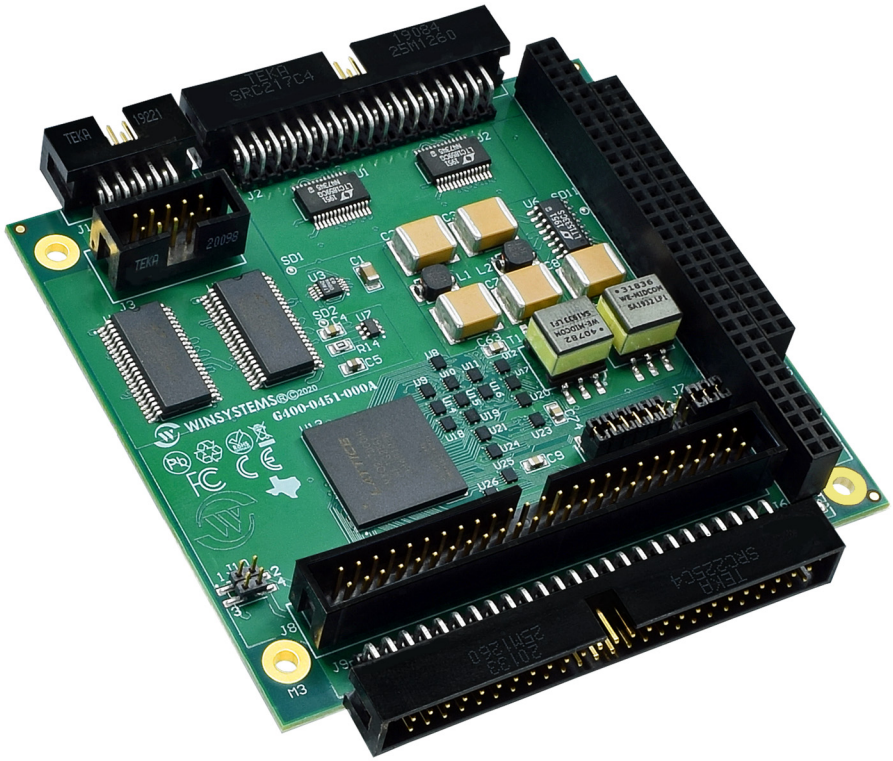
WINSYSTEMS®

PCM-MIO-A-1

16 Channel, 16-bit Analog Inputs
8 Channel, 12-bit Analog Outputs, and
48 Digital I/O

Product Manual



WINSYSTEMS, Inc. | 2890 112th Street, Grand Prairie, Texas 75050 | 817-274-7553 | info@winsystems.com | www.winsystems.com

Revision History

Document Version	Last Updated Date	Brief Description of Change
v1.0	8/27/20	Initial release
v1.1	7/29/2025	Updated Conformal Coating, added Warranty link, updated all links

Copyright and Trademarks

Trademark Acknowledgments

Table of Contents

1. Before You Begin

Review the warnings in this section and the best practice recommendations (see “Best Practices” on page 31) when using and handling the WINSYSTEMS PCM-MIO-A-1 board. Following these recommendations provides an optimal user experience and prevents damage. Read through this document and become familiar with the PCM-MIO-A-1 before proceeding.

APPLYING CONFORMAL COATING AFTER PURCHASE WILL VOID YOUR WARRANTY. FAILING TO COMPLY WITH THESE BEST PRACTICES MAY DAMAGE THE PRODUCT AND VOID YOUR WARRANTY.

1.1 Warnings

Only qualified personnel should configure and install the PCM-MIO-A-1. While observing best practices, pay particular attention to the following.

Avoid Electrostatic Discharge (ESD)
Only handle the circuit board and other bare electronics when electrostatic discharge (ESD) protection is in place. Having a wrist strap and a fully grounded workstation is the minimum ESD protection required before the ESD seal on the product bag is broken.

2. Introduction

This manual provides configuration and usage information for the PCM-MIO-A-1 data acquisition module. If you still have questions, contact Technical Support at (817) 274-7553, Monday through Friday, between 8 AM and 5 PM Central Standard Time (CST).

Refer to the WINSYSTEMS website at https://www.winsystems.com/ for other accessories (including cable drawings and pinouts) that can be used with your PCM-MIO-A-1.

3. Functionality

The PCM-MIO-A-1 is a versatile, PC/104 analog input, analog output, and digital I/O board designed for high accuracy and high channel count analog and digital I/O. The board is based upon Linear Technology’s state-of-the-art precision converters and voltage references, which require no external calibration.

3.1 Base I/O Address

The PCM-MIO-A-1 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 “Software Summary” on page 17.

3.2 Interrupts

The PCM-MIO-A-1 provides flexible interrupt configuration options. Each D/A converter and 24 digital I/O lines 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 “Software Summary” on page 17.

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

3.3 Analog-to-Digital

The PCM-MIO-A-1 analog-to-digital conversion inputs are implemented by using two, 16-bit Linear Technology 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 operates in single-ended or differential mode and also selects the desired input range (0 V to 5 V, 0 V to 10 V, ±5 V and ±10 V).

The LTC-1859CG 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.

3.4 Digital-to-Analog

The PCM-MIO-A-1 provides digital-to-analog conversion using two 12-bit Linear Technology LTC-2704 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 output ranges (0 V to 5 V, 0 V to 10 V, ±2.5 V, ±5 V, ±10 V and -2.5 V to 7.5 V).

4. Features

The PCM-MIO-A-1 provides the following features.

