Error Mapping

Application Note

www.agito-akribis.com
Member of Akribis Systems group

Revision History

Version	Description	Date
1.0	Initial release	13 March 2025

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

Products Rights

AGDx, AGCx, AGMx, AGAx, AGIOx, 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 document 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 document without notice at any time.

Trademarks

Agito PCSuite is a trademark of Agito Akribis Systems Ltd..

Contents

1 Introduction

1.1 About this Application Note

This manual mainly introduces the use of the error compensation (Error Mapping) function of the Agito Motion Controller. The narrative in the manual only comprise the configuration content related to Error Mapping in detail.

1.2 Introduction

In an ideal situation, we assume that the encoder feedback position and the actual physical position of the motor (or load, depending on the mechanical structure) are completely consistent. However, in some specific applications or mechanical manufacturing processes, the encoder feedback reading cannot accurately reflect the actual physical position of the load, such as the following situations:

- The mechanical transmission between the motor and the load is not fixed.
- The mechanical structure of the linear motion system is not entirely linear and has bending.
- The XY platform is not orthogonal.

Due to these circumstances, there is a need to compensate the error on the system to maximize the consistency between the load encoder reading and the actual physical position to achieve the goal of precise position control.

Agito controller supports 1D, 2D and 3D error compensations functions.

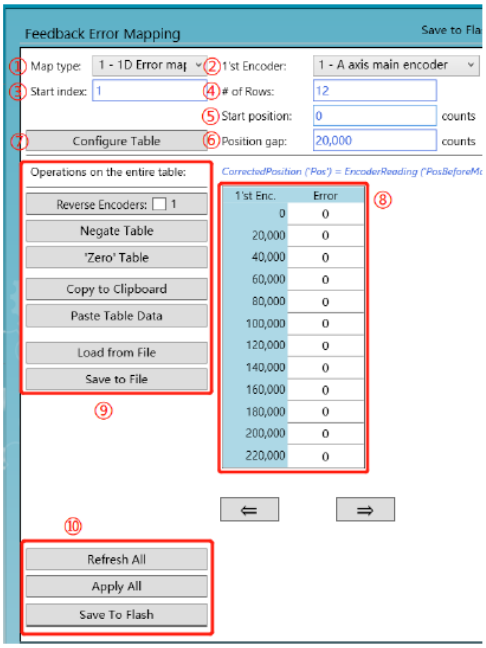
1.3 Notes on the Error Compensation Function

The error compensation function requires the use of a laser interferometer for error measurement calibration.

- The position reading correction is performed in each sampling cycle and is corrected on the encoder feedback reading. Not only the final motion position will be corrected, but the motion speed will be corrected as well.
- Before homing, the error mapping function needs to be turned off. This can be configured in the homing steps option “Set Error Mapping Type” to disable it at the start and setting it to turn on at the last step of the homing process.
- Before performing error compensation, make sure that the homing step (including the point mentioned above) has been set and can perform homing process normally and the motor zero position (Pos=0) at this time is the zero-position required by the user. This step is a key step as the compensation position uses this zero point as the position reference point.

2 PCSuite Error Mapping Interface Introduction

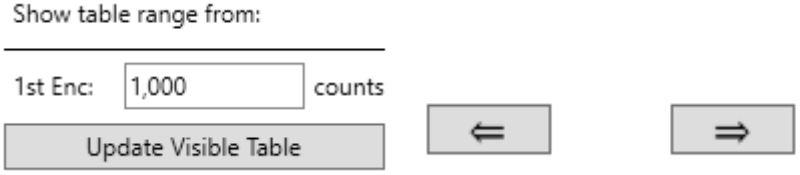
❖ 1D Error Compensation: Open Error Mapping, the default Map type is “0-No Error Map”, select Map type as “1-1D Error Map”, and the 1D error compensation function will be enabled. The interface is as follows:



