Application Note

CMMP Endless positioning

Using CMMP-AS in combination with multiturn encoders for applications, which require endless positioning.

CMMP-AS

100003

Title .... CMMP Endless positioning
Version 1.10
Document no. 100003
Original .... en
Author .... Festo

Last saved 01.10.2014

Legal Notice

Hardware, software, operating systems and drivers may only be used for the applications described and only in conjunction with components recommended by Festo AG & Co. KG.

Festo AG & Co. KG does not accept any liability for damages arising from the use of any incorrect or incomplete information contained in this documentation or any information missing therefrom.

Defects resulting from the improper handling of devices and modules are excluded from the warranty.

The data and information specified in this document should not be used for the implementation of safety functions relating to the protection of personnel and machinery.

No liability is accepted for claims for damages arising from a failure or functional defect. In other respects, the regulations with regard to liability from the terms and conditions of delivery, payment and use of software of Festo AG & Co. KG, which can be found at www.festo.com and can be supplied on request, shall apply.

All data contained in this document do not represent guaranteed specifications, particularly with regard to functionality, condition or quality, in the legal sense.

The information in this document serves only as basic information for the implementation of a specific, hypothetical application and is in no way intended as a substitute for the operating instructions of the respective manufacturers and the design and testing of the respective application by the user.

The operating instructions for Festo products can be found at www.festo.com.

Users of this document (application note) must verify that all functions described here also work correctly in the application. By reading this document and adhering to the specifications contained therein, users are also solely responsible for their own application.

Internet: http://www.festo.com
E-Mail: service_international@festo.com

Table of contents

1 Components/Software used

Type/Name	Version Software/Firmware	Date of manufacture
CMMP-AS- -M3	Up to version 4.0.1501.1.3	
CMMP-AS- -M0	Up to version 4.0.1501.1.3	
CMMP-AS-	Up to version 3.5.1501.2.1	

Table 1.1: Components/Software used

2 Application description

Endless positioning with CMMP-AS Servo drive and motors with multiturn encoder.

Whenever using a multiturn encoder in an application where endless relative movements are necessary, there can be a problem after one or more overflows from the multiturn encoder.

The multiturn encoder can store only 4096 absolute rounds (12 bits). After a power cycle the encoder will give a value which is inside this range (0...4096). This range is mapped into a range from -2048 to +2048 inside the controller. The controller converts this value with the factor group into the user defined position unit or modulo position.

The problem only occurs on unlimited axis or conveyor applications when the encoder range has one or more overflow(s) and an "all together gear ratio", which gives back not an integer number inside the multiturn range from the application working positions. For example a linear axis (Toothbelt conveyor) has to run every time 100 mm or a rotary table has to stop each 60°. The application working stroke has to be part of the calculation.

If there is more than one gearbox used in the application you have to calculate all the transmission translations to check, is the result an integer number.

After a power cycle the controller didn't know if or how often the overflow occurred. He will get only the new absolute position from the encoder and map and convert it like described.

The solution for this application is using a gear ratio which gives back an integer number inside the multiturn range.

That means that the absolute (modulo) position is always correct (before and after a power cycle).

2.1 Example 1: Block diagram of the machine / Conveyor

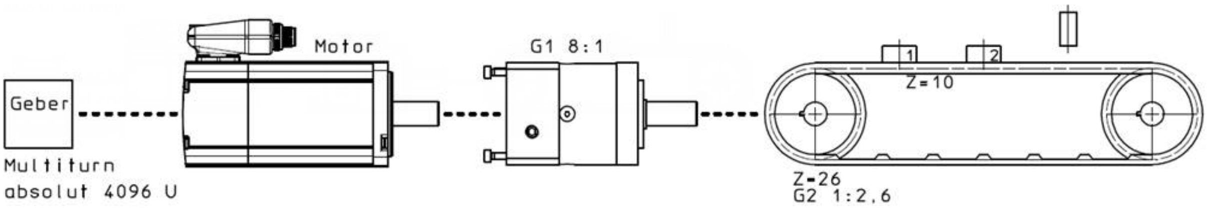

Fig. 2.1

The customer application is an endless rotative positioning system with tooth-belt. On the tooth-belt are goods carriers (nests) to bring the goods to a fixed processing position. The nest pitch is always 100mm.

The customer denied using any reference switch because the multiturn encoder always delivers the actual position.

Problem: After switching OFF/ON the machine, the encoder position does not fit to the conveyor position.

$$\text{Ü} = \frac{1}{8} * \frac{2,6}{1} * 4096 = 1331.2$$

Value is not an integer!

