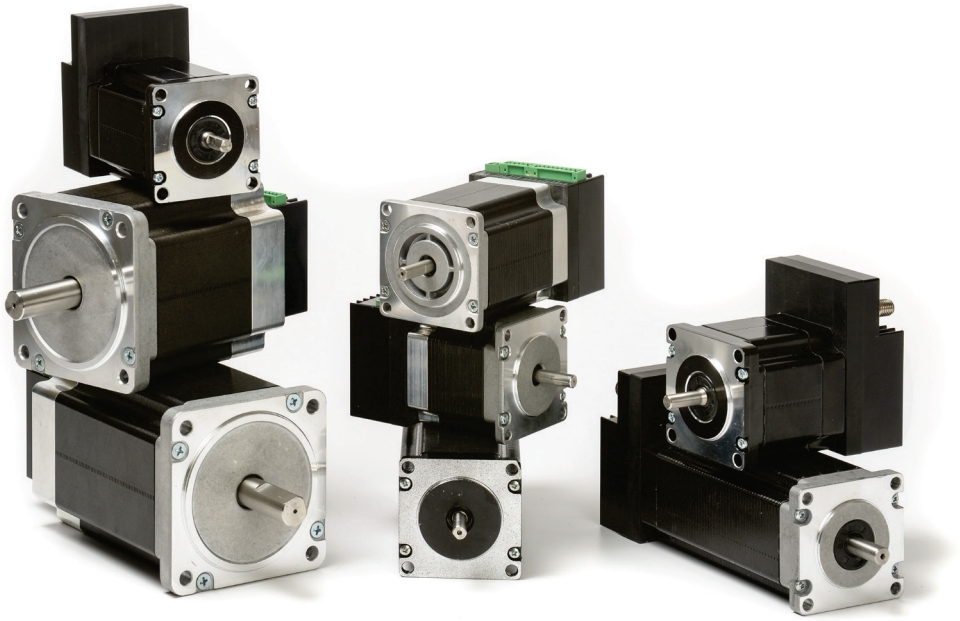
stepIM

Integrated Closed Loop Stepper Motor

User Manual

Manual Revision 1.1



SERVOTRONIX
always in motion™

Revision History

Doc. Rev.	Date	Remarks
1.1	April 2016	Minor corrections
1.0	April 2016	Initial release

Copyright Notice

Disclaimer

This product documentation was accurate and reliable at the time of its release. Servotronix Motion Control Ltd. reserves the right to change the specifications of the product described in this manual without notice at any time.

Trademarks

All marks in this manual are the property of their respective owners.

Contact Information

Servotronix Motion Control Ltd.
21C Yagia Kapayim Street
Petach Tikva 49130, Israel

Tel: +972 (3) 927 3800
Fax: +972 (3) 922 8075

Website: www.servotronix.com

Technical Support

If you need assistance with the installation and configuration of the stepIM drive, contact Servotronix technical support: tech.support@servotronix.com

Contents

1 Introduction

1.1 stepIM Product Series

The stepIM is an integrated closed-loop stepper motor – a single unit comprising motor, drive electronics and position sensor. The electronic control board is attached to the motor and includes control electronics, power stage and magnetic encoder.

The combination of an integrated stepper solution with closed-loop commutation and control provides a number of advantages for machine builders:

- Reduces wiring and assembly time.
- Frees space and reduces heat in the cabinet.
- Reduces machine complexity since fewer components and part numbers, and a smaller cabinet, are used.
- Enhances machine design and flexibility due to the modular structure.
- Provides the high performance of a servo motor at the low cost of a stepper motor.

1.2 About This Manual

This documentation describes the stepIM integrated closed-loop stepper motor.

It provides the information required for installation, configuration and basic operation of the stepIM.

This documentation is intended for persons who are qualified to assemble, commission, and maintain the equipment described herein.

1.3 Documentation Set for stepIM

This manual is part of a documentation set, which consists of the following:

- **stepIM User Manual**. Hardware installation, configuration and operation.
- **stepIM CANopen Reference Manual**. Implementation of CANopen protocol in stepIM.
- **stepIM ServoStudio Reference Manual**. Guide for graphical software interface.

1.4 Ordering Options

IS T – 23M 1 2 CO 1 0 – 0

Integrated Stepper Motor

Type	
T	High torque

Frame Size and Length	
23S	NEMA 23 Short
23M	NEMA 23 Medium
23L	NEMA 23 Long
34M	NEMA 34 Medium
34L	NEMA 34 Long

Shaft	
1	Single flat Frame size 23 only
2	Double flat Frame size 34 only
3	Key upon request. Frame size 23 only
Customized shafts available	

Connector and Degree of Protection	
2	Crimp connectors IP 20
6	M-connectors IP 65 NEMA 23 frames NEMA 34 frames – *upcoming*

Communication	
CO	CANopen
EC	EtherCAT – *upcoming*

Feedback	
1	Standard – 12-bit absolute single turn

Brake	
0	Without brake

Options	
0	**Standard:** Frame size 23, 14–48V 4A Frame size 34, 14–70V 7A – *upcoming*
1	Frame size 34, 14–48V 4A

Available Part Numbers
IST-23S12CO10-0
IST-23M12CO10-0
IST-23L12CO10-0
IST-34M22CO10-1
IST-34L22CO10-1
IST-23S16CO10-0
IST-23M16CO10-0
IST-23L16CO10-0

2 Safety and Standards

Only qualified persons may perform the installation procedures. You do not need to be an expert in motion control to install and operate the stepIM. However, you must have a basic understanding of electronics, computers, mechanics, and safety practices.

The stepIM utilizes hazardous voltages.
Be sure the drive is properly grounded.

Before you install the stepIM, review the safety instructions in this manual.

Failure to follow the safety instructions may result in personal injury or equipment damage.

2.1 Standards Compliance

The stepIM has been tested and according to the following standards.

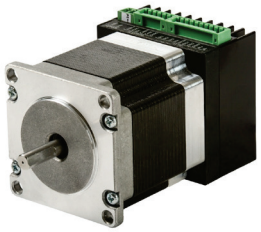
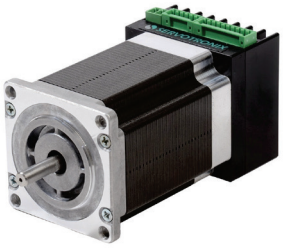
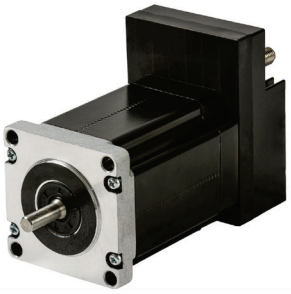
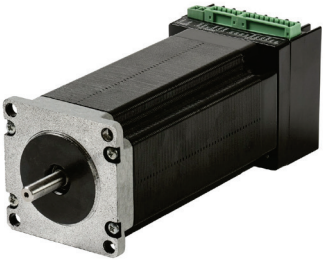
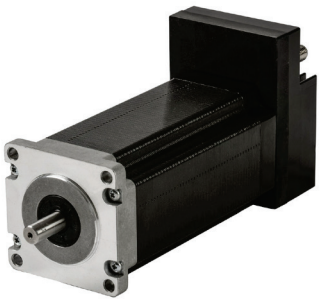
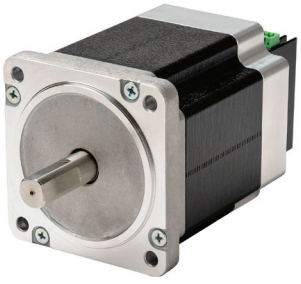
Table 2-1. Standards Compliance

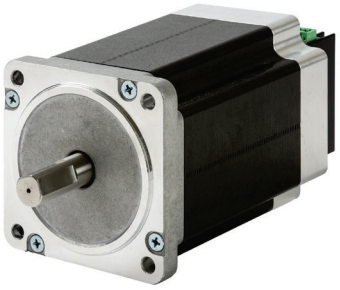
Standard	Directive/Description	Certif. Mark
IEC 61800-3	Electromagnetic Compatibility (EMC) Directive 2004/108/E Adjustable speed electrical power drive systems.	CE
EN 50581:2012	European Regulations 2011/65/EU RoHS (Restriction of Hazardous Substances) Technical documentation required for declaring compliance with the applicable substance restrictions.	CE

IEC International Electrotechnical Commission
EN European Standard (Euro Norm)

3 Specifications

3.1 stepIM Product Models

Model	IP20	IP65
NEMA 23 Short		
NEMA 23 Medium		
NEMA 23 Long		
NEMA 34 Medium		

Model	IP20	IP65
NEMA 34 Long		

3.2 stepIM Architecture

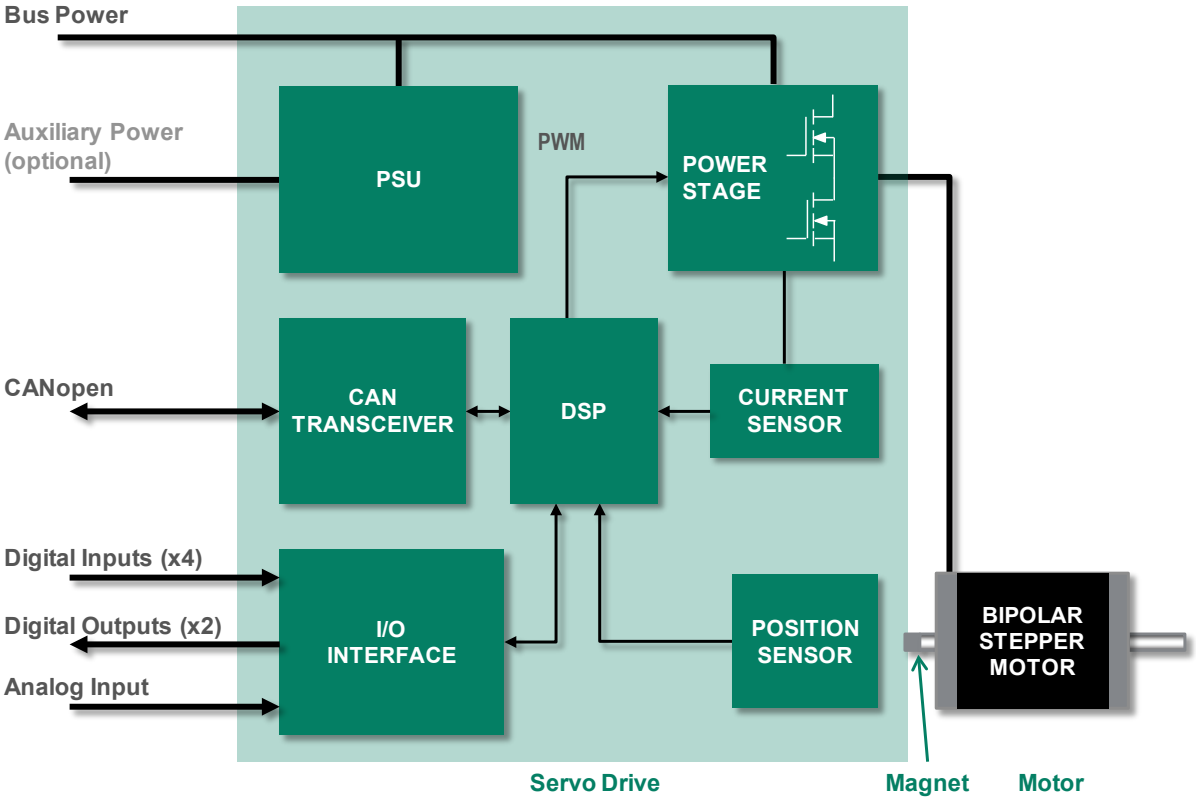


Figure 3-1. stepIM Hardware

3.3 Dimensions and Mounting

stepIM Nema 23

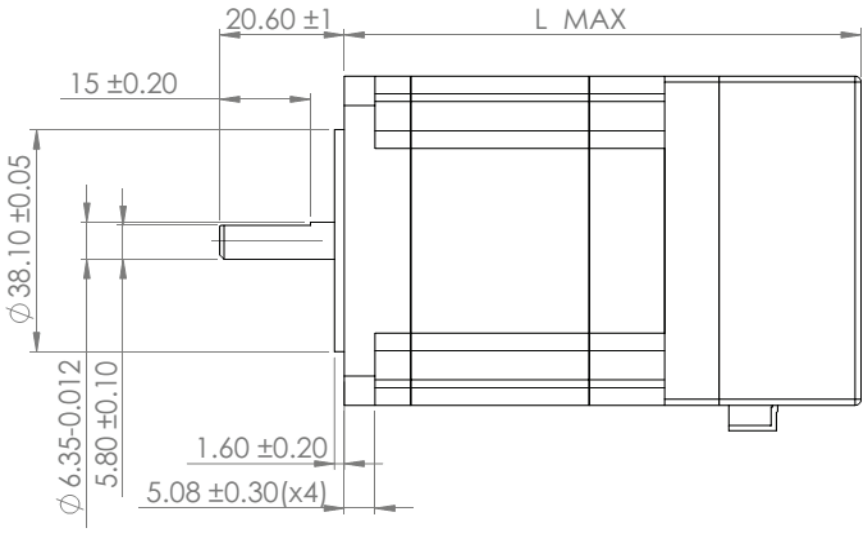


Figure 3-2. stepIM Nema 23 (IP 20) - Dimensions (mm)

