Serial Encoder Guide

P/N 16-01055

Revision 00

August 2026

Table of Contents

1 About This Manual 4
1.1 Overview and Scope 4
1.2 Related Documentation 4
1.3 Comments 4
1.4 Copyrights 4
1.5 Document Validity 4
1.6 Product Warnings 5
2 Introduction 6
3 Absolute Encoders 7
3.1 Supported Absolute Encoders 7
3.2 Firmware Requirements 7
3.3 Emulated Encoder Output 8
3.4 Dual Feedback 9
3.5 SSI Encoders (Type 12) 11
3.5.1 SSI Encoder and Drive Communication 11
3.5.2 SSI Encoder Wiring Example 11
3.5.3 SSI Encoder Configuration Example 12
3.6 EnDat Encoders (Type 11) 14
3.6.1 EnDat Encoder and Drive Communication 14
3.6.2 EnDat 2.2 Encoder Wiring Example 14
3.6.3 EnDat Encoder Configuration Example 15
3.6.4 EnDat 2.1 Wiring Example 16
3.7 EnDat 2.1 Encoder Configuration Example 17
3.8 Absolute A Encoders (Type 14) 19
3.8.1 Absolute A Encoder and Drive Communication 19
3.8.2 Absolute A Encoder Wiring Example 19
3.8.3 Absolute A Configuration Example 20
3.9 Incremental A Encoders (Type 14) 21
3.9.1 Incremental A Encoder and Drive Communication 21
3.9.2 Incremental A Encoder Wiring Example 21
3.9.3 Incremental A Configuration Example 22
3.10 BiSS Encoders (Type 13) 23
3.10.1 BiSS Encoder and Drive Communication 23
3.10.2 BiSS Encoder Wiring Example 23
3.10.3 BiSS Encoder Configuration Example 24
4 Encoder Management 26
4.1 CME Encoder Errors 26
4.1.1 Control Panel 26
4.2 Error Log 27
4.2.1 Troubleshooting Table 27
4.3 Encoder Status Words 28
4.3.1 0x12E Motor Encoder Status 28
4.3.2 0x12F Load Encoder Status 28
4.4 Encoder Status Bits 29
4.4.1 SSI Encoders (Type 12) 29
4.4.2 EnDat Encoders (Type 11) 29
4.4.3 Absolute A Encoders (Type 14) 29

4.4.4 Incremental A Encoders (Type 14) 30
4.4.5 BiSS Encoders (Type 13) 30
4.5 Encoder Option Words 30
4.5.1 0x12A Motor Encoder Options 30
4.5.2 0x12B Load Encoder Options 31
4.5.3 Encoder Option Bits 32
4.5.4 SSI Encoders (Type 12) 32
4.5.5 EnDat Encoders (Type 11) 32
4.5.6 Absolute A Encoders (Type 14) 32
4.5.7 Incremental A Encoders (Type 14) 33
4.5.8 BiSS Encoders (Type 13) 33
4.6 Clearing Faults 33
4.6.1 Clearing Faults in an Absolute Encoder 33
4.7 ASCII Interface 34
4.7.1 Binary Serial Interface 35
4.8 CANopen and EtherCAT 35
4.8.1 MACRO 36
4.9 Clearing Faults in the Drive 36
4.10 ASCII Command using an External Application 37
4.10.1 ASCII Clear Command via CME 37
4.11 Managing Absolute Position 38
4.11.1 Lost Battery 38
4.11.2 Mechanical Slipping 38
4.12 Calibration 39
4.12.1 Manual Calibration 39
4.12.2 Homing Method 39
4.12.3 Writing to an Encoder 39
4.12.4 Using CME 40
4.12.5 Through CANopen/EtherCAT 40

1 About This Manual

1.1 Overview and Scope

This book describes the use and configuration of encoders on Copley Controls' Plus drives.

1.2 Related Documentation

- *CME User Guide*
- *ASCII Programmers Guide*
- *Copley Amplifier Parameter Dictionary*
- *Copley Motion Objects Programmer's Guide*

Links to these publications can be found under Software Documents and Communications Protocols at: http://www.copleycontrols.com.

In addition, the following documents provide more information.

- *CANopen Programmer's Manual*
- *CML Reference Manual*
- *Copley Programming Language (CPL) User Guide*

1.3 Comments

Copley Controls welcomes your comments on this manual. For additional information, refer to the Copley website: https://www.copleycontrols.com.

1.4 Copyrights

No part of this document may be reproduced in any form or by any means, electronic or mechanical, including photocopying, without express written permission of Copley Controls.

The following are the trademarks.

- Xenus®, Accelnet®, and Stepnet® are registered trademarks of Copley Controls.
- CME® is a registered trademark of Copley Controls.
- Windows XP and Windows 7 are trademarks or registered trademarks of the Microsoft Corporation.

1.5 Document Validity

Copley Controls reserves the right to modify their products. The information in this document is subject to change without notice and does not represent a commitment by Copley Controls. This document is for informational purposes only. Copley Controls assumes no responsibility or warranties for the use of this material described in this document.

Revision History

Revision	Date	Comments
00	August 2026	Initial Release-Update information.

1.6 Product Warnings

Observe all relevant state, regional and local safety regulations when installing and using Copley Controls' drives. For safety and to assure compliance with documented system data, only Copley Controls should perform repairs to servo drives.

 **DANGER**	**DANGER: HAZARDOUS VOLTAGES.**
	Exercise caution when installing and adjusting Copley drives. Persons responsible for installing and commissioning Copley drives must be experienced and trained in all aspects of electrical equipment installations. **FAILURE TO HEED THIS WARNING CAN CAUSE EQUIPMENT DAMAGE, INJURY, OR DEATH.**
 **DANGER**	**RISK OF ELECTRIC SHOCK.**
	On some Copley Controls drives, high-voltage circuits are connected to mains power. Refer to hardware documentation. **FAILURE TO HEED THIS WARNING CAN CAUSE EQUIPMENT DAMAGE, INJURY, OR DEATH.**
 **DANGER**	**RISK OF UNEXPECTED MOTION WITH NON-LATCHED FAULTS.**
	After the cause of a non-latched fault is corrected, the drive re-enables the PWM output stage without operator intervention. In this case, motion may re-start unexpectedly. Configure faults as latched unless a specific situation calls for non-latched behavior. When using non-latched faults, be sure to safeguard against unexpected motion. **FAILURE TO HEED THIS WARNING CAN CAUSE EQUIPMENT DAMAGE, INJURY, OR DEATH.**
 **DANGER**	**USING COPLEY SOFTWARE OR SERIAL COMMANDS MAY AFFECT OR SUSPEND CAN OR ETHERCAT OPERATIONS.**
	When operating the drive as a CAN or EtherCAT node over a network, the use of CME or ASCII serial commands may affect network operations in progress. Using such commands to initiate motion may cause network operations to suspend. Network operations may restart unexpectedly when the commanded motion is stopped. **FAILURE TO HEED THIS WARNING CAN CAUSE EQUIPMENT DAMAGE, INJURY, OR DEATH.**
 **DANGER**	**LATCHING AN OUTPUT DOES NOT ELIMINATE THE RISK OF UNEXPECTED MOTION WITH NON-LATCHED FAULTS.**
	Associating a fault with a latched, custom-configured output does not latch the fault itself. After the cause of a non-latched fault is corrected, the drive re-enables without operator intervention. In this case, motion may re-start unexpectedly. For more information, see the *CME User Guide.* When operating the drive as a CAN or DeviceNet node, the use of CME or ASCII serial commands may affect operations in progress. Using such commands to initiate motion may cause network operations to suspend. Operation may restart unexpectedly when the commanded motion is stopped. **FAILURE TO HEED THIS WARNING CAN CAUSE EQUIPMENT DAMAGE, INJURY, OR DEATH.**
 **DANGER**	**USE EQUIPMENT AS DESCRIBED**.
	Operate drives within the specifications provided in the relevant hardware manual or data sheet. **FAILURE TO HEED THIS WARNING CAN CAUSE EQUIPMENT DAMAGE, INJURY, OR DEATH.**
 **WARNING**	**When using this product, adhere to the following requirements.** **•Observe all relevant state, regional and local safety regulations when installing and using this product.** **•Be sure that all wiring complies with the National Electrical Code (NEC) or its national equivalent, and all prevailing local codes.** **FAILURE TO COMPLY WITH THESE WARNINGS CAN CAUSE EQUIPMENT DAMAGE, INJURY, OR DEATH.**

2 Introduction

Absolute Encoders provide position data without referencing to a mechanical home. For every position, a unique binary word is generated.

- In single-turn absolute encoders, words are repeated for every revolution of the encoder's shaft.
- In multi-turn absolute encoders, words are unique for every position, through multiple rotations of the shaft.

For both Single and Multi-turn Encoders, the resolution of a single-turn, and the number of turns, are programmable. The total number of bits is limited to 32.

In this guide, the encoder selection and configuration examples use Copley Controls' CME commissioning software. For more detailed information, refer to the datasheets for encoders and Copley drives posted on the Copley website: https://www.copleycontrols.com.

3 Absolute Encoders

This chapter describes the configuration of absolute encoders using Copley's CME commissioning software. Note that the configuration steps may be continued on subsequent pages.

3.1 Supported Absolute Encoders

The table below is a list of Absolute Encoder types that are supported by the following Copley Controls drives: Accelnet Plus, Xenus Plus, and Accelnet MACRO. These Copley drives may also be used with Incremental Encoders.

Encoder Type	Universal and Manufacturer Standards
SSI	Binary, Gray Code, Renishaw RLS, Baumer SSI, Zettlex Standard
EnDat	Heidenhain 2.1, Heidenhain 2.2, Kollmorgen EnDat
Absolute A	A format, Sanyo Denki Absolute, Panasonic Absolute A, Tamagawa Absolute A
Incremental A	Panasonic Incremental A
BiSS	Hengstler BiSS, Kollmorgen BiSS, Renishaw BiSS C
Hiperface	

3.2 Firmware Requirements

To access and download the updated firmware, refer to the Copley website: https://www.copleycontrols.com.

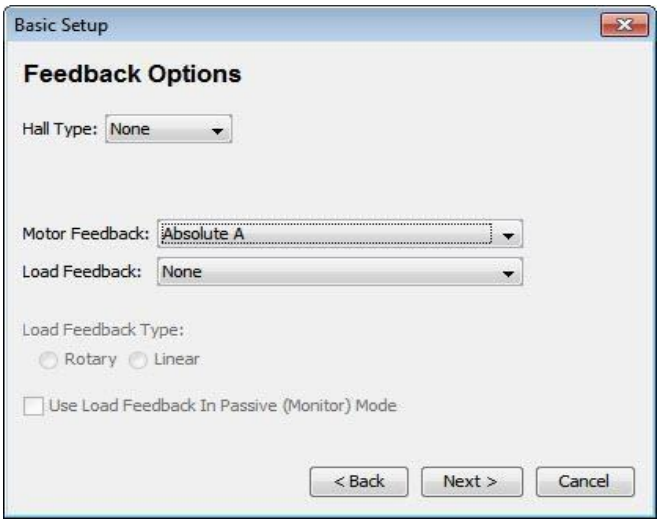
3.3 Emulated Encoder Output

Copley Controls drives can receive position feedback originating from sensors on the motor, through the Primary Feedback channel. They produce emulated digital encoder output using the drive's multi-mode port.

