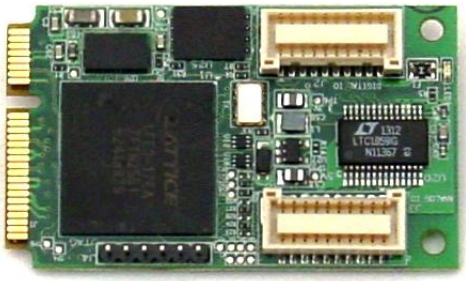
DS-MPE-DAQ0804

PCIe MiniCard Data Acquisition Port Module

Rev A.2 June 2016



Revision	Date	Comment
A.0	8/27/2014	Initial release
A.1	6/18/2015	Updated
A.2	6/20/2016	Data acquisition section expanded

FOR TECHNICAL SUPPORT PLEASE CONTACT:

support@diamondsystems.com

Diamond Systems Corporation
158 Commercial Street
Sunnyvale, CA 94086 USA
Tel 1-650-810-2500
Fax 1-650-810-2525
www.diamondsystems.com

CONTENTS

1. IMPORTANT SAFE HANDLING INFORMATION

WARNING!

ESD-Sensitive Electronic Equipment

Observe ESD-safe handling procedures when working with this product.

Always use this product in a properly grounded work area and wear appropriate ESD-preventive clothing and/or accessories.

Always store this product in ESD-protective packaging when not in use.

Safe Handling Precautions

This board contains a high density connector with many connections to sensitive electronic components. This creates many opportunities for accidental damage during handling, installation and connection to other equipment. The list here describes common causes of failure found on boards returned to Diamond Systems for repair. This information is provided as a source of advice to help you prevent damaging your Diamond (or any vendor's) embedded computer boards.

ESD damage – This type of damage is usually almost impossible to detect, because there is no visual sign of failure or damage. The symptom is that the board eventually simply stops working, because some component becomes defective. Usually the failure can be identified and the chip can be replaced. To prevent ESD damage, always follow proper ESD-prevention practices when handling computer boards.

Damage during handling or storage – On some boards we have noticed physical damage from mishandling. A common observation is that a screwdriver slipped while installing the board, causing a gouge in the PCB surface and cutting signal traces or damaging components.

Another common observation is damaged board corners, indicating the board was dropped. This may or may not cause damage to the circuitry, depending on what is near the corner. Most of our boards are designed with at least 25 mils clearance between the board edge and any component pad, and ground / power planes are at least 20 mils from the edge to avoid possible shorting from this type of damage. However these design rules are not sufficient to prevent damage in all situations.

A third cause of failure is when a metal screwdriver tip slips, or a screw drops onto the board while it is powered on, causing a short between a power pin and a signal pin on a component. This can cause overvoltage / power supply problems described below. To avoid this type of failure, only perform assembly operations when the system is powered off.

Sometimes boards are stored in racks with slots that grip the edge of the board. This is a common practice for board manufacturers. However our boards are generally very dense, and if the board has components very close to the board edge, they can be damaged or even knocked off the board when the board tilts back in the rack. Diamond recommends that all our boards be stored only in individual ESD-safe packaging. If multiple boards are stored together, they should be contained in bins with dividers between boards. Do not pile boards on top of each other or cram too many boards into a small location. This can cause damage to connector pins or fragile components.

Power supply wired backwards – Our power supplies and boards are not designed to withstand a reverse power supply connection. This will destroy each IC that is connected to the power supply (i.e. almost all ICs). In this case the board will most likely will be unrepairable and must be replaced. A chip destroyed by reverse power or by excessive power will often have a visible hole on the top or show some deformation on the top surface due to vaporization inside the package. **Check twice before applying power!**

Overvoltage on digital I/O line – If a digital I/O signal is connected to a voltage above the maximum specified voltage, the digital circuitry can be damaged. On most of our boards the acceptable range of voltages connected to digital I/O signals is 0-5V, and they can withstand about 0.5V beyond that (-0.5 to 5.5V) before being damaged. However logic signals at 12V and even 24V are common, and if one of these is connected to a 5V logic chip, the chip will be damaged, and the damage could even extend past that chip to others in the circuit

2. INTRODUCTION

2.1 Description

The DS-MPE-DAQ0804 is a rugged data acquisition PCIe MiniCard module with both analog and configurable digital I/O. It offers 8 single ended or 4 differential 16-bit analog inputs with an aggregate maximum sample rate of 100KHz, 2048 sample A/D FIFO, 4 16-bit analog outputs, and 14 configurable digital I/O lines. The buffered digital I/O lines can be optionally configured as either pulse width modulators or counter/timers. Diamond System' Universal Driver software provides driver support for all functions.

2.2 Features

- 8 single ended / 4 differential 16-bit analog inputs
- 100KHz maximum aggregate sample rate
- 2048 sample A/D FIFO with programmable threshold
- 4 analog input ranges: +/-10V, +/-5V, 0-10V, 0-5V
- 4 16-bit analog outputs
- 2 analog output ranges: 0-5V, 0-2.5V
- 14 digital I/O lines optionally configurable as:
  - 4 24-bit pulse width modulators
  - 8 32-bit counter/timers
- Latching connectors for increased ruggedness
- Universal Driver support for all functions

2.3 Operating System Support

- Linux 3.2.x
- Windows Embedded Standard 7, XP

2.4 Mechanical, Electrical, Environmental

- PCIe MiniCard full size format
- Dimensions: 50.95mm x 30mm (2" x 1.18")
- -40°C to +85°C ambient operating temperature
- Power input requirements: +3.3VDC +/- 5

2.5 Models

The DS-MPE-DAQ0804 product is available in two models as described below.

Product Number	Description
DS-MPE-DAQ0804	Analog I/O PCIe MiniCard Module with 8 A/D, 4 D/A & 14 DIO
DS-MPE-DAQ0800	Analog I/O PCIe MiniCard Module with 8 A/D & 14 DIO

3. PACKING LIST

The DS-MPE-DAQ0804 product comes with the PCIe MiniCard hardware assembly, a cable kit with one digital and one analog cable, and a hardware kit containing mounting screws.

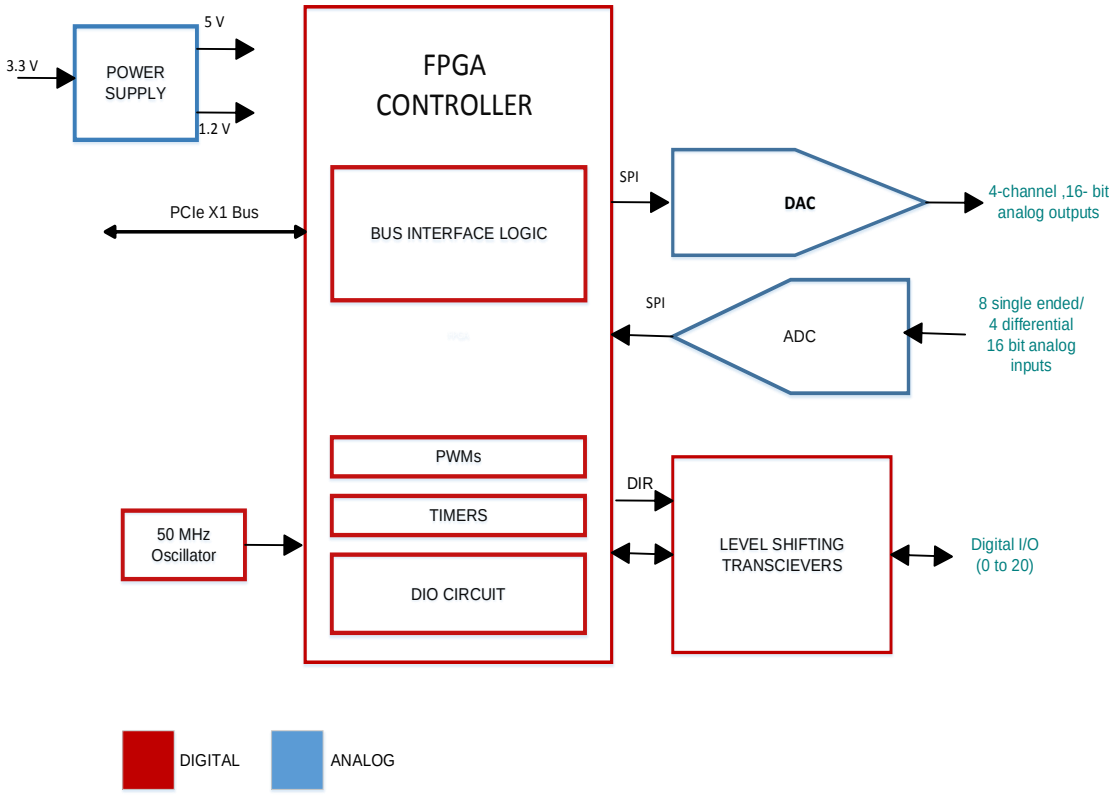
Quantity	Part Number	Description
1	915047x	DS-MPE-DAQ080x hardware assembly
1	6800502	Hardware Kit with mounting screws
1	CK-DAQ02	Cable Kit with analog and digital cables