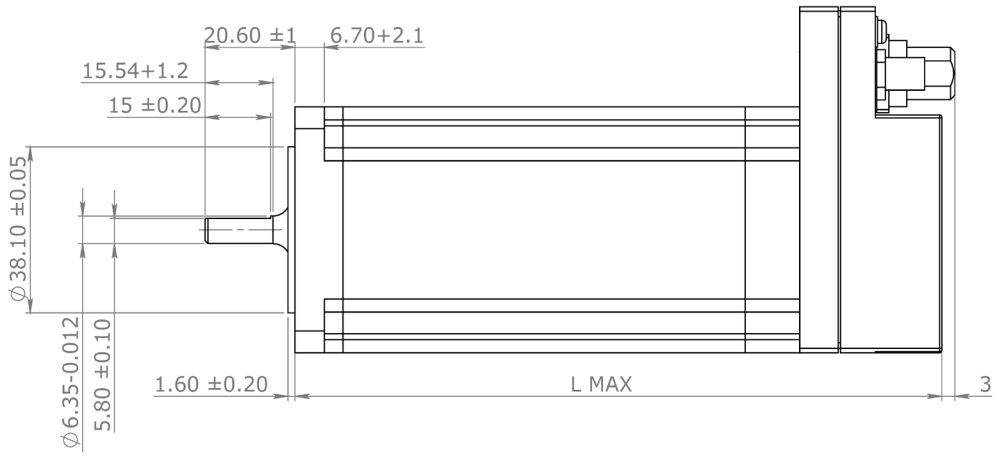


Figure 3-3. stepIM Nema 23 (IP 65) - Dimensions (mm)

Table 3-1. stepIM Nema 23 – Dimensions

Model	L (mm)
IST-23S12CO10-0	86.4
IST-23M12CO10-0	108.4
IST-23L12CO10-0	145.4
IST-23S16CO10-0	91.4
IST-23M16CO10-0	112.4
IST-23L16CO10-0	148.4

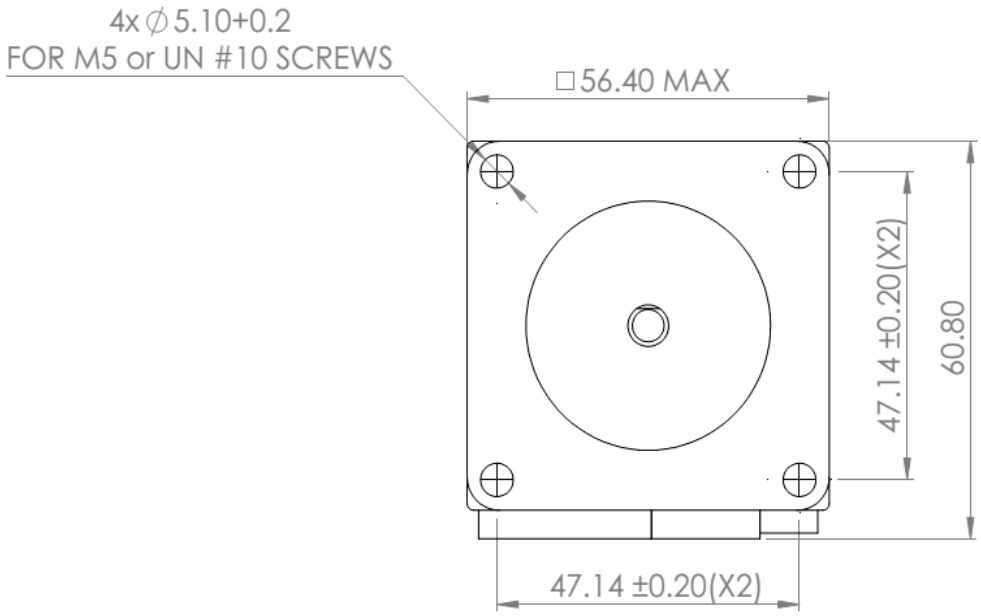


Figure 3-4. stepIM Nema 23 (IP 20) - Mounting (mm)

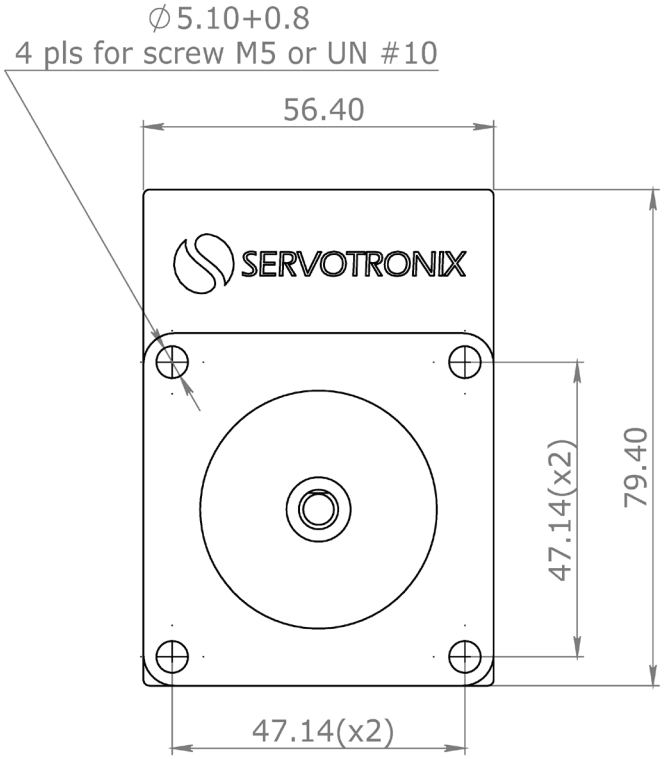


Figure 3-5. stepIM Nema 23 (IP 65) - Mounting (mm)

stepIM Nema 34

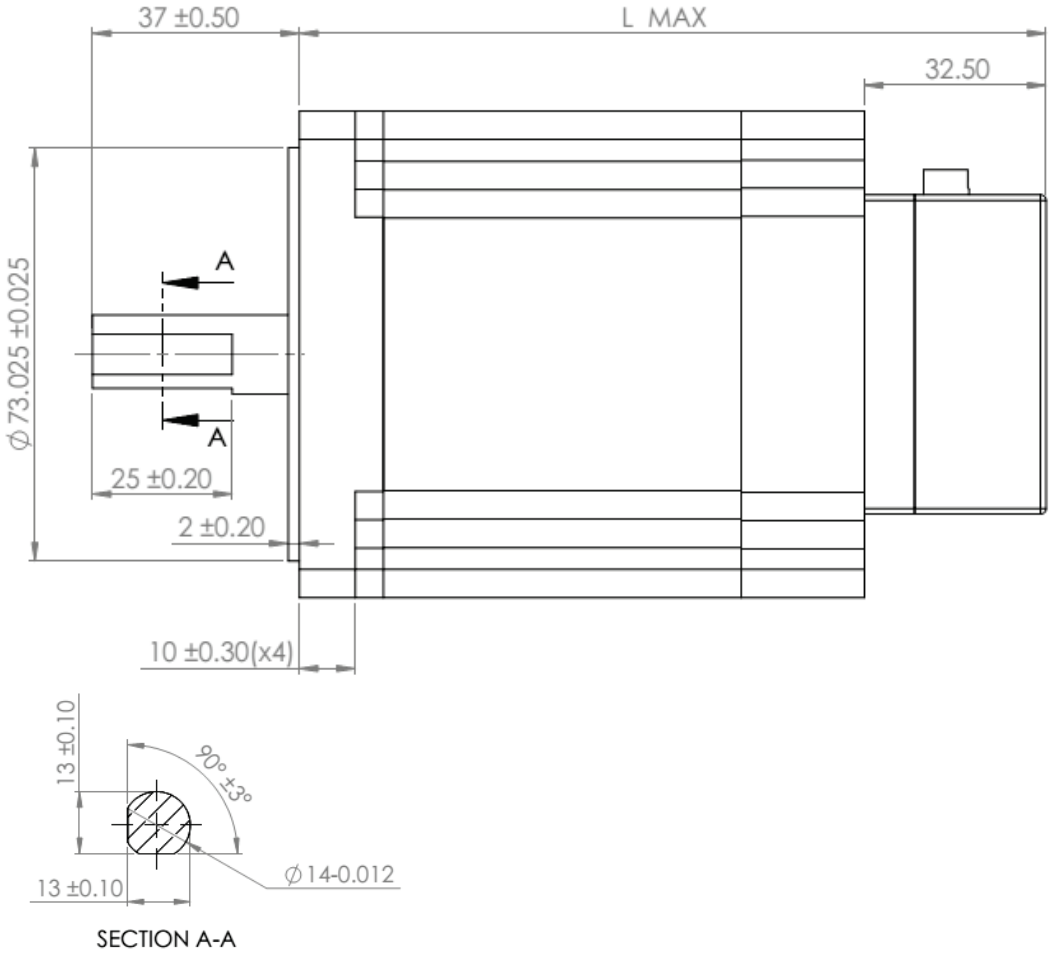


Figure 3-6. stepIM Nema 34 - Dimensions (mm)

Table 3-2. stepIM Nema 34 – Dimensions Data

Model	L (mm)
IST-34M22CO10-1	133.9
IST-34L22CO10-1	163.4

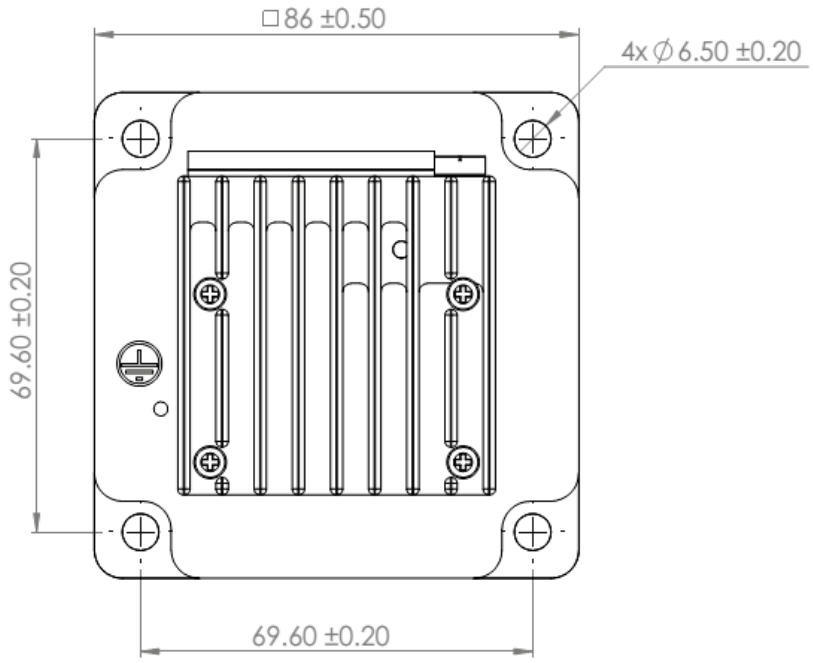


Figure 3-7. stepIM Nema 34 - Mounting (mm)

3.4 Mechanical and Electrical

Table 3-3. Motor and Unit Specifications

Specification		23S...-0	23M...-0	23L...-0	34M...-1	34L...-1*
Input Power, Nominal (±10%)	VDC	14-48				
Input Power, Current Maximum	A	4.5				
Auxiliary Input Power, Nominal (±10%)	VDC	6-24				
Auxiliary Input Power, Maximum	W	1				
Resistance	ohm	0.75	0.75	1.2	0.35	0.49
Inductance	mH	2.2	3.1	6.2	3	5.4
Detent Torque	mNm	40	70	120	250	350
Torque	Nm	1.2	1.8	2.6	3.4	5.4
Overhang Load Limit	kg	0.6	1	1.5	2.7	3.8
Rotor Inertia	g·cm2	260	460	750	1850	2750
Continuous Output Current	A	4.5				
Peak Output Current (application dependent)	A	6.5				
Step Angle	deg	1.8				

Specification		23S...-0	23M...-0	23L...-0	34M...-1	34L...-1*
Magnetic Encoder, Resolution	ppr	4096				
Circuit Loss	W	6				
Weight	kg	0.6	1.0	1.5	2.7	3.8
Connection Hardware Screw Size/Torque	Nm	3	3	3	5.2	5.2
Under-Voltage Trip, Nominal	VDC	Logic				
Over-Voltage Trip	VDC	Logic				

