PEG & Position Lock

Application Note

agito

www.agito-akribis.com

Member of Akribis Systems group

Revision History

Version	Description	Date
1.0	Initial Release	3 December 2025

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 Agito motion control products.
We are committed to providing advanced motion control solutions that combine high speed and precision, supported by comprehensive technical service.

This manual focuses on the use of PEG (Position Event Generation) and Position Lock (Probe) features of the Agito motion controller. Only configurations related to PEG and Position Lock are described in detail.

For other parameter settings, please refer to the *Agito Quick Start Guide* and related documents.

1.2 Overview

PEG (Position Event Generation), also known as “Fly Capture,” outputs an I/O signal when the motor reaches a preset position. The output can be a single-ended or differential digital signal, which can be used to trigger cameras, lasers, relays, cutters, etc.

Position Lock (Probe) captures the encoder position upon receiving an external event input.
The trigger source can be an encoder Index (Z) signal, a single-ended or differential digital input, allowing users to capture position data corresponding to physical signal events.

Note

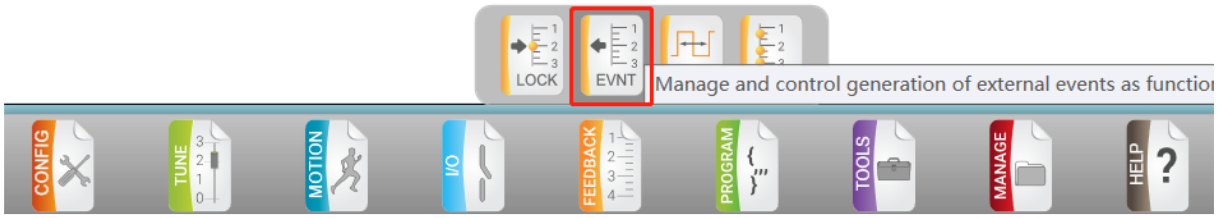
Only the AGM800-CI model supports enabling PEG and Position Lock simultaneously. Other AGC and AGD series controllers can only use one function at a time—enabling one will automatically disable the other.

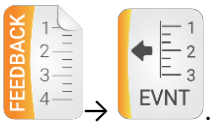
2 PEG Function

Before using PEG or Position Lock, ensure that:

- The motor operates normally according to your motion requirements.
- The required I/O signals are correctly connected to the corresponding controller ports.

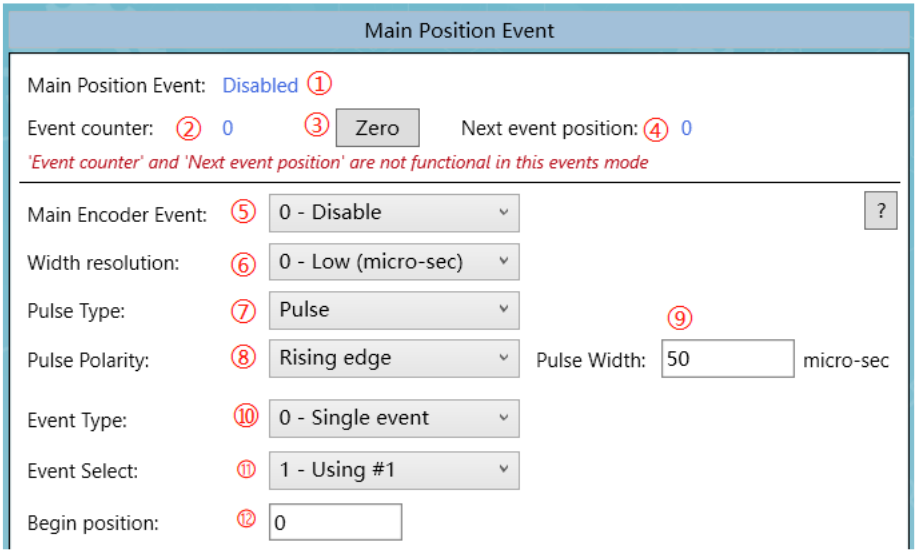
2.1 Parameter Configuration Interface



Navigate to 

→ .

PEG offers multiple output modes such as equal interval or table-based, supporting both single-ended and differential outputs.

