



www.triomotion.com

Trio Motion Technology Ltd.
Shannon Way, Tewkesbury,
Gloucestershire. GL20 8ND
United Kingdom
Tel: +44 (0)1684 292333
Fax: +44 (0)1684 297929

1000 Gamma Drive
Suite 206
Pittsburgh, PA 15238
United States of America
Ph: +1 412.968.9744
Fx: +1 412.968.9746

Doc No.: TN20-69
Version: 1.0
Date: 22ND July 2004
Subject: Interfacing EnDat Absolute Encoders

Application Information

Contents:

Contents:.....	1
Introduction:.....	1
System Requirements:.....	1
Wiring Configuration:.....	2
Configuring the Motion Coordinator:.....	2
Relevant BASIC Commands:.....	3
Error Handling:.....	4

Introduction:

The EURO205X and PCI208 Motion Coordinators can be configured to interface directly to EnDat absolute encoders. These encoders respond on a dedicated Clock and Data 1 Mhz RS485 serial interface when their position is requested by the controller. When set to the relevant encoder mode, the axis transmits an information request to the encoder on a fixed 250usec cycle. The data returned is available to BASIC and can be used to servo a motor.

System Requirements:

ENDAT Encoders require the FPGA of the Motion Coordinator to be fitted with a revised program. This needs to be fitted by TRIO and is specified with a different order code.

EURO205X Motion Coordinator with system software v1.6324 or higher and FPGA version 9706 or higher. The axis must be feature enabled for servo operation to function

PCI208 can interface axes to EnDat provided they are in pairs 0 and 1, 2 and 3, 4 and 5, 6 and 7. The FPGA of the PCI208 must be programmed to version ????

Motion Perfect 2 version yyyyyyy is useful although not essential

Wiring Configuration:

The encoder connects directly to the "Z" and "/Z" connections of the EURO205X axis. The following table gives the encoder connections to the EURO205X backplane connector:

Encoder Signal	Wire Colour	Axis 0	Axis 1	Axis 2	Axis 3
SD	Blue	A26	A24	A22	A20
/SD	Blue/Black	A25	A23	A21	A19
GND	Black + Brown/Black	C3/B3/A3	C3/B3/A3	C3/B3/A3	C3/B3/A3

The encoder must be connected to its 5v supply and optional battery supply if that is required.

Configuring the Motion Coordinator:

Using the incremental or absolute encoder ports requires that the relevant "feature enable" codes on the EURO205X Motion Coordinator are installed:

Axis Number	Feature Code for Encoder Use
0	4
1	5
2	6
3	7

On the PCI 208 Motion Coordinator:

Axis Number	Feature Code for Encoder Use
0	8
1	9
2	10
3	11
4	12
5	13
6	14
7	15

After setting the feature enable codes the axes will power up as *incremental* encoder inputs. It is necessary to set the configuration by changing the axis ATYPE using a BASIC program:

Axis ATYPE	Description
33	Unused
34	EnDat absolute encoder with servo control

In either case the +/-10volt DAC's (if fitted) may be used from BASIC when the servo is off.

Before the ATYPE is set it is essential to set the "ENCODER_BITS" parameter for the axis.

Relevant BASIC Commands:

The BASIC implements some new keywords to give the user access to the encoder data.

ENCODER_BITS

This axis parameter configures the interface for the number of encoder bits. Bits 0..7 of the parameter should be set to the total number of encoder bits. Bits 8..14 should be set to the number of multi-turn bits to be used.

For example: A 25 bit encoder has 12 multi-turn and 13 bits/turn resolution. The BASIC program initialises with:

```
ENCODER_BITS=25 + (256 * 12)
ATYPE=34
```

Note that the parameter must be set for each axis.

ENCODER_ID

Unused on the EnDat encoder.

ENCODER_TURNS

This axis parameter returns the number of multi-turn counts from the encoder.

ENCODER_STATUS

Unused on the EnDat encoder.

ENCODER

This axis parameter returns the position within one turn of the encoder.

ENCODER_CONTROL

Endat encoders can be set to either cyclically return their position, or they can be set to a parameter read/write mode. The mode is controlled with the parameter ENCODER_CONTROL.

ENCODER_CONTROL = 1 ` sets parameter read/write mode

ENCODER_CONTROL = 0 ` sets cyclic position return mode

ENCODER_CONTROL is set to 0 on power up or reset. Using the ENCODER_READ or ENCODER_WRITE functions will set the parameter to 1 automatically.

On the PCI 208 the ENCODER_CONTROL should be set for the axis pairs 0/1, 2/3, 4/5 or 6/7 at the same time due to the configuration of the interface transceivers.

ENCODER_READ(address)

Returns a 16 bit encoder parameter. Bits 8..15 of the "address" are the "MRS" setting (see EnDat interface data) and bits 0..7 are the offset within the MRS block. If a CRC error occurs

the function will return -1. ENCODER_READ is an axis function and can use the axis modifier within the Trio BASIC language:

Example:

```
ENCODER_READ($A10D) AXIS(7) ` read the number of encoder bits
```

ENCODER_WRITE(address, value)

Writes to an encoder parameter. Bits 8..15 of the "address" are the "MRS" setting (see EnDat interface data) and bits 0..7 are the offset within the MRS block. If an error occurs the function will return 0 otherwise it will return -1. Writing to address 0 performs an encoder reset function which is used as part of a position shift function:

Example:

```
` Set axes to parameter mode in a pair (PCI 208)
```

```
ENCODER_CONTROL AXIS(0)=1
```

```
ENCODER_CONTROL AXIS(1)=1
```

```
` Set value to shift by and perform reset:
```

```
ENCODER_WRITE($A700,$2000) AXIS(0)
```

```
ENCODER_WRITE($A701,0) AXIS(0)
```

```
ENCODER_WRITE(0,0) AXIS(0)
```

Error Handling:

When the axis receives a data packet from the with a CRC error, or fails to receive a data packet from the encoder, it will set bit 2 of the AXISSTATUS axis parameter. If subsequent packets are received successfully the bit will be cleared.

If bit 2 of the ERRORMASK axis parameter is set the encoder data error will trip the WDOG enable relay.