



Agito PEG & Position Lock 使用手册 v1.0



www.agito-akribis.com

Member of Akribis Systems group

版本记录

版本	描述	日期
1.0	首版发布	2023/10/7

※本公司保留不定期更新的权利，根据产品硬件及软件的升级或更新迭代以及市场需求，本手册将会不定期进行内容上的更新调整，恕不另行告知，如需最新本本文档，请联系 Agito-Akribis 公司获取相应支持。

目录

1	介绍	4
1.1	关于手册	4
1.2	PEG 和 Position Lock 简介	4
2	PEG 功能使用方法	5
2.1	参数配置界面	5
2.2	操作步骤	7
3	Position Lock 功能使用方法	10
3.1	参数配置界面	10
3.2	操作步骤	10
4	关键字介绍	11

1 介绍

1.1 关于手册

感谢您选择 Agito 系列运动控制产品，我们将竭力为您提供追求速度与精度的极致运动控制方案，并提供全方位的技术支持。

本手册主要介绍 Agito 运动控制器 PEG（飞拍）和 Position Lock（探针）功能使用方法。

手册中仅详细介绍与 PEG（飞拍）和 Position Lock（探针）相关的配置内容，其他参数设置可参阅《Agito 快速入门手册》及其他相关手册中的详细介绍，本文档将不再赘述。

1.2 PEG 和 Position Lock 简介

PEG (Position Event), 或称为“飞拍”，指的是电机运动到用户设定的位置时，触发 I/O 信号输出，I/O 信号包括单端数字输出信号、差分数字输出信号，用户可以使用该信号执行相机拍照、激光、继电器、切片等应用。

Position Lock, 或称为“探针”，指的是根据外部事件输入触发捕捉位置，包括编码器 Index 信号（或称 Z 信号）、单端数字输入信号、差分数字输入信号，用户可用于扫描锁定各种物理信号的应用。

2 PEG 功能使用方法

在使用 PEG 或 Position Lock 功能前，请确保电机可以按用户需求正常运动，并且所需的 I/O 信号已正确连接到控制器对应 I/O 端口上正常工作。

2.1 参数配置界面



切换菜单栏到 FEEDBACK ---> Feedback Event 界面如下：

PEG 有多种输出方式供用户选择，如等间距、查表等，输出信号支持单端信号和差分信号，

Main Position Event

Main Position Event: Disabled ①

Event counter: ② 0 ③ Zero Next event position: ④ 0

'Event counter' and 'Next event position' are not functional in this events mode

Main Encoder Event: ⑤ 0 - Disable ?

Width resolution: ⑥ 0 - Low (micro-sec) ?

Pulse Type: ⑦ Pulse

Pulse Polarity: ⑧ Rising edge Pulse Width: ⑨ 50 micro-sec

Event Type: ⑩ 0 - Single event

Event Select: ⑪ 1 - Using #1

Begin position: ⑫ 0

- ①Main Position Event: 显示当前 Event 功能状态，Disabled-开启，Enable-关闭；
 - ②Event counter: [EventCntr]，事件计数，表示当前已输出多少个事件信号；
 - ③Zero: EventCntr=0，将当前事件计数清零；
 - ④Next event position: 显示下一次输出事件的编码器位置；
 - ⑤Main Encoder Event: [EventOn]，EventOn=0 开启 Event 输出，EventOn=1 关闭 Event 输出；
 - ⑥Width resolution: [EventPulseRes]，脉宽分辨率，=0 表示 ms，=1 表示 ns；
 - ⑦Pulse Type: [EventPulseWid]，输出信号类型，Pulse-脉冲型，Toggle-跳变；
 - ⑧Pulse Polarity(仅当 Pulse Type 选择“Pulse”时显示该选项): [EventPulseWid]，脉冲极性，Rising edge-上升沿，Falling edge-下降沿；
 - ⑨Pulse Width: [EventPulseWid]，输出脉冲宽度，
- 注意：输出信号（EventPulseWid）由⑥⑦⑧⑨共同决定；

- ⑩ **Event Type**: 选择输出事件类型，输出类型有 3 种形式，

- ◆ 单事件输出 (EventType=0) :

Event Type: ← 单个事件
 Event Select: ← 选择事件输出数字口
 Begin position: ← 事件输出起始位置

- ◆ 等间隔输出 (EventType=1) :

Event Type: ← 等间隔输出
 Event Select: ← 选择事件输出数字口
 Begin position: ← 起始位置 Gap: ← 位置间隔 End: ← 停止位置 user-units

- ◆ 查表输出 (EventType=2) :

Event Type: ← 查表输出
 Event table start index: ← 起始行 End index: ← 停止行
 Select table to use: ← 选择事件输出表