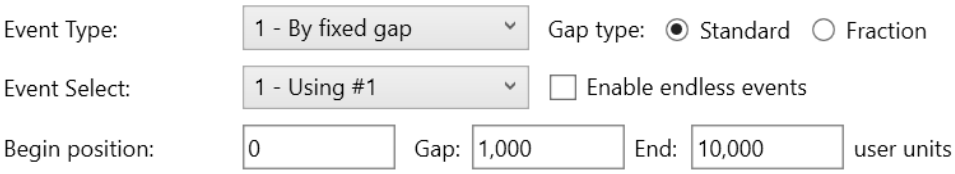


① *Main Position Event:* displays current event status (Disabled / Enabled).
② *Event Counter:* [*EventCntr*], shows number of event signals already output.
③ *Zero:* resets Event Counter to zero.
④ *Next Event Position:* displays encoder position for the next event output.
⑤ *Main Encoder Event:* [*EventOn*], 0 = disable Event Output; 1 = enable Event Output.
⑥ *Width Resolution:* [*EventPulseRes*], pulse width resolution: 0 = µs, 1 = ns.
⑦ *Pulse Type:* [*EventPulseWid*], output type, *Pulse / Toggle*.
⑧ *Pulse Polarity:* Rising edge / Falling edge, displayed only when Pulse Type = Pulse.
⑨ *Pulse Width:* [*EventPulseWid*], output pulse width — determined by items ⑥⑦⑧⑨.

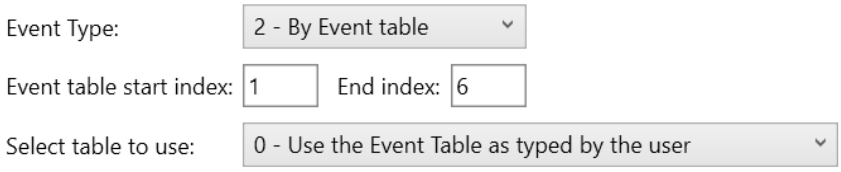
⑩ *Event Type:* 0 – Single Event / 1 – By Fixed Gap / 2 – By Event Table.

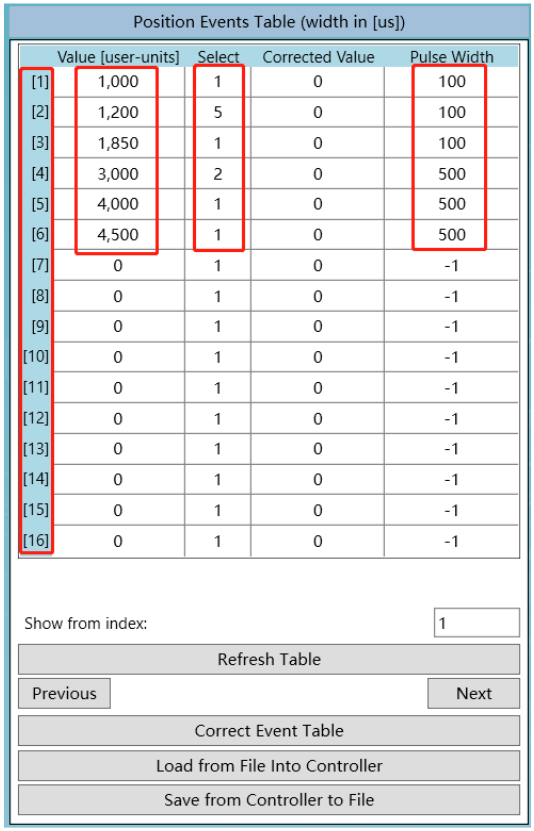
- ***0 – Single Event：***

- ***1 – By Fixed Gap:***



- ***2 – By Event Table：***

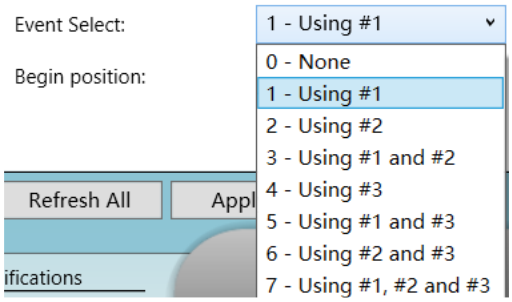




Event Table Output Interface:

- *Pulse Width: EventTableWid[x]*
- *Select: EventTableSel[x],* 7 options of output channel to choose from
- *Value: EventTable[x],* event output position
- *Event Output Position Index Number:* up to 100, max display 16 items per page.
- *Show from index:* select the starting index number to display the table and use it in conjunction with Refresh Table.
- *Refresh Table:* Refresh the table context.
- *Previous/Next:* Table pagination.
- *Correct Event Table:* [*EventCorrect*], add the error compensation value to the position (only use when *EventTableSrc*=1).
- *Load from File Into Controller:* import local table files (*.aef format).
- *Save from Controller to File:* export the event table to local file (*.aef format).

⑪ *Event Select*: choose PEG output signal channel.