* Supports limited time peak of 50% more torque at 7A

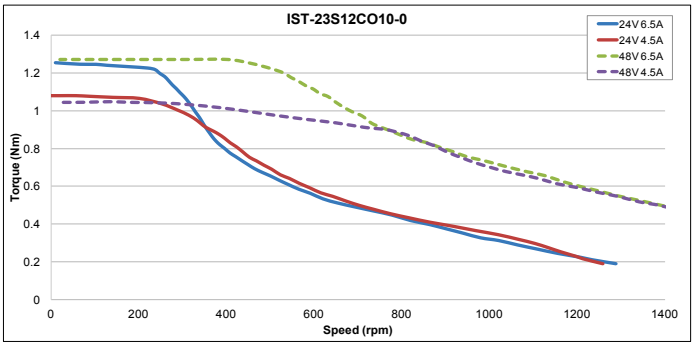


Figure 3-8. Speed/Torque - NEMA 23 Short

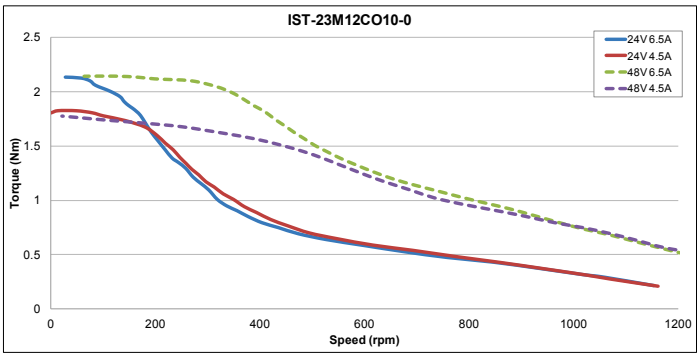


Figure 3-9. Speed/Torque - NEMA 23 Medium

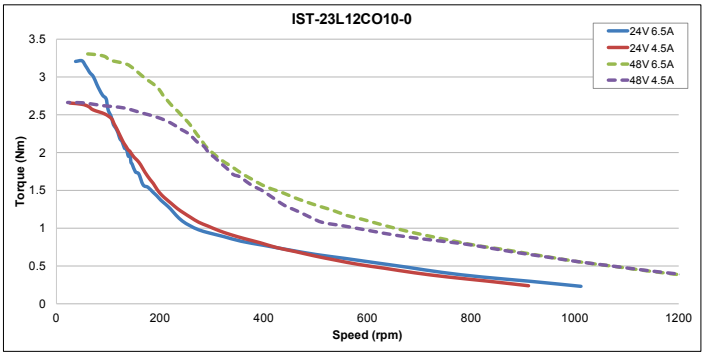


Figure 3-10. Speed/Torque - NEMA 23 Long

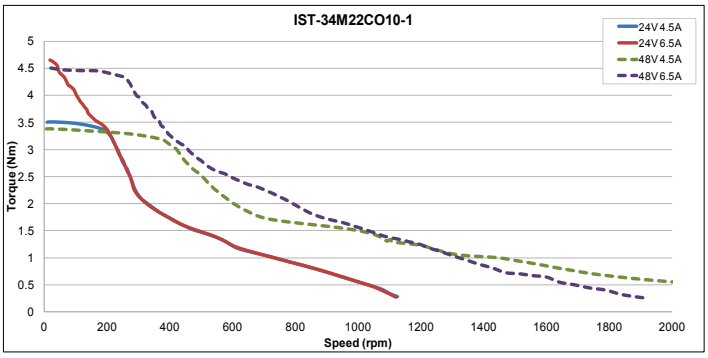


Figure 3-11. Speed/Torque - NEMA 34 Medium

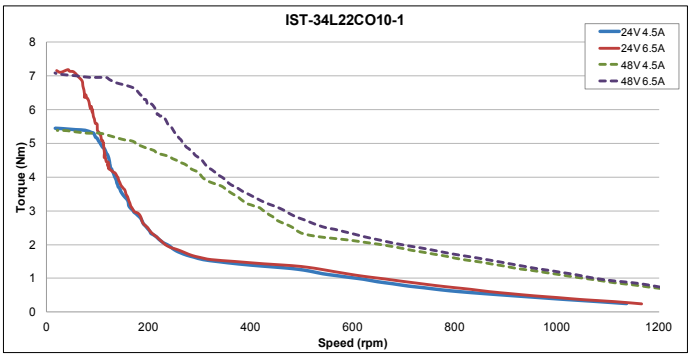


Figure 3-12. Speed/Torque - NEMA 34 Long

Regeneration

During forced deceleration, the motor becomes a generator. The free-wheeling diodes inside the power chip rectify the sinusoidal BEMF voltage into DC current that flows back to the power supply. Not all power supplies can handle regeneration energy. The amount of power also depends on the inertia and speed.

Since the stepIM does not have means of absorbing this energy, a suitable power supply or special accessory is required.

3.5 Power Specifications

Table 3-4. Power Specifications

Type	Units	Value
Supply voltage range	VDC	14 – 48
Auxiliary supply voltage range	VDC	6 – 24
Maximum continuous phase current	A	4.5
Maximum peak phase current *	A	6.5

* Peak current maximum duration: 2 seconds

Auxiliary Power Supply

The auxiliary power supply is optional. The stepIM logic voltage is derived from the bus voltage, and can work with the main bus supply only.

If auxiliary voltage is connected, it will power the stepIM digital components, allowing communication and diagnostics if bus voltage is disconnected.

Warning: Use separate power supplies for bus and auxiliary power supply. Do not connect the VIN+ and VAUX to the same power source.

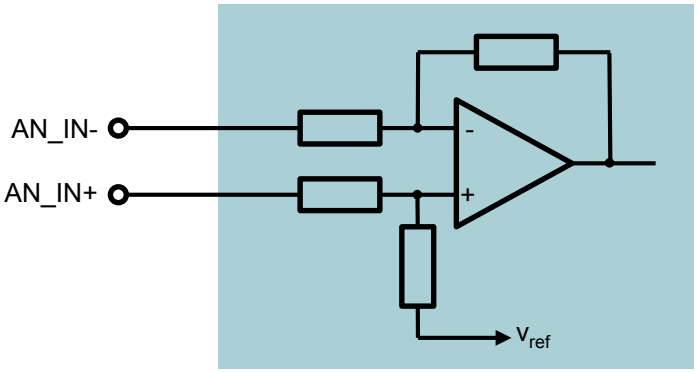


Figure 3-13. Analog Input Interface Scheme

3.6 Control

Table 3-5. Power Specifications

Feature	Specification	
Operation Modes	Selectable	Profile position, Velocity, Profile velocity, Profile torque, Homing, Cyclic synchronous position
Display		Bi-color LED
Software Tools	User Interface	ServoStudio Windows-based application
	Functions	Connection settings, Drive info, Power info, I/O configuration, Motion settings and tuning, Fault history/display
Rotary Units	Position	counts
	Velocity	rpm/100
	Acceleration/ Deceleration	rpm/100/s

3.7 Communication

Table 3-6. Communication Specifications

Feature	Specification	Default value
Baud rate	10 Kbps - 1 Mbps	1 Mbps
CAN ID	1 –127	101
CANopen	CiA 301 application layer and CiA 402 device profile for drives and motion control Heartbeat producer, SDO, PDO (variable mapping)	

3.8 Protection and Environmental Specifications

Table 3-7. Protection and Environmental Specifications

Feature	Specification
Protective Functions	I2T limit, Over-voltage, Under-voltage, Drive over-temperature, Over-speed, Velocity error, Position error, Magnet missing, Power stage fault, PLL lock lost, Position command error, Acceleration / deceleration violation
Standards	CE
Environment	Ambient temperature: Operation 0 – 40°C, Storage 0 – 70°C Heat sink max. temperature: 100°C Motor max. temperature: 120°C
	Humidity: 10 – 90%
	Altitude: If in accordance with specified clearances, per IEC 61800-5-1, the stepIM is rated for use at altitudes up to 2000m
	Vibration: under review
Operating Conditions	Protection class: IP20 and IP65 Pollution degree: 2 as per IEC 60664-1 Do not use where the following are present: corrosive gases, flammable gases, water, oil, chemicals, dust (including iron dust and salts)
Configuration	Flange mounting

Caution!

During continuous motor operation, the motor body and the drive's power stage heat up. The drive shuts down when its power stage heats to 105°C.

The motor can be damaged if its winding temperature exceeds 130°C.

As a precaution, use proper airflow or connect the stepIM to a heat-sink to prevent reaching 100°C on the drive and 120°C on the motor.

3.9 Inputs/Outputs

Table 3-8. I/O Specifications

Feature	Specification	
Analog Input	Signal	±10 V differential
	Functions	User defined
	Input Resolution	12 bit
	Input Resistance	94 kΩ
	Max. Input Frequency	8 kHz
Digital Inputs IP20 x4 IP65 x3	Signal	Configurable opto-isolated. User defined compatibility with sinking output or sourcing output.
	Functions	Homing, limit switch, remote enable, start motion command for profiled position operation mode
	Voltage High Level Input	30 V
	Min. High Level Input V_{IH}	11 V
	Max. Low Level Input V_{IL}	5 V
	Input Resistance	2.2 kΩ
	Max. Input Frequency	1 kHz
	Isolation Voltage	2500 Vrms
	Max. Input Current	According to max. voltage level, input current is not limited, drive limits the input current
	Propagation Delay Time	1 ms
Digital Outputs IP20 x2 IP65 x1	Signal	Configurable open collector. User defined compatibility with either opto-isolated sinking output or sourcing output.
	Functions	Motor speed set, Current, Motor speed set clear, Regen resistor control, Motion completed, In position, Zero speed, Software position limit switch, Active, User selectable.
	Voltage	30 V
	Max. Current	500 mA
	Min. Load Resistance	60 Ω
	Output Voltage (VO)	0.25 V
	Min. Propagation Delay Time	1 ms (may be longer if load current is lower)

4 Wiring

4.1 Wiring Guidelines

Be sure to use conductors according to the interface specifications.

- Use the shortest cable possible.
- Follow the wiring guidelines defined by the connector manufacturers.
- To reduce the effects of EMI, use twisted pairs for the following cables:
  - Power supply
  - CANopen communication

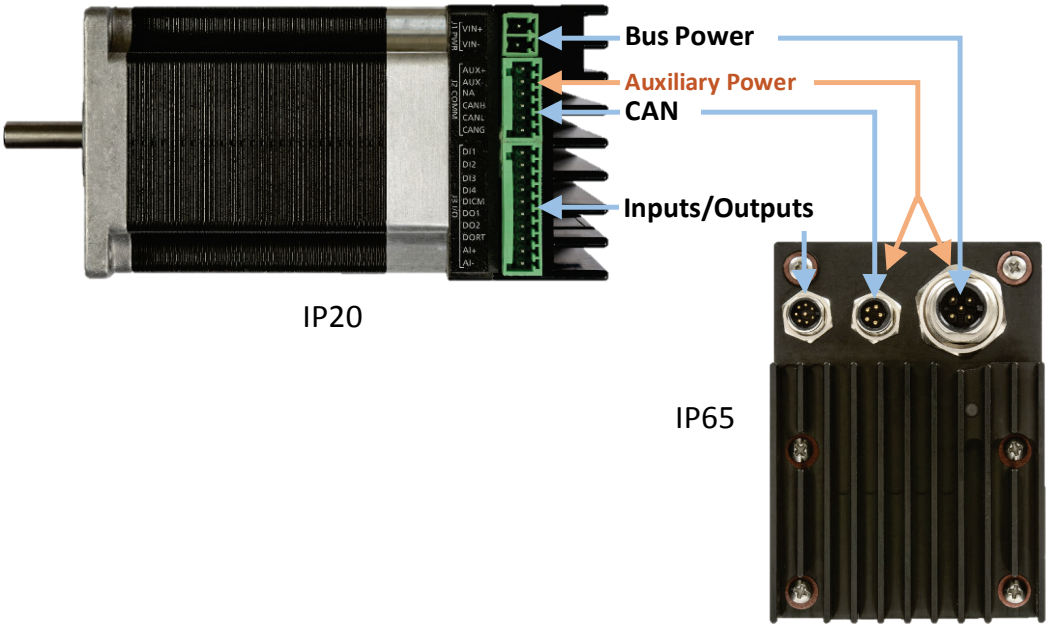


Figure 4-1. stepIM connectors (all models)

- Connect components and cables according to manufacturer specifications.
- Use the shortest cables possible.
- To reduce the effects of EMI, use twisted pairs for the following cables:
  - Power supply
  - CANopen communication

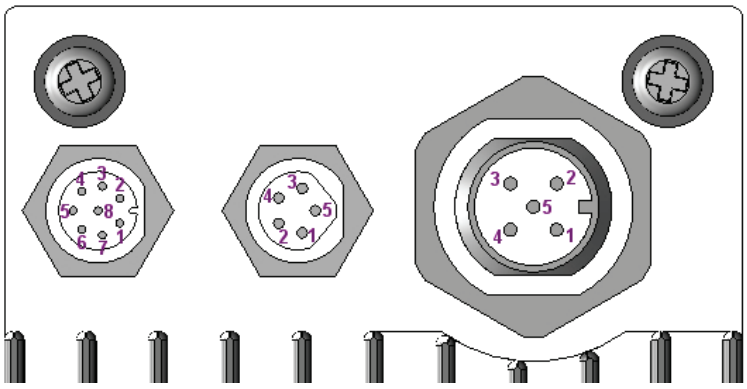


Figure 4-2. stepIM – NEMA23 IP65 Connector Pinouts (all models)

4.2 Power Supply

The stepIM has two separate power supply inputs.

- The **Bus** power supply can be used exclusively to supply power to both the control electronics and the motor.
- The **Auxiliary** power supply is optional. Its supplies power to the control electronics, but not the motor.

Warning: Use separate power sources for the Bus and the Auxiliary power supplies. Do not connect the VIN+ and VAUX to the same power source.

Use twisted pair cables to reduce EMI.

When bus voltage is greater than 32 VDC, use an isolated power supply source for UL compliance.

Warning: No reverse polarity protection on supply input. Incorrect wiring can cause severe damage to the stepIM.

4.2.1 Bus Power – IP20 Models

Table 4-1. Bus Power Interface – IP20 Models

Connector	Phoenix Contact 1803277	
Pitch	3.81 mm	
Pinout	1	VIN
	2	VIN_RET
Mating connector	Phoenix Contact: 1851041 (spring) or Phoenix Contact: 1803578 (screw)	
Wire gauge	16 – 28 AWG	

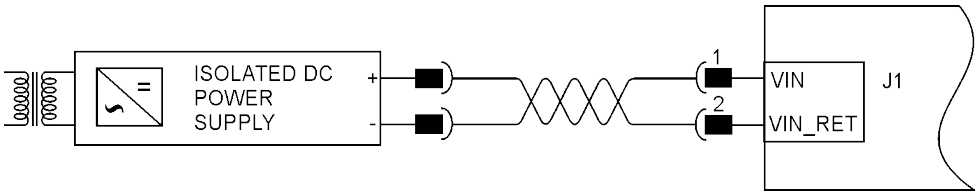


Figure 4-3. Bus Power Supply Wiring

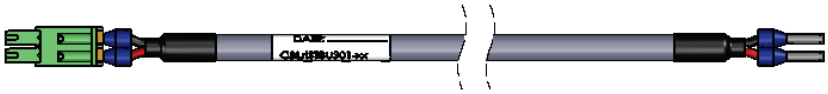


Figure 4-4. Bus Power Cable Example

4.2.2 Bus Power – IP65 Models

Table 4-2. Bus Power Interface – IP65 Models

Connector	M12, 5 pins male, A-code	
Pinout	1	VIN+
	2	VIN-
	3	VAUX+
	4	VAUX-
	5	PE

4.3 CAN

4.3.1 CAN | Auxiliary Power – IP20 Models

Table 4-3. J2 – CAN | Auxiliary Power Interface – IP20 Models

Connector	Phoenix Contact 1881480	
Pitch	2.5 mm	
Pinout	1	VAUX
	2	VAUX_RET
	3	Not connected
	4	CAN_H
	5	CAN_L
	6	CAN_GND
Mating connector	Phoenix Contact: 1881367	
Wire gauge	20 – 28 AWG	

If Auxiliary voltage is connected, it will power the stepIM digital components, enabling communication and diagnostics if Bus voltage is disconnected.