In the following example, it shows the Basic Setup screen, to use to configure the drive with an absolute encoder on the motor and producing an emulated output from the multi-mode port.

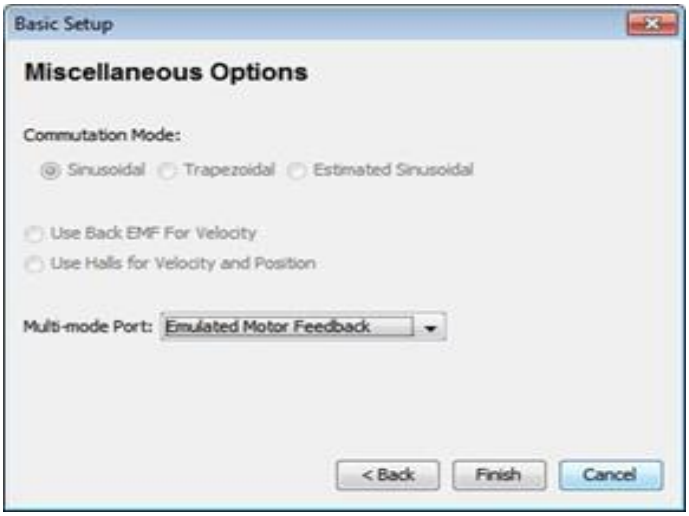
1. In CME, navigate to the *Feedback Options* screen: **Amplifier→Basic Setup→Change Settings→Next**.

   On the *Feedback Options* screen, set **Motor Feedback** to Absolute A, for example.



Basic Setup (Absolute A) Screen

2. From the **Multi-mode Port** drop down, select Emulated Motor Feedback.



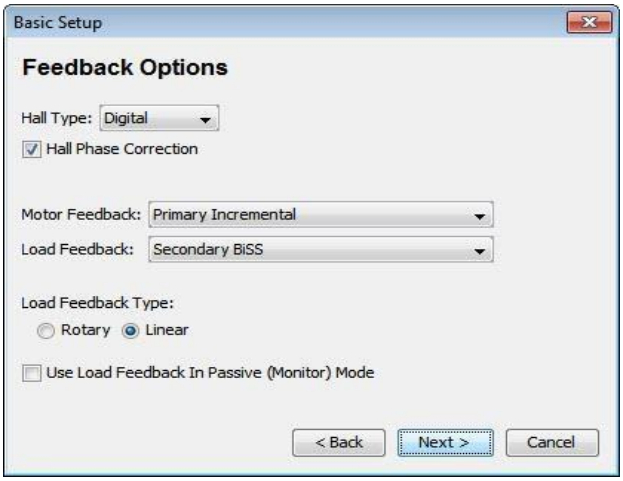
Basic Setup (Emulated Motor Feedback) Screen

3.4 Dual Feedback

Copley Controls drives can receive position feedback from sensors on the motor and the load through the Primary and Secondary feedback channels. In dual loop control mode, an absolute encoder attached to the load provides position loop feedback. The motor encoder or resolver provides velocity loop feedback.

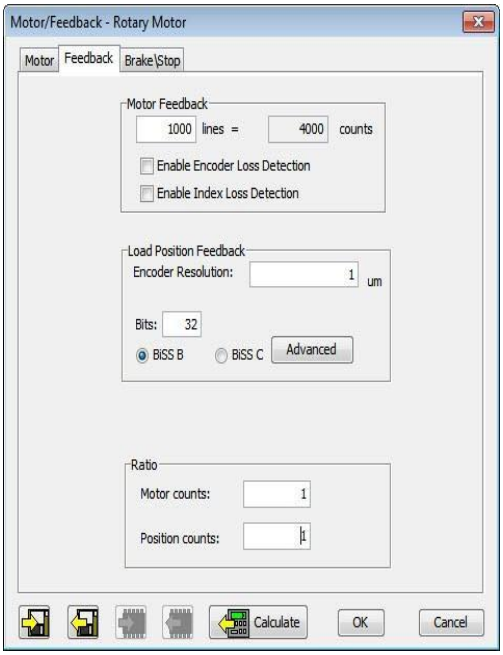
In the dual-feedback setup shown below, the drive receives feedback from an incremental motor encoder through the Primary feedback channel. Position (load) encoder feedback is received through the multi-mode port. There is also an option to use the feedback as simple (passive) monitoring.

1. In CME, navigate to the *Feedback Options* screen: **Amplifier→Basic Setup→Change Settings→Next**. On the CME *Basic Setup Feedback Options* screen, set **Motor Feedback** to Primary Incremental and **Load Feedback** to Secondary BiSS, for example.



Basic Setup (Primary Incremental) Screen

2. Open the *Motor/Feedback* screen *Feedback* tab.



Motor/Feedback-Rotary Motor Screen

To view or change the settings, refer to the following table.

Field	Description
Lines	The number of encoder lines. Quadrature counts/rev will be 4 times the number of lines.
Enable Encoder Loss Detection	An encoder fault will occur when a differential encoder signal, A /A or B /B, is lost.
Enable Index Loss Detection	An encoder fault will occur when a differential index signal, X /X, is lost.
Encoder Resolution	The number of rotary counts/rev or linear distance/count.
Bits	The number of bits used for calculating position. Sign extended to 32 bits.
BiSS B	Mode choice at each start cycle.
BiSS C	Continuous mode.
Motor counts	The position from motor feedback.
Position counts	The position from load feedback.

3.5 SSI Encoders (Type 12)

This section describes the use of Synchronous Serial Interface (SSI) Absolute Encoders used with Copley Controls drives.

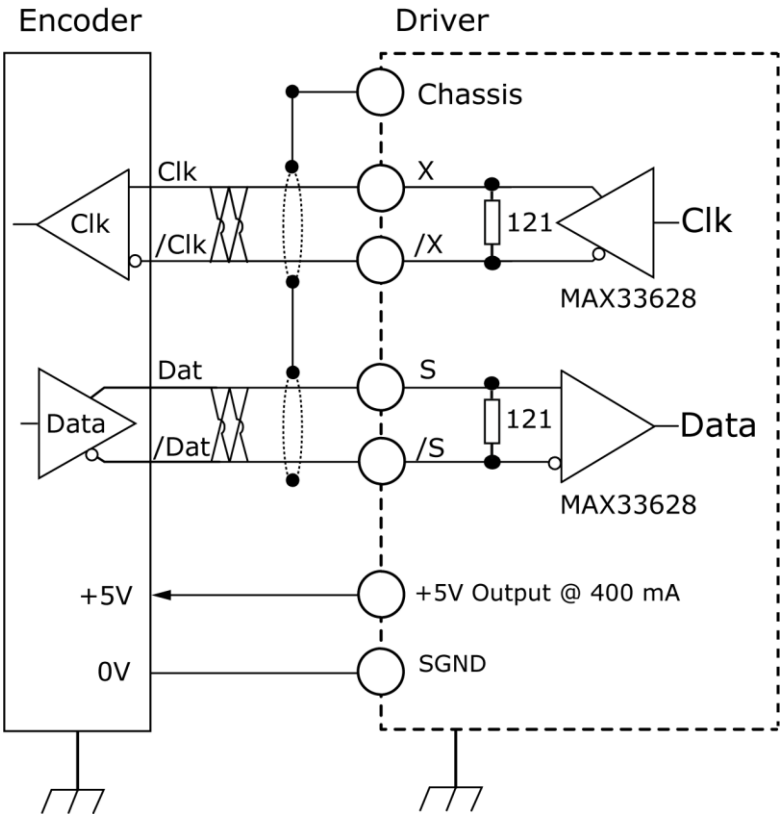
3.5.1 SSI Encoder and Drive Communication

SSI Encoders provide all digital one way communications to the selected Copley drive. A train of clock pulses from the drive initiates the transmission of position data by sending subsequent clock pulses. The clock and data are wired as differential pairs.

An SSI Encoder should have a clock frequency in the range of 1 to 4 MHz. This clock frequency allows all the data to be transferred at a speed that is sufficient for the position loop update rate.

3.5.2 SSI Encoder Wiring Example

The following diagram shows the typical SSI Encoder wiring.
For information on the detailed wiring specifications, refer to the datasheets. The datasheets are posted on the Copley website: https://www.copleycontrols.com.

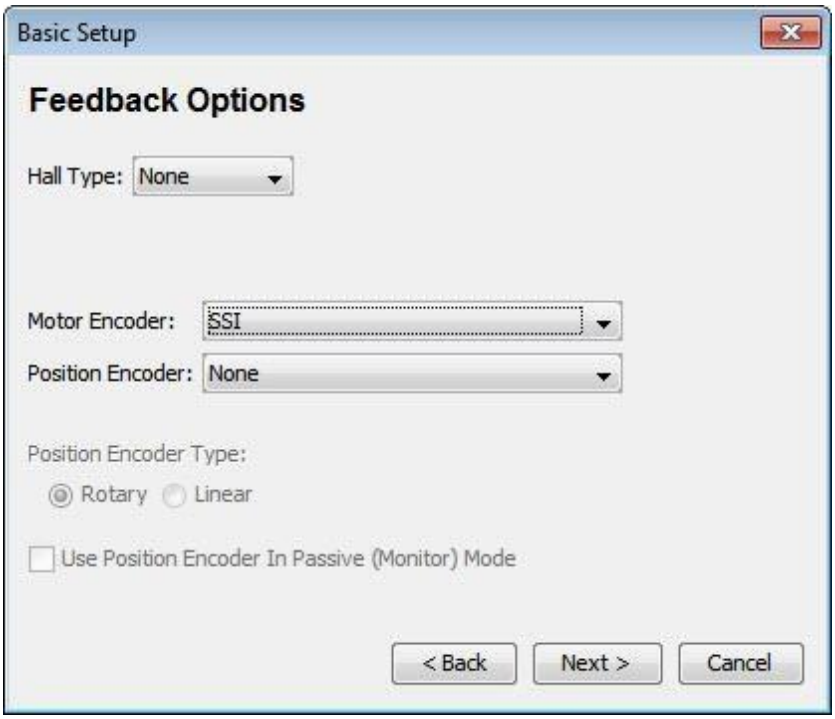


SSI Encoder Wiring Diagram

3.5.3 SSI Encoder Configuration Example

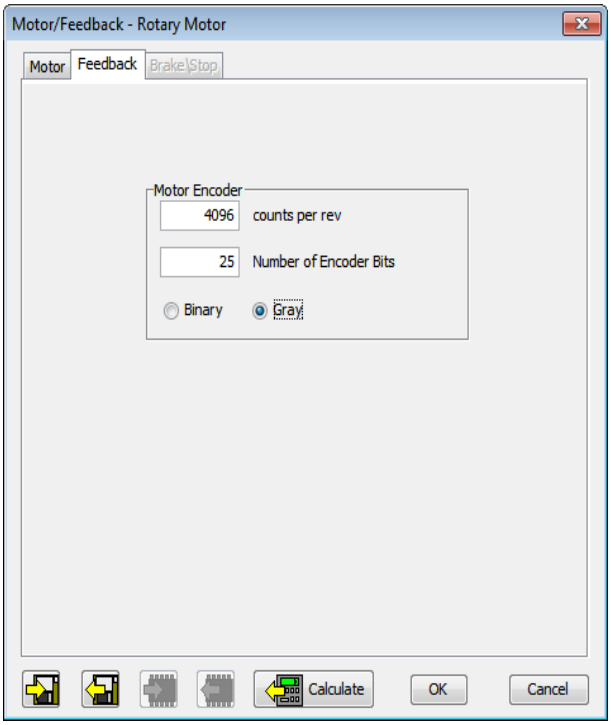
1. In CME, navigate to the Feedback Options screen: **Amplifier→Basic Setup→Change Settings→Next**.

