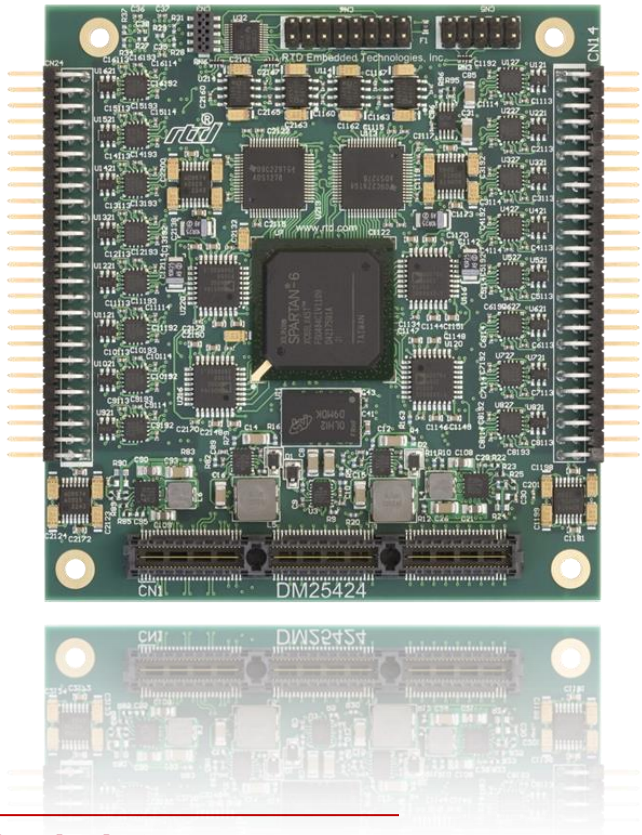
DM35424HR/DM35224HR

PCI Express Analog I/O dataModule

User's Manual

BDM-610010044 Rev. C



RTD Embedded Technologies, Inc.
AS9100 and ISO 9001 Certified

RTD Embedded Technologies, Inc.
103 Innovation Boulevard
State College, PA 16803 USA
Telephone: 814-234-8087
Fax: 814-234-5218

www.rtd.com

sales@rtd.com
techsupport@rtd.com

Revision History

Rev A	Initial Release
Rev B	Added Information about DM35224
Rev C	Updated DIO IDAN Pinout

Table of Contents

Table of Figures

Table of Tables

1 Introduction

1.1 Product Overview

The DM35424 is a software configurable simultaneous sampling data acquisition module. This module provides 16 differential analog input channels, with programmable gains. It also provides 16 individually controlled analog outputs and digital I/O.

The DM35224 is a software configurable simultaneous sampling data acquisition module. This module provides 8 differential analog input channels, with programmable gains. It also provides 8 individually controlled analog outputs and digital I/O.

This board is targeted to sensors that require high precision with a low signal level, such as accelerometers, pressure transducers, and Resistance Temperature Detectors (RTD). The DAC output can provide sufficient current and voltage for the excitation voltage of most sensors.

1.2 Board Features

- PCIe x 1 Interface
  - Universal Board can be used with a PCIe/104 Type 1 or Type 2 host
  - Dedicated DMA channel per I/O for maximum efficiency
- Analog Inputs
  - 16 Input Channels (DM35424) / 8 Input Channels (DM35224)
    - Programmable Differential or Single Ended
    - ±2.5V Differential Analog Input range
    - 0-5V Single Ended Input range
    - Programmable gains of 1, 2, 4, 8, 16, 32, 64, & 128
  - 24-bit delta sigma ADC with simultaneous sampling
  - 9.48µs max conversion time (105 KHz throughput)
  - Sampling modes and triggers are configurable independently
  - Threshold detection can generate an interrupt, or be used as a start or stop trigger
  - Configurable IIR filter on each channel
- Analog outputs
  - 16 Single ended channels (DM35424) /8 Single ended channels (DM35224)
  - 16-bit resolution
  - 10 µs full-scale settling time (±5V range)
  - ±5V output range
  - 10mA output current
- Digital I/O
  - 14 bits of Digital I/O
  - Bit programmable direction
  - 5V tolerant

1.3 Ordering Information

The DM35424 is available in the following options:

Table 1: Ordering Options

Part Number	Description
DM35424HR	PCIe/104 Analog I/O dataModule
DM35224HR	PCIe/104 Analog I/O dataModule
IDAN-DM35224HR-62S	PCIe/104 Analog I/O dataModule in IDAN enclosure with 62-pin D-Sub Connector
IDAN-DM35424HR-62S	PCIe/104 Analog I/O dataModule in IDAN enclosure with 62-pin D-Sub Connector
IDAN-DM35224HR-68S	PCIe/104 Analog I/O dataModule in IDAN enclosure with 68-pin High-Density Connector
IDAN-DM35424HR-68S	PCIe/104 Analog I/O dataModule in IDAN enclosure with 68-pin High-Density Connector

Note: Throughout this document, DM35424 refers to both DM35424 and DM35224 unless otherwise noted

The Intelligent Data Acquisition Node (IDAN™) building block can be used in just about any combination with other IDAN building blocks to create a simple but rugged 104™ stack. This module can also be incorporated in a custom-built RTD HiDAN™ or HiDANplus High Reliability Intelligent Data Acquisition Node. Contact RTD sales for more information on our high reliability systems.

1.4 Contact Information

1.4.1 Sales Support

For sales inquiries, you can contact RTD Embedded Technologies sales via the following methods:

Phone: 1-814-234-8087 Monday through Friday, 8:00am to 5:00pm (EST).
E-Mail: sales@rtd.com

1.4.2 Technical Support

If you are having problems with you system, please try the steps in the Troubleshooting section of this manual.

For help with this product, or any other product made by RTD, you can contact RTD Embedded Technologies technical support via the following methods:

Phone: 1-814-234-8087 Monday through Friday, 8:00am to 5:00pm (EST).
E-Mail: techsupport@rtd.com

2 Specifications

2.1 Operating Conditions

Table 2: Operating Conditions

Symbol	Parameter	Test Condition	Min	Max	Unit
V_{cc5}	5V Supply Voltage		4.75	5.25	V
V_{cc3}	3.3V Supply Voltage		n/a	n/a	V
V_{cc12}	12V Supply Voltage		n/a	n/a	V
T_a	Operating Temperature		-40	+85	C
T_s	Storage Temperature		-55	+125	C
RH	Relative Humidity	Non-Condensing	0	90%	%
MTBF	Mean Time Before Failure	Telcordia Issue 2 30°C, Ground benign, controlled		TBD	Hours

2.2 Electrical Characteristics

Table 3: Electrical Characteristics

Symbol	Parameter	Test Condition	Min	Typ	Max	Unit
P	Power Consumption (DM35424)	V_{cc5} = 5.0V		7.52		W
I_{cc}	5V Input Supply Current (DM35424)	Active		1504		mA
P	Power Consumption (DM35224)	V_{cc5} = 5.0V		TBD		W
I_{cc}	5V Input Supply Current (DM35224)	Active		TBD		mA
		PCIe/104 Bus				
	Differential Output Voltage		0.8		1.2	V
	DC Differential TX Impedance		80		120	Ω
	Differential Input Voltage		0.175		1.2	V
	DC Differential RX Impedance		80		120	Ω
	Electrical Idle Detect Threshold		65		175	mV
		Analog to Digital Converter				
	Linear Input Voltage	IN+ or IN -	.05		4.95	V
FSR	Full-scale Differential Input Voltage	V_{IN}=(IN+ - IN-)	-2.5 - 1LSB		+2.5	V
	Linear Differential Input Voltage	V_{IN}=(IN+ - IN-) G = PGA Gain	$\frac{-2.3}{G}$		$\frac{+2.3}{G}$	V
	Resolution				24	Bits
	Data Rate	High-Speed Mode High-Resolution Mode Low-Power Mode Low-Speed Mode			105469 52734 52734 10547	SPS
	Passive Filter -3dB Frequency				64	KHz
	Internal Filter -3dB Frequency				.49 f_{DATA}	Hz
	Internal Filter Stop Band Attenuation	High-Resolution Mode All other modes	95 100			dB
	Internal Filter Stop Band	High-Resolution Mode All other modes	0.547 f_{DATA} 0.547 f_{DATA}		127.453 f_{DATA} 63.453 f_{DATA}	Hz
	Front-End Gain Flatness	Frequency:0 – 35Khz			1	dB
	Settling Time	Complete Settling			78	Samples
	Delay	High-Resolution Mode All other modes			39 38	Samples
	ENOB(1)	High-Resolution Mode All other modes		15.2 15.4		Bits
	SNR(1)	High-Resolution Mode All other modes		93.17 94.41		dB

Table 3: Electrical Characteristics

Symbol	Parameter	Test Condition	Min	Typ	Max	Unit
	THD(1)	High-Resolution Mode All other modes		-84.74 -85.97		dB
	Noise Free Bits(2)	High-Resolution Mode All other modes		15.5 14.6		Bits
G	Gains		1		128	
Digital to Analog Converter						
	Full-scale Analog Output Voltage		-5.0000		4.9998	V
	Resolution				16	Bits
INL	Relative Accuracy		-4		+4	LSB
	Full-Scale Accuracy					LSB
	Non-Linearity					LSB
	Settling Time				10	µs
	Output Current				10	mA
	-3dB Frequency				106440	Hz
	Slew Rate				5	V/µs
Digital I/O						
V_{IL}	Input High Voltage		2		5	V
V_{IH}	Input Low Voltage		-0.5		0.8	V
V_{OL}	Output Low Voltage	I_O =-12mA	0.0		0.4	V
V_{OH}	Output High Voltage	I_O = -12mA	2.6		3.3	V
Reference Adjustment Digital Potentiometer						
	Settling Time	Change from 0x0 – 0xFF			0.5	µs
	Write to Nonvolatile				12	ms
	Read Nonvolatile				1	µs
	Write to Output				1	µs

[1] Calculated with Input = -1.2 dbFS @ 1 KHz, Max Sample Rate
[2] Calculated with Inputs grounded, Max Sample Rate

2.2.1 ANALOG INPUT FFT PLOTS

In Figure 1, the ADC was set to high speed mode and a coherent 1KHz sine wave signal was attached to input ADC0 Channel 0. The FFT absolute value was calculated using a Blackman-Hanning three term window. Three samples were taken, and the FFT data averaged.

In Figure 2, the ADC was set to high resolution mode and a coherent 1KHz sine wave signal was attached to input ADC0 Channel 0. The FFT absolute value was calculated using a Blackman-Hanning three term window. Three samples were taken, and the FFT data averaged.

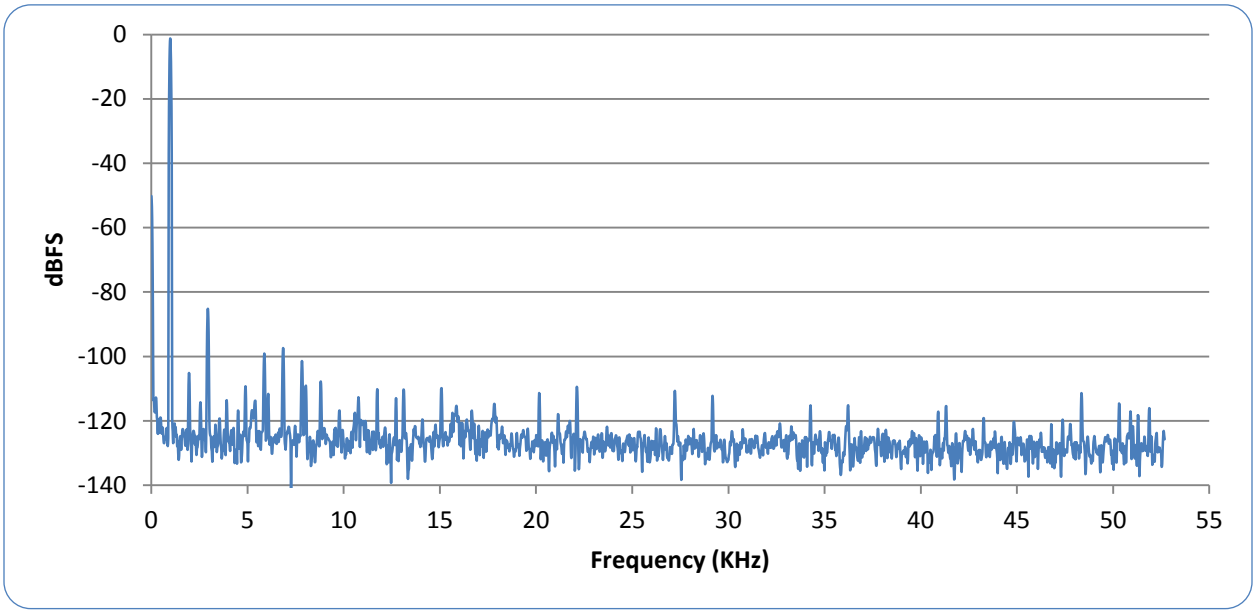


Figure 1: High Speed Mode FFT

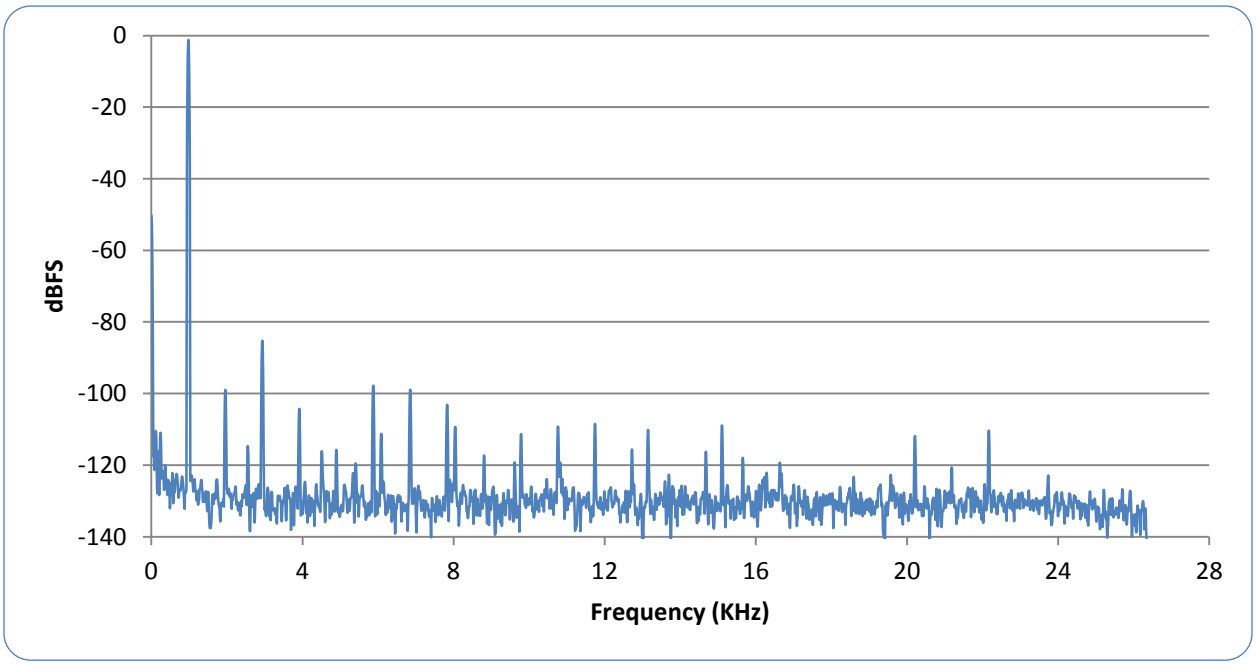


Figure 2: High Resolution Mode FFT

3 Board Connection

3.1 Board Handling Precautions

To prevent damage due to Electrostatic Discharge (ESD), keep your board in its antistatic bag until you are ready to install it into your system. When removing it from the bag, hold the board at the edges, and do not touch the components or connectors. Handle the board in an antistatic environment, and use a grounded workbench for testing and handling of your hardware.

