



# Pulse & Direction Output Command



**Application Note**



[www.agito-akribis.com](http://www.agito-akribis.com)

Member of Akribis Systems group



## Revision History

| Version | Description      | Date       |
|---------|------------------|------------|
| 1.0     | Initial release. | 2 Oct 2023 |

## Contact Information

|              |   |
|--------------|---|
| Manufacturer | Agito Akribis Systems Ltd., Member of Akribis Systems Group |
| Address      | 56 Serangoon North Avenue 4, Singapore 555851               |
| Email        | info@akribis-sys.com  |
| Website      | www.agito-akribis.com                                       |

## Copyright Notice

©2026 Agito Akribis Systems Ltd.

All rights reserved. This work may not be edited in any form or by any means without written permission of Agito Akribis Systems Ltd.

## Products Rights

AGDx, AGCx, AGMx, AGAx, AGIx, and AGLx are products designed by Agito Akribis Systems Ltd. in Israel.

Sales of the products are licensed to Akribis Systems Pte Ltd. under intercompany license agreement.

Agito Akribis Systems Ltd. has full rights to distribute above products worldwide.

## Disclaimer

This product documentation was accurate and reliable at the time of its release.

Agito Akribis Systems Ltd. reserves the right to change the specifications of the product described in this manual without notice at any time.

## Trademarks

Agito PCSuite is a trademark of Agito Akribis Systems Ltd.

## Contents

|     |                                   |    |
|-----|-----------------------------------|----|
| 1   | About this Application Note       | 4  |
| 1.1 | Introduction to Pulse & Direction | 4  |
| 2   | Wiring Interface                  | 5  |
| 2.1 | AGL101 PD Output Pinouts          | 5  |
| 2.2 | AGL102 PD Output Pinouts          | 5  |
| 2.3 | AGC301/AGD301 PD Output Pinouts   | 6  |
| 2.4 | AGC300/AGD200 PD Output Pinouts   | 6  |
| 3   | Configuration Setup (via PCSuite) | 7  |
| 3.1 | PD Output Settings                | 7  |
| 3.2 | Feedback Settings                 | 8  |
| 3.3 | IO Mode Functionalities           | 9  |
| 4   | Motion                            | 11 |
| 5   | PD Output Limitations             | 12 |
| 6   | PD Output Relevant Keywords       | 13 |

# 1 About this Application Note

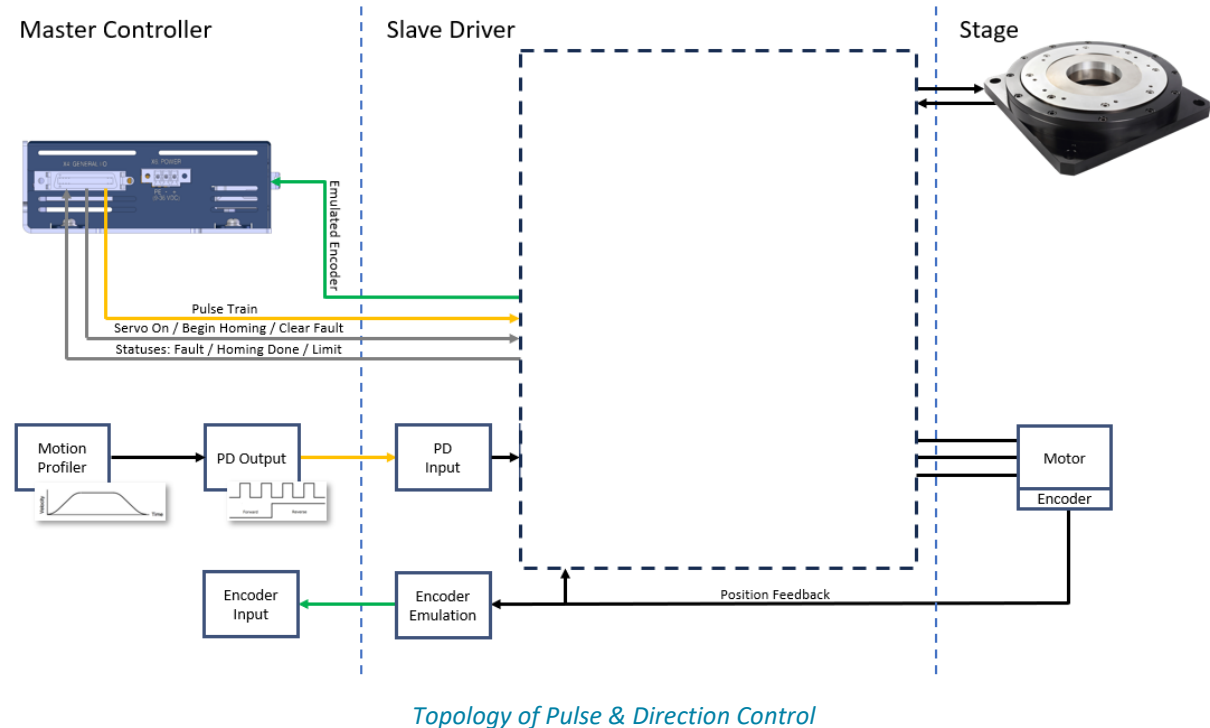
This application note explains how to setup and configure the Agito controller as a master that does the motion profiling, and outputs pulse and direction position command to a 3<sup>rd</sup> party driver.