Analog Input

- Two 8-channel, 16-bit analog-to-digital (A/D) (LTC-1859CG) with sample-and-hold-circuit support
- Conversion rate: 100 Ksps max.
- Any combination of up to 16 single-ended input channels and up to 8 differential input channels
- Each channel independently programmable for input type and range
- Input ranges: 0-5 V, 0-10 V, ±5 V or ±10 V
- Input protection: ±25 V
- Supports industry-standard signal conditioners
- Programmable interrupts

Analog Output (not available on the PCM-MIO-A-AD-1)

- Two 4-channel, 12-bit digital-to-analog (D/A) (LTC-2704CGW-12)
- Output ranges: 0-5 V, 0-10 V, ±5 V or ±10 V, +/-2.5 V, -2.5 V to 7.5 V
- Each channel independently programmable for output type and range
- Output channels can be updated and cleared individually or simultaneously
- Interrupt I/O supported
- Supports industry-standard signal conditioners

Digital I/O

- 48 bidirectional lines with input, output, or output with readback (FPGA with WS16C48 compatible programmed logic)
- 12 mA sink current per line
- Ability to generate an interrupt on signal change-of state (24 bits)
- Write-protection mask register for each port
- Programmable edge polarity

Power

- +5 V required, 500 mA typical

Operating Temperature

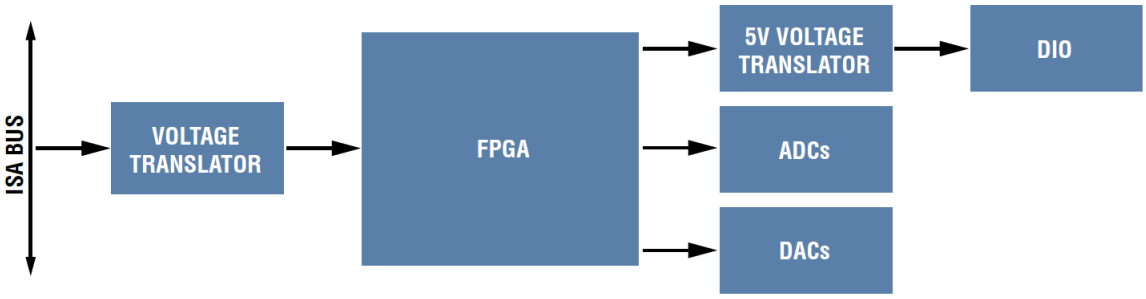
- -40°C to 85°C (-40°F to 185°F)

Form Factor

- 3.6 x 3.8 in. (90 x 96 mm)

5. General Operation

5.1 System Block Diagram



The PCM-MIO-A-1 utilizes a Lattice MachXO2 FPGA to communicate between the PC/104 ISA bus and the ADCs, DACs, and 48 DIO. The analog 16-bit inputs provide up to 100 Ksps, and ±25 V protection via the Linear Technology LTC1859 ADC converters. The DACs provide 12-bit outputs via the Linear Technology LTC2704.

6. Specifications

The PCM-MIO-A-1 adheres to the following specifications and requirements.

Feature	Specification
Electrical	
V_{CC}	+5 V ±5% @ 500 mA (typ.), all outputs unloaded
Models	PCM-MIO-A-1 Includes: ADC, DAC, DIO PCM-MIO-A-AD-1 Includes: ADC, DIO
Absolute Maximums	A/D input protection: • **Board off:** 20 kΩ input impedance, up to ± 20 V • **Board on:** 25 kΩ input impedance, up to ± 25 V
Mechanical	
Dimensions	3.6 in x 3.8 in (90 mm x 96 mm)
Weight	3.20 oz. (90.72 g)
PCB Thickness	0.078 in. (1.98 mm)

Feature	Specification
Environmental	
Temperature	-40 to +85°C (-40 to +185°F)
Humidity (RH)	5% to 95% non-condensing
Mean time between failure (MTBF) [a]	MTBF (hours) 1074379 MTBF (years) 122.646
RoHS Compliant	Yes
Operating Systems	
Compatible with Windows and Linux	

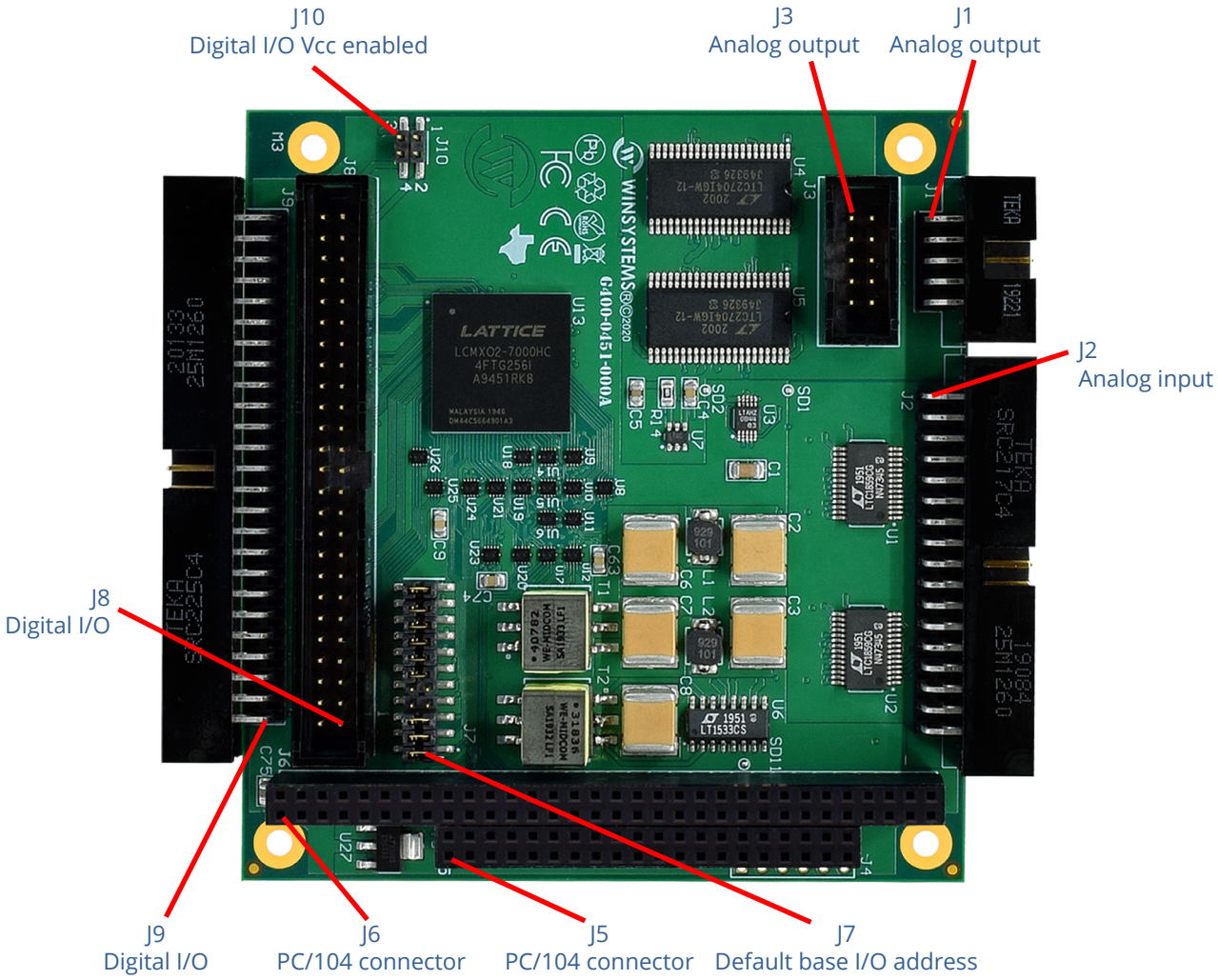
a. A MTBF measurement is based on a statistical sample and is not intended to predict any one specific unit's reliability; thus MTBF is not, and should not be construed as, a warranty measurement.

