Eagle Technologies

PCI-703-16/64A DAQ Module

Technical Manual

Revision 3.00 November 2000

TABLE OF CONTENTS

TABLE OF FIGURES & TABLES

1. Module Features

- The PCI-64 DAQ module is PCI 2.2 compliant at 32 bits and 33 Mhz.
- The modules are 3.3V PCI slot compatible.
- I/O Connector compatible with and functionally similar to the National Instruments PCI-6071E module
- The modules support both master and target PCI functions with an integrated programmable DMA controller.
- There are three PCI-703 DAQ module option ranges. The ranges and data acquisition features and functions of each are defined in Table 1.

Feature	PCI-703-16	PCI-703-16A	PCI-703-64A
Number of Analogue Input Channels	16	16	64
Number of Analogue Output Channels	0	2	2
A/D Resolution (bits) @ 400 Khz max.	14	14	14
A/D Sample FIFO Depth	4096	4096	4096
A/D Channel List FIFO Depth	4096	4096	4096
D/A Resolution (bits) @ 100 Khz	-	14	14
D/A FIFO Depth (programmable)	-	4096	4096
D/A waveform generation capability	-	YES	YES
Triggering Capability.	Internal, External. Analogue & Digital. Post & Pre-Trigger.	Internal, External. Analogue & Digital. Post & Pre-Trigger.	Internal, External. Analogue & Digital. Post & Pre-Trigger.
Counter Timers – 16 bit.	3	3	3
Frequency Output Timer, or Counter Timer pre-scaler – 8 bit.	1	1	1
Number of Digital I/O Lines (TTL)	8	8	8
Number of programmable function I/O's	10	10	10

Table 1 PCI-703-XX Module Options

2. Hardware Overview

This chapter provides a hardware overview of the PCI-64 DAQ module. A block diagram of the module is shown in Figure 1.

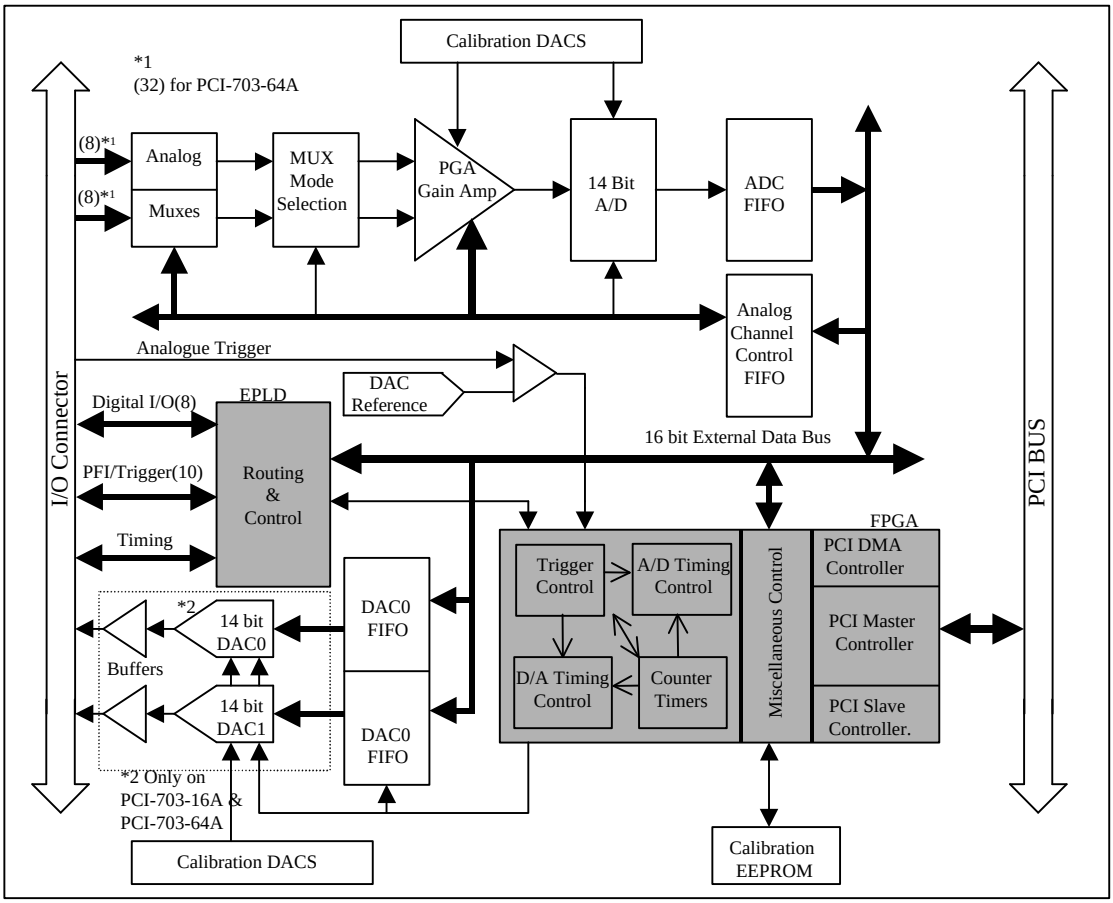

Figure 1 PCI-703-XX Functional Block Diagram

2.1 Module Address Map

The module uses the Actel Target + DMA PCI core to interface to the PCI bus. This core thus defines much of the software protocol used to access the module.

The PCI-703 module's control registers and FIFO' s are mapped as per Table 2.

Configuration register BAR0 used as the base.		
Offset Address (Hex)	**Register Name**	**Description**
00	A/D_FIFO_DATA	Read/Write. Contains the A/D conversion results. Only bits <13:0> are valid.
04	A/D_CHANNEL_LIST	Read/Write. Write data to A/D Channel List FIFO. Only bits <13:0> are valid.
08	DAC_FIFO(0)	Read/Write. Write data to D/A Channel 0 FIFO. Only bits <13:0> are valid.
0C	DAC_FIFO(1)	Read/Write. Write data to D/A Channel 1 FIFO. Only bits <13:0> are valid.
3C:20	GAIN_CALIBRATION_BUFFER	Read/Write. 8 words x 14 bits. Contains the gain calibration constants for the PGIA D/A. Only bits <13:0> are valid.
Configuration register BAR1 used as the base.		
Offset Address (Hex)	**Register Name**	**Description**
100	DAC_CNTRL(0)	DAC0 Control Register
104	DAC_CNTRL(1)	DAC1 Control Register
200	CNTR_CNTRL(0)	Counter 0 Control Register
208	CNTR_DIVR(0)	Counter 0 Divisor Register
210	CNTR_CNTRL(1)	Counter 1 Control Register
218	CNTR_DIVR(1)	Counter 1 Divisor Register
220	CNTR_CNTRL(2)	Counter 2 Control Register
228	CNTR_DIVR(2)	Counter 2 Divisor Register
23(X)	FREQ_CNTRL (PRESCALER)	FREQ_OUT Control Register
300	A/D_CNTRL	A/D Control Register

400	DIO_CONFIG_REG	Controls the direction of the DIO pins.
410	DIO_DATA_REG	DIO write or read data register
420	PIO_CONFIG_REG	Controls the direction of the PIO pins.
430	PIO_DATA_REG	PIO write or read data register
500	GLOBAL_STATUS_REGISTER	Configuration, Interrupt and Status Flags register

Table 2 PCI-703 Register Address Map

2.2 Global Status Register

This read/write register defines bits used to configure and monitor the status of the module. With the exception of the data FIFO' s it is the only register that can be read by the host firmware – simplifying the FPGA interface.