4. FUNCTIONAL OVERVIEW

The DS-MPE-DAQ0804 is a PCIe MiniCard I/O module containing a combination of A/D, D/A, and DIO features using a PCIe interface. The A/D and D/A circuits are based on a high-integration A/D chip, LTC1859, with built-in single-ended/differential multiplexor and input range select circuit. A quad D/A chip AD5696 is used for the D/A features. Both analog components are powered by +5VDC and have their own integrated precision, low-drift voltage references.

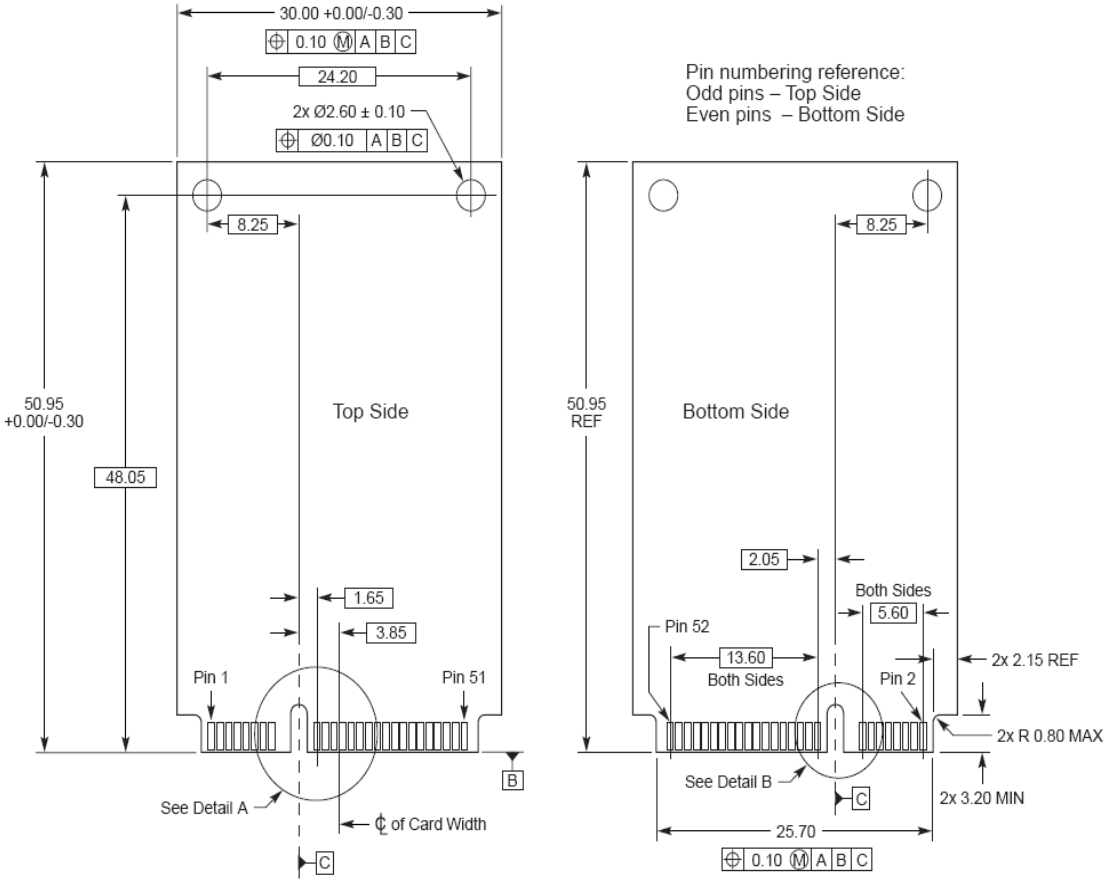
4.1 Functional Block Diagram

The DS-MPE-DAQ0804 block diagram is shown below.



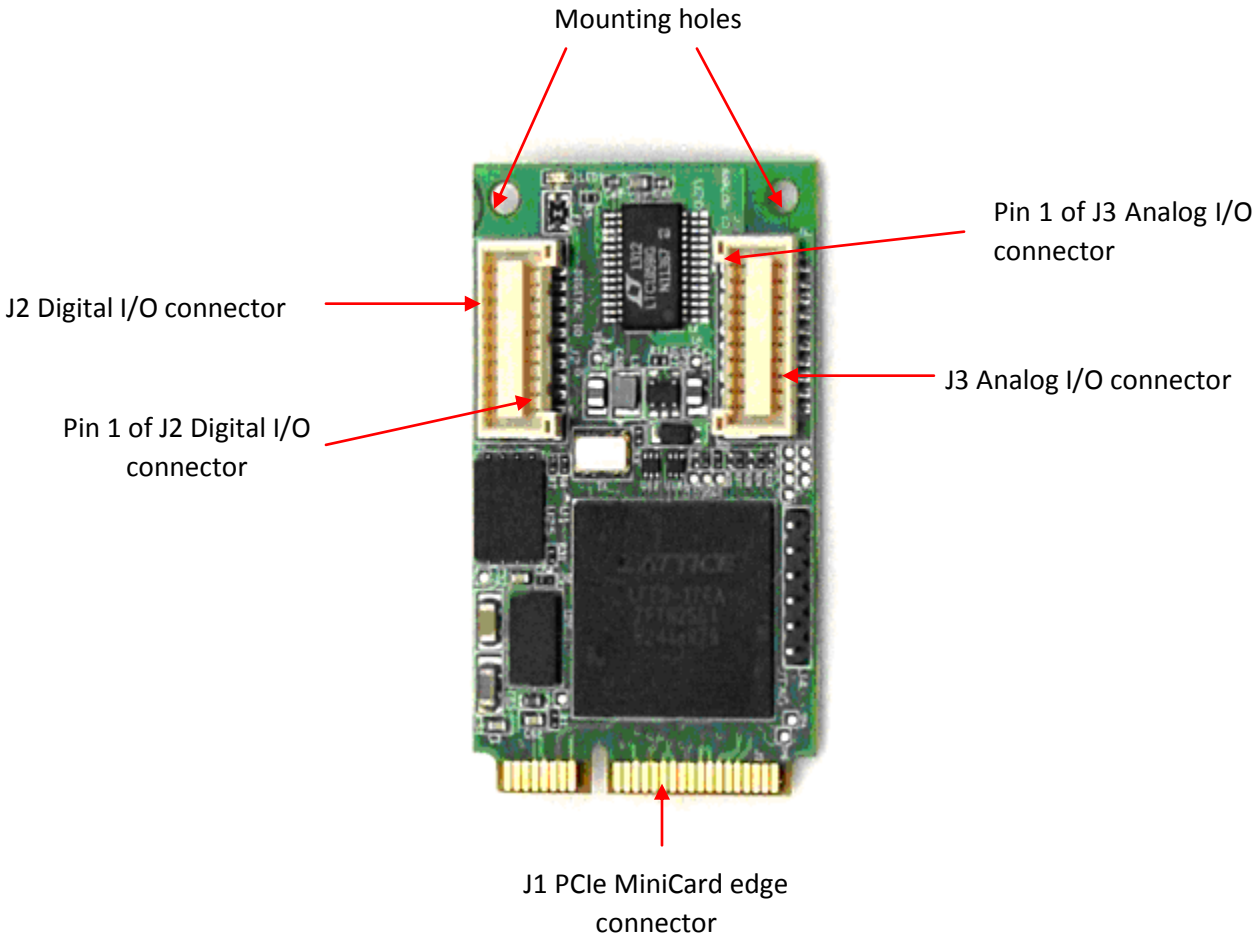
4.2 Mechanical Board Drawing

The DS-MPE-DAQ0804 conforms to the PCIe MiniCard electromechanical specification revision 1.2, full size format. Overall dimensions are 50.95mm L x 30.00mm W. The two mounting holes are isolated from the CPU ground and not connected to any ground lines.