3.2 Physical Characteristics

- Weight: Approximately 80 g (0.18 lbs.)
- Dimensions: 90.17 mm L x 95.89 mm W (3.550 in L x 3.775 in W)

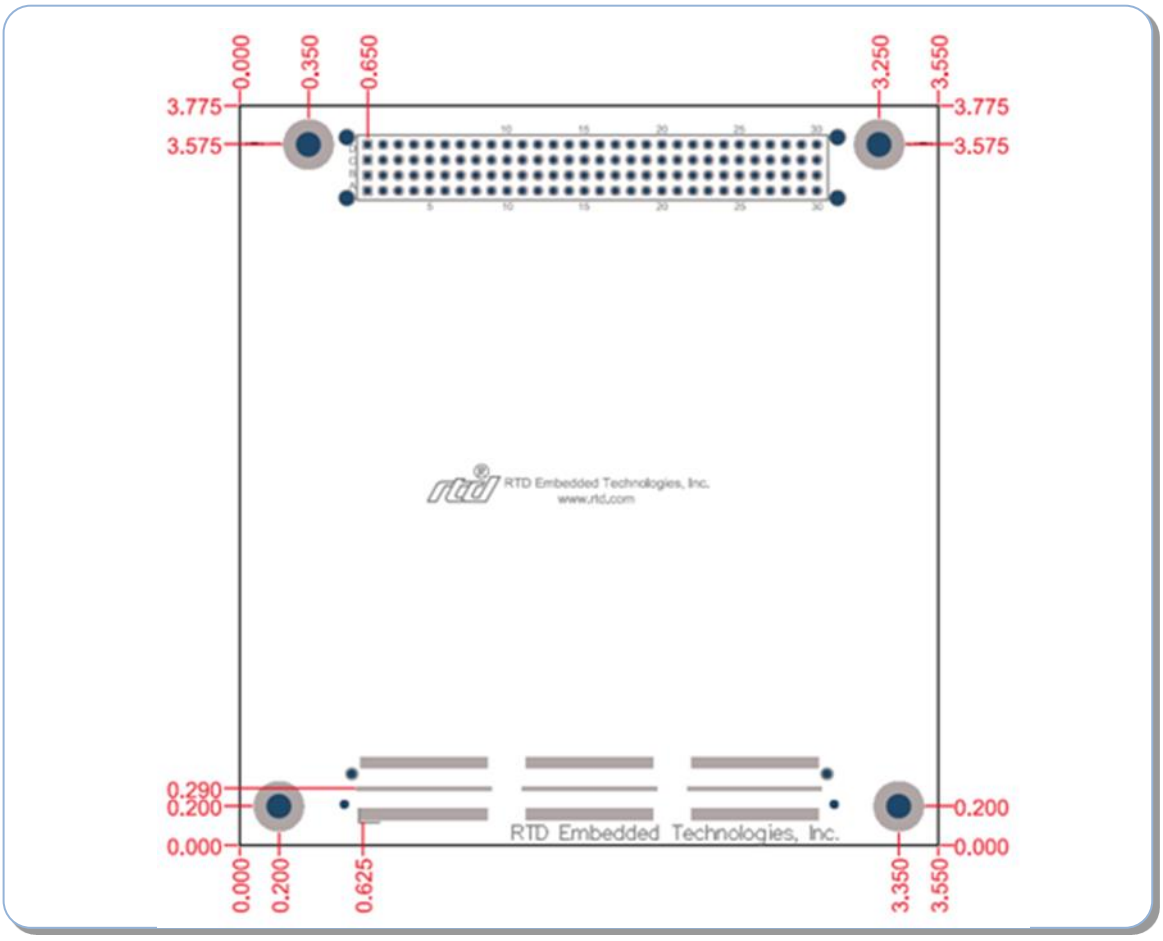


Figure 3: Board Dimensions

3.3 Connectors and Jumpers

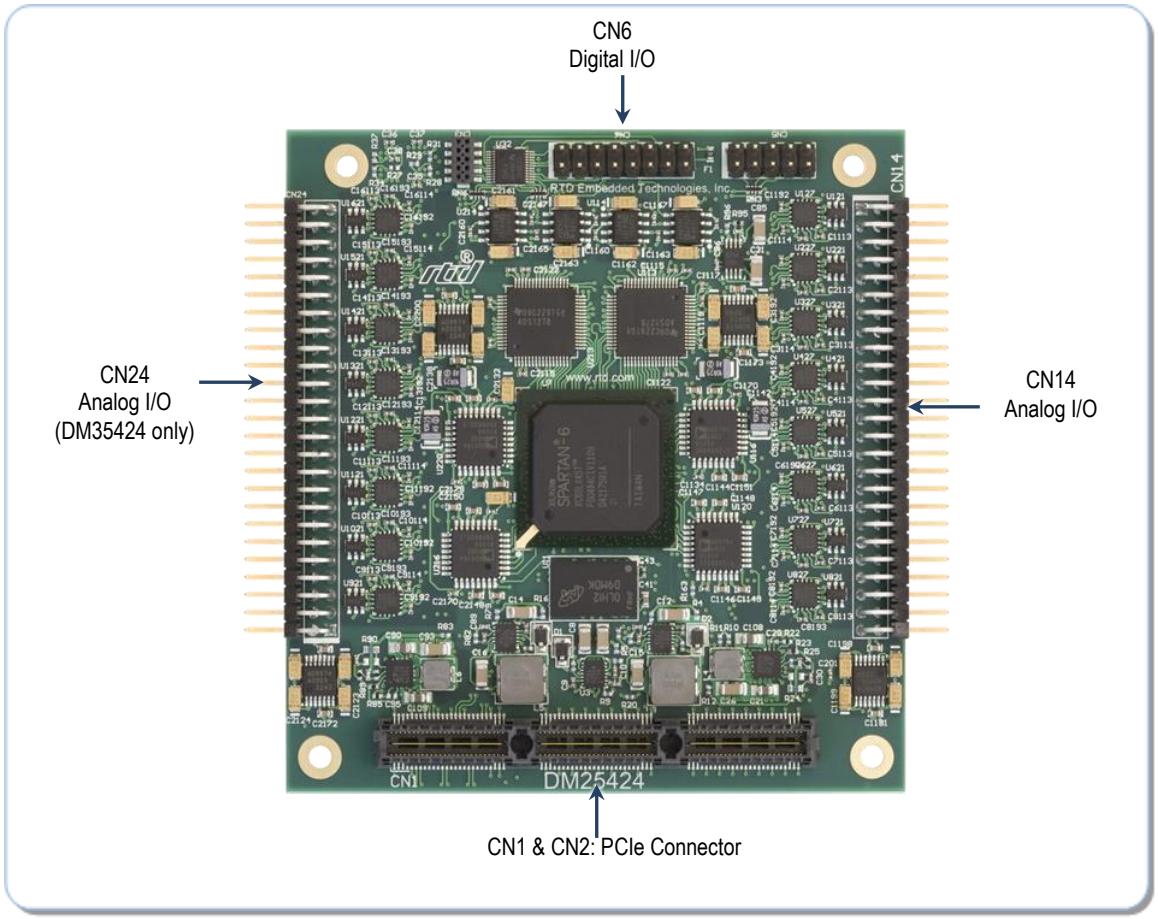


Figure 4: Board Connections

3.3.1 Bus Connectors

CN1(Top) & CN2(Bottom): PCIe Connector

The PCIe connector is the connection to the system CPU. The position and pin assignments are compliant with the *PCI/104-Express Specification*. (See PC/104 Specifications on page 52)

The DM35424 is a "Universal" board, and can connect to either a Type 1 or Type 2 PCIe/104 connector.

3.3.2 DM35424 External I/O Connectors

Digital I/O Connectors: CN6

CN6 contains one 14 bit port of digital I/O. The pin assignment is listed below.

Table 4: CN6 Digital I/O Pin Assignments

PORT0_1	2	1	PORT0_0
PORT0_3	4	3	PORT0_2
PORT0_5	6	5	PORT0_4
PORT0_7	8	7	PORT0_6
PORT0_9	10	9	PORT0_8
PORT0_11	12	11	PORT0_10
PORT0_13	14	13	PORT0_12
+5V	16	15	GND

Analog I/O Connectors: CN14 and CN24

CN14 and CN24 contain the signals for the 16 analog I/O channels. The pin assignment is listed below.

Table 5: CN14 Analog I/O Pin Assignments

AGND	2	1	AGND
AGND	4	3	DAC0 Channel 0
ADC0 Channel 0-	6	5	ADC0 Channel 0+
AGND	8	7	AGND
AGND	10	9	DAC0 Channel 1
ADC0 Channel1-	12	11	ADC0 Channel 1+
AGND	14	13	AGND
AGND	16	15	DAC0 Channel 2
ADC0 Channel 2-	18	17	ADC0 Channel 2+
AGND	20	19	AGND
AGND	22	21	DAC0 Channel 3
ADC0 Channel 3-	24	23	ADC0 Channel 3+
AGND	26	25	AGND
AGND	28	27	DAC1 Channel 0
ADC0 Channel 4-	30	29	ADC0 Channel 4+
AGND	32	31	AGND
AGND	34	33	DAC1 Channel 1
ADC0 Channel 5-	36	35	ADC0 Channel 5+
AGND	38	37	AGND
AGND	40	39	DAC1 Channel 2
ADC0 Channel 6-	42	41	ADC0 Channel 6+
AGND	44	43	AGND
AGND	46	45	DAC1 Channel 3
ADC0 Channel 7-	48	47	ADC0 Channel 7+
AGND	50	49	AGND

Table 6: CN24 Analog I/O Pin Assignments

AGND	2	1	AGND
AGND	4	3	DAC2 Channel 0
ADC1 Channel 0-	6	5	ADC1 Channel 0+
AGND	8	7	AGND
AGND	10	9	DAC2 Channel 1
ADC1 Channel1-	12	11	ADC1 Channel 1+
AGND	14	13	AGND
AGND	16	15	DAC2 Channel 2
ADC1 Channel 2-	18	17	ADC1 Channel 2+
AGND	20	19	AGND
AGND	22	21	DAC2 Channel 3
ADC1 Channel 3-	24	23	ADC1 Channel 3+
AGND	26	25	AGND
AGND	28	27	DAC3 Channel 0
ADC1 Channel 4-	30	29	ADC1 Channel 4+
AGND	32	31	AGND
AGND	34	33	DAC3 Channel 1
ADC1 Channel 5-	36	35	ADC1 Channel 5+
AGND	38	37	AGND
AGND	40	39	DAC3 Channel 2
ADC1 Channel 6-	42	41	ADC1 Channel 6+
AGND	44	43	AGND
AGND	46	45	DAC3 Channel 3
ADC1 Channel 7-	48	47	ADC1 Channel 7+
AGND	50	49	AGND

Note: CN3 is for Factory Use only

3.3.3 DM35224 External I/O Connectors

Digital I/O Connectors: CN6

CN6 contains one 14 bit port of digital I/O. The pin assignment is listed below.

Table 7: CN6 Digital I/O Pin Assignments

PORT0_1	2	1	PORT0_0
PORT0_3	4	3	PORT0_2
PORT0_5	6	5	PORT0_4
PORT0_7	8	7	PORT0_6
PORT0_9	10	9	PORT0_8
PORT0_11	12	11	PORT0_10
PORT0_13	14	13	PORT0_12
+5V	16	15	GND

Analog I/O Connectors: CN14

CN14 contain the signals for the 16 analog I/O channels. The pin assignment is listed below.

Table 8: CN14 Analog I/O Pin Assignments

AGND	2	1	AGND
AGND	4	3	DAC0 Channel 0
ADC0 Channel 0-	6	5	ADC0 Channel 0+
AGND	8	7	AGND
AGND	10	9	DAC0 Channel 1
ADC0 Channel1-	12	11	ADC0 Channel 1+
AGND	14	13	AGND
AGND	16	15	DAC0 Channel 2
ADC0 Channel 2-	18	17	ADC0 Channel 2+
AGND	20	19	AGND
AGND	22	21	DAC0 Channel 3
ADC0 Channel 3-	24	23	ADC0 Channel 3+
AGND	26	25	AGND
AGND	28	27	DAC1 Channel 0
ADC0 Channel 4-	30	29	ADC0 Channel 4+
AGND	32	31	AGND
AGND	34	33	DAC1 Channel 1
ADC0 Channel 5-	36	35	ADC0 Channel 5+
AGND	38	37	AGND
AGND	40	39	DAC1 Channel 2
ADC0 Channel 6-	42	41	ADC0 Channel 6+
AGND	44	43	AGND
AGND	46	45	DAC1 Channel 3
ADC0 Channel 7-	48	47	ADC0 Channel 7+
AGND	50	49	AGND

Note: CN3 is for Factory Use only

3.3.4 JUMPERS

There are no jumpers on the DM35424.

3.4 Steps for Installing

1. Always work at an ESD protected workstation, and wear a grounded wrist-strap.
2. Turn off power to the PC/104 system or stack.
3. Select and install stand-offs to properly position the module on the stack.
4. Remove the module from its anti-static bag.
5. Check that pins of the bus connector are properly positioned.
6. Check the stacking order; make sure all of the busses used by the peripheral cards are connected to the cpuModule.
7. Hold the module by its edges and orient it so the bus connector pins line up with the matching connector on the stack.
8. Gently and evenly press the module onto the PC/104 stack.
9. If any boards are to be stacked above this module, install them.
10. Attach any necessary cables to the PC/104 stack.
11. Re-connect the power cord and apply power to the stack.
12. Boot the system and verify that all of the hardware is working properly.