7. Configuration

This section describes the PCM-MIO-A-1 components and configuration.

7.1 Component Layout

7.1.1 Top View



Item	Description	Reference
J1, J3	D/A Analog Output Headers	page 13
J2	ADC Analog Input Header	page 12
J5, J6	PC/104 Connectors	page 16
J7	Base I/O Address Jumpers	page 11
J8, J9	Digital I/O Headers	page 14
J10	Digital I/O VCC Enable Jumpers	page 12

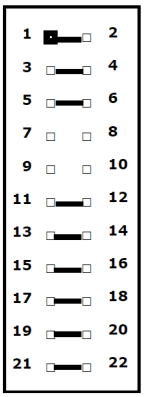
7.2 Jumper Reference

NOTE Jumper part numbers W/S# G201-0002-005 and SAMTEC 2SN-BK-G are applicable to all jumpers. These are available in a five piece kit from WINSYSTEMS (Part# KIT-JMP-G-200).

7.2.1 J7 - Base I/O Address Jumpers

ThePCM-MIO-A-1 is I/O mapped and requires 32 sequential port addresses. The base address is jumper selectable at **J7**. Take care to choose an I/O area that does not conflict with other resources in the system. Address bits A0 to A4 are not selectable. J7 begins at address bit A5. The specific device locations and register offsets are discussed in more detail in “Software Summary” on page 17.

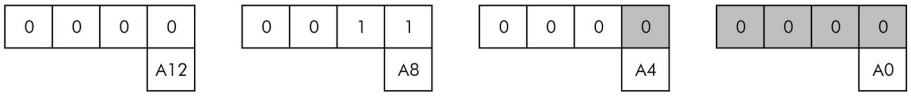
Layout and Pin Reference

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

Example

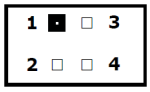
300HEX Default

- Jumper OPEN = 1
- Jumper INSTALLED = 0



7.2.2 J10 - Digital I/O VCC Enable Jumpers

Layout and Pin Reference

Pin	Name
	
3-4	+5V is provided at pin 49 of J9 3-4
1-2	+5V is provided at pin 49 of J8 1-2
OPEN	No power at pin 49 of J8/J9 (default)

7.3 Connector Reference

7.3.1 J2 - ADC Analog Input Header

The PCM-MIO-A-1 and PCM-MIO-A-AD-1 analog-to-digital conversion inputs are implemented by using two, 16-bit Linear Technology 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 operates in single-ended or differential mode and also selects the desired input range (0 V to 5 V, 0 V to 10 V, ±5 V, and ±10 V).

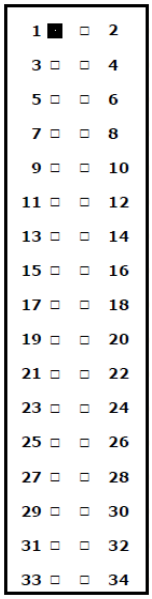
The LTC-1859CG uses a successive approximation algorithm and an internal sample-and-hold circuit to convert an analog input signal to 16-bit digital data. The output is two's complement binary for bipolar mode and offset binary for unipolar mode.

Programming information for the A/D controller is provided in "Software Summary" on page 17.

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

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.

The analog input channels are terminated at **J2**. The pin definitions are shown in the illustration below.Layout and Pin Reference

Diagram	Pin	Name	Pin	Name
	1	AD1 CH0	2	GND
	3	AD1 CH1	4	GND
	5	AD1 CH2	6	GND
	7	AD1 CH3	8	GND
	9	AD1 CH4	10	GND
	11	AD1 CH5	12	GND
	13	AD1 CH6	14	GND
	15	AD1 CH7	16	GND
	17	AD2 CH0	18	GND
	19	AD2 CH1	20	GND
	21	AD2 CH2	22	GND
	23	AD2 CH3	24	GND
	25	AD2 CH4	26	GND
	27	AD2 CH5	28	GND
	29	AD2 CH6	30	GND
	31	AD2 CH7	32	GND
	33	GND	34	GND

Connector

- PCB connector: TEKA SRC217C425M126-0
- Mating connector: ITW-PANCON 050-034-455A

7.3.2 J1, J3 - D/A Analog Output Headers

NOTE The D/A converters are not available on the PCM-MIO-A-AD-1.