5. INSTALLATION

The DS-MPE-DAQ0804 plugs in to any socket meeting the PCIe MiniCard specifications. It has two connectors, one for the analog I/O and one for the digital I/O, and a pair of mounting holes. To install the DS-MPE-DAQ0804, fully insert the board into a PCIe MiniCard connector and secure in place by inserting one screw from the hardware kit into each of the mounting holes, see the diagram below.



6. CONNECTOR PINOUT AND PIN DESCRIPTION

6.1 PCIe MiniCard Edge Connector (J1)

The DS-MPE-DAQ0804 module is compatible with the standard Mini PCIe socket pin out as shown below.

WAKE#	1	2	+3.3VAUX_3
COEX1	3	4	GND9
COEX2	5	6	+1.5V_1
CLKREQ#	7	8	UIM_PWR
GND1	9	10	UIM_DATA
REFCLK-	11	12	UIM_CLK
REFCLK+	13	14	UIM_RESET
GND2	15	16	UIM_VPP
	KEY		
RSVD(UIM_C8)	17	18	GND10
RSVD(UIM_C4)	19	20	W_DISABLE#
GND3	21	22	PERST#
PERN0	23	24	+3.3VAUX_4
PERP0	25	26	GND11
GND4	27	28	+1.5V_2
GND5	29	30	SMB_CLK
PETN0	31	32	SMB_DATA
PETP0	33	34	GND12
GND6	35	36	USB_D-
GND7	37	38	USB_D+
+3.3VAUX_1	39	40	GND13
+3.3VAUX_2	41	42	LED_WWAN#
GND8	43	44	LED_WLAN#
RSVD1	45	46	LED_WPAN#
RSVD2	47	48	+1.5V_3
RSVD3	49	50	GND14
RSVD4	51	52	+3.3VAUX_5

6.2 Digital I/O (J2)

The digital I/O signals are provided on a miniature 20-pin latching connector (J2).

DIO 0	1	2	DIO 1
DIO 2	3	4	DIO 3
DIO 4	5	6	DIO 5
CTR 0 I/O / DIO 6	7	8	DIO 7 / CTR 1 I/O
CTR 2 I/O / DIO 8	9	10	DIO 9 / CTR 3 I/O
CTR 4 I/O / DIO 10	11	12	DIO 11 / CTR 5 I/O
CTR 6 I/O / DIO 12	13	14	DIO 13 / CTR 7 I/O
PWM 0 Out / DIO 14	15	16	DIO 15 / PWM 1 Out
PWM 2 Out / DIO 16	17	18	DIO 17 / PWM 3 Out
+3.3V (fused)	19	20	Digital Ground

Connector Part Number / Description
JST BM20B-GHDS-G-TF 20-pin (2x10) 1.25mm pitch vertical SMT latching connector

6.3 Analog I/O (J3)

The analog I/O signals are provided on a miniature 20-pin latching connector (J3).

D/A 0	**1**	**2**	D/A 1
D/A 2	**3**	**4**	D/A 3
Analog Ground	**5**	**6**	Analog Ground
A/D 0	**7**	**8**	A/D 1
A/D 2	**9**	**10**	A/D 3
A/D 4	**11**	**12**	A/D 5
A/D 6	**13**	**14**	A/D 7
Analog Ground	**15**	**16**	Analog Ground
DIO 18	**17**	**18**	DIO 19 / A/D ext trigger
D/A ext trigger / DIO 20	**19**	**20**	Digital Ground

Connector Part Number / Description
JST BM20B-GHDS-G-TF 20-pin (2x10) 1.25mm pitch vertical SMT latching connector

7. ARCHITECTURE OVERVIEW

7.1 Bus Interface

The FPGA utilizes a PCI Express x1 bus interface. The design includes a PCIe core to implement the PCIe interface.

7.2 FPGA

The FPGA is a Lattice Semiconductor ECP3 family (LFE3) in BGA256 package. The FPGA includes an SPI core to gain access to the FPGA configuration flash memory. This allows the FPGA code to be updated in the field without requiring a JTAG cable or 3rd party software.

7.3 A/D Circuit

The module uses the Linear Technology LT1859 high-integration A/D converter chip for the A/D functionality and includes the following features:

- 16-bit A/D with 100KHz sampling rate
- 8 channel single ended / 4 channel differential multiplexor
- Input protection up to +/-25V
- Programmable input ranges: 0-10V, 0-5V, +/-10V, +/-5V
- Precision 2.5V low-drift reference voltage

7.4 D/A Circuit

The module uses the Analog Devices AD5686R D/A converter chip for the D/A functionality and includes the following features:

- 4 channel 16-bit D/A
- Single channel and simultaneous update modes
- Programmable output ranges: 0-2.5V, 0-5V
- Precision 2.5V low-drift reference voltage

7.5 Digital I/O

The 14 digital I/O lines are provided by the FPGA. They can operate in simple I/O mode in the form of 8-bit and 1-bit ports or in counter/timer and pulse width modulator modes. All bits have independent 1-bit drivers with independent direction control.

The digital I/O output voltage is +3.3V. The digital I/O lines can be software-configured for pull-up / down resistors. All lines are configured together for up or down.

7.6 Counter Timers and Clock Sources

The FPGA offers 8 32-bit counter/timers with programmable up/down counting, divide-by-n function, and square wave / pulse output. The Counters can be latched and read while counting.

7.7 Pulse Width Modulators

The FPGA includes 4 24-bit pulse-width modulator (PWM) circuits. Each circuit includes a period register as well as a duty cycle register. Both registers may be updated in real-time without stopping the PWM. Duty cycles from 0-100% inclusive are supported, as well as both positive and negative output polarity. The PWM clock may be selected from the on-board 50MHz clock or a 1MHz clock derived from the 50MHz clock. The PWM outputs are enabled on general purpose I/O pins with limited voltage and current capability. The user must determine whether these pins provide the appropriate voltage and current levels for the intended application or whether additional buffering or amplification is required.

.

7.8 Interrupt Circuit

Interrupts enable the board to request service independently of the program operation, typically in response to a user defined time interval or external event. The board supports interrupts from variety of sources including the digital I/O channels and counters/timers. The application is responsible for providing the interrupt service routine to respond to the interrupt request. An unserviced interrupt request may cause unpredictable 12ibble12. Diamond's Universal Driver software includes built-in interrupt handling routines that can link to user-defined code. This software lets you define the conditions that will generate an interrupt and then define the behavior of the system when an interrupt occurs.

