Handwheel

Application Note

agito

www.agito-akribis.com

Member of Akribis Systems group

Revision History

Version	Description	Date
1.0	Initial release.	24 July 2026

Contact Information

Manufacturer	Agito Akribis Systems Ltd., Member of Akribis Systems Group
Address	56 Serangoon North Avenue 4 Singapore 555851
Telephone	+65 6484 3357
Website	www.agito-akribis.com

Copyright Notice

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..

Warranty

This product is warranted to be free of defects in material and workmanship and conforms to the specifications listed in this manual, for a period of 12 months from the shipment date from factory.

Contents

1 Introduction

1.1 About This Manual

Thank you for choosing an Agito motion control product. We are committed to providing you with the ultimate motion control solution for speed and precision, together with full technical support.

This manual mainly describes how to use a handwheel together with an Agito motion controller. For other parameter settings, please refer to the detailed descriptions in the “Agito Quick Start Guide”; they are not repeated here.

Note that handwheel-related operations should only be carried out after the basic parameter settings have been completed and the motor has been confirmed to run normally.

1.2 Handwheel Overview

An electronic handwheel (Manual Pulse Generator, MPG), also known as a manual pulse generator, is commonly referred to as a “handwheel” or “MPG.” It uses the pulse signal generated by a hand-cranked pulse encoder to control the motion of each servo axis. The pulse frequency and pulse count control the speed and displacement of the axis motion, respectively.

Figure 1. Handwheel

2 Handwheel Operating Procedure

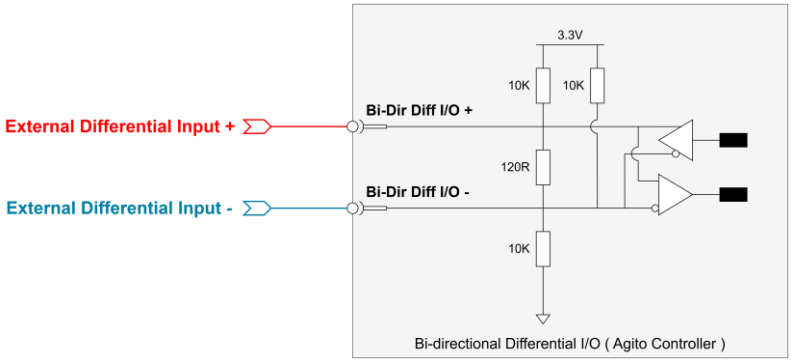
To use the handwheel, wire it to the controller according to the correct wiring scheme, then enable PD mode in PCSuite. Turning the handwheel then sends pulses to the motor to drive it. Before performing handwheel-related operations, make sure the relevant motor parameters have been configured correctly and tuned, and that the motor can move normally.

2.1 Handwheel Wiring

1. Wiring Principle

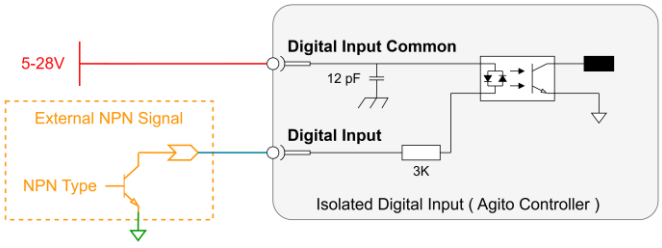
Since handwheel wiring only involves differential input signals and single-ended input signals, only these two signal types are described here.

(1) Differential digital input

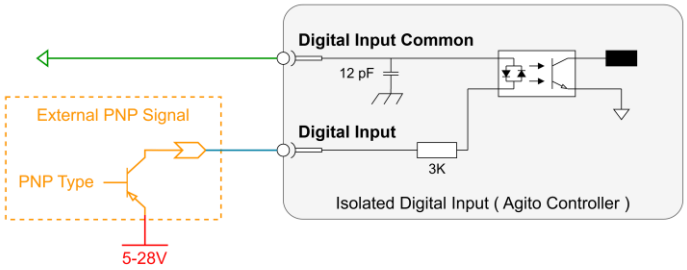


(2) Single-ended digital input

① External NPN type (external sink type):



② External PNP type (external source type):