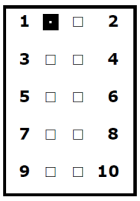
The PCM-MIO-A-1 provides digital-to-analog conversion output using two of the 12-bit Linear Technology LTC-2704 devices. These SoftSpan quad digital-to-analog converters (DACs) are software programmable for either unipolar or bipolar mode with specific voltage ranges on a per channel basis. Each of the 8 channels can be programmed to any one of the six output ranges (0 V to 5 V, 0 V to 10 V, ±2.5 V, ±5 V, ±10 V, and -2.5 V to 7.5 V).

Programming information for the D/A controller is provided in “Software Summary” on page 17.

The analog output channels are terminated at J1 and J3. The pin definitions are shown in the illustration below.

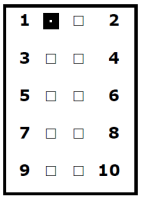
Layout and Pin Reference

J1



Pin	Name	Pin	Name
1	DAC1 CH0	2	GND
3	DAC1 CH1	4	GND
5	DAC1 CH2	6	GND
7	DAC1 CH3	8	GND
9	GND	10	GND

J3



Pin	Name	Pin	Name
1	DAC2 CH0	2	GND
3	DAC2 CH1	4	GND
5	DAC2 CH2	6	GND
7	DAC2 CH3	8	GND
9	GND	10	GND

Connector

- PCB connector: TEKA SRC205C425M126-0 (J1), TEKA SVC205B3580135-0 (J3)
- Mating connector: ITW-PANCON 050-010-455A

7.3.3 J8, J9 Digital I/O Headers

The PCM-MIO-A-1 and PCM-MIO-A-AD-1 use the Lattice MachXO2 FPGA with WINSYSTEMS WS16C48 ASIC compatible programmed logic. The 48 lines are each individually programmable as input or output and the first 24 lines are capable of fully latched event sensing with edge polarity being software programmable.

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 shown below.

Layout and Pin Reference

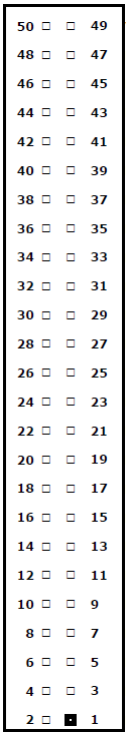
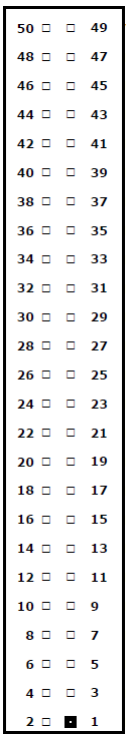
Diagram	Pin	Name	Pin	Name
J9 	50	GND	49	VCC
	48	GND	47	Port 0 Bit 0
	46	GND	45	Port 0 Bit 1
	44	GND	43	Port 0 Bit 2
	42	GND	41	Port 0 Bit 3
	40	GND	39	Port 0 Bit 4
	38	GND	37	Port 0 Bit 5
	36	GND	35	Port 0 Bit 6
	34	GND	33	Port 0 Bit 7
	32	GND	31	Port 1 Bit 0
	30	GND	29	Port 1 Bit 1
	28	GND	27	Port 1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

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

Connector

- PCB connector: TEKA SVC225B3580135-0 (J8), TEKA SRC225C425M126-0 (J9)
- Mating connector: ITW-PANCON 050-050-455A

7.3.4 J5, J6 - PC/104 Connectors

The PC/104 bus is electrically equivalent to the 16-bit ISA bus. Standard PC/104 I/O cards can be populated on PCM-MIO-A-1's connectors, located at J5 and J6. The interface does not support hot swap capability. The PC/104 bus connector pin definitions are provided here for reference. Refer to the PC/104 Bus Specification for specific signal and mechanical specifications.

Layout and Pin Reference

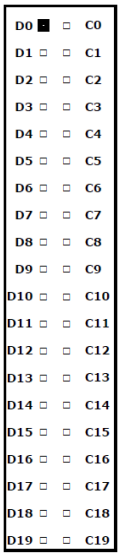
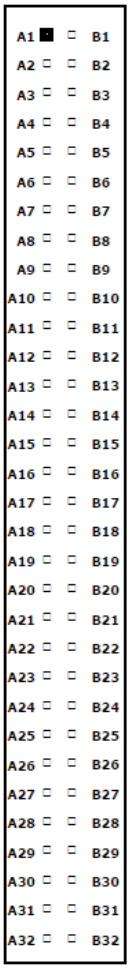
Diagram	Pin	Name	Pin	Name
J5 	D0	GND	C0	GND
	D1	MEMCS16#	C1	SBHE#
	D2	IOCS16#	C2	LA23
	D3	IRQ10	C3	LA22
	D4	IRQ11	C4	LA21
	D5	IRQ12	C5	LA20
	D6	IRQ15	C6	LA19
	D7	IRQ14	C7	LA18
	D8	DACK0#	C8	LA17
	D9	DRQ0	C9	MEMR#
	D10	DACK5#	C10	MEMW#
	D11	DRQ5	C11	SD8
	D12	DACK6#	C12	SB9
	D13	DRQ6	C13	SB10
	D14	DACK7#	C14	SB11
	D15	DRQ7	C15	SB12
	D16	+5V	C16	SB13
	D17	MASTER#	C17	SB14
	D18	GND	C18	SB15
	D19	GND	C19	KEY

