DAQe-2016/DAQe-2010

PCI Express® 4-CH, Simultaneous,
High-Performance, Multi-Function
Data Acquisition Card

User's Manual

Manual Rev. 2.00

Revision Date: July 27, 2006

Part No: 50-11035-2000

Recycled Paper

Advance Technologies; Automate the World.

Trademarks

Getting Service from ADLINK

Customer Satisfaction is top priority for ADLINK Technology Inc. Please contact us should you require any service or assistance.

ADLINK TECHNOLOGY INC.

Web Site:	http://www.adlinktech.com
Sales & Service:	Service@adlinktech.com
TEL:	+886-2-82265877
FAX:	+886-2-82265717
Address:	9F, No. 166, Jian Yi Road, Chungho City, Taipei, 235 Taiwan

Please email or FAX this completed service form for prompt and satisfactory service.

Company Information	
Company/Organization	
Contact Person	
E-mail Address	
Address	
Country	
TEL	FAX:
Web Site	
Product Information	
Product Model	
Environment	OS: M/B: CPU: Chipset: Bios:

Please give a detailed description of the problem(s):

Table of Contents

List of Tables

List of Figures

1 Introduction

The ADLINK DAQe-2010 and DAQe-2016 are the first data acquisition cards based on the PCI Express® technology. Both cards offer four channels of simultaneous data sampling and multi-functionality using the dedicated PCI Express® x1 lane. PCI Express® features a point-to-point and high-speed serial I/O bus.

The DAQe-2010/2016 is capable of simultaneously sampling four analog input channels with differential input configuration to achieve maximum noise elimination. It also supports multiple card synchronization through SSI (system synchronization interface) bus for high-channel count applications.

Aside from flexible analog and digital trigger modes, the DAQe-2010/2016 provides multi-functionality including a 2-CH, 12-bit, 1 MS/s analog output; 2-CH, 16-bit timer/counters; and 24-CH TTL DIO. With software auto-calibration, the DAQe-2010/2016 provides a convenient way to maintain accuracy for long-range measurements.

The DAQe-2010/2016 software driver is backward-compatible with existing PCI-based applications, enabling DAQ-2010/2016 users to easily migrate to PCI Express®-based platform and enjoy the advantages of PCI Express® bus.

1.1 Features

The DAQe-2010/2016 data acquisition card provides the following advanced features:

- PCI Express® specification r1.0a-compliant
- 4-channel simultaneous differential analog inputs
- DAQe-2010: 14-bit Analog input resolution with sampling rate up to 2MS/s
- DAQe-2016: 16-bit Analog input resolution with sampling rate up to 800kS/s
- Programmable bipolar/unipolar analog input
- Programmable gain (x1, x2, x4, x8)
- DAQe-2010: Total 8K samples A/D FIFO
- DAQe-2016: Total 512 samples A/D FIFO
- Versatile trigger sources: software trigger, external digital trigger, analog trigger and trigger from System Synchronization Interface (SSI).
- A/D 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
- 2K samples output data FIFO for DA channels
- DA Data transfer: software update and bus-mastering DMA with Scatter/Gather functionality
- System Synchronization Interface (SSI)
- Full A/D and D/A auto-calibration
- Completely jumper-less and software-configurable

1.2 Applications

- Automotive testing
- Cable testing
- Transient signal measurement
- Automatic testing equipment
- Laboratory automation
- Biotech measurement

1.3 Specifications

Analog Input (AI)

- Number of channels: 4 differential
- A/D converter:
  - 2010: LTC1414 or equivalent
  - 2016: A/D7671 or equivalent
- Max sampling rate:
  - 2010: 2 MS/s
  - 2016: 800 kS/s
- Resolution:
  - 2010: 14 bits, no missing code
  - 2016:16 bits, no missing code
- FIFO buffer size:
  - 2010:8K samples
  - 2016: 512 samples
- Programmable input range:
  - Bipolar: ±10V, ±5V, ±2.5V, ±1.25V
  - Unipolar: 0~10V, 0~5V, 0~2.5V, 0~1.25V
- Operational common mode voltage range: ±11V
- Overvoltage protection:
  - Power on: continuous ±30V
  - Power off: continuous ±15V
- Input impedance: 1GΩ/100pF
- -3dB small signal bandwidth: (25°C typical)

Device	Input Range	Bandwidth (-3dB)	Input Range	Bandwidth (-3dB)
2010	±10V	1170 kHz	0 ~ 10V	1090 kHz
	±5V	1050 kHz	0 ~ 5V	1020 kHz
	±2.5V	800 kHz	0 ~ 2.5V	790 kHz
	±1.25V	530 kHz	0 ~ 1.25V	530 kHz
2016	±10V	840kHz	0 ~ 10V	900 kHz
	±5V	825kHz	0 ~ 5V	800 kHz
	±2.5V	710kHz	0 ~ 2.5V	690 kHz
	±1.25V	530kHz	0 ~ 1.25V	530 kHz

Table 1-1: -3dB small signal bandwidth

- Large signal bandwidth (1% THD): 300 kHz
- System Noise: (Typical)

Device	Input Range	System noise	Input Range	System noise
2010	±10V	0.6 LSBrms	0 ~ 10V	0.8 LSBrms
	±5V	0.6 LSBrms	0 ~ 5V	0.8 LSBrms
	±2.5V	0.6 LSBrms	0 ~ 2.5V	0.8 LSBrms
	±1.25V	0.7 LSBrms	0 ~ 1.25V	0.9 LSBrms
2016	±10V	1.9 LSBrms	0 ~ 10V	3.0 LSBrms
	±5V	1.9 LSBrms	0 ~ 5V	3.3 LSBrms
	±2.5V	2.0 LSBrms	0 ~ 2.5V	3.3 LSBrms
	±1.25V	2.0 LSBrms	0 ~ 1.25V	3.4 LSBrms

Table 1-2: System Noise

- CMRR: (DC to 60Hz, Typical)

Device	Input Range	CMRR	Input Range	CMRR
2010	±10V	89 dB	0~10V	86 dB
	±5V	90 dB	0~5V	89 dB
	±2.5V	94 dB	0~2.5V	92 dB
	±1.25V	96 dB	0~1.25V	95 dB

Table 1-3: CMRR: (DC to 60Hz)

Device	Input Range	CMRR	Input Range	CMRR
2016	±10V	82 dB	0~10V	82 dB
	±5V	88 dB	0~5V	86 dB
	±2.5V	94 dB	0~2.5V	92 dB
	±1.25V	99 dB	0~1.25V	96 dB

Table 1-3: CMRR: (DC to 60Hz)

- 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
- Input coupling: DC
- Offset error:
  - Before calibration: ±60mV max
  - After calibration: ±1mV max
- Gain error:
  - Before calibration: ±0.6% of output max
  - After calibration: ±0.1% of output max for DAQe-2010, ±0.03% of output max for DAQe-2016

Analog Output (AO)

- Number of channels: 2 channel voltage output
- 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 DA 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: ±10V or ±AOEXTREF
  - Unipolar: 0~10V or 0~AOEXTREF
- Settling time: 3µS to 0.5 LSB accuracy
- Slew rate: 20V/µS
- Output coupling: DC
- Protection: Short-circuit to ground
- Output impedance: 0.3Ω typical
- Output driving current: ±5mA max.
- Stability: Any passive load, up to 1500pF
- Power-on state: 0V steady-state
- Power-on glitch: ±1.5V/500uS
- Relative accuracy:
  - ±0.5 LSB typical, ±1 LSB max
- DNL:
  - ±0.5 LSB typical, ±1.2 LSB max
- Offset error:
  - Before calibration: ±80mV max
  - After calibration: ±1mV max
- Gain error:

- ▷ Before calibration: ±0.8% of output max
- ▷ After calibration: ±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: VIL=0.8V max; IIL=0.2mA max.
  - ▷ High: VIH=2.0V max; IIH=0.02mA max
- Output voltage:
  - ▷ Low: VOL=0.5V max; IOL=8mA max.
  - ▷ High: VOH=2.7V min; IOH=400µA
- Synchronous Digital Inputs (SDI, for DAQe-2010 only)
- Number of channels: 8 digital inputs sampled simultaneously with the analog signal input
- Compatibility: TTL/CMOS
- Input voltage:
  - ▷ Logic Low: VIL=0.8V max; IIL=0.2mA max.
  - ▷ Logic High: VIH=2.7V min; IIL=0.02mA 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: 10 MHz

Analog Trigger (A.Trig)

- Source:
  - All analog input channels; external analog trigger (EXTATRIG)
- Level: ±Full-scale, internal; ±10V external
- Resolution: 8 bits
- Slope: Positive or negative (software selectable)
- Hysteresis: Programmable
- Bandwidth: 400khz

External Analog Trigger Input (EXTATRIG)

- Input Impedance:
  - 40kΩ for DAQe-2010
  - 2kΩ for DAQe-2016
- Coupling: DC
- Protection: Continuous ±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: ±2ppm/°C
  - Long-term stability: 6ppm/1000Hr

Physical

- Dimensions:
  - 175mm by 107mm
- I/O connector: 68-pin female VHDCI type (e.g. AMP-787254-1)

Power Requirement (typical)

- DAQe-2010
  - +12VDC @ 0.448A
  - +3.3VDC @ 1.246A
- DAQe-2016
  - +12VDC @ 0.569A

Operating Environment

- Ambient temperature: 0 to 55°C
- Relative humidity: 10% to 90% non-condensing

Storage Environment

- Ambient temperature: -20 to 80°C
- Relative humidity: 5% to 95% non-condensing

Interface Connector: 68-pin AMP-787254-1 or equivalent

1.4 Software Support

The DAQe-2010/2016 supports the following operating systems:

- Windows® 2000/XP
- Linux: Fedora Core 3 and 4; Mandrakelinux 10.1; RedHat Enterprise Linux WS 3 and WS 4; SuSE LINUX Professional 9.3

NOTE: Visit http://www.adlinktech.com/TM/linux.daq.html for details on Linux support.

ADLINK provides versatile software drivers and packages for users' different approach to building up a system. ADLINK not only provides programming libraries such as DLL for most Windows based systems, but also provide drivers for other software packages such as LabVIEW® and MATLAB®. The DAQe-2010/2016 software drivers are backward-compatible with DAQ-2000 card series drivers enabling them to easily integrate to PCI Express®-based platform with the same set of drivers and applications.

All software options are included in the ADLINK All-in-One CD. Non-free software drivers are protected with licensing codes. Without the software code, you can install and run the demo version for two hours for trial/demonstration purposes. Please contact ADLINK dealers to purchase the formal license.

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® 2000/XP. DLL is binary compatible across Windows® 2000/XP. This means all applications developed with D2K-DASK are compatible across Windows® 2000/XP. 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\Software Package\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\Software Package\D2K-DASK-X.)

DAQ-LVIEW PnP: LabVIEW® Driver

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

(\\Manual\Software Package\DAQ-LVIEW PnP)

D2K-OCX: ActiveX Controls

We suggest customers who are familiar with ActiveX controls and VB/VC++ programming use D2K-OCX ActiveX control component libraries for developing applications. D2K-OCX is designed for Windows® 2000/XP. For more detailed information about D2K-OCX, refer to the user's guide in the CD.

(\\Manual\Software Package\D2K-OCX)

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

In addition, ADLINK supplies ActiveX control software DAQBench. DAQBench is a collection of ActiveX controls for measurement or automation applications. With DAQBench, you can easily develop custom user interfaces to display your data, analyze data you acquired or received from other sources, or integrate with popular applications or other data sources. For more detailed information about DAQBench, refer to the user's guide in the CD.