8. DATA ACQUISITION CIRCUIT DESCRIPTION

8.1 Features

A/D Features

- 16 analog voltage inputs
- 16-bit resolution (1 part in 65536)
- Programmable input ranges: 0-5V, 0-10V, +/-5v, +/-10V
- Single-ended and differential input configuration options
- Precision, low-drift 2.5V reference voltage
- 100KHz maximum total A/D sample rate (all active channels combined)
- Integrated 2048-sample FIFO and interrupt service for efficient high-speed sampling

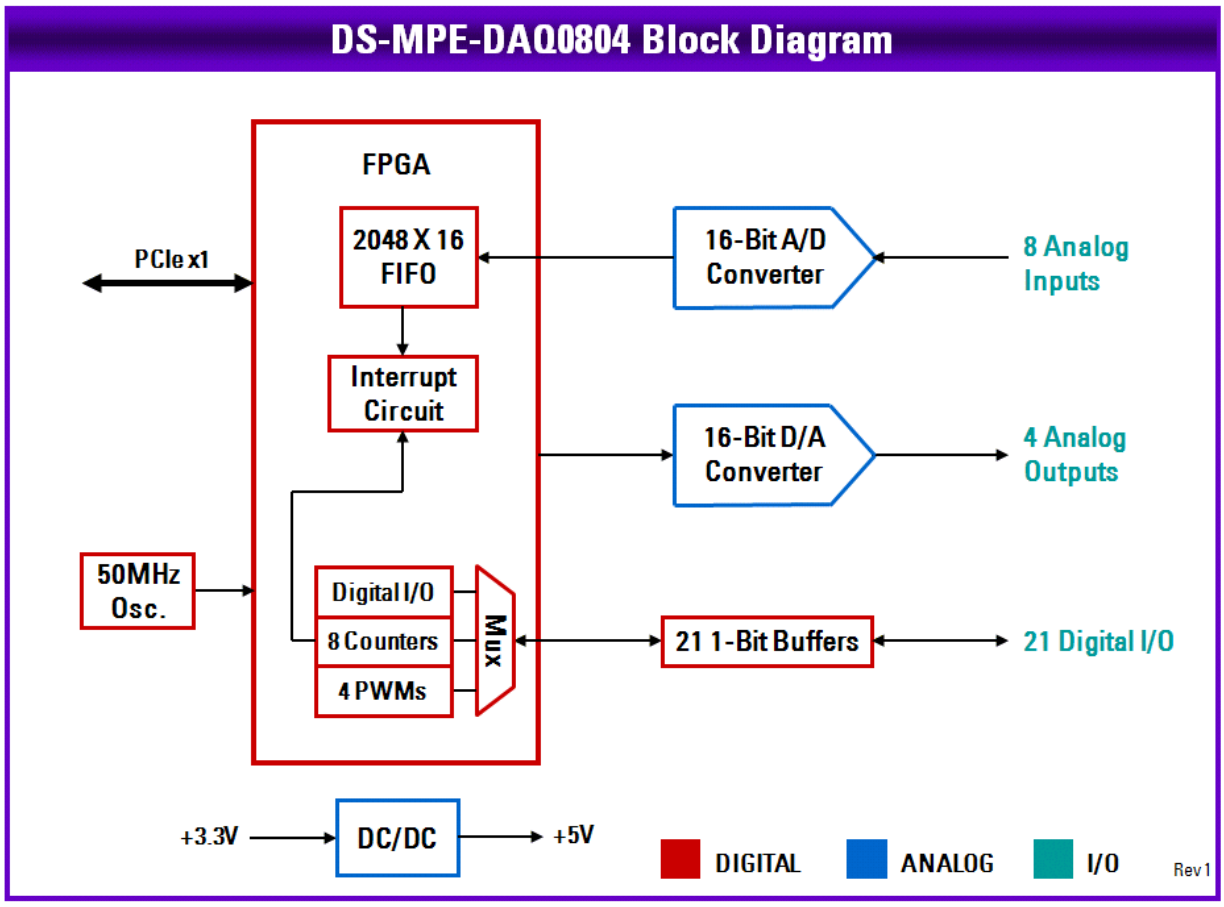
D/A Features

- 4 analog voltage outputs
- 16-bit resolution (1 part in 65536)
- Single-channel and multi-channel simultaneous update modes
- Programmable output range: 0-5V, 0-2.5V
- 30KHz update rate capability
- Waveform generator on 1 to 4 outputs with user-defined waveforms and 2048-sample waveform buffer

Digital I/O features

- 21 digital I/O lines
- 3.3V logic levels
- User-configurable 10K pull-up / pull-down resistors
- Individually programmable direction for each line
- 8-bit programmable edge detection circuit
- Buffers for protection and higher current drive
- 8 32-bit counter/timers with up counting, down counting, pulse output, and interrupt features
- 4 24-bit pulse-width modulators with programmable duty cycle and output polarity
- Interrupt support for A/D, digital I/O, and counter/timer circuits

8.2 Block Diagram



8.3 Analog-to-Digital Circuit

8.3.1 A/D Circuit Overview

The DS-MPE-DAQ0804 implements an analog input circuit based on the LTC1859 16-bit A/D converter, or ADC. The LTC1859 is capable of operating at up to 100,000 samples per second in total, either on a single channel or on any number of channels. The ADC supports programmable input ranges of 0-5V, 0-10V, +/-5V, and +/-10V. The input range can be changed on a sample by sample or channel by channel basis.

The ADC includes a built-in channel multiplexor which can treat the 8 inputs as either 8 single-ended channels or 4 differential channels. In single-ended mode, each channel is measured with respect to analog ground, i.e. the input signal must be connected between the input pin and the analog ground pin on the I/O connector. In differential mode, the voltage is measured as the difference between the two input pins, and the analog ground pin is not part of the measurement. However, both the high and low side of the input signal pair must remain within the “common mode voltage” of the A/D at all times to ensure accurate readings.

8.3.2 A/D Channel Selection, Sampling, and Timing

A/D conversions may be triggered by several sources, including a software command, a programmable clock in the circuit’s FPGA, and an external digital signal connected to one of the board’s digital I/O ports. These options provide maximum flexibility in tailoring the circuit’s performance to a wide variety of real-world applications. Software triggering is mostly used for low-speed sampling needs or where the exact elapsed time between samples is not critical. External triggering is used when the A/D sampling must be synchronized to some external

activity such as a rotary or linear encoder. Clock timing is used for high speed sampling or where the time between samples must be precise.

A channel sequencer circuit in the FPGA controls which channels sampled by the ADC. The sequencer can be programmed to select any single channel repeatedly, or it can be programmed to select any consecutive group of channels. As described above, the A/D front end consists of two 8-channel multiplexors connected to two analog inputs of the ADC. In single-ended mode, the channel sequencer will treat the 16 inputs as channels 0-15 with reference to analog ground, and in differential mode the sequencer will treat them as 8 differential channels, with channels 0-7 forming the high side of the inputs (renamed as channels 0+ to 7+) and channels 8-15 forming the low side of the inputs (renamed as channels 0- to 7-).

The sequencer works in conjunction with two available sampling modes called sample and scan. In sample mode, each A/D clock or trigger event causes one A/D conversion to occur. If the sequencer is programmed for a single channel, each successive A/D conversion will sample the same channel. This method is useful for high speed sampling of an AC signal. If the sequencer is programmed for a range of channels, then each successive A/D clock will cause the next channel in the selected range to be sampled, with the result that all the selected input channels are sampled one at a time in round robin fashion.

In scan mode, each A/D clock generates one A/D conversion on all channels selected by the sequencer (called the scan range) in tight succession. This method is typically used with the programmable clock circuit to measure a group of channels all at once. The number of selected channels is called the scan size. The timing of scan mode differs significantly from sample mode in that all selected channels are sampled as closely together as possible in time instead of being spread out equally in time as is done in sample mode. The analog input circuit does not support simultaneous sampling mode, where all inputs are sampled at exactly the same time.

One very important fact to remember is that because scan sampling generates one A/D conversion for every channel in the scan range, the total sample rate of the circuit is equal to the clock rate times the scan size. This total sample rate must be kept at or below the hard 100KHz limit of the A/D converter.

A second very important fact is that the fastest possible sample rate for any channel is 100KHz divided by the scan size in scan mode or the number of selected channels in sample mode. Thus it is not possible to sample more than one channel at 100KHz. If two channels are selected, the maximum sample rate for each channel is 50KHz, and so on.

