



# Agito 误差补偿使用手册 v1.0



[www.agito-akribis.com](http://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	2023/9/19

※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	误差补偿 (Error Mapping) 简介	4
1.3	关于误差补偿功能注意事项	4
2	PCSuite Error Mapping 界面介绍	5
3	操作步骤	7
3.1	1D Error Mapping	7
3.2	2D ErrMapping	9
3.3	3D ErrMapping	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 运动控制器误差补偿（Error Mapping）功能使用方法。

手册中仅详细介绍与 Error Mapping 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 误差补偿（Error Mapping）简介

---

通常我们假定编码器反馈的位置和电机（或负载，取决于机械结构）真实物理位置是完全吻合的，然而在某些特定应用或机械制造工艺等原因导致编码器反馈读数无法精确反映负载的真实物理位置，例如以下情形：

- 电机与负载之间机械传动不固定；
- 直线运动系统机械结构非严格线性，存在弯曲；
- XY 平台不垂直，非绝对 90° 角；

此时需要对系统位置进行误差补偿，最大程度使负载编码器读数和真实物理位置吻合，达到精确位置控制的目标。

Agito 控制器支持 1D、2D、3D 误差补偿功能。

## 1.3 关于误差补偿功能注意事项

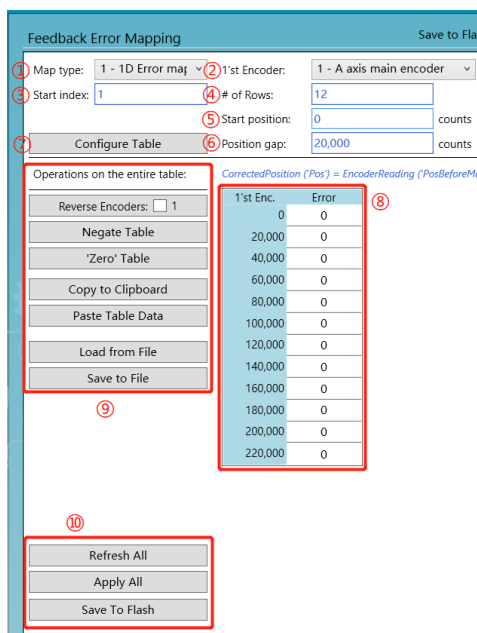
---

- 误差补偿功能需要使用到激光干涉仪进行误差测量标定；
- 由于位置读数校正是在每个采样周期内进行的，并且是在编码器反馈读数上进行校正的，因此不仅会校正最终的运动位置，同时也会修正运动速度。
- 在进行回零（Homing）前，需要关闭补偿功能，此功能在回零步骤中可配置，详细请参阅《Agito 回零使用手册》中的内容介绍。
- 在误差补偿表中补偿值确定后，**一般将误差补偿功能的开启置于回零步骤最后一步，详细步骤请参阅《Agito 回零使用手册》中的 2.3 章节内容介绍**，本文将不再赘述。
- 进行误差补偿前，请确保回零步骤已设定完成，可以正常回零，并且此时电机零位（Pos=0）位置即用户所需的零位，**此步骤为关键步骤，因为补偿位置是以该零点作为位置参考点**；

## 2 PCSuite Error Mapping 界面介绍



- ◆ **1 维误差补偿**: 打开 Error Mapping, 默认 Map type 为 “0-No Error Map”, 选择 Map type 为 “1-1D Error map”, 将开启 1 维误差补偿功能, 界面如下:



- ① Map type: [MapType], 误差补偿表类型, 0-关闭误差补偿功能, 1-开启一维误差补偿, 2-开启二维误差补偿, 3-开启三维误差补偿;
  - ② 1'st Encoder: [MapEncoder[1]], 选择用于误差补偿的编码器信号源;
  - ③ Start index: [MapStartIndex], 选择误差补偿表生效起始行;
  - ④ # of Rows: [MapLength[1]], 定义位置误差补偿表总行数, 最大 1600 点;
  - ⑤ Start Position: [MapStartPos[1]], 定义误差补偿表起始位置 (编码器读数);
  - ⑥ Position gap: [MapPosGap[1]], 位置采样间隔;
  - ⑦ Configure Table: 根据以上用户设定值建立误差补偿表;
- 注: [\*]内为对应关键字 (Keyword) .

