

OPERATIONS MANUAL

PCM-MIO-G

NOTE: *This manual has been designed and created for use as part of the WinSystems' Technical Manuals CD and/or the WinSystems' website. If this manual or any portion of the manual is downloaded, copied or emailed, the links to additional information (i.e. software, cable drawings) will be inoperable.*

WinSystems reserves the right to make changes in the circuitry and specifications at any time without notice.
©Copyright 2008 by WinSystems. All Rights Reserved.

REVISION HISTORY

P/N 400-0324-000

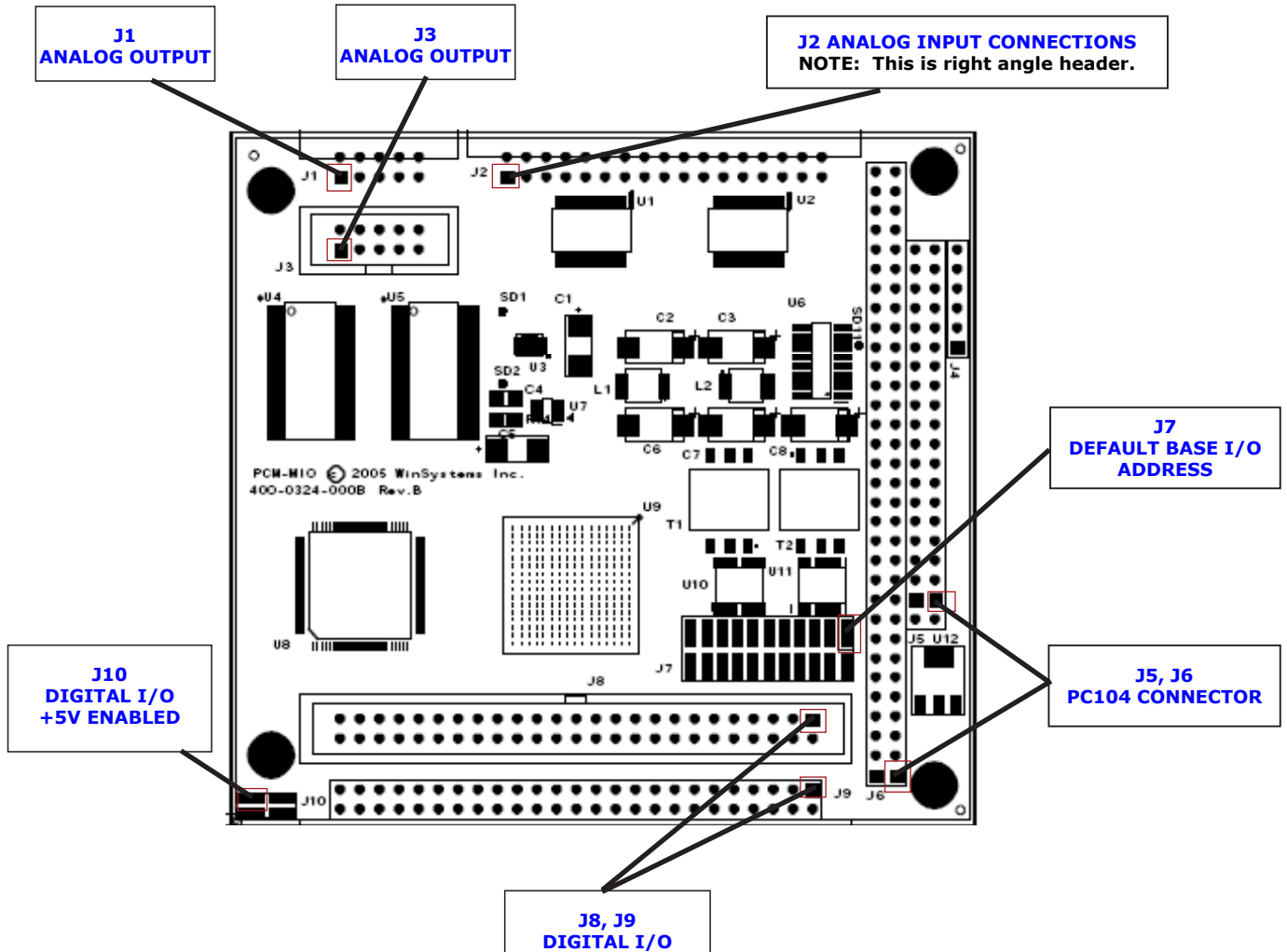
ECO Number	Date Code	Rev Level
ORIGINATED	060103	Preliminary
	061025	A
07-66	070904	B
07-151	080108	B.1
08-47	080409	B.2

Table of Contents

Visual Index – Quick Reference	i
Introduction	1
General Information	1
Features	1
Analog Input	1
Analog Output	2
Digital I/O	2
Functional Capability	3
Default Base I/O Address	3
Interrupts	3
DMA Support	4
A/D Section	4
D/A Section	5
Parallel I/O	5
VCC Enable	5
Parallel I/O Connectors	6
PC/104 Bus Interface	7
Software Summary	8
A/D Converters	8
D/A Converters	13
Parallel I/O	19
Software Drivers & Examples	21
Mechanical Drawing	22
Jumper Reference	23
Specifications	24
WARRANTY REPAIR INFORMATION	25

Visual Index – Quick Reference

For the convenience of the user, a copy of the Visual Index has been provided with direct links to connector and jumper configuration data.



NOTE: The reference line to each component part has been drawn to Pin 1, where applicable. Pin1 is also highlighted with a red square, where applicable.

Introduction

This manual is intended to provide the necessary information regarding configuration and usage of the PCM-MIO-G board. WinSystems maintains a Technical Support Group to help answer questions regarding usage and programming of the board. For answers to questions not adequately addressed in this manual, contact Technical Support at (817) 274-7553 between 8AM and 5PM Central Standard Time.

General Information

The PCM-MIO-G is a versatile, PC/104-based analog input, analog output, and digital I/O board designed to meet customer demands for high-accuracy and high channel count analog and digital I/O. The board is based upon Linear Technologies' state of the art precision converters and voltage references which require no external calibration.

The PCM-MIO-G operates over the industrial temperature range of -40°C to +85°C.

The board is pin out compatible with three other WinSystems' PC/104 analog conversion modules: PCM-ADIO, PCM-A/D16 (includes PCM-A/D12), and PCM-D/A12.

Features

- Multifunction Analog and Digital I/O Module
- The standard configuration includes 16-bit A/D, 12-bit D/A, 48 DIO
- Special OEM configurations are available for other analog and digital I/O combinations.
- No adjustment potentiometers or calibration needed.
- Software programmable interrupt configuration.
- Standard 0.100" headers for easy cable access.
- I/O pinout compatible with WinSystems PCM-ADIO, PCM-A/D16 (PCM-A/D12), and PCM-D/A12.
- Free software drivers in C, Windows and Linux.
- Operating temperature: -40°C to +85°C
- +5VDC operation
- Small size: 3.6" x 3.8" (90 mm x 96 mm)

Analog Input

- 16-bit Analog to Digital (A/D) converter with sample-and-hold circuit.
- Conversion speed: 100k samples per second.
- Analog Input Ranges: 0-5V, 0-10V, $\pm 5V$, and $\pm 10V$.
- Any combination of up to 16 single ended input channels and up to 8 differential input channels.
- $\pm 25V$ input protection on each channel.
- Each channel independently software programmable for input type and range
- No missing codes over full range.
- Low noise DC/DC converter.
- High-precision, low drift reference.
- No adjustment potentiometers or calibration needed.
- DMA or interrupt I/O supported.
- Supports industry standard signal conditioners.

Analog Output

- Eight, 12-bit Digital to Analog (D/A) converters. (Optional 16-bit D/A available for OEM's).
- Output Ranges: 0-5V, 0-10V, $\pm 5V$, $\pm 10V$.
- Each individual channel independently software programmable.
- Output channels can be updated and cleared individually or simultaneously.
- No adjustment potentiometers or calibration needed.
- Interrupt I/O supported.
- Supports industry standard signal conditioners.

Digital I/O

- 48 bi-directional TTL-compatible digital I/O lines with 24 capable of event sense interrupt generation.
- 12 mA sink current per line.
- Pinout compatible with industry-standard, optically isolated, digital I/O racks.



