

**PROA10V80A/
PROA20V80A
SA-CAT**

**Programmable Servo Drive
with EtherCAT® for Step,
DC, Brushless DC and AC
Motors**



Programmable Servo Drive

ElectroCraft Document Number
A11357 Rev 1

**Technical
Reference**

ELECTROCRAFT

PRO-A10V80A-SA-CAT
PRO-A20V80A-SA-CAT

Technical Reference

ElectroCraft Document Number
A11357 Revision 1

ElectroCraft
1 Progress Drive
Dover, NH 03820

www.electrocrafter.com

Read This First

While ElectroCraft believes that the information and guidance given in this manual is correct, all parties must rely upon their own skill and judgment when making use of it. ElectroCraft does not assume any liability to anyone for any loss or damage caused by any error or omission in the work, whether such error or omission is the result of negligence or any other cause. Any and all such liability is disclaimed.

All rights reserved. No part or parts of this document may be reproduced or transmitted in any form or by any means, electrical or mechanical including photocopying, recording or by any information-retrieval system without permission in writing from ElectroCraft, Inc.

The information in this document is subject to change without notice.

About This Manual

This book is a technical reference manual for the **PRO-Ax0V80A-SA-CAT** family of programmable servo drives, including the following products:

PRO-A10V80A-SA-CAT (p/n P029.025.E221)

PRO-A20V80A-SA-CAT (p/n P029.026.E221)

In order to operate the **PRO-Ax0V80** drives, you need to pass through 3 steps:

- Step 1 Hardware installation**
- Step 2 Drive setup** using ElectroCraft **PRO Config** software for drive commissioning
- Step 3 Motion programming** using one of the options:
 - A **EtherCAT® master**
 - The drive's **built-in motion controller** executing a ElectroCraft's Motion Programming Language (**MPL**) program developed using ElectroCraft **MotionPRO Developer** software
 - A **distributed control** approach which combines the above options, like for example a host calling motion functions programmed on the drives in **MPL**

This manual covers **Step 1** in detail. It describes the **PRO-Ax0V80A-SA-CAT** hardware including the technical data, the connectors and the wiring diagrams needed for installation. The manual also presents an overview of the following steps, and includes the scaling factors between the real SI units and the drive internal units. For detailed information regarding the next steps, refer to the related documentation.

Notational Conventions

This document uses the following conventions:

- **PRO-Ax0V80**– all products described in this manual
- **IU units** – Internal units of the drive
- **SI units** – International standard units (meter for length, seconds for time, etc.)
- **MPL** – ElectroCraft's Motion Programming Language
- **CoE** – CAN application protocol over EtherCAT®

Trademarks

EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.

Related Documentation

Help Screens within the PROconfig software – describes how to use **PROconfig** to quickly setup any ElectroCraft PRO Series drive for your application using only 2 dialogue boxes. The output of PROconfig is a set of setup data that can be downloaded into the drive EEPROM or saved on a PC file. At power-on, the drive is initialized with the setup data read from its EEPROM. With PROconfig it is also possible to retrieve the complete setup information from a drive previously programmed. PROconfig is part of the ElectroCraft Motion PRO Suite. Motion PRO Suite is available as part of a PRO Series Drive Evaluation Kit. Please contact ElectroCraft or your local ElectroCraft sales representative for more information on obtaining MotionPRO Suite or an evaluation kit.

CoE Programming (Document No. A11358) – explains how to program the ElectroCraft programmable drives using **CAN application protocol over EtherCAT®** and describes the associated object dictionary.

Motion Programming using ElectroCraft MotionPRO Suite (Document No. A11229) – describes how to use the MotionPRO Suite to create motion programs using the ElectroCraft Motion PROgramming Language (MPL). The MotionPRO Suite platform includes **PROconfig** for the drive/motor setup, and a **Motion Wizard** for the motion programming. The Motion Wizard provides a simple, graphical way of creating motion programs and automatically generates all the MPL instructions. *With MotionPRO Suite you can fully benefit from a key advantage of ElectroCraft drives – their capability to execute complex moves without requiring an external motion controller, thanks to their built-in motion controller.* Motion PRO Suite is available as part of a PRO Series Drive Evaluation Kit. Please contact ElectroCraft or your local ElectroCraft sales representative for more information on obtaining MotionPRO Suite or an evaluation kit.

If you Need Assistance ...

If you want to ...	Contact ElectroCraft at ...
Visit ElectroCraft online	World Wide Web: www.electrocraft.com
Receive general information or assistance (see Note)	World Wide Web: www.electrocraft.com Email: drivesupport@electrocraft.com
Ask questions about product operation or report suspected problems (see Note)	Tel : +1 734.662-7771 Email: drivesupport@electrocraft.com
Make suggestions about, or report errors in documentation (see Note)	Mail: ElectroCraft 1 Progress Drive Dover, NH 03820 USA

Contents

Read This First	III
<i>Trademarks</i>	IV
<i>Related Documentation</i>	IV
1. Safety information	11
1.1. Warnings	11
1.2.	11
1.3. Cautions	12
2. Product Overview	13
2.1. Introduction	13
2.2. Key Features	14
2.3. Supported Motor-Sensor Configurations	16
2.4. PRO-Ax0V80A-SA-CAT Drive Dimensions	23
2.5. Identification Labels	23
2.6. Electrical Specifications	24
2.6.1. Operating Conditions	24
2.6.2. Storage Conditions	24
2.6.3. Mechanical Mounting	24
2.6.4. Environmental Characteristics	24
2.6.5. Logic Supply Input (+V _{LOG})	25
2.6.6. Motor Supply Input (+V _{MOT})	25
2.6.7. Enable circuit inputs (ENA1, ENA2)	25
2.6.8. Motor Outputs (A/A+, B/A-, C/B+, BR/B-)	26
2.6.9. Digital Inputs (IN0, IN1, IN2/LSP, IN3/LSN)	27
2.6.10. Digital Outputs (OUT0, OUT1, OUT2/Error, OUT3/Ready)	28
2.6.11. Digital Hall Inputs (Hall1, Hall2, Hall3)	29
2.6.12. Encoder Inputs (A1+, A1-, B1+, B1-, Z1+, Z1-, A2+, A2-, B2+, B2-, Z2+, Z2-)	29
2.6.13. Sin-Cos Encoder Inputs (Sin+, Sin-, Cos+, Cos-)	30
2.6.14. Analog 0...5V Inputs (REF, FDBK)	30
2.6.15. RS-232	30
2.6.16. EtherCAT®	31
2.6.17. Supply Output (+5V)	31

3.	Step 1. Hardware Installation	32
3.1.	Mechanical Mounting	32
3.2.	Mating Connectors	32
3.3.	Connectors and Connection Diagrams	33
3.3.1.	Connector Layout	33
3.3.2.	J1 Power supply input connector pinout.....	33
3.3.3.	J2 Motor output.....	34
3.3.4.	J3 Primary feedback connector pinout.....	34
3.3.5.	J4 Secondary feedback connector pinout	35
3.3.6.	J5 Digital, analog I/O and logic supply connector pinout.....	35
3.3.7.	J6, J7 EtherCAT® connectors	36
3.3.8.	J8 Enable circuit connector pinout	36
3.3.9.	J9 RS232 connector pinout.....	36
3.3.10.	AxisID DIP switch settings	36
3.3.11.	24V Digital I/O Connection	37
3.3.12.	5V Digital I/O Connection	39
3.3.13.	Analog Inputs Connection.....	40
3.3.14.	Motor connections	42
3.3.15.	Feedback connections.....	46
3.3.16.	Power Supply Connection	51
3.3.17.	Serial RS-232 connection	54
3.3.18.	Recommendations for EtherCAT® Wiring	55
3.3.19.	Disabling the setup table at startup	57
3.4.	Axis ID Selection for PRO-Ax0V80A-SA-CAT drives with EtherCAT	57
3.5.	DIP-Switch Settings.....	58
3.6.	LED Indicators.....	59
3.6.1.	EtherCAT® Status indicator.....	60
4.	Step 2. Drive Setup.....	62
4.1.	Installing PRO Config	62
4.2.	Getting Started with PRO Config.....	62
4.2.1.	Establish communication	63
4.2.2.	Setup drive/motor	64
4.2.3.	Selecting NPN/PNP inputs type in Setup	66
4.2.4.	Download setup data to drive/motor	66
4.2.5.	Evaluate drive/motor behavior (optional).....	67
4.3.	Changing the drive Axis ID	67

4.4.	Creating an Image File with the Setup Data	68
5.	Step 3. Motion Programming	69
5.1.	Using an EtherCAT® master	69
5.1.1.	Modes of Operation	69
5.1.2.	Checking Setup Data Consistency.....	69
5.2.	Using the built-in Motion Controller and MPL	69
5.2.1.	ElectroCraft's Motion Programming Language Overview.....	69
5.2.2.	Installing MotionPRO Developer	70
5.2.3.	Getting Started with MotionPRO Developer.....	70
5.2.4.	Creating an Image File with the Setup Data and the MPL Program.....	76
5.3.	Combining CoE with MPL	76
5.3.1.	Using MPL Functions to Split Motion between Master and Drives	76
5.3.2.	Executing MPL programs.....	77
5.3.3.	Loading Automatically Cam Tables Defined in MotionPRO Developer	77
5.3.4.	Customizing the Homing Procedures.....	78
5.3.5.	Customizing the Drive Reaction to Fault Conditions	78
6.	Scaling factors	79
6.1.	Position units.....	79
6.1.1.	Brushless / DC brushed motor with quadrature encoder on motor.....	79
6.1.2.	DC brushed motor with quadrature encoder on load and tacho on motor	79
6.1.3.	Step motor open-loop control. No feedback device	80
6.1.4.	Step motor open-loop control. Incremental encoder on load.....	80
6.1.5.	Brushless motor with sine/cosine encoder on motor	81
6.2.	Speed units	82
6.2.1.	Brushless / DC brushed motor with quadrature encoder on motor.....	82
6.2.2.	DC brushed motor with quadrature encoder on load and tacho on motor	82
6.2.3.	DC brushed motor with tacho on motor	83
6.2.4.	Step motor open-loop control. No feedback device	83
6.2.5.	Step motor closed-loop control. Incremental encoder on motor	84
6.2.6.	Brushless motor with sine/cosine encoder on motor	84
6.3.	Acceleration units.....	85
6.3.1.	Brushless / DC brushed motor with quadrature encoder on motor.....	85
6.3.2.	DC brushed motor with quadrature encoder on load and tacho on motor	85
6.3.3.	Step motor open-loop control. No feedback device	85
6.3.4.	Step motor open-loop control. Incremental encoder on load.....	85

6.3.5.	Step motor closed-loop control. Incremental encoder on motor	86
6.3.6.	Brushless motor with sine/cosine encoder on motor	87
6.4.	Jerk units	88
6.4.1.	Brushless / DC brushed motor with quadrature encoder on motor	88
6.4.2.	DC brushed motor with quadrature encoder on load and tacho on motor	88
6.4.3.	Step motor open-loop control. No feedback device	89
6.4.4.	Step motor open-loop control. Incremental encoder on load	89
6.4.5.	Step motor closed-loop control. Incremental encoder on motor	89
6.4.6.	Brushless motor with sine/cosine encoder on motor	89
6.5.	Current units	91
6.6.	Voltage command units	91
6.7.	Voltage measurement units	91
6.8.	Time units	92
6.9.	Master position units	92
6.10.	Master speed units	92
6.11.	Motor position units	92
6.11.1.	Brushless / DC brushed motor with quadrature encoder on motor	92
6.11.2.	DC brushed motor with quadrature encoder on load and tacho on motor	92
6.11.3.	Step motor open-loop control. No feedback device	93
6.11.4.	Step motor open-loop control. Incremental encoder on load	93
6.11.5.	Step motor closed-loop control. Incremental encoder on motor	93
6.11.6.	Brushless motor with sine/cosine encoder on motor	93
6.12.	Motor speed units	93
6.12.1.	Brushless / DC brushed motor with quadrature encoder on motor	93
6.12.2.	DC brushed motor with quadrature encoder on load and tacho on motor	94
6.12.3.	DC brushed motor with tacho on motor	94
6.12.4.	Step motor open-loop control. No feedback device or incremental encoder on load	95
6.12.5.	Step motor closed-loop control. Incremental encoder on motor	95
6.12.6.	Brushless motor with sine/cosine encoder on motor	96

7. Memory Map 97

This page is empty.

1. Safety information

Read carefully the information presented in this chapter before carrying out the drive installation and setup! It is imperative to implement the safety instructions listed hereunder.

This information is intended to protect you, the drive and the accompanying equipment during the product operation. Incorrect handling of the drive can lead to personal injury or material damage.

Only qualified personnel may install, set up, operate and maintain the drive. A “qualified person” has the knowledge and authorization to perform tasks such as transporting, assembling, installing, commissioning and operating drives.

The following safety symbols are used in this manual:



WARNING!

SIGNALS A DANGER THAT MIGHT CAUSE BODILY INJURY TO THE OPERATOR. MAY INCLUDE INSTRUCTIONS TO PREVENT THIS SITUATION



CAUTION!

SIGNALS A DANGER FOR THE DRIVE, WHICH MIGHT DAMAGE THE PRODUCT OR OTHER EQUIPMENT. MAY INCLUDE INSTRUCTIONS TO AVOID THIS SITUATION

1.1. Warnings



WARNING!

TO AVOID ELECTRIC ARCING AND HAZARDS, NEVER PLUG / UNPLUG CABLES INTO THE PRO-Ax0V80A-SACAT WHILE THE POWER SUPPLIES ARE ON!



WARNING!

THE DRIVE MAY HAVE HOT SURFACES DURING OPERATION.



WARNING!

DURING DRIVE OPERATION, THE CONTROLLED MOTOR WILL MOVE. KEEP AWAY FROM ALL MOVING PARTS TO AVOID INJURY

1.2.

1.3. Cautions



CAUTION! THE POWER SUPPLIES CONNECTED TO THE DRIVE MUST COMPLY WITH THE PARAMETERS SPECIFIED IN THIS DOCUMENT



CAUTION! TROUBLESHOOTING AND SERVICING ARE PERMITTED ONLY FOR PERSONNEL AUTHORISED BY ELECTROCRRAFT



CAUTION! ELECROMAGNETIC COMPATIBILITY (EMC) COMPLIANCE REQUIRES ADDITION OF SPECIFIC COUNTERMEASURES OUTSIDE THE DRIVE, AS SHOWN IN THIS DOCUMENT IN THE CONNECTION DIAGRAMS

2. Product Overview

2.1. Introduction

The **PRO-Ax0V80A-SA-CAT** is part of a family of fully digital programmable servo drives, based on the latest DSP technology and they offer unprecedented drive performance combined with an embedded motion controller.

Suitable for control of brushless DC, brushless AC (vector control), DC brushed motors and step motors, the PRO-Ax0V80A-SA-CAT drives accept as position feedback incremental encoders (quadrature or sine/cosine).

All drives perform position, speed or torque control and work in single, multi-axis or stand-alone configurations. Thanks to the embedded motion controller, the PRO-Ax0V80A-SA-CAT drives combine controller, drive and PLC functionality in a single compact unit and are capable to execute complex motions without requiring intervention of an external motion controller. Using the high-level ElectroCraft's Motion Programming Language (**MPL**) the following operations can be executed directly at drive level:

- Setting various motion modes (profiles, PVT, PT, electronic gearing or camming, etc.)
- Changing the motion modes and/or the motion parameters
- Executing homing sequences
- Controlling the program flow through:
 - Conditional jumps and calls of MPL functions
 - MPL interrupts generated on pre-defined or programmable conditions (protections triggered, transitions on limit switch or capture inputs, etc.)
 - Waits for programmed events to occur
- Handling of digital I/O and analog input signals
- Executing arithmetic and logic operations
- Performing data transfers between axes¹
- Controlling motion of an axis from another one via motion commands sent between axes¹

By implementing motion sequences directly at drive level you can really distribute the intelligence between the master and the drives in complex multi-axis applications, reducing both the development time and the overall communication requirements. For example, instead of trying to command each movement of an axis, you can program the drives using MPL to execute complex motion tasks and inform the master when these tasks are done. Thus, for each axis control the master job may be reduced at: calling MPL functions stored in the drive EEPROM and waiting for a message, which confirms the MPL functions execution completion.

For PRO-Ax0V80 commissioning **PRO Config** or **MotionPRO Developer** PC applications may be used.

PRO Config is a subset of MotionPRO Developer, including only the drive setup part. The output of PRO Config is a set of setup data that can be downloaded into the drive EEPROM or saved on a PC file. At power-on, the drive is initialized with the setup data read from its EEPROM. With PRO Config it is also

¹ In preparation. Please contact ElectroCraft for availability

possible to retrieve the complete setup information from a drive previously programmed. PRO Config shall be used for drive setup in all cases where the motion commands are sent exclusively from a master. Hence neither the PRO-Ax0V80 MPL programming capability nor the drive camming mode, are used.

PRO Config can be downloaded free of charge from ElectroCraft web page.

MotionPRO Developer platform includes PRO Config for the drive setup, and a **Motion Wizard** for the motion programming. The Motion Wizard provides a simple, graphical way of creating motion programs and automatically generates all the MPL instructions. *With MotionPRO Developer you can fully benefit from a key advantage of ElectroCraft drives – their capability to execute complex motions, thanks to their built-in motion controllers.* MotionPRO Developer, shall be used to program motion sequences in MPL. MotionPRO Developer can also be used with EtherCAT® protocol, if the user wants to call MPL functions stored in the drive EEPROM or to use the camming mode. With camming mode, MotionPRO Developer offers the possibility to quickly download and test a cam profile and also to create a **.sw** file (see par. 5.2.4) with the cam data. The **.sw** file can be afterwards stored in a master and downloaded to the drive, wherever needed. **A demo version of MotionPRO Developer (with PRO Config part fully functional) can be downloaded free of charge from ElectroCraft web page.**

2.2. Key Features

- Fully digital servo drive suitable for the control of rotary or linear brushless, DC brush, and step motors with EtherCAT® interface and built-in motion controller with high-level MPL motion language
- Very compact design
- Sinusoidal (FOC) or trapezoidal (Hall-based) control of brushless motors
- Open or closed-loop control of 2 and 3-phase steppers
- Drive enable function. Two enable inputs are provided, which when left not connected will disable the motor outputs. This provides a dual redundant hardware protection that cannot be overdriven by the software or other hardware components.
- Various modes of operation, including: cyclic synchronous torque, velocity or position control; position or velocity profiles, external analog reference or sent via a communication bus
- ElectroCraft's Motion Programming Language (MPL) instruction set for the definition and execution of motion sequences
- Standalone operation with stored motion sequences
- Communication:
 - RS-232 serial up to 115kbits/s
 - Dual 100Mbps RJ45 EtherCAT® interfaces, for use in daisy-chaining topologies
- Digital and analog I/Os:
 - 4 digital inputs: 5-36 V, programmable polarity: sourcing/NPN or sinking/PNP: Enable, 2 Limit switches and 2 general-purpose
 - 4 digital outputs: 5-36 V, 0.5 A, sinking/NPN open-collector (Ready, Error and 2 general-purpose)
 - 2 analog inputs: 12 bit, +/-10V and 0-5V: Reference and Feedback or general purpose
- Primary feedback devices supported:
 - Incremental quadrature encoder (differential)
 - Analog sine/cosine encoder (differential 1Vpp)
 - Digital Hall sensors
- Secondary feedback devices supported:

-
- Incremental quadrature encoder (differential)
 - Pulse & Direction interface (RS-422 differential) for external (master) digital reference
 - Various motion programming modes:
 - Position profiles with trapezoidal or S-curve speed shape
 - Position, Velocity, Time (PVT) 3rd order interpolation
 - Position, Time (PT) 1st order interpolation
 - Electronic gearing and camming
 - 35 Homing modes
 - 16K × 16 internal SRAM memory for data acquisition
 - 16K × 16 E²ROM to store MPL programs and data
 - PWM switching frequency up to 125kHz
 - Motor supply: 9-80V
 - Logic supply: 9-36V. Separate supply is optional
 - Output current: PRO-A10V80A-SA-CAT: 10A¹ continuous; 20A peak
PRO-A20V80A-SA-CAT: 20A² continuous; 40A peak
 - Operating ambient temperature: 0-40°C (over 40°C with derating)
 - Protections:
 - Short-circuit between motor phases
 - Short-circuit from motor phases to ground
 - Over-voltage
 - Under-voltage
 - Over-current
 - Over-temperature
 - Communication error
 - Control error

¹ 10A cont. with DC, step and BLDC motors (trapezoidal), 10A amplitude (7.1A_{RMS}) for PMSM (sinusoidal)

² 20A cont. with DC, step and BLDC motors (trapezoidal), 20A amplitude (14.2A_{RMS}) for PMSM (sinusoidal)

2.3. Supported Motor-Sensor Configurations

PRO-Ax0V80A-SA-CAT supports the following configurations:

1. Position, speed or torque control of a **brushless AC rotary motor** with an **incremental quadrature encoder** on its shaft. The brushless motor is vector controlled like a permanent magnet synchronous motor. It works with **sinusoidal** voltages and currents. Scaling factors take into account the transmission ratio between motor and load (rotary or linear). Therefore, the motion commands (for position, speed and acceleration) expressed in SI units (or derivatives) always refer to the load¹.

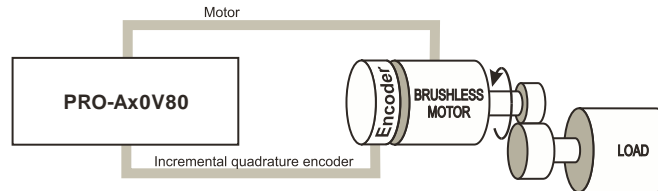


Figure 2.1. Brushless AC rotary motor. Position / speed / torque control. Quadrature encoder on motor.

2. Position, speed or torque control of a **brushless AC rotary motor** with an **incremental sine/cosine encoder** on its shaft. The brushless motor is vector controlled like a permanent magnet synchronous motor. It works with **sinusoidal** voltages and currents. Scaling factors take into account the transmission ratio between motor and load (rotary or linear). Therefore, the motion commands (for position, speed and acceleration) expressed in SI units (or derivatives) always refer to the load¹.