当选择查表方式输出时，将会在右侧窗口显示如下表格：

	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
[7]	0	1	0	-1
[8]	0	1	0	-1
[9]	0	1	0	-1
[10]	0	1	0	-1
[11]	0	1	0	-1
[12]	0	1	0	-1
[13]	0	1	0	-1
[14]	0	1	0	-1
[15]	0	1	0	-1
[16]	0	1	0	-1

Show from index:

Refresh Table

Previous Next

Correct Event Table

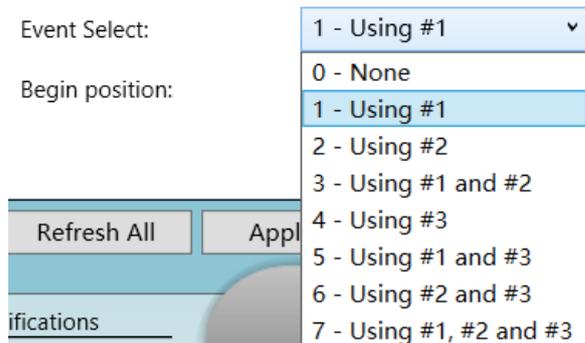
Load from File Into Controller

Save from Controller to File

- 脉冲宽度[EventTableWid[x]];
- 事件输出类型[EventTableSel[x]]: 有 7 种方式可以选择;
- 事件输出位置点[EventTable[x]];
- 事件输出位置点索引号: 最多 100 点, 每页最多显示 16 个点位 (x) ;
- Show from index: 选择显示表格的起始索引号 (x), 配合 Refresh Table 使用;
- Refresh Table: 刷新表格内容;
- Previous/Next: 表格翻页;

- Correct Event Table[EventCorrect](仅当 EventTableSrc=1 时使用): 将误差补偿补偿值叠加到位置点中;
- Load from File Into Controller: 导入本地保存的 Event 表 (*.aef 格式);
- Save from Controller to File: 将 Event 表导出到本地文件 (*.aef 格式);

- ⑪ **Event Select:** 选择 PEG 输出信号口



2.2 操作步骤

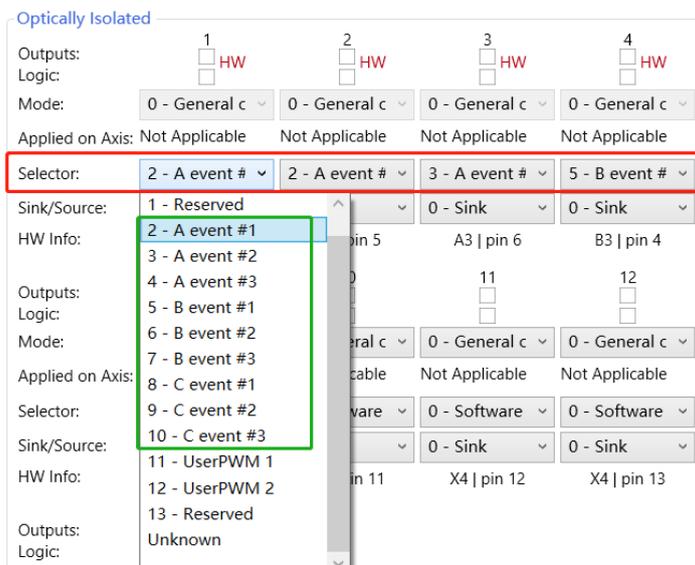
• 第 1 步：I/O 接线

PEG 使用数字输出信号输出事件，在使用前根据用户需求连接好数字输出信号（单端或差分信号），数字输出电气接线部分可参阅《Agito 快速入门手册》中 2.4.2 章节及所使用对应控制器产品手册中数字输出相关内容，本文档将不再赘述。

• 第 2 步：绑定事件

切换菜单栏到 **Discrete ---> Discrete Outputs** 界面，将对应的数字输出口绑定轴事件号（如图示中将 DO_1 绑定 A 轴 1#事件，DO_2 也绑定 A 轴 1#事件，DO_3 绑定 A 轴 2#事件，DO_4 绑定 B 轴 1#事件）：

最多可以使用 3 个事件号，同一事件号可以在不同数字输出口重复使用，使用时注意应用轴；



• 第 3 步：配置 PEG 参数

切换菜单栏到 **FEEDBACK ---> Feedback Event** 界面，根据本文 2.1 章节的内容设置 Event 参数，

其中 Event Select 处选择第 2 步中绑定的事件号，共有 7 种方式（即 1#、2#、3#事件自由组合）供选择（以下以等间距输出为例）：

Event Type: 1 - By fixed gap

Event Select: 1 - Using #1

Begin position: 0 End: 10,000 user-units

Refresh All App

1 - Using #1
2 - Using #2
3 - Using #1 and #2
4 - Using #3
5 - Using #1 and #3
6 - Using #2 and #3
7 - Using #1, #2 and #3
Unknown