- ①Map type: [MapType], Error Compensation table type, 0-disable error compensation function, 1- Enable 1D error compensation, 2- Enable 2D error compensation, 3-Enable 3D error compensation.
- ②1’st Encoder: [MapEncoder[1]], Select the encoder signal source for error compensation.
- ③Start index: [MapStartIndex], select the effective starting row of the error compensation table.
- ④# of Rows: [MapLength[1]], define the total number of rows in the position error compensation table, up to 1600 points.
- ⑤Start Position: [MapStartPos[1]], define the starting position of the error compensation table (encoder reading)
- ⑥Position gap: [MapPosGap[1]], position sampling interval
- ⑦Configure Table: Create an error compensation table according to the above user settings.

Note: [*] is the corresponding keyword.

- ⑧ Error compensation table: After clicking “⑦Configure Table”, the error compensation table shown in ⑧ will be automatically generated (Note: the table can display a maximum of 15 rows per page. If the total number of rows exceeds 15, the following setting window will be displayed. Change the value of 1’st Enc to change the starting position of the table, and then click “Update Visible Table” to update the error compensation table to the position set by the user). The first page and the first row of the compensation table are MapTable[1], and so on or click on the arrow to navigate to the next page of the error compensation table.



- ⑨ Error compensation table operation bar:

Reverse Encoders: This function is under development and is not supported for use yet.

Negate Table: Negate all compensation values in the error compensation table.

‘Zero’ Table: Set all the compensation values in the error compensation table to 0.

Copy to Clipboard: Copy all compensation values in the error compensation table to the clipboard.

Paste Table Data: Paste the compensation values on the clipboard

(Note: When the number of rows in the pasted data and the compensation table are inconsistent, a prompt window will pop up. Select “Yes” to replace the data according to the shortest matching length, and the number of rows exceeding will remain unchanged).

Load From File: Import the locally saved error compensation table.

Save to File: Export the error compensation table to the local.

⑩ ErrMapping function operation bar:

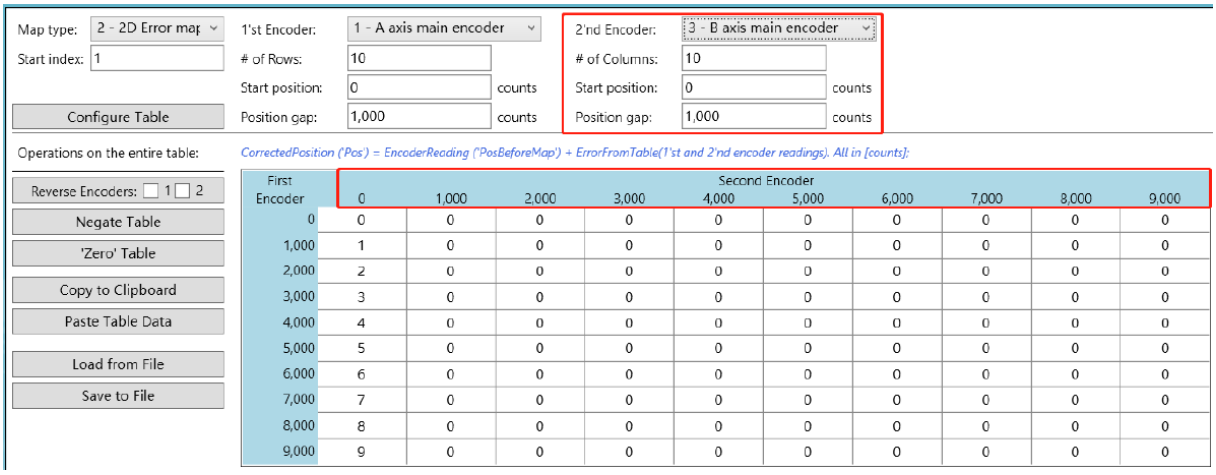
Reflash All: Refresh the page, it will revert back to the status that was previously applied upon “Apply All” or entering “enter button”.

Apply All: Apply the current parameter (the same function as clicking the "Enter" key), and the power is not saved if it is lost.

