Register Map for the PCIM-DAS16JR/16

MEASUREMENT
COMPUTING™

Document Revision 1.0, September, 2003

Register Description

Overview

PCIM-DAS16JR/16 operation registers are mapped into I/O space. This board has several base address regions, each corresponding to a reserved block of addresses in I/O space.

Only experienced programmers should attempt register-level programming. As an alternative to register level programming, the PCIM-DAS16JR/16 is fully supported by the optional Universal Library software, as well as most high-level data acquisition and control application packages (for example, SoftWIRE., HP Vee or Labtech Notebook.)

Of the six Base Address Regions (BADRs) available in the industry standard PCI 2.1 specification, four are implemented in this design and are summarized in Table 1.

Table 1. PCIM-DAS16JR/16 BADR Mapping

I/O Region	Function	Operations
BADR0	PCI memory mapped configuration registers	32-bit Double Word
BADR1	PCI I/O mapped configuration registers	32-bit Double Word
BADR2	ADC and DAC data registers	16-bit Word
BADR3	Pacer, Counter, Trigger, Interrupt, and Digital I/O configuration registers	8-bit Byte

BADR*n* will likely be different on different machines. These Base Address values are assigned by the PCI BIOS, and cannot be guaranteed to be the same, even on subsequent power-on cycles of the same machine. All software must interrogate BADR0 at run-time with a READ_CONFIGURATION_DWORD instruction to determine the BADR*n* values.

BARD0 and BADR1 Registers

The PCIM-DAS16JR/16 board uses the PLX PCI9030 PCI Bus Interface chip. BADR0 and BADR1 are used for PCI configuration. Only the PCI Interrupt Control/Status Register (BADR1+4Ch) should be used. All other registers should not be written to. For more information on these registers, refer to the *PCI 9030 Data Book* (May, 2002), available from PLX Technology® at www.plxtech.com/products/9030/default.htm.

Note: 0 indicates a read operation. All unused bits are denoted by an X. Actual writes to these bit positions are "don't care", meaning that it does not matter what data is written to these bits.

BARD1 Register

BADR1 is used for PCI configuration. Only the PCI Interrupt Control/Status Register (BADR1+4Ch) should be used. All others should not be written to.

Register	Read Function	Write Function
BADR1 + 4Ch	Interrupt Status	Interrupt Control

9030 Interrupt Status & Control

BADR1 + 4Ch

Read/Write

31:24	23:16
X	X

15	14	13	12	11	10	9	8
0	0	0	0	0	INTCLR	0	INTSEL

7	6	5	4	3	2	1	0
X	PCIINT	1	0	0	INT	INTPOL	INTE

This register is 32-bits long, and controls the interrupt features of the PLX PCI 9030. For proper operation, do not change the predefined bits (bits 3, 4, 9, and 11 to 15 = 0, and bit 5 = 1). The rest of the register's bits have specific control functions.

If you write to the Interrupt Status register, be sure to read the current status. Only change the bits you want, and then write the new status word. X indicates "don't care", which means that it does not matter what data is written to these bits.

INTE Local Interrupt Enable:

0 = disabled

1 = enabled (default)

INTPOL Interrupt Polarity:

0 = active low (default)

1 = active high

INT Interrupt Status:

0 = interrupt is not active

1 = interrupt is active

PCIINT PCI Interrupt Enable:

0 = disabled

1 = enabled (default)

INTSEL Interrupt Edge/Level Select:

0 = level-sensitive (default)

1 = edge-sensitive

Note that this bit only has an effect when the INTPOL bit is set to a high-polarity mode (INTPOL=1). The interrupt defaults to level-sensitive when the INTPOL bit is set to low-polarity (INTPOL=0).

INTCLR Clears the PCI INT interrupt:

0 = no effect

1 = clear the PCI INT interrupt

You must set both PCIINT and INTE to 1 to enable interrupts. There is also an interrupt enable bit (INTE) in BADR3 + 4 that you must set to 1 to enable interrupts.

