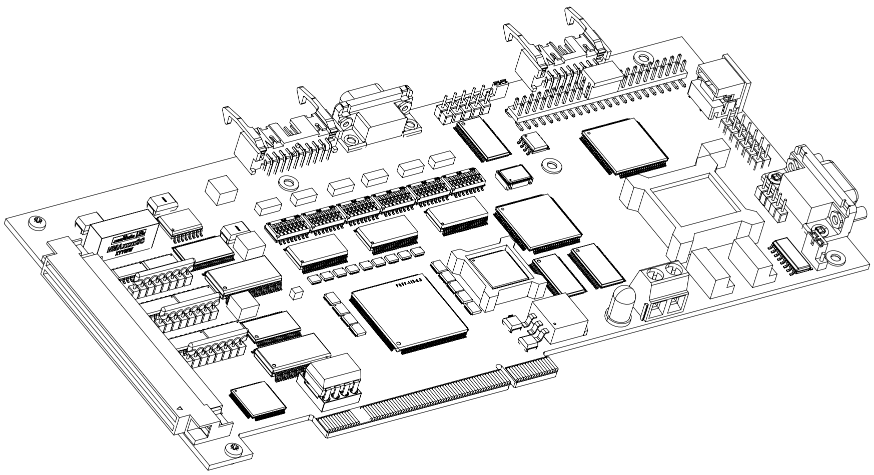


Magellan®
Motion Control IC

DK58420 Developer Kit
User Manual

Revision 1.7/ May 2024

Performance Motion Devices, Inc.

80 Central Street, Boxborough, MA 01719

www.pmdcorp.com

NOTICE

Warranty

Performance Motion Devices, Inc. warrants that its products shall substantially comply with the specifications applicable at the time of sale, provided that this warranty does not extend to any use of any Performance Motion Devices, Inc. product in an Unauthorized Application (as defined below). Except as specifically provided in this paragraph, each Performance Motion Devices, Inc. product is provided "as is" and without warranty of any type, including without limitation implied warranties of merchantability and fitness for any particular purpose.

Performance Motion Devices, Inc. reserves the right to modify its products, and to discontinue any product or service, without notice and advises customers to obtain the latest version of relevant information (including without limitation product specifications) before placing orders to verify the performance capabilities of the products being purchased. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgment, including those pertaining to warranty, patent infringement and limitation of liability.

Unauthorized Applications

Performance Motion Devices, Inc. products are not designed, approved or warranted for use in any application where failure of the Performance Motion Devices, Inc. product could result in death, personal injury or significant property or environmental damage (each, an "Unauthorized Application"). By way of example and not limitation, a life support system, an aircraft control system and a motor vehicle control system would all be considered "Unauthorized Applications" and use of a Performance Motion Devices, Inc. product in such a system would not be warranted or approved by Performance Motion Devices, Inc.

By using any Performance Motion Devices, Inc. product in connection with an Unauthorized Application, the customer agrees to defend, indemnify and hold harmless Performance Motion Devices, Inc., its officers, directors, employees and agents, from and against any and all claims, losses, liabilities, damages, costs and expenses, including without limitation reasonable attorneys' fees, (collectively, "Damages") arising out of or relating to such use, including without limitation any Damages arising out of the failure of the Performance Motion Devices, Inc. product to conform to specifications.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent procedural hazards.

Disclaimer

Performance Motion Devices, Inc. assumes no liability for applications assistance or customer product design. Performance Motion Devices, Inc. does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of Performance Motion Devices, Inc. covering or relating to any combination, machine, or process in which such products or services might be or are used. Performance Motion Devices, Inc.'s publication of information regarding any third party's products or services does not constitute Performance Motion Devices, Inc.'s approval, warranty or endorsement thereof.

Patents

Performance Motion Devices, Inc. may have patents or pending patent applications, trademarks, copyrights, or other intellectual property rights that relate to the presented subject matter. The furnishing of documents and other materials and information does not provide any license, express or implied, by estoppel or otherwise, to any such patents, trademarks, copyrights, or other intellectual property rights.

Patents and/or pending patent applications of Performance Motion Devices, Inc. are listed at https://www.pmdcorp.com/company/patents.

Related Documents

Magellan Motion Control IC User Guide

Complete description of the Magellan Motion Control IC features and functions with detailed theory of operations.

C-Motion Magellan Programming Reference

Descriptions of all Magellan Motion Control IC commands, with coding syntax and examples, listed alphabetically for quick reference.

MC58000 Electrical Specifications

For DC Brush, Brushless DC, microstepping, and pulse & direction Magellan MC58000 multi-axis motion control ICs.

MC55000 Electrical Specifications

For pulse & direction Magellan MC55000 multi-axis motion control ICs.

Atlas Digital Amplifier User Manual

Description of the Atlas Digital Amplifier electrical and mechanical specifications along with a summary of its operational features.

Table of Contents

This page intentionally left blank.

List of Figures

This page intentionally left blank.

1. Installation

1

In This Chapter

- Introduction
- Magellan Motion Control IC Family Overview
- Components List
- DK58420 Board
- Installation Overview
- Software Installation
- Recommended Hardware
- Preparing the Board for Installation
- Connection Summary
- Applying Power
- First Time System Verification
- Going Further with Pro-Motion

1.1 Introduction

The PMD DK58420 Developer Kit is an integrated board/software package that serves as an electrical and software design tool for building MC58420 and MC55420 based systems.

Two different developer kits are available as shown below:

Developer Kit P/N	ICs Supported	Installed IC	Motors Supported
DK58420	MC58420, MC58320, MC58220, MC58120	MC58420	DC Brush, Brushless DC, step motor
DK55420	MC55420, MC55320, MC55220, MC55120	MC55420	Pulse & direction

Both of the above developer kit versions share the same physical PC board. They differ in the specific type of chip that is installed in the board. To build a complete control system the DK board is combined with one or more motion axis amplifiers. For DK58420 PMD's Atlas Digital Amplifiers are directly supported via a plug-in connector. Alternatively, connectors are provided to integrate other off-the-shelf amplifiers, or user-built amplifier circuitry.

Because the installed chip for each developer's kit supports the maximum number of axes allowed, it can be used to develop systems based on chipsets with fewer axes simply by disabling the unused axes. Alternatively, because the CP chip is socketed in the DK58420, the MC58320, MC58220, or MC58120 IC can be purchased separately and substituted for the installed MC58420 IC. And similarly for the DK55420, the MC55320, MC55220, or MC55120 IC can be purchased separately and substituted for the installed MC55420 IC.

Throughout this manual the term MC58420 may be used to mean just the specific MC58420 IC, the four members of the MC58000 IC group (MC58420, MC58320, MC58220, and MC58120) or the eight members of the combined MC58000/MC55000 IC group (MC58420, MC58320, MC58220, MC58120, MC55420, MC55320, MC5220, and MC5120).

Throughout this manual the term MC58000 means just the four members of the MC58000 IC group (MC58420, MC58320, MC58220, and MC58120, and the term MC55000 means just the four members of the MC55000 IC group (MC55420, MC55320, MC55220, and MC55120).

1.2 Magellan Motion Control IC Family Overview

The following table presents a feature summary of the products in the Magellan Motion Control IC product family:

	MC58000 Series (Except MC58113)	MC55000 Series	MC58113 Series
# of axes	1, 2, 3, 4	1, 2, 3, 4	1+ (primary & aux channel encoder input)
Motor types supported	DC brush, brushless DC, step motor	Step motor	DC brush, brushless DC, step motor
Output format	SPI Atlas, PWM, DAC, Pulse & direction	Pulse & direction	SPI Atlas, PWM, DAC, Pulse & direction
Parallel host communication	✓	✓	
Serial host communication	✓	✓	✓
CAN 2.0B host communication	✓	✓	✓
SPI host communication			✓
Incremental encoder input	✓	✓	✓
Parallel word device input	✓	✓	
Index & Home signals	✓	✓	✓
Position capture	✓	✓	✓
Directional limit switches	✓	✓	✓
PWM output	✓		✓
Parallel DAC output	✓		
SPI Atlas interface	✓		✓
SPI DAC output	✓		✓
Pulse & direction output	✓	✓	✓
Digital current control	✓ (with Atlas)		✓
Field oriented control	✓ (with Atlas)		✓
Under/overvoltage sense	✓ (with Atlas)		✓
I^2T Current foldback	✓ (with Atlas)		✓
DC Bus shunt resistor control			✓
Overtemperature sense	✓ (with Atlas)		✓
Short circuit sense	✓ (with Atlas)		✓
Ground fault detection	✓ (with Atlas)		
Trapezoidal profiling	✓	✓	✓
Velocity profiling	✓	✓	✓
S-curve profiling	✓	✓	✓
Electronic gearing	✓	✓	✓
On-the-fly changes	✓	✓	✓
PID position servo loop	✓		✓
Dual biquad filters	✓		✓

	MC58000 Series (Except MC58113)	MC55000 Series	MC58113 Series
Dual encoder loop	✓ (multi-axis configurations only)		✓
Programmable derivative sampling time	✓		✓
Feedforward (accel & vel)	✓		✓
Data trace/diagnostics	✓	✓	✓
Motion error detection	✓	✓ (with encoder)	✓
Axis settled indicator	✓	✓ (with encoder)	✓
Analog input	✓	✓	✓
Programmable bit output	✓	✓	✓
Software-invertible signals	✓	✓	✓
User-defined I/O	✓	✓	
Internal Trace Buffer			✓
External RAM support	✓	✓	
Multi-chip synchronization	✓		✓
Chipset configurations	MC58420 (4 axes, 2 ICs) MC58320 (3 axes, 2 ICs) MC58220 (2 axes, 2 ICs) MC58120 (1 axis, 2 ICs) MC58110 (1 axis, 1 IC)	MC55420 (4 axes, 2 ICs) MC55320 (3 axes, 2 ICs) MC55220 (2 axes, 2 ICs) MC55120 (1 axis, 2 ICs) MC55110 (1 axis, 1 IC)	MC51113 (1+ axis, 1 IC) MC53113 (1+ axis, 1 IC) MC54113 (1+ axis, 1 IC) MC58113 (1+ axis, 1 IC)
IC Package: CP chip	MC58x20: 144 pin TQFP MC58110: 144 pin TQFP	MC55x20: 144 pin TQFP MC55110: 144 pin TQFP	100 pin TQFP
IC Package: IO chip	MC58x20: 100 pin TQFP MC58110: NA	MC55x20: 100 pin TQFP MC55110: NA	N/A
Motion control IC developer kit p/n's	DK58420 DK58320 DK58220 DK58120 DK58110	DK55420 DK55320 DK55220 DK55120 DK55110	DK51113 DK53113 DK54113 DK58113 DK58113S

1.3 Components List

The DK58420 and DK55420 Developer Kits contain the following components:

- 4-axis DK58420 Developer Kit board
- 100-Pin Connector to dual 50-pin header converter cable (3' length)
- USB to 9-pin serial cable

The following software and design materials are part of the DK58420 and DK55420 developer kits and can be downloaded from the PMD website. See Section 1.6, "Software Installation" for details:

- Pro-Motion Windows-based exerciser
- C-Motion Magellan Software Developer Kit
- PDFs of all documentation

1.4 DK58420 Board

The heart of the DK58420 and DK55420 developer kits is the DK58420 printed circuit board that contains interface and amplifier circuitry to allow various features of the MC58420 ICs to be accessed. Here is a summary of the features provided by the DK58420 board:

- Four axis full function motion control board
- Supports step, DC Brush, and Brushless DC motors
- Socketed CP and IO chips allow different MC58420 chips to be swapped out
- Interfaces to external Atlas, off-the-shelf, user-designed, or pulse & direction amplifiers
- RS-232, RS-485, and CANbus host communications
- Single DC-voltage supply
- Quadrature signal input with Index and Home capture
- Hall sensor, Home, limits, AxisIn and AxisOut signals
- Pulse & Direction signals with AtRest for use with external step motor amplifiers

1.5 Installation Overview

1. Before using the DK board, the software must be installed. See Section 1.6, “Software Installation” for instructions on installing the software.
2. For a normal installation of the DK58420 board you will need to configure the board. See Section 1.8, “Preparing the Board for Installation” for a description of configuring the board.
3. Next, connect the system’s motor windings, encoder(s), and sensors to operate the motion hardware. See Section 1.9, “Connection Summary” for details.
4. Connect the DK58420 board to the host PC via a Serial cable. This is described in Section 1.9.8, “Communication Connections.”
5. Once this hardware configuration is complete, the final step to finish the installation is to perform a functional test of the finished system. See Section 1.11, “First Time System Verification” for a description of this procedure.

Once these steps have been accomplished, the installation is complete, and the board is ready for operation.

1.6 Software Installation

The software distribution for the DK58420 developer kit is downloaded from the PMD website at the URL: https://www.pmdcorp.com/resources/software.

All software applications are designed to work with Microsoft Windows.

To install the software:

1. Go to the Software Downloads section of PMD’s website located at https://www.pmdcorp.com/resources/software and select download for "Developer Kit Software".

2 After selecting download you will be prompted to register your DK, providing the serial # for the DK and other information about you and your motion application.

3 After selecting submit the next screen will provide a link to the software download. The software download is a zip file containing various installation programs. Select this link and downloading will begin.

4 Once the download is complete extract the zip file and execute the desired install programs from the list below. Every first-time installation should install Pro-Motion, and at least one of the two SDK options. However you may install both SDKs if desired. When installing the SDKs you will be given the option to download the documentation and/or the complete SDK content.

- Pro-Motion – an application for communicating to, and exercising PMD ICs, modules, or boards.
- C-Motion Magellan SDK – an SDK (Software Developer Kit) for creating motion applications using the C/C++ programming language for PMD products that utilize a direct Magellan or Juno formatted protocol.
- C-Motion PRP SDK – an SDK for creating PC and downloadable user code for systems utilizing either a PRP (PMD Resource Access Protocol) protocol device or a Magellan/Juno protocol device. C-Motion PRP is also used in motion applications that will use the .NET (C#, VB) programming languages.

*Adobe Acrobat Reader is required for viewing these files. If the Adobe Acrobat Reader is not installed on your computer, it may be freely downloaded from http://www.adobe.com.

Here is more information on each of these software packages.

1.6.1 Pro-Motion

Pro-Motion is a sophisticated, easy-to-use exerciser program which allows all Magellan IC parameters to be set and/or viewed, and allows all features to be exercised. Pro-Motion features include:

- Motion oscilloscope graphically displays motion control parameters in real-time
- AxisWizard to automate axis setup and configuration
- Project window for accessing motion resources and connections
- Ability to save and load settings
- Distance, time, and electrical units conversion
- Frequency sweep and bode plot analysis tools
- Motor-specific parameter setup
- Axis shuttle performs continuous back and forth motion between two positions
- C-Motion Engine monitor debug window
- C-Motion Engine user application code download

1.6.2 C-Motion

C-Motion provides a convenient set of callable routines comprising the C language code required for controlling Magellan ICs. C-Motion includes the following features:

- Magellan axis virtualization
- Ability to communicate to multiple PMD motion boards or modules
- Ability to communicate via PC/104 bus, serial, CANbus, Ethernet, SPI (Serial Peripheral Interface), or 8/16 bit parallel bus
- Provided as source code, allowing easy compilation & porting onto various run-time environments including a PC, microprocessor, embedded board, or C-Motion Engine
- Can be easily linked to any C/C++ application

There are three different versions of C-Motion; C-Motion Magellan, C-Motion PRP, and C-Motion PRP II. C-Motion Magellan is used with PMD products that utilize a direct Magellan or Juno formatted protocol. C-Motion PRP is used in systems that support both Magellan/Juno protocol device and PRP (PMD Resource Access Protocol) protocol devices. C-Motion PRP is also used in motion applications that will use the .NET (C#, VB) programming languages. C-Motion PRP II is used with ION/CME N-Series Digital Drives.

C-Motion Magellan is described in the *C-Motion Magellan Programming Reference,* C-Motion PRP is described in the *C-Motion PRP Programming Reference,* C-Motion PRP II is described in the *C-Motion PRP II Programming Reference.*

1.6.3 .NET Language Support

A complete set of methods and properties is provided for developing applications in Visual Basic and C# using a dynamically loaded library (DLL) containing PMD library software. The DLL may also be used from any language capable of calling C language DLL procedures, such as Labview, but no special software support is provided.

Includes the following features:

- Magellan axis virtualization
- Ability to communicate to multiple PMD motion boards or modules
- Ability to communicate via PC/104 bus, serial, CAN, Ethernet, SPI, or 8/16-bit parallel bus
- Provided as a single DLL and Visual Basic .NET source code for easy porting onto various PC environments

1.7 Recommended Hardware

To install a DK58420 board the following hardware is recommended.

- Intel (or compatible) processor, 1 Gbyte of available disk space, 256 MB of available RAM, and a CD ROM drive. The supported PC operating systems are Windows XP, Vista, Windows 7, and Windows 8.\
- One to four Atlas amplifiers, or other PWM input, analog-input, or pulse and direction amplifiers depending on the DK being used and control mode used. Amplifiers are motor-type specific driving either DC Brush, Brushless DC, or step motors.
- One to four step motors, DC Brush, or Brushless DC motors. These motors may or may not provide encoder position feedback. Servo motors must have encoder feedback, while for step motors encoder feedback is optional.

- A single 100-pin header-type connector to interface to your motion hardware such as encoders, limit switches, and other connections. Most users will use the provided 100-pin connector which converts to two 50-pin header connectors. See Section 3.2.3, “Motion Peripherals Connector (J4)” for details on signal breakout hardware that can be used with this connector.
- Additional connectors as required to connect to amplifiers and motor peripherals. If Atlas amplifiers are used the DK58420 supports a direct connection to Atlas developer kits via a DB9 connector.
- Power supply, power cable, and communication cable.

1.8 Preparing the Board for Installation

To prepare your board for installation various user-settable hardware options should be checked. To aid with this the following diagram shows the location of various DK58420 board elements including resistor packs RS1, RS2, RS3, motor type switches, and other components such as connectors.

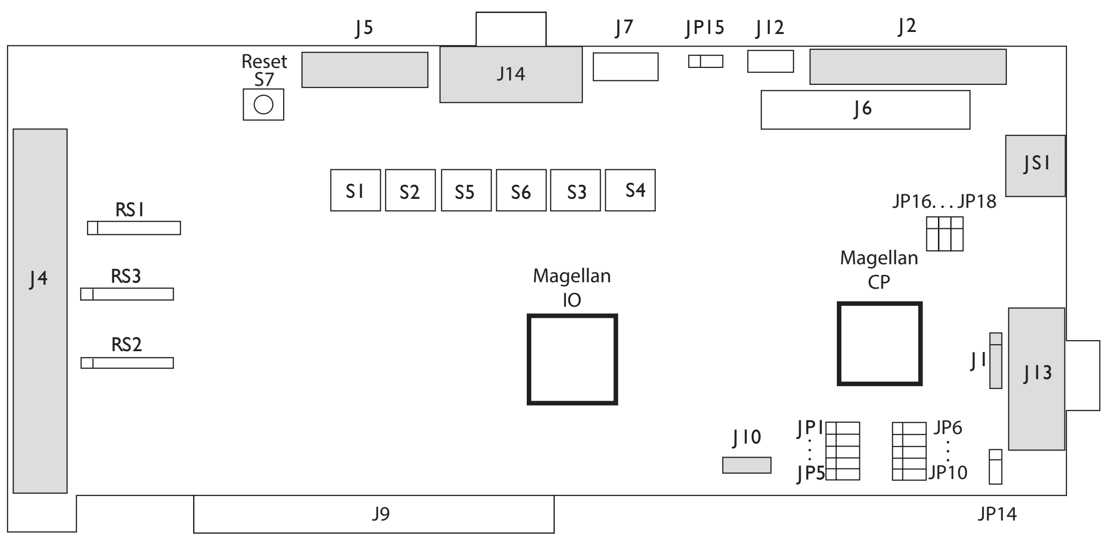


Figure 1-1: Location of Various DK58420 Board Elements

1.8.1 Resistor Packs

The first step to configuring the DK58420 board is to install or remove the encoder resistor packs. See the table below for information.

If you are not using encoders in your system you may skip to the next section.

Item	How to Set	Description
Resistor packs RS1, RS2, RS3	Installed *this is the default setting of resistor packs RS1-RS3*	If you are using differential encoder connections leave these resistor packs installed.
	Removed	If you are using single-ended encoder connections, remove the resistor packs.

1.8.2 Motor Type Switch Settings

When using the DK55420 only pulse and direction motors are used and it is not necessary to set switches related to motor type.

When using the DK58420 it is possible to support any combination of DC Brush, Brushless DC, microstepping, and pulse & direction motors all on the same board and dip switches 5 and 6 must be set to indicate the motor type that will be used. When configuring the dip switches and connecting your motors to the Developer's Kit Board, the following information may be helpful:

- *Brushless DC* means the board expects to connect to a Brushless DC motor with an encoder, and possibly with Hall sensors. With this connection, the Developer's Kit board performs the commutation and outputs a multi-phase signal.
- *Pulse and direction* means the board expects to connect to a step motor which uses an Atlas step motor amplifier or standard pulse and direction amplifier. Quadrature feedback is optional with this type of motor.
- Microstepping means the Developer's Kit board expects to connect to a step motor amplifier with multi-phase signal command input, providing 2 or 3 phases per axis. Quadrature feedback is optional with this type of motor.
- *DC Brush* means the boards expects to connect to a DC Brush motor with an encoder, or an externally commutated Brushless DC motor (amplifier performs commutation). With this motor type the board outputs one phase per axis.

1.8.2.1 Motor Type Switch Settings (for DK58420 only)

When referring to the table below the switch up position is relative to the bottom of the board where the PCI connector is located. The up position on the switch is marked **on**.

Item	Switches	Description			
Dip switch S5	S5-1 S5-2 S5-3	Axis #1 Motor type setting Set S5 1-3 dip switches according to the motor type you will be using on axis #1			
		5-1	5-2	5-3	Axis #1
		up	up	up	Brushless DC (3 phase)
		down	up	up	Closed loop stepper
		up	down	up	Microstepping (3 phase)
		down	down	up	Microstepping (2 phase)
		up	up	down	Pulse & direction
		down	down	down	DC Brush (default setting)
Dip switch S5	S5-5 S5-6 S5-7	Axis #2 Motor type setting Set S5 5-7 dip switches according to the motor type you will be using on axis #2			
		5-5	5-6	5-7	Motor type setting
		up	up	up	Brushless DC (3 phase)
		down	up	up	Closed loop stepper
		up	down	up	Microstepping (3 phase)
		down	down	up	Microstepping (2 phase)
		up	up	down	Pulse & direction
		down	down	down	DC Brush (default setting)
Dip switch S6	S6-1 S6-2 S6-3	Axis #3 Motor type setting Set S6 1-3 dip switches according to the motor type you will be using on axis #3			
		6-1	6-2	6-3	Motor type setting
		up	up	up	Brushless DC (3 phase)
		down	up	up	Closed loop stepper
		up	down	up	Microstepping (3 phase)
		down	down	up	Microstepping (2 phase)
		up	up	down	Pulse & direction
		down	down	down	DC Brush (default setting)

Item	Switches	Description			
Dip	S6-5	Axis #4 Motor type setting			
switch	S6-6	Set S6 5-7 dip switches according to the motor type you will be using on axis #4			
S6	S6-7	6-5	6-6	6-7	Motor type setting
		up	up	up	Brushless DC (3 phase)
		down	up	up	Closed loop stepper
		up	down	up	Microstepping (3 phase)
		down	down	up	Microstepping (2 phase)
		up	up	down	Pulse & direction
		down	down	down	DC Brush (default setting)

Unconnected motors can be left at the default setting of *DC Brush*.

