

NuDAQ®  
DAQ-2010  
14-bit, 4-CH, 2MS/s  
Simultaneous A/D Multi-function Card  
User's Guide



Recycled Paper



©Copyright 2000~2001 ADLINK Technology Inc.

All Rights Reserved.

Manual Rev. 1.00: September 07, 2001

P/N: 50-11020-100

The information in this document is subject to change without prior notice in order to improve reliability, design and function and does not represent a commitment on the part of the manufacturer.

In no event will the manufacturer be liable for direct, indirect, special, incidental, or consequential damages arising out of the use or inability to use the product or documentation, even if advised of the possibility of such damages.

This document contains proprietary information protected by copyright. All rights are reserved. No part of this manual may be reproduced by any mechanical, electronic, or other means in any form without prior written permission of the manufacturer.

### **Trademarks**

NuDAQ<sup>®</sup>, NuIPC<sup>®</sup>, NuDAM<sup>®</sup>, NuPRO<sup>®</sup> are registered trademarks of ADLINK Technology Inc. Other product names mentioned herein are used for identification purposes only and may be trademarks and/or registered trademarks of their respective companies.

# Getting service from ADLINK

- Customer Satisfaction is always the most important thing for ADLINK Tech Inc. If you need any help or service, please contact us and get it.

ADLINK Technology Inc.			
Web Site	http://www.adlinktechnology.com http://www.adlink.com.tw		
Sales & Service	service@adlink.com.tw		
Technical Support	NuDAQ + USBDAQ	nudaq@adlink.com.tw	
	NuDAM	nudam@adlink.com.tw	
	NuIPC	nuipc@adlink.com.tw	
	NuPRO	nupro@adlink.com.tw	
	Software	sw@adlink.com.tw	
TEL	+886-2-82265877	FAX	+886-2-82265717
Address	9F, No. 166, Jian Yi Road, Chungho City, Taipei, 235 Taiwan, R.O.C.		

- Please inform or FAX us of your detailed information for a prompt, satisfactory and constant service.

Detailed Company Information			
Company/Organization			
Contact Person			
E-mail Address			
Address			
Country			
TEL		FAX	
Web Site			
Questions			
Product Model			
Environment to Use	OS: _____ Computer Brand: _____ M/B: _____ CPU: _____ Chipset: _____ BIOS: _____ Video Card: _____ Network Interface Card: _____ Other: _____		
Challenge Description			
Suggestions for ADLINK			

# Table of Contents

<b>Chapter 1 Introduction</b> .....	<b>1</b>
1.1 Features .....	1
1.2 Applications .....	3
1.3 Specifications.....	3
1.4 Software Support .....	7
1.4.1 <i>Programming Library</i> .....	7
1.4.2 <i>PCIS-LVIEW: LabVIEW<sup>®</sup> Driver</i> .....	7
1.4.3 <i>DAQBench<sup>™</sup>: ActiveX Controls</i> .....	8
<b>Chapter 2 Installation</b> .....	<b>9</b>
2.1 What You Have .....	9
2.2 Unpacking .....	10
2.3 DAQ-2010 Layout .....	11
2.4 PCI Configuration .....	12
<b>Chapter 3 Signal Connections</b> .....	<b>13</b>
3.1 Connectors Pin Assignment .....	13
3.2 Analog Input Signal Connection .....	16
3.2.1 <i>Types of signal sources</i> .....	16
3.2.2 <i>Single-Ended Measurements</i> .....	16
3.2.3 <i>Differential Measurements</i> .....	17
<b>Chapter 4 Operation Theorem</b> .....	<b>19</b>
4.1 A/D Conversion .....	19
4.1.1 <i>AD Transfer Modes</i> .....	19
4.1.1.1 <i>Software Polling</i> .....	20
4.1.1.2 <i>Bus-mastering DMA</i> .....	21
4.1.2 <i>AD Trigger Modes</i> .....	23
4.1.2.1 <i>Post-Trigger Acquisition</i> .....	23
4.1.2.2 <i>Delay Trigger Acquisition</i> .....	23

	4.1.2.3	<i>Pre-Trigger Acquisition</i> .....	24
	4.1.2.4	<i>Middle-Trigger Acquisition</i> .....	25
	4.1.2.5	<i>Post-Trigger or Delay-trigger Acquisition with Retrigger</i> .....	25
4.2		D/A Conversion .....	27
	4.2.1	<i>Software Update</i> .....	27
	4.2.2	<i>Timed Waveform Generation</i> .....	27
	4.2.2.1	<i>Post-Trigger Generation</i> .....	29
	4.2.2.2	<i>Delay-Trigger Generation</i> .....	29
	4.2.2.3	<i>Repetitive Waveform Generation</i> .....	30
	4.2.2.4	<i>Delay2 in Repetitive Waveform Generation</i> .....	30
	4.2.2.5	<i>Retriggered Waveform Generation</i> .....	31
	4.2.2.6	<i>Stop Modes of Scan Update</i> .....	31
4.3		Digital I/O .....	33
4.4		General Purpose Timer/Counter Operation .....	34
	4.4.1	<i>Timer/Counter functions basics</i> .....	34
	4.4.2	<i>General Purpose Timer/Counter modes</i> .....	34
	4.4.2.1	<i>Mode1: Simple Gated-Event Counting</i> .....	35
	4.4.2.2	<i>Mode2: Single Period Measurement</i> .....	35
	4.4.2.3	<i>Mode3: Single Pulse-width Measurement</i> .....	36
	4.4.2.4	<i>Mode4: Single Gated Pulse Generation</i> .....	36
	4.2.2.5	<i>Mode5: Single Triggered Pulse Generation</i> .....	37
	4.2.2.6	<i>Mode6: Re-triggered Single Pulse Generation</i> .....	37
	4.2.2.7	<i>Mode7: Single Triggered Continuous Pulse Generation</i> .....	38
	4.2.2.8	<i>Mode8: Continuous Gated Pulse Generation</i> .....	38
4.5		Triggers .....	39
	4.5.1	<i>Software-Trigger</i> .....	39
	4.5.2	<i>External Analog Trigger</i> .....	39
	4.5.2.1	<i>Below-Low analog trigger condition</i> .....	40
	4.5.2.2	<i>Above-High analog trigger condition</i> .....	40
	4.5.2.3	<i>Inside-Region analog trigger condition</i> .....	41
	4.5.2.4	<i>High-Hysteresis analog trigger condition</i> .....	42
	4.5.2.5	<i>Low-Hysteresis analog trigger condition</i> .....	42
	4.5.3	<i>External Digital Trigger</i> .....	43
4.6		Timing Signals .....	44

4.6.1	<i>Auxiliary Function Inputs</i> .....	44
4.6.2	<i>System Synchronization Interface</i> .....	45
<b>Chapter 5</b>	<b>Calibration</b> .....	<b>46</b>
5.1	Loading Calibration Constants .....	46
5.2	Auto-calibration .....	47
5.3	Saving Calibration Constants .....	47
<b>Product Warranty/Service</b>	.....	<b>48</b>

# How to Use This Guide

This manual is designed to help you to use the DAQ-2010. The manual describes the versatile functions and the operation theorem of the DAQ-2010. It is divided into six chapters:

- Chapter 1,** Introduction gives an overview of the product features, applications, and specifications.
- Chapter 2,** Installation describes how to install the DAQ-2010. The layout of DAQ-2010, including the positions of connectors, the connectors' pin assignments are specified.
- Chapter 3,** Signal Connections describes the connector of the DAQ-2010, and the signal connection between the DAQ-2010 and external devices. The layout of DAQ-2010, including the positions of connectors, the connectors' pin assignments are specified.
- Chapter 4,** Operation Theorem describes how to operate DAQ-2010. The A/D, D/A, GPIO, timer/counter, trigger and timing signal routing are introduced.
- Chapter 5,** Calibration describes how to calibrate the DAQ-2010 for accurate measurement.

# 1

## Introduction

The DAQ-2010 is an advanced data acquisition card based on the 32-bit PCI architecture. High performance designs and the state-of-the-art technology make this card ideal for data logging and signal analysis applications in medical, process control, etc.

---

### 1.1 Features

The DAQ-2010 Advanced Data Acquisition Card provides the following advanced features:

- 32-bit PCI-Bus, plug and play
- 4-channel simultaneous analog inputs
- 14-bit ADC with sampling rate up to 2MHz
- Programmable bipolar/unipolar analog input
- Programmable gain of x1, x2, x4, x8
- AD FIFO size: 8K samples
- Versatile trigger sources: software trigger, external digital trigger, analog trigger and trigger from System Synchronization Interface (SSI).
- AD Data transfer: software polling & bus-mastering DMA with Scatter/Gather functionality
- Four A/D trigger modes: post-trigger, delay-trigger, pre-trigger and middle-trigger
- 2 channel DA outputs with waveform generation capability

- 1K output data FIFO per DA channel
- DA Data transfer: software update and bus-mastering DMA with Scatter/Gather functionality
- System Synchronization Interface (SSI)
- AD/DA fully auto-calibration
- Completely jumper-less and software configurable

---

## 1.2 Applications

- Automotive Testing
- Cable Testing
- Transient signal measurement
- ATE
- Laboratory Automation
- Biotech measurement

---

## 1.3 Specifications

### ◆ Analog Input (AI)

- **Number of channels:** 4 differential
- **AD converter:** LTC1414 or equivalent
- **Max sampling rate:** 2MS/s
- **Resolution:** 14 bits
- **FIFO buffer size:** 8K samples
- **Programmable input range:**  
Bipolar:  $\pm 10V$ ,  $\pm 5V$ ,  $\pm 2.5V$ ,  $\pm 1.25V$   
Unipolar:  $0\sim 10V$ ,  $0\sim 5V$ ,  $0\sim 2.5V$ ,  $0\sim 1.25V$
- **Overvoltage protection:** continuous  $\pm 35V$  maximum
- **3dB bandwidth:** 400KHz
- **Input impedance:**  $> 10\text{ M}\Omega$
- **Time-base source:** Internal 40MHz or External clock Input (fmax.: 40MHz, fmin.: 1MHz, 50% duty cycle)
- **Trigger modes:** post-trigger, delay-trigger, pre-trigger and middle-trigger
- **Data transfers:** Programmed I/O, and bus-mastering DMA with scatter/gather
- **Data throughput:** 4MB/s for single channel enabled, maximum 16MB/s for four channels enabled
- **Input coupling:** DC