The following topics will be covered in this document,

- PD Output wiring interface summary for Agito products
- IO Mode functionality for communication with a 3<sup>rd</sup> party slave driver
- PCSuite configuration steps and settings related to PD Output
- Motion and observation
- Relevant keywords summary

## 1.1 Introduction to Pulse & Direction

The Pulse & Direction command is a traditional method to decouple the controller that does the profiling, from the amplifier/driver that drives the motor. With this, controllers with profiling algorithm for specific applications (E.g., CNC Machine or Laser cutter) can be used with any stage system that supports this mode of command. Therefore, allowing for flexibility in pairing of drivers of different sizes and types (E.g., 3-phase servo/2-phase stepper/5-phase stepper drives).



## 2 Wiring Interface

The PD Output functionality is product dependent. The following tables depict how the PD Output functionality is assigned to the outputs.

These are partial wiring descriptions, be sure to wire up other relevant lines such as ground for the differential IO. Please refer to the relevant Product Manual for more information regarding the hardware and circuitry.

### 2.1 AGL101 PD Output Pinouts

For AGL101, the PD Output functionality can be flexibly assigned to any of the outputs. It is suggested to use the first 2 differential outputs for PD output.

| Output No.    | Type                        | Connector    | Function               |
|---------------|-----------------------------|--------------|------------------------|
| 1             | Optically-isolated          | X3: Pin 3    | Pulse/Direction Output |
| 2             | Optically-isolated          | X3: Pin 12   | Pulse/Direction Output |
| 3             | Optically-isolated          | X3: Pin 11   | Pulse/Direction Output |
| 4             | Non-isolated open collector | X2: Pin 4    | Pulse/Direction Output |
| 5             | Non-isolated open collector | X2: Pin 17   | Pulse/Direction Output |
| 6             | Non-isolated open collector | X3: Pin 3    | Pulse/Direction Output |
| 7 (Suggested) | Differential                | X2: Pin 7,20 | Pulse/Direction Output |
| 8 (Suggested) | Differential                | X2: Pin 6,19 | Pulse/Direction Output |
| 9             | Differential                | X2: Pin 7,16 | Pulse/Direction Output |

### 2.2 AGL102 PD Output Pinouts

For AGL102, the PD Output functionality can be flexibly assigned to any of the outputs. AGL102 only has non-isolated open collector outputs.