- ⑧ 误差补偿表: 点击 “⑦ Configure Table” 后, 会自动生成 ⑧ 所示误差补偿表 (注意: 该表每页最多显示 15 行, 如总行数超过 15 行将会显示以下设置窗, 更改 1'st Enc 的值更改表格的起始位置, 然后点击 “Update Visible Table” 将会更新误差补偿表到用户设定的位置), 补偿表第 1 页第 1 行为 MapTable[1], 依此类推。

Show table range from:

1'st Enc:  counts

- ⑨ 误差补偿表操作栏:

**Reverse Encoders:**  1: 该功能开发中, 暂不支持使用;

**Negate Table:** 将误差补偿表中的所有补偿值取反;

**'Zero' Table:** 将误差补偿表中的所有补偿值置 0;

**Copy to Clipboard:** 将误差补偿表中的所有补偿值复制到剪贴板;

**Paste Table Data:** 粘贴剪贴板上的补偿值 (注意: 当粘贴的数据和补偿表行数不一致时, 会弹出提示窗, 选择 “Yes” 将按照最短的匹配长度进行数据替换, 超出的行数保持保持不变);

**Load from File:** 导入本地保存的误差补偿表;

**Save to File:** 将误差补偿表导出到本地;

• ⑩ErrMapping 功能操作栏:

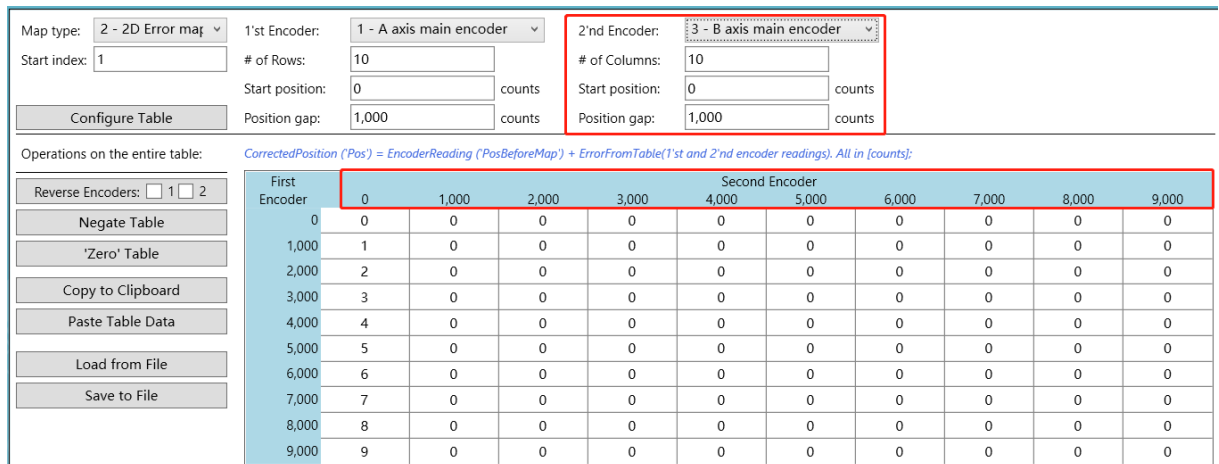
**Reflash All:** 刷新当前蓝色状态参数（蓝色字体）为保存过的参数;

**Apply All:** 应用当前参数（和点“回车”键作用相同），掉电不保存;

**Save to flash:** 将当前参数保存到控制器 Flash 中，掉电保存;

◆ **2 维误差补偿:** 选择 Map type 为“2-2D Error map”，将开启 2 维误差补偿功能，界面如下:

相比 1 维误差补偿，2 维误差即增加另一维度编码器位置参考，在“2'nd Encoder”选择需要做 2 维运动的另一关联轴编码器，方法可参考 1 维位置误差补偿，与第 1 个编码器位置公共用于查询误差补偿表，查询到的误差补偿值加上轴编码器位置，得到校正后的位置;

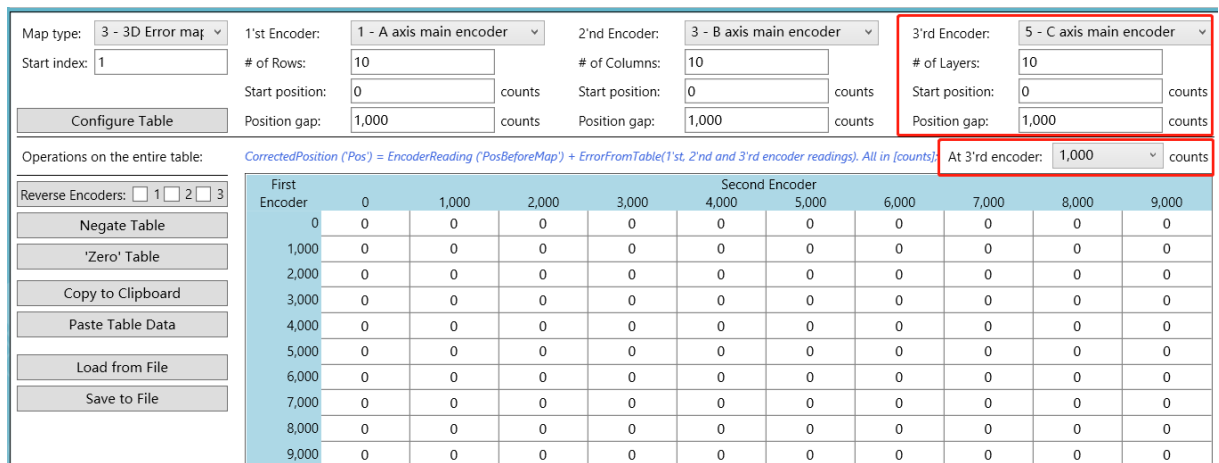


Operations on the entire table:  $CorrectedPosition (Pos) = EncoderReading (PosBeforeMap) + ErrorFromTable(1'st \text{ and } 2'nd \text{ encoder readings}). All \text{ in } [counts];$

First Encoder	Second Encoder									
	0	1,000	2,000	3,000	4,000	5,000	6,000	7,000	8,000	9,000
0	0	0	0	0	0	0	0	0	0	0
1,000	1	0	0	0	0	0	0	0	0	0
2,000	2	0	0	0	0	0	0	0	0	0
3,000	3	0	0	0	0	0	0	0	0	0
4,000	4	0	0	0	0	0	0	0	0	0
5,000	5	0	0	0	0	0	0	0	0	0
6,000	6	0	0	0	0	0	0	0	0	0
7,000	7	0	0	0	0	0	0	0	0	0
8,000	8	0	0	0	0	0	0	0	0	0
9,000	9	0	0	0	0	0	0	0	0	0

◆ **3 维误差补偿:** 选择 Map type 为“3-3D Error map”，将开启 3 维误差补偿功能，界面如下:

相比 2 维误差补偿，3 维误差即再增加另一维度编码器位置参考，在“3'nd Encoder”选择需要做 3 维运动的另一关联轴位置范围，方法可参考 1 维位置误差补偿，三维误差补偿表中显示的是第 3 维每个 Gap 处对应的 2 维误差补偿值，3 维位置点通过“3'ed encoder”选择对应位置切换，由三个编码器共同用于查询位置误差补偿表;



Operations on the entire table:  $CorrectedPosition (Pos) = EncoderReading (PosBeforeMap) + ErrorFromTable(1'st, 2'nd \text{ and } 3'rd \text{ encoder readings}). All \text{ in } [counts];$  At 3'rd encoder: 1,000 counts