Warning: Use separate power sources for the Bus and the Auxiliary power supplies. Do not connect the VIN+ and VAUX to the same power source.

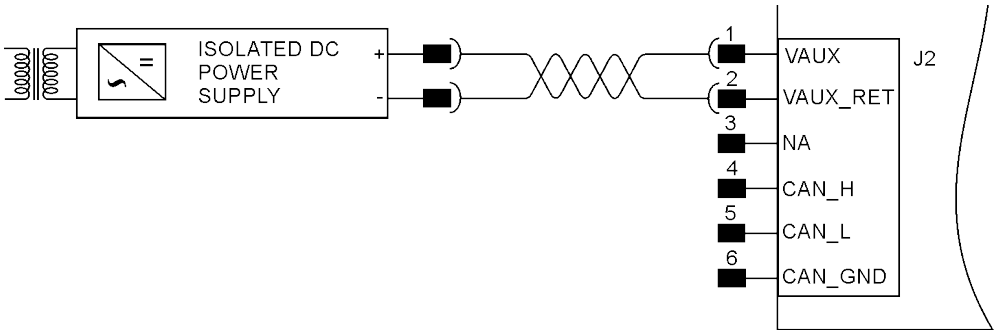


Figure 4-5. Auxiliary Power Supply Wiring

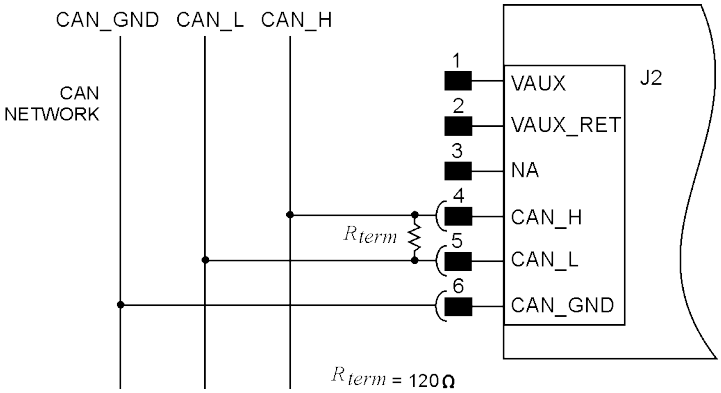


Figure 4-6. CAN Wiring

For CANopen termination, connect an external 120Ω/0.25W resistor between CAN_H and CAN_L terminals (pins 4 and 5 on interface J2).

In addition, use a termination resistor on the host side.

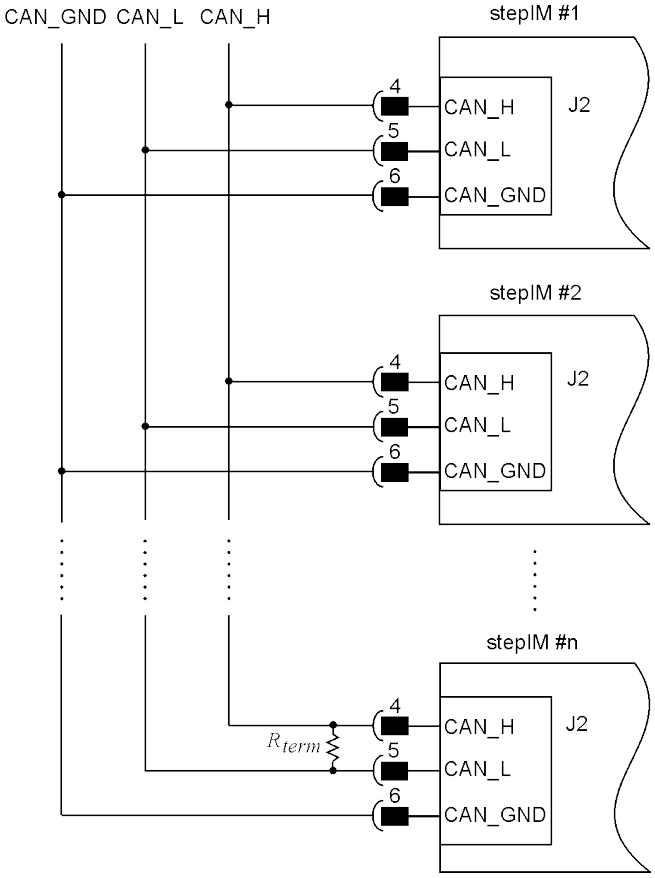


Figure 4-7. CANopen Network Wiring

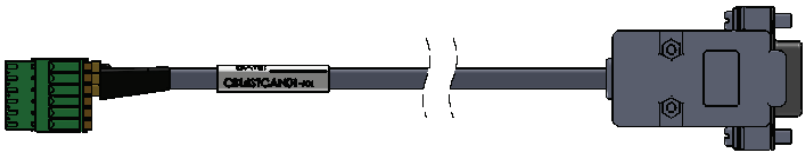


Figure 4-8. CANopen Cable Example

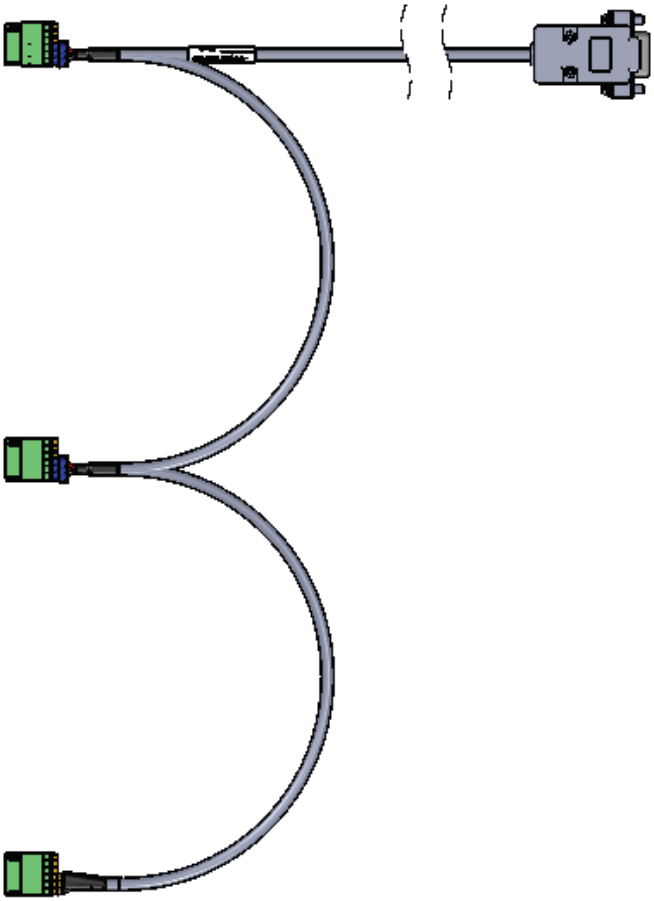


Figure 4-9. CANopen Network Cable Example

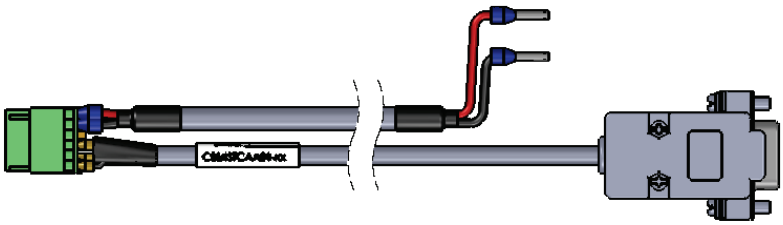


Figure 4-10. One stepIM CAN and VAUX Cable Example

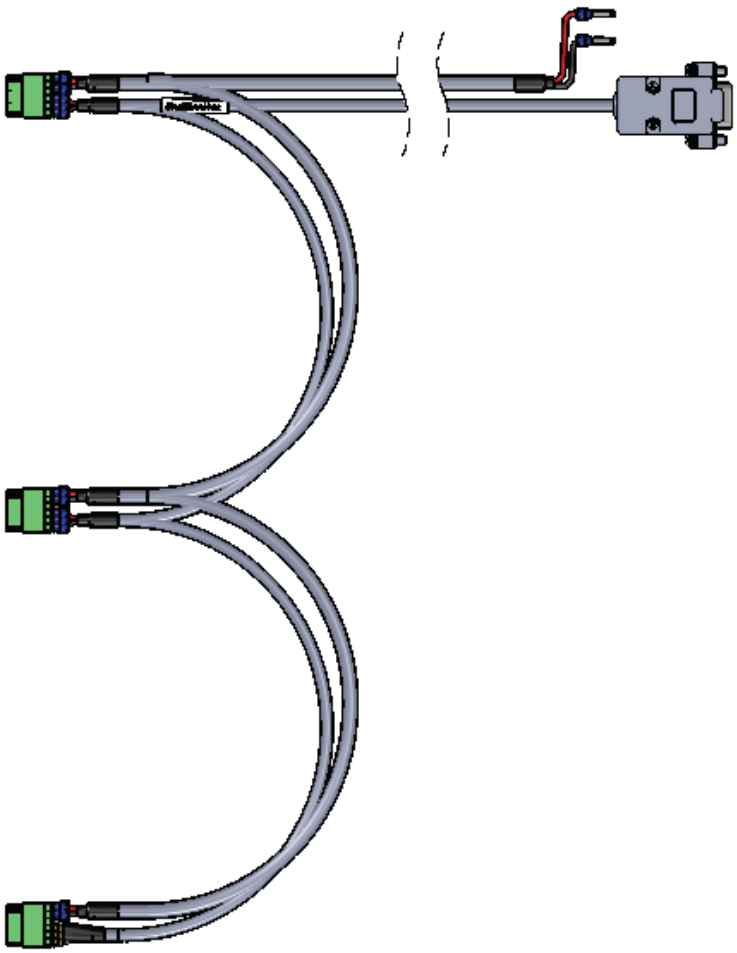


Figure 4-11. Three stepIM CAN and VAUX Cable Example

4.3.2 CAN – IP65 Models

Table 4-4. CAN Interface – IP65 Models

Connector	M8, 5 pins male, B-code	
Pinout	1	PE
	2	VAUX+
	3	GND
	4	CAN_H
	5	CAN_L

4.4 Inputs/Outputs

Table 4-5. Inputs/Outputs Interface – IP20

Connector	Phoenix Contact 1881529	
Pitch	2.5 mm	
Pinout	1	DIN_1
	2	DIN_2
	3	DIN_3
	4	DIN_4
	5	DIN_COM
	6	DOUT_1
	7	DOUT_2
	8	DOUT_RET
	9	AIN+
	10	AIN-
Mating connector	Phoenix Contact: 1881406	
Wire gauge	20 – 28 AWG	

Table 4-6. Inputs/Outputs Interface – IP65

Connector	M8, 8 pins male A-code	
Pinout	1	DI3
	2	DI1
	3	AI-
	4	AI+
	5	DO1
	6	DORT
	7	DICM
	8	DI2

4.4.1 Digital Inputs

The stepIM has 4 digital inputs, with 1 common port that can be used as common ground or common supply.

Follow the I/O specifications for using this interface properly.