⑫ *Begin Position*: the begin position where the event is triggerred.

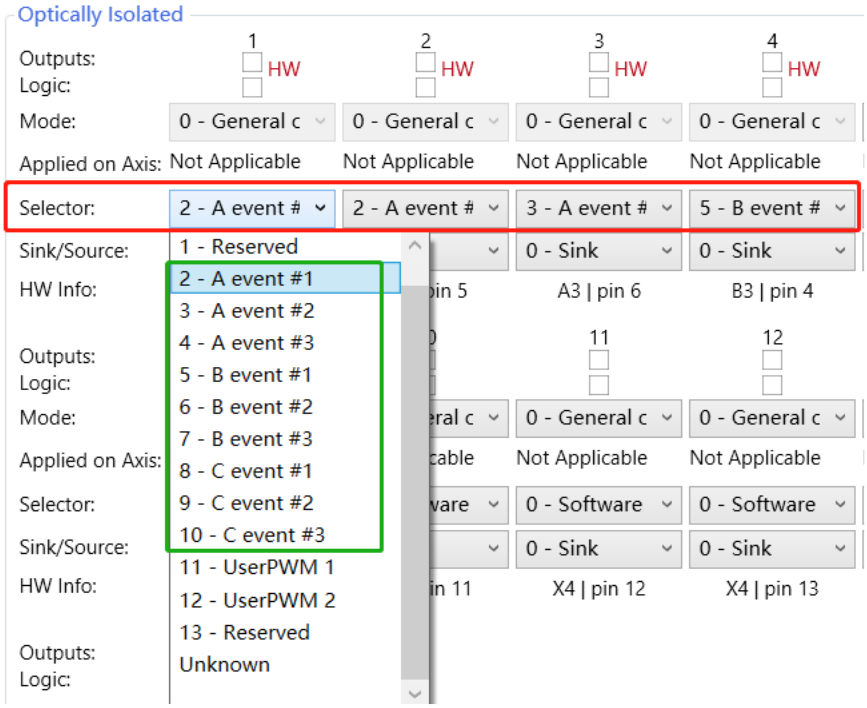
2.2 Operation Steps

Step 1: I/O Wiring

Connect the required digital output (single-ended or differential) as your application.
Refer to the *Agito Quick Start Guide* and the specific controller manual for wiring details.

Step 2: Bind Event Channel

Go to *Discrete* → *Discrete Outputs* and bind the digital output port to an axis event number (As shown in the figure, DO_1 is bound to A-axis #1 event, DO_2 is also bound to A-axis #1 event, DO_3 is bound to A-axis #2 event, and DO_4 is bound to B-axis #1 event).
Up to 3 event numbers can be used. The same event number can be reused across multiple outputs.



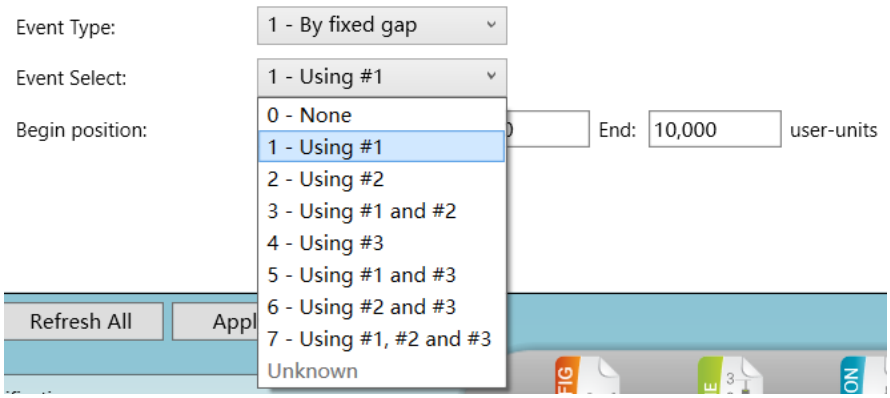
Step 3: Configure PEG Parameters

Navigate to 

→ 

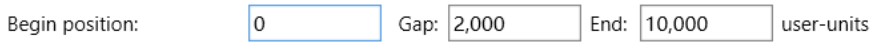
and set the parameters as described in 2.1.

Select the event number configured in Step 2 (e.g., 1#, 2#, or 3# combination).
Let's take *Fixed Gap* Event Type as example.