| Output No.    | Type                        | Connector  | Function               |
|---------------|-----------------------------|------------|------------------------|
| 1 (Suggested) | Non-isolated open collector | X3: Pin 7  | Pulse/Direction Output |
| 2 (Suggested) | Non-isolated open collector | X3: Pin 19 | Pulse/Direction Output |
| 3             | Non-isolated open collector | X3: Pin 6  | Pulse/Direction Output |
| 4             | Non-isolated open collector | X3: Pin 18 | Pulse/Direction Output |



#### Note

The AGL102 is designed as a low-cost analog command adapter. While it is technically possible to use the AGL102 to do PD Output, the low frequency limitation from using slow non-isolated outputs will make it impractical for most applications.

## 2.3 AGC301/AGD301 PD Output Pinouts

For AGC301/AGD301, the PD Output functionality is tied to specific outputs. This is necessary to support multiple axes.

| Output No.            | Type         | Connector     | Function                  |
|-----------------------|--------------|---------------|---------------------------|
| <b>18 (Diff IO 1)</b> | Differential | X4: Pin 1,2   | A Axis – Pulse Output     |
| <b>19 (Diff IO 2)</b> | Differential | X4: Pin 19,20 | A Axis – Direction Output |
| <b>20 (Diff IO 3)</b> | Differential | X4: Pin 3,4   | B Axis – Pulse Output     |
| <b>21 (Diff IO 4)</b> | Differential | X4: Pin 21,22 | B Axis – Direction Output |
| <b>22 (Diff IO 5)</b> | Differential | X4: Pin 25,26 | C Axis – Pulse Output     |
| <b>23 (Diff IO 6)</b> | Differential | X4: Pin 9,10  | C Axis – Direction Output |

## 2.4 AGC300/AGD200 PD Output Pinouts

For AGC300, AGD200, the PD Output functionality is tied to specific outputs. This is necessary to support multiple axes.

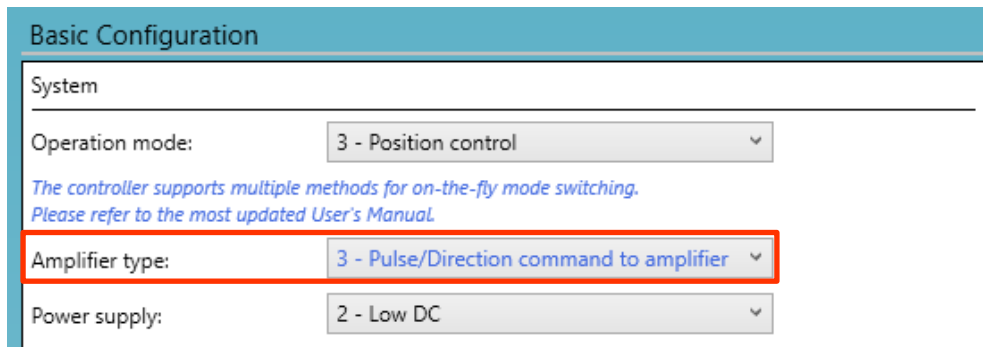
| Output No. | Type         | Connector      | Function                  |
|------------|--------------|----------------|---------------------------|
| <b>5</b>   | Differential | X10: Pin 21,22 | A Axis – Pulse Output     |
| <b>6</b>   | Differential | X10: Pin 23,24 | A Axis – Direction Output |
| <b>7</b>   | Differential | X10: Pin 25,26 | B Axis – Pulse Output     |
| <b>8</b>   | Differential | X10: Pin 27,28 | B Axis – Direction Output |

## 3 Configuration Setup (via PCSuite)

### 3.1 PD Output Settings

The steps below explain how to configure the driver for PD Output operation.

- Under the  tab, navigate to the  page. Set the amplifier type to “**3 – PD Command to 3rd party drive**”.