   On the *Feedback Options* screen, set **Motor Encoder** to SSI.



Basic Setup (SSI) Screen

2. Open the *Motor/Feedback* screen *Feedback* tab.



Motor/Feeback – Rotary Motor Screen

To view or change the settings, refer to the Fields described in the following table.

Field	Description
Counts per rev	Counts per single revolution of the encoder shaft.
Number of Encoder Bits	Resolution of the encoder output.
Binary	Parallel output.
Gray	Parallel output with single bit change.

3.6 EnDat Encoders (Type 11)

This section describes the use of EnDat Absolute Encoders used with Copley Controls drives.

3.6.1 EnDat Encoder and Drive Communication

EnDat Encoders can be single or multi-turn. They provide all digital bidirectional communication sent to the Copley drive.

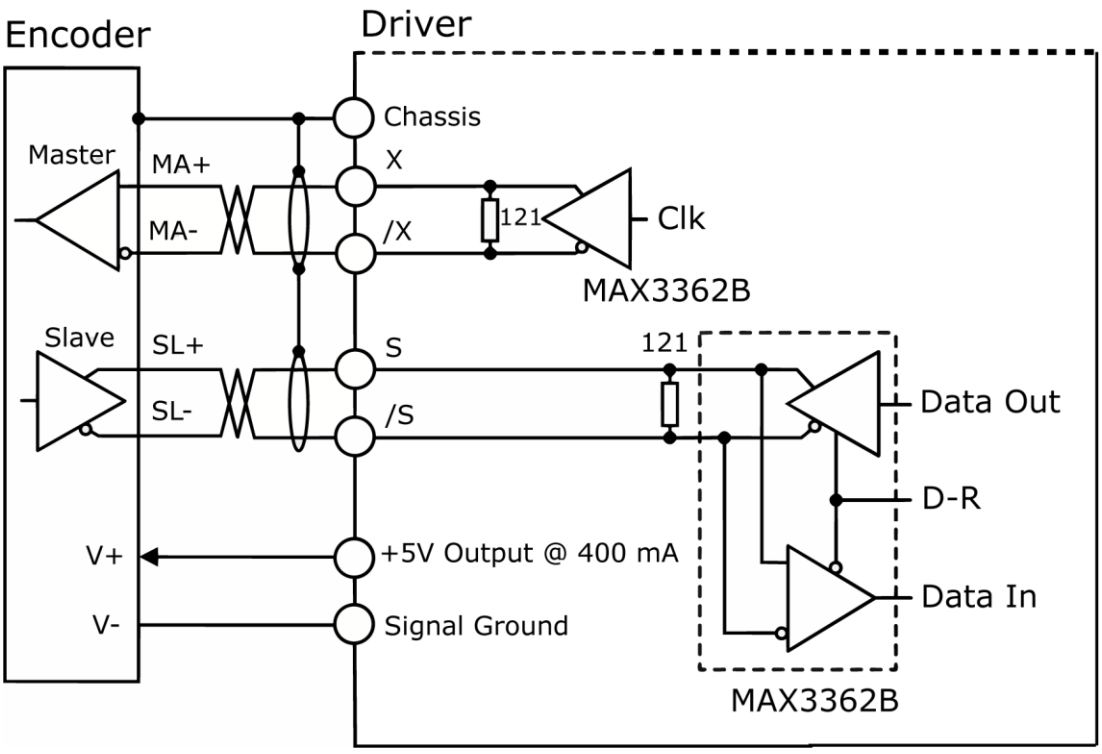
The type of transmission (position values, parameters, diagnostics, etc.) is selected by using the Mode commands that the drive sends to the encoder. The drive can read, and update information stored in the encoder, or save new information. The encoder's zero reference point may be set by the drive. The clock and data are wired as differential pairs.

An Endat Encoder transmits data in synch with the drive's clock signal.

Because EnDat 2.1 Encoders have slow update rates, the transmission of position values requires sin/cos inputs (1V p-p). EnDat 2.2 Encoders have fast update rates and do not require sin/cos inputs.

3.6.2 EnDat 2.2 Encoder Wiring Example

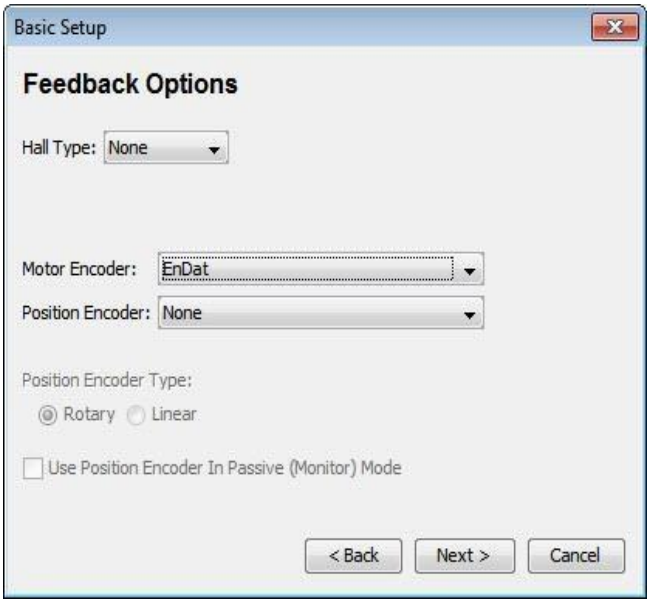
The following diagram Below is a typical EnDat 2.2 encoder wiring diagram. Use of programmable sin/cos incremental signals is not required. For information on the detailed wiring specifications, refer to the datasheets. The datasheets are posted on the Copley website: https://www.copleycontrols.com.



EnDat 2.2 Encoder Wiring Diagram

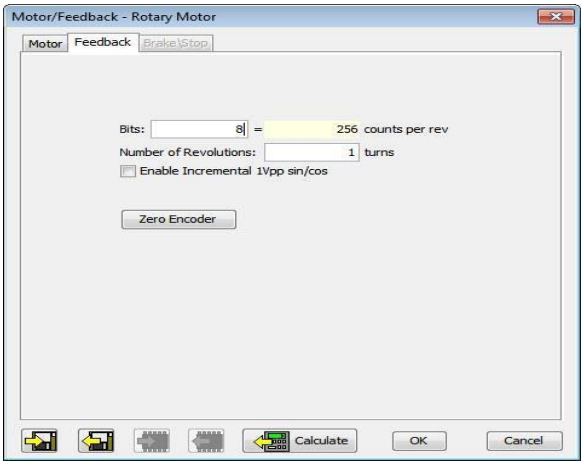
3.6.3 EnDat Encoder Configuration Example

1. In CME, navigate to the Feedback Options screen: **Amplifier→Basic Setup→Change Settings→Next**.
On the *Feedback Options* screen, set **Motor Encoder** to EnDat.



Basic Setup (EnDat) Screen

2. Open the *Motor/Feedback* screen's *Feedback* tab.



Motor/Feedback - Rotary Motor Screen

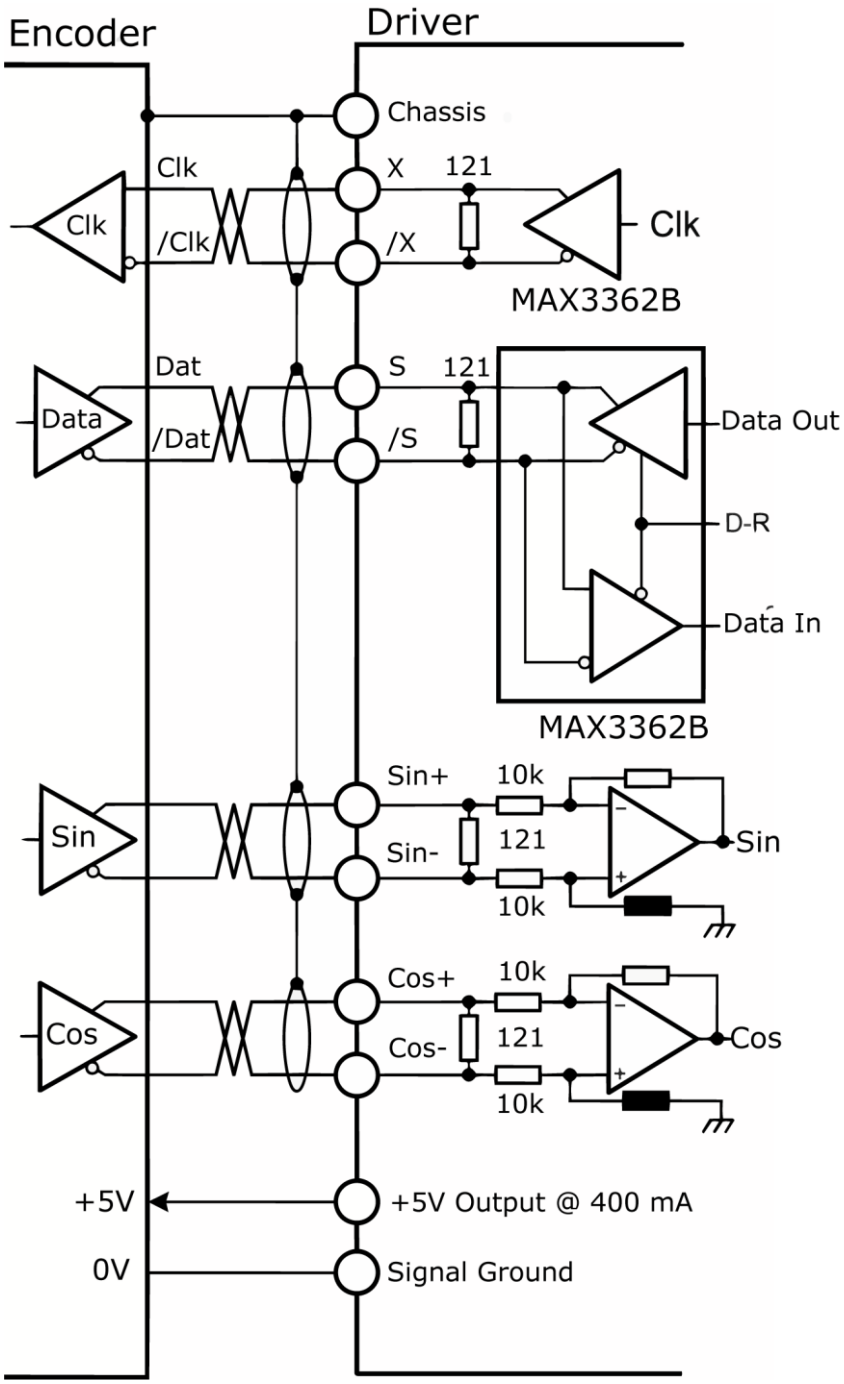
To view or change the settings, refer to the Fields described in the following table.