1.8.3 Atlas Amplifier Operation

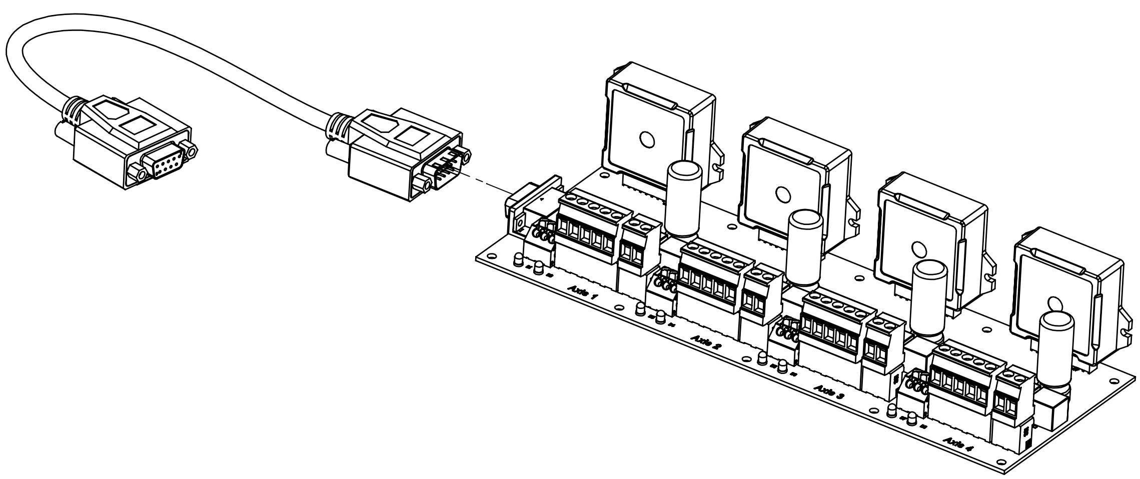


Figure 1-2: Connecting DB9 Cable to DK Board

If you are using Atlas amplifiers with your DK, jumpers JP16, JP17 and JP18 should be set as follows:

Item	Configuration	Description
JP16, JP17, JP18	jumpers installed at 2-3 for all three jumpers	To operate the DK58420 with Atlas amplifiers jumpers should be installed connecting the 2-3 pins for the JP16, JP17, and JP18 jumpers. See Figure 1-1 for location of these jumpers.
	jumpers installed at 1-2	To operate the DK58420 with non-Atlas external amplifiers jumpers should be installed connecting the 2-3 pins for the JP16 and JP17 jumpers, and JP18 should be connected at 1-2. See Figure 1-1.

1.9 Connection Summary

The following sections summarize the connections you should make for various motor types. The connections are broken into two overall groups, connections required when Atlas amplifiers are used, and connections when other amplifiers are used. You may use Atlas amplifiers for some axes and non-Atlas amplifiers for other axes. In addition, you may use different motor types on each axes.

1.9.1 DC Brush Motor With Atlas Amplifier

The following table summarizes connections to the DK58420 Developer Kit board when DC Brush motors are used with an Atlas amplifier.

Atlas amplifier connections are made via J14, the Atlas DK bus connector. See the *Atlas Digital Amplifier User Manual, Appendix A* for detailed instructions on connecting Atlas amplifiers.

All other connections are made through connector J4, the primary 100-pin connector indicated in Figure 1-1. See Section 3.2.3, "Motion Peripherals Connector (J4)" for a detailed list of connections.

Signal Category	Signal Description
Encoder input signals: *(per axis)*	A quadrature channel input B quadrature channel input Index pulse channel input
Amplifier signals:	SPI Atlas bus
Other control signals: *(optional, per axis)*	Home signal input Limit switch inputs AxisIn input AxisOut output
Miscellaneous signals:	Digital GND +5 V (for encoder power)

1.9.2 Brushless DC Motor With Atlas Amplifier

The following table summarizes connections to the DK58420 Developer Kit board when Brushless DC motors are used with an Atlas amplifier.

Atlas amplifier connections are made via J14, the Atlas DK bus connector. See *Atlas Digital Amplifier User Manual, Appendix A* for detailed instructions on connecting Atlas amplifiers.

All other connections are made through connector J4, the primary 100-pin connector. See Section 3.2.3, "Motion Peripherals Connector (J4)" for detailed signal descriptions.

Signal Category	Signal Description
Encoder input signals: (per axis)	A quadrature channel input B quadrature channel input Index pulse channel input
Amplifier signals:	SPI Atlas bus
Hall inputs:	Hall (phase A) Hall (phase B) Hall (phase C)
Other control signals: (optional, per axis)	Home signal channel input Positive limit switch input Negative limit switch input AxisIn input AxisOut output
Miscellaneous signals:	GND +5 V (for encoder power)

1.9.3 Step Motor With Atlas Amplifier

The following table summarizes connections to the DK58420 Developer Kit board when two-phase step motors are used with an Atlas amplifier.

Atlas amplifier connections are made via J14, the Atlas DK bus connector. See *Atlas Digital Amplifier User Manual, Appendix A* for detailed instructions on connecting Atlas amplifiers.

All other connections are made through connector J4, the primary 100-pin connector, indicated on Figure 1-1. See Section 3.2.3, "Motion Peripherals Connector (J4)" for detailed signal descriptions.

Signal Category	Signal Description
Encoder input signals: *(per axis)*	A quadrature channel input B quadrature channel input Index pulse channel input
Amplifier signals:	SPI Atlas bus
Other control signals: *(optional, per axis)*	Home signal channel input Positive limit switch input Negative limit switch input AxisIn input AxisOut output
Miscellaneous signals:	GND +5 V (for encoder power)

1.9.4 Non-Atlas DC Brush Motor Connections

The following table summarizes connections to the DK58420 Developer Kit board when DC-brush motors are used with a non-Atlas format amplifier.

All connections are made through connector J4, the primary 100-pin connector indicated in the figure above. See Section 3.2.3, "Motion Peripherals Connector (J4)" for a detailed list of connections.

Signal Category	Signal Description
Encoder input signals: *(per axis)*	A quadrature channel input B quadrature channel input Index pulse channel input
Amplifier output signals: *(per axis, if PWM sign, magnitude used)*	PWM direction PWM magnitude
Amplifier output signals: *(per axis, if PWM 50/50 used)*	PWM magnitude
Amplifier output signals: *(per axis, if analog output used)*	Analog out (DAC output)
Other control signals: *(optional, per axis)*	Home signal input Limit switch inputs AxisIn input AxisOut output
Miscellaneous signals:	Digital GND +5 V (for encoder power)

1.9.5 Non-Atlas Brushless DC Motors

The following table summarizes connections to the DK58420 Developer Kit board when Brushless DC motors are used with a non-Atlas format amplifier. All of these connections are made through connector J4, the primary 100-pin connector. See Section 3.2.3, "Motion Peripherals Connector (J4)" for detailed signal descriptions.

Signal Category	Signal Description
Encoder input signals: (per axis)	A quadrature channel input B quadrature channel input Index pulse channel input
Amplifier output signals: (per axis, if PWM 50/50 used)	PWM magnitude (phase A) PWM magnitude (phase B) PWM magnitude (phase C)
Amplifier output signals: (per axis, if analog output used)	Analog out (phase A) Analog out (phase B)
Hall inputs:	Hall (phase A) Hall (phase B) Hall (phase C)
Other control signals: (optional, per axis)	Home signal channel input Positive limit switch input Negative limit switch input AxisIn input AxisOut output
Miscellaneous signals:	GND +5 V (for encoder power)

1.9.6 Non-Atlas Pulse & Direction Motor Connections

The following table summarizes connections to the DK58420 Developer Kit board when pulse & direction interface step motors are used. All connections are made through connector J4, the primary 100-pin connector, indicated on Figure 1-1. See Section 3.2.3, "Motion Peripherals Connector (J4)" for detailed signal descriptions.

Signal Category	Signal Description
Encoder input signals: (optional, per axis)	A quadrature channel input B quadrature channel input Index pulse channel input
Amplifier output signals:	Pulse Direction
Other control signals: (optional, per axis)	AtRest signal output Home signal channel input Positive limit switch input Negative limit switch input AxisIn input AxisOut output
Miscellaneous signals:	GND +5 V (for encoder power)

1.9.7 Non-Atlas Microstepping or Closed Loop Stepper Motor Connections

The following table summarizes connections to the DK58420 Developer Kit board when multi-phase connections to step motors are used with a non-Atlas format amplifier. All of these connections are made through connector J4, the primary 100-pin connector. See Section 3.2.3, "Motion Peripherals Connector (J4)" for detailed signal descriptions.

Signal Category	Signal Description
Encoder input signals: (per axis)	A quadrature channel input B quadrature channel input Index pulse channel input
Amplifier output signals: (per axis, if PWM sign, magnitude used)	PWM magnitude (phase A) PWM magnitude (phase B) PWM direction (phase A) PWM direction (phase B)
Amplifier output signals: (per axis, if PWM 50/50 used)	PWM magnitude (phase A) PWM magnitude (phase B)
Amplifier output signals: (per axis, if analog output used)	Analog out (phase A) Analog out (phase B)
Other control signals: (optional, per axis)	Home signal channel input Positive limit switch input Negative limit switch input AxisIn input AxisOut output
Miscellaneous signals:	GND +5 V (for encoder power)

1.9.8 Communication Connections

The included USB to DB9 cable should be used to connect the PC to the DK board's serial port connector. At the DK board the serial cable should connect to J13. See Figure 1-1 to locate the J13 connector.

The DK board's default serial port baud rate is 57,600 bits per second. This rate should work with most PCs. To set the baud rate to a different value use the S2 switch settings described in Section 2.3.2, "Serial Transceiver" of this manual. If you still have trouble communicating to the board using the serial port contact PMD for assistance.

1.9.9 Power Connections

1.9.9.1 Power For The DK Board

The final electrical installation step is to connect power to the DK board and amplifiers.

The power connection for the DK58420 board is shown in the table below.

Signal	Connector	Description
+5V	J10	J10 is a two-signal (+5V, Gnd) terminal-screw-style connector that provides five volts to the DK board. See Figure 1-1 for location of J10 connector. Use a 1.0 Amp supply or greater to ensure adequate power.

1.9.9.2 Power For The Amplifiers

Whether using Atlas amplifiers or another amplifier type you will need to apply power to the amplifiers. To do so, carefully follow the instructions provided for your amplifier to safely connect power and make other necessary connections such as ground or case/shield connections.

In the case of Atlas amplifiers the input voltage range is 12-56 volts. Consult the *Atlas Digital Amplifier User Manual* for additional information.

1.10 Applying Power

Once you have made the necessary power and signal connections to your external amplifiers, motors, motor encoders, and other signals required for your application, hardware installation is complete and the board is ready for operation.

Most systems will first power the DK board. Upon power up, the board will be in a reset condition. In this condition the motor output signals provided by the DK board will send a “zero command” signal to the amplifier.

Next, amplifier power is enabled. After power is applied the motors should remain stationary. If the motors move or jump, power down the board and check the amplifier and other connections. If anomalous behavior is still observed, call PMD for assistance.

1.11 First Time System Verification

The first time system verification procedure summarized below has two overall goals. The first is to connect the DK board with the PC that is being used so that they are communicating properly, and the second is to initialize each axis of the system and bring it under stable control capable of making trajectory moves. While there are many additional capabilities that Pro-Motion and the DK board provide, these steps will create a foundation for further, successful exploration and development.

Here is a summary of the steps that will be used during first time system verification. Each of these steps will be described below in a separate section.

1. Initiate Pro-Motion and establish communication between the PC and the board using the serial link.
2. Run Pro-Motion’s Axis wizard for each axis of your system to initialize parameters such as encoder direction and safe servo parameters (if using a servo motor).
3. Execute a simple trajectory profile on each axis demonstrating that it is operating correctly and under stable control.

During this first time system setup you may find it useful to refer to other PMD manuals such as the *Magellan Motion Control IC User Guide* to familiarize yourself with operation of the Magellan Motion Control IC, which lies at the heart of all PMD Motion boards.

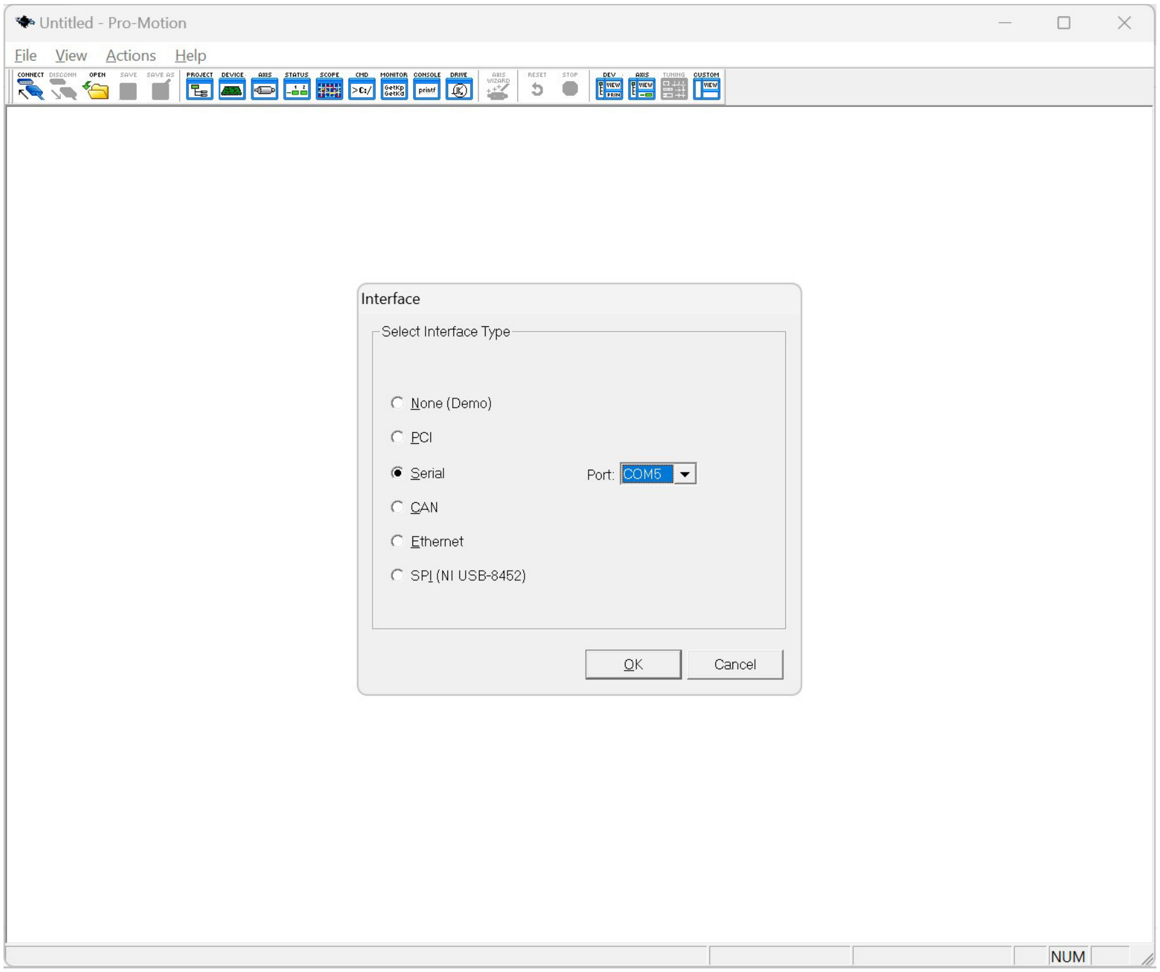
1.11.1 Establishing Communications

To establish serial communications:

1. Make sure the DK58420 board is powered and connected to the PC via the USB to DB9 serial cable.
2. Launch the Pro-Motion application.

   When Pro-Motion is launches you will be prompted with an Interface selection window. A typical screen view when first launching Pro-Motion appears below.

The purpose of the Interface dialog box is to indicate to Pro-Motion how your PMD controller is connected to the PC. It provides various selectable communication options such as serial, CAN, Ethernet.



3 Click Serial and view the available COM ports listed in the Port field. If you know which COM port your 3-pin programming cable is connected to, select it.

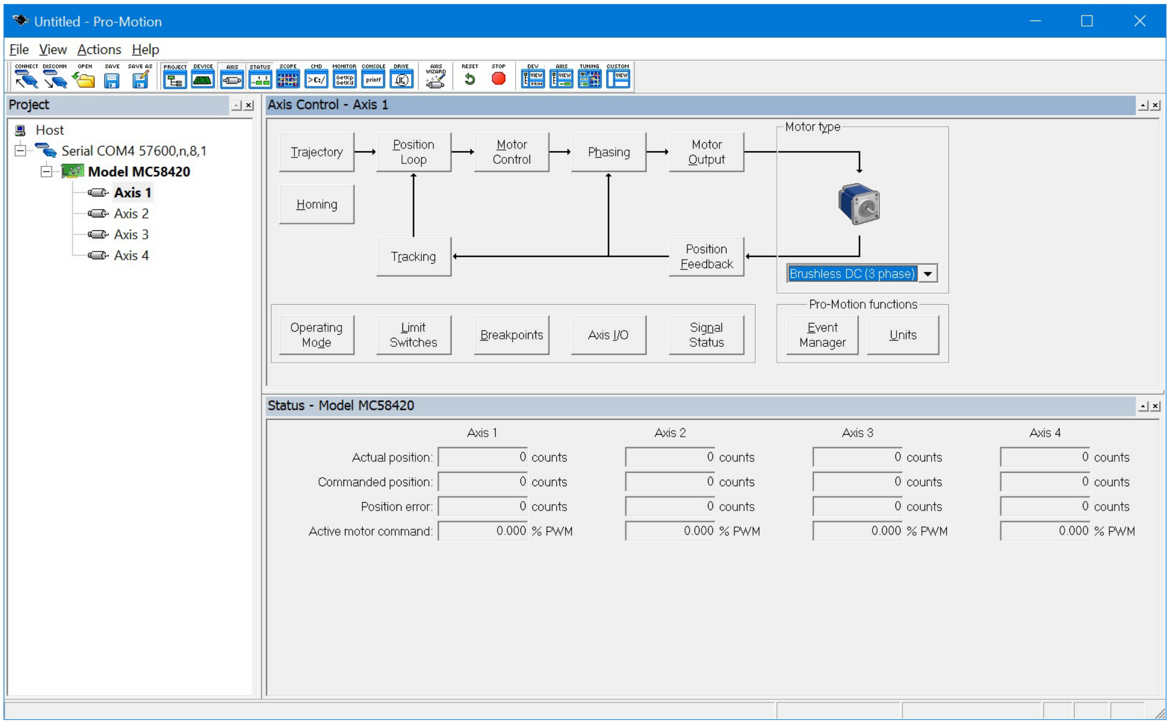
If you are not sure which of the listed COM ports is the correct one, you may use the following procedure:

- First, unplug the 3-pin cable from your USB port and exit the Interface dialog box. Now re-enter the Interface dialog box by clicking "Connect" which is an icon at the far left of the top icon bar. Select Serial, and view the COM port list. Record the list of COM ports.
- Next plug the 3-pin cable back into the USB port and once again exit and re-enter the Interface dialog box. Select Serial and now when you view the COM port list you should see a new COM port listed. This is the COM port that is connected via the 3-pin cable provided with the DK.
- Select this COM port and hit the OK button.

The Serial Port dialog box displays with default communication values of 57,600 baud, no parity, 1 stop bit, and point to point protocol.

4 Click OK without changing any of these settings.

If serial communication is correctly established, after a brief pause a set of object graphics loads into the Project window to the left, as shown in the following figure.



If serial communications are not correctly established, a message appears indicating that an error has occurred. If this is the case, recheck your connections and repeat from step 1.

1.11.2 Running the Axis Wizard

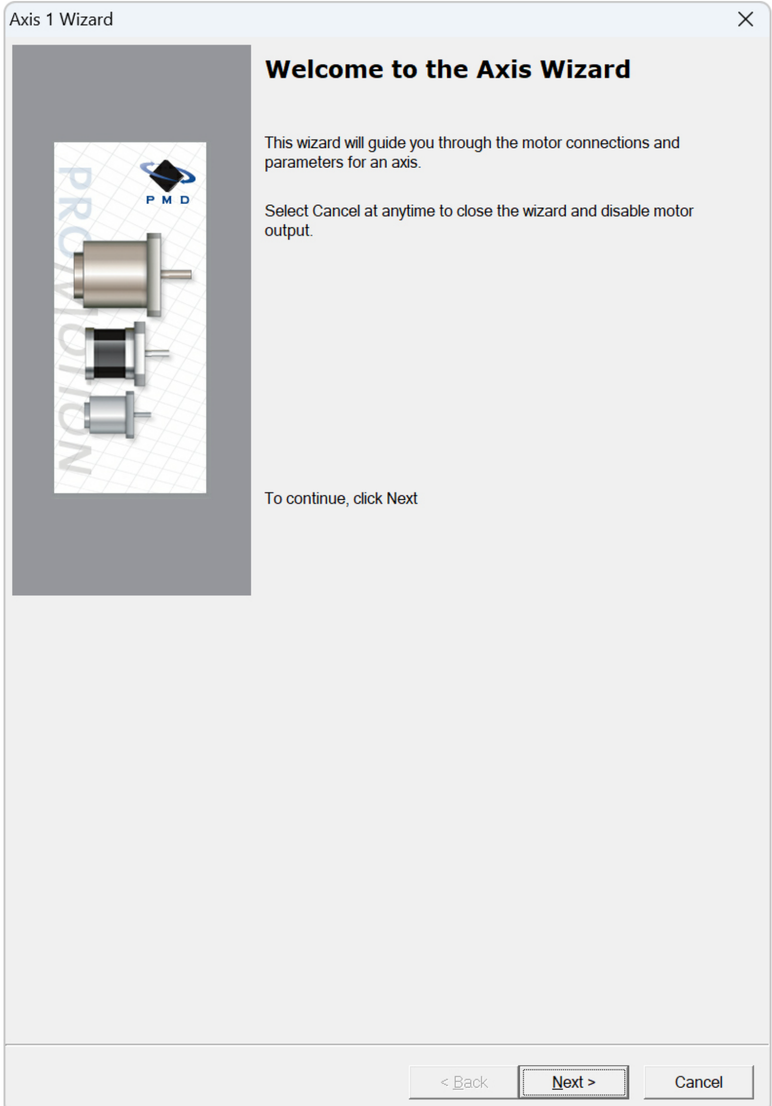
The next step is to initialize each axis in your system, thereby verifying correct encoder feedback connections (if an encoder is used), and other functions. All of this can be conveniently accomplished using Pro-Motion's Axis Wizard.

Before selecting the Axis Wizard icon however you should select the axis # to initialize. This is accomplished via the Project window, which is located to the left, by selecting the desired axis. After clicking on the desired axis it will become highlighted and the Axis Control Window title will change to reflect the newly selected axis #. If the axis you want to operate on is already highlighted there is no need to select the axis.

Once the desired axis to initialize has been selected you should:

1 Click the Axis Wizard toolbar button which is located right of center in the row of icons toward the top of the window.

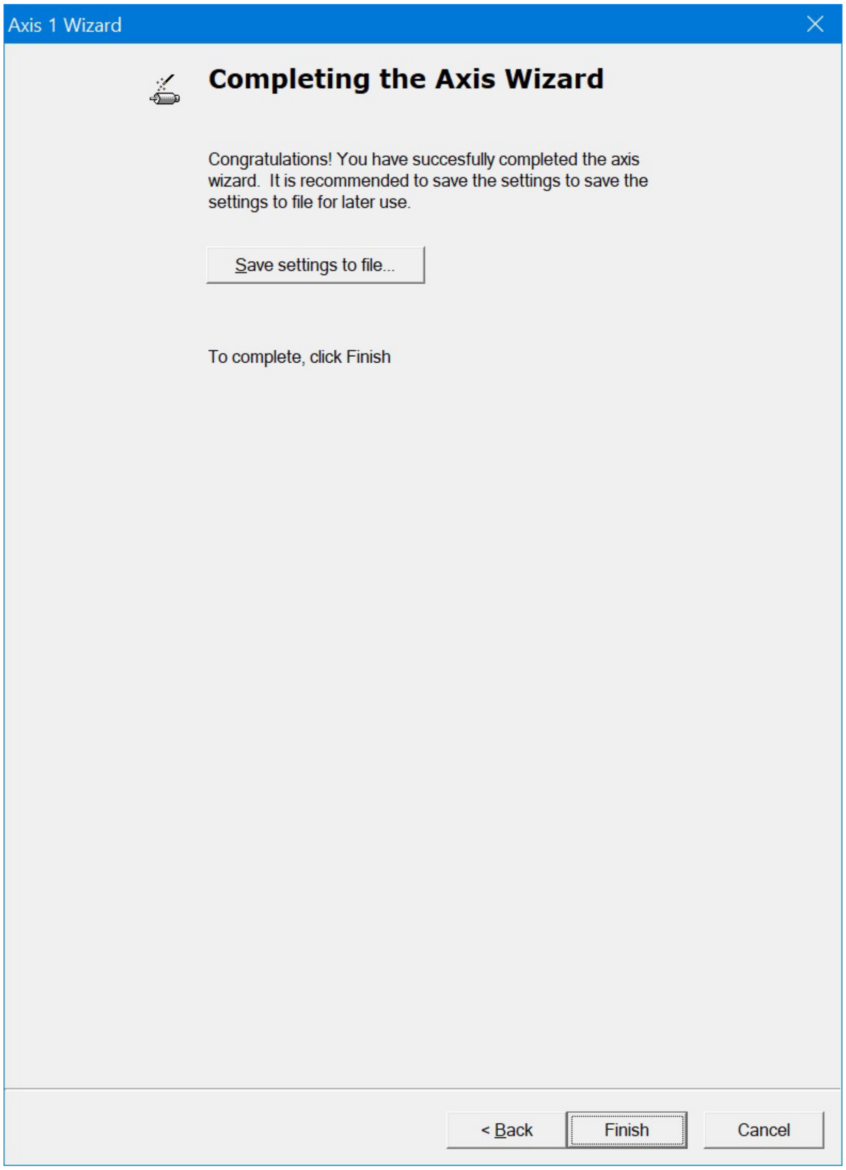
The Axis Wizard initialization window appears.