The bit definitions are defined in Table 3.

Global Status Register bit configuration:		
Bit 0	Global Interrupt Enable	Enables or disables all interrupts from the module. "0" : Interrupts are disabled. No PCI INTA can be asserted. "1" : Interrupts are enabled. Note that the various interrupt sources are individually configured using the specific functions Control Register. However, the status of these individual interrupts can only be accessed via this register.
Bit 1	A/D FIFO_EMPTY_STAT	Reflects the status of the A/D Data FIFO's Empty Flag. (Read only) "0" : The FIFO is not empty "1" : The FIFO is empty.
Bit 2	Channel_List_Done_STAT	Gets set to '1' whenever the last command in the channel list has been executed. Used by the host to know when the number of A/D samples defined in the Channel List FIFO have been converted. The A/D result buffer can now be read. Cleared when the host write a '0' to this bit.
Bit 3	A/D FIFO_EMPTY_INTR	Set whenever A/D FIFO_EMPTY_STAT changes to the NOT EMPTY state if the corresponding interrupt is enabled in the A/D Control Register. Cleared by writing a zero to this bit.
Bit 4	Channel_List_Done_INTR	Set whenever bit (2) is set and the corresponding interrupt is enabled in the A/D Control Register. Cleared when bit(2) or the Interrupt enable is cleared.
Bit 5	DAC(0) FIFO_EMPTY_STAT	Reflects the status of the DAC(0) Data FIFO's Empty Flag. (Read only) "0" : The FIFO is not empty "1" : The FIFO is empty.
Bit 6	DAC(0) FIFO_HALF_STAT	Reflects the status of the DAC(0) Data FIFO's Half Full Flag. (Read only) "0" : The FIFO level is less than half full. "1" : The FIFO level has reached at least half full

Global Status Register bit configuration:		
Bit 7	DAC(0) FIFO_EMPTY_INTR	Set whenever DAC(0) FIFO_EMPTY_STAT changes to the EMPTY state if the corresponding interrupt is enabled in the A/D Control Register. Cleared by writing a zero to this bit.
Bit 8	DAC(0) FIFO_HALF_INTR	Set whenever DAC(0) FIFO_HALF_STAT changes to the LESS THAN HALF EMPTY state if the corresponding interrupt is enabled in the A/D Control Register. Cleared by writing a zero to this bit.
Bit 9	DAC(1) FIFO_EMPTY_STAT	Reflects the status of the DAC(1) Data FIFO's Empty Flag. (Read only) "0" : The FIFO is not empty "1" : The FIFO is empty.
Bit 10	DAC(1) FIFO_HALF_STAT	Reflects the status of the DAC(1) Data FIFO's Half Full Flag. (Read only) "0" : The FIFO level is less than half full. "1" : The FIFO level has reached at least half full
Bit 11	DAC(1) FIFO_EMPTY_INTR	Set whenever DAC(1) FIFO_EMPTY_STAT changes to the EMPTY state if the corresponding interrupt is enabled in the A/D Control Register. Cleared by writing a zero to this bit.
Bit 12	DAC(1) FIFO_HALF_INTR	Set whenever DAC(1) FIFO_HALF_STAT changes to the LESS THAN HALF EMPTY state if the corresponding interrupt is enabled in the A/D Control Register. Cleared by writing a zero to this bit.
Bit 13	CNTR(0) EXPIRED	Set whenever Counter Register 0 has reached it's terminal count if the corresponding interrupt is enabled in the Counter Control Register. Cleared by writing a zero to this bit.
Bit 14	CNTR(1) EXPIRED	Set whenever Counter Register 1 has reached it's terminal count if the corresponding interrupt is enabled in the Counter Control Register. Cleared by writing a zero to this bit.
Bit 15	CNTR(2) EXPIRED	Set whenever Counter Register 2 has reached it's terminal count if the corresponding interrupt is enabled in the Counter Control Register. Cleared by writing a zero to this bit.
Bit 16	EEPROM_ClK	Maps directly to the EEPROM' s CLK pin. (Read/Write)
Bit 17	EEPROM_CS	Maps directly to the EEPROM' s CS pin. (Read/Write)
Bit 18	EEPROM_DI	Maps directly to the EEPROM' s DI pin. (Read/Write)

Global Status Register bit configuration:		
Bit 19	EEPROM_DO	Maps directly to the EEPROM' s DO pin. (Read/Write)
Bit 20	CAL_DAC_CLK	Maps directly to the Calibration DAC' s CLK pin. (Read/Write)
Bit 21	CAL_DAC_CS	Maps directly to the Calibration DAC' s CS pin. (Read/Write)
Bit 22	CAL_DAC_LD	Maps directly to the Calibration DAC' s LD pin. (Read/Write)
Bit 23	CAL_DAC_SDI	Maps directly to the Calibration DAC' s SDI pin. (Read/Write)

Table 3 Global Status Register Bit Definitions

2.3 DIO I/O Signals

The module supports eight programmable digital input/ output signals mapped to the I/O connector DIO(7:0) pins. These pins are configured as per Table 4 and Table 5.

DIO_CONFIG_REG:		
Bits (7:0)		Defines the bits as inputs or outputs. (Write Only) Bit Definition: “0” : The corresponding bit is an input. This is the reset state. “1” : The corresponding bit is an output.

Table 4 DIO Configuration Register bit definitions

DIO_DATA_REG:		
Bits (7:0)		WRITE: Provides output data for all bits configured as OUTPUTS READ: Provides the I/O state of all pins.

Table 5 DIO Data Register bit definitions

2.4 PIO I/O Signals

The module supports ten programmable digital multifunction input/output signals mapped to the I/O connector PIO(9:0) pins. From a software viewpoint these signals behave like normal I/O signals. Their multifunction nature occurs when other functions are configured. For example if PFI(2)/CONVERT is used in the A/D setupt then PFI(2) must be configured as an input. These pins are configured as per Table 6 and Table 7.

PIO_CONFIG_REG:		
Bits (9:0)		Defines the bits as inputs or outputs. (Write Only) Bit Definition: “0” : The corresponding bit is an input. This is the reset state. “1” : The corresponding bit is an output.

Table 6 PIO Configuration Register bit definitions

PIO_DATA_REG:		
Bits (9:0)		WRITE: Provides output data for all bits configured as OUTPUTS READ: Provides the I/O state of all pins.

Table 7 PIO Data Register bit definitions

2.5 Analogue Input

A block diagram of the analogue input configuration is presented in Figure 2.