Field	Description
Bits	Resolution of the encoder output.
Number of Revolutions	The number of turns on a multi-turn encoder.
Enable Incremental 1Vpp sin/cos	Not available or needed with EnDat 2.2.
Zero Encoder	Parallel output with single bit change.

3.6.4 EnDat 2.1 Wiring Example

The following diagram shows a typical EnDat 2.1 Encoder Wiring diagram.

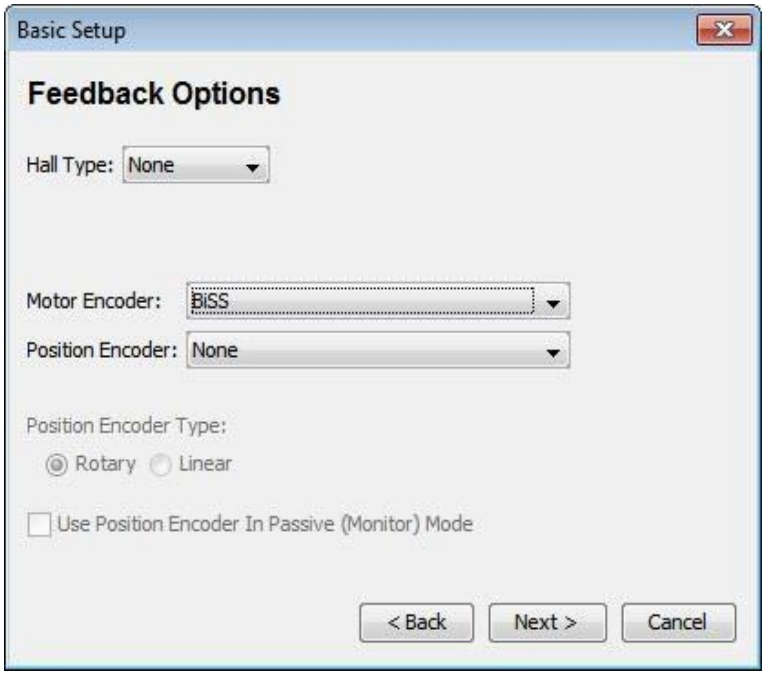
For information on the detailed wiring specifications, refer to the datasheets. The datasheets are posted on the Copley website: https://www.copleycontrols.com.



EnDat 2.1 Encoder Wiring Diagram

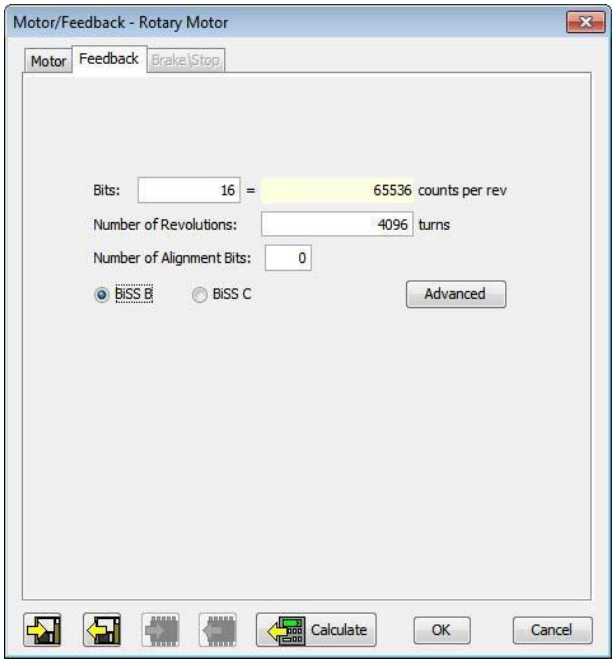
3.7 EnDat 2.1 Encoder Configuration Example

1. In CME, navigate to the Feedback Options screen: **Amplifier□Basic Setup□Change Settings□Next**.
   On the *Feedback Options* screen, set **Motor Encoder** to BiSS.



Basic Setup (BiSS) Screen

2. Open the *Motor/Feedback* screen's *Feedback* tab.

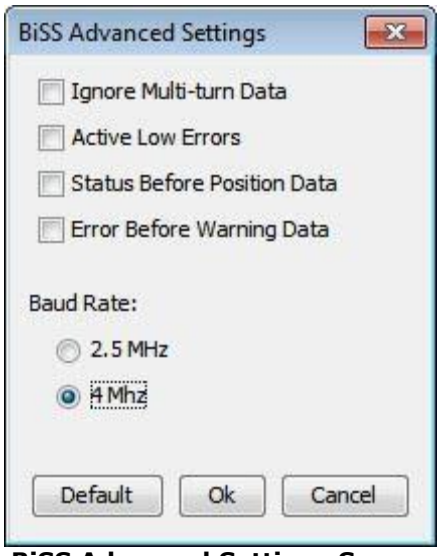


Motor/Feedback – Rotary Motor Screen

To view or change the settings, refer to the Fields described in the following table.

Field	Description
Bits	Resolution of the encoder output.
Number of Revolutions	The number of turns on a multi-turn encoder.
Number of alignment Bits	Bits (which contain no useful information), that may be ignored, after position data has been sent.
BiSS B	Mode choice at each start cycle.
BiSS C	Continuous mode.

1. On the *Feedback* screen click the **Advanced** button to open the *BiSS Advanced Settings* window.



BiSS Advanced Settings Screen

To view or change the settings, refer to the Fields described in the following table.

Field	Description
Ignore Multi-turn Data	Use only single turn data.
Active Low Errors	Active low option. Active high is the default.
Status Before Position Data	Status before position data option. Position data before status is the default.
Error Before Warning Data	Error before warning option. Warning before error is the default.
Baud Rate	A baud rate option of 2.5 MHz. 4 MHz is the default.

3.8 Absolute A Encoders (Type 14)

This section describes the use of Absolute A Encoders in Copley Controls drives.

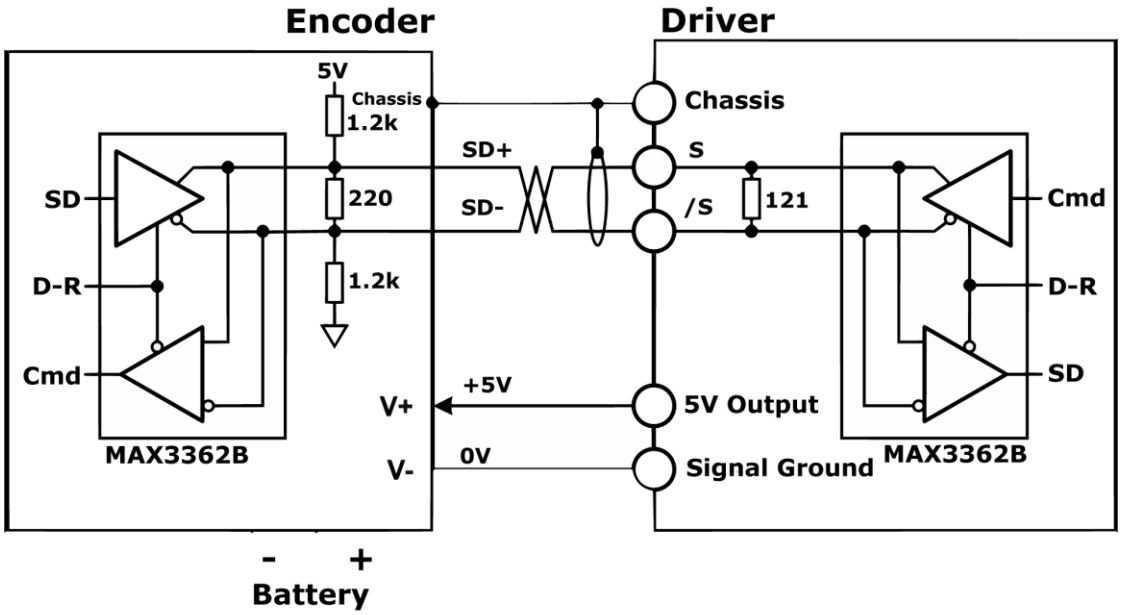
3.8.1 Absolute A Encoder and Drive Communication

Absolute A Encoders can be single or multi-turn. They provide all digital one way communications to the selected Copley drive with a bi-directional option.

The Absolute A interface uses 2-wire, half-duplex communication to handle position values and diagnostics.

3.8.2 Absolute A Encoder Wiring Example

The following diagram shows the typical Absolute A Encoder wiring. Absolute A Encoders typically use batteries to save multi-turn position data in case of power failure. For information on the detailed wiring specifications, refer to the datasheets. The datasheets are posted on the Copley website: https://www.copleycontrols.com.

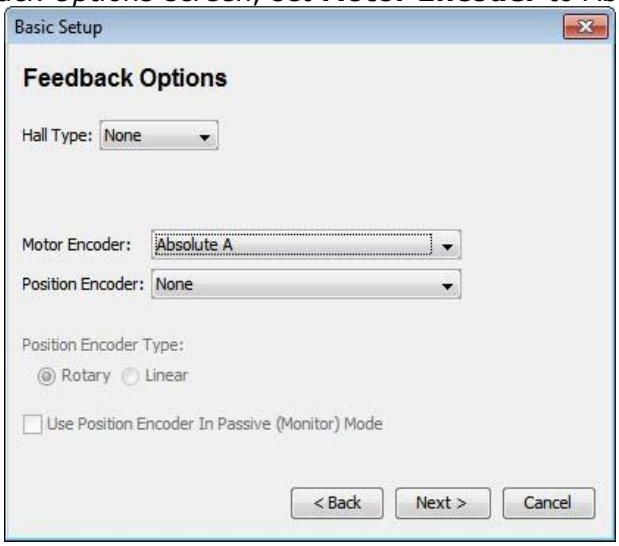


Absolute A Encoder Wiring Diagram

3.8.3 Absolute A Configuration Example

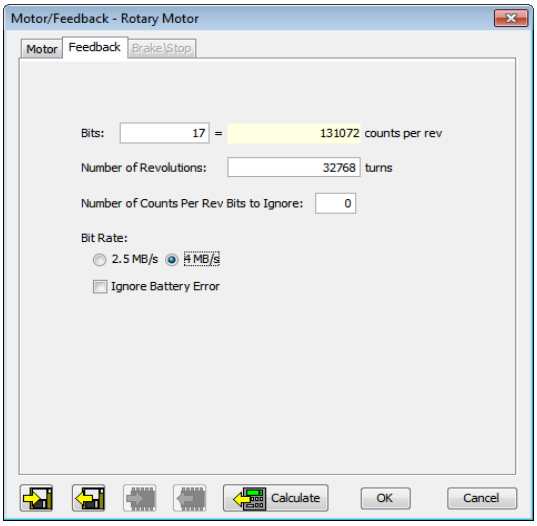
In the example below, Tamagawa, Panasonic, and Sanyo Denki Absolute A encoders may be substituted for *Absolute A* in the Feedback Options screen's Motor Encoder entry drop down menu.