First Encoder	Second Encoder									
	0	1,000	2,000	3,000	4,000	5,000	6,000	7,000	8,000	9,000
0	0	0	0	0	0	0	0	0	0	0
1,000	0	0	0	0	0	0	0	0	0	0
2,000	0	0	0	0	0	0	0	0	0	0
3,000	0	0	0	0	0	0	0	0	0	0
4,000	0	0	0	0	0	0	0	0	0	0
5,000	0	0	0	0	0	0	0	0	0	0
6,000	0	0	0	0	0	0	0	0	0	0
7,000	0	0	0	0	0	0	0	0	0	0
8,000	0	0	0	0	0	0	0	0	0	0
9,000	0	0	0	0	0	0	0	0	0	0

查询表内的点与点之间进行线性插值，并且假设这些表为等间距排列。如果编码器读数超出表格范围（低于起始位置或高于结束位置），则表中第一个点或最后一个点的数值将被用作误差校正。

### 3 操作步骤

在进行误差补偿前，请先完成以下事项：

- 电机可以正常运动且满足用户运动整定误差要求；
- 回零步骤已设定完成，且可以正常回零，确保此时电机零位（Pos=0）位置即用户所需的零位，**此步骤为关键步骤，因为补偿位置是以该零点作为位置参考点**；
- 在打激光运动标定之前请先回零，并且确保误差补偿功能已关闭（MapType=0）。