2 Click Next and follow the Axis Wizard instructions for each page of the axis initialization process.

A typical axis wizard sequence takes 5-10 minutes.

3 The last Axis wizard screen allows you to save the various control parameters you have specified while in the Axis Wizard. This is convenient because it allows you to quickly load your control settings and make them active without the need to re-run the Axis Wizard.



4 To save the Axis Wizard settings select "Save settings to file..." and specify a destination directory and file name. The Axis Wizard will create the file with your parameters loaded into it.

5 When you have specified a file name and saved your settings select Finish at the bottom of the screen. You will now be returned to the main Pro-Motion screen.

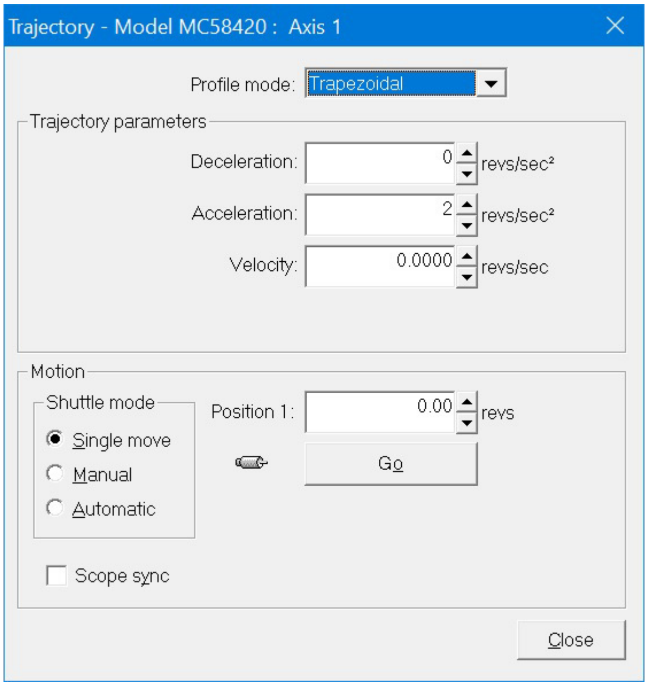
At this point you have the choice of selecting a new axis to initialize via the Axis Wizard, or to continue working with the already-selected axis and proceed to Section 1.11.3, "Exercising The Motor." Most users prefer to initialize and then exercise each axis immediately, but this is not required and is ultimately your choice.

1.11.3 Exercising The Motor

The next section will provide instructions on exercising your motor and verifying that it is reacting correctly to programmed commands.

To perform a simple move:

1 Click the Trajectory button in the Axis Control window. The Trajectory dialog box appears, shown below.



2 In the Profile mode list, select Trapezoidal, and in the Motion box select Single move.

3 Enter motion profiles for deceleration, acceleration, velocity, and destination position (Position 1) that are safe for your system and will demonstrate proper motion. The units of these parameters should match the units you requested earlier in the Axis Wizard Setup process. If you would like to change the units you can do this by going to the Axis Control Window and clicking the Units box which is to the far lower right on that window. You should then exit and re-enter the Trajectory dialog box for the units change to be visible.

4 Click Go and confirm that the motion occurred in a stable and controlled fashion.

Congratulations! First-time system verification for this axis is now complete.

Section 1.12, "Going Further with Pro-Motion" provides a guided tour of frequently used features of Pro-Motion. You may find this helpful if you are new to Pro-Motion and PMD's Magellan Motion Control IC, but taking the guided tour is optional and not required to initialize your motor or your setup. Having completed the above sequence your motor is fully initialized and ready for operation.

1.12 Going Further with Pro-Motion

In this section we provide more information on Pro-Motion to help familiarize you with its most commonly used features. We will cover the following areas:

- Restarting a Saved Project
- Axis Control Window
- Trace & Motion Oscilloscope
- Adjusting The Position Loop Parameters
- Troubleshooting Suggestions

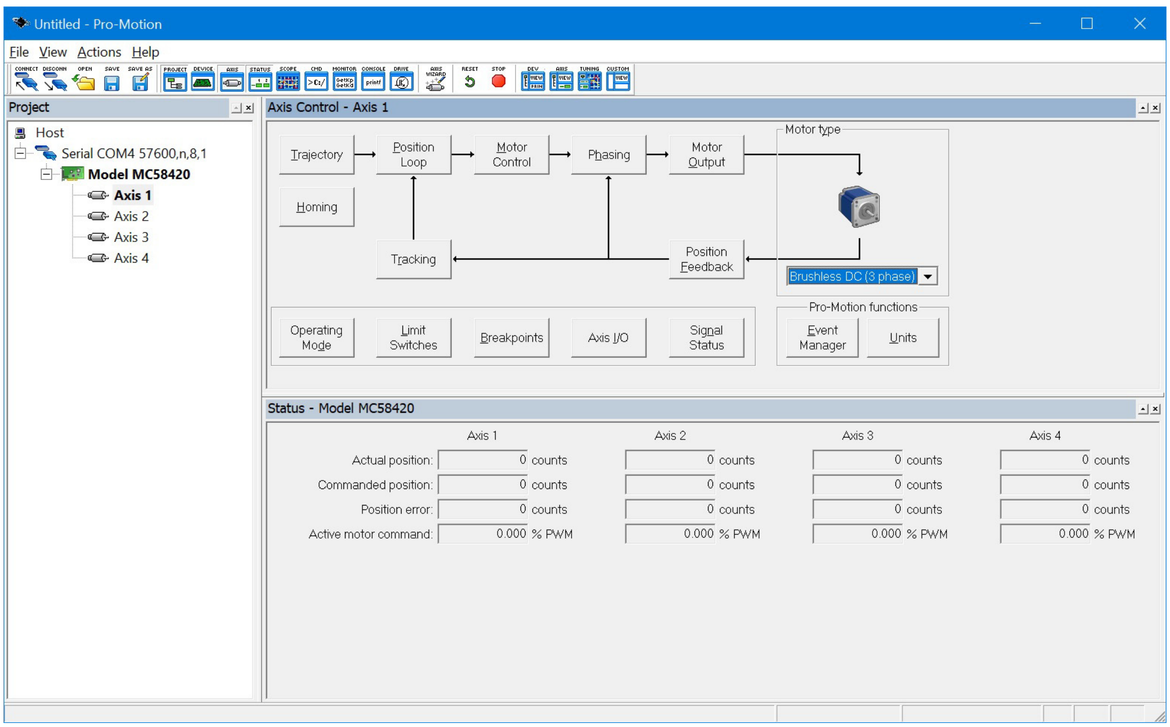
1.12.1 Re-Starting A Saved Project

Project files, which have an extension of ".PMD", are an important part of using Pro-Motion. This section provides information on how to save and load projects.

While using the Axis Wizard, at the very last screen, you were given the option of saving the configuration. If you specified a file and saved the configuration a project file was created. Note that what was saved at that time was the configuration of all axes. This is true whether or not you had initialized all axes. In addition to axis control parameters certain other system characteristics were saved in the project file such as the position of Pro-Motion Windows and other Pro-Motion settings such as display units.

Named Projects are useful because they allow you to store multiple configurations and later load up these saved configurations. In addition to saving a project at the end of the Axis Wizard you can save a project at any time while working with Pro-Motion by specifying File/Save Project or File/Save Project As from the top menu.

If you are starting a new Pro-Motion session and the PMD controller is in a reset condition (which will be the case after first powering it on) when you launch Pro-Motion and establish communications the screen will look as shown below:



To call up a saved project select File and Open Project. You will be asked to specify a project file. Once you specify a file and select Open in the dialog box the script will be input by Pro-Motion and the content sent to the PMD controller unit. In the event that there are errors during input, or errors that occur as the PMD controller receives and processes the saved content, a dialog box will appear indicating this.

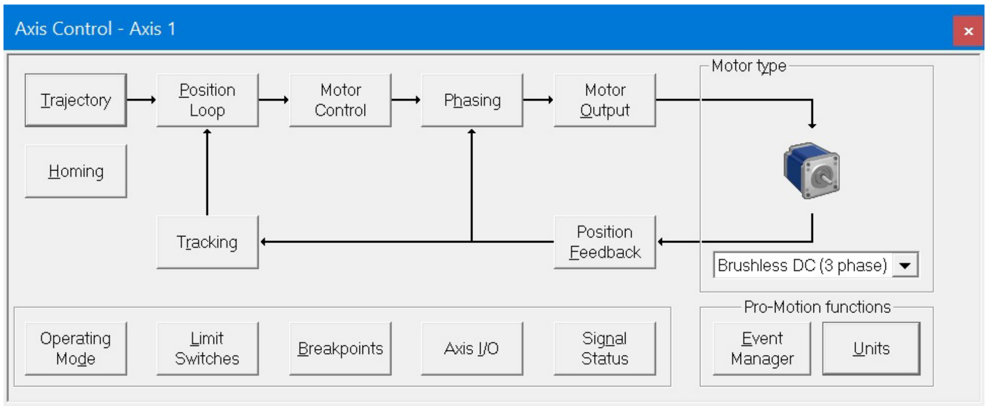
Once the project file has processed a dialog box will appear asking whether you are ready to energize and initialize the motors. In the case of a Brushless DC motor this means the commutation is initialized and depending on the motor control mode setting the current loop and the position loop may be enabled. For DC Brush motors, similarly, the position loop and current loop may be activated depending on the control systems in the project file. For microstepping motors the drive current or holding current will be applied depending again on the programmed control systems.

During a working session you may develop new control parameters, and you may therefore decide to update your existing project file content using File/Save Project or create an entirely new named project using File/Save Project As.

Once initialized, the motor is ready to execute commands provided by the user.

1.12.2 Axis Control Window

We have already used the Axis Control Window in our getting started procedure. Each selectable box (technically these are Windows buttons) in the Axis Control Window results in a dialog box being opened letting you access a sub-set of the functions provided by the Magellan Control IC. For example there are boxes that access the Magellan IC's trajectory generator (labeled 'Trajectory', the position servo loop (labeled 'Position Loop'), drive safety settings (labeled 'Drive Safety') etc.



Visually the Axis Window presents these selectable boxes such that the overall control flow for the motor control mode selected is evident. For example if the boxes displayed for a Brushless DC motor are somewhat different than for a step motor.

Some of the boxes at the bottom of the Axis Control Window are not connected via the control flow arrows because they set or view the status of system-wide functions. These include access to settings for Limit Switches, Breakpoints, Axis I/O, Signal Status, and others.

The box labeled 'Operating Mode' provides control of whether major axis control functions are active. These control functions are trajectory, position loop, current loop, and motor output. In Position and Velocity Control mode all of the operating mode functions are active, but there may be circumstances, for example if a motion error occurs, where some will get disabled by the Magellan IC for safety reasons and may need to be manually re-enabled. For the Torque Control and Voltage modes the control diagram and operating mode settings are somewhat different. For details refer to Section 4.4, "Motor Control Mode Operation."

Underpinning the control flow arrows and selectable boxes in the Axis Control Window is the control architecture of PMD's Magellan Motion Control IC. The reference manual that describes this is the *Magellan Motion Control IC User Guide*. For example if you want to know what trajectory profile modes are available, and exactly how they function and what parameter settings they require this manual contains this information. The same applies for the other selectable boxes and associated functions within the Axis Control Window. As you become familiar with how the Magellan IC controls the motor axis it will be easier and more intuitive for you to access those motion control functions via Pro-Motion.

1.12.3 Trace & Motion Oscilloscope

Many users of the DK58420 will have a specific performance goal in mind, or will want to characterize the performance of the motors when connected to the hardware in their application.

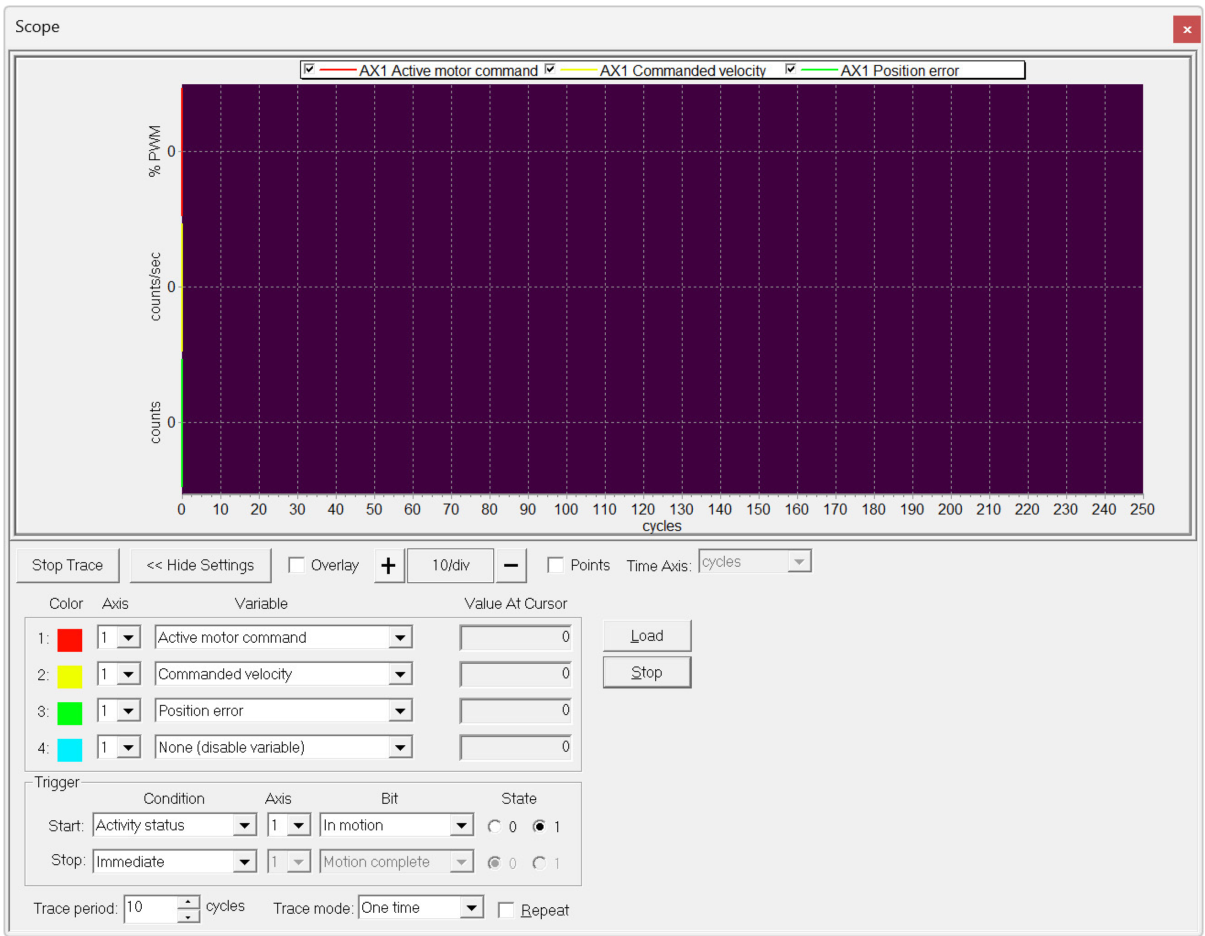
Both optimizing and measuring performance can be accomplished using Pro-Motion's Scope function. These Oscilloscope functions tie directly to a very powerful feature of the Magellan Motion Control IC called trace, which provides the ability to capture and store in hardware memory up to four motion registers simultaneously, at up to 20 kHz speed.

Once a trace operation is specified the Magellan IC captures the motion values at a programmed time interval and stores these values in its local RAM memory. Once the data has been captured Pro-Motion sends commands to retrieve the data from the Magellan IC and then display this data graphically. In addition to being displayed, traced data can be captured to a file for import to spreadsheets or other graphing and analysis software.

There are over 100 different trace variables selectable, but common traced variables include the motor output command, servo position error, commanded position, commanded velocity, and the actual position.

1.12.3.1 Using Pro-Motion Scope Function

To access the Pro-Motion scope function click the icon at the top bar labeled "Scope". A window will open up which you can resize and move if desired.



Here are some of the key settable fields of the Scope Window:

Trace Variables – The core of the trace and scope function is the list of motion registers that will be traced and graphed. These are shown as Variables 1 through 4. For each trace variable click the down arrow which in turn displays a list of trace categories such as Commanded Trajectory, Feedback, Position Loop, etc... Selecting one of these categories then shows the specific available traceable motion variables.

Data Graph – The top portion of the scope window graphs captured data. Up to four variables can be graphed at the same time. The horizontal scale is time, with selectable units via the Time Axis field of cycles, milliseconds, and seconds.

Trigger Controls – Similar to a regular oscilloscope the conditions by which trace data collection can start or end is settable. Use the down arrow key to see the list of available trigger registers, and the associated bit or signal for each register. Popular bits to start trace on are the In Motion bit of the Activity Status word, specific external signals in the Signal Status Register, and one of the two breakpoint bits of the Event Status register. For each trigger bit selected the specific state, low or high also needs to be set.

Trace Control – Two other fields in the scope window are especially useful. Trace Period and Trace mode. The trace period is expressed in cycles, which are 51.2 µSec each in length. So specifying a period of 1 cycle captures data at 20 kHz. Trace mode can be set to One-time or Rolling Buffer Mode. Rolling buffer is useful for monitoring events at slower speeds (higher trace periods). One-time capture fills the trace buffer only once when the trigger conditions are satisfied.

When using Rolling buffer mode caution should be exercised to avoid overflowing the buffer. This can occur if the rate of data being put into the trace buffer by the Magellan IC is faster then the rate at which Pro-Motion can request it. When a buffer overflow occurs a box in the scope window indicates this, and the data being graphed may show a discontinuity. In general selecting One time capture for the trace mode is recommended unless a particular reason for selecting Rolling buffer exists.

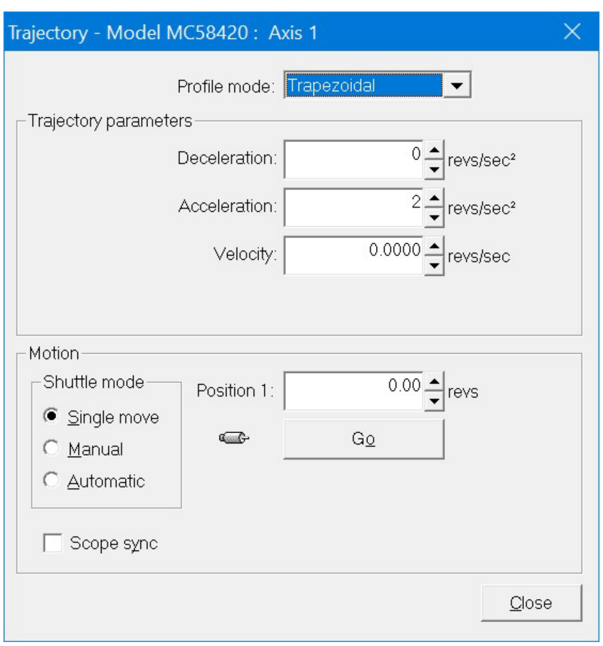
Trace Length – The total number of trace samples can be specified via this setting. Setting this value may help you better-manage how your data displays or how it exports to a file. The programmed value represents the number of data set captures. For example if this value is set to 500 and one variable is traced a total of 500 data values will be captured, if two variables are traced a total of 1,000 data values will be captured etc...

1.12.3.2 Example Performance Optimization Session

Many different optimized control parameter settings can be explored using the trace function's extensive list of traceable variables. In the sequence below we will provide an example that shows how to iteratively find the profile parameters to maximize acceleration speed.

Our goal is to determine what maximum acceleration the motor and attached mechanisms can achieve without exceeding the safe current limit specification for the motor. In this example we will assume motor is driven in position control mode.

1 Start by opening up the Trajectory dialog box, which can be accessed from the Axis Control Window. Specify Trapezoidal profile mode.



2 In the Shuttle mode box at the lower left select Single move.

3 Enter an acceleration value, and a velocity value, and a destination position. A deceleration of 0 commands the deceleration rate to match the acceleration rate.

4 Next open up the scope window and enter the following settings:

Trace variable 1 - Active motor command

Trace variable 2 - Commanded velocity

Trace variable 3 - Position error

Trace Period: 10 cycle

Trace mode: One-time

Start trigger condition Activity Status register, In motion bit, State = 1

Stop trigger condition - Immediate

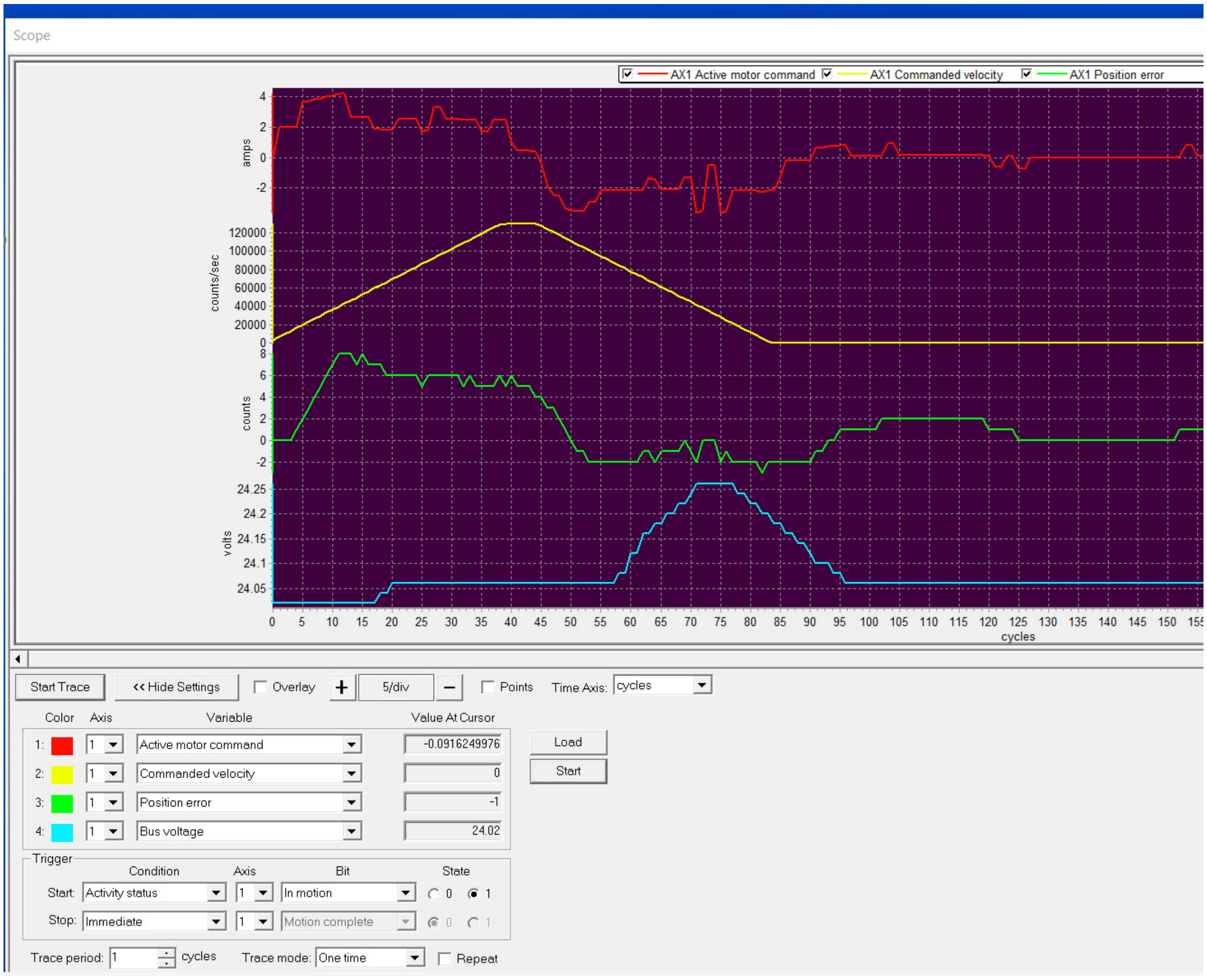
5 Select the Start Trace button in the upper left of the trace window. The data graph display should not change. If it immediately begins to graph data wait till graph display completes and select Start Trace again. When in the proper state the graph area should go blank and although the Start Trace has been activated trace capture (or graphing) does not begin.