1. In CME2, navigate to the Feedback Options screen: **Amplifier□Basic Setup□Change Settings□Next**.
On the *Feedback Options* screen, set **Motor Encoder** to Absolute A.



Basic Setup (Absolute A) Screen

2. Open the Motor/Feedback screen's Feedback tab.



Motor/Feedback Screen

To view or change the settings, refer to the Fields described in the following table.

Field	Description
Bits	Resolution of the encoder output.
Number of Revolutions	The number of turns on a multi-turn encoder.
Number of Counts Per Rev Bits to Ignore	Bits (which contain no useful information) to ignore, after the position data is sent.
Bit Rate	Used for communication. 2.5MB/s or 4 MB/s

3.9 Incremental A Encoders (Type 14)

This section describes the use of Incremental A Encoders used with Copley Controls drives.

The Incremental A Encoder is considered a subset of Absolute encoders because it provides absolute Hall information used for commutation.

3.9.1 Incremental A Encoder and Drive Communication

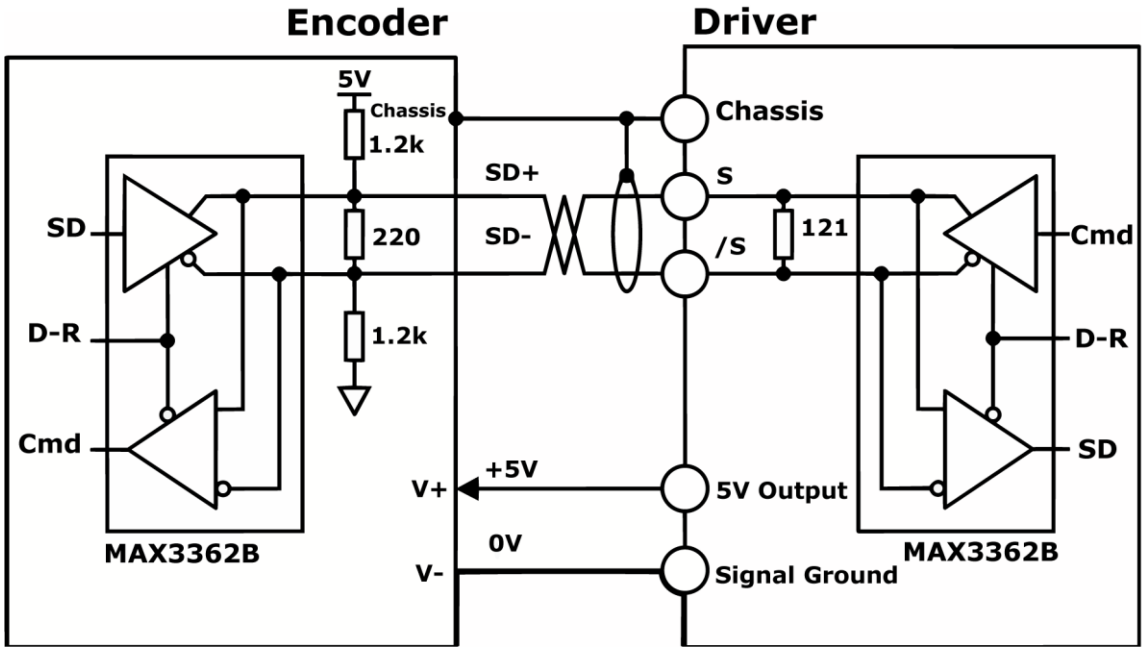
Incremental A Encoders provide incremental position and Hall data to the selected Copley drive.

The Incremental A interface uses 2-wire, half-duplex communication to handle position values and diagnostics. The data is wired as a differential pair.

3.9.2 Incremental A Encoder Wiring Example

The following diagram shows the typical Incremental A Encoder wiring. Because there is no absolute position, a battery is not required.

For information on the detailed wiring specifications, refer to the datasheets. The datasheets are posted on the Copley website: https://www.copleycontrols.com.

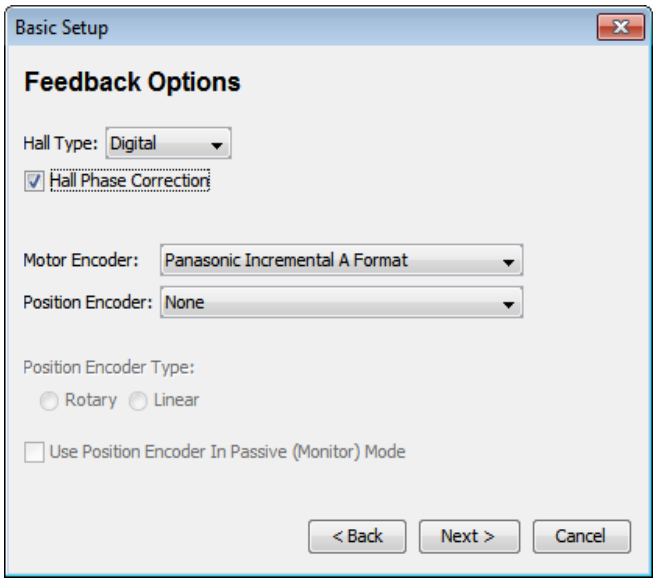


Incremental A Encoder Wiring Diagram

3.9.3 Incremental A Configuration Example

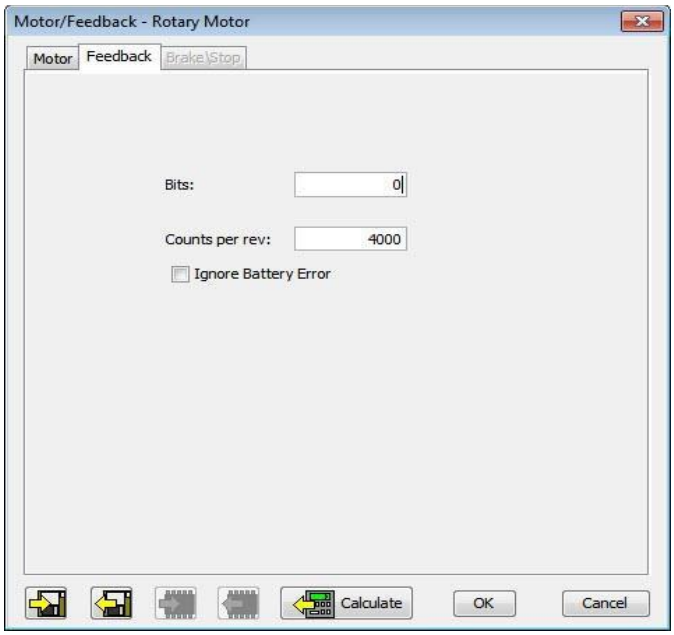
1. In CME2, navigate to the Feedback Options screen: **Amplifier →Basic Setup→Change Settings→Next**.

On the *Feedback Options* screen, set **Motor Encoder** to *Incremental A Format*, for example.



Basic Setup (Panasonic Incremental A Format) Screen

2. Open the *Motor/Feedback* screen's *Feedback* tab.



Motor/Feedback – Rotary Motor Screen

To view or change the settings, refer to the Fields described in the following table.

Field	Description
Bits	Resolution of the encoder output.
Counts per rev	The number of counts/rev post quadrature.
Ignore Battery Error	Ignores battery error status from the encoder.

3.10 BiSS Encoders (Type 13)

This section describes the use of BiSS encoders in Copley Controls drives.

3.10.1 BiSS Encoder and Drive Communication

BiSS encoders can be single or multi-turn. They provide all digital one way communications to the selected Copley drive with a bi-directional option.

A train of clock pulses from the drive initiates the transmission of position data by subsequent clock pulses. The clock and data are wired as differential pairs.

BiSS is an open source digital interface for sensors and actuators. It refers to principles of well known industrial standards for Serial Synchronous Interfaces like SSI, with additional options:

- BiSS B-protocol: Mode choice at each cycle start
- BiSS C-protocol: Continuous mode
- Synchronous high speed communication

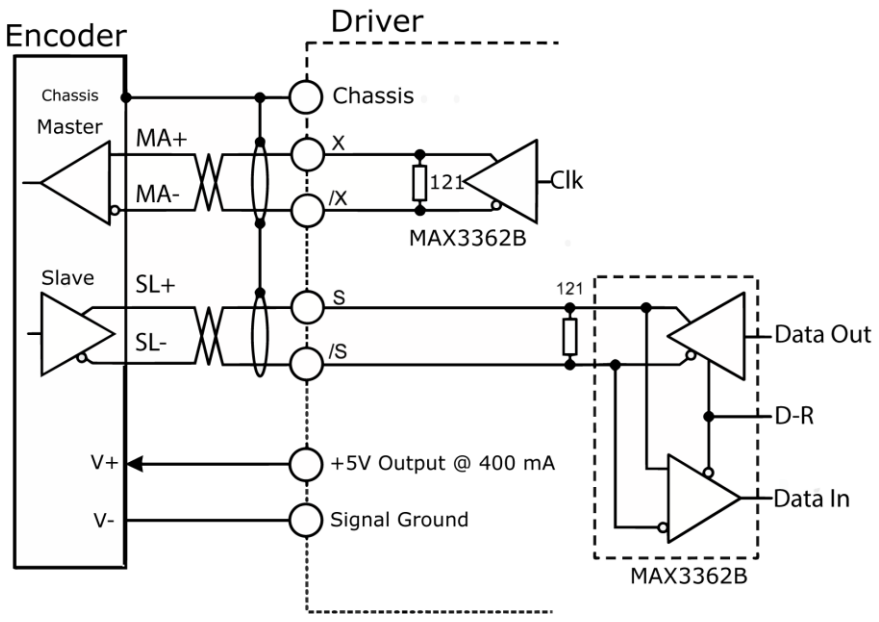
Line delay compensation for high speed data transfer

- Request for data generation at slaves
- Safety capable: CRC, Errors, Warnings

3.10.2 BiSS Encoder Wiring Example

The following diagram shows the typical BiSS Encoder wiring.

For information on the detailed wiring specifications, refer to the datasheets. The datasheets are posted on the Copley website: https://www.copleycontrols.com.