Save to flash: Save the current parameter to the controller Flash and save after power off.

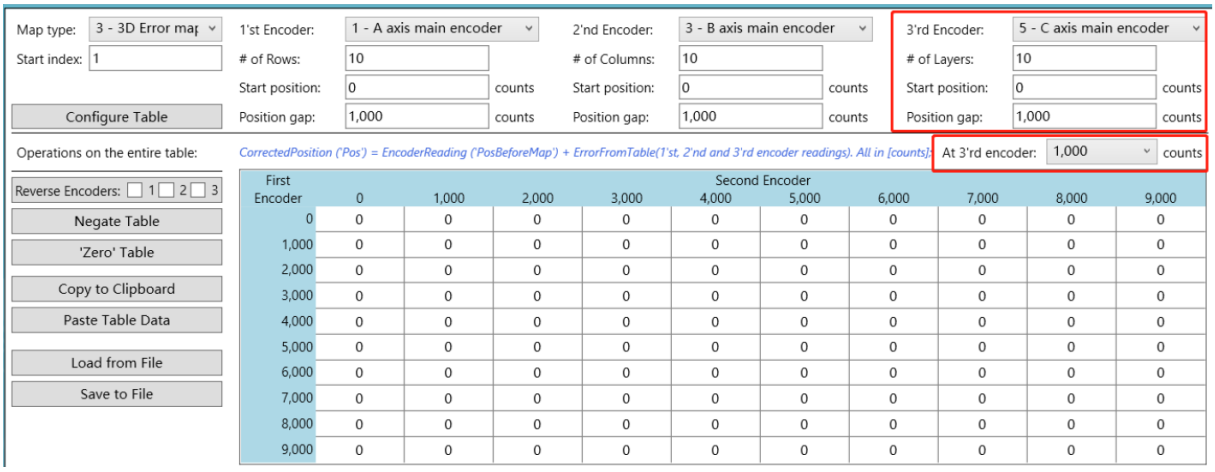
❖ 2D Error Compensation: Selecting the Map type as "2-2D Error map" will enable the 2-dimensional error compensation function, and the interface is as follows:

Compared with 1-dimensional error compensation, 2-dimensional error increases the position reference of another dimension coder. In "2'nd Encoder", select another associated axis encoder that needs to perform 2-dimensional motion. The method can refer to 1-dimensional position error compensation and is used in conjunction with the first encoder position common to query the error compensation table. The query error compensation value plus the axis encoder position to obtain the corrected position.



❖ 3D Error Compensation: 3D error compensation: Select Map type as "3-3D Error map", and the 3D error compensation function will be enabled. The interface is as follows:

Compared with 2-dimensional error compensation, 3-dimensional error is to add another dimension encoder position reference. In "3'rd Encoder", select another correlation axis position range that needs to be performed in 3-dimensional motion. The method can refer to 1-dimensional position error compensation. The three-dimensional error compensation table shows the 2-dimensional error compensation value corresponding to each Gap in the third dimension. The 3-dimensional position point selects the corresponding position switching through "At 3'rd encoder". The three encoders are used to query the position error compensation table.



Linear interpolation is performed between points in the query table and assume that these tables are arranged at equal intervals. If the encoder readings are outside the table (below the start position or above the end position), the value of the first or last point in the table will be used as error correction.

3 Operation steps

Before enabling error compensation function, please complete the following:

- The motor can move normally and meet the user's motion setting error requirements.
- The homing sequence has been set and can be normal to ensure that the motor zero position (Pos=0) position is the zero-position required by the user. This step is a key step because the compensation position uses zero point as the position reference point.
- Please return to zero before activating laser motion calibration and make sure that the error compensation function is turned off (MapType=0)