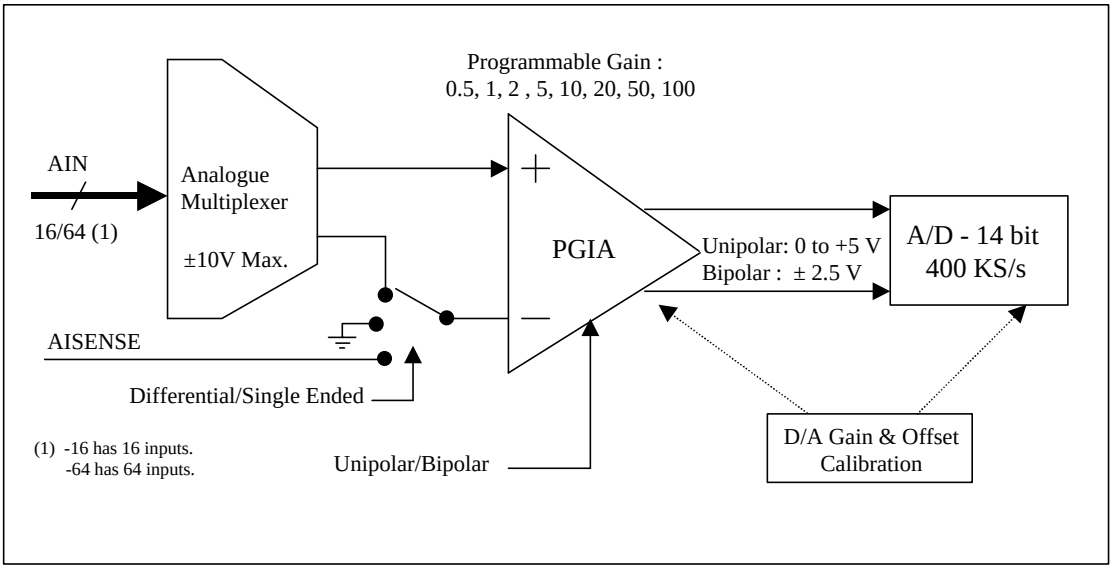

Figure 2 PCI-703- XX Analogue Input Configuration

2.5.1 Analogue Multiplexer

A block diagram of the analogue multiplexer section is shown in Figure 3. Note that this drawing only depicts the multiplexer structure for sixteen analogue input channels. The sixty four input architecture is similar the difference being that there are a total of eight rather than two input multiplexers i.e. the “EnGrp(x)” numbers run from 0 to 7. Although not shown in Figure 3 there is also a built in test multiplexer that is enabled with the control signal “EnGrp(8)”.

The bank on analogue input multiplexers on the left are always configured as differential inputs and used to select one of thirty two input pairs. Note that a differential pair is always configured as AIN(X) and AIN(X +8). The control signals MUX_A(1:0) are common to all input multiplexers and a specific group is selected using the controls signals EnGrp(8:0). The single analogue multiplexer on the right uses control signals MUX_A(01:00) right to select the final channel configuration as follows:

“00” Selects the differential pair as enabled by EnGrp(8:0).
“01” Selects the single ended input selected by the top half on the primary input multiplexer.
“10” Selects the single ended input selected by the bottom half on the primary input multiplexer.
“11” Selects a differential Analogue ground input used for offset adjustment and calibration.

In single ended mode the analogue signal can either be referenced to a common signal “AISENSE” available on the I/O connector or to the DAQ modules internal analogue ground. This option is controlled by the control signal “RSE_MODE”.

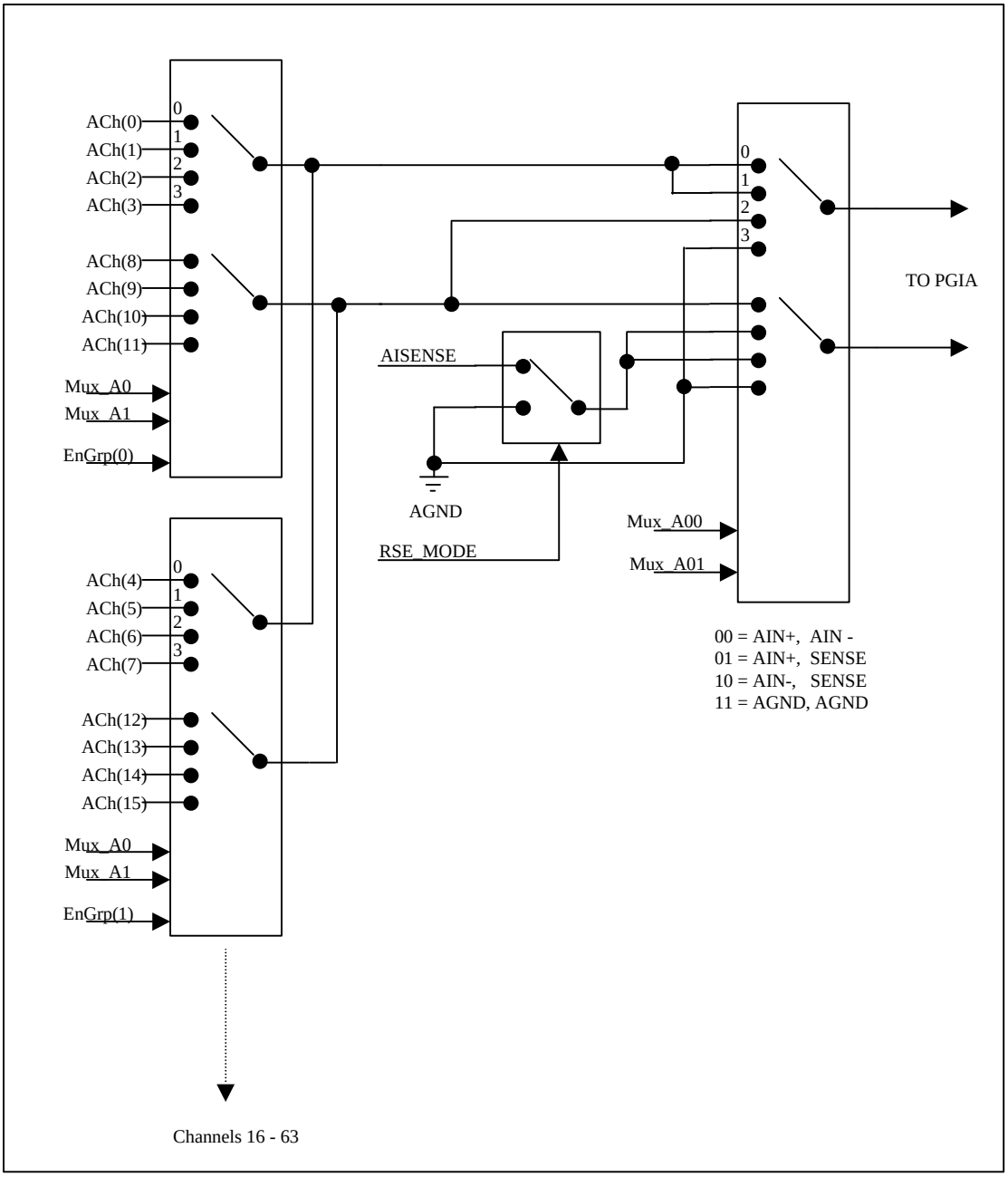

Figure 3 Analogue Multiplexer Architecture

Note that the analogue input multiplexer configuration is controlled by configuration bits in the Channel List FIFO. The EnGrp(7:0) signals are generated by the FPGA by decoding ChanSel(4:2). Refer to paragraph 2.5.4 .