This register is only used to enable the local and PCI interrupt bits, so the interrupt generated by the on board logic can propagate through the PLX PCI9030 interface to the PCI bus INTA. The board has both edge and level sensitive interrupts. The edge-sensitive interrupts, EndOfAcquisition, EndOfBurst, and EndOfConversion must be cleared by writing a 0 to the INT bit in BADR3+4 at the end of your interrupt service routine. The level-sensitive interrupts, FifoHalfFull and FifoNotEmpty, will be regenerated after you service the interrupt if their condition is still true. Refer to the section on BADR3 + 4 for more details.

BADR2 Registers

The I/O Region defined by BADR2 contains the 16-bit ADC data and the two 12-bit DAC data registers.

Register	Read Function	Write Function
BADR2 + 0	ADC Data	Begin single conversion

ADC Data/Convert

BADR2 + 0

Read

15	14	13	12	11	10	9	8
AD15	AD14	AD13	AD12	AD11	AD10	AD9	AD8
MSB							

7	6	5	4	3	2	1	0
AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
							LSB

AD[15:0] This register contains the current ADC data word. Data format is dependent upon offset mode:

Bipolar Mode: Offset Binary Coding

0000 h = –FS

7FFFh = Mid-scale (0V)

FFFFh = +FS – 1LSB

Unipolar Mode: Straight Binary Coding

0000 h = –FS (0V)

7FFFh = Mid-scale (+FS/2)

FFFFh = +FS – 1LSB

Write

Writing to this register is only valid for SW initiated conversions. The ADC Pacer source must be set to software polled (see BADR3 + 5). A null write to BADR2 + 0 will begin a single conversion. Conversion status may be determined by polling the EOC bit in BADR3 + 2.

BADR3 Registers

The I/O Region defined by BADR3 contains the data and control registers for the ADC pacer, counter, trigger, interrupt, and high-drive digital I/O bytes. This region supports 8-bit BYTE operations.

REGISTER	READ FUNCTION	WRITE FUNCTION
BADR3 + 0	MUX scan limits	MUX scan limits
BADR3 + 1	Main Connector Digital Inputs	Main Connector Digital Outputs
BADR3 + 2	ADC Channel Status and Switch Settings	
BADR3 + 3	ADC Conversion Status	
BADR3 + 4	Interrupt Settings /Status	Interrupt Control
BADR3 + 5	A/D Pacer Clock Settings	A/D Pacer Clock Control
BADR3 + 6	Burst Mode and Converter Settings	Burst Mode and Converter Control
BADR3 + 7	Programmable Gain Settings	Programmable Gain Control
BADR3 + 8	82C54 Counter 1 Data	82C54 Counter 1 Data
BADR3 + 9	82C54 Counter 2 Data	82C54 Counter 2 Data
BADR3 + 0Ah	82C54 Counter 3 Data	82C54 Counter 3 Data
BADR3 + 0Bh	82C54 Counter Control Data	82C54 Counter Control Data
BADR3 + 0Ch	User Counter Clock Setting	User Counter Clock Control
BADR3 + 0Dh		Residual Counter upper 2 bits
BADR3 + 0Eh		Residual Counter lower byte

MUX Scan Limits Register

BADR3 + 0

Read/Write

7	6	5	4	3	2	1	0
CH H3	CH H2	CH H1	CH H0	CH L3	CH L2	CH L1	CH L0

Read

The current channel scan limits are read as one byte. The high channel number scan limit is in the **most** significant four bits. The low channel scan limit is in the **least** significant four bits.

Write

The channel scan limits desired are written as one byte. The high channel number scan limit is in the **most** significant four bits. The low channel scan limit is in the **least** significant four bits.

Note: Every write to this register sets the current A/D channel MUX setting to the number in bits 0-3, and resets the FIFO. You should delay 10 µs after setting the MUX (to allow for settling time) before initiating a conversion.

Main Connector Digital Input and Digital Output Registers

BADR3 + 1

Read

7	6	5	4	3	2	1	0
1	1	1	1	DI3	DI2, CTR0 GATE	DI1	DI0, EXT TRIG, EXT PACER, EXT GATE

The signals present at the inputs are read as one byte. The most significant 4 bits (bit 4 – bit 7) are always 1. Digital inputs 2 and 0 have multiple functions: Digital input 2 may also be used as the gate to Counter 1 of the 82C54, which is available on the Main connector. Refer to BADR3 + 6 for a more detailed description. Digital input 0 may also be used as a trigger, a pacer, or a gate for the ADC. Refer to BADR3 + 5 for more details.