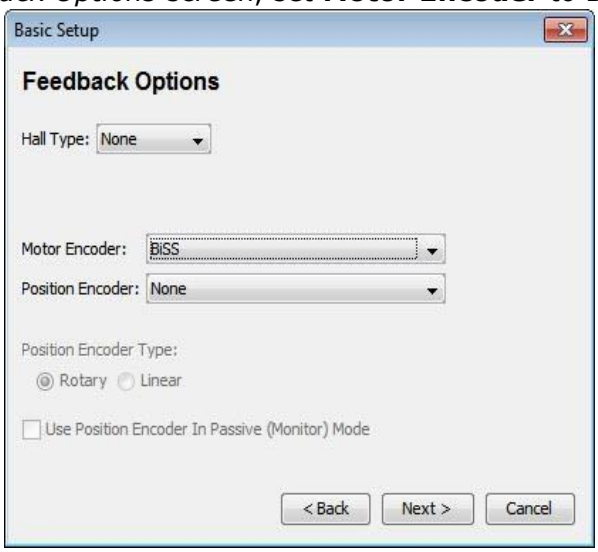


Biss Encoder Wiring Diagram

3.10.3 BiSS Encoder Configuration Example

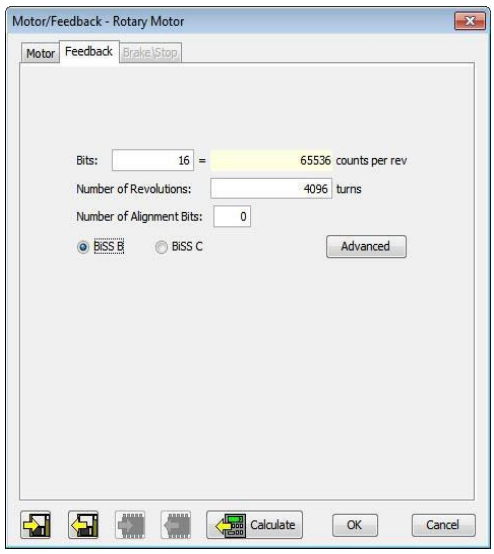
3. In CME2, navigate to the Feedback Options screen: **Amplifier →Basic Setup→Change Settings→Next**.

On the *Feedback Options* screen, set **Motor Encoder** to BiSS.



Basic Setup (BiSS) Screen

1. Open the *Motor/Feedback* screen's *Feedback* tab.

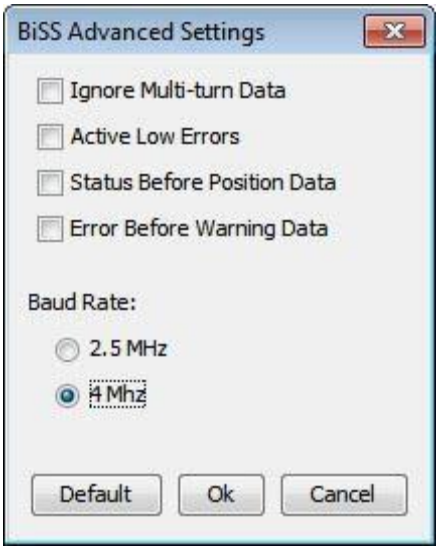


MotorFeedback – Rotary Motor Screen

To view or change the settings, refer to the Fields described in the following table.

Field	Description
Bits	Resolution of the encoder output.
Number of Revolutions	The number of turns on a multi-turn encoder.
Number of Alignment Bits	Bits (which contain no useful information), that may be ignored, after position data has been sent.
BiSS B	Mode choice at each start cycle.
BiSS C	Continuous mode.

1. On the *Feedback* screen click the **Advanced** button to open the *BiSS Advanced Settings* window.



Biss Advanced Settings Screen

To view or change the settings, refer to the Fields described in the following table.

Field	Description
Ignore Multi-turn Data	Use only single turn data.
Active Low Errors	Active low option. Active high is the default.
Status Before Position Data	Status before position data option. Position data before status is the default.
Error Before Warning Data	Error before warning option. Warning before error is the default.
Baud Rate	A baud rate option of 2.5 MHz. 4 MHz is the default.

4 Encoder Management

This chapter describes the following:

- Management of Encoder Errors
- ASCII Programming using CME
- Clearing Faults
- Absolute Position

CME uses ASCII variables to read status information and to configure absolute encoders. The mapping of bits to function depends on the encoder type. For details pertaining to specific encoder types, refer to the Encoder Status Words and Encoder Option Words sections.

- For a list of parameters, refer to the Copley Amplifier Parameter Dictionary.
- For detailed ASCII programming (which includes error codes), see the Copley ASCII Interface Programmer's Guide.

To access and download the Guides, refer to the Copley website: https://www.copleycontrols.com.

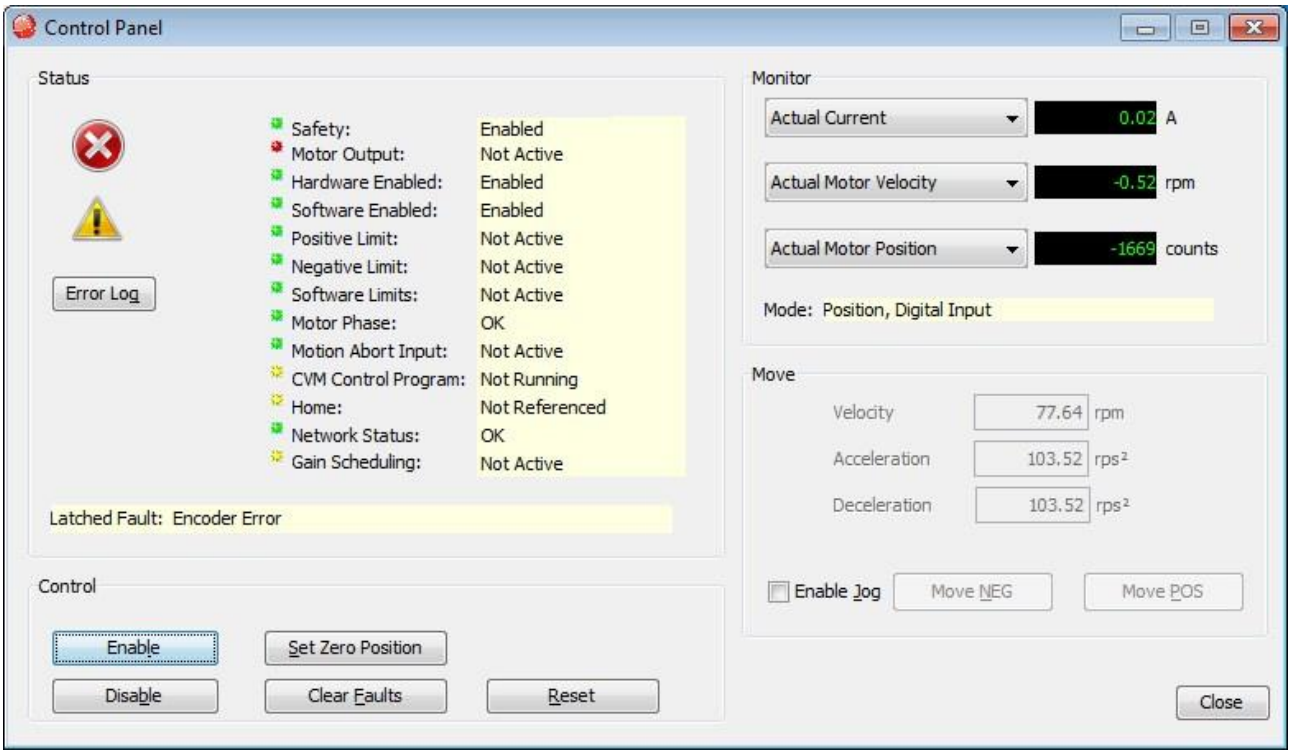
4.1 CME Encoder Errors

CME displays fault information via the Control Panel and Error Log windows.

4.1.1 Control Panel

The CME Control Panel will indicate if there is an encoder error. If any red lights are displayed in the Status section, it indicates that motion will be inhibited. After the problems are fixed, click the **Clear Faults** or the **Reset** button to clear the fault indicators.

To open the *Control Panel*, click on the icon  from the Toolbar, or, from the Main Menu choose **Amplifier→Control Panel**.

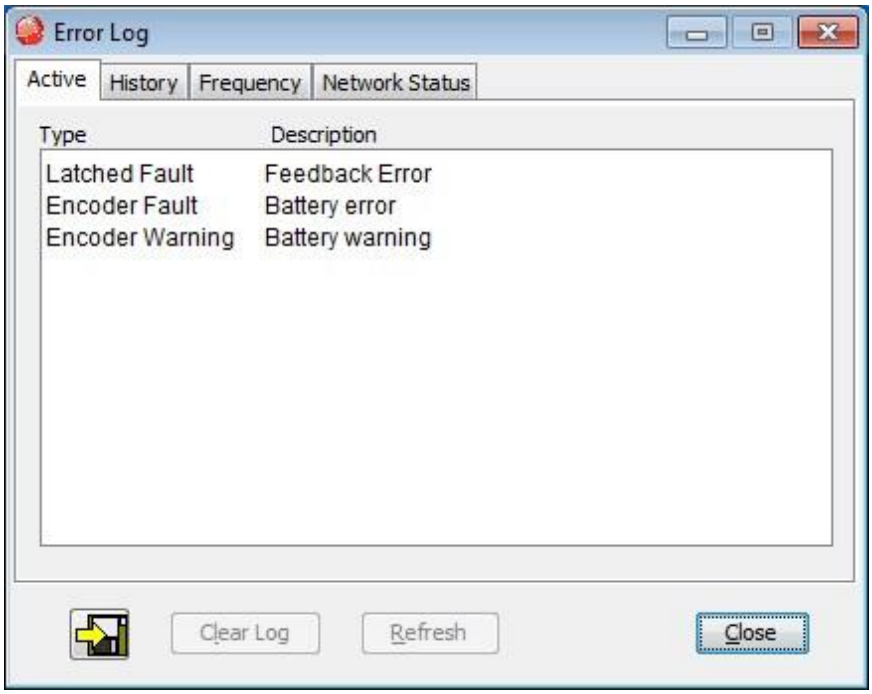


Control Panel

4.2 Error Log

The Error Log window will display the type of encoder error and its description. At the Error Log, troubleshooting may begin. The window below shows examples of errors.

To open the *Error Log*, click on the icon from the Toolbar, or, from the Main Menu, choose **Amplifier→Error Log**.



Error Log

4.2.1 Troubleshooting Table

The table below shows examples of errors described in the Error Log window and their likely solutions.

Description	Problem	Solution
Feedback Error	This could indicate an open wire.	Fix and verify good connections.
CRC Error	CRC (Cyclical Redundancy Check). Check on feedback data. This is an indication of poor cabling, grounding or shielding.	Check cabling system lengths, shield path to Earth, and drive grounding.
Battery Warning	Battery is not in the circuit or dead, but encoder still has power.	Replace battery.
Battery Error	Battery was not in the circuit or dead while the power was OFF.	Replace battery and manage possible lost multi-turn position.

After the faults are resolved, clear them using the Control Panel **Clear Faults** button.