2.5.2 PGIA

The Programmable Gain Instrumentation Amplifier is actually implemented as per Figure 4.

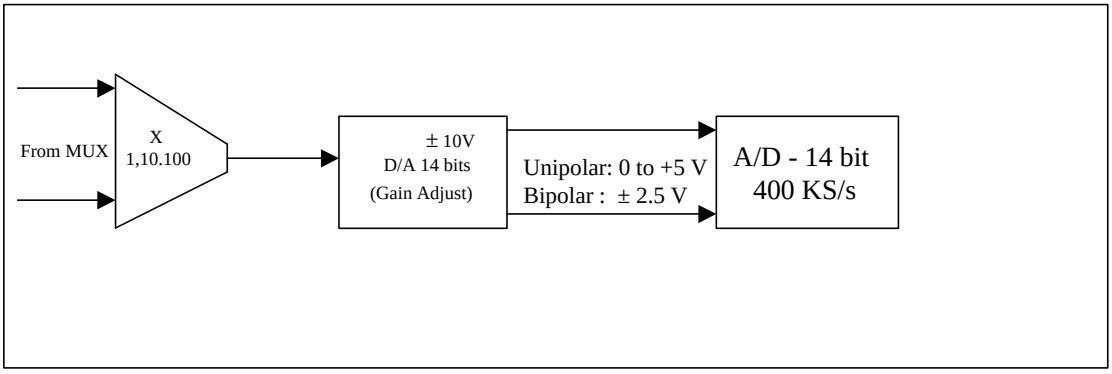

Figure 4 PGIA Implementation Architecture

The fourteen bit D/A is used as programmable attenuator and is used to scale the voltage to the A/D into a ±2.5 or 0 to +5V full scale signal. Because of gain errors within the amplifier stage, the D/A and the A/D itself the D/A scale factor is adjusted using calibration constants stored within the EEPROM. All gain errors within the channel can thus be calibrated out. The gain selection bits in the Channel List FIFO are used by the FPGA to automatically load the applicable gain selection calibration constant into the D/A.

ONE: Input voltage of ± 10V and a gain factor of 0.25

The PGA is set to a gain of 1.
The D/A scale factor is set to 0.25, resulting in a ± 2.5V full scale signal.

TWO: Input voltage of ± 100 mV and a gain factor of 25.

The PGA is set to a gain of 100.
The D/A scale factor is set to 0.25, resulting in a ±2.5V full scale signal.

As the accuracy of the various gain combinations is primary determined by 0.1% resistors these errors need to be calibrated out to within the resolution of the A/D. The D/A is used to scale these to within 1 LSB of the ideal.

Ignoring gain errors the D/A constants for the various gain options are defined in Table 8.

Channel List **Gain (Max. Input)**	**PGA GAIN**	**D/A Constant** **Output = Input(Constant/16384)**
0.25 (±10V)	1	4096
0.50 (±5V)	1	8192
1.0 (±2.5V)	1	16384
2.5 (±1.0V)	10	4096
5 (±500mV)	10	8192
10 (±250mV)	10	16384

Channel List Gain (Max. Input)	PGA GAIN	D/A Constant Output = Input(Constant/16384)
25 (±100mV)	100	4096
50 (±50mV)	100	8192

Table 8 Gain vs D/A Constants

2.5.3 Gain Calibration Buffer

As there are eight programmable gain settings, eight D/A scale/calibration constants are required. These are stored in the serial EEPROM during the module's calibration process. During module initialisation these serial constants need to be read by the host and made available to the A/D state controller in parallel format as they are accessed continuously together with the Channel List FIFO data. The FPGA supports a special 8 x 14 bit data buffer into which these constants should be stored. The format of this buffer is as per Table 9.

Offset into Gain Calibration Buffer	Constant Stored
Index 0	Calibration constant for Gain = 0.25
Index 1	Calibration constant for Gain = 0.5
Index 2	Calibration constant for Gain = 1
Index 3	Calibration constant for Gain = 2.5
Index 4	Calibration constant for Gain = 5
Index 5	Calibration constant for Gain = 10
Index 6	Calibration constant for Gain = 25
Index 7	Calibration constant for Gain = 50

Table 9 D/A Calibration Constant Table

2.5.4 Channel List FIFO

The Channel List FIFO is a circular buffer with a depth equal to half that of the programmable depth A/D DATA FIFO. Each entry in this circular buffer is used to configure the channel parameters (channel number, gain etc) for the next scheduled A/D sample. The A/D state controller in the FPGA repeatedly loops through this buffer converting the selected channel as per the setup configuration. This list must be configured and setup by the host software before the A/D scanning is enabled. Although the FIFO depth is the same as that of the A/D DATA FIFO normally the full range cannot be used and the practical depth depends on the latency of the host in reading the A/D DATA FIFO.

In normal operation the Channel List FIFO is loaded "N" (N < 4096) commands. When the A/D is enabled these commands are executed at each A/D time tick. When all N commands have been executed and "N" A/D data results available a Channel List Done status and interrupt flag is set. The host uses this status to unload the "N" results form the A/D data buffer. Meanwhile, the Channel List FIFO wraps and the command execution

repeats with the results again being loaded into the A/D DATA FIFO. It is to ensure that the A/D DATA FIFO never reaches the full state that the Channel List FIFO should typically never contain more than 2048 commands. The format of the Channel List FIFO is as defined in Table 10.

Channel List FIFO Configuration		
Bits (1:0)	Mapped by the FPGA to Mux_A(1:0).	Used in conjunction with BITS (6:2) to select a specific pair of differential input or single ended analogue signals. **Differential:** For example if Bits (6:5) = “00” and Bits (4:2) = “000” then these two bits would select the specific pair as follows: “00”: AIN(0) and AIN(8) “01”: AIN(1) and AIN(9) “10”: AIN(2) and AIN(10) “11”: AIN(3) and AIN(11). **Single Ended**: For example if Bits (6:5) = “01” and Bits (4:2) = “000” then these two bits would select a single input as follows: “00”: AIN(0) “01”: AIN(1) “10”: AIN(2) “11”: AIN(3). For example if Bits (6:5) = “10” and Bits (4:2) = “000” then these two bits would select a single input as follows: “00”: AIN(8) “01”: AIN(9) “10”: AIN(10) “11”: AIN(11). **BIT Selected**: if Bits (6:5) = “11” the BIT signals are selected as follows: “00” : Loopback of DAC 0 Output “01” : Loopback of DAC 1 Output “10”: 4.06V Reference of the A/D. “11”: AGND and AGND.
Bits (4:2)	Mapped by the FPGA to EnGrp(7:0)	Used to select one of eight primary input multiplexers. Mapped to EnGrp(7:0) by the FPGA and selects the differential groups as follows: **Primary Group** **Secondary Group** “000” : AIN(3:0) with pair AIN(11:8) “001” : AIN(7:4) with pair AIN(15:12) “010” : AIN(19:16) with pair AIN(27:24) “011” : AIN(23:20) with pair AIN(31:28) “100” : AIN(35:32) with pair AIN(43:40) “101” : AIN(39:36) with pair AIN(47:44) “110” : AIN(51:48) with pair AIN(59:56) “111” : AIN(55:52) with pair AIN(64:60)