(\\Manual\Software Package\DAQBench Evaluation)

You can also get a free 4-hour evaluation version of DAQBench from the CD.

DAQBench is not free. Contact ADLINK or its local representative to purchase the software license.

2 Installation

This chapter describes how to install the DAQe-2010/2016. The contents of the package and unpacking information that you should be aware of are outlined first.

The DAQe-2010/2016 performs an automatic configuration of the IRQ, and port address. Users can use software utility, PCI_SCAN to read the system configuration.

2.1 Contents of Package

In addition to this User's Guide, the package should include the following items:

- DAQe-2010/2016 data acquisition card
- ADLINK All-in-One CD
- Software installation guide

If any of these items are 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 DAQe-2010/2016 card contains electrostatic sensitive components that can be easily be damaged by static electricity.

Therefore, 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 damages. Shipping and han-dling may cause damage to your module. Be sure there are no shipping and handling damages on the modules carton before continuing.

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 damages. 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.

You are now ready to install the DAQe-2010/2016.

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

2.3 DAQe-2010/2016 Layout

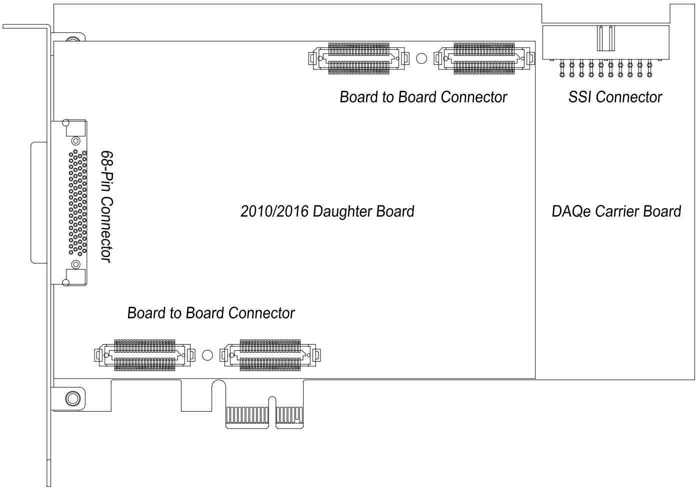

Figure 2-1: PCB Layout of the DAQe-2010/2016

3 Signal Connections

This chapter describes the connectors of the DAQe-2010/2016, and the signal connection between the DAQe-2010/2016 and external devices.

3.1 Connectors Pin Assignment

The DAQe-2010/2016 is equipped with one 68-pin VHDCI-type connector (AMP-787254-1). It is used for digital input/output, analog input / output, and timer/counter signals, etc. One 20-pin ribbon male connector is used for SSI (System Synchronous Interface). The pin assignments of the connectors are defined in Table 3-1 and Table 3-2.

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 / NC*	9	43	SDI3_0 / NC*
SDI2_1 / NC*	10	44	SDI2_0 / NC*
SDI1_1 / NC*	11	45	SDI1_0 / NC*
SDI0_1 / NC*	12	46	SDI0_0 / NC*
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

Table 3-1: 68-pin VHDCI-type pin assignment

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

Table 3-1: 68-pin VHDCI-type pin assignment

* SDI for DAQe-2010 only; NC for DAQe-2016

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 (2010) NC (2005/2006/2016)	DGND	Input	Synchronous digital inputs
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>

Table 3-2: 68-pin VHDCI-type Connector Legend

Pin #	Signal Name	Reference	Direction	Description
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 TIME-BASE
21,28,49, 50,54,62	DGND	--------	--------	Digital ground
22,56,23, 57,24,58, 25,59	PB<7,0>	DGND	PIO*	Programmable DIO pins of 8255 Port B
26,60,27, 61,29,63, 30,64	PC<7,0>	DGND	PIO*	Programmable DIO pins of 8255 Port C
31,65,32, 66,33,67, 34,68	PA<7,0>	DGND	PIO*	Programmable DIO pins of 8255 Port A
35~38	CH<0..3>-	--------	Input	Differential nega-tive 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 (2010) NC (2005/2006/2016)	DGND	Input	Synchronous dig-ital inputs
47	EXTWFTRIG	DGND	Input	External AO waveform
trigger				
48	EXTDTRIG	DGND	Input	External AI digital trigger

Table 3-2: 68-pin VHDCI-type Connector Legend

Pin #	Signal Name	Reference	Direction	Description
55	AFI0	DGND	Input	Auxiliary Function Input 0 (ADCONV, AD_START)
21	AFI1	DGND	Input	Auxiliary Function Input 1 (DAWR, DA_START)

Table 3-2: 68-pin VHDCI-type Connector Legend

*PIO means programmable I/O

SSI_TIMEBASE	1	2	DGND
SSI_ADCONV	3	4	DGND
SSI_DAWR	5	6	DGND
SSI_SCAN_START	7	8	DGND
RESERVED	9	10	DGND
SSI_AD_TRIG	11	12	DGND
SSI_DA_TRIG	13	14	DGND
RESERVED	15	16	DGND
RESERVED	17	18	DGND
RESERVED	19	20	DGND

Table 3-3: SSI connector (JP3) pin assignment for DAQe-2010/2016

Legend:

SSI timing signal	Functionality
SSI_TIMEBASE	SSI master: send the TIMEBASE out SSI slave: accept the SSI_TIMEBASE to replace the internal TIMEBASE signal.
SSI_ADCONV	SSI master: send the ADCONV out SSI slave: accept the SSI_ADCONV to replace the internal ADCONV signal.
SSI_SCAN_START	SSI master: send the SCAN_START out SSI slave: accept the SSI_SCAN_START to replace the internal SCAN_START signal.
SSI_AD_TRIG	SSI master: send the internal AD_TRIG out SSI slave: accept the SSI_AD_TRIG as the digital trigger signal.
SSI_DAWR	SSI master: send the DAWR out. SSI slave: accept the SSI_DAWR to replace the internal DAWR signal.
SSI_DA_TRIG	SSI master: send the DA_TRIG out. SSI slave: accept the SSI_DA_TRIG as the digital trigger signal.

Table 3-4: Legend of SSI connector

3.2 Analog Input Signal Connection

The DAQe-2010/2016 provides 4 differential analog input channels. The analog signal can be converted to digital values by the A/D converter. To avoid ground loops and get more accurate measurements from the A/D conversion, it is quite important to understand the signal source type and how to connect the analog input signals.

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 DAQe-2010/2016, assuming that the computer is plugged into the same power system. Non- isolated outputs of instruments and devices that plug into the buildings power system are ground-referenced signal sources.

Floating Signal Sources

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

Single-Ended Measurements

For single-ended connection, the analog input signal is 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 in-stead of floating. Refer to Figure 3-1.

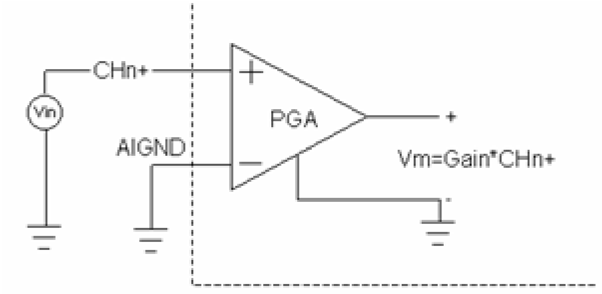

Figure 3-1: Single-Ended connections

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 minimal wire connections are necessary.

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 3-2 shows the connection of ground-referenced signal sources under the differential input mode.

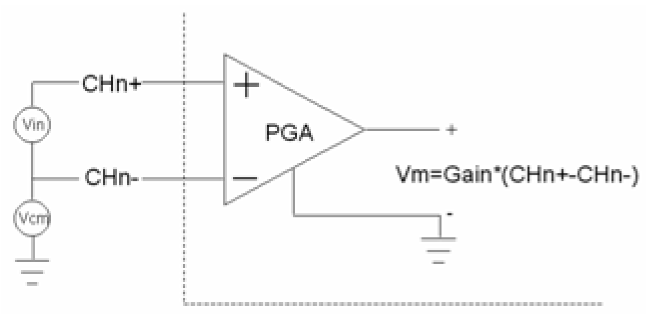

Figure 3-2: Ground-referenced source and differential input

Differential Connection for Floating Signal Sources

Figure 3-3 shows how to connect a floating signal source to DAQe-2010/2016 in differential input mode. For floating signal sources, you need to add a resistor at each channel 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 100 ohms, you can simply connect the negative side of the signal to AGND as well as the negative input of the Instru-mentation Amplifier, without any resistors at all. In differential input mode, less noise couples into the signal connections than in single-ended mode.

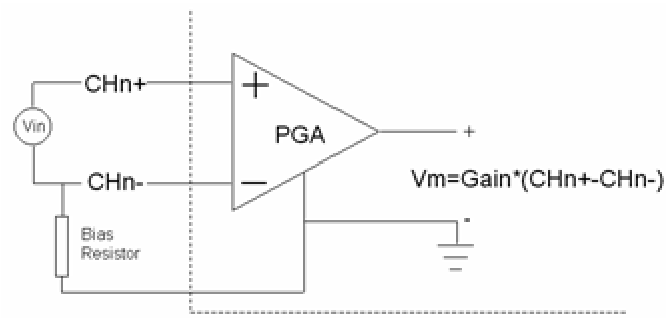

Figure 3-3: Floating source and differential input

4 Operation Theory

The operation theory of the functions on the DAQe-2010/2016 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 theory can help you understand how to configure and program the DAQe-2010/2016.

4.1 A/D Conversion

When using an A/D converter, users should first know about the properties of the signal to be measured. Users can decide which channel to use and where to connect the signals to the card. Refer to Section 3.2 for signal connections. In addition, users should define and control the A/D signal configurations, including channels, gains, and A/D signal types.

There are 2 ways to initiate A/D conversion, either by Software Polling or Programmable Scan Acquisition; these are described below.

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 once 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 the PC's memory for further processing.

DAQe-2010 AI Data Format

Synchronous Digital Inputs (for DAQe-2010 only)

When each A/D conversion is completed, the 14-bits converted digital data accompanied with 2 bits of SDI<1..0>_X per channel from J5 will be latched into the 16-bit register and data FIFO, as shown in Figure 4-1 and Figure 4-2 on page 24. Therefore, users can simultaneously sample one analog signal with four digital signals. The data format of every acquired 16-bit data is as follows:

```
D13, D12, D11 ....... D1, D0, b1, b0
Where
```

```
D13, D12, D11 ....... D1, D0: 2’s complement A/D
     14-bit data
b1, b0: Synchronous Digital Inputs SDI<1..0>
```

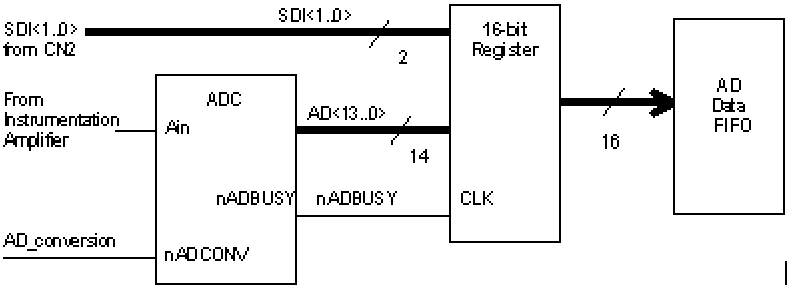

Figure 4-1: Synchronous Digital Inputs Block Diagram

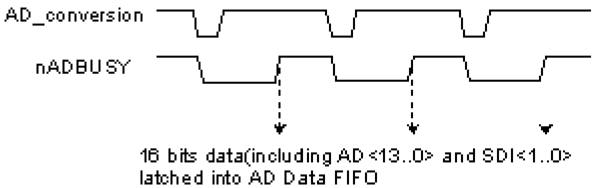

Figure 4-2: Synchronous Digital Inputs timing

Note: Since the analog signal is sampled when an A/D conversion starts (falling edge of A/D_conversion signal), while SDI<1..0> are sampled right after an A/D conversion completes (rising edge of nADBUSY signal). Precisely SDI<1..0> are sampled within 220 to 400ns lag to the analog signal, due to the variation of the conver-sion time of the A/D converters.