3.1 1D Error Mapping

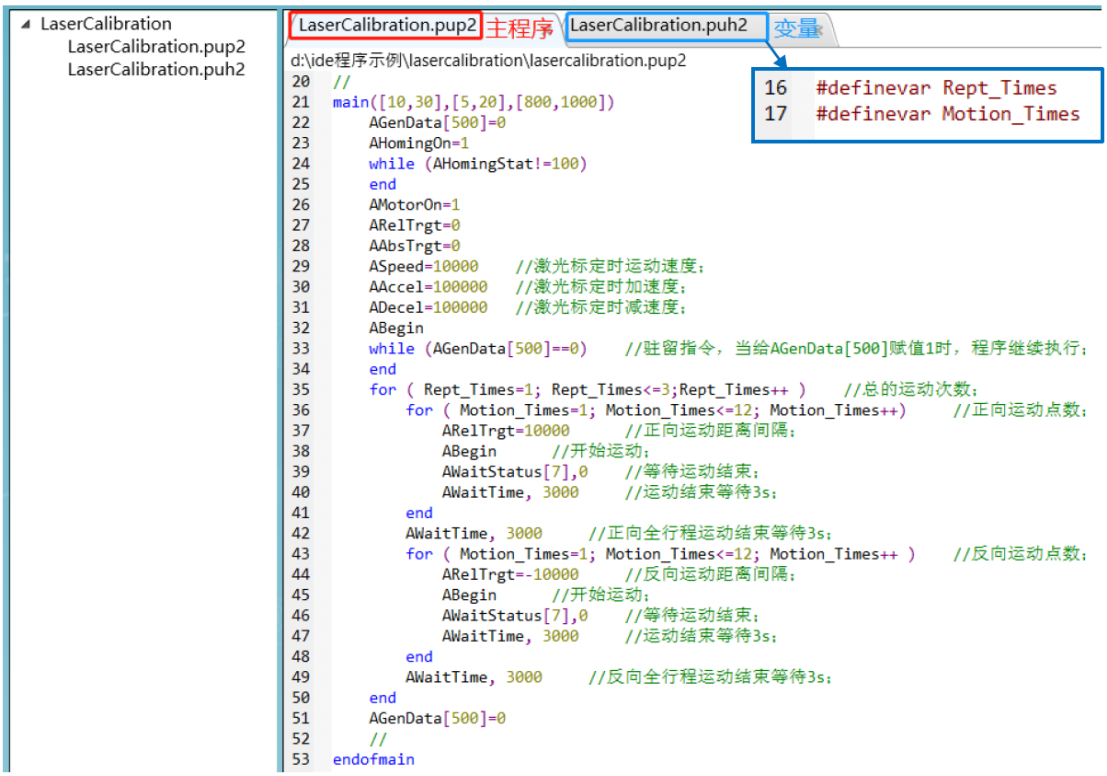
Step 1: Homing

User to run their desired homing sequence (excluding error mapping function).

Step 2: Write the motor running IDE Program

The following sample program demonstrates that the A axis moves forward continuously 12 times, then it will move backwards continuously 12 times, the full stroke will repeat for 3 times. The user can adjust the relevant parameters according to actual needs.