Functional Capability

Default Base I/O Address

The PCM-MIO-G is I/O mapped and requires 32 sequential port addresses. The base address is jumper selectable at **J7**. Care should be taken to choose an I/O area that does not conflict with other resources in the system. The specific device locations and register offsets are discussed in more detail in the [Software Summary](#) section.

J7

A5	1 0 0 2
A6	3 0 0 4
A7	5 0 0 6
A8	7 0 0 8
A9	9 0 0 10
A10	11 0 0 12
A11	13 0 0 14
A12	15 0 0 16
A13	17 0 0 18
A14	19 0 0 20
A15	21 0 0 22

EXAMPLE: 320HEX

0	0	0	0	0	0	1	1	0	0	1	X	X	X	X	X
A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0

Jumper OPEN = 1
Jumper INSTALLED = 0

Interrupts

The PCM-MIO-G provides flexible interrupt configuration options. Each A/D converter, each D/A converter and 24 Digital I/O are capable of generating an interrupt. They can be setup to use individual interrupts, a single shared interrupt, or any combination of the two. The interrupts are completely software configurable and require no jumpers or other configuration. The individual registers and configuration for each device are discussed in the [Software Summary](#) section and under each device.

The PCM-MIO-G can be configured to use IRQ's 3, 4, 5, 6, 7, 9, 10, 11, 12, 14, or 15 depending on availability in the system. IRQ's 0, 1, 2, 8, and 13 are not supported.

DMA Support

The PCM-MIO-G supports DMA on the A/D and D/A devices. Commands can be configured on DMA channels 0, 1, 2, and 3, which are 8 bit channels. Data can be configured on DMA channels 5, 6, and 7, which are 16 bit channels.

DMA Channel 4 is not available.

A/D Section

The PCM-MIO-G analog-to-digital conversion inputs are implemented by using two, 16-bit Linear Technologies LTC-1859CG devices. The board supports up to 16 single ended input channels, 8 differential input channels or various combinations of both. The channel configuration word selects whether an input will operate in single ended or differential mode and also selects the desired input range (0V to 5V, 0V to 10V, $\pm 5V$ and $\pm 10V$).

The LTC1859CG uses a successive approximation algorithm and an internal sample-and-hold circuit to convert an analog input signal to 16-bit digital data. The output is two's complement binary for bipolar mode and offset binary for unipolar mode.

Programming information for the A/D controller is provided in the [Software Summary](#) section of this manual.

The 32 analog input channels are terminated at **J2**. The pin definitions are shown in the illustration.

J2

AD1 CH 0	1 0 0 2	GND
AD1 CH 1	3 0 0 4	GND
AD1 CH 2	5 0 0 6	GND
AD1 CH 3	7 0 0 8	GND
AD1 CH 4	9 0 0 10	GND
AD1 CH 5	11 0 0 12	GND
AD1 CH 6	13 0 0 14	GND
AD1 CH 7	15 0 0 16	GND
AD2 CH 0	17 0 0 18	GND
AD2 CH 1	19 0 0 20	GND
AD2 CH 2	21 0 0 22	GND
AD2 CH 3	23 0 0 24	GND
AD2 CH 4	25 0 0 26	GND
AD2 CH 5	27 0 0 28	GND
AD2 CH 6	29 0 0 30	GND
AD2 CH 7	31 0 0 32	GND
GND	33 0 0 34	GND

Note: In differential input mode only the even channel numbers (0,2,4, . . .) are used and the signal is applied between the even channel number and the next odd channel input pin.

NOTE: When the board is powered off, the A/D input has a 20K input impedance and is protected to $\pm 20V$. When the board is powered on, the A/D differential channel input has a 31K input impedance and the single-ended channel input has a 42K input impedance. Power on inputs are protected to $\pm 25V$.

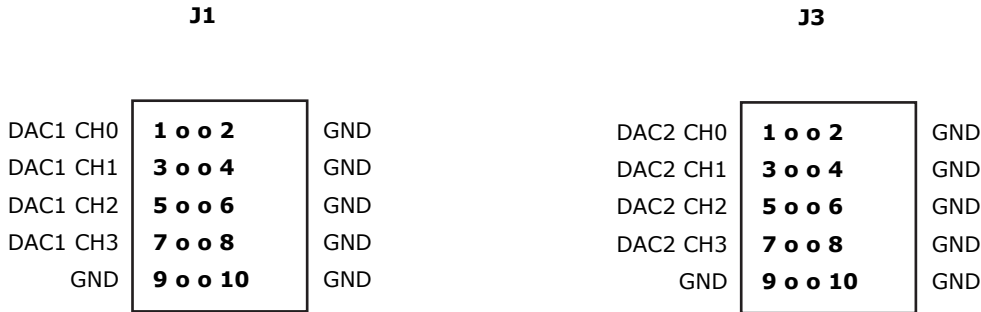


D/A Section



The PCM-MIO-G provides digital-to-analog conversion input using 2 of the 12-bit Linear Technologies LTC2704 devices. These SoftSpan™ quad Digital-to-Analog converters (DACs) are software programmable for either unipolar or bipolar mode with specific voltage ranges on a per channel basis. Each of the 8 channels can be programmed to any one of the six output ranges (0V to 5V, 0V to 10V, ±2.5V, ±5V, ±10V and -2.5V to 7.5V).

The analog output channels are terminated at **J1** and **J3**. The pin out definitions are:

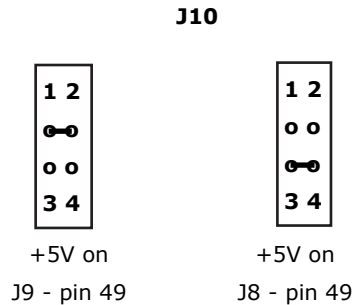


Parallel I/O

The PCM-MIO-G utilizes the WinSystems WS16C48 ASIC high-density I/O chip. The 48 lines are each individually programmable as input or output and the first 24 lines are capable of fully latched event sensing with sense polarity being software programmable.

VCC Enable

The digital I/O connector can provide +5 volts to an I/O rack, when required. +5V is provided at pin 49 of connector J9 when **J10** is jumpered, pin 3-4. +5V is provided at pin 49 of connector J8 when **J10** is jumpered, pin 1-2. It is the user's responsibility to limit current to a safe value (less than 400mA) to avoid damaging the CPU board.



Parallel I/O Connectors

The 48 lines of parallel I/O are terminated through two 50-pin connectors at **J8** and **J9**. The **J9** connector handles I/O ports 0-2 while **J8** handles ports 3-5. The pin definitions for **J8** and **J9** are:



J9			J8		
PORT 2 BIT 7	1 0 0 2	GND	PORT 5 BIT 7	1 0 0 2	GND
PORT 2 BIT 6	3 0 0 4	GND	PORT 5 BIT 6	3 0 0 4	GND
PORT 2 BIT 5	5 0 0 6	GND	PORT 5 BIT 5	5 0 0 6	GND
PORT 2 BIT 4	7 0 0 8	GND	PORT 5 BIT 4	7 0 0 8	GND
PORT 2 BIT 3	9 0 0 10	GND	PORT 5 BIT 3	9 0 0 10	GND
PORT 2 BIT 2	11 0 0 12	GND	PORT 5 BIT 2	11 0 0 12	GND
PORT 2 BIT 1	13 0 0 14	GND	PORT 5 BIT 1	13 0 0 14	GND
PORT 2 BIT 0	15 0 0 16	GND	PORT 5 BIT 0	15 0 0 16	GND
PORT 1 BIT 7	17 0 0 18	GND	PORT 4 BIT 7	17 0 0 18	GND
PORT 1 BIT 6	19 0 0 20	GND	PORT 4 BIT 6	19 0 0 20	GND
PORT 1 BIT 5	21 0 0 22	GND	PORT 4 BIT 5	21 0 0 22	GND
PORT 1 BIT 4	23 0 0 24	GND	PORT 4 BIT 4	23 0 0 24	GND
PORT 1 BIT 3	25 0 0 26	GND	PORT 4 BIT 3	25 0 0 26	GND
PORT 1 BIT 2	27 0 0 28	GND	PORT 4 BIT 2	27 0 0 28	GND
PORT 1 BIT 1	29 0 0 30	GND	PORT 4 BIT 1	29 0 0 30	GND
PORT 1 BIT 0	31 0 0 32	GND	PORT 4 BIT 0	31 0 0 32	GND
PORT 0 BIT 7	33 0 0 34	GND	PORT 3 BIT 7	33 0 0 34	GND
PORT 0 BIT 6	35 0 0 36	GND	PORT 3 BIT 6	35 0 0 36	GND
PORT 0 BIT 5	37 0 0 38	GND	PORT 3 BIT 5	37 0 0 38	GND
PORT 0 BIT 4	39 0 0 40	GND	PORT 3 BIT 4	39 0 0 40	GND
PORT 0 BIT 3	41 0 0 42	GND	PORT 3 BIT 3	41 0 0 42	GND
PORT 0 BIT 2	43 0 0 44	GND	PORT 3 BIT 2	43 0 0 44	GND
PORT 0 BIT 1	45 0 0 46	GND	PORT 3 BIT 1	45 0 0 46	GND
PORT 0 BIT 0	47 0 0 48	GND	PORT 3 BIT 0	47 0 0 48	GND
+5V	49 0 0 50	GND	+5V	49 0 0 50	GND