- **Offset error:**  
Before calibration:  $\pm 60\text{mV}$  max  
After calibration:  $\pm 3\text{mV}$  max
- **Gain error :**  
Before calibration:  $\pm 0.6\%$  of output max  
After calibration:  $\pm 0.03\text{mV}$  of output max
- ◆ **Analog Output (AO)**
  - **Number of channels:** 2 voltage
  - **DA converter:** LTC7545 or equivalent
  - **Max update rate:** 1MS/s
  - **Resolution:** 12 bits
  - **FIFO buffer size:** 1K samples per channel, when both channels are enabled for timed output, and 2K samples when only one channel is used for timed DA output
  - **Data transfers:** Programmed I/O, and bus-mastering DMA with scatter/gather
  - **Output range:**  
Bipolar:  $\pm 10\text{V}$   
Unipolar: 0~10V
  - **Settling time for full-scale step:**  $2\mu\text{S}$
  - **Slew rate:**  $20\text{V}/\mu\text{S}$
  - **Output coupling:** DC
  - **Protection:** Short-circuit to ground
  - **Output impedance:**  $0.1\Omega$ . max.
  - **Output driving:**  $\pm 5\text{mA}$  max.
  - **Power-on state:** 0V steady-state
  - **Power-on glitch:**  $\pm 600\text{mV}/500\mu\text{S}$
  - **Offset error :**  
Before calibration:  $\pm 80\text{mV}$  max  
After calibration:  $\pm 2\text{mV}$  max

- **Gain error :**
  - Before calibration:  $\pm 0.8\%$  of output max
  - After calibration:  $\pm 0.02\%$  of output max
- ◆ **General Purpose Digital I/O (G.P. DIO, 82C55A)**
  - **Number of channels:** 24 programmable Input/Output
  - **Compatibility:** TTL/CMOS
  - **Input voltage:**
    - Logic Low:  $V_{IL}=0.8\text{ V max.}; I_{IL}=0.2\text{mA max.}$
    - High:  $V_{IH}=2.0\text{V max.}; I_{IH}=0.02\text{mA max}$
  - **Output voltage:**
    - Low:  $V_{OL}=0.5\text{ V max.}; I_{OL}=8\text{mA max.}$
    - High:  $V_{OH}=2.7\text{V min}; I_{OH}=400\mu\text{A}$
- ◆ **Simultaneous Digital I/O (SDI)**
  - **Number of channels:** 8 inputs (2 inputs per AI channel)
  - **Compatibility:** TTL/CMOS
  - **Input voltage:**
    - Logic Low:  $V_{IL}=0.8\text{ V max.}; I_{IL}=0.2\text{mA max.}$
    - High:  $V_{IH}=2.0\text{V max.}; I_{IH}=0.02\text{mA max}$
- ◆ **General Purpose Timer/Counter (GPTC)**
  - **Number of channel:** 2 Up/Down Timer/Counters
  - **Resolution:** 16 bits
  - **Compatibility:** TTL
  - **Clock source:** Internal or external
  - **Max source frequency:** 10MHz
- ◆ **Analog Trigger (A.Trig)**
  - **Source:** All analog input channels; external analog trigger (EXTATRIG)
  - **Level:**  $\pm$ Full-scale, internal;  $\pm 10\text{V}$  external
  - **Resolution:** 8 bits
  - **Slope:** Positive or negative (software selectable)

- **Hysteresis:** Programmable
- **Bandwidth:** 400khz
- **External Analog Trigger Input (EXTATRIG)**  
 Impedance: 40KO  
 Coupling: DC  
 Protection: Continuous  $\pm 35V$  maximum
- ◆ **Digital Trigger (D.Trig)**
  - **Compatibility:** TTL/CMOS
  - **Response:** Rising or falling edge
  - **Pulse Width:** 10ns min
- ◆ **System Synchronous Interface (SSI)**
  - **Trigger lines:** 7
- ◆ **Stability**
  - **Recommended warm-up time:** 15 minutes
  - **On-board calibration reference:**  
 Level: 5.000V  
 Temperature coefficient:  $\pm 2\text{ppm}/^\circ\text{C}$   
 Long-term stability: 6ppm/1000Hr
- ◆ **Physical**
  - **Dimension:** 175mm by 107mm
  - **I/O connector:** 68-pin female VHDCI type (e.g. AMP-787254-1)
- ◆ **Power Requirement**
  - **+5VDC:** 1.82 A (typical)
- ◆ **Operating Environment**
  - **Ambient temperature:** 0 to 55°C
  - **Relative humidity:** 10% to 90% non-condensing
- ◆ **Storage Environment**
  - **Ambient temperature:** -20 to 70°C
  - **Relative humidity:** 5% to 95% non-condensing

---

## 1.4 Software Support

ADLINK provides versatile software drivers and packages for users' different approach to built-up a system. We not only provide programming library such as DLL for many Windows systems, but also provide drivers for other software package such as LabVIEW®.

All the software options are included in the ADLINK CD. The non-free software drivers are protected with serial licensed code. Without the software serial number, you can still install them and run the demo version for two hours for demonstration purpose. Please contact ADLINK dealer to purchase the formal license serial code.

### 1.4.1 Programming Library

For customers who are writing their own programs, we provide function libraries for many different operating systems, including:

- **D2K-DASK** : Include device drivers and DLL for **Windows 98**, **Windows NT** and **Windows 2000**. DLL is binary compatible across Windows 98, Windows NT and Windows 2000. That means all applications developed with D2K-DASK are compatible across Windows 98, Windows NT and Windows 2000. The developing environment can be VB, VC++, Delphi, BC5, or any Windows programming language that allows calls to a DLL. The user's guide and function reference manual of D2K-DASK are in the CD. (\\Manual\_PDF\\Software\\D2K-DASK)
- **D2K-DASK/X** : Include device drivers and shared library for Linux. The developing environment can be Gnu C/C++ or any programming language that allows linking to a shared library. The user's guide and function reference manual of D2K-DASK/X are in the CD. (\\Manual\_PDF\\Software\\D2K-DASK-X.)

### 1.4.2 D2K-LVIEW: LabVIEW® Driver

D2K-LVIEW contains the VIs, which are used to interface with NI's LabVIEW® software package. The D2K-LVIEW supports Windows 98/NT/2000. The LabVIEW® drivers are free shipped with the board. You can install and use them without license. For detailed information about D2K-LVIEW, please refer to the user's guide in the CD.

(\\Manual\_PDF\\Software\\D2K-LVIEW)

The above software drivers are shipped with the board. Please refer to the "**Software Installation Guide**" in the package to install these drivers.

### **1.4.3 DAQBench™: ActiveX Controls**

We suggest the customers who are familiar with ActiveX controls and VB/VC++ programming use DAQBench™ ActiveX Control components library for developing applications. DAQBench™ is designed for Windows 98/NT/2000. For more detailed information about DAQBench, please refer to the user's guide in the CD.

(\\Manual\_PDF\Software\DAQBench\DAQBench Manual.PDF)

The above software is charged software. Please contact ADLINK dealer or ADLINK to purchase the software license.

# 2

## Installation

This chapter describes how to install the DAQ-2010. At first, the contents in the package and unpacking information that you should be careful of are described.

The DAQ-2010 performs an automatic configuration of the IRQ, and port address. Users can use software utility, PCI\_SCAN to read the system configuration.

---

### 2.1 What You Have

In addition to this *User's Guide*, the package includes the following items:

- DAQ-2010 Multi-function Data Acquisition Card
- ADLINK All-in-one Compact Disc
- Software Installation Guide

If any of these items is missing or damaged, contact the dealer from whom you purchased the product. Save the shipping materials and carton in case you want to ship or store the product in the future.

---

## 2.2 Unpacking

Your DAQ-2010 card contains sensitive electronic components that can be easily damaged by static electricity.

The card should be handled on a grounded anti-static mat. The operator should be wearing an anti-static wristband, grounded at the same point as the anti-static mat.

Inspect the card module carton for obvious damage. Shipping and handling may cause damage to your module. Be sure there are no shipping and handling damages on the module before processing.

After opening the card module carton, extract the system module and place it only on a grounded anti-static surface with component side up.

Again inspect the module for damage. Press down on all the socketed IC's to make sure that they are properly seated. Do this only with the module place on a firm flat surface.

---

**Note :** DO NOT APPLY POWER TO THE CARD IF IT HAS BEEN DAMAGED.

---

***You are now ready to install your DAQ-2010.***

---

## 2.3 DAQ-2010 Layout

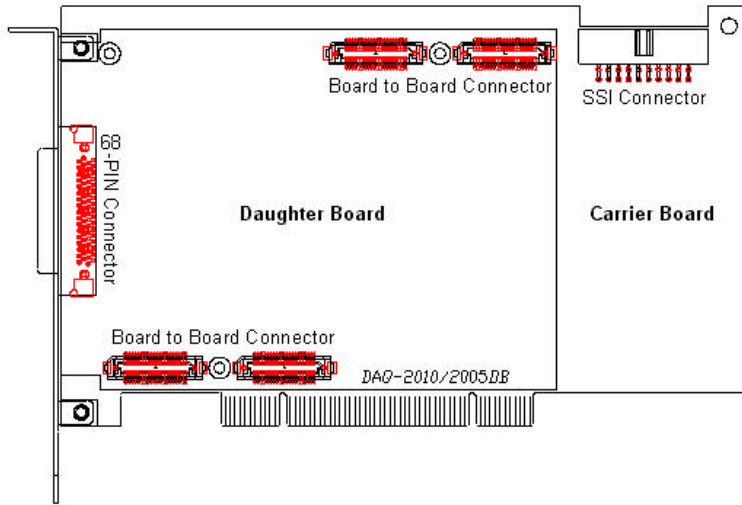


Figure 2.1 PCB Layout of the DAQ-2010

---

## 2.4 PCI Configuration

### 1. Plug and Play:

As a plug and play component, the board requests an interrupt number via its PCI controller. The system BIOS responds with an interrupt assignment based on the board information and on known system parameters. These system parameters are determined by the installed drivers and the hardware load seen by the system.