Channel List FIFO Configuration		
Bits (6:5)	Mapped by the FPGA to MUX_A(01:00).	Used to select the operational mode of the channel as follows. “00” : Channel is fully differential “01” : Single ended. Selected from Primary Pair group “10” : Single ended. Selected from Secondary Pair group “11” : Differential input derived from the BIT multiplexer .
Bit 7	RSE_MODE	In Single Ended Mode this signal selects the reference for the signal: “0”: Reference is the external AISENSE signal “1”: Reference is local AGND
Bit 8	BI_POLAR	Selects whether the signal is uni-polar or bi-polar. “0”: Selects uni-polar mode. “1”: Selects bi-polar mode.
Bit 10:9	RESERVED	
Bits(13:11)	Channel Gain	Select the required channel gain as follows: “000” : 0.25 “001” : 0.5 “010” : 1 “011” : 2.5 “100” : 5 “101” : 10 “110” : 25 “111” : 50

Table 10 Channel List FIFO Bit Definitions

2.6 A/D Channel

The A/D conversion process essentially consists of the following events:

1. Wait for trigger event to be met.
2. Index the Channel List FIFO to select the next analogue input source and associated channel parameters.
3. Wait for the next sample clock period.
4. Start the A/D conversion process.
5. Store the results in the A/D Data FIFO
6. Loop to 2.

The key requirements to configure in this process are the trigger event, the Channel List and the source input used to generate the sample clock.

As soon as the A/D state controller is enabled the process loops waiting for a trigger event to occur. This trigger event essentially moves the state controller from idle mode to acquisition mode. Once in this mode a channel is converted synchronously with the sample clock and the Channel List FIFO pointer updated. This process loops until stopped by the host processor.

The trigger event can be generated from any one of the following sources:

1. Always triggered.
2. External input “PFI(0)/TRIG1”.
3. A selected input analogue channel (any of ACh(63..0)) being greater or less than the trigger reference voltage set on DAC Channel 1. This mode includes triggering on LEVEL or EDGE conditions. **Note that for this trigger mode the first entry in the Channel List FIFO must always be the entry that specifies the channel selected as the trigger source.**

The sample clock can be generated from any of the following sources:

1. *Processor using a Programmed I/O bit*: In this mode the processor initiates each conversion by setting a bit in the A/D Control Register. The FIFO EMPTY flag can be used to poll the status of the result. This mode is primarily used for calibration.
2. *Using Counter 2*: In this mode counter two is used to generate the sample clock source. One conversion is performed at each terminal count point. The FIFO HALF FULL flag can be polled to determine when one scan buffer is full. The DMA controller can be programmed to read the FIFO and copy the results to system memory.
3. *Using external I/O pin PFI(2)/CONVERT*: In this mode each rising edge of the external CONVERT signal is used to trigger a conversion. If CONVERT is driven by a clock signal then the mode of operation is the same as that defined in (2).

A typical host interface flow could be defined as follows:

1. Reset the A/D FIFO’ s as follows: Set bit 0 of the A/D control register to ‘1’. Reset bit 0 of the A/D control register to a ‘0’ (the enable to disable condition triggers the FIFO reset). Wait for at least 10us for the FIFO’ s to clear.
2. Write “N” commands to the Channel List FIFO.
3. Set the A/D sample clock to use counter 2 as the source.

4. Setup counter 2 to generate a 200 KHz sample clock frequency.
5. Set the trigger option to “Always Triggered”.
6. Enable the A/D
7. Poll (can be set to interrupt) Channel List Done STAT flag until set.
8. Programme the DMA to empty “N” conversion results. Wait for the DMA to complete.
9. Loop to 7 OR stop the sampling by disabling the A/D.

2.6.1 Programming Information

The A/D channel is programmed using the registers as per Table 11.

The status of the FIFO control signals can be read via the GLOBAL_STATUS_REG.

REGISTER NAME	DESCRIPTION
A/D_CNTRL	**ADC CONTROL REGISTER**
BIT Definitions	
< 0 >	*A/D Operation Mode* : Write Only. < 0 > : Disables the A/D . This is the reset state. < 1 > : Enables Channel List Scan Mode Note: Moving this bit from a ‘1’ to a ‘0’ automatically resets the Data and Channel List FIFO’ s. The FIFO reset takes at least 10 us, so no FIFO commands should be issued before this period after a reset condition.
< 2:1 >	*A/D Scan Clock Source* : Sets the source for the scan clock. Each clock results in a data conversion cycle with a subsequent Channel List Pointer update. Note that the source is gated with the trigger event. < 0:0 > : A/D_CNTRL(10) is the source. < 0:1 > : Counter Timer 3 is the source. < 1:0 > : I/O pin PFI(2)/CONVERT is the source. < 1:1 > : Reserved
< 4:3 >	*A/D Trigger Source* : Sets the source for triggering the scanning. A trigger event is required before the selected scan source clock initiates conversion cycles. Once triggered the process loops until stopped. < 0:0 > : The trigger is always enabled. This is the reset state. < 0:1 > : The Analogue Compare Reference trigger is used. < 1:0 > : I/O pin TRIG1 is used. < 1:1 > : Reserved

REGISTER NAME	DESCRIPTION
< 7:5 >	*Analogue Trigger Mode* : Selects the mode in which the analogue trigger operates: Applicable only when AD_CNTRL < 4:3 > = < 0:1 > Write Only. < 000 > : LEVEL. Selected Ach(x) is > DAC(1) Reference. < 001 > : LEVEL. Selected Ach(x) is < DAC(1) Reference. < 010 > : EDGE. Selected Ach(x) rises above DAC(1) Reference. < 011 > : EDGE. Selected Ach(x) falls below DAC(1) Reference.
< 9:8 >	*A/D Channel FIFO Depth* : Write Only. FIFO Depth is always 4096.
< 10 >	*Initiate Sample* : Writing a '1' to this bit will initiate a single A/D conversion cycle if the trigger condition is satisfied. This is a "soft" bit and does not need to be cleared. Write Only.
< 11 >	*Enable FIFO_EMPTY Interrupts* : Write Only < 0 > : This interrupt is disabled. This is the reset value < 1 > : This interrupt is enabled.
< 12 >	*Enable End of Channel List Command Interrupts* : Write Only < 0 > : This interrupt is disabled. This is the reset value < 1 > : This interrupt is enabled.
A/D_FIFO	**ADC FIFO DATA REGISTER**
BIT Definitions	
< 13:0 >	Read Only: Reading this register (FIFO) passes the fourteen bits of A/D data from the results FIFO to the host. For BI-POLAR conversions the ideal A/D results are coded as follows: **CODE A/D Input Voltage** 01111111111111 = 2.5V – 1 LSB 00000000000000 = 0 10000000000000 = - (2.5V – 1 LSB). For UNI-POLAR conversions the ideal A/D results are coded as follows: **CODE A/D Input Voltage** 01111111111111 = 5V – 1 LSB 00000000000000 = 2,5V 10000000000000 = 0 + 1 LSB

Table 11 A/D Programming Register Definitions

2.7 D/A Channels

Each D/A channel has the following operational modes.