Write

7	6	5	4	3	2	1	0
X	X	X	X	DO3	DO2	DO1	DO0

The upper four bits are ignored. The lower four bits are latched TTL outputs. Once written, the state of the inputs cannot be read back, because a read-back would read the separate digital input lines (see above).

Note: Digital lines 0-3 and analog connector pins 3, 4, 5, 6, 22, 23, 24, and 25 should not be used as ON/OFF digital I/O.

The digital inputs have multiple functions, as described above. The digital outputs are also used by the CIO-EXP32 32 channel analog multiplexer/amplifier. We suggest that the 4-bit ports that are available at the board's main I/O connector be kept free for analog multiplexing control lines.

ADC Channel Status and Switch Settings Registers

BADR3 + 2

Read Only

7	6	5	4	3	2	1	0
EOC	U/B	MUX	CLK	MA3	MA2	MA1	MA0

EOC 1 = the A/D converter is busy.

0 = the A/D converter is free.

EOC is in both BADR3 + 2 and BADR3 + 3 for convenience in software programming.

U/B 1 = the Analog Input Polarity Switch is set to Unipolar.

0 = the Analog Input Polarity Switch is set to Bipolar.

MUX 1 = the Analog Input Mode Switch is set to 16 single-ended.

0 = the Analog Input Mode Switch is set to 8 differential.

CLK 1 = the Pacer Clock jumper is set to 10 MHz.

0 = the Pacer Clock jumper is set to 1 MHz.

MA3, MA2, MA1, and MA0 are binary numbers between 0 and 15 that indicate the MUX channel currently selected, and are valid only when EOC = 0. The channel MUX increments shortly after EOC = 1, and may be in a state of transition when EOC = 1.

ADC Conversion Status Register

BADR3 + 3

Read Only

7	6	5	4	3	2	1	0
EOC	EOB	EOA	FNE	FHF	OVERRUN	0	0

EOC 1 = the A/D converter is busy.

0 = the A/D converter is free.

EOC is in both BADR3 + 2 and BADR3 + 3 for convenience in software programming.

EOB 1 = an ADC Burst has been completed.

0 = an ADC Burst is in progress or has not started.

EOA 1 = the residual # of samples have been written to the FIFO.

0= the residual # of samples have not been written to the FIFO.

EOA is cleared by writing a 0 to the INT bit in BADR3 + 4.

EOA is in both BADR3 + 3 and BADR3 + 4 for convenience in software programming.

FNE 1 = FIFO memory contains at least one sample.

0 = FIFO memory contains no samples.

FHF 1 = FIFO memory contains at least 512 samples.

0 = FIFO memory contains less than 512 samples.

OVERRUN 1 = FIFO memory has overrun.

0 = FIFO memory has not overrun.

OVERRUN is in both BADR3 + 3 and BADR3 + 4 for convenience in software programming.

Interrupt Status and Control

BADR3 + 4

Read/Write

7	6	5	4	3	2	1	0
INTE	INT	X	OVERRUN	EOA	EOA_INT_SEL	INTSEL1	INTSEL0

INTSEL[1:0] Used to select the source of the interrupt. With the exception of EOA, end of acquisition, you can only select one interrupt source.

INTSEL1	**INTSEL0**	**INTERRUPT SOURCE**
0	0	EOC, End of Conversion
0	1	FIFO not empty
1	0	EOB, End of Burst
1	1	FIFO half full/EOA

EOA_INT_SEL 1 = Interrupt on end of acquisition

0 = No interrupt on end of acquisition

EOA_INT_SEL is used in conjunction with the residual counter. See BADR3 + 0Dh.

EOA 1 = the residual # of samples have been written to the FIFO

0 = the residual # of samples have not been written to the FIFO

EOA is cleared by writing a 0 to the INT bit. Refer below to INT.

EOA is in both BADR3 + 3 and BADR3 + 4 for convenience in software programming.

OVERRUN 1 = FIFO memory has overrun

0 = FIFO memory has not overrun

OVERRUN is in both BADR3 + 3 and BADR3 + 4 for convenience in software programming.