Diagram	Pin	Name	Pin	Name
J6 	A1	IOCHK#	B1	GND
	A2	SD7	B2	RESET
	A3	SD6	B3	+5V
	A4	SD5	B4	IRQ
	A5	SD4	B5	-5V
	A6	SD3	B6	DRQ2
	A7	SD2	B7	-12V
	A8	SD1	B8	SRDY#
	A9	SD0	B9	+12V
	A10	IOCHRDY	B10	KEY
	A11	AEN	B11	SMEMW#
	A12	SA19	B12	SMEMR#
	A13	SA18	B13	IOW#
	A14	SA17	B14	IOR#
	A15	SA16	B15	DACK3#
	A16	SA15	B16	DRQ3
	A17	SA14	B17	DACK1#
	A18	SA13	B18	DRQ1
	A19	SA12	B19	REFRESH#
	A20	SA11	B20	BCLK
	A21	SA10	B21	IRQ7
	A22	SA9	B22	IRQ6
	A23	SA8	B23	IRQ5
	A24	SA7	B24	IRQ4
	A25	SA6	B25	IRQ3
	A26	SA5	B26	DACK2#
	A27	SA4	B27	TC
	A28	SA3	B28	BALE
	A29	SA2	B29	+5V
	A30	SA1	B30	OSC
	A31	SA0	B31	GND
	A32	GND	B32	GND

Notes:

= Active low signal

Rows C and D are not required on 8-bit modules.

B10 and C19 are key locations. WINSYSTEMS uses key pins as connections to GND.

Signal timing and function are as specified in ISA specification.

Signal source/sink current differ from ISA values.

PCB connectors

- TEKA PC232-A-W1A7-M (J6)
- TEKA PC220-A-W1A7-M (J5)

8. Software Summary

8.1 A/D Converters

The PCM-MIO-A-1 uses two Linear Technology LTC-1859CG 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.

8.1.1 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.

Register	Address (Base+)	Read/ Write	7	6	5	4	3	2	1	0
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	DON'T CARE				INTERRUPT ROUTING ASSIGNMENT IRQ[15-3] 0, 1, 2, 8, AND 13 NOT AVAILABLE			
			X	X	X	X	BIT 3	BIT 2	BIT 1	BIT 0
DIO RESOURCE (3)	2	R/W	X	X	X	X	BIT 3	BIT 2	BIT 1	BIT 0
RESOURCE ENABLE	3	W	X	X	X	REGISTER SELECT		X	X	INTERRUPT ENABLE (3)
						BIT 1	BIT 0	X	X	
STATUS	3	R	DATA READY	X	X	INTERRUPT REQUEST PENDING (3)	REGISTER SELECT STATUS	X	X	INTERRUPT ENABLE BIT STATUS (3)

Notes:

(1) Accessed when REGISTER SELECT (BASE +3 bit 3) = 00
(2) Accessed when REGISTER SELECT (BASE +3 bit 3) = 01
(3) Accessed when REGISTER SELECT (BASE +7, bit 3) = 1X
(4) 0=Disabled, 1=Enabled

8.1.2 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).

Bit 4 is a don't care in this case, so 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
DATA_LO	4	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	5	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)	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	DON'T CARE				INTERRUPT ROUTING ASSIGNMENT IRQ[15-3] 0, 1, 2, 8, AND 13 NOT AVAILABLE			
			X	X	X	X	BIT 3	BIT 2	BIT 1	BIT 0
DIO RESOURCE (3)	2	R/W	X	X	X	X	BIT 3	BIT 2	BIT 1	BIT 0
RESOURCE ENABLE	7	W	X	X	X	X	REGISTER SELECT	X	X	INTERRUPT ENABLE (3)
								X	X	
STATUS	7	R	DATA READY	X	X	INTERRUPT REQUEST PENDING (3)	REGISTER SELECT STATUS	X	X	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=Enabled

8.1.3 Command Register

Each A/D device contains an 8-bit command register to configure the inputs as single-ended or differential with a desired input range of 0 V–5 V, 0 V–10 V, ±5 V, and ±10 V. The following describes the register options.

				INPUT RANGE			
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
SGL / DIFF	ODD SIGN	SELECT 1	SELECT 0	UNI	GAIN	NAP	SLEEP
MUX CHANNEL SELECTION						**NOT SUPPORTED**	

8.1.4 Multiplexer Channel Selection

MUX ADDRESS				DIFFERENTIAL CHANNEL SELECTION								MUX ADDRESS				SINGLE-ENDED CHANNEL SELECTION								
SGL/ DIFF	ODD SIGN	SELECT		0	1	2	3	4	5	6	7	SGL/ DIFF	ODD SIGN	SELECT		0	1	2	3	4	5	6	7	COM
		1	0											1	0									
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 assigns the channel configuration for the requested conversion. The converter measures the voltage between the two channels indicated by the + and - signs in the table below. In differential mode, measurements are from any of the four 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 defined the selection options.

Input Range Selection

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

Examples of Multiplexer Options

Examples of multiplexer options and changing the MUX assignment on The fly are shown below.

CHANNEL	
0,1	+(-) -(+)
2,3	+(-) -(+)
4,5	+(-) -(+)
6,7	+(-) -(+)

4 Differential

CHANNEL	
0	+
1	+
2	+
3	+
4	+
5	+
6	+
7	+
	COM(-)

8 Single-Ended

CHANNEL	
0,1	+ -
2,3	- +
4	+
5	+
6	+
7	+
	COM(-)

Combinations of Differential and Single-Ended

CHANNEL	
....	....
4,5	+ -
6,7	+ -
	COM(-) (UNUSED)

1st Conversion

CHANNEL	
....	....
4,5	- +
	+
6	+ -
7	COM(-)

2nd Conversion

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. Note that all readings are offset by one action due to the latching of the serial input 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. S	et 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. Note that all readings are offset by one action due to the latching of the serial input mechanism.	
8. Additional readings are achieved by repeating steps 5 through 7.	