It is not possible to reproduce the zero position after multiturn overflow!

2.2 Example 2: Block diagram of the machine / Rotary table

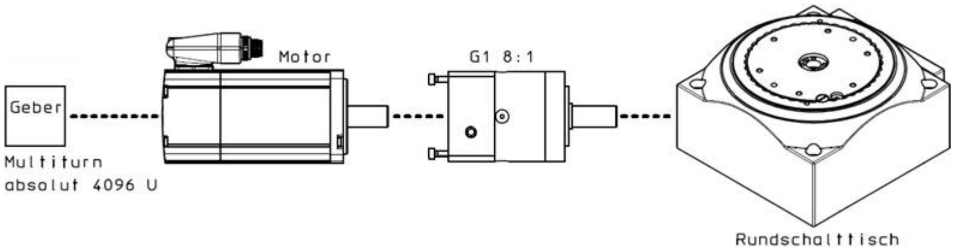

Fig. 2.2

Customer application is an endless rotary table with working positions every 90°.

$$\ddot{U} = \frac{1}{8} * 4096 * 4 = 512$$

Value is an integer ➔ good.
Multiturn overflow does not have any impact.

Customer application is an endless rotary table with working positions every 60°. Gear factor 12:1

$$\ddot{U} = \frac{1}{12} * 4096 * 6 = 2048$$

Value is an integer ➔ good.

Customer application is an endless rotary table with working positions every 90°. Gear factor 12:1

$$\ddot{U} = \frac{1}{12} * 4096 * 4 = 1365.33333$$

Value is not an integer ➔ not good.

- Please ensure in your application, that the total gear ratio of the combination from gear factor and table pitch ends up in an INTEGER.

3 Using the CMMP-AS to support the application

There are no special needs for the controller setup. Just set the axis in the configuration to “unlimited”.

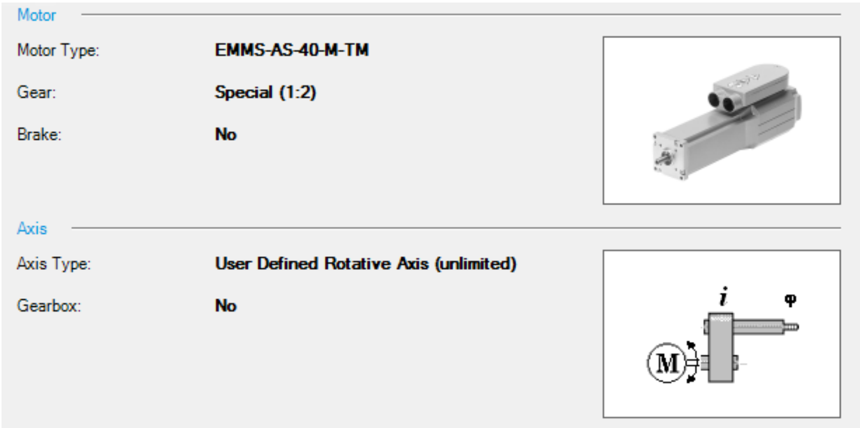

Fig. 3.1

The endless positioning function is selected by the selection hook "working / positioning unlimited". The selection is only for user defined linear - and rotary axes.

i The way to cover any kind of positioning differences after several encoder overflows is to get an integer number inside the multiturn range with the used application stroke.

The total gear translation must be calculated from all single gear translations used in the application.

3.1 Possible Solutions for Example 1

In the shown application 100 mm distance from nest to nest should end in an integer number inside the multiturn range. Here the calculation with the original application conditions.

Number of nest intervals in multiturn range:

$$Ü = \frac{1}{8} * \frac{2.6}{1} * 4096 = 1331{,}2$$

3.1.1 Conveyor Application

Current Setting / Current Conveyor		
Motor	4096	turns absolute position
Gearbox	8 1	primary secondary
Output gear	512	turns absolute position
Conveyor		
– Primary Role	26	teeth
– Tooth pitch	10	mm
– Pitch for nests	100 10	mm teeth / nest interval
– Gear Ratio conveyor	2.6	
– No. of nest intervals in multiturn range	1331.2	not good - result is not an integer!

Table 3.1

3.1.2 Examples to reach the required integer number inside the multiturn range.

Example: Conveyor modification 1		
Motor	4096	4096
Gearbox	8 1	primary secondary
Output gear	512	turns absolute position
Conveyor		
– Primary Role	26	teeth
– Tooth pitch	10	mm
– Pitch for nests	130 13	mm teeth / nest interval
– Gear Ratio conveyor	2	
– No. of nest intervals in multiturn range	1024	good!