PC/104 Bus Interface

The control, data, and power signals are wired to a 16-bit stackthrough PC/104 connector located at **J5** and **J6**. The pin definitions are:



J5			J6		
GND	D0 o o C0	GND	IOCHK#	A1 o o B1	GND
MEMCS16#	D1 o o C1	SBHE#	SD7	A2 o o B2	RESET
IOCS16#	D2 o o C2	LA23	SD6	A3 o o B2	+5V
IRQ10	D3 o o C3	LA22	SD5	A4 o o B4	IRQ9
IRQ11	D4 o o C4	LA21	SD4	A5 o o B5	-5V
IRQ12	D5 o o C5	LA20	SD3	A6 o o B6	DRQ2
IRQ15	D6 o o C6	LA19	SD2	A7 o o B7	-12V
IRQ14	D7 o o C7	LA18	SD1	A8 o o B8	SRDY#
DACK0#	D8 o o C8	LA17	SD0	A9 o o B9	+12V
DRQ0	D9 o o C9	MEMR#	IOCHRDY	A10 o o B10	KEY
DACK5#	D10 o o C10	MEMW#	AEN	A11 o o B11	SMEMW#
DRQ5	D11 o o C11	SD8	SA19	A12 o o B12	SMEMR#
DACK6#	D12 o o C12	SD9	SA18	A13 o o B13	IOW#
DRQ6	D13 o o C13	SD10	SA17	A14 o o B14	IOR#
DACK7#	D14 o o C14	SD11	SA16	A15 o o B15	DACK3#
DRQ7	D15 o o C15	SD12	SA15	A16 o o B16	DRQ3
+5V	D16 o o C16	SD13	SA14	A17 o o B17	DACK1#
MASTER#	D17 o o C17	SD14	SA13	A18 o o B18	DRQ1
GND	D18 o o C18	SD15	SA12	A19 o o B19	REFRESH#
GND	D19 o o C19	KEY	SA11	A20 o o B20	BCLK
			SA10	A21 o o B21	IRQ7
			SA9	A22 o o B22	IRQ6
			SA8	A23 o o B23	IRQ5
			SA7	A24 o o B24	IRQ4
			SA6	A25 o o B25	IRQ3
			SA5	A26 o o B26	DACK2#
			SA4	A27 o o B27	TC
			SA3	A28 o o B28	BALE
			SA2	A29 o o B29	+5V
			SA1	A30 o o B30	OSC
			SA0	A31 o o B31	GND
			GND	A32 o o B32	GND

= Active Low Signal

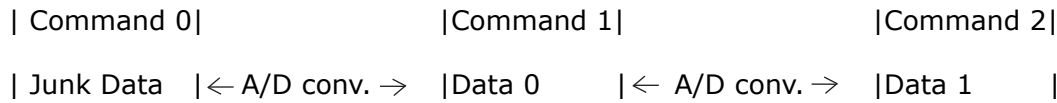
NOTES:

1. Rows C and D are not required on 8-bit modules.
2. B10 and C19 are key locations. WinSystems uses key pins as connections to GND.
3. Signal timing and function are as specified in ISA specification.
4. Signal source/sink current differ from ISA values.

Software Summary

A/D Converters

The PCM-MIO-G uses two Linear Tech LTC1859CG 8-channel A/D converters. Each device is independently software configurable to support the listed input modes and ranges. The devices use a full duplex serial interface which transmits and receives data simultaneously. An 8-bit command is shifted into the ADC interface to configure it for the next conversion. At the same time, the data from the previous conversion is shifted out of device. Consequently, the conversion result is delayed by one conversion from the command word.



A/D1 - Starting at BASE +0

The COMMAND register, RESOURCE register and RESOURCE ENABLE registers are used to configure the A/D device operation. The specific options of each register are detailed here.

			7	6	5	4	3	2	1	0	
Register	Address (Base+)	Read/Write									
A/D1	DATA_LO	0	R	LOW ORDER DATA BYTE							
				DATA-BIT 7	DATA-BIT 6	DATA-BIT 5	DATA-BIT 4	DATA-BIT 3	DATA-BIT 2	DATA-BIT 1	DATA-BIT 0
	DATA_HI	1	R	HIGH ORDER DATA BYTE							
				DATA-BIT 15	DATA-BIT 14	DATA-BIT 13	DATA-BIT 12	DATA-BIT 11	DATA-BIT 10	DATA-BIT 9	DATA-BIT 8
	COMMAND (1)	2	R/W	COMMAND							
				CMD - BIT 7	CMD - BIT 6	CMD - BIT 5	CMD - BIT 4	CMD - BIT 3	CMD - BIT 2	CMD - BIT 1	CMD - BIT 0
	RESOURCE (2)	2	R/W	DMA CHANNEL ASSIGNMENT				INTERRUPT ROUTING ASSIGNMENT IRQ[15-3] 0,1,2,8 AND 13 NOT AVAILABLE			
				DATA REGISTER BIT 1	DATA REGISTER BIT 0	CMD REGISTER BIT 1	CMD REGISTER BIT 0	BIT 3	BIT 2	BIT 1	BIT 0
	RESOURCE ENABLE	3	W	X	X	X	REGISTER SELECT		DATA DRQ ENABLE (3)	CMD DRQ ENABLE (3)	INTERRUPT ENABLE (3)
							BIT 1	BIT 0			
STATUS	3	R	DATA READY	DATA DMA REQUEST PENDING (3)	CMD DMA REQUEST PENDING (3)	INTERRUPT REQUEST PENDING (3)	REGISTER SELECT STATUS	DATA DRQ ENABLE STATUS (3)	CMD DRQ ENABLE STATUS (3)	INTERRUPT ENABLE BIT STATUS (3)	

- Notes:
- (1) Accessed when REGISTER SELECT (BASE +3 bit 4 & 3) = 00
 - (2) Accessed when REGISTER SELECT (BASE +3 bit 4 & 3) = 01
 - (3) 0=Disabled, 1=Enable

A/D2 - Starting at BASE +4

The interface for the second A/D device is almost identical to the first device. Notice in this table that the REGISTER SELECT function within the RESOURCE ENABLE register is only a single bit. In both software examples shown below, action 1 (one) should be replaced with:

1. Write 'xxxx1xxx' to bit 3 of base+7 (select access to Resources)

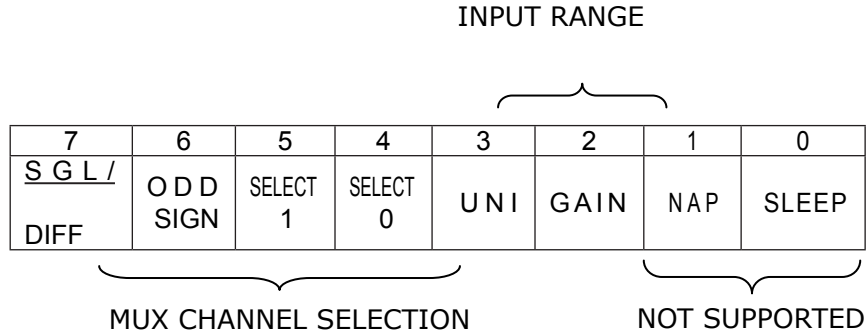
Since bit 4 is a 'don't care' in this case, it is possible to use identical code for both devices just noting the different base address.