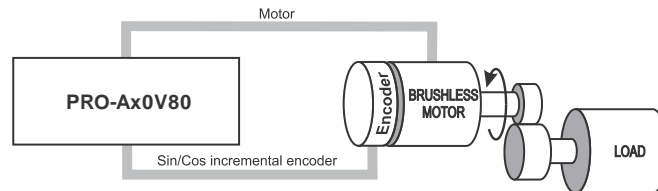


Figure 2.2. Brushless AC rotary motor. Position/speed/torque control. Sine/cosine incremental encoder on motor

¹ Motion commands can be referred to the motor by setting in PRO Config a rotary to rotary transmission with ratio 1:1

- Position, speed or torque control of a **brushless DC rotary motor** with **digital Hall sensors** and an **incremental quadrature encoder** on its shaft. The brushless motor is controlled using Hall sensors for commutation. It works with rectangular currents and **trapezoidal BEMF** voltages. Scaling factors take into account the transmission ratio between motor and load (rotary or linear). Therefore, the motion commands (for position, speed and acceleration) expressed in SI units (or derivatives) always refer to the load¹.

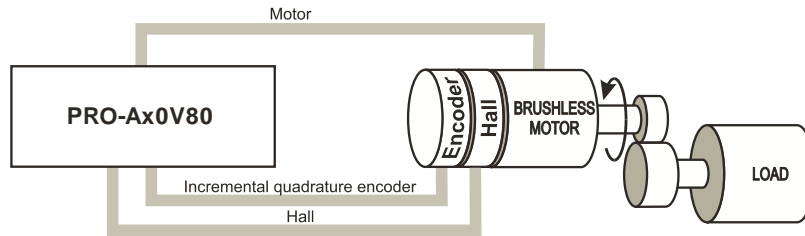


Figure 2.3. Brushless DC rotary motor. Position / speed / torque control. Hall sensors and quadrature encoder on motor.

- Dual loop position and speed control of a **brushless DC rotary motor** with **digital Hall sensors**, an **incremental quadrature or Sin/Cos encoder** on one feedback and another incremental encoder on the second feedback. The speed loop is controlled with the encoder on the motor and the position loop is controlled with the encoder on the load.
Remark: the Sin/Cos encoder is present only on the Feedback #1 interface.

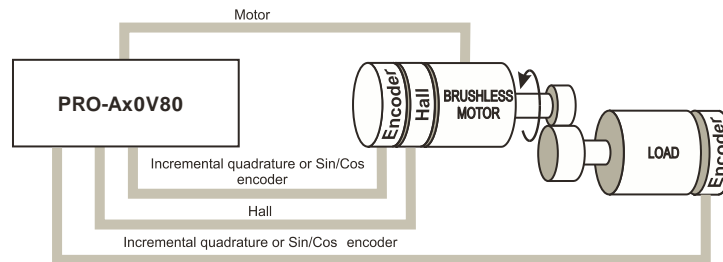


Figure 2.4. Brushless DC rotary motor. Position and speed control. Hall sensors and encoders on motor and load.

¹ Motion commands can be referred to the motor by setting in PRO Config a rotary to rotary transmission with ratio 1:1

5. Position, speed or torque control of a **brushless AC linear motor** with an **incremental sine/cosine encoder**. The brushless motor is vector controlled like a permanent magnet synchronous motor. It works with **sinusoidal** voltages and currents. Scaling factors take into account the transmission ratio between motor and load (rotary or linear). Therefore, the motion commands (for position, speed and acceleration) expressed in SI units (or derivatives) always refer to the load¹.

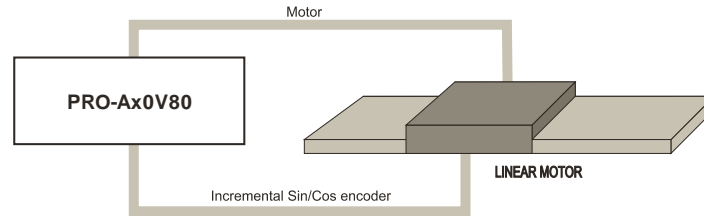


Figure 2.5. Brushless AC linear motor. Position/speed/torque control.
Sine/cosine incremental encoder on motor

6. Position, speed or torque control of a **brushless AC linear motor** with an **incremental quadrature linear encoder** on the track. The brushless motor is vector controlled like a permanent magnet synchronous motor. It works with **sinusoidal** voltages and currents. Scaling factors take into account the transmission ratio between motor and load (linear or rotary). Therefore, the motion commands (for position, speed and acceleration) expressed in SI units (or derivatives) always refer to the load¹.

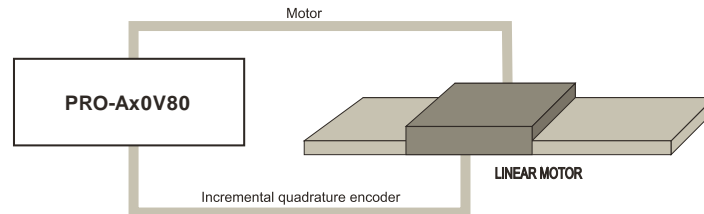


Figure 2.6. Brushless AC rotary motor with incremental quadrature linear encoder.
Position / speed / torque control.

¹ Motion commands can be referred to the motor by setting in PRO Config a rotary to rotary transmission with ratio 1:1

7. Dual loop position and speed control of a **brushless DC linear motor** with **digital Hall sensors**, an **incremental quadrature or Sin/Cos encoder** on one feedback and another incremental encoder on the second feedback. The speed loop is controlled with the encoder on the motor and the position loop is controlled with the encoder on the load.
- Remark: the Sin/Cos encoder is present only on the Feedback #1 interface.*

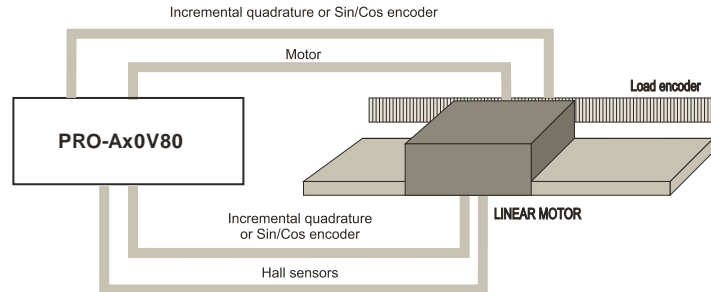


Figure 2.7. Brushless DC rotary motor. Position and speed control. Hall sensors and encoders on motor and load.

8. Position, speed or torque control of a **DC brushed rotary motor** with an **incremental quadrature encoder** on its shaft. Scaling factors take into account the transmission ratio between motor and load (rotary or linear). Therefore, the motion commands (for position, speed and acceleration) expressed in SI units (or derivatives) always refer to the load¹.

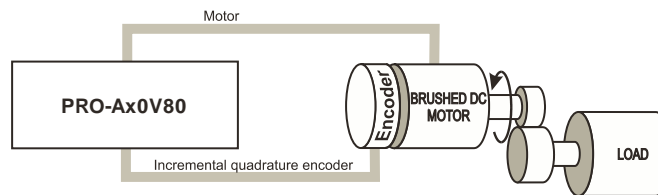


Figure 2.8. DC brushed rotary motor. Position / speed / torque control. Quadrature encoder on motor.

¹ Motion commands can be referred to the motor by setting in PRO Config a rotary to rotary transmission with ratio 1:1

9. Dual loop, position and speed control of a **brushed DC rotary motor** with an **incremental quadrature** on the motor and an incremental encoder on the load. The speed loop is controlled with the encoder on the motor and the position loop is controlled with the encoder on the load.

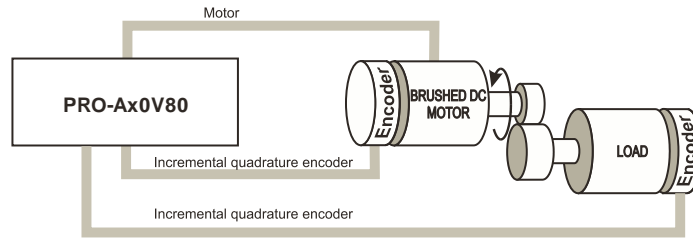


Figure 2.9. DC brushed rotary motor. Position and speed control. Quadrature encoders on motor and load.

10. Load position control using an **incremental quadrature encoder** on load, combined with speed control of a **DC brushed rotary motor** having a **tachometer** on its shaft. The motion commands (for position, speed and acceleration) expressed in SI units (or derivatives) always refer to the load¹.

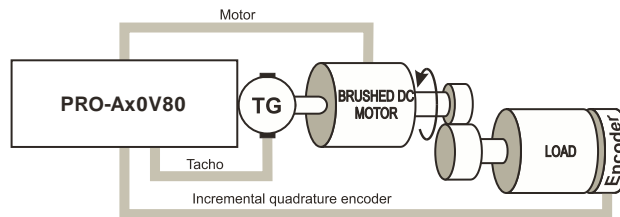


Figure 2.10. DC brushed rotary motor. Position / speed / torque control. Quadrature encoder on load plus tachometer on motor.

¹ Motion commands can be referred to the motor by setting in PRO Config a rotary to rotary transmission with ratio 1:1

-
11. Speed or torque control of a **DC brushed rotary motor** with a **tachometer** on its shaft. Scaling factors take into account the transmission ratio between motor and load (rotary or linear). Therefore, the motion commands (for position, speed and acceleration) expressed in SI units (or derivatives) always refer to the load¹.

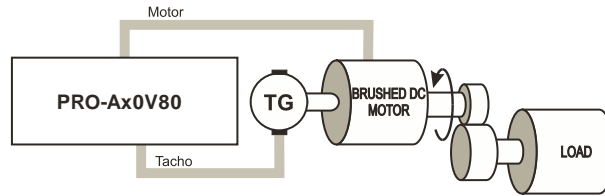


Figure 2.11. DC brushed rotary motor. Speed/torque control.
Tachometer on motor.

12. Open-loop control of a 2 or 3-phase **step motor** in position or speed. Scaling factors take into account the transmission ratio between motor and load (rotary or linear). Therefore, the motion commands (for position, speed and acceleration) expressed in SI units (or derivatives) always refer to the load¹.

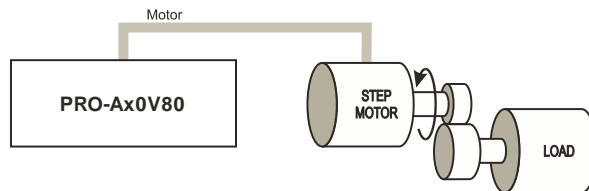


Figure 2.12. No position or speed feedback.
Open-loop control: motor position or speed.

¹ Motion commands can be referred to the motor by setting in PRO Config a rotary to rotary transmission with ratio 1:1

-
13. Closed-loop control of **load position using an encoder on load**, combined with open-loop control of a **2 or 3-phase step motor** in speed, with speed reference provided by the position controller. The motion commands in both SI and IU units refer to the load.

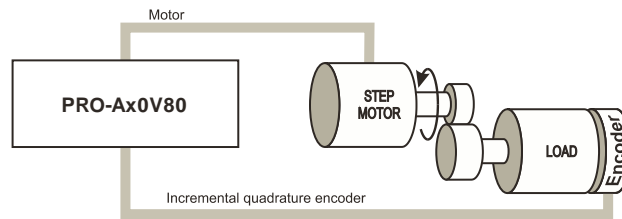


Figure 2.13. Encoder on load.
Closed-loop control: load position, open-loop control: motor speed.

14. Closed-loop control of a **2-phase step motor** in position, speed or torque. Scaling factors take into account the transmission ratio between motor and load (rotary or linear). Therefore, the motion commands expressed in SI units (or derivatives) refer to the load¹, while the same commands, expressed in IU units refer to the motor.

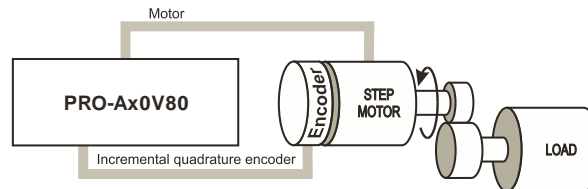


Figure 2.14. Encoder on motor shaft.
Closed-loop control: motor position, speed or torque.

¹ Motion commands can be referred to the motor by setting in PRO Config a rotary to rotary transmission with ratio 1:1

2.4. PRO-Ax0V80A-SA-CAT Drive Dimensions

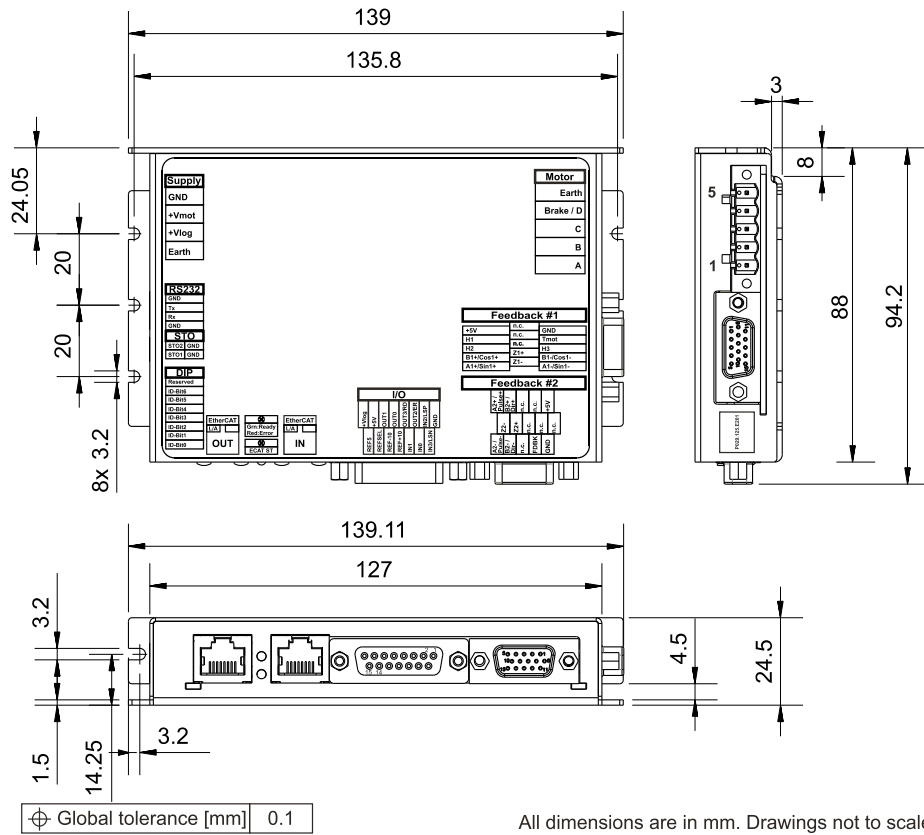


Figure 2.15. PRO-Ax0V80A-SA-CAT drive dimensions

All dimensions are in mm. The drawings are not to scale.

2.5. Identification Labels

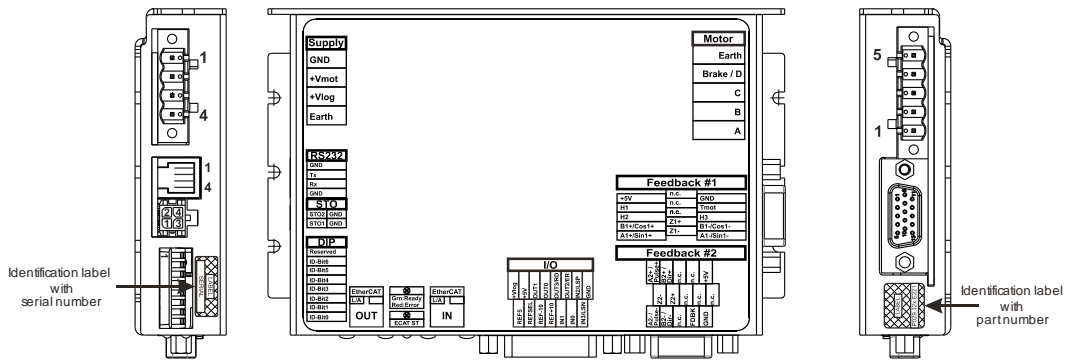


Figure 2.16. PRO-Ax0V80A-SA-CAT Identification Labels

2.6. Electrical Specifications

All parameters measured under the following conditions (unless otherwise specified):

$T_{amb} = 0 \dots 40^{\circ}\text{C}$, $V_{LOG} = 24 V_{DC}$; $V_{MOT} = 80V_{DC}$; Supplies start-up / shutdown sequence: -any-
Load current (sinusoidal amplitude / continuous BLDC, DC, stepper) = 8A

2.6.1. Operating Conditions

		Min.	Typ.	Max.	Units
Ambient temperature		0		+40	$^{\circ}\text{C}$
Ambient humidity	Non-condensing	0		90	%Rh
Altitude / pressure ¹	Altitude (referenced to sea level)	-0.1	$0 \div 2.5$	²	Km
	Ambient Pressure	0^2	$0.75 \div 1$	10.0	atm

2.6.2. Storage Conditions

		Min.	Typ.	Max.	Units
Ambient temperature		-40		+85	$^{\circ}\text{C}$
Ambient humidity	Non-condensing	0		100	%Rh
Ambient Pressure		0		10.0	atm

2.6.3. Mechanical Mounting

Airflow		natural convection, closed box
---------	--	--------------------------------

2.6.4. Environmental Characteristics

		Min.	Typ.	Max.	Units
Size (Length x Width x Height)	Without mating connector	139 x 94.2 x 24.6			mm
		~5.47 x 3.7 x 0.97			inch
Weight	Without mating connectors	240			g
Power dissipation	Idle (no load)		3.6		W
	Operating		11		
Efficiency			98		%
Cleaning agents	Dry cleaning is recommended	Only Water- or Alcohol- based			
Protection degree	According to IEC60529, UL508	IP30			-

¹ PRO-Ax0V80 can be operated in vacuum (no altitude restriction), but at altitudes over 2,500m, current and power rating are reduced due to thermal dissipation efficiency.

2.6.5. Logic Supply Input (+V_{LOG})

		Min.	Typ.	Max.	Units
Supply voltage	Nominal values	9		36	V _{DC}
	Absolute maximum values, drive operating but outside guaranteed parameters	8		40	V _{DC}
	Absolute maximum values, surge (duration ≤ 10ms) †	-1		+45	V
Supply current	No Load on Digital Outputs	+V _{LOG} = 9V		400	mA
		+V _{LOG} = 12V		300	
		+V _{LOG} = 24V		150	
		+V _{LOG} = 40V		90	

2.6.6. Motor Supply Input (+V_{MOT})

		Min.	Typ.	Max.	Units	
Supply voltage	Nominal values	12	80	90	V _{DC}	
	Absolute maximum values, drive operating but outside guaranteed parameters	11		94	V _{DC}	
	Absolute maximum values, surge (duration ≤ 10ms) †	-1		95	V	
Supply current	Idle		1	5	mA	
	Operating	10V80	-20	±10	+20	A
		20V80	-40	±20	+40	
	Absolute maximum value, short-circuit condition (duration ≤ 10ms) †	10V80			22.5	A
		20V80			45	

2.6.7. Enable circuit inputs (ENA1, ENA2)

		Min.	Typ.	Max.	Units
Enable function		Disables motor power when either ENA1 or ENA2 is disconnected from the power source			
Mode compliance		PNP			
Default state	Input floating (wiring disconnected)	Logic Low			
Input voltage	Logic "LOW"	-10		2.2	V
	Logic "HIGH"	6.3		36	
	Absolute maximum, continuous	-10		+39	
Input current	Logic "LOW", pulled to GND		0		mA
	Logic "HIGH", pulled to +Vlog			0.4	
Pulse duration	Ignored low-high-low			tbd	ms
	Ignored high-low-high			tbd	
	Accepted pulse	tbd			

Fault reaction time	From internal fault detection to register DER bit 14 =1 and Output 2 high-to-low			tdb	ms
ESD protection	Human body model	±2			KV

2.6.8. Motor Outputs (A/A+, B/A-, C/B+, BR/B-)

		Min.	Typ.	Max.	Units
Nominal output current, continuous	for DC brushed, steppers and BLDC motors with Hall-based trapezoidal control			10	A
	for PMSM motors with FOC sinusoidal control (sinusoidal amplitude value)			10	
	for PMSM motors with FOC sinusoidal control (sinusoidal effective value)			7.1	
Motor output current, peak	maximum 2.5s	-38.3		+38.3	A
Short-circuit protection threshold	measurement range			±45	A
Short-circuit protection delay		5	10		µS
On-state voltage drop	for nominal output current; including typical mating connector contact resistance		±0.3	±0.5	V
Off-state leakage current			±0.5	±1	mA
Motor inductance (phase-to-phase)	Recommended value, for ripple ±5% of measurement range; +V _{MOT} = 36 V	F _{PWM} = 20 kHz	250		µH
		F _{PWM} = 40 kHz	120		
		F _{PWM} = 80 kHz	60		
		F _{PWM} = 100 kHz	45		
	Absolute minimum value, limited by short-circuit protection; +V _{MOT} = 36 V	F _{PWM} = 20 kHz	75		µH
		F _{PWM} = 40 kHz	25		
		F _{PWM} = 80 kHz	10		
		F _{PWM} = 100 kHz	5		
Motor electrical time-constant (L/R)	Recommended value, for ±5% current measurement error due to ripple	F _{PWM} = 20 kHz	250		µs
		F _{PWM} = 40 kHz	125		
		F _{PWM} = 80 kHz	63		
		F _{PWM} = 100 kHz	50		
Current measurement accuracy	FS = Full Scale		±4	±8	%FS

2.6.9. Digital Inputs (IN0, IN1, IN2/LSP, IN3/LSN)¹

		Min.	Typ.	Max.	Units
Mode compliance		PNP			
Default state	Input floating (wiring disconnected)	Logic LOW			
Input voltage	Logic "LOW"	-10	0	2.2	V
	Logic "HIGH"	6.3		36	
	Floating voltage (not connected)		0		
	Absolute maximum, continuous	-10		+39	
	Absolute maximum, surge (duration ≤ 1S) †	-20		+40	
Input current	Logic "LOW"; Pulled to GND		0		mA
	Logic "HIGH"		1.3	2	

		Min.	Typ.	Max.	Units
Mode compliance		NPN			
Default state	Input floating (wiring disconnected)	Logic HIGH			
Input voltage	Logic "LOW"	-10		2.2	V
	Logic "HIGH"	6.3		36	
	Floating voltage (not connected)		3		
	Absolute maximum, continuous	-10		+36	
	Absolute maximum, surge (duration ≤ 1S) †	-20		+40	
Input current	Logic "LOW"; Pulled to GND	-1.6	0.6	1	mA
	Logic "HIGH"; Pulled to +24V	0	0	0.3	

Input frequency		0		150	KHz
Minimum pulse width		3.3			μS
ESD protection	Human body model	±2			KV