Table 4-1 and Table 4-2 illustrate the ideal transfer characteristics of various input ranges of DAQe-2010/2016. The converted digital codes for DAQe-2010 are 14-bit and 2’s complement, and here we present the codes as hexa-decimal numbers. Note that the last 2 bits of the transferred data, which are the synchronous digital input (SDI), should be ignored when retrieving the analog data.

Description	Bipolar Analog Input Range				Digital code
Full-scale Range	±10V	±5V	±2.5V	±1.25V	
Least significant bit	1.22mV	0.61mV	0.305mV	0.153mV	
FSR-1LSB	9.9988V	4.9994V	2.4997V	1.2499V	1FFF
Midscale +1LSB	1.22mV	0.61mV	0.305mV	0.153mV	0001
Midscale	0V	0V	0V	0V	0000
Midscale –1LSB	-1.22mV	-0.61mV	-0.305mV	-0.153mV	3FFF
-FSR	-10V	-5V	-2.5V	-1.25V	2000

Table 4-1: Bipolar analog input range and the output digital code on DAQe-2010 (Note that the last 2 digital codes are SDI<1..0>)

Description	Unipolar Analog Input Range				Digital code
Full-scale Range	0V to 10V	0 to +5V	0 to +2.5V	0 to +1.25V	
Least significant bit	0.61mV	0.305mV	0.153mV	76.3uV	
FSR-1LSB	9.9994V	4.9997V	2.9999V	1.2499V	1FFF
Midscale +1LSB	5.00061V	2.50031V	1.25015V	625.08mV	0001
Midscale	5V	2.5V	1.25V	625mV	0000
Midscale –1LSB	4.99939V	2.49970V	1.24985V	624.92mV	3FFF
-FSR	0V	0V	0V	0V	2000

Table 4-2: Unipolar analog input range and the output digital code on DAQe-2010 (Note that the last 2 digital codes are SDI<1..0>)

DAQe-2016 AI Data Format

The data format of the acquired 16-bit A/D data is Binary coding. Table 4-3 and Table 4-4 illustrate the valid input ranges and the ideal transfer characteristics. The converted digital codes for DAQe-2016 are 16-bit and direct binary, and here we present the codes as hexadecimal numbers.

Description	Bipolar Analog Input Range				Digital code
Full-scale Range	±10V	±5V	±2.5V	±1.25V	
Least significant bit	305.2uV	152.6uV	76.3uV	38.15uV	
FSR-1LSB	9.999695V	4.999847V	2.499924V	1.249962V	FFFF
Midscale +1LSB	305.2uV	152.6uV	76.3uV	38.15uV	8001
Midscale	0V	0V	0V	0V	8000
Midscale -1LSB	-305.2uV	-152.6uV	-76.3uV	-38.15uV	7FFF
-FSR	-10V	-5V	-2.5V	-1.25V	0000

Table 4-3: Bipolar analog input range and the output digital code on DAQe-2016

Description	Unipolar Analog Input Range				Digital code
Full-scale Range	0V to 10V	0 to +5V	0 to +2.5V	0 to +1.25V	
Least signifi-cant bit	152.6uV	76.3uV	38.15uV	19.07uV	
FSR-1LSB	9.999847V	4.999924V	2.499962V	1.249981V	FFFF
Midscale +1LSB	5.000153V	2.500076V	1.250038V	0.625019V	8001
Midscale	5V	2.5V	1.25V	0.625V	8000
Midscale -1LSB	4.999847V	2.499924V	1.249962V	0.624981V	7FFF
-FSR	0V	0V	0V	0V	0000

Table 4-4: Unipolar analog input range and the output digital code on DAQe-2016

Software conversion with polling data transfer acquisition mode (Software Polling)

This is the easiest way to acquire a single A/D data. The A/D converter starts one conversion whenever the dedicated software command is executed. Then the software would poll the conversion status and read the A/D data back when it is available.

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

Specifying Channel, Gain, and Polarity

In both the Software Polling and programmable scan acquisition mode, the channel, gain, and polarity for each channel can be specified and selected. With this configuration, signal sources must be connected to the right connector as the specified settings.

When the specified channels have been sampled from the first to the last data, the settings applied to each channel would be the same until next change.

Example:

Typically you can set the input configuration for different channels:

```
Ch1 with unipolar ±10V
Ch2 with bipolar ±2.5V
Ch3 with no signal input (disabled)
Ch4 with bipolar ±1.25V
```

Programmable scan acquisition mode

Scan Timing and Procedure

It's recommended that this mode be used if your applications need a fixed and precise A/D sampling rate. You can accurately program the period between conversions of individual channels. There are at least 2 counters, which need to be specified:

```
SI_counter (24 bit):Specify the Scan Interval =
     SI_counter / TIMEBASE
PSC_counter (24 bit):Specify Post Scan Counts,
     i.e. the total sample count after a trigger
     event,
```

The acquisition timing and the meanings of the 2 counters are illustrated in Figure 4-3. The SCAN_START signal is derived from the SI_counter, which will lead to the A/D conversion signal generation. Note that the DAQe-2010/2016 card is a simultaneous sampling A/D card, so the “scan interval” equals to the “sampling interval”.

Example: (Post-trigger acquisition)

```
Set
     SI_counter = 160
     PSC_counter = 30
     TIMEBASE = Internal clock source
Then
Scan Interval  = 160/40M s = 4 us
Total acquisition time = 30 X 4 us = 120 us
```

TIMEBASE clock source

In scan acquisition mode, all the A/D conversions start on the output of counters, which use TIMEBASE as the clock source. By software you can specify the TIMEBASE to be either an internal clock source (on-board 40MHz clock) or an external clock input (EXTTIMEBASE) on J5 connector (68-pin VHDCI). The external TIMEBASE is useful when you want to ac-quire data at rates not available with the internal A/D sample clock. The external clock source should generate TTL-compatible continuous clocks; with a maximum frequency of 40MHz while the minimum should be 1 MHz. Refer to section **4.6 User-controllable Timing Signals** for information of user-controllable timing signals.

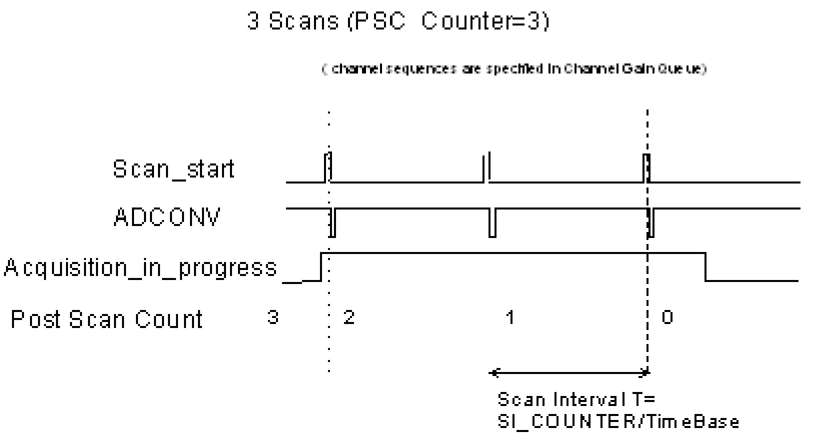

Figure 4-3: Scan Timing

There are 4 trigger modes to start the scan acquisition. Refer to section **4.5 Trigger Sources** for more details. The data transfer mode is discussed below.

Note:

1. The maximum A/D sampling rate is 2 MHz for DAQe-2010 and 800 kHz for DAQe-2016. Therefore, the minimum setting of SI_counter is 20 for DAQe-2010 and 50 for DAQe-2016 while using the internal TIMEBASE.
2. The SI_counter is a 24-bit counter. Therefore, the maximum scan interval while using an internal TIMEBASE = 224/40M s = 0.419s.

Trigger Modes

DAQe-2010/2016 provides 4 trigger sources (internal software trigger, external analog trigger, external digital trigger or SSI trigger signals). You must select one of them as the source of the trigger event. A trigger event occurs when the specified condition is detected on the selected trigger source (For example, a rising edge on the external digital trigger input). Refer to section **4.6 User-controllable Timing Signals** for more information about SSI signals.

There are 4 trigger modes (pre-trigger, post-trigger, middle-trigger, and delay-trigger) working with the 4 trigger sources to initiate different scan data acquisition timing when a trigger event occurs. They are described as follows. For information on trigger sources, refer to **4.5 Trigger Sources**.

Pre-Trigger Acquisition

Use pre-trigger acquisition in applications where you want to collect data before a trigger event. The A/D starts to sample when you execute the specified function calls to begin the pre-trigger operation, and it stops when the trigger event occurs. Users must program the value M in M_counter (16 bits) to specify the amount of the stored scans before the trigger event. If an external trigger occurs, the program only stores the last M scans of data converted before the trigger event, as illustrated in Figure 4-6, where M_counter = M =3, PSC_counter = 0. The post scan count is 0 because there is no sampling after the trigger event in pre-trigger acquisition. The total stored amount of data = Number of enabled channels * M_counter.

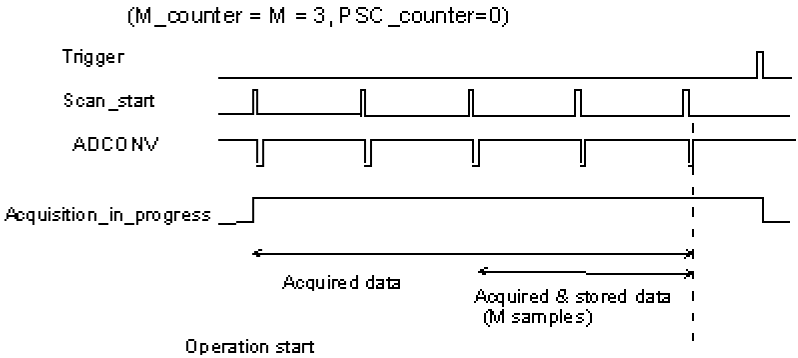

Figure 4-4: Pre-trigger (trigger occurs after at least M scans acquired)

Note that If the trigger event occurs when a conversion is in progress, the data acquisition won't stop until this conversion is completed, and the stored M scans of data include the last scan, as illustrated in Figure 4-5, where M_counter = M =3, PSC_counter = 0.