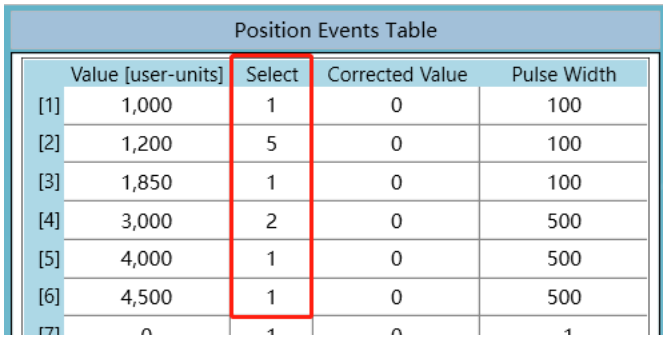


Set the event begin position.

Notes that ensure the unit of *EventGap* matches the unit of (*EventEndPos* − *EventBegPos*).



In table mode, the number under *Select* corresponds to the bound event number.



Position Events Table

	Value [user-units]	Select	Corrected Value	Pulse Width
[1]	1,000	1	0	100
[2]	1,200	5	0	100
[3]	1,850	1	0	100
[4]	3,000	2	0	500
[5]	4,000	1	0	500
[6]	4,500	1	0	500

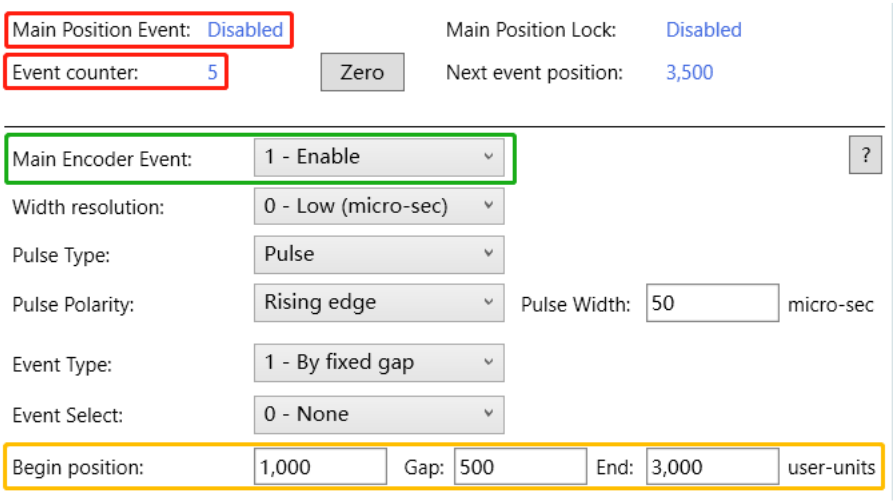
Step 4: Enable PEG

Set *EventOn* = 1 to enable PEG.

Note

When the event counter reaches the total number calculated from *EventBegPos*, *EventGap*, and *EventEndPos*, the controller will automatically disable PEG *(EventOn = 0).*

Example: If the positions are 1000, 1500, 2000, 2500, 3000, PEG will turn off automatically after 5 events.

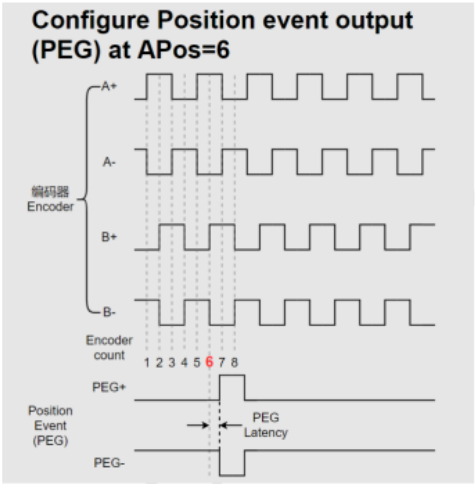


When enabling PEG while the motor is already between *EventBegPos* and *EventEndPos*, *EventCounter* will automatically jump to the corresponding count value.

For example, If PEG is enabled while the motor is at Pos = 1800, the EventCounter value will automatically be set to 2, and the counting direction will be the same as the Gap direction (no counting in the reverse direction).

PEG Response Delay:

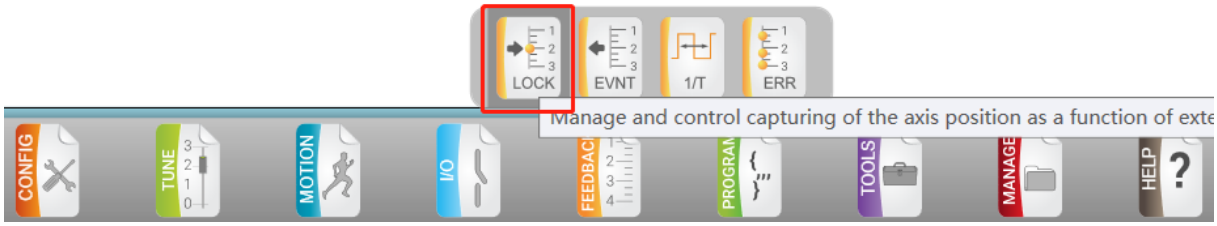
- AqB encoder: < 1 µs
- Sin/Cos encoder: < 10 µs



3 Position Lock Function

Before use, ensure motor can motion normally and I/O connections are correct.