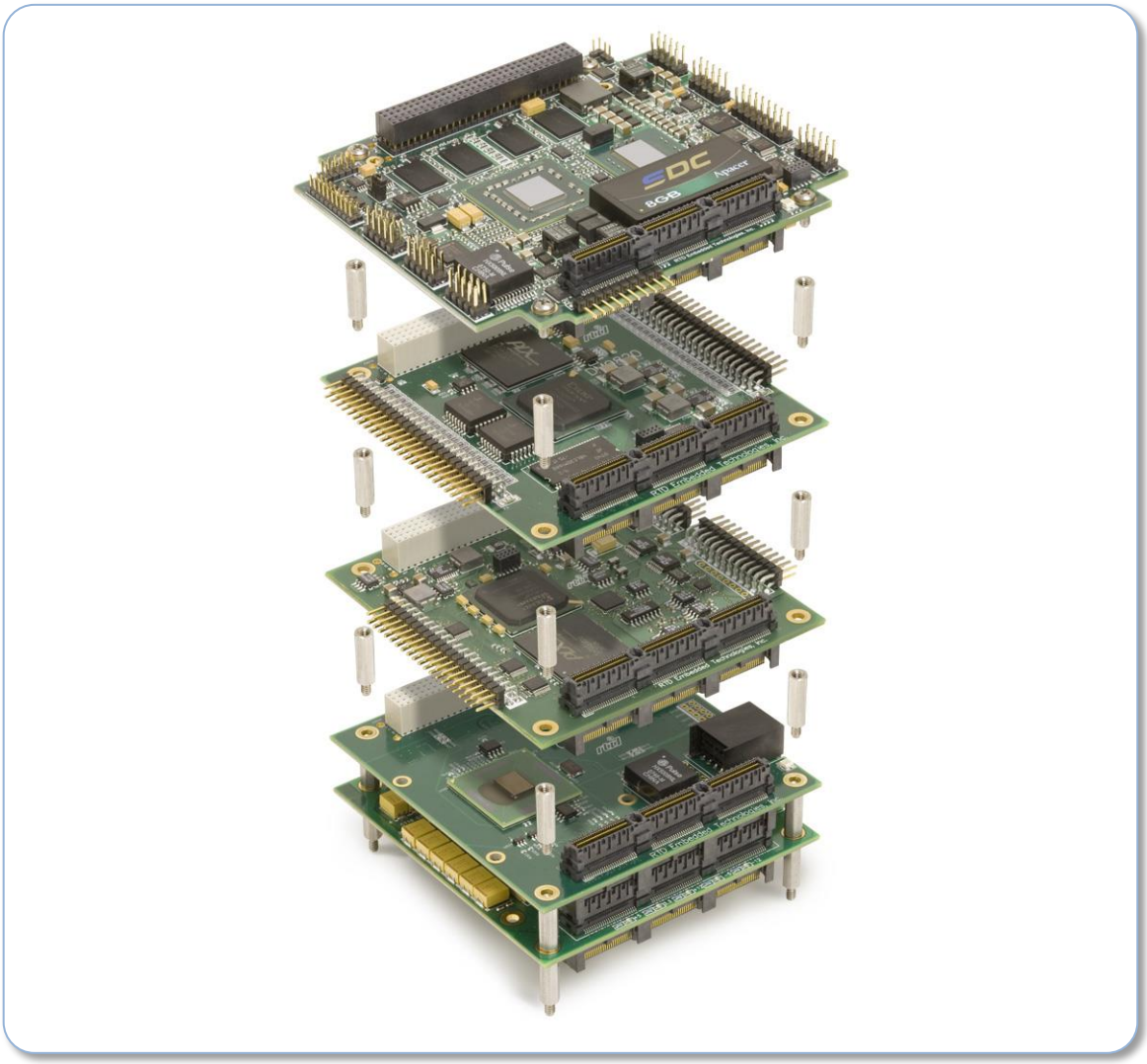


Figure 5: Example 104™ Stack

4 IDAN Connections

4.1 Module Handling Precautions

To prevent damage due to Electrostatic Discharge (ESD), keep your module in its antistatic bag until you are ready to install it into your system. When removing it from the bag, hold the module by the aluminum enclosure, and do not touch the components or connectors. Handle the module in an antistatic environment, and use a grounded workbench for testing and handling of your hardware.

4.2 Physical Characteristics

- Weight: Approximately 0.21 Kg (0.46 lbs.)
- Dimensions: 151.972 mm L x 129.978 mm W x 16.993 mm H (5.983 in L x 5.117 in W x 0.669 in H)

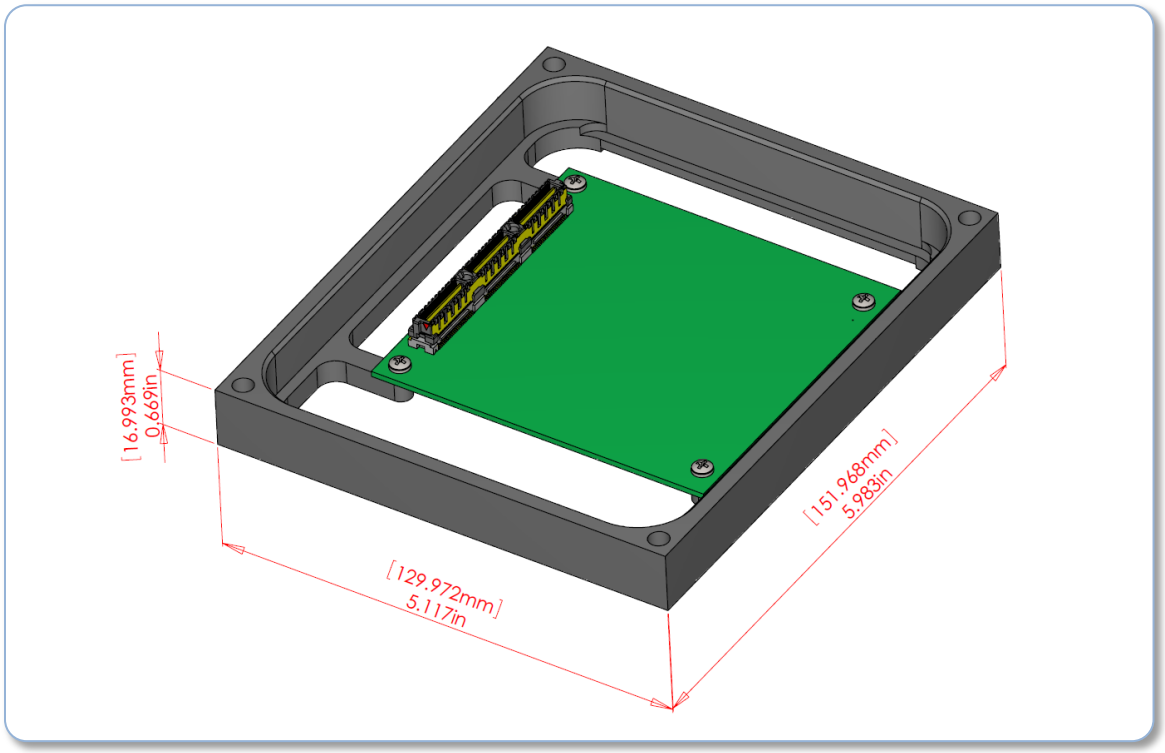


Figure 6: IDAN Dimensions

4.3 Connectors

4.3.1 DM35424 External I/O Connectors

AIO 1-8 Connector - 68-pin Subminiature "D" Female Connector

Connector Part #: Amp 749070-7 **Sample Mating Connector:** Amp 786090-7(IDC Crimp)

Table 9: IDAN- DM35424 68-Pin Subminiature "D" Connector

IDAN Pin#	Signal	DM35424 Pin #	
1	AGND	CN14	1
2	AGND	CN14	2
3	DAC0 Channel 0	CN14	3
4	AGND	CN14	4
5	ADC0 Channel 0+	CN14	5
6	ADC0 Channel 0-	CN14	6
7	AGND	CN14	7
8	AGND	CN14	8
9	DAC0 Channel 2	CN14	9
10	AGND	CN14	10
11	ADC0 Channel 1+	CN14	11
12	ADC0 Channel 1-	CN14	12
13	AGND	CN14	13
4	AGND	CN14	14
15	DAC0 Channel 2	CN14	15
16	AGND	CN14	16
17	ADC0 Channel 2+	CN14	17
18	ADC0 Channel 2-	CN14	18
19	AGND	CN14	19
20	AGND	CN14	20
21	DAC0 Channel 3	CN14	21
22	AGND	CN14	22
23	ADC0 Channel 3+	CN14	23
24	ADC0 Channel 3-	CN14	24
25	AGND	CN14	25
26	AGND	CN14	26
27	DAC1 Channel 0	CN14	27
28	AGND	CN14	28
29	ADC0 Channel 4+	CN14	29
30	ADC0 Channel 4-	CN14	30
31	AGND	CN14	31
32	AGND	CN14	32
33	DAC1 Channel 1	CN14	33
34	AGND	CN14	34

Table 9: IDAN- DM35424 68-Pin Subminiature "D" Connector

IDAN Pin#	Signal	DM35424 Pin #	
35	ADC0 Channel 5+	CN14	35
36	ADC0 Channel 5-	CN14	36
37	AGND	CN14	37
38	AGND	CN14	38
39	DAC1 Channel 2	CN14	39
40	AGND	CN14	40
41	ADC0 Channel 6+	CN14	41
42	ADC0 Channel 6-	CN14	42
43	AGND	CN14	43
44	AGND	CN14	44
45	DAC1 Channel 3	CN14	45
46	AGND	CN14	46
47	ADC0 Channel 7+	CN14	47
48	ADC0 Channel 7-	CN14	48
49	AGND	CN14	49
50	AGND	CN14	50
51	N/C		
52	N/C		
53	N/C		
54	N/C		
55	N/C		
56	N/C		
57	N/C		
58	N/C		
59	N/C		
60	N/C		
61	N/C		
62	N/C		
63	N/C		
64	N/C		
65	N/C		
66	N/C		
67	N/C		
68	N/C		

AIO 5-16 Connector - 68-pin Subminiature “D” Female Connector

Connector Part #: Amp 749070-7 **Sample Mating Connector:** Amp 786090-7(IDC Crimp)

Table 10: IDAN- DM35424 68-Pin Subminiature "D" Connector

IDAN Pin#	Signal	DM35424 Pin #	
1	AGND	CN24	1
2	AGND	CN24	2
3	DAC2 Channel 0	CN24	3
4	AGND	CN24	4
5	ADC1 Channel 0+	CN24	5
6	ADC1 Channel 0-	CN24	6
7	AGND	CN24	7
8	AGND	CN24	8
9	DAC2 Channel 2	CN24	9
10	AGND	CN24	10
11	ADC1 Channel 1+	CN24	11
12	ADC1 Channel 1-	CN24	12
13	AGND	CN24	13
4	AGND	CN24	14
15	DAC2 Channel 2	CN24	15
16	AGND	CN24	16
17	ADC1 Channel 2+	CN24	17
18	ADC1 Channel 2-	CN24	18
19	AGND	CN24	19
20	AGND	CN24	20
21	DAC2 Channel 3	CN24	21
22	AGND	CN24	22
23	ADC1 Channel 3+	CN24	23
24	ADC1 Channel 3-	CN24	24
25	AGND	CN24	25
26	AGND	CN24	26
27	DAC3 Channel 0	CN24	27
28	AGND	CN24	28
29	ADC1 Channel 4+	CN24	29
30	ADC1 Channel 4-	CN24	30
31	AGND	CN24	31
32	AGND	CN24	32
33	DAC3 Channel 1	CN24	33
34	AGND	CN24	34

Table 10: IDAN- DM35424 68-Pin Subminiature "D" Connector

IDAN Pin#	Signal	DM35424 Pin #	
35	ADC1 Channel 5+	CN24	35
36	ADC1 Channel 5-	CN24	36
37	AGND	CN24	37
38	AGND	CN24	38
39	DAC3 Channel 2	CN24	39
40	AGND	CN24	40
41	ADC1 Channel 6+	CN24	41
42	ADC1 Channel 6-	CN24	42
43	AGND	CN24	43
44	AGND	CN24	44
45	DAC3 Channel 3	CN24	45
46	AGND	CN24	46
47	ADC1 Channel 7+	CN24	47
48	ADC1 Channel 7-	CN24	48
49	AGND	CN24	49
50	AGND	CN24	50
51	N/C		
52	N/C		
53	N/C		
54	N/C		
55	N/C		
56	N/C		
57	N/C		
58	N/C		
59	N/C		
60	N/C		
61	N/C		
62	N/C		
63	N/C		
64	N/C		
65	N/C		
66	N/C		
67	N/C		
68	N/C		

AIO 1-8 Connector - 62-pin High Density "D" Female Connector

Connector Part #: VALCONN HDB-62S **Sample Mating Connector:** VALCONN HDB-62P

Table 11: IDAN- DM35424 62-Pin High Density "D" Connector

IDAN Pin#	Signal	DM35424 Pin #	
1	AGND	CN14	1
2	AGND	CN14	4
3	AGND	CN14	7
4	AGND	CN14	10
5	AGND	CN14	13
6	AGND	CN14	16
7	AGND	CN14	19
8	AGND	CN14	22
9	AGND	CN14	25
10	AGND	CN14	28
11	AGND	CN14	31
12	AGND	CN14	34
13	AGND	CN14	37
14	AGND	CN14	40
15	AGND	CN14	43
16	AGND	CN14	46
17	AGND	CN14	49
18	Reserved		
19	Reserved		
20	Reserved		
21	Reserved		
22	AGND	CN14	2
23	ADC0 Channel 0+	CN14	5
24	AGND	CN14	8
25	ADC0 Channel 1+	CN14	11
26	AGND	CN14	14
27	ADC0 Channel 2+	CN14	17
28	AGND	CN14	20
29	ADC0 Channel 3+	CN14	23
30	AGND	CN14	26
31	ADC0 Channel 4+	CN14	29

Table 11: IDAN- DM35424 62-Pin High Density "D" Connector

IDAN Pin#	Signal	DM35424 Pin #	
32	AGND	CN14	32
33	ADC0 Channel 5+	CN14	35
34	AGND	CN14	38
35	ADC0 Channel 6+	CN14	41
36	AGND	CN14	44
37	ADC0 Channel 7+	CN14	47
38	AGND	CN14	50
39	Reserved		
40	Reserved		
41	Reserved		
42	Reserved		
43	DAC0 Channel 0	CN14	3
44	ADC0 Channel 0-	CN14	6
45	DAC0 Channel 1	CN14	9
46	ADC0 Channel 1-	CN14	12
47	DAC0 Channel 2	CN14	15
48	ADC0 Channel 2-	CN14	18
49	DAC0 Channel 3	CN14	21
50	ADC0 Channel 3-	CN14	24
51	DAC1 Channel 0	CN14	27
52	ADC0 Channel 4-	CN14	30
53	DAC1 Channel 1	CN14	33
54	ADC0 Channel 5-	CN14	36
55	DAC1 Channel 2	CN14	39
56	ADC0 Channel 6-	CN14	42
57	DAC1 Channel 3	CN14	45
58	ADC0 Channel 7-	CN14	48
59	Reserved		
60	Reserved		
61	Reserved		
62	Reserved		

AIO 5-16 Connector - 62-pin High Density “D” Female Connector

Connector Part #: VALCONN HDB-62S **Sample Mating Connector:** VALCONN HDB-62P

Table 12: IDAN- DM35424 62-Pin High Density "D" Connector

IDAN Pin#	Signal	DM35424 Pin #	
1	AGND	CN24	1
2	AGND	CN24	4
3	AGND	CN24	7
4	AGND	CN24	10
5	AGND	CN24	13
6	AGND	CN24	16
7	AGND	CN24	19
8	AGND	CN24	22
9	AGND	CN24	25
10	AGND	CN24	28
11	AGND	CN24	31
12	AGND	CN24	34
13	AGND	CN24	37
14	AGND	CN24	40
15	AGND	CN24	43
16	AGND	CN24	46
17	AGND	CN24	49
18	Reserved		
19	Reserved		
20	Reserved		
21	Reserved		
22	AGND	CN24	2
23	ADC1 Channel 0+	CN24	5
24	AGND	CN24	8
25	ADC1 Channel 1+	CN24	11
26	AGND	CN24	14
27	ADC1 Channel 2+	CN24	17
28	AGND	CN24	20
29	ADC1 Channel 3+	CN24	23
30	AGND	CN24	26
31	ADC1 Channel 4+	CN24	29