¹ The digital inputs are software selectable as PNP or NPN

2.6.10. Digital Outputs (OUT0, OUT1, OUT2/Error, OUT3/Ready)

		Min.	Typ.	Max.	Units
Mode compliance	All outputs (OUT0, OUT1, OUT2/Error, OUT3/Ready)	NPN 24V			
Default state	Not supplied (+V _{LOG} floating or to GND)	High-Z (floating)			
	Immediately after power-up	OUT0, OUT1	Logic "HIGH"		
		OUT2/Error, OUT3/Ready	Logic "LOW"		
	Normal operation	OUT0, OUT1, OUT2/Error	Logic "HIGH"		
OUT3/Ready		Logic "LOW"			
Output voltage	Logic "LOW"; output current = 0.5A			0.8	V
	Logic "HIGH"; output current = 0, no load	OUT2/Error, OUT3/ Ready	2.9	3	
		OUT0, OUT1	4	4.5	5
	Logic "HIGH", external load to +V _{LOG}		V _{LOG}		
	Absolute maximum, continuous	-0.5		V _{LOG} +0.5	
Absolute maximum, surge (duration ≤ 1S) †	-1		V _{LOG} +1		
Output current	Logic "LOW", sink current, continuous	OUT0		2	A
		OUT1, OUT2, OUT3		0,5	
	Logic "LOW", sink current, pulse ≤ 5 sec	OUT0		4	A
		OUT1, OUT2, OUT3		1	
	Logic "HIGH", source current; external load to GND; V _{OUT} ≥ 2.0V	OUT2/Error, OUT3/Ready		2	mA
		OUT0, OUT1		4	mA
Logic "HIGH", leakage current; external load to +V _{LOG} ; V _{OUT} = V _{LOG} max = 40V		0.1	0.2	mA	
Minimum pulse width		2			μS
ESD protection	Human body model	±15			KV

2.6.11. Digital Hall Inputs (Hall1, Hall2, Hall3)

		Min.	Typ.	Max.	Units
Mode compliance		TTL / CMOS / Open-collector			
Default state	Input floating (wiring disconnected)	Logic HIGH			
Input voltage	Logic "LOW"		0	0.8	V
	Logic "HIGH"	2	5		
	Floating voltage (not connected)		4.4		
	Absolute maximum, surge (duration $\leq 1S$) [†]	-10		+15	
Input current	Logic "LOW"; Pull to GND			1.2	mA
	Logic "HIGH"; Internal 4.7K Ω pull-up to +5	0	0	0	
Minimum pulse width		2			μS
ESD protection	Human body model	± 5			KV

2.6.12. Encoder Inputs (A1+, A1-, B1+, B1-, Z1+, Z1-, A2+, A2-, B2+, B2-, Z2+, Z2-)¹

		Min.	Typ.	Max.	Units
Differential mode compliance		TIA/EIA-422-A			
Input voltage, differential mode	Hysteresis	± 0.06	± 0.1	± 0.2	V
	Differential mode	-14		+14	
	Common-mode range (A+ to GND, etc.)	-11		+14	
Input impedance, differential	A1+, A2+, B1+, B2+, Z1+, Z2+		2.2		K Ω
	A1-, A2-, B1-, B2-, Z1-, Z2-		1.6		
	Differential mode	0		10	MHz
	Differential mode	50			nS
ESD protection	Human body model	± 1			KV

¹ All differential input pins have internal 120 Ω termination resistors connected across

2.6.13. Sin-Cos Encoder Inputs (Sin+, Sin-, Cos+, Cos-)¹

		Min.	Typ.	Max.	Units
Input voltage, differential	Sin+ to Sin-, Cos+ to Cos-		1	1.25	V _{PP}
Input voltage, any pin to GND	Operational range	-1	2.5	4	V
	Absolute maximum values, continuous	-7		+7	
	Absolute maximum, surge (duration ≤ 1S) [†]	-11		+14	
Input impedance	Differential, Sin+ to Sin-, Cos+ to Cos-	4.2	4.7		KΩ
	Common-mode, to GND		2.2		KΩ
Interpolation Resolution	Depending on software settings			11	bits
Frequency	Sin-Cos interpolation	0		450	KHz
	Quadrature, no interpolation	0		10	MHz
ESD protection	Human body model	±1			KV

2.6.14. Analog 0...5V Inputs (REF, FDBK)

		Min.	Typ.	Max.	Units
Input voltage	Operational range	0		5	V
	Absolute maximum values, continuous	-12		+18	
	Absolute maximum, surge (duration ≤ 1S) [†]			±36	
Input impedance	To GND		6.8		KΩ
Resolution			12		bits
Integral linearity				±2	bits
Offset error			±2	±10	bits
Gain error			±1%	±3%	% FS ²
Bandwidth (-3dB)	Depending on software settings	0		1	KHz
ESD protection	Human body model	±5			KV

2.6.15. RS-232

		Min.	Typ.	Max.	Units
Standards compliance		TIA/EIA-232-C			
Bit rate	Depending on software settings	9600		115200	Baud
Short-circuit protection	232TX short to GND	Guaranteed			
ESD protection	Human body model	±2			KV

¹ All differential input pins have internal 120Ω termination resistors connected across

² "FS" stands for "Full Scale"

2.6.16. EtherCAT®

		Min.	Typ.	Max.	Units
Standards compliance		IEEE802.3, IEC61158			
Transmission line specification	According to TIA/EIA-568-5-A	Cat.5e.UTP			
J5, J6 pinout	EtherCAT® supports MDI/MDI-X auto-crossover	TIA/EIA-568-A or TIA/EIA-568-B			
Software protocols compatibility		CoE, CiA402, IEC61800-7-301			
Node addressing	Hardware: by DIP switch	1-127 & 255			-
	By software, via PRO Config	1 ÷ 255			-
MAC addressing	EtherCAT® uses no MAC address	none			-
ESD protection	Human body model	±15			KV

2.6.17. Supply Output (+5V)

		Min.	Typ.	Max.	Units
+5V output voltage	Current sourced = 250mA	4.8	5	5.2	V
+5V output current		200			mA
Short-circuit protection		NOT protected			
Over-voltage protection		NOT protected			
ESD protection	Human body model	±1			KV

† Stresses beyond values listed under “absolute maximum ratings” may cause permanent damage to the device. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

3. Step 1. Hardware Installation

3.1. Mechanical Mounting

The PRO-Ax0V80A-SA-CAT drive is intended to be mounted vertically or horizontally on a metallic support using the provided mounting holes and the recommended mating connectors, as specified in chapter 3.2. The metallic support must act as a cooling heat sink.

3.2. Mating Connectors

Connector	Description	Manufacturer	Part Number
J1	Supply input, 4x5.08 female counter part for cable	Camden	CTBA9208/4FL
J2	Motor power, 5x5.08 female counter part for cable	Camden	CTBA9208/5FL
J3,J4	Feedback #1 +Hall & Feedback #2		generic 15-pin High Density D-Sub male
J9	RS232		generic RJ10-4/4 phone plug
J5	I/O ; Analog		generic 15-pin D-Sub male, DB15
J8	MICROFIT RECEPTACLE HOUSING, 2x2 WAY	MOLEX	43045-0400
J8	CRIMP PIN, MICROFIT, 5A	MOLEX	43030-0007
J6,J7	Standard 8P8C modular jack (RJ-45) male	-	-

3.3. Connectors and Connection Diagrams

3.3.1. Connector Layout

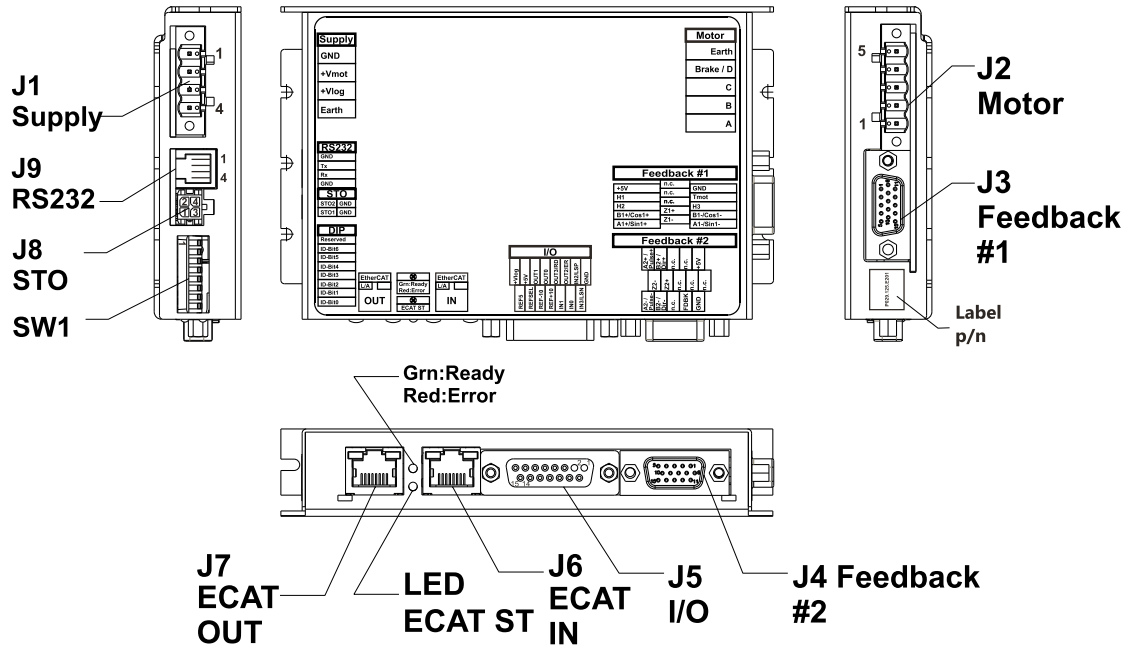


Figure 3.1. PRO-Ax0V80A-SA-CAT drive connectors

3.3.2. J1 Power supply input connector pinout

Connector description			
Pin	Name	Type	Description
1	GND	-	Negative return (ground) of the power supply
2	+V _{MOT}	I	Positive terminal of the motor supply: 9 to 80V _{DC} .
3	+V _{LOG}	I	Positive terminal of the logic supply input: 9 to 36V _{DC}
4	Earth	-	Earth connection

3.3.3. J2 Motor output

Connector description			
Pin	Name	Type	Description
1	A/A+	O	Phase A for 3-ph motors, A+ for 2-ph steppers, Motor+ for DC brush motors
2	B/A-	O	Phase B for 3-ph motors, A- for 2-ph steppers, Motor- for DC brush motors
3	C/B+	O	Phase C for 3-ph motors, B+ for 2-ph steppers
4	BR/B-	O	Brake resistor / Phase B- for step motors
5	Earth	-	Earth connection

3.3.4. J3 Primary feedback connector pinout

Connector description			
Pin	Name	Type	Description
1	+5V _{OUT}	O	5V output supply for I/O usage
2	Hall 1	I	Digital input Hall 1 sensor
3	Hall 2	I	Digital input Hall 2 sensor
4	B1+/Cos+	I	Incr. encoder1 B+ diff. input, or analog encoder Cos+ diff. input
5	A1+/Sin+	I	Incr. encoder1 A+ diff. input, or analog encoder Sin+ diff. input
6	n.c.		
7	n.c.		
8	n.c.		
9	Z1+	I	Incr. encoder1 Z+ diff. input
10	Z1-	I	Incr. encoder1 Z- diff. input
11	GND	-	Return ground for sensors supply
12	Temp Mot	I	Analog input, 12-bit, 0-5V. Used to read an analog temperature value
13	Hall 3	I	Digital input Hall 3 sensor
14	B1-/Cos-	I	Incr. encoder1 B- diff. input, or analog encoder Cos- diff. input
15	A1-/Sin-	I	Incr. encoder1 A- diff. input, or analog encoder Sin- diff. input

3.3.5. J4 Secondary feedback connector pinout

Connector description			
Pin	Name	Type	Description
1	+5V _{OUT}	O	5V output supply for I/O usage
2	n.c.		
3	n.c.		
4	B2+/Dir+	I	Incr. encoder2 B+ diff. input, or Dir+
5	A2+/Pulse+	I	Incr. encoder2 A+ diff. input, or analog encoder Sin+ diff. input
6	n.c.		
7	n.c.		
8	n.c.		
9	Z2+/PWM+	I	Incr. encoder2 Z+ diff. input
10	Z2-/PWM-	I	Incr. encoder2 Z- diff. input
11	GND	-	Return ground for sensors supply
12	FDBK	I	Analog input, 12-bit, 0-5V. Used to read an analog position or speed feedback (as tach), or used as general purpose analog input
13	n.c.		
14	B2-/Dir-	I	Incr. encoder2 B- diff. input, or Dir-
15	A2-/Pulse-	I	Incr. encoder2 A- diff. input, or Pulse-

3.3.6. J5 Digital, analog I/O and logic supply connector pinout

Connector description			
Pin	Name	Type	Description
1	GND	-	Return ground for I/O pins
2	IN2/LSP	I	5-36V digital PNP/NPN input. Positive limit switch input
3	OUT2/Error	O	5-36V 0.5A, drive Error output, active low, NPN open-collector/TTL pull-up. Also drives the red LED
4	OUT3/Ready	O	5-36V 0.5A, drive Ready output, active low, NPN open-collector/TTL pull-up. Also drives the green LED.
5	OUT0	O	5-36V 0.5A, general-purpose digital output, NPN open-collector/TTL pull-up
6	OUT1	O	5-36V 0.5A, general-purpose digital output, NPN open-collector/TTL pull-up
7	+5V _{OUT}	O	5V output supply for I/O usage
8	+V _{LOG}	I	Positive terminal of the logic supply input: 9 to 36V _{DC}
9	IN3/LSN	I	5-36V digital PNP/NPN input. Negative limit switch input
10	IN0	I	5-36V general-purpose digital PNP/NPN input
11	IN1	I	5-36V general-purpose digital PNP/NPN input
12	REF+10	I	Analog input, 12-bit, +/-10V input. Used to read an analog position, speed or torque reference.
13	REF-10	I	Connect REF-10 to GND when REF5 is used.
14	REFSEL	I	Analog selection, floating for +/-10V input, GND connected when REF5+ is used.
15	REF5	I	Analog input, 12-bit, 0-5V input. Used to read an analog position, speed or torque reference.

3.3.7. J6, J7 EtherCAT® connectors

J6 and J7 are standard RJ45 Ethernet connectors, compatible with IEEE802.3 100BASE-T (100Mbit/s).
Accept STP/UTP wiring.

J6 – EtherCAT IN connector

J7 – EtherCAT OUT connector

3.3.8. J8 Enable circuit connector pinout

Connector description			
Pin	Name	Type	Description
J8 1	ENA1	I	Enable circuit input1; connect ENA1&ENA2 to +24V to activate motor operation
2	ENA2	I	Enable circuit input2; connect ENA1&ENA2 to +24V to activate motor operation
3	GND	-	Return ground
4	GND	-	Return ground

Remark: *ENA pins set HIGH usually means they should be connected to +Vlog.*

3.3.9. J9 RS232 connector pinout

Connector description			
Pin	Name	Type	Description
J9 1	GND	-	Return ground for RS-232 pins
2	232TX	O	RS-232 Data Transmission
3	232RX	I	RS-232 Data Reception
4	GND	-	Return ground for RS-232 pins

3.3.10. AxisID DIP switch settings

Connector description			
Pin	Name	Type	Description
SW1 1	Reserved	-	Leave switch off (UP position)
2	ID-Bit6	-	
3	ID-Bit5	-	
4	ID-Bit4	-	
5	ID-Bit3	-	Hardware AxisID selection switches
6	ID-Bit2	-	They represent the first 7 bits of an 8 bit Axis ID number.
7	ID-Bit1	-	
8	ID-Bit0	-	

3.3.11. 24V Digital I/O Connection

3.3.11.1 PNP inputs

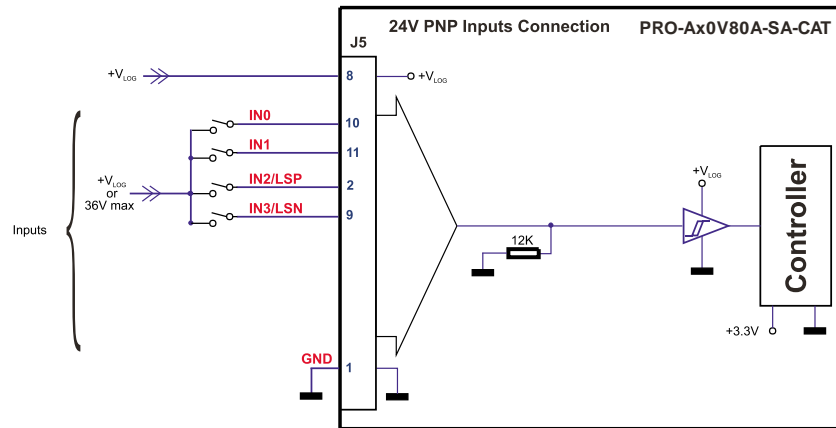


Figure 3.2. 24V Digital PNP Inputs connection

Remarks:

1. The inputs are selectable as PNP/ NPN by software.
2. The inputs are compatible with PNP type outputs (input must receive a positive voltage value (5-36V) to change its default state)

3.3.11.2 NPN inputs

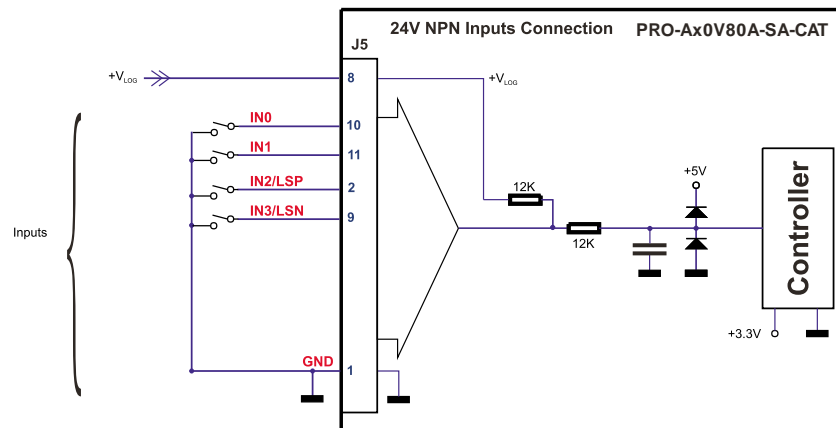


Figure 3.3. 24V Digital NPN Inputs connection

Remarks:

1. The inputs are selectable as PNP/ NPN by software.
2. The inputs are compatible with NPN type outputs (input must be pulled to GND to change its default state)

3.3.11.3 NPN outputs

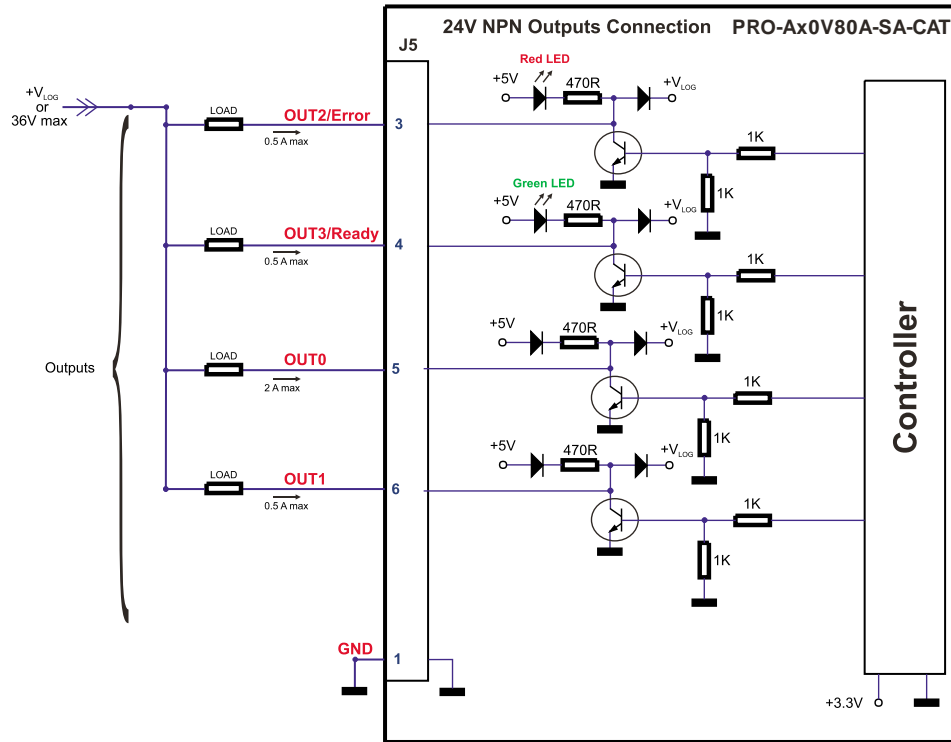


Figure 3.4. 24V Digital NPN Outputs connection

Remarks:

1. The outputs are compatible with NPN type inputs (load is tied to common $+V_{LOG}$, output pulls to GND when active and is floating when inactive)

3.3.12. 5V Digital I/O Connection

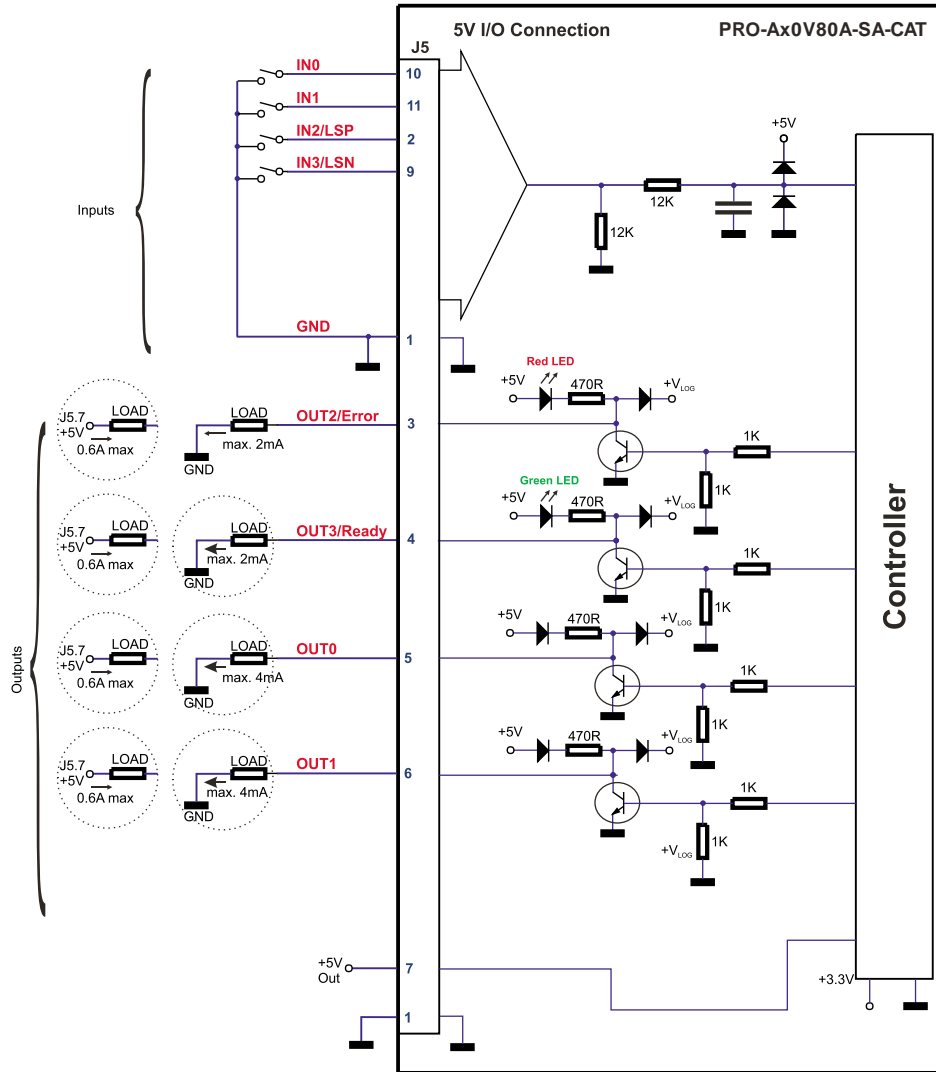


Figure 3.5. 5V Digital I/O connection

Remarks:

1. The inputs are selectable as PNP/ NPN by software. For this connection they are selected as PNP.
2. The inputs are compatible with TTL(5V), LVTTTL(3.3V), CMOS (3.3V-24V) outputs
3. The outputs are compatible with TTL (5V) and CMOS (5V) inputs
4. The output loads can be individually and independently connected to +5V or to GND.