2.7.1 Disabled

The D/A channel is disabled.

The I/O controller, FIFO and FIFO status flags reset.

2.7.2 Programmed I/O Mode.

Under programmed I/O mode the host processor updates the D/A value by simply writing to the appropriate D/A Data Channel FIFO. Although the data is written to the FIFO it is moved to the D/A channel immediately without waiting for any timer ticks.

In this mode the associated timer can be used as a general purpose interrupt timer.

2.7.3 FIFO Non Loop Mode.

The FIFO and associated timer combine to update the D/A channel synchronously with the assertion of the Q output from the timer. If the FIFO is empty the scheduled D/A update cycle is ignored.

A programmable interrupt can be generated on either (or both) of the FIFO status signals: FIFO_EMPTY or FIFO_HALF_FULL..

The suggested host interface flow for this mode of operation is:

1. Reset the D/A FIFO' s as follows: Set bit 0 of the DAC control register to '1'. Reset bit 0 of the A/D control register to a '0' (the enable to disable condition triggers the FIFO reset). Wait for at least 10us for the FIFO' s to clear.

2. Disable the associated counter timer.

3. Load the D/A channel FIFO with at least 2048 + N words, where N *(Timer period) equals the latency of the host to respond to the half empty interrupt and reload more data. The DMA controller can be used.

4. Enable the D/A channel.

5. Enable a D/A channel interrupt on FIFO_HALF_FULL.

6. Setup and enable the associated counter timer.

7. On receipt of the D/A FIFO_HALF_FULL interrupt use the DMA controller to re-load new data to the FIFO.

Having a 4096 deep FIFO with an half full interrupt does not present a problem where the user's data set is smaller. Assuming the user wishes to initially pre-load 32 D/A samples and be interrupted after each 16 samples have been processed with a subsequent refill of 16 new samples. The flow would be as follows:

1. Reset the DAC FIFO' s as follows: Set bit 0 of the DAC control register to '1'. Reset bit 0 of the A/D control register to a '0' (the enable to disable condition triggers the FIFO reset). Wait for at least 10us for the FIFO' s to clear.

2. Disable the associated counter timer.

3. Load the D/A channel FIFO with 2048 "power-up" or default output conditions.

4. Load the D/A channel FIFO with the initial 32 D/A words.

5. Enable the D/A channel and the D/A channel interrupt on FIFO_HALF_FULL.

6. Setup and enable the associated counter timer.

7. On receipt of the D/A FIFO_HALF_FULL interrupt use the DMA controller to re-load 16 new D/A words to the FIFO.

8. Loop to 7.

2.7.4 FIFO Pattern Mode.

The FIFO and associated timer combine to update the D/A channel synchronously with the assertion of the Q output from the timer. If the FIFO is empty the FIFO pointer is reset and the data loop repeated.

Although a programmable interrupt can be generated on either (or both) of the FIFO status signals: FIFO_ EMPTY or FIFO_HALF_FULL. These are meaningless in this operational mode.

The suggested host interface flow for this mode of operation is:

1. Reset the DAC FIFO' s as follows: Set bit 0 of the DAC control register to '1'. Reset bit 0 of the A/D control register to a '0' (the enable to disable condition triggers the FIFO reset). Wait for at least 10us for the FIFO' s to clear.

2. Disable the associated counter timer.

3. Load the D/A channel FIFO with as many words as required by the pattern loop. The only restriction on the word count is that the maximum FIFO depth not be exceeded. The DMA controller can be used.

4. Enable the D/A channel.

5. Setup and enable the associated counter timer.

6. Stop the loop by disabling the D/A Channel or Timer.

2.7.5 Programming Information

Each D/A channel is programmed using the registers as per Table 12 .

The status of the FIFO control signals can be read via the GLOBAL_STATUS_REG.

REGISTER NAME	DESCRIPTION
DAC_CNTRL(X)	**DAC CONTROL REGISTER**
BIT Definitions	
< 1:0 >	*D/A Operation Mode* : Write Only. < 0:0 > : Disables the D/A . This is the reset state. < 0:1 > : Enables Programmed I/O mode. < 1:0 > : Enables FIFO Non Loop Mode < 1:1 > : Enables FIFO Pattern Mode Note: Moving the DAC from an enable to a disable state automatically resets the FIFO' s. This reset takes at least 10us after which the FIFO' s can be written.
< 3:2 >	*D/A Channel FIFO Depth* : Write Only. FIFO Depth is 4096 D/A words.
< 4 >	*Enable FIFO_EMPTY Interrupts* : Write Only < 0 > : This interrupt is disabled. This is the reset value < 1 > : This interrupt is enabled.
< 5 >	*Enable FIFO_HALF_FULL Interrupts* : Write Only < 0 > : This interrupt is disabled. This is the reset value < 1 > : This interrupt is enabled.
DAC_FIFO(X)	**DAC FIFO DATA REGISTER**
BIT Definitions	

REGISTER NAME	DESCRIPTION
< 13:0 >	Write Only: Writing to this register writes the fourteen bits of DAC data into the DAC Channel FIFO. The DAC is configured as a bipolar 14 bit ± 10V unit. The output bit definition is thus: **DAC CODE OUTPUT (V)** 11111111111111 = 10*(8191 / 8192) 10000000000001 = 10*(1 / 8192) 10000000000000 = 0 01111111111111 = -10*(1 / 8192) 00000000000000 = -10*(8191 / 8192)

Table 12 DAC Programming Register Definitions

2.8 Counter Timers

The PCI-703 module contains three identical sixteen bit programmable down counters. These three resources are generally allocated (this is programmable) as follows:

CNTR_0: Always mapped to the I/O signal GPCTR0_OUT. Can also be selected to control D/A Channel 0 timing.

CNTR_1: Always mapped to the I/O signal GPCTR1_OUT. Can also be selected to control D/A Channel 1 timing.

CNTR_2: A/D Channel timing.

Timers 0 and 1 can use one of three source clocks namely: Internal FPGA sources of 20 Mhz or 100 KHz or an externally generated TTL source – not exceeding 100 KHz. NOTE: The counters are always actually clocked at 20 Mhz with the rising edge of the selected clock source synchronised to the 20 MHz and used as a clock enable signal. Timer 3 can only use the internal FPGA sources of 20 Mhz or 100 KHz.

Each timer has a single Q output that has two programmable modes of operation. These are: 50% duty cycle mode or a single high level pulse equal in pulse width to the period of the selected clock source. The Q output changes state whenever the counter is about to wrap from a count of one back to it's pre-load count.

