PowerDAQ™ for LabVIEW

Software Manual

DSP-based High Performance Data Acquisition boards for PCI Bus
LabVIEW® driver software manual

October 2000 Edition

First Edition

October 2000 Printing

Contacting Omega Engineering

✉ Address:

OMEGA Engineering, Inc.

One Omega Drive

Stamford, Connecticut 06907-0047

U.S.A.

☎ Support:

Telephone: 1-800-622-2378

Fax: 1-800-848-4271

💻Internet Access:

Support support@omega.com

Web site http://www.omega.com

FTP site ftp://ftp.omega.com

Table of Contents

How to Use This Manual

Introduction

The National Instruments™ LabVIEW® is one of the most popular rapid development environment and graphic language (G). The PowerDAQ™ driver for the LabVIEW® was designed to support all available features of wide range of PowerDAQ™ boards from Omega Engineering, Inc.

Who Should Read This Book?

This manual has been designed to benefit the user of PowerDAQ™ boards. To use PowerDAQ™, it is assumed that you have basic PC skills, and that you are familiar with Microsoft Windows NT/2000 and/or 95/98 operating environments and National Instruments™ LabVIEW® development environment.

Organization of This Manual

This manual includes the brief information about PowerDAQ™ boards available and explains how to use low-level library and quick-start examples VI's (Visual Instruments) . Each library VI refers to one or more example, which show how to use it.

The advanced topics at the end of this manual highlight the special questions about how to create reliable data acquisition applications.

The PowerDAQ™ driver for the LabVIEW® supports the Windows 95/98, Windows NT 4.0/2000 operation systems. The LINUX version of the driver is under development and will be released soon.

The PowerDAQ™ driver for the LabVIEW® User Manual is organized as follows:

Chapter I – Introduction

This chapter provides an overview of the PowerDAQ™ driver for the LabVIEW® and information about hardware and operation systems supported.

Chapter II – Library

This chapter is a reference to PowerDAQ™ driver for the LabVIEW® low-level Library. You can find the description of each VI included to the Library and explanation how to use it in a best way.

Chapter III – Examples

This chapter is dedicated to the Quick-Start Examples. The wide set of Examples provided allows to quick start the field application development.

Chapter IV – Advanced topics

In this chapter you can find the advanced information about how to use the advantages of the PowerDAQ™ boards under the LabVIEW® and some design techniques.

Appendix A – Quick reference

Appendix A contains reference of the all files installed with the PowerDAQ™ driver for the LabVIEW® and navigates throughout them.

Appendix B –

TBD

Appendix C – Common Questions and Support

This appendix contains a list of commonly asked questions and their answers relating to usage and special features of your PowerDAQ™ PD2-AO board. Should you require assistance while installing or using PowerDAQ™ PD2-AO, support service details are also listed.

Appendix D – Warranty

This appendix contains a detailed explanation of PowerDAQ™ LabVIEW® driver warranty.

Glossary

The Glossary contains an alphabetical list and description of terms used in this manual.

Index

The Index alphabetically lists topics covered in this manual.

How to read this book

There are a two ways to use this book:

- As a Quick Reference for the Library and Example VI. In this situation the best way to find a VI which cause a problem or misunderstanding in the Index or Content of this manual and just read the associated topics. Some of the topics contain the links to the another topics, in this situation you should take a look on them also. This way is faster and could be useful for the engineers who already have an experience how to work with the PowerDAQ™ or National Instruments® data acquisition boards. And even for these users we are recommend to read Introduction to each chapter and Advanced Topics (Chapter IV).
- As a Complete Manual for PowerDAQ™ LabVIEW® driver. This is recommended way – to read this book completely from Introduction to the Appendixes an use it all the time when you learn the PowerDAQ™ boards functionality under LabVIEW® and need some help. And the best way to learn is to open example by example and Run them and use them as a start point for you own applications.

Conventions Used in This Manual

These are the main conventions used to help you get the most out of this manual:

Tips are designed to highlight quick ways to get the job done, or good ideas you might not discover on your own.

Note Notes alert you to important information.

CAUTION! Caution advises you of precautions to take to avoid injury, data loss, or system crash.

Text formatted in **bold** typeface may also represent type that should be entered verbatim or a command, as in the following example:

You can instruct users how to run setup using a command such as **setup.exe**.

Feedback

We are interested in any feedback you might have concerning our products and manuals. A Reader Evaluation form is available on the last page of the manual.

1

Introduction

Introduction

This chapter provides an overview of the PowerDAQ™ driver for the LabVIEW® and information about hardware and operation systems supported.

About the PowerDAQ™ LabVIEW® driver

Thank you for purchasing a PowerDAQ™ board. All PowerDAQ™ PD2-AO board was designed from to overcome the problems associated with previous ISA-based boards and fits to any user requirements.

The associated PowerDAQ™ software has been written specifically for these products.

Overview

The PowerDAQ™ driver for LabVIEW® is a true 32-bit software. The main idea of the driver is to provide a reliable data acquisition driver that is as close as possible to the National Instruments™ data acquisition Visual Instrument set. The wide varieties of examples (more than 50 different examples provided free with PowerDAQ™ SDK) that allow to quick start you own application development.

Driver was designed for the LabVIEW 5.0 and compatible with 5.1 version of the LabVIEW® software. All VI's are stored in 5.0 format.

The PowerDAQ™ driver includes the VI's and the driver DLL that talks directly with the board driver DLL.

PowerDAQ™ for LabVIEW® VI set is divided into two parts: the basic components and the high-level Quick-Start example VI's.

When we constructed our low-level library, we created a compatible interface with National Instruments intermediate and advanced data acquisition VI's. It is extremely easy to convert NI VI's to support the PowerDAQ boards. The PowerDAQ low-level advanced VI's support additional features that are not provided in the National Instruments VI's. High-level example VI's show the hi-speed advantages of the PowerDAQ boards and allow dramatically increase throughput during acquisition.

Features

The major features of the PowerDAQ™ LabVIEW® driver are:

- Supports LabVIEW® 5.x for Windows 95/98/NT 4.0 (Service Pack 3) - full multithread support
- Stream to Disk at full speed using any PowerDAQ™ multifunction board - we call the PowerDAQ™ stream to disk functions within LabVIEW®
- Use PowerDAQ™ and National Instruments boards in the same PC
- No custom code - this driver is written using standard LabVIEW® calls
- Includes extensive PowerDAQ™ example VI's-Analog Input, Analog Output, Digital Input, Digital Output
- Complete support for three 16-bit counter timers (8254) including conversion and frequency count VI's.
- Digital I/O and Analog Output boards support
- Thermocouple Rack VI's
- Multiple board support
- Simultaneous Analog Input, Analog Output, Digital Input, Digital Output and Counter/Timer VI

VI Naming

To not conflict with the naming of the functions already present in LabVIEW® all OMEGA ENGINEERING PowerDAQ™ driver functions have an "PD" prefix. For example Analog Input Read function is called "PD AI Read.vi".

How start you own application development

The easiest way to start you own application development is to open an existing example (or examples), save them as you own VI and start to add you own interface and functionality. Please refer to Appendix A to find a full list of Example VI and a brief description of them.

Note For the full list of VI, *see Appendix A: Quick reference.*

Hardware supported

The table bellow summarizes the hardware supported by PowerDAQ™ LabVIEW® driver. All the board models in PowerDAQ I and PowerDAQ II families are supported. Also included VI's to support PowerDAQ Thermocouple Rack (type J and K).

#	Board Model	#	Board Model
1	PD-MF-16-1M/12L	30	PD2-MFS-4-1M/12
2	PD-MF-16-1M/12H	31	PD2-MFS-8-1M/12
3	PD-MF-64-1M/12L	32	PD2-MFS-4-400/14
4	PD-MF-64-1M/12H	33	PD2-MFS-8-400/14
5	PD-MF-16-300/12L	34	PD2-MFS-4-800/14
6	PD-MF-16-300/12H	35	PD2-MFS-8-800/14
7	PD-MF-64-300/12L	36	PD2-MFS-4-300/16
8	PD-MF-64-300/12H	37	PD2-MFS-8-300/16
9	PD-MF-16-250/16L	38	PD2-DIO-64
10	PD-MF-16-250/16H	39	PD2-DIO-128
11	PD-MF-64-250/16L	40	PD2-AO-8/16
12	PD-MF-64-250/16H	41	PD2-AO-16/16
13	PD-MF-16-50/16L	42	PD2-AO-32/16
14	PD-MF-16-50/16H	43	PD-TCR-16 (with PDx-MF)
15	PD-MFS-6-1M/12		
16	PD2-MF-16-1M/12L		
17	PD2-MF-16-1M/12H		
18	PD2-MF-64-1M/12L		
19	PD2-MF-64-1M/12H		
20	PD2-MF-16-400/14L		
21	PD2-MF-16-400/14H		
22	PD2-MF-64-400/14L		
23	PD2-MF-64-400/14H		
24	PD2-MF-16-333/16L		
25	PD2-MF-16-333/16H		
26	PD2-MF-64-333/16L		
27	PD2-MF-64-333/16H		
28	PD2-MF-16-50/16L		
29	PD2-MF-16-50/16H		

PowerDAQ™ Low-level Library

PowerDAQ Low-level library is a set of board-level, analog input, analog output, digital input and output and counter-timers visual instruments most of them were designed to provide compatibility with National Instruments™ VI's. The low-level library provides a direct communication interface with PowerDAQ™ boards and is a base for all application development. The library support all advantages of multithreading environment such as Windows 95/98 or NT/2000. Each subsystem runs in its separate thread to provide maximum independence and increase speed of board operation under LabVIEW®. You can install multiple PowerDAQ™ boards into the one PC. You are limited to the number of PCI slots in your PC. For detail description about each Library VI please refer to the **Chapter II.**

Generic Examples

Generic Examples provided with PowerDAQ™ driver for the LabVIEW® includes the examples for:

- Analog Input Subsystem Examples
  - Single-Channel Acquisition
  - Multiple-Channels Acquisition
  - FFT
  - Thermocouple
  - Stream-To-Disk
  - Single-Point Acquisition
- Analog Output Examples
  - Single-Update Mode
  - Buffered Mode
  - PowerDAQ™ Analog Output Boards Support Examples
- Digital Input and Output Examples
  - Single-Update Mode

- Buffered Mode
- PowerDAQ™ Digital I/O Boards Support Examples

➢ Counter/Timers Example

- 82C54-based (MF/MFS series)
- DSP-based 24-bit counter/timers usage (AO/DIO series)
- Frequency Measurement

The goal of the each generic example is to show specific feature or features associated with specific task or subsystem in the simplest way.

For details please see **Chapter III**

Multi-boards and Multi-subsystem Examples

Thanks to PowerDAQ™ boards and software design the all subsystems available on board are ready to access simultaneously at any time without limitations. You can run high-speed analog input data acquisition process and work with counters digital I/O and analog output at the same time. To make the more complex end-customer application development easier the multi-board and multi-subsystem examples are provided.

The multi-boards support examples shows how to use two, three, four, eight and even ten PowerDAQ™ boards on the same PC.

The multi-subsystems support examples shows how to integrate the different subsystems of the PowerDAQ™ together and some advanced techniques for analog triggering.

All details are described in **Chapter III**, *Multi-subsystem* and *Multi-board* section .

2

Library

Library

This chapter is a reference to PowerDAQ™ driver for the LabVIEW® low-level Library. You can find the description of each VI included to the Library and explanation how to use it in a best way.

The Low-level library is a set of basic and more advanced VI's which can be used to develop your own data-acquisition-based application using the PowerDAQ boards.

The following sub-division of the Library VI's can be defined:

- Board-level
- Analog Input
- Analog Output
- Digital Input
- Digital Output
- Counter-Timers

Note The some VI's are dedicated to the specific board type and can not be used with another board type. In this situation the appropriate error code will be generated.

Design Notes (Before You Start)

From the LabVIEW® view the PowerDAQ™ board can be considered as set of subsystems. Generally the following subsystems can be available on a board (less or more, depends of the board type):

- Analog Input (PDx-MF(S) models only)

  Generally this subsystem has associated with it a certain number of input channels (channel list), each of which have an input range. The analog input subsystem can also have conversion and channel list clock, analog and digital trigger and input buffer to store the data from the board.

- Analog Output (PDx-MF(S) and PD2-AO models)

  Generally this subsystem has associated with it a certain number of input channels (channel list), each of which have an input range (fixed +/-10V for PDx boards). The analog input subsystem can also have programmable conversion clock, which allows channels to be updated at regular intervals.

- Digital Input and Digital Output

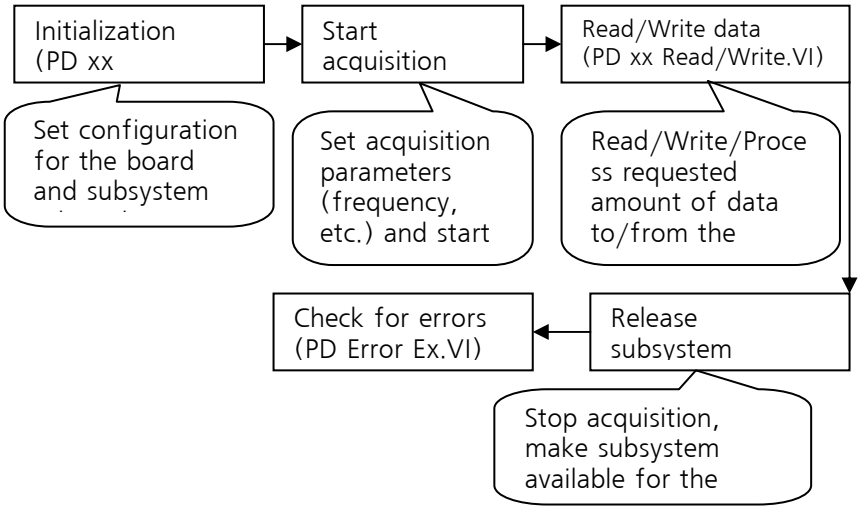
  This subsystem generally has associated with it a certain number of digital ports, 8 or 16 bits wide. Most but not all boards have ports that have dedicated input or output assigning.

- Counter/Timers

  This subsystem generally has associated with it a certain number of user-accessible on-board counter/timers, 16 or 24 bits wide. They can be programmed either as inputs (counters) – to measure frequency, pulse width/period or outputs (timers) to provide variety of output frequencies.

- Calibration (PDx-MF(S) and PD2-AO models)

Note The calibration subsystem of the PowerDAQ™ boards has no access from LabVIEW®. Use the calibration software provided with PowerDAQ™ SDK to calibrate your board. Please refer to specific board model manual or datasheet for details about subsystems available and other information.



The simplified flow of the LabVIEW®-based data acquisition application is (see picture above)

- Initialization (generally performed by *"PD xx Config.VI"* call)
- Start the acquisition process (generally performed by *"PD xx Start.VI"* call)
- Read/Write operations for the subsystem selected (generally performed by *"PD xx Read/Write.VI"* call), data processing
- Stop the Acquisition (generally performed by *"PD xx Close.VI"* call)
- Analyze the possible error codes and messages

Note That second and third items can be exchanged depends of the subsystem type (input or output). Some of intermediate level VI can incorporate all those stages inside and in this situation all acquisition process can be reduced to the single VI call (see "**PD AI Wave.VI**" for example)

Summary

PowerDAQ LabVIEW driver library is a foundation for the high-level applications. The set of VIs provided in the library allows you to build more complex VIs using the low-level ones as basic elements. The Library VIs are updated each time when we release the new version of PowerDAQ™ SDK. Normally we include one or two new low-level and intermediate-level Library VIs into the new release. Thanks to our "click and replace" technology you can easy convert you NI-boards oriented sources to the PowerDAQ boards.

Common Inputs and Outputs

Some inputs and outputs names are used by almost every Library VI and it would be better to describe them once. The common inputs and outputs and they description are listed below:

- *taskId in*

taskId Unique hex number of task associated with specified subsystem. Each VI except the "PD xx Config.vi" passes the value from the *taskId in* to the *taskId out*. Issued by "PD xx Config.vi" taskId should be used in all down-stream VIs to provide an access to the board/subsystem initialized in "PD xx Config.vi" call.

taskId out

taskId Unique hex number of task associated with specified subsystem and initially issued by "PD xx Config.vi".

error in

error in error in is a cluster of three elements

- status
- code
- source

the error state of upstream VI. The VI executes normally only if no incoming error exists, otherwise it passes the er*ror in* value to *error out*

status The status Boolean is either TRUE for an error, or FALSE for no error or a warning.

code The code input identifies the error or warning.

You can use “PD Error Ex.vi” for an explanation of non-zero error codes

source The source string describes the origin of the error or warning. Usually this string contains "No error" message or name of VI where error occurred. The “PD Error Ex.VI” explains non-zero error codes.

- *error out*

error out error out is a cluster of three elements

- status
- code
- source

Describes the error state of upstream VI or error code generated by current VI. The VI executes normally only if no incoming error exists, otherwise it passes the error in value to error out.

Status The status Boolean is either TRUE for an error, or FALSE for no error or a warning.

Code The code input identifies the error or warning.

You can use “PD Error Ex.VI” for an explanation of non-zero error codes.

source The source string describes the origin of the error or warning. Usually this string contains "No error" message or name of VI where error occurred. Use "PD Error Ex.vi" for an explanation of non-zero error codes.

Board-Level and Service VIs

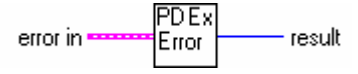
The following list of Library Visual Instruments below can be defined as board level and service VI.

VI name	Description
PD Error Ex.vi	Convert error code to string message
PD Get Capabilities.vi	Get board capabilities information
PD Shell.vi	Call shell function
PD String Array to Int.vi	Convert arrays

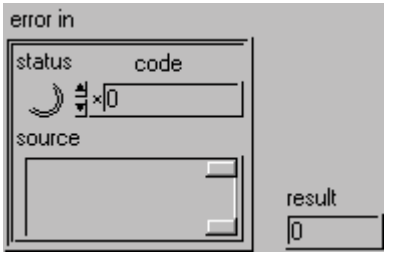
Pd Error Ex.VI

Explains non-zero error codes and shows dialog box with information about error. In case of zero error code do nothing.

Connector Panel



Front Panel



Controls and Indicators

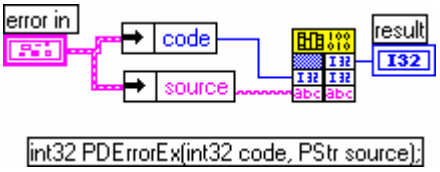
error in

see **Common Inputs and Outputs** topic for the description

Result always return 0x0

(reserved for future use)

Block Diagram

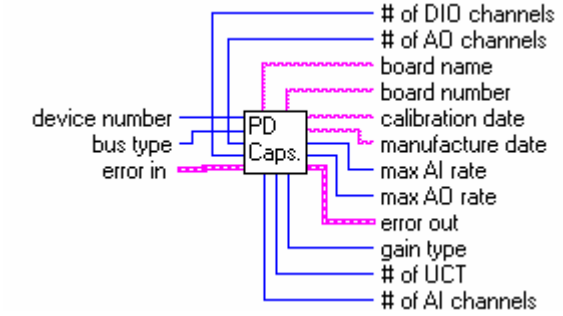


PD Get Capabilities.vi

Board detection VI.

Collects information about specified board at specified bus, return non- zero error code if board is not found. See "PD Diagnostic.VI" or "PD AI Single channel realtime display.VI" for examples.

Connector Panel



Controls and Indicators

device number Number of device at bus specified (starts from 1)

bus type Bus type where PowerDAQ device installed

error in

see **Common Inputs and Outputs** topic for the description

board name Type of PowerDAQ board installed

board number Board serial # (from EPROM)

calibration date Board calibration date (from EPROM)

manufacture date Board manufacture date (from EPROM)

# of AI channels Number of analog input channels

# of AO channels Number of analog output channels

# of DIO channels Number of digital input-output channels

# of UCT Number of user counter-timers

gain type Indicates type of board gains

3 means only gain 1 available

2 means 1,2,5,10 gains available

1 means 1,10,100,1000 gains available

0 means 1,2,4,8 gains supported by board

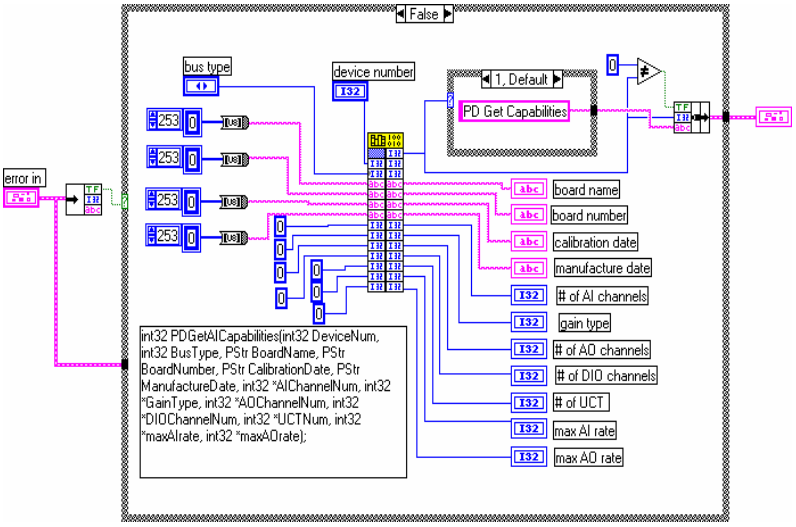
max AI rate Maximum value of analog input rate that recommended by PowerDAQ technical documentation

max AO rate Maximum value of analog output rate that recommended by PowerDAQ technical documentation

error out

see **Common Inputs and Outputs** topic for the description

Block Diagram



See also

- PD Get AO Capabilities.VI
- PD Get AI Capabilities.VI

PD Shell.VI

PowerDAQ service VI.

Execute *ShellExecuteA* function for open default Web-browser with OMEGA ENGINEERING web-site hot link. See *Windows API* documentation for details about the *ShellExecuteA* function.

Connector Panel

Controls and Indicators

OMEGA ENGINEERING on-line hot link button. When ON(TRUE) – this VI calls *ShellExecuteA* function

command command line to be processed by *open* command

HInstance Result of *ShellExecuteA* function call. Void.

See also

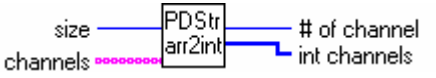
- All PowerDAQ examples with hot-link button.

PD String Array to Int.vi

This PowerDAQ library VI reserved for internal usage only.

Converts string array to array of integers with specified size. Used in "PD xx Config.VI" to convert LabVIEW channel list in string array to the PowerDAQ-compatible integer array channel list.

Connector Panel



Controls and Indicators

Channels channels - string array: specifies the set of analog input channels for a group and task. PowerDAQ limitations: only digits should be used. See "**PD AI Config.VI**" for details.

channel channels - string array: specifies the set of analog input channels for a group and task. PowerDAQ limitations: only digits should be used. See "**PD AI Config.VI**" for details.

Size size of output array

int channels output channel list in the internal presentation

output channel list in the internal presentation

# of channels output – number of channels used in channel list

Analog Input

The Analog Input part of the PowerDAQ™ MF(S) series boards is a most advanced and complicated subsystem. The different type of clocks, triggers, buffer settings associated with Analog Input subsystem are described in topics below.

For information about clock and buffer settings please refer to:

- Conversion Clock and Channel List Clock
- Optimizing performance using the buffer settings

The Analog Input VIs in the PowerDAQ™ Library include:

- Configuration Functions
  - PD AI Config.VI
  - PD AI Fine Tune.VI
  - PD AI Start.VI
  -

PD AI Stream Init.VI

- PD AI Clear.VI
- PD Get AI Capabilities.VI

- Service Functions
  - PD AI Active channels.VI
  - PD AI Convert Ranges.VI
  - PD AI Data Count.VI
  - PD AI Init Arrays.VI
  - PD AI Frame Size.VI

  - PD AI Thermocouple Control.VI

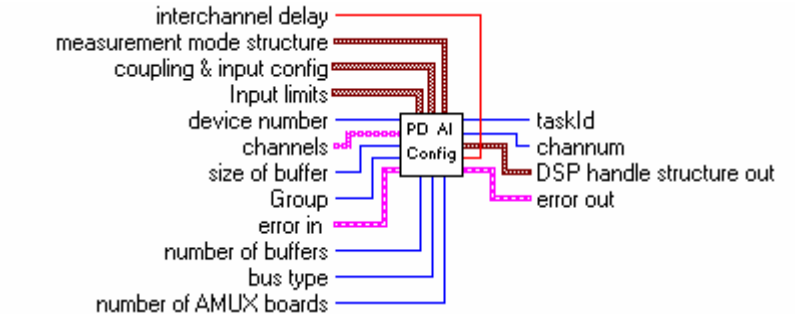
  - PD AI Thermocouple Display.VI
- Analog Input Functions

  - PD AI Read Async.VI
  - PD AI Read.VI
  - PD AI Stream.VI
- Simplified Easy Functions
  - PD AI Read One Scan.vi
  - PD AI Sample Channel.vi
  - PD AI Single Scan.vi
  - PD AI Wave.vi

PD AI Config.VI

Basically this VI is used to configure the upper and lower input limits (and calculate gain for each channel and input range for whole board automatically), set the channel list, acquisition buffer size and inter-channel delay for the board, specified in the *device number* parameter. This VI checks the board availability and if specified board available and Analog Input subsystem exists on the board and not used by another VI the unique *taskId* will be issued. It should be used in all sub-sequential VI to provide access to the same subsystem of the same board. Use "PD AI Clear.VI" to release the Analog Input subsystem.

Connector Panel



Controls and Indicators

I16 **device number** Number of the device(beginning from 1) at the bus specified

bus type Bus type where PowerDAQ™ device installed

SGL **interchannel delay** delay between acquiring each individual channel in the channel list in seconds ,0 - use default value

I32 **size of buffer** Set size of buffer for storing the data from the board – 10000 by default. Buffer allocated and released inside the PowerDAQ driver. The buffer size is defined in bytes. To convert bytes into samples, the bytes should be divided by two (one sample is a one 16-bit word which is equal to the two bytes). The frame size in LabVIEW is defined in scans. One scan is a set of the samples – one for each channel in the channel list. This was done in this way because the most AI xx functions accept the number of scans as an input parameter, for

the amount of the data requested. To find out the frame size the following equation should be used:

FrameSize = (((BufferSize / 2) / (Number Of The Frames))/Number Of The Acquired Channels)

The buffer size (in bytes) should be defined using the rule: at least 1.5 times the Acquisition Rate, for rates between 10K and 200K. And, 2-4 times the Acquisition Rate for high frequencies. Increasing the buffer size increases the stability and reliability of the acquisition system at high acquisition speeds.

Channels ([string]) channels: specifies the set of analog input channels for a group and task.

Example : if only first row of the array used -

'0,1,2,4' defines four channels in the channel list,

'0' – define empty channel list, '1' define one channel in the channel list (channel 0),

'0,' define one channel in the channel list (channel 0),

'1,' define one channel in the channel list (channel 1).

Channel input – string array channels: specifies the set of analog input channels for a group and task. PowerDAQ limitations : only digital numbers of channels, ',',';' should be used in channel list definition as a channel delimiters.

If more then one row used, they are OR-ed together and following rule is used – '0' in row n means that channel n is not present in the channel list, '1' means that channel n present in the channel list, string separated by commas will be processed using the rule described above for the first row.

Input limits input limits is an array of clusters. You can use it for tune board gain and input type. We use this input for capability with National Instruments™ "G" sources.

Specify low and high level of input signals (in Volts) or leave empty array, which means the input limits keep their default settings.

The default value is +/-10V for all PD(2)-MF(S) boards.

You can set gains per channel directly in "PD AI Fine Tune.VI"

input - cluster of input limits: an array of clusters, of which each array element specifies the range limits for the channel(s) in the corresponding element of the

channels array. If there are fewer elements in this array than the number of channels, the VI uses the default values for the rest of the channels. Each cluster contains the following parameters:

high limit – single : specifies the maximum scaled data in Volts (10.0 - default)

low limit – single : specifies the minimum scaled data in Volts (-10.0 - default)

High limit (10.0) high limit - single : specifies the maximum scaled data in Volts (10.0 – default)

Low limit (-10.0) low limit - single : specifies the minimum scaled data in Volts (-10.0 - default)

coupling & input config coupling & input config is an array of clusters. PowerDAQ™ LabVIEW® driver use only type of inputs (differential/single-ended) and, if both of them specified differential have high priority and used to set board configuration

input – cluster coupling & input config:

an array of clusters, of which each array element specifies the coupling and input configuration for the channel(s) in the corresponding element of the channels array.

PowerDAQ™ LabVIEW® driver use only type of inputs (differential/single-ended) and, if both of them specified differential have high priority and will be used to set board configuration

The default input is an empty array, which means the parameters do not change from their default settings. Each cluster contains the following parameters:

coupling:

This input is not used by PowerDAQ™ boards and ignored (both AC and DC modes are supported and boards have a perfect specs in full

bandwidth)
input config:
0: Do not change the input config setting.
1: Differential.
2: Not used (reserved)
3: Non-referenced single-ended

Coupling (no change:0) coupling.

This input is not used by PowerDAQ™ boards and ignored (both AC and DC modes are supported and boards have a perfect specs in full bandwidth)

Input config (no change:0) input config.

0: Do not change the input config setting.
1: Differential.
2: Not used (reserved)
3: Non-referenced single-ended

This value used for all analog input channels, but if you specify it for more than one channel the differential mode have a highest priority

error in

see **Common Inputs and Outputs** topic for the description

number of buffers This input is not used by PowerDAQ™ boards and ignored

Group This input is not used with PowerDAQ boards because only one upper level VI can use Analog Input subsystem of board at time (see *PowerDAQ™ API Programmer Guide*)

number of AMUX boards This input is not used by PowerDAQ™ boards and ignored

measurement mode structure This input is not used by PowerDAQ™ boards and ignored

measurement mode This input is not used by PowerDAQ™ boards and ignored

reserved

Allocation mode (no change:0) PowerDAQ driver allocates the memory buffer automatically but if 999 specified as input value for this input buffer will not be allocated. It is useful for single scan operations.

taskId Unique hex number of task associated with specified subsystem. Each VI except the "PD xx Config.vi" passes the value from the *taskId in* to the *taskId out*. Issued by "PD xx Config.vi" taskId should be used in all down-stream VIs to provide an access to the board/subsystem initialized in "PD xx Config.vi" call.

channum Total number of channels in channel list. Use "PD AI Active Channels.VI" to receive this number via *taskId* at any time after "PD AI Config.VI" has been called.

error out

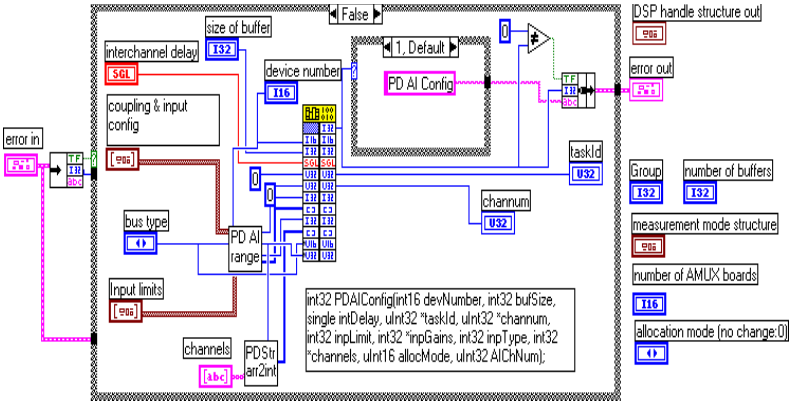
see **Common Inputs and Outputs** topic for the description

DSP handle structure out This input is not used by PowerDAQ™ boards and ignored

size This input is not used by PowerDAQ™ boards and ignored

DSP memory handle This input is not used by PowerDAQ™ boards and ignored

Block Diagram



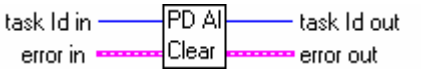
See also

- PD AI Clear.VI
- PD AI Start.VI
- Analog Input Examples

PD AI Clear.VI

This VI stops any analog input acquisition process, frees resources, clear buffers and returns zero *taskId*. After you have called this VI you can reuse analog input subsystem for another task in LabVIEW.

Connector Panel



Controls and Indicators

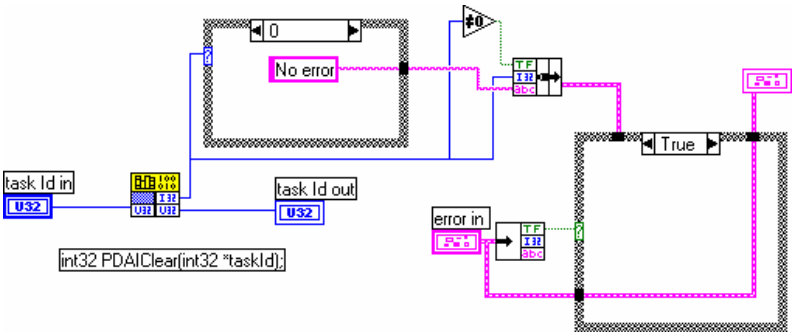
error in
see **Common Inputs and Outputs** topic for the description

task Id in Unique hex number of task to be cleared
see **Common Inputs and Outputs** topic for the description

task Id out output - always 0

error out
see **Common Inputs and Outputs** topic for the description

Block Diagram



See also

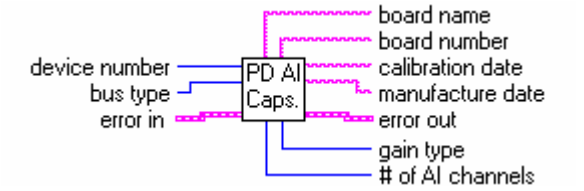
- PD AI Config.VI

- PD AI Start.VI
- Analog Input Examples

PD Get AI Capabilities.VI

Collects information about analog input subsystem of specified board at specified bus , return non zero error code (from PD Get Capabilities) if board is not found

Connector Panel



Controls and Indicators

device number Number of device at bus specified (beginning from 1)

bus type Bus type where PowerDAQ device installed

error in

see **Common Inputs and Outputs** topic for the description

board name Type of PowerDAQ board installed

board number Board serial # (from EEPROM)

calibration date Board calibration date (from EEPROM)

manufacture date Board manufacture date (from EEPROM)

# of AI channels Number of analog input channels

max AI rate Maximum value of analog input rate that recommended by PowerDAQ technical documentation

gain type Indicates type of board gains

3 means only gain 1 available

2 means 1,2,5,10 gains available

1 means 1,10,100,1000 gains available

0 means 1,2,4,8 gains supported by board

error out

see **Common Inputs and Outputs** topic for the description

See also

- "

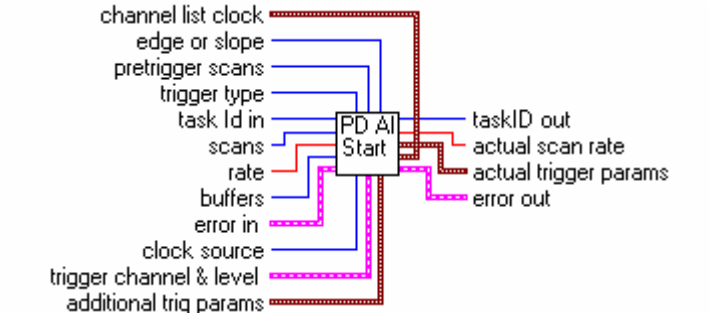
PD AO Config.VI"

- "PD Get Capabilities.VI"

PD AI Start.VI

This VI configures the rate, channel list and conversion clock source and triggering conditions (digital or analog), sets total number of scans to acquire or continuos mode and starts analog input subsystem of PowerDAQ board which specified via taskId.

Connector Panel



Controls and Indicators

error in

see Common Inputs and Outputs topic for the description

task Id in

see Common Inputs and Outputs topic for the description

scans Total number of scans to be acquired. After the board acquire requested amount of scans acquisition will be stopped automatically. Specify zero value or leave unconnected for continuous acquisition.

rate Frequency of A/D conversion clock (CV Clock). If you acquiring more than 1 channel actual frequency for each channel should be calculated as

(Scan rate)/(Number of active channels)

Note There are two different type of clocks available on PowerDAQ board to clock acquisition – channel list clock and A/D conversion clock. Please refer to *Conversion Clock and Channel List Clock* section in Advanced Topics for details.

buffers now ignored by PowerDAQ

trigger type trigger type

0 – no triggering (default input).

1 – analog trigger (default setting) - one of the analog channels in channel list used for triggering.

2 – digital trigger A - for PowerDAQ boards - digital trigger on rising edge.

3 – digital trigger A and B - for PowerDAQ boards - digital trigger on falling edge.

4 – external high-speed hardware trigger (see usage notes below).

Note For trigger type 1 one analog channel that exists in channel list could be specified in trigger channel control (0 - default channel). For trigger type 2 and 3 any numbers of digital channels could be specified.

For the trigger type 4 the following values in the first entry in the channel list could be used:

0 – do not use external trigger line to start acquisition.

1 – use rising edge of the external trigger line to start acquisition.

2 – use falling edge of the external trigger line to start acquisition.

3 – reserved

For the trigger type 4 the follows values in the second entry in the channel list could be used:

0 – do not use external trigger line to stop acquisition.

1 – use rising edge of the external trigger line to stop acquisition.

2 – use falling edge of the external trigger line to stop acquisition.

3 – reserved

pretrigger scans The pre-trigger data size, specified in scans. The pretrigger data size is limited by Frame Size. See "PD AI Frame

Size.VI" for details.

edge or slope edge or slope.

0 - Do not change the default setting (default input).

1 - rising.

2 - falling.

trigger channel & level trigger channel(s) and level

Analog level and analog or digital channel(s) for triggering should be specified via this cluster.

trigger channel Specify trigger channel number(s)

For trigger type 1 analog channel existing in channel list could be specified in trigger channel control (0 - default channel). For trigger type 2 and 3 any numbers of digital channels could be specified.

level

Level (measured in Volts) which analog source must cross for a trigger to occur. You must also specify whether level must be crossed on a leading or trailing slope with the edge or slope input. The default input for level is 0.0.

additional trig params advanced trigger parameters for best triggering

Hysteresis –

The hysteresis of the signal in Volts. The default input and setting are 0.0. The hysteresis value refers to a limit above or below the actual trigger level, which will need to be surpassed before it is considered to be a valid trigger. This compensates for the possibility of a noise spike causing an accidental trigger condition to be detected

coupling coupling of signal

0: Do not change the trigger coupling setting (default input).

1: DC.

2: AC (reserved)

delay pre-triggering delay in seconds. The default input and setting are 0.0 seconds.

skip count skip count is the number of triggers the VI skips before triggering the acquisition.

-1 - no change the skip count setting

0 - default - no skip any triggers.

time limit time limit set the amount of time for PowerDAQ LabVIEW driver waits for the trigger to occur.

-1.0 -no change the time limit setting

0 - (default) no waiting.

The "time limit" control sets the windowing size to be analyzed in the data stream before control is returned to the other sections of the VI. If a trigger is not detected during this window, the trigger "Timeout" situation will flash and control will return to the other subsystems before returning to analyze another window of data. The minimum length of the trigger signal is the number of channels divided by the Scan rate. (If you don't acquire the trigger, you can't trigger on it.)

clock source A/D clock source (CV Clock Source)

0 : software

1 : internal(default)

2 : external raising

3 : external falling

4 : continuos

Note: For an external clock you should provide expected scan rate (5000..MaximumFrequency] or specify timeout when PD AI Read called

channel list clock There are two main clocking signals for analog input of PowerDAQ boards - channel list start clock (CL Clock) and A/D (CV Clock) acquisition clock. Acquisition clock controls each acquisition and channel list start clock controls the start of each scan (channel list) acquiring... Please refer to *Conversion Clock and Channel List Clock* section in Advanced Topics for details.

clock source channel list (CL) clock source

0 : software

1 : internal(default)

2 : external raising

3 : external falling

4 : continuos

Note: For an external clock you should provide expected scan rate (5000..MaximumFrequency] or specify timeout when PD AI Read called

rate Channel list clock rate. By default channel list start continuously but if other specified this value used for channel list frequency, calculations are following : rate <= (board rate) / (number of channels). Using high A/D frequency and low channel list frequency you can receive a "Virtual Sample and Hold effect".

error out

see Common Inputs and Outputs topic for the description

taskID out

see Common Inputs and Outputs topic for the description

actual scan rate Actual acquisition rate, CV clock based (returned from driver). If you acquiring more than 1 channel actual frequency for each channel could be calculated as

(Scan rate)/(Number of active channels)

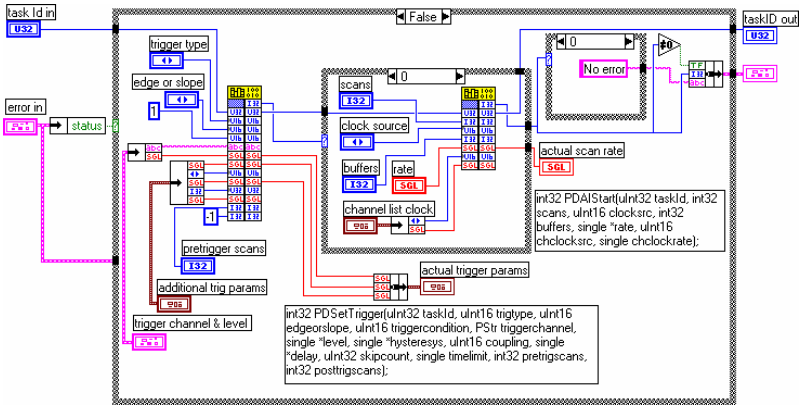
actual trigger params actual trigger params may differ slightly from the requested trigger inputs, depending on the hardware capabilities.

level actual level of the analog trigger used.

hysteresis actual hysteresis is the hysteresis the VI used.

delay actual delay is the delay the VI used.

Block Diagram



See also

- PD AI Clear.VI
- PD AI Config.VI
- Analog Input Examples

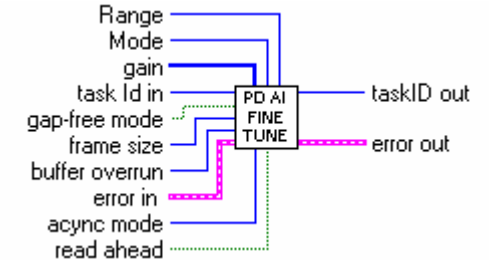
PD AI Fine Tune.vi

The "PD AI Fine Tune.VI" is used to set advanced PowerDAQ board analog input settings that are not available via "PD AI Config.VI" because of compatibility. The following parameters can be changed using this VI :

- Buffer/FIFO overrun mode
- Analog Input Range and Mode (Single-Ended/Differential)
- Gains (per channel)

This VI should be called after "PD AI Config.VI" but before "PD AI [Async] Read.VI" or "PD AI Stream.VI".

Connector Panel



Controls and Indicators

U32 task Id in

see Common Inputs and Outputs topic for the description

error in

see Common Inputs and Outputs topic for the description

TF gap-free mode

This parameter is a Boolean and dedicated to the on-board A/D FIFO buffer. When set to ON (TRUE) any on-board FIFO overrun will cause an error and stop acquisition. If this parameter set to OFF (FALSE) the on-board FIFO overrun situation will restart acquisition without any error messages.

Set this parameter to OFF for most tasks except the gap-free critical stream to disk applications.

U32 frame size Set user-defined number of scans in frame.

(0 - use default Frame Size)

In general the PowerDAQ™ buffer is a driver-allocated space in the host PC memory which is used to store the acquired data. The buffer is divided by some number of logical segments (16 by default). The application will be notified that new data is available only after the next frame is done. The buffer size is defined in bytes. To convert bytes into samples, the bytes should be divided by two (one sample is a one 16-bit word which is equal to the two bytes). The frame size in LabVIEW is defined in scans. One scan is a set of the samples – one for each channel in the channel list. This was done in this way because the most "PD AI xx" functions accept the number of scans as an input parameter, for the amount of the data requested. To find out the frame size the following equation

should be used:

FrameSize = (((BufferSize / 2) / (Number Of The Frames))/Number Of The Acquired Channels)

Note: This parameter should be changed only after "PD AI Config.VI" call and before "PD AI Start.VI" call.