Table 3.2

$$Ü = \frac{1}{8} * \frac{2}{1} * 4096 = 1024$$

Example: Conveyor modification 2	Change Gearbox	
Motor	4096	turns absolute position
Gearbox	208 20	primary secondary
Output gear	393,846154	turns absolute position
Conveyor		
– Primary Role	26	teeth
– Tooth pitch	10	mm
– Pitch for nests	100 10	mm teeth / nest interval
– Gear Ratio conveyor	2.6	
– No. of nest intervals in multiturn range	1024	good!

Table 3.3

$$Ü = \frac{20}{208} * \frac{2.6}{1} * 4096 = 1024$$

Example: Conveyor modification 3	**Change primary Role**	
Motor	4096	turns absolute position
Gearbox	8 1	primary secondary
Output gear	512	turns absolute position
Conveyor		
– Primary Role	25	teeth
– Tooth pitch	10	mm
– Pitch for nests	100 10	mm teeth / nest interval
– Gear Ratio conveyor	2.5	
– No of nest intervals in multiturn range	1280	good!

Table 3.4

$$Ü = \frac{1}{8} * \frac{2.5}{1} * 4096 = 1280$$

Example: Conveyor modification 3a	**Change gearbox + pitch of nests**	
Motor	4096	turns absolute position
Gearbox	13 1	primary secondary
Output gear	315,076923	turns absolute position
Conveyor		
– Primary Role	26	teeth
– Tooth pitch	10	mm
– Pitch for nests	80 8	mm teeth / nest interval
– Gear Ratio conveyor	3.25	
– No of nest intervals in multiturn range	1024	good!

Table 3.5

$$Ü = \frac{1}{13} * \frac{3.25}{1} * 4096 = 1024$$

Example: Conveyor modification 3b	change gearbox + pitch of nests	
Motor	4096	turns absolute position
Gearbox	26 3	primary secondary
Output gear	472,615385	turns absolute position
Conveyor		
– Primary Role	26	teeth
– Tooth pitch	10	mm
– Pitch for nests	120 12	mm teeth / nest interval
– Gear Ratio conveyor:	2.16666667	
– No of nest intervals in multiturn range:	1024	good!

Table 3.6

$$\text{Ü} = \frac{3}{26} * \frac{2.16666667}{1} * 4096 = 1024$$

3.2 Possible Solutions for Example 2

The Solution is to translate the multiturn range of the encoder (typical 4096 turns) into an integer number of turns at the working stroke. If there is more than one gear mounted in the application, you have to calculate all gear factors to the total gear factor.

The total gear factor multiplicated with the 4096 multiturn range must result in an integer number.

Possible modifications:

- Change number of working positions (pitch).
- Change gear from the motor.
- Use a second gear in the rotary table.

$$\text{Ü} = \frac{1}{3} * \frac{1}{2} * \frac{X}{X} * 4096 = \text{INTEGER VALUE}$$

4 Controller specific Rounding with relative movements

If one is using relative movements, please note the following:

The CMMP controller is a 32-bit controller. That means, the controller calculates internal with 4294967296 increments position range. The controller calculates with integer values. When traversing in which no integer is as a result, the controller round up to the nearest integer. This so called quantization error might lead to deviations in the continuous positioning.

Example to show the rounding

Rotary indexing table 4 positions. (90 °) = 4294967296 : 4 = 1073741824 ➔ Integer.
6 positions. (60 °) = 4294967296 : 6 = 715827882.666666 ➔ not Integer.

The controller will calculate with 715827883

i This rounding deviation, even with a 32-bit controller, can play a role in practice, depending on the clock rate and the required accuracy of the system.

4.1 Examples for postion deviation caused by quantisation error

Mechanics / Application

– 5 cycles per second.
– Direct drive motor drives gearwheel and toothbelt.
– Feed 100 mm / revolution
– Accuracy: 0.05 mm required for continuous operation.

Resulting requirement on the positioning accuracy of the shaft: 0.0005 revolutions (0.5 per thousandth)

4.1.1 Calculation with CMMS-AS (16-Bit controller)

Accurate position adjustment: $1/2^{16}$ revolutions

The maximum deviation from the ideal relative position width is 1/216 == 0.00001526

Number of cycles, until limit accuracy is reached:

$$\mathrm{N} = \left(\mathrm{INT}\ \frac{0.0005}{\left(\frac{1}{2^{16}}\right)}\right) + 1 = 33$$

$$33 * 0.2\ \mathrm{s} = 6.5\ \mathrm{s}$$

The accuracy limit of 0.05 mm might be reached in worst case after 6.5 s, if no precautions for compensation are considered.