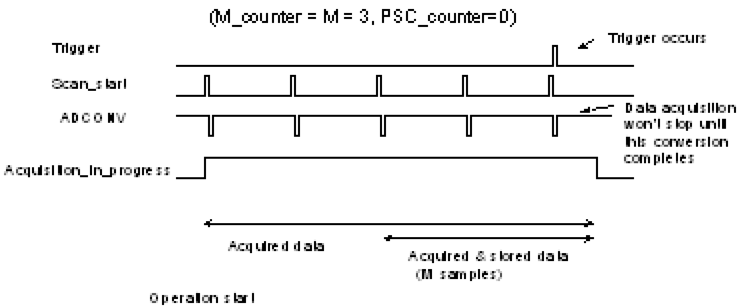

Figure 4-5: Pre-trigger scan acquisition (trigger occurs when a conversion is in progress)

When the trigger signal occurs before the first M scans of data are con-verted, the amount of stored data could be fewer than the originally specified amount M_counter, as illustrated in Figure 4-6. This situation can be avoided by setting M_enable. If M_enable is set to 1, the trigger signal will be ignored until the first M scans of data are converted, and it assures the user M scans of data under pre-trigger mode, as illustrated in Figure 4-7. However, if M_enable is set to 0, the trigger signal will be accepted any time. Note that the total amount of stored data is always equal to the number in the M_counter because the data acquisition do not stop until a scan is completed.

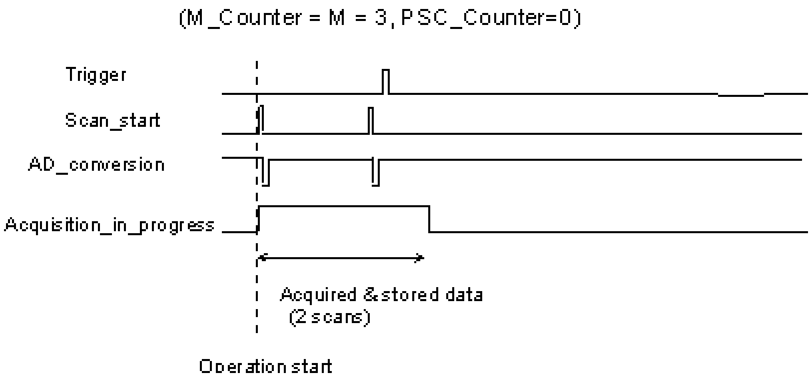

Figure 4-6: Pre-trigger with M_enable = 0 (Trigger occurs before M scans)

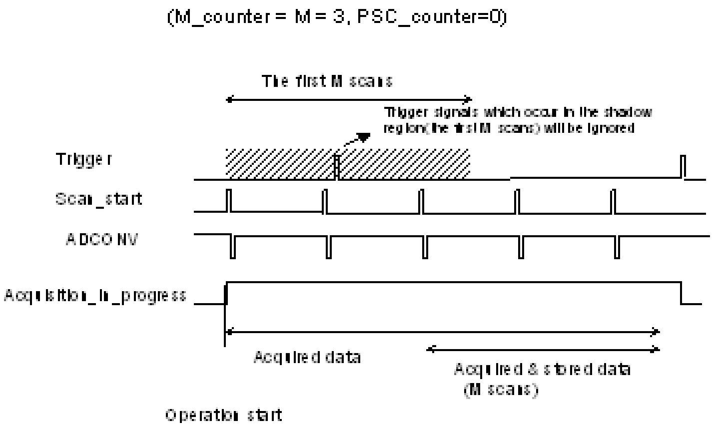

Figure 4-7: Pre-trigger with M_enable = 1

Note: The PSC_counter is set to 0 in pre-trigger acquisition mode.

Middle-Trigger Acquisition

Use middle-trigger acquisition in applications where you want to collect data before and after a trigger event. The number of scans (M) stored before the trigger is specified in M_counter, while the number of scans (N) after the trigger is specified in PSC_counter.

Like pre-trigger mode, the number of stored data could be less than the specified amount of data (M+N), if an external trigger occurs before M scans of data are converted. The M_enable bit in middle-trigger mode takes the same effect as in pre-trigger mode. If M_enable is set to 1, the trigger signal will be ignored until the first M scans of data are converted, and it assures the user with (M+N) scans of data under middle-trigger mode. However, if M_enable is set to 0, the trigger signal will be accepted at any time. Figure 4-8 shows the acquisition timing with M_enable=1.

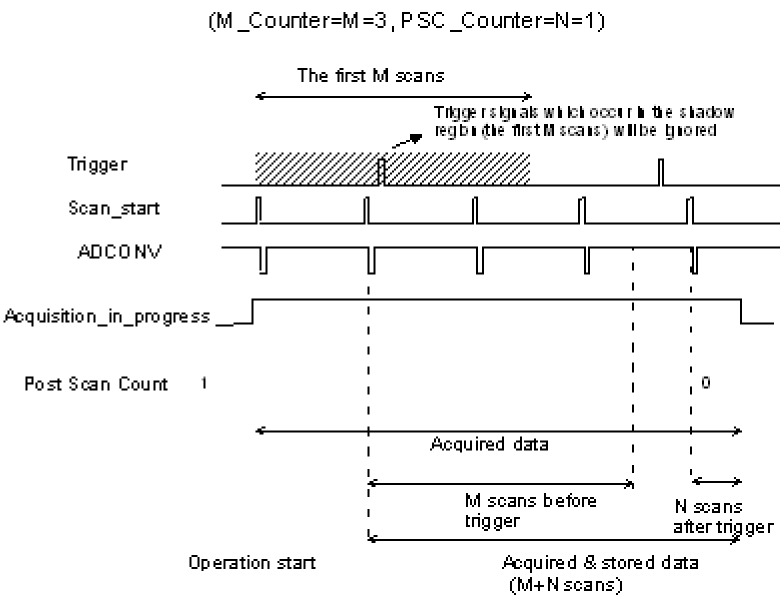

Figure 4-8: Middle trigger with M_enable = 1

If the trigger event occurs when a scan is in progress, the stored N scans of data would include this scan, as illustrated in Figure 4-9.

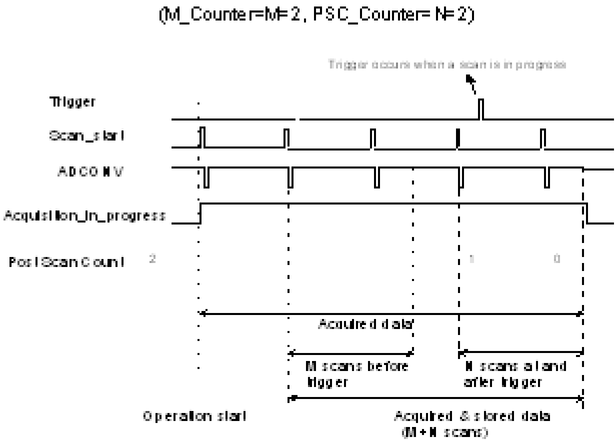

Figure 4-9: Middle trigger (trigger occurs when a scan is in progress)

Note:M_counter defined in Middle-Trigger is different from that of Pre-Trigger. In Middle-trigger, M_Counter ends counting before the trigger event while in Pre-Trigger, M_Counter ends counting right at or before trigger event. Refer to Figure 4-6 and Figure 4-9.

Post-Trigger Acquisition

Use post-trigger acquisition in applications where you want to collect data after a trigger event. The number of scans after the trigger is specified in PSC_counter, as illustrated in Figure 4-10. The total acquired data length = number of enable-channel * PSC_counter.

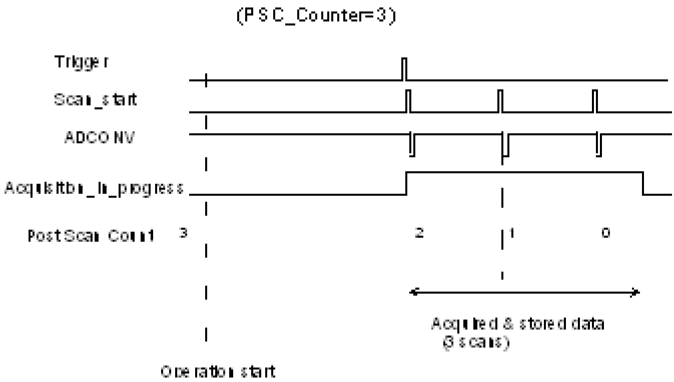

Figure 4-10: Post trigger

Delay Trigger Acquisition

Use delay trigger acquisition in applications where you want to delay the data collection after the occurrence of a specified trigger event. The delay time is controlled by the value, which is pre-loaded in the Delay_counter (16-bit). The counter counts down on the rising edge of the Delay_counter clock source after the trigger condition is met. The clock source can be software programmed either by the TIMEBASE clock (40 MHz) or A/D sampling clock (TIMEBASE / SI_counter). When the count reaches 0, the counter stops and the card starts to acquire data. The total acquired data length = number of enable-channel * PSC_counter.

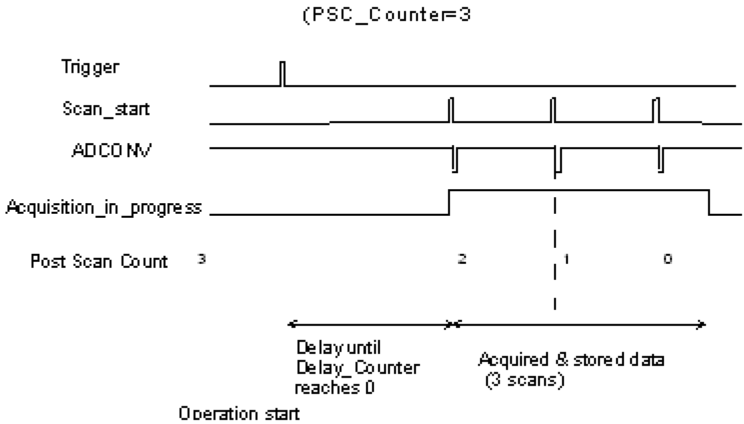

Figure 4-11: Delay trigger

Note: When the Delay_counter clock source is set to TIMEBASE, the maximum delay time = 216/40M s = 1.638ms, and when the source is set to A/D sampling clock, the maximum delay time can be as higher as (216 * SI_counter / 40M).

Post-Trigger or Delay-trigger Acquisition with re-trigger

Use post-trigger or delay-trigger acquisition with re-trigger function in ap-plications where you want to collect data after several trigger events. The number of scans after each trigger is specified in PSC_counter, and users could program Retrig_no to specify the re-trigger numbers. Figure 4-12 illustrates an example. In this example, 2 scans of data is acquired after the first trigger signal, then the card waits for the re-trigger signal (re-trigger signals which occur before the first 2 scans is completed will be ignored). When the re-trigger signal occurs, 2 more scan is performed. The process repeats until specified amount of re-trigger signals are detected. The total acquired data length = number of enable-channel * PSC_counter * Retrig_no.

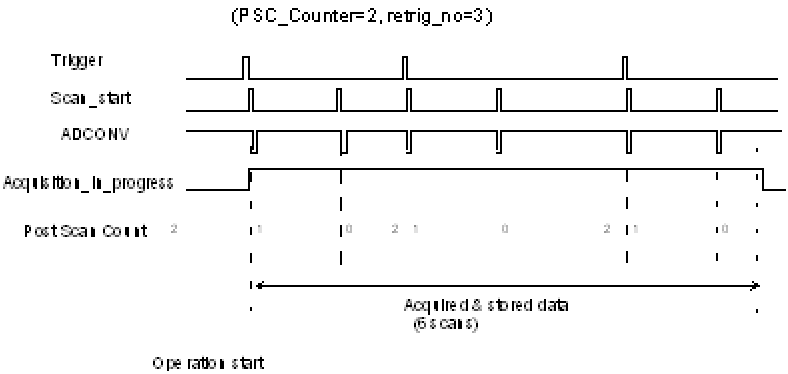

Figure 4-12: Post trigger with re-trigger

Bus-mastering DMA Data Transfer

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 the PCIe bridge, controls the PCIe when it becomes the master of the bus. Bus mastering reduces the size of the onboard 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 PCIe bus. Once the analog input operation starts, control returns to your program. The hardware temporarily stores the acquired data in the on-board 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 the bus-mastering DMA.