The number of frames in the buffer is sixteen by default, the current Frame Size can be determined using the "PD AI Frame Size.VI"

The frame size should not be less than One-Half of the on-board FIFO size (in scans). This is not a required but recommended because of the internals of transferring mechanism.

The Number Of Frames in a buffer should not be less than 4, when the high response is a requirement the frame size should be minimized and amount of frames in the buffer increased.

buffer overrun Buffer overrun mode

Integer parameter with four possible values. This parameter is dedicated to PowerDAQ LabVIEW driver buffer in host PC memory space and can be used depends of the application requirements.

0 - the buffer overrun is not allowed any buffer overrun will cause an error and stops acquisition.

1 - the buffer overrun is allowed, but driver use buffer in recycle mode, when newest data can override the old one even if this data is not read yet. No errors will be generated in this situation.

2 - the buffer overrun is allowed, any buffer overrun will restart acquisition.

3 - buffer will be acquired only once and acquisition will be stopped.

acync mode Operation mode (synchronous/asynchronous)

0 - do not change (default)

1 - asynchronous operation ("PD AI Read Async.VI" should be used)

2 - synchronous operation ("PD AI Read.VI" should be used)

When synchronous mode used each “PD AI Read[Stream].VI” call will wait until board will collect required amount of the data or user-specified timeout will be expired and only after that control will be returned to the LabVIEW®. This means that VI execution will frees for the time necessary to collect the data.

If asynchronous mode used the control will be returned to the LabVIEW immediately from “PD AI Read Async.VI” call regardless of data availability. If there are not enough data available the specific error code will be returned.

[I32] gain input – array that set gain level for each AI channel.

[I32] gain input - [i32] array used to set desired gains for the analog inputs directly and override the settings made in the "PD AI Config.VI". Each array item represents one channel in the channel list. There are following options available to select from:

-1 (default input) do not change the gain

0 set the gain 1

1 set the gain 2

2 set the gain 4

3 set the gain 8

4 set the gain 10 for the boards with gains 1/10/100/1000

5 set the gain 100

6 set the gain 1000

7 set the gain 5

8 set the gain 10 for the boards with gains 1/2/5/10

Note The gains are available only in sets of 1/2/4/8, 1/10/100/1000, 1/2/5/10 or just 1, depends of the board installed, to determine the gain type use "PD Get AI Capabilities.VI". For this particular input it is a user responsibility to use the correct gain code.

[TF] read ahead This parameter is a Boolean and responsible for behavior of how the “PD AI xx Read[Stream].VI” retrieving the data from the PowerDAQ driver buffer. When set to ON (TRUE) “PD AI Read[Stream].VI” will check how many data is inside the buffer not read yet. (To see this inside the LabVIEW the “PD AI Data Count.VI” provided. If more than S of the acquisition buffer is already contains the data each call of “PD AI Read[Stream]. VI” will try to retrieve two frames at time

instead of one. If it OFF (FALSE) only one frame a time will be requested from the PowerDAQ driver.

Mode Analog Input mode

Allow to set analog input mode directly and override the settings made in the "PD AI Config.VI".

0 - do not change

1- single-ended

2 – differential

Range Analog Input Range

Allow to set analog input mode directly and override the settings made in the "PD AI Config.VI".

0 - do not change

1 - 0..5 V

2 - +/-5 V

3 - 0..10 V

4 - +/-10 V

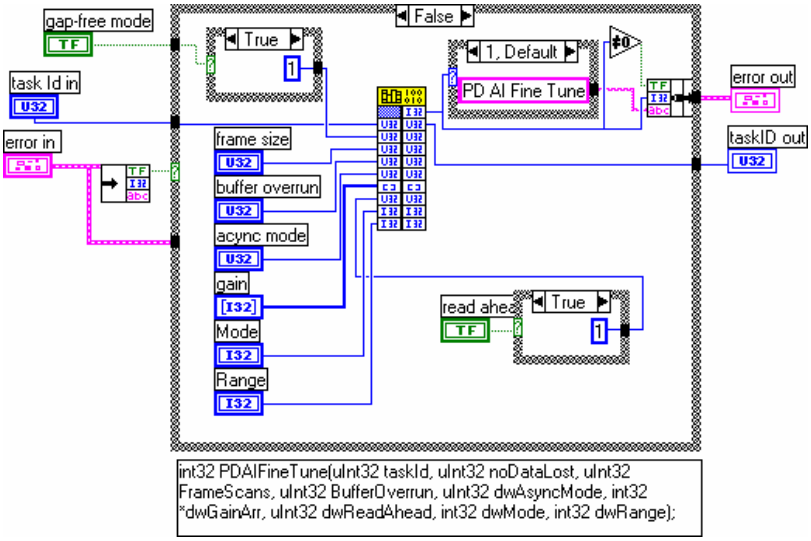
taskID out

see Common Inputs and Outputs topic for the description

error out

see Common Inputs and Outputs topic for the description

Block Diagram



See also

- PD AI Clear.VI
- PD AI Config.VI
- PD AI Start.VI
- Optimizing performance using the buffer settings
- Analog Input Examples

PD AI Stream Init.vi

Prepares analog input subsystem of the PowerDAQ board for the stream-to-disk operation. Used with "PD AI Stream.VI"

Connector Panel

Controls and Indicators

U32 task Id in

see Common Inputs and Outputs topic for the description

error in

see Common Inputs and Outputs topic for the description

abc file name File name for accept acquiring data. Catalog must be exist.

U32 frames to stream Total number of the frames. Specify zero for continuos streaming or any number.

TF No data lost mode Allows to select one of two acquisition modes:

- OFF: with small possibility of data loss but non-stop acquisition.
- ON: set PowerDAQ LabVIEW driver to "no data lost" mode.

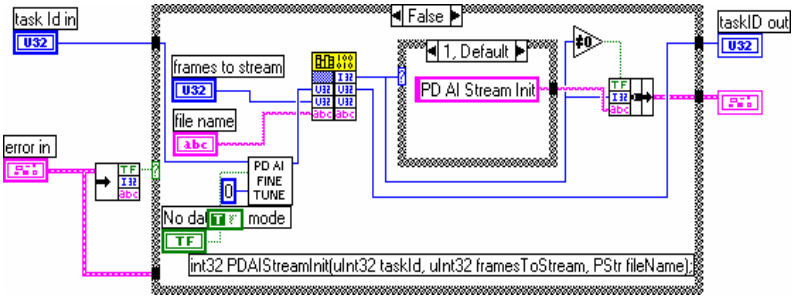
U32 task Id out

see Common Inputs and Outputs topic for the description

error out

see Common Inputs and Outputs topic for the description

Block Diagram



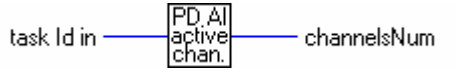
See also

- PD AI Stream.VI
- PD AI Stream To Disk.VI (Example)
- Optimizing performance using the buffer settings
- Multi-board streaming Examples

PD AI Active channels.vi

PowerDAQ LabVIEW advanced VI - returns the number of active channels in the channel list for specified taskId. In case of zero or invalid taskId returns 0.

Connector Panel



Controls and Indicators

U32 task Id in

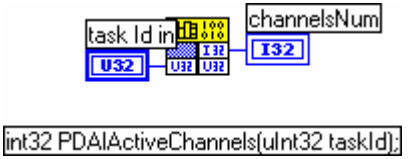
see Common Inputs and Outputs topic for the description

I32 channelsNum output - total number of analog input channels used for specified taskId.

This is an advanced VI and be careful with passed parameter -

if any error detected, it returns zero as a result.

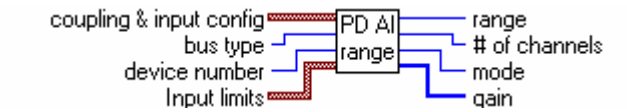
Block Diagram



PD AI Convert Ranges.vi

This VI converts input limits/ranges for analog input channels that are specified in the standard LabVIEW format into PowerDAQ compatible values. This VI is dedicated to the internal usage and should not be used in user applications.

Connector Panel



Controls and Indicators

I16 device number
see PD AI Config.VI topic for the description

bus type
see PD AI Config.VI topic for the description

Input limits
see PD AI Config.VI topic for the description

coupling & input config
see PD AI Config.VI topic for the description

I32 range analog input range code formatted to the internal PowerDAQ presentation

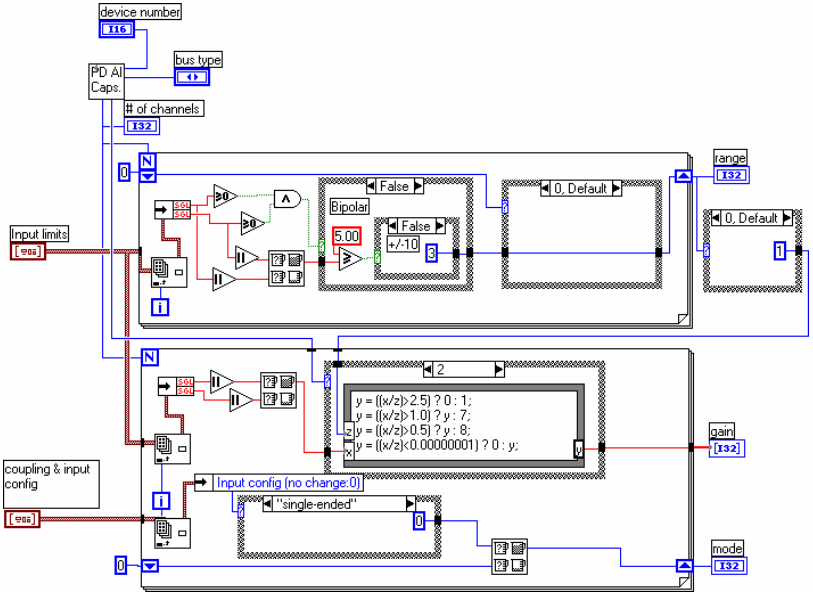
I32 mode analog input mode code formatted to the internal

PowerDAQ presentation

[I32] gain output - array that set gain level for each AI channel.

I32 # of channels output - total number of analog input channels on specified PowerDAQ board

Block Diagram



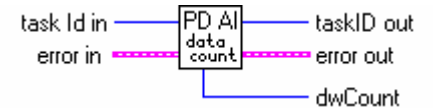
PD AI Data Count.VI

PD AI Data Count is an Advanced Library VI that can be used to determine the stability of an acquisition system.

To keep the acquisition system healthy the maximum value of the result of this function call should be less then S of the acquisition buffer size measured in samples (one sample is equal to two bytes).

Note Another parameter which could be helpful for performance monitoring tasks is the Kernel time usage (see your WIN32 API documentation).

Connector Panel



Controls and Indicators

U32 task Id in
see Common Inputs and Outputs topic for the description

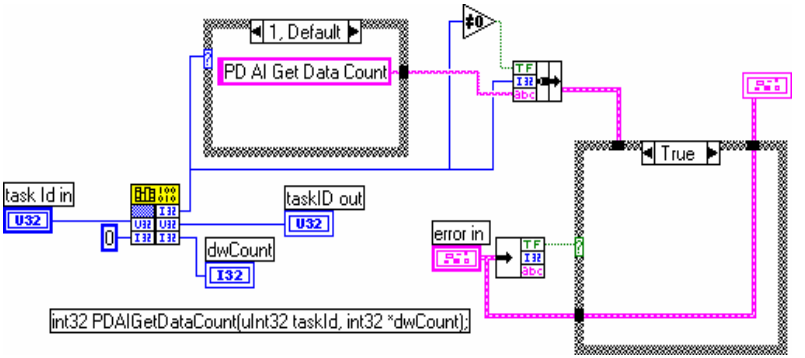
error in
see Common Inputs and Outputs topic for the description

U32 taskID out
see Common Inputs and Outputs topic for the description

error out
see Common Inputs and Outputs topic for the description

I32 dwCount uint32 - amount of data in the acquisition buffer (in samples) that has not been read into LabVIEW. This value is a different than *backlog* parameter from "PD AI Read.VI" call. See "PD AI Read.VI" fo details.

Block Diagram



See also

- PD AI Acquire Eight Channels.VI (Example)
- Optimizing performance using the buffer settings

PD AI Init Arrays.vi

Used to pre-allocate LabVIEW arrays for "PD AI Async Read.VI" call. This VI is dedicated to the internal usage and should not be used in user applications.

PD AI Frame Size.VI

Returns the current analog input buffer frame size measured in scans. This or even to this number of scans requested could be passed into "PD AI Read.VI" to acquire the data without any loss. In case of zero or invalid *taskId* returns 0.

See **Input buffers. General Information** topic for the detailed description about PowerDAQ buffering mechanism.

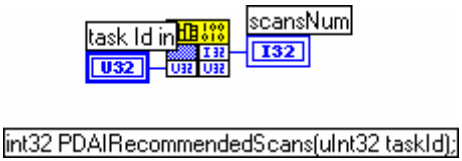
Connector Panel

Controls and Indicators

U32 **task Id in**
see **Common Inputs and Outputs** topic for the description

I32 **scansNum** output - recommended number of data (in scans).

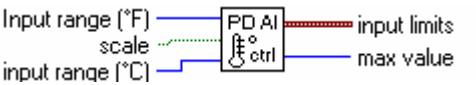
Block Diagram



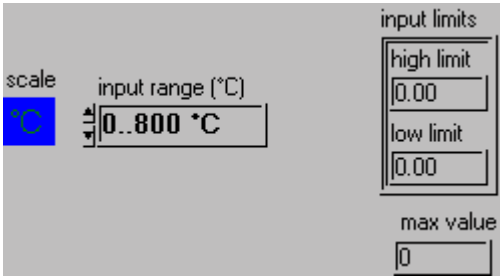
PD AI Thermocouple Control.VI

This VI used to convert visual controls (temperature scale and input range for selected scale) to cluster of input limits that should be connected to the Build Array LabVIEW VI which should be connected to the "**PD AI Config.VI**". See "PD AI 16 Channels Thermocouple Rack.VI" for example. For temperature data use "PD AI Thermocouple Display.VI".

Connector Panel



Front Panel



Controls and Indicators

scale Measurement mode switch : °F /°C

input range (°C) Choosing the correct input range increases precision of temperature measurement

Input range (°F) Choosing the correct input range increases precision of temperature measurement

input limits – output.

This cluster calculated depends of the input range selected and board type installed.

Input limits is a cluster of two elements

High limit (Volts)

Low limit (Volts.)

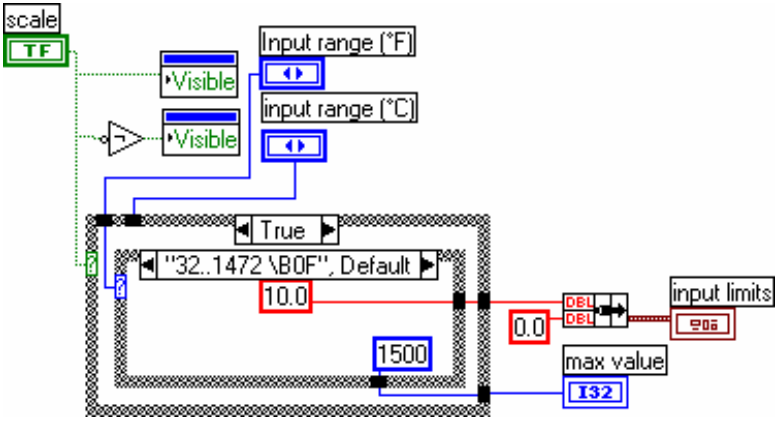
Low limit always is zero and high limit is calculated inside the VI.

high limit high limit calculated depends of the input temperature range selected

low limit low limit is always zero

max value output - maximum value of temperature in scale selected. Should be connected to *max* value input of "PD AI Thermocouple Display.VI."

Block Diagram



See also

- "PD AI 16 Channels Thermocouple Rack.VI" (Example)

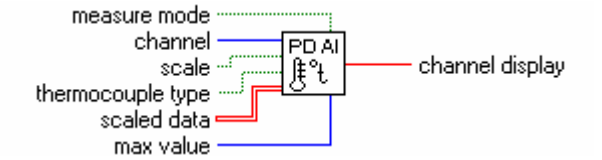
"

- PD AI Thermocouple Display.VI"

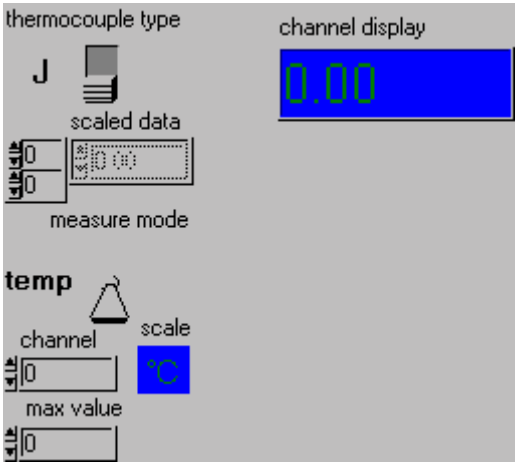
PD AI Thermocouple Display.VI

This VI displays data (temperature or Volts) on digital indicator. For temperature it converts voltage into the temperature for scale (F/C) and thermocouple type (J or K) selected. **–9999.00** means 'data out of range'. See "PD AI 16 channels thermocouple rack.VI" for example. For tune channel which thermocouple connected use "PD AI Thermocouple Control.VI"

Connector Panel



Front Panel



Controls and Indicators

SGL **scaled data** Scaled data (in Volts) from the PowerDAQ board driver

SGL Scaled data (in Volts) from the PowerDAQ board driver

TF **measure mode** Temperature/voltage display mode switch

I32 **channel** Channel number for display

TF **thermocouple type** Thermocouple type switch. There is two type of thermocouple available with PowerDAQ board - J and K.

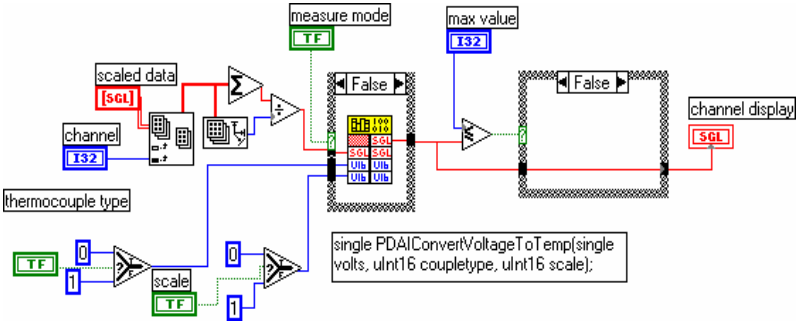
TF **scale** Measurement mode switch : °F /°C

I32 **max value** Input - maximum value of temperature in scale selected. If overloaded passes 9999.00 to output

SGL **channel display** Displays the data from selected channel in

selected form

Block Diagram



See also

- "PD AI 16 Channels Thermocouple Rack.VI" (Example)

"

- PD AI Thermocouple Control.VI"

PD AI Read Async.VI

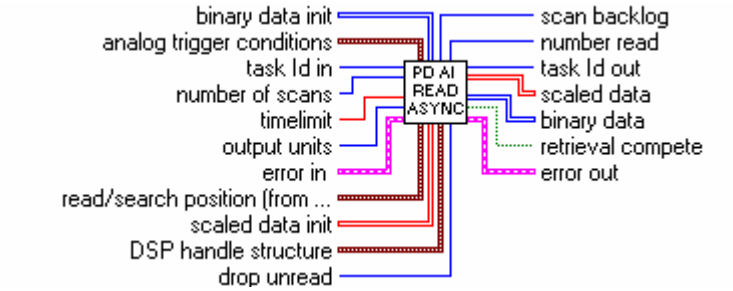
"PD AI Async Read.VI" incorporates all the functionality of "PD AI Read.VI" and provides an asynchronous feature that allows work with multiple (more than two) boars. Reads the specified number of scans and returns the data: 'as is' in 16-bits words array and 'scaled' - in output units in 4-bytes 'single' floating point numbers array. If trigger condition specified - waits for trigger or timeout.

For speed-up operation there are two pre-allocated in "**PD AI Init Arrays.VI**" arrays has to be passed to the VI call. The size of both arrays should be equal or greater than number of scans requested.

Special *drop unread* input allows set data read mode in accordance with current task.

Note For asynchronous operation *timelimit* should be zero and if analog trigger enabled timeout should be specified **in "PD AI Start VI"**.

Connector Panel



Controls and Indicators

U32 task Id in

see Common Inputs and Outputs topic for the description

I32 number of scans input – the number of scans requested from the driver. Use "PD AI Frame Size.VI" to find out the amount of scans to be requested from the "PD AI Read Async.VI" without any data loss. See Optimizing performance using the buffer settings topic for details.

SGL timelimit input – maximum timeout for data wait from driver.

0.0 – asynchronous operation

-1.0 – timelimit value will be calculated automatically by driver.

error in

see Common Inputs and Outputs topic for the description

analog trigger conditions input cluster - used for set analog trigger conditions. See cluster members description for details. Analog trigger settings allow synchronizing the data using one of the analog input channels (first channel in the channel list by default) as trigger.


mode input — trigger mode :

off - clear all triggers

on - add analog trigger conditions

no change - leave trigger configuration unchanged


channel index Specify trigger channel number

For analog trigger input channel existing in channel list could be specified in *trigger channel* control (0 - default channel).


slope edge or slope.

0 - do not change the default setting (default input).

1 - rising.

2 - falling.


level level (measured in Volts) which analog source must cross for a trigger to occur. You must also specify whether level must be crossed on a leading or trailing slope with the edge or slope input. The default input for level is 0.0.


hysteresis The hysteresis of the signal in Volts. The default input and setting are 0.0. The hysteresis value refers to a limit above or below the actual trigger level, which will need to be surpassed before it is considered to be a valid trigger. This compensates for the possibility of a noise spike causing an accidental trigger condition to be detected


skip count skip count is the number of triggers the VI skips before triggering the acquisition.

-1 — do no change the skip count setting

0 - default — do not skip any triggers.


offset offset in scans after trigger conditions retrieved - post trigger scan number


DSP handle structure This input is not used with PowerDAQ boards and ignored


output units This input is not used with PowerDAQ boards and ignored


read/search position (from mark) This input is not used with PowerDAQ boards and ignored


drop unread Set acquisition buffer control mode:

0 - gap-free mode(used in stream to disk applications);

1 - drop the data when buffer overflow detected;

2 - drop the data up to the end of current frame in current

read;

4 - combines 1 and 2;

binary data init pre-allocated array (use “PD AI Init Arrays.VI” to create an array)

pre-allocated array (use “PD AI Init Arrays.VI” to create an array)

scaled data init pre-allocated array (use “PD AI Init Arrays.VI” to create an array)

pre-allocated array (use “PD AI Init Arrays.VI” to create an array)

task Id out

see Common Inputs and Outputs topic for the description

scaled data scaled data is a 2D array that contains analog input data in scaled data units. The data appears in columns, where each column contains the data for a single channel. The second (bottom) dimension selects channel column. The first (top) dimension selects a single data point for that channel.

scaled data: a two-dimensional array that contains scaled analog input data if output units requests scaled data. The first dimension is scans, the second dimension is channels.

binary data binary data is a 2D array that contains binary analog input data if output units requests binary data.

The data appears in columns where each column contains the data for a single channel. The second (or bottom) dimension selects channel column. The first (or top) dimension selects a single data point for that channel.

binary data: a two-dimensional array that contains binary analog input data if output units requests binary data. The first dimension is scans, the second dimension is channels.

number read output - number of scans that was read from PowerDAQ board

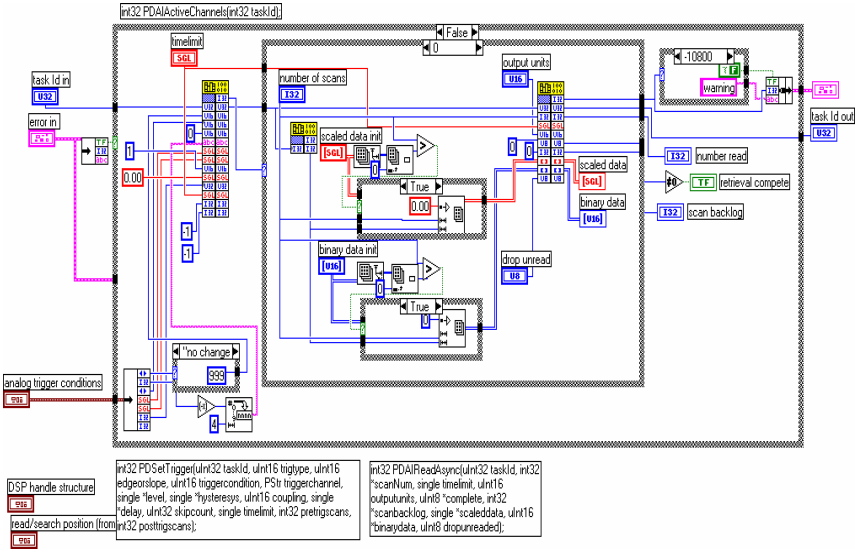
scan backlog output - number of scans of lost data. This output works differently in than with NI boards. The value returned is an amount of data in scans, which was unread from the last frame requested from the board. To determine the amount of unread data in acquisition buffer use “PD AI Data Count.VI”

error out

see Common Inputs and Outputs topic for the description

retrieval compete TRUE if all requested data was read from driver

Block Diagram



See also

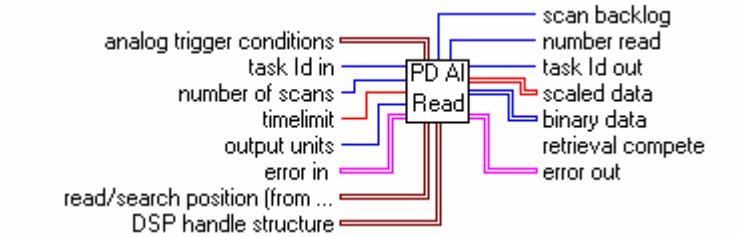
- "PD AI Config.VI"
- "PD AI Start.VI"
- "PD AI Read.VI"
- "PD AI Async.VI (Single-channel example)
- "PD AI Async Acquire Eight Channels.VI (Multiple-channels example)
- "PD AI Two Boards Realtime Display.VI (Multiple boards example)

PD AI Read.vi

"PD AI Read.VI" is a main function that allows receiving the data from the PowerDAQ board continuously. "PD AI Read.VI" reads the specified number of scans and returns the data: 'as is' in 16-bits words array and 'scaled' - in output units in 4-bytes 'single' floating point numbers array. If trigger condition specified - check for trigger or timeout.

Note That board should be properly initialized using "PD AI Config.VI" and acquisition started in "PD AI Start VI". Additional acquisition setting can be provided in "PD AI Fine Tune.VI"

Connector Panel



Controls and Indicators

U32 task Id in

see Common Inputs and Outputs topic for the description

I32 number of scans input - the number of scans requested from the driver. Use "PD AI Frame Size.VI" to find out the amount of scans to be requested from the "PD AI Read Async.VI" without any data loss. See Optimizing the performance using the buffer settings topic for details.

SGL timelimit input - maximum timeout for data wait from driver.

0.0 - asynchronous operation

-1.0 - timelimit value will be calculated automatically by driver.

error in

see Common Inputs and Outputs topic for the description

analog trigger conditions input cluster - used for set analog trigger conditions. See cluster members description for details. Analog trigger settings allow synchronizing the data using one of the analog input channels (first channel in the channel list by default) as trigger.

mode input - trigger mode :

off - clear all triggers

on - add analog trigger conditions

no change - leave trigger configuration unchanged

channel index Specify trigger channel number

For analog trigger input channel existing in channel list could be specified in trigger channel control (0 - default channel).

slope edge or slope.

0 - do not change the default setting (default input).

1 - rising.

2 - falling.

level level (measured in Volts) which analog source must cross for a trigger to occur. You must also specify whether level must be crossed on a leading or trailing slope with the edge or slope input. The default input for level is 0.0.

hysteresis The hysteresis of the signal in Volts. The default input and setting are 0.0. The hysteresis value refers to a limit above or below the actual trigger level, which will need to be surpassed before it is considered to be a valid trigger. This compensates for the possibility of a noise spike causing an accidental trigger condition to be detected

skip count skip count is the number of triggers the VI skips before triggering the acquisition.

-1 - do no change the skip count setting

0 - default - do not skip any triggers.

offset offset in scans after trigger conditions retrived - post trigger scan number

DSP handle structure This input is not used with PowerDAQ boards and ignored

size size is not used.

DSP memory handle This input is not used with PowerDAQ boards and ignored

output units

read/search position (from mark) This input is not used with PowerDAQ boards and ignored

position This input is not used with PowerDAQ boards and ignored

read offset This input is not used with PowerDAQ boards and ignored

task Id out

see Common Inputs and Outputs topic for the description

scaled data scaled data is a 2D array that contains analog input data in scaled data units. The data appears in columns, where each column contains the data for a single channel. The second (bottom) dimension selects which channel column. The first (top) dimension selects a single data point for that channel.

([[sgl]]) scaled data: a two-dimensional array that contains scaled analog input data if output units requests scaled data. The first dimension is scans, the second dimension is channels.

binary data binary data is a 2D array that contains unscaled analog input data if output units requests binary data.

The data appears in columns where each column contains the data for a single channel. The second (or bottom) dimension

selects which channel column. The first (or top) dimension selects a single data point for that channel.

U16 ([[i16]]) binary data: a two-dimensional array that contains unscaled analog input data if output units requests binary data. The first dimension is scans, the second dimension is channels.

I32 number read output - number of scans that really readed from PowerDAQ board

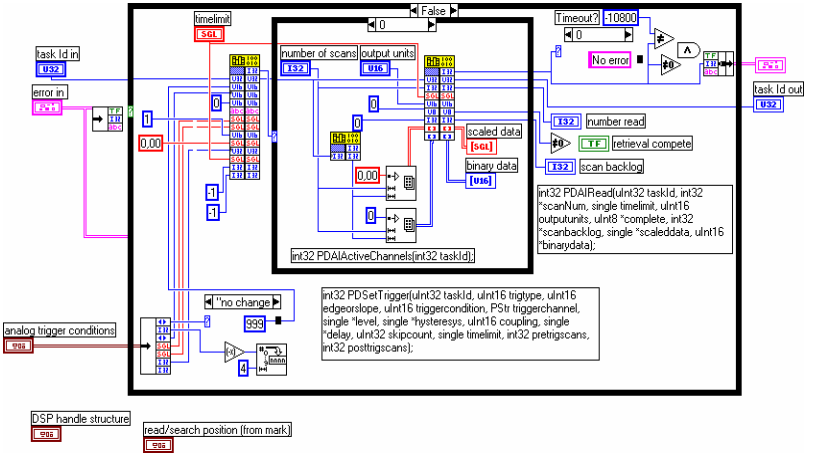
I32 scan backlog output - number of scans of losed data

error out

see Common Inputs and Outputs topic for the description

TF retrieval compete TRUE if all requested data readed from driver

Block Diagram



See also

- "PD AI Config.VI"
- "PD AI Start.VI"

- "PD AI Read Async.VI"
- "PD AI Single Channel Realtime Display.VI (Single-channel example)
- "PD AI Acquire Eight Channels.VI (Multiple-channels example)
- Multiple boards example
- "PD AI All Subsystems with Analog trigger.VI"

PD AI Stream.vi

This VI used to flush acquisition buffer to the disk and check stream-to-disk operation status which was started by "

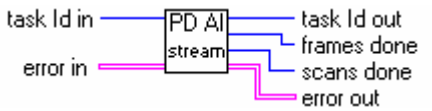
PD AI Stream Init.VI". After specified in "

PD AI Stream Init.VI" numbers of frames to be stored on disk complete, it stops acquisition and closes the file. Stream-to-disk operation will be continued until disk will be full or acquisition stopped because of performance considerations if frame number to be acquired is not specified or set to zero.

Note That data is never comes into the LabVIEW in a stream-to-disk operation. The acquisition buffer flushed to the disk inside the PowerDAQ LabVIEW driver to speed up stream-to-disk operation. See **Optimizing performance using the buffer settings** topic for details about how to acquire data without any loss. Use **"PD AI Start VI"/"PD AI Read[Async].VI**" to retrieve data into the LabVIEW. Additional acquisition setting can be provided in **"PD AI Fine Tune.VI**"

Note The one of the biggest concern for engineer who developing the data acquisition-based application is "Am I loosing any critical data?". Please refer to *Advanced topics* section for complete information. Briefly PowerDAQ board provides a choice to the user how to set up analog input subsystem depends of application requirement. Set *Buffer overrun* to 0 and *Gap Free Mode* to ON(TRUE) using the "**PD AI Fine Tune.VI**". Check interrupts distribution on your PC. For best performance PowerDAQ board should not share the same interrupt with mass-storage device(s) or video card.

Connector Panel



Controls and Indicators

task Id in

see Common Inputs and Outputs topic for the description

error in

see Common Inputs and Outputs topic for the description

task Id out

see Common Inputs and Outputs topic for the description

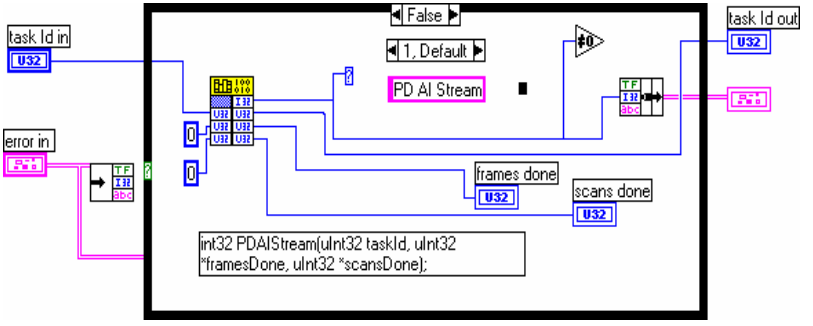
frames done Number of frames of data that already stored on disk. Frame size can be found using “PD AI Frame Size.VI”.

scans done Number of scans received from PowerDAQ LabVIEW driver and stored on disk. Each scan is equal to two bytes.

error out

see Common Inputs and Outputs topic for the description

Block Diagram



See also

- “PD AI Config.VI”
- “

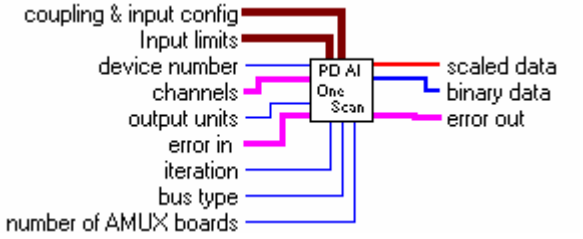
PD AI Stream Init.VI"

- "PD AI Stream To Disk.VI" (Single board example)
- Multiple boards stream-to-disk examples

PD AI Read One Scan.vi

"PD AI Read One Scan.VI" reads one scan of data from specified channel list and returns the data : 'as is' in 16-bits words array and 'scaled' - in output units in 4-bytes 'single' floating point numbers array. Iteration input maybe used to avoid continuos initialization of the board. This VI does not use buffering mechanism and interrupts/events to retrieve the data from PowerDAQ board. For low acquisition rates such as few Hertz or even milli-Hertz when acquisition clocked using the PC clock this VI is a best way to receive an analog input data from board.

Connector Panel



Controls and Indicators

output units output units specifies whether the VI returns binary data or scaled data.

0: use default settings for this input

1: return scaled data only. The binary data output array is filled by zeroes.

2: return binary data only. The scaled data output array is filled by zeroes.

3: return scaled and binary data(default).

error in

see **Common Inputs and Outputs** topic for the description

iteration input - iteration: controls when "PD AI Read One Scan.VI" performs initialization. Zero means that the VI configures the acquisition with the hardware specifications connected to the VI, then starts the acquisition and reads the data. If iteration is greater than zero, then "PD AI Read One Scan.VI" reads the data without initialization.

bus type Bus type where PowerDAQ(tm) device installed

channels ([string]) channels: specifies the set of analog input channels for a group and task.

Example : if only first row of the array used -

'0,1,2,4' defines four channels in the channel list,

'0' - define empty channel list, '1' define one channel in the channel list (channel 0),

'0,' define one channel in the channel list (channel 0),

'1,' define one channel in the channel list (channel 0).

channel input - string array channels: specifies the set of analog input channels for a group and task. PowerDAQ limitations : only digital numbers of channels, ',',';' should be used in channel list definition as a channel delimiters.

If more then one row used, they are OR-ed together and following rule is used - '0' in row n means that channel n is not present in the channel list, '1' means that channel n present in the channel list, string separated by commas will be processed using the rule described above for the first row.

number of AMUX boards This input is not used with PowerDAQ boards and ignored

Input limits input limits is an array of clusters. You can use it for tune board gain and input type. We use this input for capability with National Instruments™ "G" sources.

Specify low and high level of input signals (in Volts) or leave empty array, which means the input limits keep their default settings.

The default value is +/-10V for all PD(2)-MF(S) boards.

input - cluster of input limits: an array of clusters, of which each array element specifies the range limits for the channel(s) in the corresponding element of the channels array. If there are fewer elements in this array than the number of channels, the VI uses the default values for the rest of the channels. Each cluster contains the following parameters:

high limit - single : specifies the maximum scaled data in Volts (10.0 - default)

low limit - single : specifies the minimum scaled data in Volts (-10.0 - default)

high limit - single : specifies the maximum scaled data in Volts (10.0 - default)

low limit (–10.0) low limit - single : specifies the minimum scaled data in Volts (-10.0 - default)

coupling & input config coupling & input config is an array of clusters. PowerDAQ LabVIEW driver use only type of inputs (differential/single-ended) and, if both of them specified differential have high priority and used for board configuration

input – cluster coupling & input config:

an array of clusters, of which each array element specifies the coupling and input configuration for the channel(s) in the corresponding element of the channels array.

PowerDAQ™ LabVIEW® driver use only type of inputs (differential/single-ended) and, if both of them specified differential have high priority and will be used to set board configuration

The default input is an empty array, which means the parameters do not change from their default settings. Each cluster contains the following parameters:

coupling:

This input is not used by PowerDAQ™ boards

and ignored (both AC and DC modes are supported and boards have a perfect specs in full bandwidth)

input config:

0: Do not change the input config setting.

1: Differential.

2: Not used (reserved)

3: Non-referenced single-ended

Coupling (no change:0) This input is not used by PowerDAQ™ boards and ignored (both AC and DC modes are supported and boards have a perfect specs in full bandwidth)

Input config (no change:0) input config.

0: Do not change the input config setting.

1: Differential.

2: Not used (reserved)

3: Non-referenced single-ended

This value used for all analog input channels, but if you specify it for more than one channel the differential mode have a highest priority

device number Number of device at bus specified (beginning from 1)

binary data binary data is an array that contains unscaled analog input data if output units requests binary data.

binary data: an array that contains unscaled analog input data if output units requests binary data.

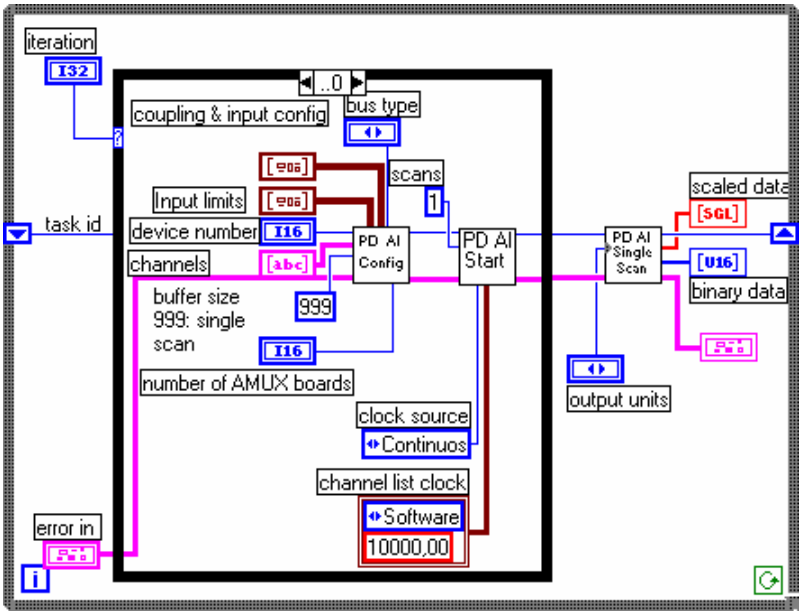
scaled data scaled data is an array that contains analog input data in scaled data units.

scaled data: an array that contains scaled analog input data if output units requests scaled data.

error out

see **Common Inputs and Outputs** topic for the description

Block Diagram



See also

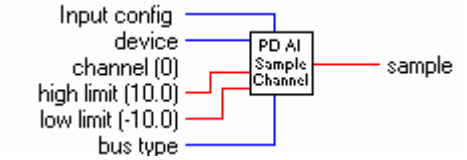
- "PD AI Config.VI" (Single channel read)
- "PD AI Single Scan.VI" (Single scan simplified VI)
- "PD AI Single Scan Example.VI" (Example VI)

PD AI Sample Channel.vi

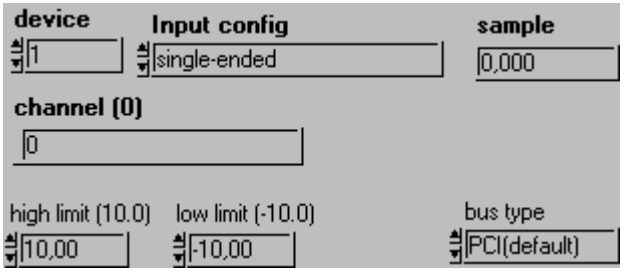
This VI simplifies non-timed single channel measurement. Measures the signal attached to the specified channel and returns the scaled measured data (in Volts).

A dialog box appears if an error occurs giving you the option to stop the VI or continue.

Connector Panel



Front Panel



Controls and Indicators

channel (0) Channel to be measured.

input - string channel: specifies the analog input channel to be measured.

PowerDAQ limitations : only digital numbers of channels, ',',';' should be used in channel list definition as a channel delimiters.

Example :

'0' - define empty channel list,

'1' define one channel in the channel list (channel 0),

'0,' define one channel in the channel list (channel 0),

'1,' define one channel in the channel list (channel 1).

low limit (–10.0) low limit - single : specifies the minimum scaled data in Volts (-10.0 - default)

high limit(10.0) - high limit - single : specifies the maximum scaled data in Volts (10.0 - default)

device device is the device number you assigned to the plug-in DAQ board during configuration. This parameter defaults to 1.

Input config input config.

0: Do not change the input config setting.

1: Differential.

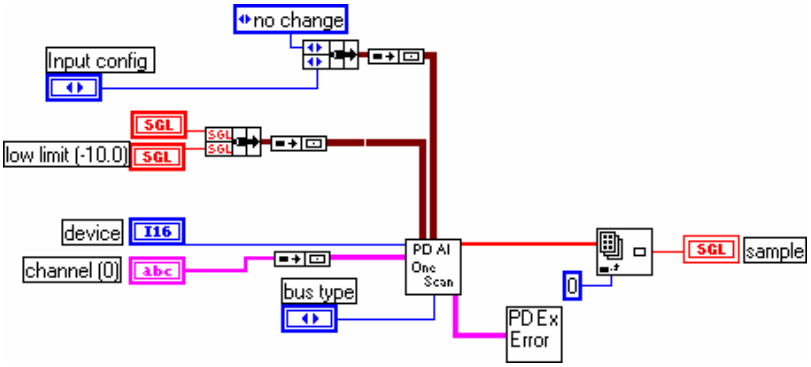
2: Not used (reserved)

3: Non-referenced single-ended

bus type Bus type where PowerDAQ device installed

sample sample: single value than contains scaled analog input data.

Block Diagram



See also

- "PD AI Config.VI"

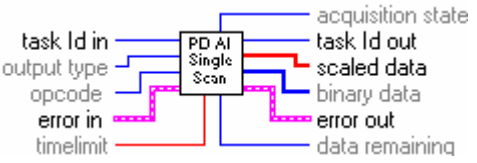
- "PD AI Single Scan.VI" (Single scan simplified VI)
- "PD AI Single Scan Example.VI" (Example VI)

PD AI Single Scan.vi

This VI returns one scan of data from the specified channels.