### 2. Configuration:

The board configuration is done on a board-by-board basis for all PCI boards on your system. Because configuration is controlled by the system and software, so there is no jumpers for base-address, DMA, and interrupt IRQ need to be set by the user.

The configuration is subject to change with every boot of the system as new boards are added or boards are removed. So, there is no idea what's going on to be installed.

### 3. Trouble shooting:

If your system doesn't boot or if you experience erratic operation with your PCI board in place, it's likely caused by an interrupt conflict (perhaps because you incorrectly configured BIOS setup). In general, the solution, once you determine it is not a simple oversight, is to consult the BIOS documentation that come with your system.

# 3

## Signal Connections

This chapter describes the connector of the DAQ-2010, and the signal connection between the DAQ-2010 and external devices.

---

### 3.1 Connectors Pin Assignment

The DAQ-2010 is equipped with one 68-pin VHDCI-type connector (AMP-787254-1). It is used for digital signal input / output, analog input / output, and timer/counter's signals, etc. The pin assignment of the connector is illustrated in the Figure 3.1.

CH0+	1	35	CH0-
CH1+	2	36	CH1-
CH2+	3	37	CH2-
CH3+	4	38	CH3-
EXTATRIG	5	39	AIGND
DA1OUT	6	40	AOGND
DA0OUT	7	41	AOGND
AOEXTREF	8	42	AOGND
SDI3_1	9	43	SDI3_0
SDI2_1	10	44	SDI2_0
SDI1_1	11	45	SDI1_0
SDI0_1	12	46	SDI0_0
AO_TRIG_OUT	13	47	EXTWFTRG
AI_TRIG_OUT	14	48	EXTDTRIG
GPTC1_SRC	15	49	DGND
GPTC0_SRC	16	50	DGND
GPTC0_GATE	17	51	GPTC1_GATE
GPTC0_OUT	18	52	GPTC1_OUT
GPTC0_UPDOWN	19	53	GPTC1_UPDOWN
EXTTIMEBASE	20	54	DGND
AFI1	21	55	AFI0
PB7	22	56	PB6
PB5	23	57	PB4
PB3	24	58	PB2
PB1	25	59	PB0
PC7	26	60	PC6
PC5	27	61	PC4
DGND	28	62	DGND
PC3	29	63	PC2
PC1	30	64	PC0
PA7	31	65	PA6
PA5	32	66	PA4
PA3	33	67	PA2
PA1	34	68	PA0

**Figure 3.1 Connector pin assignment**

**Legend :**

Pin #	Signal Name	Reference	Direction	Description
1~4	CH<0..3>+	CH0<0..3>-	Input	Differential positive input for AI channel <0..3>
5	EXTATRIG	AIGND	Input	External AI analog trigger
6	DA0OUT	AOGND	Output	AO channel 0
7	DA1OUT	AOGND	Output	AO channel 1
8	AOEXTREF	AOGND	Input	External reference for AO channels
9~12	SDI<3..0>_1	DGND	Input	Simultaneous DI channel 1 for AI channel <0..3>
13	AO_TRIG_OUT	DGND	Output	AO trigger signal
14	AI_TRIG_OUT	DGND	Output	AI trigger signal
15,16	GPTC<0,1>_SRC	DGND	Input	Source of GPTC<0,1>
17,51	GPTC<0,1>_GATE	DGND	Input	Gate of GPTC<0,1>
18,52	GPTC<0,1>_OUT	DGND	Input	Output of GPTC<0,1>
19,53	GPTC<0,1>_UPDOWN	DGND	Input	Up/Down of GPTC<0,1>
20	EXTTIMEBASE	DGND	Input	External Timebase
21,28,49,50,54,62	DGND	-----	-----	Digital ground
22,56,23,57,24,58,25,59	PB<7,0>	DGND	PIO*	Programmable DIO of 8255 Port B
26,60,27,61,29,63,30,64	PC<7,0>	DGND	PIO*	Programmable DIO of 8255 Port C
31,65,32,66,33,67,34,68	PA<7,0>	DGND	PIO*	Programmable DIO of 8255 Port A
35~38	CH<0..3>-	-----	Input	Differential negative input for AI channel <0..3>
39	AIGND	-----	-----	Analog ground for AI
40~42	AOGND	-----	-----	Analog ground for AO
43~46	SDI<3..0>_0	DGND	Input	Simultaneous DI channel 0 for AI channel <0..3>
47	EXTWFTRIG	DGND	Input	External AO waveform trigger
48	EXTDTRIG	DGND	Input	External AI digital trigger
21	AFI1	DGND	Input	Auxiliary Function Input 1 (ADCONV, AD_START)
55	AFI0	DGND	Input	Auxiliary Function Input 0 (DAWR, DA_START)

\*PIO means programmable I/O

---

## 3.2 Analog Input Signal Connection

The DAQ-2010 provides 4 differential analog input channels. The analog signal can be converted to digital value by the A/D converter. To avoid ground loops and get more accurate measurement of A/D conversion, it is quite important to understand the signal source type and how to connect the analog input signals.

### 3.2.1 Types of signal sources

#### *Ground-Referenced Signal Sources*

A ground-referenced signal means it is connected in some way to the building system. That is, the signal source is already connected to a common ground point with respect to the DAQ-2010, assuming that the computer is plugged into the same power system. Non-isolated outputs of instruments and devices that plug into the building power system are ground-referenced signal sources.

#### *Floating Signal Sources*

A floating signal source means it is not connected in any way to the building ground system. A device with an isolated output is a floating signal source, such as optical isolator output, transformer output, and thermocouples.

### 3.2.2 Single-Ended Measurements

The single-ended connection is that the analog input signals are referenced to the common ground of the system. In this case, all the negative ends of analog input channels should be connected to the AIGND on the connector instead of floating. Please refer to the figure 3.3

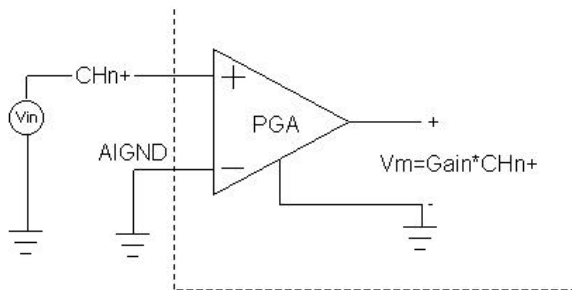


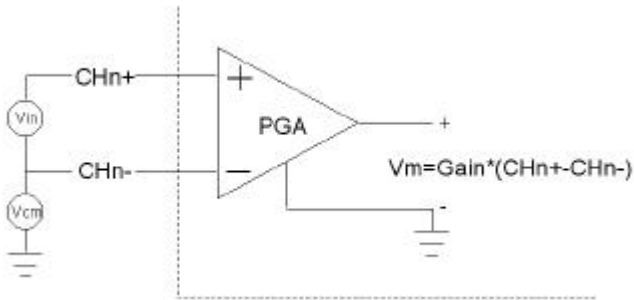
Figure 3.3 Single-Ended connections

In single-ended configurations, more electrostatic and magnetic noise couples into the single connections than in differential configurations. Therefore, the single-ended connection is not recommended unless fewer wire connections are necessary.

### 3.2.3 Differential Measurements

#### *Differential Connection for Grounded-Reference Signal Sources*

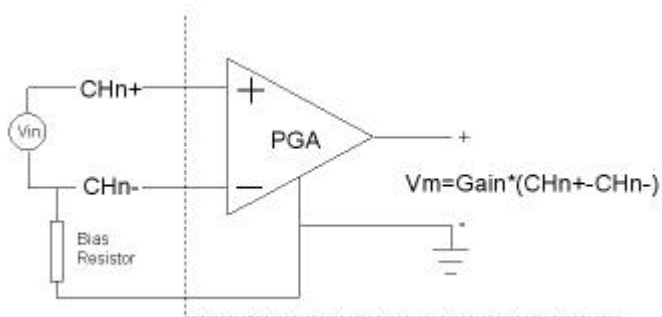
The differential analog input provides two inputs that respond to the signal voltage difference between them. If the signal source is ground-referenced, the differential mode can be used for the common-mode noise rejection. Figure 3.3 shows the connection of ground-referenced signal sources under the differential input mode.



**Figure 3.3** Ground-referenced source and differential input

### ***Differential Connection for Floating Signal Sources***

Fig3.4 shows how to connect a floating signal source to DAQ-2010 in differential input mode. For floating signal sources, you need to add a resistor to provide a bias return path. The resistor value should be about 100 times the equivalent source impedance. If the source impedance is less than 100ohms, you can simply connect the negative side of the signal to AGND as well as the negative input of the Instrumentation Amplifier, without any resistors at all. In differential input mode, less noise couples into the signal connections than in single-ended mode.



**Figure 3.4 Floating source and differential input**

# 4

## Operation Theorem

The operation theorem of the functions on DAQ-2010 is described in this chapter. The functions include the A/D conversion, D/A conversion, Digital I/O and General Purpose Counter / Timer. The operation theorem can help you understand how to manipulate and program the DAQ-2010.

---

### 4.1 A/D Conversion

When using an A/D converter, users should know about the properties of the signal to be measured at first. Users can decide which channel to use and connect the signals to the DAQ-2010. Please refer to 3.2. In addition, users should define and control the A/D signal configurations, including channels, gains, and A/D signal types.

The A/D acquisition is initiated by a trigger source, users must decide how to trigger the A/D conversion. The data acquisition will start when a trigger condition is matched.