Table 12: IDAN- DM35424 62-Pin High Density "D" Connector

IDAN Pin#	Signal	DM35424 Pin #	
32	AGND	CN24	32
33	ADC1 Channel 5+	CN24	35
34	AGND	CN24	38
35	ADC1 Channel 6+	CN24	41
36	AGND	CN24	44
37	ADC1 Channel 7+	CN24	47
38	AGND	CN24	50
39	Reserved		
40	Reserved		
41	Reserved		
42	Reserved		
43	DAC2 Channel 0	CN24	3
44	ADC1 Channel 0-	CN24	6
45	DAC2 Channel 1	CN24	9
46	ADC1 Channel 1-	CN24	12
47	DAC2 Channel 2	CN24	15
48	ADC1 Channel 2-	CN24	18
49	DAC2 Channel 3	CN24	21
50	ADC1 Channel 3-	CN24	24
51	DAC3 Channel 0	CN24	27
52	ADC1 Channel 4-	CN24	30
53	DAC3 Channel 1	CN24	33
54	ADC1 Channel 5-	CN24	36
55	DAC3 Channel 2	CN24	39
56	ADC1 Channel 6-	CN24	42
57	DAC3 Channel 3	CN24	45
58	ADC1 Channel 7-	CN24	48
59	Reserved		
60	Reserved		
61	Reserved		
62	Reserved		

DIO Connectors - 15-pin “D” Female Connector

Connector Part #: Amp 747052-3 **Sample Mating Connector:** Amp 1658611-3

Table 13: IDAN- DM35424 15-Pin "D" Connector

IDAN Pin#	Signal	DM35424 Pin #	
1	PORT0_0	CN6	1
2	PORT0_2	CN6	3
3	PORT0_4	CN6	5
4	PORT0_6	CN6	7
5	PORT0_8	CN6	9
6	PORT0_10	CN6	11
7	PORT0_12	CN6	13
8	GND	CN6	15
9	PORT0_1	CN6	2
10	PORT0_3	CN6	4
11	PORT0_5	CN6	6
12	PORT0_7	CN6	8
13	PORT0_9	CN6	10
14	PORT0_11	CN6	12
15	PORT0_13	CN6	14

4.3.2 DM35224 EXTERNAL I/O CONNECTORS

AIO 1-8 Connector - 68-pin Subminiature "D" Female Connector

Connector Part #: Amp 749070-7 **Sample Mating Connector:** Amp 786090-7(IDC Crimp)

Table 14: IDAN- DM35224 68-Pin Subminiature "D" Connector

IDAN Pin#	Signal	DM35224 Pin #	
1	AGND	CN14	1
2	AGND	CN14	2
3	DAC0 Channel 0	CN14	3
4	AGND	CN14	4
5	ADC0 Channel 0+	CN14	5
6	ADC0 Channel 0-	CN14	6
7	AGND	CN14	7
8	AGND	CN14	8
9	DAC0 Channel 2	CN14	9
10	AGND	CN14	10
11	ADC0 Channel 1+	CN14	11
12	ADC0 Channel 1-	CN14	12
13	AGND	CN14	13
14	AGND	CN14	14
15	DAC0 Channel 2	CN14	15
16	AGND	CN14	16
17	ADC0 Channel 2+	CN14	17
18	ADC0 Channel 2-	CN14	18
19	AGND	CN14	19
20	AGND	CN14	20
21	DAC0 Channel 3	CN14	21
22	AGND	CN14	22
23	ADC0 Channel 3+	CN14	23
24	ADC0 Channel 3-	CN14	24
25	AGND	CN14	25
26	AGND	CN14	26
27	DAC1 Channel 0	CN14	27
28	AGND	CN14	28
29	ADC0 Channel 4+	CN14	29
30	ADC0 Channel 4-	CN14	30
31	AGND	CN14	31
32	AGND	CN14	32
33	DAC1 Channel 1	CN14	33
34	AGND	CN14	34

Table 14: IDAN- DM35224 68-Pin Subminiature "D" Connector

IDAN Pin#	Signal	DM35224 Pin #	
35	ADC0 Channel 5+	CN14	35
36	ADC0 Channel 5-	CN14	36
37	AGND	CN14	37
38	AGND	CN14	38
39	DAC1 Channel 2	CN14	39
40	AGND	CN14	40
41	ADC0 Channel 6+	CN14	41
42	ADC0 Channel 6-	CN14	42
43	AGND	CN14	43
44	AGND	CN14	44
45	DAC1 Channel 3	CN14	45
46	AGND	CN14	46
47	ADC0 Channel 7+	CN14	47
48	ADC0 Channel 7-	CN14	48
49	AGND	CN14	49
50	AGND	CN14	50
51	N/C		
52	N/C		
53	N/C		
54	N/C		
55	N/C		
56	N/C		
57	N/C		
58	N/C		
59	N/C		
60	N/C		
61	N/C		
62	N/C		
63	N/C		
64	N/C		
65	N/C		
66	N/C		
67	N/C		
68	N/C		

AIO 1-8 Connector - 62-pin High Density “D” Female Connector

Connector Part #: VALCONN HDB-62S **Sample Mating Connector:** VALCONN HDB-62P

Table 15: IDAN- DM35224 62-Pin High Density "D" Connector

IDAN Pin#	Signal	DM35224 Pin #	
1	AGND	CN14	1
2	AGND	CN14	4
3	AGND	CN14	7
4	AGND	CN14	10
5	AGND	CN14	13
6	AGND	CN14	16
7	AGND	CN14	19
8	AGND	CN14	22
9	AGND	CN14	25
10	AGND	CN14	28
11	AGND	CN14	31
12	AGND	CN14	34
13	AGND	CN14	37
14	AGND	CN14	40
15	AGND	CN14	43
16	AGND	CN14	46
17	AGND	CN14	49
18	Reserved		
19	Reserved		
20	Reserved		
21	Reserved		
22	AGND	CN14	2
23	ADC0 Channel 0+	CN14	5
24	AGND	CN14	8
25	ADC0 Channel 1+	CN14	11
26	AGND	CN14	14
27	ADC0 Channel 2+	CN14	17
28	AGND	CN14	20
29	ADC0 Channel 3+	CN14	23
30	AGND	CN14	26
31	ADC0 Channel 4+	CN14	29

Table 15: IDAN- DM35224 62-Pin High Density "D" Connector

IDAN Pin#	Signal	DM35224 Pin #	
32	AGND	CN14	32
33	ADC0 Channel 5+	CN14	35
34	AGND	CN14	38
35	ADC0 Channel 6+	CN14	41
36	AGND	CN14	44
37	ADC0 Channel 7+	CN14	47
38	AGND	CN14	50
39	Reserved		
40	Reserved		
41	Reserved		
42	Reserved		
43	DAC0 Channel 0	CN14	3
44	ADC0 Channel 0-	CN14	6
45	DAC0 Channel 1	CN14	9
46	ADC0 Channel 1-	CN14	12
47	DAC0 Channel 2	CN14	15
48	ADC0 Channel 2-	CN14	18
49	DAC0 Channel 3	CN14	21
50	ADC0 Channel 3-	CN14	24
51	DAC1 Channel 0	CN14	27
52	ADC0 Channel 4-	CN14	30
53	DAC1 Channel 1	CN14	33
54	ADC0 Channel 5-	CN14	36
55	DAC1 Channel 2	CN14	39
56	ADC0 Channel 6-	CN14	42
57	DAC1 Channel 3	CN14	45
58	ADC0 Channel 7-	CN14	48
59	Reserved		
60	Reserved		
61	Reserved		
62	Reserved		

DIO Connectors - 15-pin “D” Female Connector

Connector Part #: Amp 747052-3 **Sample Mating Connector:** Amp 1658611-3

Table 16: IDAN- DM35224 15-Pin "D" Connector

IDAN Pin#	Signal	DM35224 Pin #	
1	PORT0_0	CN6	1
2	PORT0_2	CN6	3
3	PORT0_4	CN6	5
4	PORT0_6	CN6	7
5	PORT0_8	CN6	9
6	PORT0_10	CN6	11
7	PORT0_12	CN6	13
8	GND	CN6	15
9	PORT0_1	CN6	2
10	PORT0_3	CN6	4
11	PORT0_5	CN6	6
12	PORT0_7	CN6	8
13	PORT0_9	CN6	10
14	PORT0_11	CN6	12
15	PORT0_13	CN6	14

4.4 Steps for Installing

1. Always work at an ESD protected workstation, and wear a grounded wrist-strap.
2. Turn off power to the IDAN system.
3. Remove the module from its anti-static bag.
4. Check that pins of the bus connector are properly positioned.
5. Check the stacking order; make sure all of the busses used by the peripheral cards are connected to the cpuModule.
6. Hold the module by its edges and orient it so the bus connector pins line up with the matching connector on the stack.
7. Gently and evenly press the module onto the IDAN system.
8. If any boards are to be stacked above this module, install them.
9. Finish assembling the IDAN stack by installing screws of an appropriate length.
10. Attach any necessary cables to the IDAN system.
11. Re-connect the power cord and apply power to the stack.
12. Boot the system and verify that all of the hardware is working properly.

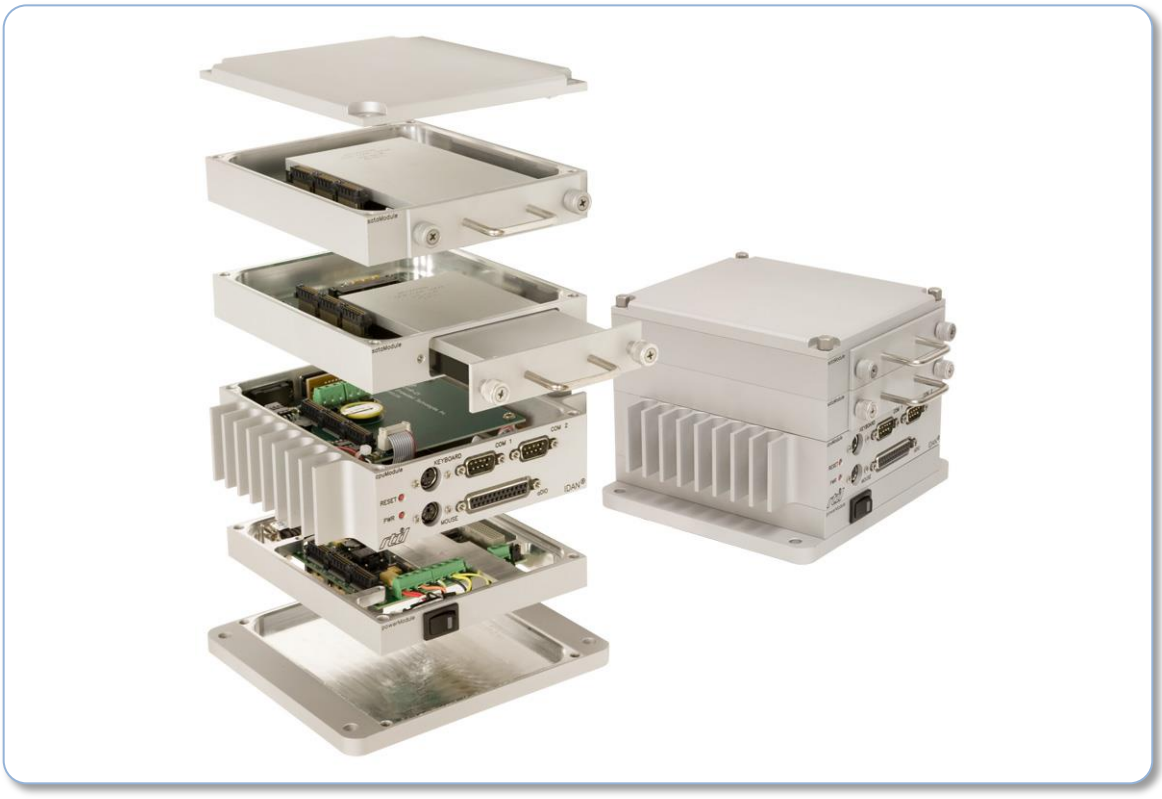


Figure 7: Example IDAN System

5 Functional Description

5.1 Block Diagram

The Figure below shows the functional block diagram of the DM35424. The various parts of the block diagram are discussed in the following sections.

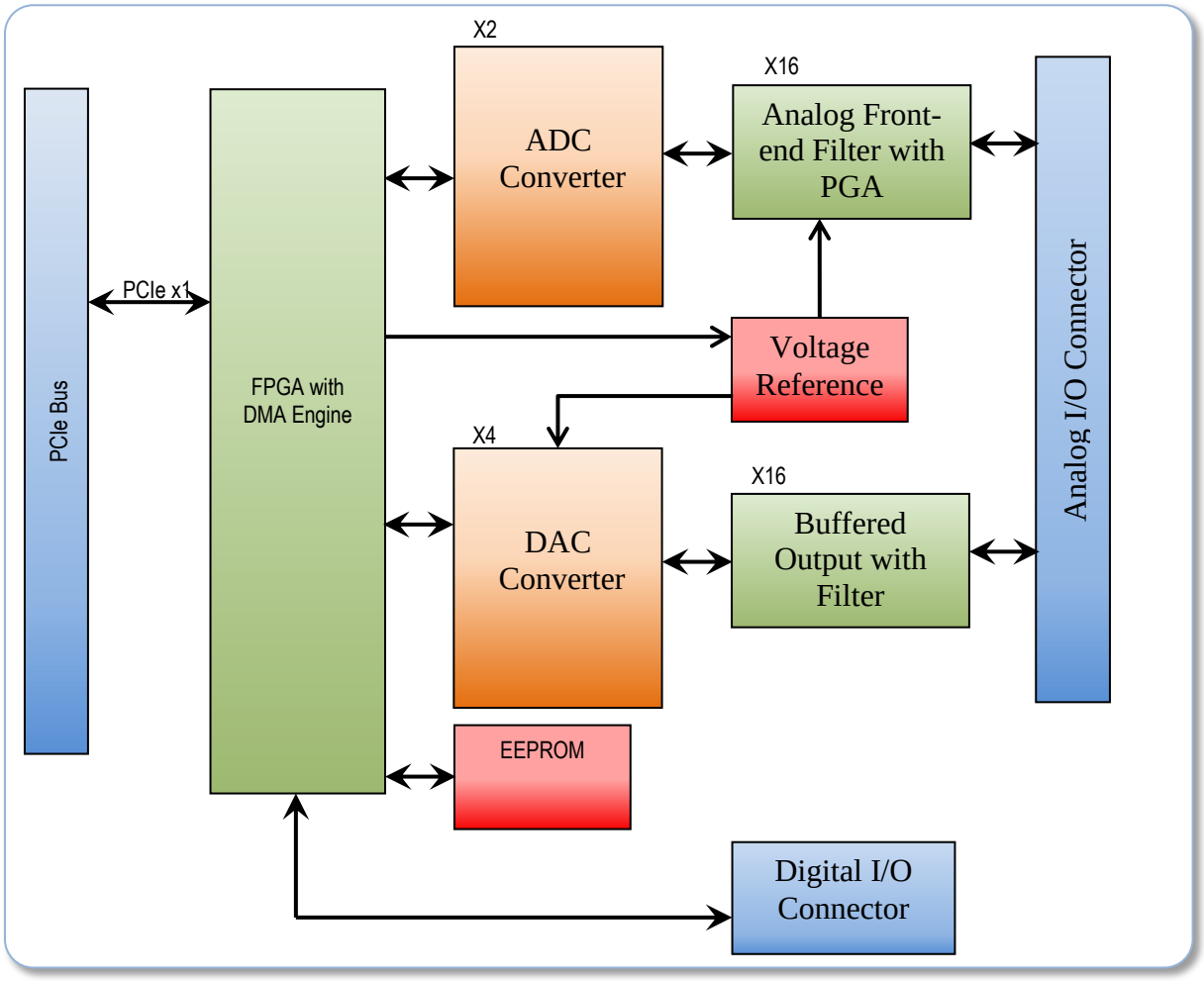


Figure 8: DM35424 Block Diagram

5.2 DMA Engine

The DM35424 features a FPGA with a built in PCI Express interface and DMA engine. The FPGA controls all communication between the bus and the control logic on the board. The FPGA also features small FIFOs for use with DMA, which is needed for continuous data transfer. Each DAC and ADC is provided with its own FIFO and DMA channel, allowing them to transfer data independent of one another.