NOTE 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 **xxxxxx1** 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.

8.2 D/A Converters

The PCM-MIO-A-1 contains two Linear Technology LTC-2704 digital-to-analog converter (DAC) devices. Each device is a 4-channel converter with software selectable output span.

8.2.1 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
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	DON'T CARE				INTERRUPT ROUTING ASSIGNMENT IRQ[15-3] 0, 1, 2, 8, AND 13 NOT AVAILABLE			
			X	X	X	X	BIT 3	BIT 2	BIT 1	BIT 0
RESOURCE ENABLE	11	W	X	X	X	READBACK ENABLE	REGISTER SELECT	X	X	INTERRUPT ENABLE (4)
STATUS	11	R	DATA READY	X	X	INTERRUPT REQUEST PENDING (4)	REGISTER SELECT STATUS	X	X	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=Enabled

8.2.2 D/A2 - Starting at BASE +12

As shown in the table below, interface to the second 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 results in the status of the Master Interrupt Status Register.

Register	Address (Base+)	Read/ Write	7	6	5	4	3	2	1	0
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	DON'T CARE				INTERRUPT ROUTING ASSIGNMENT IRQ[15-3] 0, 1, 2, 8, AND 13 NOT AVAILABLE			
			X	X	X	X	BIT 3	BIT 2	BIT 1	BIT 0
RESOURCE ENABLE	15	W	X	X	D/A2 SELECT	READBACK ENABLE	REGISTER SELECT	X	X	INTERRUPT ENABLE (4)
STATUS (5)	15	R	DATA READY	X	X	INTERRUPT REQUEST PENDING (4)	REGISTER SELECT STATUS	X	X	INTERRUPT ENABLE BIT STATUS (4)
IRQ REGISTER (6)	15	R		X	X	DA/2 IRQ PENDING	DIO IRQ PENDING	DA/1 IRQ PENDING	AD/2 IRQ PENDING	AD/1 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=Enabled

(5) Accessed when D/A2 SELECT (BASE + 15 bit 5) = 0

(6) Accessed when D/A2 SELECT (BASE + 15 bit 5) = 1

The Linear Technology LTC-2704 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 buffers 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.

8.2.3 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

8.2.4 Command Codes

NOTE Codes not shown are reserved and should not be used.

C3	C2	C1	C0	Command	Readback Point 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 Span DAC n	Set by Previous Command	B1 Span 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 Span DAC n
0	**1**	**1**	**0**	Write to B1 Span DAC n Update B1 -> B2 DAC n	Set by Previous Command	B1 Span DAC n
0	**1**	**1**	**1**	Write to B1 Code DAC n Update B1 -> B2 DAC n	Set by Previous Command	B2 Span 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 Span 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 Span 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 Span DAC n	
1	**1**	**1**	**1**	No Operation	Set by Previous Command	B2 Span DAC n

8.2.5 Address Codes

NOTE Codes not shown are reserved and should not be used.

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

8.2.6 Span Codes

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

NOTE Codes not shown are reserved and should not be used.

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

8.2.7 Readback Enable

Each time a command is issued to one of Linear Technology LTC-2704 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 returns the last value written to that register not the Readback value of the actual buffers.

8.2.8 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 **xxxxxx1** 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.

8.2.9 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 6Xh and 7Xh (for configuration and output data, respectively), which pre-loads 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 2Xh and 3Xh and then present the values to the DAC either individually (with command value 4Xh) or simultaneously (with command value 5Xh).

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)** 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
8. Additional channels are then programmed by repeating steps 2 through 7.	

Example 2 - Double Instruction Sequence

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

This example demonstrates 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 Configuration data 00000000 (zero)** to BASE +8.	Set **Span** = 0 V 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 Configuration data 00000001** to BASE +8.	Set **Span** = 0 V to 10 V
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 Configuration 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 Configuration data 00000011** to BASE +8.	Set **Span** = -10 V to 10 V
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 0101xxxx** to BASE+10.	Move SPAN/DATA B1>>B2 All Channels

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

8.3 Registers

8.3.1 Register Definitions (WS16C48 Logic)

The PCM-MIO-A-1 and PCM-MIO-A-AD-1 use a Lattice MachXO2 FPGA with WINSYSTEMS WS16C48 ASIC compatible programmed logic. This provides 48 lines of digital I/O. There are 17 unique registers within the WS16C48

logic. The following table summarizes the registers, and the text that follows provides details on each of the internal registers.

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

8.3.2 Register Details

Port 0 through 5 I/O

Each I/O bit in each of the six 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 10 KΩ 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 reads as a **0** in the register. Writing a **1** to a bit position causes that output pin to sink current (up to 12 mA), 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 three 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 register allows an Interrupt Service Routine to quickly determine if any interrupts are pending and which I/O port has a pending interrupt.

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 through 0 allow for locking the I/O ports. A **1** written to the I/O port position prohibits further writes to the corresponding I/O port.

IRQ_REG

This register is accessible when Page 0 is selected. It is used to select the bus routine for the generated interrupt.

REV_LO

This register is accessible when Page 0 is selected. This register returns the low byte of the current revision level for the FPGA firmware.

REV_HIGH

This register is accessible when Page 0 is selected. This register returns the high byte of the current revision level for the FPGA firmware.

POL0 - 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

These registers are accessible when Page 2 is selected. They allow for port-by-port and bit-by-bit enabling of the edge detection interrupts. When set to a **1**, the edge detection interrupt is enabled for the corresponding port and bit. When cleared to **0**, the bit's edge detection interrupt is disabled. Note that this register can be used to individually clear a pending interrupt by disabling and re-enabling the pending interrupt.