**Basic Configuration**



System

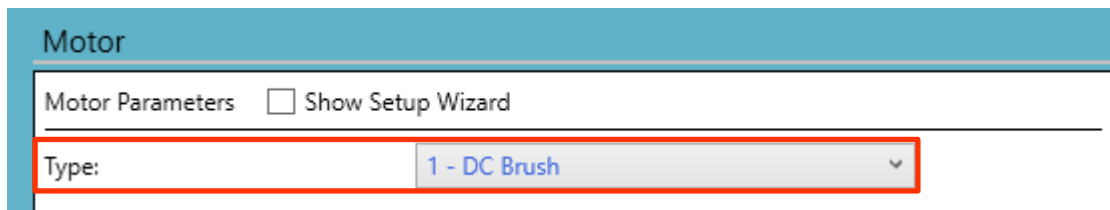
Operation mode: 3 - Position control

*The controller supports multiple methods for on-the-fly mode switching. Please refer to the most updated User's Manual.*

Amplifier type: 3 - Pulse/Direction command to amplifier

Power supply: 2 - Low DC



- Under the  tab, navigate to the  page. Set the motor type to “**1 – DC Brush**” or “**2 – Voice Coil**” or “**5 – Stimulation**”. This is to indicate to the master controller that no commutation is required on its side. If commutation is required, it will be handled by the slave driver.

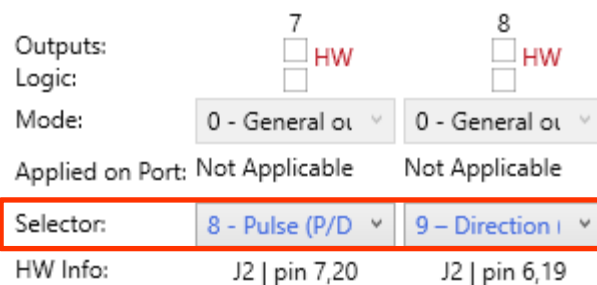


**Motor**

Motor Parameters  Show Setup Wizard

Type: 1 - DC Brush

- Under the  tab, navigate to the  page, **Discrete Outputs** tab.  
For Central-I products (AGL101/102), configure the selector of the outputs to “**8 – Pulse (P/D)**” and “**9 – Direction (P/D)**” according.



Outputs: 7  HW 8  HW

Logic:

Mode: 0 - General ou 0 - General ou

Applied on Port: Not Applicable Not Applicable



Selector: 8 - Pulse (P/D) 9 - Direction (P/D)

HW Info: J2 | pin 7,20 J2 | pin 6,19

For standalone products (AGC300, AGC301, AGD200, AGD301), configure the selector of the outputs to “**13 – x Enc. Emul. Line x**”. Although the naming of the selector is different, they both achieve the same functionality.

|                  |                             |                             |
|------------------|-----------------------------|-----------------------------|
| Outputs:         | 18                          | 19                          |
| Logic:           | <input type="checkbox"/> HW | <input type="checkbox"/> HW |
| Mode:            | 0 - General ot              | 0 - General ot              |
| Applied on Axis: | Not Applicable              | Not Applicable              |
| Selector:        | 13 - A Enc. en              | 13 - A Enc. en              |
| Direction:       | 1 - Output                  | 1 - Output                  |
| HW Info:         | X4   pin 1,2                | X4   pin 19,20              |



- Under the  tab, navigate to the  page. Configure the virtual encoder source to use “APosRef”, or “BPosRef” if you are configuring for B axis. The virtual encoder feature is used emulate the profiled position (PosRef).

Configure the output type as necessary. PD Output and AqB are supported types.

The scaling factor allows the user to scale the output. However, in PD output mode, the user controls the profiler, so there is no real need to use a factor that is not 1.