Each DMA channel can be programmed to transfer data from FPGA to PCI bus or from the PCI bus to the FPGA. Each DMA channel also features a 64-bit PCI addressing and a 16MB maximum buffer for memory accessing.

5.3 Analog Front-End

The DM35424 analog front-end consist of two filters and a differential programmable gain amplifier, PGA. The two filters are, one RF filter with a 3dB cutoff frequency of 1MHZ and the second filter is a single-pole pass filter with a 3dB cutoff frequency of 64KHz. Refer to Figure 9 for analog front-end design.

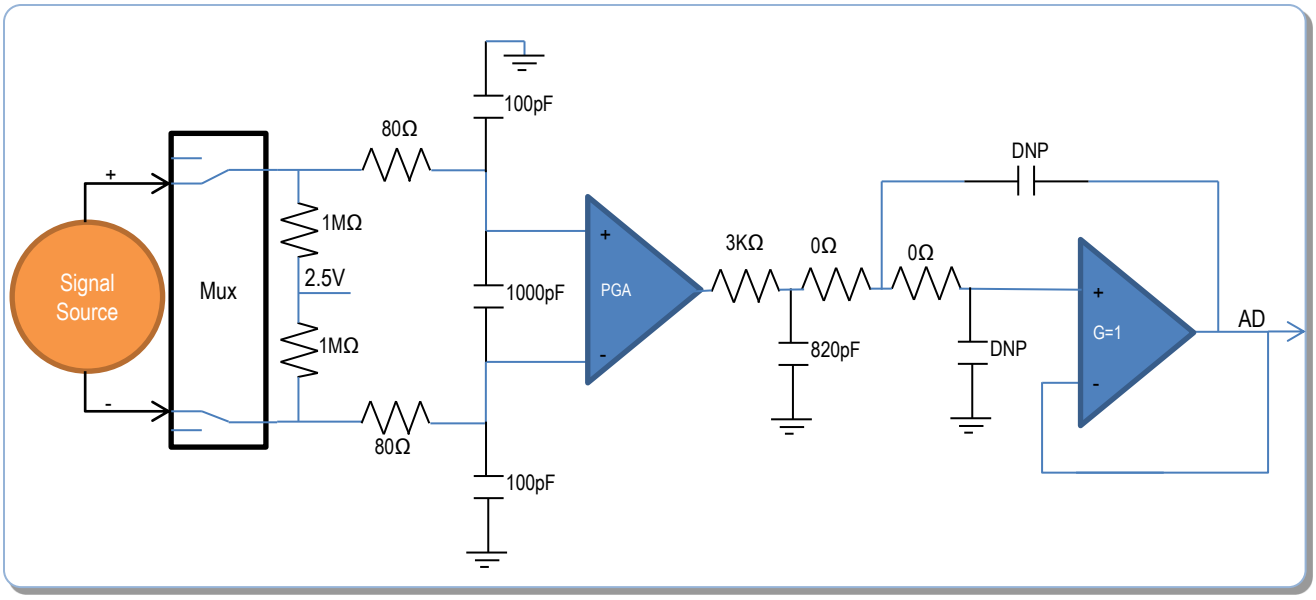


Figure 9: DM35424 Front-End

Note: For custom cutoff frequency for front-end filter contact RTD Sales.

The DM35424 also features four input modes. These modes can be selected by using the CHn_FRONT_END_CONFIG (Maskable Read/Write) register of the ADC on page 41. A description of each mode can be found below.

DAC Loopback Mode

The DAC Loopback mode is a single-ended mode which the DAC channel is connected internally to its corresponding ADC channel. Table 17 below shows the DAC channel with its corresponding ADC channel. The ADC input must stay within ±2.5V, at a gain of 1, with respect to the board's voltage reference (2.5V).

Table 17: Corresponding DAC to ADC channel

DAC Channel	ADC Channel
DAC0 Channel 0	ADC0 Channel 0
DAC0 Channel 1	ADC0 Channel 1
DAC0 Channel 2	ADC0 Channel 2
DAC0 Channel 3	ADC0 Channel 3
DAC1 Channel 0	ADC0 Channel 4
DAC1 Channel 1	ADC0 Channel 5
DAC1 Channel 2	ADC0 Channel 6
DAC1 Channel 3	ADC0 Channel 7
DAC2 Channel 0	ADC1 Channel 0
DAC2 Channel 1	ADC1 Channel 1
DAC2 Channel 2	ADC1 Channel 2
DAC2 Channel 3	ADC1 Channel 3
DAC3 Channel 0	ADC1 Channel 4
DAC3 Channel 1	ADC1 Channel 5
DAC3 Channel 2	ADC1 Channel 6
DAC3 Channel 3	ADC1 Channel 7

Single-Ended Positive Input Mode

In single-ended positive input mode, the input signal is measured in reference to the board's GND. In this mode the input signal is connected to input ADC0 Channel 1+ through ADC1 Channel 8+ and the low side to any of the GND pins available on the Analog Connector. When in single-ended positive input mode, the high side must stay within ±2.5V, at a gain of 1, with respect to the board's voltage reference (2.5V).

Single-Ended Negative Input Mode

In single-ended negative input mode, the input signal is measured in reference to the board's GND. In this mode the input signal is connected to input ADC0 Channel 1- through ADC1 Channel 8- and the low side to any of the GND pins available on the Analog Connector. When in single-ended negative input mode, the high side must stay within ±2.5V, at a gain of 1, with respect to the board's voltage reference (2.5V).

Differential Input Mode

In this mode your signal source may or may not have a separate ground reference. In differential mode, the high side input is measured in reference to the low side input. In this mode you connect the high side of the input signal to the analog input, ADC0 Channel 1+ through ADC1 Channel 8+, and connect the low side to corresponding ADC **-** pin. When in differential mode, both the high and low side must stay within ±2.5V, at a gain of 1, with respect to the board's voltage reference (2.5V). In most cases, the board ground must still be attached to the device that is generating the input signal.

5.4 Analog to Digital Converter

The DM35424 uses a 24 bit delta sigma ADC converter, which provides a very high digital resolution of the dynamic input voltage. This ADC converter has a max throughput rate of 105 KHz, allowing conversion speeds up to 9.6µs. This ADC features simultaneous sampling for all 16 channels, an internal antialiasing filter and four different operational modes allowing the user to optimize for speed, resolution and power.

The DM35424 conversions are controlled by a pacer clock and the sampling ratio set by the delta sigma ADC. (This ratio is listed in Sampling on page 41.)

Each ADC converter supports a 511 sample FIFO for DMA. Each sample is 32 bits.

5.4.1 Initialization

There are several steps to initialize the Analog to Digital converter. The initialization prepares the converter and the front-end to capture samples. Following the example programs and using the drivers provided by RTD will ensure that these steps are followed in the correct order. Initialization of the ADC is performed as follows:

1. Set the ADC to the Uninitialized state (MODE = Uninitialized)
2. Setup the DMA for the channel
3. Set the input mode (CH_FRONT_END_CONFIG)
4. Delay for at least 3us. This allows the front-end to settle.
5. Set the AD_CONFIG register.
6. Set the start and stop triggers (START_TRIG, STOP_TRIG)
7. Set the clock source (CLK_SOURCE)
8. Set the sample rate (CLK_DIV_CNTR)
9. Set the Pre and/or Post Capture counters (PRE_TRIGGER_CAPTURE, POST_STOP_CAPTURE)
10. Set the ADC to the Reset state (MODE = Reset)
11. Start the DMA
12. Start the ADC (MODE = Go)

5.4.2 Analog to Digital Converter Delay

The Delta-Sigma converter used on this board has a linear phase filter. Linear phase filters exhibit constant delay times versus frequency, or constant group delay. However, there is a long delay from the time that a sample is taken until it is captured by the FIFO. On this board, that delay is either 39 samples in High-Resolution mode, or 38 samples in all other modes. The delay can be viewed as an additional FIFO with a fixed amount of data in it. All of the data that is visible to the driver is delayed by 38 (or 39) samples.

After the ADC is initialized, it will continue to sample until the board is reset, or the ADC is placed into an uninitialized state. When capturing is started by setting the ADC Mode to "Go" and having a start trigger, the first sample captured is actually 38 (or 39) samples old. This is important to recognize when correlating captured data with other events. For example, if you are using the rising edge of an external signal as the start trigger, and you are also sampling that signal, the first 38 (or 39) samples captured will be low, followed by the rising edge. If the application requires correlation with a start and stop events, then the first 38 (or 39) samples should be discarded, and always collect 38 (or 39) more samples than what is needed. (When a Stop Trigger is used, the POST_STOP_CAPTURE register can be used to capture additional samples)

This is also true when pausing the ADC. Let's assume a sample rate of 1 Hz using high-resolution mode. If you pause the ADC at t=100s, the last sample captured into the FIFO is from t=61s. The ADC continues to sample, but the samples are discarded. If you then resume sampling at t=200s, the next sample captured into the FIFO is from t=161s.

5.4.3 DATA FORMATTING

The converted data is stored as a signed 32-bit value in ADC channel's CH_LAST_SAMPLE (Read-Only) and in its DMA channel. Below are the equations of how to convert stored conversion data to a voltage.

Differential	Single-Ended
$Voltage = \frac{2.5}{Gain \times (2^{23}-1)} \times Conversion\ Data$	$Voltage = \left[\frac{2.5}{Gain \times (2^{23}-1)} \times (Conversion\ Data - 0x7FFFFF)\right] + 2.5$

5.5 Digital to Analog Converter

The DM35424 digital to analog (DAC) circuitry features 16 independent 16-bit analog output channels with a programmable ±5V range. With the use of DMA data can be continuously written to the DAC to produce a repetitive and non-repetitive output waveform.

Each DAC converter supports a 511 sample FIFO for DMA. Each sample is 32 bits.

5.5.1 INITIALIZING THE DAC CONVERTER

The following is a list of the typical steps needed to initialize the DAC converter and begin sampling

1. Set the DAC to the Uninitialized state (MODE = Uninitialized)
2. Setup the DMA for the channel
3. Set the start and stop triggers (START_TRIG, STOP_TRIG)
4. Set the clock source (CLK_SOURCE)
5. Set the sample rate (CLK_DIV_CNTR)
6. Set the Post Capture counter (POST_STOP_CAPTURE)
7. Set the DAC to the Reset state (MODE = Reset)
8. Start the DMA
9. Start the DAC (MODE = Go)

5.6 Digital I/O

The DM35424 has 14 buffered TTL/CMOS digital I/O lines, which can be used to transfer data between the system and external devices. The 14 bits can be programmed as either inputs or outputs.

5.7 Voltage Reference

The DM35424 has an adjustable voltage reference controlled by a digital potentiometer. This allows the voltage references that are used for the ADC's and DAC's to be adjusted by ±0.5%. By adjusting the reference to a partially ADC or DAC, you can correct the gain error of the device.

6 Register Address Space

The DM35424 was built as a modular design, which allows each board function to have its own Functional Block (FB). Each functional block was design to work independent of each other. For this reason, we provide individual DMA channels, interrupts, clock, and FIFO to each functional block.

The registers are described by their PCIe Base Address Register (BAR), which is defined in the PCI configuration space for this board. The configuration space is generally handled by the operating system. For more information on how to use the configuration space, consult the *PCI Local Bus Specification, Revision 3.0* from the PCI-SIG.

Register Types

There are several different types of registers that are referred to in this section. A description of each type is below.

- Read/Write Registers: The value that is written to this register can also be read back.
- Maskable Registers: This is a 32 bit register that consists of 16-bit data field in the upper word and a 16-bit mask value in the lower word. For each bit in the data field, it is only written to the register if the corresponding bit in the mask field is '1'.
- Sticky Registers: This is a status read register. When bit in this register has a value of '1', a '1' needs written to that bit to reset the register to '0'. This is typically used for interrupt status registers.
- Read Only: This register can only be read.

NOTE: Writing to Read-Only registers may have unexpected results.

Clock and Trigger Source

Below is the list of clock and trigger sources, and the register value needed to select the source. These sources are used in the functional blocks below.

0x00: System clock/immediate

0x01: Never

0x02: CLK_GBL2

0x03: CLK_GBL3

0x04: CLK_GBL4

0x05: CLK_GBL5

0x06: CLK_GBL6

0x07: CLK_GBL7

0x08: Local Clock Source

0x09: Inverted Local Clock Source

0x0A: CLK_GBL2 Inverted

0x0B: CLK_GBL3 Inverted

0x0C: CLK_GBL4 Inverted

0x0D: CLK_GBL5 Inverted

0x0E: CLK_GBL6 Inverted

0x0F: CLK_GBL7 Inverted

6.1 BAR0 – General Board Control

The BAR0 region is a Memory Mapped register space which contains some global registers. It also contains a table describing the different Function Blocks of the board, and the offsets into BAR2 of the registers for that Function Block. For maximum flexibility, the driver must read the table in BAR0 to calculate the offset to each Function Block in BAR2.

Table 18: BAR0

Offset	0x03	0x02	0x01	0x00
0x00	GBC_BRD_RST	GBC_EOI	GBC_REV	GBC_FMT
0x04	GBC_PDP			
0x08	GBC_BUILD			
0x0C	*Reserved*		GBC_SYS_CLK_FREQ	
0x10	GBC_IRQ_STATUS			
0x14				
0x18	GBC_DIRQ_STATUS			
0x1C				
0x20	FB0_ID			
0x24	FB0_OFFSET			
0x28	FB0_OFFSET_DMA			
0x2C	Reserved			
0x30	FB1_ID			
0x34	FB1_OFFSET			
0x38	FB1_OFFSET_DMA			
0x3C	Reserved			
…				
0x20+0x10*n	FBn_ID			
0x24+0x10*n	FBn_OFFSET			
0x28+0x10*n	FBn_OFFSET_DMA			
0x2C+0x10*n	Reserved			
…				
0xA0	FB8_ID			
0xA4	FB8_OFFSET			
0xA8	FB8_OFFSET_DMA			
0xAC	Reserved			

6.1.1 GBC_BRD_RST (READ/WRITE)

This register is used to send a reset command to the board. Write 0xAA to this register to reset the board.

6.1.2 GBC_EOI (READ/CLEAR)

This register is used to acknowledge an interrupt. It is used to safeguard against missing an interrupt. At the end of the Interrupt Service Routines (ISR), write a 0x01 to this register. If there is another interrupt pending in the status registers, the interrupt line is toggled (Legacy Mode), or another interrupt is sent (MSI Mode).