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.

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 among all the on-board 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

9. Accessories

The following table lists available WINSYSTEMS accessories for the PCM-MIO-A-1.

Go to www.winsystems.com for more information on WINSYSTEMS cables and batteries.

Item	Part Number	Description
Termination board	ISM-TRM-RELAY	Industrial PC/104 relay board, 16 SPDT relays
Termination board	ISM-TRM-ISO-OUT	Termination board, 24 isolated outputs
Termination board	ISM-TRM-ISO-IN	Termination board, 24 isolated inputs
Termination board	ISM-TRM-COMBO	Termination board, 8 in, 8 out, and 8 relays

Standoff kits are available and recommended for use with the PCM-MIO-A-1. The following table lists the items contained in each kit.

Kit	Component	Description	Qty
KIT-G-PCM-STANDOFF-4 4 pc. nylon hex PC/104 standoff kit	Standoff	Nylon 0.25” hex, 0.600” long male/female 4-40	4
	Hex nut	Hex nylon 4-40	4
	Screw	Phillips-pan head (PPH) 4-40 x 1/4” stainless steel	4
KIT-PCM-STANDOFF-B-4 4 pc. brass hex PC/104 standoff kit	Standoff	Brass 5 mm hex, 0.600” long male/female 4-40	4
	Hex nut	4-40 x 0.095 thick, nickel finish	4
	Screw	Phillips-pan head (PPH) 4-40 x 1/4” stainless steel	4

10. Software Drivers

Go to www.winsystems.com for information on available software drivers.

Appendix A. Best Practices

The following paragraphs outline the best practices for operating the PCM-MIO-A-1 in a safe, effective manner, that does not damage the board. Read this section carefully.

Power Supply

Avoid Electrostatic Discharge (ESD)
Only handle the circuit board and other bare electronics when electrostatic discharge (ESD) protection is in place. Having a wrist strap and a fully grounded workstation is the minimum ESD protection required before the ESD seal on the product bag is broken.

Power Supply Budget

Evaluate your power supply budget. It is usually good practice to budget twice the typical power requirement for all of your devices.

Zero-load Power Supply

Use a zero-load power supply whenever possible. A zero-load power supply does not require a minimum power load to regulate. If a zero-load power supply is not appropriate for your application, then verify that the embedded system's typical load is not lower than the power supply's minimum load. If the embedded system does not draw enough power to meet the power supply's minimum load, then the power supply does not regulate properly and can cause damage to the PCM-MIO-A-1.

Use Proper Power Connections (Voltage)
When verifying the voltage, measure it at the power connector on the PCM-MIO-A-1. Measuring it at the power supply does not account for voltage drop through the wire and connectors.

The PCM-MIO-A-1 requires 5 V to operate. Verify the power connections. Incorrect voltages can cause catastrophic damage.

Power Harness

Minimize the length of the power harness. This reduces the amount of voltage drop between the power supply and the PCM-MIO-A-1.

Gauge Wire

Use the largest gauge wire that you can. Most connector manufacturers have a maximum gauge wire they recommend for their pins.

Contact Points

WINSYSTEMS boards mostly use connectors with gold finish contacts. Gold finish contacts are used exclusively on high-speed connections. Power and lower speed peripheral connectors may use a tin finish as an alternative contact surface. It is critical that the contact material in the mating connectors is matched properly (gold to gold and tin to tin). Contact areas made with dissimilar metals can cause oxidation/corrosion, resulting in unreliable connections.

Pin Contacts

Often the pin contacts used in cabling are not given enough attention. The ideal choice for a pin contact would include a design similar to Molex or Trifurcon designs, which provide three distinct points to maximize the contact area and improve connection integrity in high shock and vibration applications.

Power Down

Make sure that power has been removed from the system before making or breaking any connections.

Power Supply OFF—Always turn off the power supply before connecting to the I/O Module. Do not hot-plug the PCM-MIO-A-1 on a host platform that is already powered.

I/O Connections OFF—Turn off all I/O connections before connecting them to the PCM-MIO-A-1 or any other I/O cards. Connecting hot signals can cause damage whether the embedded system is powered or not.

Mounting and Protecting the I/O Module

To avoid damage, mount the PCM-MIO-A-1 properly. Standoff kits are available and recommended for use with the PCM-MIO-A-1.

Placing the PCM-MIO-A-1 on mounting standoffs—Be careful when placing the PCM-MIO-A-1 on the mounting standoffs. Sliding the board around until the standoffs are visible from the top can cause component damage on the bottom of the board.

Do not bend or flex the PCM-MIO-A-1—Bending or flexing can cause irreparable damage. Embedded computer modules are especially sensitive to flexing or bending around ball grid array (BGA) devices. BGA devices are extremely rigid by design, and flexing or bending the embedded computer module can cause the BGA to tear away from the printed circuit board.

Mounting holes—The mounting holes are plated on the top, bottom, and through the barrel of the hole. The mounting holes are connected to the embedded computer module's ground plane.

- Never use a drill or any other tool in an attempt to make the holes larger.
- Never use screws with oversized heads. The head could come in contact with nearby components causing a short or physical damage.
- Never use self-tapping screws; they compromise the walls of the mounting hole.
- Never use oversized screws that cut into the walls of the mounting holes.
- Always use all of the mounting holes. By using all of the mounting holes, you provide the support the embedded computer module needs to prevent bending or flexing.

Plug or unplug connectors only on fully mounted boards—Never plug or unplug connectors on a board that is not fully mounted. Many of the connectors fit rather tightly and the force needed to plug or unplug them could cause the embedded computer module to be flexed.