8.3.3 A/D FIFO and High Speed Sampling

The FPGA contains a 2048 sample FIFO which supports high-speed A/D sampling. The FIFO enables the CPU to avoid having to respond every time an A/D conversion occurs, which would consume too much processor time when high speed sampling is being executed. Instead, the A/D samples are stored in the FIFO, and when the number of samples reaches a user-defined threshold, an interrupt occurs. The software can then respond to the interrupt and read out a large number of samples all at once. The Diamond Systems Universal Driver software provides full support for high speed A/D sampling with FIFO and interrupt support. Please refer to that user manual for operating details. When using Universal Driver software, the interrupt rate is equal to the total A/D sample rate divided by the programmed FIFO threshold, because each time the interrupt service routine runs, it will read out the number of samples equal to the threshold value. The threshold is programmable so that the application software can optimize the interrupt rate for its needs. In general the interrupt rate should not exceed 1KHz, and in most cases an interrupt rate of 100-200Hz is recommended.

8.3.4 A/D Operation

In essence, an A/D converter converts an analog voltage to a number that indicates the ratio between that voltage and the full range of voltages that the A/D can measure. A 16-bit A/D converter means that all input voltages are represented by a 16-bit binary number ranging from 0 to 65535 (2^{16}-1), or binary 0000 0000 0000 0000 to 1111 1111 1111 1111. There are two groups of input ranges, unipolar and bipolar. Unipolar ranges consist of only positive voltages (0-5V, 0-10V), and bipolar ranges include both negative and positive voltages (+/-5V, +/-10V). For a unipolar input range, the A/D values are straight binary and will range from 0 (binary 0000 0000 0000 0000) for 0V to 65535 (binary 1111 1111 1111 1111) for the top end of the range. For a bipolar input range, the A/D

values are 2s complement and will range from -32768 (binary 1000 0000 0000 0000) to +32767 (binary 0111 1111 1111 1111).

The tables below summarize the relationship between input voltage and A/D values for the general case. Note that the nominal upper limit of the input range (5.0000V or 10.0000V) is not achievable, since this voltage would require a 17 bit number (2^{16} or 1 0000 0000 0000 0000). V_{FS} means the full-scale input voltage, either 5V or 10V.

Bipolar Input Ranges

A/D Code	A/D code binary	Formula	+/-5V input range	+/-10V input range
-32768	1000 0000 0000 0000	$-V_{FS}$	-5.0000V	-10.0000V
-32767	1000 0000 0000 0001	$-V_{FS}$ + 1 LSB	-4.9998V	-9.9997V
-1	1111 1111 1111 1111	-1 LSB	-0.153mV	-0.305mV
0	0000 0000 0000 0000	0V	0.0000V	0.0000V
1	0000 0000 0000 0001	+1 LSB	0.153mV	.305mV
32767	0111 1111 1111 1111	V_{FS} - 1 LSB	4.9998V	9.9997V

Unipolar Input Ranges

A/D Code	A/D code binary	Formula	0-5V input range	0-10V input range
0	0000 0000 0000 0000	0V	0.0000V	0.0000V
1	0000 0000 0000 0001	1 LSB (V_{FS} / 65536)	0.076mV	0.153mV
32767	0111 1111 1111 1111	V_{FS} / 2 - 1 LSB	2.4999V	4.9998V
32768	1000 0000 0000 0000	V_{FS} / 2	2.5000V	5.0000V
32769	1000 0000 0000 0001	V_{FS} / 2 + 1 LSB	2.5001V	5.0002V
65535	1111 1111 1111 1111	V_{FS} - 1 LSB	4.9999V	9.9998V

8.3.5 A/D Resolution

The smallest change in input voltage that can be detected is 1/(2^{16}), or 1/65536, of the full-scale input range. This smallest change results in an increase or decrease of 1 in the A/D code, and is referred to as 1 LSB (1 Least Significant Bit). The resolution is always 16 bits, but the value of 1 LSB will vary with the input range.

Polarity	*Input Range*	*Resolution (1LSB)*
Bipolar	±10V	305µV
Bipolar	±5V	153µV
Unipolar	0 - 10V	153µV
Unipolar	0 - 5V	76µV

8.3.6 Input Range Selection

The input range of the A/D circuit is programmable in software and can be selected from the 4 values shown in the table above. Refer to the Universal Driver software user manual A/D section for details. The input range can be changed anytime, so that you can use different ranges for different input signals based on the best match. In

general, you should select the highest gain (smallest input range) that allows the A/D converter to read the full range of voltages over which the input signals will vary. For example, if you have a signal that ranges from 0V minimum to 3V maximum, use the 0-5V range for best resolution. An input range that is too small causes the A/D converter to clip at either the high end or low end, and you will not be able to read the full range of voltages on your input signals.

8.3.7 Converting A/D Readings to Volts or Engineering Units

The A/D always returns a 16-bit binary number that represents the value of the input voltage relative to the selected input range. This number needs to be converted to a meaningful value in order to be used in your application. The first step is to convert it back to the actual measured voltage. Afterwards, you may need to convert the voltage to some other engineering units such as temperature in degrees C or weight in grams.

Since there are many possible formulas for converting the input voltage to engineering values, this secondary step is not described here. Only conversion to input voltage is described. However, you can combine both voltage and engineering unit conversions into a single formula if desired.

To convert the A/D value to its corresponding input voltage, use the following formulas.

Conversion Formula for Bipolar Input Ranges

Input voltage = A/D value / 32768 * Full-scale input voltage

where the A/D value is a 2s complement number ranging from -32768 to 32767, and the full-scale voltage is the nominal maximum value, either 5V or 10V.

Example:
For bipolar input range ±5V, full-scale input voltage = 5V

For an A/D value of 17761: Input voltage = 17761 / 32768 * 5V = 2.710V

For an A/D value of -12345: Input voltage = -12345 / 32768 * 5V = -1.884V

Conversion Formula for Unipolar Input Ranges

Input voltage = A/D value / 65536 * Full-scale input voltage

where the A/D value is a straight binary number ranging from 0 to 65535, and the full-scale voltage is the nominal maximum value, either 5V or 10V.

Example:
For unipolar input range 0-5V, full-scale input voltage = 5V.

For an A/D value of 17761: Input voltage = 17761 / 65536 * 5V = 1.355V

8.3.8 Measurement Accuracy and Calibration

Although the A/D circuit has 16-bit resolution (meaning it can resolve input voltages to within $1/2^{16}$ or 1/65536 of its input range), the measurement accuracy is not necessarily the same. All A/D circuits exhibit two inherent errors known as offset and gain errors. In addition all A/D circuits will experience some minor drift as the ambient temperature changes, as well as lesser drift over long periods of time.

To minimize the size and cost of the A/D circuit while still providing reasonable performance, the A/D circuit on DS-MPE-DAQ0804 does not provide calibration features to eliminate these errors. Instead the circuit was designed to minimize the inherent offset, gain, and drift characteristics. The specifications at the end of this manual indicate the worst case offset, gain, and temperature drift errors in the circuit. Most boards will exhibit errors much lower than these values (typically 25-50% of the published values), so the A/D measurements can typically be used satisfactorily in your application without concern.

If higher accuracy is required, a software or digital calibration technique can be implemented in your application. Measure two known voltages close to the lower and upper limits of the input range you are using. The ideal values are between 1% and 5% at the bottom end and 95% and 99% at the top end. Compare the actual A/D readings of these input voltages to the expected values, and then store the differences in a file. Then you can use these errors to apply an offset and gain factor to each A/D reading on that board forevermore. Although this technique does not eliminate temperature and drift errors, it will eliminate the largest portion of the total error and provide more accurate measurements and better overall product performance.

When considering accuracy, it is important to consider all the contributions to measurement error. In addition to the inherent A/D errors, the sensors connected to the A/D also have their own offset, gain, and temperature/time drift errors. Usually these errors will be published as a simple accuracy figure in a percentage form. One percent error on a 16-bit A/D is 655 LSBs, so if your sensor has a 1% error specification, that is already more than 10x the typical A/D reading error, and therefore software calibration of the A/D readings alone will not provide any material benefit. However the same technique described above can be applied with the sensor connected using known input conditions (for example empty scale and scale with a precision weight applied), so that you can eliminate total system error.

To include A/D error values into the input voltage calculation, use this formula:

Input voltage = ((ADC – Cmin) / (Cmax – Cmin)) * (Vmax – Vmin) + Vmin

ADC = A/D code

Vmax = Test input voltage at top end of A/D scale used in calibration procedure

Vmin = Test input voltage at bottom end of A/D scale used in calibration procedure

Cmax = A/D code at Vmax input

Cmin = A/D code at Vmin input

This formula works for both unipolar and bipolar input ranges.