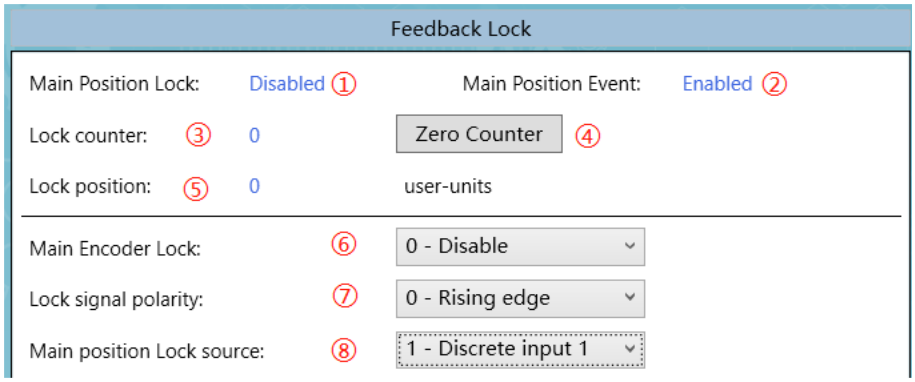
3.1 Parameter Configuration Interface



Navigate to 

→ 

.



① *Main Position Lock:* displays current Lock status (Disabled / Enabled).
② *Main Position Event:* displays Event status (may not appear in some firmware).
③ *Lock Counter:* shows lock capture count.
④ *Zero Counter:* resets *Lock Counter*.
⑤ *Lock Position:* displays the most recently captured position.
⑥ *Main Encoder Lock:* [*LockEn*], 0 = Disable, 1 = Enable.
⑦ *Lock Signal Polarity:* [*LockSrc (Sign)*], 0 = Rising edge, 1 = Falling edge.
⑧ *Main Position Lock Source:* [*LockSrc* (Value)], select input source (digital input, encoder index, or event etc).

3.2 Operation Steps

Step 1: Configure Position Lock parameters according to Section 3.1.
Step 2: Enable Position Lock by setting *LockEn* = 1 (Enable).

4 Keyword Reference

These keywords can be used in PCSuite Terminal, IDE programming, or ASCII string communication.

Keywords	Description
EventOn	PEG state: 0 = Off, 1 = On.
EventCntr	PEG counter — number of events after EventOn. Restarting the EventOn will reset the EventCntr.
EventNextPos	Next event position.
EventPulseRes	Pulse width resolution (0 = µs, 1 = ns).
EventPulseWid	PEG output type: EventPulseWid = 0 -> toggle type. EventPulseWid ≠ 0 -> pulse type (positive = rising edge, negative = falling edge).
EventType	Output event type: 0 = Single Event, 1 = Fixed Gap Event, 2 = Table Event.
EventSelect	PEG output channel, which output signals based on the bound events.
EventTableBeg / EventTableEnd	PEG start/end position.
EventTableSrc	Event output table type: 0 = current position table, 1 = compensated position table.
EventTable[x]	Event output positions (up to 100). EventTable[x] = b, where x = 1, 2, … (x ≤ 100). Here, b represents the event output position, and up to 100 positions are supported.
EventTableSel[x]	Selected output signal channel (up to 100). EventTableSel[x] = c, where x = 1, 2, … (x ≤ 100). Here, c represents the selected output signal channel, based on the bound event to discrete output.
EventTableCor[x]	Compensated position values: EventTableCor[x] = EventTable[x] – MapTable[x] (x ≤ 100).
EventTableWid[x]	The event trigger pulse width at EventTable[x] is defined as EventTableWid[x] = e: • When EventTableWid[x] = -1, the pulse width setting for this row will be ignored, and the value from EventTableWid[x-1] will be used instead. • If this point is the starting trigger position, the pulse width setting will also be ignored, and the default value from EventPulseWid will be used.
LockEn	Position Lock state (0 = Off, 1 = On).
LockCntr	Lock counter.
LockVal	Latest captured Lock position.

Keywords	Description
LockSrc	Position Lock source (positive = rising edge, negative = falling edge).