3.3.13. Analog Inputs Connection

3.3.13.1 0-5V Input Range

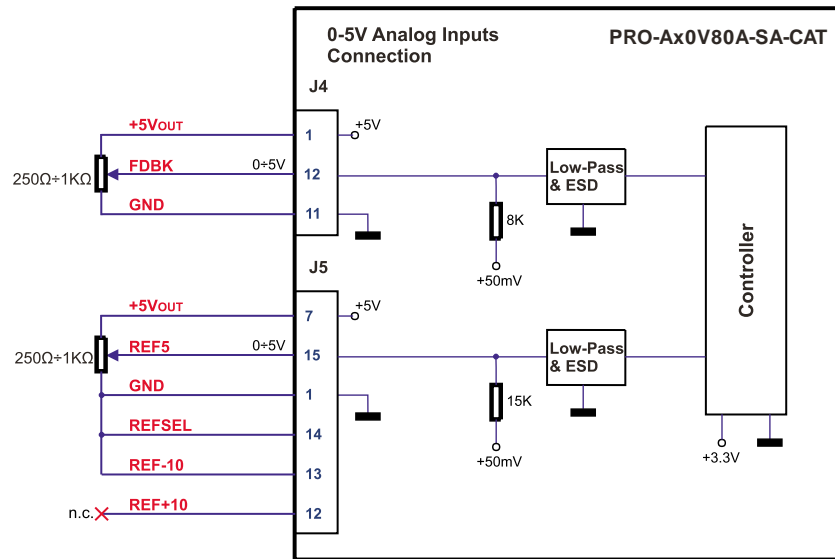


Figure 3.6. 0-5V Analog inputs connection

3.3.13.2 +/-10V Input Range

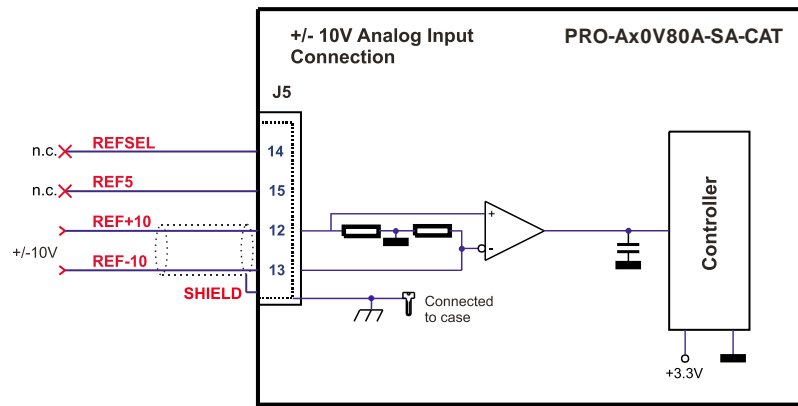


Figure 3.7. +/-10V Analog inputs connection

3.3.13.3 Recommendations for Analog Signals Wiring

- a) If the analog signal source is single-ended, use a 2-wire shielded cable as follows: 1st wire connects the live signal to the drive positive input (+); 2nd wire connects the signal ground to the drive negative input(-).
- b) If the analog signal source is differential and the signal source ground is isolated from the drive GND, use a 3-wire shielded cable as follows: 1st wire connects the signal plus to the drive positive input (+); 2nd wire connects the signal minus to the drive negative input (-) and 3rd wire connects the source ground to the drive GND
- c) If the analog signal source is differential and the signal source ground is common with the drive GND, use a 2-wire shielded cable as follows: 1st wire connects the signal plus to the drive positive input (+); 2nd wire connects the signal minus to the drive negative input (-)
- d) For all of the above cases, connect the cable shield to the drive GND and leave the other shield end unconnected to the signal source. To further increase the noise protection, use a double shielded cable with inner shield connected to drive GND and outer shield connected to the motor chassis (earth).

3.3.14. Motor connections

3.3.14.1 Brushless Motor connection

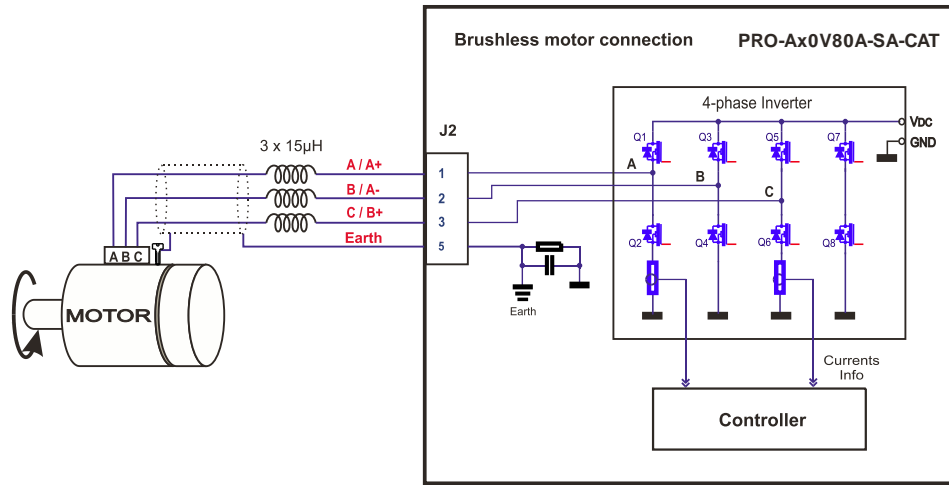


Figure 3.8. Brushless motor connection

Remark: To hardware enable the motor outputs, set ENA1 and ENA2 pins HIGH. This usually means they should be connected to +Vlog.

3.3.14.2 2-phase Step Motor connection

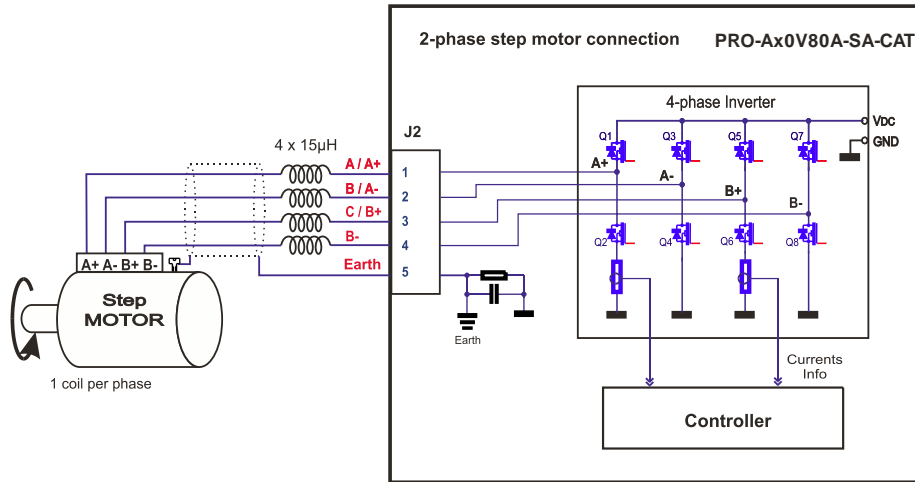


Figure 3.9. 2-phase step motor connection, one coil per phase

Remark: To hardware enable the motor outputs, set ENA1 and ENA2 pins HIGH. This usually means they should be connected to +Vlog.

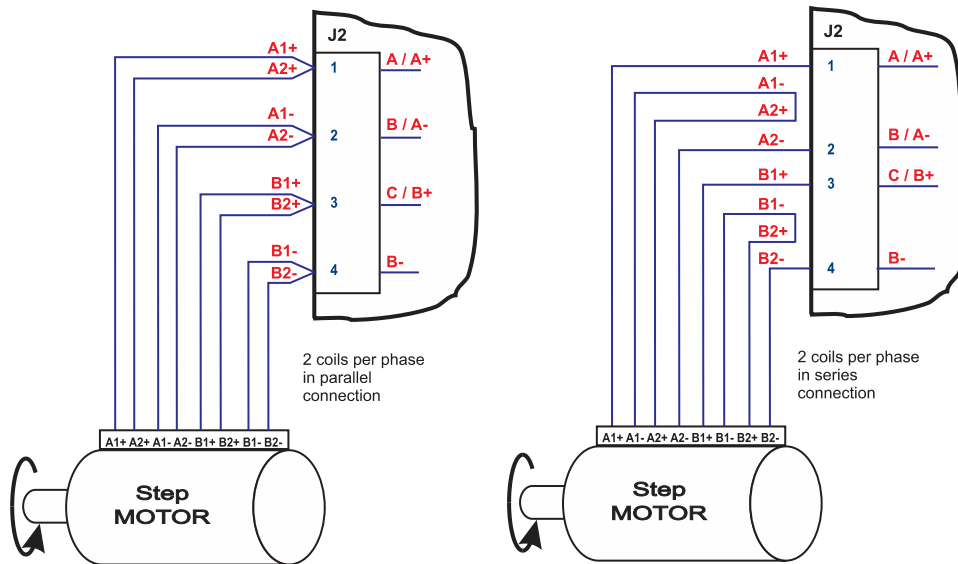


Figure 3.10. 2-phase step motor connection, two coils per phase

3.3.14.3 3-Phase Step Motor connection

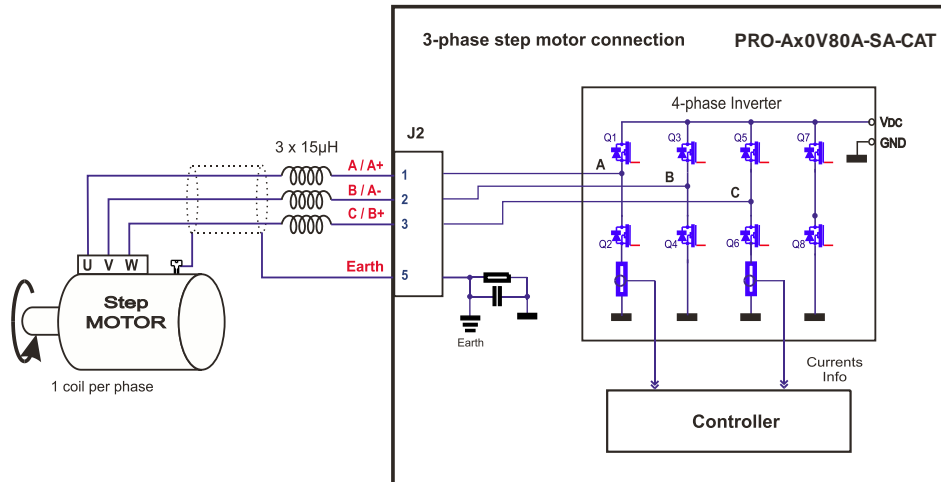


Figure 3.11. 3-phase step motor connection

Remark: To hardware enable the motor outputs, set ENA1 and ENA2 pins HIGH. This usually means they should be connected to +Vlog.

3.3.14.4 DC Brushed Motor connection

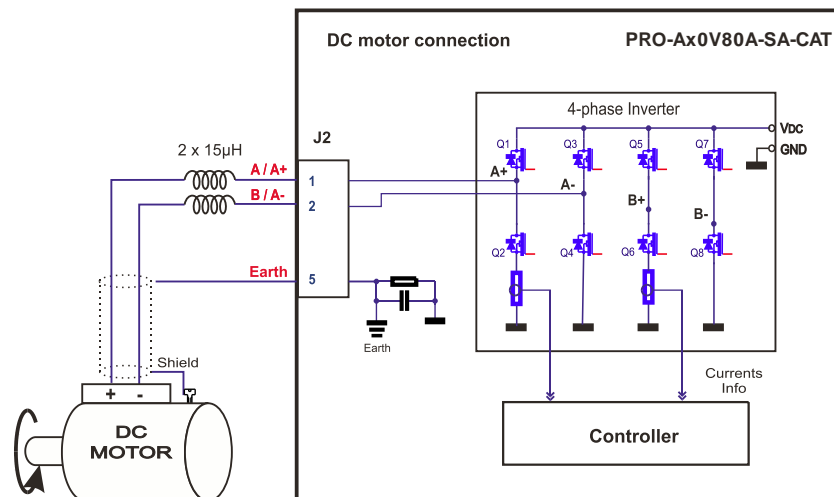


Figure 3.12. DC Motor connection

Remark: To hardware enable the motor outputs, set ENA1 and ENA2 pins HIGH. This usually means they should be connected to +Vlog.

3.3.14.5 Recommendations for motor wiring

- a) Avoid running the motor wires in parallel with other wires for a distance longer than 2 meters. If this situation cannot be avoided, use a shielded cable for the motor wires. Connect the cable shield to the PRO-Ax0V80 GND pin. Leave the other end disconnected.
- b) In order to meet requirements of EMC radiated emissions it is necessary to add series inductors between drive outputs and motor cable. Each inductor shall be 15 μ H minimum, rated for the motor peak current and preferably magnetically shielded (toroidal, for example).
- c) A good shielding can be obtained if the motor wires are running inside a metallic cable guide.

3.3.15. Feedback connections

3.3.15.1 Differential Incremental Encoder1 Connection

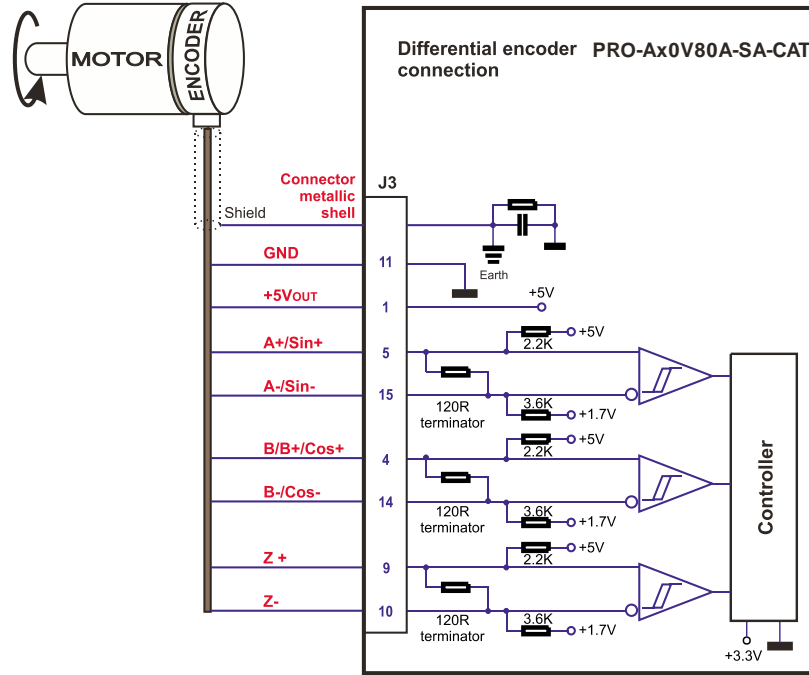


Figure 3.13. Differential incremental encoder connection

Remark: 120Ω termination resistors are present in the drive.

3.3.15.2 Digital Hall Connection

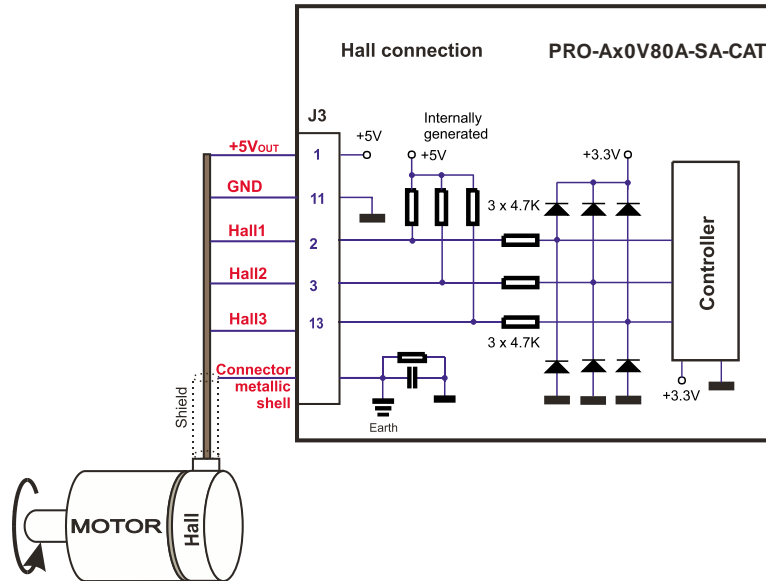


Figure 3.14. Digital Hall connection

3.3.15.3 Pulse and direction connection

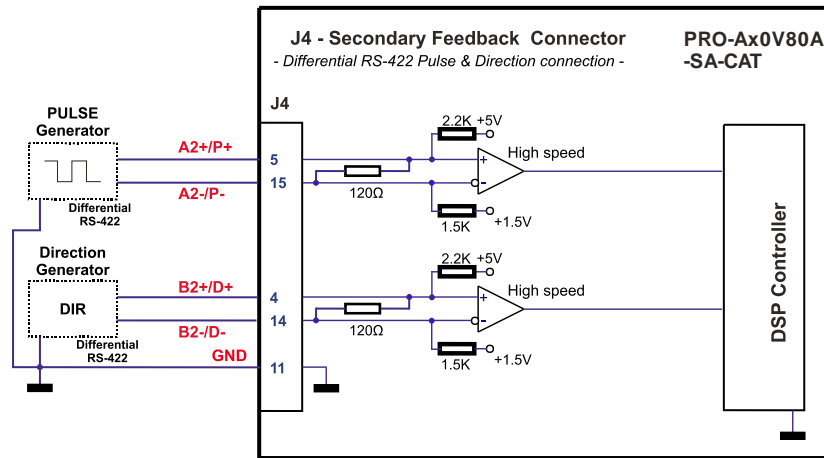


Figure 3.15. J4 – Differential (RS-422) Pulse & Direction connection

Remark: Termination resistors (120Ω) are present in the drive.

3.3.15.4 Differential Incremental Encoder2 Connection

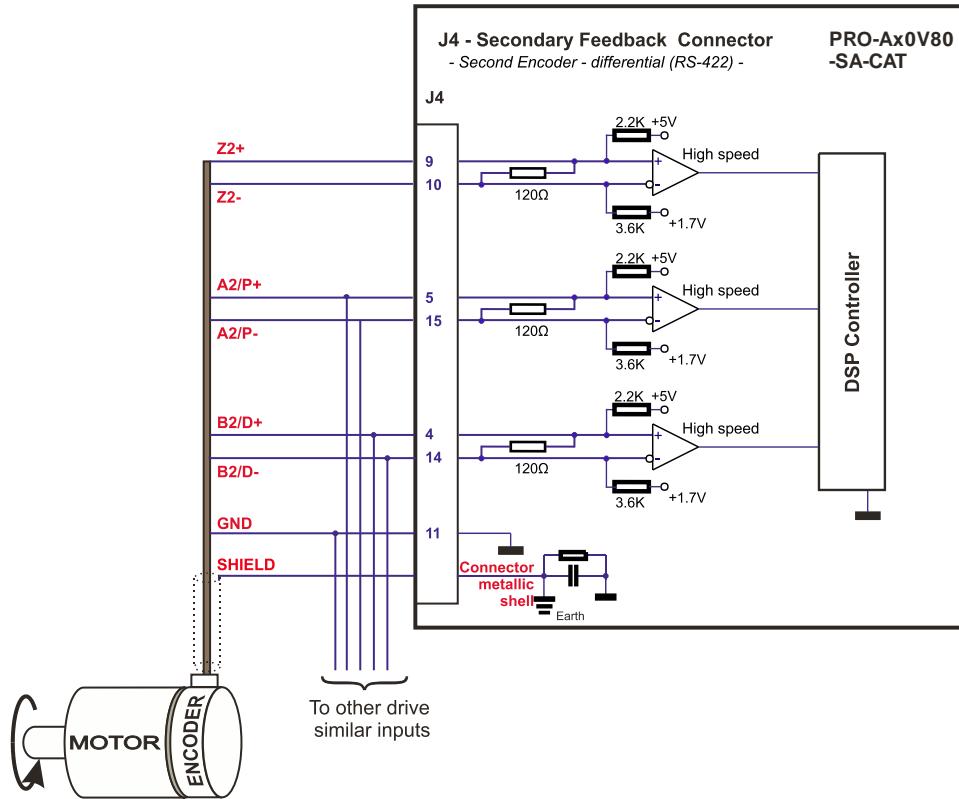


Figure 3.16. J9 – Second encoder – differential (RS-422) connection

Remark: Termination resistors (120Ω) are present in the drive.