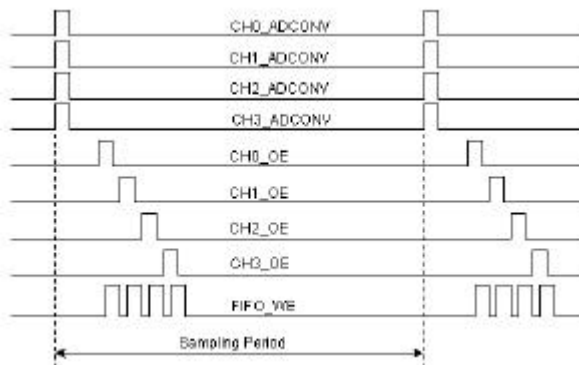
After the end of A/D conversion, the A/D data is buffered in a Data FIFO. The A/D data should be transferred into PC's memory for further processing.

#### 4.1.1 AD Transfer Modes

Two of AD data transfer modes: Software Polling and Bus-Mastering DMA are described separately in the following.

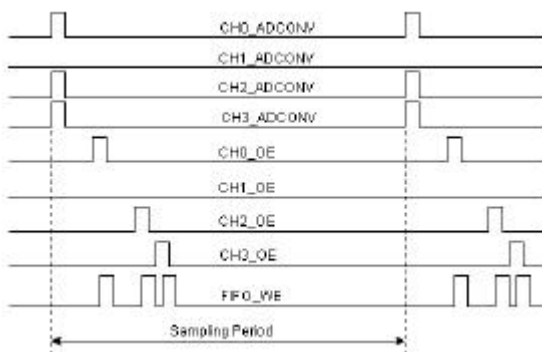
### 4.1.1.1 Software Polling

This is the easiest way to acquire A/D data. As figure 4.1, after the four A/D converters are initialized by software conversion synchronously, the converted digital data will be saved to the AD Data FIFO sequentially by asserting CHn\_OE (Channel Output Enable) and FIFO\_WE (FIFO Write Enable) one after another. That means users can acquire four synchronous data by reading the AD Data FIFO then.



**Figure 4.1.1 Timing of AD Sampling**

In figure 4.1.1, all the four channels are enabled. Each channel obtains 2K FIFO buffer size. Any of the channels can be disabled and the enabled channels will share the 8K equally, for example of figure 4.1.2, AD channel 1 is disabled. The AD data of channel 0, 2, 3 will be save to the AD Data FIFO sequentially. Each enabled channel obtains 2.66K buffer size.



**Figure 4.1.2 Timing of AD Sampling with channel 1 disabled**

This method is very suitable for the application that needs to process AD data in real time. Under this mode, the timing of the A/D conversion is fully controlled by software. However, it is difficult to control the fixed A/D conversion rate.

#### **4.1.1.2 Bus-mastering DMA**

PCI bus-mastering DMA is necessary for high speed DAQ in order to utilize the maximum PCI bandwidth. The bus-mastering controller, which is built-in into the PLX IOP-480 PCI controller, controls the PCI bus when it becomes the master of the bus. Bus mastering reduces the size of on-board memory and reduces the CPU loading because data is directly transferred to the computer's memory without host CPU intervention.

Bus-mastering DMA provides the fastest data transfer rate on PCI-bus. Once the analog input operation starts, control returns to your program. The hardware temporarily stores the acquired data in the onboard AD Data FIFO and then transfers the data to a user-defined DMA buffer memory in the computer. Please note that even when the acquired data length is less than the Data FIFO, the AD data will not be kept in the Data FIFO but directly transferred into host memory by bus-mastering DMA.

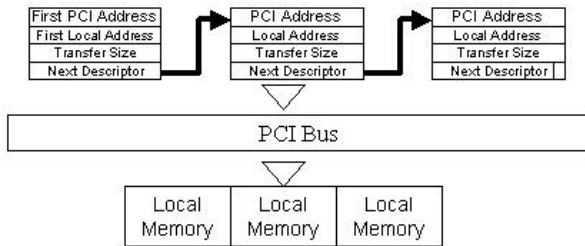
The DMA transfer mode is very complex to program. We recommend using a high-level program library to manipulate this card. If you want to program the software that can handle the DMA bus master data transfer, please refer to more information about the PCI controller. ([www.plxtech.com](http://www.plxtech.com))

By using a high-level program library for high speed DMA data acquisition, users simply need to assign the sampling period and the number of conversion into the specific counters. After the AD trigger condition is matched, the data will be transferred to the system memory by bus-mastering DMA.

The PCI controller also supports the function of scatter/gather bus mastering DMA, which helps the users to transfer a large amount of data by linking the all memory blocks into a continuous linked list.

In the multi-user or multi-tasking OS, like Microsoft Windows, Linux, and so on, it is difficult to allocate a large continuous memory block to do the DMA transfer. Therefore, the PLX IOP-480 provides the function of scatter/gather or chaining mode DMA to link the non-continuous memory blocks into a linked list so that users can transfer a very large amount of data without limiting by the fragment of small size memory. Users can configure the linked list for the input DMA channel or the output DMA channel. The figure 4.1.3 shows the linked list that is constructed by three DMA descriptors. Each descriptor contains a PCI address, a local address, a transfer size, and the pointer to the next descriptor. Users can allocate many small size memory blocks and chain their associative DMA de-

scriptors altogether by their application programs. The DAQ-2010 software driver provides the easy settings of the scatter/gather function, and some sample programs are also provided within the ADLINK all-in-one CD.



**Figure 4.1.3 Scatter/gather DMA for digital output**

In non-chaining mode, the maximum DMA data transfer size is 2M double words (8M bytes). However, by using chaining mode, scatter/gather, there is no limitation on DMA data transfer size. Users can also link the descriptors nodes circularly to achieve a multi-buffered mode DMA.

## 4.1.2 AD Trigger Modes

There are 5 trigger modes: software-trigger, pre-trigger, post-trigger, middle-trigger, and delay-trigger to start the data acquisition. All but software trigger are external triggers. They are described as follows.

### 4.1.2.1 Post-Trigger Acquisition

Use post-trigger acquisition in applications when you want to collect data after a trigger event. The number of scans after the trigger is specified in SC\_counter, as illustrated in fig 4.1.4.

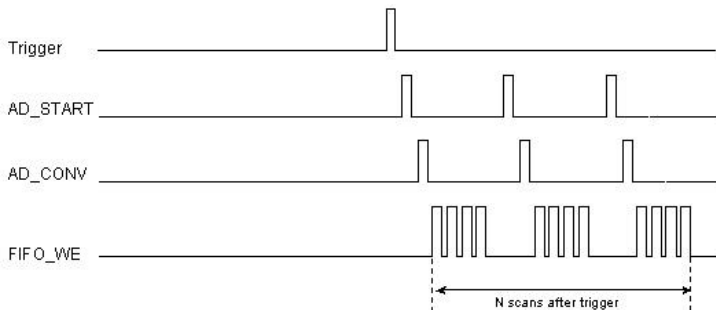
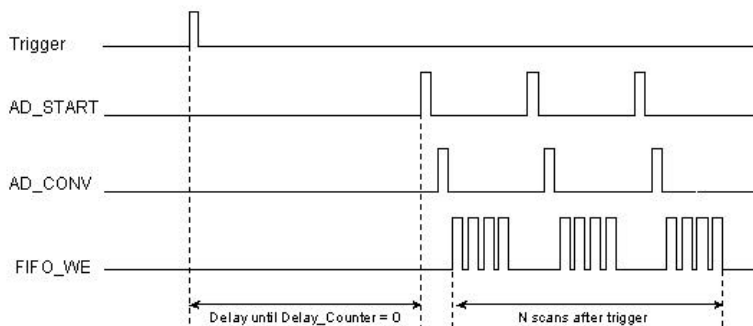


Figure 4.1.4 Post trigger

### 4.1.2.2 Delay Trigger Acquisition

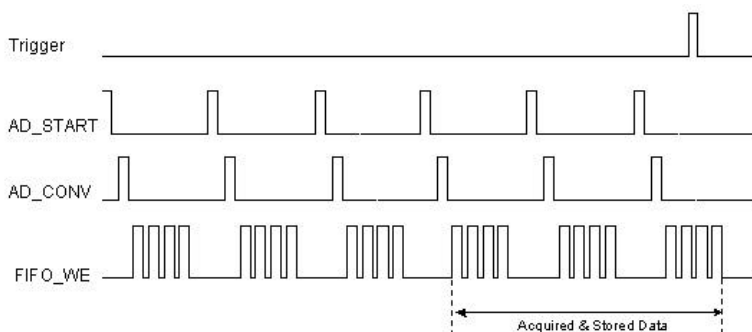
Use delay trigger acquisition in applications when you want to delay the data collection after the occurrence of a specified trigger event. The delay time is controlled by the value that is pre-loaded in the **Delay\_counter**. Then the counter counts down on the rising edge of Delay\_counter clock source after the trigger condition is met. When the count reaches 0, the counter stops and DAQ-2010 starts to acquire data. This Delay counter is 16-bit wide and users could set the delay time in the unit of Timebase or in the unit of sampling period, such that the delay time could reach a wide range.



**Figure 4.1.5 Delay trigger**

### 4.1.2.3 Pre-Trigger Acquisition

Use pre-trigger acquisition in applications when you want to collect data before a trigger event. The A/D starts when you execute the specified function calls to begin the operation, and it stops when the trigger event occurs. Users could program the value  $M$  in **M\_counter** to specify the amount of conversion before the trigger event. If the external trigger occurs after  $M$  scans of data are converted, the program only stores the last  $M$  scans of data, as illustrated in fig.4.1.6, where  $M\_counter = 3$



**Figure 4.1.6 Pre-trigger**

When the trigger signal occurs before the first  $M$  data are converted, the amount of stored data could be fewer than the originally specified. Users could choose which kind of acquisition is preferred by software setting, to let the trigger signal be ignored until the first  $M$  scans of data are converted,

or stop the acquisition immediately. Thus the function could assure that users can get M scans of data pre-trigger mode.

#### 4.1.2.4 Middle-Trigger Acquisition

Use middle-trigger acquisition in applications when you want to collect data before and after a trigger event. The number of scans before the trigger is specified in M\_counter, while the number of scans after the trigger is specified in SC\_counter.