Current version of this VI used the 50000 Hz clock rate for acquisition.

Connector Panel



Controls and Indicators

U32 task Id in

see Common Inputs and Outputs topic for the description

U16 output type output type specifies whether the VI returns binary data or scaled data.

0: use default settings for this input

1: return scaled data only. The binary data output array is filled by zeroes.

2: return binary data only. The scaled data output array is filled by zeroes.

3: return scaled and binary data (default).

opcode Ignored by PowerDAQ boards because PowerDAQ LabVIEW driver performs only one scan of all channels in channel list and returns the data read.

opcode: specifies the type of data retrieval the VI performs.

0: do not change the opcode setting (default input)
1: read oldest data (default setting)
2: read newest data
3: return data remaining only
4: empty the FIFO only

timelimit input - timeout for wait data from driver. If not specified calculated automatically by driver.


error in

see Common Inputs and Outputs topic for the description

task Id out

see Common Inputs and Outputs topic for the description

binary data binary data is an array that contains binary analog input data if output units requests binary data.

binary data: an array that contains binary analog input data if output units requests binary data.

scaled data scaled data is an array that contains analog input data in scaled data units.

scaled data: an array that contains scaled analog input data if output units requests scaled data.

data remaining data remaining

1 when data still in the FIFO or buffer when the VI is about to return.
0 when the FIFO is empty.

Always 0 for PowerDAQ boards

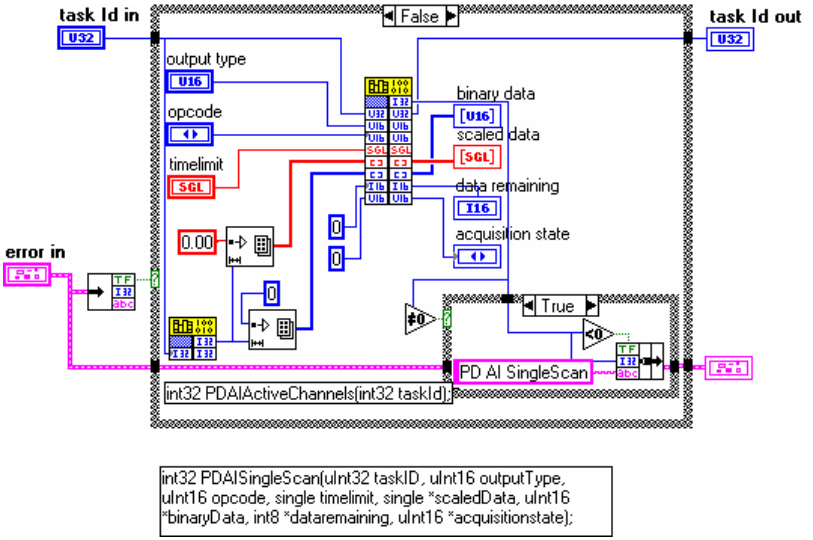

acquisition state acquisition state:

0: running
1: finished with backlog
2: finished with no backlog
3: paused
4: no acquisition

if function call finished successfully PowerDAQ LabVIEW driver return 2 and 4 in case of error

error out
see Common Inputs and Outputs topic for the description

Block Diagram



See also

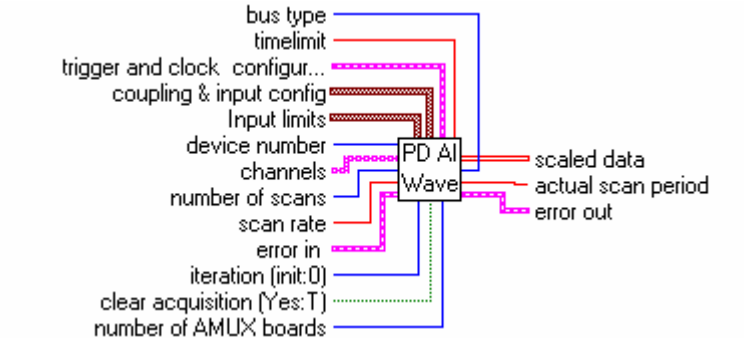
- "PD AI Config.VI"
- "PD AI Read One Scan.VI"
- "PD AI Single Scan Example.VI" (Example VI)

PD AI Wave.VI

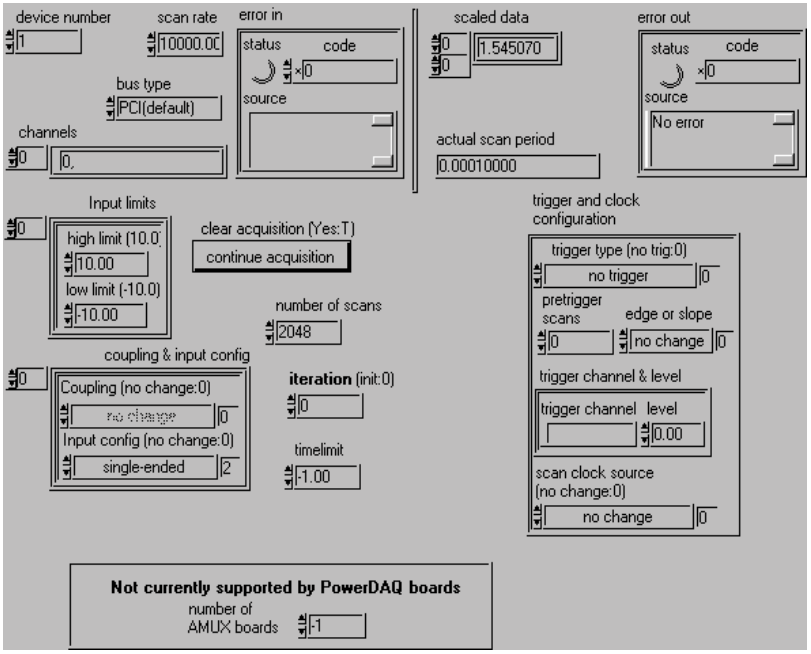
This VI is a most advanced low-level analog input VI from PowerDAQ library, which allows initializing, starting, acquiring data and stopping buffered acquisition using the single VI call.

"PD AI Wave.VI" performs multi-channel analog input operations from specified PowerDAQ board. Channel list, scan rate, input limits and types can be changed. Buffer size calculated automatically using the amount of data requested. The buffer used in "acquire once and stop" mode (see **Optimizing performance using the buffer settings** section of this manual for details).

Connector Panel



Front Panel



Controls and Indicators

I16 **device number** Number of the device (beginning from 1) at the bus specified

[abc] **channels** ([string]) channels: specifies the set of analog input channels for a group and task.

Example : if only first row of the array used -

'0,1,2,4' defines four channels in the channel list,

'0' - define empty channel list, '1' define one channel in the channel list (channel 0),

'0,' define one channel in the channel list (channel 0),

'1,' define one channel in the channel list (channel 0).

abc **channel** input - string array channels: specifies the set of analog input channels for a group and task. PowerDAQ limitations : only digital numbers of channels, ',',';' should be used in channel list definition as a channel delimiters.

If more then one row used, they are OR-ed together and following rule is used - '0' in row n means

that channel n is not present in the channel list, '1' means that channel n present in the channel list, string separated by commas will be processed using the rule described above for the first row.

Input limits input limits is an array of clusters. You can use it for tune board gain and input type. We use this input for capability with National Instruments™ "G" sources.

Specify low and high level of input signals (in Volts) or leave empty array, which means the input limits keep their default settings.

The default value is +/-10V for all PD(2)-MF(S) boards.

input - cluster of input limits: an array of clusters, of which each array element specifies the range limits for the channel(s) in the corresponding element of the channels array. If there are fewer elements in this array than the number of channels, the VI uses the default values for the rest of the channels. Each cluster contains the following parameters:

high limit - single : specifies the maximum scaled data in Volts (10.0 - default)

low limit - single : specifies the minimum scaled data in Volts (-10.0 - default)

high limit (10.0) - single : specifies the maximum scaled data in Volts (10.0 - default)

low limit (-10.0) low limit - single : specifies the minimum scaled data in Volts (-10.0 - default)

coupling & input config coupling & input config is an array of clusters. PowerDAQ LabVIEW driver use only type of inputs (differential/single-ended) and, if both of them specified differential have upper priority and used for board configuration

input - cluster coupling & input config:

an array of clusters, of which each array element specifies the coupling and input configuration for the channel(s) in the corresponding element of the channels array.

PowerDAQ™ LabVIEW® driver use only type of inputs (differential/single-ended) and, if both of them specified differential have high priority and will be used to set board configuration

The default input is an empty array, which means the

parameters do not change from their default settings. Each cluster contains the following parameters:

coupling:

This input is not used by PowerDAQ™ boards and ignored (both AC and DC modes are supported and boards have a perfect specs in full bandwidth)

input config:

0: Do not change the input config setting.

1: Differential.

2: Not used (reserved)

3: Non-referenced single-ended

Coupling (no change:0) This input is not used by PowerDAQ™ boards and ignored (both AC and DC modes are supported and boards have a perfect specs in full bandwidth)

Input config (no change:0) input config.

0: Do not change the input config setting.

1: Differential.

2: Not used (reserved)

3: Non-referenced single-ended

This value used for all analog input channels, but if you specify it for more than one channel the differential mode have a highest priority

clear acquisition (Yes:T) clear acquisition determines whether the VI clears the task after reading the specified number of scans. The VI should pass a TRUE value to this parameter when reading the last set of scans for a given acquisition. The default is TRUE, which means that the VI reads data only once if you do this input remains unwired. Usually you wire this input to the terminating condition of a loop, so that when the loop finishes, the VI clears the acquisition.

number of scans input - the number of scans requested from the driver. Use "PD AI Frame Size.VI" to find out the amount of scans to be requested from the "PD AI Read Async.VI" without any data loss. See Optimizing the performance using the buffer settings topic for details.

iteration (init:0) input - iteration: controls when PD AI Wave performs initialization. Zero means that the VI configures the

acquisition with the hardware specifications connected to the VI, then starts the acquisition and reads the data. If iteration is greater than zero, then PD AI Wave read the data.

timelimit input - maximum timeout for data wait from driver.

0.0 - asynchronous operation

-1.0 - timelimit value will be calculated automatically by driver.

scan rate Frequency of A/D conversion clock (CV Clock). If you acquiring more than 1 channel actual frequency for each channel should be calculated as

(Scan rate)/(Number of active channels)

Note There are two different type of clocks available on PowerDAQ board to clock acquisition - channel list clock and A/D conversion clock. Please refer to Conversion Clock and Channel List Clock section in Advanced Topics of PowerDAQ Manual for details.

error in

see **Common Inputs and Outputs** topic for the description

number of AMUX boards This input is not used with PowerDAQ boards and ignored

trigger and clock configuration trigger and clock cluster contains parameters controlling the acquisition clock and triggering conditions. Because of compatibility with standard LabVIEW library only few of PowerDAQ features provided

trigger type (no trig:0) trigger type

0 - no triggering (default input).

1 - analog trigger (default setting) - one of the analog channels in channel list used for triggering.

2 - digital trigger A - for PowerDAQ boards - digital trigger on rising edge.

3 - digital trigger A and B - for PowerDAQ boards - digital trigger on falling edge.

4 - external high-speed hardware trigger (see usage notes below).

Note : For trigger type 1 one analog channel that exists in channel list could be specified in trigger channel control (0 - default channel). For trigger type 2 and 3 any numbers of digital channels could be specified.

For the trigger type 4 the following values in the first

entry in the channel list could be used:

0 - do not use external trigger line to start acquisition.

1 - use rising edge of the external trigger line to start acquisition.

2 - use falling edge of the external trigger line to start acquisition.

3 - reserved

For the trigger type 4 the follows values in the second entry in the channel list could be used:

0 - do not use external trigger line to stop acquisition.

1 - use rising edge of the external trigger line to stop acquisition.

2 - use falling edge of the external trigger line to stop acquisition.

3 - reserved

pretrigger scans The pre-trigger data size, specified in scans. The pretrigger data size is limited by Frame Size. See "PD AI Frame Size.VI" for details.

edge or slope edge or slope.

0 - Do not change the default setting (default input).

1 - rising.

2 - falling.

trigger channel & level Analog level and analog or digital channel(s) for triggering should be specified via this cluster.

trigger channel Specify trigger channel number(s)

For trigger type 1 analog channel existing in channel list could be specified in trigger channel control (0 - default channel). For trigger type 2 and 3 any numbers of digital channels could be specified.

level Level (measured in Volts) which analog source must cross for a trigger to occur. You must also specify whether level must be crossed on a leading or trailing slope with the edge or slope input. The default input for level is 0.0.

scan clock source (no change:0) clock source

0 : no change

1 : internal(default)

2 : external

3 : continuos

Note : For external clock you should provide scan rate (5000..MaximumFrequency] or specify timeout when PD AI Wave calling

bus type Bus type where PowerDAQ(tm) device installed

scaled data scaled data is a 2D array that contains analog input data in scaled data units. The data appears in columns, where each column contains the data for a single channel. The second (bottom) dimension selects which channel column. The first (top) dimension selects a single data point for that channel.

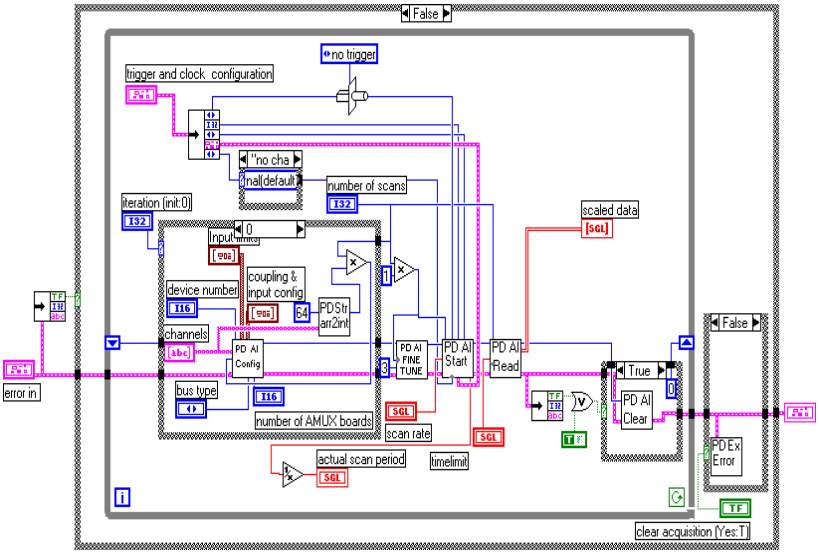
scaled data: a two-dimensional array that contains scaled analog input data if output units requests scaled data. The first dimension is scans, the second dimension is channels.

actual scan period Calculated as 1/(actual scan rate). Actual scan rate parameter may be different than scan rate whether of board capabilities.

error out

see **Common Inputs and Outputs** topic for the

Block Diagram



See also

- "PD AI Config.VI"
- "PD AI Start.VI"
- "PD AI Read.VI"
- "PD AI 16 Channels Thermocouple Rack.VI" (Example VI)
- "PD AI Wave Spectrum Analyzer.VI" (Example VI)

Analog Output

Generally this subsystem has associated with it a certain number of input channels (channel list), each of which have an input range (fixed +/-10V for PDx boards). The analog input subsystem can also have programmable conversion clock, which allows channels to be updated at regular intervals.

PowerDAQ II series boards include multifunction boards with analog output subsystem on it and analog output boards itself. Both of them are supported by PowerDAQ LabVIEW driver and only difference between them in data representation.

The Analog Output VIs in the PowerDAQ™ Library include:

- Configuration Functions
  -

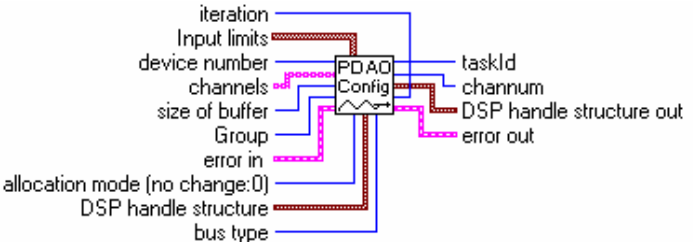
PD AO Config.vi
- PD AO Start.vi
- PD AO Clear.vi
- PD AO Clock Config.vi

- Analog Output Functions
  - PD AO Write.vi
  - PD AO Buffered Wave.vi
  - PD AO Wave.vi
- Simplified Easy Functions
  - PD AO Write One Update.vi
  - PD AO Update Channel.vi

PD AO Config.VI

Basically this VI is used to configure the upper and lower input limits (which are reserved for now, +/- 10V fixed output range used), set the channel list and acquisition buffer size for the board, specified in the *device number* parameter. This VI checks the board availability and if specified board available and Analog Output subsystem exists on the board and not used by another VI the unique *taskId* will be issued. It should be used in all sub-sequential VI to provide access to the same subsystem of the same board. Use "PD AO Clear.VI" to release the Analog Output subsystem.

Connector Panel



Controls and Indicators

device number Number of the device (beginning from 1) at the bus specified

bus type Bus type where PowerDAQ(tm) device installed

size of buffer Set size of buffer (in samples) for storing data for analog output 512 default but this value should be as low as 1024 if you use clear wave options. Buffer allocated and released inside the PowerDAQ driver.

channels (array of string) channels: specifies the set of analog output channels for a group and task.

Example : if only first row of the array used -

'0,1,2,4' defines four channels in the channel list,

'0' - define empty channel list, '1' define one channel

in the channel list (channel 0),

'0,' define one channel in the channel list (channel 0),

'1,' define one channel in the channel list (channel 0).

channel input - string array channels: specifies the set of analog output channels for a group and task. PowerDAQ limitations : only digital numbers of channels, ',',';' should be used in channel list definition as a channel delimiters.

If more then one row used, they are OR-ed together and following rule is used - '0' in row n means that channel n is not present in the channel list, '1' means that channel n present in the channel list, string separated by commas will be processed using the rule described above for the first row.

Output limits output limits is an array of clusters. This input is reserved for the future use and for now all PowerDAQ boards provides the same +/- 10V fixed analog output range.

- high limit: specifies the maximum scaled data in Volts at a particular channel.

- low limit: specifies the minimum scaled data in Volts at a particular channel.

high limit (10.0) high limit - single : specifies the maximum scaled data in Volts (10.0 - default)

low limit (–10.0) low limit - single : specifies the minimum scaled data in Volts (-10.0 - default)

reference source reference source.

0: Do not change the reference source setting (default input).

1: Internal (default setting).

2: External (reserved).

error in

see **Common Inputs and Outputs** topic for the description

Group This input is not used with PowerDAQ boards because only one upper level VI can use Analog Input subsystem of board at time (see PowerDAQ programmer manual)

allocation mode (no change:0) This input is not used with PowerDAQ boards and ignored

DSP handle structure This input is not used with PowerDAQ

boards and ignored

iteration (reserved)

taskId

see **Common Inputs and Outputs** topic for the description

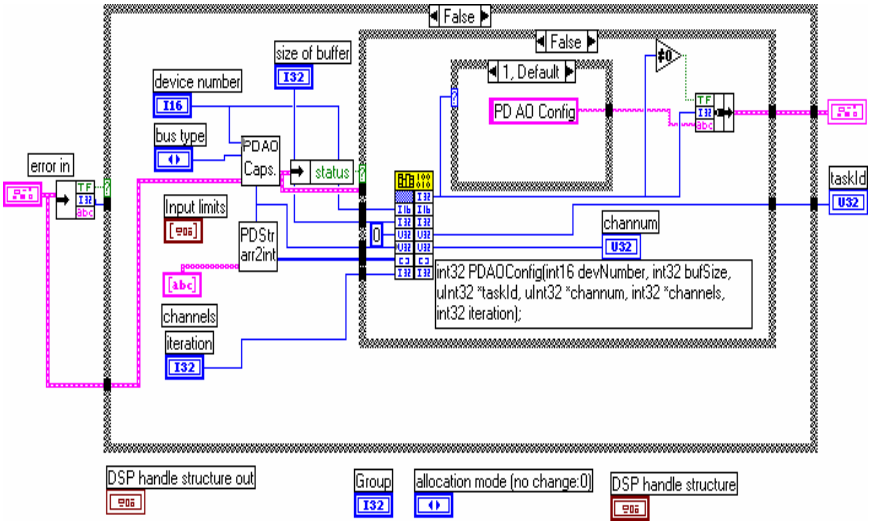
channum Total number of channels in channel list.

error out

see **Common Inputs and Outputs** topic for the description

DSP handle structure out This input is not used with PowerDAQ boards and ignored

Block Diagram



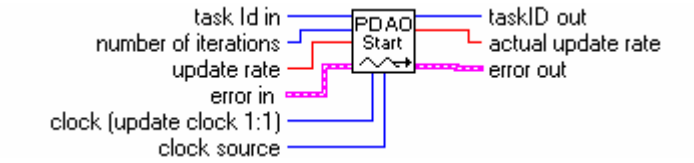
See also

- "PD AO Start.VI"
- "PD AO Write.VI"
- "PD AO Clear.VI"
- "PD AO Buffered Update.VI" (Example VI)

PD AO Start.vi

This VI configures the rate, clock source, sets total number of iterations or continuos mode and starts analog output subsystem on PowerDAQ board, which specified via taskId initially created by "PD AO Config.VI".

Connector Panel



Controls and Indicators

error in

see Common Inputs and Outputs topic for the description

task Id in

see Common Inputs and Outputs topic for the description

number of iterations

Total number of iterations of output buffer that will be passed to the analog output and then acquisition stopped. Specify zero or leave unconnected for continuos acquisition.

Note if iterations value is zero analog output operations will be continued until "PD AI Clear.VI" called.

update rate Frequency of acquisition. For PD-MF(S) boards actual frequency for each channel is the same number because two channels are updated simultaneously. For PD2-AO boards the actual rate per channel can be calculated as (update rate/number of channels updated).

clock This input is used to specify an analog output mode for the PowerDAQ board.

0 – no change

1 – single update mode (used with "PD AO Write.VI" only)

2 – buffered event-based update mode (used with "PD AO

Wave.VI only)

(1024 samples/scans per "PD AO Wave.VI" call maximum)

3 – buffered continuous update mode (used with "PD AO Wave.VI only)

(2048 samples/scans per "PD AO Wave.VI" call maximum)

4 – DSP-based auto-regeneration update mode (used with "PD AO Wave.VI only) (2048 stored on DSP samples/scans maximum)

5 – PowerDAQ buffering buffered update mode (used with "PD AO Wave.VI only) (no limitations for the buffer size)

See *Analog output modes* topic for the details.

U16 clock source clock source.

0: Do not change the clock source setting

1: Internal (LabVIEW picks one; default setting)

2: Use Asynchronous Update Mode (used with "PD AO Wave.VI")

3-7: This values is not supported by PowerDAQ boards and ignored

error out

see Common Inputs and Outputs topic for the description

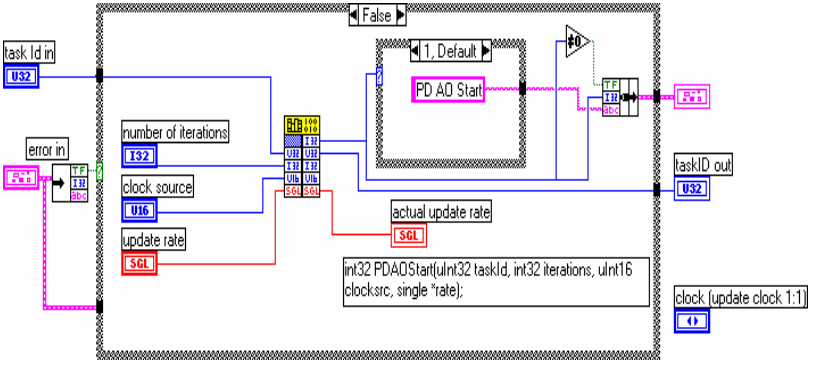
U32 taskId task Id out

see Common Inputs and Outputs topic for the description

SGL actual update rate

actual aggregate output update rate.

Block Diagram



See also

- "PD AO Config.VI"
- "PD AO Write.VI"
- "PD AO Clear.VI"
- "PD AO Buffered Update.VI" (Example VI)

PD AO Clear.VI

This VI stops any analog output acquisition process, frees resources, clear buffers and returns zero taskId. If analog output running in regeneration mode - stops the waveform generation. After you have called this VI you can reuse analog output subsystem for another task under the LabVIEW. All analog outputs can be set to 0 using this VI.

Connector Panel

Controls and Indicators

error in
see **Common Inputs and Outputs** topic for the description

task Id in
see **Common Inputs and Outputs** topic for the description

set to zero If ON(TRUE) set all analog output availiable on board to zero Volts after clearing

task Id out
see **Common Inputs and Outputs** topic for the description

Note *task Id out* is always 0 after "PD AO Clear.VI" call.

error out
see **Common Inputs and Outputs** topic for the description

See also

- "

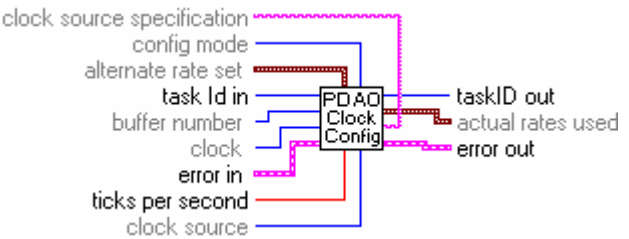
PD AO Config.VI"

- "PD AO Start.VI"
- "PD AO Write.VI"
- "PD AO Buffered Update.VI" (Example VI)

PD AO Clock Config.VI

This VI configures an update or interval clock for analog output.

Connector Panel



Controls and Indicators

U32 task Id in

see Common Inputs and Outputs topic for the description

clock now ignored by PowerDAQ

U16 clock source input- clock source:

0: do not change the clock source setting (default input)

1: internal--LabVIEW picks the clock (default setting)

2-15: This values is not used with PowerDAQ boards and ignored

SGL ticks per second input - update frequency.

(-1.0 - do not change)

alternate rate set input – cluster of alternate rate set: contains the following parameters

- seconds per tick
- timebase source

- timebase signal
- timebase divisor

SGL seconds per tick input - seconds per tick:

-1.0 - default input (not to use this parameter)

0.0 - turns off the clock

timebase source now ignored by PowerDAQ

(return 1)

SGL timebase signal Not used with PowerDAQ boards and ignored

(default timebase for all boards is 11,000,000)

I32 timebase divisor timebase divisor: the divide-down value the VI uses to create the clock rate. The default input is -1, which tells LabVIEW not to change timebase divisor

config mode input - config mode:

0: do not change the config mode setting (default input)

1: change rate immediately (default setting)

2: change rate at the end of the current buffer iteration

3: translate only. Do not change the clock rate settings.

I32 buffer number now ignored by PowerDAQ

error in

see Common Inputs and Outputs topic for the description

abc clock source specification now ignored by PowerDAQ

U32 task Id out

see Common Inputs and Outputs topic for the description

actual rates used output cluster with actual rates used:

- ticks per second used
- seconds per tick used
- timebase signal used
- timebase divisor used: the timebase divisor that the VI used.

sticks per second used ticks per second used

sseconds per tick used seconds per tick

stimebase signal used timebase signal used: the timebase signal expressed in hertz that the VI used when you set timebase source to 1.

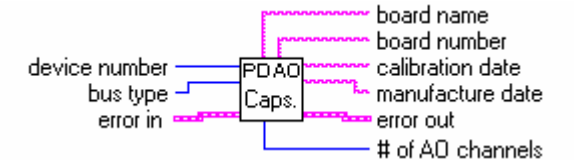
Itimebase divisor used timebase divisor used: the timebase divisor that the VI used.

error out
see Common Inputs and Outputs topic for the description

PD Get AO Capabilities.VI

Collects the information about analog output subsystem of specified board at specified bus, return non zero error code (from “PD Get Capabilities.VI”) if board is not found.

Connector Panel



Controls and Indicators

device number Number of device at bus specified (starts from 1)

bus type Bus type where PowerDAQ device installed

error in
see **Common Inputs and Outputs** topic for the description

board name Type of PowerDAQ board installed

board number Board serial # (from EPROM)

calibration date Board calibration date (from EPROM)

manufacture date Board manufacture date (from EPROM)

# of AO channels Number of analog output channels

max AO rate Maximum value of analog output rate that recommended by PowerDAQ technical documentation.

error out
see **Common Inputs and Outputs** topic for the description

See also

- “

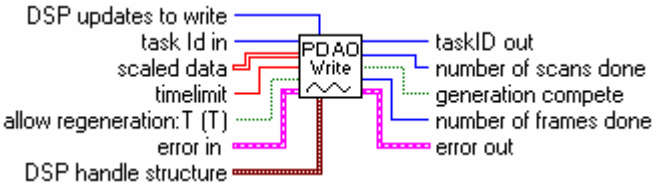
PD AO Config.VI"

- "PD Get Capabilities.VI"

PD AO Write.VI

The "PD AI Write.VI" writes the specified number of scans to the analog output of PowerDAQ board. If regeneration flag is ON (TRUE) true the data will be written continuously until specified in "**PD AO Start.VI**" number of buffers/iterations will be passed to the output or "**PD AO Clear.VI**" called. Note that this VI can be used only with Single Update Mode (see "PD AO Start.VI", *clock* input description).

Connector Panel



Controls and Indicators

task Id in
see **Common Inputs and Outputs** topic for the description

timelimit input - timeout. If not specified calculated automatically by driver

error in
see **Common Inputs and Outputs** topic for the description

scaled data scaled data is a 2D array that contains analog output data in Volts. The data appears in columns, where each column contains the data for a single channel. The second (bottom) dimension selects which channel column. The first (top) dimension selects a single data point for that channel.

scaled data: a two-dimensional array that contains

scaled analog output data . The first dimension is scans, the second dimension is channels.


DSP handle structure This input is not used with PowerDAQ boards and ignored


allow regeneration:T (T) allow regeneration. Regeneration means generating the same data pattern continuously. If allow regeneration is ON (TRUE), when all the data in the buffer has been generated, PowerDAQ board starts generating again from the beginning of the buffer, regardless of whether the data there is new or old. If allow regeneration is OFF (FALSE) and the data about to be generated has already been generated, this VI returns an error. This parameter defaults to TRUE.


number of scans input - number of scans to be written in this VI call


taskID out
see **Common Inputs and Outputs** topic for the description

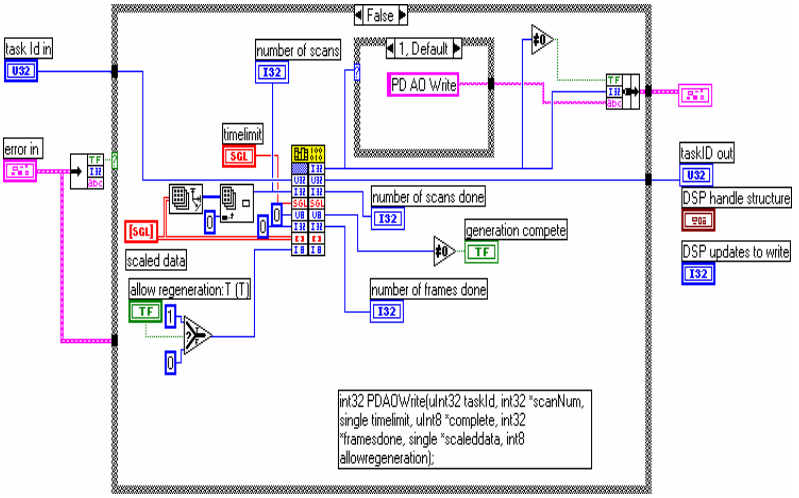

number of scans done output - number of scans that has been written to PowerDAQ board


number of frames done output - number of buffers that has been written to PowerDAQ board


error out
see **Common Inputs and Outputs** topic for the description


generation compete true - if driver already generated all data outputted

Block Diagram



See also

- ”

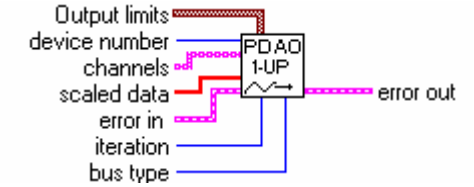
PD AO Config.VI"

- "PD AO Start.VI"
- "PD AO Clear.VI"
- Analog output Example VIs

PD AO Write One Update.VI

Performs single update of each analog output channel in the channel list provided.

Connector Panel



Controls and Indicators

iteration 0 : initialize

else - skip initialization

error in

see **Common Inputs and Outputs** topic for the description

scaled data scaled data is a one-dimensional (1D) array that contains data expressed in the physical units of your signal (Volts). The channel order of the data must be identical to the channel order specified in the *channels* control.

SGL scaled data: a one-dimensional array that contains data expressed in the physical units of your signal, one value for each channel given in the channels array.

Output limits output limits is an array of clusters.

See "**PD AO Config.VI**" for details.

channels (array of string) channels: specifies the set of analog output channels for a group and task.

See "**PD AO Config.VI**" for details.

device number Number of device at bus specified (beginning from 1)

bus type Bus type where PowerDAQ device installed

error out

see **Common Inputs and Outputs** topic for the description

See also

- “

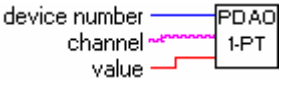
PD AO Config.VI"

- "PD AO Two Channels Single Update.VI" (Example VI)

PD AO Update Channel.VI

"PD AO Update Channel.VI" writes a single value to an analog output channel specified. This VI calls the "**PD AO Write One Update.VI**". This is a simplest way to perform a single update of the single analog output channel.

Connector Panel



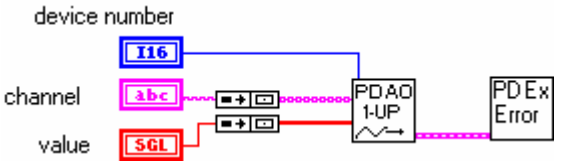
Controls and Indicators

device number Number of the PowerDAQ device (beginning from 1)

channel channel: specifies the analog output channel to be updated

value value: output data in Volts

Block Diagram



PD AO Wave.vi

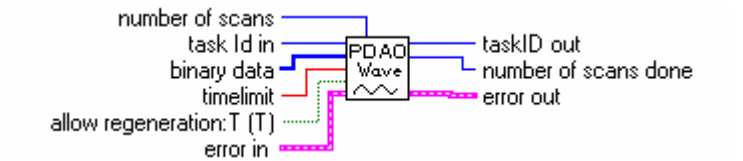
The "PD AO Wave.VI" is almost analog of "PD AO Write.VI" but allowf to use three different buffered waveform modes, which send the data directly to the DSP on-board output buffer.

If Auto-regeneration mode selected the data stored in the on-DSP buffer an application writes data to the buffer of the PD2- MF(S)/AO board and each time the end of buffer is reached, it starts to re-send the same buffer again until "PD AI Clear.VI" called.

Note That this VI can be used only with Buffered Event-Based Update Mode, Buffered Continuous Update Mode and Auto-regeneration Update Mode which specified when "PD AO Start.VI" called (see "**PD AO Start.VI**", *clock* input description). See *Analog output modes* topic for the details about output modes supported.

Note For applications, which require more than 2048 points(PD2-AO)/scans(PD2-MF[S]) use PD AO Buffered Wave.VI.

Connector Panel



Controls and Indicators

U32 task Id in

see Common Inputs and Outputs topic for the description

I32 number of scans input - number of scans to be written in this VI

call

timelimit input - timeout. If not specified calculated automatically by driver

error in

see Common Inputs and Outputs topic for the description

binary data scaled data is a 1D array that contains analog output data in uint32 format. For the PD2-MF/MFS boards each point contains value for two channels (bits 31..24 are ignored, bits 23..12 - channel1 data, bits 11..0 - channel0 data), for the PD2-AO boards each value represents the channel# to write and the data to write (bit 20..16 - channel#, bit 15..0 - 16-bit value).

allow regeneration:T (T) allow regeneration.

taskID out

see Common Inputs and Outputs topic for the description

number of scans done output - number of scans that has been written to PowerDAQ board. This value can be used to keep track of the data outputted.

error out

see Common Inputs and Outputs topic for the description

See also

- “

PD AO Config.VI"

- "PD AO Start.VI"
- "PD AO Clear.VI"
- "PD AO Buffered Wave.VI"
- "PD AO32 Wave.VI" (Example VI)

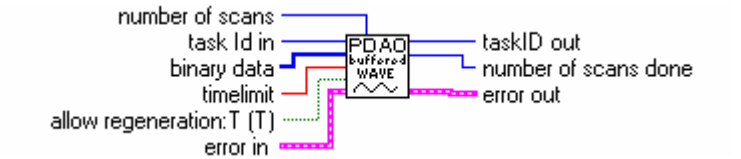
PD AO Buffered Wave.VI

The "PD AO Buffered Wave.VI" is an advanced and specialized version of "PD AO Wave.VI" and allow to use unlimited output buffer size (we recommend to keep buffer size in 1 Megabyte limits). Only one special output mode supported (this mode is a specific for the PowerDAQ LabVIEW driver).

Note That this VI can be used only with PowerDAQ Buffered Event-Based Update Mode which specified when "PD AO Start.VI" called (see "PD AO Start.VI", clock input description). See *Analog output update modes* topic for the details about output modes supported.

Note This VI should be used for applications, which require more than 2048 points (PD2-AO)/scans(PD2-MF[S]) of data in the buffer.

Connector Panel



Controls and Indicators

U32 task Id in

see Common Inputs and Outputs topic for the description

I32 number of scans input - number of scans to be written in this VI call

SGL timelimit input - timeout. If not specified calculated

automatically by driver

error in

see Common Inputs and Outputs topic for the description

binary data scaled data is a 1D array that contains analog output data in uint32 format. For the PD2-MF/MFS boards each point contains value for two channels (bits 31..24 are ignored, bits 23..12 - channel1 data, bits 11..0 - channel0 data), for the PD2-AO boards each value represents the channel# to write and the data to write (bits 20..16 - channel#, bits 15..0 - 16-bit value).

allow regeneration:T (T) allow regeneration.

taskID out

see Common Inputs and Outputs topic for the description

number of scans done output - number of scans that has been written to PowerDAQ board.

this value can be used to keep track of the data outputted.

error out

see Common Inputs and Outputs topic for the description

See also

- "

PD AO Config.VI"

- "PD AO Start.VI"
- "PD AO Clear.VI"
- "PD AO Wave.VI"
- "PD AO32 Buffered Wave.VI" (Example VI)

Digital I/O

Digital Input and Output subsystems are additional subsystems for PowerDAQ MF[S]/AO boards and main subsystems on the DIO boards.

Those subsystems are generally has associated with it a certain number of digital ports, 8 or 16 bits wide. Most but not all boards have ports that have dedicated input or output assigning.

The Digital I/O VIs in the PowerDAQ™ Library include:

- Configuration Functions
  - PD DIO Config.vi
  - PD DIO Start.vi
  - PD DIO Clear.vi
- Digital I/O Functions
  - PD DIO Read.vi
  - PD DIO Write.vi
  - PD DIO-128 Read.vi
  - PD DIO-128 Write.vi
  - PD DIO Buffer Read.vi
  - PD DIO Buffer Write.vi
- Simplified Easy Functions
  - PD DIO Port Read.vi
  - PD DIO Port Write.vi
  - PD Read from Digital Line.vi
  - PD Read from Digital Port.vi
  - PD Write to Digital Line.vi
  - PD Write to Digital Port.vi

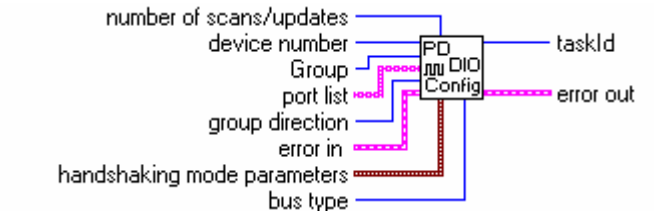
DIO Boards Support

Most of the PowerDAQ Digital Input and Output VIs in the Library are generally can be used with both MF[S]/AO and DIO series boards. If some of them are dedicated to use with only one board type this is noted in the VI description.

PD DIO Config.VI

This VI is used to configure the digital ports as input or output and returns taskId, set the channel (port) list and acquisition buffer size for digital input or output operation for the board, specified in the *device number* parameter. This VI checks the board availability and if specified board available and Digital Input or Output subsystem exists on the board and not used by another VI the unique *taskId* will be issued. It should be used in all sub-sequential VI to provide access to the same subsystem of the same board. Use "PD DIO Clear.VI" to release the Digital Input or Output subsystem.

Connector Panel



Controls and Indicators

device number Number of device at bus specified (beginning from 1)

bus type Bus type where PowerDAQ device installed

number of scans/updates Set size of buffer (in bytes) for storing the data to/from board - 1000 default

port list (array of string) channels: specifies the set of digital input or output channels for a group and task.

Example : if only first row of the array used -

'0,1,2,4' defines four channels in the channel list,

'0' - define empty channel list, '1' define one channel in the channel list (channel 0),

'0,' define one channel in the channel list (channel 0),

'1,' define one channel in the channel list (channel 1).

channel input - string array channels: specifies the set of

analog output channels for a group and task. PowerDAQ limitations : only digital numbers of channels, ',','-' should be used in channel list definition as a channel delimiters.

If more then one row used, they are OR-ed together and following rule is used - '0' in row n means that channel n is not present in the channel list, '1' means that channel n present in the channel list, string separated by commas will be processed using the rule described above for the first row.

error in

see **Common Inputs and Outputs** topic for the description

Group This input is not used with PowerDAQ boards because only one upper level VI can use Analog Input subsystem of board at time (see PowerDAQ programmer manual)

handshaking mode parameters This input is not used with PowerDAQ boards and ignored

group direction group direction sets the direction for the channel list specified in *port list* parameter.

0: Do not change the group direction setting (reserved)

1: Input (default setting)

2: Output

3: Bi-directional (reserved)

taskId

see **Common Inputs and Outputs** topic for the description

error out

see **Common Inputs and Outputs** topic for the description

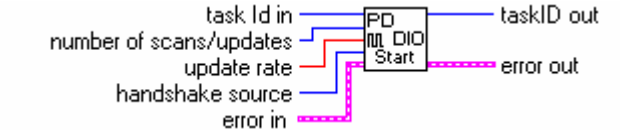
See also

- "PD DIO Start.VI"
- "PD DIO Clear.VI"
- "PD DIO Buffered Read[Write].VI" (Example VI)

PD DIO Start.VI

This VI configures the rate and sets total number of scans to be read or write from/to DIO subsystem in PowerDAQ board that specified via taskId.

Connector Panel



Controls and Indicators

error in

see Common Inputs and Outputs topic for the description

task Id in

see Common Inputs and Outputs topic for the description

number of scans/updates number of scans/updates to read or write specifies the total number of scans that you ask this VI to read or write before the operation completes.

Note -1 leave the number of scans/updates unchanged(size of buffer used) 0 means that PowerDAQ LabVIEW driver acquires or generates data indefinitely, until you clear the operation using the "PD DIO Clear.vi".

update rate Frequency/rate of Digital I/O operation.

handshake source Not currently supported by PowerDAQ boards

error out

see Common Inputs and Outputs topic for the description

taskID out

see Common Inputs and Outputs topic for the description

See also

- "PD DIO Config.VI"

- "PD DIO Clear.VI"
- "PD DIO Buffered Read[Write].VI" (Example VI)

PD DIO Clear.VI

This VI stops any digital input or output acquisition process, frees resources, clear buffers and returns zero taskId. If digital output running in regeneration mode - stops the pattern generation. After you have called this VI you can reuse digital input or output subsystem for another task under the LabVIEW. All digital outputs can be set to 0 using this VI.

Connector Panel

Controls and Indicators

error in
see Common Inputs and Outputs topic for the description

task Id in
see Common Inputs and Outputs topic for the description

set to zero If ON(TRUE) set all digital output available on board to logic zero after clearing

task Id out output - always 0
see Common Inputs and Outputs topic for the description

error out
see Common Inputs and Outputs topic for the description

See also

- "PD DIO Config.VI"
- "PD DIO Start.VI"

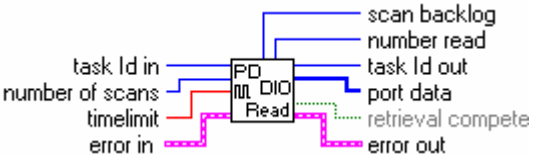
- "PD DIO Buffered Read[Write].VI" (Example VI)

PD DIO Read.VI

"PD DIO Read.VI" is a main function that allows receiving the digital data from the PowerDAQ board continuously. "PD DIO Read.VI" reads the specified number of scans and returns the data.