The DMA transfer mode is very complex to program. We recommend using a high-level program library to configure this card. If users would like to know more about programs/software's that can handle the DMA bus master data transfer, refer to http://www.plxtech.com for additional information on PCIe bridge.

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

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

In a multi-user or multi-tasking OS, like Windows, Linux, etc., it is difficult to allocate a large continuous memory block to do the DMA transfer. Therefore, the PCIe bridge 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 very large amounts of data without being limited by the fragment of small size memory. Users can configure the linked list for the input DMA channel or the output DMA channel. Figure 4-13 shows a linked list that is constructed by three DMA descriptors. Each descriptor contains a PCIe 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 descriptors altogether by their application programs. DAQe-2010/2016 software driver provides simple settings of the scatter/gather function, and some sample programs are also provided within the ADLINK All-in-One CD.

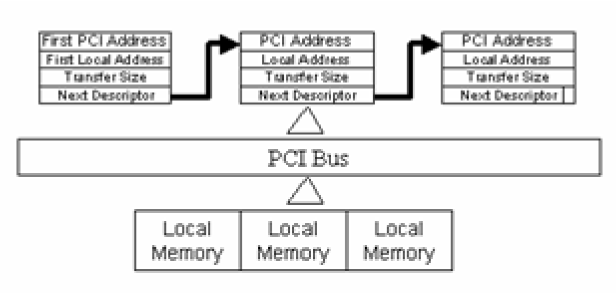

Figure 4-13: Scatter/gather DMA for data transfer

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

4.2 D/A Conversion

There are 2 channels of 12-bit D/A output available in the DAQe-2010/2016. When using D/A converters, users should assign and control the D/A converter reference sources for the 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 inputs are available in the DAQe-2010/2016. The 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 with the reference to generate the analog output. While using internal 10V reference, the full range would be –10V ~ +9.9951V in the bipolar output mode, and 0V ~ 9.9976V in the unipolar output mode. While using an external reference, users can reach different output ranges by connecting different references. For example, if connecting a DC –5V with the external reference, then the users can get a full range from –4.9976V to +5V in the bipolar output with inverting char-acteristics due to the negative reference voltage. Users could also have an amplitude modulated

(AM) output by feeding a sinusoidal signal into the reference input. The range of the external reference should be within ±10V. Table 4-5 and Table 4-6 illustrates the relationship between digital code and output voltages.

Digital Code	Analog Output
111111111111	Vref * (2047/2048)
100000000001	Vref * (1/2048)
100000000000	0V
011111111111	-Vref * (1/2048)
000000000000	-Vref

Table 4-5: Bipolar output code table
(Vref=10V if internal reference is selected)

Digital Code	Analog Output
111111111111	Vref * (4095/4096)
100000000000	Vref * (2048/4096)
000000000001	Vref * (1/4096)
000000000000	0V

Table 4-6: Unipolar output code table
(Vref=10V if internal reference is selected)

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 to a buffering Data FIFO.

There are two D/A conversion modes: **Software Update** and **Timed Waveform Generation**. These modes are described in the following sections including timing, trigger source control, trigger modes and data transfer methods. Either mode may be applied independently on each D/A channels. You can update the DA CH0 software while generating timed waveforms on CH1.