6.1.3 GBC_REV (READ-ONLY)

This register contains the PDP revision for this board. A=1, B=2, etc.

6.1.4 GBC_FMT (READ-ONLY)

This register contains the format ID that is used in this board. The current value is 0x01.

6.1.5 GBC_PDP (READ-ONLY)

This register contains the PDP number for this board.

6.1.6 GBC_BUILD (READ-ONLY)

This register contains a unique 32-bit build number for every FPGA build.

6.1.7 GBC_SYS_CLK_FREQ (READ ONLY)

This register contains the measured frequency of the system clock. Units are 10 kHz, i.e. (Frequency in Hertz) = (GBC_SYS_CLK_FREQ * 10 kHz). This value is not available (will read 0) until 100us after a Board Reset, and is continually updated.

6.1.8 GBC_IRQ_STATUS (READ/CLEAR)

This is a 64-bit interrupt status register for non-DMA interrupts. Each bit in this register corresponds to one of the Function Blocks; bit 0 corresponds to FB0 (whose ID and OFFSET are at 0x020), etc. Bits 60 through 63 are reserved. This is a Sticky Register, so the driver clears it by writing a '1' to the appropriate bit.

6.1.9 GBC_DIRQ_STATUS (READ/CLEAR)

This is a 64-bit interrupt status register for DMA interrupts. Each bit in this register corresponds to one of the Function Blocks; bit 0 corresponds to FB0 (whose ID and OFFSET are at 0x020), etc. Bits 60 through 63 are reserved. This is a "sticky" register, and the driver clears it by writing a '1' to the appropriate bit.

6.1.10 FBN_ID (READ-ONLY)

This is a 32-bit value that identifies the type of Function Block in slot 'n'.

0x00011000 – ADC

0x00012000 – DAC

0x0000F000 – Voltage Reference

0x00003000 – Digital I/O

0x0000F001 – Temperature Sensor

6.1.11 FBN_OFFSET (READ-ONLY)

This is the offset from the beginning of the Functional Block section in BAR2 that this Functional Block resides in.

6.1.12 FBN_OFFSET_DMA (READ-ONLY)

This is the offset from the beginning of the Functional Block section in BAR2 that the Functional Block DMA Registers reside in.

6.2 BAR2: Functional Block Standard DMA

This section describes a standard DMA implementation is used by the Functional Blocks. There is a single DMA engine that services all of the DMA channels used by the Function Block. Each DMA channel has a block of registers associated with it to configure the DMA channel, as well as set up the descriptors for the buffers in system memory. In the sections below, "m" is used to enumerate the DMA channels, and "n" is used to enumerate the buffer descriptors within a channel.

Table 19: DMA Registers

Offset	0x03	0x02	0x01	0x00
D + 0x00	FB_DMAm_Stat_Underflow	FB_DMAm_Stat_ Overflow	FB_DMAm_Setup	FB_DMAm_Action
D + 0x04	FB_DMAm_Current_Buffer	FB_DMAm_Count		
D + 0x08	FB_DMAm_RD_FIFO_CNT		FB_DMAm_WR_FIFO_CNT	
D + 0x0C	FB_DMAm_Last_Action	FB_DMAm_Stat_ Complete	FB_DMAm_Stat_ Invalid	FB_DMAm_Stat_ Used
D + 0x10	FB_DMAm_CTRL0	FB_DMAm_STAT0	Reserved	
D + 0x14	Reserved	FB_DMAm_SIZE0		
D + 0x18	FB_DMAm_ADDRESS0			
D + 0x1C				
D + 0x20	FB_DMAm_CTRL1	FB_DMAm_STAT1	Reserved	
D + 0x24	Reserved	FB_DMAm_SIZE1		
D + 0x28	FB_DMAm_ADDRESS1			
D + 0x2C				
…				
D + 0x10 + (0x10 * n)	FB_DMAm_CTRLn	FB_DMAm_STATn	Reserved	
D + 0x14 + (0x10 * n)	Reserved	FB_DMAm_SIZEn		
D + 0x18 + (0x10 * n)	FB_DMAm_ADDRESSn			
D + 0x1C + (0x10 * n)				

6.2.1 FB_DMAm_Action (Read/Write)

This register is the overall control for this DMA channel. After writing to the Action register, the driver should poll the Last_Action register (below) until it reads the same value. This shows that the action has been performed by the DMA state machine. This is especially important when entering and exiting the Clear state.

0x00 = Clear: Clear the Current Buffer field, the internal offset counters, and the FIFO. DMA is stopped.

0x01 = Go: Starts DMA

0x02 = Pause: DMA transfers are stopped, but all internal registers maintain their state. During PAUSE you will still receive Stat_Underflow and Stat_Overflow interrupts. After PAUSE, you may transition to GO or CLEAR.

0x03 = Halt: Buffer has been filled that has the HALT bit set, attempted to use a buffer with the Valid bit cleared. After HALT, you must transition to CLEAR.

NOTE: The DMA engine also writes to this register when a buffer is completed with the HALT bit set, or it encounters an invalid buffer. When changing this register from the Go to the Clear state, be sure to read it back to make sure the DMA engine did not change it to the Halt state.

6.2.2 FB_DMAM_LAST_ACTION (READ/WRITE)

The DMA Engine writes the value of FB_DMAm_Action to this register after it has completed the action. This indicates to the user that the last command has been processed. It specifically aids the transition to the Clear state. When transitioning to Clear, the user should wait until FB_DMAm_Last_Action indicates that the Clear has been processed before initiating any other Action changes.

The user may also write a value to this register and then poll the register to see when the value changes. This method can be used to detect when the DMA engine services the channel without an Action change.

6.2.3 FB_DMAm_Setup (Read/Write)

B0: IntEna: Set to '1' to enable the DMA engine to generate interrupts on completion of a buffer.

B1: ErrIntEna: Set to '1' to enable the DMA engine to generate interrupts on error.

B2: Direction: Set to '1' to transfer from the board to the PCI bus. Clear to '0' to transfer from the PCI bus to the board. Note that although the DMA channel always supports both directions, the Function Block that the channel is associated with may only support one direction.

B3: IgnoreUsed: Set to '1' to prevent an error condition when accessing a buffer with the Used bit set. Examples are continuous output from a DAC, or very large Pre-trigger buffering using system memory.

6.2.4 FB_DMAm_Stat_Used (Read/Write)

This register is used to determine the source of a DMA interrupt. The bits are cleared by writing 0x00 to the byte. Stat_Used will be set regardless of having ErrIntEna set to '1'.

B0: Used_Desc (R/C). Set to '1' by the DMA engine if it attempting to use a descriptor with the Used bit set.

6.2.5 FB_DMAm_Stat_Invalid (Read/Write)

This register is used to determine the source of a DMA interrupt. The bits are cleared by writing 0x00 to the byte. Stat_Invalid will be set regardless of having ErrIntEna set to '1'.

B0: Invalid_Desc (R/C). Set to '1' by the DMA engine if it attempting to use a descriptor with the Valid bit cleared.

6.2.6 FB_DMAm_Stat_Overflow (Read/Write)

This register is used to determine the source of a DMA interrupt. The bits are cleared by writing 0x00 to the byte. Stat_Overflow will be set regardless of having ErrIntEna set to '1'. If an overflow occurs the DMA engine will PAUSE.

B0: Overflow (R/C). Set to '1' by the DMA engine if an overflow occurred on the FIFO.

6.2.7 FB_DMAm_Stat_Underflow (Read/Write)

This register is used to determine the source of a DMA interrupt. The bits are cleared by writing 0x00 to the byte. Stat_Underflow will be set regardless of having ErrIntEna set to '1'. If an underflow occurs the DMA engine will PAUSE.

B0: Underflow (R/C). Set to '1' by the DMA engine if an underflow occurred on the FIFO.

6.2.8 FB_DMAm_Stat_Complete (Read/Write)

This register is used to determine the source of a DMA interrupt. The bits are cleared by writing 0x00 to the byte.

B0: Buffer_Complete (R/C). Set to '1' by the DMA engine when a buffer is filled that has the Interrupt bit set.

6.2.9 FB_DMAm_Current_Buffer (Read-Only)

This is the ID for the buffer that will be used for the next access. The driver may use this to track the progress of the DMA activity. This value is displayed in Bytes.

6.2.10 FB_DMAm_COUNT (Read-Only)

This is the offset in the DMA buffer for the next access. The driver may use this to track the progress of the DMA activity. This value is displayed in Bytes.

6.2.11 FB_DMAm_RD_FIFO_CNT (Read-Only)

B[9:0] This is the amount of data available in the read FIFO in bytes. Software can use this to determine when the FIFO is empty. A value of 0x3FC indicates that there are 1020 or more bytes available.

B15: RD_EMPTY- '1' indicates that the read FIFO is empty

6.2.12 FB_DMAm_WR_FIFO_CNT (Read-Only)

B[9:0] This is the amount of space available in the write FIFO in bytes. Software can use this to determine when the FIFO is full. A value of 0x3FC indicated that there are 1020 or more bytes available.

B15: WR_FULL- '1' indicates that the write FIFO is full

6.2.13 FB_DMAm_ADDRESSn (Read/Write)

This is the 64-bit PCI address for DMA Channel m, buffer n. It must be double-word aligned (i.e. b[1:0] are reserved).

6.2.14 FB_DMAm_SIZEn (Read/Write)

This is the size in bytes of the buffer for DMA Channel m, buffer n. It must be an integer number of double-words (i.e. b[1:0] are reserved). The actual size is FB_DMAm_SIZEn + 4 Bytes. The maximum buffer size is 16MB.

6.2.15 FB_DMAm_CTRLn (Read/Write)

B0: Valid: Driver sets to '1' to indicate that this contains valid information. The DMA engine will set the error bit and halt if it is ready to use this descriptor and it is not valid.

B1: Halt: Driver sets to '1' to halt the DMA engine after this buffer is full.

B2: Loop: Driver sets to '1' to start back at descriptor 0 after this buffer is full. This has a higher priority than the HALT bit.

B3: Interrupt: Driver sets to '1' to generate an interrupt after this buffer is full.

If the last buffer is reached, and the HALT and LOOP bits are both '0', the DMA engine will loop.

If the last buffer is reached, and the HALT and LOOP bits are both '1', the DMA engine will halt and the Current_Buffer will be set to 0.

6.2.16 FB_DMAm_STATn (Read/Clear)

B0: Used (R/C): DMA engine sets to '1' to indicate that it has completely used this descriptor. The driver must clear this bit when it is ready to be used again. The DMA engine will set the error bit and PAUSE if it is ready to use this descriptor and the Used bit is set, unless the IgnoreUsed bit is set. The bits are cleared by writing 0x00 to the byte.

6.3 BAR2 – ADC Functional Block

This Function Block is for an Analog to Digital converter. This ADC has multiple channels, however all channels must use the same pacer clock. Each channel has a dedicated FIFO and DMA Channel.

Table 20: Multi-Channel ADC Functional Block

	Offset	0x03	0x02	0x01	0x00
Header	FB + 0x00	FB_ID			
	FB + 0x04	FB_DMA_BUFFERS	FB_DMA_CHANNELS	*Reserved*	*Reserved*
ADC Control	FB + 0x08	STOP_TRIG	START_TRIG	CLK_SRC	MODE_STATUS
	FB + 0x0C	CLK_DIV			
	FB + 0x10	CLK_DIV_CNTR			
	FB + 0x14	PRE_TRIGGER_CAPTURE (limited by FIFO size)			
	FB + 0x18	POST_STOP_CAPTURE			
	FB + 0x1C	SAMPLE_CNT			
	FB + 0x20	INT_ENA (Sample, Start, Stop, Threshold, Pacer Tick, etc)			
	FB + 0x24	INT_STAT		*Reserved*	
	FB + 0x28	CLK_SRC_GBL3	CLK_SRC_GBL2		*Reserved*
	FB + 0x2C	CLK_SRC_GBL7	CLK_SRC_GBL6	CLK_SRC_GBL5	CLK_SRC_GBL4
	FB + 0x30	AD_CONFIG (Maskable register – 16-bit)			
ADC Channel	FB + 0x34	CH0_FRONT_END_CONFIG (Maskable register – 16-bit)			
	FB + 0x38	CH0_FIFO_DATA_CNT			
	FB + 0x3C	CH0_THRESH_INT_ENA	CH0_THRESH_INT_STAT	CH0_FILTER	*Reserved*
	FB + 0x40	CH0_THRESH_HIGH			
	FB + 0x44	CH0_THRESH_LOW			
	FB + 0x48	CH0_LAST_SAMPLE			
	FB + 0x4C	CH1_FRONT_END_CONFIG (Maskable register – 16-bit)			
	FB + 0x50	CH1_FIFO_DATA_CNT			
	FB + 0x54	CH1_THRESH_INT_ENA	CH1_THRESH_INT_STAT	CH1_FILTER	*Reserved*
	FB + 0x58	CH1_THRESH_HIGH			
	FB + 0x5C	CH1_THRESH_LOW			
	FB + 0x60	CH1_LAST_SAMPLE			
	FB + 0x64	CH2_FRONT_END_CONFIG (Maskable register – 16-bit)			
	FB + 0x68	CH2_FIFO_DATA_CNT			
	FB + 0x6C	CH2_THRESH_INT_ENA	CH2_THRESH_INT_STAT	CH2_FILTER	*Reserved*
	FB + 0x70	CH2_THRESH_HIGH			
	FB + 0x74	CH2_THRESH_LOW			
	FB + 0x78	CH2_LAST_SAMPLE			
	FB + 0x7C	CH3_FRONT_END_CONFIG (Maskable register – 16-bit)			
	FB + 0x80	CH3_FIFO_DATA_CNT			
	FB + 0x84	CH3_THRESH_INT_ENA	CH3_THRESH_INT_STAT	CH3_FILTER	*Reserved*
	FB + 0x88	CH3_THRESH_HIGH			
	FB + 0x8C	CH3_THRESH_LOW			
	FB + 0x90	CH3_LAST_SAMPLE			
	FB + 0x94	CH4_FRONT_END_CONFIG (Maskable register – 16-bit)			
	FB + 0x98	CH4_FIFO_DATA_CNT			
	FB + 0x9C	CH4_THRESH_INT_ENA	CH4_THRESH_INT_STAT	CH4_FILTER	*Reserved*
	FB + 0xA0	CH4_THRESH_HIGH			
	FB + 0xA4	CH4_THRESH_LOW			
	FB + 0xA8	CH4_LAST_SAMPLE			
	FB + 0xAC	CH5_FRONT_END_CONFIG (Maskable register – 16-bit)			
	FB + 0xB0	CH5_FIFO_DATA_CNT			
	FB + 0xB4	CH5_THRESH_INT_ENA	CH5_THRESH_INT_STAT	CH5_FILTER	*Reserved*
	FB + 0xB8	CH5_THRESH_HIGH			
	FB + 0xBC	CH5_THRESH_LOW			
	FB + 0xC0	CH5_LAST_SAMPLE			
	FB + 0xC4	CH6_FRONT_END_CONFIG (Maskable register – 16-bit)			
	FB + 0xC8	CH6_FIFO_DATA_CNT			
	FB + 0xCC	CH6_THRESH_INT_ENA	CH6_THRESH_INT_STAT	CH6_FILTER	*Reserved*
	FB + 0xD0	CH6_THRESH_HIGH			
	FB + 0xD4	CH6_THRESH_LOW			

Table 20: Multi-Channel ADC Functional Block

	Offset	0x03	0x02	0x01	0x00
	FB + 0xD8	CH6_LAST_SAMPLE			
	FB + 0xDC	CH7_FRONT_END_CONFIG (Maskable register – 16-bit)			
	FB + 0 xE0	CH7_FIFO_DATA_CNT			
	FB + 0xE4	CH7_THRESH_INT_ENA	CH7_THRESH_INT_STAT	CH7_FILTER	*Reserved*
	FB + 0xE8	CH7_THRESH_HIGH			
	FB + 0xEC	CH7_THRESH_LOW			
	FB + 0xF0	CH7_LAST_SAMPLE			

6.3.1 FB_ID (READ-ONLY)

This is the functional block ID. This registry should read 0x00011000 for the ADC functional block.