Register	Address (Base+)	Read/Write	7	6	5	4	3	2	1	0		
			LOW ORDER DATA BYTE									
DATA_LO	4	R	DATA-BIT 7	DATA-BIT 6	DATA-BIT 5	DATA-BIT 4	DATA-BIT 3	DATA-BIT 2	DATA-BIT 1	DATA-BIT 0		
DATA_HI	5	R	HIGH ORDER DATA BYTE									
COMMAND (1)	6	R/W	COMMAND									
			CMD - BIT 7	CMD - BIT 6	CMD - BIT 5	CMD - BIT 4	CMD - BIT 3	CMD - BIT 2	CMD - BIT 1	CMD - BIT 0		
RESOURCE (2)	6	R/W	DMA CHANNEL ASSIGNMENT				INTERRUPT ROUTING ASSIGNMENT IRQ[15-3] 0,1,2,8 AND 13 NOT AVAILABLE					
			DATA REGISTER BIT 1	DATA REGISTER BIT 0	CMD REGISTER BIT 1	CMD REGISTER BIT 0	BIT 3	BIT 2	BIT 1	BIT 0		
RESOURCE ENABLE	7	W	X	X	X	X	REGISTER SELECT	DATA DRQ ENABLE (3)	CMD DRQ ENABLE (3)	INTERRUPT ENABLE (3)		
STATUS	7	R	DATA READY	DATA DMA REQUEST PENDING (3)	CMD DMA REQUEST PENDING (3)	INTERRUPT REQUEST PENDING (3)	REGISTER SELECT STATUS	DATA DRQ ENABLE STATUS (3)	CMD DRQ ENABLE STATUS (3)	INTERRUPT ENABLE BIT STATUS (3)		

- Notes:
- (1) Accessed when REGISTER SELECT (BASE+7 bit 3) = 0
 - (2) Accessed when REGISTER SELECT (BASE+7 bit 3) = 1
 - (3) 0=Disabled, 1=Enable

Command Register

Each A/D device contains an 8-bit command register to configure the inputs to single ended or differential mode and the desired input range (0V – 5V, 0V – 10V, +/- 5V and +/- 10V). The following describes the register options.



Multiplexer Channel Selection

MUX ADDRESS				DIFFERENTIAL CHANNEL SELECTION							MUX ADDRESS				SINGLE-ENDED CHANNEL SELECTION										
SGL/ DIFF	ODD SIGN	SELECT 1	0	0	1	2	3	4	5	6	7	SGL/ DIFF	ODD SIGN	SELECT 1	0	0	1	2	3	4	5	6	7	COM	
0	0	0	0	+	-							1	0	0	0	+									-
0	0	0	1			+	-					1	0	0	1			+							-
0	0	1	0					+	-			1	0	1	0					+					-
0	0	1	1							+	-	1	0	1	0							+			-
0	1	0	0	-	+							1	1	0	0		+								-
0	1	0	1			-	+					1	1	0	1				+						-
0	1	1	0					-	+			1	1	1	0						+				-
0	1	1	1							-	+	1	1	1	1								+		-

Channel Selection

Bits 7 - 4 of the command register assign the channel configuration for the requested conversion. The converter will measure the voltage between the two channels indicated by the + and - signs in the table below. In differential mode measurements are from any of 4 adjacent input pairs in either polarity. In single-ended mode, all input channels are measured with respect to GND. Both the + and - inputs are sampled simultaneously so common mode noise is rejected.

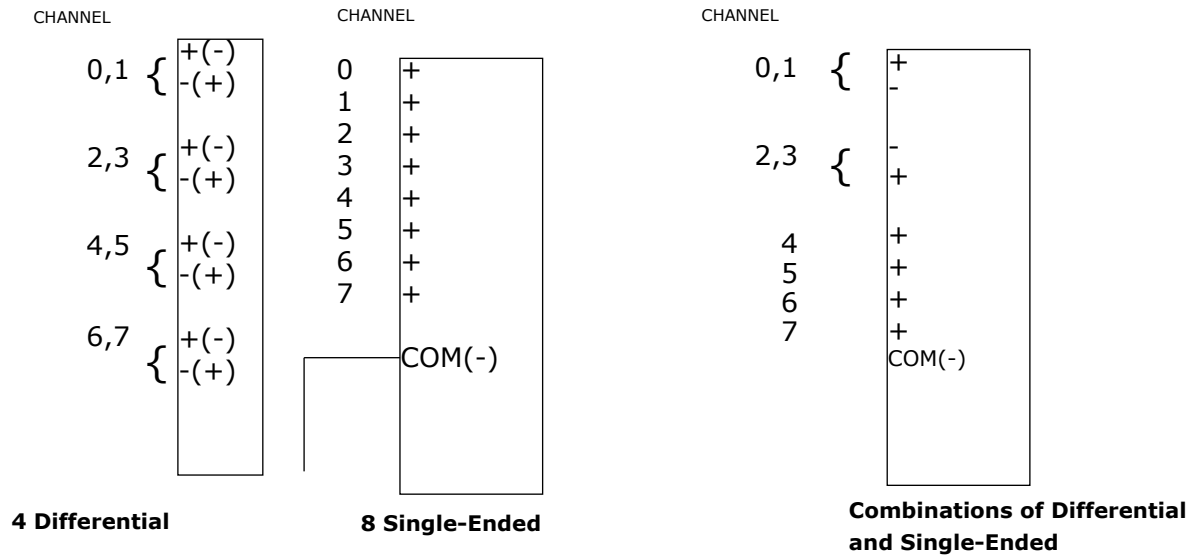
Range Selection



Bits 3 and 2 of the command register determine the input range for the conversion. Setting UNI to a logical one selects a unipolar conversion while a zero selects bipolar. The GAIN bit selects the input span for the conversion in conjunction with the UNI bit. The table below defines the selection options.

INPUT RANGE SELECTION		3	2	INPUT RANGE
UNI	GAIN			
0	0			±5V
1	0			0V to 5V
0	1			±10V
1	1			0V to 10V

Examples of Multiplexer Options



Changing the MUX Assignment "On the Fly"

Example 1 - ADC1

The following is a polled mode example for A/D1.

1. Write 'xxx00xxx' to bits 3&4 of BASE+3 (select access to CMD)
2. Write CMD selection to BASE+2 (set mux channel operation & range)
3. Read data from BASE+0 and discard (lo_byte unknown data)
4. Read data from BASE+1 and discard (hi_byte unknown data)
5. Write CMD selection to BASE+2 again (set mux channel operation & range)
6. Read data from BASE+0 (lo_byte)
7. Read data from BASE+1 (hi_byte)

The data received is actually the result of the first CMD written. Realize that all readings will be offset by one action due to the latching of the serial input data mechanism.

8. Additional readings are achieved by repeating steps 5 through 7.

Example 2 - ADC2

The following is a polled mode example for A/D2, note the difference in starting address and resource enable.

1. Write 'xxxx0xxx' to bit 3 of BASE+7 (select access to CMD)
2. Write CMD selection to BASE+6 (set mux channel operation & range)
3. Read data from BASE+4 and discard (lo_byte unknown data)
4. Read data from BASE+5 and discard (hi_byte unknown data)
5. Write CMD selection to BASE+6 again (set mux channel operation & range)
6. Read data from BASE+4 (lo_byte)
7. Read data from BASE+5 (hi_byte)

The data received is actually the result of the first CMD written. Realize that all readings will be offset by one action due to the latching of the serial input data mechanism.

8. Additional readings are achieved by repeating steps 5 through 7.

Please read the documentation included with the sample programs and drivers for more complex examples.

A/D Interrupts

To operate using interrupt mode, IRQ routing must be configured and interrupts enabled for each device. This is achieved with the Resource and Resource Enable registers. The following would apply to A/D1:

1. Write 'xxx01xxx' to bits 4 & 3 of base +3 (select access to Resource Register).
2. Write IRQ selection (0-15 hex) to bits 3-0 of base+2 ('xF Hex' = IRQ 15).
3. Write 'xxxxxxx1' base+3 to enable the IRQ.

Enabling an interrupt for A/D2 can be achieved in the same manner with the appropriate offset.