Similar to the pre-trigger mode, users could choose which kind of acquisition is preferred by software setting, to let the trigger signal be ignored until the first M scans of data are converted, or stop the acquisition immediately. Thus the function could assure that users can get M+N scans of data in the middle-trigger mode.

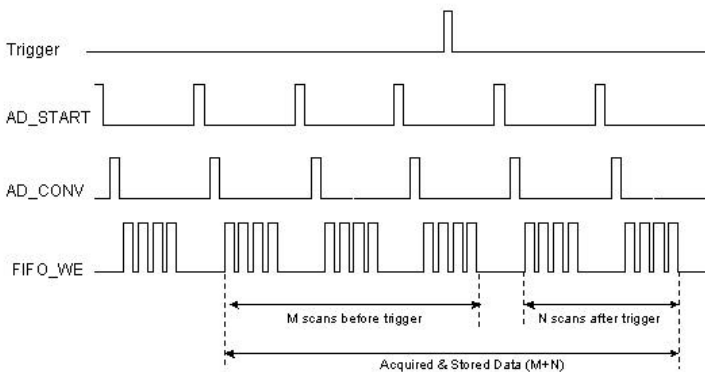
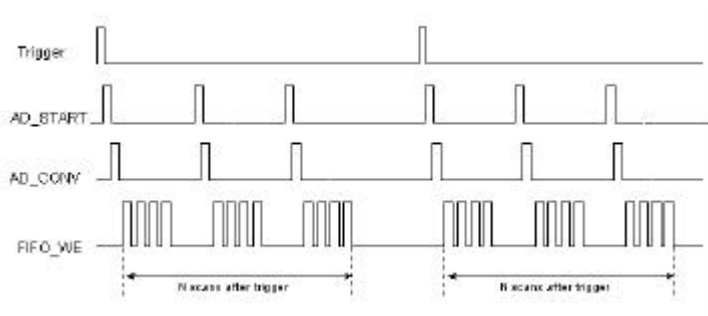


Figure 4.1.7 Middle trigger

#### 4.1.2.5 Post-Trigger or Delay-trigger Acquisition with Retrigger

Use post-trigger or delay-trigger acquisition with retrigger function in applications when you want to collect data after several trigger events. The number of scans after each trigger is specified in SC\_counter, and users could program to specify the retrigger numbers. Fig.4.1.8 illustrates an example. In this example, 3 scans of data is acquired after the first trigger signal, then the board waits for the 2<sup>nd</sup> trigger signal. When the trigger signal occurs, the board acquires 3 scans of data again. The process repeats until specified amount of retrigger signals are detected. Note that the retrigger function only supports the post-trigger mode and the delay-trigger mode.



**Figure 4.1.8 Post trigger with re-trigger**

---

## 4.2 D/A Conversion

There are 2 channels D/A output available in the DAQ-2010. When using D/A converters, users should define and control the D/A converter reference sources, D/A operation mode and D/A channels. Users could also select the output polarity: unipolar or bipolar.

The reference selection control lets users fully utilize the multiplying characteristics of the D/A converters. Internal 10V reference and external reference input are available in the DAQ-2010. The full range of the D/A output is directly related to the reference. The digital codes that are updated to the D/A converters will multiply the reference to generate the analog output. While using internal 10V reference, the full range would be  $-10V \sim +10V$  in the bipolar output mode, and  $0V \sim 10V$  in the unipolar output mode. While using the external reference, users could reach different output full ranges by connecting different references. For example, if connecting a DC  $-4.096V$  with the external reference, then users could get the full range  $-4.096V$  to  $+4.096V$  in the bipolar output with inverting characteristics due to the negative reference voltage. Users could also reach amplitude modulation (AM) output by feeding a sinusoidal signal into the reference input.

The D/A conversion is initiated by a trigger source. Users must decide how to trigger the D/A conversion. The data output will start when a trigger condition is met. Before the start of D/A conversion, D/A data is transferred from PC's main memory and buffered in a Data FIFO.

Two of the D/A conversion modes: Software Update and Timed Waveform Generation are described in the following, including the timing, trigger source control, trigger modes and data transfer methods. **Either way may be applied to D/A channels independently.**

### 4.2.1 Software Update

This is the easiest way to generate D/A output. First, users should specify D/A output channels, set output polarity: unipolar or bipolar, and reference source: internal 10V or external AOEXTREF. Then update digital values into D/A data registers through a software output command.

### 4.2.2 Timed Waveform Generation

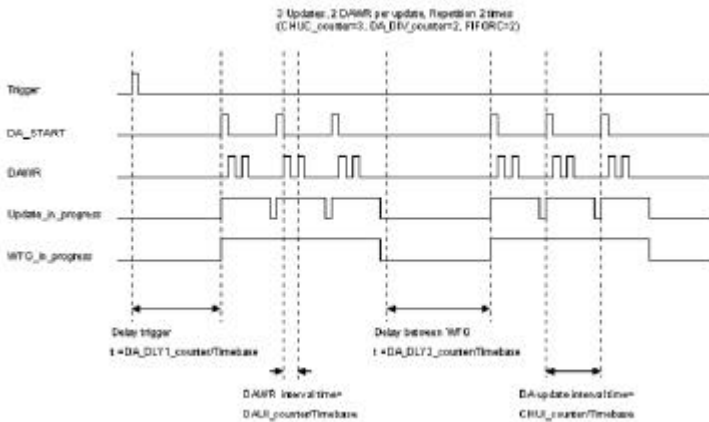
This mode could provide your applications a precise D/A output with a fixed update rate. It could be used to generate an infinite or finite waveform. You can accurately program the update period of the D/A converters.

The D/A output timing is provided through a combination of counters in the FPGA on board. There are totally 7 counters to be specified, but users

don't have to know how to program these counters in detail as long as you use the software library we provide. These counters are:

- CHUI\_counter(24 bits): specify the DA Update Interval =  $\text{CHUI\_counter}/\text{Timebase}$ .
- DAUI\_counter(16 bits): specify the D/A converter Update Interval =  $\text{DAUI\_counter}/\text{Timebase}$ .
- CHUC\_counter(24 bits): specify the number of total update count.
- DA\_DIV\_counter(9 bits): specify the number of DAWR per update.
- FIFORC\_counter(24 bits): specify the repetitioncount number of update start.
- DA\_DLY1\_counter(16 bits): specify the delay from the trigger to the first update start.
- DA\_DLY2\_counter(16 bits): specify the delay between two consecutive waveform generations.

Figure 4.2.1 shows the typical D/A timing diagram.



**Figure 4.2.1 Typical D/A timing of waveform generation**

There are two trigger modes for D/A outputs: post-trigger mode and pre-trigger mode.

### 4.2.2.1 Post-Trigger Generation

Figure 4.2.2 shows a infinite continuous waveform generation after a trigger signal is detected, exactly the same meaning as in the A/D acquisition. The trigger signal could come from a software command, the analog trigger or the digital trigger. Please refer to the section 4.5 for detailed information.

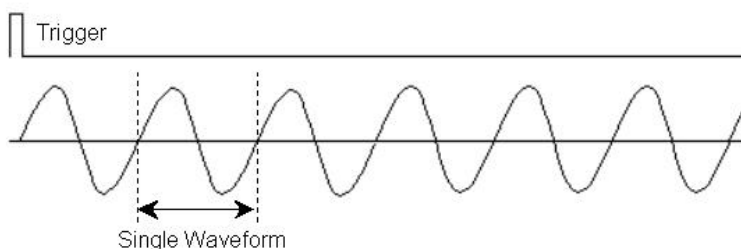


Figure 4.2.2 continuous waveform generation

### 4.2.2.2 Delay-Trigger Generation

In figure 4.2.3, DA\_DLY1\_counter determines the time of Delay\_1, the delay time after the trigger signal to the start of the waveform generation. The delay time is controlled by the value that is pre-loaded in the **Delay1\_counter**. Then the counter counts down on the rising edge of Delay1\_counter clock source after the trigger condition is met. When the count reaches 0, the counter stops and DAQ-2010 starts the waveform generation. This Delay1 counter is 16-bit wide and users could set the delay time in the unit of Timebase or in the unit of update period, such that the delay time could reach a wide range

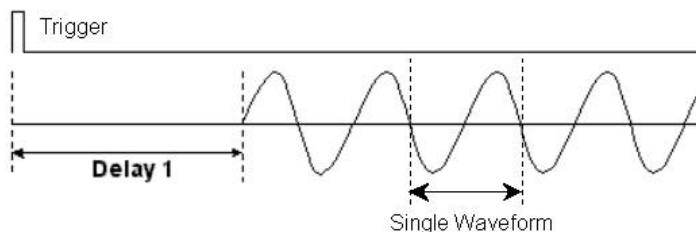


Figure 4.2.3 continuous waveform generation with Delay1

### 4.2.2.3 Repetitive Waveform Generation

Users could use the FIFO on board to store the necessary data for repetitive waveform generation, and then initialize the waveform generation by re-transmitting the data in the FIFO (performed by FPGA) to achieve the repetitive waveform generation. The FIFORC counter stores the repetition number, and the repetition could be finite or infinite. In figure 4.2.4, the value of FIFORC counter is set to be 4, while one cycle of the sinewave-like waveform is composed of the data stored in the FIFO.