6.3.2 FB_DMA_CHANNELS (READ -ONLY)

This register contains the number of DMA Channels in this Function Block. Each Channel contains a control register, and a set of Buffer Descriptor Registers.

6.3.3 FB_DMA_BUFFERS (READ-ONLY)

This register contains the number of Buffer Descriptors in each DMA Channel.

6.3.4 MODE_STATUS (READ/WRITE, READ-ONLY)

Selects the current mode of operation and indicates its triggering status.

B[3:0] : Mode

- 0x04 – Uninitialized. This is the power-on state. No converter initialization has taken place. Sampling is stopped, and all counters are reset and the triggering state machine is reset. Transition to any of the other Modes will start converter initialization (sampling will not start until initialization is complete).
- 0x00 – Reset. Sampling is stopped. All counters are reset and the triggering state machine is reset.
- 0x01 – Paused. Sampling is stopped, but the counters and triggering state machine maintain their state.
- 0x02 – Go, Single-Shot. After filling the buffer with the Post-Stop samples, capturing stops. The Mode must be set back to RESET in order to capture more samples.
- 0x03 – Go, Re-arm. After filling the buffer with the Post-Stop samples and the FIFO is empty, the triggering state machine is restarted, i.e. FIFO is filled with Pre-Start samples and waits for a start trigger.

B[7:4] : Status

- 0x08 – Uninitialized – The status when in the “Uninitialized” mode and the converter requires initialization.
- 0x09 - Initializing
- 0x00 – Stopped – The status when in the “Reset” mode, or in the “Uninitialized” mode and the converter does not require initialization.
- 0x01 – Filling Pre-Trigger buffer
- 0x02 – Waiting for start trigger
- 0x03 – Sampling/Waiting for stop trigger
- 0x04 – Filling Post-Stop buffer
- 0x05 – Wait to re-arm - Waiting until local FIFO is empty so that we can fill pre-trigger buffer
- 0x07 – Done capturing

6.3.5 CLK_SRC (READ/WRITE)

Selects the source for CLK_DIV from the clock bus.

Refer to Clock and Trigger Source on page 32 for list of valid values. For the ADC Function block, the Local Clock Source is Channel Threshold – One of the channels has exceeded the High or Low threshold.

6.3.6 START_TRIG (READ/WRITE)

Selects the start trigger from the clock bus. CLK_DIV will start counting after the start trigger, unless PRE_TRIGGER_CAPTURE is non-zero in which case CLK_DIV will start counting immediately.

Refer to Clock and Trigger Source on page 32 for list of valid values. For the ADC Function block, the Local Clock Source is Channel Threshold – One of the channels has exceeded the High or Low threshold.

6.3.7 STOP_TRIG (READ/WRITE)

Selects the stop trigger from the clock bus.

Refer to Clock and Trigger Source on page 32 for list of valid values. For the ADC Function block, the Local Clock Source is Channel Threshold – One of the channels has exceeded the High or Low threshold.

6.3.8 CLK_DIV (READ/WRITE)

Divider for the pacer clock. Pacer Clock Frequency = (Clk_Src_Frequency) / (1 + CLK_DIV). If synchronizing with the pacer clock from another Function Block (by using one of the CLK_GBL signals), this is typically set to 0.

6.3.9 CLK_DIV_CNTR (READ ONLY)

The current value of the Clock Divide Counter. This counter starts at a value of CLK_DIV, and counts down. When it reaches zero, a sample is taken. This is useful when using a slow sample clock.

6.3.10 PRE_TRIGGER_CAPTURE (READ/WRITE)

Number of samples to collect before the Start Trigger. The length is limited by the FIFO size – writing a value larger than the FIFO size will have indeterminate results.

6.3.11 POST_STOP_CAPTURE (READ/WRITE)

Number of samples to collect after the Stop Trigger.

6.3.12 SAMPLE_CNT (READ ONLY)

Total number of samples collected. This does not increment while in the “Waiting For Start Trigger” state. It also continues counting after a Re-Arm.

6.3.13 INT_ENA (MASKABLE READ/WRITE)

Each bit corresponds to an interrupt source. A value of ‘1’ enables the source, and a value of ‘0’ disables it. See below for a description of the sources.

6.3.14 INT_STAT (READ/CLEAR)

Each bit corresponds to an interrupt source. Reading a value of ‘1’ indicates that an event has occurred. Reading a value of ‘0’ indicates that the event has not occurred. Writing a ‘1’ will clear that bit.

B0: Sample – A sample has been taken.

B1: Channel Threshold – One of the channels has exceeded the High or Low threshold. Check the CH_THRESH_STAT registers.

B2: Pre-Start Buffer Filled

B3: Start Trigger

B4: Stop Trigger

B5:Post-Stop Buffer Filled

B6: Sampling has completed and the FIFO is empty (all data transferred to host)

B7: Pacer – The pacer clock has ticked.

6.3.15 CLK_SRC_GBLN

NOTE: If a CLK_SRC_GBL is unassigned in all function blocks, it defaults to System Clock/Immediate.

Selects the event to drive onto Clock Bus signal N.

B[7:0]:

0x00: Disables Clock Source

0x80: Sample – A sample has been taken.

0x81: Channel Threshold – One of the channels has exceeded the High or Low threshold. Check the CH_THRESH_STAT registers.

0x82: Pre-Start Buffer Filled

0x83: Start Trigger

0x84: Stop Trigger

0x85: Post-Stop Buffer Filled

0x86: Sampling has completed and the FIFO is empty (all data transferred to host)

0x87: Pacer – The pacer clock has ticked.

6.3.16 AD_CONFIG (MASKABLE READ/WRITE)

This provides up to 16 bits to configure the ADC Converter.

B[2]: MODE1

B[1]: MODE0

B[0]: CLKDIV

Sampling

The following Table describes the maximum sample rates and the Pacer Clock to Sample Clock ratios

Mode	MODE[1:0]	CLKDIV	Pacer:Sample	Max Pacer Clock(MHz)	Max Sample Rate(SPS)
High-Speed	00	1	256:1	27	105,468
High-Resolution	01	1	512:1	27	52,734
Low-Power	10	0	256:1	13.5	52,734
Low-Speed	11	0	512	5.4	10,547

6.3.17 CHN_FRONT_END_CONFIG (MASKABLE READ/WRITE)

This provides up to 16 bits to configure the Front End for this ADC Channel, to allow adjustment of gains, ranges, input calibration, etc.

B[7]: /PWDN 0 = ADC Channel is in low power mode 1= ADC channel is active

B[6]: /PGASD 0 = PGA in shutdown mode 1= PGA active

B[5]: SW_EN 0 = input switch tri-state 1 = Input switch enabled

B[4:3]: SW_IN[1:0]

- SW_IN[1:0] = 00: DAC Loopback
- SW_IN[1:0] = 01: Single- Ended Positive Input (Unipolar Mode)
- SW_IN[1:0] = 10: Single- Ended Negative Input (Unipolar Mode)
- SW_IN[1:0] = 11: Differential Input (Bipolar Mode)

B[2:0]: PGA_A[2:0] Sets the Range/Gain as below:

Table 21: Differential Mode Range/Gain Table

PGA_A[2:0]	Range	Gain
000	± 2.5 V	1
100	± 1.25 V	2
010	± 625 mV	4
110	± 312.5 mV	8
001	± 156.25 mV	16
101	± 78.125 mV	32
011	± 39.0626 mV	64
111	± 19.53125 mV	128

Table 22: Single-Ended Mode Range/Gain Table

PGA_A[2:0]	Range	Gain
000	0 - 5V	1
100	1.25 - 3.75 V	2
010	1.875 - 3.125V	4
110	2.1875 - 2.8125V	8
001	2.34375 - 2.65625 V	16
101	2.421875 - 2.578125V	32
011	2.460938 - 2.539063V	64
111	2.480469 - 2.519531V	128

NOTE: The Front End may take up to 100us to settle after writing to this register.

6.3.18 CHn_FILTER (READ/WRITE)

The programmable digital filter provides a single pole Infinite Impulse Response (IIR) filter on each channel. This a unity-gain filter. The filtered data has a value of:

$$D_n = \frac{D_{n-1} \times (2^{ORDER} - 1) + NewSample}{2^{ORDER}}$$

The response of the filter is shown in the Figure 2 below. Table 1 below shows the -3dB cutoff for each of the filter settings. Both the figure and the table are relative to the per-channel sample rate (f_s).

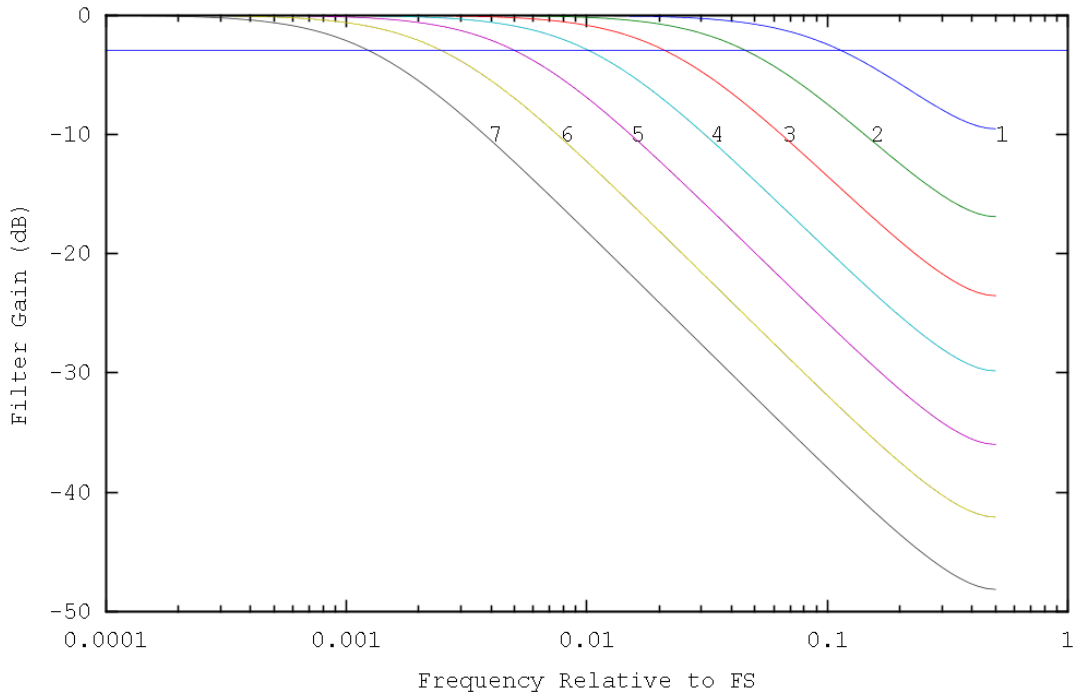


Figure 10: Filter Response with each ORDER Value

ORDER	-3 dB Cutoff
0	n/a
1	0.114791 * f_s
2	0.045995 * f_s
3	0.021236 * f_s
4	0.010255 * f_s

ORDER	-3 dB Cutoff
5	0.005042 * f_s
6	0.002501 * f_s
7	0.001246 * f_s

6.3.19 CHN_THRESH_STAT (READ/CLEAR)

This is the status register for the Threshold Detection. Reading a '1' indicates that the threshold has been crossed. Writing a '1' will clear the bit.

B0 – '1' = Low Threshold has been crossed

B1 – '1' = High Threshold has been crossed

6.3.20 CHN_THRESH_ENA (READ/WRITE)

These are interrupts enables for the threshold detection. Bit defines are above. An interrupt is generated (if not already generated) each time a sample is taken and the value is above the high threshold or below the low threshold.

6.3.21 CHN_THRESH_LOW (READ/WRITE)

Signed 32-bit value indicating the low threshold. If the input signal drops below this value, an interrupt or clock can be generated until the signal goes above this value. The 9 least significant bits are ignored from the actual threshold value.

NOTE: The threshold value should not exceed the ADC range. If the threshold value exceeds the ADC range unexpected results will occur.

6.3.22 CHN_THRESH_HIGH (READ/WRITE)

Signed 32-bit value indicating the high threshold. If the input signal goes above this value, an interrupt or clock can be generated until the signal goes below this value. The 9 least significant bits are ignored from the actual threshold value.

NOTE: The threshold value should not exceed the ADC range. If the threshold value exceeds the ADC range unexpected results will occur.

6.3.23 CH_LAST_SAMPLE (READ-ONLY)

The last sample read from the ADC Converter, after filtering. This is the same value that is written to the DMA FIFO.

6.4 BAR2 – DAC Functional Block

This Function Block is for a Digital to Analog converter. This DAC has multiple channels, however all channels must use the same sample clock. Each channel has a dedicated FIFO and DMA Channel.

Table 23: Multi-Channel DAC Functional Block

	Offset	0x03	0x02	0x01	0x00
Header	FB + 0x00	FB_ID			
	FB + 0x04	FB_DMA_BUFFERS	FB_DMA_CHANNELS	*Reserved*	*Reserved*
DAC Control	FB + 0x08	STOP_TRIG	START_TRIG	CLK_SRC	MODE_STATUS
	FB + 0x0C	CLK_DIV			
	FB + 0x10	CLK_DIV_CNTR			
	FB + 0x14	Reserved			
	FB + 0x18	POST_STOP_CONVERSIONS			
	FB + 0x1C	CONVERSION_CNT			
	FB + 0x20	INT_ENA (Conversion, Start, Stop, Error, Channel)			
	FB + 0x24	INT_STAT		*Reserved*	
	FB + 0x28	CLK_SRC_GBL3	CLK_SRC_GBL2		BOOKMARK_TRIG
	FB + 0x2C	CLK_SRC_GBL7	CLK_SRC_GBL6	CLK_SRC_GBL5	CLK_SRC_GBL4
	FB + 0x30	DA_CONFIG (Maskable register – 16-bit)			
DAC Channel	FB + 0x34	CH0_FRONT_END_CONFIG (Maskable register – 16-bit)			
	FB + 0x38	CH0_FIFO_DATA_CNT			
	FB + 0x3C	CH0_MARK_INT_ENA	CH0_MARK_INT_STAT	*Reserved*	*Reserved*
	FB + 0x40	*Reserved*			
	FB + 0x44	CH0_LAST_CONVERSION			
	FB + 0x48	CH1_FRONT_END_CONFIG (Maskable register – 16-bit)			
	FB + 0x4C	CH1_FIFO_DATA_CNT			
	FB + 0x50	CH1_ MARK_INT_ENA	CH1_ MARK_INT_STAT	*Reserved*	*Reserved*
	FB + 0x54	*Reserved*			
	FB + 0x58	CH1_LAST_ CONVERSION			
	FB + 0x5C	CH2_FRONT_END_CONFIG (Maskable register – 16-bit)			
	FB + 0x60	CH2_FIFO_DATA_CNT			
	FB + 0x64	CH2_ MARK_INT_ENA	CH2_ MARK_INT_STAT	*Reserved*	*Reserved*
	FB + 0x68	*Reserved*			
	FB + 0x6C	CH2_LAST_ CONVERSION			
	FB + 0x70	CH3_FRONT_END_CONFIG (Maskable register – 16-bit)			
	FB + 0x74	CH3_FIFO_DATA_CNT			
	FB + 0x78	CH3_ MARK_INT_ENA	CH3_ MARK_INT_STAT	*Reserved*	*Reserved*
	FB + 0x7C	*Reserved*			
	FB + 0x80	CH3_LAST_ CONVERSION			

6.4.1 FB_ID (READ-ONLY)

This is the functional block ID. This registry should read 0x00012000 for the DAC functional block.