It is possible for both devices to share an interrupt or use individual interrupts. When sharing interrupts, the most efficient method to determine which device generated an interrupt request is to utilize the Master Interrupt Status Register.

DMA Support

DMA operation is available for this device. A sample of these operations under DOS is provided in the attached software section. These operations under other operating systems can be quite complex and are beyond the scope of this manual.

D/A Converters

The PCM-MIO-G contains two Linear Tech LTC2704 Digital-to-Analog Converter (DAC) devices. Each device is a 4-channel converter with software-selectable output span.

D/A1 - Starting at BASE+8

The COMMAND register, RESOURCE register and RESOURCE ENABLE registers are used to configure the D/A device operation. The specific options of each register are detailed here.

	Register	Address (Base+)	Read/Write	7	6	5	4	3	2	1	0
D/A1	DATA_LO	8	R/W	LOW ORDER DATA BYTE							
				DATA-BIT 7	DATA-BIT 6	DATA-BIT 5	DATA-BIT 4	DATA-BIT 3	DATA-BIT 2	DATA-BIT 1	DATA-BIT 0
	READBACK (1)	8	R	LOW ORDER DATA BYTE							
				DATA-BIT 7	DATA-BIT 6	DATA-BIT 5	DATA-BIT 4	DATA-BIT 3	DATA-BIT 2	DATA-BIT 1	DATA-BIT 0
	DATA_HI	9	R/W	HIGH ORDER DATA BYTE							
				DATA-BIT 15	DATA-BIT 14	DATA-BIT 13	DATA-BIT 12	DATA-BIT 11	DATA-BIT 10	DATA-BIT 9	DATA-BIT 8
	READBACK (1)	9	R	HIGH ORDER DATA BYTE							
				DATA-BIT 15	DATA-BIT 14	DATA-BIT 13	DATA-BIT 12	DATA-BIT 11	DATA-BIT 10	DATA-BIT 9	DATA-BIT 8
	COMMAND (2)	10	R/W	COMMAND							
				CMD - BIT 7	CMD - BIT 6	CMD - BIT 5	CMD - BIT 4	CMD - BIT 3	CMD - BIT 2	CMD - BIT 1	CMD - BIT 0
RESOURCE (3)	10	R/W	DMA CHANNEL ASSIGNMENT				INTERRUPT ROUTING ASSIGNMENT IRQ[15-3] 0,1,2,8 AND 13 NOT AVAILABLE				
			DATA REGISTER BIT 1	DATA REGISTER BIT 0	CMD REGISTER BIT 1	CMD REGISTER BIT 0	BIT 3	BIT 2	BIT 1	BIT 0	
RESOURCE ENABLE	11	W	X	X	X	READBACK ENABLE	REGISTER SELECT	DATA DRQ ENABLE (4)	CMD DRQ ENABLE (4)	INTERRUPT ENABLE (4)	
STATUS	11	R	DATA READY	DATA DMA REQUEST PENDING (4)	CMD DMA REQUEST PENDING (4)	INTERRUPT REQUEST PENDING (4)	REGISTER SELECT STATUS	DATA DRQ ENABLE STATUS (4)	CMD DRQ ENABLE STATUS (4)	INTERRUPT ENABLE BIT STATUS (4)	

- Notes:
- (1) Accessed when READBACK ENABLE (BASE+11 bit 4) = 1
 - (2) Accessed when REGISTER SELECT (BASE+11 bit 3) = 0
 - (3) Accessed when REGISTER SELECT (BASE+11 bit 3) = 1
 - (4) 0=Disabled, 1=Enable

D/A2 - Starting at BASE+12

As shown in the table below, interface to the 2nd device is almost identical to the first with a change in the base address. The RESOURCE ENABLE register does contain an additional register select bit, MASTER IRQ / DA2 SELECT. If this bit is set to 1, reading BASE+15 will result in the status of the Master Interrupt Status Register.

Register	Address (Base+)	Read/Write	7	6	5	4	3	2	1	0		
			LOW ORDER DATA BYTE									
DATA_LO	12	R/W	LOW ORDER DATA BYTE									
			DATA-BIT 7	DATA-BIT 6	DATA-BIT 5	DATA-BIT 4	DATA-BIT 3	DATA-BIT 2	DATA-BIT 1	DATA-BIT 0		
READBACK (1)	12	R	LOW ORDER DATA BYTE									
			DATA-BIT 7	DATA-BIT 6	DATA-BIT 5	DATA-BIT 4	DATA-BIT 3	DATA-BIT 2	DATA-BIT 1	DATA-BIT 0		
DATA_HI	13	R/W	HIGH ORDER DATA BYTE									
			DATA-BIT 15	DATA-BIT 14	DATA-BIT 13	DATA-BIT 12	DATA-BIT 11	DATA-BIT 10	DATA-BIT 9	DATA-BIT 8		
READBACK (1)	13	R	HIGH ORDER DATA BYTE									
			DATA-BIT 15	DATA-BIT 14	DATA-BIT 13	DATA-BIT 12	DATA-BIT 11	DATA-BIT 10	DATA-BIT 9	DATA-BIT 8		
COMMAND (2)	14	R/W	COMMAND									
			CMD - BIT 7	CMD - BIT 6	CMD - BIT 5	CMD - BIT 4	CMD - BIT 3	CMD - BIT 2	CMD - BIT 1	CMD - BIT 0		
RESOURCE (3)	14	R/W	DMA CHANNEL ASSIGNMENT				INTERRUPT ROUTING ASSIGNMENT IRQ[15-3] 0,1,2,8 AND 13 NOT AVAILABLE					
			DATA REGISTER BIT 1	DATA REGISTER BIT 0	CMD REGISTER BIT 1	CMD REGISTER BIT 0	BIT 3	BIT 2	BIT 1	BIT 0		
RESOURCE ENABLE	15	W	X	X	/DA2 SELECT	READBACK ENABLE	REGISTER SELECT	DATA DRQ ENABLE (4)	CMD DRQ ENABLE (4)	INTERRUPT ENABLE (4)		
STATUS (5)	15	R	DATA READY	DATA DMA REQUEST PENDING (4)	CMD DMA REQUEST PENDING (4)	INTERRUPT REQUEST PENDING (4)	REGISTER SELECT STATUS	DATA DRQ ENABLE STATUS (4)	CMD DRQ ENABLE STATUS (4)	INTERRUPT ENABLE BIT STATUS (4)		
IRQ REGISTER (6)	15	R	DATA READY	DATA DMA REQUEST PENDING (4)	CMD DMA REQUEST PENDING (4)	DA2 IRQ PENDING	DIO IRQ PENDING	DA1 IRQ PENDING	AD2 IRQ PENDING	AD1 IRQ PENDING		

- Notes:
- (1) Accessed when READBACK ENABLE (BASE+15 bit 4) = 1
 - (2) Accessed when REGISTER SELECT (BASE+15 bit 3) = 0
 - (3) Accessed when REGISTER SELECT (BASE+15 bit 3) = 1
 - (4) 0=Disabled, 1=Enable
 - (5) Accessed when MASTER IRQ/DA2 SELECT (BASE+15 bit 5) = 0
 - (6) Accessed when MASTER IRQ/DA2 SELECT (BASE+15 bit 5) = 1

The Linear Tech LTC2704 devices are unique in that each channel consists of a double-buffered data register (B1 Code and B2 Code) and a double-buffered span register (B1 Span and B2 Span). B1 are the holding buffers and data is loaded into each one using a write operation, the DAC outputs are not affected. The contents of the B2 buffers can only be updated by copying the contents of B1 into B2 via an update operation initiated by the Command Code. The contents of the B2 buffers (both DAC Span and DAC Code) directly control the DAC output voltage or the DAC output range. Configuration, programming and writing of the D/A data is achieved through a series of control registers listed below for each DAC.

Command Register

Each DAC contains a command register used to configure the span and load the data. The command word consists of a 4 bit command and a 4 bit address, as shown. Each DAC contains a command register used to configure the span and load the data. The command word consists of a 4 bit command and a 4 bit address, as shown.