Software Update

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

Timed Waveform Generation

This mode can provide your applications with a precise D/A output with a fixed update rate. It can 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 a total of 5 counters to be specified. These counters include:

```
UI_counter (24 bits): specify the DA Update
     Interval = CHUI_counter/TIMEBASE.
UC_counter (24 bits): specify the total Update
     Counts in a single waveform
IC_counter (24 bits): specify the Iteration
     Counts of waveform.
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 4-14 shows a typical D/A timing diagram. D/A updates its output on each rising edge of DAWR. The meaning of the counters above is discussed more in the following sections.

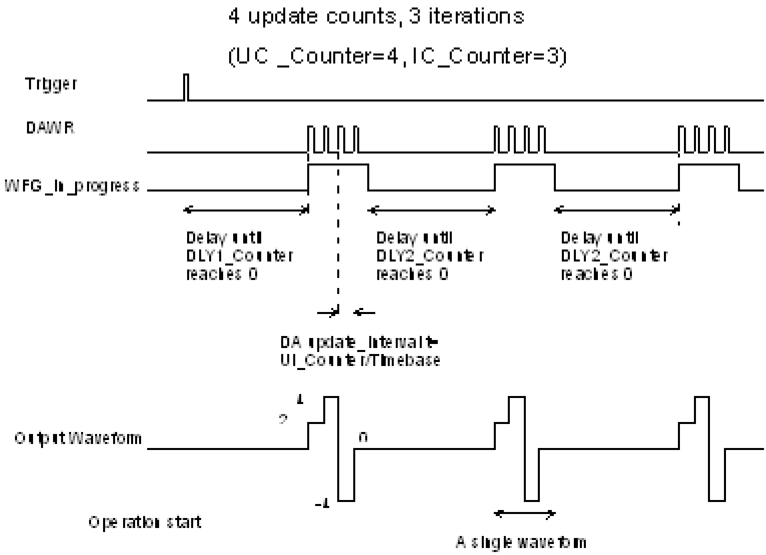

Figure 4-14: Typical D/A timing of waveform generation
(Assuming the data in the data buffer are 2V, 4V, -4V, 0V)

Note: The maximum D/A update rate is 1MHz. Therefore, the minimum setting of the UI_counter is 40 while using an internal TIMEBASE(40MHz).

Trigger Modes

Post-Trigger Generation

Use post trigger when you want to perform DA waveform right after a trigger event occurs. In this trigger mode DLY1_Counter is not used and you don't need to specify it. Figure 4-15 shows a single waveform generated right after a trigger signal is detected. The trigger signal could come from a software command, an analog trigger or a digital trigger. Please refer to section 4.5 for detailed information.

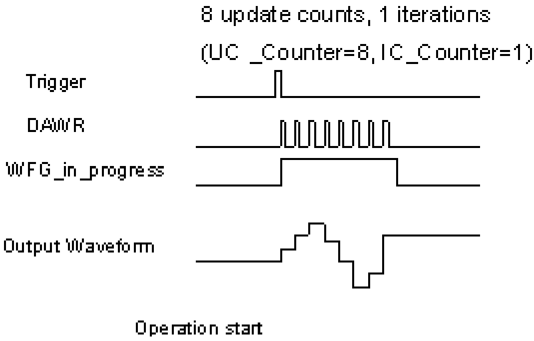

Figure 4-15: Post trigger waveform generation
(Assuming the data in the data buffer are 2V, 4V, 6V, 3V, 0V, -4V, -2V, 4V)

Delay-Trigger Generation

Use delay trigger when you want to delay the waveform generation after a trigger event. In Figure 4-16, DA_DLY1_counter determines the delay time from the trigger signal to the start of the waveform generation. DLY1_counter counts down on the rising edge of its clock source after the trigger condition is met. When the count reaches 0, the counter stops and the DAQe-2010/2016 starts the waveform generation. This DLY1_Counter is 16 bits wide and users can set the delay time in units of TIMEBASE (delay time = DLY1_Counter/TIMEBASE) or in units of update period (delay time = DLY1_Counter * UI_counter/TIMEBASE), such that the delay time can reach a wider range.

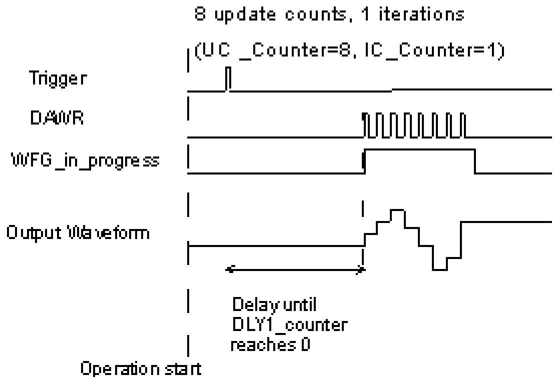

Figure 4-16: Delay trigger waveform generation
(Assuming the data in the data buffer are 2V, 4V, 6V, 3V, 0V, -4V, -2V, 4V)

Post-Trigger or Delay-Trigger with Re-trigger

Use post-trigger or delay-trigger with re-trigger function when you want to generate waveform after more than one trigger events. The re-trigger function can be enabled or disabled by software setting. In Figure 4-17, each trigger signal will initiate a waveform generation. However, the trigger event would be ignored while the waveform generation is ongoing.

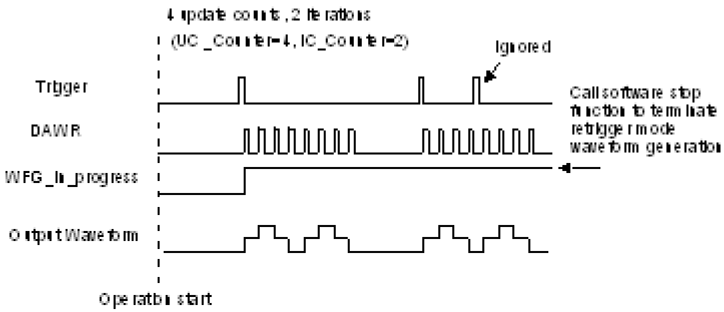

Figure 4-17: Re-triggered waveform generation
(Assuming the data in the data buffer are 2V, 4V, 2V, 0V)

Iterative Waveform Generation

Set IC_Counter in order to generate iterative waveforms from the data of a single waveform. The counter stores the iteration number, and the iterations can be finite as shown in Figure 4-18 or infinite as shown in Figure 4-19.

A data FIFO on board is used to buffer the digital data for DA output. If the data size of a single waveform you specified (That is, Update Counts in UC_counter) is less than the FIFO size, after initially transferring the data from the host PC memory to the FIFO on board, the data in the FIFO will be automatically re-transmitted whenever a single waveform is completed. Therefore, it won't occupy the PCI bandwidth when repetitive waveforms are performed. However, if the size of a single waveform were larger than that of the FIFO, it needs to be intermittently loaded from the host PC's memory via DMA, when a repetitive waveforms is performed thus PCI bandwidth would be occupied.

The data FIFO size on DAQe-2010 is 2k samples while the DAQe-2016 FIFO size is 512 samples.

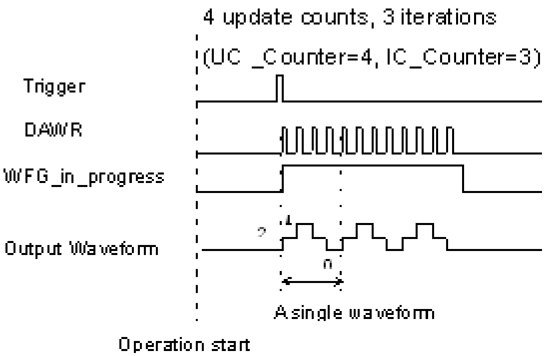

Figure 4-18: Finite iterative waveform generation with Post-trigger and DLY2_Counter = 0

(Assuming the data in the data buffer are 2V, 4V, 2V, 0V)

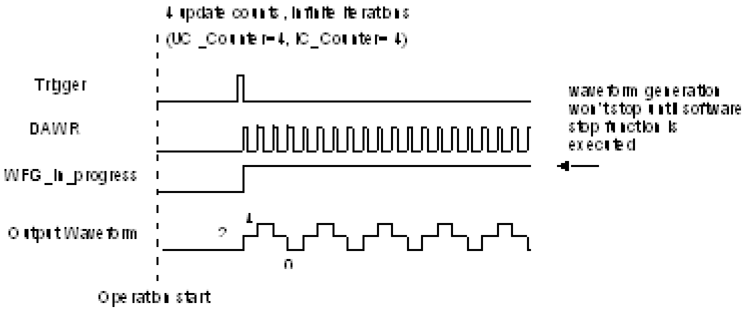

Figure 4-19: Infinite iterative waveform generation with Post-trigger and DLY2_Counter = 0

(Assuming the data in the data buffer are 2V, 4V, 2V, 0V)

Note:

1. When running infinite iterative waveform generation, setting IC_Counter is ineffective to the waveform generation. It only makes a difference when setting stop mode III, refer to Figure 4-22: Stop mode III.
2. Refer to the software manual to set finite and infinite iterative waveform generation.

Delay2 in Repetitive Waveform Generation

To diversify the D/A waveform generation, we add a DLY2 Counter to separate 2 consecutive waveforms in repetitive waveform generation. The time between two waveforms is set by the value of DLY2 Counter. The Delay2 counter starts to count down after a waveform generation finishes, and the next waveform generation starts right after it counts down to zero, just as shown in Figure 4-20. This DLY2_Counter is 16-bits wide and users can set the delay time in units of TIMEBASE (delay time = DLY2_Counter/TIMEBASE) or in units of update period (delay time = DLY2_Counter * UI_counter/TIMEBASE), such that the delay time can reach a wider range

Stop Modes of Scan Update

You can call software stop function to stop waveform generation when it is still in progress. Three stop modes are provided for timed waveform generation, which means when it is to stop the waveform generation. You can apply these 3 modes to stop waveform generation no matter infinite or finite waveform generation mode is selected.

Figure 4-20 illustrates an example for stop mode I, in this mode the waveform stops immediately when software command is asserted.

In stop mode II, after a software stop command is given, the waveform generation won't stop until a complete single waveform is finished. Take Figure 4-21 as an example. Since UC_counter is set to 4, the total DA update counts (that is, number of pulses of DAWR signal) must be a multiple of 4.(update counts = 20 in this example)

In stop mode III, after a software stop command is given, the waveform generation won't stop until the performed number of waveforms is a mul-tiple of IC_Counter. Take Figure 4-22 as an example, since IC_Counter is set to 3, the total generated waveforms must be a multiple of 3(waveforms = 6 in this example), and the total DA update counts must be a multiple of 12(UC_counter * IC_Counter). You can compare these three figures to see their differences.

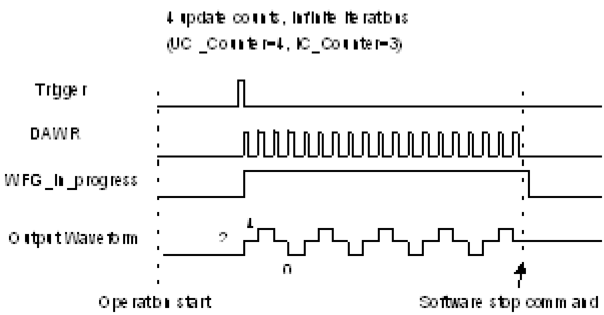

Figure 4-20: Stop mode I
(Assuming the data in the data buffer are 2V, 4V, 2V, 0V)

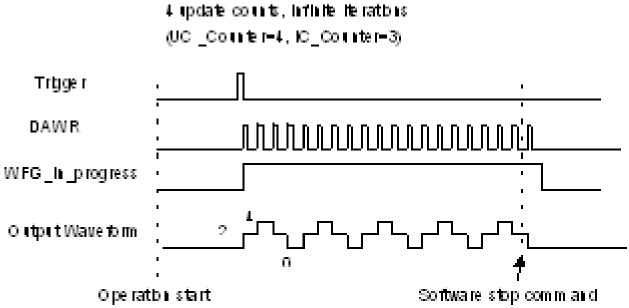

Figure 4-21: Stop mode II

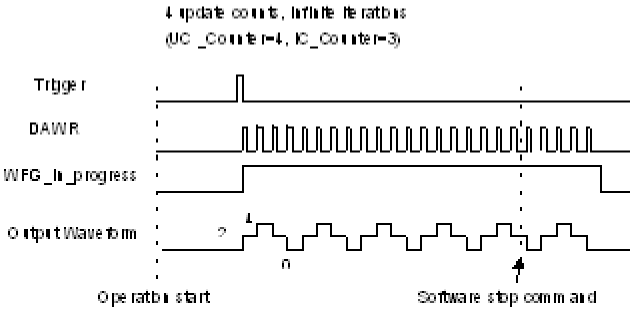

Figure 4-22: Stop mode III

4.3 Digital I/O

The DAQe-2010/2016 contains 24-lines of general-purpose digital I/O (GPIO), which is provided through a 82C55A chip.

The 24-line GPIO are separated into three ports: Port A, Port B and Port C. High nibble (bit[7…4]), and low nibble (bit[3…0]) of each port can be indi-vidually programmed to be either inputs or outputs. Upon system startup or reset, all the GPIO pins are reset to high impedance inputs.

DAQe-2010 also provides 2 digital inputs per channel (SDI from J5), which are sampled simultaneously with an analog signal input and is stored with the 14-bit AD data. Refer to section **4.1 A/D Conversion** for the more details.

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 10 MHz)
- 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

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 resister about 10K ohms. Then GPTC_OUT goes low after the DAQe-2010/2016 is initialized.

All the polarities of input/output signals can be programmed by software. In this chapter, for easy explanation, all GPTC_CLK, GPTC_GATE, and GPTC_OUT are assumed to be active high or rising-edge triggered in the figures.

General Purpose Timer/Counter modes

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

Mode 1: 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 4-23 illustrates the operation with initial count = 5, count-down mode.

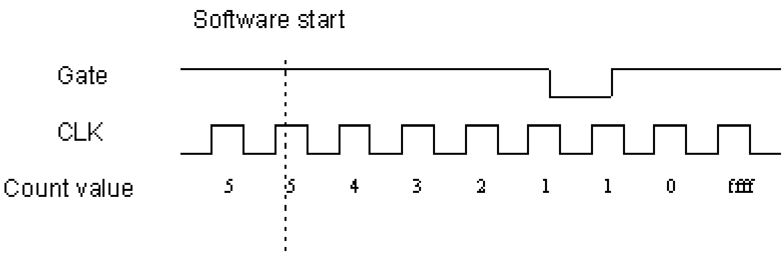

Figure 4-23: Mode 1 Operation

Mode 2: 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 com-pletion of the period interval on GPTC_GATE, GPTC_OUT outputs high and then current count value can be read-back by software.

Figure 4-24 illustrates the operation where initial count = 0, count-up mode.

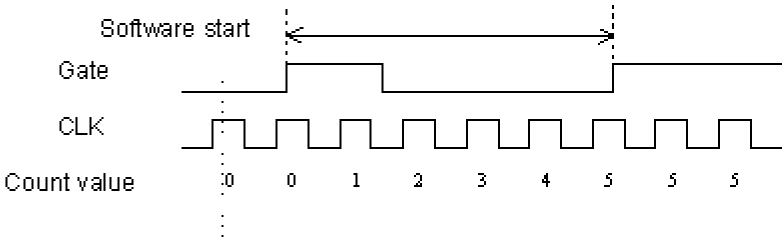

Figure 4-24: Mode 2 Operation

Mode 3: 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 out-puts high and then current count value can be read-back by software. Figure 4-25 illustrates the operation where initial count = 0, count-up mode.

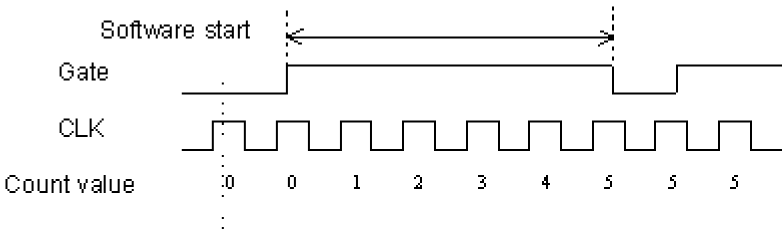

Figure 4-25: Mode 3 Operation

Mode 4: 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 4-26 illustrates the generation of a single pulse with a pulse delay of two and a pulse-width of four.

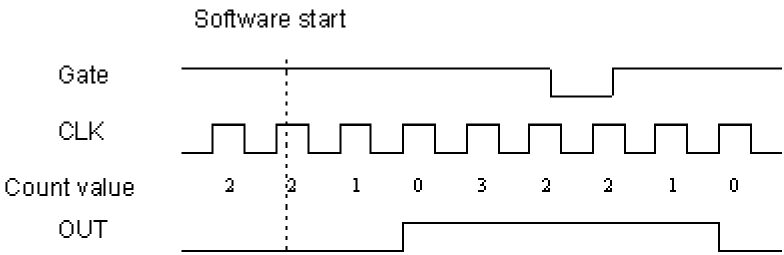

Figure 4-26: Mode 4 Operation

Mode 5: Single Triggered Pulse Generation

This function generates a single pulse with programmable delay and pro-grammable 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 4-27 illustrates the generation of a single pulse with a pulse delay of two and a pulse-width of four.

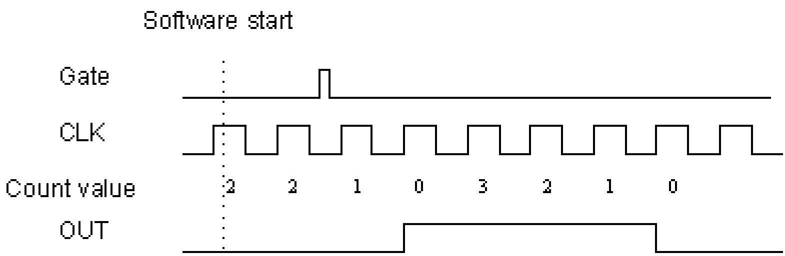

Figure 4-27: Mode 5 Operation

Mode 6: 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. Any GPTC_GATE triggers that occur when the prior pulse is not completed would be ignored. Figure 4-28 illustrates the generation of two pulses with a pulse delay of two and a pulse-width of four.

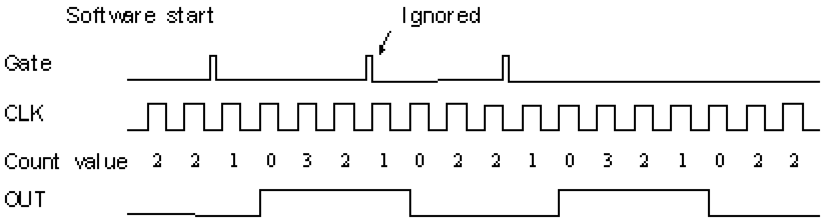

Figure 4-28: Mode 6 Operation

Mode 7: 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 4-29 illustrates the generation of two pulses with a pulse delay of four and a pulse-width of three.