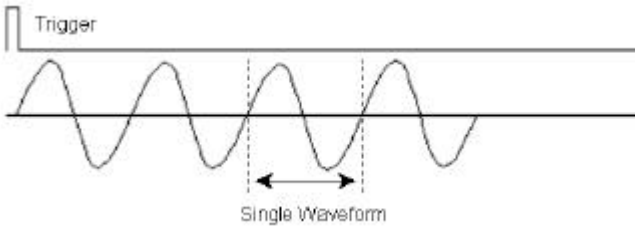


Figure 4.2.4 finite waveform generation

### 4.2.2.4 Delay2 in Repetitive Waveform Generation

To stretch out the flexibility of the D/A waveform generation, we add a Delay2 counter to separate 2 consecutive waveforms in repetitive waveform generation. The time between two waveforms is assigned by setting the value of Delay2 counter. The Delay2 counter counts down after a complete waveform generation, and when it counts down to zero, the next waveform generation will start, just as shown in figure 4.2.5. This Delay2 counter is 16-bit wide and users could set the delay time in the unit of Timebase or in the unit of update period, such that the delay time could reach a wide range

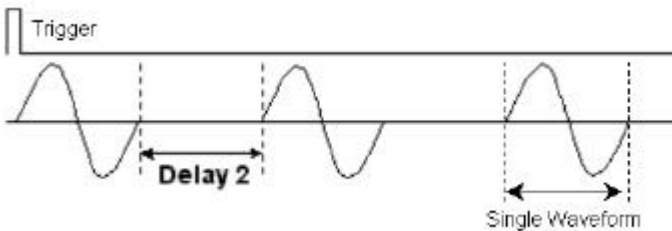


Figure 4.2.5 continuous waveform generation with Delay2

#### 4.2.2.5 Retriggered Waveform Generation

The retrigger function of waveform generation can be enabled or disabled by software setting. In figure 4.2.6, in retrigger mode, each trigger signal will initiate a waveform generation. However, the trigger signal inserts when the previous waveform generation is still ongoing would be ignored.

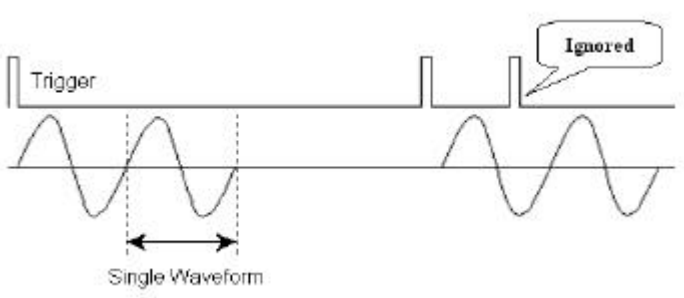


Figure 4.2.6 Retriggered waveform generation

#### 4.2.2.6 Stop Modes of Scan Update

Three stop modes are provided for timed waveform generation, which means the when to stop the waveform generation. Users can stop the waveform generation immediately after the stop command being received, stop after a completion of waveform generation, or stop after a completion of repetition of waveform generation.

In figure 4.2.7, a waveform generation with FIFORC counter=2 and Delay2 counter enabled stops DA output immediately when stop command asserts. In figure 4.2.8, the waveform stops after a complete waveform is finished. And in figure 4.2.9, the waveform generator stops only after the complete repetition of waveform is finished.

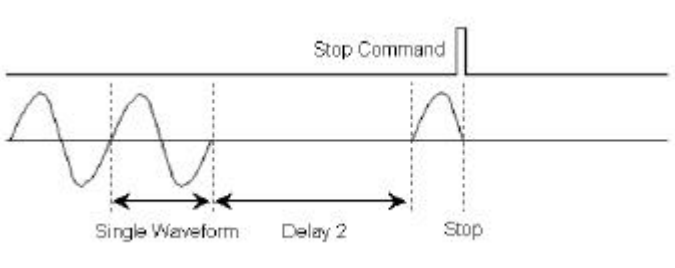
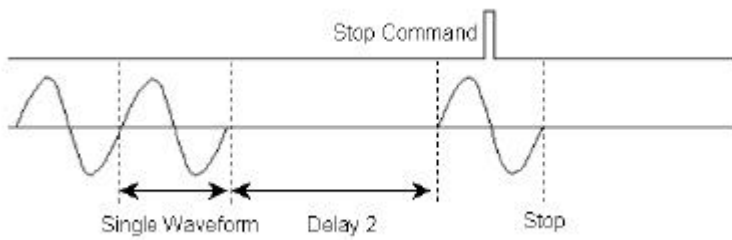
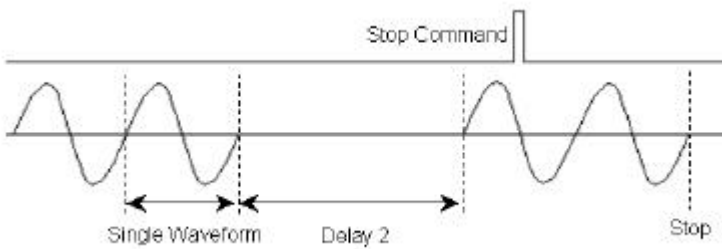


Figure 4.2.7 Stop mode I



**Figure 4.2.8 Stop mode II**



**Figure 4.2.9 Stop mode III**

---

### 4.3 Digital I/O

The DAQ-2010 contains 24-lines of general-purpose digital I/O (GPIO), which is provided through a 82C55A chip, and 8 lines of synchronous digital inputs (SDI).

The 24-lines GPIO are separated into three ports: Port A, Port B and Port C. Port A, Port B, Port C high nibble (bit-4 to bit-7), and low nibble (bit 0 to bit 3) can be programmed to be input or output individually. At system startup and reset, all the I/O pins are all reset to be input configuration, that is, high impedance.

When each AD conversion is completed, 2 lines of SDI will be saved to the register accompanied with the 14-bit AD output data as shown as figure 4.3.1. That means users can simultaneously sample one analog signal and two digital signals. And totally, 4 analog signals and 8 digital signals can be sampled simultaneously.

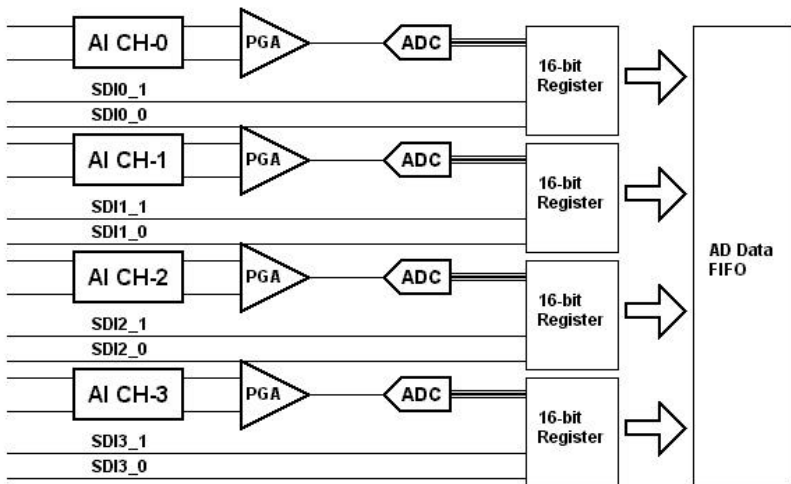


Figure 4.3.1 Synchronous digital inputs

---

## 4.4 General Purpose Timer/Counter Operation

Two independent 16-bit up/down timer/counter are designed within FPGA for various applications. They have the following features:

- Count up/down controlled by hardware or software
- Programmable counter clock source (internal or external clock up to 10MHz)
- Programmable gate selection (hardware or software control)
- Programmable input and output signal polarities (high active or low active)
- Initial Count can be loaded from software
- Current count value can be read-back by software without affecting circuit operation

### 4.4.1 Timer/Counter functions basics

Each timer/counter has three inputs that can be controlled via hardware or software. They are clock input (GPTC\_CLK), gate input (GPTC\_GATE), and up/down control input (GPTC\_UPDOWN). The GPTC\_CLK input provides a clock source input to the timer/counter. Active edges on the GPTC\_CLK input make the counter increment or decrement. The GPTC\_UPDOWN input controls whether the counter counts up or down. The GPTC\_GATE input is a control signal which acts as a counter enable or a counter trigger signal under different applications.

The output of timer/counter is GPTC\_OUT. After power-up, GPTC\_OUT is pulled high by a pulled-up resistor about 10K ohms. Then GPTC\_OUT goes low after DAQ-2010 initialization.

All the polarities of input/output signals can be programmed by software. In this chapter, all the figures of timing assume that GPTC\_CLK, GPTC\_GATE, and GPTC\_OUT are set to be high active or rising-edge trigger.

### 4.4.2 General Purpose Timer/Counter modes

Eight programmable timer/counter modes are provided. All the modes start operations following the software-start signal that is set by software. The GPTC software reset initializes the status of the counter and re-loads the initial value to the counter. The operation remains stop until the software-start is re-executed. The operating theorems under different modes are described as follows.

#### 4.4.2.1 Mode1: Simple Gated-Event Counting

In this mode, the counter counts the number of pulses on the GPTC\_CLK after the software-start. Initial count can be loaded from software. Current count value can be read-back by software any time without affecting the counting. GPTC\_GATE is used to enable/disable counting. When GPTC\_GATE is inactive, the counter halts the current count value. Figure 4.4.1 illustrates the operation with initial count = 5, count-down mode.

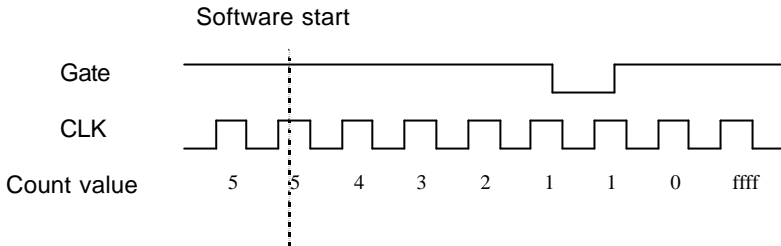


Figure 4.4.1 Mode 1 Operation

#### 4.4.2.2 Mode2: Single Period Measurement

In this mode, the counter counts the period of the signal on GPTC\_GATE in terms of GPTC\_CLK. Initial count can be loaded from software. After the software-start, the counter counts the number of active edges on GPTC\_CLK between two active edges of GPTC\_GATE. After the completion of the period interval on GPTC\_GATE, GPTC\_OUT outputs high and then current count value can be read-back by software. Figure 4.4.2 illustrates the operation where initial count = 0, count-up mode.