8.3.9 Input Impedance

Another factor that can significantly affect A/D measurement accuracy is the ratio between the input impedance of the A/D circuit and the output impedance of the signal being measured. The voltage seen by the A/D converter is a simple resistor divider between the two impedances, with the input signal at the top of the divider and the A/D in the middle. The higher the output impedance of the input signal, the greater the error will be. Conversely, the higher the input impedance of the A/D circuit, the lower the error will be. Due to severe space constraints of the small minicard form factor, it was not possible to install a high-impedance front end to the A/D converter, so the overall circuit input impedance is equal to the A/D converter's input impedance. For bipolar input ranges, the input impedance is 31K Ohms typical, and for unipolar input ranges the input impedance is 42K Ohms typical. For low errors the formula for calculating the maximum output impedance of the input signal may be approximated as follows:

Maximum input signal impedance = A/D impedance x tolerance

For an allowable 0.1% error (66 LSB), the input signal should have an output impedance of 31 ohms max for bipolar input ranges and 42 ohms max for unipolar input ranges. Ideally the output impedance of the circuit should be less than 10 ohms.

8.4 Digital-to-Analog Circuit

8.4.1 Overview

The DS-MPE-DAQ0804 data acquisition circuit uses a 16-bit D/A converter. This means that all output voltages are defined by a 16-bit binary number, or D/A code, ranging from 0 to 65535 (2^{16}-1), or binary 0000 0000 0000 0000 to 1111 1111 1111 1111. The DS-MPE-DAQ0804 D/A circuit supports only unipolar output voltage ranges (positive voltages only). For a unipolar output range, the D/A codes are straight binary and will range from 0 (binary 0000 0000 0000 0000) for 0V to 65535 (binary 1111 1111 1111 1111) for the top end of the range.

The tables below summarize the relationship between D/A codes and output voltages. Note that the nominal upper limit of the output range (5.0000V or 2.5000V) is not achievable, since this voltage would require a 17 bit number (2^{16} or 1 0000 0000 0000 0000). V_{FS} means the full-scale output voltage, either 5V or 2.5V.

D/A Output Codes to Output Voltages

D/A Code	D/A code binary	Formula	0-5V output range	0-2.5V output range
0	0000 0000 0000 0000	0V	0.0000V	0.00000V
1	0000 0000 0000 0001	1 LSB (V_{FS} / 65536)	0.0763mV	0.0381mV
32767	0111 1111 1111 1111	V_{FS} / 2 - 1 LSB	2.4999V	1.24996V
32768	1000 0000 0000 0000	V_{FS} / 2	2.5000V	1.25000V
32769	1000 0000 0000 0001	V_{FS} / 2 + 1 LSB	2.5001V	1.25004V
65535	1111 1111 1111 1111	V_{FS} - 1 LSB	4.9999V	2.49996V

8.4.2 D/A Resolution

The smallest change in output voltage that can be obtained is 1/(2^{16}), or 1/65536, of the full-scale output range. This smallest change results from an increase or decrease of 1 in the D/A code and is therefore referred to as 1 LSB (1 Least Significant Bit). The D/A resolution is always 16 bits, but the value of 1 LSB will vary with the output range.

Polarity	*Output Range*	*Resolution (1LSB)*
Unipolar	0 - 5V	76.3µV
Unipolar	0 – 2.5V	38.1µV

8.4.3 Output Range Selection

The output range of the D/A circuit is programmable in software and can be selected from the 2 values shown in the table above. Refer to the Universal Driver user manual D/A functions for details. The output range can be changed anytime. However unlike the A/D circuit, changing the output range affects all output channels simultaneously. In general, you should select the smallest output range that allows the D/A converter to cover the full range of output voltages you need for your application. For example, if you need the output to vary from 0V minimum to 2V maximum, use the 0-2.5V output range for best resolution. Note that the DS-MPE-DAQ0804 D/A circuit cannot output negative voltages.

8.4.4 D/A Conversion Formula

The D/A code written to the D/A circuit is always a straight binary integer ranging between 0 and 65535 (2^{16}-1). You may choose to round up or down to achieve the most accurate output value.

To calculate the required D/A code for the desired output voltage:

D/A code = (Desired output voltage / Full scale voltage) * 65536

where Full scale voltage is either 5V or 2.5V depending on the selected output range.

To determine the output voltage resulting from a given D/A code:

Output voltage = (D/A code / 65536) * Full scale voltage

Examples:
For unipolar output range 0-5V, full-scale voltage = 5.000V.

For a desired output voltage of 1.000V: D/A code = (1.000V / 5.000V) * 65536 = 13107

For unipolar output range 0-2.5V, full-scale voltage = 2.500V.

For a desired output voltage of 1.000V: D/A code = (1.000V / 2.500V) * 65536 = 26124

8.4.5 Output Accuracy and Calibration

Although the D/A circuit has 16-bit resolution (meaning it can resolve output voltages to within 1/2^{16} or 1/65536 of its output range), the output accuracy is not necessarily the same. As with A/D circuits described earlier, all D/A circuits exhibit two inherent errors known as offset and gain errors. In addition all D/A circuits will experience some minor drift as the ambient temperature changes, as well as lesser drift over long periods of time.

To minimize the size and cost of the D/A circuit while still providing reasonable performance, the D/A circuit on DS-MPE-DAQ0804 does not provide calibration features to eliminate these errors. Instead the circuit was designed to minimize the inherent offset, gain, and drift characteristics. The specifications at the end of this manual indicate the worst case offset, gain, and temperature drift errors in the circuit. Most boards will exhibit errors much lower than these values (typically 25-50% of the published values), so the D/A can typically be used satisfactorily in your application without concern.

If higher accuracy is required, a software or digital calibration technique can be implemented in your application. Write the bottom and top output codes (0 and 65535) to the D/A circuit and measure the actual output voltages. Compare the actual voltages to the expected values, and then store the differences in a file. Then you can use these errors to apply an offset and gain factor to the D/A conversion formula that board forevermore. Although this technique does not eliminate temperature and drift errors, it will eliminate the largest portion of the total error and provide more accurate measurements and better overall product performance.

When considering accuracy, it is important to consider the requirements of your application. In many applications, 16-bit resolution is better than the system requires for proper operation. For example if you are controlling the brightness of a lamp viewed by the human eye, a resolution of 8 bits (1/256) would probably be sufficient in most scenarios. D/A errors only need to be taken into account if they will have a material impact on the performance of your application.

To incorporate error measurements into the D/A code calculation, use this formula:

D/A code = max((Target voltage – Vmin) / (Vmax – Vmin) * 65535), 0)

Target voltage = the desired output voltage

Vmin is the voltage measured with D/A code 0

Vmax is the actual output voltage with D/A code 65535

Examples for 0-5V output range:
offset voltage = .003V, full-scale voltage = 5.012V, desired output voltage = 1.000V

D/A code = (1V - .003V) / (5.012V - .003V) * 65535 = **13044**

offset voltage = -.003V, full-scale voltage = 4.996V, desired output voltage = 1.000V

D/A code = (1V + .003V) / (4.996V - .003V) * 65535 = **13149**

Note: In this formula, the binary code scale value is 65535, not 65536. This is because the full scale voltage Vmax was measured with the actual output code of 65535.

Note: The output values are limited to voltages between the 0 code voltage and the 65535 code voltage. If the offset voltage is positive, you cannot output 0V.

Note: D/A codes are limited to positive values in the range 0-65535. Therefore, a positive offset voltage (output voltage is greater than zero when D/A code = 0) cannot be corrected, since correction would require a D/A code less than zero. This is why the formula above uses the max() function to force a minimum D/A code of zero.