#### 3.1 1D Error Mapping

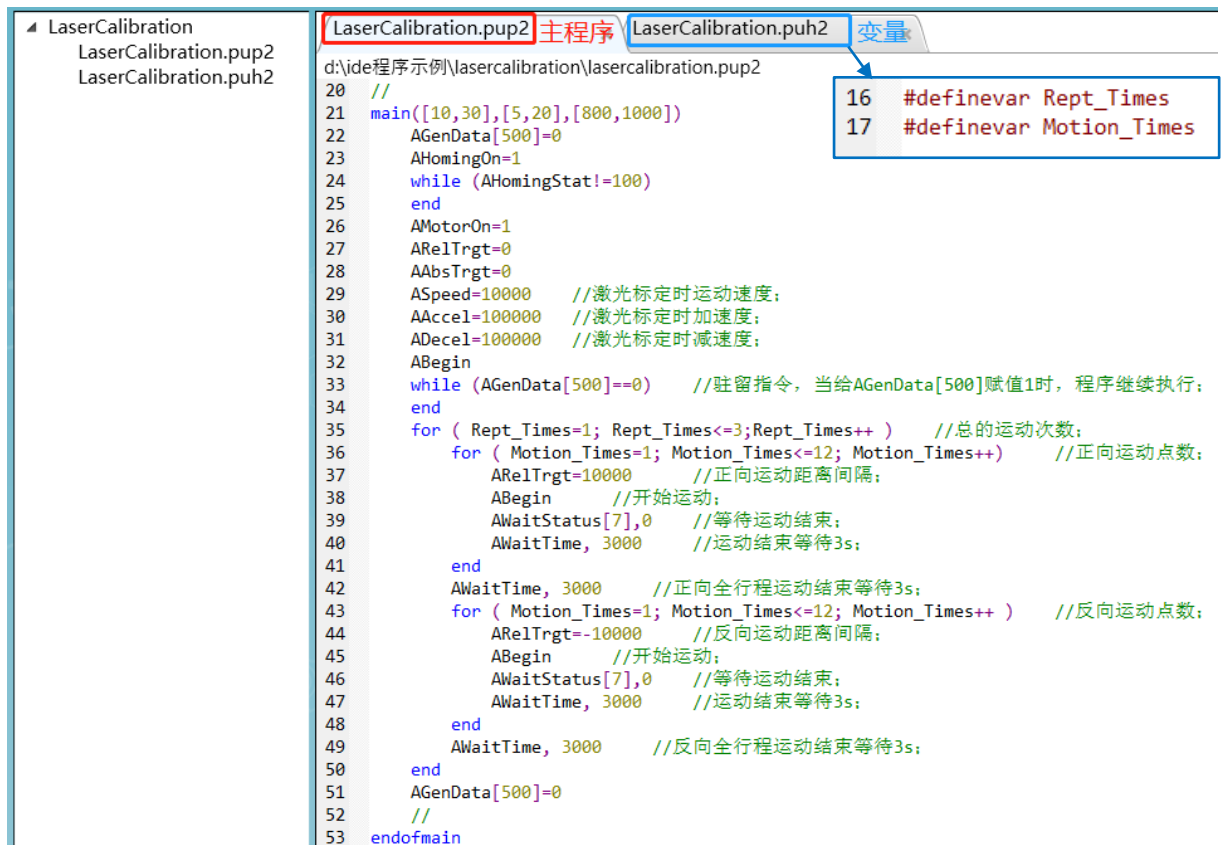
- **第 1 步：回零**

回零详细步骤请参阅《Agito 回零使用手册》；

- **第 2 步：编写电机运行 IDE 程序**

以下示例程序以 A 轴正向连续运动 12 次，反向连续运动 12 次，全行程重复 3 次为演示，用户根据实际需要调整相关参数；

设定好参数后依次点击“Compile and Debug”-->“Run/Restart”运行程序，程序将会驻留在第 33、34 行，直到用户发送运动指令（见步骤 3）；



```

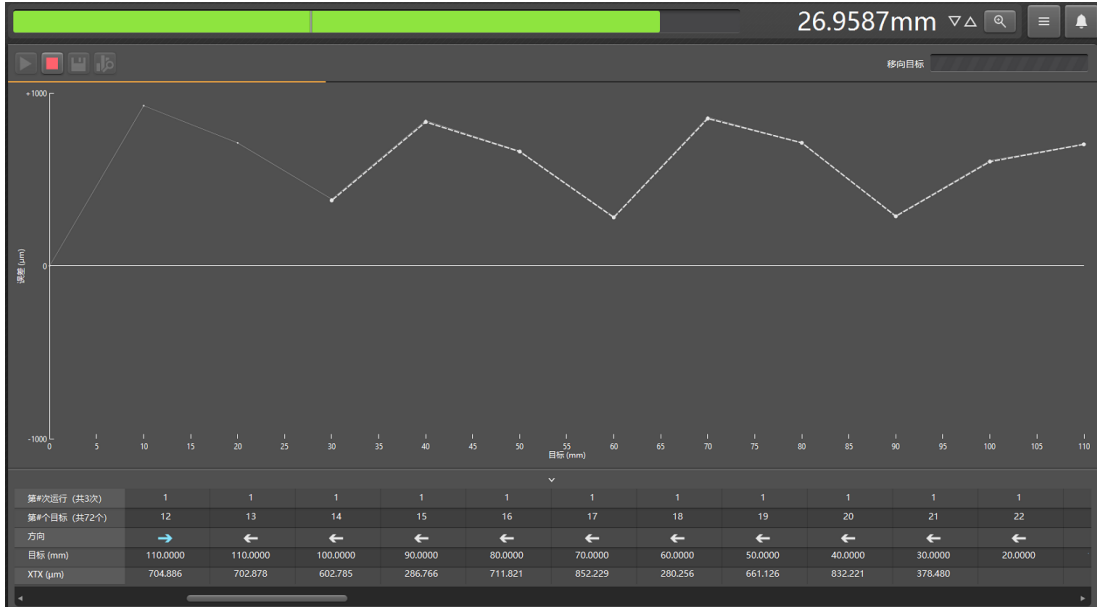
d:\ide程序示例\lasercalibration\lasercalibration.pup2
20 //
21 main([10,30],[5,20],[800,1000])
22 AGenData[500]=0
23 AHomingOn=1
24 while (AHomingStat!=100)
25 end
26 AMotorOn=1
27 ARelTrgt=0
28 AAbsTrgt=0
29 ASpeed=10000 //激光标定时运动速度;
30 AAccel=100000 //激光标定时加速度;
31 ADecel=100000 //激光标定时减速度;
32 ABegin
33 while (AGenData[500]==0) //驻留指令，当给AGenData[500]赋值1时，程序继续执行;
34 end
35 for ( Rept_Times=1; Rept_Times<=3;Rept_Times++ ) //总的运动次数;
36   for ( Motion_Times=1; Motion_Times<=12; Motion_Times++ ) //正向运动点数;
37     ARelTrgt=10000 //正向运动距离间隔;
38     ABegin //开始运动;
39     AWaitStatus[7],0 //等待运动结束;
40     AWaitTime, 3000 //运动结束等待3s;
41   end
42   AWaitTime, 3000 //正向全行程运动结束等待3s;
43   for ( Motion_Times=1; Motion_Times<=12; Motion_Times++ ) //反向运动点数;
44     ARelTrgt=-10000 //反向运动距离间隔;
45     ABegin //开始运动;
46     AWaitStatus[7],0 //等待运动结束;
47     AWaitTime, 3000 //运动结束等待3s;
48   end
49   AWaitTime, 3000 //反向全行程运动结束等待3s;
50 end
51 AGenData[500]=0
52 //
53 endofmain
  