2. Wiring Connection Description

AGD301 Pin	Name	Function	Handwheel Name
7 (X4)	5V	Indicator	Indicator L+ (also tied to 5V)
		handwheel power supply	handwheel power supply 5V
8 (X4)	GND	Indicator	Indicator L– (also tied to 0V)
		handwheel power supply	handwheel power supply 0V
1 (X4)	Differential Input 1+	Pulse generation	A+
2 (X4)	Differential Input 1-		A-
19 (X4)	Differential Input 2+		B+
20 (X4)	Differential Input 2-		B-
9 (A3)	5V	Common terminal for handwheel single-ended signals	Common terminal
1 (A3)	Digital Input 1	Axis-select signal	X
12 (A3)	Digital Input 2		Y
2 (A3)	Digital Input 3		Z
13 (A3)	Digital Input 4	Multiplier signal	X1
3 (A3)	Digital Input 5		X10
14 (A3)	Digital Input 6		X100
11 (A3)	Digital Input Common (tied to A3-Pin10 GND)	Controller single-ended signal common terminal	—
10 (A3)	GND		

Notes:

① Both the handwheel indicator L+ and the pulse generator 5V are connected to the controller's 5V pin.

② Both the handwheel indicator L- and the pulse generator 0V are connected to the controller's GND pin.

③ The pulse generator's A+/A-/B+/B- are connected respectively to the controller I/O's differential inputs 1+/1-/2+/2-.

④ To enable axis switching, connect the handwheel's A+/A-/B+/B- signals in parallel to the differential inputs of the other axes — for example, differential inputs 3+/3-/4+/4- for B axis, and 5+/5-/6+/6- for C axis.

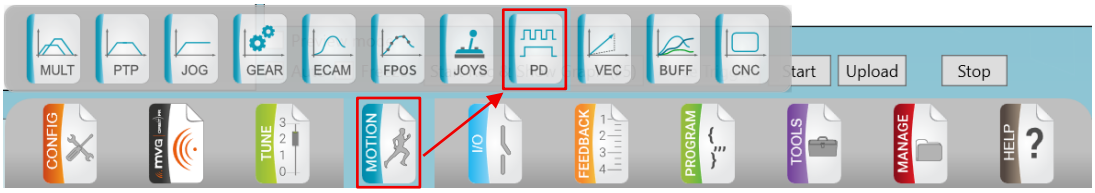
⑤ The digital input pins have no fixed assignment; the axis-select and multiplier signals may be wired to any digital input pins according to personal preference, provided the digital input common terminal is wired correctly.

⑥ If only pulse generation from the handwheel is required, it is sufficient to connect 5V, 0V, and A+/A-/B+/B-.

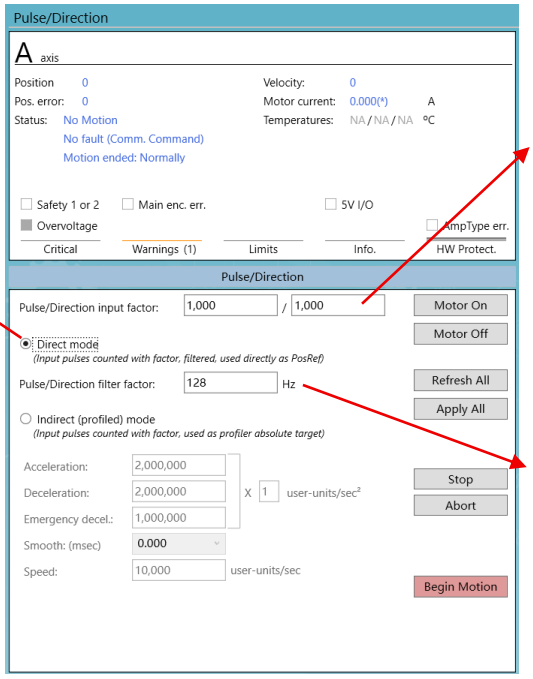
2.2 Driving the Motor via PD Mode

1. Operating Procedure

① Click **MOTION** -> **PD** button to enter the Pulse/Direction screen.



Direct mode: this is the mode normally used for the handwheel.



Pulse/Direction input factor (numerator/denominator): the displacement produced by one positioning pulse output by the controller. The numerator may be negative.