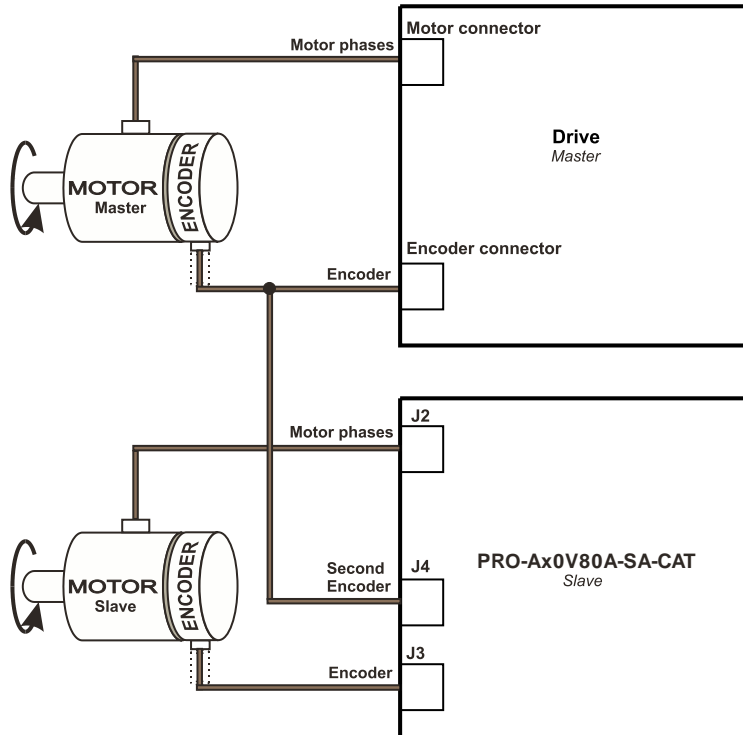


Figure 3.17. J4 – Master – Slave connection using second encoder input

3.3.15.5 Sine-Cosine Analog Encoder Connection

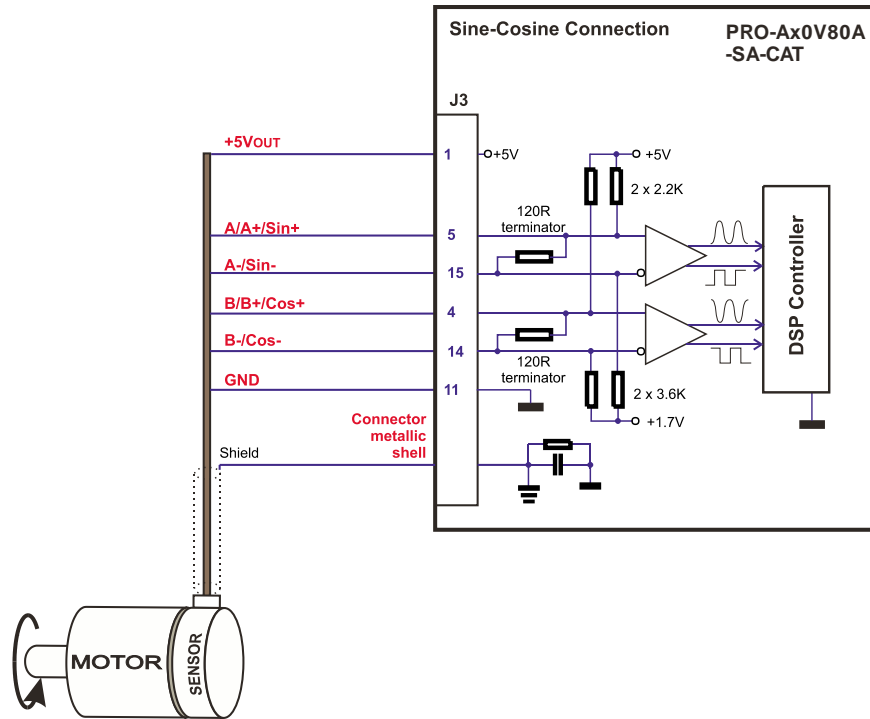


Figure 3.18. Sine-Cosine analog encoder connection

Remark: Termination resistors (120Ω) are present in the drive.

3.3.15.6 Recommendations for wiring

- Always connect both positive and negative signals when the position sensor is differential and provides them. Use one twisted pair for each differential group of signals as follows: A+/Sin+ with A-/Sin-, B+/Cos+ with B-/Cos-, Z+ with Z-. Use another twisted pair for the 5V supply and GND.
- If the PRO-Ax0V80 5V supply output is used by another device (like for example an encoder) and the connection cable is longer than 5 meters, add a decoupling capacitor near the supplied device, between the +5V and GND lines. The capacitor value can be 1...10 μF , rated at 6.3V.

3.3.16. Power Supply Connection

3.3.16.1 Supply Connection

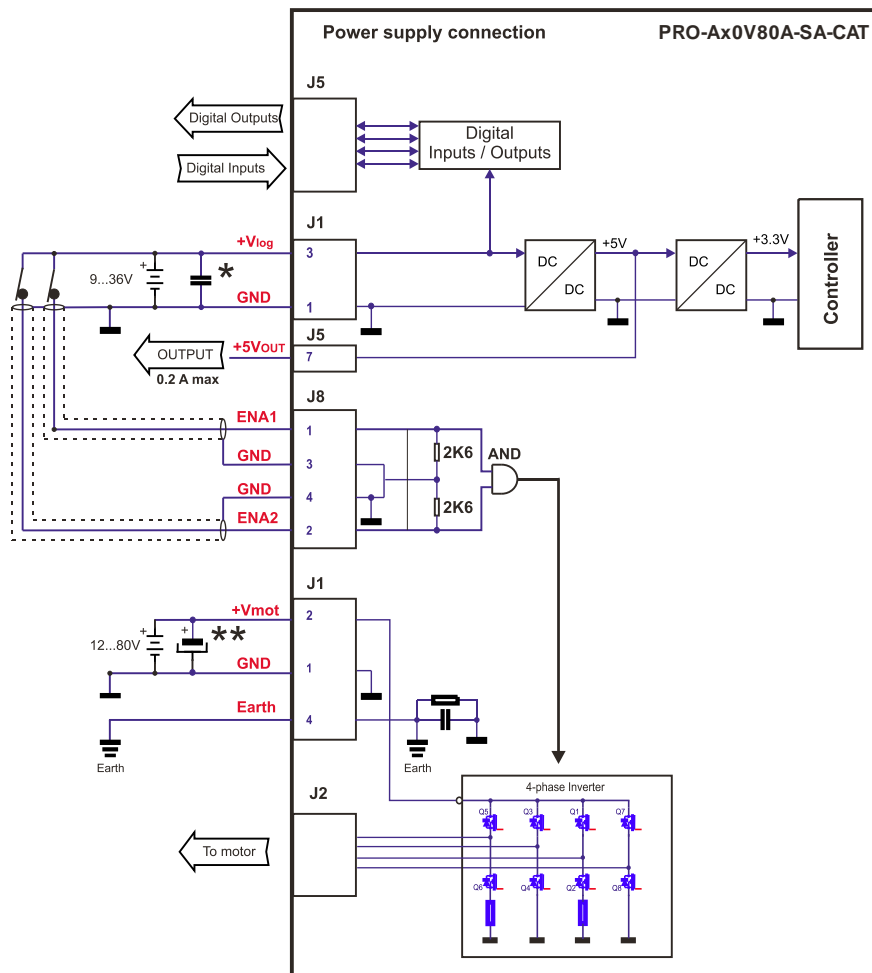


Figure 3.19. Supply connection

Remarks:

- * For compliance to EMC surge immunity, an external capacitor must be added between +Vlog and GND. The capacitor shall be of 100µF minimum, rated for the logic supply voltage.
- **An external electrolytic capacitor may be added between +Vmot and GND, to help reduce over-voltage during load braking/ reversals. See paragraph 3.3.16.3 for details.

3.3.16.2 Recommendations for Supply Wiring

The PRO-Ax0V80A-SA-CAT always requires two supply voltages: V_{log} and V_{mot} .

Use short, thick wires between the PRO-Ax0V80 and the motor power supply. Connect power supply wires to all the indicated pins. If the wires are longer than 2 meters, use twisted wires for the supply and ground return. For wires longer than 20 meters, add a capacitor of at least 4,700 μ F (rated at an appropriate voltage) right on the terminals of the PRO-Ax0V80.

It is recommended to connect the negative motor supply return (GND) to the Earth protection near the power supply terminals.

3.3.16.3 Recommendations to limit over-voltage during braking

During abrupt motion brakes or reversals the regenerative energy is injected into the motor power supply. This may cause an increase of the motor supply voltage (depending on the power supply characteristics). If the voltage bypasses 84V, the drive over-voltage protection is triggered and the drive power stage is disabled. In order to avoid this situation you have 2 options:

Option 1. Add a capacitor on the motor supply big enough to absorb the overall energy flowing back to the supply. The capacitor must be rated to a voltage equal or bigger than the maximum expected over-voltage and can be sized with the formula:

$$C \geq \frac{2 \times E_M}{U_{MAX}^2 - U_{NOM}^2}$$

where:

U_{MAX} = 84V is the over-voltage protection limit

U_{NOM} is the nominal motor supply voltage

E_M = the overall energy flowing back to the supply in Joules. In case of a rotary motor and load,

E_M can be computed with the formula:

$$E_M = \underbrace{\frac{1}{2}(J_M + J_L)\omega_M^2}_{\text{Kinetic energy}} + \underbrace{(m_M + m_L)g(h_{\text{initial}} - h_{\text{final}})}_{\text{Potential energy}} - \underbrace{3I_M^2 R_{Ph} t_d}_{\text{Copper losses}} - \underbrace{\frac{t_d \omega_M}{2} T_F}_{\text{Friction losses}}$$

where:

J_M – total rotor inertia [kgm^2]

J_L – total load inertia as seen at motor shaft after transmission [kgm^2]

ω_M – motor angular speed before deceleration [rad/s]

m_M – motor mass [kg] – when motor is moving in a non-horizontal plane

m_L – load mass [kg] – when load is moving in a non-horizontal plane

g – gravitational acceleration i.e. 9.8 [m/s^2]

h_{initial} – initial system altitude [m]

h_{final} – final system altitude [m]

I_M – motor current during deceleration [A_{RMS}/phase]

R_{Ph} – motor phase resistance [Ω]

t_d – time to decelerate [s]

T_F – total friction torque as seen at motor shaft [Nm] – includes load and transmission

In case of a linear motor and load, the motor inertia J_M and the load inertia J_L will be replaced by the motor mass and the load mass measured in [kg], the angular speed ω_M will become linear speed measured in [m/s] and the friction torque T_F will become friction force measured in [N].

Option 2. Connect a brake resistor R_{BR} between phase BR/B- (J2/ pin 4) and ground (J1/ pin 1), and activate the software option of dynamic braking (see below).

This option is not available when the drive is used with a step motor.

The dynamic braking option can be found in the Drive Setup dialogue within MotionPRO Developer / PRO Config. The braking will occur when DC bus voltage increases over U_{BRAKE} . This parameter (U_{BRAKE}) should be adjusted depending on the nominal motor supply. Optimally (from a braking point of view), U_{BRAKE} should be a few volts above the maximum nominal supply voltage. This setting will activate the brake resistor earlier, before reaching dangerous voltages – when the over-voltage protection will stop the drive. Of course, U_{BRAKE} must always be less than U_{MAX} – the over-voltage protection threshold.

Remark: This option can be combined with an external capacitor whose value is not enough to absorb the entire regenerative energy E_M but can help reducing the brake resistor size.

Brake resistor selection

The brake resistor value must be chosen to respect the following conditions:

1. to limit the maximum current below the drive peak current $I_{PEAK} = 38.3A$

$$R_{BR} > \frac{U_{MAX}}{I_{PEAK}}$$

2. to sustain the required *braking power*:

$$P_{BR} = \frac{E_M - \frac{1}{2}C(U_{MAX}^2 - U_{brake}^2)}{t_d}$$

where C is the capacitance on the motor supply (external), i.e:

$$R_{BR} < \frac{U_{BRAKE}^2}{2 \times P_{BR}}$$

3. to limit the average current below the drive nominal current $I_{NOM}=10A$

$$R_{BR} > \frac{P_{BR} \times t_d}{t_{CYCLE} \times I_{NOM}^2}$$

where t_{CYCLE} is the time interval between 2 brakes in case of repetitive moves.

4. to be rated for an average power $P_{AV} = \frac{P_{BR} \times t_d}{t_{CYCLE}}$ and a peak power $P_{PEAK} = \frac{U_{MAX}^2}{R_{BR}}$

Remarks:

1. If $\frac{U_{MAX}}{I_{PEAK}} > \frac{U_{BRAKE}^2}{2 \times P_{BR}}$ the braking power P_{BR} must be reduced by increasing either t_d – the time to decelerate or C – the external capacitor on the motor supply
2. If $\frac{P_{BR} \times t_d}{t_{CYCLE} \times I_{NOM}^2} > \frac{U_{BRAKE}^2}{2 \times P_{BR}}$ either the braking power must be reduced (see Remark 1) or t_{CYCLE} – the time interval between braking cycles must be increased

	WARNING!	THE BRAKE RESISTOR MAY HAVE HOT SURFACES DURING OPERATION.
---	-----------------	---

3.3.17. Serial RS-232 connection

3.3.17.1 Serial RS-232 connection

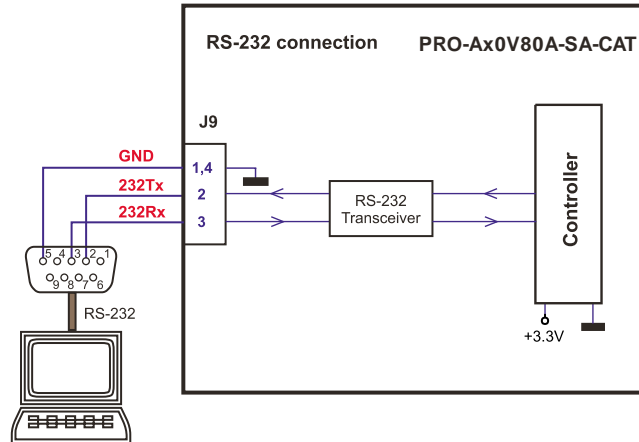


Figure 3.20. Serial RS-232 connection

3.3.17.2 Recommendation for wiring

- If you build the serial cable, you can use a 3-wire shielded cable with shield connected to BOTH ends. Do not use the shield as GND. The ground wire (pin 1 or 4 of J9) must be included inside the shield, like the 232Rx and 232Tx signals
- Always power-off all the PRO-Ax0V80A-SA-CAT supplies before inserting/removing the RS-232 serial connector
- Do not rely on an earthed PC to provide the PRO-Ax0V80 GND connection! The drive must be earthed through a separate circuit. Most communication problems are caused by the lack of such connection



CAUTION! DO NOT CONNECT/DISCONNECT THE RS-232 CABLE WHILE THE DRIVE IS POWERED ON. THIS OPERATION CAN DAMAGE THE DRIVE

3.3.18. Recommendations for EtherCAT® Wiring

- a) Build EtherCAT® network using UTP (unshielded twisted pair) cables rated CAT5E or higher (CAT6, etc.). Cables with this rating must have multiple characteristics, as described in TIA/EIA-568-B. Among these are: impedance, frequency attenuation, cross-talk, return loss, etc.
- b) It is acceptable to use STP (shielded twisted pair) or FTP (foil twisted pair) cables, rated CAT5E or higher (CAT6, etc.). The added shielding is beneficial in reducing the RF (radio-frequency) emissions, improving the EMC emissions of the application.
- c) The maximum length of each network segment must be less than 100 meters.
- d) The network topology is daisy-chain. All connections are done using point-to-point cables. The global topology can be one of the two:
 - Linear, when the J7 / OUT port of the last drive in the chain remains not connected. Master is connected to J6 / IN port of the first drive; J7 / OUT of the first drive is connected to J6 / IN of the following drive; J7 / OUT of the last drive remains unconnected.
See *Figure 3.22* for a visual representation of the linear topology.
 - Ring, when the J7 / OUT port of the last drive in the chain is connected back to the master controller, on the 2nd port of the master. This topology consists of the linear topology described above, plus an extra connection between the master, which has two RJ45 ports, to J5 / OUT of the last drive.
See *Figure 3.23* for a visual representation of the ring topology.
- e) Ring topology is preferred for its added security, since it is insensitive to one broken cable / connection along the ring (re-routing of communication is done automatically, so that to avoid the broken cable / connection)
- f) It is highly recommended to use qualified cables, assembled by a specialized manufacturer. When using CAT5E UTP cables that are manufactured / commissioned / prepared on-site, it is highly recommended to check the cables. The check should be performed using a dedicated Ethernet cable tester, which verifies more parameters than simple galvanic continuity (such as cross-talk, attenuation, etc.). The activation of “Link” indicators will NOT guarantee a stable and reliable connection! This can only be guaranteed by proper quality of cables used, according to TIA/EIA-568-B specifications.

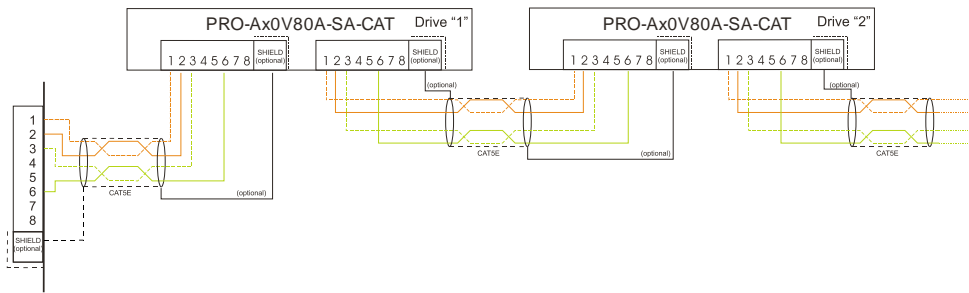


Figure 3.21. EtherCAT® Wiring

Linear Topology

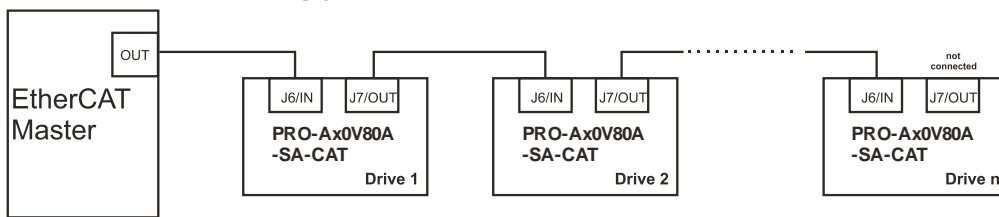


Figure 3.22. EtherCAT® Network - Linear Topology

Ring Topology

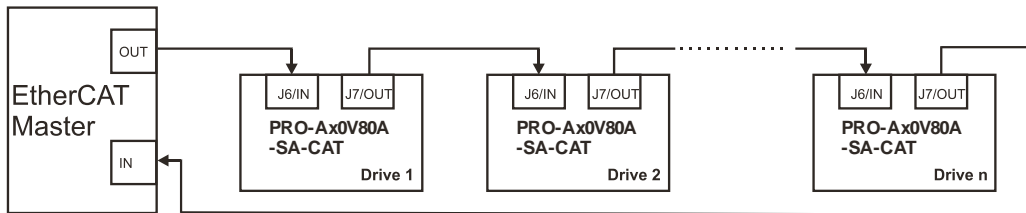


Figure 3.23. EtherCAT® Network - Ring Topology

3.3.19. Disabling the setup table at startup

In some very rare cases, the setup table might be corrupted, leading to a loop where the drive resets continuously. This behavior can be noticed by seeing both the Ready and Error LED blinking for short periods of time continuously.

To recover from this behavior, the setup table can be invalidated by connecting all digital Hall inputs to GND, as shown in *Figure 3.24*.

On the next power on, the drive will load setup default settings and the Motion Error Register (MER) bit 2 will be 1. After a new valid setup table is loaded onto the drive, disconnect the hall sensors from GND and execute a new power off/ power on cycle.

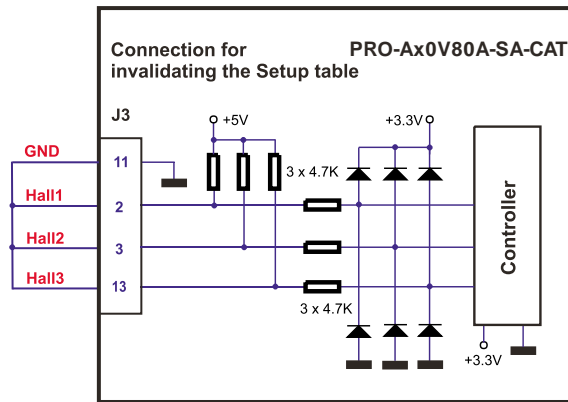


Figure 3.24. Temporary connection during power-on to disable the setup table

3.4. Axis ID Selection for PRO-Ax0V80A-SA-CAT drives with EtherCAT

The PRO-Ax0V80A-SA-CAT drives support all EtherCAT standard addressing modes. In case of device addressing mode based on node address, the PRO-Ax0V80A-SA-CAT drive sets the *configured station alias* address with its AxisID value. The drive AxisID value is set after power on by:

- Software, setting via PRO Config a specific AxisID value in the range 1-255.
- Hardware, by setting h/w in PRO Config and selecting a value between 1-127 from the switch SW1

3.5. DIP-Switch Settings

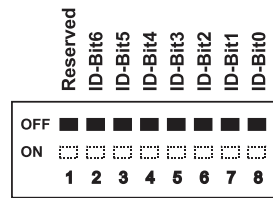


Figure 3.25. SW1 – DIP Switch

- **Position 1:** Reserved, do not use
- **Positions 2 ... 8:** ID-Bitx.
 - **Axis ID switches** The drive axis/address number is set when H/W is selected in Drive Setup under AxisID field or when the Setup is invalid.
 - The axis ID is an 8 bit unsigned number. Its first 7 bits are controlled by the ID-bit0 to ID-bit6. Bit7 of this variable is always 0. In total, 127 axis ID HW values can result from the DIP switch combinations.

Remark: All switches are sampled at power-up, and the drive is configured accordingly

3.6. LED Indicators

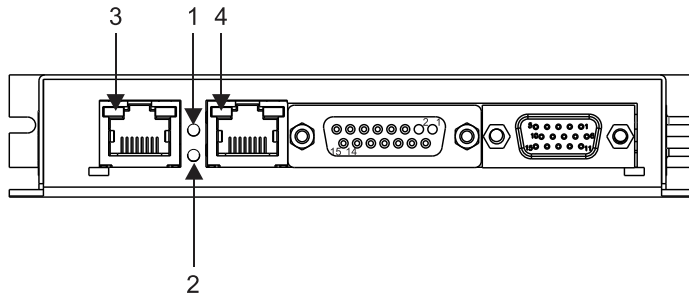


Figure 3.26. LED locations

Table 3.1 – LED Indicators

LED no.	LED name	LED color	Function
1	Drive Ready/ Error	green	Lit after power-on when the drive initialization ends. Turned off when an error occurs.
		red	Turned on when the drive detects an error condition or when OUT2/Error is set to +Vlog with OUT(2)=0 MPL instruction.
3	EtherCAT® OUT Link/Activity	green	Shows the state of the physical link and activity.
4	EtherCAT® IN Link/Activity	green	Shows the state of the physical link and activity.
5	EtherCAT® ST	red and green	EtherCAT® ERROR and RUN indicators combined. Shows the state of the EtherCAT® Status Machine

3.6.1. EtherCAT® Status indicator

The STATUS indicator displays the RUN states with a 180 degree phase shift to the ERROR states. An example is noted in **Figure 3.27**.