Note That board should be properly initialized using "PD DIO Config.VI" and buffered operation started in "PD DIO Start VI".

Connector Panel



Controls and Indicators

U32 task Id in

see Common Inputs and Outputs topic for the description

I32 number of scans input - number of scans to be read from driver.

read location input - cluster with read location

I32 read offset read offset.The default setting and input are 0.

read mode read mode

0 - no change,

1 – from read mark,

2 – from start of buffer,

3 – from end of data

SGL timelimit input - timeout for wait data transfer timeout to driver. If not specified calculated automatically by driver

error in

see Common Inputs and Outputs topic for the description

TF retrieval compete TRUE if all requested data are read from driver

error out

see Common Inputs and Outputs topic for the description

[U8] port data port data: a 1D array that contains the digital data that the VI obtained from the internal buffer (in bytes).

Note That for the PD-DIO boards input data is 16 bits wide, you may want to use the Type Cast primitive in the Miscellaneous Function menu to convert your data to an array of u16s.

U8 Data from driver

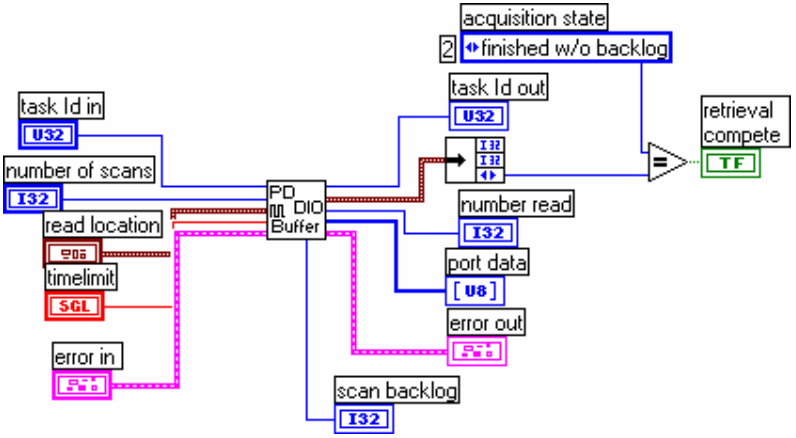
I32 number read output – number of scans that was read from PowerDAQ board

U32 task Id out

see Common Inputs and Outputs topic for the description

I32 scan backlog output – number of scans of lost data

Block Diagram



See also

- "PD DIO Config.VI"
- "PD DIO Start.VI"

- "PD DIO Clear.VI"
- "PD DIO Buffered Read[Write].VI" (Example VI)

PD DIO Write.VI

This VI writes/updates an array of data to the PowerDAQ driver buffers for digital output.

Note That board should be properly initialized using "PD DIO Config.VI". Buffered operation should be started by "PD DIO Start VI" after this VI call.

Connector Panel

Controls and Indicators

U32 task Id in

see Common Inputs and Outputs topic for the description

write location Cluster which contains information about write position

I32 write offset Input - write offset: offset of the write mark .(-1 - do not change the write offset settings)

write mode Input - write mode(s):

0: do not change the write mode setting (default input)

1: relative to the write mark (default setting)

2: relative to the beginning of the buffer

[U8] port data port data: a 1D array of bytes that contains the digital data that this VI writes to the internal buffer.

Note That for the PD-DIO boards data is 16 bits wide, you

may want to use the Type Cast primitive in the Miscellaneous Function menu to convert your data to an array of u16s.

U8 Port data to be written

error in

see Common Inputs and Outputs topic for the description

SGL timelimit input – timeout for wait data transfer to driver. If not specified calculated automatically by driver

U32 task ID out

see Common Inputs and Outputs topic for the description

error out

see Common Inputs and Outputs topic for the description

I32 buffer iterations output- indicates the number of complete iterations of the buffer (frame) that has been written to the digital output

TF generation compete true - if driver already generated all required data

See also

- "PD DIO Config.VI"
- "PD DIO Start.VI"
- "PD DIO Clear.VI"
- "PD DIO Buffered Read[Write].VI" (Example VI)

PD DIO-128 Read.vi

The "PD DIO-128 Read.VI" reads the logical state of a digital port specified. Ideal for non-timed Digital I/O operation

Note That board should be properly initialized using "PD DIO Config.VI". Only PD2-DIO-64 and PD2-DIO-128 boards supported by this VI.

Connector Panel

Controls and Indicators

U32 task Id in

see Common Inputs and Outputs topic for the description

error in

see Common Inputs and Outputs topic for the description

U16 Port input - Port # to be read. Can be 0..3 for PD2-DIO-64 and 0..7 for PD2-DIO-128 boards

error out

see Common Inputs and Outputs topic for the description

U32 task Id out

see Common Inputs and Outputs topic for the description

U16 Value output - 16-bits data read from the port specified

See also

- "PD DIO-128 Write.VI"
- "PD DIO-128 Single Read and Write.VI" (Example VI)

PD DIO–128 Write.VI

The "PD DIO-128 Write.VI" write the data specified in value parameter to the digital port specified via taskIdIn. Ideal for non-timed Digital I/O operation

Note That board should be properly initialized using "PD DIO Config.VI". If specified port is not enabled to be output, the data will be stored in output register but output state will be high-impedance. Only PD2-DIO-64 and PD2-DIO-128 boards supported by this VI.

Connector Panel

Controls and Indicators

U32 task Id in

see Common Inputs and Outputs topic for the description

error in

see Common Inputs and Outputs topic for the description

U16 port input - Port # to be written. Can be 0..3 for PD2-DIO-64 and 0..7 for PD2-DIO-128 boards

U16 value input - 16-bits port data to be written

error out

see Common Inputs and Outputs topic for the description

U32 task Id out

see Common Inputs and Outputs topic for the description

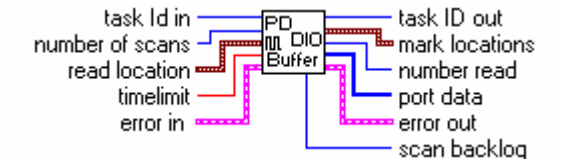
See also

- "PD DIO-128 Read.VI"
- "PD DIO-128 Single Read and Write.VI" (Example VI)

PD DIO Buffer Read.VI

This VI returns digital input data from the internal data buffer.

Connector Panel



Controls and Indicators

task Id in

see Common Inputs and Outputs topic for the description

number of scans input - number of scans for read from driver

read location This cluster contains information about read location

- read offset read offset
  The default setting and input are 0
- read mode read mode
  - 0 - no change
  - 1 - from read mark
  - 2 - from start of buffer
  - 3 - from end of buffer

timelimit input - timeout for wait data from driver. If not specified calculated automatically by driver

error in

see Common Inputs and Outputs topic for the description

mark locations This cluster contains information about next read location

- read mark scan read mark scan: the number of the scan that the VI will read the next time you call this VI
- end of data scan end of data scan: the number of the most recently acquired scan
- acquisition state acquisition state: has the following values.
  - 0: running
  - 1: finished with backlog
  - 2: finished with no backlog
  - 3: paused
  - 4: no acquisition

number read output - number of scans that really readed from PowerDAQ board

task ID out

see Common Inputs and Outputs topic for the description

scan backlog output - number of scans of loosed data

port data port data: a 1D array that contains the digital data that the VI obtained from the internal buffer

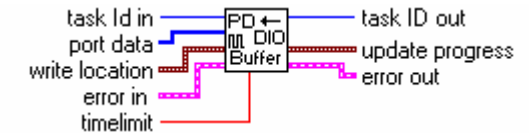
- port data

error out
see Common Inputs and Outputs topic for the description

PD DIO Buffer Write.VI

This VI writes the digital data to the array that created in "PD DIO Config.VI"

Connector Panel



Controls and Indicators

task Id in
see Common Inputs and Outputs topic for the description

write location Contains information about write position

write offset Input - write offset: offset of the write mark .(-1 - do not to change the write offset setting). The default setting is 0.

write mode Input - write mode(s):

0: do not change the write mode setting (default input)

1: relative to the write mark (default setting)

2: relative to the beginning of the buffer

timelimit input - timeout for wait data transfer timeout to the driver. If not specified calculated automatically by driver

port data port data: a 1D array that contains the digital data that the VI written to the internal buffer.

array with output data

error in
see Common Inputs and Outputs topic for the description

update progress Cluster - contains information about output progress

I32 write mark output - points to the next update to which the VI will write.

I32 output mark output - points to the next update the VI will generate.

I32 buffer iterations output- indicates the number of complete iterations of the buffer (frame) that written to digital output

buffer state output- buffer state: has the following values

0: no such buffer (default setting)

1: waiting for data

2: ready

3: active

4: finished

for current release of PowerDAQ LabVIEW driver this output always

is 2 (ready)

U32 task ID out

see Common Inputs and Outputs topic for the description

error out

see Common Inputs and Outputs topic for the description

PD DIO Port Read.vi

The "PD DIO Port Read.VI" reads the logical state of a digital port specified. Ideal for non-timed Digital I/O operation.

Note That board should be properly initialized using "PD DIO Config.VI". Only PD2-MF[S] and PD2-AO boards supported by this VI.

Connector Panel

Controls and Indicators

task Id in

see Common Inputs and Outputs topic for the description

error in

see Common Inputs and Outputs topic for the description

line mask input - 32-bits line mask determines which lines this VI reads. The default setting for line mask is -1(all channels presents in output pattern).

The VI sets the bits of pattern that correspond to lines the VI has not read to 0

error out

see Common Inputs and Outputs topic for the description

task Id out

see Common Inputs and Outputs topic for the description

pattern output - 32-bits pattern: a bit pattern representing the state of the lines in the port

See also

- "PD DIO Port Write.VI"
- "PD DIO Config.VI"
- "PD DIO Single Read.VI" (Example VI)

PD DIO Port Write.vi

The "PD DIO Port Write.VI" write the data specified in value parameter to the digital port specified via taskIdIn. Ideal for non-timed Digital I/O operation

Note That board should be properly initialized using "PD DIO Config.VI". Only PD2-MF[S] and PD2-AO boards are supported by this VI.

Connector Panel

Controls and Indicators

task Id in
see **Common Inputs and Outputs** topic for the description

error in
see **Common Inputs and Outputs** topic for the description

line mask input - 32-bits line mask determines which lines this VI writes. The default setting for line mask is -1(all channels presents in output pattern).

The VI sets the bits of pattern that correspond to lines the VI has not write to 0.

pattern input - 32-bits pattern: a bit pattern representing the state of the lines in the port.

error out
see **Common Inputs and Outputs** topic for the description

task Id out
see **Common Inputs and Outputs** topic for the description

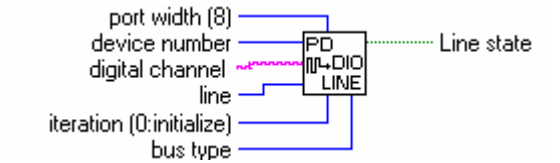
See also

- "PD DIO Port Read.VI"
- "PD DIO Config.VI"
- "PD DIO Single Write.VI" (Example VI)

PD Read from Digital Line.VI

The "PD Read from Digital Line.VI" reads the logical state of a digital line specified via line and port number. A Boolean result indicates the value read from the line as either logic high or logic low.

Connector Panel



Controls and Indicators

I32 iteration (0:initialize) 0 : initialize

else - skip initialization

I16 device number Number of device at bus specified (beginning from 1)

bus type Bus type where PowerDAQ device installed

abc digital channel Not currently supported by PowerDAQ boards - specify channel via port and line number

I16 port width port width is the total width in bits of the port.

8 bits for PD-MF/AO boards and 16 bits for PD2-MF[S]/DIO boards

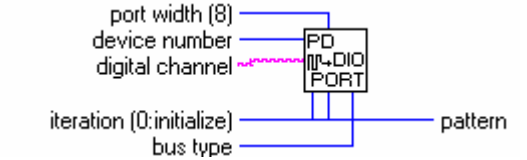
I16 line line specifies the individual port bit or line to be used for I/O

TF Shows the state of specified digital line (TRUE or FALSE)

PD Read from Digital Port.vi

The PD Read from Digital Port VI reads the logical state of the digital port that specified via device number.

Connector Panel



Controls and Indicators

line mask input - 32-bits line mask determines which lines this VI reads. The default setting for line mask is -1(all channels presents in output pattern).

The VI sets the bits of pattern that correspond to lines the VI has not read to 0.

port width (8) port width is the total width in bits of the port.

8 bits for PD-MF/AO boards and 16 bits for PD2-MF[S]/DIO boards

digital channel Not currently supported by PowerDAQ boards

bus type Bus type where PowerDAQ device installed

device number Number of device at bus specified (beginning from 1)

iteration (0:initialize) 0 : initialize

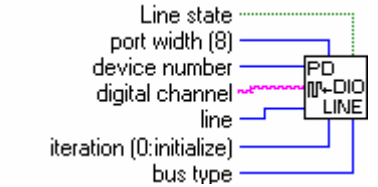
else - skip initialization

pattern output - 32-bits pattern: a bit pattern representing the state of the lines in the port.

PD Write to Digital Line.vi

The "PD Write to Digital Line.VI" writes the logical state to the digital line that specified via line and port number. A Boolean result indicates the value write to the line as either logic high or logic low.

Connector Panel



Controls and Indicators

I32 iteration (0:initialize) 0 : initialize
else - skip initialization

I16 device number Number of device at bus specified (beginning from 1)

bus type Bus type where PowerDAQ device installed

abc digital channel Not currently supported by PowerDAQ boards - specify channel via port and line number

I16 port width (8) port width is the total width in bits of the port.

8 bits for PD-MF/AO boards and 16 bits for PD2-MF[S]/DIO boards

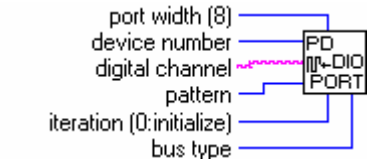
I16 line line specifies the individual port bit or line to be used for I/O.

TF Line state Shows the state of specified digital line (TRUE or FALSE)

PD Write to Digital Port.vi

The "PD Write to Digital Port.VI" writes the logical state to the digital port that specified via device number.

Connector Panel



Controls and Indicators

line mask input - 32-bits line mask determines which lines this VI writes. The default setting for line mask is -1(all channels presents in output pattern).

The VI sets the bits of pattern that correspond to lines the VI has not write to 0.

pattern output - 32-bits pattern: a bit pattern representing the state of the lines in the port.

port width (8) port width is the total width in bits of the port.

8 bits for PD-MF/AO boards and 16 bits for PD2-MF[S]/DIO boards

digital channel Not currently supported by PowerDAQ boards

bus type Bus type where PowerDAQ device installed

device number Number of device at bus specified (beginning from 1)

iteration (0:initialize) 0 : initialize

else - skip initialization

Counters/Timers

This subsystem generally has associated with it a certain number of user-accessible on-board counter/timers, 16 or 24 bits wide. They can be programmed either as inputs (counters) – to measure frequency, pulse width/period or outputs (timers) to provide variety of output frequencies.

The Counter/Timer VIs in the PowerDAQ™ Library include:

- PD UCT Control.vi
- PD UCT Freq Get.vi
- PD UCT Freq Ready.vi
- PD UCT Freq Start.vi
- PD UCT Get timebase.vi
- PD UCT DSP Config.vi
- PD UCT DSP Read.vi

Multifunction Boards Support (82C54–based)

The multifunction boards (PDx-MF[S] series) uses the standard 82C54-based counter/timer. Standard UCT support includes :

- PD UCT Control.vi
- PD UCT Get timebase.vi

All three counters are available to the user application. Thanks to the on-board DSP all three counters are able to measure the input frequency in range 1Hz –65534Hz. The frequency mesurement VIs includes :

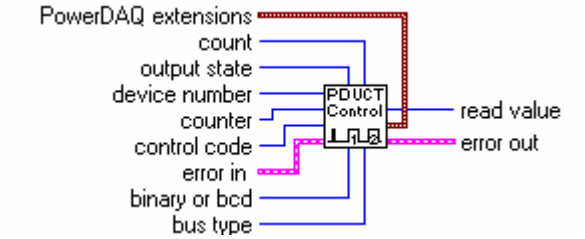
- PD UCT Freq Get.vi
- PD UCT Freq Ready.vi
- PD UCT Freq Start.vi

PD UCT Control.VI

This VI provides an access to the 82c54 counter/timer based UCT subsystem on the PowerDAQ MF[S] board.

PowerDAQ boards counter/timer subsystem provides additional flexibility with using software gate/clock, external clock and frequency counting.

Connector Panel



Controls and Indicators

device number Number of device at bus specified (beginning from 1)

bus type Bus type where PowerDAQ device installed

counter Specify 82c54 counter number from 0 through 2.

control code Input - control code determines the mode in which the counter operates. Modes from 0 to 5 corresponding to 82c54 modes.

0: Interrupt or terminal count (default)
1: Hardware retriggerable one-shot
2: Rate generator
3: Square wave rate generator
4: Software-triggered strobe
5: Hardware-triggered strobe (retriggerable)
6: Read counter/timer value
7: Reset counter/timer subsystem
8: Leave mode unchanged
9: Reserved
10: Frequency measurement mode start
11: Frequency measurement result read

In mode 0, the output goes low after the mode set operation, and the counter begins to count down while the gate

(software or hardware) input is high. The output goes high when counter reaches the terminal count (that is, when the counter decrements to 0) and stays high until counter mode will be written again.

In mode 1, the output goes low on the count following the leading edge of the gate (software or hardware) input and goes high on terminal count.

In mode 2, the output goes low for one period of the clock input. count indicates the period from one output pulse to the next.

In mode 3, the output stays high for one-half of the count clock pulses and stays low for the other half.

In mode 4, the output is initially high, and the counter begins to count down while the gate (software or hardware) input is high. On terminal count, the output goes low for one clock pulse, then goes high again.

Setup mode 5 is similar to mode 4, except that the gate (software or hardware) input triggers the count to start.

Mode 10 enforces counter to start count external event coming in 1.00000 second input interval. The result can be read using the mode 11.

Mode 11 allows read the result of frequency measurement. Note that frequency should be in 1-65534Hz limits. 0 Hz will be read as 65535.

See the 82c54 Programmable Interval Timer data sheet in your product user manual for a detailed description of these modes and the associated timing diagrams.

count input - 16-bit unsigned value that defines the period from one output pulse to the next. If control code is 0, 1, 4, or 5, count can be 0 through 65,535 in binary counter operation and 0 through 9,999 in BCD counter operation. If control code is 2 or 3, count can be 2 through 65,535 and 0 in binary counter operation and 2 through 9,999 and 0 in BCD counter operation.

binary or bcd Input - defines the counter/time count mode

0: 4-decade BCD counter

1: 16-bit binary counter

output state Reserved

error in

see **Common Inputs and Outputs** topic for the description

PowerDAQ extensions

Additional features for PowerDAQ boards

- clock source
- gate source
- software gate state

clock source Defines the clock source for the timer/counter counts.

PowerDAQ board provides three types of counting base clock :

Software – each count-down event specified via software

1 MHz - 1 MHz internal timebase clock

External - external clock provided

if this input sets to 'no change' value previous settings still take effect

gate source Specify gate input for counter/timer source

PowerDAQ board provides two types of timer gate :

Software – gate turned on and off via software

External - external gate used

if this input sets to 'no change' value previous settings still take effect

software gate Software gate value (used with gate source = 'software')

Enabled - enable counting

Disabled - disable counting

if this input sets to 'no change' value previous settings still take effect

read value output - 16-bit unsigned read value: When you set control code to 6/11 (read), read value returns the value the VI read from the counter specified.

Error out

see **Common Inputs and Outputs** topic for the description

See also

- PD UCT Freq Get.vi
- PD UCT Freq Ready.vi
- PD UCT Freq Start.vi
- PD UCT Frequency Counter.VI (Example VI)
- PD UCT Multimode Counter.VI (Example VI)

PD UCT Get timebase.vi

This VI returns PowerDAQ board counter/timer subsystem base frequency in Hz.

Connector Panel

Controls and Indicators

error in

see Common Inputs and Outputs topic for the description

device number Number of device at bus specified (beginning from 1)

bus type Bus type where PowerDAQ device installed

error out

see Common Inputs and Outputs topic for the description

timebase frequency PowerDAQ board counter/timer subsystem base frequency in Hz.

This value has sense only when internal clock used.

timebase period PowerDAQ board counter/timer subsystem

timebase period in seconds.

PD UCT Freq Start.vi

Starts the UCT-based frequency measurement.

This VI provides access to 82c54 counter/timer based UCT subsystem on PowerDAQ board for the specific task - input frequency measurement. The input frequency can be measured at any of the counter input. The unused counters will continue to perform they operation. The "PD UCT Freq Start.VI" starts 1 sec DSP-based measurement interval. For the frequency in range 1 Hz - 65534 Hz PowerDAQ MF/MFS boards provides 1Hz accuracy in a results achieved. For the higher frequencies (up to 11 MHz) one counter can be used as a divider of the input frequency.

This VI should be used in combination with "PD UCT Freq Get.VI" and "PD UCT Freq Ready.VI"

The following algorithm should be used:

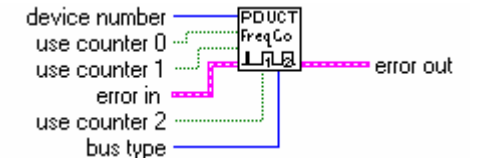
1. Call "PD UCT Freq Start.VI"

2. Wait more than 1 sec or use "PD UCT Freq Ready.VI" to find out when the measurement results are ready.

3. Read the results using "PD UCT Freq Get.VI".

Note For the complete example see "PD UCT Frequency Counter.VI" and "PD AIn & UCT.VI."

Connector Panel



Controls and Indicators

device number Number of device at bus specified (beginning from 1)

bus type Bus type where PowerDAQ device installed

error in

see **Common Inputs and Outputs** topic for the description

use counter 0 If true (ON) - use counter 0 for the frequency measurement

use counter 1 If true (ON) - use counter 1 for the frequency measurement

use counter 2 If true (ON) - use counter 2 for the frequency measurement

error out

see **Common Inputs and Outputs** topic for the description

See also

- PD UCT Freq Get.vi
- PD UCT Freq Ready.vi
- PD UCT Control.vi
- PD UCT Frequency Counter.VI (Example VI)

PD UCT Freq Ready.vi

Checks for the availability of the frequency measurement results.

This VI provides access to the 82c54 counter/timer based UCT subsystem on the PowerDAQ board to perform an input frequency measurement. The input frequency can be measured at any of the counter inputs. The unused counters will continue to perform the operation. An error code (-10800) will be set and zero frequency values will be returned if the measured data is not ready. For the frequency ranges 2 Hz - 65500 Hz, the PowerDAQ MF/MFS boards provide 1 Hz accuracy. For higher frequencies (up to 11 MHz) one counter can be used as a divider for the input frequency.

This VI should be used in combination with "PD UCT Freq Get.VI" and "PD UCT Freq Ready.VI"

The following algorithm should be used:

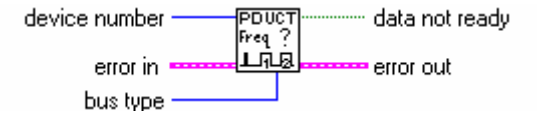
1. Call "PD UCT Freq Start.VI"

2. Wait more than 1 sec or use "PD UCT Freq Ready.VI" to find out when the measurement results are ready.

3. Read the results using "PD UCT Freq Get.VI".

Note For the complete example see "PD UCT Frequency Counter.VI" and "PD AIn & UCT.VI."

Connector Panel



Controls and Indicators

device number Number of device at bus specified (beginning from 1)

bus type Bus type where PowerDAQ device installed

error in

see **Common Inputs and Outputs** topic for the description

error out

see **Common Inputs and Outputs** topic for the description

TF **data not ready** true (ON) if frequency measurement process is in a progress

I32 **result** reserved

See also

- PD UCT Freq Get.vi
- PD UCT Freq Start.vi
- PD UCT Control.vi
- PD UCT Frequency Counter.VI (Example VI)

PD UCT Freq Get.VI

Reads the result of the UCT-based frequency measurement.

This VI provides access to the 82c54 counter/timer based UCT subsystem on the PowerDAQ board to perform an input frequency measurement. The input frequency can be measured at any of the counter inputs. The unused counters will continue to perform the operation. An error code will be set and zero frequency values will be returned if the measured data is not ready. For the frequency ranges 2 Hz - 65500 Hz, the PowerDAQ MF/MFS boards provide 1 Hz accuracy. For higher frequencies (up to 11 MHz) one counter can be used as a divider for the input frequency.

This VI should be used in combination with "PD UCT Freq Start.VI" and "PD UCT Freq Ready.VI"

The following algorithm should be used:

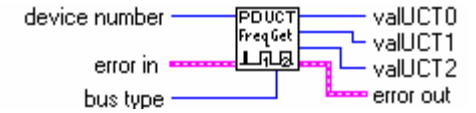
1. Call "PD UCT Freq Start.VI"

2. Wait more than 1 sec or use "PD UCT Freq Ready.VI" to find out when the measurement results are ready.

3. Read the results using "PD UCT Freq Get.VI".

Note For the complete example see "PD UCT Frequency Counter.VI" and "PD AIn & UCT.VI."

Connector Panel



Controls and Indicators

device number Number of device at bus specified (beginning from 1)

bus type Bus type where PowerDAQ device installed

error in

see **Common Inputs and Outputs** topic for the description

error out

see **Common Inputs and Outputs** topic for the description

valUCT0 u16 - result of frequency measurement at UCT 0 input

valUCT1 u16 - result of frequency measurement at UCT 0 input

valUCT2 u16 - result of frequency measurement at UCT 0 input

See also

- PD UCT Freq Ready.vi
- PD UCT Freq Start.vi
- PD UCT Control.vi
- PD UCT Frequency Counter.VI (Example VI)

DSP-based Counters Support

Depending on your PowerDAQ™ PD2-DIO/AO operation mode, the board can support up to the three DSP based 24-bit counter/timers with a maximum count rate up to 33 MHz for an internal base clock and 16.5 MHz for the external clock. The minimum count rate is 0.001 Hz for the internal clock and has no low limits for the external clock.

Note For all waveform modes, only Timer0 and Timer1 are available to use.

Programming of the counter/timer subsystem of the PD2-DIO/AO board require a minimal understanding of the Triple Timer Module of the Motorola 56301 DSP. For this information please refer to the *Motorola DSP 56301 User Manual* (Motorola P/N DSP56301UM/AD).

Each timer can be used for internal or external clocking and can interrupt the DSP56301 after a specified number of events (clocks) or signal an external device after counting internal events. Each timer can also be used to trigger DMA transfers after a specified number of events (clocks) have occurred. Each timer connects to the external world through one bi-directional pin TIO that is 7kV ESD protected. When TIO is configured as input the timer functions as an external event counter or can measure external pulse width/signal period. When TIO is used as output, it functions as a timer, watchdog or Pulse Width Modulator.

Some common timer/counter/output functions that microprocessors require are:

- Real time clock,
- Event counter
- Digital one-shot
- Programmable rate generator
- Square wave generator
- Binary rate multiplier
- Complex digital wave form generator
- Complex motor control

Note That

Each counter is a 24-bits wide count-up counter

After power-up, the count value and output of all counters are set to zero

Each counter must be programmed before it can be used

Unused counters need not to be programmed

Counters are programmed by using the PowerDAQ™ LabVIEW driver VIs

Each counter is fully independent of the others. Only prescaler, when used, is a common part of the counter-timer subsystem. Each counter may operate in a different mode

PD UCT DSP Config.VI

Set/change configuration of 24-bits wide DSP counter/timer.

Note That this VI can be used only with PD2-AO/DIO boards.

PowerDAQ Examples

PowerDAQ™ Examples

The PowerDAQ LabVIEW driver includes extensive set of examples, which you can use as a start point to build you own application. In the our installation we are divided all example VIs to the six categories depends of they functionality:

- Analog Input
- Analog Output
- Digital Input and Output
- Counter/Timer
- Multiple Subsystem
- Multiple Boards

Most of the examples are simple programs, which allows to perform only one specific task or highlights specific feature of the specific subsystem. You can easily combine them to provide more functionality or use more complex examples to start you design. In this manual each example showed contains Front Panel, Diagram and brief explanation about what this particular VI does. Additionally, if some more complex or advanced features provided there are a special topic in each example called **Features** which explains them.

Note that goal of this part of the manual was not to describe each and every sub-VI of the example VI but give you a necessary information which allows you to start or continue you own task more faster and reliable.

Common Controls

There are some common controls that could be presents on examples

- *deviceId*

Device Device: the device number you assigned to the PowerDAQ board during configuration. This parameter defaults to 1. Use this parameter for multiple boards.

- *channels*

Channels channels: specifies the set of input/output channels for a group and task.
Example : if only first row of the array used -
'0,1,2,4' defines four channels in the channel list,
'0' – define empty channel list, '1' define one channel in the channel list (channel 0),
'0,' define one channel in the channel list (channel 0),
'1,' define one channel in the channel list (channel 1).

channel input - string array channels: specifies the set of input/output channels for a group and task.
PowerDAQ limitations : only digital numbers of channels, ',',';' should be used in channel list definition as a channel delimiters.
If more then one row used, they are OR-ed together and following rule is used - '0' in row n means that channel n is not present in the channel list, '1' means that channel n present in the channel list, string separated by commas will be processed using the rule described above for the first row.

- *web-link button*

Click to visit our web-site to download the latest version of SDK or up-to-date information about our products

- *stop button*

stop Stops the current VI execution

Diagnostic VI

For the quick diagnostic the PowerDAQ Library includes the "PD Diagnostic.VI". This VI checks the board availability communicates with the board and shows the detailed information about PowerDAQ board(s) installed.

For the functional check we recommend to use following VIs, depends of type of the board to be checked:

"PD All Subsystem.VI" – for the PDx-MF[S] series

"PD DIO-128 Single Read And Write.VI " – for the PD2-DIO boards

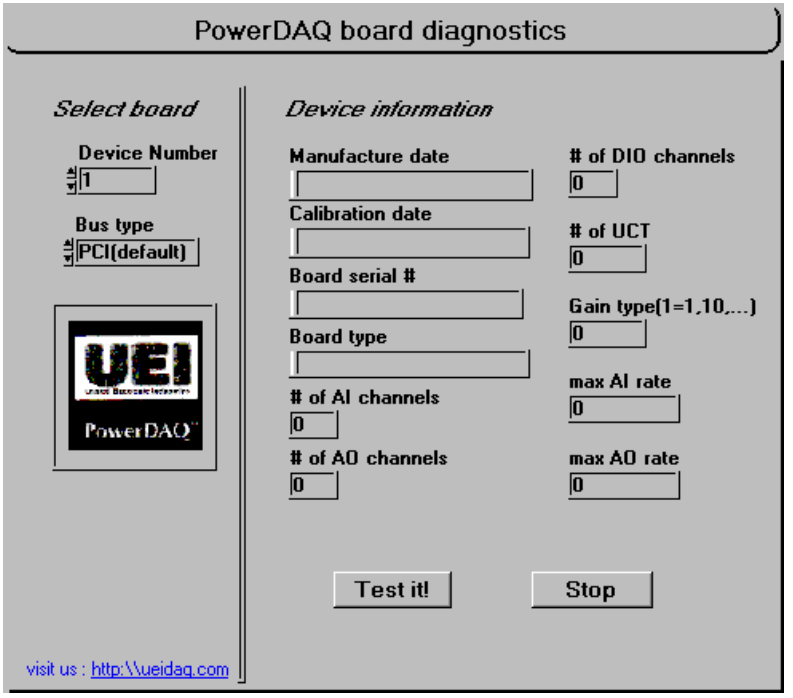
"PD AO32 Wave.VI" – for the PD2-AO series boards.

PD Diagnostic.VI

PowerDAQ board detection example VI.

This VI detects any type of PowerDAQ board installed into PC. *'Test it!'* button finds the PowerDAQ board through board *(device)* number and bus type specified and shows their parameters. If board is not present, error code is returned and dialog box will be displayed.

Front Panel



Controls and Indicators

- I32 Device Number see Common Controls topic for the description
- Bus type Bus type where PowerDAQ device installed
- TF Test it! Test board specified at Device Number and Bus Type fields
- TF Stop see Common Controls topic for the description
- TF Web-link button see Common Controls topic for the description
- abc Manufacture date Board manufacture date (from EEPROM)
- abc Calibration date Board calibration date (from EEPROM)
- abc Board serial # Board serial # (from EEPROM)
- abc Board type Type of PowerDAQ board installed
- I32 # of AI channels Number of analog input channels
- I32 # of AO channels Number of analog output channels
- I32 # of DIO channels Number of digital input-output channels

I32 # of UCT Number of user counter-timers

I32 Gain type(1=1,10,...)

Indicates type of board gains

3 means only gain 1 available

2 means 1,2,5,10 gains available

1 means 1,10,100,1000 gains available

0 means 1,2,4,8 gains supported by board

I32 max AI rate Maximum value of analog input rate that recommended by PowerDAQ technical documentation

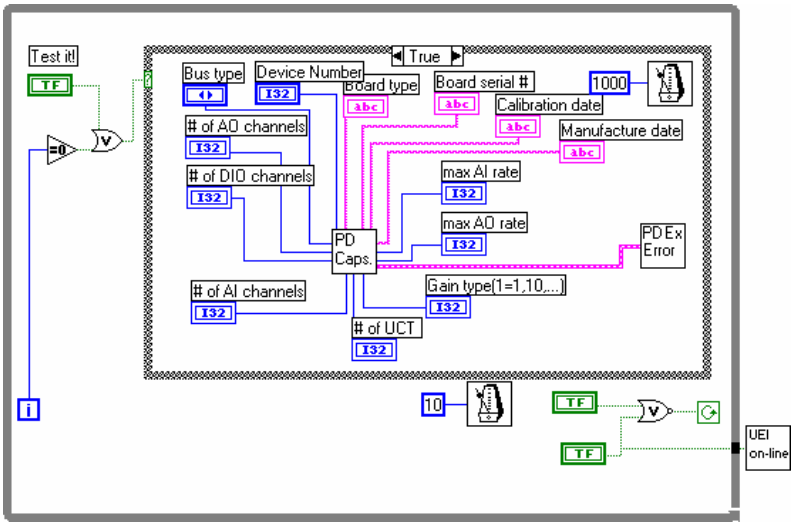
I32 max AO rate Maximum value of analog output rate that recommended by PowerDAQ technical documentation

See also

- PD Get Capabilities.VI
- PD DIO-128 Single Read And Write.VI

- PD AO32 Wave.vi
- PD AI All Subsystems.VI (Example VI)

Block Diagram



Analog Input Examples

The Analog Input Example VIs set includes:

- PD AI 16 channels thermocouple rack.VI
- PD AI Acquire & Display all channels.VI
- PD AI Acquire eight channels.VI
- PD AI Analog & Digital Triggering.VI
- PD AI Analog triggering.VI
- PD AI Async acquire eight channels.VI
- PD AI Async.VI
- PD AI Digital triggering.VI
- PD AI Single channel real-time display.VI
- PD AI Single scan example.VI
- PD AI Spectrum Analyzer.VI
- PD AI Stream data into LabVIEW array.VI
- PD AI Stream to disk.VI
- PD AI Wave Spectrum Analyzer.VI

All those VIs are located in **the <LabVIEW folder>\VI.LIB\PowerDAQ\Examples\AnalogIn** directory if you have PowerDAQ SDK installed on you PC.

Analog Input Examples Overview

The table below provides a brief description about each analog input example VI

	VI file name	Description
1	PD AI 16 channels thermocouple rack.VI	This VI associated with PowerDAQ thermocouple rack hardware which allows to monitor up to the 16 temperature sensors but can be used as an example for "**PD AI Wave.VI**"
2	PD AI Acquire & Display all channels.VI	Automatically detects number of analog input channels of the selected board, allows to acquire all of them and display any eight on Waveform Graph.
3	PD AI Acquire eight channels.VI	Eight channels example includes some advanced features such as variable frame size and backlog monitoring.
4	PD AI Analog & Digital Triggering.VI	This example shows how to use two trigger types to trigger acquisition on PowerDAQ board. Note that two different types of digital trigger available – hardware and a software. Analog trigger available only in software mode.
5	PD AI Analog triggering.VI	Advanced analog triggering example allows to change a lot of trigger parameters on the fly (slope/level/hysteresis, etc.). Note that one of the analog input channels should provide trigger signal. For even more complicated example when all other on-board subsystems involved to the acquisition process see "PD All Subsystems with Analog trigger.VI"
6	PD AI Async acquire eight channels.VI	Almost the same as "PD AI Acquire Eight channels.VI" but uses different way to receive data acquired.

		Asynchronous mode means that there are no wait for the data inside the driver – if data is not available control will be returned to the LabVIEW immediately and special error code set.
7	PD AI Async.VI	Almost the same as "PD AI Single Channel Real-Time Display.VI" but uses different way to receive data acquired. Asynchronous mode means that there are no wait for the data inside the driver – if data is not available control will be returned to the LabVIEW immediately and special error code set.
8	PD AI Digital triggering.VI	This example shows how to use digital trigger to trigger analog input acquisition on PowerDAQ board. Note that two different types of digital trigger available – hardware and a software.
9	PD AI Single channel real-time display.VI	High-speed single channel acquisition example.
10	PD AI Single scan example.VI	Simplest example – allows to acquire single scan of channel list provided. (Non-timed acquisition). Can be used in data log applications.
11	PD AI Spectrum Analyzer.VI	Continuous acquisition spectrum analyzer. Allows to show the data acquired and spectrum of this data. Performs different FFT modes on the fly.
12	PD AI Stream data into LabVIEW array.VI	Continuous streaming data into LabVIEW array without any data loss.
13	PD AI Stream to disk.VI	Simple stream-to-disk application. Use StreamReader application from PowerDAQ SDK to convert stream binary file format to comma-delimited spreadsheet format. Please refer to Advanced Topics to find out how to configure you PC hardware for best steam-to-disk performance.
14	PD AI Wave Spectrum	Extensive example of "**PD AI**

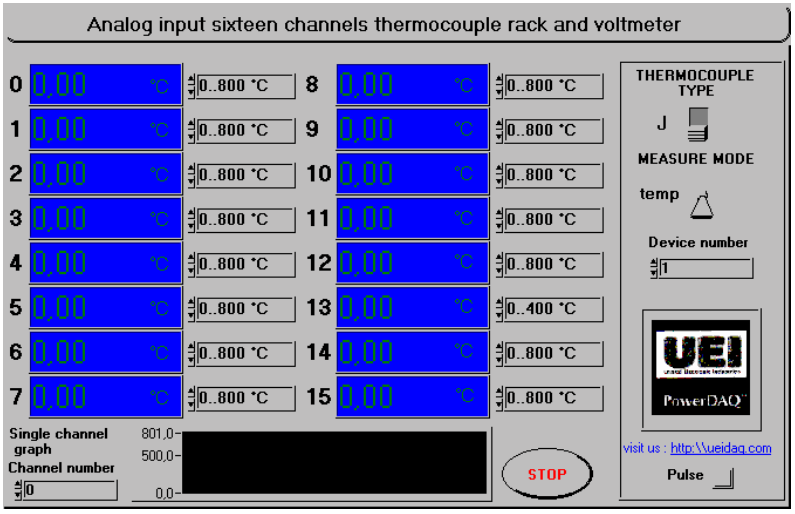
	Analyzer.VI	**Wave.VI**" usage. Allows to change acquisition parameters on the fly because of start-acquire buffer-stop nature of this example.

PD AI 16 channels thermocouple rack.VI

This VI acquires sixteen channels (from 0 to 15) and displays results in temperature or Volts. The acquisition runs in start-acquire-stop mode. Both measurement modes: Celsius and Fahrenheit degrees available at the same time. For better precision you can set a different temperature range for the each channel displayed. Choose any one of the first eight channels to display on temperature graph. In this example the heart of acquisition - "PD AI Wave.VI" used on the fixed acquisition rate (20K Samples/board). The "PD AI Thermocouple Control" is used to create a channel list entries using the best input range/gain available on the board installed and selected in device number control. Similarly the "PD AI Thermocouple Display.VI" used to display data in mode selected (Celsius, Fahrenheit degrees or Volts).

Note That PD-TCR hardware required do display proper temperature values.

Front Panel



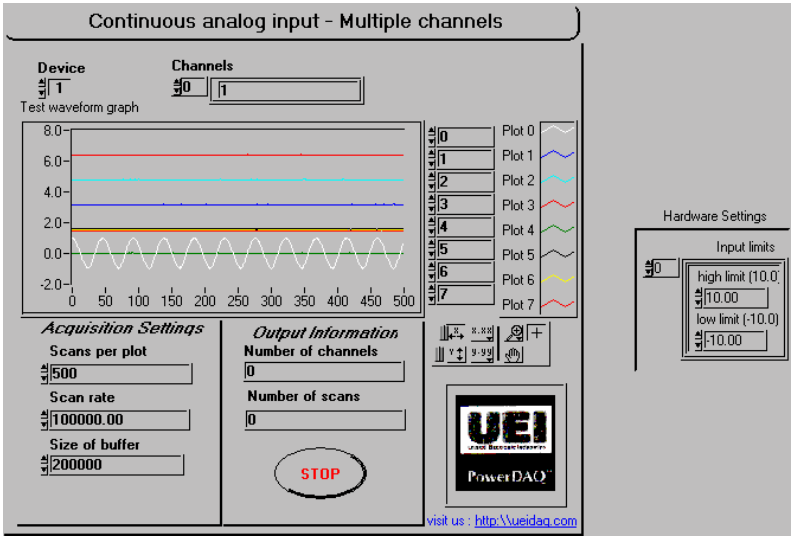
PD AI Acquire & Display all channels.VI

Acquires all analog input channels of the PowerDAQ board and displays any eight of these channels in a graph. Rate, buffer size and number of points per plot can be changed.

This VI provides a simple example for continuos acquisition when the most of acquisition settings are set to they defaults. The output array rebuild to provide required set of input channels to be displayed. Note that this operation required a lot of processor resources.

Note That buffer size should be at least 1.5*rate for rates between 10K and 200K and 2 to 3*rate for high frequencies. Increasing the buffer increases the stability and reliability of acquisition at high frequencies. Use hardware settings for changing default values of input limits (and thus gains).

Front Panel



Controls and Indicators

I16 Device see Common Controls topic for the description

I32 Scans per plot Number of samples that will be showed at graph during the acquisition (and requested from "PD AI Read.VI". There is no sense to show more then 1-3K scans because graphical operation takes too much processor time under the LabVIEW.

Note That PowerDAQ board works with frames of data and if frame size is greater than amount of data requested each "PD AI Read.VI" call the rest of the frame data will be lost. This is okay for most of applications such as FFT analysis and scope-like applications but for critical applications you can request exact amount of data which equal to the frame size from the "PD AI Read.VI". See *Optimizing performance using the buffer settings* topic for the details.

SGL Scan rate Frequency of acquisition. If you acquiring more than 1 channel actual frequency for each channel should be calculated as

(Scan rate)/(Number of active channels)

Note That actual scan rate can be found using the "PD AI Start.VI" and actual number of channels in channel list is returned from "PD AI Config.VI" call.

Size of buffer set size of buffer (in bytes) for storing data from board - 200000 default. Use the following rule: buffer size should be at least 1.5*rate for rates between 10K and 200K and 2-3*rate for high frequencies. Increasing the buffer increase the stability and reliability of acquisition system at the high frequencies.

stop Stops the acquisition after the next read


Input limits input limits is an array of clusters. You can use it to set board gain and input type (unipolar/bipolar) for the each channel. Please refer to "PD AI Config.VI" documentation for the details.

Channels see Common Controls topic for the description

Click to visit our web-site to download the latest version of SDK or up-to-date information about our products

Number of channels Number of active channels in channel list. Zero means error in channel list definition.