6 Review the trace length field located in the middle bottom of the screen and increase or decrease as desired. 250 to 500 is a typical number because it generally displays one oscilloscope screen-full without 'scrolling'. But most PMD products offer far more storage than this. Larger traces may be useful if data will be exported for analysis in a spreadsheet or other analysis tool. To export traced data to a file use the Pro-Motion File/Export Trace menu function.

7 From the Trajectory dialog box select Go. You should see the trace data graphing area immediately begin to display captured data and continue till a full buffer has been captured. The reason this is the case is that by pressing Go the Magellan trajectory generator is activated and the Activity Status Register In Motion bit goes from 0 (false) to 1 (true).

To manage the data being displayed you can use the scope's **-** or **+** buttons to expand or reduce the horizontal scale. Alternatively if you would like to speed up the trace buffer update, see Section 1.12.3.3, "Trace Buffer Display Optimization" for suggestions.

8 The image below shows an example of what a trace may look like when optimizing the acceleration:



This screen capture shows data from a motor with a current drive limit of 4 amps, optimized to accelerate as fast as possible without exceeding the current drive limit. The resultant point-to-point trapezoidal move traverses 300 counts (~ 25 motor shaft degrees) in approximately 4 milliseconds. Note that despite this aggressive move the on-the-fly position error never exceeds 8 encoder counts, and the motor settles to within 2 counts error after 85 cycles, which corresponds to 4.25 mSec.

In a typical session a series of profile moves would be made exploring the effect of different accelerations. The limit of the acceleration command is determined by the maximum current rating of the motor. If too high the motor output limit is exceeded and the motor command value results in rapid increases in the position error.

Other parameters that are commonly optimized to develop the highest possible performance of the motion hardware include the position loop gain settings, the acceleration feed forward values, the maximum velocity, and the servo settling time.

1.12.3.3 Trace Buffer Display Optimization

The speed of the scope display update is related to the size of the buffer that fills with data in the Magellan IC, and the speed with which the Windows PC can retrieve this data from the Magellan IC. For higher speed interfaces such as CAN or Ethernet optimizing these settings is usually not needed. However for serial interface, improving the communication speed may be useful. To set the baud rate to a new value use the S2 switch settings described in Section 2.3.2, “Serial Transceiver.”

Another important way you can speed up serial communications is to reduce the latency between message packet sends. This is done via the following sequence; First, open the Windows Device Manager by typing "Device Manager" in the Windows search box. Next, find Ports (COM & LPT) from the list and click on it, then right-click the USB Serial Port (COMn) of the serial port you are using and then select Properties. Click on the Port Settings tab and then select the Advanced button. Change the field for Latency Timer (msec) to a value of 1. Click OK on both windows and close the Device Manager.

1.12.4 Adjusting Position Loop Parameters

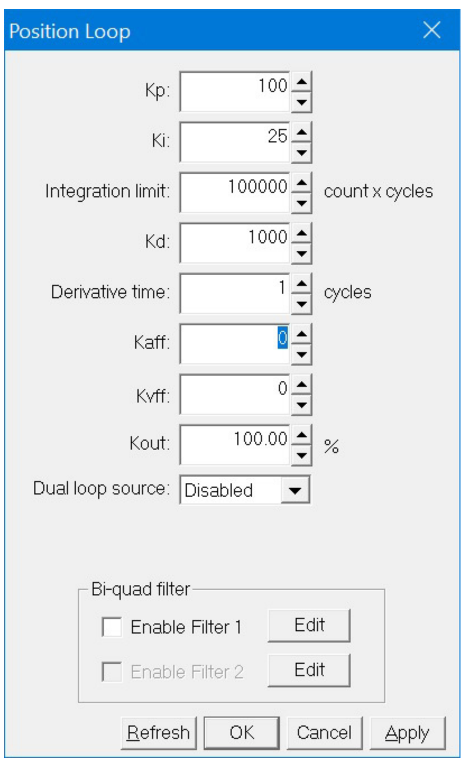
This section applies only for servo controlled motors.

The next few sections provides a simple and straightforward approach to developing position loop gain settings. The results you obtain may not be completely optimum, but should work well in a large majority of cases.

> Re-tuning the position loop parameters is strongly recommended if hardware has been attached to the motor resulting in a change of inertia, or if for any reason the motor motion is altered by being incorporated/installed into the machine hardware.

1.12.4.1 Setup

1 To set up a tuning session the motor should be energized and operating with existing position loop parameters. Open the position loop dialog box by clicking the Position Loop box in the Axis Control window. The diagram below shows what this dialog box looks like:



2 Next, click the Tuning button on the right side of the top icon bar of Pro-Motion. After doing so the trace scope will be displayed along with a Step Response Dialog box. You may want to adjust the location and size of the scope window to conveniently fit your monitor size.

3 In the scope window select Active Motor Command for the first variable to trace, select Commanded Position as the second variable, and select Actual Position as the third variable. If desired you can select a fourth variable to trace but for the purpose of this description this is not necesssary.

4 In the Step Response Dialog box specify a move distance. A typical move distance for motors with an encoder is 50 or 100 counts. For motors using Halls to track the position a typical move distance is 5 to 10 counts.

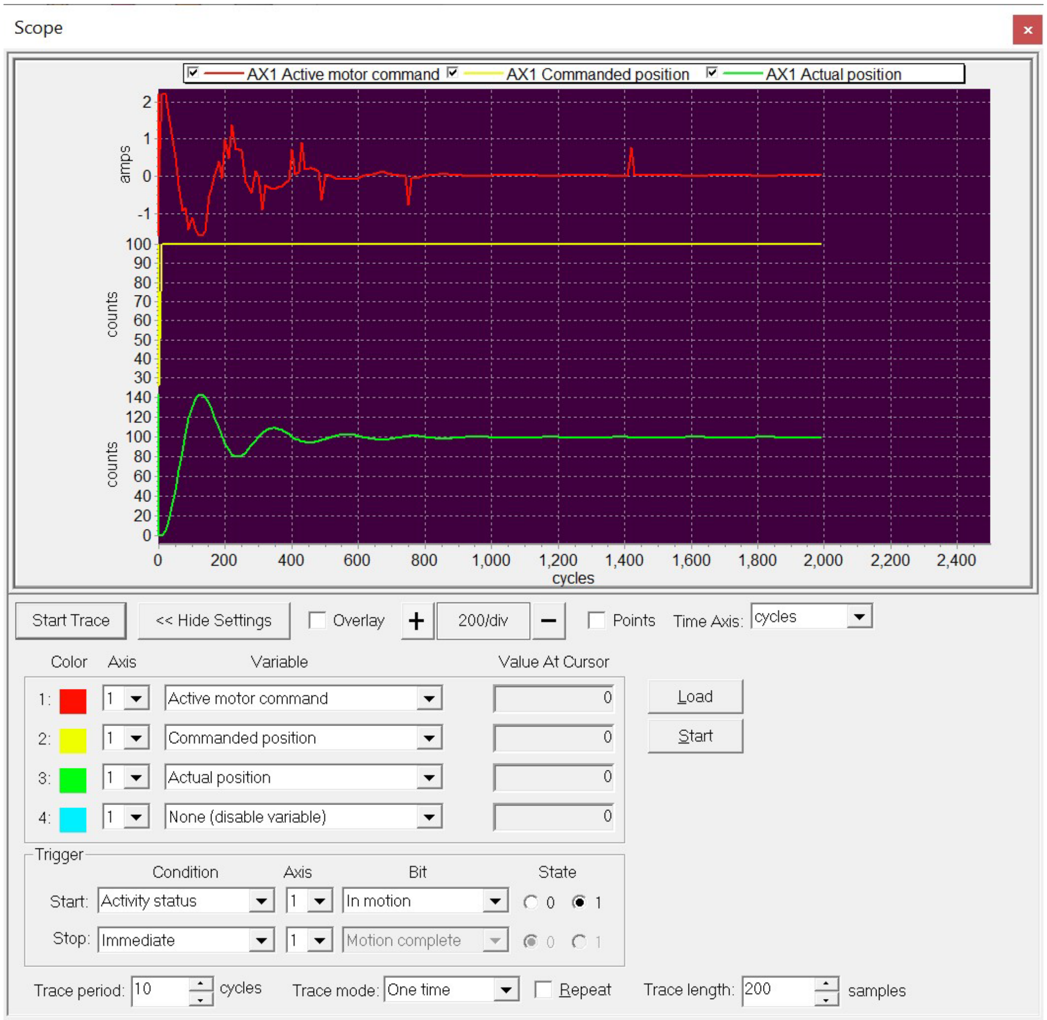
5 As confirmation of a correct setup click the right arrow key in the Step response dialog box. This should result in the motor quickly (in a single instantaneous step) moving by the programmed amount, and shortly thereafter the trace scope window should begin to display data. If this doesn't happen review the instructions above and try again.

6 To develop new tuning parameters set Ki, Integration limit, Kaff, and Kvff to zero. Set Kp to a small value, typically 10 to 50 for motors with encoders and 100 to 500 for motors with Halls only. Set the derivative value to a value 10 times greater than the Kp value you entered. Set the derivative time to 1 and set KOut to 100%.

7 Hit the Apply button and observe the physical motor. If it oscillates or vibrates set Kp to a lower value and re-apply this setting. The scope trace should display the results of the settings you entered. To tune the Kp and Kd you will look at the display results and determine whether the response is underdamped, critically damped, or overdamped and increase. These characterizations will guide you to how to change the gain settings as described in the following three sections.

1.12.4.2 Underdamped Response

The screen capture below shows a typical underdamped response.

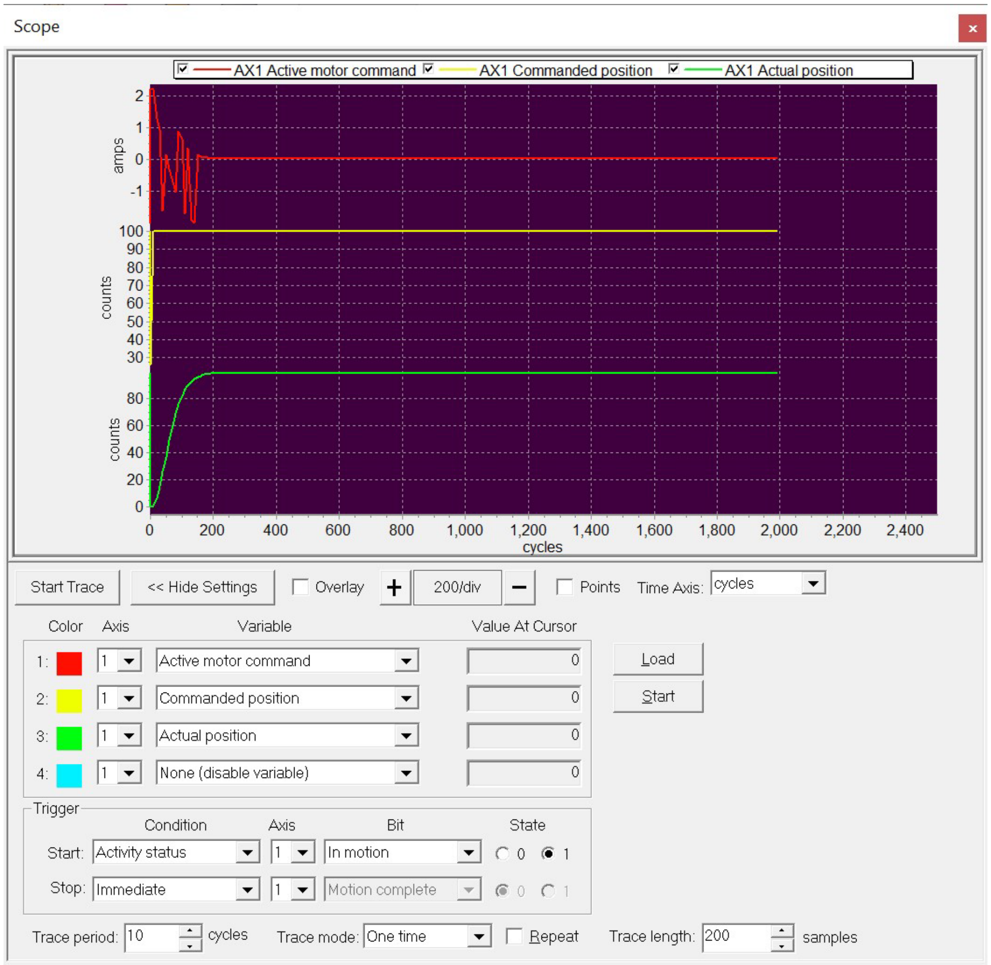


The yellow trace shows that the commanded position instantly changes from a position of 0 to a position of 100 (in your setup this jump will equal whatever you entered into the step response dialog box). The green trace shows the resultant actual motor location.

From this trace, although the actual position does eventually settle to a position value of 100, it overshoots significantly and oscillates several times. Whenever the actual axis position overshoots the commanded position the system is considered underdamped, and to correct for this you should increase the derivative term. The goal of the Kd setting is to determine a value that results in a critically damped response, as shown in the next section.

1.12.4.3 Critically Damped Response

The screen capture below shows a critically damped response.

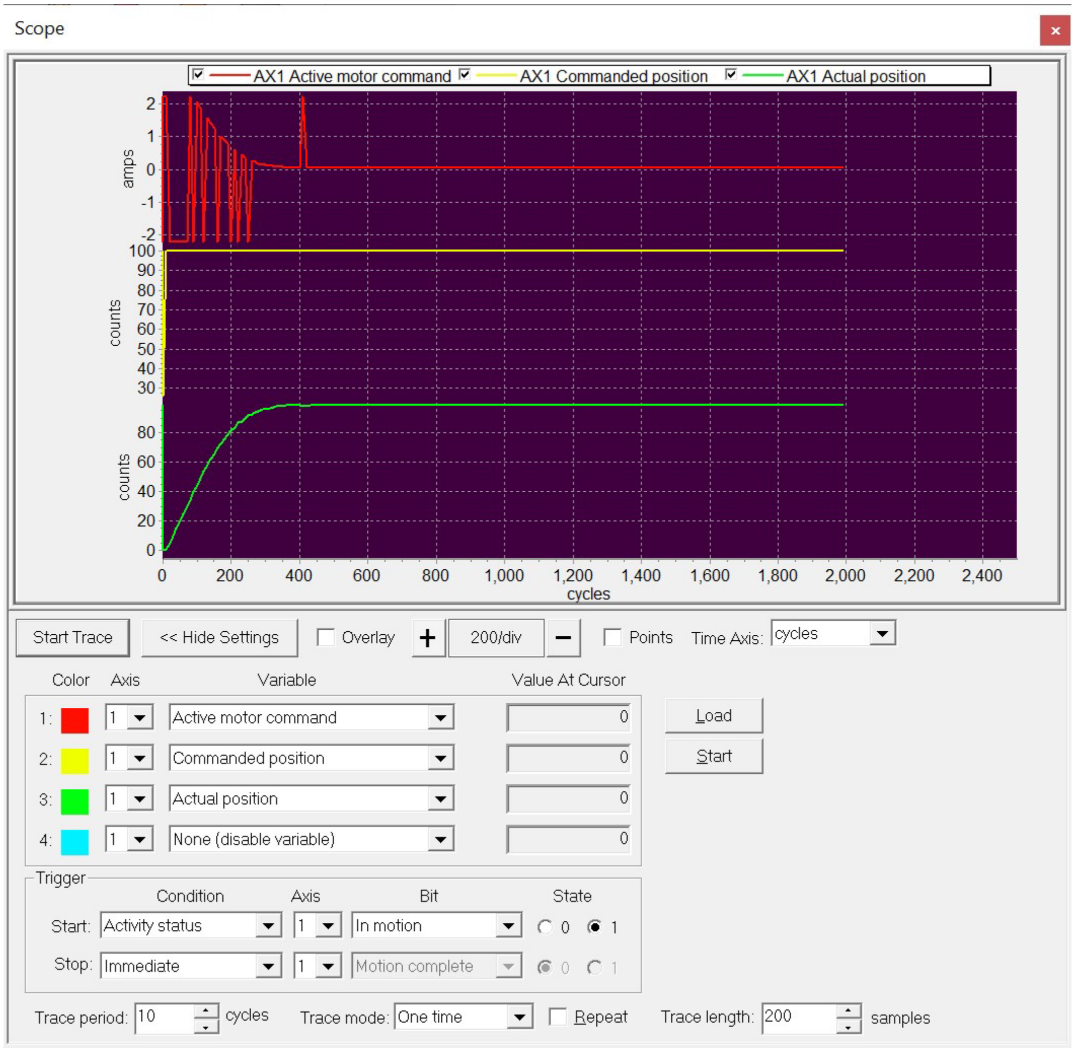


In this trace the actual position does not overshoot that commanded position, and it settles quickly for this particular motor in 10 mSec to the commanded position of 100.

It's worth noting that many Kd settings could result in a response that appears critically damped. The distinction between critically damped and overdamped, in particular, is somewhat subjective when using this manual trial and error process. In general you should try to find the smallest setting of Kd which still provides a critically damped response.

1.12.4.4 Overdamped Response

The screen capture below shows an overdamped response.



This can be seen by comparing with the critically damped response in the previous section. An overdamped system will be stable but unnecessarily slow in its response to changes in the position command.

If the response is overdamped you should reduce Kd trying to find the smallest value of Kd that still gives a critically damped response.

> For purposes of illustration in the above three example graphs the Kp setting was 100 and the Kd settings were 2,000, 6,000, and 25,000 for the underdamped, critically damped, and overdamped responses respectively.

1.12.4.5 Getting To Final Gain Settings

Your overall goal is to set the Kp to as high a value as possible while still finding associated Kd values that can create a critically damped response. Higher Kp values will result in more accurate tracking and faster responses to commanded position changes.

As you try higher and higher Kp values, at some value of Kp you will find that there is no value of Kd that can create a critically damped response. At this point you should reduce the Kp setting by 35-50% and determine the associated critically damped Kd setting. Backing down from the 'borderline' Kp value is important for improving stability, accommodating small differences in controller, motor, and attached hardware behavior, and for handling changes that may occur to the system as it operates in the field over time.

Once Kp and Kd have been set, you can enter a value of Ki to improve tracking accuracy. Typically, Ki settings are 1/10 to 1/2 of the Kp value. Smaller motors typically use relatively large ratios, and larger motors with larger loads typically use smaller Ki ratios. Higher Ki settings will increase tracking accuracy during and after the motor move is complete but can also reduce system stability. So you should set Ki to the smallest value that can achieve your goals for final or dynamic tracking while not distorting the critically damped response the Kp and Kd settings created.

The Integration limit can be set to a high value, perhaps 100,000. The Magellan IC PID engine has built in anti-windup logic so it is very unlikely this integration limit will ever be reached.

The final 'core' PID setting, known as derivative time, can often be left at one. Setting it to higher values means the period at which the derivative contribution is calculated is increased. This can be useful to lower the frequency of Kd 'chatter' to below audible, should there be any in the system. Another benefit to increasing the derivative time may be to increase the influence of the Kd term. The impact of the Kd on the PID is a direct multiple of the derivative time. For example if the derivative time is set to 10 and the Kd value is 100 this would be an equivalent impact as a Kd of 1,000 with a derivative time of 1.

1.12.4.6 Frequency Based Tools & Analysis

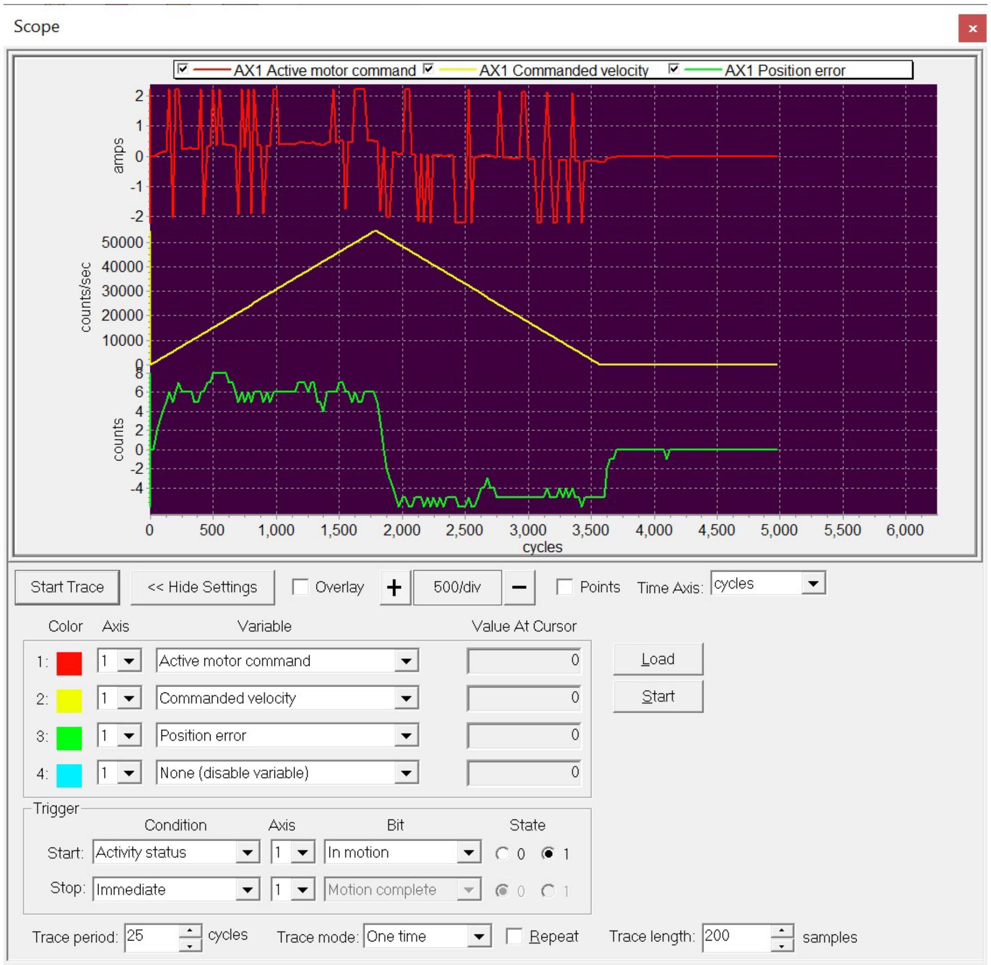
The above process represents a popular and relatively straightforward approach toward tuning the PID loop. There are more sophisticated approaches however both for determining PID parameters and for verifying the behavior of a given PID setting. PMD provides frequency-based tools known as Bode plots to support such methods.

Bode plots can be used in different ways and can help characterize your mechanical system as well as determine important characteristics of the operating control loop such as the bandwidth and phase margin. The Pro-Motion Bode plot tool can be accessed from the View/BodePlots Pro-Motion menu selection.

One output of a Bode-based system characterization may be the identification of natural resonances in the mechanical system. While a properly tuned PID can help reduce the impact of this, the Magellan IC also provide two general purpose biquad filters in the position loop. Biquad filters can be used to create low pass and notch filters. For more information on biquads as well as the Magellan Position PID loop refer to the *Magellan Motion Control IC User Guide*.