INT 1 = Interrupt generated

0 = No interrupt generated

INT must be cleared after each edge sensitive interrupt (EOC, EOB, and EOA) by setting it to 0.

INTE 1 = Interrupts are enabled.

0 = Interrupts are disabled.

To enable interrupts, you must also set bits in BADR1 + 4Ch.

A/D Pacer Clock Status and Control

BADR3 + 5

Read/Write

7	6	5	4	3	2	1	0
X	GATE_STATUS	GATE_POL	GATE_LATCH	GATE_EN	EXT_PACER_POL	PS1	PS0

PS[1:0] Control the source of the A/D Pacing according to the values in the following table:

PS1	**PS0**	
0	X	Software polled A/D
1	0	External Pacer Clock (digital input 0, pin 25)
1	1	Internal Pacer Clock (CTR 2 OUT, no external access)

EXT_PACER_POL 1 = the external pacer polarity is set to negative edge for non burst mode and burst mode.

0 = the external pacer polarity is set to positive edge for non-burst mode and burst mode.

This bit is only used when the external pacer clock is selected. We recommend setting this bit to positive edge.

The remaining bits are only used when the internal pacer is selected.

Note: The trigger edge select hardware jumper, P8 (labeled **TRIG** on the board), selects whether the rising edge or falling edge of the pulse from the internal pacer is used to pace acquisitions. We recommended that this jumper be set to a positive-going rising edge (the "**R"** position on the board).

GATE_EN 1 = the gate to the internal pacer is always on regardless of the signal on pin 25. In this mode, the bits below are ignored.

0 = the gate to the internal pacer is controlled by the signal on pin 25.

GATE_ LATCH 1 = the signal on pin 25 will act as an edge trigger to the internal pacer. It is latched in hardware. Software must clear latch by writing a "0" to the GATE_STATUS bit.

0 = the signal on pin 25 will act as a level gate to the internal pacer.

GATE_POL 1 = the trigger / gate polarity is set to a negative-going edge / low level for non burst mode, and a positive-going edge / high level for burst mode.

0 = the trigger / gate polarity is set to a positive-going edge / high level for non burst mode, and a negative-going edge / low level for burst mode.

on a read: GATE_STATUS = 1, the gate to the internal pacer is on.

GATE_STATUS = 0, the gate to the internal pacer is off.

on a write: GATE_STATUS = 0 clears the hardware latch when LATCH = 1

Burst mode and Converter Control

BADR3 + 6

Read/Write

7	6	5	4	3	2	1	0
X	X	X	X	X	X	BME	CONV_EN

CONV_EN 1 = conversions are enabled

0 = conversions are disabled

BME 1 = bursting is enabled. When burst mode is enabled, the mux channel select bits in BADR3 + 0 are used to specify the channels in the burst.

0 = bursting is disabled

The burst mode generator is a clock signal that paces the A/D at the maximum multi-channel sample rate, and then periodically performs additional maximum rate scans. In this way, the channel to channel skew (time between successive samples in a scan) is minimized without taking a large number of undesired samples (Figure 1).

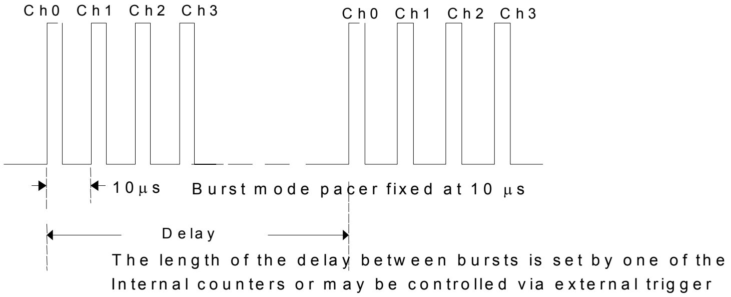
Ch0 Ch1 Ch2 Ch3
Ch0 Ch1 Ch2 Ch3
10µs
Burst mode pacer fixed at 10 µs
Delay
The length of the delay between bursts is set by one of the
Internal counters or may be controlled via external trigger

Figure 1. Burst Mode Timing

The PCIM-DAS16Jr/16 burst mode generator takes advantage of the fast A/D. The burst mode skew is 10 µs between channels for the PCIM-DAS16Jr/16. For the CIO-DAS1602/16, the burst mode skew is 13.3 µs between channels.