4.3 Encoder Status Words

There are two encoder status words available:

- one used for motor encoder status: ASCII **0x12E**
- one used for load encoder status: ASCII **0x12F**

4.3.1 0x12E Motor Encoder Status

The Encoder Status Word returns status information used for the motor encoder. Bits set in the status word are latched and cleared when the status value is read (See Encoder Status Bits below). The format of this status word is dependent on the encoder type. Many error bits are taken directly from the encoder data stream. For a description of what these error bits mean, consult the Encoder Manufacturer.

The table below includes the equivalent MACRO, CANopen, and EtherCAT (use the CAN ID) status words.

From the Copley Amplifier Parameter Dictionary					
ASCII	**DvcNet**	**Macro**	**CAN ID:sub**	**Bank**	**Type**
0x12E	0x12F	0x52E	0x2224	R*	U32
*NOTE: Read-only.					

4.3.2 0x12F Load Encoder Status

This status word is the same as 0x12E, but it is used for the load encoder.

The table below includes the equivalent MACRO, CANopen, and EtherCAT (use the CAN ID) status words.

From the Copley Amplifier Parameter Dictionary					
ASCII	**DvcNet**	**Macro**	**CAN ID:sub**	**Bank**	**Type**
0x12F	0x12G	0x52F	0x2225	R*	INT32
*NOTE: Read-only.					

For more details, refer to the Copley Amplifier Parameter Dictionary. To access and download the Parameter Dictionary, refer to the Copley website: http://www.copleycontrols.com.

4.4 Encoder Status Bits

The following subsections include the Status Bit Tables for commonly used encoder types.

4.4.1 SSI Encoders (Type 12)

Bits	Description
0 - 6	Fault flags returned from the encoder.
15	Encoder data invalid bit set.

4.4.2 EnDat Encoders (Type 11)

Bits	Description
0	CRC error on data received from encoder.
1	Failed to detect encoder connected to amplifier.
2	Error bit on encoder stream is active.
3	Encoder failed to respond to request for position.

4.4.3 Absolute A Encoders (Type 14)

Bits	Description
0	Battery warning reported by encoder.
1	Battery error reported by encoder.
3	Over speed reported by encoder.
4	Memory error reported by encoder.
5	STERR reported by encoder.
6	PSERR reported by encoder.
7	Busy error reported by encoder.
8	Memory busy reported by encoder.
9	Over temperature reported by encoder.
15	CRC error on data received from encoder.

4.4.4 Incremental A Encoders (Type 14)

Bits	Description
0	Battery warning reported by encoder.
1	Battery error reported by encoder.
3	Over speed reported by encoder.
4	Memory error reported by encoder.
5	STERR reported by encoder.
6	PSERR reported by encoder.
7	Busy error reported by encoder.
8	Memory busy reported by encoder.
9	Over temperature reported by encoder.
15	CRC error on data received from encoder.

4.4.5 BiSS Encoders (Type 13)

Bits	Description
0	CRC error on data received from encoder.
1	Encoder failed to transmit data to amp.
2	Error bit on encoder stream is active.
3	Warning bit on encoder stream is active.
4	Encoder transmission delay is too long.

4.5 Encoder Option Words

Encoder option words are used to configure motor or load encoders. There are two encoder options of words available.

- One used for Motor Encoders: ASCII **0x12A**, and
- One used for Load or Position Encoders: ASCII **0x12B**.

The mapping of option bits to function depends on the encoder type, see Encoder Option Bits below.

4.5.1 0x12A Motor Encoder Options

0x12A specifies various configuration options for the motor encoder.

The table below includes the equivalent MACRO, CANopen, and EtherCAT (use the CAN ID) status words.

From the Copley Amplifier Parameter Dictionary					
ASCII	**DvcNet**	**Macro**	**CAN ID:sub**	**Bank**	**Type**
0x12A	0x12B	0x52A	0x2222	F	INT32

4.5.2 0x12B Load Encoder Options

This 0x12B word is the same as 0x12A, but it is used for the Load or Position Encoder. The table below includes the equivalent MACRO, CANopen, and EtherCAT (use the CAN ID) status words.

From the Copley Amplifier Parameter Dictionary					
ASCII	**DvcNet**	**Macro**	**CAN ID:sub**	**Bank**	**Type**
0x12B	0x12C	0x52B	0x2223	F	INT32

For more details, refer to the Copley Amplifier Parameter Dictionary. To access and download the Parameter Dictionary, refer to the Copley website: http://www.copleycontrols.com.

4.5.3 Encoder Option Bits

Below are option bit tables for configuring commonly used encoder types.

4.5.4 SSI Encoders (Type 12)

Bits	Description
0-5	Number of bits of position data available.
8-10	Number of extra status bits sent after position data.
12	If set, ignore the first bit of data sent by the encoder.
13	If set, encoder outputs position data using Gray code.
14	If set, pull clock low briefly after data (custom for Codechamp encoder).
15	If set, data is sent on LSB first.
16-21	Encoder bit rate in 100 kHz units. If zero, default to 1MHz.
22	If set, use setting of encoder counts/rev to determine how many data bits to use.
24	If set, first bit is 'data valid' bit.

4.5.5 EnDat Encoders (Type 11)

Bits	Description
0-4	Number of bits of single turn data available from encoder.
8-12	Number of bits of multi-turn data available from encoder.
16	Set if analog inputs are supplied by encoder.
17	If set, use multi-mode port.
18	If set, read position using EnDat 2.2 style commands rather than the default 2.1 style.
20-23	Number of least significant bits of the encoder reading to discard.

4.5.6 Absolute A Encoders (Type 14)

Bits	Description
0-5	Number of bits of single turn data.
8-12	Number of bits of multi-turn data.
16-19	Number of LSB to discard from reading.
20-22	Number of consecutive CRC errors to ignore before generating an error.
24-27	Encoder sub-type (0=Tamagawa, 1=Panasonic absolute, 2=HD systems, 3=Panasonic Incremental, 4=Sanyo Denki).
28	Bit rate (set for 4 Mbit, clear for 2.5 Mbit).
30	If set, treat encoder battery errors as warnings.

4.5.7 Incremental A Encoders (Type 14)

Bits	Description
0-5	Number of bits of single turn data.
8-12	Number of bits of multi-turn data.
16-19	Number of LSB to discard from reading.
20-22	Number of consecutive CRC errors to ignore before generating an error.
24-27	Encoder sub-type (0=Tamagawa, 1=Panasonic absolute, 2=HD systems, 3=Panasonic Incremental, 4=Sanyo Denki).
28	Bit rate (set for 4 Mbit, clear for 2.5 Mbit).
30	If set, treat encoder battery errors as warnings.

4.5.8 BiSS Encoders (Type 13)

Bits	Description
0-5	Number of bits of single turn data.
8-12	Number of bits of multi-turn data.
15	If set, assume the encoder position data wraps after the number of encoder counts programmed in parameter 0x62.
16	Set for mode-C encoder format.
20	Set if encoder error and warning bits are active low.
21	Set if encoder status bits are sent before position data, clear if status bits are sent after position data.
22	Set if encoder error bit is transmitted before warning bit. Clear for warning bit sent first.
24-26	Number of alignment bits (reserved bits sent before position info).
28	Use multi-mode port if set. If clear use primary encoder interface.
30	Set for 2.5MHz baud rate, clear for 4MHz baud rate.

4.6 Clearing Faults

In the Encoder: Before an encoder's latched *faults* can be cleared, any faulty mechanical or electrical conditions in the encoder must be corrected.

In the Drive: Before a drive's latched *encoder faults* can be cleared, any faulty mechanical or electrical conditions in the encoder, drive or mechanical system must be corrected.

4.6.1 Clearing Faults in an Absolute Encoder

After faulty conditions have been corrected, any internal latched faults in an encoder must be cleared to resume normal encoder use.

Latched faults in an encoder may be cleared through Copley drives or, with some smart encoders, through the encoders themselves.

The following interfaces can be used to clear latched faults in an encoder:

- ASCII Interface
- Binary Serial Interface
- CANopen and EtherCAT
- MACRO

NOTE: power cycling an encoder may not clear an *internal* encoder fault condition.

4.7 ASCII Interface

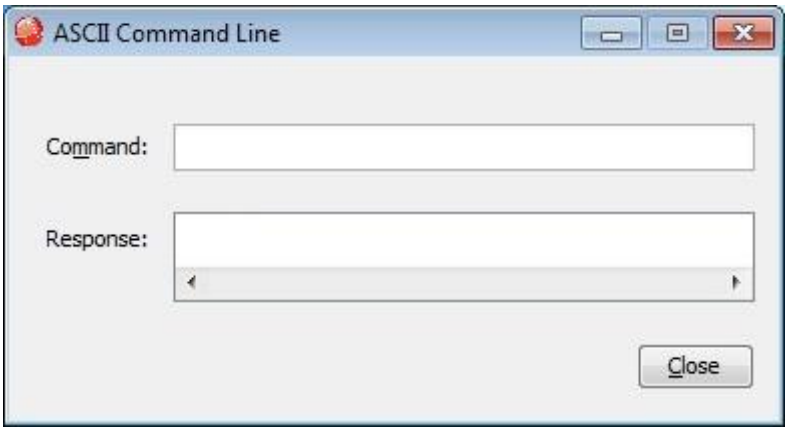
Copley drives support a command on CME's ASCII interface that can be used to clear encoder faults.

- To clear a fault on a motor encoder, use the CME ASCII command **enc clear.**
- To clear a fault on a load encoder, use the CME ASCII command **ldenc clear.**

Upon success, these commands will return '**ok**'. If an error occurs or the drive's firmware does not support clearing latched faults for the specific type of encoder, an error response will be returned.

To enter a command using CME's ASCII interface, do the following steps.

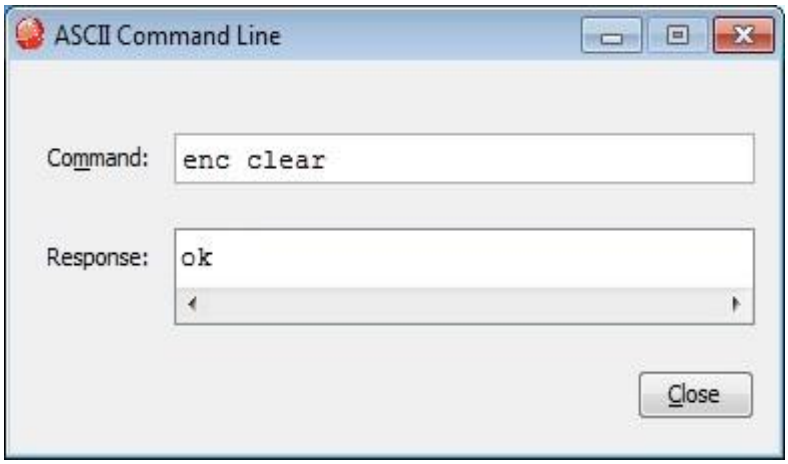
1. From the Main screen, choose **Tools→ASCII Command Line** to open the tool.



ASCII Command Line

2. Enter the ASCII Command **enc clear** or **Idenc clear** in the **Command** field.

3. Press the **Enter** key to send the command to the drive. Observe the **Response** field. In the **Response** field, ok appears.