1.12.4.7 Feedforward Terms

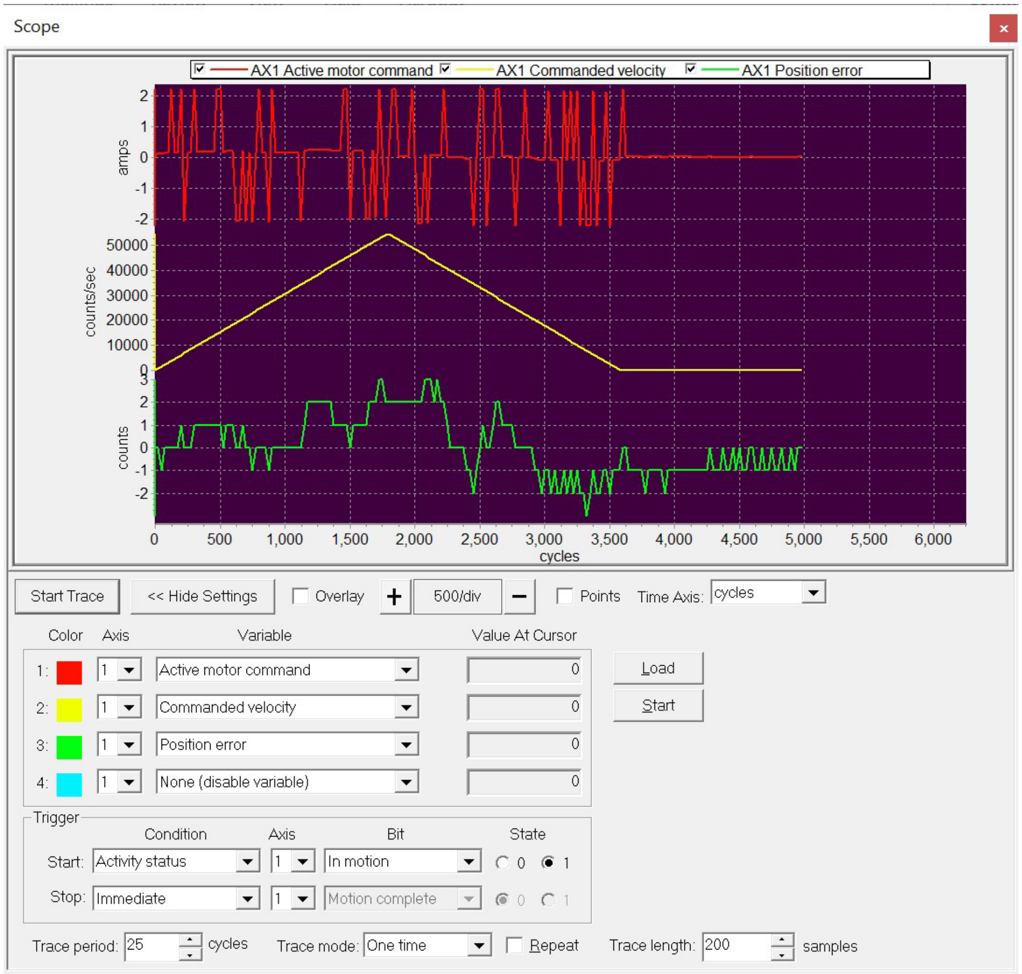
The graph below shows a complete point to point profile move using the critically damped system settings from above. The yellow line represents the commanded trajectory velocity, and the green line represents the position error (the difference between the commanded and the actual position). Notice that despite the fact that the axis is accelerating and decelerating quite aggressively (the entire move duration is less than 200 mSec) the motion is stable throughout and the tracking accuracy is quite good even during the motion, varying from +8 counts maximum position error to -5 counts.



Nevertheless Acceleration feedforward (Aff for short) can reduce the dynamic tracking error even further. Acceleration feedforward works by adding a torque command proportional to commanded acceleration. PMD's Magellan IC also supports velocity feedforward, which similarly adds to the motor command proportional to the commanded velocity. Velocity feedforward is used less often than acceleration feedforward, typically to compensate for fluid friction such as lubricated bearings or fluid pumping applications.

The reason Aff improves tracking accuracy is that from the position error graph above we can see that the form of the position error echoes the acceleration value. In this trapezoidal profile the acceleration is a constant positive value as the profile increases speed and a constant negative value as the profile speed decreases. Therefore by 'pre-injecting' this feedforward value the servo loop is required to do less work, resulting in better tracking accuracy.

After adding an appropriate Aff gain to the PID loop the results are shown below. Although the tracking is not perfect, the magnitude of the position error is halved during the motion.



One caution with acceleration feedforward is that it is load specific. If you tune the Aff value by increasing and decreasing till the net position error during motion is minimized, you will find that if the load on the motor changes you will need to re-tune the Aff value to achieve a similar result. This means systems that typically carry a variable load may not be good candidates for acceleration feedforward.

1.12.5 Troubleshooting Suggestions

If the first-time verification sequence detailed in Section 1.11, "First Time System Verification" was successful problems while using Pro-Motion are not that common. Nevertheless below is a list of issues users may encounter while exercising their motor or developing their motion applications and suggestions for correcting the issue.

The Motor Stops Responding. If a motor previously moving and responding to your commands stops responding the most likely reason is that the operating mode is not set correctly. Although most such 'transitions' from normal operation to an error/protection state result in a dialog box appearing, to check the operating mode select the Operating Mode box in the Axis Control Window and adjust the settings if needed. See Section 1.12.2, "Axis Control Window" for more on the Operating mode dialog box.

The Step Motor Stops Responding. In addition to checking the Operating Mode as detailed above, step motors controlled via microstepping mode may stop moving because the motor command has been set to zero. This may occur during safety related actions or when certain command functions are sent. To check and restore the motor command setting select the Motor Control box in the Axis Control Window.

Events Aren't Being Reported. As the Magellan Motion Control IC executes its control settings some occurrences representing a change in the control system state may occur. These changes are called events and include items such as motion error, over temperature, breakpoints, and more. To check the Pro-Motion settings for which events will result in a dialog box popping up click the Event Manager box in the lower right of the Axis Control Window. For detailed information on Magellan events refer to the

An Overvoltage Event Occurred. If during a motion profile an overvoltage event is indicated the most likely reason is that during deceleration of the motor the inertia of the motor and load resulted in net generation of energy. Depending on the design of the power supply you are using this can result in the HV voltage rising and eventually exceeding the overvoltage limit. Remedies are to reduce the profile's rate of deceleration or add more capacitance to the HV supply.

An Undervoltage Event Occurred. If during a motion profile an undervoltage event is indicated the most likely reason is that during acceleration the power supply was not able to deliver the needed amount of current, or was not able to deliver an increase in current fast enough. Undervoltage (and overvoltage) events may also occur if the current loop or position loop is unstable. Remedies are to increase the current output limit/rating of the power supply, add more capacitance to the HV supply, or reduce the profile's rate of acceleration or re-tune the servo loops.

A Motion Error Event Occurred. If during a motion profile a motion error event is indicated there are a few potential reasons. Motion error, also called position error, means the difference between the commanded position and the actual motor position has exceeded a user-settable threshold. Potential reasons are that the axis motion was blocked, that the position loop is not well tuned, that the commanded profile velocity exceeds the achievable motor velocity, or that an aggressive profile has temporarily resulted in a motion error during the move. Remedies depend on the cause, and are generally straightforward once the cause is determined.

Communication Errors Are Occurring. Communication errors between Pro-Motion and the controller are rare, but depending on the connection type and motion setup may occur. The 3-pin Programming Port for example is susceptible to noise if high current moves are commanded because it does not use a differential signaling scheme. For whatever communication link is used, you should try to determine if there are specific operating conditions that result in communication errors. If signal noise may be at fault consider switching to a link that uses a differential signaling scheme such as RS-232, CAN, or Ethernet.

> Depending on the type of amplifier used some of the events described above may not be reportable by the control system. For example if Atlas amplifiers are used Overvoltage and Undervoltage event reporting will occur correctly, but if user-designed or non-PMD amplifiers are used these events will not be detected or reported by Pro-Motion.

1.12.6 Summary

Hopefully this brief guided tour and additional information about Pro-Motion has been useful to help you to better understand Pro-Motion and your PMD controller product. For detailed information on the Magellan Motion Control IC, refer to the *Magellan Motion Control IC User Guide*. For information on Magellan command mnemonics and C-motion syntax refer to the *C-Motion Magellan Programming Reference*.

2. Operation

In This Chapter

The PMD DK58420 is a high performance PCI-bus board that provides motion control for DC Brush, Brushless DC, and step motors. These boards are based on Magellan Motion Control ICs which perform motion command interpretation and other real time functions.

The overall board function is divided amongst a number of modules. These modules are shown in the block diagram below:

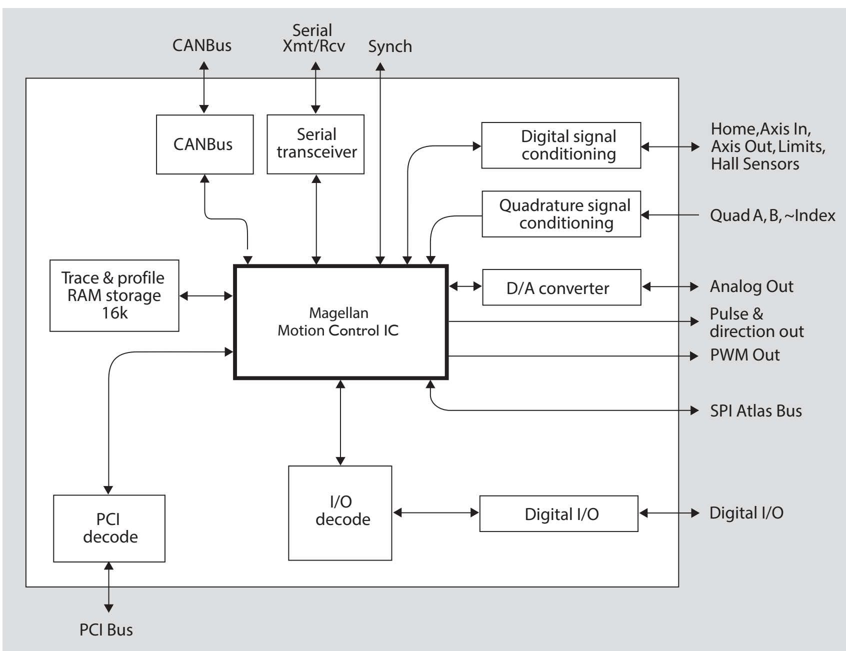


Figure 2-1: DK58420 Developer Kit Board Internal Block Diagram

2.1 Board Function Overview

DK58420 Developer Kit board resources can be broken into three overall categories:

Magellan Motion Control IC functions—These are programmable functions which reside in the motion control IC such as profile generation, servo loop closure, and much more. These functions are accessed using Magellan Motion Control IC commands, of which there are roughly 150 in total, to allow sophisticated control of the board's overall behavior.

Board-specific functions—These are programmable functions which are controlled by the motion control IC using commands ReadIO and WriteIO, but which reside in various portions of the board circuitry. These functions include general purpose digital IO, and other board-specific capabilities.

Signal processing & hardware functions—A substantial portion of the board provides signal conditioning and other functions associated with non-programmable, signal-related processing.

2.1.1 Standalone Operation

The DK58420 board is operated standalone and is communicated to via serial or CANbus communications. In addition, you will need to provide power to the board via an external connection, J10. See Section 3.2.6, "Standalone Power Connector (J10)" for a detailed description.

2.2 Magellan Motion Control IC

The Magellan Motion Control IC pictured in Figure 2-1 is comprised of 2 ICs, a CP (command processor) and an IO (input/output) IC. A summary list of the functions provided by the motion control IC is as follows:

- Profile generation
- Motor output signal generation (PWM and analog)
- Quadrature counting, index capture
- Servo loop closure (for DC Brush or Brushless DC motors only)
- Breakpoint processing
- PLC-function processing (AxisIn and AxisOut signals)
- Trace
- Motion error detection, tracking windows, and at-settled indicator
- Limit switches

C-Motion function calls, one for each Magellan command, is used to communicate to the board. For a complete list of Magellan commands see the *C-Motion Magellan Programming Reference*.

The PC on which the DK58420 Developer Kit board is installed can control the board through the pre-compiled program Pro-Motion, or through a program of the user's own construction, using C-Motion calls as the basic interface. During axis setup, the communications method (serial port, or CANbus) and other parameters are specified, which allow C-Motion to create a virtualized axis handle, that from then on is the reference for all C-Motion commands.

Available C-Motion commands correspond one-for-one with those listed in the *C-Motion Magellan Programming Reference*. All C-Motion commands preface the Magellan command with the letters "PMD," so the Magellan command (for example) SetPosition becomes the C-callable routine.

PMDSetPosition

Example

The following simple example, to set up and execute a simple trapezoidal profile, illustrates just a small part of the overall command set.

```
PMDSetProfileMode(axis_handle, trapezoidal);
PMDSetPosition(axis_handle, position_value);
PMDSetVelocity(axis_handle, velocity_value);
```

PMDSetAcceleration(axis_handle, acceleration_value);
PMDSetDeceleration(axis_handle, deceleration_value);
PMDUpdate(axis_handle);

Two separate manuals describe how the Magellan Motion Control IC operates and how it is programmed, the *Magellan Motion Control IC User Guide* and *C-Motion Magellan Programming Reference.*

2.3 Board Specific Functions

Board-specific functions are those functions that are mapped through the motion control IC's ReadIO and WriteIO facility, but are implemented in the board circuitry.

Board-specific functions are detailed in this document rather than the *Magellan Motion Control IC User Guide* or the *C-Motion Magellan Programming Reference.*

2.3.1 General Purpose Digital IO

In addition to numerous special-purpose digital signals that are input or output to the board such as ***AxisIn, AxisOut, Home, QuadA,*** etc., the DK58420 board supports 8 general-purpose inputs, and 8 general-purpose outputs. These signals provide a convenient way of accessing additional general purpose digital IO. Although access to these signals occurs through the motion control IC's **ReadIO** and **WriteIO** command, the signals present at these various connections do not directly affect the motion control IC's behavior. Thus the motion control IC simply passes them through.

ReadIO and WriteIO Commands

The 8 inputs and outputs are read using the **ReadIO** command and **WriteIO** command, with an IO address of 0. The table below shows this, along with the bit locations of the input and output signals.

Command	Bit location	Signals
ReadIO 0	0-7	DigitalIn0-7
WriteIO 0	0-7	DigitalOut0-7

To read the 8 general purpose digital I/Os, a **ReadIO** command is performed at address offset 0. To write new signal values to the 8 digital outputs, a **WriteIO** command to address offset 0 is sent, and the values on bits 0-7 will be output to the signal connections. The value of bits 8-15 are ignored.

Example

To write a sequence 0xaa to bits 0-7, the C-Motion command **PMDWriteIO(axis_handle, 0, 0xaa)** is used. Assuming that a signal pattern of 0x55 is present on the 8 input connections, if the command **PMDReadIO(axis_handle, 0, &load_reg)** is used, load_reg will contain 0x55.

Connections & Associated Signals

The general-purpose IO signals are direct single-signal digital inputs and outputs. There are no associated connections that need to be made for these signals to function properly; however, one or more of the digital grounds must be connected. The default value, upon powerup, for all general-purpose digital outputs is low.

See Chapter 3, "Magellan Developer's Kit Electrical Specifications" for a complete description of the pinout connections to/from the board.

2.3.2 Serial Transceiver

This module and associated signals provide the capability to operate the DK58420 board using an asynchronous serial port, or to allow certain monitoring operations to be performed through the serial port.

Two connectors provide serial port signals from the DK58420 board to external devices. J13 is a standard DB9 connector, and is used for point-to-point serial communications. This connector is designed such that it can directly connect to a PC serial port without a null modem. J1 is a 5-pin connector which is used for multi-drop communications with an external driver board. For more information on external driver boards for multi-drop communications, contact your PMD representative.

Regardless of which connector is used, the following information is useful to select baud rates and set other parameters associated with serial pot communications.

The serial port can be operated at various baud rates from 1,200 to 460,800, and different configurations of stop, start, and parity codes. In addition, three connection modes are provided, point-to-point, multi-drop (address bit mode), and multi drop (idle line mode). The following table shows how these parameters can be set on the board.

Switch block S1 sets the transmission rate, parity, stop bits, and protocol. Switch block S2 selects the device address when using the multi-drop protocol. When referring to the table below, the switch up position is relative to the bottom of the board where the PCI connector is located. The up position on the switch is marked **on**.

		S1-1	S1-2	S1-3	S1-4	S1-5	S1-6	S1-7	S1-8	S2-1
Transmission rate (bits per second)	1200	up	up	up	up					
	2400	down	up	up	up					
	9600	up	down	up	up					
	19200	down	down	up	up					
	57600	up	up	down	up					
	115200	down	up	down	up					
	230400	up	down	down	up					
	460800	down	down	down	up					
Parity	None					up	up			
	Odd					down	up			
	Even					up	down			
Stop bits	1							up		
	2							down		
Protocol	Point-to-point								up	up
	Address bit								up	down
	Idle line								down	down

See Figure 1-1 for locations of switches S1 and S2.

2.3.3 S2: Serial Device Address

Switch block S3 sets the device address for multi-drop protocol systems. When referring to the table below, the switch up position is relative to the bottom of the board where the PCI connector is located. The up position on the switch is marked **on**.

S2-2 and S2-3 must be left in the up position.

Multi-drop address selection (S2)

Address	S2-4	S2-5	S2-6	S2-7	S2-8
0	up	up	up	up	up
1	down	up	up	up	up
2	up	down	up	up	up
3	down	down	up	up	up
4	up	up	down	up	up
5	down	up	down	up	up
6	up	down	down	up	up
7	down	down	down	up	up
8	up	up	up	down	up
9	down	up	up	down	up
10	up	down	up	down	up
11	down	down	up	down	up
12	up	up	down	down	up
13	down	up	down	down	up
14	up	down	down	down	up
15	down	down	down	down	up
16	up	up	up	up	down
17	down	up	up	up	down
18	up	down	up	up	down
19	down	down	up	up	down
20	up	up	down	up	down
21	down	up	down	up	down
22	up	down	down	up	down
23	down	down	down	up	down
24	up	up	up	down	down
25	down	up	up	down	down
26	up	down	up	down	down
27	down	down	up	down	down
28	up	up	down	down	down
29	down	up	down	down	down
30	up	down	down	down	down
31	down	down	down	down	down

2.3.4 CANbus Transceiver

In addition to a serial port, it is possible to operate the DK58420 Developer Kit board via CANbus. The CANbus interface is useful to communicate to the board when one or more boards co-exist on the same network. In addition, compared to a RS-485 interconnect, CANbus communication is faster and more robust.

The following table shows how to configure the CANbus port. When referring to the table below, the switch up position is relative to the bottom of the board where the PCI connector is located. The up position on the switch is marked **on**.

> S3-8 and S4-1 through S4-5 should be left in the on position.

Switches		Description								
S3	1-7	Node ID Setting Switches 1-7 of the S3 DIP switch control the CANbus nodeID, allowing the board to be uniquely addressed on the CANbus network. Switches 1-7 are set binary-encoded, with total range of allowed values 0 - 127, and with 1 being least significant, 7 most significant. If a switch is up, it encodes a bit value of 0 and if it is down it encodes a bit value of 1.								
		S3-1	*S3-2*	*S3-3*	*S3-3*	*S3-4*	*S3-5*	*S3-6*	*S3-7*	*NodeID*
		up	up	up	up	up	up	up	up	0
		down	up	up	up	up	up	up	up	1
		up	down	up	up	up	up	up	up	2
		down	down	up	up	up	up	up	up	3
		up	up	down	up	up	up	up	up	4
		down	up	down	up	up	up	up	up	5
		up	down	down	up	up	up	up	up	6
		down	down	down	up	up	up	up	up	7
		. . .								
		down	down	down	down	down	down	down	down	127
S4	8-16	Communication rate. These switches set the CANbus communication rate as follows:								
		S4-6	*S4-7*	*S4-8*	*Baud setting*					
		up	up	up	1,000,000 baud					
		down	up	up	800,000 baud					
		up	down	up	500,000 baud					
		down	down	up	250,000 baud					
		up	up	down	125,000 baud					
		down	up	down	50,000 baud					
		up	down	down	20,000 baud					
		down	down	down	10,000 baud					

> If the attached CANbus device is the last node on the CAN network, then JP15 should be left in the default position of 2-3 jumpered. If this it is not the last device, then it should be installed with 1-2 jumpered.

C-Motion Commands

CANbus parameters can also be set using C-Motion commands through the CANbus interface or another interface. Although this is not commonly done, it can be useful to test communication ports.

C-Motion Command	Argument(s)	Description
SetCANMode	axis_handle	Sets the CAN mode communication information. The binary word sent *mask* is encoded per the bit field above.
GetCANMode	axis_handle	Gets the CAN mode communication information. The returned word is encoded per the bit field described above

Connections & Associated Signals

A special DIN-style 4-pin connector is used to connect to the CANbus port. Section 3.2.9, "CANbus Connector (JS1)" provides a detailed description.

See Chapter 3, "Magellan Developer's Kit Electrical Specifications" for a complete description of the pinout connections to/from the board.

2.3.5 Reset

The DK58420 board can be reset manually as well as electrically. To reset the board manually, depress the reset button on the board. This button can be located using Figure 1-1.

Although a reset occurs automatically during power up, a user-initiated reset can also be performed explicitly through the board's PCI-bus interface. There are two methods by which this can be done. They are summarized in the table below:

Method	Type of reset	Description
Reset through motion control IC	Soft reset	The C-Motion command PMDReset sends the command Reset to the motion control IC, which causes a "soft" reset of the motion control IC only. See the *Magellan Motion Control IC User Guide* for more information.

After a reset occurs, the motion control IC and other related output signals will be driven to known states, depending on the type of reset performed. These are summarized in the table below:

Signal name	Reset condition	
AxisOut1-4	High	
PWMMag1A-4C	DC Brush motor:	Low
	Brushless DC motor:	50/50 High/Low
	Microstepping motor:	Low
PWMSign1A-4B	DC Brush motor:	High
	Brushless DC motor:	Low
	Microstepping motor:	High
DAC1A-DAC4B	0.0 volts	

If Atlas amplifiers are used, a Magellan reset will result in the commanded output torque being set to a zero value. A Magellan reset does not result in an explicit reset of connected Atlas units. If, for whatever reason, an explicit reset of the Atlas units is desired a command can be sent via the Magellan and addressed to the Atlas, or power to the Atlas can be cycled. To learn more about sending Atlas commands via an attached Magellan IC see the *Magellan Motion Control IC User Guide*.

C-Motion Commands

The available C-Motion callable functions for this feature are:

C-Motion Command	Arguments	Function Description
PMDReset	axis_handle	This function causes a "soft" reset of the motion control IC.

2.4 Signal Processing and Hardware Functions

Signal processing and hardware functions are board functions which are not directly user-programmable. These are board characteristics which are encoded in hardware. Primarily this consists of various types of signals. The following sections lists these related groups of signals and provides information that may be helpful when connecting your motion system.

2.4.1 Home, AxisIn, AxisOut, Limits, Hall Sensors

These signals are conditioned by the board, but are output or input directly to the motion control IC. The *Magellan Motion Control IC User Guide* explains the functions provided in connection with these various signals. Most of the signals are optional, and are connected depending on the nature of the application being used.

These signals are named ***Home1-4***, ***AxisIn1-4***, ***AxisOut1-4***, ***PosLim1-4*** (positive direction limit input), ***NegLim1-4*** (negative direction limit input), and ***Hall1A-4C*** (12 signals in all).

Connections & Associated Signals

This group of signals are direct single-signal digital inputs to the board, with the exception of ***AxisOut***, which is a single-ended output. There are no associated connections that need to be made for these signals to function properly; however, one or more of the digital grounds must be connected. The default value, upon powerup, for all ***AxisOut*** signals, is high.

See Chapter 3, "Magellan Developer's Kit Electrical Specifications" for a complete description of the pinout connections to/from the board.

2.4.2 QuadA, QuadB, Index

This group of signals provides position feedback to the controller which is used to track motor position, and for servo motors, to generate a motor command. For DC Brush and Brushless DC motors, they are required for proper operation. For microstepping or pulse and direction motors, they are optional.