Number of scans Number of scans received from PowerDAQ LabView driver - show progress of acquisition.

Note That PowerDAQ board works with frames of data and if frame size is greater than amount of data requested each "PD AI Read.VI" call the rest of the frame data will be lost. This is okay for most of applications such as FFT analysis and scope-like applications but for critical applications you can request exact amount of data which equal to the frame size from the "PD AI Read.VI".

See *Optimizing performance using the buffer settings* topic for the details.

Test waveform graph Shows results of acquisition.

See also

- PD AI Config.vi
- PD AI Start.vi
- PD AI Read.vi
- PD AI All Channels with Analog Trigger.VI (Example VI)

PD AI Acquire eight channels.VI

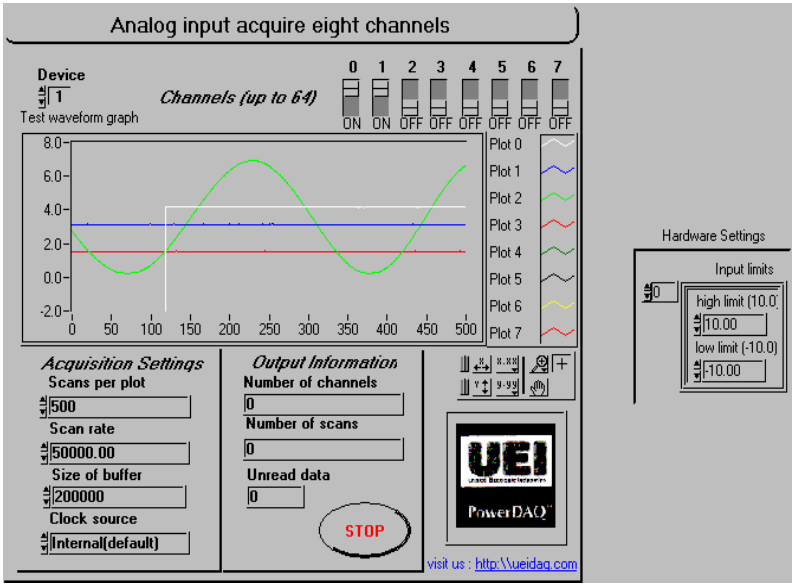
Eight channels example includes some advanced features such as variable frame size and backlog monitoring.

Acquires up to the eight analog input channels of the PowerDAQ board and displays results of the acquisition process on the graph. Rate, buffer size and number of points per plot can be changed. This VI acquires and read from the "**PD AI Read.VI**" all the data acquired. The "**PD AI Fine Tune.VI**" used to change default frame size and set some additional acquisition settings. Depends of your task you can set the values of *buffer overrun, gap-free mode* and *read ahead allowed* constants on the diagram. The *Unread data* indicator shows the amount of backlog data in the acquisition buffer. Changing *clock source* control you can change the CV clock source which clocks A/D conversion. Note that reducing amount of data displayed using *Scans per plot* control you can increase the aggregate performance of this example.

This example is a good start point for the data-critical applications development.

Note That buffer size should be at least 1.5*rate for rates between 10K and 200K and 2 to 3*rate for high frequencies. Increasing the buffer increases the stability and reliability of acquisition at high frequencies. Use hardware settings for changing default values of input limits (and thus gains).

Front Panel



Controls and Indicators

I16 Device see Common Controls topic for the description

TF 0 Channel switch. ON - specified channel present in channel list.

TF 1 Channel switch. ON - specified channel present in channel list.

TF 2 Channel switch. ON - specified channel present in channel list.

TF 3 Channel switch. ON - specified channel present in channel list.

TF 4 Channel switch. ON - specified channel present in channel list.

TF 5 Channel switch. ON - specified channel present in channel list.

TF 6 Channel switch. ON - specified channel present in channel list.

TF 7 Channel switch. ON - specified channel present in channel list.

I32 Scans per plot Number of samples that will be showed at graph during the acquisition (note that it might be different than requested from "PD AI Read.VI"). There is no sense to show more then 1-3K scans because graphical operation takes too much processor time under the LabVIEW.

Note That PowerDAQ board works with frames of data and if frame size is greater than amount of data requested each "PD AI Read.VI " call the rest of the frame data will be lost. This is okay for most of applications such as FFT analysis and scope-like

applications but for data-critical applications you can request exact amount of data which equal or even to the frame size from the "PD AI Read.VI ". You can either specify you own frame size like we did in this example using PD AI Fine Tune.VI" or find out current frame size using "PD AI Frame Size.VI". See *Optimizing performance using the buffer settings* topic for the details.

Scan rate Frequency of acquisition. If you acquiring more than 1 channel actual frequency for each channel should be calculated as

(Scan rate)/(Number of active channels)

Note That actual scan rate can be found using the "PD AI Start.VI" and actual number of channels in channel list is returned from "PD AI Config.VI" call.

Size of buffer set size of buffer (in bytes) for storing data from board - 200000 default. Use the following rule: buffer size should be at least 1.5*rate for rates between 10K and 200K and 2-3*rate for high frequencies. Increasing the buffer increase the stability and reliability of acquisition system at the high frequencies.

stop see Common Controls topic for the description

Input limits input limits is an array of clusters. You can use it to set board gain and input type (unipolar/bipolar) for the each channel. Please refer to "PD AI Config.VI" documentation for the details.

Click to visit our web-site to download the latest version of SDK or up-to-date information about our products

Clock source A/D clock source (CV Clock Source)

0 : software

1 : internal(default)

2 : external raising

3 : external falling

4 : continuos

Note For an external clock you should provide expected scan rate (5000..MaximumFrequency] or specify timeout when PD AI Read called

Please refer to the Conversion Clock and Channel List Clock topic in the manual for the details about CV and CL clock sources.

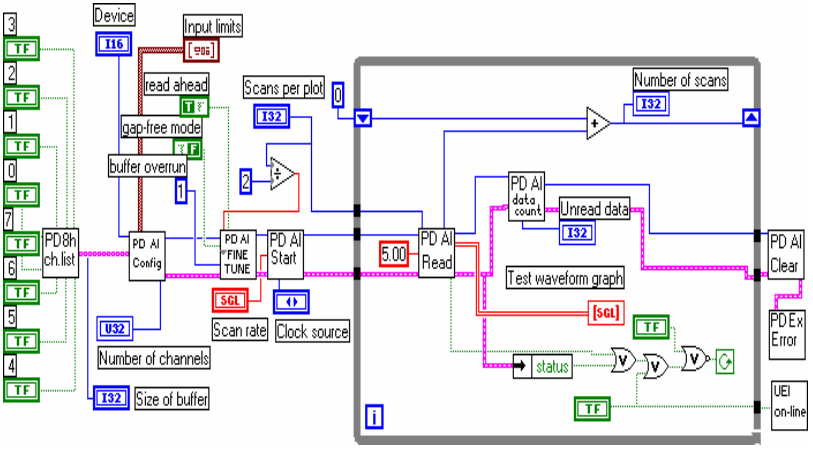
Number of channels Number of active channels in channel list. Zero means error in channel list definition.

Number of scans Number of scans received from PowerDAQ LabView driver - show progress of acquisition.

I32 **Unread data** uint32 - Amount of data in the acquisition buffer (in samples) that has not been read into LabVIEW.

Block Diagram

Note That "PD AI Fine Tune.VI" could be used to set an additional acquisition settings



See also

- PD AI Config.vi
- PD AI Start.vi
- PD AI Read.vi
- PD AI All Channels with Analog Trigger.VI (Example VI)

PD AI Analog & Digital Triggering.VI

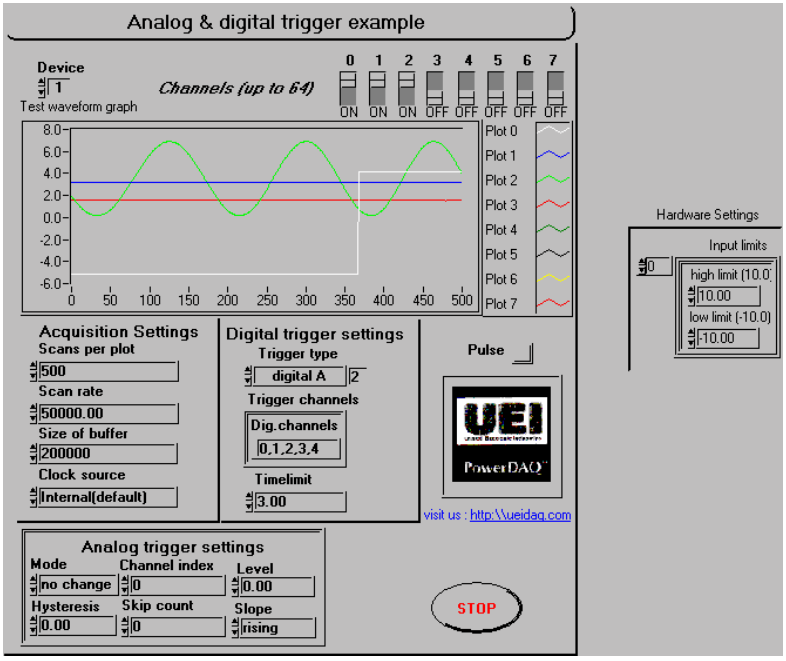
PowerDAQ analog input with simultaneously digital and analog trigger example.

Acquires up to the eight analog input channels of the PowerDAQ board and displays results of the acquisition process on the graph.

Any of the analog input channels can be used to trigger acquisition. The analog trigger settings can be changed on fly but digital trigger settings change will affect the results only after VI will be restarted. Rate, buffer size and number of points per plot can be changed. Changing *clock source* control you can change the CV clock source which clocks A/D conversion. Note that reducing amount of data displayed using *Scans per plot* control you can increase the aggregate performance of this example.

Note The trigger window is limited by frame size, PowerDAQ LabVIEW driver looking for the trigger conditions in the current and previous frame. If trigger conditions are not found it skips to the next frame until time-out specified will be expired. The buffer size should be at least 1.5*rate for rates between 10K and 200K and 2 to 3*rate for high frequencies. Increasing the buffer increases the stability and reliability of acquisition at high frequencies. Use hardware settings for changing default values of input limits (and thus gains).

Front Panel



Controls and Indicators

I16 Device see Common Controls topic for the description

TF 0 Channel switch. ON - specified channel present in channel list.

TF 1 Channel switch. ON - specified channel present in channel list.

TF 2 Channel switch. ON - specified channel present in channel list.

TF 3 Channel switch. ON - specified channel present in channel list.

TF 4 Channel switch. ON - specified channel present in channel list.

TF 5 Channel switch. ON - specified channel present in channel list.

TF 6 Channel switch. ON - specified channel present in channel list.

TF 7 Channel switch. ON - specified channel present in channel list.

I32 Scans per plot Number of samples that will be showed at graph during the acquisition (note that it might be different than requested from "PD AI Read.VI"). There is no sense to show more then 1-3K scans because graphical operation takes too much processor time under the LabVIEW.

Note That PowerDAQ board works with frames of data and

if frame size is greater than amount of data requested each "PD AI Read.VI " call the rest of the frame data will be lost. This is okay for most of applications such as FFT analysis and scope-like applications but for data-critical applications you can request exact amount of data which equal or even to the frame size from the "PD AI Read.VI ". You can either specify you own frame using PD AI Fine Tune.VI" or find out current frame size using "PD AI Frame Size.VI". See *Optimizing performance using the buffer settings* topic for the details.

Scan rate Frequency of acquisition. If you acquiring more than 1 channel actual frequency for each channel should be calculated as

(Scan rate)/(Number of active channels)

Note That actual scan rate can be found using the "PD AI Start.VI" and actual number of channels in channel list is returned from "PD AI Config.VI" call.

Size of buffer set size of buffer (in bytes) for storing data from board - 200000 default. Use the following rule: buffer size should be at least 1.5*rate for rates between 10K and 200K and 2-3*rate for high frequencies. Increasing the buffer increase the stability and reliability of acquisition system at the high frequencies.

stop see Common Controls topic for the description

Input limits input limits is an array of clusters. You can use it to set board gain and input type (unipolar/bipolar) for the each channel. Please refer to "PD AI Config.VI" documentation for the details.

Web-site see Common Controls topic for the description

Clock source A/D clock source (CV Clock Source)

0 : software

1 : internal(default)

2 : external raising

3 : external falling

4 : continuos

Note For an external clock you should provide expected scan rate (5000..MaximumFrequency] or specify timeout when PD AI Read called

Please refer to the Conversion Clock and Channel List Clock topic in the manual for the details about CV and CL clock sources.

Trigger type trigger type

0 - no triggering (default input).

1 - analog trigger (default setting) - one of the analog channels in channel list used for triggering.

2 - digital trigger A - for PowerDAQ boards - digital trigger on rising edge.

3 - digital trigger A and B - for PowerDAQ boards - digital trigger on falling edge.

4 - external high-speed hardware trigger (see usage notes below).

Note For trigger type 1 one analog channel that exists in channel list could be specified in trigger channel control (0 - default channel). For trigger type 2 and 3 any numbers of digital channels could be specified.

For the trigger type 4 the following values in the first entry in the channel list could be used:

0 - do not use external trigger line to start acquisition.

1 - use rising edge of the external trigger line to start acquisition.

2 - use falling edge of the external trigger line to start acquisition.

3 - reserved

For the trigger type 4 the follows values in the second entry in the channel list could be used:

0 - do not use external trigger line to stop acquisition.

1 - use rising edge of the external trigger line to stop acquisition.

2 - use falling edge of the external trigger line to stop acquisition.

3 - reserved

Trigger channels trigger channel(s) and level

Analog level and analog or digital channel(s) for triggering should be specified via this cluster.

Dig.channels Specify trigger channel number(s)

For trigger type 1 one analog channel that exists in channel list could be specified in trigger channel control (0 - default channel). For trigger type 2 and 3 any numbers of digital channels could be specified. Triggering conditions should be in ANY of digital channel specified.

SGL **level** level is the Volts of the analog source must cross for a trigger to occur. You must also specify whether level must be crossed on a leading or trailing slope with the edge or slope input. The default input for level is 0.0.

SGL **Timelimit** time limit set the amount of time for PowerDAQ LabVIEW driver waits for the trigger to occur.

-1.0 -no change the time limit setting

0 - (default) no waiting.

The "time limit" control sets the windowing size to be analyzed in the data stream before control is returned to the other sections of the VI. If a trigger is not detected during this window, the trigger "Timeout" situation will flash and control will return to the other subsystems before returning to analyze another window of data. The minimum length of the trigger signal is the number of channels divided by the Scan rate. (If you don't acquire the trigger, you can't trigger on it.)

analog trigger conditions input cluster - used for set analog trigger conditions. See cluster members description for details

Mode input - trigger mode :

off - clear all triggers

on - add analog trigger conditions

no change - leave trigger configuration unchanged

I32 **Channel index** Specify trigger channel number(s)

For trigger type 1 analog channel existing in channel list could be specified in trigger channel control (0 - default channel). For trigger type 2 and 3 any numbers of digital channels could be specified.

Slope edge or slope.

0 - Do not change the default setting (default input).

1 - rising.

2 - falling.

SGL **Level** Level (measured in Volts) which analog source must cross for a trigger to occur. You must also specify whether level must be crossed on a leading or trailing slope with the edge or slope input. The default input for level is 0.0.

SGL **Hysteresis** The hysteresis of the signal in Volts. The default input and setting are 0.0. The hysteresis value refers to a limit above or below the actual trigger level, which will need to be surpassed before it is considered to be a valid trigger. This compensates for the possibility of a noise spike causing an accidental trigger condition to

be detected

Skip count skip count is the number of triggers the VI skips before triggering the acquisition.

-1 - no change the skip count setting

0 - default - no skip any triggers.

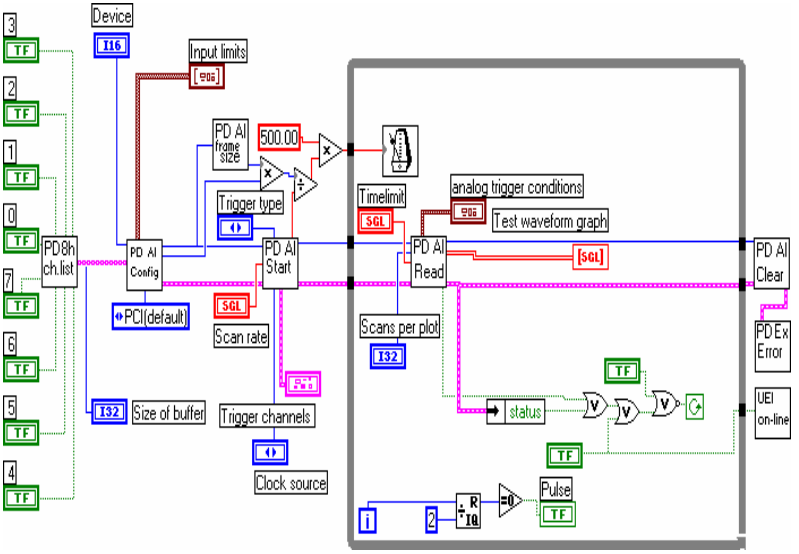
Offset offset in scans after trigger conditions retrieved - post trigger scan number

Test waveform graph Shows the results of acquisition.

Pulse Shows the 'live' pulse of acquisition

Note That "PD AI Frame Size.VI" could be find our the current frame size

Block Diagram



See also

- PD AI Config.vi
- PD AI Start.vi

- PD AI Read.vi
- PD AI All Channels with Analog Trigger.VI (Example VI)

PD AI Analog triggering.VI

PowerDAQ analog input continuous buffered acquisition with analog trigger example.

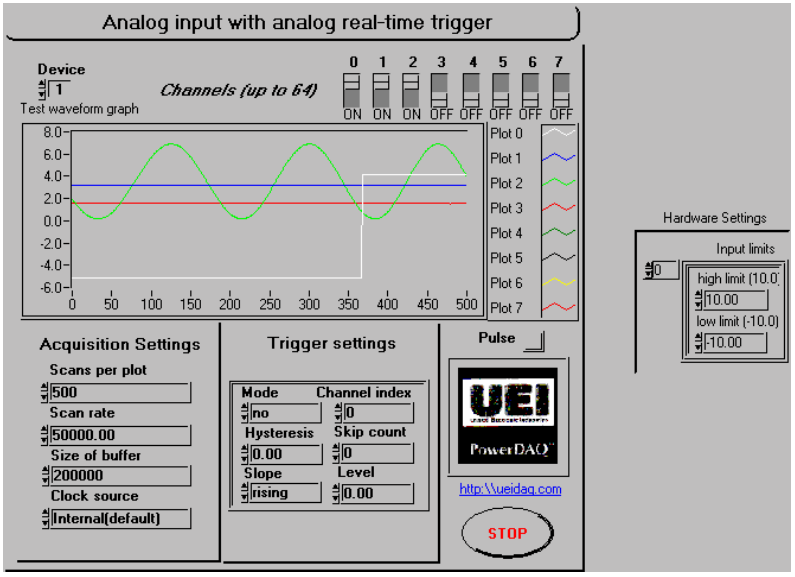
Acquires up to the eight analog input channels of the PowerDAQ board and displays results of the acquisition process on the graph.

Any of the analog input channels can be used to trigger acquisition. The analog trigger settings can be changed on fly. Rate, buffer size and number of points per plot can be changed. Changing *clock source* control you can change the CV clock source which clocks A/D conversion. Note that reducing amount of data displayed using *Scans per plot* control you can increase the aggregate performance of this example. The advanced trigger settings such as *hysteresis slope* and *level* allows to tune-up your trigger conditions.

This example is a good start point for the scope-like triggered applications. For more advanced triggering with different frame size and multiple subsystem usage see "PD All Subsystems with Analog Trigger.VI" example.

Note The trigger window is limited by frame size, PowerDAQ LabVIEW driver looking for the trigger conditions in the current and previous frame. If trigger conditions are not found it skips to the next frame until time-out specified will be expired. The buffer size should be at least 1.5*rate for rates between 10K and 200K and 2 to 3*rate for high frequencies. Increasing the buffer increases the stability and reliability of acquisition at high frequencies. Use hardware settings for changing default values of input limits (and thus gains).

Front Panel



Controls and Indicators

I16 Device see Common Controls topic for the description

TF 0 Channel switch. ON - specified channel present in channel list.

TF 1 Channel switch. ON - specified channel present in channel list.

TF 2 Channel switch. ON - specified channel present in channel list.

TF 3 Channel switch. ON - specified channel present in channel list.

TF 4 Channel switch. ON - specified channel present in channel list.

TF 5 Channel switch. ON - specified channel present in channel list.

TF 6 Channel switch. ON - specified channel present in channel list.

TF 7 Channel switch. ON - specified channel present in channel list.

I32 Scans per plot Number of samples that will be showed at graph during the acquisition (note that it might be different than requested from "PD AI Read.VI"). There is no sense to show more then 1-3K scans because graphical operation takes too much processor time under the LabVIEW.

Note That PowerDAQ board works with frames of data and if frame size is greater than amount of data requested each "PD AI Read.VI " call the rest of the frame data will be lost. This is okay for most of applications such as FFT analysis and scope-like

applications but for data-critical applications you can request exact amount of data which equal or even to the frame size from the "PD AI Read.VI ". You can either specify you own frame using PD AI Fine Tune.VI" or find out current frame size using "PD AI Frame Size.VI". See *Optimizing performance using the buffer settings* topic for the details.

Scan rate Frequency of acquisition. If you acquiring more than 1 channel actual frequency for each channel should be calculated as

(Scan rate)/(Number of active channels)

Note That actual scan rate can be found using the "PD AI Start.VI" and actual number of channels in channel list is returned from "PD AI Config.VI" call.

Size of buffer set size of buffer (in bytes) for storing data from board - 200000 default. Use the following rule: buffer size should be at least 1.5*rate for rates between 10K and 200K and 2-3*rate for high frequencies. Increasing the buffer increase the stability and reliability of acquisition system at the high frequencies.

stop see Common Controls topic for the description

Input limits input limits is an array of clusters. You can use it to set board gain and input type (unipolar/bipolar) for the each channel. Please refer to "PD AI Config.VI" documentation for the details.

Web-site link

see Common Controls topic for the description

Clock source A/D clock source (CV Clock Source)

0 : software

1 : internal(default)

2 : external raising

3 : external falling

4 : continuos

Note For an external clock you should provide expected scan rate (5000..MaximumFrequency] or specify timeout when PD AI Read called

Please refer to the Conversion Clock and Channel List Clock topic in the manual for the details about CV and CL clock sources.

Analog trigger conditions input cluster - used to set an analog trigger conditions. See cluster members description for details. Analog trigger settings allow synchronizing the data using one of the analog input channels (first channel in the channel list by default) as trigger.

Mode input - trigger mode :

off - clear all triggers

on - add analog trigger conditions

no change - leave trigger configuration unchanged

Channel index Specify trigger channel number

For analog trigger input channel existing in channel list could be specified in trigger channel control (0 - default channel).

Slope edge or slope.

0 - do not change the default setting (default input).

1 - rising.

2 - falling.

Level level (measured in Volts) which analog source must cross for a trigger to occur. You must also specify whether level must be crossed on a leading or trailing slope with the edge or slope input. The default input for level is 0.0.

Hysteresis The hysteresis of the signal in Volts. The default input and setting are 0.0. The hysteresis value refers to a limit above or below the actual trigger level, which will need to be surpassed before it is considered to be a valid trigger. This compensates for the possibility of a noise spike causing an accidental trigger condition to be detected

Skip count skip count is the number of triggers the VI skips before triggering the acquisition.

-1 - do no change the skip count setting

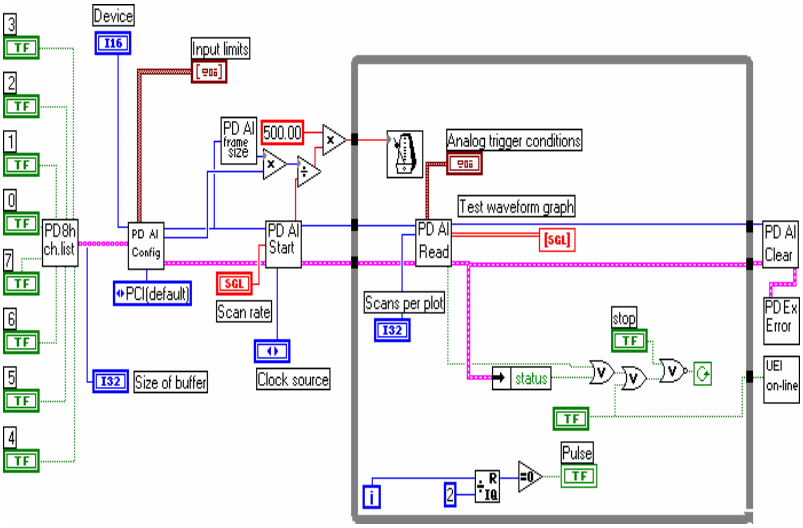
0 - default - do not skip any triggers.

Test waveform graph Shows the results of acquisition.

Pulse Shows the 'live' pulse of acquisition

Block Diagram

Note That "PD AI Frame Size.VI" could be find our the current frame size



See also

- PD AI Config.vi
- PD AI Start.vi
- PD AI Read.vi
- PD AI Analog & Digital Triggering.VI
- PD AI All Channels with Analog Trigger.VI (Example VI)

PD AI Async.VI

Single-channel asynchronous continuos buffered acquisition demo.

This analog input example highlights asynchronous way to acquire the analog input data. The main difference that if requested amount of data is not available yet PowerDAQ LabVIEW driver returns control to the LabVIEW immediately and arise dedicated error code (-10802 for instance) which shows that. So, the data should be processed only if zero error code returned. Normally "**PD AI Read.VI**" should be used to retrieve the data from the acquisition buffer and "

PD AI Read Async**.VI**" is dedicated for multi-board systems.

Acquires any one of the available on board installed and specified via *deviseId* analog input channel and displays results of the acquisition process on the graph. Rate, buffer size and number of points per plot can be changed. This VI acquires and read the data acquired from the "

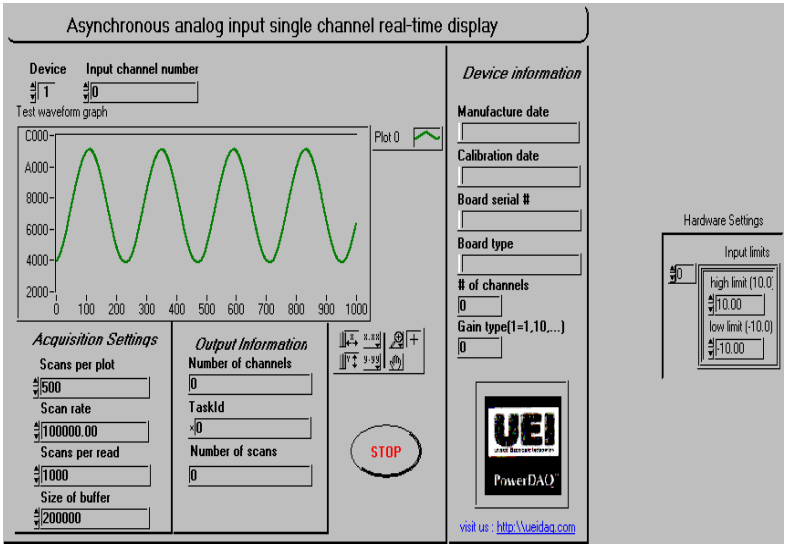
PD AI Read Async**.VI**". The "**PD AI Fine Tune.VI**" used to change default frame size set asynchronous mode and set some additional acquisition settings. Depends of your task you can set the values of *buffer overrun, gap-free mode* and *read ahead allowed* constants on the diagram. Note that there are two user-controlled parameters are control amount of data requested – the total amount of scans requested from the "

PD AI Read Async**.VI**" and amount of the data to be displayed.

Device information sheet at the right-hand side of this VI provides an addition information about selected board.

Note That buffer size should be at least 1.5*rate for rates between 10K and 200K and 2 to 3*rate for high frequencies. Increasing the buffer increases the stability and reliability of acquisition at high frequencies. Use hardware settings for changing default values of input limits (and thus gains).

Front Panel



Controls and Indicators

I16 Device see Common Controls topic for the description

I32 Scans per plot Number of samples that will be showed at graph during the acquisition. There is no sense to show more then 1-3K scans because graphical operation takes too much processor time under the LabVIEW.

SGL Scan rate Frequency of acquisition. If you acquiring more than 1 channel actual frequency for each channel should be calculated as

(Scan rate)/(Number of active channels)

Note That actual scan rate can be found using the "PD AI Start.VI" and actual number of channels in channel list is returned from "PD AI Config.VI" call.

I32 Scans per read Total number of scans that will be transferred from PowerDAQ LabView driver into this VI.

I32 Size of buffer set size of buffer (in bytes) for storing data from board - 200000 default. Use the following rule: buffer size should be at least 1.5*rate for rates between 10K and 200K and 2-3*rate for high frequencies. Increasing the buffer increase the stability and reliability of acquisition system at the high frequencies.


stop see Common Controls topic for the description


Input limits input limits is an array of clusters. You can use it to set board gain and input type (unipolar/bipolar) for the each channel. Please refer to "PD AI Config.VI" documentation for the details.


Input channel number (0..Number of AI channels-1)


Web-site link

see Common Controls topic for the description


Test waveform graph Show the results of acquisition


Number of channels Number of active channels in channel list. Zero means error in channel list definition.


TaskId Internal task ID associated with acquisition process


Number of scans Number of scans received from PowerDAQ LabView driver - show progress of acquisition.

Note That PowerDAQ board works with frames of data and if frame size is greater than amount of data requested each "

PD AI Read Async.VI" call the rest of the frame data will be lost. This is okay for most of applications such as FFT analysis and scope-like applications but for critical applications you can request exact amount of data which equal to the frame size from the "

PD AI Read Async.VI".

See *Optimizing performance using the buffer settings* topic for the details.


Manufacture date Board manufacture date (from EEPROM)


Calibration date Board calibration date (from EEPROM)


Board serial # Board serial # (from EEPROM)


Board type Type of the PowerDAQ board installed


# of channels Number of analog input channels


Gain type(1=1,10,...) Indicates type of board gains

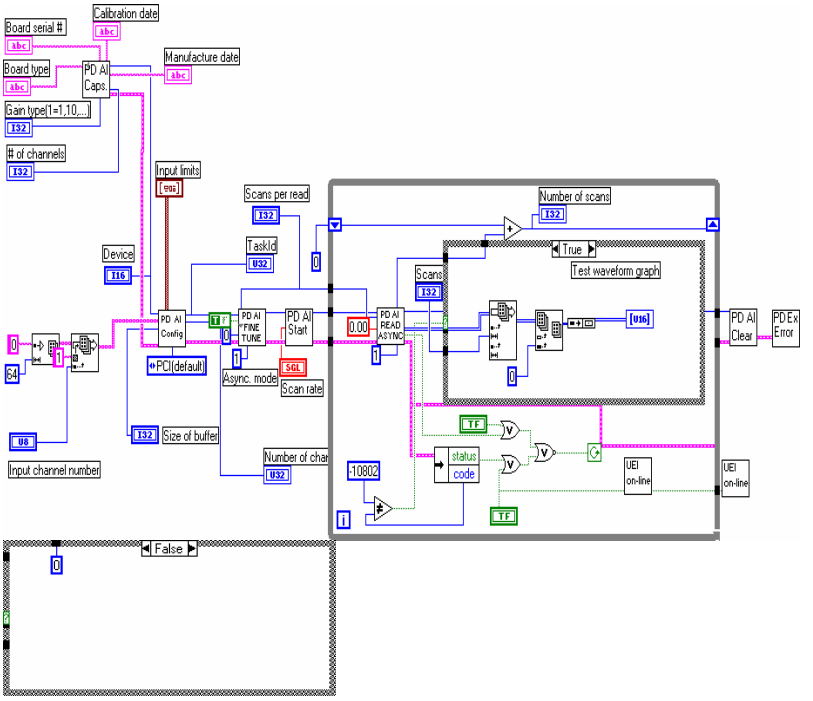
3 means only gain 1 available

2 means 1,2,5,10 gains available

1 means 1,10,100,1000 gains available

0 means 1,2,4,8 gains supported by board

Block Diagram



See also

- PD AI Config.vi
- PD AI Start.vi

- PD AI Read Async.vi
- PD AI Async Acquire Eight Channels.VI (Example VI)

PD AI Async Acquire Eight Channels.VI

PowerDAQ Multiple-channel asynchronous continuos buffered acquisition demo with advanced triggering.

This analog input example highlights asynchronous way to acquire the analog input data. The main difference that if requested amount of data is not available yet PowerDAQ LabVIEW driver returns control to the LabVIEW immediately and arise dedicated error code (-10802 for instance) which shows that. So, the data should be processed only if zero error code returned. Normally "**PD AI Read.VI**" should be used to retrieve the data from the acquisition buffer and "

PD AI Read Async**.VI**" is dedicated for multi-board systems.

Acquires up to the eight of the available on board analog input channels and displays the results on the graph. Rate, buffer size and number of points per plot can be changed. This VI acquires and read the data acquired from the "

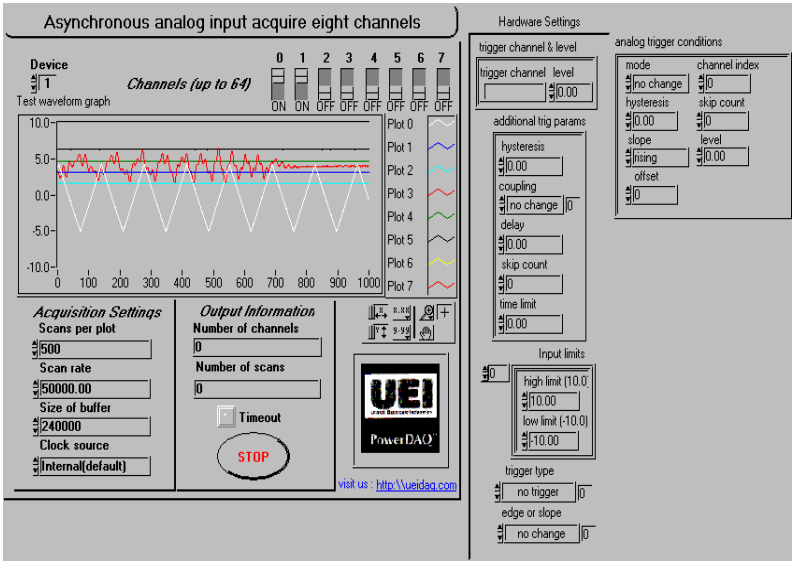
PD AI Read Async**.VI**". The "**PD AI Fine Tune.VI**" used to change default frame size set asynchronous mode and set some additional acquisition settings. Depends of your task you can set the values of *buffer overrun, gap-free mode* and *read ahead allowed* constants on the diagram. Note that there are two user-controlled parameters are control amount of data requested – the total amount of scans requested from the "

PD AI Read Async**.VI**" and amount of the data to be displayed.

The extensive triggering features were added to provide an ability to test each and every analog input trigger modes available on PowerDAQ board

Note That buffer size should be at least 1.5*rate for rates between 10K and 200K and 2 to 3*rate for high frequencies. Increasing the buffer increases the stability and reliability of acquisition at high frequencies. Use hardware settings for changing default values of input limits (and thus gains).

Front Panel



Controls and Indicators

I16 **Device** see Common Controls topic for the description

TF 0 Channel switch. ON - specified channel present in channel list.

TF 1 Channel switch. ON - specified channel present in channel list.

TF 2 Channel switch. ON - specified channel present in channel list.

TF 3 Channel switch. ON - specified channel present in channel list.

TF 4 Channel switch. ON - specified channel present in channel list.

TF 5 Channel switch. ON - specified channel present in channel list.

TF 6 Channel switch. ON - specified channel present in channel list.

TF 7 Channel switch. ON - specified channel present in channel list.

I32 **Scans per plot** Number of samples that will be showed at graph during the acquisition. There is no sense to show more then 1-3K scans because graphical operation takes too much processor time under the LabVIEW.

SGL **Scan rate** Frequency of acquisition. If you acquiring more than 1 channel actual frequency for each channel should be calculated as

(Scan rate)/(Number of active channels)

Note That actual scan rate can be found using the "PD AI

Start.VI" and actual number of channels in channel list is returned from "PD AI Config.VI" call.

Size of buffer set size of buffer (in bytes) for storing data from board - 200000 default. Use the following rule: buffer size should be at least 1.5*rate for rates between 10K and 200K and 2-3*rate for high frequencies. Increasing the buffer increase the stability and reliability of acquisition system at the high frequencies.

stop Stops the acquisition after next read

Input limits input limits is an array of clusters. You can use it to set board gain and input type (unipolar/bipolar) for the each channel. Please refer to "PD AI Config.VI" documentation for the details.

Web-site link

see Common Controls topic for the description

Clock source A/D clock source (CV Clock Source)

0 : software

1 : internal(default)

2 : external raising

3 : external falling

4 : continuos

Trigger channels trigger channel(s) and level

Analog level and analog or digital channel(s) for triggering should be specified via this cluster.

Dig.channels Specify trigger channel number(s)

For trigger type 1 one analog channel that exists in channel list could be specified in trigger channel control (0 - default channel). For trigger type 2 and 3 any numbers of digital channels could be specified. Triggering conditions should be in ANY of digital channel specified.

level level is the Volts of the analog source must cross for a trigger to occur. You must also specify whether level must be crossed on a leading or trailing slope with the edge or slope input. The default input for level is 0.0.

additional trig params input cluster – see members description for the details

hysteresis The hysteresis of the signal in Volts. The default input and setting are 0.0. The hysteresis value refers to a limit above or below the actual trigger level, which will need to be surpassed before it is considered to be a valid trigger. This compensates for the possibility of a noise spike causing an accidental trigger condition to

be detected

coupling coupling of signal

0: Do not change the trigger coupling setting (default input).

1: DC.

2: AC (reserved)

SGL delay pretriggering delay in seconds. The default input and setting are 0.0 seconds.

I32 skip count skip count is the number of triggers the VI skips before triggering the acquisition.

-1 - no change the skip count setting

0 - default - no skip any triggers.

SGL time limit time limit set the amount of time for PowerDAQ LabVIEW driver waits for the trigger to occur.

-1.0 -no change the time limit setting

0 - (default) no waiting.

The "time limit" control sets the windowing size to be analyzed in the data stream before control is returned to the other sections of the VI. If a trigger is not detected during this window, the trigger "Timeout" situation will flash and control will return to the other subsystems before returning to analyze another window of data. The minimum length of the trigger signal is the number of channels divided by the Scan rate. (If you don't acquire the trigger, you can't trigger on it.)

trigger type trigger type

0 - no triggering (default input).

1 - analog trigger (default setting) - one of the analog channels in channel list used for triggering.

2 - digital trigger A - for PowerDAQ boards - digital trigger on rising edge.

3 - digital trigger A and B - for PowerDAQ boards - digital trigger on falling edge.

4 - external high-speed hardware trigger (see usage notes below).

Note For trigger type 1 one analog channel that exists in channel list could be specified in trigger channel control (0 - default channel). For trigger type 2 and 3 any numbers of digital channels could be specified.

For the trigger type 4 the following values in the first entry in

the channel list could be used:

0 - do not use external trigger line to start acquisition.

1 - use rising edge of the external trigger line to start acquisition.

2 - use falling edge of the external trigger line to start acquisition.

3 - reserved

For the trigger type 4 the follows values in the second entry in the channel list could be used:

0 - do not use external trigger line to stop acquisition.

1 - use rising edge of the external trigger line to stop acquisition.

2 - use falling edge of the external trigger line to stop acquisition.

3 - reserved


edge or slope edge or slope.

0 - Do not change the default setting (default input).

1 - rising.

2 - falling.


analog trigger conditions input cluster - used for set analog trigger conditions. See cluster members description for details


Mode input - trigger mode :

off - clear all triggers

on - add analog trigger conditions

no change - leave trigger configuration unchanged


Channel index Specify trigger channel number(s)

For trigger type 1 analog channel existing in channel list could be specified in trigger channel control (0 - default channel). For trigger type 2 and 3 any numbers of digital channels could be specified.


Slope edge or slope.

0 - Do not change the default setting (default input).

1 - rising.

2 - falling.


Level Level (measured in Volts) which analog source must cross for a trigger to occur. You must also specify whether level must be crossed on a leading or trailing

slope with the edge or slope input. The default input for level is 0.0.

SGL **Hysteresis** The hysteresis of the signal in Volts. The default input and setting are 0.0. The hysteresis value refers to a limit above or below the actual trigger level, which will need to be surpassed before it is considered to be a valid trigger. This compensates for the possibility of a noise spike causing an accidental trigger condition to be detected

I32 **Skip count** skip count is the number of triggers the VI skips before triggering the acquisition.

-1 - no change the skip count setting

0 - default - no skip any triggers.

I32 **Offset** offset in scans after trigger conditions retrieved - post trigger scan number

U32 **Number of channels** Number of active channels in channel list. Zero means error in channel list definition.

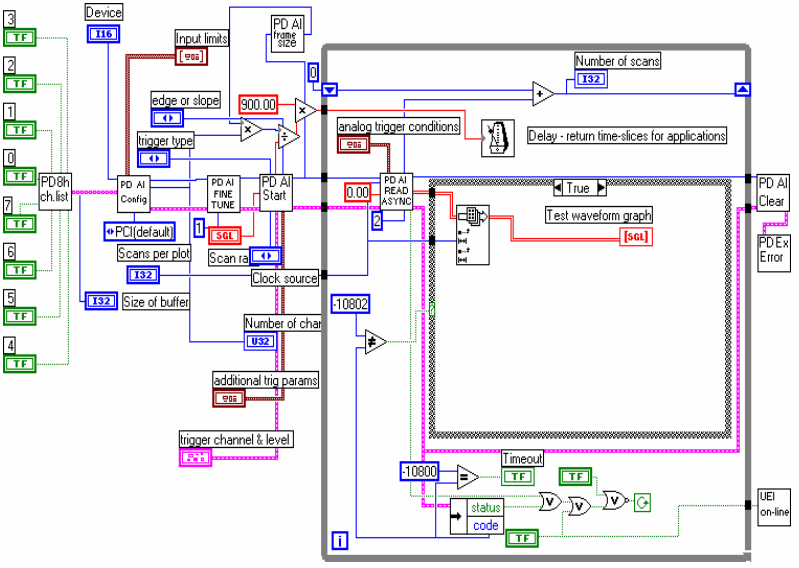
I32 **Number of scans** Number of scans received from PowerDAQ LabView driver - shows progress of acquisition.

[SGL] **Test waveform graph** Shows the results of acquisition

TF **Timeout** Indicator. ON if trigger timeout was detected

Block Diagram

Note That only trigger parameters which passed to the "PD AI Read Async.VI" can be changed on the fly. For all other changes VI has to be restarted.



See also

- PD AI Config.vi
- PD AI Start.vi

- PD AI Read Async.vi
- PD AI Async.VI (Example VI)

PD AI Digital triggering.VI

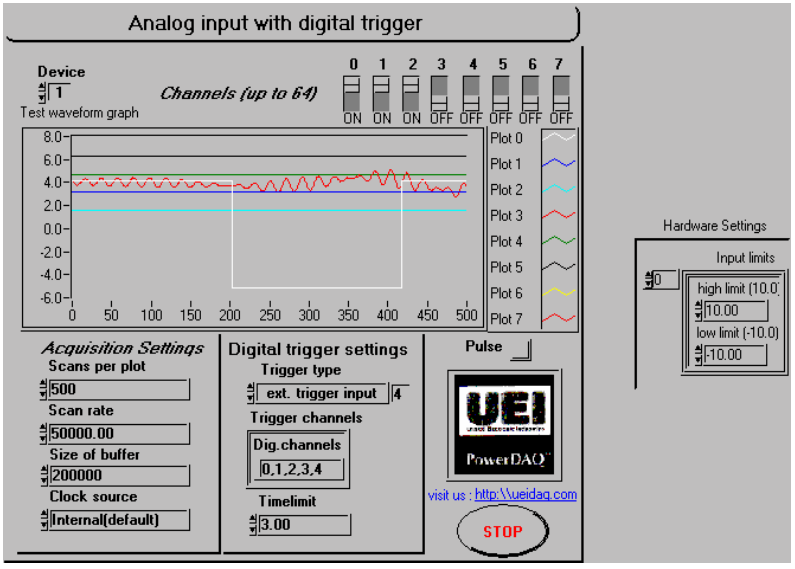
PowerDAQ analog input with digital trigger example.