8.5 Digital I/O Features

8.5.1 Overview

The 21 digital I/O lines are organized as 3 ports A, B, and C. Ports A, and B are 8 bits wide, and port C is 5 bits wide. All lines operate with 3.3V logic levels. All lines may be configured via software for 10K ohm pull-up or pull-down resistors. The pull resistors are organized in two groups: Group 1 consists of I/O lines 0-15, and Group 2 consists of I/O lines 16-20. The selected pull configuration is stored in nonvolatile memory and is automatically recalled during power-up.

All digital I/O lines utilize 1-bit logic buffers (transceivers) to provide enhanced output current capability and protect the FPGA from faulty connections. The directions for all lines are individually configurable. For safety and to prevent glitches, on power up or reset, all lines reset to input mode and all port data registers reset to all 0.

8.5.2 Edge Detection Circuit

An edge detection circuit is available on any single digital I/O line. The edge detection circuit can be used to notify the processor when a particular event occurs, such as a door opening, a switch being pressed, or a light curtain being penetrated.

The edge detection circuit on DS-MPE-DAQ0804 works by selecting the desired digital I/O line and the desired polarity. The programmer provides custom code to define the functionality to be performed whenever the specified edge occurs. When a qualifying edge occurs, the circuit will generate an interrupt. If the Diamond Universal Driver software is being used, the driver will respond to the interrupt and pass control to the user's custom code. After executing the application-specific functionality, the custom code must clear the interrupt request in order to be ready for the next event.

8.5.3 Support for Special Functions

The digital I/O lines may be defined as inputs or outputs for the 8 counter/timers as well as outputs for the 4 PWMs, as indicated below. Configuring these features is explained in the following sections and in the Diamond Universal Driver user manual.

DIO Bit	Alternate Signal
6	CTR 0 I/O
7	CTR 1 I/O
8	CTR 2 I/O
9	CTR 3 I/O
10	CTR 4 I/O
11	CTR 5 I/O
12	CTR 6 I/O
13	CTR 7 I/O
14	PWM 0 out
15	PWM 1 out
16	PWM 2 out
17	PWM 3 out
19	A/D external trigger
20	D/A waveform external trigger

8.6 Counter/Timer Features

8.6.1 Overview

DS-MPE-DAQ0804 provides 8 32-bit counter/timers with a wide array of features and programmability. Both up and down counting are supported, and the clock for each counter can be selected from an internal 50MHz or 1MHz clock or an external digital signal. In down counting mode, an optional programmable-width output pulse may be enabled each time the counter reaches zero. Counters can also be used to generate programmable interrupts, enabling custom code to be run at precise user-defined intervals.

8.6.2 Counter Commands

The counters are configured and managed with a series of commands. These commands are implemented in the Diamond Systems Universal Driver software. Commands may operate on either a single counter or any combination of counters simultaneously using a user-defined selection mask.

Command	Function
0	Clear one or more counters. If a counter is running at the time it is cleared, it will continue
1	Load counter with user-defined 32-bit value
2	Select count direction, up or down
4	Enable / disable counting
5	Latch counter; a counter must be latched before its count value can be read back. The latch is a snapshot of the counter value at one moment in time. The counter continues to run after latching.
6	Select clock source for one or more counters. Options include internal 50MHz clock, internal 10MHz clock, or the counter's designated I/O pin on port C.
7	Enable / disable auto-reload
8	Enable counter output pulse on designated I/O pin on port C

9	Configure counter output pulse width; options include 1, 10, 100, or 1000 clock periods
10	Read counter value. A counter must be latched with command 5 before its contents can be read back.
15	Reset one more counters. When a counter is reset its configuration and contents are lost.

8.6.3 Counter I/O Signals

Counter clock and output signals may be made available on digital I/O pins as defined below. A counter can use its designated I/O pin for either an external clock or its output signal, but not both at the same time.

DIO pin	Input	Output
DIO6	Counter 0 clock	Counter 0 output
DIO7	Counter 1 clock	Counter 1 output
DIO8	Counter 2 clock	Counter 2 output
DIO9	Counter 3 clock	Counter 3 output
DIO10	Counter 4 clock	Counter 4 output
DIO11	Counter 5 clock	Counter 5 output
DIO12	Counter 6 clock	Counter 6 output
DIO13	Counter 7 clock	Counter 7 output

8.6.4 Counter Advanced Features

The DAQ circuit uses counter/timers to control the A/D sample rate for high speed sampling as well as the D/A waveform generator data rate. These operations are managed by the Universal Driver software. Refer to the Universal Driver user manual for details on these advanced features.

8.7 Pulse Width Modulator Features

8.7.1 Overview

DS-MPE-DAQ0804 provides 4 24-bit pulse width modulator circuits (PWMs). These circuits can be programmed to produce an output square wave up to 25MHz with a duty cycle anywhere from 0% to 100% (these limits are of course DC signals). The output polarity is programmable. PWM outputs are available on digital I/O port C pins.

PWM operation is as follows: Each PWM contains a period counter and a duty cycle counter which are programmed for the desired values. When the PWM starts, both counters start to count down simultaneously, and the output pulse is set to the desired active polarity. When the duty cycle counter reaches 0, it stops, and the output changes to the inactive polarity. When the period counter reaches zero, both counters reload, the output is made active again, and the cycle repeats.

The PWM duty cycle can be updated in real time without having to stop the circuit from running.

8.7.2 PWM Commands

The PWMs are configured and managed with a series of commands. These commands are implemented in the Diamond Systems Universal Driver software. Configuration commands operate on a single PWM at a time, however start, stop, and reset commands may operate on a single PWM or all PWMs at the same time.

Command	Function
0	Stop one or all PWMs. When a PWM is stopped, its output returns to its inactive state, and the counters are reloaded with their initial values. If the PWM is subsequently restarted, it will start at the beginning of its waveform, i.e. the start of the active output pulse. Stopping a PWM is not the same as resetting it. See command 4 below.
1	Load the period or duty cycle counter with user-defined value
2	Select output pulse polarity
3	Enable / disable PWM output. This must be done in conjunction with command 5 below.
4	Reset one or all PWMs. When a PWM is reset, it stops running, and any DIO line assigned to that PWM for output is released to normal DIO operation. The direction of the DIO line will revert to its value prior to the PWM operation.
5	Enable / disable PWM output signal on designated Port C DIO pin.
6	Select clock source for period and duty cycle counters, either 50MHz or 1MHz. Both counters will use the same clock source.
7	Start one or all PWMs. All selected PWMs will start their active output pulses at the same time.

8.7.3 PWM Output Signals

When a PWM output is enabled with command 5, the corresponding DIO pin is forced to output mode regardless of its current configuration. To make the pulse appear on the output pin, command 3 must also be executed, otherwise the output will be held in inactive mode (the opposite of the selected polarity for the PWM output). When a PWM is reset, the corresponding digital I/O line returns to its previous direction and data status.

PWM signals are available on digital I/O pins as defined below.

DIO Bit	*Alternate Signal*
14	PWM 0 output
15	PWM 1 output
16	PWM 2 output
17	PWM 3 output

8.8 Interrupt Operation

8.8.1 Overview

Interrupts provide a means for a circuit to request service from the processor autonomously without requiring the application software to continuously poll the circuit. When the circuit requires attention, it generates an interrupt request signal to the processor. The processor responds by passing program control to a dedicated software routine, which provides the functionality required by the circuit.

The maximum interrupt rate is not a fixed value but varies with the operating system and the duration that the interrupt service routine runs. The maximum rate must be determined by trial and error. However a general guideline is that for Linux and Windows the rate should not exceed 1KHz. Because all interrupts require some processing overhead, the lower the rate the better. However, in the case of A/D sampling, a slower interrupt rate also delays the availability of new data. The programmer must find the ideal balance between processor load and response time.

8.8.2 Interrupt Sources

Interrupts can come from several sources in the data acquisition circuit. A status register is used to indicate which circuit or circuits are requesting interrupt service, so the application software can respond accordingly.

A/D converter: For high speed sampling, the data acquisition circuit supports the use of interrupts to pass data back to the processor in large blocks instead of one sample at a time. This greatly reduces the processor time required to handle the data flow. A programmable FIFO holds the A/D data and generates an interrupt request when the number of samples in the FIFO reaches a user-specified threshold. This is described more fully in the data acquisition section above.