The encoder-processing circuitry provides a multi-pass digital filter of the ***QuadA***, ***QuadB***, and ***Index*** signals for each axis. This provides additional protection against erroneous noise spikes, thereby improving reliability and motion integrity.

These signals are named ***QuadA1+*** through ***QuadB4-*** (16 signals), and ***Index1+*** through ***Index4-*** (8 signals).

Connections & Associated Signals

This group of signals are connected in one of two ways. Single-ended termination means that only one wire per signal is used, while differential (dual) mode means two wires encode each signal (labeled + and -). Differential encoding is generally recommended for the highest level of reliability because it provides greater noise immunity than a single-ended connection scheme.

If single-ended connections are used only the + signal is connected, and the - signal should be left floating. For example, in connecting to the A quadrature input, ***QuadA1+*** connects to the signal, and ***QuadA1-*** is left floating. If differential connections are used, both the + and - signals are used.

Differential or single-ended termination must be selected through resistor pack installation. (See table in Section 1.8.1, "Resistor Packs" for details.) Note that all quadrature and index connections should be in either single-ended or differential mode. It is not possible to mix on a signal-by-signal basis.

When using the system with differential connections, if desired, the polarity of the differential signal can be reversed by swapping the + and - connections. This may be useful for altering the motor and/or encoder direction; however, this same function can also be accomplished through commands to the motion control IC. See the *Magellan Motion Control IC User Guide,* for more information.

Associated connections that are supported by the board are the +5V output signals. These are provided as a convenience, as they are generally connected to a corresponding input on the encoder, to power its circuitry. As was the case for the digital input signals, one or more of the digital grounds must also be connected.

See Chapter 3, "Magellan Developer's Kit Electrical Specifications" for a complete description of the pinout connections to/from the board.

2.4.3 Pulse & Direction Input Over QuadA, QuadB

The DK58420 board supports an alternate decoding scheme for the QuadA and QuadB position input signals consisting of pulse & direction position feedback interpretation.

Normally, the QuadA and QuadB signals are interpreted to be quadrature encoded. After passing through the differential receiver circuitry they are output as digital logic levels directly to the Magellan IC, which only supports quadrature encoded signals at these two pins.

The DK58420 board includes logic circuitry external to the Magellan IC which, when enabled, converts incoming pulse & direction signals into corresponding quadrature encoded signals. After conversion, these quadrature-encoded signals are passed to the Magellan IC where they are used to accumulate the feedback position in the normal manner.

This capability can be set on a per-axis basis, meaning that some axes can be set to input quadrature-encoded signals, and other axes can input pulse & direction-encoded signals. Selection of standard (no conversion) or converting (pulse & direction to quadrature conversion) signal processing is controlled via dip switches S5-4, S5-8, S6-4 and S6-8.

The primary use of this feature is to allow position command input from external pulse & direction output motion controllers. Inputting this position data stream into Magellan is useful for:

- Electronic gear or cam position command input
- Analysis of an external controller's pulse & direction profile
- Setup and optimization of systems that will use Atlas amplifiers in pulse & direction signal input mode.

The table below shows how to set the QuadA and QuadB input mode:

Item	Switch	Description
Dip switch S5-4	up	Sets QuadA and QuadB interpretation for Axis 1 to quadrature
	down	Sets QuadA and QuadB interpretation for Axis 1 to pulse & direction
Dip switch S5-8	up	Sets QuadA and QuadB interpretation for Axis 2 to quadrature
	down	Sets QuadA and QuadB interpretation for Axis 2 to pulse & direction
Dip switch S6-4	up	Sets QuadA and QuadB interpretation for Axis 3 to quadrature
	down	Sets QuadA and QuadB interpretation for Axis 3 to pulse & direction
Dip switch S6-8	up	Sets QuadA and QuadB interpretation for Axis 4 to quadrature
	down	Sets QuadA and QuadB interpretation for Axis 4 to pulse & direction

2.4.4 Analog Input

The *Analog0-7* signals provide general purpose input of 8 analog signals. If connected, the voltages present at these various connections do not directly affect the motion control IC's behavior. However they can be read through the motion control IC, and thus provide a convenient way of bringing in analog signal levels that may be acted upon by the user's application code located on the PC. These signals are read using the Magellan command ReadAnalog. In conjunction with the *Analog0-7* signals, the user must also provide a number of other signals that provide analog reference scaling to the Magellan Motion Control IC. These signals are summarized in the table below:

Signal name	Function
AnalogRefLow	Provides minimum allowed analog voltage input signal. Has an allowed range of 0 to 3.3V. Generally connected to 0 volts.
AnalogRefHigh	Provides maximum allowed analog voltage input signal. Has an allowed range of 0 to 3.3V, but must be greater than ***AnalogRefLow***. Generally connected to 3.3 volts.
AnalogGND	Provides ground return for reference and analog input signals

All of the analog signals described in this section are directly connected to the corresponding pins on the Magellan Motion Control IC. For more information on reading the value of these analog inputs, see the *Magellan Motion Control IC User Guide*.

Connections & Associated Signals

For analog voltages to be read correctly, in addition to the analog signals *Analog0-7* themselves, *AnalogRefLow*, *AnalogRefHigh*, and *AnalogGND* must be connected. See the preceding table for more information.

2.4.5 SPI Atlas Bus

Atlas amplifiers are directly supported via SPI Atlas bus signals. This bus connects the Magellan to the Atlas amplifier units, and allows bi-directional communication so that Magellan may continually specify the desired amplifier output command as well as query Atlas for parameters or other control/status information. Communication on the SPI Atlas bus is in the form of packets using a command protocol defined by the Atlas amplifiers.

The bus hardware signaling is the same regardless of motor type. Motor-specific differences are represented in the command protocol rather than differences in signal connections. For detailed information on SPI hardware timing and command protocols refer to the *Atlas Digital Amplifier Complete Technical Reference*.

Connections & Associated Signals

There are seven signals associated with the SPI Atlas bus: SPIClock, SPIXmt, SPIRcv, and SPIEnable1, SPIEnable2, SPIEnable3, and SPIEnable4. All of these signals directly connect to the Magellan chipset.

This group of signals, along with ground connections, are available at J14, and comprise an interface directly compatible with the Atlas Developer's Kit boards. See Section 3.2.8, "SPI Atlas Bus Connector (J14)" for a detailed description of J14, the SPI Atlas bus connector.

2.4.6 Pulse and Direction Out

For pulse & direction amplifiers these signals provide an output stream of pulse and direction data compatible with a wide variety of off-the-shelf step motor amplifiers. These signals are directly generated by the Magellan Motion Control IC. For more information on output waveforms, pulse rates, and related information, see the *Magellan Motion Control IC User Guide*.

These signals are named *pulse1-4, direction1-4*.

The default value, upon powerup, for all pulse & direction output signals is high.

Connections & Associated Signals

This group of signals are direct single-signal digital outputs. There are no associated connections that need to be made for these signals to function properly; however, one or more of the digital grounds must be connected.

See Chapter 3, "Magellan Developer's Kit Electrical Specifications" for a complete description of the pinout connections to/from the board.

2.4.7 PWM Out

For servo or microstepping motors these signals provide PWM (pulse width modulated) motor command signals when the motor output mode is set to **PWMSignMagnitude** or **PWM5050Magnitude**. See the *Magellan Motion Control IC User Guide* for complete information. The number of signals per axis varies depending on the motor type being connected to, the number of phases that the motor has, and the motor drive method (sign/magnitude or 50/50). Chapter 3, "Magellan Developer's Kit Electrical Specifications" has complete connection tables for various motor configurations.

These signals are named ***PWMMag1A-4C*** (12 signals) and ***PWMSign1A-4B*** (8 signals).

Connections & Associated Signals

This group of signals are direct single-signal digital outputs. There are no associated connections that need to be made for these signals to function properly; however, one or more of the digital grounds must be connected.

See Chapter 3, "Magellan Developer's Kit Electrical Specifications" for a complete description of the pinout connections to/from the board.

2.4.8 Motor Command

For servo or microstepping motors, these signals contain the analog motor command when the motor output mode is set to DAC (digital to analog converter). These signals vary between -10V and +10V. See the *Magellan Motion Control IC User Guide* for more information. The number of signals per axis varies depending on the motor type being connected to.

These signals are named ***DAC1A - DAC4B*** (8 signals).

Connections & Associated Signals

For analog voltages to be output correctly, ***GND*** (motor command ground) must be connected.

See Chapter 3, "Magellan Developer's Kit Electrical Specifications" for a complete description of the pinout connections to/from the board.

This page intentionally left blank.

3. Electrical Specifications

3

In This Chapter

- Magellan Configuration Switch Blocks
- Magellan Connectors
- Outputs to Motor Amplifiers
- Encoder Inputs
- Environmental and Electrical Ratings

Figure 3-1 shows the locations of the principal components of the developer kit. The component side of the board is shown. All component locations in this manual refer to this orientation. In Figure 3-1 the motion control IC's CP chip and IO chip are identified for reference. All other components of interest to the user are identified by their board label:

- Switch blocks S1 and S2 set the serial interface configuration and multi-drop address
- Switch blocks S3 and S4 set the CANbus interface configuration
- Switch blocks S5 and S6 select the motor type (used with MC58000 family products only) along with the encoding format of the QuadA and QuadB signals
- Resistor Packs RS1-RS3 select whether single-ended or differential encoder are connected
- Jumper JP15 selects whether CANbus node is last in the network
- Jumpers JP16-18 configure the board to work with Atlas-compatible Magellans, or with pre-Atlas Magellans
- Connectors J1, J2, J4, J5, J10, J13, J14 and JS1 provide connections to and from external components

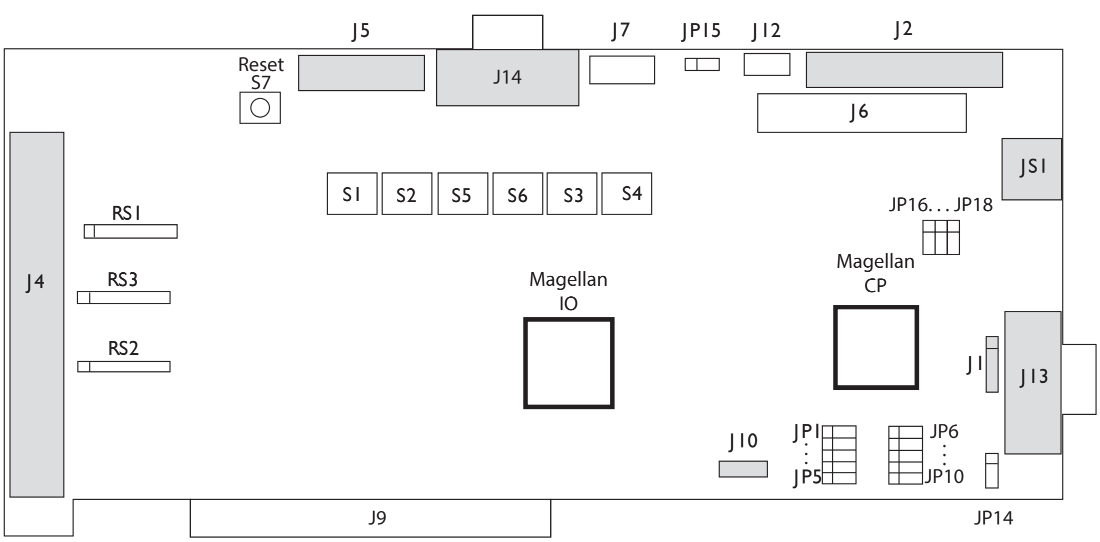


Figure 3-1: Location of Various Board Elements

3.1 Magellan Configuration Switch Blocks

These switch blocks are oriented horizontally on the developer's kit board; on is up.

3.1.1 S1: Serial Transmission Parameters

Switch block S1 sets the transmission rate, parity, stop bits, and protocol. Switch block S2 selects the device address when using the multi-drop protocol. When referring to the table below, the switch up position is relative to the bottom of the board where the PCI connector is located. The up position on the switch is marked **on.**

		S1-1	S1-2	S1-3	S1-4	S1-5	S1-6	S1-7	S1-8	S2-1
Transmission rate (bits per second)	1200	up	up	up	up					
	2400	down	up	up	up					
	9600	up	down	up	up					
	19200	down	down	up	up					
	57600	up	up	down	up					
	115200	down	up	down	up					
	230400	up	down	down	up					
	460800	down	down	down	up					
Parity	None					up	up			
	Odd					down	up			
	Even					up	down			
Stop bits	1							up		
	2							down		
Protocol	Point-to-point								up	up
	Address bit								up	down
	Idle line								down	down

3.1.2 S2: Serial Device Address

Switch block S3 sets the device address for multi-drop protocol systems. When referring to the table below, the switch up position is relative to the bottom of the board where the PCI connector is located. The up position on the switch is marked **on**.

S2-2 and S2-3 must be left in the up position.

Multi-drop address selection (S2)

Address	S2-4	S2-5	S2-6	S2-7	S2-8
0	up	up	up	up	up
1	down	up	up	up	up
2	up	down	up	up	up
3	down	down	up	up	up
4	up	up	down	up	up
5	down	up	down	up	up

Multi-drop address selection (S2)

Address	S2-4	S2-5	S2-6	S2-7	S2-8
6	up	down	down	up	up
7	down	down	down	up	up
8	up	up	up	down	up
9	down	up	up	down	up
10	up	down	up	down	up
11	down	down	up	down	up
12	up	up	down	down	up
13	down	up	down	down	up
14	up	down	down	down	up
15	down	down	down	down	up
16	up	up	up	up	down
17	down	up	up	up	down
18	up	down	up	up	down
19	down	down	up	up	down
20	up	up	down	up	down
21	down	up	down	up	down
22	up	down	down	up	down
23	down	down	down	up	down
24	up	up	up	down	down
25	down	up	up	down	down
26	up	down	up	down	down
27	down	down	up	down	down
28	up	up	down	down	down
29	down	up	down	down	down
30	up	down	down	down	down
31	down	down	down	down	down

3.1.3 CANbus Transceiver

The following table shows how to configure the CANbus port. When referring to the table below, the switch up position is relative to the bottom of the board where the PCI connector is located. The up position on the switch is marked **on**.

> S3-8 and S4-1 through S4-5 should be left in the on position.

Switches		Description
S3	1-7	Node ID Setting Switches 1-7 of the S3 DIP switch control the CANbus nodeID, allowing the board to be uniquely addressed on the CANbus network. Switches 1-7 are set binary-encoded, with total range of allowed values 0 - 127, and with 1 being least significant, 7 most significant. If a switch is up, it encodes a bit value of 0 and if it is down it encodes a bit value of 1.

S3-1	*S3-2*	*S3-3*	*S3-3*	*S3-4*	*S3-5*	*S3-6*	*S3-7*	*NodeID*
up	up	up	up	up	up	up	up	0
down	up	up	up	up	up	up	up	1
up	down	up	up	up	up	up	up	2
down	down	up	up	up	up	up	up	3
up	up	down	up	up	up	up	up	4
down	up	down	up	up	up	up	up	5
up	down	down	up	up	up	up	up	6
down	down	down	up	up	up	up	up	7
. . .								
down	down	down	down	down	down	down	down	127

Switches		Description
S4	8-16	Communication rate. These switches set the CANbus communication rate as follows:

S4-6	*S4-7*	*S4-8*	*Baud setting*
up	up	up	1,000,000 baud
down	up	up	800,000 baud
up	down	up	500,000 baud
down	down	up	250,000 baud
up	up	down	125,000 baud
down	up	down	50,000 baud
up	down	down	20,000 baud
down	down	down	10,000 baud

If the attached CANbus device is the last node on the CAN network, then JP15 should be left in the default position of 2-3 jumpered. If this it is not the last device, then it should be installed with 1-2 jumpered.

3.1.4 Motor Type Switch Settings

When using the DK55000 only pulse and direction motors are used, and it is not necessary to set jumpers related to motor type. When using the DK58000 it is possible to support any combination of DC Brush, Brushless DC, microstepping, and pulse & direction motors all on the same board. To accomplish this dip switches 5 and 6 must be set to indicate the motor type that will be used for each axis.

When configuring the dip switches and connecting your motors to the DK58420 board, the following information may be helpful:

- *Brushless DC* means the board expects to connect to a Brushless DC motor with Hall sensors and an encoder. With this connection, the DK58420 board performs the commutation and outputs a multi-phase signal, 2 or 3 phases per axis, to the amplifier. For Atlas-connected axes only 2 phase may be selected.
- *Pulse and direction* is used with step motor Atlas amplifiers, or to connect to a step motor which uses a standard pulse and direction amplifier. Quadrature feedback is optional with this type of motor.

- *Microstepping* means the DK58420 board outputs a multi-phase signal, 2 or 3 phases per axis, to a step motor amplifier that can accept this type of output. Quadrature feedback is optional with this type of motor.
- *DC Brush* means the board expects to connect to a DC Brush motor with an encoder, or an externally commutated Brushless DC motor. With this motor type the board outputs one phase per axis.

Motor Type Jumper settings (for DK58000 only)

When referring to the table below the switch up position is relative to the bottom of the board where the PCI connector is located. The up position on the switch is marked on.

Item	Switches	Description			
Dip	S5-1	Axis #1 Motor type setting			
switch	S5-2	Set S5 1-3 dip switches according to the motor type you will be using on axis #1			
S5	S5-3	5-1	5-2	5-3	Axis #1
		up	up	up	Brushless DC (3 phase)
		down	up	up	Closed loop stepper
		up	down	up	Microstepping (3 phase)
		down	down	up	Microstepping (2 phase)
		up	up	down	Pulse & direction
		down	down	down	DC Brush (default setting)
Dip	S5-5	Axis #2 Motor type setting			
switch	S5-6	Set S5 5-7 dip switches according to the motor type you will be using on axis #2			
S5	S5-7	5-5	5-6	5-7	Motor type setting
		up	up	up	Brushless DC (3 phase)
		down	up	up	Closed loop stepper
		up	down	up	Microstepping (3 phase)
		down	down	up	Microstepping (2 phase)
		up	up	down	Pulse & direction
		down	down	down	DC Brush (default setting)
Dip	S6-1	Axis #3 Motor type setting			
switch	S6-2	Set S6 1-3 dip switches according to the motor type you will be using on axis #3			
S5	S6-3	6-1	6-2	6-3	Motor type setting
		up	up	up	Brushless DC (3 phase)
		down	up	up	Closed loop stepper
		up	down	up	Microstepping (3 phase)
		down	down	up	Microstepping (2 phase)
		up	up	down	Pulse & direction
		down	down	down	DC Brush (default setting)
Dip	S6-5	Axis #4 Motor type setting			
switch	S6-6	Set S6 5-7 dip switches according to the motor type you will be using on axis #4			
S5	S6-7	6-5	6-6	6-7	Motor type setting
		up	up	up	Brushless DC (3 phase)
		down	up	up	Closed loop stepper
		up	down	up	Microstepping (3 phase)
		down	down	up	Microstepping (2 phase)
		up	up	down	Pulse & direction
		down	down	down	DC Brush (default setting)

Unconnected motors can be left at the default setting of *DC Brush*.

3.1.5 QuadA, QuadB Signal Interpretation

The QuadA and QuadB position feedback signals can be provided to the Magellan as is, or after conversion from pulse & direction encoding to quadrature encoding. For quadrature encoders and most position feedback devices, the default encoding of quadrature should be used.

To input position information via the QuadA and QuadB signals from pulse & direction controllers, enable the conversion by selecting pulse & direction encoding.

Item	Switch	Description
Dip switch S5-4	up	Sets QuadA and QuadB interpretation for Axis 1 to quadrature
	down	Sets QuadA and QuadB interpretation for Axis 1 to pulse & direction
Dip switch S5-8	up	Sets QuadA and QuadB interpretation for Axis 2 to quadrature
	down	Sets QuadA and QuadB interpretation for Axis 2 to pulse & direction
Dip switch S6-4	up	Sets QuadA and QuadB interpretation for Axis 3 to quadrature
	down	Sets QuadA and QuadB interpretation for Axis 3 to pulse & direction
Dip switch S6-8	up	Sets QuadA and QuadB interpretation for Axis 4 to quadrature
	down	Sets QuadA and QuadB interpretation for Axis 4 to pulse & direction

3.1.6 Resistor Packs

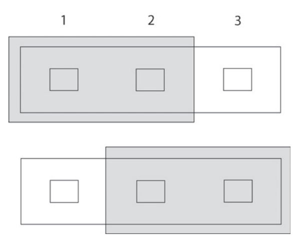
To prepare your board for installation the following user-settable hardware option should be checked:

Item	How to set	Description
Resistor packs -	Installed this is the default setting of resistor packs RS1-RS3	If you are using differential connections leave these resistor packs installed.
RS1, RS2, RS3	Removed	If you are using single-ended encoder connections, remove the resistor packs.

3.1.7 CANbus Termination Jumper (JP15)

If you are using the DK58420 Developer Kit board with a CANbus network, then the CANbus Termination Jumper (JP15) should be set.

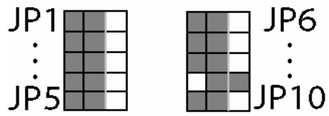
Jumper 1-2 if this board is not the last node on the network, and jumper 2-3 (default jumpering) if board is the last (or only) node on network. See diagram below for more details.



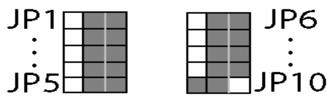
3.1.8 Device Type Selection Jumpers (JP1-JP10)

These jumpers are used to select signal connection paths appropriate for the type of Magellan Motion Control IC that is installed. These jumpers are set at the factory and should not be changed unless a motion control IC is installed that requires alternate settings. If this is the case a PMD representative will provide specific instructions on the new jumper settings.

Jumper positions for the MC58x20 and MC55x20



Jumper positions for the MC58110 and MC55110



3.1.9 Atlas-Compatible Magellan Jumpers (JP16-JP18)

To select signal connection paths appropriate for Atlas Digital Amplifiers jumpers JP16, JP17, and JP18 should be connected at the 2-3 position.

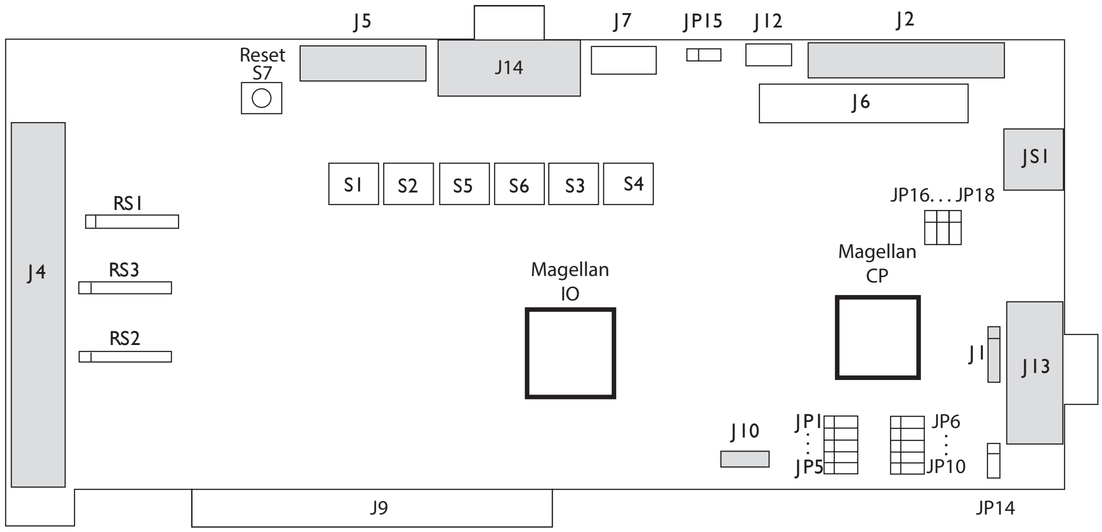
To select signal paths appropriate for non-Atlas Digital external amplifiers jumpers JP16 and JP17 should be connected at the 2-3 position and JP18 to 1-2.