```

- **第 3 步：控制电机运动及使用激光干涉仪采集位置**

架设好激光干涉仪并设置相关参数开始采集，并在 PCSuite Terminal 终端里发送指令“AGenData[500]=1”，电机将开始按设定参数运动

```

-----
AGenData[500]=1
OK>
-----
  
```



← 误差补偿 - RDM030-A-B2 : XTX

← 配置

补偿类型: 双向

计算类型: 绝对式

补偿单位: mm

小数位: 4

Backlash resolution: 0

补偿分辨率: 1

符号规约: 计算补偿值

类型: LEC.REN

基准位置: 0 mm

补偿开始: 0 mm

补偿结束: 110 mm

补偿间距: 10.0000 mm

补偿点数量: 12

补偿表 图形补偿

绝对式 误差补偿表 (mm)

系数	位置 (mm)	正向 (比例: 1)	反向 (比例: 1)
1	0.0000	0.0000	0.0005
2	10.0000	-0.4677	-0.4600
3	20.0000	-0.4479	-0.4448
4	30.0000	0.1355	0.1402
5	40.0000	-0.3045	-0.2966
6	50.0000	-0.5181	-0.5135
7	60.0000	0.1011	0.1067
8	70.0000	-0.2351	-0.2276
9	80.0000	-0.4678	-0.4655
10	90.0000	0.2045	0.2078
11	100.0000	-0.4422	-0.4346
12	110.0000	-0.7609	-0.7626

• **第 4 步: 编辑位置误差补偿表**

根据以上步骤采集到的误差补偿值换算为 Counts（注意：控制器只能以 Counts 作为位置单位），将对应的补偿值填入 PCSuite 对应轴的误差补偿表中：

Feedback Error Mapping

Map type: 1 - 1D Error map

Start index: 1

1'st Encoder: 1 - A axis main encoder

# of Rows: 12

Start position: 0 counts

Position gap: 20,000 counts

Configure Table

Operations on the entire table:

Reverse Encoders:  1

Negate Table

'Zero' Table

Copy to Clipboard

Paste Table Data

Load from File

Save to File

CorrectedPosition (Pos) = EncoderReading (PosBeforeMap)

1'st Enc.	Error
0	0
20,000	944
40,000	975
60,000	-300
80,000	589
100,000	1,154
120,000	-229
140,000	457
160,000	1,037
180,000	-469
200,000	846
220,000	1,750

- 注意误差补偿值的符号和偏差符号相反；
- 补偿值需要换算为 Counts；
- 超过 15 行时，通过更改起始行更改 1' st Enc 的值更改表格的起始位置，然后点击“Update Visible Table”，请参阅第 5 页；