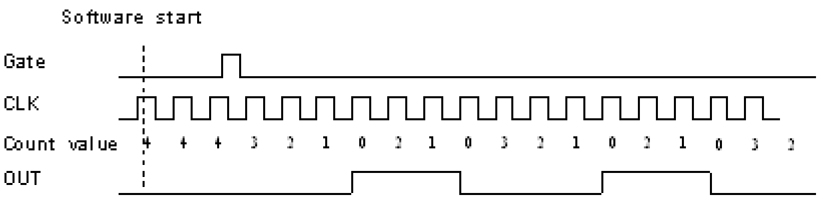

Figure 4-29: Mode 7 Operation

Mode 8: 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 4-30 illustrates the generation of two pulses with a pulse delay of four and a pulse-width of three.

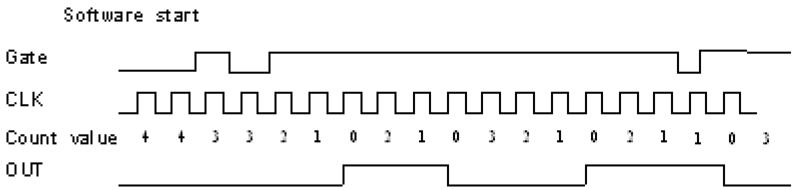

Figure 4-30: Mode 8 Operation

4.5 Trigger Sources

We provide flexible trigger selections in DAQe-2010/2016. In addition to the internal software trigger, DAQe-2010/2016 also supports external analog, digital triggers and SSI triggers. You 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.

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 can receive an individual software trigger.

External Analog Trigger

The analog trigger circuitry routing is shown in the Figure 4-31. The analog multiplexer can select either a direct analog input from the EXTATRIG pin (SRC1 in Figure 4-31) in the 68-pin connector or the input signal of ADC (SRC2 in Figure 4-31). That is, one of the 4 channel inputs you can select as a trigger source). Both trigger sources can be used for all trigger modes. The range of trigger level for SRC1 is ±10V and the resolution is 78mV (Refer to Table 4-6), while the trigger range of SRC2 is the full-scale range of the selected channel input, and the resolution is the desired range divided by 256. For example, if the channel input selected to be the trigger source is set bipolar and ±5V range, the trigger voltage would be 4.96V when the trigger level code is set to 0xFF while 0V when the code is set to 0x80.

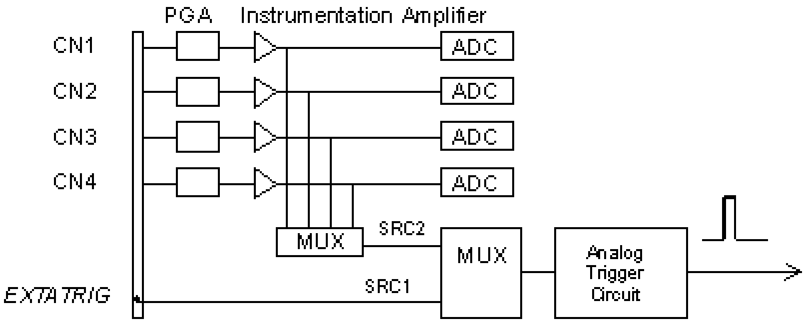

Figure 4-31: Analog trigger block diagram

Trigger Level digital setting	Trigger voltage
0xFF	9.92V
0xFE	9.84V
---	---
0x81	0.08V
0x80	0
0x7F	-0.08V
---	---
0x01	-9.92V

Table 4-7: Analog trigger SRC1 (EXTATRIG) ideal transfer characteristic

The trigger signal is generated when the analog trigger condition is satis-fied. There are five analog trigger conditions in the DAQe-2010/2016. The DAQe-2010/2016 uses 2 threshold voltages, Low_Threshold and High_Threshold to build the 5 different trigger conditions. You may configure the trigger conditions easily via software.

Below-Low analog trigger condition

Figure 4-32 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.

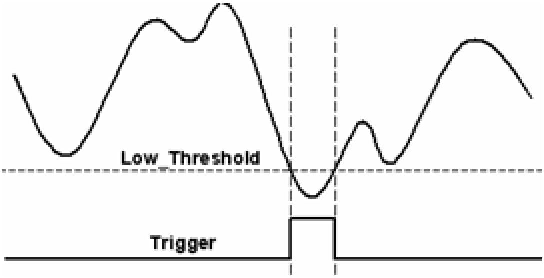

Figure 4-32: Below-Low analog trigger condition

Above-High analog trigger condition

Figure 4-33 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.

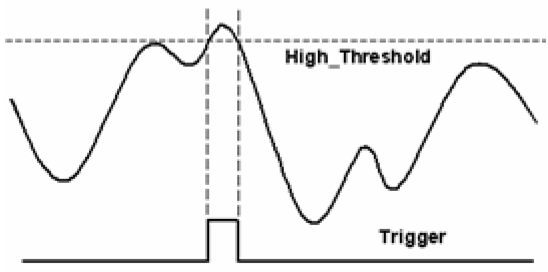

Figure 4-33: Above-High analog trigger condition

Inside-Region analog trigger condition

Figure 4-34 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. Note: the High_Threshold setting should be always higher then the Low_Threshold voltage setting.

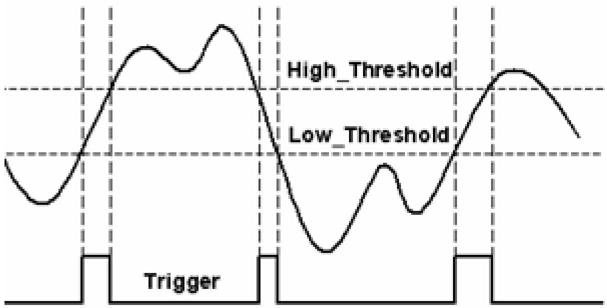

Figure 4-34: Inside-Region analog trigger condition

High-Hysteresis analog trigger condition

Figure 4-35 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 Low_Threshold voltage determines the hysteresis duration. Note the High_Threshold setting should be always higher then the Low_Threshold voltage setting.

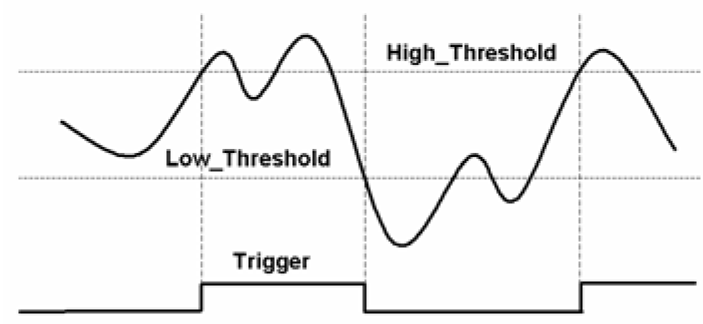

Figure 4-35: High-Hysteresis analog trigger condition

Low-Hysteresis analog trigger condition

Figure 4-36 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

High_Threshold voltage determines the hysteresis duration. Note the High_Threshold setting should be always higher then the Low_Threshold voltage setting.

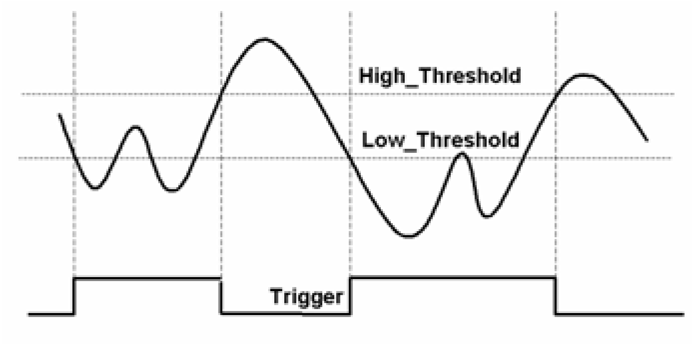

Figure 4-36: Low-Hysteresis analog trigger condition

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 EXTDTRIG or the EXTWFTRG of the 68-pin connector for external digital trigger. The EXTDTRIG is dedicated for A/D process, and the EXTWFTRG is used for D/A process. Users can program the trigger polarity through ADLINK's software drivers easily. Note that the signal level of the external digital trigger signals should be TTL-compatible, and the minimum pulse is 20ns.

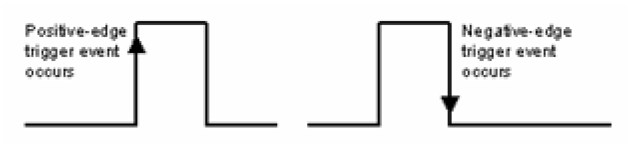

Figure 4-37: External digital trigger

4.6 User-controllable Timing Signals

In order to meet the requirements for user-specific timing and the re-quirements for synchronizing multiple cards, the DAQe-2010/

2016 card provides flexible user-controllable timing signals to connect to external circuitry or additional cards.

The whole DAQ timing of the DAQe-2010/2016 series is composed of a bunch of counters and trigger signals in the FPGA. These timing signals are related to the A/D, D/A conversions and Timer/Counter applications. These timing signals can be inputs to or outputs from the I/O connectors and SSI connector. Therefore the internal timing signals can be used to control external devices and/or circuitry.

We implemented signal multiplexers in the FPGA to individually choose the desired timing signals for the DAQ operations, as shown in the Figure 4-38.

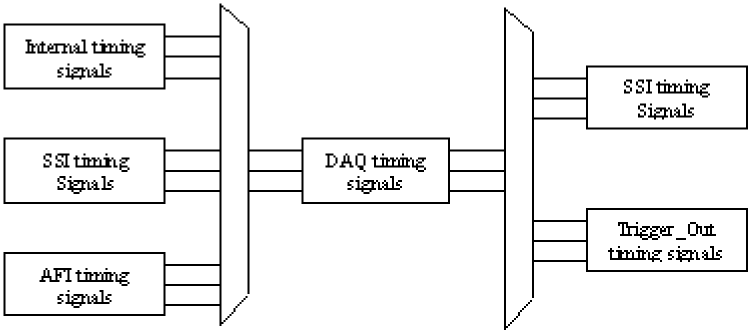

Figure 4-38: DAQ signals routing

Users can utilize the flexible timing signals through our software drivers, and simply and correctly connect the signals with the DAQe-2010/2016 card. Here is the summary of the DAQ timing signals and the corresponding functionalities:

Timing signal category	Corresponding functionality
SSI signals	Multiple cards synchronization
AFI signals	Control DAQe-2010/2016 by external timing signals
AI_Trig_Out, AO_Trig_Out	Control external circuitry or boards

Table 4-8: Summary of user-controllable timing signals and the corresponding functionalities

DAQ timing signals

The user-controllable internal timing-signals contain: (Refer to section **4.1 A/D Conversion** for internal timing signal definitions)

1. TIMEBASE, providing TIMEBASE for all DAQ operations, which could be from internal 40 MHz oscillator, EXTTIMEBASE from I/O connector or the SSI_TIMEBASE. Note that the frequency range of the EXTTIMEBASE is 1 MHz to 40 MHz, and the EXTTIMEBASE should be TTL-compatible.
2. AD_TRIG, the trigger signal for the A/D operation, which could come from external digital trigger, analog trigger, internal software trigger and SSI_AD_TRIG. Refer to section **4.5 Trigger Sources** for detailed description.
3. SCAN_START, the signal to start a scan, which would bring the following ADCONV signals for AD conversion, and could come from the internal SI_counter, AFI[0] and SSI_AD_START. This signal is synchronous to the TIMEBASE. Note that the AFI[0] should be TTL-compatible and the minimum pulse width should be the pulse width of the TIMEBASE to guarantee correct functionalities.
4. ADCONV, the conversion signal to initiate a single conversion, which could be derived from internal counter, AFI[0] or SSI_ADCONV. Note that this signal is edge-sensitive. When using AFI[0] as the external ADCONV source, each rising edge of AFI[0] would bring an effective conversion signal. Also note that the AFI[0] signal should be TTL-compatible and the minimum pulse width is 20ns.
5. DA_TRIG, the trigger signal for the D/A operation, which could be derived from external digital trigger, analog trigger, internal software trigger and SSI_AD_TRIG. Refer to section **4.5 Trigger Sources** for detailed description.
6. DAWR, the update signal to initiate a single D/A conversion, which could be derived from internal counter, AFI[1] or SSI_DAWR. Note that this signal is edge-sensitive. When using AFI[1] as the external DAWR source,

each rising edge of AFI[1] would bring an effective update signal. Also note that the AFI[1] signal should be TTL-compatible and the minimum pulse width is 20ns.

Auxiliary Function Inputs (AFI)

Users could use the AFI in applications that take advantage of external circuitry to directly control the DAQe-2010/2016 card. The AFI includes 2 categories of timing signals: one group is the dedicated input, and the other is the multi-function input. Table 4-9 illustrates this categorization.

Table 4-9 summarizes the auxiliary function input signals and the corresponding functionalities

Category	Timing signal	Functionality	Constraints
Dedicated input	EXTTIMEBASE	Replace the internal TIMEBASE	1. TTL-compatible
			2. 1 MHz to 40 MHz
			3. Affects on both A/D and D/A operations
	EXTDTRIG	External digital trigger input for A/D operation	1. TTL-compatible
			2. Minimum pulse width = 20ns
			3. Rising edge or falling edge
	EXTWFTRG	External digital trigger input for D/A operation	1. TTL-compatible
			2. Minimum pulse width = 20ns
			3. Rising edge or falling edge

Table 4-9: Auxiliary function input signals and the corresponding functionalities

Category	Timing signal	Functionality	Constraints
Multi-function input	AFI[0] (Dual functions)	Replace the internal ADCONV	1. TTL-compatible
			2. Minimum pulse width = 20ns
			3. Rising–edge sensitive only
		Replace the internal SCAN_START	1. TTL-compatible
			2. Minimum Pulse width > 2/TIMEBASE
	AFI[1]	Replace the internal DAWR	1. TTL-compatible
			2. Minimum pulse width = 20ns
			3.Rising–edge sensitive only

Table 4-9: Auxiliary function input signals and the corresponding functionalities

EXTDTRIG and EXTWFTRIG

EXTDTRIG and EXTWFTRIG are dedicated digital trigger input signals for A/D and D/A operations respectively. Please refer to section **4.5 Trigger Sources** for detailed descriptions.

EXTTIMEBASE

When the applications needs specific sampling frequency or update rate that the card could not generate from its internal TIMEBASE, the 40 MHz clock, users could utilize the EXTTIMEBASE with internal counters to achieve the specific timing intervals for both A/D and D/A operations. Note that once you choose the TIMEBASE source, both A/D and D/A operations will be affected because A/D and D/A operations share the same TIMEBASE.

AFI[0]

Alternatively, users can also directly apply an external A/D conversion signal to replace the internal ADCONV signal. This is another way to achieve customized sampling frequencies. The external ADCONV signal can only be inputted from the AFI[0]. As section 4.1 describes, the SI_counter triggers the generation of the A/D conversion signal, ADCONV, but when using the AFI[0] to replace the internal ADCONV signal, then the SI_counter and the internally generated SCAN_START will not be effective. By controlling the ADCONV externally, users can sample the data according to external events. In this mode, the Trigger signal and trigger mode settings will are not available.