### Virtual encoder

Configuration

Virtual encoder: 1 - Enable

|       |           |          |
|-------|-----------|----------|
| Axis: | Category: | Keyword: |
| A     | Position  | PosRef   |

VEnc src.: APosRef

Output type: 0 - Pulse/Directio



|         |        |   |        |
|---------|--------|---|--------|
| Factor: | 65,536 | / | 65,536 |
|---------|--------|---|--------|

Delay: 0 micro-sec

## 3.2 Feedback Settings

The steps below explain how to setup the controller to receive position feedback from the slave driver.



- Under the  tab, navigate to the  page. If the slave driver provides for emulated position feedback, connect it to the encoder port as if it were an encoder. Pulse direction feedback will be supported as a future feature.

Feedbacks

---

Dual loop

Dual Loop mode: 0 - No dual loop ?

Main encoder      Reading: 0      Last index: 0

---

Type: 1 - Incremental ?

*Please properly configure max. speed limitation at Config/Pos window*

Sub Type: 0 - A quad B encoder ?

Resolution: 1 counts / rotation (or pitch) ?

Invert direction: 0 - No      Max. speed: 0 - 100,000,000 ?

Modulus range: 0 user-units

Emulation divider value: 0      Direction: Normal ?

User units' factor: 65,536 / 65536      counts/user-units

### 3.3 IO Mode Functionalities

The steps below explain how to setup the IO's to communicate the master controller's commands and to read the slave driver's status.

- 
 Under the **I/O** tab, navigate to the  page, **Discrete Inputs** tab. Wire and configure the input signals as necessary.

**Fault (ext. drive) trigger** – if this input is high while in motor-on state, a fault will be triggered.

This signal is usually connected to the **Fault/Alarm status** from the slave driver. If the slave driver meets a fault, the master controller will be alerted.

Inputs: 1

Logic: 1

Mode: 24 - Fault (ext)

Applied on Axis: A

HW Info:      A3 | pin 1

- 
 Under the **I/O** tab, navigate to the  page, **Discrete Outputs** tab. Wire and configure the output signals as necessary.

**Motor On status** – if motor is enabled, the output will be high.

This signal is usually connected to the **Motor On & Begin trigger** input on the slave driver side. When the master controller is enabled, the output will go high. In turn, the slave driver will be triggered to do motor-on and begin.

|                  |                          |
|------------------|--------------------------|
| Outputs:         | 1                        |
| Logic:           | <input type="checkbox"/> |
| Mode:            | 2 - Motor On             |
| Applied on Axis: | A                        |
| Selector:        | 0 - Software [           |
| Sink/Source:     | 0 - Sink                 |
| HW Info:         | A3   pin 4               |

**Fault/Alarm status** – if there is a fault, the output will be high.

This signal is usually **inverted** and connected to the **Reset/Clear Fault/Alarm trigger** on the slave driver side. When a fault occurs, this output will drop low due to the inversion of logic. Doing “ConFlt=0” will clear the error on the master controller. In turn, this output will become high, and trigger the to the slave driver to reset the faults on its side too.

|                  |                                     |
|------------------|-------------------------------------|
| Outputs:         | 1                                   |
| Logic:           | <input checked="" type="checkbox"/> |
| Mode:            | 9 - Fault/Alarr                     |
| Applied on Axis: | A                                   |
| Selector:        | 0 - Software [                      |
| Sink/Source:     | 0 - Sink                            |
| HW Info:         | A3   pin 4                          |



**Note**

Some of the discrete inputs/outputs’ special functionalities assignments (“Mode”) will operate properly only after power off/on or reset of the controller. This is mainly relevant for the case of turning off a special functionality. It might be still functional until the controller is powered off/on or reset.

Once setting up all special functionalities of the discrete inputs/outputs, apply them, save the new settings to the Flash, and consider resetting or power-cycling the controller.