Acquires up to the eight analog input channels of the PowerDAQ board and displays results of the acquisition process on the graph.

There are a two different types of digital trigger available on PowerDAQ board – hardware digital trigger when dedicated input trigger line used to trigger (start/stop) acquisition and software digital I/O-based trigger. For both of them proper de-bouncing circuitry should be used to avoid re-triggering. The digital trigger settings change will affect the results only after VI will be restarted. Rate, buffer size and number of points per plot can be changed. Changing clock source control you can change the CV clock source which clocks A/D conversion. Note that reducing amount of data displayed using *Scans per plot* control you can increase the aggregate performance of this example.

Note That buffer size should be at least 1.5*rate for rates between 10K and 200K and 2 to 3*rate for high frequencies. Increasing the buffer increases the stability and reliability of acquisition at high frequencies. Use hardware settings for changing default values of input limits (and thus gains).

Front Panel



Controls and Indicators

I16 Device see Common Controls topic for the description

TF 0 Channel switch. ON - specified channel present in channel list.

TF 1 Channel switch. ON - specified channel present in channel list.

TF 2 Channel switch. ON - specified channel present in channel list.

TF 3 Channel switch. ON - specified channel present in channel list.

TF 4 Channel switch. ON - specified channel present in channel list.

TF 5 Channel switch. ON - specified channel present in channel list.

TF 6 Channel switch. ON - specified channel present in channel list.

TF 7 Channel switch. ON - specified channel present in channel list.

I32 Scans per plot Number of samples that will be showed at graph during the acquisition (note that it might be different than requested from "PD AI Read.VI"). There is no sense to show more then 1-3K scans because graphical operation takes too much processor time under the LabVIEW.

Note That PowerDAQ board works with frames of data and if frame size is greater than amount of data requested each "PD AI Read.VI " call the rest of the frame data will be lost. This is okay for most of applications such as FFT analysis and scope-like applications but for data-critical applications you can request exact amount of data which equal or even to the frame size from the "PD AI Read.VI ". You can either specify you own frame using PD AI Fine Tune.VI" or find out current frame size using "PD AI Frame Size.VI". See *Optimizing performance using the buffer settings* topic for the details.

SGL Scan rate Frequency of acquisition. If you acquiring more than 1 channel actual frequency for each channel should be calculated as

(Scan rate)/(Number of active channels)

Note That actual scan rate can be found using the "PD AI Start.VI" and actual number of channels in channel list is returned from "PD AI Config.VI" call.

I32 Size of buffer set size of buffer (in bytes) for storing data from board - 200000 default. Use the following rule: buffer size should be at least 1.5*rate for rates between 10K and 200K and 2-3*rate for high frequencies. Increasing the buffer increase the stability and reliability of acquisition system at the high frequencies.

TF stop Stops the acquisition after next read

Input limits input limits is an array of clusters. You can use it to set board gain and input type (unipolar/bipolar) for the each

channel. Please refer to "PD AI Config.VI" documentation for the details.

Web-site link

see Common Controls topic for the description

Clock source A/D clock source (CV Clock Source)

0 : software

1 : internal(default)

2 : external raising

3 : external falling

4 : continuos

Note For an external clock you should provide expected scan rate (5000..MaximumFrequency] or specify timeout when "PD AI Read.VI" called

Please refer to the Conversion Clock and Channel List Clock topic in the manual for the details about CV and CL clock sources.

Trigger type trigger type

0 - no triggering (default input).

1 - analog trigger (default setting) - one of the analog channels in channel list used for triggering.

2 - digital trigger A - for PowerDAQ boards - digital trigger on rising edge.

3 - digital trigger A and B - for PowerDAQ boards - digital trigger on falling edge.

4 - external high-speed hardware trigger (see usage notes below).

Note For trigger type 1 one analog channel that exists in channel list could be specified in trigger channel control (0 - default channel). For trigger type 2 and 3 any numbers of digital channels could be specified.

For the trigger type 4 the following values in the first entry in the channel list could be used:

0 - do not use external trigger line to start acquisition.

1 - use rising edge of the external trigger line to start acquisition.

2 - use falling edge of the external trigger line to start acquisition.

3 - reserved

For the trigger type 4 the follows values in the second entry in the channel list could be used:

0 - do not use external trigger line to stop acquisition.

1 - use rising edge of the external trigger line to stop acquisition.

2 - use falling edge of the external trigger line to stop acquisition.

3 - reserved

Trigger channels trigger channel(s) and level

Analog level and analog or digital channel(s) for triggering should be specified via this cluster.

Dig.channels Specify trigger channel number(s)

For trigger type 1 one analog channel that exists in channel list could be specified in trigger channel control (0 - default channel). For trigger type 2 and 3 any numbers of digital channels could be specified. Triggering conditions should be in ANY of digital channel specified.

level level is the Volts of the analog source must cross for a trigger to occur. You must also specify whether level must be crossed on a leading or trailing slope with the edge or slope input. The default input for level is 0.0.

Timelimit time limit set the amount of time for PowerDAQ LabVIEW driver waits for the trigger to occur.

-1.0 -no change the time limit setting

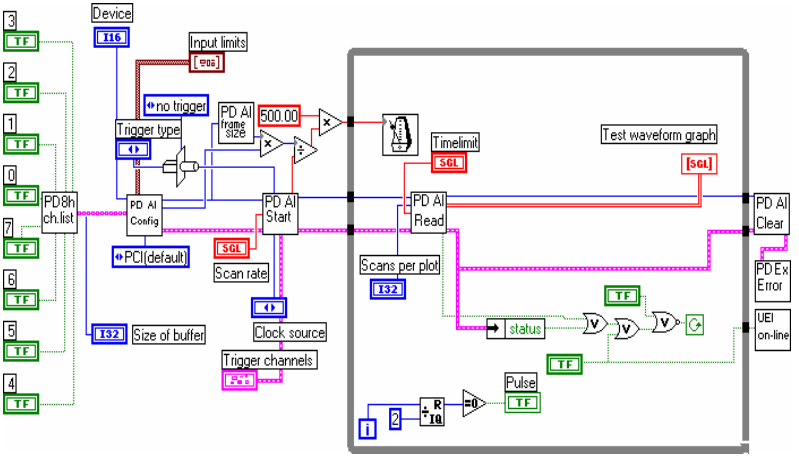
0 - (default) no waiting.

The "time limit" control sets the windowing size to be analyzed in the data stream before control is returned to the other sections of the VI. If a trigger is not detected during this window, the trigger "Timeout" situation will flash and control will return to the other subsystems.

Test waveform graph Shows the results of acquisition.

Pulse Shows the 'live' pulse of acquisition

Block Diagram



See also

- PD AI Config.vi
- PD AI Start.vi
- PD AI Read.vi
- PD AI All Channels with Analog Trigger.VI (Example VI)

PD AI Single channel real-time display.VI

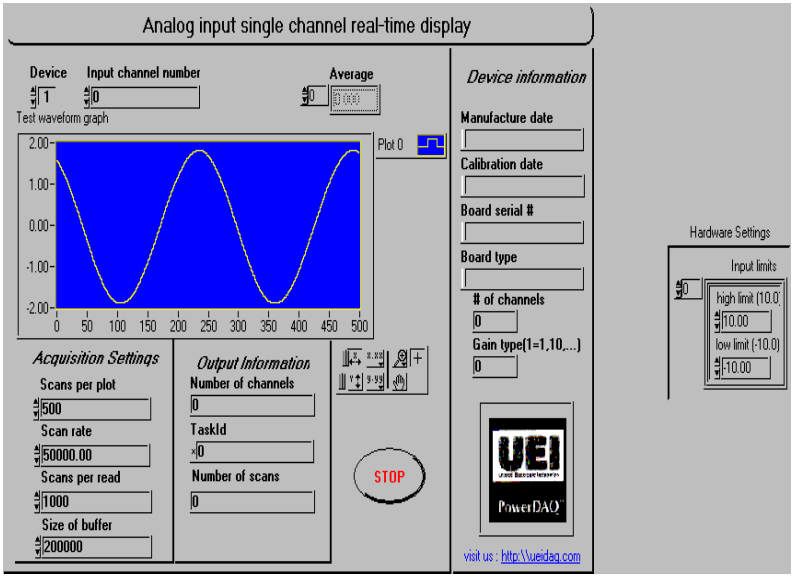
Single-channel real-time continuos buffered acquisition demo.

Acquires any one of the available on board installed and specified via *deviseId* analog input channel and displays results of the acquisition process on the graph. Rate, buffer size and number of points per plot can be changed. This VI acquires and read the data acquired from the "**PD AI Read.VI**". Note that there are two user-controlled parameters are control amount of data requested – the total amount of scans requested from the "**PD AI Read.VI**" and amount of the data to be displayed.

Device information sheet at the right-hand side of this VI provides an addition information about selected board.

Note That buffer size should be at least 1.5*rate for rates between 10K and 200K and 2 to 3*rate for high frequencies. Increasing the buffer increases the stability and reliability of acquisition at high frequencies. Use hardware settings for changing default values of input limits (and thus gains).

Front Panel



Controls and Indicators

I16 Device see Common Controls topic for the description

I32 Scans per plot Number of samples that will be showed at graph during the acquisition. There is no sense to show more then 1-3K scans because graphical operation takes too much processor time under the LabVIEW.

SGL Scan rate Frequency of acquisition. If you acquiring more than 1 channel actual frequency for each channel should be calculated as

(Scan rate)/(Number of active channels)

Note That actual scan rate can be found using the "PD AI Start.VI" and actual number of channels in channel list is returned from "PD AI Config.VI" call.

Scans per read Total number of scans that will be transferred from PowerDAQ LabView driver into this VI.

Size of buffer set size of buffer (in bytes) for storing data from board - 200000 default. Use the following rule: buffer size should be at least 1.5*rate for rates between 10K and 200K and 2-3*rate for high frequencies. Increasing the buffer increase the stability and reliability of acquisition system at the high frequencies.

stop Stops the acquisition after next read.

Input limits input limits is an array of clusters. You can use it to set board gain and input type (unipolar/bipolar) for the each channel. Please refer to “PD AI Config.VI” documentation for the details.

Input channel number (0..Number of AI channels-1)

Web-site link

see Common Controls topic for the description

Test waveform graph Show the results of acquisition

Number of channels Number of active channels in channel list. Zero means error in channel list definition.

TaskId Internal task ID associated with acquisition process

Number of scans Number of scans received from PowerDAQ LabView driver - show progress of acquisition.

Note That PowerDAQ board works with frames of data and if frame size is greater than amount of data requested each “PD AI Read.VI” call the rest of the frame data will be lost. This is okay for most of applications such as FFT analysis and scope-like applications but for critical applications you can request exact amount of data which equal to the frame size from the “PD AI Read.VI”.

See *Optimizing performance using the buffer settings* topic for the details.

Manufacture date Board manufacture date (from EEPROM)

Calibration date Board calibration date (from EEPROM)

Board serial # Board serial # (from EEPROM)

Board type Type of the PowerDAQ board installed

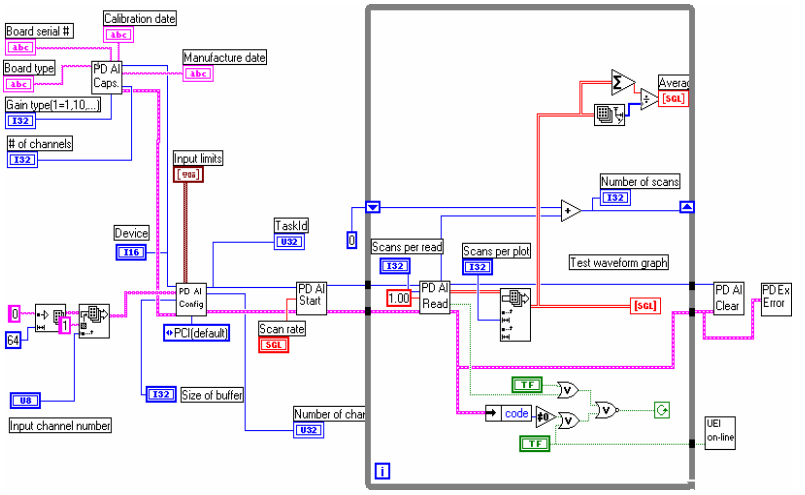
# of channels Number of analog input channels

Gain type(1=1,10,...) Indicates type of board gains

3 means only gain 1 available
2 means 1,2,5,10 gains available
1 means 1,10,100,1000 gains available
0 means 1,2,4,8 gains supported by board

[SGL] Average Average of data read

Block Diagram



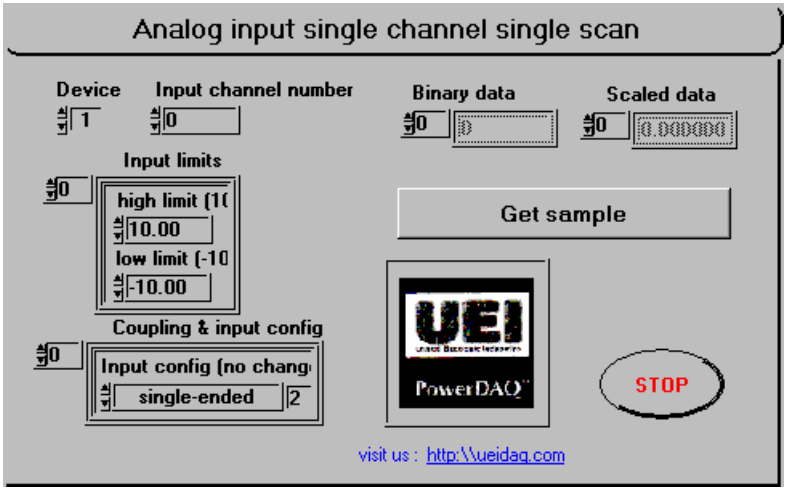
PD AI Single scan example.VI

PowerDAQ board single scan/sample example.

Simplest example – allows you to acquire single scan of channel list provided. (Non-timed non-buffered acquisition example). Can be used as a start point for data log applications.

This example just interfaces necessary inputs and outputs to "**PD AI Single Scan.VI**" Library VI.

Front Panel



Controls and Indicators

I16 Device see Common Controls topic for the description

U8 Input channel (0..Number of AI channels-1)

TF Web-site link

see Common Controls topic for the description

[2D8] Coupling & input config coupling & input config is an array of clusters. PowerDAQ LabVIEW driver use only type of inputs (differential/single-ended) and, if both of them specified differential have high priority and used for board configuration

[2D8] Input limits input limits (high and low, measured in Volts)

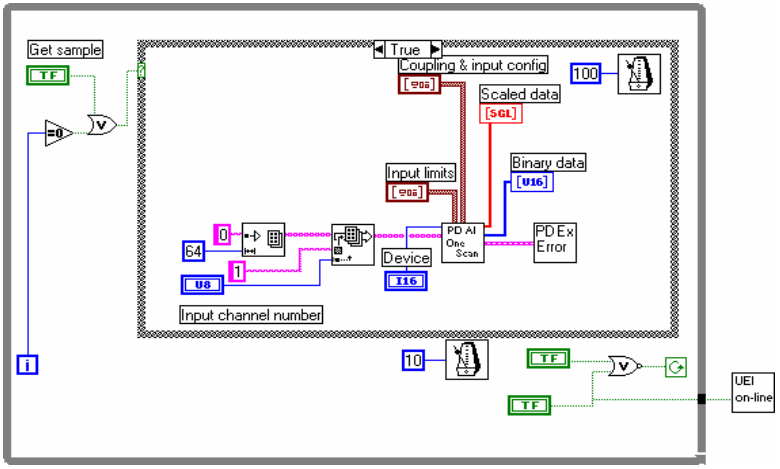
TF stop see Common Controls topic for the description

TF Get sample Acquire and get data

[U16] Binary data binary data is an array that contains binary analog input data if output units requests binary data.

[SGL] Scaled data scaled data is an array that contains analog input data in scaled data units.

Block Diagram



See also

- PD AI Single Scan.VI
- PD AI Sample Channel.VI

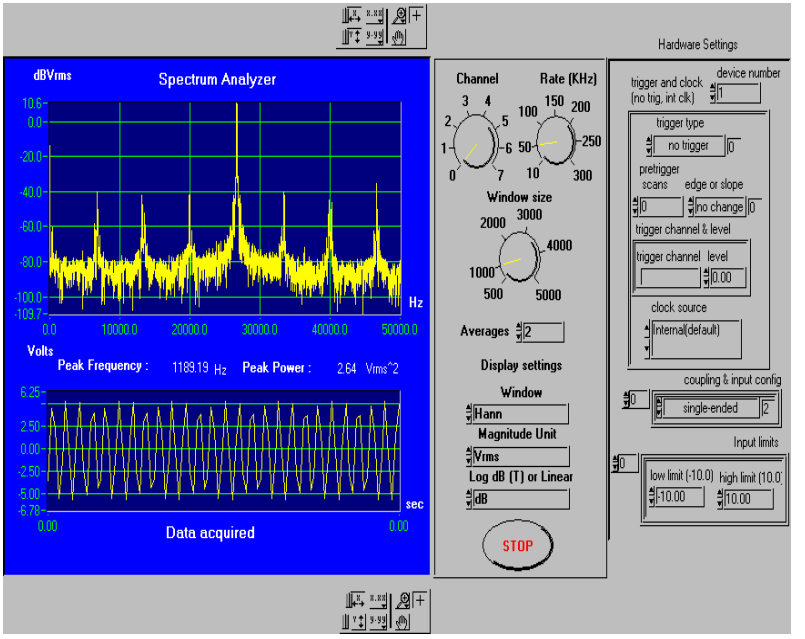
PD AI Spectrum Analyzer.VI

This VI performs non-stop acquisition and analyzes input signal spectrum on fly. For start-acquire-stop-analyze mode "PD AI Wave spectrum analyzer.VI" should be used.

Continuos Spectrum Analyzer computes the RMS averaged power spectrum on a single channel and also detect the peak frequency component and the actual frequency and power.

All acquisition parameter changes will take effect only after acquisition will be restarted.

Front Panel



Controls and Indicators

TF **stop** see Common Controls topic for the description

I32 **Window** Select the desired window type

I32 **Magnitude Unit** Select the units for the power spectrum display

U16 **Log dB (T) or Linear** Select linear or dB modes for the magnitude display.

If dB is selected, dBV(rms) or dBVpk is used for the display for the amplitude, auto power, and cross power spectrums where the reference is 1Vrms or 1Vpk respectively. dB is used for amplitude spectrum because that is already a ratio. Coherence, impulse response and time waveform are never shown in dB.

U32 **Averages** Select the number of averages you wish to compute. For most functions RMS averaging is performed. For the amplitude spectrum, only additive averaging is performed. The amplitude phase and time domain waveform just show the most recent signal acquired (no averaging).

The number of averages is limited only by the amount of memory available.

For a coherence measurement, you must have at least two averages.


coupling & input config coupling & input config is an array of clusters. PowerDAQ Labview driver use only type of inputs (differential/single-ended) and, if both of them specified differential have high priority and used for board configuration


Input limits input limits is an array of clusters. You can use it for tune board gain and input type. We use this input for capability with NI G sources.

Specify low and high level of input signals (in Volts) or leave empty array, which means the input limits keep their default settings.


Channel Analog input channel number to be analyzed


Rate (KHz) Acquisition rate


device number see Common Controls topic for the description


Window size Select the desired window size


Data acquired Waveform graph – shows the data acquired from the selected channel


Peak Frequency The pick frequency in current spectrum


Peak Power The power of the pick frequency in current spectrum

PD AI Stream data into LabVIEW array.VI

This example includes some advanced features such as special buffer settings.

Acquires any one analog input channel of the PowerDAQ board and reads the acquisition process to the LabVIEW array. Rate and buffer size can be changed. This VI acquires and read from the "**PD AI Read.VI**" all the data acquired. The "**PD AI Fine Tune.VI**" used to change default acquisition settings. Depends of your task you can set the values of *buffer overrun, gap-free mode* and *read ahead allowed* constants on the diagram.

This example is a good start point for the data-critical applications development.

For stream-to-disk applications use "PD AI Stream-to-disk example.VI".

Note That buffer size should be at least 1.5*rate for rates between 10K and 200K and 2 to 3*rate for high frequencies. Increasing the buffer increases the stability and reliability of acquisition at high frequencies. Use hardware settings for changing default values of input limits (and thus gains).

Controls and Indicators

Device see Common Controls topic for the description

Scan rate Frequency of acquisition. If you acquiring more than 1 channel actual frequency for each channel should be calculated as

(Scan rate)/(Number of active channels)

Note That actual scan rate can be found using the "PD AI Start.VI" and actual number of channels in channel list is returned from "PD AI Config.VI" call.

Size of buffer set size of buffer (in bytes) for storing data from board - 200000 default. Use the following rule: buffer size should be at least 1.5*rate for rates between 10K and 200K and 2-3*rate for high frequencies. Increasing the buffer increase the stability and reliability of acquisition system at the high frequencies but frame size is a function from buffer size unless you override this dependence using the *Frame Size* input of "PD AI Fine Tune.VI".

stop see Common Controls topic for the description


Input limits input limits is an array of clusters. You can use it to set board gain and input type (unipolar/bipolar) for the each channel. Please refer to "PD AI Config.VI" documentation for the details.

Input channel number (0..Number of AI channels-1)

Click to visit our web-site to download the latest version of SDK or up-to-date information about our products

Scans per read Indicator – total amount of scans that will be transferred from PowerDAQ LabVIEWdriver into this VI each "PD

AI Read.VI" call.

[SGL] scaled data scaled data is a 2D array that contains analog input data in scaled data units. The data appears in columns, where each column contains the data for a single channel. The second (bottom) dimension selects which channel column. The first (top) dimension selects a single data point for that channel.

SGL scaled data: a two-dimensional array that contains scaled analog input data if output units requests scaled data. The first dimension is scans, the second dimension is channels.

U32 Number of channels (0..Number of AI channels-1)

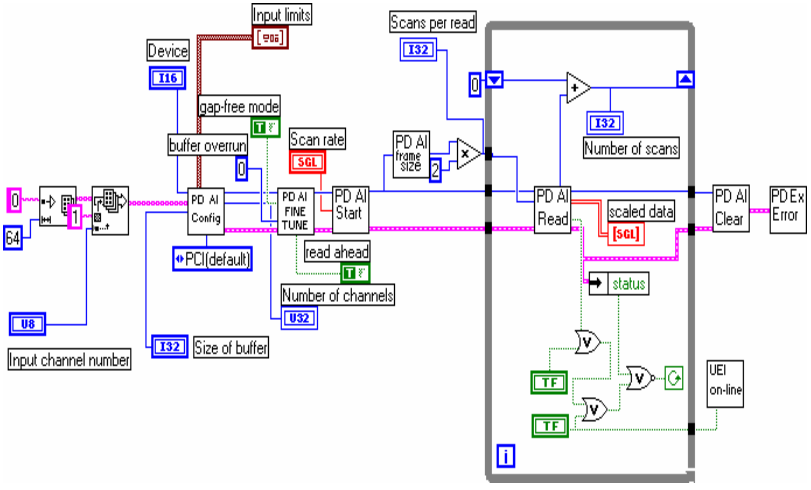
I32 Number of scans Number of scans received from PowerDAQ LabView driver - show progress of acquisition.

Note That PowerDAQ board works with frames of data and if frame size is greater than amount of data requested each "PD AI Read.VI" call the rest of the frame data will be lost. This is okay for most of applications such as FFT analysis and scope-like applications but for critical applications you can request exact amount of data which equal to the frame size from the "PD AI Read.VI".

See *Optimizing performance using the buffer settings* topic for the details.

Block Diagram

Note That "PD AI Fine Tune.VI" could be used to set an additional acquisition settings



See also

- PD AI Config.vi
- PD AI Start.vi
- PD AI Read.vi
- PD AI Fine Tune.VI
- PD AI Stream to disk.VI (Example VI)

PD AI Stream to disk.VI

PowerDAQ high-speed stream-to-disk example.

Acquires up to the eight of analog input channels of the PowerDAQ board and writes the results of the acquisition process to the file specified. Rate and file name can be changed. This VI uses "

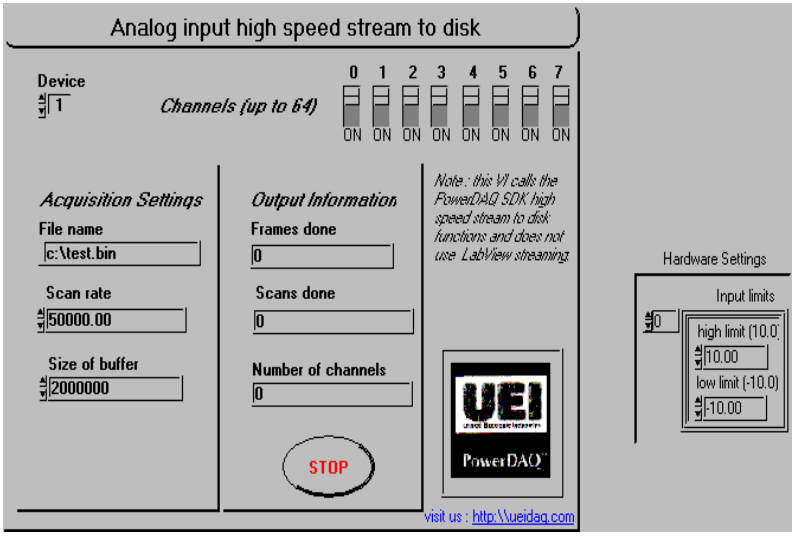
PD AI Stream Init**.VI"** to set an additional stream-to-disk specifix acquisition paramerers and writes to the disk using "**PD AI Stream.VI"** all the data acquired. This example is a good start point for the data-critical stream-to-disk applications development.

For stream-to-memory applications "**PD AI Stream data into LabVIEW array.VI**".

Note Check the loading of your system using Windows NT Task Manager or similar program. It's should not exceed 80-85 percent for reliable acquisition.

Note That buffer size should be at least 1.5*rate for rates between 10K and 200K and 2 to 3*rate for high frequencies. Increasing the buffer increases the stability and reliability of acquisition at high frequencies. Use hardware settings for changing default values of input limits (and thus gains).

Front Panel



Controls and Indicators

I16 Device see Common Controls topic for the description

TF 0 Channel switch. ON - specified channel present in channel list.

TF 1 Channel switch. ON - specified channel present in channel list.

TF 2 Channel switch. ON - specified channel present in channel list.

TF 3 Channel switch. ON - specified channel present in channel list.

TF 4 Channel switch. ON - specified channel present in channel list.

TF 5 Channel switch. ON - specified channel present in channel list.

TF 6 Channel switch. ON - specified channel present in channel list.

TF 7 Channel switch. ON - specified channel present in channel list.

abc File name File name to store the data acquired.

SGL Scan rate Frequency of acquisition. If you acquiring more than 1 channel actual frequency for each channel should be calculated as

(Scan rate)/(Number of active channels)

Note That actual scan rate can be found using the "PD AI Start.VI" and actual number of channels in channel list is returned from "PD AI Config.VI" call.

I32 Size of buffer set size of buffer (in bytes) for storing data from board - 200000 default. Use the following rule: buffer size

should be at least 1.5*rate for rates between 10K and 200K and 2-3*rate for high frequencies. Increasing the buffer increase the stability and reliability of acquisition system at the high frequencies.

stop see Common Controls topic for the description

Input limits input limits is an array of clusters. You can use it to set board gain and input type (unipolar/bipolar) for the each channel. Please refer to "PD AI Config.VI" documentation for the details.

Web-site link

see Common Controls topic for the description

Frames done Number of frames of data that already stored on disk.

The frame size in LabVIEW is defined in scans. One scan is a set of the samples – one for each channel in the channel list. This was done in this way because the most "PD AI xx" functions accept the number of scans as an input parameter, for the amount of the data requested. To find out the frame size the following equation should be used:

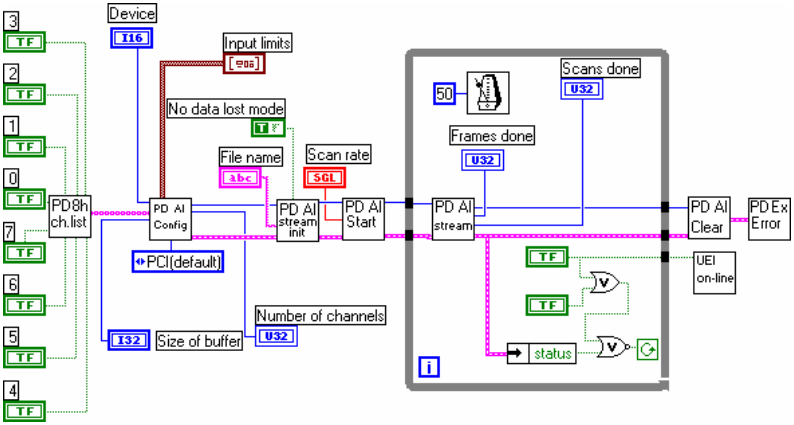
FrameSize = (((BufferSize / 2) / (Number Of The Frames))/Number Of The Acquired Channels)

There is a VI called "PD AI Frame Size.VI" which should be used to determine the current frame size.

Scans done Number of scans received from PowerDAQ LabView driver and stored on disk.

Number of channels Number of active channels in the channel list. Zero means error in channel list definition.

Block Diagram



See also

- PD AI Config.vi
- PD AI Start.vi
- PD AI Read.vi
- PD AI Fine Tune.VI
- PD AI Stream data into LabVIEW array.VI (Example VI)
- Multiple boards stream-to-disk examples

PD AI Wave Spectrum Analyzer.VI

This VI performs start-acquire-stop-analyze acquisition and analyzes input signal spectrum on fly. For non-stop mode "**PD AI Spectrum Analyzer.VI**" should be used.

Spectrum Analyzer computes the RMS averaged power spectrum on a single channel and also detect the peak frequency component and the actual frequency and power.

All acquisition parameter changes will take effect immediately.

For controls description see "**PD AI Spectrum Analyzer.VI**".

Analog Output Examples

The Analog Output Example VIs set includes:

- PD AO Buffered Update.VI
- PD AO Two channels single update.VI
- PD AO Two Channels.VI
- PD AO32 Buffered Wave.VI
- PD AO32 Wave.VI

All those VIs are located in the **<LabVIEW folder>\VI.LIB\PowerDAQ\Examples\AnalogOut** directory if you have PowerDAQ SDK installed on you PC.

Analog Output Examples Overview

The table below provides a brief description about each analog output example VI

	VI file name	Description
1	PD AO Buffered Update.VI	PowerDAQ MF[S] series high-speed buffered analog output example
2	PD AO Two channels single update.VI	Two channels single update example
3	PD AO Two Channels.VI	Two channels buffered update using the standard LabVIEW buffering mechanism (works fine up to 1K Sample/sec update rate).
4	PD AO32 Buffered Wave.VI	PowerDAQ AO series high-speed buffered analog output example with extended buffer
5	PD AO32 Wave.VI	PowerDAQ AO series high-speed buffered analog output example with DSP buffer

PD AO Buffered Update.VI

PowerDAQ analog output two channels continuous buffered write example.

In this example DSP-based on-board buffer used directly to store the data outputted. Following update modes are supported (**PD AO Start.VI**" for details) :

2 – buffered event-based update mode (used with "PD AO Wave**.VI** only)

(1024 samples/scans per "PD AO Wave.VI" call maximum)

3 – buffered continuous update mode (used with "**PD AO Wave.VI** only)

(2048 samples/scans per "PD AO Wave.VI" call maximum)

4 – DSP-based auto-regeneration update mode (used **PD AO Wave.VI** only) (2048 stored on DSP samples/scans maximum).

This example will work correctly only with MF[S] board series because the data formatted into MF[S] series board analog output format. See the

Internal Data Presentation for the PowerDAQ™ I, II MF(S)/AO/DIO boards topic for the details.

Controls and Indicators

Device see **Common Controls** topic for the description

Output rate Frequency of acquisition (500..100000Hz).
Note That for PDx-MF[S] boards update rate is always channel list clock because two available on board analog output channels are updated simultaneously. For the PD2-AO boards the frequency is refer to aggregate throughput and to find update rate per channel you should divide this frequency to the number of the channels updated.

stop Stops the acquisition after next buffer write

Output array size set size of array that will be passed

into the analog output (1024..2048 scans)

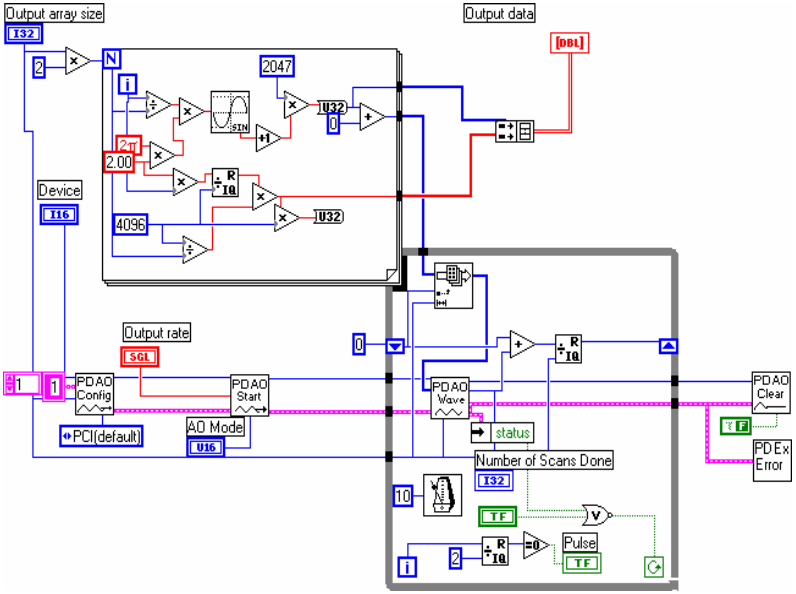
U16 **AO Mode** This input is used to specify an analog output mode for the PowerDAQ board.
0 – no change
1 – single update mode (used with "**PD AO Write.VI**" only)
2 – buffered event-based update mode (used with "PD AO Wave.VI only)
(1024 samples/scans per "PD AO Wave**.VI**" call maximum)
3 – buffered continuous update mode (used with "PD AO Wave.VI only)
(2048 samples/scans per "PD AO Wave.VI" call maximum)
4 – DSP-based auto-regeneration update mode (used with "PD AO Wave**.VI** only) (2048 stored on DSP samples/scans maximum)
5 – PowerDAQ buffering buffered update mode (used **with "PD AO Buffered Wave.VI** only) (no limitations for the buffer size)
See *Analog output modes* topic for the details

TF **Pulse** Shows the 'live' pulse of acquisition

DBL **Output data** Graph. Shows the analog output data passed to the "PD AO Wave**.VI**".

I32 **Number of Scans Done** output - number of scans that had written to the on-board DSP buffer when "PD AO Wave**.VI**" was called last time

Block Diagram



See also

-

PD AO Config.vi

- PD AO Start.vi
- PD AO Wave.vi

- PD AO32 Wave.VI (Example VI)

- PD AO32 Buffered Wave.VI (Example VI)

PD AO Two channels.VI

PowerDAQ analog output two channels continuous buffered write example using the LabVIEW buffering mechanism. The update frequency rate should not exceed 1K Sample/sec. For more advanced high-speed applications see "**PD AO Buffered Update.VI**" or "PD AO32 Buffered Wave.VI".

In this example user-defined size buffer allocated by PowerDAQ LabVIEW driver to store the data outputted. The data itself is an array filled by user-defined dummy values. Only single update mode is supported (**PD AO Start.VI**" for details) :

1 – single update mode (used with "**PD AO Write.VI**" only)

This example will work correctly only with MF[S] board series because the data formatted into MF[S] series board analog output format. See the

Internal Data Presentation for the PowerDAQ™ I, II MF(S)/AO/DIO boards topic for the details.

Controls and Indicators

Device see Common Controls topic for the description

0 Channel switch. ON - specified channel present in channel list.

1 Channel switch. ON - specified channel present in channel list.

Output rate Frequency of acquisition (1..1000Hz).

Note That for PDx-MF[S] boards update rate is always channel list clock because two available on board analog output channels are updated simultaneously. For the PD2-AO boards the frequency is refer to aggregate throughput and to find update rate per channel you should divide this frequency to the number of the channels updated.

Size of buffer Set size of buffer (in samples) for storing data for analog output - 512 default but you could increase this value up to 10000 if you want to create clear waves. Buffer allocated inside the PowerDAQ driver.

stop Stops the acquisition after next buffer write

Web-site link

see Common Controls topic for the description

Channel 0 value output value for channel 0 in Volts

Channel 1 value output value for channel 1 in Volts

Output array size set size of array that will be passed into the analog output

Allow regeneration allow regeneration. Regeneration means generating the same data more than once. If allow regeneration is TRUE, when all the data in the buffer has been generated, LabVIEW starts generating again from the beginning of the buffer, regardless of whether the data there is new or old. If allow regeneration is FALSE and the data about to be generated has already been generated, the VI returns an error. This parameter defaults to TRUE.

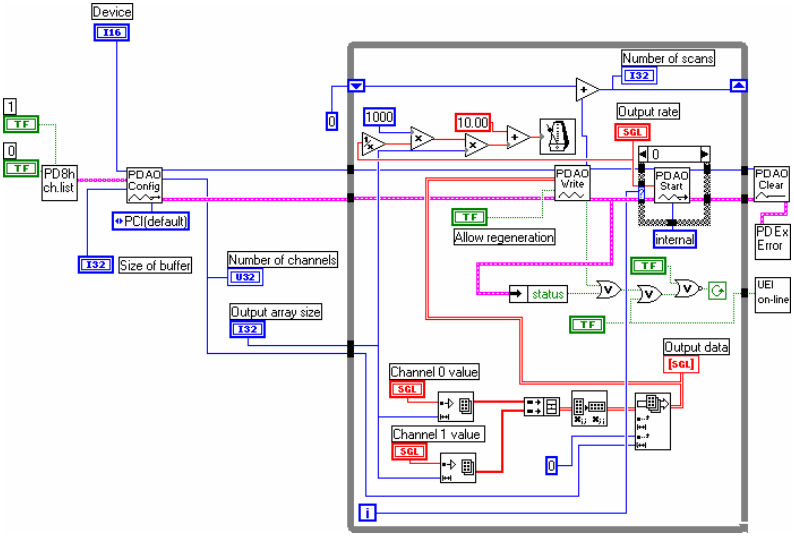
Number of channels Number of active channels in channel list. Zero means error in channel list definition.

Number of scans Number of scans passed to PowerDAQ LabView driver - show progress of acquisition.

Output data scaled data is a 2D array that contains analog input data in scaled data units. The data appears in columns, where each column contains the data for a single channel. The second (bottom) dimension selects which channel column. The first (top) dimension selects a single data point for that channel.

scaled data: a two-dimensional array that contains scaled analog output data. The first dimension is scans, the second dimension is channels.

Block Diagram



See also

-

PD AO Config.vi

- PD AO Start.vi
- PD AO Wave.vi
- PD AO Two channels single update.VI (Example VI)

- PD AO32 Buffered Wave.VI (Example VI)

PD AO Two channels single update.VI

PowerDAQ analog output two channels single update.

This example allows to perform non-timed analog output operation. For more advanced high-speed buffered analog output applications see "**PD AO Buffered Update.VI**" or "PD AO32 Buffered Wave.VI".

In this example the one of the simplest analog output Library VI used : "**PD AO Write One Update.VI**".

This example will work correctly both MF[S] and AO board series because the scaled data (not binary) used.

Controls and Indicators

Device see Common Controls topic for the description

stop see Common Controls topic for the description

Web-site link
see Common Controls topic for the description

Channel 0 value Output value (in Volts) for channel 0

Channel 1 value Output value (in Volts) for channel 1

PD AO32 Buffered Wave.VI

PowerDAQ analog output multiple channels continuous buffered write example.

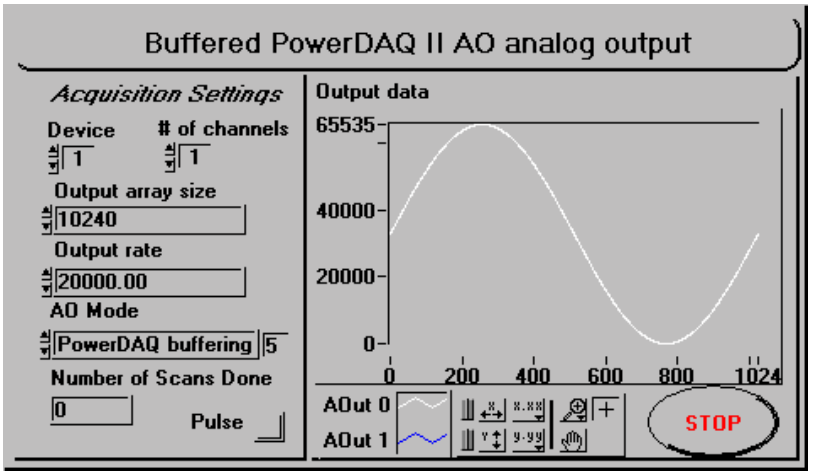
In this example DSP-based on-board buffer used directly to store the data outputted. Following update mode is supported (see "**PD AO Start.VI**" for details) :

5 - PowerDAQ buffering buffered update mode (used with "**PD AO Buffered Wave.VI**" only) (no limitations for the buffer size)

Note This example will work correctly only with AO board series because the data formatted into AO series board analog output format. See the

Internal Data Presentation for the PowerDAQ™ I, II MF(S)/AO/DIO boards topic for the details.

Front Panel



Controls and Indicators

I16 Device see Common Controls topic for the description

SGL Output rate Frequency of acquisition (500..100000Hz).

Note That for PDx-MF[S] boards update rate is always channel list clock because two available on board analog output channels are updated simultaneously. For the PD2-AO boards the frequency is refer to aggregate throughput and to find update rate per channel you should divide this frequency to the number of the channels updated.

TF stop see Common Controls topic for the description

I32 Output array size set size of buffer allocated to store the analog output data

(1024..2048000 samples)

U16 AO Mode This input is used to specify an analog output mode for the PowerDAQ board.

0 - no change

1 - single update mode (used with "PD AO Write.VI" only)

2 - buffered event-based update mode (used with "PD AO Wave.VI only)

(1024 samples/scans per "PD AO Wave.VI" call maximum)

3 - buffered continuous update mode (used with "PD AO Wave.VI

only)

(2048 samples/scans per "PD AO Wave.VI" call maximum)

4 - DSP-based auto-regeneration update mode (used with "PD AO Wave.VI only) (2048 stored on DSP samples/scans maximum)

5 - PowerDAQ buffering buffered update mode (used with "PD AO Buffered Wave.VI only) (no limitations for the buffer size)

See Analog output modes topic for the details

# of channels # of channels: specify the number of channels to be updated

Pulse Shows the 'live' pulse of acquisition

Output data Graph. Shows the analog output data passed to the "PD AO Buffered Wave.VI".

Number of Scans Done output - number of scans(MF[S])/samples(AO) that had written to the on-board DSP buffer when "PD AO Buffered Wave.VI" was called last time

See also

-

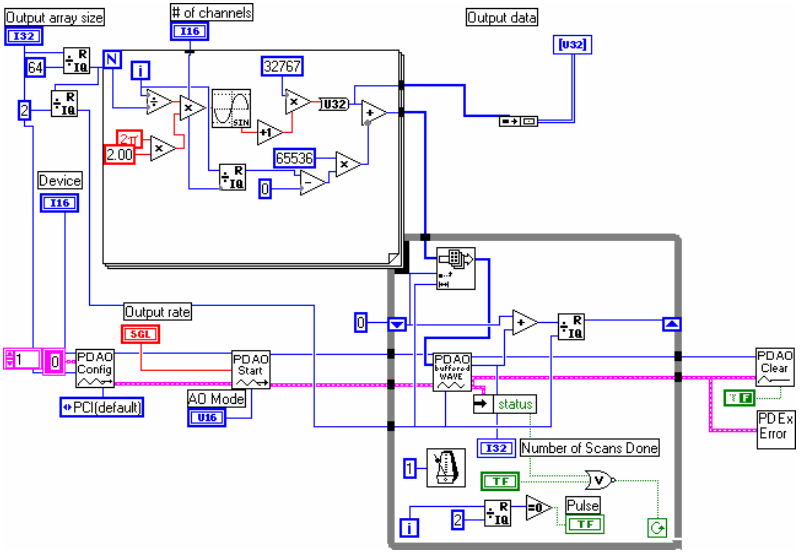
PD AO Config.vi

- PD AO Start.vi
- PD AO Buffered Wave.vi
- PD AO Two channels single update.VI (Example VI)

- PD AO32 Wave.VI (Example VI)

Block Diagram

Note That the data prepared to PD2-AO boards format when in 32-bit DWORD 16 LSBs represents the data and next 5 MSBs represents the channel #



PD AO32 Wave.VI

PowerDAQ analog output multiple channels continuous buffered write example.