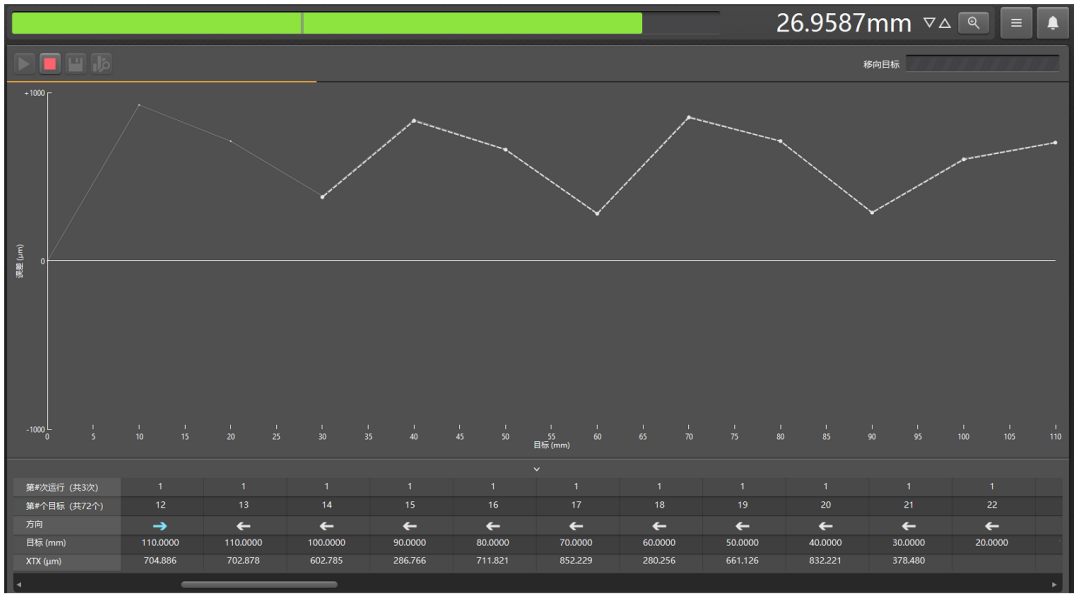
After setting the parameters, click “Compile and Debug” → “Run/Restart” to run the program. The program will reside in line 33 and 34 until the user sends a trigger to turn on “AGendata[500]”



Step 3: Start the motion and use the laser interferometer to collect the position data

Set up the laser interferometer and set the relevant parameters to start collecting the position discrepancy data. Once it is completed, user can send the command “AGenData[500]=1” in the PCSuite Terminal and motor will move according how it was programmed in the IDE.

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



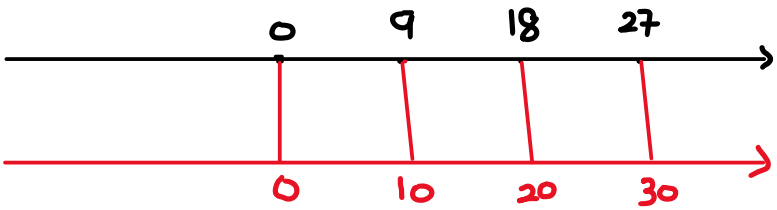
Step 4: Edit the Error Compensation Table

According to the above steps, convert the error compensation values collected into Counts (note: the controller can only use Counts as the position unit), and fill the corresponding compensation values into the error compensation table of the corresponding axis in PCSuite.

(Note: Please ensure that the laser interferometer direction convention is the same as the encoder direction else there is a need to negate the error that is collected from the data)

Example 1: Laser Interferometer Convention is the same as Encoder Direction

Laser → Positive Convention, Encoder → Positive Convention



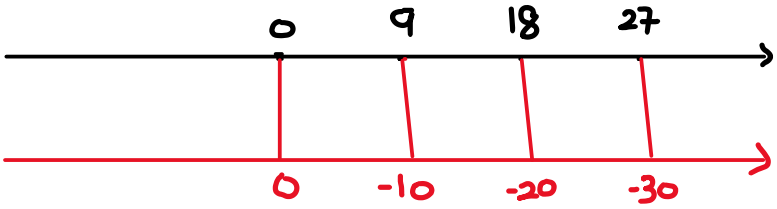
Let's say at Encoder (10mm, 20mm, 30mm), in an ideal situation I would be expecting my actual position to be at 10mm,20mm,30mm. However, this is not always the case and when the laser measurement was taken, it recorded that instead of 10mm,20mm,30mm, it was 9mm,18mm,27mm. Hence, the error collected would be (-1mm, -2mm, -3mm).

In the error mapping table, we will key in the error we read in counts as we want the encoder raw reading to move further in order to reach its target position. The equation we used for error mapping is "CorrectedPosition('Pos') = EncoderReading ("PosBeforeMap") + ErrorFromTable", hence if we were to attain 'Pos' to be at 10mm, we will need to compensate for the -1mm error which in order words increasing my EncoderReading. "10mm = 11mm – 1mm", we will need to key in -1mm (normalized to counts) in the error mapping table.