In single pulse mode the counter divides (Q) the input clock frequency by it's divider setting. In 50% mode the output pulse mode frequency is halved.

Any timer's Q output can also be configured to generate a PCI interrupt.

2.8.1 Programming Information

Each counter is programmed using the registers as per Table 13.

REGISTER NAME	DESCRIPTION
CNTR_CNTRL(X)	**COUNTER CONTROL REGISTER**
BIT Definitions	
< 0 >	Global counter enable bit. When set the counter is enabled to count down. Set to zero on reset . Write Only.
< 2:1 >	*Counter Source Clock Select* : Write Only. < 0:0 > : The source clock is 20Mhz. This is the reset value < 0:1 > : The source clock is 100 KHz < 1:X > : The source clock is the external clock. For Counters 0 & 1 only
< 3 >	*Q Output Mode* : Write Only < 0 > : The Q output is single pulse only. This is the reset value < 1 > : The Q output toggles at each terminal count.
<4 >	*Interrupt Mode* : Write Only < 0 > : Terminal counts do not set the Cntr(x)_INTR flag. This is the reset value < 1 > : Terminal counts set the Cntr(x)_INTR flag.

REGISTER NAME	DESCRIPTION
CNTR_DIVR(X)	**COUNTER DIVISOR REGISTER**
BIT Definitions	
< 15:0 >	Counter Divider: Write Only. This register contains the value that is automatically re-loaded into the down counter whenever the count is about to wrap below one. Whenever this register is written the actual counter is automatically also loaded with this value and the Q output set to zero. The counter will re-load continuously unless it is disabled. Assuming the counter is loaded with 10, the count sequence is: 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 10, 9……

Table 13 Counter Programming Registers

For predictable Q output behaviour the correct programming sequence should be: Disable the counter (the counter will stop, Q will freeze in it's current state. Programme a new divisor (Q will be reset to 0). Enable the counter.

2.9 FREQ OUT TIMER (PRESCALER)

The PCI-703 module contains a single eight bit programmable down counter used as a frequency divider. The terminal count of this divider toggles an output flip-flop which is always mapped to the I/O signal FREQ_OUT. The input clock to the divider can be selected from either a 20Mhz or 100 KHz clock source.

The divider is programmed using the registers as per Table 14.

REGISTER NAME	DESCRIPTION
FREQ_CNTRL	**FREQ_OUT CONTROL REGISTER**
BIT Definitions	
< 0 >	0: Source clock is 20Mhz. 1: Source clock is 100 Khz Bit is write only.
< 4:1 >	Counter Divider: Write Only. This register contains the value that is automatically re-loaded into the down counter whenever the count is about to wrap below one. A value of 0 will stop the counter. Whenever this register is written the actual counter is automatically also loaded with this value and the Q output set to zero. The counter will re-load continuously unless it is disabled. Assuming the counter is loaded with 10, the count sequence is: 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0, 10, 9……

Based on the above count sequence the formula for the output frequency is:

Frequency out = (Source frequency) / (2(n+1))

Table 14 FREQ_OUT Timer Programming Registers

2.10 EEPROM and Calibration DAC

An eight channel eight bit serial DAC is used to calibrate out various offset errors as well as set the two output DAC' s –10V reference level. The calibration DAC can generate a bi-polar output of ± 2.5V. As this DAC is used to calibrate errors for fourteen bit A/D's and DAC's the output is generally resistively scaled down by a factor of 47 with each bit thus representing an offset adjustment of 415 uV.

The eight offset constants are derived during the modules calibration process and are stored in a calibration EEPROM. During module initialisation these constants need to be copied from the EEPROM to the calibration DAC. This same EEPROM also contains the 14 bit gain calibration constants that are copied to the Gain Calibration Buffer.

The EEPROM is organised as 64 x 16 bit words with the organisational structure defined in Table 15 .

EEPROM Offset and	**Stored Constant**	**Mapped to Calibration DAC Channel No:**	**Nominal Value (ideal)**	**Valid Bit Range**
0	PGA_OFFSET	0	128	8 bits (7:0).
1	GAIN_DAC_OFFSET	1	128	8 bits (7:0).
2	DAC(1)_REFERENCE	2	128	8 bits (7:0).
3	DAC(1)_OFFSET	3	128	8 bits (7:0).
4	DAC(0)_REFERENCE	4	128	8 bits (7:0).
5	DAC(0)_OFFSET	5	128	8 bits (7:0).
6	BIPOLAR_OFFSET	6	128	8 bits (7:0).
7	UNIPOLAR_OFFSET	7	128	8 bits (7:0).
	Gain Calibration Constants	**Gain Buffer Offset (Mapped into 32 bit PCI space)**		
10	GAIN_0.5	0	See Table 8	14 bits (13:0).
11	GAIN_1	4	See Table 8	14 bits (13:0).
12	GAIN_2	8	See Table 8	14 bits (13:0).
13	GAIN_5	C	See Table 8	14 bits (13:0).
14	GAIN_10	10	See Table 8	14 bits (13:0).

15	GAIN_20	14	See Table 8	14 bits (13:0).
16	GAIN_50	18	See Table 8	14 bits (13:0).
17	GAIN_100	1C	See Table 8	14 bits (13:0).

Table 15 EEPROM Address Map

Both the EEPROM and Calibration DAC are accessed as serial devices. As these only need be accessed once during module initialisation this task needs to be performed by the host driver software by manipulating the serial control bits available in the GLOBAL_STATUS_REGISTER.

The EEPROM protocol and timing is defined in the data sheet for the device: MicroChip 93LC46B. To maintain compatibility with other manufacturers the suggested CLK rate should be limited to 1 Mhz.

The calibration DAC protocol is defined in the data sheet for the device : Analogue Devices AD8842.

3. Specifications

3.1 Maximum Transfer Bandwidth.

The A/D and D/A's share a data path to the FIFO interface. This limits the maximum bandwidth available to transfer data between the FIFO' s, the appropriate device and across PCI.

The maximum number of conversion cycles (14 bit A/D and D/A samples) is limited to 450 000 per second. This defines the maximum cycle clocking as per the following example:

(1) Assume A/D sampling clock set to 400 KHz.

(2) Assume D/A #1 sampling clock set to 40 KHz.

(3) Maximum D/A #2 sampling clock can be 10 KHz.

3.2 Analogue Input

Input Characteristics

Input signal ranges

Channel Gain (Programmable per channel)	**Bipolar Range** **(**Programmable per channel)	**Unipolar Range** (Programmable per channel)
0.25	± 10 V	-
.5	± 5 V	0 to 10 V
1	± 2.5 V	0 to 5 V
2.5	± 1 V	0 to 3 V
5	± 500 mV	0 to 1 V
10	± 250 mV	0 to 500 mV
25	± 100 mV	0 to 200 mV
50	± 50 mV	0 to 100 mV

Input Coupling DC

Maximum working voltage ± 11 V relative to module ground.

Over voltage protection ± 25 V with power ON (relative to module ground)

	± 35 V with power OFF (relative to module ground)

FIFO Buffer Size	Maximum 4096 .
Channel List Buffer Size	Tracks FIFO buffer size. (Depends on number of entries)
Data Transfers	Triggered interrupts, DMA or programmed I/O.