In this example DSP-based on-board buffer used directly to store the data outputted.

Following update modes are supported (see "**PD AO Start.VI** " for details) :

2 - buffered event-based update mode (used with "**PD AO Wave.VI** only)

(1024 samples/scans per "PD AO Wave.VI" call maximum)

3 - buffered continuous update mode (used with "**PD AO Wave.VI** I only)

(2048 samples/scans per "PD AO Wave.VI" call maximum)

4 - DSP-based auto-regeneration update mode (used **PD AO Wave.VI"** only) (2048 stored on DSP samples/scans maximum).

Note This example will work correctly only with PD2-AO board series because the data formatted into AO series board analog output format. See the

Internal Data Presentation for the PowerDAQ™ I, II MF(S)/AO/DIO boards topic for the details.

Controls and Indicators

See "

PD AO32 Buffered Wave**.VI**" for controls description and a block diagram details.

Digital I/O Examples

The Digital Input and Output Example VIs set includes:

- PD DIO Buffered Read.VI
- PD DIO Buffered Write.VI
- PD DIO Single Read.VI
- PD DIO Single Write.VI
- PD DIO-128 Single Read And Write.VI

All those VIs are located in the **<LabVIEW folder>\VI.LIB\PowerDAQ\Examples\DigitalIO** directory if you have PowerDAQ SDK installed on you PC.

Digital IO Examples Overview

The table below provides a brief description about each digital input and output example VI

	VI file name	Description
1	PD DIO Buffered Read.VI	Digital Input buffered read using the standard LabVIEW buffering mechanism (works fine up to 1K Sample/sec update rate).
2	PD DIO Buffered Write.VI	Digital Output buffered update using the standard LabVIEW buffering mechanism (works fine up to 1K Sample/sec update rate).update example
3	PD DIO Single Read.VI	Digital Input non-timed read example
4	PD DIO Single Write.VI	Digital Output non-timed read example
5	PD DIO-128 Single Read And Write.VI	PowerDAQ II DIO boards series example

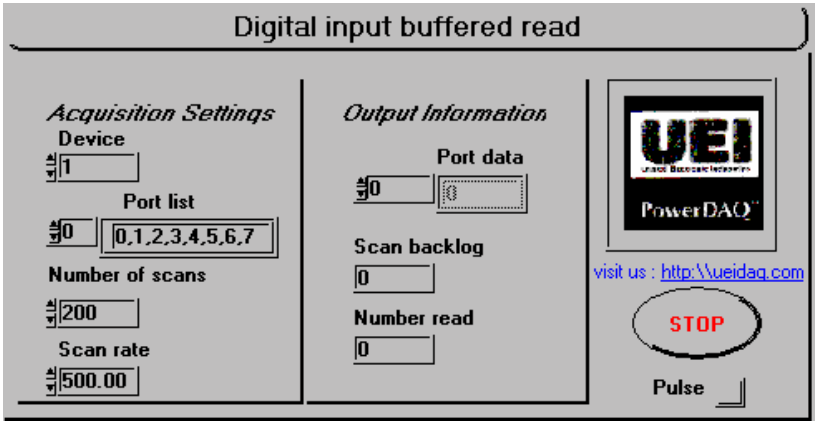
PD DIO Buffered Read.VI

PowerDAQ digital input continuos read example.

This VI continuously reads the digital data from the digital lines specified.

Note That only PDx-MF[S] and PD2-AO boards are supported by this VI

Front Panel



Controls and Indicators

TF stop see Common Controls topic for the description

TF Web-site link

see Common Controls topic for the description

SGL Scan rate Frequency of acquisition. For the PDx-MF[S] boards actual frequency for each channel is same.

I32 Number of scans Set size of buffer (in bytes) for storing data from board - 128 default if zero specified

abc Port list

see Common Controls topic for the description

U16 Device see Common Controls topic for the description

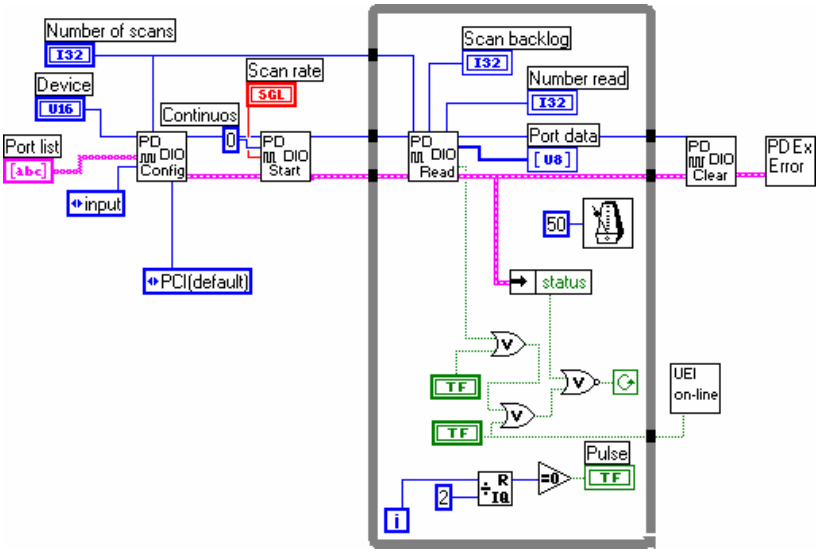
I32 Number read output - number of scans that was read from PowerDAQ board

[U8] Port data port data: a 1D array that contains the digital data that VI obtained from the internal buffer

I32 Scan backlog output - number of scans of unread data

TF Pulse Shows the 'live' pulse of acquisition

Block Diagram



See also

- PD DIO Config.VI
- PD DIO Start.VI
- PD DIO Read.VI
- PD DIO Write.VI
- PD DIO-128 Single Read And Write.VI (Example VI)

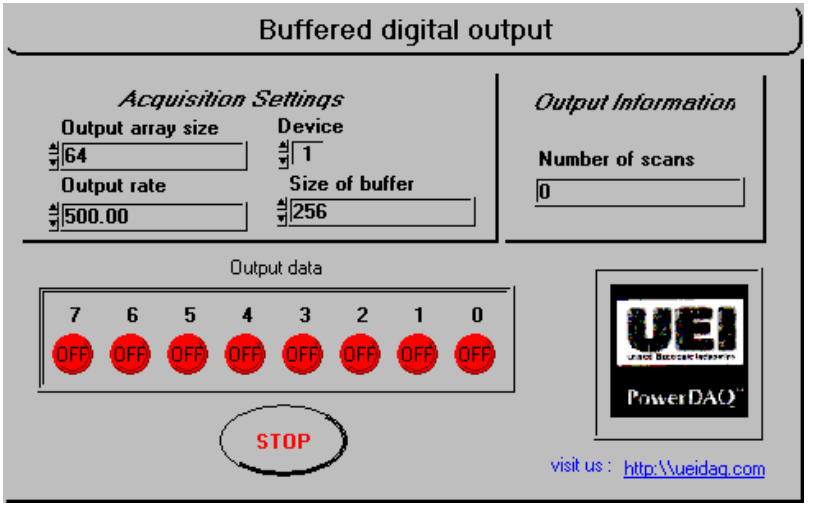
PD DIO Buffered Write.vi

PowerDAQ digital output buffered write example.

This VI shows PowerDAQ LabVIEW driver buffered digital output feature. Specified digital data value converted into the array and writes into the digital output.

Note That only PDx-MF[S] and PD2-AO boards are supported by this VI

Front Panel



Controls and Indicators

I16 **Device** see Common Controls topic for the description

SGL **Output rate** Frequency of acquisition (1..1000Hz)

I32 **Size of buffer** Set size of buffer (in samples) for storing data for the digital output, 512 by default.

TF **stop** see Common Controls topic for the description

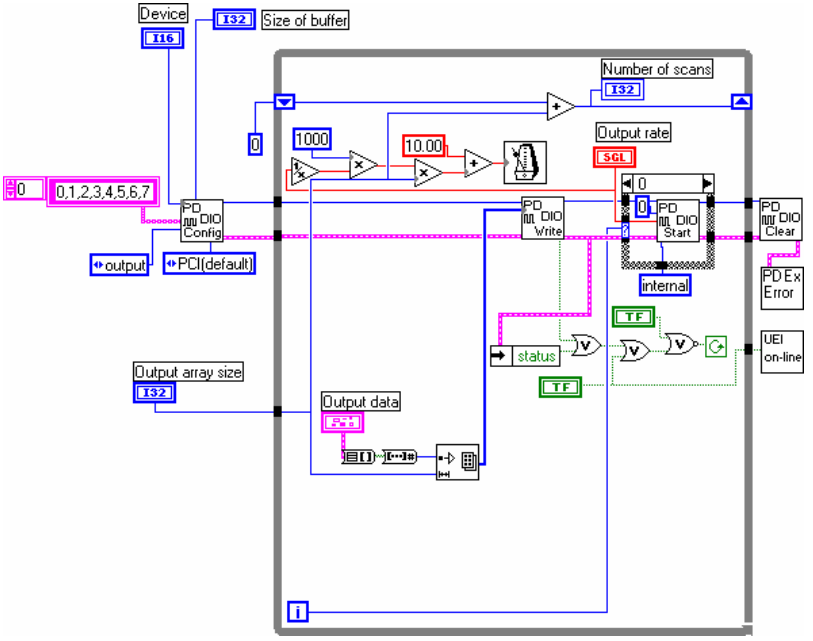
TF **Web-site link**
see Common Controls topic for the description

Output array size set size of array that will be passed into the digital output

Output data Digital data to be written to digital port

Number of scans Number of scans passed to PowerDAQ LabView driver – shows the progress of acquisition.

Block Diagram



See also

- PD DIO Config.VI
- PD DIO Start.VI
- PD DIO Read.VI
- PD DIO Write.VI
- PD DIO-128 Single Read And Write.VI (Example VI)

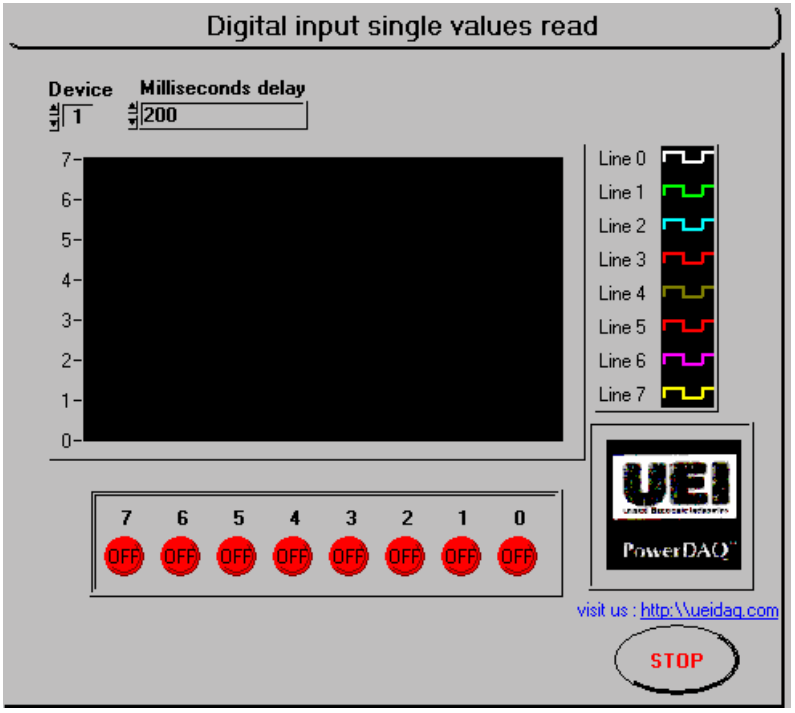
PD DIO Single Read.VI

PowerDAQ digital input single read example.

This VI reads the digital data from the PowerDAQ board and displays it. Delay (PC-clock based timed) and device ID can be changed.

Note That only PDx-MF[S] and PD2-AO boards are supported by this VI

Front Panel



Controls and Indicators


Device see Common Controls topic for the description


stop see Common Controls topic for the description

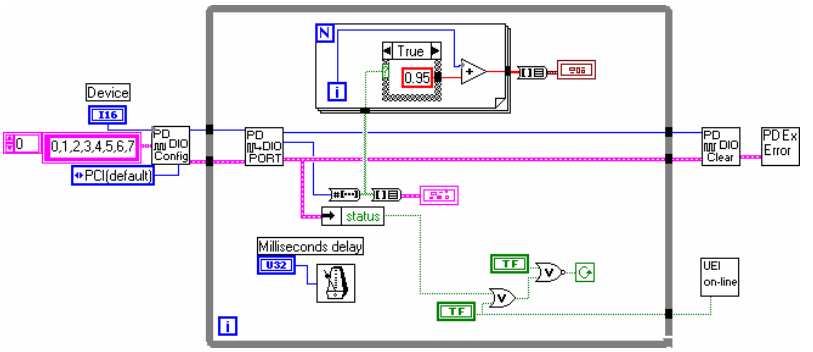

Web-site link
see Common Controls topic for the description


Milliseconds delay PC-clock based delay between readings


Digital data from of digital port


Time-based digital waveform of data read

Block Diagram



See also

- PD DIO Config.VI
- PD DIO Start.VI
- PD DIO Read.VI
- PD DIO Write.VI
- PD DIO-128 Single Read And Write.VI (Example VI)

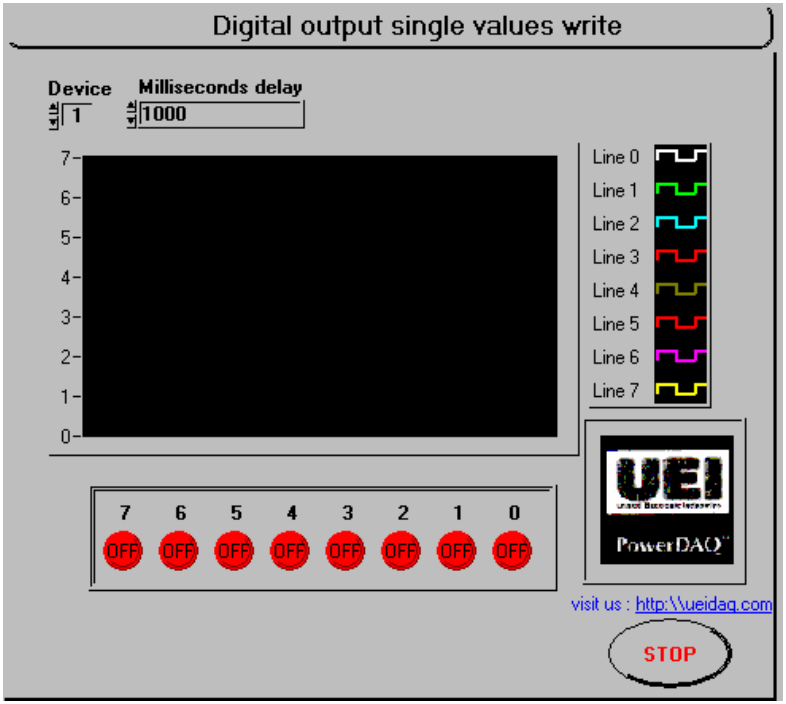
PD DIO Single Write.VI

PowerDAQ digital output single write example.

This VI writes the digital data to the PowerDAQ board and displays it. Delay (PC-clock based timed) and device ID can be changed.

Note That only PDx-MF[S] and PD2-AO boards are supported by this VI

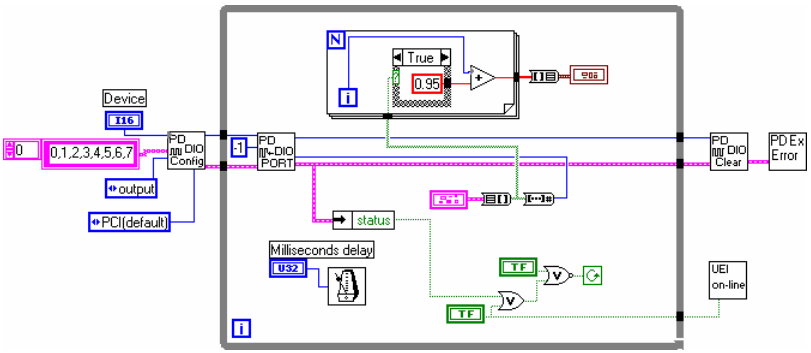
Front Panel



Controls and Indicators

Device see Common Controls topic for the description
stop see Common Controls topic for the description
Web-site link
see Common Controls topic for the description
Digital data to be written to digital port
Milliseconds delay PC-clock based delay between writings
Time-based digital waveform of data been written

Block Diagram



See also

- PD DIO Config.VI
- PD DIO Start.VI
- PD DIO Read.VI
- PD DIO Write.VI
- PD DIO-128 Single Read And Write.VI (Example VI)

PD DIO-128 Single Read And Write.VI

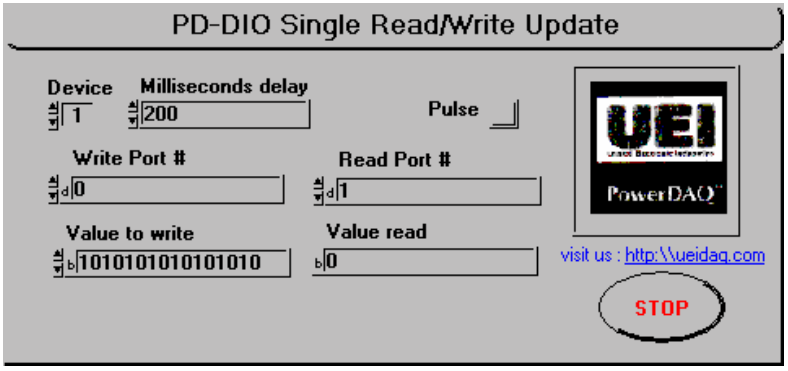
PowerDAQ DIO board digital input and output single read-write example.

This VI writes the digital data to the PowerDAQ II DIO board specified in deviceID and reads it back. Delay (PC-clock based timed) and device ID can be changed.

The channel list definition is on the Diagram. Note that there are two different channel lists for input and output operations. If port not specified as output the outputs of this port will be tri-stated but you can read value of any port regardless of is this port specified as input or as output.

Note That only PD2-DIO boards are supported by this VI

Front Panel



Controls and Indicators

I16 **Device** see Common Controls topic for the description

TF **stop** see Common Controls topic for the description

TF Web-site link
see Common Controls topic for the description

U32 **Milliseconds delay** PC-clock based delay

U16 **Write Port #** input - 16-bits port #. Can be 0..3 for PD-DIO-64

and 0..7 for PD-DIO-128.

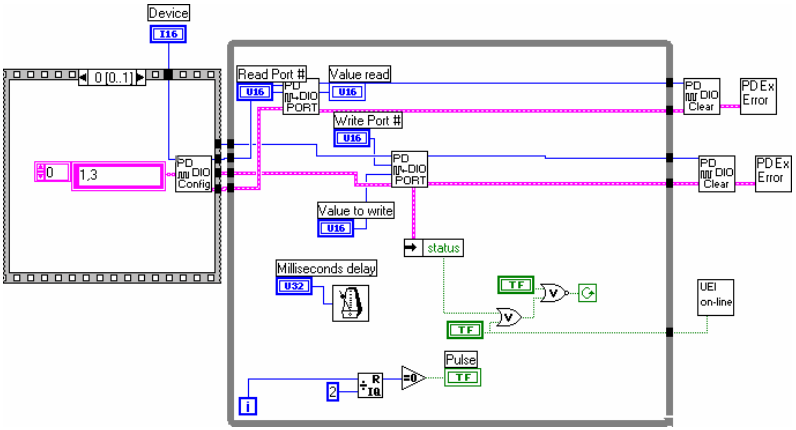
U16 **Value to write** input - 32-bits pattern: a bit pattern representing the state of the lines in the port.

U16 **Read Port #** input - Port # to read. Can be 0..3 for PD-DIO-64 and 0..7 for PD-DIO-128 boards

U16 **Value read** output - 16-bits data read from the port specified

TF **Pulse** Shows the 'live' pulse of acquisition

Block Diagram



See also

- PD DIO Config.VI
- PD DIO Start.VI
- PD DIO-128 Read.VI
- PD DIO-128 Write.VI

Counters/Timers Examples

Counters/Timers Example VIs set includes:

- PD UCT Frequency Counter.vi
- PD UCT Multimode counter.vi

All those VIs are located in the **<LabVIEW folder>\VI.LIB\PowerDAQ\Examples\Counters** directory if you have PowerDAQ SDK installed on you PC.

Counters/Timers Examples Overview

The table below provides a brief description about each counter-timer example VI

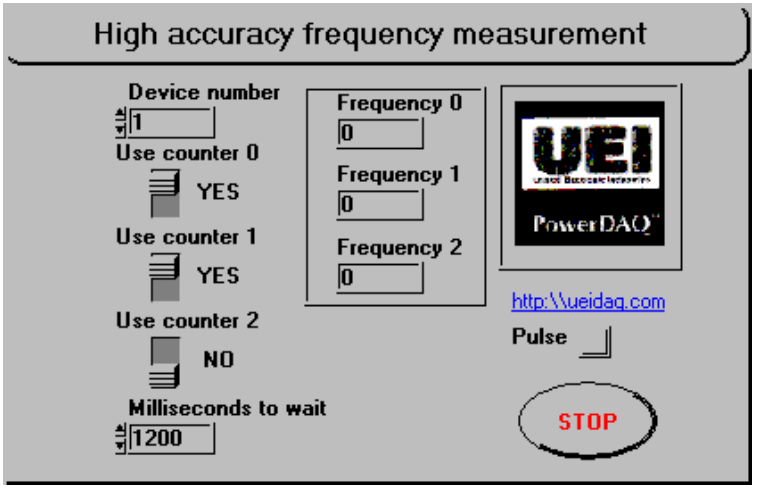
	VI file name	Description
1	PD UCT Frequency Counter.vi	PowerDAQ frequency counter VI allows to use up to three of the 82C54 counters for the frequency measurement. Frequency input range is from 1Hz to 65534 Hz.
2	PD UCT Multimode counter.vi	This VI provides direct interface to on-board 82C54 counter chip

PD UCT Frequency Counter.VI

This VI allows to measure up to the three frequency inputs using the 82C54 counter-timer and on-board DSP Timer (for the time-base operation). The frequency range is 1-65534 Hz for one counter. You can cascade them to measure higher frequencies.

Note That only PDx-MF[S] boards are supported by this VI . For the some counting modes correct external wired connection required.

Front Panel



Controls and Indicators

I16 **Device number** see Common Controls topic for the description
TF **Use counter 0** Use UCT 0 for measuring the frequency
TF **Use counter 1** Use UCT 1 for measuring the frequency
TF **Use counter 2** Use UCT 2 for measuring the frequency
U32 **Milliseconds to wait** Milliseconds delay to complete frequency

measurement. Should be greater than 1000 ms.

TF **stop copy** see Common Controls topic for the description

TF **Web-site link**
see Common Controls topic for the description

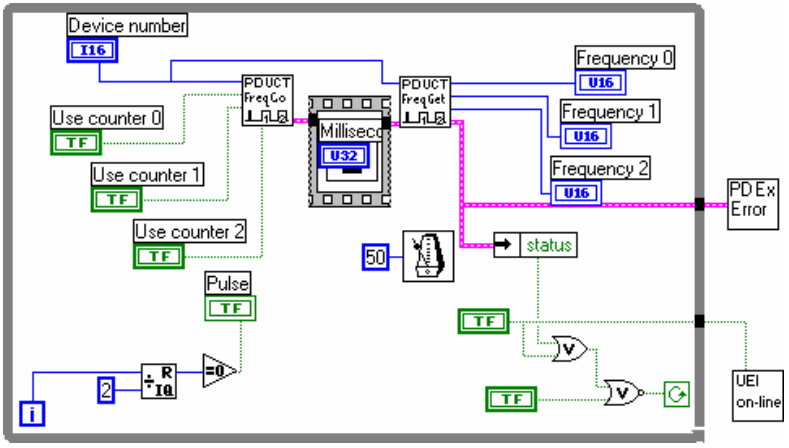
U16 **Frequency 0** Frequency measurement result at UCT0

U16 **Frequency 1** Frequency measurement result at UCT1

U16 **Frequency 2** Frequency measurement result at UCT2

TF **Pulse** Shows the 'live' pulse of acquisition

Block Diagram



See also

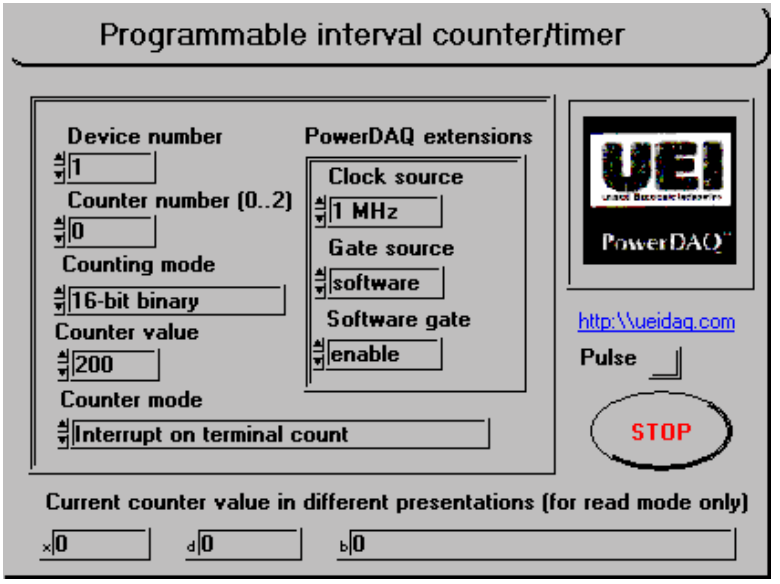
- PD UCT Control.VI
- PD UCT Freq Start.VI
- PD UCT Freq Ready.VI

PD UCT Multimode counter.VI

This VI shows all possibilities of the PowerDAQ counter/timer subsystem. It i

Note That for some modes external inputs should be connected properly.

Front Panel



Controls and Indicators

TF Web-site link

see **Common Controls** topic for the description

TF **stop** see **Common Controls** topic for the description

Timer settings. Cluster. See member descriptions for details.

I16 **Device number** see **Common Controls** topic for the description

I16 **Counter number (0..2)** Specify 82c54 counter number from 0

through 2.

Counter mode Input - control code determines the mode in which the counter operates. Modes from 0 to 5 corresponding to 82c54 modes.

0: Interrupt or terminal count (default)

1: Hardware retriggerable one-shot

2: Rate generator

3: Square wave rate generator

4: Software-triggered strobe

5: Hardware-triggered strobe (retriggerable)

6: Read counter/timer value

7: Reset counter/timer subsystem

8: Leave mode unchanged

9: Reserved

10: Frequency measurement mode start

11: Frequency measurement result read

In mode 0, the output goes low after the mode set operation, and the counter begins to count down while the gate (software or hardware) input is high. The output goes high when counter reaches the terminal count (that is, when the counter decrements to 0) and stays high until counter mode will be written again.

In mode 1, the output goes low on the count following the leading edge of the gate (software or hardware) input and goes high on terminal count.

In mode 2, the output goes low for one period of the clock input. count indicates the period from one output pulse to the next.

In mode 3, the output stays high for one-half of the count clock pulses and stays low for the other half.

In mode 4, the output is initially high, and the counter begins to count down while the gate (software or hardware) input is high. On terminal count, the output goes low for one clock pulse, then goes high again.

Setup mode 5 is similar to mode 4, except that the gate

(software or hardware) input triggers the count to start.

Mode 10 enforces counter to start count external event coming in 1.00000 second input interval. The result can be read using the mode 11.

Mode 11 allows read the result of frequency measurement. Note that frequency should be in 1-65534Hz limits. 0 Hz will be read as 65535.

See the 82c54 Programmable Interval Timer data sheet in your product user manual for a detailed description of these modes and the associated timing diagrams.

Counting mode Input - defines the counter/time count mode

0: 4-decade BCD counter

1: 16-bit binary counter

Counter value input - 16-bit unsigned value that defines the period from one output pulse to the next. If control code is 0, 1, 4, or 5, count can be 0 through 65,535 in binary counter operation and 0 through 9,999 in BCD counter operation. If control code is 2 or 3, count can be 2 through 65,535 and 0 in binary counter operation and 2 through 9,999 and 0 in BCD counter operation.

PowerDAQ extensions Additional features for PowerDAQ boards

- clock source
- gate source
- software gate state

Clock source Defines the clock source for timer/counter counts.

PowerDAQ board provides three types of counting base clock :

Software - each count-down event specified via software

1 MHz - 1 MHz internal timebase clock

External - extarnal clock provided

if this input set to 'no change' value previous settings still take effect

Gate source Specify gate input for counter/timer source

PowerDAQ board provides two types of timer gate :
Software - gate turned on and off via software
External - extarnal gate used

if this input set to 'no change' value previous settings still take effect

Software gate Software gate value (used with gate source = 'software')

Enabled - enable counting
Disabled - disable counting

if this input set to 'no change' value previous settings still take effect

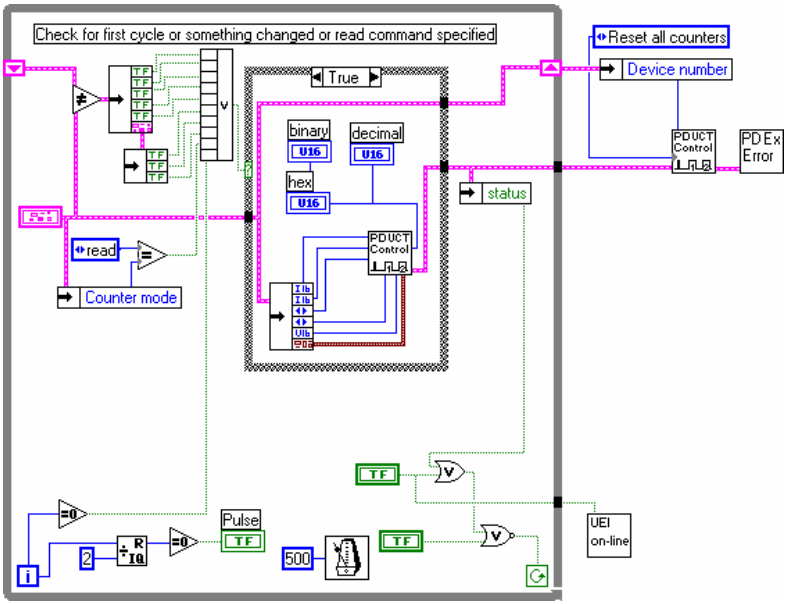
Pulse Shows the 'live' pulse of acquisition

Hex output - 16-bit unsigned read value: When you set control code to 6/11 (read), read value returns the value the VI read from the counter specified.

Binary output - 16-bit unsigned read value: When you set control code to 6/11 (read), read value returns the value the VI read from the counter specified.

Decimal output - 16-bit unsigned read value: When you set control code to 6/11 (read), read value returns the value the VI read from the counter specified.

Block Diagram



See also

- PD UCT Control.VI
- PD UCT Frequency Counter.VI (Example VI)

Multi-subsystem Examples

The Multi-subsystem Example VIs set includes:

- PD AIn & UCT.vi
- PD AIn & UCT with analog trigger.vi
- PD All Subsystem.vi

All those VIs are located in the **<LabVIEW folder>\VI.LIB\PowerDAQ\Examples\AllInOne** directory if you have PowerDAQ SDK installed on you PC.

Multi–subsystem Overview

The table below provides a brief description about each Multi-subsystem example VI

	VI file name	Description
1	PD AIn & UCT.vi	PowerDAQ buffered Analog Input and UCT simultaneous usage example
2	PD AIn & UCT with analog trigger.vi	Most advanced PowerDAQ example – extensive analog input with triggering and UCT example.
3	PD All Subsystem.vi	Buffered analog input with triggering and single-update analog output, digital input and output and UCT example.

PD Aln & UCT with analog trigger.VI

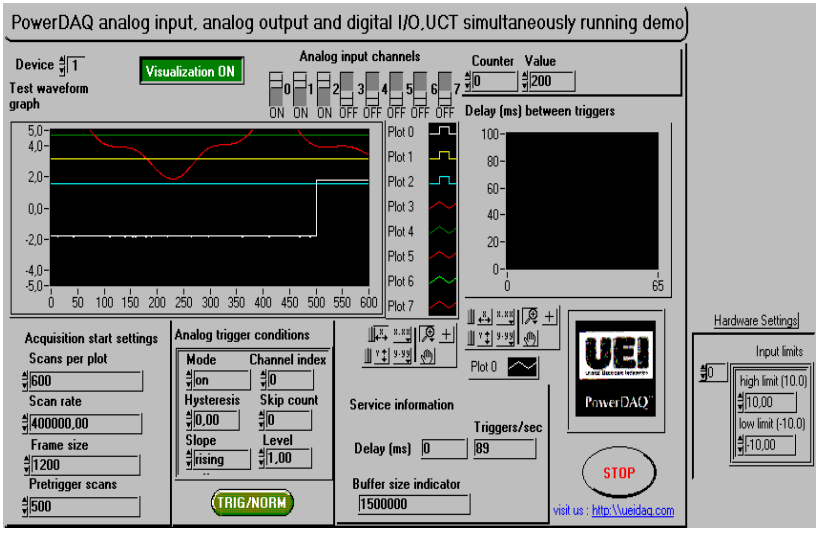
PowerDAQ example for Analog Input and Counter/Timer subsystems running simultaneously and an analog threshold trigger for triggering the analog input.

Note That there are no limitations to use any other subsystems such as Analog Output or Digital I/O available on PowerDAQ board simultaneously with already employed in this example. They are not showed on this example to make it simple. You can just cut and copy them from "PD All Subsystems.VI"

This example allows you to acquire up to 8 analog input channels and show the results of the acquisition on a multi-plot graph using an analog threshold trigger on any one channel. Both pre- and post-trigger data are displayed. The amount of pre- and post-trigger data can only be modified prior to execution. The pre-trigger data size is specified in scans and modified in the "Pretrigger scans" text box. The post-trigger data size is set by the "Scans per plot" text box and is equal to this value minus the value in the "Pretrigger scans" text box. The "Analog trigger conditions" cluster allows you to control the channel, which is being triggered on, the voltage level to trigger on, the type of edge (rising or falling), the "Hysteresis" value of the trigger, and the number of triggers, which will be ignored before accepting a trigger ("Skip count"). [The hysteresis value refers to a limit above or below the actual trigger level, which will need to be surpassed before it is considered to be a valid trigger. This compensates for the possibility of a noise spike causing an accidental trigger condition to be detected.] The "timelimit" constant (in the diagram of the VI) sets the windowing size to be analyzed in the data stream before control is returned to the other sections of the VI. If a trigger is not detected during this window, the trigger "Timeout" situation will flash and control will return to the other subsystems before returning to analyze another window of data. The minimum length of the trigger signal is the number of channels divided by the Scan rate. (If you don't acquire the trigger, you can't trigger on it.)

The rate, buffer size and number of points per plot can be changed. Use the following rules: the buffer size should be at least 1.5* the rate (for rates between 10K and 200K) and 2 -3* the rate (for higher frequencies). Increasing the buffer size increases the stability and reliability of the acquisition system at high frequencies. Use the hardware settings to change the default values for the input limits (and thus gains and range).

Front Panel



Controls and Indicators

I16 Device see Common Controls topic for the description

TF 0 Channel switch. ON - specified channel present in channel list.

TF 1 Channel switch. ON - specified channel present in channel list.

TF 2 Channel switch. ON - specified channel present in channel list.

TF 3 Channel switch. ON - specified channel present in channel list.

TF 4 Channel switch. ON - specified channel present in channel list.

TF 5 Channel switch. ON - specified channel present in channel list.

TF 6 Channel switch. ON - specified channel present in channel list.

TF 7 Channel switch. ON - specified channel present in channel list.

I32 Scans per plot Number of scans in this example defines two things- first this is the amount of date that will be displayed each time when analog trigger was occurred (in Trigger ON mode) or "PD AI Read.VI" called (in Trigger OFF mode). And second this number minus "Pre-trigger scans" number gives you a "Post-trigger scans" number. On the data sample displayed there is a 500 pre-trigger scans and 100 post-trigger scans in a total 600 scans/per plot quantity.

SGL Scan rate Frequency of acquisition (in a CV clock terms). Use channel list clock input in the "PD AI Start.VI" to define separate channel list clock. This is extremely important to use CL clock

instead of CV clock for PD2-MFS boards because of the nature of sample-and-hold amplifiers.

If you acquiring more than one channel using CV clock and set CL clock to continuous mode, actual frequency for each channel should be calculated as

(Scan rate)/(Number of active channels)


stop see Common Controls topic for the description


Input limits input limits is an array of clusters. See “PD AI Config.VI” for details.


Web-site link

see Common Controls topic for the description


Counter Specify 82c54 counter number from 0 through 2.

NOTE, that in this example counter is used as rate-generator and if you want to use it as a trigger source your should either increase the acquisition rate to 1 MHz or change timer mode to the squire-wave generator.


Value input - 16-bit unsigned value that defines the period from one output pulse to the next. If control code is 0, 1, 4, or 5, count can be 0 through 65,535 in binary counter operation and 0 through 9,999 in BCD counter operation. If control code is 2 or 3, count can be 2 through 65,535 and 0 in binary counter operation and 2 through 9,999 and 0 in BCD counter operation.


Pretrigger scans Number of scans to be displayed before trigger situation. Cannot be greater than frame size.


Analog trigger conditions The "Analog trigger conditions" cluster allows you to control the channel, which is being triggered on, the voltage level to trigger on, the type of edge (rising or falling), the "Hysteresis" value of the trigger, and the number of triggers, which will be ignored before accepting a trigger ("Skip count").


Mode input - trigger mode:

Off - clear all triggers

On – add analog trigger conditions

No change - leave trigger configuration unchanged


Channel index index of trigger channel in analog input channel list (beginning from 0)


Slope trigger signal direction

Rising – trigger on rising signal (default)

Falling – trigger on falling signal

No change – leave this condition unchanged


Level level of trigger signal in Volts. If over ranged - default 0.0 Volts used.

SGL Hysteresis The hysteresis value refers to a limit above or below the actual trigger level, which will need to be surpassed before it is considered to be a valid trigger. This compensates for the possibility of a noise spike causing an accidental trigger condition to be detected.

I32 Skip count number of times when triggering conditions skips.

TF Visualization The 'Visualization ON/OFF' Control enables or disables the real-time display and data processing. This control does not affect acquisition itself.

You can use this control to check the performance of your data acquisition system. Use delay (ms) between triggers graph to see it.

I32 Frame size The frame size (in this VI) defines the trigger window to catch the trigger. PowerDAQ LabVIEW driver will check input data for the trigger in the current frame and the rest of the in the previous frame. If there is no trigger next frame will be analyzed for trigger conditions.

At the same time there is some dependence between minimum frame size, on-board FIFO size, acquisition rate and maximum number of triggers that PowerDAQ LabVIEW driver is able to catch each second.

First of all frame size should not be less than 1/2 FIFO size measured in scans. Just divide the 1/2 FIFO-size from PowerDAQ Control Panel applet to the number of channels acquired to find out this number.

Second, PowerDAQ driver cannot catch more triggers than "number of frames/per second" value. Keep it in your mind when you define the frame size.

I32 Buffer size indicator indicates the buffer size calculated

[SGL] Test waveform graph Shows the results of the triggered acquisition

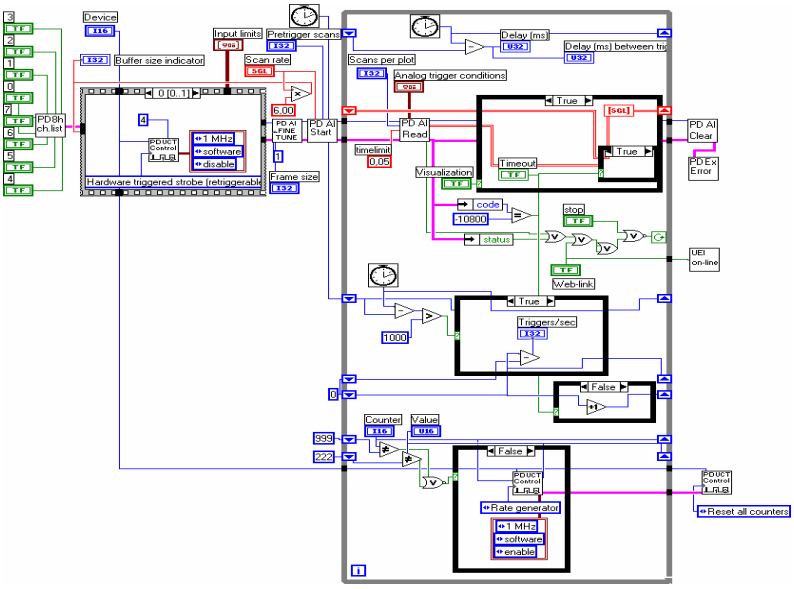
TF Timeout Indicates normal/trigger ON mode or trigger timeout whatever occurs

U32 Delay (ms) Shows the actual delay between the triggers.

U32 Delay (ms) between triggers “Delay between the triggers” graph shows the actual delay between the triggers. This is some kind of performance control. You can see a difference in performance clicking on "Visualization" button.

I32 Triggers/sec indicates how many triggers PowerDAQ LabVIEW driver catches every second.

Block Diagram



See also

- PD AI Acquire & Display all channels.VI
- PD UCT Control.VI

PD AIn & UCT .VI

Acquires all analog input channels of the PowerDAQ board and displays any eight of these channels in a graph. Additionally measure the frequency at UCT0 (1Hz-65534Hz). Rate, buffer size and number of points per plot can be changed.

This VI provides a simple example for continuous acquisition when the most of acquisition settings are set to they defaults. The output array rebuilds to provide required set of input channels to be displayed. Note that this operation required a lot of processor resources.

Note See "**PD AI Acquire & Display all channels.VI**" for details. That buffer size should be at least 1.5*rate for rates between 10K and 200K and 2 to 3*rate for high frequencies. Increasing the buffer increases the stability and reliability of acquisition at high frequencies. Use hardware settings for changing default values of input limits (and thus gains).

See also

- PD AI Acquire & Display all channels.VI
- PD UCT Control.VI
- PD UCT Freq Start.VI
- PD UCT Freq Ready.VI

PD All Subsystem.VI

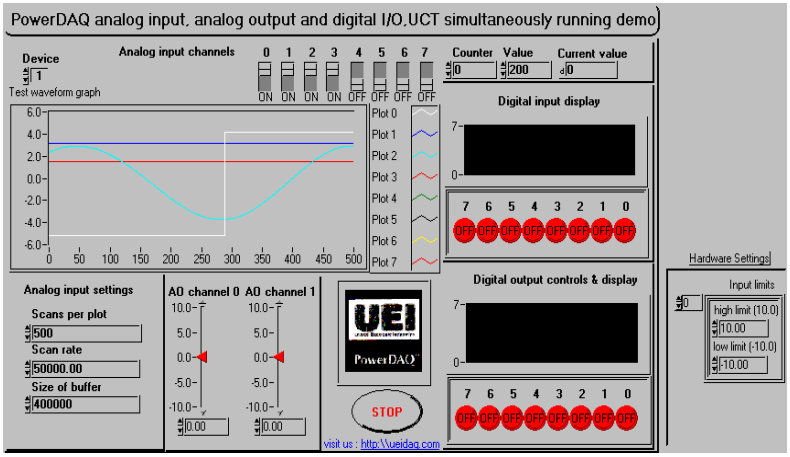
PowerDAQ example for all subsystems running simultaneously.