3.2 Magellan Connectors

This section describes the pinouts for the following cable connectors on the DK58420 board (shaded areas in diagram below):

J1	5-pin serial communication connector
J2	16-pin analog input connector
J4	100-pin main connector containing encoder input, Hall input, AxisIn signals, AxisOut signals, Motor output signals, and limit switch inputs
J5	20-pin user-defined digital I/O connector
J10	4-pin 5v power connector
J13	9-pin DB9 serial port connector
J14	9-pin DB9 SPI ATLAS connector
JS1	4-pin CANbus connector

Figure 3-2: Connector Locator



3.2.1 Serial Communications Connector (J1)

J1 is a 5-pin single row header (0.1" spacing).

Connector J1 is used with multi-drop serial communications. Connector J13 is used with serial point-to-point communications. See Section 2.3.2, "Serial Transceiver" for more information.

Pin number	Signal name
1	SrlXmt (CP pin 44)
2	SrlRcv (CP pin 43)
3	SrlEnable (CP pin 99)
4	GND
5	3.3V

3.2.2 Analog Input Connector (J2)

This is a 16-pin header (2x13, 0.1" spacing).

Pin number	Signal name	Pin number	Signal name
1	Analog0	9	AnalogRefHigh
2	Analog1	10	AnalogRefLow
3	Analog2	11	AnalogGND
4	Analog3	12	AnalogGND
5	Analog4	13	AnalogVcc
6	Analog5	14	~HostIntrpt*
7	Analog6	15	Gnd*
8	Analog7	16	3.3V*

*~HostIntrpt, along with Gnd and 3.3V are connected to the corresponding pins on the Magellan Motion Control IC. They can be used to externalize the ~HostIntrpt signal from the Magellan Motion Control IC if this is desired.

3.2.3 Motion Peripherals Connector (J4)

This is a 100-pin high-density connector (2x50, 0.05" spacing). The accompanying cable assembly supplied with your developer's kit consists of two 36" flat ribbon cables terminating together at one end in the matching 100-pin connector. At the other end, each ribbon terminates in a 50-pin header (2x25, 0.1" spacing). The ribbons are labeled Hdr1 and Hdr2. Pins 1-50 on Hdr1 connect to pins 1-50 of J4. Pins 1-50 of Hdr2 connect to pins 51-100 of J4.

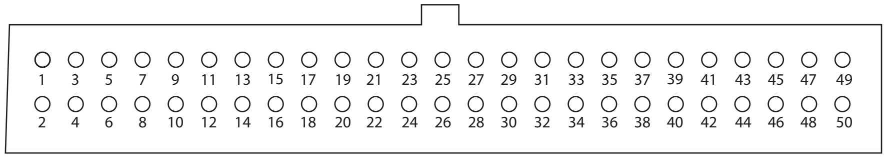
Header 1 (to J4 pins 1-50)				**Header 2 (to J4 pins 51-100)**			
Pin	Signal name	Pin	Signal name	Pin	Signal name	Pin	Signal name
1	QuadA1+	26	QuadA2+	1	QuadA3+	26	QuadA4+
2	QuadA1-	27	QuadA2-	2	QuadA3-	27	QuadA4-
3	QuadB1+	28	QuadB2+	3	QuadB3+	28	QuadB4+
4	QuadB1-	29	QuadB2-	4	QuadB3-	29	QuadB4-
5	Index1+	30	Index2+	5	Index3+	30	Index4+
6	Index1-	31	Index2-	6	Index3-	31	Index4-
7	Vcc (encoder)	32	Vcc (encoder)	7	Vcc (encoder)	32	Vcc (encoder)
8	GND (encoder)	33	GND (encoder)	8	GND (encoder)	33	GND (encoder)
9	Hall1A	34	Hall2A	9	Hall3A	34	Hall4A
10	Hall1B	35	Hall2B	10	Hall3B	35	Hall4B
11	Hall1C	36	Hall2C	11	Hall3C	36	Hall4C
12	GND (Hall)	37	GND (Hall)	12	GND (Hall)	37	GND (Hall)
13	PosLim1	38	PosLim2	13	PosLim3	38	PosLim4
14	NegLim1	39	NegLim2	14	NegLim3	39	NegLim4
15	Home1	40	Home2	15	Home3	40	Home4
16	AxisIn1	41	AxisIn2	16	AxisIn3	41	AxisIn4
17	AxisOut1	42	AxisOut2	17	AxisOut3	42	AxisOut4
18	PWMMagA1/Pulse1	43	PWMMagA2/Pulse2	18	PWMMagA3/Pulse3	43	PWMMagA4/Pulse4
19	PWMMagB1	44	PWMMagB2	19	PWMMagB3	44	PWMMagB4
20	PWMMagC1/ PWMsignB1/ AtRest1	45	PWMMagC2/ PWMsignB2/ AtRest2	20	PWMMagC3/ PWMsignB3/ AtRest3	45	PWMMagC4/ PWMsignB4/ AtRest4
21	PWMsignA1/Direction1	46	PWMsignA2/Direction2	21	PWMsignA3/Direction3	46	PWMsignA4/Direction4
22	DACA1*	47	DACA2*	22	DACA3*	47	DACA4*
23	DACB1*	48	DACB2*	23	DACB3*	48	DACB4*
24	GND (DAC)	49	GND (DAC)	24	GND (DAC)	49	GND (DAC)
25	3.3V	50	3.3V	25	3.3V	50	3.3V

> Unused signals may be left unconnected. Each converter cable has labels to indicate HDR1 and HDR2.

3.2.4 Connector Pin Layout

The following diagram shows the pin layout for the two 50-pin header cables used with the Magellan board.

Figure 3-3: Bottom view of 50-pin Header Connector



For testing purposes, this connector can be mated with a terminal block. PMD suggests Phoenix Contact (http://www.phoenixcon.com/), part number FLKM 50 (Digi-Key part number 2281089-ND).

3.2.5 User-defined Digital IO Connector (J5)

These general-purpose IO signals source up to 4mA and can sink up to 8mA. These pins are accessed using the Magellan commands ReadIO/WriteIO.

This is a 20-pin header (2x10, 0.1" spacing).

Pin number	Signal name	Pin number	Signal name
1	PrlIn0	10	PrlOut4
2	PrlOut0	11	PrlIn5
3	PrlIn1	12	PrlOut5
4	PrlOut1	13	PrlIn6
5	PrlIn2	14	PrlOut6
6	PrlOut2	15	PrlIn7
7	PrlIn3	16	PrlOut7
8	PrlOut3	17, 19	GND
9	PrlIn4	18, 20	3.3V

3.2.6 Standalone Power Connector (J10)

This is a green 2-connection terminal block.

Pin number	Signal name
1	+5V
2	GND

3.2.7 Serial Port Connector (J13)

This is a male DB9 connector.

Connector J1 is used with multi-drop serial communications. Connector J13 is used with serial point-to-point communications. See Section 2.3.2, "Serial Transceiver" for more information.

Pin number	Signal name
1	No connection
2	Serial Transmit
3	Serial Receive
4	No connection
5	Gnd
6	No connection
7	No connection
8	No connection
9	No connection

3.2.8 SPI Atlas Bus Connector (J14)

This is a female DB-9 connector.

This connector is designed to be compatible with the Atlas DK bus system of cables and boards. See the *Atlas Digital Amplifier User Manual*, *Appendix A*, for more information on Atlas DKs

Pin number	Signal name
1	SPIEnable3
2	SPIEnable2
3	GND
4	GND
5	SPIRcv
6	SPIEnable1
7	SPIEnable4
8	SPIClock
9	SPIXmt

3.2.9 CANbus Connector (JS1)

This is a 4-pin mini-DIN connector.

Pin number	Signal name
1	CANH
2	CANL
3	GND
4	No connection

3.3 Outputs to Motor Amplifiers

When connecting any axis to an Atlas amplifier the SPI ATLAS bus connector (J14) is used.

For axes that are not connected to Atlas amplifiers, there are four types of output to the motor amplifiers:

DAC	Analog signals from the on-board D/A converters
PWM 50/50	Pulse-width modulated square-wave signals with a 50% duty cycle
PWM sign-magnitude	Pulse-width modulated signals with definable duty cycle and direction
Pulse & direction	Step-motor output digital pulse and direction signals

These outputs should be connected from the designated J4 pins to the appropriate amplifier inputs, as shown in the following tables. The names of the input pins may vary among amplifiers; common names are shown.

3.3.1 DK58420

DC Brush Motors

			J4 connection (Header-pin)			
	Signal name	*Amplifier input*	*Axis 1*	*Axis 2*	*Axis 3*	*Axis 4*
DAC	DACA*n*	Ref+ or V+	Hdr1-22	Hdr1-47	Hdr2-22	Hdr2-47
	GND*n*	Ref- or Gnd	Hdr1-24	Hdr1-49	Hdr2-24	Hdr2-49
PWM sign/magnitude	PWMMagA*n*	PWM magnitude	Hdr1-18	Hdr1-43	Hdr2-18	Hdr2-43
	PWMSignA*n*	PWM direction	Hdr1-21	Hdr1-46	Hdr2-21	Hdr2-46
	GND	GND	Hdr1-8	Hdr1-33	Hdr2-8	Hdr2-33

3.3.2 Three-Phase Brushless Motors

			J4 connection (Header-pin)			
	Signal name	*Amplifier input*	*Axis 1*	*Axis 2*	*Axis 3*	*Axis 4*
DAC	DACA*n*	Ref1+ or V1+	Hdr1-22	Hdr1-47	Hdr2-22	Hdr2-47
	DACB*n*	Ref2+ or V2+	Hdr1-23	Hdr1-48	Hdr2-23	Hdr2-48
	GND*n*	Ref- or Gnd	Hdr1-24	Hdr1-49	Hdr2-24	Hdr2-49
PWM 50/50	PWMMagA*n*	PWM phase 1	Hdr1-18	Hdr1-43	Hdr2-18	Hdr2-43
	PWMMagB*n*	PWM phase 2	Hdr1-19	Hdr1-44	Hdr2-19	Hdr2-44
	PWMMagC*n*	PWM phase 3	Hdr1-20	Hdr1-45	Hdr2-20	Hdr2-45
	GND	GND	Hdr1-8	Hdr1-33	Hdr2-8	Hdr2-33

3.3.3 Two-Phase Step Motors

			J4 connection (Header-pin)			
	Signal name	*Amplifier input*	*Axis 1*	*Axis 2*	*Axis 3*	*Axis 4*
DAC	DACA*n*	Ref+ or V+	Hdr1-22	Hdr1-47	Hdr2-22	Hdr2-47
	DACBn	Ref+ or V+	Hdr1-23	Hdr1-48	Hdr2-23	Hdr2-48
	GND*n*	Ref- or Gnd	Hdr1-24	Hdr1-49	Hdr2-24	Hdr2-49
PWM sign/magnitude	PWMMagA*n*	PWM magnitude	Hdr1-18	Hdr1-43	Hdr2-18	Hdr2-43
	PWMSignA*n*	PWM direction	Hdr1-21	Hdr1-46	Hdr2-21	Hdr2-46
	PWMMagB*n*	PWM magnitude	Hdr1-19	Hdr1-44	Hdr2-19	Hdr2-44
	PWMSignB*n*	PWM direction	Hdr1-20	Hdr1-45	Hdr2-20	Hdr2-45
	GND	GND	Hdr1-8	Hdr1-33	Hdr2-8	Hdr2-33

3.3.4 DK58420 and DK55420

Pulse & Direction Motors

		J4 connection (Header-pin)			
Signal name	*Amplifier input*	*Axis 1*	*Axis 2*	*Axis 3*	*Axis 4*
Pulse *n*	Pulse or step	Hdr1-18	Hdr1-43	Hdr2-18	Hdr2-43
Direction *n*	Direction	Hdr1-21	Hdr1-46	Hdr2-21	Hdr2-46
GND	GND	Hdr1-8	Hdr1-33	Hdr2-8	Hdr2-33

3.4 Encoder Inputs

Resistor packs RS1 - RS3

These three resistor packs are at the left end of the developer's kit board, next to the 100-pin connector J4. When using differential encoders, leave these packs in place. When using open-ended encoders, remove all three packs and connect encoder signals to the positive encoder input only. The negative input can be left unconnected. Encoder connections are shown below.

Encoder connections when using differential encoder input

Signal	**J4 pin connections**			
	Axis 1	Axis 2	Axis 3	Axis 4
QuadAn+	Hdr1-1	Hdr1-26	Hdr1-51	Hdr1-76
QuadAn-	Hdr1-2	Hdr1-27	Hdr1-52	Hdr1-77
QuadBn+	Hdr1-3	Hdr1-28	Hdr1-53	Hdr1-78
QuadBn-	Hdr1-4	Hdr1-29	Hdr1-54	Hdr1-79
Indexn+	Hdr1-5	Hdr1-30	Hdr1-55	Hdr1-80
Indexn-	Hdr1-6	Hdr1-31	Hdr1-56	Hdr1-81
Vcc	Hdr1-7	Hdr1-32	Hdr1-57	Hdr1-82
GND	Hdr1-8	Hdr1-33	Hdr1-58	Hdr1-83

Encoder connections when using single-ended encoder input

Signal	**J4 pin connections**			
	Axis 1	Axis 2	Axis 3	Axis 4
QuadAn	Hdr1-1	Hdr1-26	Hdr1-51	Hdr1-76
QuadBn	Hdr1-3	Hdr1-28	Hdr1-53	Hdr1-78
Indexn	Hdr1-5	Hdr1-30	Hdr1-55	Hdr1-80
Vcc	Hdr1-7	Hdr1-32	Hdr1-57	Hdr1-82
GND	Hdr1-8	Hdr1-33	Hdr1-58	Hdr1-83

3.5 Environmental and Electrical Ratings

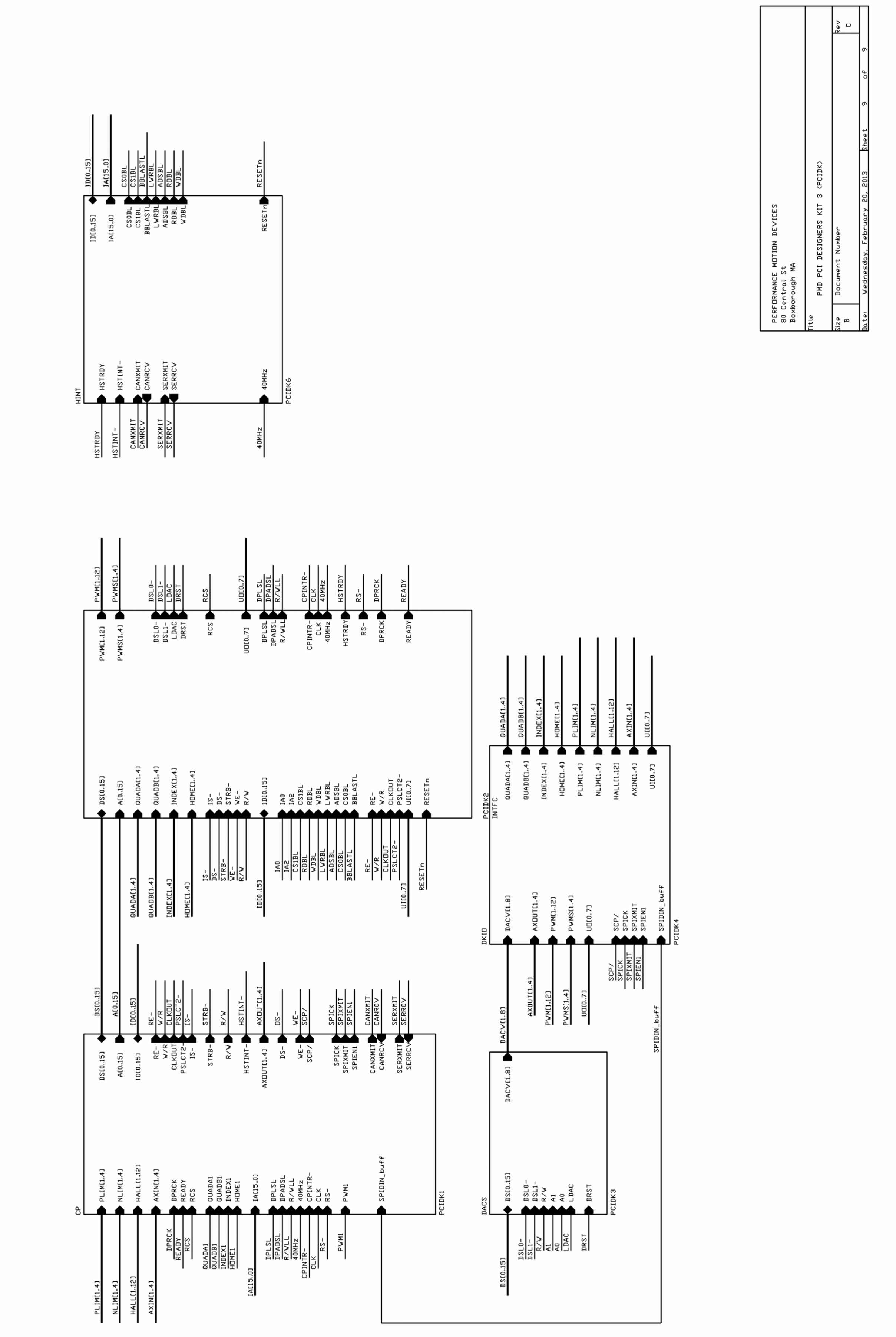
All ratings and ranges are for both the IO and CP chips.

Dimensions	4.25" x 9.25", PCI Adapter
Storage Temperature	-40 °C to 125 °C
Operating Temperature	0 °C to 70 °C*
Power Consumption	1A @ 5V; 83mA @ 12V
Supply Voltage Limits	-0.3V to +7.0V
Supply Voltage Operating Range	4.75V to 5.25V
Analog Output Range	-10.0V to 10.0V
Analog Input Range	0.0V to 3.3V
Digital Output Range	0.0V to 3.3V

A.Reference Schematics

A

Figure A-1: Developer's Kit overview schematic



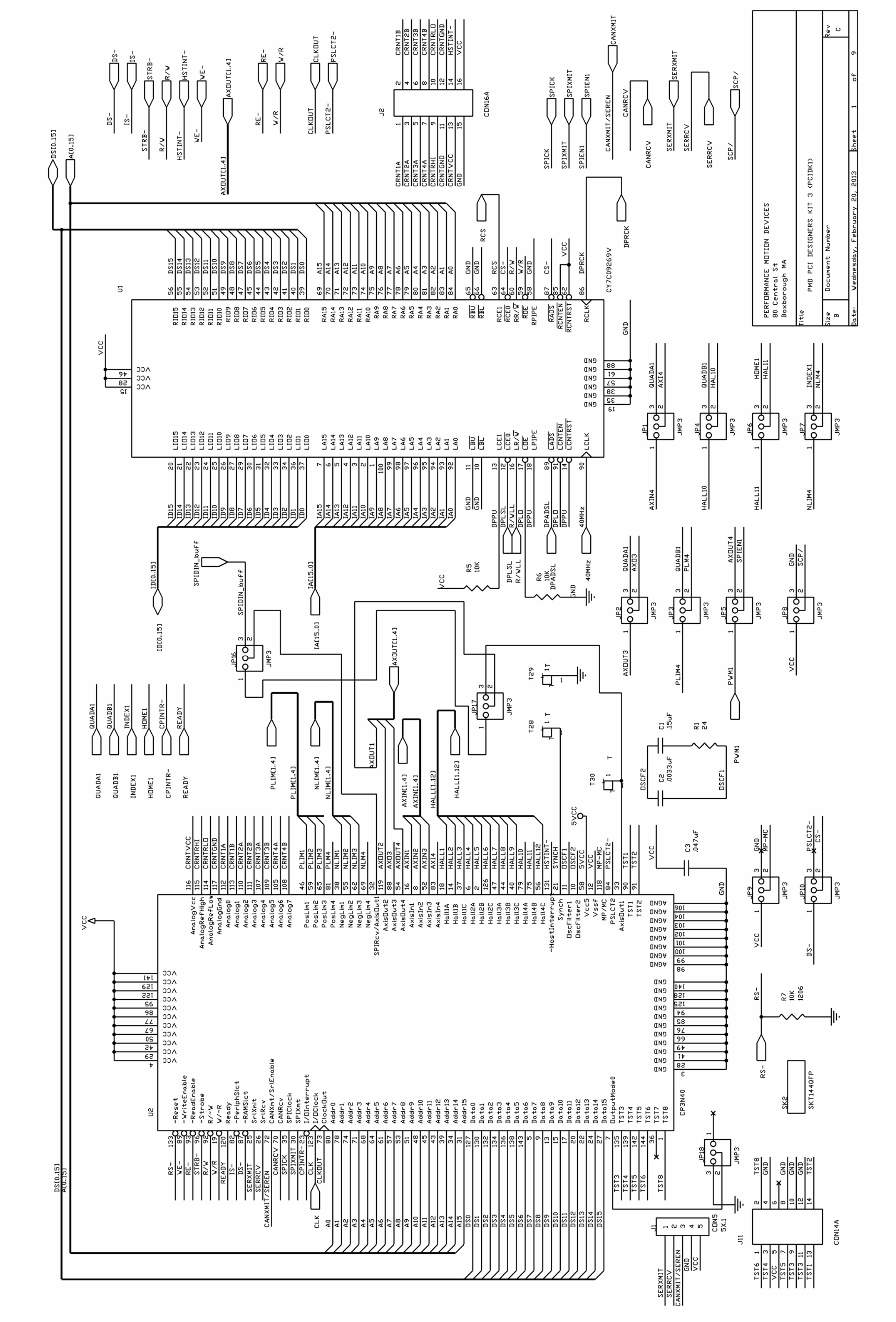
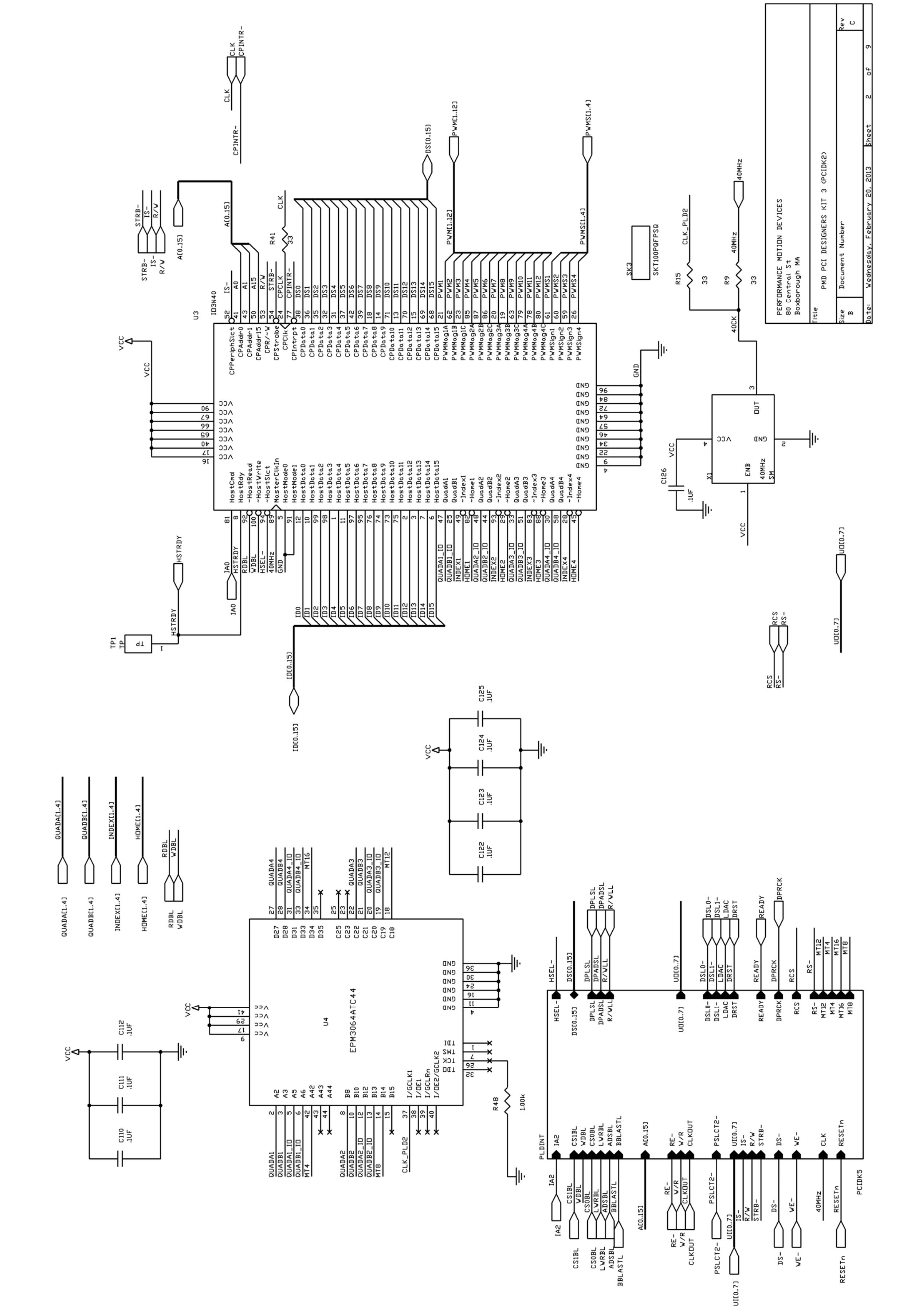