ASCII Command Line

4.7.1 Binary Serial Interface

The binary serial interface of a Copley drive supports clearing latched encoder faults through op-code 0x1B. This op-code uses a single word of data to identify the type of command to send to the encoder. The following tables show the data word formatted.

Bits	Description
0	Identify which encoder to clear; 0 for motor encoder, 1 for load encoder.
1-3	Reserved.
4-7	Sub-command code (Should be 1 to clear encoder errors).
8-11	Reserved.
12	Axis number; 0 for axis 1, 1 for axis 2.
13-15	Reserved.

The following encoder sub-commands are currently supported.

Code	Description
0	Read a register from within the encoder. The register number is passed as a second word of data to the encoder command.
1	Reset any errors currently latched on the encoder
2	Zero the encoder's internal position.

For example:

To clear latched faults on the motor encoder of axis 1, the data word would be **0x0010**.

To clear latched faults on the load encoder of axis 2, the data word would be **0x1011**.

The full binary command consists of a 4 byte header followed by the two byte data word. To clear latched faults on the motor encoder of axis 1, the full command sent to the drive would be:

0x00 0x50 0x01 0x1B 0x00 0x10

For more details on the Binary Serial Protocol, please contact Copley Controls.

4.8 CANopen and EtherCAT

To clear latched encoder faults over an EtherCAT or CANopen interface, **object 0x2000** is used. This object allows binary serial commands to be transmitted over the CANopen/EtherCAT interface. Serial port commands can be sent using object 0x2000 by first writing the command to the object using an SDO. The data written to 0x2000 consists of the op-code code 0x1B (for encoder commands), followed by any data words.

- Note that when communicating over the CANopen/EtherCAT interface, the data words are sent via LSB first.
- When communicating over the normal binary serial interface, the data words are sent via MSB first.

For example:

To clear latched encoder faults on the motor encoder of axis 1, use op-code **0x1B** and a single data word **0x0010**. To write this command to object **0x2000**, pass the following three bytes of data: **0x1B 0x10 0x00.**

To read back the result of the command, read object **0x2000**. It should contain a single byte of data which displays as **0x00** to indicate success, or an error code displays to indicate failure.

4.8.1 MACRO

To clear latched encoder faults over the MACRO interface, the MACRO node I-variable 1018 should be used. This I-variable allows binary serial commands to be sent over the MACRO interface.

The 48-bit value written to 1018 consists of an operation code in the upper 16-bits, and data in the lower 32-bits. A more detailed description of this I-variable is available in Copley's MACRO interface documentation.

As described above, the serial port command code used to access absolute encoders is 0x1B. To clear latched encoder faults, a single additional word of data needs to be passed. To clear faults on the motor encoder of axis 1, the passed data word is 0x0010.

Sending this command over the MACRO interface involves two writes to I-1018:

1. Save the data word to the internal buffer associated with 1018:
   MS0,i1018 = 0x000200100000

2. Execute serial port command code 0x1B: MS0,i1018 = 0x0007001**B**0000

4.9 Clearing Faults in the Drive

This section describes the following five ways available to clear faults in the drive:

- CME Control Panel
- CANopen or EtherCAT
- Power Cycling
- ASCII Command using an External Application
- ASCII clear command via CME

After all fault conditions are corrected, the following actions may be taken.

1. CME Control Panel

Open the CME Control Panel and click on the **Clear Faults** or **Reset** button.
See the CME User Guide for more details.

2. CANopen or EtherCAT

To clear a driver's *Encoder Faults* over a CANopen or EtherCAT network, use control word **0x6040**, bit 7.
For more information, see Copley's CANopen/EtherCAT Programmer's Manual.

3. Power Cycling

Power cycle the drive to clear a driver's encoder faults.
See Related Documentation, for manuals mentioned in this section.

4.10 ASCII Command using an External Application

An external control application (HMI, PLC, PC, etc.) can use the drive's RS-232 serial bus to set up and direct the serial control of the drive. The control application can issue drive commands from the set of ASCII format commands that make up the Copley Controls ASCII Interface.

For example:

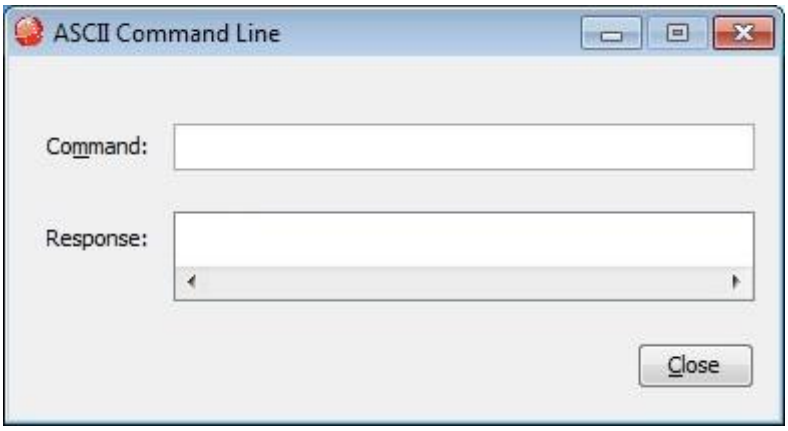
> To clear a driver's encoder faults using an external control application, write a **1** to bit **7** of the fault register **0xa4**.

For more information, see Copley's ASCII Programmers Guide.

CME may also be used to send an ASCII command to clear latched *Encoder Faults* on the drive as described in the procedure in Section 4.10.1.

4.10.1 ASCII Clear Command via CME

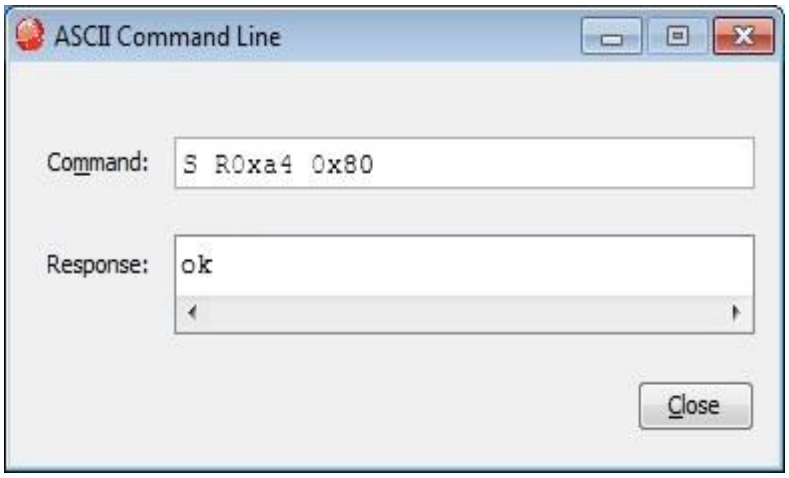
1. From the Main screen, choose Tools☐ASCII Command Line to open the tool.



ASCII Command Line Screen

2. Type **S R0xa4 0x80** in the *Command* field.

3. Press the **Enter** key to send the command to the amplifier. Observe the *Response* field. The response should read *ok*.



ASCII Command Line Screen

4.11 Managing Absolute Position

4.11.1 Lost Battery

If Multi-turn Absolute Encoders are used, do not ignore lost battery faults and warnings. If a battery is removed and the system is powered OFF for an extended length of time, the absolute position may need to be re-calibrated.

Some Multi-turn Encoders use mechanical linkage and do not require a battery to hold absolute multi-turn information.

Single-turn Encoders do not usually require a battery.

4.11.2 Mechanical Slipping

Absolute position may be lost due to mechanical slipping between the Absolute Encoder and the mechanical system to the load. Always verify that there is reliable mechanical linkage with no slipping in the mechanical system.

 **DANGER!**	**Mechanical Slipping Warning:**
	If multi-turn absolute position is lost, damage to equipment or life may be at risk. **FAILURE TO COMPLY WITH THIS WARNING CAN CAUSE EQUIPMENT DAMAGE, INJURY, OR DEATH.**

4.12 Calibration

All Absolute Encoders need to be calibrated to zero or home. This section describes three choices: Manual Calibration, the Homing Method, and Writing to an Encoder (using CME or through CANopen/EtherCAT).

4.12.1 Manual Calibration

This procedure begins with the encoder not mounted to the mechanical system.

1. Rotate the encoder shaft until the zero position is displayed. CME may also be used with the drive to move the encoder to the zero position.

2. Move the mechanical system to a zero reference.

3. Keeping the encoder in zero position, mount the encoder to the mechanical system.

4.12.2 Homing Method

This procedure begins with the encoder mounted to the mechanical system. The Absolute Encoder will power up with an arbitrary position value, and the drive will add an offset to reference the encoder to zero.

1. Rotate the motor to a good Home position.

2. In CME, run the Homing Method *Absolute Position Home*. For information on running the Homing Method, see the CME User Guide.

4.12.3 Writing to an Encoder

Some encoders allow the zero calibration position to be written directly to the encoder. In this case the following applies.

 **DANGER!**	**Lost Commutation Warning:**
	Writing zero to the encoder will cause loss of the motor manufacturer's standard calibration of phase relationship to position and cause lost commutation. Motor manufactures phase calibration will need to be performed again. **FAILURE TO COMPLY WITH THESE WARNINGS CAN CAUSE EQUIPMENT DAMAGE, INJURY, OR DEATH.**

4.12.4 Using CME

From the CME main screen, choose **Tools→ASCII Command Line**. In the Command field enter ***enc zero***.

OR

From CME, navigate to the Manual Phase screen and press the **Zero Encoder** button.



4.12.5 Through CANopen/EtherCAT

To zero an encoder over an EtherCAT or CANopen interface, a command is sent to object 0x2000 (via the binary serial interface) using an SDO. The data written to 0x2000 consists of the op-code 0x1B, followed by any data words.

- Note that when communicating over the CANopen/EtherCAT interface, the data words are sent via the least significant word first.
- When communicating over the normal binary serial interface, the data words are sent via the most significant word first.

For example:

- To zero the motor encoder of axis 1, use op-code 0x1B and a single data word 0x0020.
- To write this command to object 0x2000, pass the following three bytes of data:

  0x1B 0x20 0x00
- To read back the result of the command, read object 0x2000. This object should contain a single byte of data which will be 0x00 for success, or an error code for failure.

Op-code 0x1B uses a single word of data to identify the type of command to send to the encoder. The data word is formatted as follows:

Bits	Description
0	Identify which encoder to clear; 0 for motor encoder, 1 for load encoder.
1-3	Reserved.
4-7	Sub-command code; 2, to zero the encoder's internal position.
8-11	Reserved.
12	Axis number; 0 for axis 1, 1 for axis 2.
13-15	Reserved.

Sub-commands are shown on the next page.

The following encoder sub-commands are currently supported:

Code	Description
0	Read a register from within the encoder. The register number is passed as a second word of data to the encoder command.
1	Reset any errors currently latched on the encoder
2	Zero the encoder's internal position.

Copley Controls Serial Encoder Guide

P/N 16-01055

Revision 00

August 2026

Copley Controls
20 Dan Road
Canton, MA 02021 USA

© 2026
All rights reserved.