Position (1mm = 1000 counts)	Error (1mm = 1000 counts)
10000	-1000
20000	-2000
30000	-3000

Example 2: Laser Interferometer Convention is the same as Encoder Direction

Laser → Positive Convention, Encoder → Negative Convention



Let's say we keep everything the same except for the convention of the encoder direction. The error collected from the data will still remain as (-1mm, -2mm and -3mm) but in the error mapping table we will have to negate the error mapping data. "-10 (10mm) = -11 (11mm) + 1mm (negated error collected from laser).

According to the equation given for example 1, it is expected for the encoder reading to move further down the → line to attain 10mm position hence we need to key in an error of 1mm instead of -1mm due to the convention of the encoder to let it travel more to reach our Target Position.

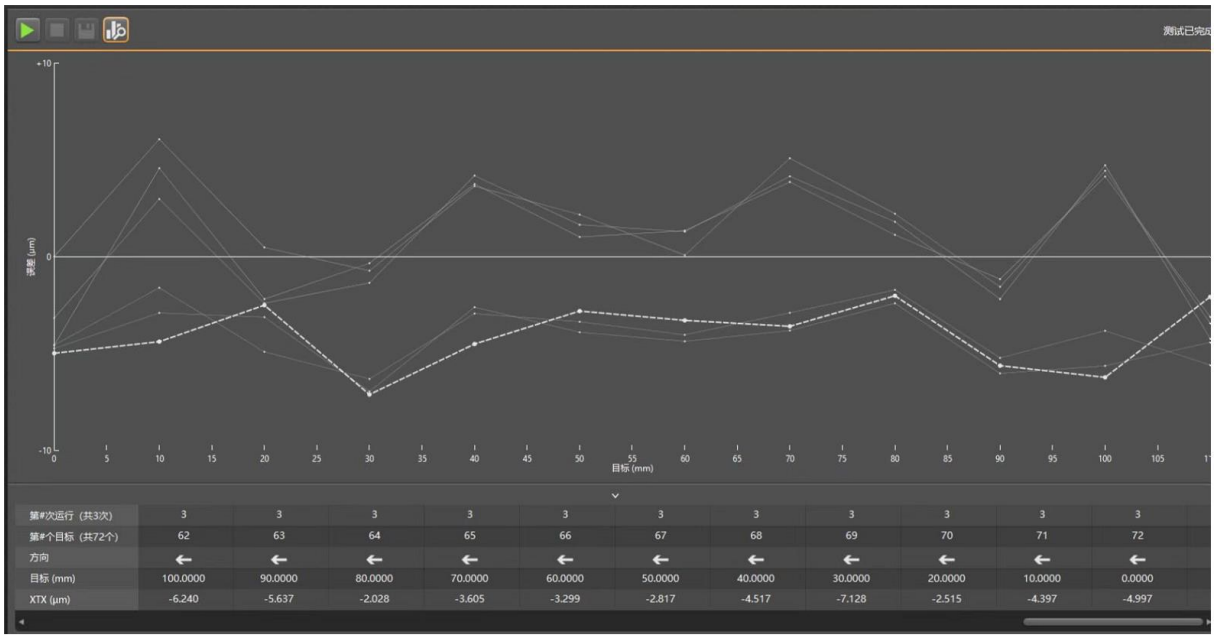
Position (1mm = 1000 counts)	Error (1mm = 1000 counts)
-10000	1000
-20000	2000
-30000	3000

Step 5: Measure the physical deviation after correction

After setting the error compensation table, click "Apply All" to apply the parameters, and then repeat steps 1, 2, and 3 to measure the physical deviation after correction using the error compensation table.

It is worth noting that before this test, error compensation needs to be turned on, "XMapType=y, X represents the axis number and y represents the error compensation type (No Error Mapping ,1D, 2D, 3D).