Figure A-2: Magellan Motion Control IC CP and DPRAM schematic

Figure A-3: Developer's Kit I/O schematic



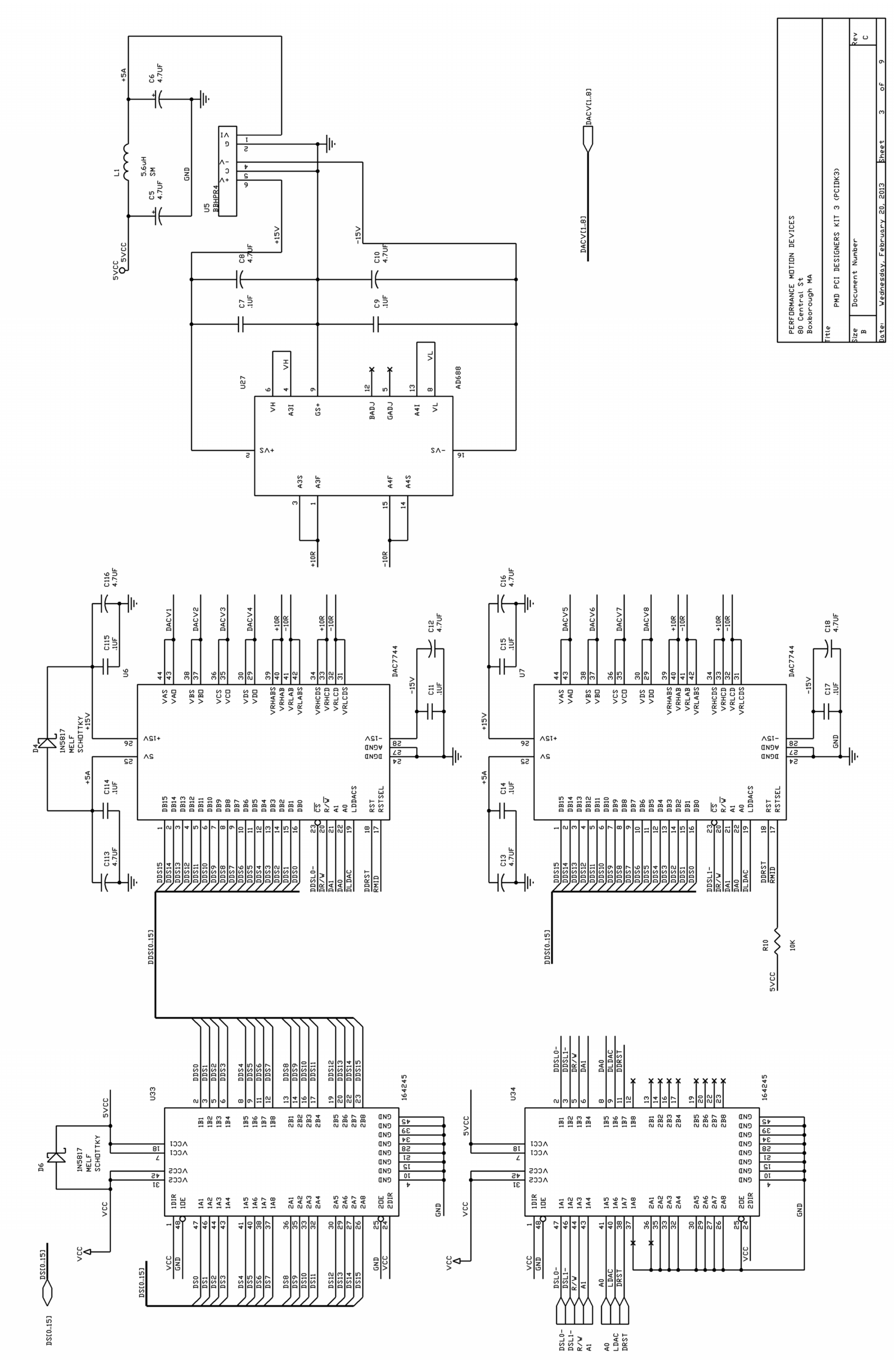
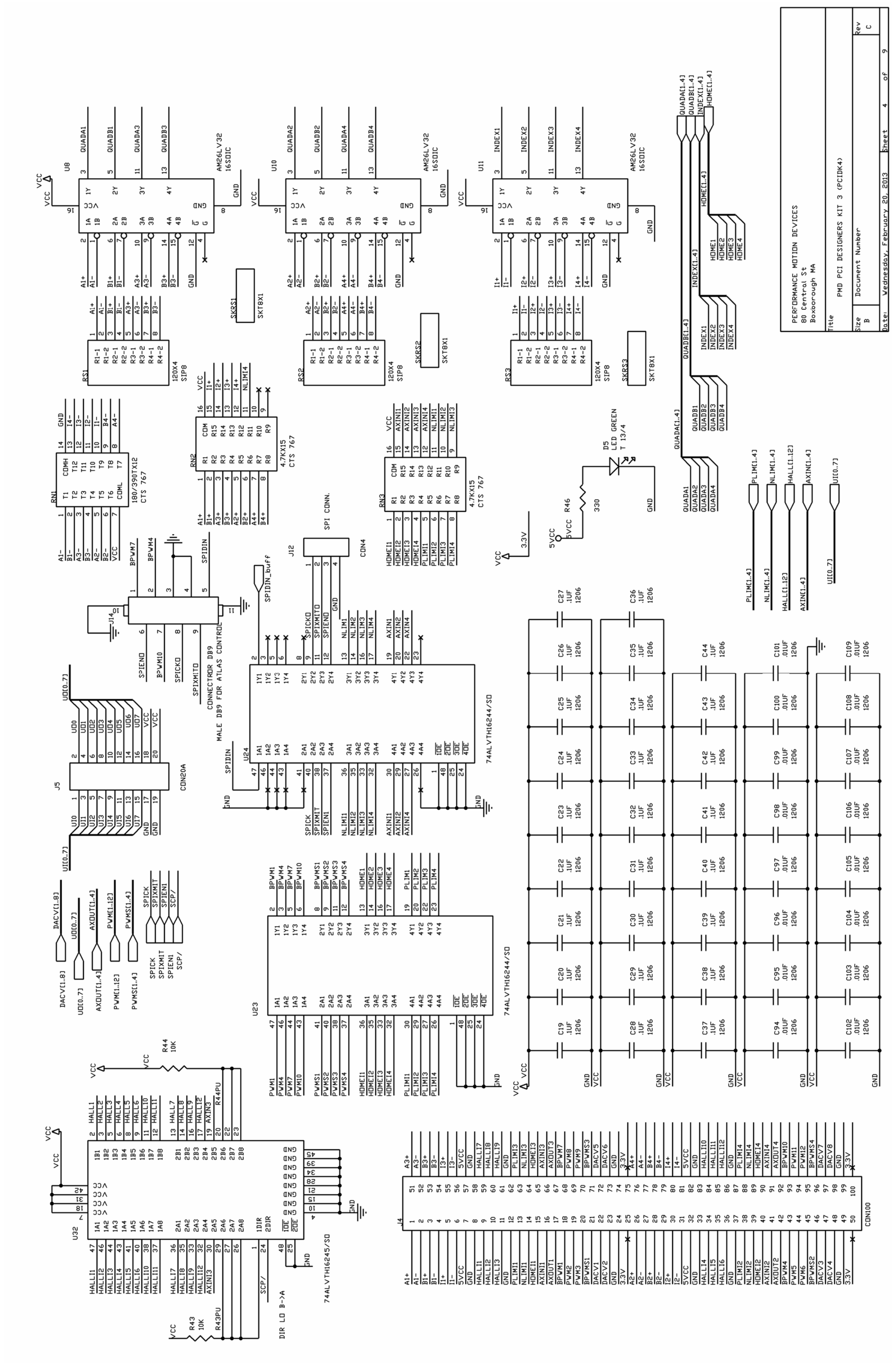


Figure A-4: Developer's Kit DAC amplifiers schematic

Figure A-5: Developer's Kit connector and quadrature schematic



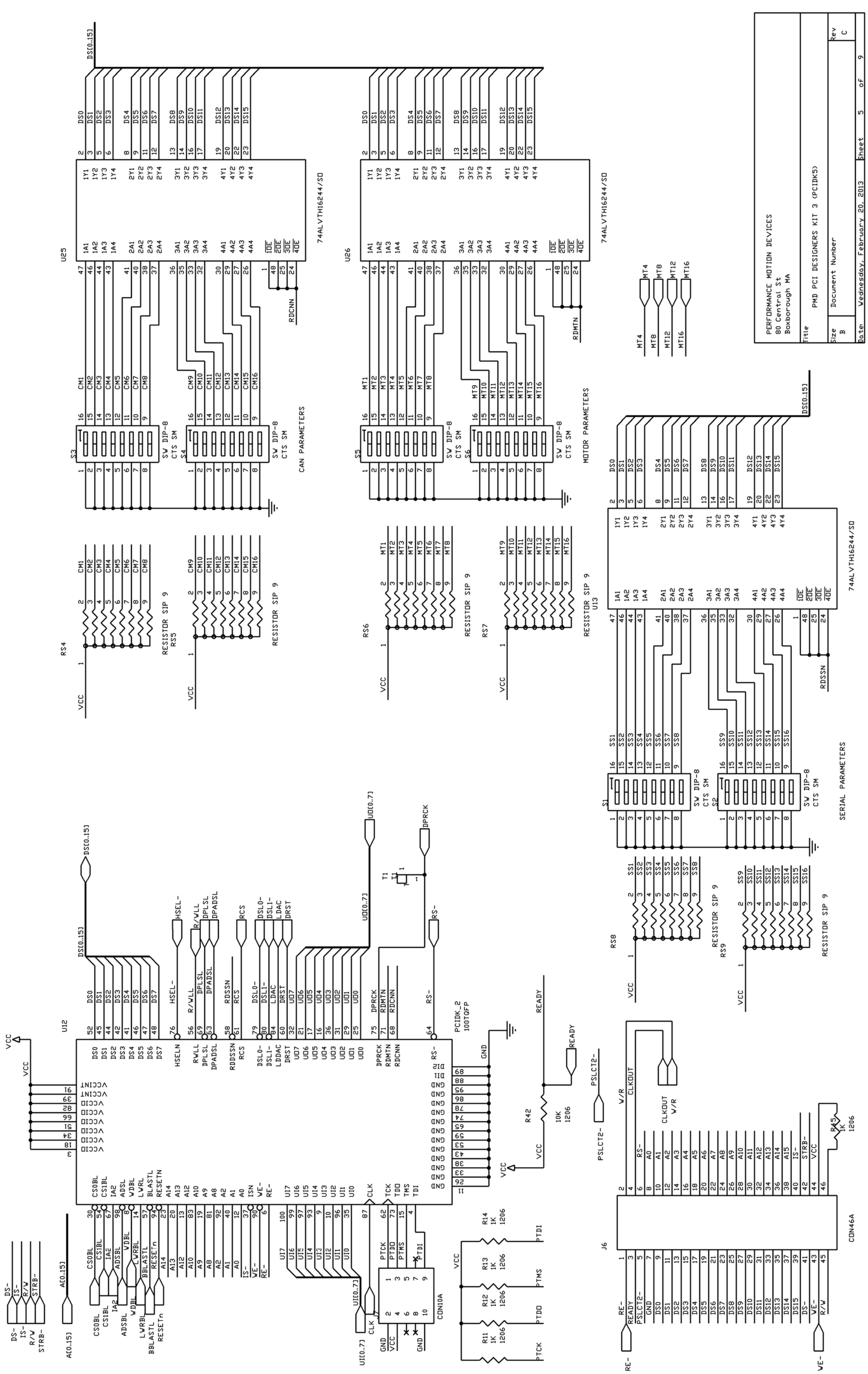
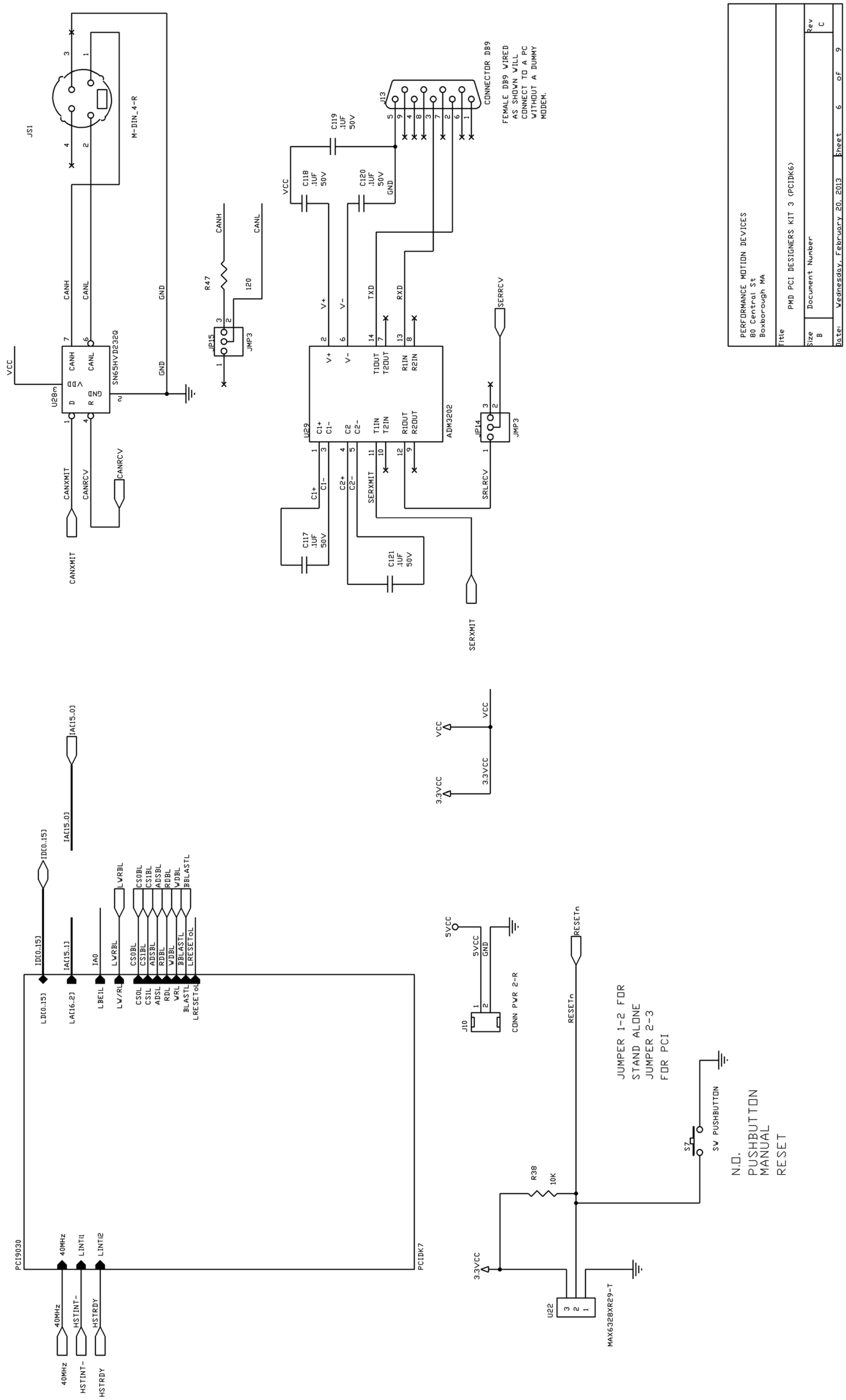


Figure A-6: Developer's Kit PLD schematic

Figure A-7: Developer's Kit transceivers schematic



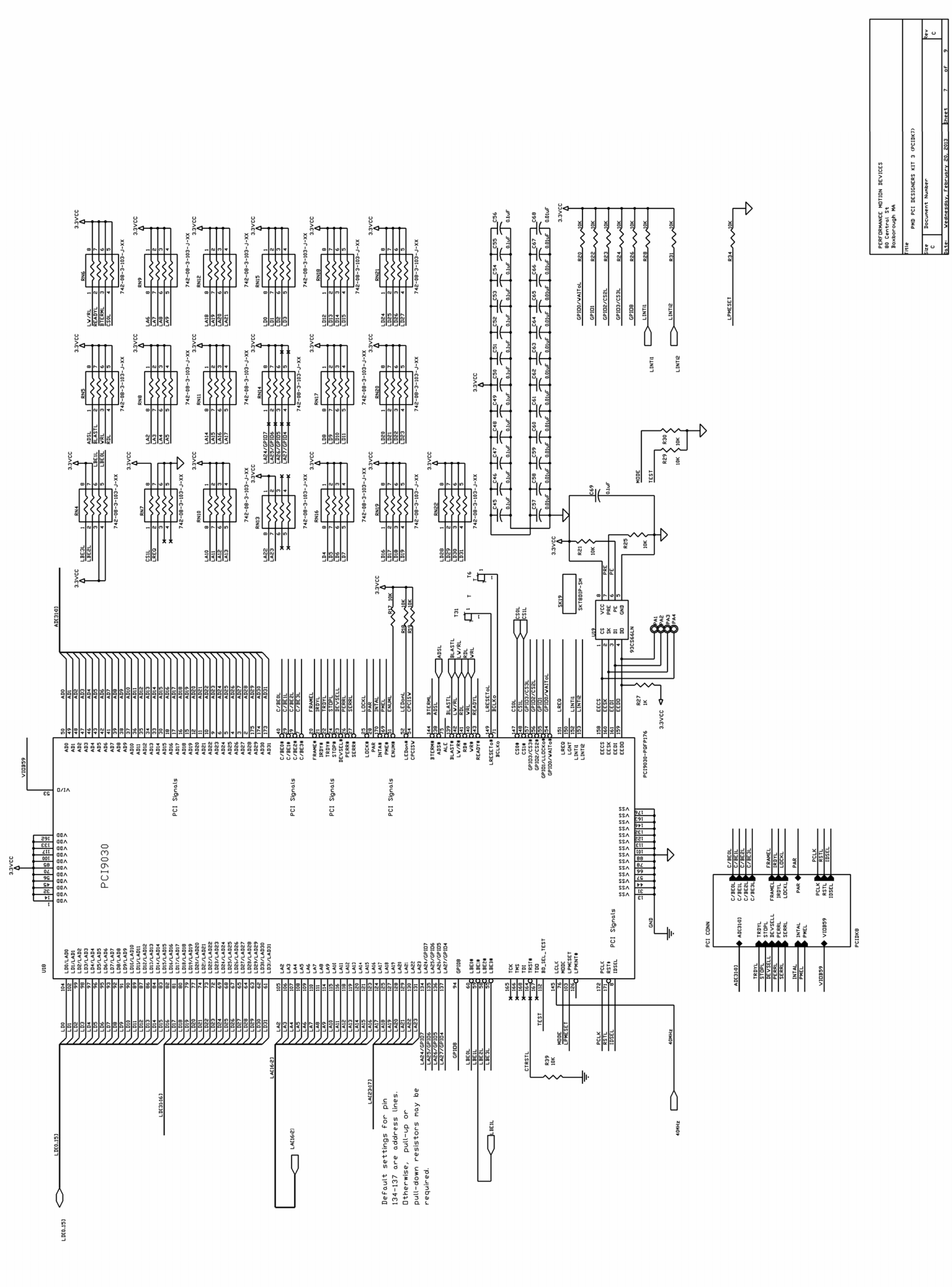
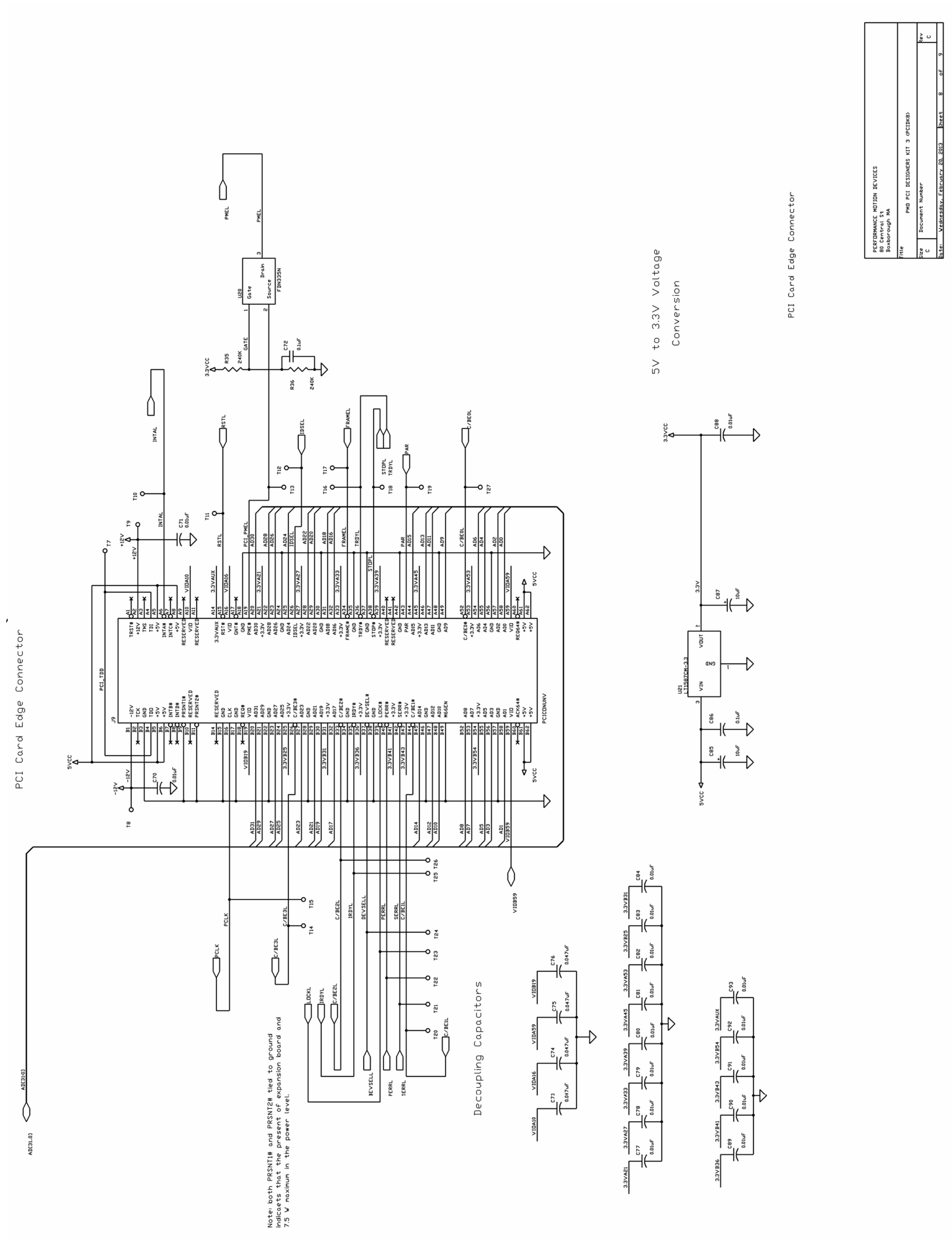


Figure A-8: Developer's Kit PLX9030 schematic

Figure A-9: Developer's Kit PCI edge connector schematic



For additional information, or for technical assistance,
please contact PMD at (978) 266-1210.

You may also e-mail your request to support@pmdcorp.com

Visit our website at http://www.pmdcorp.com

PMD

Performance Motion Devices
80 Central Street
Boxborough, MA 01719

This page intentionally left blank.