In case of a conflict between turning the indicator on green versus red, the indicator shall be turned to red. Apart from this situation, the bi-color STATUS indicator combines the behavior of the RUN indicator specified in **Table 3.2** and the behavior of the ERROR indicator specified in **Table 3.3**.

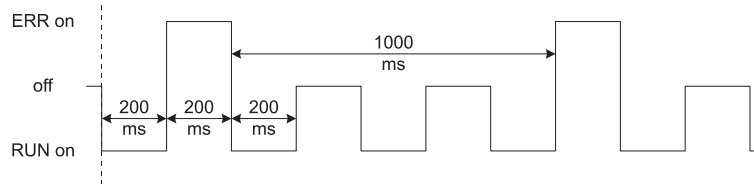


Figure 3.27. STATUS indicator Example

Table 3.2 – RUN Indicator States

Indicator states	Slave State	Description
Off	INITIALISATION	The drive is in state INIT
Blinking	PRE-OPERATIONAL	The drive is in state PRE-OEPRATIONAL
Single Flash	SAFE-OPERATIONAL	The drive is in state SAFE-OPERATIONAL
On	OPERATIONAL	The drive is in state OPERATIONAL

Table 3.3 – ERROR Indicator States

ERR state	Error name	Description
On	Application controller failure	An critical communication or application controller error has occurred
Double Flash	Process Data Watchdog Timeout/ EtherCAT Watchdog Timeout	An application watchdog timeout has occurred.
Single Flash	Local Error	Slave device application has changed the EtherCAT state autonomously, due to local error (see ETG.1000 part 6 EtherCAT State Machine). Error Indicator bit is set to 1 in AL Status register.
Blinking	Invalid Configuration	General Configuration Error
Flickering	Booting Error	Booting Error was detected. INIT state reached, but Error Indicator bit is set to 1 in AL Status register
Off	No error	The EtherCAT communication of the device is in working condition

For a more detailed description of EtherCAT® LED functionalities please read ETG.1300 S (R) V1.0.1 available at www.EtherCAT.org .

4. Step 2. Drive Setup

4.1. Installing PRO Config

PRO Config is a PC software platform for the setup of the ElectroCraft drives. PRO Config comes with an **Update via Internet tool** through which you can check if your software version is up-to-date, and when necessary download and install the latest updates.

You will need MotionPRO Developer only if you plan to use the advanced features presented in Section 5.3 Combining CoE with MPL.

4.2. Getting Started with PRO Config

Using PRO Config you can quickly setup a drive for your application. The drive can be directly connected with your PC **via a serial RS 232 link.**

The output of PRO Config is a set of *setup data*, which can be downloaded into the drive EEPROM or saved on your PC for later use.

PRO Config includes a set of evaluation tools like the Data Logger, the Control Panel and the Command Interpreter which help you to quickly measure, check and analyze your drive commissioning.

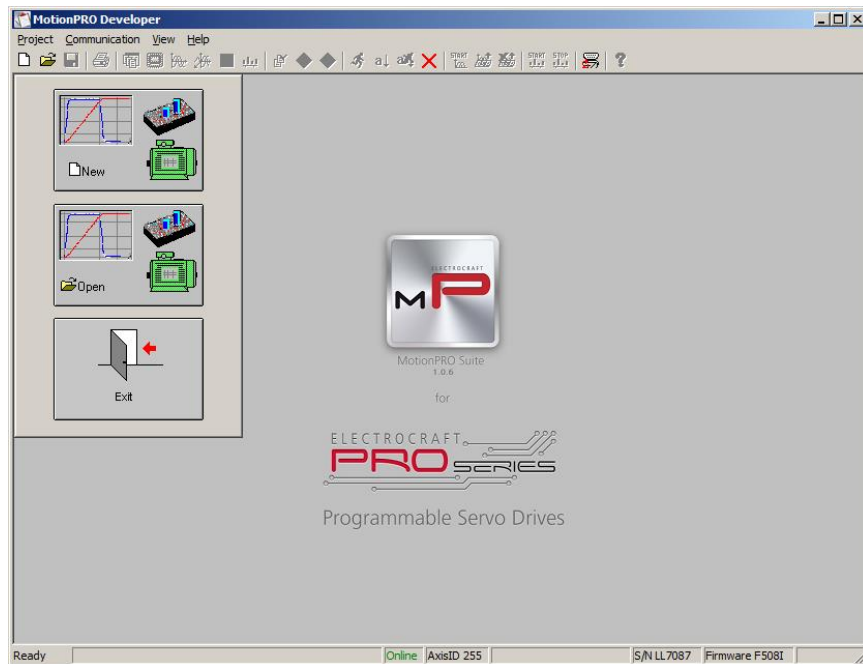
PRO Config works with **setup** data. A **setup** contains all the information needed to configure and parameterize a ElectroCraft drive. This information is preserved in the drive EEPROM in the *setup table*. The setup table is copied at power-on into the RAM memory of the drive and is used during runtime. With PRO Config it is also possible to retrieve the complete setup information from a drive previously programmed.

Note that with PRO Config you do only your drive/motor commissioning. For motion programming you have the following options:

- Use a **EtherCAT® master**
- Use **MotionPRO Developer** to create and download a **MPL** program into the drive/motor memory
- **Implement** on your master the MPL commands you need to send to the drives using one of the supported communication channels. The implementation must be done according with ElectroCraft communication protocols.
- **Combine** MPL programming at drive level with one of the other options (see Section 5.3)

4.2.1. Establish communication

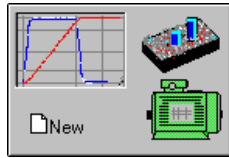
PRO Config starts with an empty window from where you can create a **New** setup, **Open** a previously created setup which was saved on your PC, or **Upload** the setup from the drive/motor.



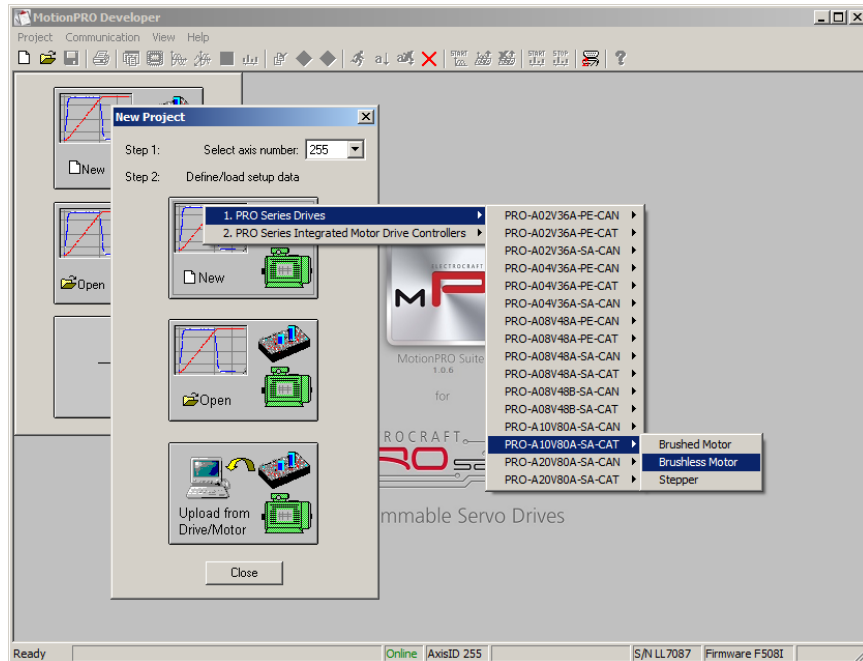
Before selecting one of the above options, you need to establish the communication with the drive you want to commission. Use menu command **Communication | Setup** to check/change your PC communication settings. Press the **Help** button of the dialogue opened. Here you can find detailed information about how to setup your drive and do the connections. Power on the drive, then close the **Communication | Setup** dialogue with OK. If the communication is established, PRO Config displays in the status bar (the bottom line) the text “**Online**” plus the axis ID of your drive/motor and its firmware version. Otherwise the text displayed is “**Offline**” and a communication error message tells you the error type. In this case, return to the **Communication | Setup** dialogue, press the Help button and check troubleshoots

Remark: When first started, PRO Config tries to communicate via RS-232 and COM1 with a drive having axis ID=255 (default communication settings). If your drive is powered and it is connected to your PC port COM1 via an RS-232 cable, the communication shall establish automatically. If the drive has a different axis ID and you don't know it, select in the **Communication | Setup** dialogue at “Axis ID of drive/motor connected to PC” the option **Autodetected**.

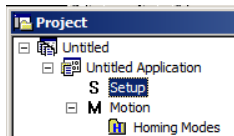
4.2.2. Setup drive/motor



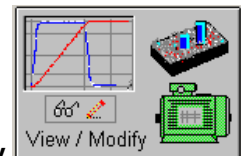
Press **New** button and select your drive type.



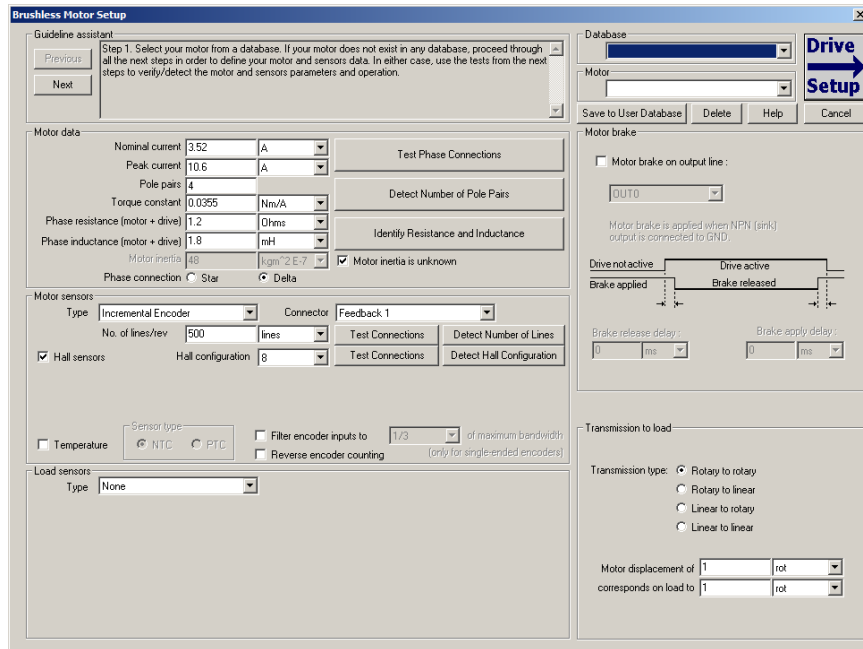
The selection continues with the motor technology (for example: brushless, brushed or stepper).



Click **Setup** in the Project window. Then press **Edit View/Modify**

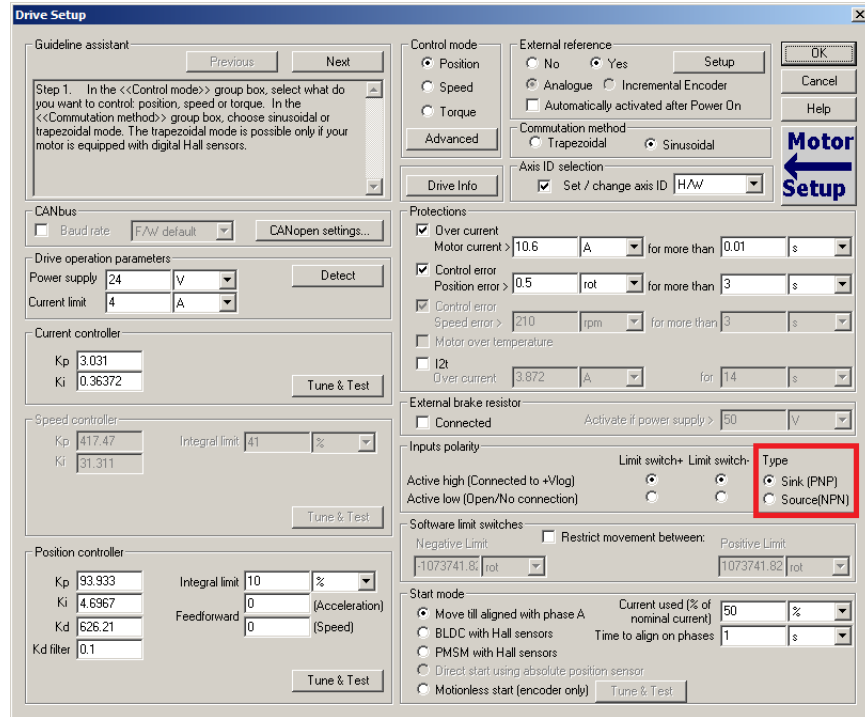


The selection opens 2 setup dialogues: for **Motor Setup** and for **Drive setup** through which you can configure and parameterize a ElectroCraft drive, plus several predefined control panels customized for the product selected.



In the **Motor setup** dialogue you can introduce the data of your motor and the associated feedback sensors. Data introduction is accompanied by a series of tests having as goal to check the connections to the drive and/or to determine or validate a part of the motor and sensors parameters. In the **Drive setup** dialogue you can configure and parameterize the drive for your application. In each dialogue you will find a **Guideline Assistant**, which will guide you through the whole process of introducing and/or checking your data. Close the Drive setup dialogue with **OK** to keep all the changes regarding the motor and the drive setup.

4.2.3. Selecting NPN/PNP inputs type in Setup

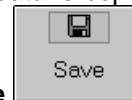


In Drive Setup, choose the inputs type PNP or NPN.

4.2.4. Download setup data to drive/motor



Press the **Download to Drive/Motor** button to download your setup data in the drive/motor EEPROM memory in the *setup table*. From now on, at each power-on, the setup data is copied into the



drive/motor RAM memory which is used during runtime. It is also possible to **Save** the setup data on your PC and use it in other applications.

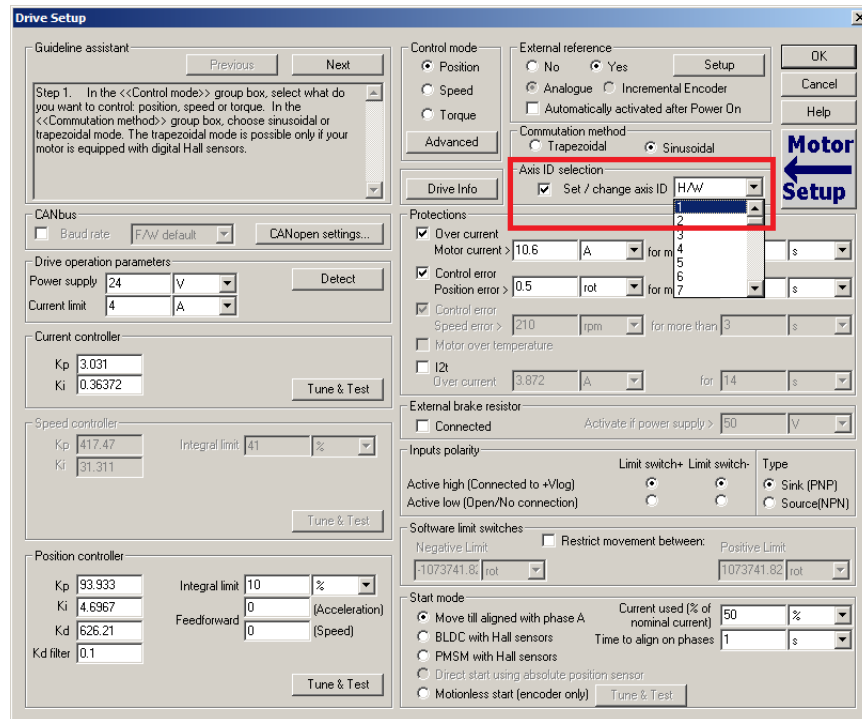
To summarize, you can define or change the setup data in the following ways:

- create a new setup data by going through the motor and drive dialogues
- use setup data previously saved in the PC
- upload setup data from a drive/motor EEPROM memory

4.2.5. Evaluate drive/motor behavior (optional)

You can use the **Data Logger** or the **Control Panel** evaluation tools to quickly measure and analyze your application behavior. In case of errors like protections triggered, use the Drive Status control panel to find the cause.

4.3. Changing the drive Axis ID



The axis ID of an PRO-Ax0V80 drive can be set in 2 ways:

- Hardware (H/W) – according with AxisID DIP switch SW1 par. 3.5.
- Software (via Setup)– any value between 1 and 255, stored in the setup table..

The axis ID is initialized at power on, using the following algorithm:

- a) If a valid setup table exists, and this setup table was created with the *Axis ID Selection* checkbox checked in the Drive Setup dialogue (see above) – with the value read from the setup table. This value can be an axis number 1 to 255.
- b) If the setup table is invalid, with the last value set from a valid setup table.
- c) If the setup table is invalid, there is no previous axis ID set from a valid setup table, will be set as 255.

Remark: If you don't know the axis ID set in a drive, you can find it in the following way:

- a) Connect the drive via a serial RS232 link to a PC where PRO Config or MotionPRO Developer are installed

-
- b) *With the drive powered, open PRO Config or MotionPRO Developer and check the status bar. If communication with the drive is established, the status bar displays **Online** in green and nearby the drive's Axis ID. If the status bar displays **Offline** in red, execute menu command "Communication|Setup..." and in the dialogue opened select at "Channel Type" **RS232** and at "Axis ID of drive/motor connected to PC" the option **Autodetected**. After closing the dialogue with OK, communication with the drive shall be established and the status bar shall display the drive's Axis ID*

4.4. Creating an Image File with the Setup Data

Once you have validated your setup, you can create with the menu command **Setup | Create EEPROM Programmer File** a software file (with extension **.sw**) which contains all the setup data to write in the EEPROM of your drive.

A software file is a text file that can be read with any text editor. It contains blocks of data separated by an empty row. Each block of data starts with the block start address, followed by data values to place in ascending order at consecutive addresses: first data – to write at start address, second data – to write at start address + 1, etc. All the data are hexadecimal 16-bit values (maximum 4 hexadecimal digits). Each row contains a single data value. When less than 4 hexadecimal digits are shown, the value must be right justified. For example 92 represents 0x0092.

The **.sw** file can be programmed into a drive:

- from an EtherCAT master, using the communication objects for writing data into the drive EEPROM
- using the EEPROM Programmer tool, which comes with PRO Config but may also be installed separately. The EEPROM Programmer was specifically designed for repetitive fast and easy programming of **.sw** files into the ElectroCraft drives during production.

5. Step 3. Motion Programming

5.1. Using an EtherCAT® master

5.1.1. Modes of Operation

The PRO-Ax0V80A-SA-CAT drive supports CAN application protocol over EtherCAT® (CoE) protocol with the following CiA 402 modes of operation:

- Profile position mode
- Profile velocity mode
- Homing mode
- Interpolated position mode
- Cyclic synchronous position mode
- Cyclic synchronous speed mode
- Cyclic synchronous torque mode

Additional to these modes, there are also several manufacturer specific modes defined:

- Electronic gearing position mode
- Electronic camming position mode

For details see the EtherCAT Programming manual (part no. P091.064.UM.xxxx)

5.1.2. Checking Setup Data Consistency

During the configuration phase, an EtherCAT® master can quickly verify using the checksum objects and a reference **.sw** file (see 5.2.4 for details) whether the non-volatile EEPROM memory of an PRO-Ax0V80 drive contains the right information. If the checksum reported by the drive doesn't match with that computed from the **.sw** file, the CANopen master can download the entire **.sw** file into the drive EEPROM using the communication objects for writing data into the drive EEPROM.

5.2. Using the built-in Motion Controller and MPL

One of the key advantages of the ElectroCraft drives is their capability to execute complex motions without requiring an external motion controller. This is possible because ElectroCraft drives offer in a single compact package both a state of art digital drive and a powerful motion controller.

5.2.1. ElectroCraft's Motion Programming Language Overview

Programming motion directly on a ElectroCraft drive requires creating and downloading a MPL (Motion Programming Language) program into the drive memory. The MPL allows you to:

- Set various motion modes (profiles, PVT, PT, electronic gearing or camming, etc.)
- Change the motion modes and/or the motion parameters
- Execute homing sequences
- Control the program flow through:
 - Conditional jumps and calls of MPL functions

-
- MPL interrupts generated on pre-defined or programmable conditions (protections triggered, transitions on limit switch or capture inputs, etc.)
 - Waits for programmed events to occur
- Handle digital I/O and analog input signals
 - Execute arithmetic and logic operations

In order to program a motion using MPL you need MotionPRO Developer software platform.

5.2.2. Installing MotionPRO Developer

MotionPRO Developer is an integrated development environment for the setup and motion programming of ElectroCraft programmable drives. It comes with an **Update via Internet tool** through which you can check if your software version is up-to-date, and when necessary download and install the latest updates.

A **demo version of MotionPRO Developer** including the **fully functional version of PRO Config** can be downloaded free of charge from ElectroCraft web page.

MotionPRO Developer is delivered on a CD. Once you have started the installation package, follow its indications. After installation, use the update via internet tool to check for the latest updates. Alternately, you can first install the demo version and then purchase a license. By introducing the license serial number in the menu command **Help | Enter registration info...**, you can transform the demo version into a fully functional version.

5.2.3. Getting Started with MotionPRO Developer

Using MotionPRO Developer you can quickly do the setup and the motion programming of a ElectroCraft a drive according with your application needs. The drive can be directly connected with your PC via a serial RS 232 link.

The output of the MotionPRO Developer is a set of setup data and a motion program, which can be downloaded to the drive/motor EEPROM or saved on your PC for later use.

MotionPRO Developer includes a set of evaluation tools like the Data Logger, the Control Panel and the Command Interpreter which help you to quickly develop, test, measure and analyze your motion application.

MotionPRO Developer works with **projects**. A project contains one or several **Applications**.