Comparing the images before and after the error compensation is turned on measured by the laser interferometer, it can be seen that the error value is reduced from the maximum of about 1000um before compensation to the maximum of about 6um



Step 6: Multiple measurements to obtain the best compensation effect

Based on the actual deviation image obtained in step 5, repeat the above steps 3, 4, and 5, fine-tune the error compensation table according to the principle of more reduction and more addition, and measure multiple times to obtain the best motion performance

3.2 2D Error Mapping

The 2D error compensation method is similar to the 1D error compensation method, except that the error compensation table is changed from a linear table to a 2D table, and X and Y are used as reference positions, which is equivalent to measuring multiple sets of 1D compensation.

(Note: The Error Compensation table is only compensating on the Axis you have turned the function on at, it does not apply to the axis you take the encoder reading from. Lets say you are taking reference for X encoder reading and Y encoder reading and this Error Mapping Function is turn on for X axis only, it will only compensate the error on the X Encoder. A dedicated error mapping for Y Axis would be necessary if you would want to compensate for Y as well).

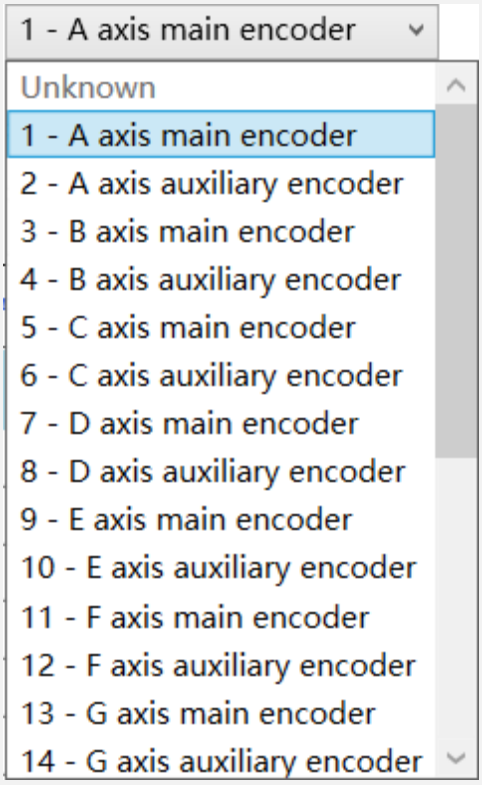
3.3 3D Error Mapping

3D error compensation adds Z direction as reference position on the basis of X and Y, which is equivalent to measuring multiple sets of 2D compensation values. 2D error is measured at each Z gap for compensation.

(Note: Refer to the note mentioned in 2D Error Mapping)

4 Keywords

The Agito keyword can be used not only in PCSuite Terminal and IDE programming environment, but also in user communication via string or ASCII.

Keyword	Description
MapType	*MapType is used to define the error compensation type.* *0-Disable error compensation* *1-Enable 1D error compensation* *2-Enable 2D error compensation* *3-Enable 3D error compensation*
MapEncoder	*MapEncoder[x] ,x=1,2,3, respectively represent the encoder signal sources in the three dimensions of X, Y, and Z;* *The following are the meanings of the values (the diagram does not show the complete picture):* 
MapLength	*MapLength[x] , x=1,2,3, respectively represents the total number of rows in the compensation table of the three dimensions X, Y, and Z. The maximum data grid is 65535 points*
MapStartPos	*MapLength[x] , x=1,2,3, respectively represents the starting position of the compensation table in the three dimensions of X, Y, and Z*
MapPosGap	*MapLength[x] , x=1,2,3, respectively represents the position interval of the compensation table in the three dimensions of X, Y, and Z*
MapTable	*MapTable[y] ,y=1,2,3…, respectively represent the error compensation table points.*
MapStartIndex	*The starting row in the MapTable is usually set to 1, unless multiple error compensation tables need to be saved.*