Programmable Gain Control Register

BADR3 + 7

Read/Write

7	6	5	4	3	2	1	0
X	X	X	X	X	UNI/BIP	G1	G0

G[1:0] Control the gain of the programmable gain amplifier according to the table below.

UNI/BIP Controls the range of the input according to the table below; 0 = Unipolar, 1 = Bipolar

UNI/BIP	G1	G0	Range
1	0	0	±10V
1	0	1	±5V
1	1	0	±2.5V
1	1	1	±1.25V
0	0	0	0 to 10V
0	0	1	0 to 5V
0	1	0	0 to 2.5V
0	1	1	0 to 1.25V

8254 Counter 1 Data - User Counter

BADR3 + 8

Read/Write

7	6	5	4	3	2	1	0
D8	D7	D6	D5	D4	D3	D2	D1

The 82C54 counter 1 is available to you as a generic counter/timer. The clock, gate and output are all available at the main 37 pin connector. Refer to BADR3 + 0C HEX for clock options.

8254 Counter 2 Data - ADC Pacer Lower Counter

BADR3 + 9

Read/Write

7	6	5	4	3	2	1	0
D8	D7	D6	D5	D4	D3	D2	D1

82C54 Counter 3 Data - ADC Pacer Upper Counter

BADR3 + 0Ah

Read/Write

7	6	5	4	3	2	1	0
D8	D7	D6	D5	D4	D3	D2	D1

Counters 2 and 3 are configured in hardware to produce a 32-bit counter for use as a pacer for the A/D converter.

82C54 Counter Control

BADR3 + 0Bh

Read/Write

7	6	5	4	3	2	1	0
D8	D7	D6	D5	D4	D3	D2	D1

This register controls the operation and loading/reading of the counters. The four 82C54 registers may be written to and read from. The operation of the 82C54 is explained in the 82C54 programmable interval timer data sheet. This data sheet is available on our web site at www.measurementcomputing.com/PDFmanuals/82C54.pdf.

User Counter Clock Control

BADR3 + 0Ch

Read/Write

7	6	5	4	3	2	1	0
X	X	X	X	X	X	X	CTR1_CLK_SEL

CTR1 _CLK_SEL 1 = the onboard 100 kHz clock signal is ANDed with the COUNTER 1 CLOCK INPUT (pin 21). A high on pin 21 will allow pulses from the onboard source into the 8254 Counter 1 input. (This input has a pull-up resistor on it, so no connection is necessary to use the onboard 100 kHz clock.)

0 = the input to 8254 Counter 1 is entirely dependent on pulses at pin 21, COUNTER 1 CLOCK INPUT.

Residual Sample Counter Registers

BADR3 + 0Dh

Read/Write

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

BADR3 + 0Eh

Read/Write

7	6	5	4	3	2	1	0
X	X	X	X	X	X	D9	D8

The residual count, data bits D9:D0 are used to specify the number of samples at the end of a paced acquisition that will be collected before the EOA (end of acquisition) interrupt is generated. This is useful when the total number of samples is not a multiple of half the FIFO size (512) or the total number of samples is less than the FIFO size (1024).

Always write the residual count before setting the EOA_INT_SEL bit. Writing to either register will reset the counter with the new values. You must write the values each acquisition even if they have not changed. Perform the following procedures for correct operation.

Total number of samples is less than 512

1. Before you start the acquisition, write the total number of samples to the residual counter, an 87h to BADR3+4 (INTE, EOA_INT_SEL, and FIFO_HALF FULL enabled), and a 67h to BADR1+4Ch (INTE and PCINTE enabled).
2. Start the acquisition.
3. The first interrupt you get will be the EOA interrupt. First clear the EOA_INT_SEL bit (bit 2 BADR3 + 4), then read 20 samples from FIFO. The last thing you should do in your interrupt service routine is to clear the INT bit (bit 6, BADR3 + 4) and disable interrupts by writing a “0” to the INTE bit (bit 7, BADR3 + 4).

Example: 20 total samples

1. Before you start the acquisition, write a 20 to the residual counter, an 87h to BADR3 + 4, and a 67h to BADR1 + 4Ch.
2. Start the acquisition.
3. You will get the EOA interrupt. Write a 03h to BADR3 + 4, read 20 samples from FIFO, and then write another 03h to BADR3 + 4.