7	6	5	4	3	2	1	0
C3	C2	C1	C0	A3	A2	A1	A0

Command Codes

C3	C2	C1	C0	COMMAND	Readback Pointer Current Input Word	Readback Pointer Next Input Word
0	0	1	0	Write to B1 Span DAC n	Set by Previous Command	B1 Span DAC n
0	0	1	1	Write to B1 Code DAC n	Set by Previous Command	B1 Code DAC n
0	1	0	0	Update B1→B2 DAC n	Set by Previous Command	B2 Span DAC n
0	1	0	1	Update B1→B2 All DAC n	Set by Previous Command	B2 Code DAC A
0	1	1	0	Write to B1 Span DAC n Update B1→B2 DAC n	Set by Previous Command	B2 Span DAC n
0	1	1	1	Write to B1 Code DAC n Update B1→B2 DAC n	Set by Previous Command	B2 Code DAC n
1	0	0	0	Write to B1 Span DAC n Update B1→B2 All DACs	Set by Previous Command	B2 Span DAC n
1	0	0	1	Write to B1 Code DAC n Update B1→B2 All DACs	Set by Previous Command	B2 Code DAC n
1	0	1	0	Read B1 Span DAC n	B1 Span DAC n	
1	0	1	1	Read B1 Code DAC n	B1 Code DAC n	
1	1	0	0	Read B2 Span DAC n	B2 Span DAC n	
1	1	0	1	Read B2 Code DAC n	B2 Code DAC n	
1	1	1	1	No Operation	Set by Previous Command	B2 Span DAC n

Codes not shown are reserved and should not be used

Address Codes

A3	A2	A1	A0	n	READBACK POINTER n
0	0	0	0	DAC A	DAC A
0	0	1	0	DAC B	DAC B
0	1	0	0	DAC C	DAC C
0	1	1	0	DAC D	DAC D
1	1	1	1	All DACs	DAC A

Codes not shown are reserved and should not be used

Span Codes

The span for each channel is set by loading the desired value into the data registers then issuing one of span commands. The last 4 bits set the span as shown below, the rest of the data should be set to 0.

S3	S2	S1	S0	SPAN
0	0	0	0	Unipolar 0V to 5V
0	0	0	1	Unipolar 0V to 10V
0	0	1	0	Bipolar -5V to 5V
0	0	1	1	Bipolar -10V to 10V
0	1	0	0	Bipolar -2.5V to 2.5V
0	1	0	1	Bipolar -2.5V to 7.5V

Codes not shown are reserved and should not be used

Readback Enable

Each time a command is issued to one of Linear Tech LTC2704 devices, the value of one of the buffers is simultaneously shifted out of the device. Except when issuing one of the specific readback commands (Ax, Bx, Cx, Dx), the data returned corresponds to the READBACK Pointer from the previous command as shown in the [Command Codes Table](#). The READBACK ENABLE bit must be set to 1 to read this data.

Note: If the READBACK ENABLE bit is set to 0, a read of the DAC data registers will return the last value written to that register not the READBACK value of the actual buffers.

D/A Interrupts

To operate using interrupt mode, IRQ routing must be configured and interrupts enabled for each device. This is achieved with the Resource and Resource Enable registers. The following would apply to D/A1:

1. Write 'xxxx1xxx' to bit 3 of base +11 (select access to Resource Register).
2. Write IRQ selection (0-15 hex) to bits 3-0 of base+10 ('xF Hex' = IRQ 15).
3. Write 'xxxxxxx1' base+11 to enable the IRQ.

Enabling an interrupt for D/A2 can be achieved in the same manner with the appropriate offset.

It is possible for both devices to share an interrupt or use individual interrupts. When sharing interrupts, the most efficient method to determine which device generated an interrupt request is to utilize the Master Interrupt Status Register.

DMA Support

DMA operation is available for this device. A sample of these operations under DOS is provided in the attached software section. These operations under other operating systems can be quite complex and are beyond the scope of this manual.

D/A Examples

The most basic method is to first set the output span for a channel and then write the output value for that channel. Notice that the configuration and data write operations can each be performed with either a single or double instruction sequence. Each channel can be updated individually using command values 6x & 7x (for configuration and output data, respectively) which will pre-load the value and present it to the DAC with a single instruction sequence. The second option is to pre-load the configuration and output data using command values 2x & 3x and then present the values to the DAC either individually (with command value 4x) or simultaneously (with command value 5x).

Example 1 - Single Instruction Sequence

To configure and write data to a DAC channel, each with a single command sequence, is very simple. The configuration must be set first and then the data output is written. Of course the span configuration is only required to be set once unless changes are required during the application.

1. Write 'xxxx0xxx' to bit 3 of base +11 (select access to CMD)
2. Write Span data '0000xxxx' to base+8 (where xxxx = span)
3. Write 00000000 (zero) to base+9 (high order byte for span)
4. Write CMD 01100xxx to base+10 (where xxx = DAC channel)
5. Write low byte data to base+8
6. Write high byte data to base+9
7. Write CMD 01110xxx to base+10 (where xxx = DAC channel)

Additional channels are then programmed by repeating steps 2-7.

Example 2 - Double Instruction Sequence

The second option is to pre-load the configuration and output data using command values 2x and 3x and then present the values to the DAC either individually using command value 4x or simultaneously with command value 5x.

This example will demonstrate pre-loading the span configuration and data output values for each DAC channel and then presenting the information simultaneously to all DAC channels.

1. Write 'xxxx0xxx' to bit 3 of base +11 (select access to CMD)
2. To set Span Configuration for channel 0
 - Write Span Config data 00000000 to base+8 (set span = 0V to 5V)
 - Write 00000000 (zero) to base+9 (high order data byte)
 - Write CMD 00100000 to base+10 (move data to B1 Span)
3. To set Span Configuration for channel 1
 - Write Span Config data 00000001 to base+8 (set span = 0V to 10V)
 - Write 00000000 (zero) to base+9 (high order data byte)
 - Write CMD 00100010 to base+10 (move data to B1 Span)
4. To set Span Configuration for channel 2
 - Write Span Config data 00000010 to base+8 (set span = -5V to 5V)
 - Write 00000000 (zero) to base+9 (high order data byte)
 - Write CMD 00100100 to base+10 (move data to B1 Span)
5. To set Span Configuration for channel 3
 - Write Span Config data 00000011 to base+8 (set span = -10V to 10V)
 - Write 00000000 (zero) to base+9 (high order data byte)
 - Write CMD 00100110 to base+10 (move data to B1 Span)
6. To pre-load Data Output for channel 0
 - Write low byte data to base+8
 - Write high byte data to base+9
 - Write CMD 00110000 to base+10 (move data to B1 Code)
7. To pre-load Data Output for channel 1
 - Write low byte data to base+8
 - Write high byte data to base+9
 - Write CMD 00110010 to base+10 (move data to B1 Code)
8. To pre-load Data Output for channel 2
 - Write low byte data to base+8
 - Write high byte data to base+9
 - Write CMD 00110100 to base+10 (move data to B1 Code)
9. To pre-load Data Output for channel 3
 - Write low byte data to base+8
 - Write high byte data to base+9
 - Write CMD 00110110 to base+10 (move data to B1 Code)
10. To simultaneously update all DAC channels
 - Write CMD 01011111 to base+10 (move all B1 --> B2 ALL Channels)

If the application requires all DAC channels to be configured to the same output span, command value 8x supports this action with a single instruction sequence. Likewise if all DAC channels will be written with the same data output then command value 9x both pre-loads and presents the value to all DAC channels with a single instruction sequence.

Parallel I/O

The PCM-MIO-G uses the WinSystems exclusive ASIC device, the WS16C48. The device provides 48 lines of Digital I/O. There are 17 unique registers within the WS16C48. The following table summarizes the registers and the text that follows provides details on each of the internal registers as supported on this board.