Avoid cutting the PCM-MIO-A-1—Never use star washers or any fastening hardware that cut into the PCM-MIO-A-1.

Avoid over-tightening of mounting hardware—Causing the area around the mounting holes to compress could damage interlayer traces around the mounting holes.

Use appropriate tools—Always use tools that are appropriate for working with small hardware. Large tools can damage components around the mounting holes.

Avoid conductive surfaces—Never allow the embedded computer module to be placed on a conductive surface. Many embedded systems use a battery to back up the clock-calendar and CMOS memory. A conductive surface such as a metal bench can short the battery causing premature failure.

Adding PC/104 to your Stack

Be careful when adding PC/104 boards to your stack—Never allow the power to be turned on when a PC/104 board has been improperly plugged onto the stack.

Conformal Coating

Conformal coating by any source other than WINSYSTEMS voids the product warranty and will not be accepted for repair by WINSYSTEMS. If such a product is sent to WINSYSTEMS for repair, it will be returned at

customer expense and no service will be performed. A WINSYSTEMS product conformally coated by WINSYSTEMS will be subject to regular WINSYSTEMS warranty terms and conditions.

Operations/Product Manuals

Every WINSYSTEMS product has an Operations manual or Product manual.

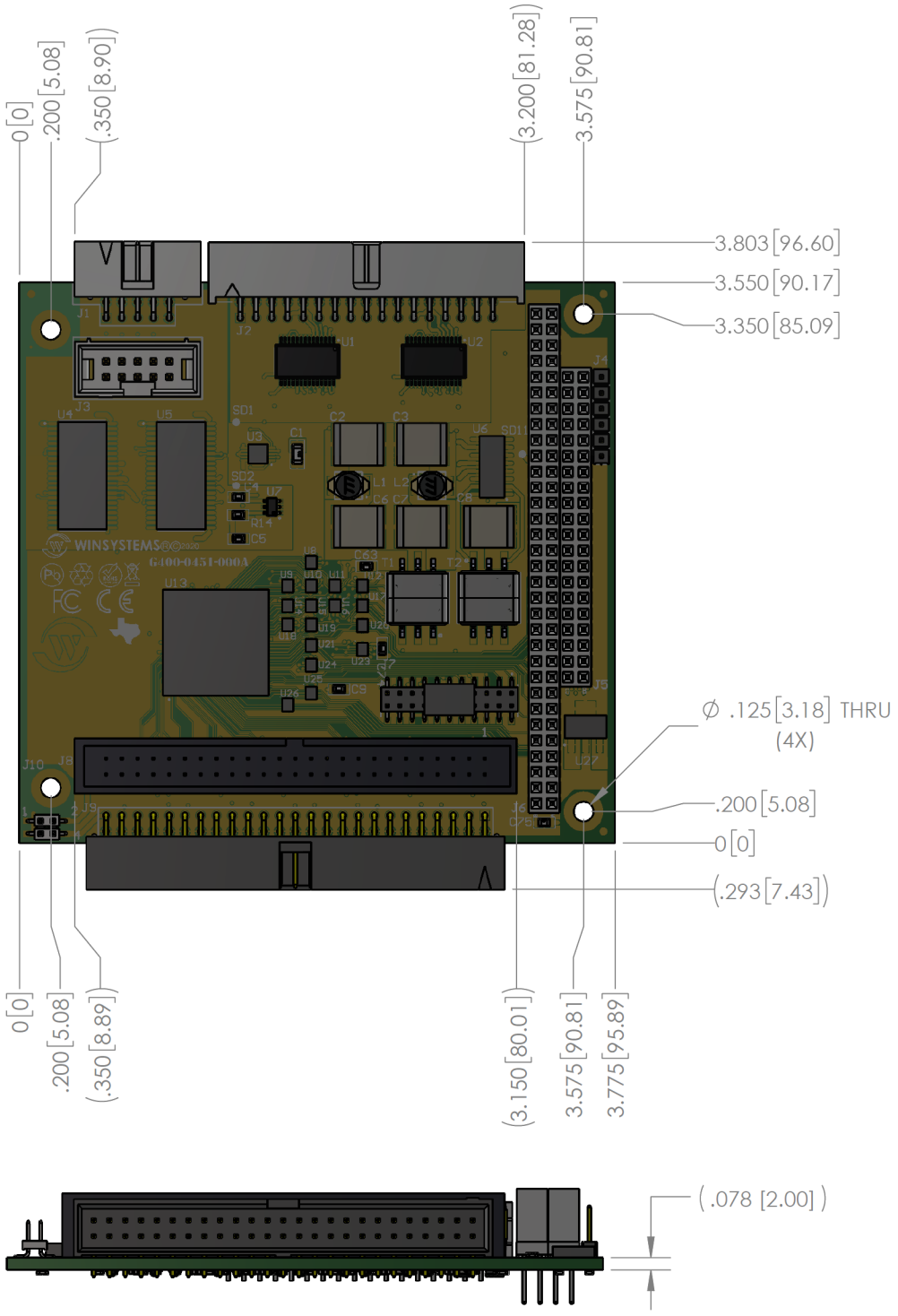
Periodic updates—Operations/product manuals are updated often. Periodically check the WINSYSTEMS website (https://www.winsystems.com) for revisions.

Check pinouts—Always check the pinout and connector locations in the manual before plugging in a cable. Many I/O modules have identical headers for different functions and plugging a cable into the wrong header can have disastrous results.

Contact an Applications Engineer—If a diagram or chart in a manual does not seem to match your system, or if you have additional questions, contact a WINSYSTEMS Applications Engineer at +1-817-274-7553.

Appendix B. Mechanical Drawings

The mechanical drawing for the PCM-MIO-A-1 is shown below.



Appendix C. Warranty Information

Full warranty information can be found at https://winsystems.com/company-policies/warranty/.