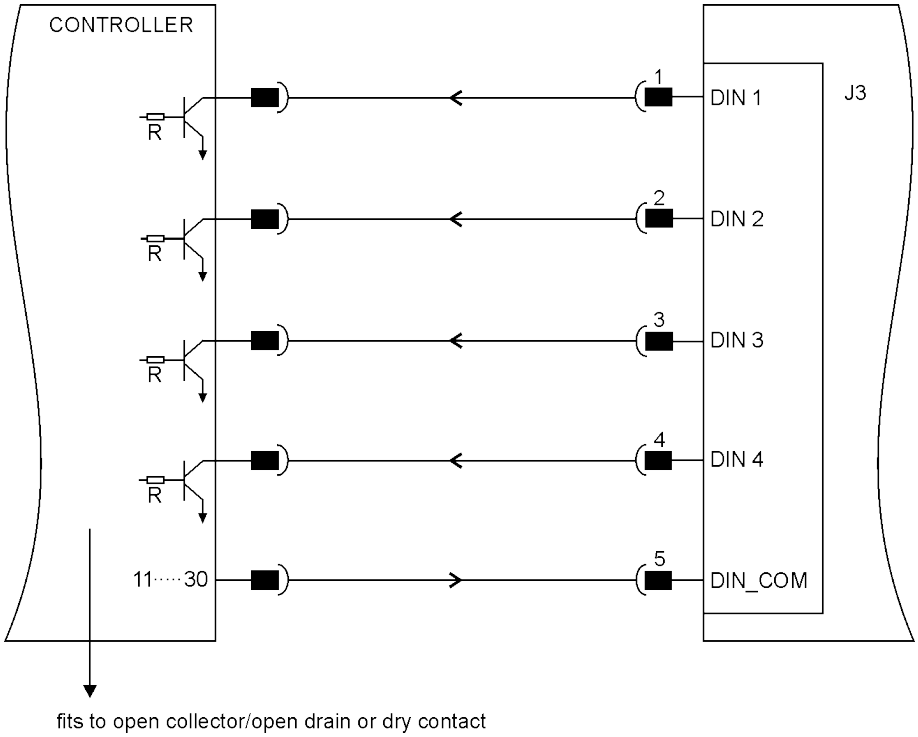


Figure 4-12. Digital Inputs Source Wiring

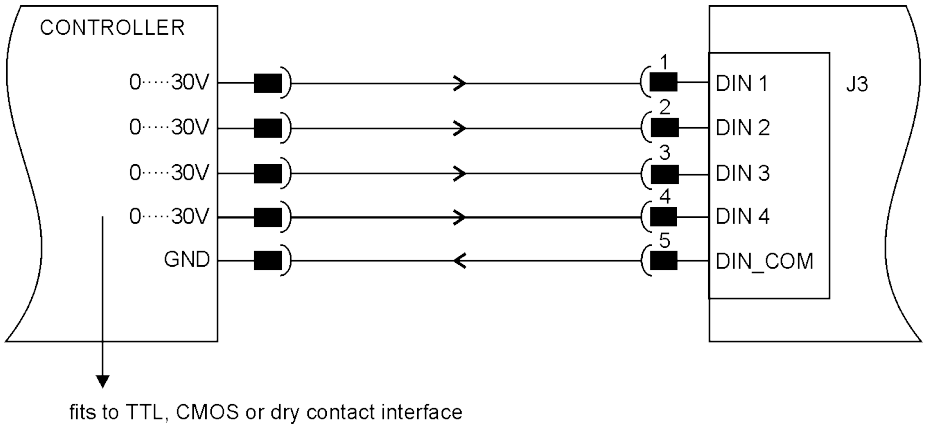


Figure 4-13. Digital Inputs Sink Wiring

4.4.2 Digital Outputs

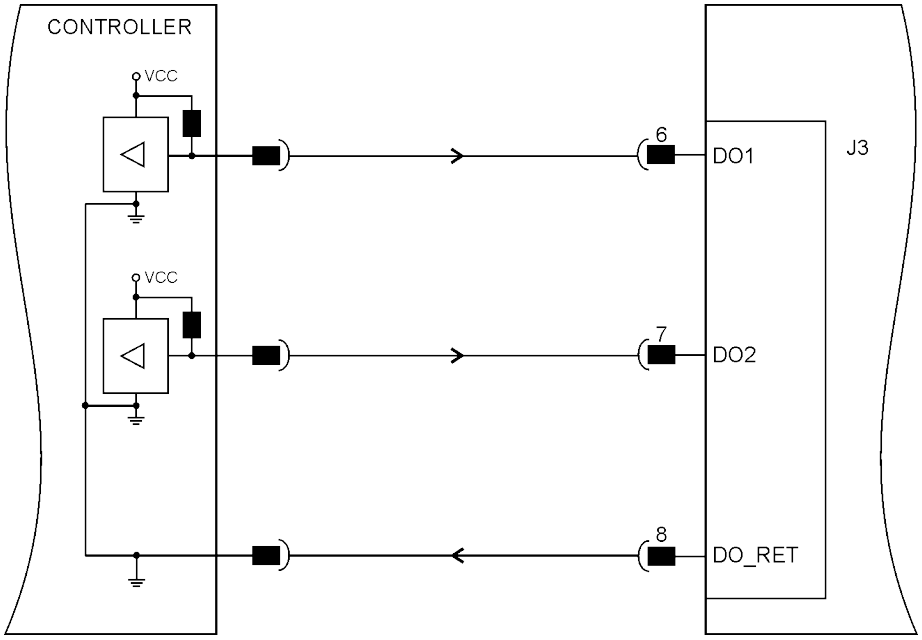


Figure 4-14. Digital Outputs Wiring

4.4.3 Analog Input

The analog input can interface two kinds of analog interfaces – single ended and differential:

- Single-ended: **AN_IN-** is connected to the controller's ground and **AN_IN+** is connected to the controller's single-ended output in the range of ±10V.
- Differential: **AN_IN+** is connected to the controller's positive output and **AN_IN-** is connected to the negative output.

Use a shielded cable and twisted pair for the analog signal.

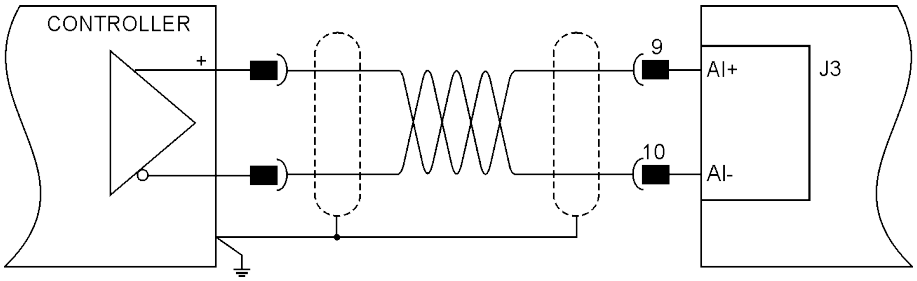


Figure 4-15. Analog Input Differential Wiring

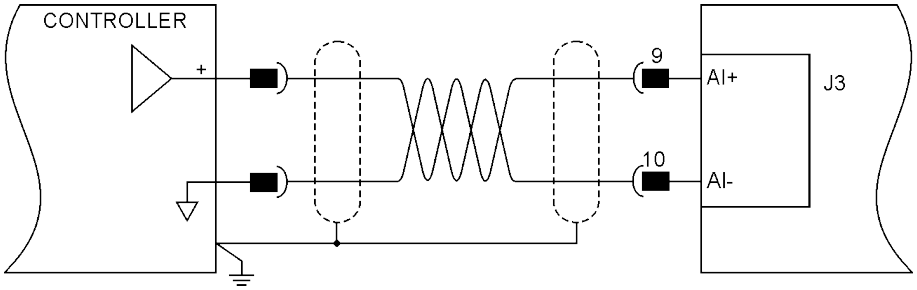


Figure 4-16. Analog Input Single Ended Wiring

5 Installation

5.1 Installation Overview

Perform the following steps to install a stepIM system.

1. Mount the stepIM. Refer to the section Dimensions and Mounting.
1. Make all electrical connections, as required by your application:
   - Digital inputs, digital outputs, analog input (J3)
   - CANopen (J2)
   - DC bus voltage (J1)
   - Optional: Auxiliary power supply (J2)
2. Connect the stepIM to the host PLC.

   Optional: Connect the stepIM to the host computer.
3. Power on the power supply that feeds the stepIM.

   Optional: If using the auxiliary power supply, power it on.
4. Using the host PLC and CANopen protocol, configure and test the stepIM.

   Optional: Download and install ServoStudio software for stepIM on the host computer, and use it to configure and test the stepIM.

5.2 Tools and Connectors

Mating Connectors Kit

- 1 x Phoenix Contact 1851041
- 1 x Phoenix Contact 1881367
- 1 x Phoenix Contact 1881406

5.3 Host Computer System (for Software)

A computer system and software are required if you intend to use ServoStudio, a graphical software interface, useful for commissioning the stepIM, upgrading firmware, and other configuration tasks.:

- 2 GHz CPU
- 1 MB RAM
- 1000 MB available on hard drive (after .net 4 is installed)
- USB port
- Operating system: Windows XP-SP3, or Windows 7
- Recommended screen resolution for ServoStudio is 1280x800.
  Minimal resolution is1024x768.
- ServoStudio software. Download from the Servotronix website or contact Technical Support.

- .Net4 (for details, refer to .NET Framework System Requirements). If .NET 4 is not installed on the computer, ServoStudio will guide you through the installation, but will not install it automatically.

5.4 Software Interface

1. Install ServoStudio software on the host computer.
2. When installation is complete, start ServoStudio from the Windows Start menu or the shortcut on your desktop.

5.5 EDS File

An EDS (electronic data sheet) file for the stepIM must be loaded on the host controller or host computer.

Download the file from the Servotronix website or contact Technical Support.

5.6 USB-CAN Interface

Kvaser Leaf Light is used to connect the stepIM to the host computer to enable communication with ServoStudio software.

1. Before attaching Kvaser hardware to the host computer, install the Kvaser driver.
   - Go the Kvaser website, and download the driver.
   - Install the driver according to the on-screen instructions.
2. Connect the Kvaser Leaf Light HS cable to the USB port on the computer.
   - The Found New Hardware Wizard will detect and complete the driver installation.
   - Confirm hardware installation by opening the Control Panel > **Kvaser Hardware**. Make sure **Kvaser Virtual CAN driver** appears in the Devices tab.
3. Make sure the green light (PWR) on the Kvaser cable is lit.
4. Connect the D9 male connector on the Kvaser cable to the female D9 connector on the RJ45 cable.

5.7 Power Up

After completing the hardware connections and software installations, turn on power to the stepIM.

Note: If logic and bus AC supplies are separate, it is recommended that logic AC be turned on before bus AC.

6 CANopen Communication

6.1 CANopen Termination

For CANopen termination, connect an external 120Ω/0.25W resistor between CAN_H and CAN_L terminals (pins 4 and 5 on connector **J2**).

6.2 CANopen Network Management

CANopen network nodes and states are controlled by network management (NMT) messages. The following diagram shows the network states and transitions.

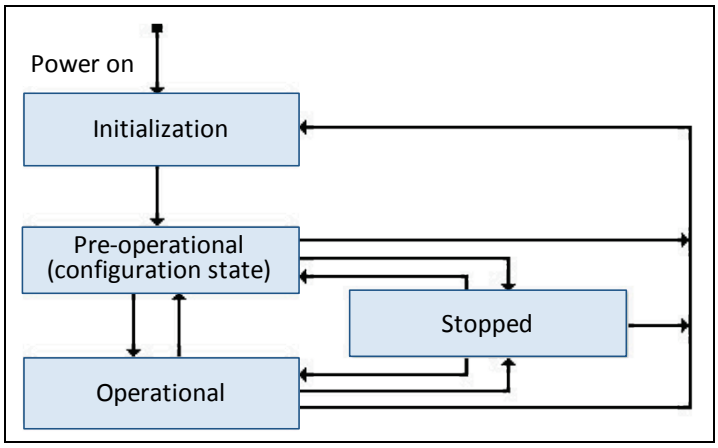


Figure 6-1. CANopen Network Management (NMT) States

- **Pre-operational state**. This state is used primarily for configuration of the CANopen device; therefore, the exchange of process date via PDOs is not possible in this state, and the device cannot be enabled in this state.
- **Operation state**. In this state the transmission of process data via PDOs is possible. This is the only state in which a device can be enabled.
- **Stopped**. A node cannot transmit or receive any other messages in this state. A device cannot be enabled when in this state.

6.3 CANopen Node ID

Note: It is also possible to configure CANopen Node IDs using ServoStudio.

Setting a New Node ID

Default Node ID is 101.

Node IDs are set by object 2F1Bh.

Node ID setting procedure:

1. Write the new Node ID value to object 2F1Bh

2. Save the new Node ID value to EEPROM in object 1010h (Store Parameter Field).
3. Cycle the stepIM power. The new Node ID will be set upon power-up.

Setting New Node IDs for Multiple Drives

When multiple stepIM drives with the same CAN ID are connected to the CANopen network, the following procedure will set a new Node ID for each drive:

1. Set the serial number of the stepIM to be configured in object 2F7Dh (Configured Serial Number).
2. Set the new Node ID in object 2F7Eh (New CAN ID Configuration).
3. Activate the update of the new Node ID within a specific serial number by writing 0 to object 2F7Fh (Operation CAN ID Configuration).
4. Save the new CAN ID in the stepIM EEPROM by writing 1 to object 2F7Fh (Operation CAN ID Configuration).
5. At the next power-up, the new Node ID is in object 2F1Bh (Drive Address).

Note: The serial number of the stepIM is contained in object 1018h (Identity object), sub-index 4; it also appears on the stepIM label. When using the serial number that appears on the label, remove the hyphen (-) from the serial number.

6.4 CANopen Baud Rate

Baud rate can be defined using the following procedure:

1. Set the new baud rate value in object 2F1Fh (CANopen Baud Rate).
2. Save the new baud rate to EEPROM in object 1010h (Store Parameter Field).
3. Cycle the stepIM power. The new baud rate will be set upon power-up.

7 Configuration

7.1 stepIM Control Loop

The following diagram shows the control loop of the stepIM.

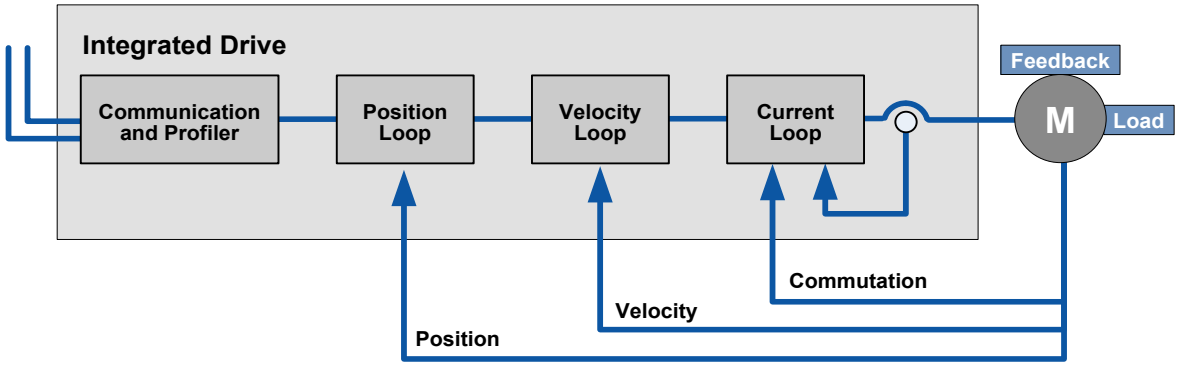


Figure 7-1. stepIM Control Loop

7.2 Drive Configuration

Drive functionality is configured by means of CANopen SDO objects. Some objects allow on-the-fly modification while the motor is in motion and the drive is enabled.