设定事件触发的位置点，值得注意的是 **EventGap** 的符号，其符号需要和（**EventEndPos-EventBegPos**）符号相同。

Begin position: 0 Gap: 2,000 End: 10,000 user-units

当选择查表输出时：表中 **Select** 中的数字（**EventTabSel[x]**）即表示绑定的事件号，同上，共有 7 种方式选择（即 **1#、2#、3#**事件自由组合）

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

第 4 步：开启 PEG 功能

将 **EventOn** 置 1 时将开启 PEG 功能：

Main Encoder Event: 1 - Enable

值得注意的是：当 **Event counter** 计数达到控制器根据 **Gap** 和起始、停止位置计算的总数时，**EventOn** 将自动置 **0-Disable**；

例如下图所示中，根据所设定的 **EventBegPos**、**EventGap**、**EventEndPos** 所计算出来总的计数是 5（即分别在位置 1000、1500、2000、2500、3000 共 5 个位置），当 **EventCounter** 计数达到 5 次时，PEG 功能将自动关闭（**Disabled**）；

如电机当前位置（**Pos**）处于 **EventBegPos** 和 **EventEndPos** 之间时开启 PEG(**EventOn: 0→1**)时，**EventCounter** 值将自动跳到该位置应该累积的次数值，如开启 PEG 时电机位于 **Pos=1800** 处时，此时 **EventCounter** 值将自动置为 2，计数方向和 **Gap** 方向一致（反向不计数）；

Main Position Event: Disabled Main Position Lock: Disabled

Event counter: 5 Zero Next event position: 3,500

Main Encoder Event: 1 - Enable

Width resolution: 0 - Low (micro-sec)

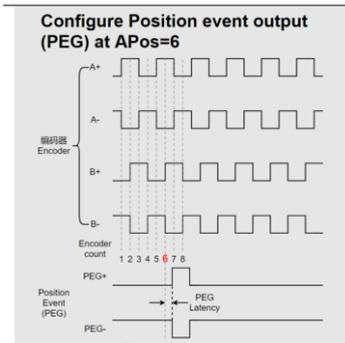
Pulse Type: Pulse

Pulse Polarity: Rising edge Pulse Width: 50 micro-sec

Event Type: 1 - By fixed gap

Event Select: 0 - None

Begin position: 1,000 Gap: 500 End: 3,000 user-units

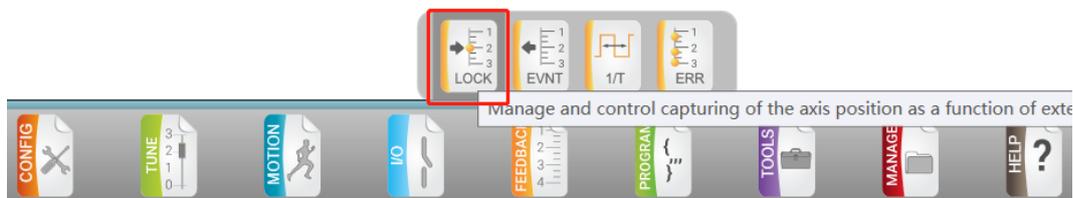


- PEG 性能 (延迟) :
AqB 编码器 <math><1\mu\text{s}</math>
Sin/Cos 编码器 <math><10\mu\text{s}</math>

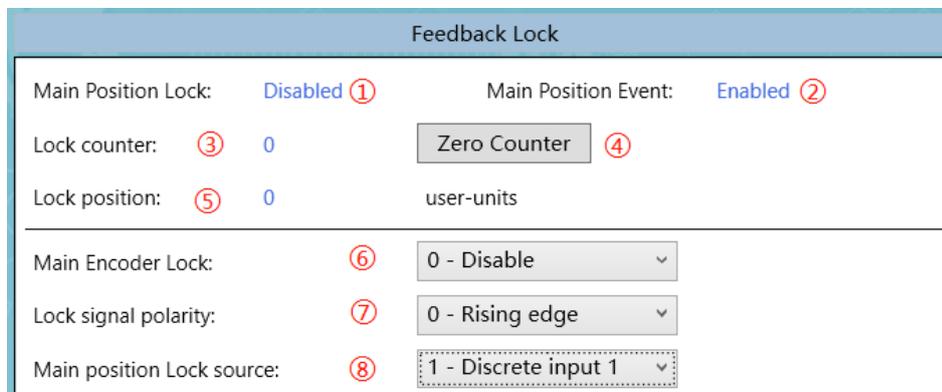
3 Position Lock 功能使用方法

在使用 PEG 或 Position Lock 功能前，请确保电机可以按用户需求正常运动，并且所需的 I/O 信号已正确连接到控制器对应 I/O 端口上正常工作。