4.1.2 Calculation with CMMP-AS (32-Bit controller)

Accurate position adjustment: $1/2^{32}$ revolutions

Number of cycles, until limit accuracy is reached:

$$\mathrm{N} = \left(\mathrm{INT}\ \frac{0.0005}{\left(\frac{1}{2^{32}}\right)}\right) + 1 = 2.14\ \mathrm{million}$$

The accuracy limit of 0.05 mm might be reached in worst case after 120 h, if no precautions for compensation are considered

4.2 Possible solutions to compensate for quantisation error and to achive optimum acurracy:

4.2.1 Reference position

- Make a reference drive to a reference switch, zero pulse or block before your system gets unaccurate.

That is the easiest solution, but not allways possible.

The disadvantage is, that it requires cyclic reference runs.

4.2.2 Choose the position width as an integer number of increments

- Choose a relative position width, which is represented by an integer number of increments inside the motor controller.
- Refer to Example 2 and using a CMMS-AS with 2^{16} increments per motor revolution:
- Chose a relative position width of...

Table	Motor shaft	Increments of motor controller
90°	720°	20000_h
45°	360°	10000_h
22,5°	180°	8000_h
11,25°	90°	4000_h
...	45°	2000_h
...	22.5°	1000_h
...	11.25°	0800_h

Table 4.1

- If you like to position at 15°, 30°... select a gearbox with the appropriate Gearsetting.

Example

Choose a gearbox with 1:6

Table	Motor shaft	Increments of motor controller
120°	720°	20000_h
60°	360°	10000_h
30°	180°	8000_h
15°	90°	4000_h
7,5°	45°,	2000_h
3,75°	22.5°	1000_h
....	11.25°	0800_h

Table 4.2

4.2.3 Add an absolute movement in modulo mode (only CMMP)

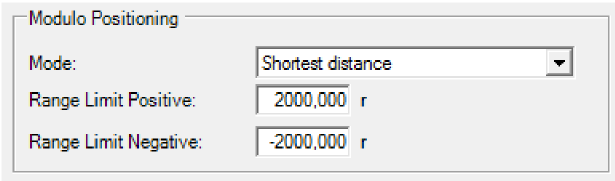

Fig. 4.1

Rotary indexing table 12 positions. (30 °) = 4294967296 : 12 = 357913941.333333 ➔ not Integer.
The controller will calculate with 357913941.

You can avoid this rounding deviation by adding an absolute positioning drive, using modulo positioning with Mode: shortest distance. This absolute positioning will balance your increasing rounding error.

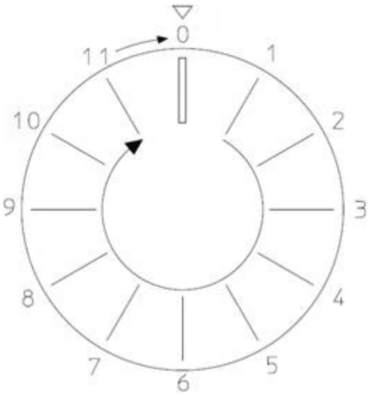

Fig. 4.2

Pitch is 30°. After 12 cycles you are at “0” again. With rounding errors, 0.333333 Increments every Pitch.
Drive eleven times with “relative” 30°.
Then an “absolute” drive Shortest distance to “0”.
This works for every value of cycles which fits to your stroke and Zero-Position.
Ex. 2400 cycles means 2399 * REL 30° + 1 * ABS 0°

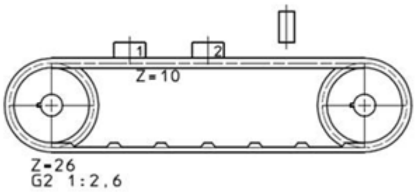

Fig. 4.3

Pitch is 100mm = 10 Teeth = (1/2,6 U) per pitch.

Ex. After 100 modulo rounds which equals 260 pitches your modulo axis will reach mechanical “0” position again.
Drive 259 times with “relative” (1/2,6)U = 0,384615U
Then an “absolute” drive Shortest distance to “0”.

This works for every value of cycles which fits to your stroke and modulo zero-position.