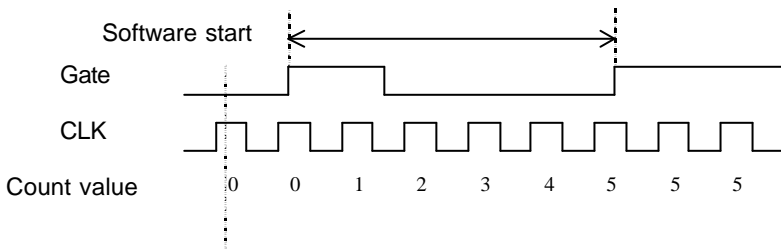


Figure 4.4.2 Mode 2 Operation

### 4.4.2.3 Mode3: Single Pulse-width Measurement

In this mode the counter counts the pulse-width of the signal on GPTC\_GATE in terms of GPTC\_CLK. Initial count can be loaded from software. After the software-start, the counter counts the number of active edges on GPTC\_CLK when GPTC\_GATE is in its active state. After the completion of the pulse-width interval on GPTC\_GATE, GPTC\_OUT outputs high and then current count value can be read-back by software. Figure 4.4.3 illustrates the operation where initial count = 0, count-up mode.

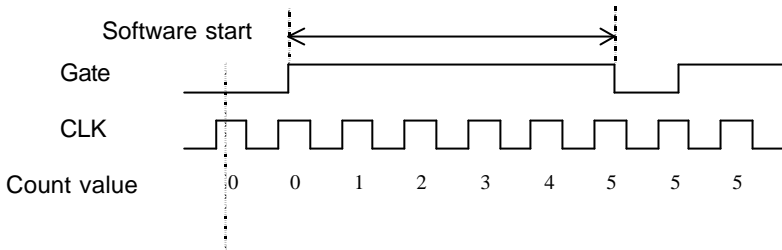


Figure 4.4.3 Mode 3 Operation

### 4.4.2.4 Mode4: Single Gated Pulse Generation

This mode generates a single pulse with programmable delay and programmable pulse-width following the software-start. The two programmable parameters could be specified in terms of periods of the GPTC\_CLK input by software. GPTC\_GATE is used to enable/disable counting. When GPTC\_GATE is inactive, the counter halts the current count value. Figure 4.4.4 illustrates the generation of a single pulse with a pulse delay of two and a pulse-width of four.

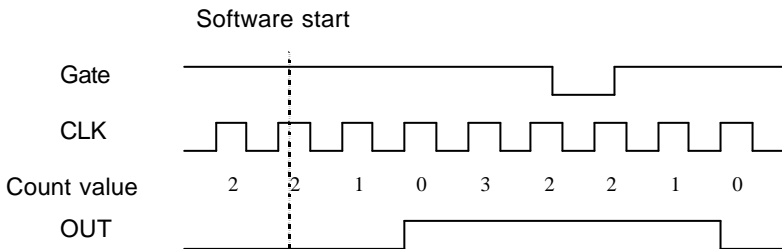


Figure 4.4.4 Mode 4 Operation

#### 4.2.2.5 Mode5: Single Triggered Pulse Generation

This function generates a single pulse with programmable delay and programmable pulse-width following an active GPTC\_GATE edge. You could specify these programmable parameters in terms of periods of the GPTC\_CLK input. Once the first GPTC\_GATE edge triggers the single pulse, GPTC\_GATE takes no effect until the software-start is re-executed. Figure 4.4.5 illustrates the generation of a single pulse with a pulse delay of two and a pulse-width of four.

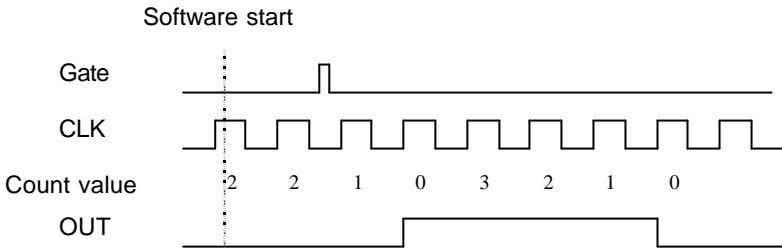


Figure 4.4.5 Mode 5 Operation

#### 4.2.2.6 Mode6: Re-triggered Single Pulse Generation

This mode is similar to mode5 except that the counter generates a pulse following every active edge of GPTC\_GATE. After the software-start, every active GPTC\_GATE edge triggers a single pulse with programmable delay and pulse-width. GPTC\_GATE trigger that occurs when the prior pulse is not completed would be ignored. Figure 4.4.6 illustrates the generation of two pulses with a pulse delay of two and a pulse-width of four.

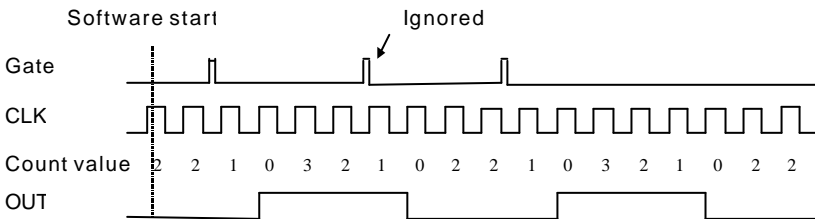


Figure 4.4.6 Mode 6 Operation

### 4.2.2.7 Mode7: Single Triggered Continuous Pulse Generation

This mode is similar to mode5 except that the counter generates continuous periodic pulses with programmable pulse interval and pulse-width following the first active edge of GPTC\_GATE. Once the first GPTC\_GATE edge triggers the counter, GPTC\_GATE takes no effect until the software-start is re-executed. Figure 4.4.7 illustrates the generation of two pulses with a pulse delay of four and a pulse-width of three.

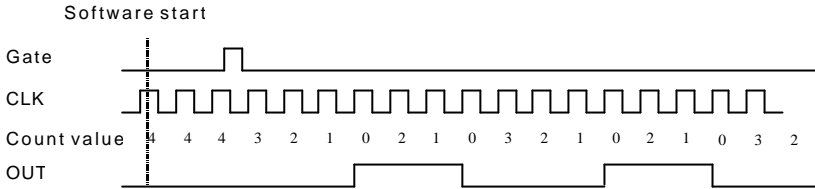


Figure 4.4.7 Mode 7 Operation

### 4.2.2.8 Mode8: Continuous Gated Pulse Generation

This mode generates periodic pulses with programmable pulse interval and pulse-width following the software-start. GPTC\_GATE is used to enable/disable counting. When GPTC\_GATE is inactive, the counter halts the current count value. Figure 4.4.8 illustrates the generation of two pulses with a pulse delay of four and a pulse-width of three.

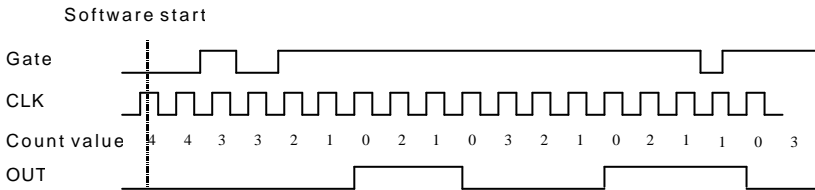


Figure 4.4.8 Mode 8 Operation

---

## 4.5 Triggers

We provide flexible trigger selection in the DAQ-2000 series products. In addition to the internal software trigger, DAQ-2010 also supports external analog and digital triggers. Users can configure the trigger source by software for A/D and D/A processes individually. **Note that the A/D and the D/A conversion share the same analog trigger.**

### 4.5.1 Software-Trigger

This trigger mode does not need any external trigger source. The trigger asserts right after you execute the specified function calls to begin the operation. A/D and D/A processes could receive an individual software trigger.

### 4.5.2 External Analog Trigger

The analog trigger circuitry routing is shown in the figure 4.5.1. The analog multiplexer could select either a direct analog input from the EXTATRIG pin in the 68-pin connector or a signal from any of the input channels. The trigger level range for the EXTATRIG is  $\pm 10V$  and the resolution is 78mV, while the trigger level range of the selected AI channel is the full-scale range of the selected AI channel, and the resolution is the desired range divided by 256.

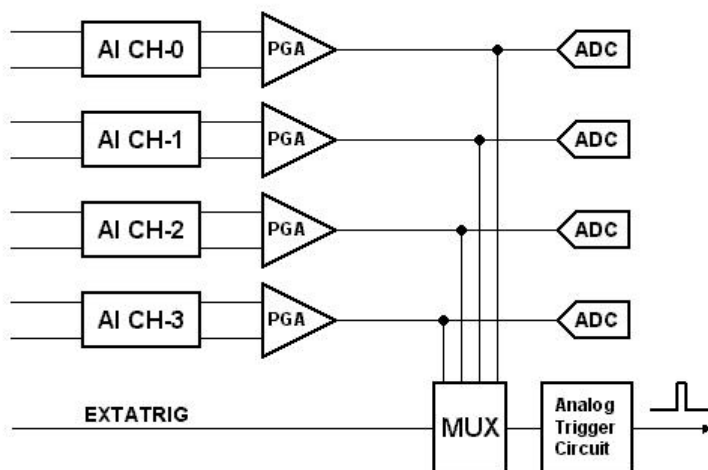
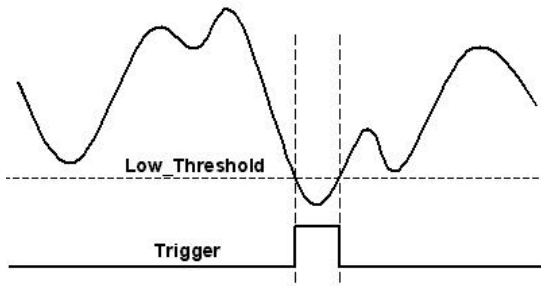


Figure 4.5.1 Analog trigger block diagram

The trigger signal is generated when the analog trigger condition is satisfied. There are five analog trigger conditions in the DAQ-2010. The DAQ-2010 uses 2 threshold voltages, Low\_Threshold and High\_Threshold to build the 5 different trigger conditions. Users could configure the trigger conditions easily by software.