General setup procedure steps:

1. Setting safety limits
2. Setting velocity loop gains
3. Setting position loop gains

7.2.1 Setting Safety Limits

Set drive safety limits parameters, if needed:

Object	Name	Description
Object 2F0Ah	Velocity Over-Speed	Velocity limit that trips the over-speed protection fault.
Object 20EEh	Velocity Limit	Software limit on the velocity command to the velocity loop.
Object 2036h	Peak Current	Current value that trips the over-current protection fault.
Object 6073h	Max Current	Software limit on the current command to the current loop.
Object 607Dh, sub-index 1	Minimum Software Position Limit	Minimum position limit in position loop (activated by object 20AC).
Object 607Dh, sub-index 2	Maximum Software Position Limit	Maximum position limit in position loop (activated by object 20AC).

Object	Name	Description
Object 20AC	Software Position Limit Mode	Enables software position limit protection when the object value is 1.
Object 6065h	Following Error Window	A limit on the position error that trips the position error fault.
Object 6083h	Profile Acceleration	Limits the acceleration in position and velocity modes.
Object 6084h	Profile Deceleration	Limits the deceleration in position and velocity modes.
Object 60C5h	Max. Acceleration	Generates a fault if acceleration or deceleration exceeds this value.

7.2.2 Setting Velocity Loop Gains

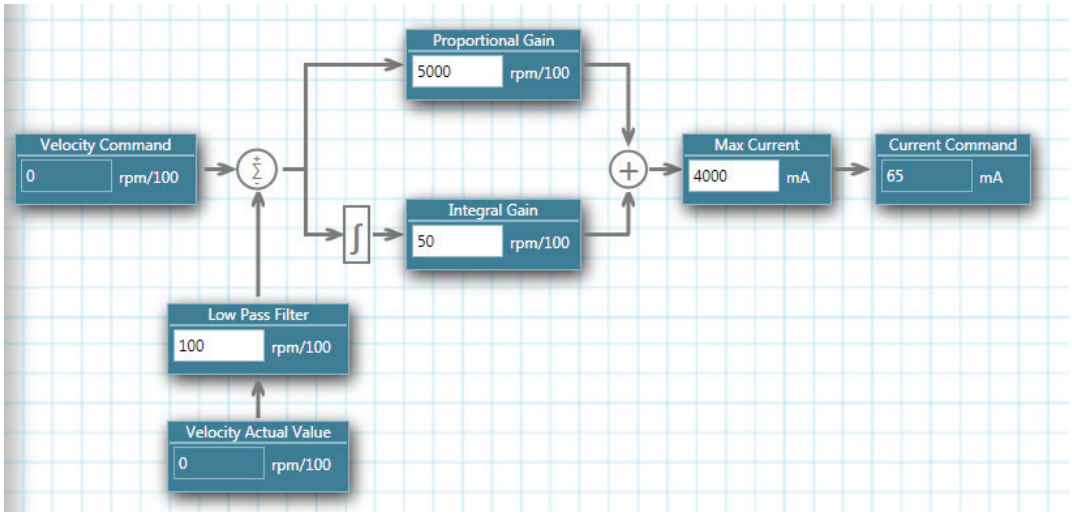


Figure 7-2.

The following parameters can be configured by user:

Object	Description
Object 2026h	Velocity Integral Gain
Object 2027h	Velocity Proportional Gain
Object 20D9h	Velocity Loop Input Filter. The low pass filter cutoff frequency for the velocity loop. It is recommended to set the filter to a cutoff frequency of 100 Hz.
Object 6073h	Maximum Current. The maximum allowed torque-creating current in the motor.

7.2.3 Setting Position Loop Gains

The position loop is a type of PID controller, with acceleration and velocity feedforward.

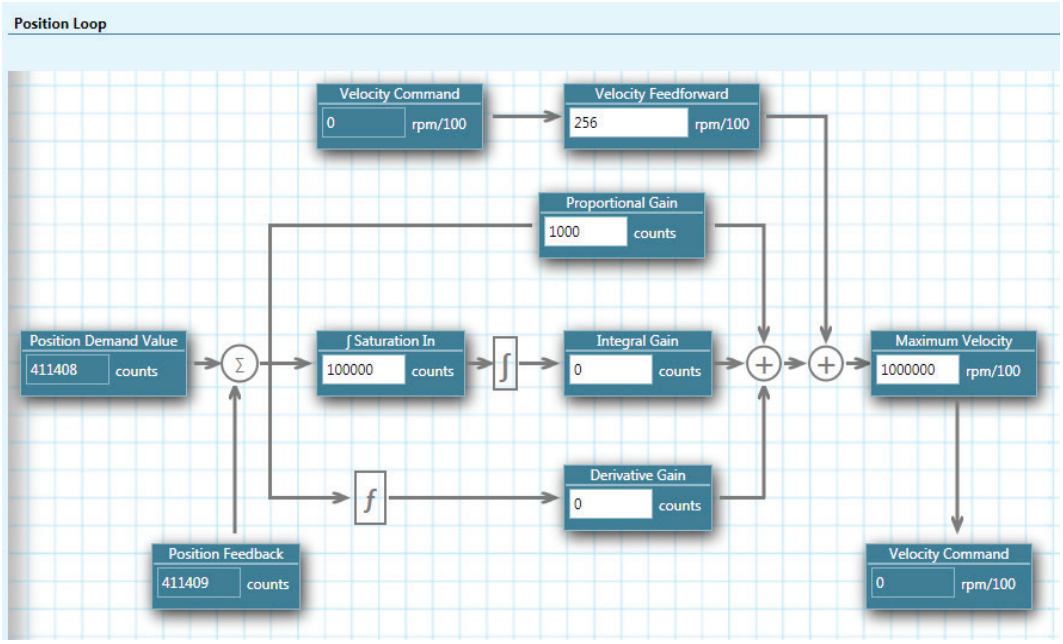


Figure 7-3.

The following parameters can be configured by user:

Object	Description
Object 2022h	Position Loop Proportional Gain
Object 2020h	Position Loop Integral Gain
Object 201Eh	Position Loop Derivative Gain
Object 2023h	Position Loop Velocity Feedforward Gain
Object 2077h	Position Integral Input Saturation
Object 20EEh	Velocity Limit. This object indicates the maximum velocity for a drive and motor.

8 CAN Operation

8.1 Drive State Machine

The drive state machine controls the sequencing of power-up and motion. It also provides the ability to respond to faults and to disable the drive if needed.

The drive device is controlled primarily by object 6040h (Controlword) with fault and status feedback provided by object 6041h (Statusword)

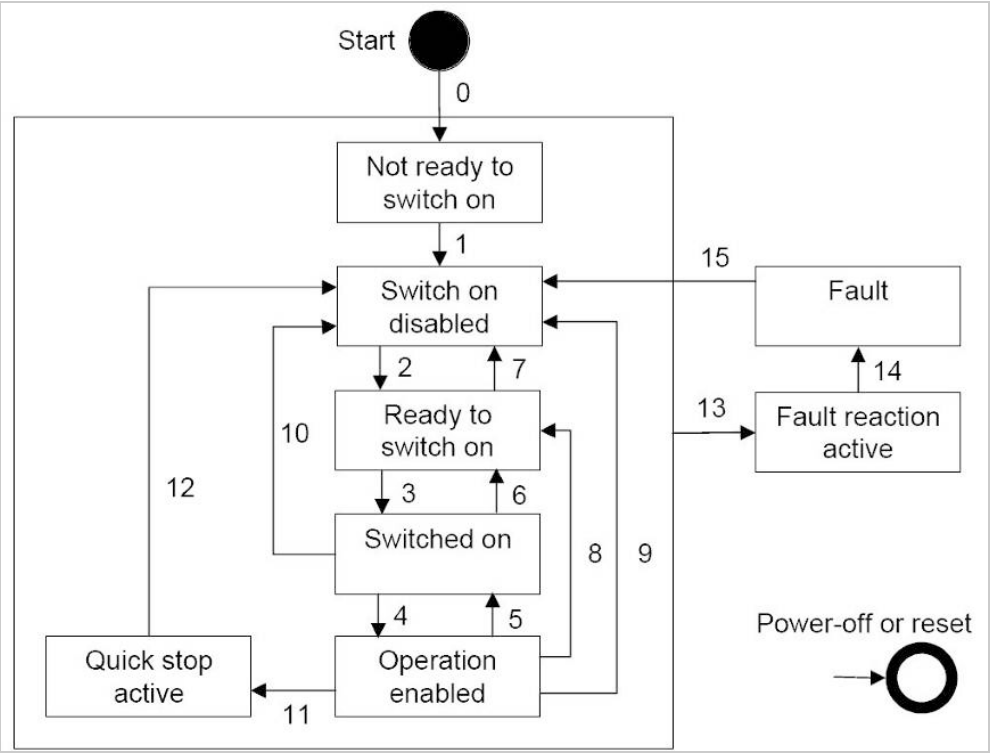


Figure 8-1. Operating States and Transitions

Diagram Notes

Not Ready to Switch On	"Not ready to operate" received from the controller.
Switch On Disabled	Ready to operate. Can read and write parameters. Motion functionality cannot be executed.
Ready to Switch On	Ready to operate. Can read and write parameters. Motion functionality cannot be executed. Bus voltage must be switched on.
Operation Enabled	Drive power stage is enabled. No fault is present. Motion functionality can be executed.
Quick Stop Active	Drive was stopped using controlled stop. Power stage is enabled. Motion functionality cannot be executed.
Fault Reaction Active	A fault has occurred. Drive is in the process of ramping down to 0 velocity.
Fault	A fault has occurred. Power stage is disabled.

The bits in object 6040h (Controlword) are used to initiate state transitions.

The bits in object 6041h (Statusword) indicate the current state of the state machine. The state reflected by object 6041h is updated when the transition has been completed.

Table 8-1. 6040h – Controlword – State Transition Command Bits

		Bit	7	3	2	1	0
		Definition	Fault Reset	Enable Operate	Quick Stop	Enable Voltage	Switch On
Fieldbus Command	**Transition**	**To State**					
Shutdown	2, 6, 8	Ready To Switch On	X	X	1	1	0
Switch On	3	Switched On	X	X	1	1	1
Disable Voltage	7, 9, 10, 12	Switch On Disabled	X	X	X	0	X
Quick Stop	7, 10, 11	Switch On Disabled Quick Stop Active	X	X	0	1	X
Disable Operation	5	Switched On	X	0	1	1	1
Enable Operation	4, 16	Operation Enabled	X	1	1	1	1
Fault Reset	15	Switch On Disabled	0 » 1 (rising edge)	X	X	X	X

Bit	Definition
Bits 4–6, 9	Operation-mode specific
Bit 8	Halt
Bits 10–15	Manufacturer-specific, configured in object 2F87h

Table 8-2. 6041h – Statusword – Bits

Bit	6	5	3	2	1	0
Definition	**Switch On Disabled**	**Quick Stop**	**Fault**	**Operation Enabled**	**Switch On**	**Ready to Switch On**
State						
Not Ready To Switch On	0	X	0	0	0	0
Switch On Disabled	1	X	0	0	0	0
Ready To Switch On	0	1	0	0	0	1
Switched On	0	1	0	0	1	1
Operation Enabled	0	1	0	1	1	1
Quick Stop Active	0	0	0	1	1	1
Fault Reaction Active	0	X	1	1	1	1
Fault	0	X	1	0	0	0

Bit	Definition
Bit 0	Ready to Switch On (1=ready to switch on)
Bit 1	Switched On (1=switched on)
Bit 2	Operation Enabled (1=enabled)
Bit 3	Fault (1=fault)
Bit 4	Voltage Enabled (1=enabled)
Bit 5	Quick Stop (0=drive is responding to a quick stop request)
Bit 6	Switch On Disabled
Bit 7	Warning
Bit 8	Manufacturer-specific
Bit 9	Remote
Bit 10	Target Reached (1=axis motion ended)
Bit 11	Internal Limit Active
Bit 12	Operation-mode specific
Bit 13	Operation-mode specific
Bit 14	Manufacturer-specific
Bit 15	Manufacturer-specific

8.2 Operation Enabled State

1. Before enabling operation, the network node state (NMT) must be Operational. Refer to the section *CANopen Network Management.*
2. First clear any faults by generating a rising edge in bit 7.
3. To enable operation, write the following command sequence to object 6040h (Controlword):

   06h – Shut down

   07h – Switch on

   0Fh – Switch on and enable operation

8.3 Operation Modes

The stepIM supports the following CANopen modes of operation:

1: Profile Position

2: Velocity

3: Profile Velocity

4: Profile Torque

6: Homing

8: Cyclic Synchronous Position

In addition, the stepIM has a manufacturer-specific operation mode:

-5: Scripted Motion

The stepIM must be disabled prior to changing the operation mode, except when switching from Homing to another mode.

Object 6060h is used to set the operation mode. Changed settings become active immediately.

8.4 Profile Position Operation Mode (1)

In the Profile Position operation mode, movements to specified target positions are performed.

The motion ends when one of the following conditions is met:

- Target position reached
- Stop caused by Halt or Quick Stop
- Stop caused by an error

Bits 10 and 12—15 in object 6041h indicate the status of the movement.