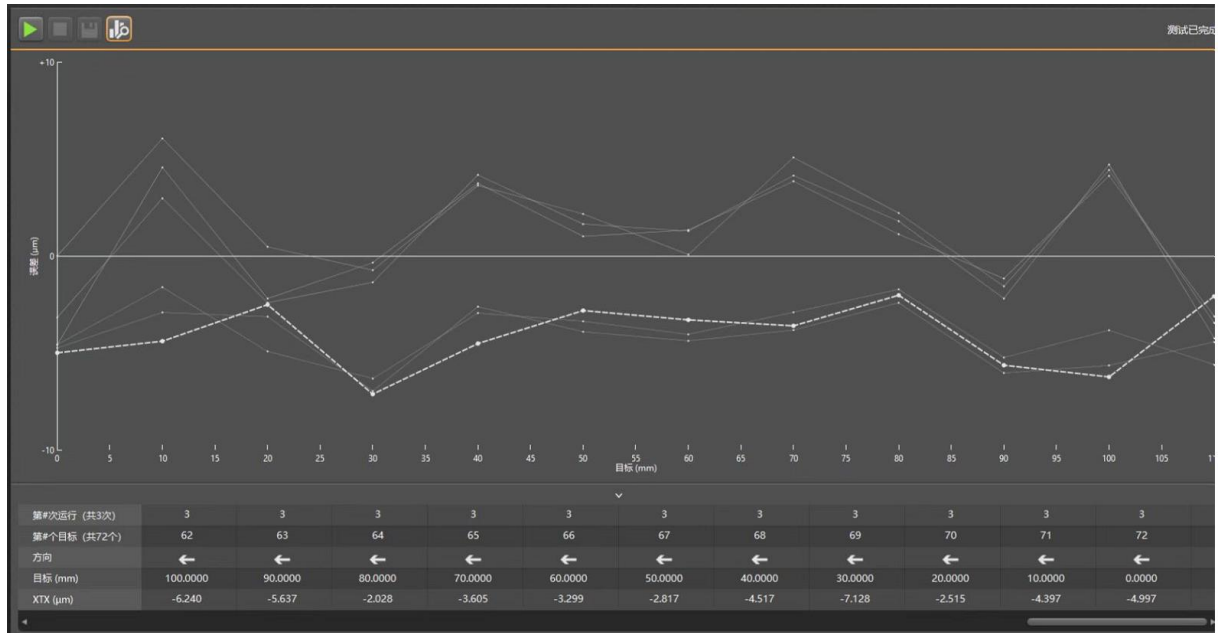
• **第 5 步：测量校正后的物理偏差**

设置好误差补偿表之后，点击“Apply All”应用参数，再重复 1、2、3 步骤测量使用误差补偿表校正后的物理偏差。

值得注意的是此次测试前，需要开启误差补偿，“XMapType=y, X 表示轴号（2 维、3 维选择 1'st Encoder 轴号），y 表示误差补偿类型（1、2、3）”；

对比激光干涉仪测量到的开启误差补偿前后的图像可看出误差值由补偿前的最大 1000um 左右减小到最大 6um 左右；

开启误差补偿后 ↓



• **第 6 步：多次测量得到最佳补偿效果**

根据第 5 步得到的实际偏差图像，重复以上 3、4、5 步骤，对误差补偿表根据多减少加的原则进行微调，多次测量以获取最佳运动性能；

### 3.2 2D ErrMapping

2 维误差补偿和 1 维误差补偿方法类似，只是误差补偿表由线性变为 2 维表格，同时以 X、Y 为参考位置，相当于测量多组 1 维补偿值；

Map type:  1'st Encoder:  2'nd Encoder:

Start index:  # of Rows:  # of Columns:

Start position:  counts Start position:  counts

Position gap:  counts Position gap:  counts

Operations on the entire table: CorrectedPosition ('Pos') = EncoderReading ('PosBeforeMap') + ErrorFromTable(1'st and 2'nd encoder readings). All in [counts];