#### **4.5.2.1 Below-Low analog trigger condition**

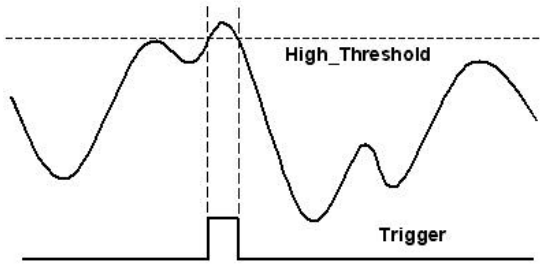
Figure 4.5.2 shows the below-low analog trigger condition, the trigger signal is generated when the input analog signal is less than the Low\_Threshold voltage, and the High\_Threshold setting is not used in this trigger condition.



**Figure 4.5.2 Below-Low analog trigger condition**

#### **4.5.2.2 Above-High analog trigger condition**

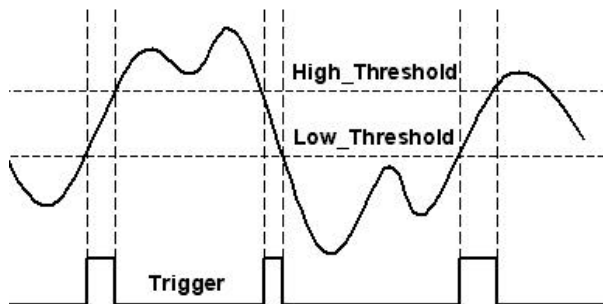
Figure 4.5.3 shows the above-high analog trigger condition, the trigger signal is generated when the input analog signal is higher than the High\_Threshold voltage, and the Low\_Threshold setting is not used in this trigger condition.



**Figure 4.5.3 Above-High analog trigger condition**

#### **4.5.2.3 Inside-Region analog trigger condition**

Figure 4.5.4 shows the inside-region analog trigger condition, the trigger signal is generated when the input analog signal level falls in the range between the High\_Threshold and the Low\_Threshold voltages.



**Figure 4.5.4 Inside-Region analog trigger condition**

#### 4.5.2.4 High-Hysteresis analog trigger condition

Figure 4.5.5 shows the high-hysteresis analog trigger condition, the trigger signal is generated when the input analog signal level is greater than the High\_Threshold voltage, and the hysteresis duration is determined by the Low\_Threshold voltage.

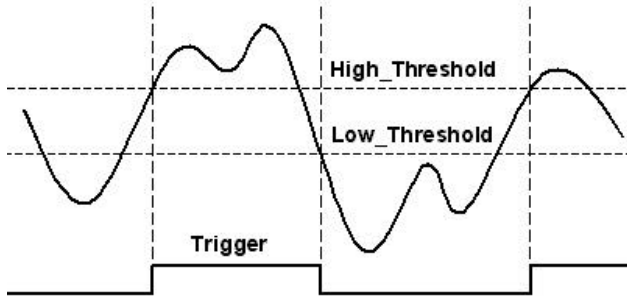


Figure 4.5.5 High-Hysteresis analog trigger condition

#### 4.5.2.5 Low-Hysteresis analog trigger condition

Figure 4.5.6 shows the low-hysteresis analog trigger condition, the trigger signal is generated when the input analog signal level is less than the Low\_Threshold voltage, and the hysteresis duration is determined by the High\_Threshold voltage.

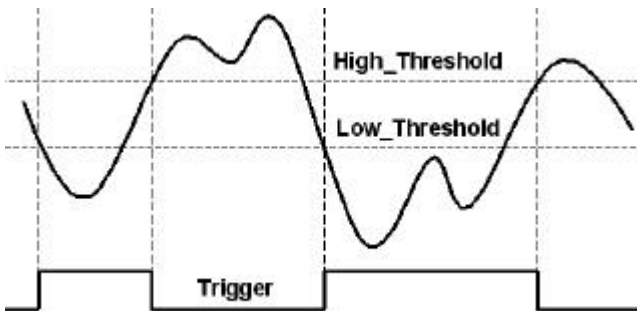


Figure 4.5.6 Low-Hysteresis analog trigger condition

### 4.5.3 External Digital Trigger

An external digital trigger occurs when a rising edge or a falling edge is detected on the digital signal connected to the EXTDRIG or the EXTWFTRG of the 68-pin connector for external digital trigger. The EXTDRIG is dedicated for the A/D process, and the EXTWFTRG.



**Figure 4.5.7 External digital trigger**

---

## 4.6 Timing Signals

In order to meet the requirements for user-specific timing and the requirements for synchronizing multiple boards, the DAQ-2000 series provides a flexible interface for connecting timing signals with external circuitry or other boards. The whole DAQ timing of the DAQ-2000 series is composed of a bunch of counters and trigger signals in the FPGA on board.

There are totally 7 timing signals related with the DAQ timing, including A/D, D/A process and GPTC usage. These 7 timing signals could be fed through the I/O connector or the SSI bus and take place of the internal timing signals. We implement a multiplexer in the FPGA to choose the desired timing signals individually as shown in the figure 4.6.1. Users could use the SSI (System Synchronization Interface) to achieve synchronization between multiple boards, and could use the AFI (Auxiliary Function Inputs) to allow external circuitry to control the timing of the DAQ-2000 series.

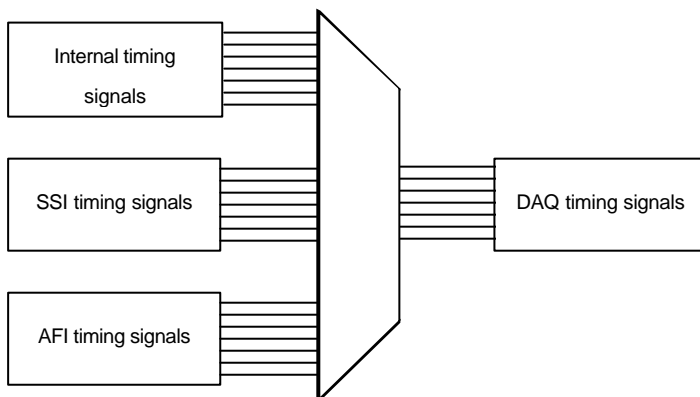


Figure 4.6.1 DAQ signals routing

### 4.6.1 Auxiliary Function Inputs

Users could use the AFI in applications that will use external circuitry to directly control the DAQ-2000 series boards. The AFI includes 2 categories of timing signals: one group is the dedicated input, and the other is a multi-function input. The EXTTIMEBASE, EXTDTTRIG and EXTWFTRG are belonged to the dedicated inputs, while the ADCONV, AD\_START, DAWR and DA\_START are belonged to the multi-function inputs. There are 2 multi-function inputs named AFI[1..0]. The AFI[0] could be used as ADCONV or AD\_START. and the AFI[1] could be used as DAWR, or DA\_START.

## 4.6.2 System Synchronization Interface

SSI (System Synchronization Interface) provides the DAQ timing synchronization of multiple boards. This interface bus use bi-directional I/O to provide a flexible connection between boards. You could choose for each of the 7 timing signals, which board to be the SSI master. The SSI master could drive the internal timing signals to the SSI slaves. With the SSI, users could achieve better synchronization between boards than using only the timing generated by individual boards.

Note that when power-up or reset, the DAQ timing signals are reset to use the internal generated timing signals.

# 5

## Calibration

This chapter introduces the calibration process to minimize AD measurement errors and DA output errors.

---

### 5.1 Loading Calibration Constants

The DAQ-2010 is factory calibrated before shipment by writing the associated calibration constants of TrimDACs to the onboard EEPROM. TrimDACs are devices which contain multiple DACs within a single package. TrimDACs do not have memory capability. That means the calibration constants do not retain their values after the system power is turned off. Loading calibration constants is the process to load the values of TrimDACs stored in the on-board EEPROM. ADLINK provides software to make it easy to read the calibration constants automatically when necessary.

There is a dedicated space for calibration constants in the EEPROM. In addition to the default bank of factory calibration constants, there are three extra user-modifiable banks. This means users can load the TrimDACs values either from the original factory calibration or from a calibration that is subsequently performed.

Because of the fact that the errors of the board measurement and the output will vary with time and temperature, it is recommended to re-calibrate when the board is installed in users environment. The auto-calibration function used to minimize the errors will be introduced in the next sub-section.

---

## 5.2 Auto-calibration

By the auto-calibration feature of the DAQ-2010, the calibration software can measure and correct for almost all the calibration errors without any external signal connections, reference voltage, or measurement devices.

The DAQ-2010 has an on-board calibration reference to ensure the accuracy of auto-calibration. The reference voltage is measured at the factory and adjusted through a digital potentiometer by using an ultra-precision calibrator. The resistance of the digital potentiometer is memorized after this adjustment. It is not recommended for users to adjust the on-board calibration reference except for an ultra-precision calibrator is available.

---

**Note:**

1. Before auto-calibration procedure starts, it is recommended to warm up the board for at least 15 minutes.
  2. Please remove the cable before an auto-calibration procedure is initiated because the DA outputs would be changed in the process of calibration.
- 

---

## 5.3 Saving Calibration Constants

After an auto-calibration is completed, users can save the new calibration constants into one of the three user-modifiable banks in the EEPROM. The date and the temperature when you run the auto-calibration function will be saved accompany with the calibration constants. That means user can stored three banks of calibrations constants according to three different environments. And users can load the calibration constants when the working environment is changed without re-calibrating the board.

## Product Warranty/Service

ADLINK warrants that equipment furnished will be free from defects in material and workmanship for a period of one year from the date of shipment. During the warranty period, we shall, at our option, either repair or replace any product that proves to be defective under normal operation.

This warranty shall not apply to equipment that has been previously repaired or altered outside our plant in any way as to, in the judgment of the manufacturer, affect its reliability. Nor will it apply if the equipment has been used in a manner exceeding its specifications or if the serial number has been removed.

ADLINK does not assume any liability for consequential damages as a result from our product uses, and in any event our liability shall not exceed the original selling price of the equipment. The remedies provided herein are the customer's sole and exclusive remedies. In no event shall ADLINK be liable for direct, indirect, special or consequential damages whether based on contract or any other legal theory.

The equipment must be returned postage-prepaid. Package it securely and insure it. You will be charged for parts and labor if the warranty period is expired or the product is proven to be misuse, abuse or unauthorized repair or modification.