Counter/timers: Counters 2 and 3 can be used to generate interrupts at programmable frequencies. Generally only one or the other counter is used for interrupts, however it is possible to use both simultaneously.

Edge detection circuit: As described above, digital I/O port B can be used to monitor the state of up to 8 digital inputs and generate an interrupt request when a specific transition is seen on any of its 8 inputs.

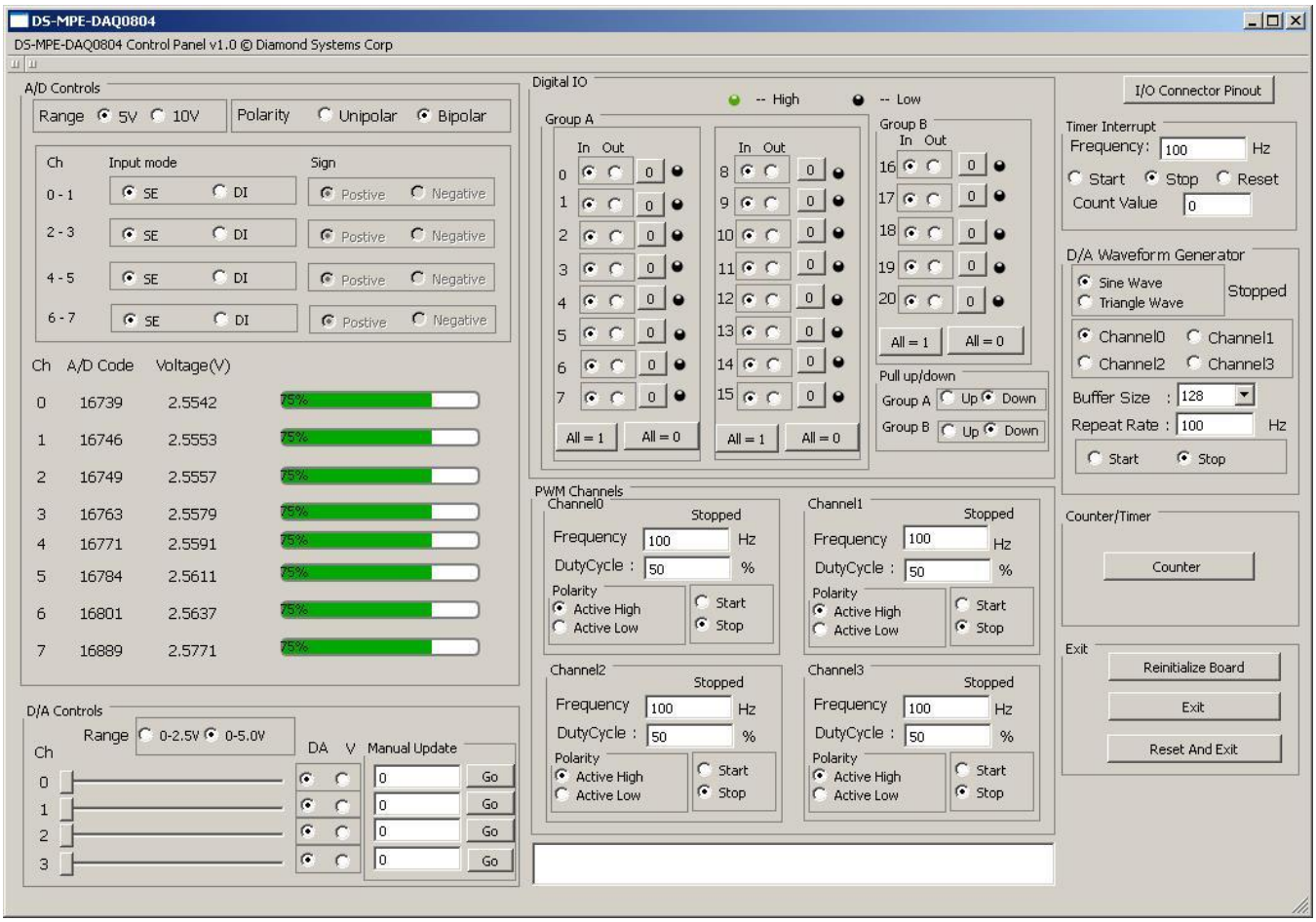
9. SOFTWARE DRIVER OVERVIEW

The DS-MPE-DAQ0804 module is configured by software. The board must first be initialized and then configured. These operations can be done either using Diamond's Universal Driver (version 7.0 or higher) or by an independent set of equivalent register operations. Please refer to the DS-MPE-DAQ0804 Control Panel Manual and DS-MPE-DAQ0804 Universal Driver Software User Manual for additional information.

9.1 Configuring Using Universal Driver

Diamond Systems' provides a device driver which will enable access to the board functionalities via an easy to use API set. This driver is called the Universal Driver and is available in Windows XP and Linux 2.6.xx operating systems. The details on the Universal driver can be found in the Universal Driver manual and can be accessed online at http://docs.diamondsystems.com/dscud/manual_Main+Page.html. The Universal Driver software comes on the Diamond System' Resource CD shipped with this product, or may be downloaded from the DS-MPE-DAQ webpage at http://www.diamondsystems.com/products/dsmpedaq0804.

The main screen of the graphical user interface is shown below.



9.2 Configuring Using Register Operations

The board can also be controlled using simple register read/write commands if you write your own driver. In typical modern operating systems, the user level applications cannot directly access the low level system information and don't have register level access. In order to communicate with any PCI device, a device driver is required.

The Universal Driver software can also be used to do register-level control, and a programmer can develop their own driver functionality that uses simple register read/write command after performing a PCI scan using the Universal Driver. Users of this type of access need to understand the board register map. This type of approach is suitable for someone who is very aware of the nature of low-level operations of hardware.

9.2.1 Interrupt level

Interrupts are used for hardware I/O operations that are independent of normal program flow. The DS-MPE-DAQ0804 can be set up to generate interrupts under several circumstances. The board can generate interrupts to transfer digital data into the board, as well as at regular intervals according to a programmable timer on the board. Individual control bits are used to enable each type of interrupt.

Since the DS-MPE-DAQ0804 board works on PCI Express bus architecture, the interrupt level is obtained as a result of a PCI scan performed by the device driver. To obtain the interrupt level used by the board, Diamond provides a default device driver, WinDriver, which can perform low level PCI commands and provide user level access to the board.

If you do not wish to use this driver and would like to develop your own driver, you need to be knowledgeable on the PCI / PCI express system architecture as well as the device driver model and architecture details for your chosen operating system.

10. SPECIFICATIONS

Analog inputs	8 16-bit single ended, 4 16-bit differential
Sample rate	100KHz maximum aggregate
Input ranges	+/-10V, +/-5V, 0-10V, 0-5V
A/D FIFO	2048 samples with programmable threshold
Input overvoltage protection	+/-25V
A/D clocking	Internal counter / timer, software command, or external clock
Analog input error	+/-28LSB (+/-0.04%) max zero scale error (no offset adjustment)
Analog outputs	4 16-bit
Output ranges	0-5V, 0-2.5V
Output updates	Simultaneous or individual channel
Output current	Up to 5mA per channel (1Kohm minimum load)
Output drift	Low drift 2ppm/^{o}C internal reference
Output error	+/-1.5mV (+/-0.03%) max zero scale error (no offset adjustment)
Analog calibration	Factory calibration for full-scale
Digital I/O	14 lines with 1-bit buffers for bit-by-bit individual direction control
DIO output voltage	+3.3V
DIO pull-up / pull-down	Software selectable
Pulse width modulators	4 24-bit, 0-100% duty cycle
Counter / timers	8 32-bit programmable
Input power	+3.3VDC +/-5%
Power consumption	0.462W @ 3.3V
Software drivers	Windows Embedded Standard 7, XP Linux 3.2.x
Universal Driver	Support for all functions
Operating temperature	-40°C to +85°C
Operating humidity	5% to 95% non-condensing
MTBF	xxx hours
Dimensions	50.95mm x 30mm (2” x 1.18”)
Weight	8.5g (0.3oz)
RoHS Compliant	Yes