Conversion Characteristics

Maximum Sampling Rate	400 KS/s. (For single channel)
Resolution	Fourteen bits
Relative Accuracy	± 1 LSB maximum
Offset error (Gain = 1)	± 0.4 mV max.
Offset error (Gain = 10)	± 0.6 mV max.
Offset error (Gain = 50)	± 0.1 mV max.
Gain error	± 0.02 % of reading max.

Input Amplifier Characteristics

Input Impedance	10 G ohm in parallel with 100 pF minimum.
Bandwidth	600 KHz
Settling Time	5 µS to 2 LSB.
(All gains, full scale step)	15 µS to 1 LSB.

System Noise	For gains less than 5:	0.6 LSB (rms)
	For gains greater than 5:	0.8 LSB (rms)

3.3 Analogue Output

Output Characteristics

Resolution	14 Bits
Maximum update rate	400 KHz to 0.02% full scale.
FIFO Buffer Size	4096 and Pattern Length. (Programmable)
Data Transfer:	Triggered interrupts, DMA or programmed IO

Conversion Characteristics (Calibrated)

Resolution	Fourteen bits
Relative Accuracy	± 1.0 LSB maximum
FULL Scale error	± 0. 9 LSB.
Zero Scale error	± 0.9 LSB.

Voltage Output Characteristics

Range	± 10 V
Output Settling Time	2.5 us to 0.02% Full scale.
Output Coupling DC	
Output Impedance	0.2 Ohm
Output Drive	± 5 mA
Power-on state	0 V

3.4 Digital I/O

Number of channels: Eight. Programmed as Input or Output per channel.

Compatibility: TTL

I/O Characteristics

Level	Min	Max
Input low voltage	0 V	0.8 V
Input high voltage	2.0 V	5.25 V
Low level input current		-100 uA
High level input current		100 uA
Output high voltage	2.40 V	
Output low voltage		0.6 V
Low level output current		-24 mA
High level output current		4 mA

3.5 Multifunction I/O (PFI)

Number of channels: Ten. Programmed as Input or Output per channel.

Can also be mapped to functions e.g. Timer output.

I/O Characteristics: As per Digital I/O

3.6 Timing I/O

Number of channels: Two : Up/Down counters/ timers

Resolution: Twenty four bits.

Base Clocks Available: 20 MHz, 100 KHz or External.

I/O Characteristics:	As per Digital I/O.

3.7 Triggers

Analogue Trigger

Trigger Source	Any of the analogue input channels.
Reference Source	Analogue Output Channel 1.
Resolution	12 bits
Level	± 10 V
Trigger Mode	Level or Edge. Greater than reference or less than reference.

Digital Trigger

Trigger Source	One PFI inputs.
Compatibility	TTL
Pulse Width	100 ns.
Trigger Mode	Level or Edge (rising or falling) .

3.8 Bus Interface

Type	PCI 2.2. Master & Slave
	3.3V or 5.0 V compatible with auto-detect.

3.9 Power requirements

+5V (±5 %)	1.3 A

4. I/O Connector Pin Outs

Pin Number	Pin Name	Description
1	FREQ_OUT	Frequency output from the pre-scaler timer
2	GPCTR0_OUT	Output from Counter 0.
3	PFI (9)	Programmable function I/O bit 9
4	DGND	Module digital ground
5	PFI (6)	Programmable function I/O bit 6
6	PFI (5)	Programmable function I/O bit 5
7	DGND	Module digital ground
8	+5V_Fused	Fused 300 mA +5V source (common with other +5V_Fused)
9	DGND	Module digital ground
10	PFI (11)	Programmable function I/O bit 10
11	PFI (10) / TRIG1	Programmable function I/O bit 10 or A/D Digital trigger
12	DGND	Module digital ground
13	DGND	Module digital ground
14	+5V_Fused	Fused 300 mA +5V source (common with other +5V_Fused)
15	DGND	Module digital ground
16	DIO (6)	Programmable Digital I/O bit 6.
17	DIO (1)	Programmable Digital I/O bit 1.
18	DGND	Module digital ground
19	DIO (4)	Programmable Digital I/O bit 4.
20	Reserved	
21	DAC1 OUT	± 10V Output from DAC # 1
22	DAC0 OUT	± 10V Output from DAC # 0
23	ACH (15)	Analogue input for channel number 15
24	AGND	Module analogue ground signal.
25	ACH (6)	Analogue input for channel number 6
26	ACH (13)	Analogue input for channel number 13
27	AGND	Module analogue ground signal.

Pin Number	Pin Name	Description
28	ACH (4)	Analogue input for channel number 4
29	AGND	Module analogue ground signal.
30	ACH (3)	Analogue input for channel number 3
31	ACH (10)	Analogue input for channel number 10
32	AGND	Module analogue ground signal.
33	ACH (1)	Analogue input for channel number 1
34	ACH (8)	Analogue input for channel number 8
35	DGND	Module digital ground
36	DGND	Module digital ground
37	PFI (5)	Programmable function I/O bit 5
38	PFI (7)	Programmable function I/O bit 7
39	DGND	Module digital ground
40	GPCTR1_OUT	Output from Counter Timer #1
41	PFI (4)	Programmable function I/O bit 4
42	PFI (3) / CPCTR1_S	PFI I/O bit 3 or source clock for counter timer #1
43	PFI (2) / CONVERT	PFI I/O bit 2 or A/D CONVERT. Rising edge of A/D Convert (INPUT) can be used to time the conversion cycles.
44	DGND	Module digital ground
45	RESERVED	DO NOT CONNECT
46	SCANCLK	100 Ns Pulse each time the Analogue selection multiplexer is about to be changed. Can be used externally to sub-multiplex a channel.
47	DIO (3)	Programmable Digital I/O bit 3.
48	DIO (7)	Programmable Digital I/O bit 7.
49	DIO (2)	Programmable Digital I/O bit 2.
50	DGND	Module digital ground
51	DIO (5)	Programmable Digital I/O bit 5.
52	DIO (0)	Programmable Digital I/O bit 0.
53	DGND	Module digital ground
54	AGND	Module analogue ground signal.
55	AGND	Module analogue ground signal.

Pin Number	Pin Name	Description
56	AGND	Module analogue ground signal.
57	ACH (7)	Analogue input for channel number 7
58	ACH (14)	Analogue input for channel number 14
59	AGND	Module analogue ground signal.
60	ACH (5)	Analogue input for channel number 5
61	ACH (12)	Analogue input for channel number 12
62	AISENSE	INPUT : External Analogue Sense (Remote)
63	ACH (11)	Analogue input for channel number 11
64	AGND	Module analogue ground signal.
65	ACH (2)	Analogue input for channel number 2
66	ACH (9)	Analogue input for channel number 9
67	AGND	Module analogue ground signal.
68	ACH (0)	Analogue input for channel number 0

5. Installation and Configuration

5.1 Related Documentation

5.2 Software Installation

5.3 Hardware Installation