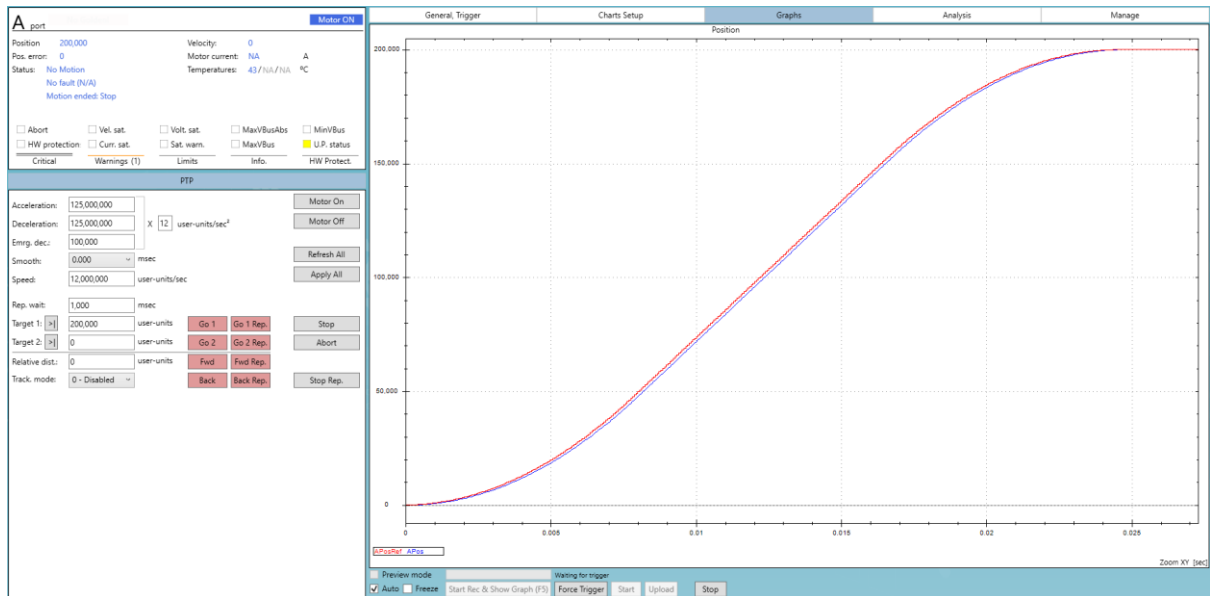
## 4 Motion

The step below explains how to execute a motion in PD Output mode (3<sup>rd</sup> party amplifier).



- Under the **MOTION** tab, navigate to the **PTP** page. The operation controls are no different from a standard controller with in-built amplifier. Do motor-on and make a motion.

The graph below is a recording of PosRef and Pos. The profiler generates the position command (PosRef) and feeds it to the slave driver. The position feedback is then emulated from the slave driver and read by the master (Pos).



### Note

The operating principal of the virtual encoder feature is to take the delta of the selected parameter (PosRef in this case) and output this delta in the next interrupt cycle. This creates an inherent lag of 61us between the PosRef and Pos.

If the slave driver has a delay in processing the emulated encoder, this will be added to the 61us. Since the position loop is not closed on the master controller, the delay will not affect performance.

## 5 PD Output Limitations

---

The differential output is limited to a maximum frequency of 5MHz. Using PD output format, the speed is limited to 5,000,000 counts/s. If AqB format is used, the speed limitation will be increased by a factor of 4 to 20,000,000 counts/s.

While internal tests have shown that it is possible to operate the feature at higher speeds of up to 9,990,144Hz, it is recommended to use output frequencies up to 5MHz.