3.1 参数配置界面



切换菜单栏到 FEEDBACK ---> Feedback Lock 界面如下：



- ①Main Position Lock: 显示当前 Lock 功能状态；
- ②Main Position Event: 显示当前 Event 功能状态（部分版本固件不显示）；
- ③Lock counter: Lock 计数；
- ④Zero Counter: 将 Lock 计数归零；
- ⑤Lock position: 显示最近一次锁存的位置信息；
- ⑥Main Encoder Lock: [LockEn], 0-Disable 关闭 Lock 功能，1-Enable 开启 Lock 功能；
- ⑦Lock signal polarity: [LockSrc (Sign)], 0-Rising edge 上升沿触发，1-Falling edge 下降沿触发；
- ⑧Main position Lock source: [LockSrc (Value)], 选择触发信号源，可以选择数字信号（单端或差分）、编码器 index 信号、Event 事件等；

3.2 操作步骤

• 第 1 步：设置信号参数

根据 3.1 章节介绍设置 Position Lock 相应参数；

• 第 2 步：开启 Lock 功能

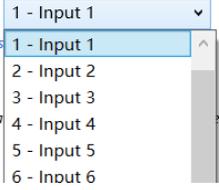
将 LockEn 置 1-Enable 将开启 Position Lock 功能；



4 关键字介绍

Agito 关键字不仅可以在 PCSuite Terminal 终端及 IDE 编程环境使用，也可用于用户通过字符串或 ASCII 通讯使用。

关键字	描述
EventOn	PEG 状态： <ul style="list-style-type: none"> EventOn=0 关闭 Event 输出， EventOn=1 开启 Event 输出；
EventCnt	PEG 计数：表示最近一次 EventOn 之后输出的事件数，重启 EventOn 将重新计数；
EventNextPos	下一次输出事件的编码器位置；
EventPulseRes	脉宽分辨率： <ul style="list-style-type: none"> EventPulseRes =0 表示 ms， EventPulseRes =1 表示 ns；
EventPulseWid	PEG 信号类型：EventPulseWid=0 时表示选择信号跳变时输出，EventPulseWid=a (a≠0) 时选择脉冲输出，a>0 时脉冲上升沿输出，a<0 时脉冲下降沿输出；
EventType	输出事件类型： <ul style="list-style-type: none"> EventType=0，单事件输出， EventType=1，等间隔输出， EventType=2 查表输出；
EventSelect	PEG 输出信号口，即根据绑定的事件输出信号；
EventTableBeg	PEG 开始位置；
EventTableEnd	PEG 停止位置；
EventTableSrc	PEG 查表输出时的表类型： <ul style="list-style-type: none"> EventTableSrc=0 时，表示使用当前位置表输出， EventTableSrc=1 时，表示使用误差补偿之后的位置表输出；
EventTable[x]	EventTable[x]=b,x=1,2,...(x≤100)，b 表示事件输出位置点，最大支持 100 个位置；
EventTableSel[x]	EventTableSel[x]=c,x=1,2,...(x≤100)，c 表示选择的输出信号口，根据数字输出通道绑定的事件输出；
EventTableCor[x]	EventTableCor[x]=d,x=1,2,...((x≤100)，d 表示补偿后的位置点， d=EventTable[x]-MapTable[x]；
EventTableWid[x]	在 EventTable[x] 处的事件触发脉冲宽度，EventTableWid[x]=e： <ul style="list-style-type: none"> 当 EventTableWid[x]=-1 时将忽略该行脉宽设置，而使用 EventTableWid[x-1] 的值，

	<ul style="list-style-type: none"> 如该点为起始触发位置点，将会忽略该该脉宽设定值，而默认使用 EventPulseWid 中的值；
LockEn	<p>PositionLock 状态：</p> <ul style="list-style-type: none"> LockEn=0 关闭 PositionLock， LockEn=1 开启 PositionLock；
LockCntr	Lock 计数；
LockVal	最近一次锁存的位置信息；
LockSrc	<p>PositionLock 信号源：正数表示上升沿，负数表示下降沿；</p> <p>Main position Lock source: </p> <p><i>Locking as a function of multiple inputs is possible</i></p> <p><i>LockValTable: The Lock feature also provides a</i></p>