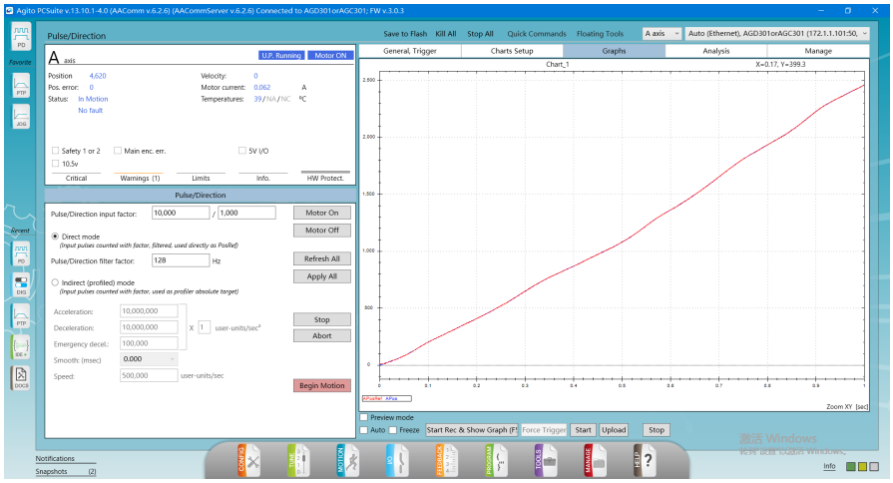
Pulse/Direction filter factor: defines the first-order filter frequency applied to the pulse and direction inputs (the keyword shown may differ between firmware versions).

② Click **Begin Motion** to enable PD mode.

③ At this point, turning the handwheel drives the motor via the pulse/direction signal.

Note

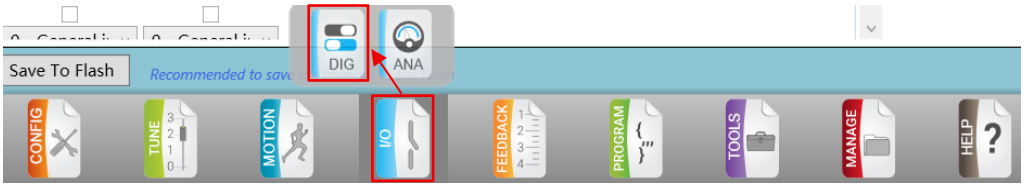
For detailed information on PD mode features, please refer to the “Pulse & Direction Input Command” application note.