Bit	Value
Bit 10: Target reached	0 = Target position not reached 1 = Target position reached
Bit 12: Target value acknowledge	0 = New position possible 1 = New target position accepted
Bit 13: Following error bit	0 = No following error 1 = Following error
Bit 14: Manufacturer-specific	
Bit 15: Manufacturer-specific	

8.4.1 Point-to-Point Movement

To initiate a point-to-point movement command, do the following:

1. Switch to Profile Position operation mode by writing 1 to object 6060h.
2. Enable operation.
3. Set the distance to the target position in object 607Ah.
4. Set the profile velocity in object 6081h.
5. Set the acceleration and the deceleration in objects 6083h and 6084h, respectively.
6. Define the type of movement in object 6040h:
   - Incremental move: set bit 6
   - Absolute move: clear bit 6
7. Define whether motion starts immediately or after previous motion in object 6040h:
   - Immediate: set bit 5
   - After previous: clear bit 5
8. Start motion by generating a rising edge in bit 4 in object 6040h.
9. Up to ten additional motion commands can be issued by using the available buffer.

8.4.2 Begin Motion On Time

The Begin Motion On Time function is applicable in Profile Position operation mode. It enables a synchronized start of motion for several axes at a pre-defined time.

This function uses the object 1013h (High Resolution Time Stamp), extended by drive PLL to allow precise time-keeping between time stamp updates.

The PLL is enabled by setting object 2F82h to 2.

To start motion at a specific time, do the following:

1. Switch to Profile Position operation mode by writing 1 to object 6060h.
2. Enable operation.
3. Set start time in object 2F83h.
4. Determine which bit in object 6040 will be used for the Begin On Time.
5. In object 2F87h, write 2 to the sub-index that correlates to the bit in object 6040h.
6. Set distance in object 607Ah (Target Position).
7. Set profile velocity in object 6081h (Profile Velocity).
8. Set the acceleration and the deceleration in objects 6083h and 6084h, respectively.
9. Define the type of movement in object 6040h:
   - Incremental move: set bit 6
   - Absolute move: clear bit 6
10. Set the selected bit in object 6040h to begin motion at the defined start time.

8.4.3 Backlash Compensation

A backlash compensation distance is applicable in Profile Position operation mode.

stepIM has two types of backlash compensation:

Type 1 Defined by object 2484h.

Prior to starting the first movement after enable, and upon every direction change, the backlash compensation distance is added to the target position.

Upon the first movement after enable, the stepIM will first move the backlash compensation distance in the opposite direction of the move command, and then it will execute the move command.

Type 2 Defined by object 2488h.

At the end of every movement in the direction of the backlash, the backlash compensation distance is added to the target position.

8.5 Velocity Operation Mode (2)

In the Velocity operation mode, a movement is made according to a specified velocity.

To initiate a velocity-controlled movement, do the following:

1. Switch the operation mode to Velocity mode by writing 2 to object 6060h.
2. Enable operation.
3. Start motion by setting the target velocity in object 60FFh.

   If needed, clear bit 8 in object 6040h to start motion.

Target velocity can be changed on-the-fly during motion.

The motion ends when one of the following conditions is met:

- Target velocity is set to 0
- Stop caused by Halt or Quick Stop
- Stop caused by an error

8.6 Profile Velocity Operation Mode (3)

In the Profile Velocity operation mode, the movement profile is defined by velocity and acceleration/decelerations commands.

To initiate a velocity-controlled profile:

1. Switch the operation mode to Profile Velocity mode by writing 3 to object 6060h.
2. Enable operation.
3. Set acceleration/deceleration in object 6083/6084h, respectively.
4. Start motion by setting the target velocity in object 60FFh.

   If needed, clear bit 8 in object 6040h to start motion.

Target velocity can be changed on-the-fly during motion.

The motion ends when one of the following conditions is met:

- Target velocity is set to 0
- Stop caused by Halt or Quick Stop
- Stop caused by an error

Bits 10, 12, 14 and 15 in object 6041h indicate the status of the movement.

Bit	Value
Bit 10 = Target reached	0 = Target velocity not reached 1 = Target velocity reached
Bit 12 = Velocity	0 = Velocity > 0 1 = Velocity = 0
Bit 14 = Manufacturer-specific	
Bit 15 = Manufacturer-specific	

8.7 Profile Torque Operation Mode (4)

In the Profile Torque operation mode, a movement is made with a specified target torque.

To initiate a torque-controlled movement:

1. Switch the operation mode to Profile Torque mode by writing 4 to object 6060h.
2. Enable operation.
3. Start motion by setting the target torque in object 6071h.

   If needed, clear bit 8 in object 6040h to start motion.

Target torque can be changed on-the-fly during motion.

The motion ends when one of the following conditions is met:

- Target torque is set to 0
- Stop caused by Halt or Quick Stop
- Stop caused by an error

8.8 Cyclic Synchronous Position Operation Mode (8)

In the Cyclic Synchronous Position operation mode, a movement to a specified target position is performed according to the value of the synchronous cyclic time.

To initiate a synchronized move command:

1. Change the operation mode to Cyclic Synchronous Position mode by writing 8 to object 6060h.
2. Enable operation.
3. Set the distance to the target position in object 607Ah.
4. Send the sync command (80h) to execute.
5. Repeat steps (3) and (4).
6. PDO may be used to synchronously update the position command in object 607Ah.

The sync command is updated periodically, as defined in object 1006h (Communication Cycle Period).

8.9 Homing (6)

In the Homing operation mode, a movement is performed in order to reach a specific reference point. The point and the path are determined by object 6098h (home type).

In addition to the 35 standard CANopen homing method, the following homing methods are also available:

-4 = homing on hard stop in positive direction with index

-3 = homing on hard stop in negative direction with index

-2 = homing on hard stop in positive direction

-1 = homing on hard stop in negative direction

To initiate homing:

1. Change the operation mode to Homing mode by writing 6 to object 6060h.
2. Enable operation.
3. Set the homing method in object 6098h.
4. Set the homing velocity fast and slow speeds in object 6099h.
5. Set the homing acceleration in object 609Ah.
6. Optional: set the homing offset in object 607Ch.
7. Start motion by generating a rising edge in bit 4 in object 6040h.

Bits 10 and 12—15 in object 6041h indicate the status of the homing.

Bit	Value
Bit 10: Target reached	0 = Homing not completed 1 = Homing completed
Bit 12: Homing attained	1 = Homing successfully completed
Bit 13: Homing error	1 = Homing error

8.10 Scripted Motion Operation Mode (-5)

In the Scripted Motion operation mode, a movement is performed according to a set of path segments, each of which is defined by its own object; the object sub-indices define the controlword, target position, acceleration, deceleration, cruise velocity, delay, number of iterations, and the next path segment to be executed.

The stepIM provides ten objects for defining a set of PTP motion paths.

Object	Description
2F90h – 2F99h	**Path Segments, numbered from 0 to 9**
Sub-index 1	**Target position**
Sub-index 2	**Cruise velocity**
Sub-index 3	**Acceleration**
Sub-index 4	**Deceleration**
Sub-index 5	**Controlword.** A digital input can be used to initiate motion if the corresponding index in object 20E0h is set to 6. This controlword will be executed after the value in sub-indices 1–4 have been set.
Sub-index 6	**Delay**. This is the duration of the pause after the motion has been completed, until the next iteration or the next segment begins.
Sub-index 7	**Number of iterations**
Sub-index 8	**Next segment index**
2F9Ah	**Motion Segment Index** The index of the motion segment that is currently being executed. Writing to this object will cause path execution to jump to the specific segment.

To initiate a scripted motion command, do the following:

1. Configure at least one path segment, starting at segment 0, in object 2F90h.
2. Switch to Scripted Motion operation mode by writing -5 to object 6060h.
3. Enable operation.
4. Start motion by generating a rising edge in bit 4 in object 6040h.

9 I/O Operation

9.1 Digital Input Modes

Object 20E0h is used to set the functionality of the stepIM digital inputs. The value can be read in object 60FDh.

Mode	Functionality
1	General
2	Homing
3	Limit switch clockwise
4	Limit switch counterclockwise
5	Remote enable
6	Start motion command for Profile Position operation mode.
7	Touch probe 1
8	Touch probe 2
9	Motion select 0
10	Motion select 1
11	Motion select 2
12	Motion select 3
13	Motion start
14	Motion stop

9.2 Motion Triggered by Digital Inputs

Digital inputs can be used for starting and stopping motion and for triggering pre-defined motion path segments. Object **20E0h** sub-indices are used to configure digital input motion triggers. Refer to section Digital Input Modes.

Objects **2F90h—2F99h** are used to configure motion path segments. Object 2F90 is motion segment 0, object 2F91h is motion segment 1, and so forth. Refer to section Scripted Motion Operation Mode .

Object **2FC8h** is used to define the functionality of Motion Select; that is, the combination of digital inputs that sets the motion segment/s to be executed.

- If the value of object 2FC8h = **0** (binary value starts motion), the binary value of the combined Motion Select inputs represents one single motion path segment (0, 1, 2, 3, 4, 5, 6 or 7). Up to eight motion path segments can be controlled this way.

  The motion path segment to be executed is determined according to the following formula:

  Segment number = [(Motion Select 3)×8] + [(Motion Select 2)×4] + [(Motion Select 1)×2] + [(Motion Select 0)]

- If the value of object 2FC8h=**1** (input starts motion), each digital input represents one motion path segment (0, 1, 2 or 3). If Motion Select 0 is set to 0, motion path segment 0 is executed, if Motion Select 1 is set to 1, motion segment path 1 is executed, and so on. Up to four motion path segments can be controlled this way.

  When object 2FC8h=1, a rising edge on the trigger input causes the stepIM to attempt to enable the motor and execute the motion path segment.

Digital inputs can serve as motion triggers when the stepIM is operating in Profile Position, Profile Velocity or Scripted Motion modes. Depending on the operation mode in effect, the various parameters defined by the sub-indices will be applied to the motion segments, as shown in the table below.

When operating in Profile Position and Scripted Motion modes, the cruise velocity value (sub-index 2 in objects 2F90h—2F99h) must be positive.

When operating in Profile Velocity mode, the cruise velocity value can be positive, negative or zero for stopping the motion.

Objects 2F90h–2F99h Sub-index	Description	Profile Position	Profile Velocity	Motion Path
Sub-index 1	Target position	Y	N	Y
Sub-index 2	Cruise velocity	Y	Y	Y
Sub-index 3	Acceleration	Y	Y	Y
Sub-index 4	Deceleration	Y	Y	Y
Sub-index 5	Controlword	Y	N	Y
Sub-index 6	Delay	N	N	Y
Sub-index 7	Number of iterations	N	N	Y
Sub-index 8	Next segment index	N	N	Y

9.3 Digital Output Modes

Object 209Ch is used to set the functionality of the stepIM digital output.

The following modes can be selected.

Mode	Functionality	Digital Output is ON While this Condition is in Effect
0	Disabled	
1	Motor Speed Set	Velocity Actual Value (object 606Ch) > Velocity Level 2 for Digital Output Definition (object 20A0h)
2	Current	Current Actual Value (object 6078h) > Current Level 2 for Digital Output Definition (object 209Ah)
3	Reserved	

Mode	Functionality	Digital Output is ON While this Condition is in Effect
4	Motor Speed Set Clear	Velocity Actual Value (object 606Ch) < Velocity Level 1 for Digital Output Definition (object 209Fh) **AND** Velocity Actual Value (object 606Ch) > Velocity Level 2 for Digital Output Definition (object 20A0h)
5	Over-Voltage (for regeneration resistor)	DC Link Circuit Voltage (object 6079h) > Voltage Level for Digital Output Definition (object 2F85h) This condition has hysteresis of ±500 millivolt
6	Motion Completed	Position In Window (20B5)
7	In Position	Position Actual Value (6063h) < Position Window (object 6067h) **AND** Position Actual Value (6063h) > The negative value of Position Window (object 6067h)
8	Zero Speed	Velocity Actual Value (object 606Ch) < Velocity Level 2 for Digital Output Definition (object 20A0h) **AND** Velocity Actual Value (object 606Ch) > - Velocity Level 2 for Digital Output Definition (object 20A0h)
9	Software Position Limit Switch	Position Actual Value (object 6063h) > Position Level 1 for Digital Output Definition (object 209Dh) OR Position Actual Value (object 6063h) < Position Level 2 for Digital Output Definition (object 209Eh)
10	Active	Operation Enabled bit of statusword is On.
11	Reserved	
12	Reserved	
13	User Selectable (XorSetClr)	Set by user in object 60F3h.

10 Recorder

10.1 General

The stepIM has an integral recorder that enables recording of up to four different objects at run time.

The recorder can start the recording on command, by fault or by evaluation of a condition.

10.2 Programming the Recorder

1. The recorder can record up to four different channels.

   Write the values of the CANopen indices to record in sub-indices 2 to 5 of object 2F10h (Recorder Channels).

2. A list of all objects that can be recorded is held in object 2F14h (recordable parameters).

   Set the number of points that will be recorded per channel in object 2F15h. This value must not exceed the maximum number of available points divided by the number of channels.

3. Set the sample time of the recorder in object 2F11h (recorder sample cycle). This value determines the frequency of the recording in multiples of 62.5 µs.

10.3 Triggering the Recorder

The recorder has three triggers types (object 2F12h, sub-index 1: recorder trigger):

- **Immediate**. The recording will start as soon as the recorder has started (object 2F16h: recorder start).
- **By condition object.** The recording will start as soon as the recorder has started (object 2F16h: recorder start) and the condition has been met. The condition consists of three elements:
  - Condition object (object 2F12h, sub-index 2: recorder condition channel index): the object index of the inspected condition.
  - Condition value (object 2F12h, sub-index 3: recorder condition value): the value that must be passed by the condition object in order to trigger the recorder.
  - Condition comparator (object 2F12h, sub-index 4: recorder condition comparator): the passing direction of the value (rising edge or falling edge).