First Encoder	Second Encoder									
	0	2,000	4,000	6,000	8,000	10,000	12,000	14,000	16,000	18,000
0	0	0	0	0	0	0	0	0	0	0
1,000	0	0	0	0	0	0	0	0	0	0
2,000	0	0	0	0	0	0	0	0	0	0
3,000	0	0	0	0	0	0	0	0	0	0
4,000	0	0	0	0	0	0	0	0	0	0
5,000	0	0	0	0	0	0	0	0	0	0
6,000	0	0	0	0	0	0	0	0	0	0
7,000	0	0	0	0	0	0	0	0	0	0
8,000	0	0	0	0	0	0	0	0	0	0
9,000	0	0	0	0	0	0	0	0	0	0

### 3.3 3D ErrMapping

3 维误差补偿在 X、Y 基础上，增加 Z 方向为参考位置，相当于测量多组 2 维补偿值，在每个 Z 向间隔点（Gap）测量 2 维误差进行补偿；

Map type:   
 Start index:

1'st Encoder:   
 # of Rows:   
 Start position:  counts  
 Position gap:  counts

2'nd Encoder:   
 # of Columns:   
 Start position:  counts  
 Position gap:  counts

3'rd Encoder:   
 # of Layers:   
 Start position:  counts  
 Position gap:  counts

Operations on the entire table: CorrectedPosition (Pos) = EncoderReading (PosBeforeMap) + ErrorFromTable(1'st, 2'nd and 3'rd encoder readings). All in [counts];
 At 3'rd encoder:  counts

Reverse Encoders:  1  2  3

First Encoder	Second Encoder									
	0	2,000	4,000	6,000	8,000	10,000	12,000	14,000	16,000	18,000
0	0	0	0	0	0	0	0	0	0	0
1,000	0	0	0	0	0	0	0	0	0	0
2,000	0	0	0	0	0	0	0	0	0	0
3,000	0	0	0	0	0	0	0	0	0	0
4,000	0	0	0	0	0	0	0	0	0	0
5,000	0	0	0	0	0	0	0	0	0	0
6,000	0	0	0	0	0	0	0	0	0	0
7,000	0	0	0	0	0	0	0	0	0	0
8,000	0	0	0	0	0	0	0	0	0	0
9,000	0	0	0	0	0	0	0	0	0	0

## 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
MapType	<p>MapType 用于用于定义误差补偿类型，以下为值含义：</p> <p>0-关闭误差补偿功能</p> <p>1-开启 1 维误差补偿</p> <p>2-开启 2 维误差补偿</p> <p>3-开启 3 维误差补偿</p>
MapEncoder	<p>MapEncoder[x],x=1,2,3, 分别表示 X、Y、Z 三个维度的编码器信号源；</p> <p>以下为值含义（图示未显示完整）：</p> <div style="border: 1px solid gray; padding: 5px; margin: 5px 0;"> <p>1 - A axis main encoder ▾</p> <p>Unknown ▲</p> <p>1 - A axis main encoder</p> <p>2 - A axis auxiliary encoder</p> <p>3 - B axis main encoder</p> <p>4 - B axis auxiliary encoder</p> <p>5 - C axis main encoder</p> <p>6 - C axis auxiliary encoder</p> <p>7 - D axis main encoder</p> <p>8 - D axis auxiliary encoder</p> <p>9 - E axis main encoder</p> <p>10 - E axis auxiliary encoder</p> <p>11 - F axis main encoder</p> <p>12 - F axis auxiliary encoder</p> <p>13 - G axis main encoder</p> <p>14 - G axis auxiliary encoder ▾</p> </div>
MapLength	<p>MapLength[x],x=1,2,3, 分别表示 X、Y、Z 三个维度补偿表总行数，三个维度总和最大 2000 行；</p>
MapStartPos	<p>MapLength[x],x=1,2,3, 分别表示 X、Y、Z 三个维度补偿表起始位置；</p>
MapPosGap	<p>MapLength[x],x=1,2,3, 分别表示 X、Y、Z 三个维度补偿表位置间隔；</p>
MapTable	<p>MapTable[y],y=1,2,3···, 分别表示误差补偿表行数。</p>
MapStartIndex	<p>MapTable 中的起始行，通常设为 1，除非是需要保存多个误差补偿表；</p>