AFI[0] could also be used as SCAN_START signal for A/D operations. Please refer to sections **4.1 A/D Conversion** and **DAQ timing signals** for detailed descriptions of the SCAN_START signal. When using external signal (AFI[0]) to replace the internal SCAN_START signal, the pulse width of the AFI[0] must be greater than two time of the period of Timebase. This feature is suitable for the DAQe2010/2016 card, which can scan multiple channels data controlled by an external event. Note that the AFI[0] is a multi-purpose input, and it can only be utilized for one function at any one time.

AFI[1]

Regarding the D/A operations, users could directly input the external D/A update signal to replace the internal DAWR signal. This is another way to achieve customized D/A update rates. The external DAWR signal can only be inputted from the AFI[1]. Note that the AFI[1] is a multi-purpose input, and it can only be utilized for one function at any one time. AFI[1] currently only has one function.

System Synchronization Interface

SSI (System Synchronization Interface) provides the DAQ timing synchronization between multiple cards. The DAQe-2010/2016 card has a designed bi-directional SSI I/O to provide flexible connection between cards and allow one SSI master to output the signal and up to three slaves to receive the SSI signal. Note that the SI signals are designed for card synchronization only and not for external devices.

.

SSI timing signal	Functionality
SSI_TIMEBASE	SSI master: send the TIMEBASE out
	SSI slave: accept the SSI_TIMEBASE to replace the internal TIMEBASE signal.
	Note: Affects on both A/D and D/A operations
SSI_AD_TRIG	SSI master: send the internal AD_TRIG out
	SSI slave: accept the SSI_AD_TRIG as the digital trigger signal.
SSI_ADCONV	SSI master: send the ADCONV out
	SSI slave: accept the SSI_ADCONV to replace the internal ADCONV signal.
SSI_SCAN_START	SSI master: send the SCAN_START out
	SSI slave: accept the SSI_SCAN_START to replace the internal SCAN_START signal.
SSI_DA_TRIG	SSI master: send the DA_TRIG out.
	SSI slave: accept the SSI_DA_TRIG as the digital trigger signal.
SSI_DAWR	SSI master: send the DAWR out.
	SSI slave: accept the SSI_DAWR to replace the internal DAWR signal.

Table 4-10: Summary of SSI timing signals and the corresponding functionalities as the master or slave

In PCI/PCI Express form factor, there is a connector on the top right corner of the card for SSI. Refer to section 2.3 for the connector position. All SSI signals are routed to the 20-pin connector from the FPGA. To synchronize multiple cards, users can connect a special ribbon cable (ACL-SSI) to all the cards in a daisy-chain configuration

The 6 internal timing signals could be routed to the SSI through software drivers. Please refer to the **DAQ timing signals** section for detailed information. Physically the signal routings are accomplished in the FPGA. Cards that are connected together through the SSI will still achieve synchronization on the 6 timing signals.

The mechanism of the SSI

1. We adopt master-slave configuration for SSI. In a system, for each timing signal, there shall be only one master, and other cards are SSI slaves or with the SSI function disabled.
2. For each timing signal, the SSI master need not be in a single card

For example:

We want to synchronize the A/D operation through the ADCONV signal for 4 DAQe-2010/2016 cards. Card 1 is the master, and Card 2, 3, 4 are slaves. Card 1 receives an external digital trigger to start the post trigger mode acquisition. The SSI setting could be:

1. Set the SSI_ADCONV signal of Card 1 to be the master.
2. Set the SSI_ADCONV signals of Card 2, 3, 4 to be the slaves.
3. Set external digital trigger for Card 1's A/D operation.
4. Set the SI_counter and the post scan counter (PSC) of all other cards.
5. Start DMA operations for all cards, thus all the cards are waiting for the trigger event.

When the digital trigger condition of Card 1 occurs, Card 1 will internally generate the ADCONV signal and output this ADCONV signal to SSI_ADCONV signal of Card 2, 3 and 4 through the SSI connectors. Thus we can achieve 16-channel acquisition simultaneously.

You could arbitrarily choose each of the 6 timing signals as the SSI master from any one of the cards. The SSI master can output the internal timing signals to the SSI slaves. With the SSI, users could achieve better card-to-card synchronization.

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

AI_Trig_Out and AO_Trig_Out

AI_Trig_Out (or AO_Trig_Out) is the signal output following one of the four trigger sources (software trigger, analog trigger, digital trigger and SSI trigger) selected by the user. That is, AI_Trig_Out follows the A/D trigger source, and AO_Trig_Out follows the D/A trigger source. These two sig-nals can be used to control external peripheral circuits or boards, or can be used as synchronization control signals. The signal level of the AI_Trig_Out and AO_Trig_Out are TTL-compatible.

Note: AI_Trig_Out and AO_Trig_Out are output pins on J5 (68-pin VHDCI). Connecting them to any signal source may cause per-manent damage.

5 Calibration

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

5.1 Loading Calibration Constants

The DAQe-2010/2016 card is factory calibrated before shipment by writing the associated calibration constants of TrimDACs to the onboard EEPROM. TrimDACs are devices containing 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 of loading the values of TrimDACs stored in the onboard 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-calibrated 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 errors in measurements and outputs will vary with time and temperature, it is recommended to re-calibratetion the card when it is installed in the user's environment. The auto-calibration function is introduced in the following sections.

5.2 Auto-calibration

By using the auto-calibration feature of the DAQe-2010/2016, the calibration software can measure and correct almost all the calibration errors without any external signal connections, reference voltages, or measurement devices.

The DAQe-2010/2016 has an onboard 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 utra-precision calibrator. The impedance of the digital potentiometer is memorized after this adjustment. It is not recommended for users to adjust the onboard calibration reference except when an ultra-precision calibrator is available.

Note:

1. Before auto-calibration procedure starts, it is recommended to warm up the card for at least 15 minutes.
2. Remove the cable before an auto-calibrating the card since the DA outputs would be changed during calibration.

5.3 Saving Calibration Constants

After completing auto-calibration, you can save the new calibration constants into one of the three user-modifiable banks in the EEPROM. The date and the temperature when you ran the auto-calibration will be saved with the calibration constants. This means that you can store three sets of calibration constants according to three different environments and re-load the calibration constants later.

Warranty Policy

Thank you for choosing ADLINK. To understand your rights and enjoy all the after-sales services we offer, please read the following carefully.

1. Before using ADLINK's products please read the user manual and follow the instructions exactly. When sending in damaged products for repair, please attach an RMA application form which can be downloaded from: http://rma.adlinktech.com/policy/home.htm.
2. All ADLINK products come with a limited two-year warranty, one year for products bought in China:
   - The warranty period starts on the day the product is shipped from ADLINK's factory.
   - Peripherals and third-party products not manufactured by ADLINK will be covered by the original manufacturers' warranty.
   - For products containing storage devices (hard drives, flash cards, etc.), please back up your data before sending them for repair. ADLINK is not responsible for any loss of data.
   - Please ensure the use of properly licensed software with our systems. ADLINK does not condone the use of pirated software and will not service systems using such software. ADLINK will not be held legally responsible for products shipped with unlicensed software installed by the user.
   - For general repairs, please do not include peripheral accessories. If peripherals need to be included, be certain to specify which items you sent on the RMA Request & Confirmation Form. ADLINK is not responsible for items not listed on the RMA Request & Confirmation Form.

3. Our repair service is not covered by ADLINK's guarantee in the following situations:
   - Damage caused by not following instructions in the User's Manual.
   - Damage caused by carelessness on the user's part during product transportation.
   - Damage caused by fire, earthquakes, floods, lightening, pollution, other acts of God, and/or incorrect usage of voltage transformers.
   - Damage caused by unsuitable storage environments (i.e. high temperatures, high humidity, or volatile chemicals).
   - Damage caused by leakage of battery fluid during or after change of batteries by customer/user.
   - Damage from improper repair by unauthorized ADLINK technicians.
   - Products with altered and/or damaged serial numbers are not entitled to our service.
   - This warranty is not transferable or extendible.
   - Other categories not protected under our warranty.
4. Customers are responsible for shipping costs to transport damaged products to our company or sales office.
5. To ensure the speed and quality of product repair, please download an RMA application form from our company website: http://rma.adlinktech.com/policy. Damaged products with attached RMA forms receive priority.

If you have any further questions, please email our FAE staff: service@adlinktech.com.