BASE+	I/O Address Offset	Page 0	Page 1	Page 2	Page 3
16	00H	Port 0 I/O	Port 0 I/O	Port 0 I/O	Port 0 I/O
17	01H	Port 1 I/O	Port 1 I/O	Port 1 I/O	Port 1 I/O
18	02H	Port 2 I/O	Port 2 I/O	Port 2 I/O	Port 2 I/O
19	03H	Port 3 I/O	Port 3 I/O	Port 3 I/O	Port 3 I/O
20	04H	Port 4 I/O	Port 4 I/O	Port 4 I/O	Port 4 I/O
21	05H	Port 5 I/O	Port 5 I/O	Port 5 I/O	Port 5 I/O
22	06H	Int_Pending	Int_Pending	Int_Pending	Int_Pending
23	07H	Page/Lock	Page/Lock	Page/Lock	Page/Lock
24	08H	N/A	Pol_0	Enab_0	Int_ID0
25	09H	N/A	Pol_1	Enab_1	Int_ID1
26	0AH	N/A	Pol_2	Enab_2	Int_ID2

Register Details

Port 0-5 I/O – Each I/O bit in each of the 6 ports can be individually programmed for input or output. Writing a '0' to a bit position causes the corresponding output pin to go to a High-Impedance state (pulled high by external 10K ohm resistors). This allows it to be used as an input. When used in the input mode, a read reflects the inverted state of the I/O pin, such that a high on the pin will read as a '0' in the register. Writing a '1' to a bit position causes the output pin to sink current (up to 12mA), effectively pulling it low.

INT_PENDING – This read-only register reflects the combined state of the INT_ID0 through INT_ID2 registers. When any of the lower 3 bits are set, it indicates that an interrupt is pending on the I/O port corresponding to the bit position(s) that are set. Reading this registers allows an Interrupt Service Routine to quickly determine if any interrupts are pending and which I/O port has pending interrupts.

PAGE/LOCK– This register serves two purposes. The upper two bits select the register page in use as shown here:

D7	D6	Page
0	0	Page 0
0	1	Page 1
1	0	Page 2
1	1	Page 3

Bits 5-0 allow for locking the I/O ports. A '1' written to the I/O port position will prohibit further writes to the corresponding I/O port.

POLO - POL2 – These registers are accessible when page 1 is selected. They allow interrupt polarity selection on a port-by-port and bit-by-bit basis. Writing a '1' to a bit position selects the rising edge detection interrupts while writing a '0' to a bit position selects falling edge detection interrupts.

ENAB0 - ENAB2 – This read-only register reflects the combined state of the INT_ID0 through INT_ID2 registers. When any of the lower 3 bits are set, it indicates that an interrupt is pending on the I/O port corresponding to the bit position(s) that are set. Reading this registers allows an Interrupt Service Routine to quickly determine if any interrupts are pending and which I/O port has pending interrupts.

INT_ID0 - INT_ID2 – These registers are accessible when page 3 is selected. They are used to identify currently pending edge interrupts. A bit when read as a '1' indicates that an edge of the polarity programmed into the corresponding polarity register has been recognized. Note that a write to this register (value ignored) clears ALL of the pending interrupts in this register.

Digital I/O Interrupts

The interrupt registers described above are specific to the internal operation of the DIO device. It is also necessary to configure the interrupt routing to the system processor. This is accomplished with the RESOURCE and RESOURCE ENABLE register as shown in this table.

			7	6	5	4	3	2	1	0
RESOURCE *	2	R/W	X	X	X	X	INT(3)	INT(2)	INT(1)	INT(0)
RESOURCE ENABLE	3	W	X	X	X	DIO ACCESS	X	X	X	X

* Interrupt Routing Assignment (IRQ15-3, 1, 2, 8 and 13 not available)

A sample Configuration:

1. Write 'XXX1XXXX' to BIT 4 of BASE+3 (select access to DIO IRQ configuration register).
2. Write IRQ selection (0-15 hex) to BITS 3-0 of BASE+2.
3. Configure DIO internal IRQ as discussed above.

Master Interrupt Status Register

Although each device contains an interrupt pending status bit, a single read only register is also available to provide the status of all devices in one location. With the limited number of interrupts available, this register is very helpful by allowing your application to share a single interrupt amongst all the onboard devices. The register is accessible by properly selecting BIT 5 at BASE +15. When BIT 5 is set to 1, the register at (BASE+15) can be read as follows:

Interrupt Pending

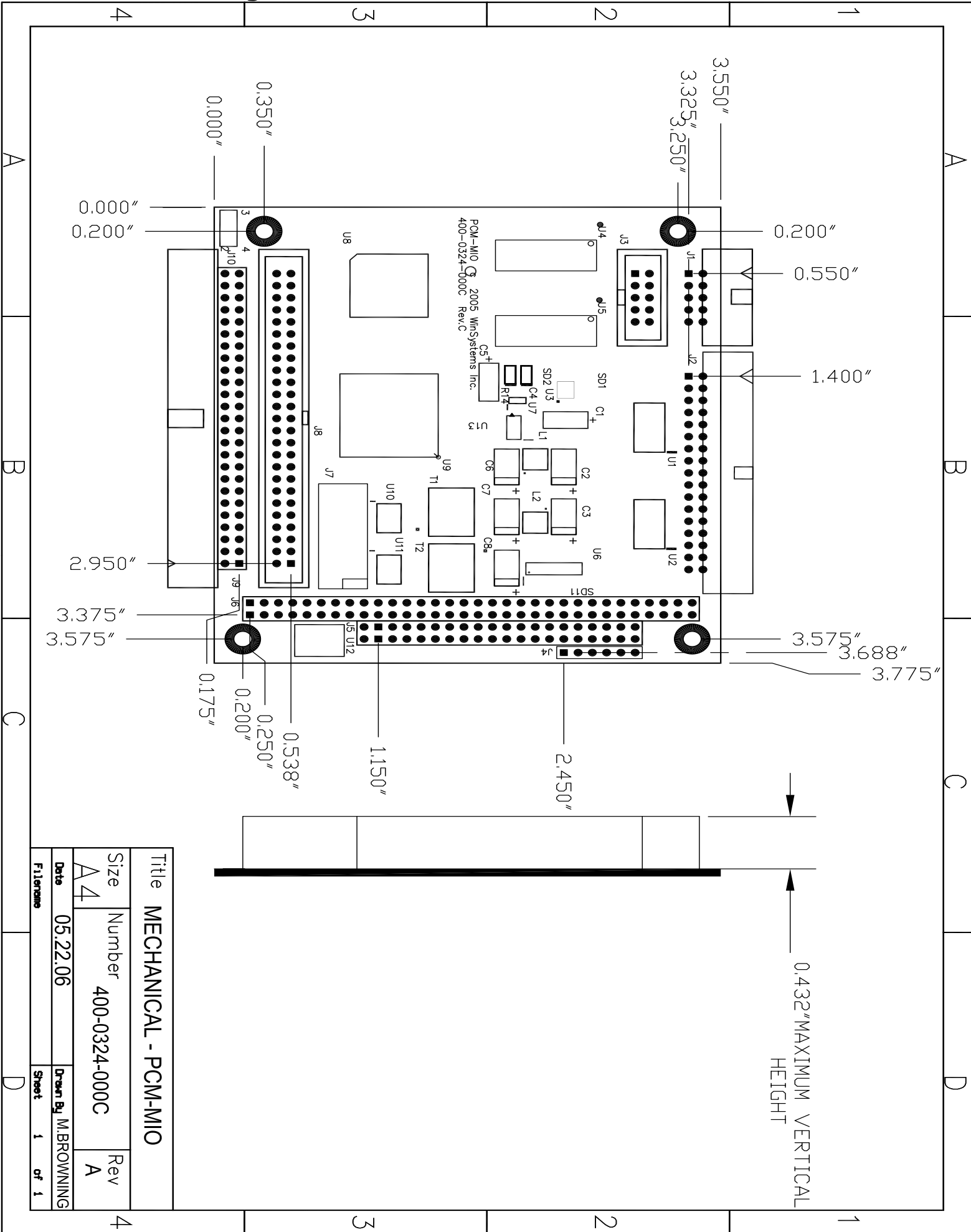
X	X	X	D/A2	DIO	D/A1	A/D2	A/D1
---	---	---	------	-----	------	------	------

Software Drivers & Examples

The following software and drivers are available from WinSystems:

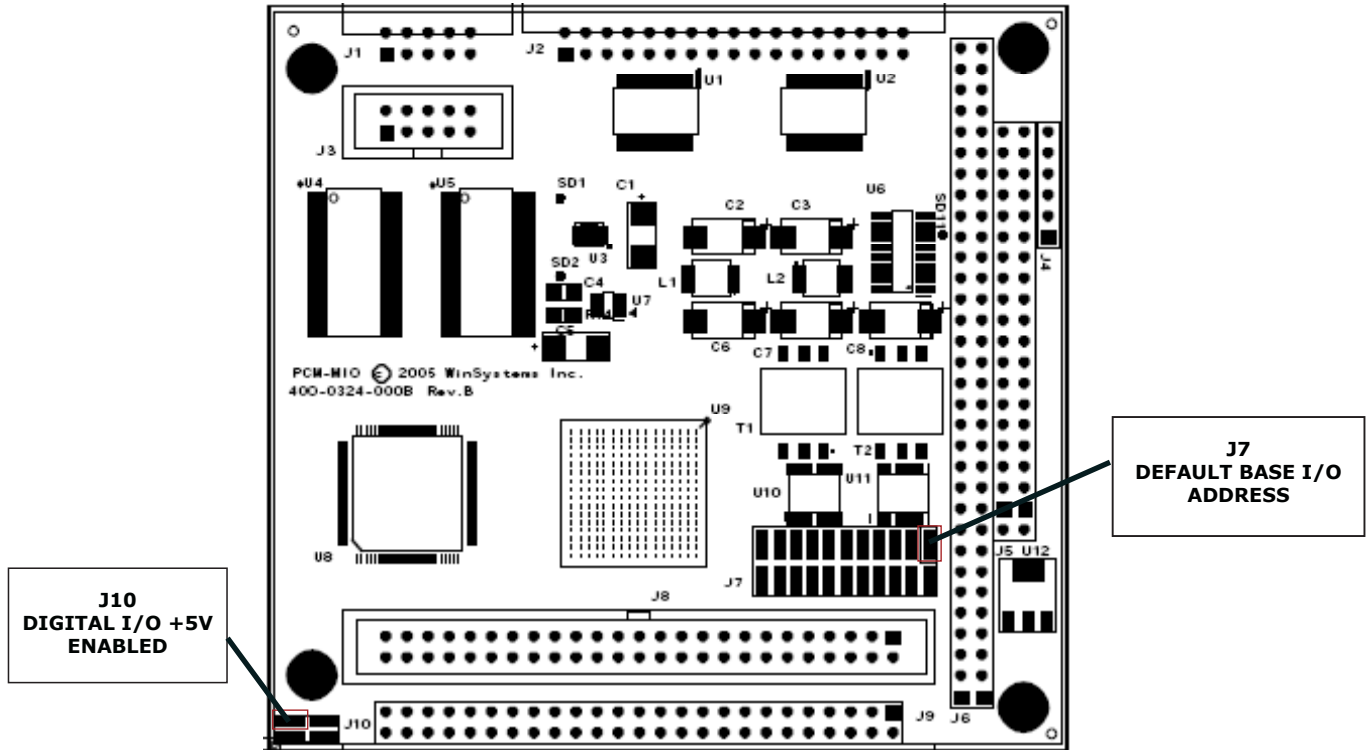
DOS Example & Driver	mio_dos.zip
Linux Driver & Example	
Kernel 2.6	pcmmio_linux.zip
Windows XP/XPEmbedded	mio_xp.zip
Windows CE	Please contact your WinSystems Application Engineer for this driver.

Mechanical Drawing



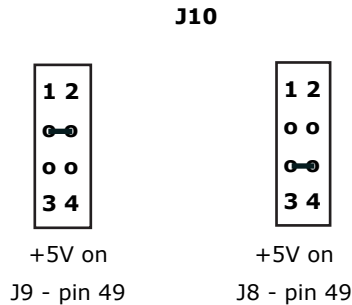
Jumper Reference

Drawings ONLY - for more detailed information on these parts, refer to the descriptions shown previously in this manual.



J10
DIGITAL I/O +5V
ENABLED

J7
DEFAULT BASE I/O
ADDRESS



J7

A5	1 0 0 2
A6	3 0 0 4
A7	5 0 0 6
A8	7 0 0 8
A9	9 0 0 10
A10	11 0 0 12
A11	13 0 0 14
A12	15 0 0 16
A13	17 0 0 18
A14	19 0 0 20
A15	21 0 0 22

Specifications

Electrical

PC/104 Bus Voltage :16-bit, stackthrough
:+5V ±5% at 500mA

A/D Section

Input :Up to 16 channels single ended,
8 channels of differential or combination thereof.
Range :0-5V, 0-10V, ±5V, and ±10V.
Resolution :16-bits (PCM-MIO-G-1)
Monotonicity :Guaranteed over temperature range
Differential :Non-linearity: No missing codes to 15-bits
Input impedance :42kΩ (typ.) unipolar mode
31kΩ (typ.) bipolar mode

D/A Section

Output : 8 channels
Range :0-5V, 0-10V, ±2.5V, ±5V, ±10V, -2.5V, -7.5V.
Resolution :12-bits, no missing codes
Settling time :2µS to 0.1% full scale step
Output Current :±10mA per output typical with ±30mA maximum per board

Digital I/O

Type :48 bits organized in six, 8-byte segments
Logic :TTL-compatible with 12mA sink for each pin

Mechanical

Dimensions :3.6" x 3.8" (90 mm x 96mm)
Weight :3.08 oz. (.0863 kg)

Connectors

A/D :One, 34-pin on 0.100" grid
D/A :Two, 10-pin on 0.100" grid
Digital I/O :Two, 50-pin on 0.100" grid
PC/104 Bus :64-pin, 0.100" (32-pin double row)
:40-pin, 0.100" (20-pin double row)
Jumpers :0.020" square posts on 2mm centers

Environmental

Operating Temperature :-40°C to +85°C
Non-condensing relative humidity : 5% to 95%
MTBF : 25.82 yrs.

WARRANTY REPAIR INFORMATION

WARRANTY

WinSystems warrants to Customer that for a period of two (2) years from the date of shipment any Products and Software purchased or licensed hereunder which have been developed or manufactured by WinSystems shall be free of any material defects and shall perform substantially in accordance with WinSystems' specifications therefore. With respect to any Products or Software purchased or licensed hereunder which have been developed or manufactured by others, WinSystems shall transfer and assign to Customer any warranty of such manufacturer or developer held by WinSystems, provided that the warranty, if any, may be assigned. Notwithstanding anything herein to the contrary, this warranty granted by WinSystems to the Customer shall be for the sole benefit of the Customer, and may not be assigned, transferred or conveyed to any third party. The sole obligation of WinSystems for any breach of warranty contained herein shall be, at its option, either (i) to repair or replace at its expense any materially defective Products or Software, or (ii) to take back such Products and Software and refund the Customer the purchase price and any license fees paid for the same. Customer shall pay all freight, duty, broker's fees, insurance charges for the return of any Products or Software to WinSystems under this warranty. WinSystems shall pay freight and insurance charges for any repaired or replaced Products or Software thereafter delivered to Customer within the United States. All fees and costs for shipment outside of the United States shall be paid by Customer. The foregoing warranty shall not apply to any Products of Software which have been subject to abuse, misuse, vandalism, accidents, alteration, neglect, unauthorized repair or improper installations.

THERE ARE NO WARRANTIES BY WINSYSTEMS EXCEPT AS STATED HEREIN, THERE ARE NO OTHER WARRANTIES EXPRESS OR IMPLIED INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, IN NO EVENT SHALL WINSYSTEMS BE LIABLE FOR CONSEQUENTIAL, INCIDENTAL OR SPECIAL DAMAGES INCLUDING, BUT NOT LIMITED TO, DAMAGES FOR LOSS OF DATA, PROFITS OR GOODWILL. WINSYSTEMS' MAXIMUM LIABILITY FOR ANY BREACH OF THIS AGREEMENT OR OTHER CLAIM RELATED TO ANY PRODUCTS, SOFTWARE, OR THE SUBJECT MATTER HEREOF, SHALL NOT EXCEED THE PURCHASE PRICE OR LICENSE FEE PAID BY CUSTOMER TO WINSYSTEMS FOR THE PRODUCTS OR SOFTWARE OR PORTION THEREOF TO WHICH SUCH BREACH OR CLAIM PERTAINS.

WARRANTY SERVICE

1. To obtain service under this warranty, obtain a return authorization number. In the United States, contact the WinSystems' Service Center for a return authorization number. Outside the United States, contact your local sales agent for a return authorization number.
2. You must send the product postage prepaid and insured. You must enclose the products in an anti-static bag to protect from damage by static electricity. WinSystems is not responsible for damage to the product due to static electricity.