6.4.2 FB_DMA_CHANNELS (READ -ONLY)

This register contains the number of DMA Channels in this Function Block. Each Channel contains a control register, and a set of Buffer Descriptor Registers.

6.4.3 FB_DMA_BUFFERS (READ-ONLY)

This register contains the number of Buffer Descriptors in each DMA Channel.

6.4.4 MODE_STATUS (READ/WRITE, READ-ONLY)

Selects the current mode of operation and indicates its triggering status.

B[3:0] : Mode

- 0x04 – Uninitialized. This is the power-on state. No converter initialization has taken place. Sampling is stopped, and all counters are reset and the triggering state machine is reset. Transition to any of the other Modes will start converter initialization (sampling will not start until initialization is complete).

- 0x00 – Reset. Sampling is stopped. All counters are reset and the triggering state machine is reset.
- 0x01 – Paused. Sampling is stopped, but the counters and triggering state machine maintain their state.
- 0x02 – Go, Single-Shot. After converting the Post-Stop number of values, converting stops. The Mode must be set back to RESET in order to convert more values.
- 0x03 – Go, Re-arm. After converting the Post-Stop number of values, the triggering state machine is restarted. DAC data is resumed from that last value sent.

B[7:4] : Status

- 0x08 – Uninitialized – The status when in the "Uninitialized" mode and the converter requires initialization.
- 0x09 - Initializing
- 0x00 – Stopped – The status when in the "Reset" mode, or in the "Uninitialized" mode and the converter does not require initialization.
- 0x01 – *Reserved*
- 0x02 – Waiting for start trigger
- 0x03 – Converting/Waiting for stop trigger
- 0x04 – Output Post-Stop buffer
- 0x05 – Wait to re-arm
- 0x07 – Done capturing

6.4.5 CLK_SRC (READ/WRITE)

Selects the source for CLK_DIV from the clock bus.

Refer to Clock and Trigger Source on page 32 for list of valid values. For the DAC Function block, the Local Clock Source is Channel Marker – One of the channels has the marker bit set.

6.4.6 START_TRIG (READ/WRITE)

Selects the start trigger from the clock bus. CLK_DIV will start counting after the start trigger.

Refer to Clock and Trigger Source on page 32 for list of valid values. For the DAC Function block, the Local Clock Source is Channel Marker – One of the channels has the marker bit set.

6.4.7 STOP_TRIG (READ/WRITE)

Selects the stop trigger from the clock bus.

Refer to Clock and Trigger Source on page 32 for list of valid values. For the DAC Function block, the Local Clock Source is Channel Marker – One of the channels has the marker bit set.

6.4.8 CLK_DIV (READ/WRITE)

Divider for the pacer clock. Pacer Clock Frequency = (Clk_Src_Frequency) / (1 + CLK_DIV). If synchronizing with the pacer clock from another Function Block (by using one of the CLK_GBL signals), this is typically set to 0.

6.4.9 CLK_DIV_CNTR (READ ONLY)

The current value of the Clock Divide Counter. This counter starts at a value of CLK_DIV, and counts down. When it reaches zero, a sample is taken. This is useful when using a slow sample clock.

6.4.10 POST_STOP_CONVERSIONS (READ/WRITE)

Number of conversions to send after the Stop Trigger.

6.4.11 CONVERSION_CNT (READ ONLY)

Total number of conversions. It also continues counting after a Re-Arm.

6.4.12 INT_ENA (MASKABLE READ/WRITE)

Each bit corresponds to an interrupt source. A value of '1' enables the source, and a value of '0' disables it. See below for a description of the sources.

6.4.13 INT_STAT (READ/CLEAR)

Each bit corresponds to an interrupt source. Reading a value of '1' indicates that an event has occurred. Reading a value of '0' indicates that the event has not occurred. Writing a '1' will clear that bit.

B0: Conversion – A value has been sent.

B1: Channel Marker – One of the channels has an enabled marker.

B2: Reserved

B3: Start Trigger

B4: Stop Trigger

B5:Post-Stop Conversions Completed

B6: Reserved

B7: Pacer – The pacer clock has ticked.

6.4.14 CLK_SRC_GBLN

NOTE: If a CLK_SRC_GBL is unassigned in all function blocks, it defaults to System Clock/Immediate.

Selects the event to drive onto Clock Bus signal N. Values are:

B[7:0] Clock Source Select

0x00:	Disable Clock Source
0x80:	Conversion – A value has been sent.
0x81:	Channel Marker – One of the channels has an enabled marker.
0x82:	Reserved
0x83	Start Trigger
0x84	Stop Trigger
0x85	Post-Stop Conversions Completed

6.4.15 DA_CONFIG (MASKABLE READ/WRITE)

Not Used

6.4.16 CH_FRONT_END_CONFIG (MASKABLE READ/WRITE)

Not Used

6.4.17 CH_MARKER_STAT (READ/CLEAR)

This is the status register for the Data Markers. Reading a '1' indicates that the Data Marker has been asserted. Writing a '1' will clear the bit.

The upper eight bits of the DAC value can be used for Markers. These Markers can be used to generate an interrupt when a certain part of the waveform is sent to the DAC. This allows an automated indication to the application software as to the state of the data being sent to the DAC. Marker bit 7 corresponds to bit 31 of the DAC data, and Marker bit 0 corresponds to bit 24 of the DAC data.

6.4.18 CH_MARKER_ENA (READ/WRITE)

These are interrupts enables for the Data Markers. Bit defines are above.

6.4.19 CH_LAST_CONVERSIONS (READ/WRITE)

The last value sent to the DAC Converter.

If the current Mode is “Reset” or the associated DMA engine is set to “Clear”, a write to this register will immediately update the DAC Converter.

6.5 BAR2 – Digital I/O Functional Block

This function block is for 14 bi-directional digital I/O. It doesn't provide any interrupts or DMA.

Table 24: Digital I/O Functional Block

	Offset	0x03	0x02	0x01	0x00
Header	FB + 0x00	FB_ID			
	FB + 0x04	FB_DMA_BUFFERS	FB_DMA_CHANNELS	*Reserved*	*Reserved*
DIO	FB + 0x08	DIO_INPUT_VAL			
	FB + 0x0C	DIO_OUTPUT_VAL			
	FB + 0x10	DIO_DIRECTION			

6.5.1 FB_ID (READ-ONLY)

This is the functional block ID. This registry should read 0x00003000 for the Digital I/O functional block.

6.5.2 FB_DMA_CHANNELS (READ -ONLY)

Has no DMA channels, reads 0

6.5.3 FB_DMA_BUFFERS (READ-ONLY)

Has no DMA buffers, reads 0

6.5.4 DIO_INPUT_VAL (READ ONLY)

This register provides the current value on the Digital I/O lines. The bits in the register correspond with the Digital I/O pins as follows:

Bit	CN6 Pin Number	Signal Name
13	14	PORT0_13
12	13	PORT0_12
11	12	PORT0_11
10	11	PORT0_10
9	10	PORT0_9
8	9	PORT0_8
7	8	PORT0_7
6	7	PORT0_6
5	6	PORT0_5
4	5	PORT0_4
3	4	PORT0_3
2	3	PORT0_2
1	2	PORT0_1
0	1	PORT0_0

6.5.5 DIO_OUTPUT_VAL (READ/WRITE)

The value to be output if the pins are configured as output. Bit assignments are the same as above.

6.5.6 DIO_DIRECTION (READ/WRITE)

Selects the direction of the I/O bit. 0=input, 1=output. Bit assignments are the same as above.

During power up Digital I/O defaults to inputs.

6.6 BAR2 – Reference Adjustment Digital Potentiometer Functional Block

This function block is for digital potentiometer SPI device that is used to adjust the on-board references. It doesn't provide any interrupts or DMA.

Each of these function blocks controls a dual potentiometer. One potentiometer adjusts the ADC reference and the other adjusts the DAC references. There are two Reference Adjustment Function Block; the first function block is for ADC0, DAC0, and DAC1, and the second function block is for ADC1, DAC2, and DAC3.

Table 25: Digital Potentiometer Functional Block

	Offset	0x03	0x02	0x01	0x00
Header	FB + 0x00	FB_ID			
	FB + 0x04	FB_DMA_BUFFERS	FB_DMA_CHANNELS	*Reserved*	*Reserved*
	FB + 0x08	*Reserved*		*Reserved*	GO_BUSY
	FB + 0x0C	OUTPUT_LATCH			
	FB + 0x10	*Reserved*			

6.6.1 FB_ID (READ-ONLY)

This is the functional block ID. This registry should read 0x0000F000 for the Digital I/O functional block.

6.6.2 FB_DMA_CHANNELS (READ -ONLY)

Has no DMA channels, reads 0

6.6.3 FB_DMA_BUFFERS (READ-ONLY)

Has no DMA buffers, reads 0

6.6.4 GO_BUSY (READ/CLEAR)

This register is used to start a write transaction to the reference adjustments. To perform a transaction, first enter data to be written into the OUTPUT_LATCH, than write a 1 to the GO_BUSY register.

B0: for reads, 0 = SPI interface is busy, 1 = SPI interface is ready. Write 0x01 to start SPI transaction.

6.6.5 OUTPUT_LATCH (READ/WRITE)

Contains the data to be written to the Digital Potentiometer SPI device.

When adjusting the digital potentiometer value, data can be written to the volatile memory to tune the voltage level. Once the voltage level is set, the digital potentiometer value can then be written to the nonvolatile in two ways, directly written the value to the nonvolatile memory or copying the current volatile memory value to the nonvolatile memory.

0x01XX: Writing to ADC adjustment volatile memory

0x02XX: Writing to DAC adjustment volatile memory

0x11XX: Writing to ADC adjustment nonvolatile memory

0x12XX: Writing to DAC adjustment nonvolatile memory

0x2100: Copy ADC adjustment volatile memory to nonvolatile memory

0x2200: Copy DAC adjustment volatile memory to nonvolatile memory

0x2300: Copy Both adjustment volatile megainmory to nonvolatile memory

0x3100: Copy ADC adjustment nonvolatile memory to volatile memory

0x3200: Copy DAC adjustment nonvolatile memory to volatile memory

0x3300: Copy Both nonvolatile memory to volatile memory

XX: is the data value range (0x0 – 0xFF)

6.7 BAR2 – Temperature Sensor Functional Block

This function block provides readout of the onboard temperature sensor. There are not any interrupts or DMA.

Table 26: Temperature Sensor Functional Block

	Offset	0x03	0x02	0x01	0x00
Header	FB + 0x00	FB_ID			
	FB + 0x04	FB_DMA_BUFFERS	FB_DMA_CHANNELS	Reserved	*Reserved*
	FB + 0x08	*Reserved*		TEMPERATURE_VAL	

6.7.1 FB_ID (READ-ONLY)

This is the functional block ID. This register should read 0x0000F001 for the Temperature Sensor functional block.

6.7.2 FB_DMA_CHANNELS (READ -ONLY)

Has no DMA channels, reads 0

6.7.3 FB_DMA_BUFFERS (READ-ONLY)

Has no DMA buffers, reads 0

6.7.4 TEMPERATURE_VAL (READ ONLY)

The current value on the temperature sensor as a signed 16-bit integer. B[2:0] of this register always read '0'. See the equation and table below for converting register value to temperature.

This register is updated every second.

$$Temperature = Register\ Value * 0.0078125\ C$$

Temperature (ºC)	Register Value
125	0x3E80
25	0x0C80
0.0625	0x0008
0	0x0000
-0.0625	0xFFF8
-25	0xF380
-55	0xE480

7 Troubleshooting

If you are having problems with your system, please try the following initial steps:

- **Simplify the System** – Remove modules one at a time from your system to see if there is a specific module that is causing a problem. Perform you troubleshooting with the least number of modules in the system possible.

- **Swap Components** – Try replacing parts in the system one at a time with similar parts to determine if a part is faulty or if a type of part is configured incorrectly.

If problems persist, or you have questions about configuring this product, contact RTD Embedded Technologies via the following methods:

Phone: +1-814-234-8087
E-Mail: techsupport@rtd.com

Be sure to check the RTD web site (http://www.rtd.com) frequently for product updates, including newer versions of the board manual and application software.

8 Additional Information

8.1 PC/104 Specifications

A copy of the latest PC/104 specifications can be found on the webpage for the PC/104 Embedded Consortium:

www.pc104.org

8.2 PCI and PCI Express Specification

A copy of the latest PCI and PCI Express specifications can be found on the webpage for the PCI Special Interest Group:

www.pcisig.com

9 Limited Warranty

RTD Embedded Technologies, Inc. warrants the hardware and software products it manufactures and produces to be free from defects in materials and workmanship for one year following the date of shipment from RTD Embedded Technologies, Inc. This warranty is limited to the original purchaser of product and is not transferable.

During the one year warranty period, RTD Embedded Technologies will repair or replace, at its option, any defective products or parts at no additional charge, provided that the product is returned, shipping prepaid, to RTD Embedded Technologies. All replaced parts and products become the property of RTD Embedded Technologies. Before returning any product for repair, customers are required to contact the factory for a Return Material Authorization (RMA) number.

This limited warranty does not extend to any products which have been damaged as a result of accident, misuse, abuse (such as: use of incorrect input voltages, improper or insufficient ventilation, failure to follow the operating instructions that are provided by RTD Embedded Technologies, “acts of God” or other contingencies beyond the control of RTD Embedded Technologies), or as a result of service or modification by anyone other than RTD Embedded Technologies. Except as expressly set forth above, no other warranties are expressed or implied, including, but not limited to, any implied warranties of merchantability and fitness for a particular purpose, and RTD Embedded Technologies expressly disclaims all warranties not stated herein. All implied warranties, including implied warranties for merchantability and fitness for a particular purpose, are limited to the duration of this warranty. In the event the product is not free from defects as warranted above, the purchaser's sole remedy shall be repair or replacement as provided above. Under no circumstances will RTD Embedded Technologies be liable to the purchaser or any user for any damages, including any incidental or consequential damages, expenses, lost profits, lost savings, or other damages arising out of the use or inability to use the product.

Some states do not allow the exclusion or limitation of incidental or consequential damages for consumer products, and some states do not allow limitations on how long an implied warranty lasts, so the above limitations or exclusions may not apply to you.

This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

RTD Embedded Technologies, Inc.
103 Innovation Boulevard
State College, PA 16803 USA
Telephone: 814-234-8087
Fax: 814-234-5218

www.rtd.com

sales@rtd.com
techsupport@rtd.com