Total number of samples is greater than 512, but less than 1024

1. Before you start the acquisition, write the total number of samples to the residual counter, an 87h to BADR3 + 4 (INTE, EOA_INT_SEL, and FIFO_HALF FULL enabled), and a 67h to BADR1 + 4Ch (INTE and PCINTE enabled).
2. Start the acquisition.
3. The first interrupt you get will be the FIFO_HALF FULL interrupt. Read 512 samples from FIFO and clear the INT bit (bit 6, BADR3 + 4).
4. The second interrupt you get will be the EOA interrupt. First clear the EOA_INT_SEL bit (bit 2 BADR3 + 4). Then read the total number of samples, less 512 from FIFO. Do not try to read the entire residual count on the EOA interrupt. You already retrieved 512 of the residual on the FIFO_HALF FULL interrupt in step 3. The last thing you should do in your interrupt service routine is to clear the INT bit (bit 6, BADR3 + 4) and disable interrupts by writing a 0 to the INTE bit (bit 7, BADR3 + 4).

Example: 1000 total samples

1. Before you start the acquisition, write a 1000 to the residual counter, an 87h to BADR3 + 4, and a 67h to BADR1 + 4Ch.
2. Start the acquisition.
3. You will get a FIFO_HALF FULL interrupt. Read 512 samples from FIFO and write an 87h to BADR3 + 4.
4. You will get the EOA interrupt. Write a 03h to BADR3 + 4, read 488 samples from FIFO, and then write another 03h to BADR3 + 4.

Total number of samples is greater than 1024

1. Before you start the acquisition, write the residual number of samples to the residual counter, an 83h to BADR3 + 4 (INTE and FIFO_HALF FULL enabled), and a 67h to BADR1+ 4Ch (INTE and PCINTE enabled). The residual number of samples is the remainder of the total number of samples divided by 512.
2. Start the acquisition.
3. The first interrupt you get will be the FIFO_HALF FULL interrupt. Read 512 samples from FIFO and clear the INT bit (bit 6, BADR3 + 4).
4. Depending on the total number of samples, you will get some number of FIFO_HALF FULL interrupts. For all but the second to last sample, repeat step 3. On the second to last one, at the very end of your interrupt service routine, you must enable the EOA_INT_SEL bit by writing a 1 to bit two of BADR3 + 4. Be sure to enable EOA_SEL_INT after you have read the FIFO because the next FIFO_HALF FULL is what triggers the residual counter to start counting.
5. After the second to last interrupt, the next interrupt you get will be a FIFO_HALF FULL interrupt. Read 512 samples from FIFO and clear the INT bit (bit 6, BADR3 + 4).

6. The next interrupt after that will be the EOA interrupt. First clear the EOA_INT_SEL bit (bit 2, BADR3 + 4). Then read the residual count from FIFO. The last thing you should do in your interrupt service routine is to clear the INT bit (bit 6, BADR3 + 4) and disable interrupts by writing a 0 to the INTE bit (bit 7, BADR3 + 4).

Example: 1537 total samples

1. Before you start the acquisition, write a 1 to the residual counter (1537 / 512 = 3, a remainder of 1), an 83h to BADR3 + 4, and a 67h to BADR1 + 4Ch.

2. Start the acquisition. You will get a FIFO_HALF FULL interrupt. Read 512 samples from FIFO and write an 83h to BADR3 + 4.

3. You will get another FIFO_HALF FULL interrupt. This is the second to last FIFO_HALF FULL interrupt. First read another 512 samples from FIFO, and then write an 87h to BADR3 + 4.

4. You will get a third and final FIFO_HALF FULL interrupt. Read 512 samples from FIFO and write an 87h to BADR3 + 4.

5. You will get the EOA interrupt. Write a 03h to BADR3 + 4, read 1 sample from FIFO, and then write another 03h to BADR3 + 4.

Measurement Computing Corporation
10 Commerce Way
Suite 1008
Norton, Massachusetts 02766
(508) 946-5100
Fax: (508) 946-9500
E-mail: info@mccdaq.com
www.mccdaq.com