Each application describes a motion system for one axis. It has 2 components: the **Setup** data and the **Motion** program and an associated axis number: an integer value between 1 and 255. An application may be used either to describe:

1. One axis in a multiple-axis system
2. An alternate configuration (set of parameters) for the same axis.

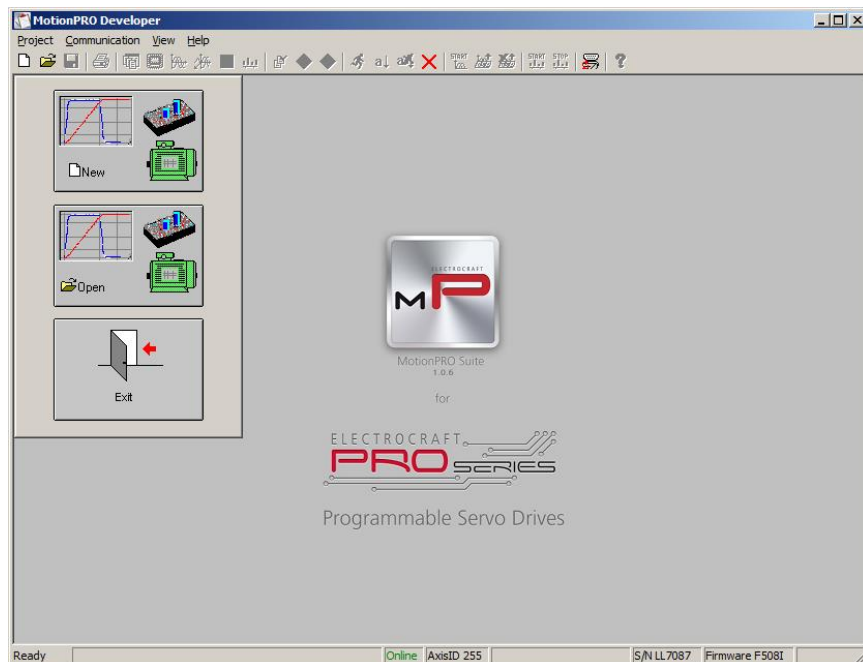
In the first case, each application has a different axis number corresponding to the axis ID of the drives/motors from the network. All data exchanges are done with the drive/motor having the same address as the selected application. In the second case, all the applications have the same axis number.

The setup component contains all the information needed to configure and parameterize a ElectroCraft drive. This information is preserved in the drive/motor EEPROM in the *setup table*. The setup table is copied at power-on into the RAM memory of the drive/motor and is used during runtime.

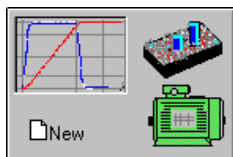
The motion component contains the motion sequences to do. These are described via a MPL (Motion Programming Language) program, which is executed by the drives/motors built-in motion controller.

5.2.3.1 Create a new project

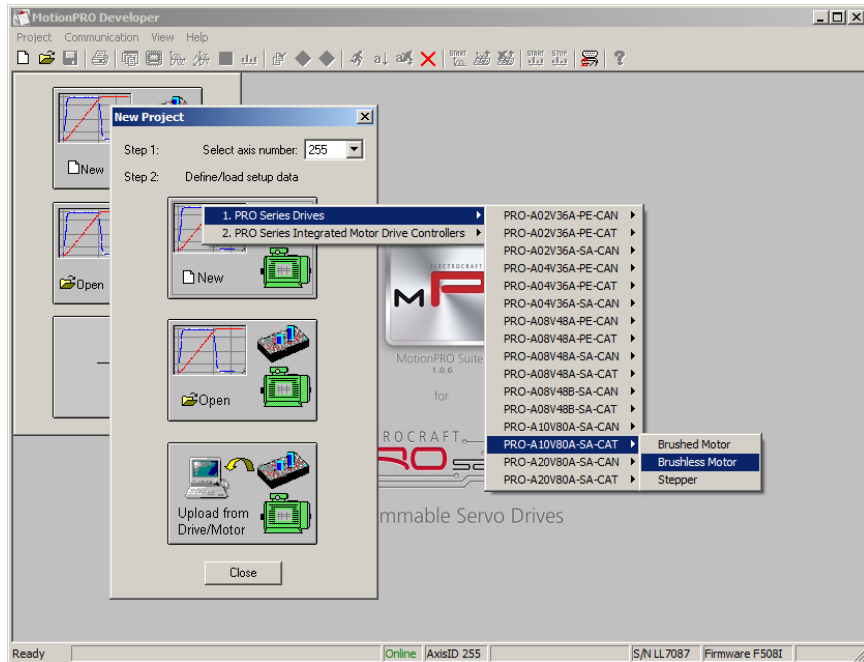
MotionPRO Developer starts with an empty window from where you can create a new project or open a previously created one.



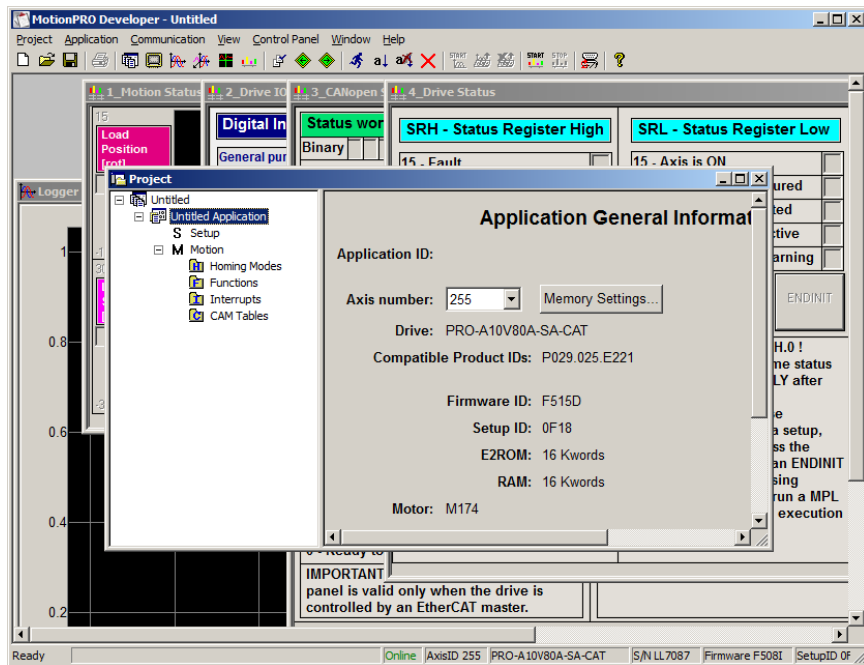
When you start a new project, MotionPRO Developer automatically creates a first application. Additional applications can be added later. You can duplicate an application or insert one defined in another project.



Press **New** button to open the “New Project” dialogue. Set the axis number for your first application equal with your drive/motor axis ID. The initial value proposed is 255 which is the default axis ID of the drives. Press **New** button and select your drive type. Depending on the product chosen, the selection may continue with the motor technology (for example: brushless or brushed) and the type of feedback device (for example: incremental encoder).



Click on your selection. MotionPRO Developer opens the Project window where on the left side you can see the structure of a project. At beginning both the new project and its first application are named “Untitled”. The application has 2 components: **S** Setup and **M** Motion (program).



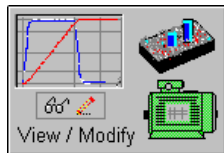
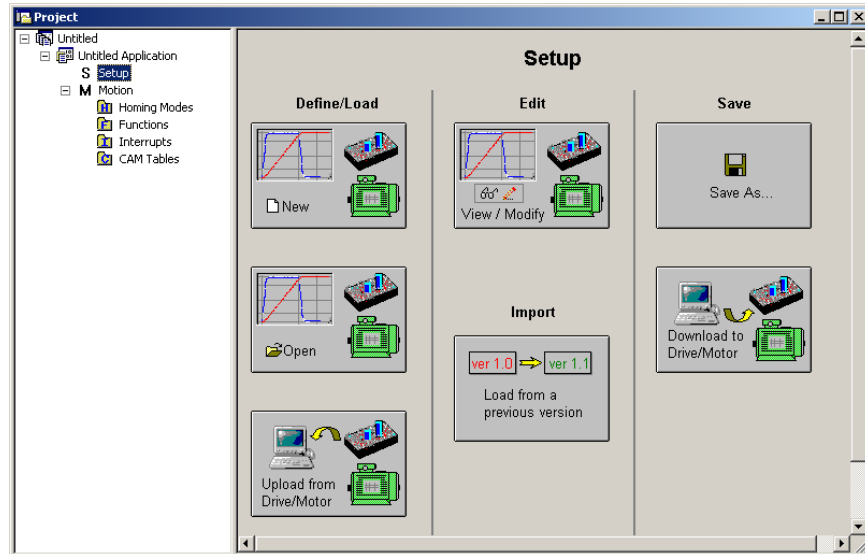
5.2.3.2 Step 2 Establish communication

If you have a drive/motor connected with your PC, now its time to check the communication. Use menu command **Communication | Setup** to check/change your PC communication settings. Press the **Help** button of the dialogue opened. Here you can find detailed information about how to setup your drive/motor and the connections. Power on the drive, then close the Communication | Setup dialogue with OK. If the communication is established, MotionPRO Developer displays in the status bar (the bottom line) the text “**Online**” plus the axis ID of your drive/motor and its firmware version. Otherwise the text displayed is “**Offline**” and a communication error message tells you the error type. In this case, return to the Communication | Setup dialogue, press the Help button and check troubleshoots.

Remark: *When first started, MotionPRO Developer tries to communicate via RS-232 and COM1 with a drive having axis ID=255 (default communication settings). If your drive is powered and it is connected to your PC port COM1 via an RS-232 cable, the communication shall establish automatically.*

5.2.3.3 Setup drive/motor

In the project window left side, select “S Setup”, to access the setup data for your application.



Press **View/Modify** button. This opens 2 setup dialogues: for **Motor Setup** and for **Drive Setup** (same like on PRO Config) through which you can configure and parameterize a ElectroCraft drive. In the **Motor setup** dialogue you can introduce the data of your motor and the associated sensors. Data introduction is accompanied by a series of tests having as goal to check the connections to the drive and/or to determine or validate a part of the motor and sensors parameters. In the **Drive setup** dialogue you can configure and parameterize the drive for your application. In each dialogue you will find a **Guideline Assistant**, which will guide you through the whole process of introducing and/or checking your data.



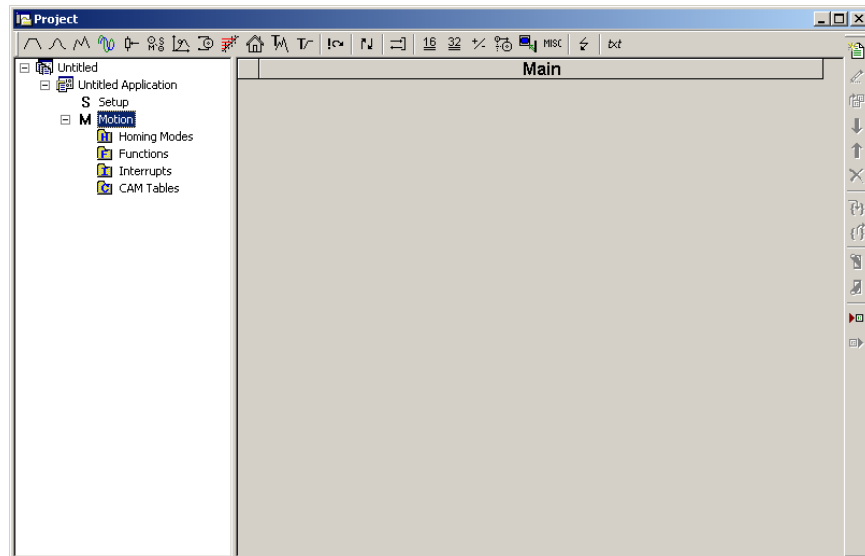
Press the **Download to Drive/Motor** button to download your setup data in the drive/motor EEPROM memory in the *setup table*. From now on, at each power-on, the setup data is copied into the drive/motor RAM memory which is used during runtime. It is also possible to save the setup data on your PC and use it in other applications. Note that you can upload the complete setup data from a drive/motor.

To summarize, you can define or change the setup data of an application in the following ways:

- create a new setup data by going through the motor and drive dialogues
- use setup data previously saved in the PC
- upload setup data from a drive/motor EEPROM memory

5.2.3.4 Program motion

In the project window left side, select “M Motion”, for motion programming. This automatically activates the **Motion Wizard**.



The Motion Wizard offers you the possibility to program all the motion sequences using high level graphical dialogues which automatically generate the corresponding MPL instructions. Therefore with Motion Wizard you can develop motion programs using almost all the MPL instructions without needing to learn them. A MPL program includes a main section, followed by the subroutines used: functions, interrupt service routines and homing procedures. The MPL program may also include cam tables used for electronic camming applications.

When activated, Motion Wizard adds a set of toolbar buttons in the project window just below the title. Each button opens a programming dialogue. When a programming dialogue is closed, the associated MPL instructions are automatically generated. Note that, the MPL instructions generated are not a simple text included in a file, but a motion object. Therefore with Motion Wizard you define your motion program as a collection of motion objects.

The major advantage of encapsulating programming instructions in motion objects is that you can very easily manipulate them. For example, you can:

- Save and reuse a complete motion program or parts of it in other applications
- Add, delete, move, copy, insert, enable or disable one or more motion objects
- Group several motion objects and work with bigger objects that perform more complex functions

As a starting point, push for example the leftmost Motion Wizard button – Trapezoidal profiles, and set a position or speed profile. Then press the **Run** button. At this point the following operations are done automatically:

- A MPL program is created by inserting your motion objects into a predefined template
- The MPL program is compiled and downloaded to the drive/motor
- The MPL program execution is started

For learning how to send MPL commands from your host/master, using one of the communication channels and protocols supported by the drives use menu command **Application | Binary Code Viewer...** Using this tool, you can get the exact contents of the messages to send and of those expected to be received as answers.

5.2.3.5 Evaluate motion application performances

MotionPRO Developer includes a set of evaluation tools like the **Data Logger**, the **Control Panel** and the **Command Interpreter** which help you to quickly measure and analyze your motion application.

5.2.4. Creating an Image File with the Setup Data and the MPL Program

Once you have validated your application, you can create with the menu command **Application | Create EEPROM Programmer File** a software file (with extension **.sw**) which contains all the data to write in the EEPROM of your drive. This includes both the setup data and the motion program. For details regarding the **.sw** file format and how it can be programmed into a drive, see paragraph 4.4.

5.3. Combining CoE with MPL

Due to its embedded motion controller, an PRO-Ax0V80A-SA-CAT offers many programming solutions that may simplify a lot the task of a EtherCAT® master. This paragraph overviews a set of advanced programming features which arise when combining MPL programming at drive level with EtherCAT® master control. A detailed description of these advanced programming features is included in the **EtherCAT Programming (part no. P091.064.UM.xxxx)** manual. All features presented below require usage of MotionPRO Developer as MPL programming tool

Remark: *If you don't use the advanced features presented below, you don't need MotionPRO Developer. In this case the PRO-Ax0V80A-SA-CAT is treated like a standard EtherCAT® drive, whose setup is done using PRO Config.*

5.3.1. Using MPL Functions to Split Motion between Master and Drives

With ElectroCraft programmable drives you can really distribute the intelligence between an EtherCAT® master and the drives in complex multi-axis applications. Instead of trying to command each step of an axis movement, you can program the drives using MPL to execute complex tasks and inform the master when these are done. Thus for each axis, the master task may be reduced at: calling MPL functions (with possibility to abort their execution) stored in the drives EEPROM and waiting for a message, which confirms the finalization of the MPL functions execution.

5.3.2. Executing MPL programs

The distributed control concept can go one step further. You may prepare and download into a drive a complete MPL program including functions, homing procedures¹, etc. The MPL program execution can be started by simply writing a value in a dedicated object,

5.3.3. Loading Automatically Cam Tables Defined in MotionPRO Developer

Apart from the standard modes of operation of CiA 402, the PRO-Ax0V80A-SA-CAT offers others like: electronic gearing, electronic camming, external modes with analog or digital reference etc. When electronic camming is used, the cam tables can be loaded in the following ways:

- a) The master downloads the cam points into the drive active RAM memory after each power on;
- b) The cam points are stored in the drive EEPROM and the master commands their copy into the active RAM memory
- c) The cam points are stored in the drive EEPROM and during the drive initialization (transition to Ready to Switch ON status) are automatically copied from EEPROM to the active RAM

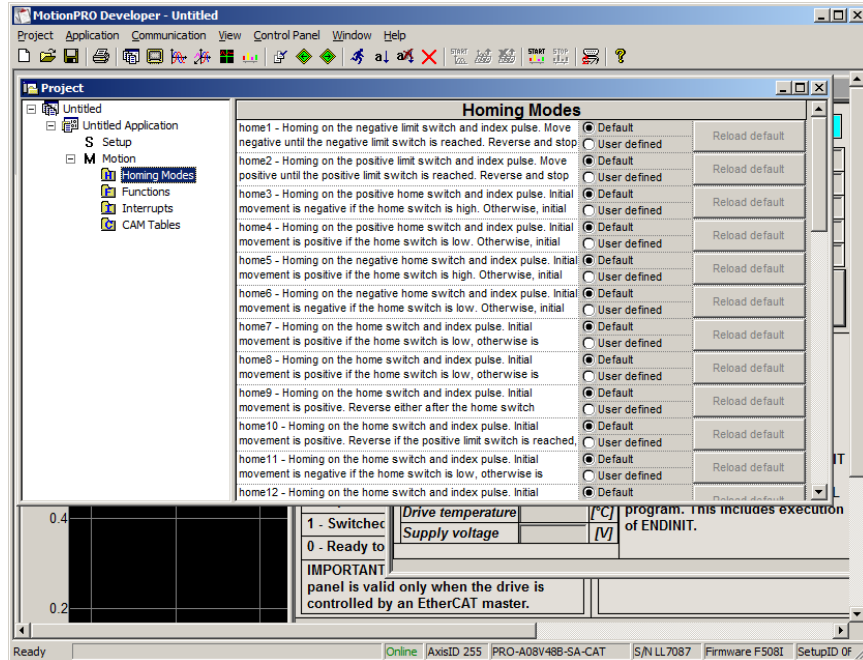
For the last 2 options the cam table(s) are defined in MotionPRO Developer and are included in the information stored in the EEPROM together with the setup data and the MPL programs/functions.

Remark: The cam tables are included in the **.sw** file generated with MotionPRO Developer. Therefore, the drives can check the cam presence in the drive EEPROM using the same procedure as for testing of the setup data.

¹ The customization of the interrupt service routines and homing routines is available only for PRO-Ax0V80 CAN executions

5.3.4. Customizing the Homing Procedures

The PRO-Ax0V80 supports all homing modes defined in CiA402 device profile, plus 4 custom based on hard stop. If needed, any of these homing modes can be customized. In order to do this you need to select the Homing Modes from your MotionPRO Developer application and in the right side to set as “User defined” one of the Homing procedures. Following this operation the selected procedure will occur under Homing Modes in a subtree, with the name *HomeX* where X is the number of the selected homing.



If you click on the *HomeX* procedure, on the right side you'll see the MPL function implementing it. The homing routine can be customized according to your application needs. Its calling name and method remain unchanged.

5.3.5. Customizing the Drive Reaction to Fault Conditions

Similarly to the homing modes, the default service routines for the MPL interrupts can be customized according to your application needs. However, as most of these routines handle the drive reaction to fault conditions, it is mandatory to keep the existent functionality while adding your application needs, in order to preserve the correct protection level of the drive. The procedure for modifying the MPL interrupts is similar with that for the homing modes.

6. Scaling factors

ElectroCraft drives work with parameters and variables represented in the drive internal units (IU). These correspond to various signal types: position, speed, current, voltage, etc. Each type of signal has its own internal representation in IU and a specific scaling factor. This chapter presents the drive internal units and their relation with the international standard units (SI).

In order to easily identify them, each internal unit has been named after its associated signal. For example the **position units** are the internal units for position, the **speed units** are the internal units for speed, etc.

6.1. Position units

6.1.1. Brushless / DC brushed motor with quadrature encoder on motor

The internal position units are encoder counts. The correspondence with the load **position in SI units**¹ is:

$$\text{Load_Position[SI]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines} \times \text{Tr}} \times \text{Motor_Position[IU]}$$

where:

No_encoder_lines – is the rotary encoder number of lines per revolution

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

6.1.2. DC brushed motor with quadrature encoder on load and tacho on motor

The internal position units are encoder counts. The motor is rotary and the transmission is rotary-to-rotary. The correspondence with the load position in SI units is:

$$\text{Load_Position[rad]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines}} \times \text{Load_Position[IU]}$$

where:

No_encoder_lines – is the encoder number of lines per revolution

¹SI units for position are: [rad] for a rotary movement, [m] for a linear movement

6.1.3. Step motor open-loop control. No feedback device

The internal position units are motor μ steps. The correspondence with the load **position in SI units** is:

$$\text{Load_Position[SI]} = \frac{2 \times \pi}{\text{No_}\mu\text{steps} \times \text{No_steps} \times \text{Tr}} \times \text{Motor_Position[IU]}$$

where:

No_steps – is the number of motor steps per revolution

No_μsteps – is the number of microsteps per step. You can read/change this value in the “Drive Setup” dialogue from PRO Config.