## 6 PD Output Relevant Keywords

| Keyword           | Description  |
|-------------------|--|
| <b>AmpType</b>    | <p>AmpType specifies the amplifier type.</p> <p><b>AmpType = 3</b> sets the controller to <b>PD Command to amplifier</b> mode. In this mode, the control loops are disabled. The motion profiler generates PosRef which can be emulated out as a command.</p> <p>To use the motion controller in PD Output mode, select the amplifier mode to be PD Command to amplifier.</p>  |
| <b>MotorType</b>  | <p>MotorType specifies the motor type.</p> <p><b>MotorType = 1</b> sets the type to <b>DC Brush motor</b>. With this motor type, no commutation is required.</p> <p><b>MotorType = 2</b> sets the type to <b>Voice Coil motor</b>. With this motor type, no commutation is required.</p> <p>To use the motion controller in PD Output mode, select the motor type to be DC Brush or Voice Coil to avoid commutation. Commutation will be done at the slave driver.</p>             |
| <b>EncType</b>    | <p>EncType specifies the encoder type.</p> <p><b>EncType = 1</b> sets the type to <b>Incremental encoder</b>.</p> <p>Typically, the emulated feedback from the slave driver is incremental. Ignore this parameter if no feedback is provided back to the master controller.</p>  |
| <b>EncSubType</b> | <p>EncSubType specifies the sub-type of the encoder input.</p> <p><b>EncSubType = 0</b> sets the sub-type to <b>quadrature (AqB)</b> mode.</p> <p><b>EncSubType = 1</b> sets the sub-type to <b>pulse and direction (PD)</b> mode.</p> <p><b>EncSubType = 2</b> sets the sub-type to <b>bit (C0/C1)</b> mode.</p> <p><b>EncSubType = 3</b> sets the sub-type to <b>Up/Down (CW/CCW)</b> mode.</p> <p>EncSubType = 1, 2, 3 are future features and not supported at the moment.</p> |
| <b>EncDir</b>     | <p>EncDir specifies the direction of the encoder.</p> <p><b>EncDir = 0</b> sets the direction to <b>not-inverted</b>.</p> <p><b>EncDir = 1</b> sets the direction to <b>inverted</b>.</p> <p>Ideally, the encoder direction between the master and slave should be aligned, so the direction of the encoder feedback should not be inverted.</p>   |

| Keyword            | Description  |
|--------------------|--|
| <b>VEncOn</b>      | <p>VEncOn enables/disables the virtual encoder feature.</p> <p><b>VEncOn = 0</b> disables the virtual encoder feature.<br/> <b>VEncOn = 1</b> enables the virtual encoder feature.</p> <p>To use the motion controller in PD Output mode, virtual encoder feature shall be enabled.</p>  |
| <b>VEncSrc</b>     | <p>VEncSrc specifies the keyword/parameter to emulate.</p> <p><b>VEncSrc = Complex CAN Code (CCC)</b> of parameter to be emulated.</p> <p>To use the motion controller in PD Output mode, VEncSrc is assigned the CCC of PosRef for the relevant axis.<br/> For ease, use PCSuite to compose the CCC. Alternatively, refer to the CCC Definition document for more information on how to compose CCC manually.</p> |
| <b>VEncType</b>    | <p>VEncType specifies the output format of the virtual encoder feature.</p> <p><b>VEncType = 0</b> sets the sub-type to <b>pulse and direction (PD)</b> mode.<br/> <b>VEncType = 1</b> sets the sub-type to <b>quadrature (AqB)</b> mode.</p> <p>Typically, pulse and direction mode is used. However, quadrature mode supports four times the max frequency of pulse and direction mode.</p>                      |
| <b>VEncFact</b>    | <p>VEncFact scales (multiplies) the emulated output by a factor.</p> <p><b>Default value: 65,536</b><br/> Max value: 16,777,215<br/> Min value: -16,777,215 (A negative value inverts the direction)</p>   |
| <b>VEncFactDen</b> | <p>VEncFactDen scales (divides) the emulated output by a factor.</p> <p><b>Default value: 65,536</b><br/> Max value: 500,000,000<br/> Min value: 1</p>   |
| <b>VEncDelay</b>   | <p>VEncDelay is applicable only if VEncType is set to pulse and direction (PD) mode.<br/> VEncDelay specifies the delay (in micro-seconds) to add in between the switching of directions.</p> <p><b>Default value: 0</b><br/> Max value: 25<br/> Min value: 0</p>  |