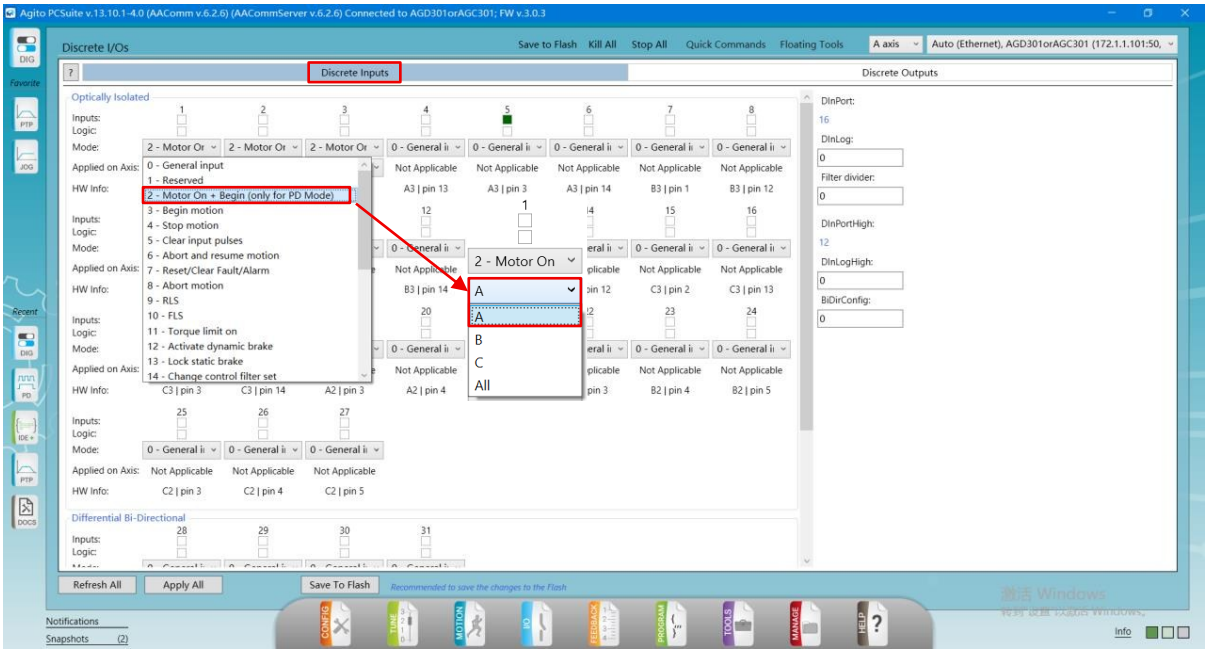
2. Axis-Select Signal I/O Configuration

After wiring the axis-select signal correctly between the handwheel and the controller, the I/O must be configured before the axis-select function can be used.

Click **I/O** → **DIG** to open the Discrete I/O window.

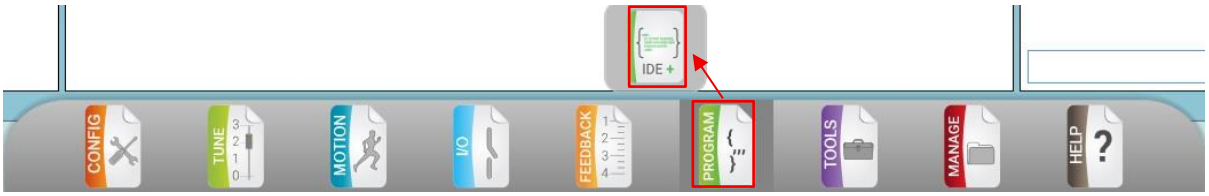


In the I/O configuration screen, set the pin used by the handwheel to send the axis-select signal (per the wiring scheme above, digital inputs 1, 2, 3) to Mode "**2 – Motor On + Begin (only for PD Mode),**" then set "Applied on Axis" to the axis to be switched to. The handwheel can then be used to perform normal axis switching.



3. Multiplier Signal Configuration

After wiring the axis-select signal correctly between the handwheel and the controller, IDE programming is required before the multiplier function can be used normally.



Per the wiring connection described earlier, the axis-select signals are digital inputs 1, 2, 3, and the multiplier signals are digital inputs 4, 5, 6. When the controller receives the corresponding signal, the IDE program changes it to the corresponding multiplier within the software.

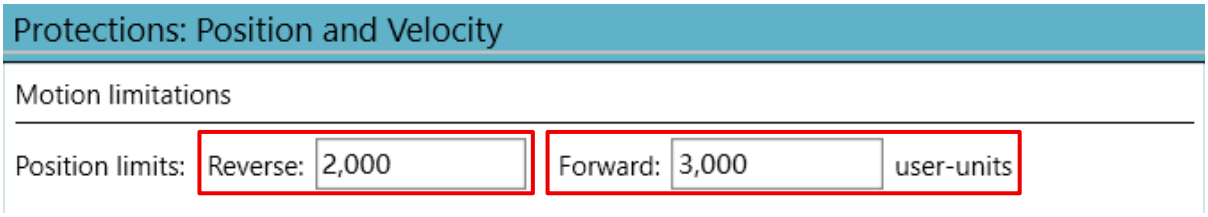
IDE program implementing this function:

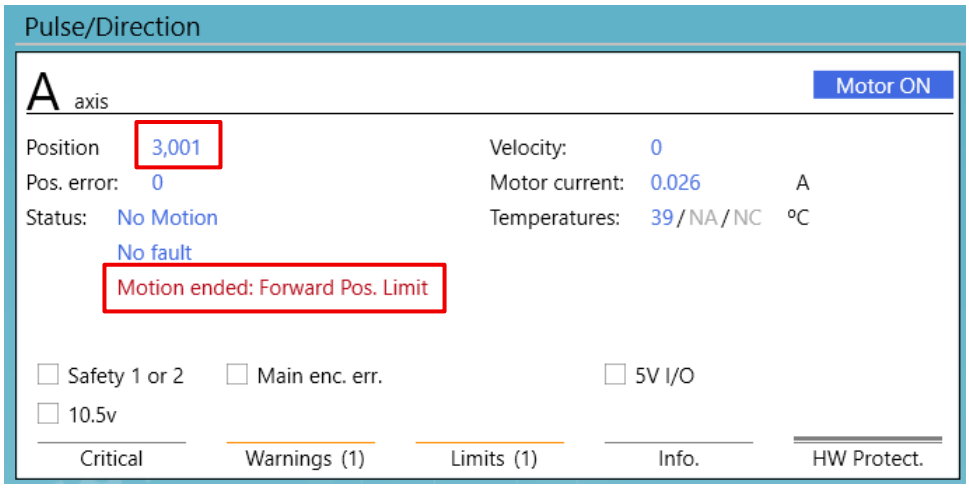
```
main([10,30],[5,20],[800,1000])
//
while(1)
        AGenData[456]=ADInPort & 63
        switch (AGenData[456])
            case 9
                APDFact=1000
        break
            case 17
                APDFact=10000
        break
            case 33
                APDFact=100000
        break
            case 10
                BPDFact=1000
        break
            case 18
                BPDFact=10000
        break
            case 34
                BPDFact=100000
        break
            case 12
                CPDFact=1000
        break
            case 20
                CPDFact=10000
        break
            case 36
                CPDFact=100000
        break
    end
end
//
endofmain
```

3 Handwheel Limit Handling

When software limits or limit-switch limits (photoelectric switch limits) have been configured, driving the motor past a limit using the handwheel will cause the motor to fault and become unable to move in the reverse direction.

1. Software Limits





To resolve this situation, an IDE program can be written to clear the fault and allow motion in the reverse direction.

```
main([10,30],[5,20],[800,1000])
//
while (1)
    if ((AMotionMode==3) && (AMotorOn==1))
        if ((AMotionMode==3) && (AMotionReason==6) && (AMotionStat==0))
            AMotionMode=1
            ARelTrgt=0
            AAbsTrgt=ARevPLim
            ABegin
            while (AMotionStat!=0)
            end
            AMotionMode=3
            ABegin
        else if ((AMotionMode==3) && (AMotionReason==7) && (AMotionStat==0))
            AMotionMode=1
            ARelTrgt=0
            AAbsTrgt=AFwdPLim
            ABegin
            while (AMotionStat!=0)
            end
            AMotionMode=3
            ABegin
        end
    end
end
//
endofmain
```

2. Limit Switch Limits (Photoelectric Switch Limits)

Similarly, when using limit switches, the following IDE program can be used to resolve the issue of moving in the reverse direction after reaching a limit.

```
main([10,30],[5,20],[800,1000])
//
while (1)
    if ((AMotionMode==3) && (AMotorOn==1))
        if ((AMotionMode==3) && (AMotionReason==4) && (AMotionStat==0))
            AMotionMode=1
            ARelTrgt=0
            AAbsTrgt=ARevPLim
            ABegin
            while (AMotionStat!=0)
            end
            AMotionMode=3
            ABegin
        else if ((AMotionMode==3) && (AMotionReason==5) && (AMotionStat==0))
            AMotionMode=1
            ARelTrgt=0
            AAbsTrgt=AFwdPLim
            ABegin
            while (AMotionStat!=0)
            end
            AMotionMode=3
            ABegin
        end
    end
end
//
endofmain
```

4 Keywords

Keyword	Description
PDFact	PDFact is a coefficient multiplied by the pulse/direction input count before use. Together with PDFactDen it determines Reference.
PDFactDen	PDFactDen divides the pulse/direction input count before use. $Reference = [Number\ of\ input\ pulses] * \frac{PDFact}{PDFactDen}$
PDFiltFact	PDFiltFact is the parameter defining the bandwidth of the pulse/direction first-order filter. (Typical value: 0.05) (In some firmware versions this keyword is shown as PDPosFilt.)
PDPosFilt	PDPosFilt is the parameter defining the bandwidth of the pulse/direction first-order filter. (Typical value: 128 Hz) (In some firmware versions this keyword is shown as PDFiltFact.)
MotionMode	PD mode is enabled when MotionMode = 3.