Allows you to acquire up to the eight analog input channels and shows the results of the acquisition on a multi-plot graph. The rate, buffer size and number of points per plot can be changed.

Additionally, it reads and writes data from the eight digital input and output channels and allows you to change the values for both analog output channels and set UCT0-2 rate.

Note That sequence control was used to provide known initialization sequence for all subsystems. The buffer size should be at least 1.5*rate for rates between 10K and 200K and 2 to 3*rate for high frequencies. Increasing the buffer increases the stability and reliability of acquisition at high frequencies. Use hardware settings for changing default values of input limits (and thus gains). This example will work only with PDx-MF [S] boards.

Front Panel



Controls and Indicators

I16 Device see Common Controls topic for the description

TF 0 Channel switch. ON - specified channel present in channel list.

TF 1 Channel switch. ON - specified channel present in channel list.

TF 2 Channel switch. ON - specified channel present in channel list.

TF 3 Channel switch. ON - specified channel present in channel list.

TF 4 Channel switch. ON - specified channel present in channel list.

TF 5 Channel switch. ON - specified channel present in channel list.

TF 6 Channel switch. ON - specified channel present in channel list.

TF 7 Channel switch. ON - specified channel present in channel list.

I32 Scans per plot Number of samples that will be showed at graph during the acquisition (grater or equal to the amount of scans requested from "PD AI Read.VI". There is no sense to show more then 1-3K scans because graphical operation takes too much processor time under the LabVIEW.

Note: That PowerDAQ board works with frames of data and if frame size is greater than amount of data requested each "PD AI Read.VI" call the rest of the frame data will be lost. This is okay for most of applications such as FFT analysis and scope-like applications but for critical applications you can request exact amount of data which equal to the frame size from the "PD AI Read.VI". See *Optimizing performance using the buffer settings* topic for the details.

SGL Scan rate Frequency of acquisition. If you acquiring more than 1

channel actual frequency for each channel should be calculated as

(Scan rate)/(Number of active channels)

Note That actual scan rate can be found using the "PD AI Start.VI" and actual number of channels in channel list is returned from "PD AI Config.VI" call.

Size of buffer set size of buffer (in bytes) for storing data from board - 200000 default. Use the following rule: buffer size should be at least 1.5*rate for rates between 10K and 200K and 2-3*rate for high frequencies. Increasing the buffer increase the stability and reliability of acquisition system at the high frequencies.

stop see Common Controls topic for the description

Input limits input limits is an array of clusters. See "PD AI Config.VI" for details.

Web-site link

see Common Controls topic for the description

AO channel 1 Output value (in Volts) for channel 1

AO channel 0 Output value (in Volts) for channel 0

Counter Specify 82c54 counter number from 0 through 2.

Value input - 16-bit unsigned value that defines the period from one output pulse to the next. If control code is 0, 1, 4, or 5, count can be 0 through 65,535 in binary counter operation and 0 through 9,999 in BCD counter operation. If control code is 2 or 3, count can be 2 through 65,535 and 0 in binary counter operation and 2 through 9,999 and 0 in BCD counter operation.

Test waveform graph Shows the results of acqusition

See also

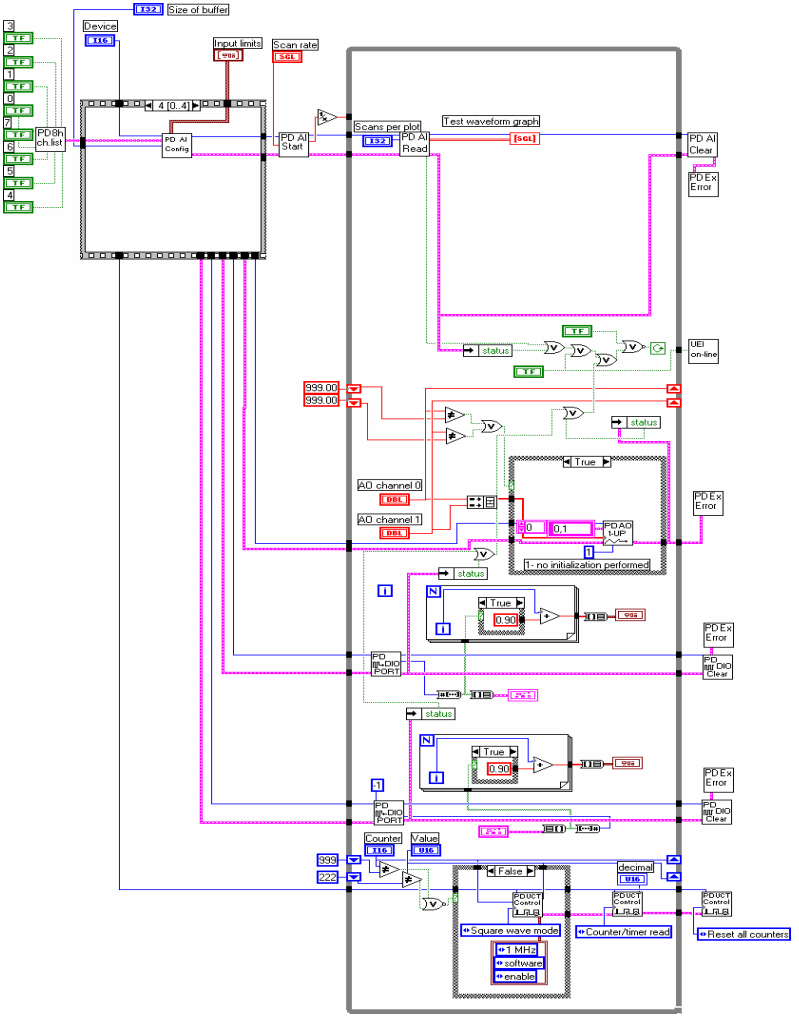
- PD AI Acquire & Display all channels.VI
- PD UCT Control.VI
- PD UCT Freq Start.VI

- PD UCT Freq Ready.VI
-

PD AO Config.vi

- PD AO Start.vi
- PD AO Wave.vi
- PD AO Two channels single update.VI (Example VI)
- PD DIO Config.VI
- PD DIO Start.VI
- PD DIO Read.VI
- PD DIO Write.VI

Block Diagram



Multi-boards Examples

The powerful feature of the PowerDAQ LabVIEW driver is multiple board support. You limited only by number of slots in your PC or backplane. We will include only description of multiple board VIs. You can download sources from our web-site www.Omega Engineeringdaq.com. See *Hardware settings* for an additionnal information about hardware setup.

The following rules should be considereted when multiple board VI designed:

- hardware settings should be optimized for better performance
- if Analog Input used "PD AI Fine Tune.VI" should be used to set desired frame size (64-512 frames/buffer recommended)
- all boards/subsystem data should be processed inside single loop

PD 8 In (1 Master + 7 Slaves) 2 Out.vi

PowerDAQ analog input multiple boards high-speed real-time display example.

This VI acquires and displays up to the eight channels from eight different

PowerDAQ boards. (Up to a total of 64 PD2-MFS or 512 PD2-MF channels.). Both the CV and CL clocks can be changed.

Note The conversion (CV) clock determines the time between the pulses that go to the A/D converter. If set to 'internal', the acquisition frequency clocks and dividers controls the rate. If set to 'external', an external device can drive the PowerDAQ's conversion clock from the board's J1 connector. If set to 'continuous', the conversion clock will run at the PowerDAQ's maximum A/D sampling rate. The channel list (CL) clock sets the frequency at which scans will be made across the channel list. In other words, with each channel list clock pulse, a single scan across all the channels entered into the channel list will be executed at the CV clock rate. The 'internal' clock entry is currently not supported and so functions as if set to 'continuous'. If set to 'external', an external device can drive the PowerDAQ's channel list clock from the board's J1 connector. If set to 'continuous', the channel list clock is effectively bypassed so that conversions happen at the CV clock rate controlled by the acquisition frequency clocks and dividers.

The user can define the frame size and the buffer size will be calculated automatically. The frame size specifies the amount of data that will be processed each time by LabVIEW.

The user can enable or disable the real-time display by clicking the 'Visualization' button. This, combined with the backlog indicator, allows you to determine the amount of processing overhead available and used by this display.

Additionally, this .VI allows you to control two PowerDAQ analog output (AO) boards. The output rate that can be set is the D/A clocking rate for all the analog outputs on each board. The output rate specified is an aggregate rate. If you have 8 analog output channels and your rate is set to 160kHz, each channel will be updated at a 20kHz rate.

This .VI also takes advantage of the PowerDAQ's capability to synchronize multiple boards. It assumes that a synchronization cable is properly installed on the J6 connector of all analog input boards. This cable allows synchronization of all boards using either the CL or CV clocks with the first board being the master and the rest slaves. This particular .VI uses the CL clock to synchronize the A/D conversions.

PD 8 In 2 Out.vi

This VI is almost the same like VI above but there is no master/slave feature provided and thus no hardware connections required.

PD AI Eight boards scan to memory.vi

This VI is almost the same like VI above but there is no analog output boards support included.

The user can enable or disable the real-time display by clicking the 'Visualization' button. This, combined with the backlog indicator, allows you to determine the amount of processing overhead available and used by this display.

PD AI Four boards realtime display.VI

PowerDAQ analog input four boards high-speed real-time display example.

This VI acquires and displays up to the eight/sixty-four channels from four different PowerDAQ boards. (Up to a total of 32 PD2-MFS or 256 PD2-MF channels.).

Note That this VI uses the "

PD AI Read Async**.VI**" to retrieve the data from the acquisition buffer. The boards initialized in the sequence structure, "**PD AI Fine Tune.VI**" used to set additional acquisition parameters.

PD AI Four boards realtime display.VI

This VI is almost the same like "PD Four boards realtime display.VI" but "PD AI Read.VI" used instead of "

PD AI Read Async**.VI**". It is makes this VI easier to use but you could loose the performance of you data acquisition system.

PD AI Four boards single wave.VI

This VI is almost the same like "PD Four boards realtime display.VI" but "**PD AO Wave.VI**" used instead of "

PD AI Read Async**.VI**". This turns you boards to the start-acquire-stop mode.

PD AI Four boards stream to disk.VI

PowerDAQ analog input four boards high-speed stream to disk example.

This VI uses "PD AI Stream Init.VI" to set an additional stream-to-disk acquisition paramerers and writes to the disk using "PD AI Stream.VI" all the data acquired. This example is a good start point for the data-critical stream-to-disk applications development.

For stream-to-memory applications "PD AI Stream data into LabVIEW array.VI".

Note Check the loading of your system using Windows NT Task Manager or similar program. It's should not exceed 80-85 percent for reliable acquisition.

Note That buffer size should be at least 1.5*rate for rates between 10K and 200K and 2 to 3*rate for high frequencies. Increasing the buffer increases the stability and reliability of acquisition at high frequencies. Use hardware settings for changing default values of input limits (and thus gains).

PD AI Three boards stream to disk.VI

This VI is almost the same like VI above but there are only three PowerDAQ boards used.

PD AI Two boards stream to disk.VI

This VI is almost the same like VI above but there are only two PowerDAQ boards used.

4

Advanced Topics

Advanced Topics

Improving performance of your the data acquisition system

The overall performance and reliability of the data acquisition system is a critical factor in the development process and this factor even more critical when data acquisition system developed under the graphic-oriented visual programming language such LabVIEW® or HP VEE™ etc. In this situation a lot of system resources might by used by development or run-time environment and some difficulties to provide a system reliability can be found. The topics below describes the internals of the analog input, analog output, digital I/O and counter/timers subsystems functionality of the PowerDAQ™ board. These tips can be helpful in application development process.

Hardware settings

It is an extremely important, especially for the data-related critical application to install PowerDAQ board properly.

When starting out, first recognize that a driver for a data-acq card differs from one for a printer, CD-ROM or other peripheral in a fundamental way: realtime operation. A printer can wait a moment before it gets the next data to print; a CD-ROM can pause for a short while to let some other activity go on. A data-acquisition board, though, typically is collecting data continuously and can pause only as long as its onboard FIFO can store intermediate results. If this buffer runs over, incoming data is lost forever.

From this point there are to things to take care about when you install PowerDAQ board:

- correct interrupt settings
- PCI slot selection

The interrupts can be assigned by BIOS of your PC and if you allow it might be re-assigned during the operation system boot up process. The PowerDAQ boards able to share the same interrupt with another PCI devices but if those devises are network card or display adapter and they have a high activity on the PCI bus the collision could happens during the PCI transfers. We recommend using separate interrupt for each PowerDAQ board. If you have motherboard with Intel Advanced Interrupt Controller – just enable it in BIOS – this will allow to use more than 16 interrupt lines. If not – use manual settings to assign the interrupt to the PCI slot where PowerDAQ board installed. You can release up to four additional interrupt lines if you will disable unused ports such as COM1 and COM2, LPT and USB support.

The another problem – which PCI slot or slots can be used. Today's PC can easily contains four, five or even more PCI slots on motherboard plus probably integrated PCI devises such as network card and/or video card. Only three or four of them can work on full PCI throughput. Please refer to you mainboard manual to find out which slots can not use interrupts and do not install PowerDAQ board into it. We recommend to use single and dual processor motherboard from Intel or ASUS.

Analog Input

In most cases the main subsystem in data acquisition system is an analog input. The high-speed, accuracy and continuous non-stop acquisition is a usual requirement here. To provide all of this we will discuss the PowerDAQ™ buffering mechanism and special options which allows to improve the parameters of the data acquisition system developed/

Conversion Clock and Channel List Clock

The conversion (CV) clock determines the time between the pulses that go to the A/D converter. If set to 'internal', the acquisition frequency clocks and dividers controls the rate. If set to 'external', an external device can drive the PowerDAQ's conversion clock from the board's J1/J2 connector. If set to 'continuous', the conversion clock will run at the PowerDAQ's maximum A/D sampling rate.

The channel list (CL) clock sets the frequency at which scans will be made across the channel list. In other words, with each channel list clock pulse, a single scan across all the channels entered into the channel list will be executed at the CV clock rate. The 'internal' clock entry is currently not supported and so functions as if set to 'continuous'. If set to 'external', an external device can drive the PowerDAQ's channel list clock from the board's J1 connector. If set to 'continuous', the channel list clock is effectively bypassed so that conversions happen at the CV clock rate controlled by the acquisition frequency clocks and dividers.

Input buffers. General Information

There are a few things, which are really important to know when using the PowerDAQ™ board for high-speed analog input applications. First, what is a PowerDAQ™ driver buffer, how it is organized inside and so on. For general information about the *OMEGA ENGINEERING Advanced Circular Buffer* please refer to the Application Notes at http://www.Omega Engineeringdaq.com/support/pnote1.html .

In general the PowerDAQ™ buffer is a driver-allocated space in the host PC memory which is used to store the acquired data. The buffer is divided by some number of logical segments (16 by default for the PowerDAQ™ LabVIEW® driver). The application will be notified that new data is available only after the next frame is done. The buffer size is defined in bytes. To convert bytes into samples, the bytes should be divided by two (one sample is a one 16-bit word which is equal to the two bytes). The frame size in LabVIEW is defined in scans. One scan is a set of the samples – one for each channel in the channel list. This was done in this way because the most "PD AI xx" functions accept the number of scans as an input parameter, for the amount of the data requested. To find out the frame size the following equation should be used:

FrameSize = (((BufferSize / 2) / (Number Of The Frames))/Number Of The Acquired Channels)

Optimizing performance using the buffer settings

In a reality there are two buffers between the A/D converter on the board and the user's LabVIEW application.

The first buffer is an on-board FIFO, which is 1K/Samples by default. It can be upgraded to 8K, 16K or 32K optionally. For the two or more boards in one system or for the stream-to-disk operation 8K or 16K FIFOs are strongly recommended.

The second buffer is a PowerDAQ driver buffer, which was briefly described in a section above.

Basically the end-user applications are divided into two classes

- Critical high-speed applications without any data loss
- Scope-like applications (FFT, etc.) when the some amount of data (not all) should be processed

The PowerDAQ LabVIEW driver provides a flexible mechanism to fit any reliability and speed requirements. There is some complexity in using the driver because of its flexibility. This should be understood when comparing this driver with most LabVIEW-based drivers.

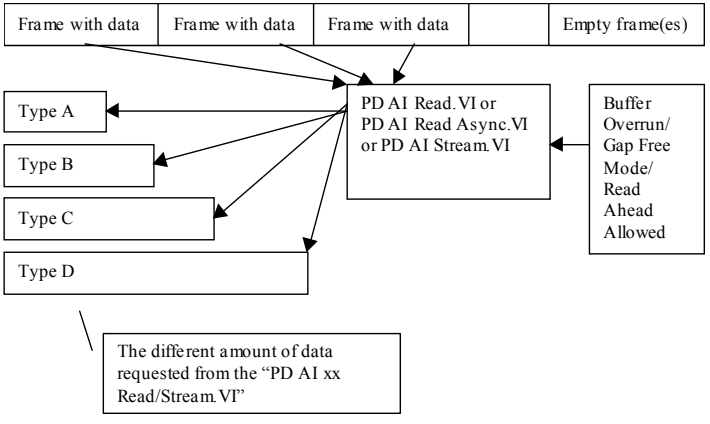
Generally the following should be considered:

- The buffer size (in bytes) should be defined using the rule: at least 1.5 times the Acquisition Rate, for rates between 10K and 200K. And, 2-4 times the Acquisition Rate for high frequencies. Increasing the buffer size increases the stability and reliability of the acquisition system at high acquisition speeds.
- The number of frames in the buffer is sixteen by default, the current Frame Size can be determined using the "PD AI Scan Size.VI"

- The Frame Size can be changed using the "**PD AI Fine Tune.VI**". The frame size should not be less than One-Half of the on-board FIFO size (in scans). This is not a required but recommended because of the internals of transferring mechanism.
- The Number Of Frames in a buffer should not be less than 4, when the high response is a requirement the frame size should be minimized and amount of frames in the buffer increased.
- There are three options which determines the buffer errors handling inside the PowerDAQ LabVIEW driver available :
  - **Gap-Free mode**. This parameter is a Boolean and dedicated to the A/D on-board FIFO buffer. When set to TRUE any on-board FIFO overrun will cause an error a stop acquisition. If this parameter set to FALSE the on-board FIFO overrun situation will restart acquisition without any error messages.

- **Buffer Overrun**. Integer parameter with four possible values. This parameter is dedicated to PowerDAQ LabVIEW driver buffer in host PC memory space and can be used depends of the application requirements
  - 0 - the buffer overrun is not allowed any buffer overrun will cause an error and stops acquisition.
  - 1 - the buffer overrun is allowed, but driver use buffer in recycle mode, when newest data can override the old one even if this data is not read yet. No errors will be generated in this situation.
  - 2 - the buffer overrun is allowed, any buffer overrun will restart acquisition.
  - 3 - buffer will be acquired only once and acquisition will be stopped.

- **Read ahead** allowed. This parameter is a Boolean and responsible for behavior of how the "PD AI xx Read[Stream].VI" retrieving the data from the PowerDAQ driver buffer. When set to ON (TRUE) "**PD AI Read**[Stream]. VI" will check how many data is inside the buffer not read yet. (To see this inside the LabVIEW the "**PD AI Data Count.VI**" provided. If more than S of the acquisition buffer is already contains the data each call of "PD AI Read[Stream]. VI" will try to retrieve two frames at time instead of one. If it OFF (FALSE) only one frame a time will be requested from the PowerDAQ driver. The picture below describes the situation



This picture shows that "PD AI Read/Stream.VI" transferring the data from the PowerDAQ driver buffer to the LabVIEW using the current acquisition settings. There are four situation can be defined depends of the amount of data requested from the VI above mentioned:

- Type A – the amount of data less than frame size. The requested amount of data will be returned to the LabVIEW and the rest of the data will be dropped – the next part of the data will be read from the next frame. The amount of unread data returned in "Scan Backlog" output. Can be used in scope-like application or FFT tasks. Note that on most applications there are no necessary to retrieve all the data.
- Type B – the amount of data is equal to the frame size. The requested amount of data will be returned to the LabVIEW and if Read Ahead Allowed is set to TRUE there is a probability of data loss if buffer becomes half full.
- Type C – the amount of data is more than one frame but not even to the frame size. The data from the consequence frames will be passed to the LabVIEW and the rest of the data in the last frame will be dropped – the next part of the data will be read from the next frame. The amount of unread data returned in "Scan Backlog" output. The Read Ahead Allowed set to TRUE could help to fill the requested data faster.
- Type D – the amount of data is even to the frame size or frame size multiplied of any factor of two. The requested amount of data will be returned to the LabVIEW.

Note That for all this modes the "Gap Free Mode" and "Buffer Overrun" options can make affect on the result achieved because for some modes the acquisition can be restarted.

Settings for the critical high–speed applications without any data loss

Buffer Size	3-6 multiply by acquisition frequency
Frame Size	not more than 24000 but we can recommend to set this value in the way when 24-36 frames will be allocated inside the acquisition buffer
"Gap Free Mode"	TRUE
"Buffer Overrun"	0 (Stop acquisition if buffer overrun detected)
"Read Ahead Allowed"	FALSE
Data to Be Read	2 multiply by frame size

Settings for the non–stop applications

Buffer Size	2-4 multiply by acquisition frequency
Frame Size	do not change or set to any reasonable value to provide required response time but it's not recommended to have a frame size bigger than 100K scans
"Gap Free Mode"	FALSE
"Buffer Overrun"	1
"Read Ahead Allowed"	TRUE
Data to Be Read	any amount but when more than 20K-50K scans requested and multiple channels are acquired the latency to create/process this data inside the LabVIEW can arise dramatically

Note That for the most applications the default driver settings are acceptable.

Analog output modes

Single Update

The PowerDAQ™ PD2-MF(S)/AO boards operate with either a single-update or streaming (waveform) output configuration. Single-update mode allows direct write access to any of the 12/16-bit DACs of the PD2- MF(S)/AO board. The update frequency is at least 1kHz for the single update mode. This single update speed is dependent on your PC system speed.

Note There is a special sub-mode of the single update mode when all DAC's outputs are updated simultaneously.

Event-based Waveform

Event-based waveform mode allows continuous waveform generation and not limited by the amount of data. The interrupt-based data requests from board will be received each time the DSP based FIFO is S full. (with 2K samples on-board FIFO, you can load a maximum of 10-24 samples at a time).

Note If the FIFO is empty or the last value is outputted, the board will continue outputting the last value.

Continuous (polled–I/O) Waveform

An alternative continuous waveform mode does not require you to use the event handling mechanism. Using polled I/O, you initialize the analog output subsystem, and write data to the output buffer (2048 samples). After the application starts, the buffer is downloaded to the DSP FIFO and the values are outputted to the DAC's

Auto–regeneration Waveform (circular waveform)

Auto-regeneration waveform mode can be used to create fixed length waveforms (2048 samples maximum) without any host PC intervention after initialization the subsystem. An application writes data to the buffer of the PD2-MF(S)/AO board and each time the end of buffer is reached, it starts to re-send the same buffer again.

PD2–AO Channel List

There are two ways do define a Channel List for the PD2-AO board. First when the output data and channel number are combined together to provide the output channel selection. In this situation, the lower 16 bits are data, the following upper 5, the channel number (see the picture below).

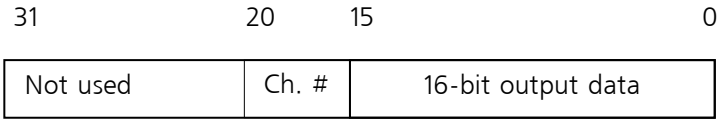


Figure 1: Channel list entries

This is the default configuration, which provides an unlimited channel list length.

Another way to define the channel list is a series of continuous channel numbers, up to the number of ports available on the AO board, starting always at channel 0. For instance, the PD2-AO-32, channels 0 through 19 can be specified as a channel list of channels to be updated in Event-based Waveform Mode and the remaining channels can be used in Single Update Mode.

Software/Hardware triggering under the LabVIEW

The general idea of the triggering of the analog input acquisition process is to keep maximum compatibility with NI version of the triggering. At the same time
PowerDAQ LabVIEW driver has some additional features such as high-speed hardware trigger.

There are two types – hardware and software – of the trigger available under the LabVIEW for the PowerDAQ boards. Also there are three different modes of the
software trigger available. Trigger type can be specified at the time when acquisition starts ("PD AI Start.VI") or during the acquisition process ("PD AI Read.VI", software analog trigger parameters only).

Trigger types

Following triggers can be used to trigger analog input acquisition process available

Hardware digital trigger

When acquisition triggered by the digital signal on the TRIG IN line.
This is a high-speed TTL trigger with a less than 1 microsecond response time.

To use this trigger customer should create a controls for the "Trigger Type" and "Trigger Channel and Level" inputs of the "PD AI START.VI" and select trigger type
4 (hardware digital trigger), specify the trigger edge for the start/stop acquisition event using the "Trigger channel" control in this VI.

Digital Channels	Hardware start trigger on:	Hardware stop trigger on:
0,0	Not used	Not used
0,1	Not used	Rising edge
0,2	Not used	Falling edge
1,0	Rising edge	Not used
1,1	Rising edge	Rising edge
1,2	Rising edge	Falling edge
2,0	Falling edge	Not used
2,1	Falling edge	Rising edge
2,2	Falling edge	Falling edge

Digital channels – should contains two entries, see table below

Internal Data Presentation for the PowerDAQ™ I, II MF(S)/AO/DIO boards

See on-line help for the "PD AI Start" for the details. See "PD AI Digital triggering.VI" for the example.

Note To insure proper operation of the Hardware stop trigger always specify conversion (CV) clock as continuous and clock acquisition using the channel list

(CL) clock. Those clocks can be also specified using the “PD AI Start.VI” – inputs “Clock source” (CV Clock) and “Channel list clock” (CL Clock).

Software analog trigger

(trigger type 1 – analog). when analog/digital input data is analyzed inside the LabVIEW driver "on the fly" and when trigger condition detected send this data to the user VI. This trigger is much more flexible and allows to use analog signal to trigger an acquisition but the response time is limited by time used to acquire one frame of the data

In this case PowerDAQ LabVIEW driver cannot detect more than one trigger occurrence per frame, so the best way to increase trigger performance is to decrease

the frame size. See We have a very good example for the software analog trigger called "PD AIn & UCT with analog trigger.VI". See this example for the details.

Note that "PD AI Fine Tune.VI" used to specify the desired frame size (see general note below)

Software digital trigger

(trigger type 2 – rising edge and 3 – fallong edge)

In this case PowerDAQ LabVIEW driver used pulled IO mode to detect desired change on the ANY of the digital input lines specifed in the "Trigger channel and level" cluster.

5

Appendix A

Appendix A: Common Questions and Support

Q What is PCI Specification 2.1?

A The PCI LocalBus is a high-performance bus that provides a processor-independent data path between the CPU and high-speed peripherals. PCI is a robust interconnect mechanism designed specifically to accommodate multiple high performance peripherals for graphics, full motion video, SCSI, LAN, etc.
The PCI Local Bus Specifications, Rev 2.1 includes the protocol, electrical, mechanical and configuration specification for the PCI Local Bus components and expansion boards.
The Rev 2.1 was published June 1, 1995 by the PCI Special Interest Group.PO Box 14070,Portland,OR 97214. Web site: www.pcisig.com

Calibration Questions

Q How often should I calibrate my board?

A The PD2-AO series board should be calibrated once a year.

Technical Support Form

Photocopy this form and update it each time you make changes to your software or hardware. Completing this form accurately before contacting us for technical support helps our application engineers answer your questions more efficiently.

What is the name and version number of the product?

What version of Windows are you using?

What programming language and version?

Is the board set at factory configuration?

_____

Have you run the board diagnostics? What were the results?

_____

Did the system ever work ? If so, what changed (moved location, installed other boards, software etc..)

Have you run the sample programs? What were the results?

_____

Have you verified that all your connections are made properly and are secure?

Have you been able to isolate the source of your problem: input or output device, board, software?

What other boards or applications are installed in your system?

How much RAM do you have?

_____

What size hard disk are you using?

_____

How fast is your CPU?

_____

How fast is your host data bus?

_____

If you are on a network, what type of network are you using and approximately how many users are on the network?

_____

Please specify whether or not the problem occurred more than once

_____

6

Appendix B

Appendix B: Warranty

Overview

IBM, IBM PC/XT/AT and IBM PS/2 are trademarks of International Business Machine Corporation.

BASIC is a trademark of Dartmouth College.

Microsoft is a trademark of Microsoft Corporation.

LabVIEW, LabWindows/CVI, DASYLab, DIADEM is a trademark of National Instruments Corporation

All PowerDAQ™ PD2-AO boards have received CE Mark certification according to the following:

- EN55011
- EN50082-1

Life Support Policy

OMEGA ENGINEERING' PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE LEGAL AFFAIRS DEPARTMENT OF OMEGA ENGINEERING CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can reasonably be expected to result in a significant injury to the user or (c) should the device or system fail to perform, may reasonably be expected to result in a significant hazard to human life, or a significant potential for injury to the user.

2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to effect its safety or effectiveness.

Omega Engineering, inc. warrants that the products furnished under this agreement will be free from material defects for a period of one year from the date of shipment. The customer shall provide notice to Omega Engineering of such defect within one week after the Customer's discovery of such defect. The sole obligation and liability of Untied Electronic Industries under this warranty shall be to repair or replace, at its option, without cost to the Customer, the product or part which is so defective and as to which such notice is given.

Upon request by Omega Engineering, the product or part claimed to be defective shall immediately be returned at the customer's expense to Omega Engineering.

There shall be no warranty or liability for any products or parts which have been subject to misuses, accident, negligence, failure or electrical power or modification by the Customer without Omega Engineering' approval. Final determination of warranty eligibility shall be made by Omega Engineering. If a warranty claim is considered invalid for any reason, the Customer will be charged for services performed and expenses incurred by Omega Engineering in handling and shipping the return item.

As to replacement parts supplied or repairs made during the original warranty period, the warranty period of the replacement or repaired part shall terminate with the termination of the warranty period with respect to the original product or part.

THE FOREGOING WARRANTY CONSTITUTES UNTIED ELECTRONICS INDUSTRIES SOLE LIABILITY AND THE CUSTOMER'S SOLE REMEDT WITH RESPECT TO THE PRODUCTS AND IS IN LIEU OF ALL OTHER WARRANTIES. LIABILITIES AND REMEDIES, EXCEPT AS THUS PROVIDED, OMEGA ENGINEERING DISCLAIMS ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Glossary

A

Asynchronous

(1) Hardware - A property of an event that occurs at an arbitrary time, without synchronization to a reference clock.

(2) Software - A property of a function that begins an operation and returns prior to the completion or termination of the operation.

B

Background Acquisition

Data is acquired by a DAQ system while another program or processing routine is running without apparent interruption.

Base Address

A memory address that serves as the starting address for programmable registers. All other addresses are located by adding to the base address.

Bit

One binary digit, either 0 or 1.

Block–Mode

A high-speed data transfer in which the address of the data is sent followed by a specified number of back-to-back data words.

Burst–Mode

A high-speed data transfer in which the address of the data is sent followed by back-to-back data words while a physical signal is asserted.

Bus

The group of conductors that interconnect individual circuitry in a computer. Typically, a bus is the expansion vehicle to which I/O or other devices are connected. Examples of PC buses are the AT PCI Bus.

Bus Master — A type of a plug-in board or controller with the ability to read and write devices on the computer bus.

Byte — Eight related bits of data, an eight-bit binary number. Also used to denote the amount of memory required to store one byte of data.

C

Cache — High-speed processor memory that buffers commonly used instructions or data to increase processing throughput.

Channel List — A variable length list of 1 to 8 channels/ports

Control Register(s) — Registers containing control bits to initiate control signals to various onboard subsystems.

Code Generator — A software program, controlled from an intuitive user interface that creates syntactically correct high-level source code in languages such as C or Basic.

Component Software — An application that contains one or more component objects that can freely interact with other component software. Examples include OLE-enabled applications such as Microsoft Visual Basic and OLE Controls for virtual instrumentation in Component Works.

Counter/Timer — A circuit that counts external pulses or clock pulses (timing), such as the Intel 8254 device.

Coupling — The manner in which a signal is connected from one location to another.

Crosstalk — An unwanted signal on one channel due to an input on a different channel.

Current Sinking The ability of a DAQ board to dissipate current for analog or digital output signals.

Current Sourcing The ability of a DAQ board to supply current for analog or digital output signals.

D

DAQ Data Acquisition

(1) Collecting and measuring electrical signals from sensors, transducers, and test probes or fixtures and inputting them to a computer for processing;

(2) Collecting and measuring the same kinds of electrical signals with A/D and/or DIO boards plugged into a PC, and possibly generating control signals with D/A and/or DIO boards in the same PC.

dB Decibel The unit for expressing a logarithmic measure of the ratio of two signal levels: dB=20log10 V1/V2, for signals in volts.

DIO Digital input/output.

DLL Dynamic Link Library A software module in Microsoft Windows containing executable code and data that can be called or used by Windows applications or other DLLs. Functions and data in a DLL are loaded and linked at run time when they are referenced by a Windows application or other DLLs.

DMA Direct Memory Access: A method by which data can be transferred to/from computer memory from/to a device or memory on the bus while the processor does something else. DMA is the

fastest method of transferring data to/from computer memory.

Drivers Software that controls a specific hardware device, such as DAQ boards.

DSP Digital signal processing.

Dual–Access Memory Memory that can be sequentially accessed by more than one controller or processor but not simultaneously accessed. Also known as shared memory.

Dual–Ported Memory Memory that can be simultaneously accessed by more than one controller or processor.

Dynamic Range The ratio of the largest signal level a circuit can handle to the smallest signal level it can handle (usually taken to be the noise level), normally expressed indB.

E

EEPROM Electrically Erasable Programmable Read-Only Memory ROM that can be erased with an electrical signal and reprogrammed.

Encoder A device that converts linear or rotary displacement into digital or pulse signals. The most popular type of encoder is the optical encoder, which uses a rotating disk with alternating opaque areas, a light source, and a photo detector.

EPROM Erasable Programmable Read-Only Memory: ROM that can be erased (usually by ultraviolet light exposure) and reprogrammed.

Events Signals or interrupts generated by a device to notify another device of an

asynchronous event. The contents of events are device-dependent.

External Trigger — A voltage pulse from an external source that triggers an event such as digital input latch signal.

F

FIFO — First-In First-Out Memory Buffer: The first data stored is the first data sent to the acceptor.

Fixed–Point — A format for processing or storing numbers as digital integers.

Floating–Point — A format for processing or storing numbers in scientific exponential notation (digits multiplied by a power of 10).

Function — A set of software instructions executed by a single line of code that may have input and/or output parameters and returns a value when executed.

G

GUI — Graphical User Interface: An intuitive, easy-to-use means of communicating information to and from a computer program by means of graphical screen displays. GUIs can resemble the front panels of instruments or other objects associated with a computer program.

H

Handler — A device driver that is installed as part of the operating system of the computer.

Hardware — The physical components of a computer system, such as the circuit boards,

plug-in boards, chassis, enclosures, peripherals, cables, and so on.

I

IMD — Intermodulation Distortion: The ratio, in dB, of the total rms signal level of harmonic sum and difference distortion products, to the overall rms signal level. The test signal is two sine waves added together according to the following standards:

Input Impedance — The measured resistance and capacitance between the input terminals of a circuit.

Input Offset Current — The difference in the input bias currents of the two inputs of an instrumentation amplifier.

Integral Control — A control action that eliminates the offset inherent in proportional control.

Interpreter — A software utility that executes source code from a high-level language such as Basic, C or Pascal, by reading one line at a time and executing the specified operation. See also Compiler.

Interrupt — A computer signal indicating that the CPU should suspend its current task to service a designated activity.

I/O — Input/Output: The transfer of data to/from a computer system involving communications channels, operator interface devices, and/or data acquisition and control interfaces.

IPC — Interprocess Communication Protocol by which processes can pass messages. Messages can be either blocks of data and information packets, or instructions and requests for pro-cess(es) to perform actions. A process can send

messages to itself, other processes on the same machine, or processes located anywhere on the network.

Isolation Voltage — The voltage that an isolated circuit can normally withstand, usually specified from input to input and/or from any input to the amplifier output, or to the computer bus.

K

k — Kilo, the standard metric prefix for 1,000, or 103, used with units of measure such as volts, hertz, and meters.

K — Kilo, the prefix for 1,024, or 210, used with B in quantifying data or computer memory.

kbytes/s — A unit for data transfer that means 1,000 or 103 bytes/s.

L

Linearity — The adherence of device response to the equation R = KS, where R =response, S = stimulus, and K = a constant.

LSB — Least significant bit.

M

M — (1) mega, the standard metric prefix for 1 million or 106, when used with units of measure such as volts and hertz;

(2) mega, the prefix for 1,048,576, or 220, when used with B to quantify data or computer memory.

Mbytes/s — A unit for data transfer that means 1 million or 106 bytes/s.

MMI — Man-Machine Interface, also Human-Machine Interface: The means by which an operator interacts with an industrial automation system; often a GUI.

Multitasking — A property of an operating system in which several processes can be run simultaneously.

N

Noise — An undesirable electrical signal. Noise comes from external sources such as the AC power line, motors, generators, transformers, fluorescent lights, soldering irons, CRT dis-plays, computers, electrical storms, welders, radio transmitters, and internal sources such as semiconductors, resistors, and capacitors.

O

OLE — Object Linking and Embedding: A set of system services that provides a means for applications to interact and interoperate. Based on the underlying Component Object Model, OLE is object-enabling system software. Through OLE Automation, an application can dynamically identify and use the services of other applications, to build powerful solutions using packaged software. OLE also makes it possible to create compound documents consisting of multiple sources of information from different applications.

OLE Controls — See ActiveX Controls.

Operating System — Base-level software that controls a computer, runs programs, interacts with users, and communicates with installed hardware or peripheral devices.

Optical Isolation The technique of using an optoelectric transmitter and receiver to transfer data without electrical continuity, to eliminate high-potential differences and transients.

Output Settling Time The amount of time required for the analog output voltage to reach its final value within specified limits.

Output Slew Rate The maximum rate of change of analog output voltage from one level to another.

Overhead The amount of computer processing resources, such as time and/or memory, required to accomplish a task.

P

Paging A technique used for extending the address range of a device to point into a larger address space

PCI Peripheral Component Interconnect: A high-performance expansion bus architecture originally developed by Intel to replace ISA and EISA. It is achieving widespread acceptance as a standard for PCs and work-stations; it offers a theoretical maximum transfer rate of 132 Mbytes/s.

PID Control A three-term control mechanism combining proportional, integral, and derivative control actions. Also see proportional control, integral control, and derivative control.

Pipeline A high-performance processor structure in which the completion of an instruction is broken into its elements so that several elements can be processed simultaneously from different instructions.

PLC Programmable logic controller: A highly reliable special-purpose computer used in industrial monitoring and control applications. PLCs typically have proprietary programming and networking protocols, and special-purpose digital and analog I/O ports.

Plug and Play ISA A specification prepared by Microsoft, Intel, and other PC-related companies that will result in PCs with plug-in boards that can be fully configured in software, without jumpers or switches on the boards.

Port A communications connection on a computer or a remote controller.

Postriggering The technique used on a DAQ board to acquire a programmed number of samples after trigger conditions are met.

Potentiometer An electrical device the resistance of which can be manually adjusted; used for manual adjustment of electrical circuits and as a transducer for linear or rotary position.

Pretriggering The technique used on a DAQ board to keep a continuous buffer filled with data, so that when the trigger conditions are met, the sample includes the data leading up to the trigger condition.

Programmed I/O The standard method a CPU uses to access an I/O device-- each byte of data is read or written by the CPU.

Propagation Delay The amount of time required for a signal to pass through a circuit. Proportional

Control A control action with an output that is to be proportional to the deviation of

the controlled variable from a desired set point.

Protocol — The exact sequence of bits, characters and control codes used to transfer data between computers and peripherals through a communications channel, such as the GPIB.

Q

Quantization Error — The inherent uncertainty in digitizing an analog value due to the finite resolution of the conversion process.

R

Real Time — A property of an event or system in which data is processed as it is acquired in-stead of being accumulated and processed at a later time.

Resource Locking — A technique whereby a device is signaled not to use its local memory while the memory is in use from the bus.

Ribbon Cable — A flat cable in which the conductors are side by side.

RTD — Resistance Temperature Detector: A metallic probe that measures temperature based upon its coefficient of resistivity.

S

SE — Single-Ended: A term used to describe an analog input that is measured with respect to a common ground.

Self-Calibrating — DAQ board that calibrates its own A/D and D/A circuits with and external reference source.

Sensor — A device that responds to a physical stimulus (heat, light, sound, pressure, motion, flow, and so on), and produces a corresponding electrical signal.

S/H — Sample-and-Hold: A circuit that acquires and stores an analog voltage/digital value on a capacitor for a short period of time.

SNR — Signal-to-Noise Ratio**:** The ratio of the overall rms signal level to the rms noise level, expressed in dB.

Software Trigger — A programmed event that triggers an event such as data acquisition.

SPDT — Single-Pole Double Throw: A property of a switch in which one terminal can be connected to one of two other terminals.

SSH — Simultaneous Sampling and Hold: A property of a system in which each input or output channel is digitized or updated at the same instant.

S/s — Samples per second; used to express the rate at which a DAQ board samples an analog signal.

Strain Gauge — A sensor whose resistance is a function of the applied force.

Subroutine — A set of software instructions executed by a single line of code that may have input and/or output parameters.

Synchronous — A property of a function that begins an operation and returns only when the operation is complete.

T

TCP/IP — A set of standard protocols for communicating across a single network

	or interconnected set of networks. The Internet Protocol (IP) for the low-level service of taking data and packaging of components, and Transmission Control Protocol (TCP) for high-reliability data transmissions.
THD	Total Harmonic Distortion: The ratio of the total rms signal due to harmonic distortion to the overall rms signal, in dB or percent.
THD+N	Signal-to-THD Plus Noise: The ratio in decibels of the overall rms signal to the rms signal of harmonic distortion plus noise introduced.
Thermistor	A semiconductor sensor that exhibits a repeatable change in electrical resistance as a function of temperature. Most thermistors exhibit a negative temperature coefficient.
Thermocouple	A temperature sensor created by joining two dissimilar metals. The junction produces a small voltage as a function of the temperature.
Throughput Rate	The data, measured in bytes/s, for a given continuous operation.
Transducer	A device that responds to a physical stimulus (heat, light, sound, pressure, motion, flow, and so on), and produces a corresponding electrical signal.
Transfer Rate	The rate, measured in bytes/s, at which data is moved from source to destination after software initialization and set up operations; the maximum rate at which the hardware can operate.

U

Unipolar A signal range that is always positive (for example, 0 to +10 V).

Z

Zero-Overhead Looping The ability of a high-performance processor to repeat instructions without requiring time to branch to the beginning of the instructions.

Zero-Wait-State Memory Memory fast enough that the processor does not have to wait during any reads and writes to the memory.

Index

Reader Evaluation

We are committed to improving the quality of our documentation, in order to serve you better. Your feedback will help us in the effort. Thanks for taking the time to fill out and return this form.

Is the manual well organized? ☐ Yes ☐ No

Can you find information easily? ☐ Yes ☐ No

Were you able to install the PD2-AO boards? ☐ Yes ☐ No

Did you find any technical errors? ☐ Yes ☐ No

Is the manual size appropriate? ☐ Yes ☐ No

Are the design, type style, and layout attractive? ☐ Yes ☐ No

Is the quality of illustrations satisfactory? ☐ Yes ☐ No

Is the quality of illustrations satisfactory? ☐ Yes ☐ No

Is the quality of illustrations satisfactory? ☐ Yes ☐ No

Is the quality of illustrations satisfactory? ☐ Yes ☐ No

How would you rate this manual? ☐ Excellent ☐ Good ☐ Fair ☐ Poor

Why?_____

_____

Suggested improvements:

_____

Other Comments:

_____

_____

Your background (optional)

Your application:

_____