- **By fault.** The recording will start as soon as the recorder has started (object 2F16h: recorder start) and a fault has occurred. The trigger consists of 1 element:
  - Buffer location (object 2F12h, sub-index 5): the position of the trigger in the recorder results buffer (i.e., all points preceding this value were recorded before the condition occurred).

10.4 Starting the Recorder

To start the recorder, write 1 to object 2F16h (Recorder Start).

Writing 0 cancels recording if it is in progress.

10.5 Retrieving the Results

Once the recorder has finished successfully (object 20E6h: record done indicator), the recorder results buffer can be retrieved from object 2F18h (Recorder Results). To retrieve the results:

- Reset the buffer index by writing 1 to object 2F18h, sub-index 1 (reset results index).
- Read object 2F18h sub-index 2 to retrieve each point's value. On each read operation the buffer is automatically advanced to the next point and the next point is retrieved. Repeat reading this object according to the value of 2F13h (Recorder Total Number of Points).

If more than a single channel was recorded, the recorded points are arranged as follows:

<1st channel 1st point>

<2nd channel 1st point>

<3rd channel 1st point>

<1st channel 2nd point>

<2nd channel 2nd point>

<3rd channel 2nd point>

.

.

.

<1st channel last point>

<2nd channel last point>

<3rd channel last point>

11 Firmware Upgrade

11.1 Firmware Upgrade via ServoStudio

11.1.1 Preparation

Contact technical support for the required firmware file.

Important: Before upgrading the firmware, do the following:

1. Backup the drive parameters since parameter settings may be lost during the upgrade. After the upgrade is completed, the parameters can be reloaded/restored.

   To backup parameters from ServoStudio, go the **Backup & Restore** screen, and click the **Backup** button.

2. Read the release note or other documentation supplied with the new firmware.

11.1.2 Upgrade Procedure

1. From the ServoStudio **Drive Information** screen, click **Download Firmware**.

   The **Firmware Upgrade** dialog box opens, and allows you to download the firmware file to the drive over CANopen.

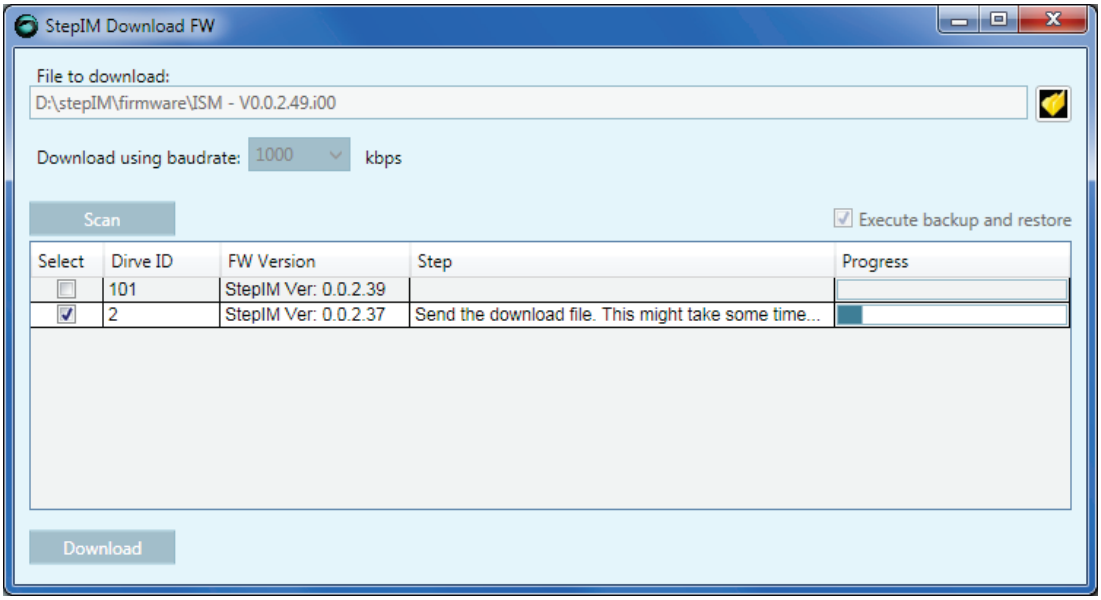


Figure 11-1. Firmware Upgrade Interface

2. Set the address of the stepIM unit to upgrade, browse to and select the firmware file, and press **Download**.

During the firmware upgrade process, the stepIM red LED is steadily-lit.

11.1.3 Resuming Operation

1. Go to the ServoStudio **Drive Information** screen, and check the drive firmware version to verify that the new firmware has been loaded.
2. To restore values to the drive parameters, go the ServoStudio **Backup & Restore** screen, and click the **Restore** button.
3. Check the version release notes, and set any parameters that may have been added to the new version.
4. Save the parameters to the non-volatile parameter memory: either via object 1010h (store parameter field), or click the **Save** button on the ServoStudio toolbar.

11.1.4 Boot Mode

If the firmware loading process has been interrupted, or the firmware is corrupted, the stepIM red LED will continue flashing after power-up.

When the stepIM is in boot mode, firmware upgrade works via ServoStudio as described in the section Upgrade Procedure.

11.2 Firmware Upgrade over CANopen

11.2.1 Firmware Upgrade Protocol

Firmware upgrade over CANopen communication is done by the bootloader. During the boot of the stepIM, the controller can access the bootloader and start the firmware upgrade procedure. The stepIM bootloader supports a minimal set of CANopen objects to enable the firmware upgrade procedure. The bootloader is stored in a protected section of the product's flash memory, and saves the new firmware to its allocated flash sectors.

After power-up, the drive is in boot mode for about 3 seconds. During this brief interval, a CANopen access to the drive prevents it from starting the firmware, and the drive can accept a firmware upgrade.

The following CANopen objects are used in the firmware upgrade procedure:

- 1000h – device type
- 2000h – main program
- 2001h sub-index 1 – flash ready
- 2001h sub-index 2 – erase Flash
- 2002h - unlock Bootloader (only in bootloader versions 2.3 and newer)
- 2800h – domain transfer

Note: Except for object 1000h, these objects are unique to the stepIM bootloader, and do not exist in the product firmware.

11.2.2 Firmware Upgrade Procedure

CAN ID during firmware upgrade:

- Bootloader versions prior to 0.0.2.3: During the boot, the CAN ID of the stepIM is 127.

- Bootloader versions 0.0.2.3 and newer: During the boot, the CAN ID of the stepIM is the last drive address that was set to the drive, or CAN ID 101 if the address has not been changed.

Perform the following steps to upgrade the firmware:

5. Power up the drive.
   - Bootloader versions prior to 2.3: within the first 5 seconds after power-up, access the drive at CAN ID 127 by reading the value of object 1000h.
   - Firmware versions 2.3 and newer: within the first 3 seconds after power-up, write 0x6E65706F ("open") to object 2002h.

6. Erase the flash memory by writing value 1 to object 2001h sub-index 2.
   - Wait 10 to 20 seconds; the drive hangs during the flash erase.

7. Read the value from object 2001h sub-index 1
   - If the value of 2001h sub-index 1 is not 0, this indicates a problem with the flash erase. Repeat the procedure from step 2. If the problem persists, contact technical support.
   - When the value of 2001h sub-index 1 is 0, continue to the next step.

8. Send the firmware file via object 2800h.

9. When the file send is done, read the value of object 2000h.
   - If the value of object 2000h is not 0, this indicates that firmware send failed. Repeat the procedure from step 2. If the problem persists, contact technical support.
   - If the value of object 2000h is 0, the firmware upgrade is successful. Restart the drive.

The following table shows the return values from erase and programming:

Description	Name	Value
Success	SUCCESS	0
Erase and programming errors	CSM_LOCKED	10
	REVID_INVALID	11
	ADDR_INVALID	12
Erase specific errors	NO_SECTOR_SPECIFIED	20
	FAIL_PRECONDITION	21
	FAIL_ERASE	22
	FAIL_COMPACT	23
	FAIL_PRECOMPACT	24
Programming specific errors	FAIL_PROGRAM	30
	FAIL_ZERO_BIT_ERROR	31
	FAIL_VERIFY	40

12 Troubleshooting

12.1 LEDs

The stepIM has green and red LED indicators.

Table 12-1. LED Indicators

Color	Function
Green	**Flashing** – The drive is operational and ready to be enabled. No faults. **ON** – The drive is enabled. No faults.
Red	**ON** – A fault has been detected and needs attention. The LED remains ON until the error is resolved. Flashing – ■ Within the first 3 seconds after power-up: the drive is in boot mode. ■ During firmware running: a fault that was detected no longer exists, but has not yet been cleared.

12.2 Built-in Protection

When a drive fault occurs, the fault is automatically latched and the drive is disabled. Faults must be explicitly cleared before the drive can be enabled.

12.3 Faults

The following table lists the fault (emergency error) codes. When an illegal state occurs in the drive, the stepIM sends the code to the master device as object 603Fh (Error Code).

Whenever the value of 603Fh is not 0, there is a fault in the drive. The CANopen state machine enters Fault mode, and the stepIM cannot be enabled.

Table 12-2. Faults

Error code	Name	Description	Action Required
2214h	Over-current	Hardware or software over-current was detected. The maximum current value is set at object 2036h.	Check the current loop parameters (IGP 2007h, IGI 2006h). Increase maximum current value (object 2036h) or reduce the current saturation value (object 6073h).

Error code	Name	Description	Action Required
2310h	I2T limit	Energy usage is higher than the I2T limit value (object 2034h). The value of I2T value (object 2033h) is greater than the value of I2T limit value (object 2034h).	Check the parameter values in the control loops. Check the demanded velocity (object 6081h), acceleration (object 6083h) and deceleration (object 6084h) and motor load. Increase I2T limit value (object 2034h) if needed, or set it to 0 to disable this functionality.
3110h	Over-voltage	Bus voltage exceeds the value that is set at the over-voltage fault level (object 20A1h).	Check the bus power voltage (object 0790h). Increase the value of object 20A1h.
3120h	Under-voltage	Bus voltage is lower than the value that is set at the under-voltage fault level (object 20CFh).	Check the bus power voltage (object 0790h). Reduce the value of the under-voltage fault level (object 20CFh).
4310h	Over-temperature	The temperature of the drive is higher than 90°C (194°F) or lower than -30°C (-22°F), or the temperature sensor has a malfunction.	Check the drive measured temperature, at object 2044h. Reduce the load on the drive.
5530h	EEPROM fault	One of the following faults occurred: Checksum error while loading parameters. EEPROM read fault: The drive firmware could not access the EEPROM during LOAD (object 1011h). EEPROM write fault: The drive firmware could not access the EEPROM during SAVE (object 1010h).	If the fault occurs when powering-up the stepIM after firmware upgrade, run the SAVE command (object 1010h). Use the command object 1011h to reload the parameters from the EEPROM. Reset the drive, and try again. The EEPROM might be damaged and the drive requires service.
7122h	Reserved	-	-
7310h	Over-speed	Actual speed exceeds the velocity over speed value (object 606Ch).	Check the velocity-loop parameters (VGI 2026h and VGP 2027h). Increase velocity over speed (object 2F0Ah), or reduce the velocity limit (object 20EEh).
8130h	Heartbeat loss	Heartbeat event when heartbeat is not received within the heartbeat consumer time.	Check that the heartbeat consumer and producer times match. Consumer heartbeat setting: object 1016h. Producer heartbeat setting (in master, not in stepIM) setting: object 1017h. To disable heartbeat monitoring, set object 1016h to 0. To disable fault in case of a heartbeat event, set object 6007h to 0.
8400h	Velocity error	The difference between the velocity command and the actual velocity is greater than the value that is set in maximum velocity error (object 2F08h).	Check the parameter values in the control loops. Check the demanded velocity (object 6081h), acceleration (object 6083h) and deceleration (object 6084h). Increase the value of the maximum velocity error (object 2F08h). If needed, or set it to 0 to disable this functionality.

Error code	Name	Description	Action Required
8611h	Position error	The difference between the position command and the actual position is greater than the value that is set in maximum position error (object 6065h)	Check the parameter values in the control loops. Check the demanded velocity (object 6081h), acceleration (object 6083h) and deceleration (object 6084h). Increase the value of the position error maximum (object 6065h) if needed.
F001h	Acceleration / deceleration violation	The motor acceleration or deceleration is greater than the value of the maximum acceleration (object 60C5h).	Check control loops parameters. Check the demanded velocity, acceleration and deceleration. Or, increase the value of the maximum acceleration object 60C5h, or set it to 0 to disable this functionality.
FF00h	Position command error	The difference between two sequential position commands is greater than the value of the maximum position derivative (object 2F0Bh). Position derivative = difference between two sequential position commands. Note: Interpolated mode only.	Check motion controller configuration. Check the demanded velocity (object 6081h), acceleration (object 6083h) and deceleration (object 6084h). Or, increase the value of the maximum position derivative (object 2F0Bh), or set it to 0 to disable this functionality.
FF03h	PLL lock lost	In synchronous motion the drive phase locked loop (PLL) on sync signal has failed.	Check CAN sync cycle parameter (object 60C2h) and increase it if needed.
FF04h	Power stage fault	Power stage generated a fault due to over or under voltage, over current or over temperature.	Check that drive operating condition are within the specification of the product.
FF05h	Encoder failure	Magnetic encoder has failed.	Try to reboot the product. If the fault persists, drive will need to be serviced.
FF06h	Gate drive voltage failure	Power chip low voltage	Check bus power supply. If power supply is OK, contact Technical Support.

stepIM
User Manual