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

Step motor closed-loop control. Incremental encoder on motor

The internal position units are motor encoder counts. The correspondence with the load **position in SI units**¹ is:

$$\text{Load_Position[SI]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines} \times \text{Tr}} \times \text{Motor_Position[IU]}$$

where:

No_encoder_lines – is the motor encoder number of lines per revolution

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

6.1.4. Step motor open-loop control. Incremental encoder on load

The internal position units are load encoder counts. The transmission is rotary-to-rotary. The correspondence with the load position in SI units is:

$$\text{Load_Position[SI]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines}} \times \text{Load_Position[IU]}$$

where:

No_encoder_lines – is the rotary encoder number of lines per revolution

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

¹ SI units for position are [rad] for a rotary movement , [m] for a linear movement

6.1.5. Brushless motor with sine/cosine encoder on motor

The internal position units are interpolated encoder counts. The correspondence with the load position in SI units is:

For rotary motors:

$$\text{Load_Position[S]} = \frac{2 \times \pi}{4 \times \text{Enc_periods} \times \text{Interpolation} \times \text{Tr}} \times \text{Motor_Position[IU]}$$

For linear motors:

$$\text{Load_Position[S]} = \frac{\text{Encoder_accuracy}}{\text{Interpolation} \times \text{Tr}} \times \text{Motor_Position[IU]}$$

where:

Enc_periods – is the rotary encoder number of sine/cosine periods or lines per revolution

Interpolation – is the interpolation level inside an encoder period. Its a number power of 2 between 1 and 256. 1 means no interpolation

Encoder_accuracy – is the linear encoder accuracy in [m] for one sine/cosine period

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

6.2. Speed units

The internal speed units are internal position units / (slow loop sampling period) i.e. the position variation over one slow loop sampling period

6.2.1. Brushless / DC brushed motor with quadrature encoder on motor

The internal speed units are encoder counts / (slow loop sampling period). The correspondence with the load **speed in SI units** is:

$$\text{Load_Speed[S]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines} \times \text{Tr} \times \text{T}} \times \text{Motor_Speed[U]}$$

where:

No_encoder_lines – is the rotary encoder number of lines per revolution

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”

6.2.2. DC brushed motor with quadrature encoder on load and tacho on motor

The internal speed units are encoder counts / (slow loop sampling period). The motor is rotary and the transmission is rotary-to-rotary. The correspondence with the load speed in SI units is:

$$\text{Load_Speed[S]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines} \times \text{T}} \times \text{Load_Speed[U]}$$

where:

No_encoder_lines – is the encoder number of lines per revolution

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”

6.2.3. DC brushed motor with tacho on motor

When only a tachometer is mounted on the motor shaft, the internal speed units are A/D converter bits. The correspondence with the load **speed in SI units**¹ is:

$$\text{Load_Speed[S]} = \frac{\text{Analogue_Input_Range}}{4096 \times \text{Tacho_gain} \times \text{Tr}} \times \text{Motor_Speed[U]}$$

where:

Analog_Input_Range – is the range of the drive analog input for feedback, expressed in [V]. You can read this value in the “Drive Info” dialogue, which can be opened from the “Drive Setup”

Tacho_gain – is the tachometer gain expressed in [V/rad/s]

6.2.4. Step motor open-loop control. No feedback device

The internal speed units are motor μ steps / (slow loop sampling period). The correspondence with the load **speed in SI units** is:

$$\text{Load_Speed[S]} = \frac{2 \times \pi}{\text{No_}\mu\text{steps} \times \text{No_steps} \times \text{Tr} \times \text{T}} \times \text{Motor_Speed[U]}$$

where:

No_steps – is the number of motor steps per revolution

No_μsteps – is the number of microsteps per step. You can read/change this value in the “Drive Setup” dialogue from PRO Config.

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”

Step motor open-loop control. Incremental encoder on load

The internal speed units are load encoder counts / (slow loop sampling period). The transmission is rotary-to-rotary. The correspondence with the load speed in SI units is:

$$\text{Load_Speed[rad/s]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines} \times \text{T}} \times \text{Load_Speed[U]}$$

where:

No_encoder_lines – is the rotary encoder number of lines per revolution

Tr – transmission ratio between the motor displacement in [rad] and load displacement in [rad] or [m]

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”.

¹ SI units for speed are [rad/s] for a rotary movement, [m/s] for a linear movement

6.2.5. Step motor closed-loop control. Incremental encoder on motor

The internal speed units are motor encoder counts / (slow loop sampling period). The correspondence with the load **speed in SI units**¹ is:

$$\text{Load_Speed[SI]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines} \times \text{Tr} \times \text{T}} \times \text{Motor_Speed[U]}$$

where:

No_encoder_lines – is the motor encoder number of lines per revolution

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”.

6.2.6. Brushless motor with sine/cosine encoder on motor

The internal speed units are interpolated encoder counts / (slow loop sampling period). The correspondence with the load speed in SI units is:

For rotary motors:

$$\text{Load_Speed[SI]} = \frac{2 \times \pi}{4 \times \text{Enc_periods} \times \text{Interpolaton} \times \text{Tr} \times \text{T}} \times \text{Motor_Speed[U]}$$

For linear motors:

$$\text{Load_Speed[SI]} = \frac{\text{Encoder_accuracy}}{\text{Interpolaton} \times \text{Tr} \times \text{T}} \times \text{Motor_Speed[U]}$$

where:

Enc_periods – is the rotary encoder number of sine/cosine periods or lines per revolution

Encoder_accuracy – is the linear encoder accuracy in [m] for one sine/cosine period

Interpolation – is the interpolation level inside an encoder period. Its a number power of 2 between 1 and 256. 1 means no interpolation

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”

¹ SI units for speed are [rad/s] for a rotary movement , [m/s] for a linear movement

6.3. Acceleration units

The internal acceleration units are internal position units / (slow loop sampling period)² i.e. the speed variation over one slow loop sampling period.

6.3.1. Brushless / DC brushed motor with quadrature encoder on motor

The internal acceleration units are encoder counts / (slow loop sampling period)². The correspondence with the load **acceleration in SI units** is:

$$\text{Load_Acceleration[SI]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines} \times \text{Tr} \times \text{T}^2} \times \text{Motor_Acceleration[IU]}$$

where:

No_encoder_lines – is the rotary encoder number of lines per revolution

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”

6.3.2. DC brushed motor with quadrature encoder on load and tacho on motor

The internal acceleration units are encoder counts / (slow loop sampling period)². The motor is rotary and the transmission is rotary-to-rotary. The correspondence with the load acceleration in SI units is:

$$\text{Load_Acceleration[SI]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines} \times \text{T}^2} \times \text{Load_Acceleration[IU]}$$

where:

No_encoder_lines – is the encoder number of lines per revolution

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”

6.3.3. Step motor open-loop control. No feedback device

The internal acceleration units are motor μ steps / (slow loop sampling period)². The correspondence with the load **acceleration in SI units** is:

$$\text{Load_Acceleration[SI]} = \frac{2 \times \pi}{\text{No_}\mu\text{steps} \times \text{No_steps} \times \text{Tr} \times \text{T}^2} \times \text{Motor_Acceleration[IU]}$$

where:

No_steps – is the number of motor steps per revolution

No_μsteps – is the number of microsteps per step. You can read/change this value in the “Drive Setup” dialogue from PRO Config.

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”

6.3.4. Step motor open-loop control. Incremental encoder on load

The internal acceleration units are load encoder counts / (slow loop sampling period)². The correspondence with the load acceleration in SI units is:

For rotary-to-rotary transmission:

$$\text{Load_Acceleration[SI]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines} \times T^2} \times \text{Load_Acceleration[IU]}$$

For rotary-to-linear transmission:

$$\text{Load_Acceleration[m/s}^2\text{]} = \frac{\text{Encoder_accuracy}}{T^2} \times \text{Load_Acceleration[IU]}$$

where:

No_encoder_lines – is the rotary encoder number of lines per revolution

Encoder_accuracy – is the linear encoder accuracy i.e. distance in [m] between 2 pulses

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”.

6.3.5. Step motor closed-loop control. Incremental encoder on motor

The internal acceleration units are motor encoder counts / (slow loop sampling period)². The transmission is rotary-to-rotary. The correspondence with the load **acceleration in SI units** is:

$$\text{Load_Acceleration[SI]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines} \times \text{Tr} \times T^2} \times \text{Motor_Acceleration[IU]}$$

where:

No_encoder_lines – is the motor encoder number of lines per revolution

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”.

6.3.6. Brushless motor with sine/cosine encoder on motor

The internal acceleration units are interpolated encoder counts / (slow loop sampling period)². The correspondence with the load **acceleration in SI units**¹ is:

For rotary motors:

$$\text{Load_Acceleratbn[SI]} = \frac{2 \times \pi}{4 \times \text{Enc_periods} \times \text{Interpolaton} \times \text{Tr} \times \text{T}^2} \times \text{Motor_Acceleratbn[IU]}$$

For linear motors:

$$\text{Load_Acceleratbn[SI]} = \frac{\text{Encoder_accuracy}}{\text{Interpolaton} \times \text{Tr} \times \text{T}^2} \times \text{Motor_Acceleratbn[IU]}$$

where:

Enc_periods – is the rotary encoder number of sine/cosine periods or lines per revolution

Encoder_accuracy – is the linear encoder accuracy in [m] for one sine/cosine period

Interpolation – is the interpolation level inside an encoder period. Its a number power of 2 between 1 and 256. 1 means no interpolation

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”

¹ SI units for acceleration are [rad/s²] for a rotary movement, [m/s²] for a linear movement

6.4. Jerk units

The internal jerk units are internal position units / (slow loop sampling period)³ i.e. the acceleration variation over one slow loop sampling period.

6.4.1. Brushless / DC brushed motor with quadrature encoder on motor

The internal jerk units are encoder counts / (slow loop sampling period)³. The correspondence with the load **jerk in SI units**¹ is:

$$\text{Load_Jerk[SI]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines} \times \text{Tr} \times \text{T}^3} \times \text{Motor_Jerk[IU]}$$

where:

No_encoder_lines – is the rotary encoder number of lines per revolution

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”

6.4.2. DC brushed motor with quadrature encoder on load and tacho on motor

The internal jerk units are encoder counts / (slow loop sampling period)³. The motor is rotary and the transmission is rotary-to-rotary. The correspondence with the load jerk in SI units is:

$$\text{Load_Jerk[SI]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines} \times \text{T}^3} \times \text{Load_Jerk[IU]}$$

where:

No_encoder_lines – is the encoder number of lines per revolution

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”

¹ SI units for jerk are [rad/s³] for a rotary movement, [m/s³] for a linear movement

6.4.3. Step motor open-loop control. No feedback device

The internal jerk units are motor μ steps / (slow loop sampling period)³. The correspondence with the load jerk in SI units¹ is:

$$\text{Load_Jerk[SI]} = \frac{2 \times \pi}{\text{No_}\mu\text{steps} \times \text{No_steps} \times \text{Tr} \times \text{T}^3} \times \text{Motor_Jerk[IU]}$$

where:

No_steps – is the number of motor steps per revolution

No_μsteps – is the number of microsteps per step. You can read/change this value in the “Drive Setup” dialogue from PRO Config.

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”

6.4.4. Step motor open-loop control. Incremental encoder on load

The internal jerk units are load encoder counts / (slow loop sampling period)³. The transmission is rotary-to-rotary. The correspondence with the load jerk in SI units is:

$$\text{Load_Jerk[SI]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines} \times \text{T}^3} \times \text{Load_Jerk[IU]}$$

where:

No_encoder_lines – is the rotary encoder number of lines per revolution

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”.

6.4.5. Step motor closed-loop control. Incremental encoder on motor

The internal jerk units are motor encoder counts / (slow loop sampling period)³. The correspondence with the load jerk in SI units is:

$$\text{Load_Jerk[SI]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines} \times \text{Tr} \times \text{T}^3} \times \text{Motor_Jerk[IU]}$$

where:

No_encoder_lines – is the motor encoder number of lines per revolution

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”.

6.4.6. Brushless motor with sine/cosine encoder on motor

¹ SI units for jerk are [rad/s³] for a rotary movement, [m/s³] for a linear movement

The internal jerk units are interpolated encoder counts / (slow loop sampling period)³. The correspondence with the load jerk in SI units is:

For rotary motors:
$$\text{Load_Jerk[SI]} = \frac{2 \times \pi}{4 \times \text{Enc_periods} \times \text{Interpolation} \times \text{Tr} \times \text{T}^3} \times \text{Motor_Jerk[IU]}$$

For linear motors:
$$\text{Load_Jerk[SI]} = \frac{\text{Encoder_accuracy}}{\text{Interpolation} \times \text{Tr} \times \text{T}^3} \times \text{Motor_Jerk[IU]}$$

where:

Enc_periods – is the rotary encoder number of sine/cosine periods or lines per revolution

Encoder_accuracy – is the linear encoder accuracy in [m] for one sine/cosine period

Interpolation – is the interpolation level inside an encoder period. Its a number power of 2 between 1 and 256. 1 means no interpolation

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”

6.5. Current units

The internal current units refer to the motor phase currents. The correspondence with the motor currents in [A] is:

$$\text{Current[A]} = \frac{2 \times I_{\text{peak}}}{65520} \times \text{Current[IU]}$$

where I_{peak} – is the drive peak current expressed in [A]. You can read this value in the “Drive Info” dialogue, which can be opened from the “Drive Setup”.

6.6. Voltage command units

The internal voltage command units refer to the voltages applied on the motor. The significance of the voltage commands as well as the scaling factors, depend on the motor type and control method used.

In case of **brushless motors** driven in **sinusoidal** mode, a field oriented vector control is performed. The voltage command is the amplitude of the sinusoidal phase voltages. In this case, the correspondence with the motor phase voltages in SI units i.e. [V] is:

$$\text{Voltagecommand[V]} = \frac{1.1 \times V_{\text{dc}}}{65534} \times \text{Voltagecommand[IU]}$$

where V_{dc} – is the drive power supply voltage expressed in [V].

In case of **brushless** motors driven in **trapezoidal** mode, the voltage command is the voltage to apply between 2 of the motor phases, according with Hall signals values. In this case, the correspondence with the voltage applied in SI units i.e. [V] is:

$$\text{Voltagecommand[V]} = \frac{V_{\text{dc}}}{32767} \times \text{Voltagecommand[IU]}$$

This correspondence is also available for **DC brushed** motors which have the voltage command internal units as the brushless motors driven in trapezoidal mode.

6.7. Voltage measurement units

The internal voltage measurement units refer to the drive V_{MOT} supply voltage. The correspondence with the supply voltage in [V] is:

$$\text{Voltage_measured[V]} = \frac{V_{\text{dcMaxMeasurable}}}{65520} \times \text{Voltage_measured[IU]}$$

where $V_{\text{dcMaxMeasurable}}$ – is the maximum measurable DC voltage expressed in [V]. You can read this value in the “Drive Info” dialogue, which can be opened from the “Drive Setup”.

Remark: the voltage measurement units occur in the scaling of the over voltage and under voltage protections and the supply voltage measurement

6.8. Time units

The internal time units are expressed in slow loop sampling periods. The correspondence with the time in [s] is:

$$\text{Time}[s] = T \times \text{Time}[IU]$$

where T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”. For example, if $T = 1\text{ms}$, one second = 1000 IU.

6.9. Master position units

When the master position is sent via a communication channel, the master position units depend on the type of position sensor present on the master axis.

6.10. Master speed units

The master speed is computed in internal units (IU) as master position units / slow loop sampling period i.e. the master position variation over one position/speed loop sampling period.

6.11. Motor position units

6.11.1. Brushless / DC brushed motor with quadrature encoder on motor

The internal motor position units are encoder counts. The correspondence with the motor **position in SI units**¹ is:

$$\text{Motor_Position}[SI] = \frac{2 \times \pi}{4 \times \text{No_encoder_lines}} \times \text{Motor_Position}[IU]$$

where:

No_encoder_lines – is the rotary encoder number of lines per revolution

6.11.2. DC brushed motor with quadrature encoder on load and tacho on motor

The motor position is not computed.

¹SI units for motor position are: [rad] for a rotary motor, [m] for a linear motor

6.11.3. Step motor open-loop control. No feedback device

The internal motor position units are motor μ steps. The correspondence with the motor **position in SI units**¹ is:

$$\text{Motor_Positior[SI]} = \frac{2 \times \pi}{\text{No_}\mu\text{steps} \times \text{No_steps}} \times \text{Motor_Positior[IU]}$$

where:

No_steps – is the number of motor steps per revolution

No_μsteps – is the number of microsteps per step. You can read/change this value in the “Drive Setup” dialogue from PRO Config.

6.11.4. Step motor open-loop control. Incremental encoder on load

In open-loop control configurations with incremental encoder on load, the motor position is not computed.

6.11.5. Step motor closed-loop control. Incremental encoder on motor

The internal motor position units are motor encoder counts. The correspondence with the motor position in SI units is:

$$\text{Motor_Positior[SI]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines}} \times \text{Motor_Positior[IU]}$$

where:

No_encoder_lines – is the motor encoder number of lines per revolution

6.11.6. Brushless motor with sine/cosine encoder on motor

The internal motor position units are interpolated encoder counts. The correspondence with the motor position in SI units is:

For rotary motors:

$$\text{Motor_Positior[SI]} = \frac{2 \times \pi}{4 \times \text{Enc_periods} \times \text{Interpolaton}} \times \text{Motor_Positior[IU]}$$

For linear motors:

$$\text{Motor_Positior[SI]} = \frac{\text{Encoder_accuracy}}{\text{Interpolaton}} \times \text{Motor_Positior[IU]}$$

where:

Enc_periods – is the rotary encoder number of sine/cosine periods or lines per revolution

Interpolation – is the interpolation level inside an encoder period. Its a number power of 2 between 1 and 256. 1 means no interpolation

Encoder_accuracy – is the linear encoder accuracy in [m] for one sine/cosine period

6.12. Motor speed units

6.12.1. Brushless / DC brushed motor with quadrature encoder on motor

The internal motor speed units are encoder counts / (slow loop sampling period). The correspondence with the motor **speed in SI units** is:

$$\text{For rotary motors: } \text{Motor_Speed[SI]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines} \times T} \times \text{Motor_Speed[IU]}$$

where:

No_encoder_lines – is the rotary encoder number of lines per revolution

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”

6.12.2. DC brushed motor with quadrature encoder on load and tacho on motor

The internal motor speed units are A/D converter bits. The correspondence with the motor **speed in SI units**¹ is:

$$\text{Motor_Speed[SI]} = \frac{\text{Analogue_Input_Range}}{4096 \times \text{Tacho_gain}} \times \text{Motor_Speed[IU]}$$

where:

Analog_Input_Range – is the range of the drive analog input for feedback, expressed in [V]. You can read this value in the “Drive Info” dialogue, which can be opened from the “Drive Setup”

Tacho_gain – is the tachometer gain expressed in [V/rad/s]

6.12.3. DC brushed motor with tacho on motor

The internal motor speed units are A/D converter bits. The correspondence with the motor speed in SI units is:

$$\text{Motor_Speed[SI]} = \frac{\text{Analogue_Input_Range}}{4096 \times \text{Tacho_gain}} \times \text{Motor_Speed[IU]}$$

where:

Analog_Input_Range – is the range of the drive analog input for feedback, expressed in [V]. You can read this value in the “Drive Info” dialogue, which can be opened from the “Drive Setup”

Tacho_gain – is the tachometer gain expressed in [V/rad/s]

¹ SI units for motor speed are [rad/s] for a rotary motor, [m/s] for a linear motor

6.12.4. Step motor open-loop control. No feedback device or incremental encoder on load

The internal motor speed units are motor μ steps / (slow loop sampling period). The correspondence with the motor **speed in SI units** is:

$$\text{Motor_Speed[SI]} = \frac{2 \times \pi}{\text{No_}\mu\text{steps} \times \text{No_steps} \times T} \times \text{Motor_Speed[IU]}$$

where:

No_steps – is the number of motor steps per revolution

No_ μ steps – is the number of microsteps per step. You can read/change this value in the “Drive Setup” dialogue from PRO Config.

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”

6.12.5. Step motor closed-loop control. Incremental encoder on motor

The internal motor speed units are motor encoder counts / (slow loop sampling period). The correspondence with the load speed in SI units is:

$$\text{Motor_Speed[SI]} = \frac{2 \times \pi}{4 \times \text{No_encoder_lines} \times T} \times \text{Motor_Speed[IU]}$$

where:

No_encoder_lines – is the motor encoder number of lines per revolution

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”.

6.12.6. Brushless motor with sine/cosine encoder on motor

The internal motor speed units are interpolated encoder counts / (slow loop sampling period). The correspondence with the motor speed in SI units is:

For rotary motors:

$$\text{Motor_Speed[SI]} = \frac{2 \times \pi}{4 \times \text{Enc_periods} \times \text{Interpolation} \times T} \times \text{Motor_Speed[IU]}$$

For linear motors:

$$\text{Motor_Speed[SI]} = \frac{\text{Encoder_accuracy}}{\text{Interpolation} \times T} \times \text{Motor_Speed[IU]}$$

where:

Enc_periods – is the rotary encoder number of sine/cosine periods or lines per revolution

Encoder_accuracy – is the linear encoder accuracy in [m] for one sine/cosine period

Interpolation – is the interpolation level inside an encoder period. Its a number power of 2 between 1 and 256. 1 means no interpolation

Tr – transmission ratio between the motor displacement in SI units and load displacement in SI units

T – is the slow loop sampling period expressed in [s]. You can read this value in the “Advanced” dialogue, which can be opened from the “Drive Setup”

7. Memory Map

PRO-Ax0V80A-SA-CAT has 2 types of memory available for user applications: 16K×16 SRAM and up to 16K×16 serial E²ROM.

The SRAM memory is mapped in the address range: C000h to FFFFh. It can be used to download and run a MPL program, to save real-time data acquisitions and to keep the cam tables during run-time.

The E²ROM is mapped in the address range: 4000h to 7FFFh. It is used to keep in a non-volatile memory the MPL programs, the cam tables and the drive setup information.

Remark: *MotionPRO Developer handles automatically the memory allocation for each motion application. The memory map can be accessed and modified from the main folder of each application*

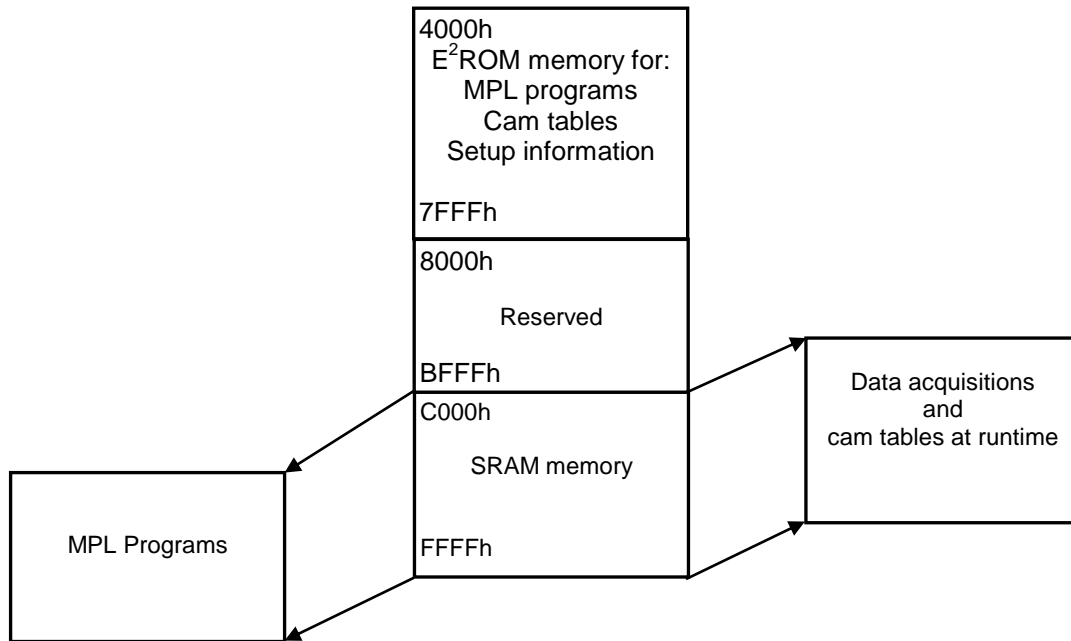


Figure 7.1. PRO-Ax0V80A-SA-CAT Memory Map

