Motion Control Products

Application note
Rotary Knife (PLCopen motion)

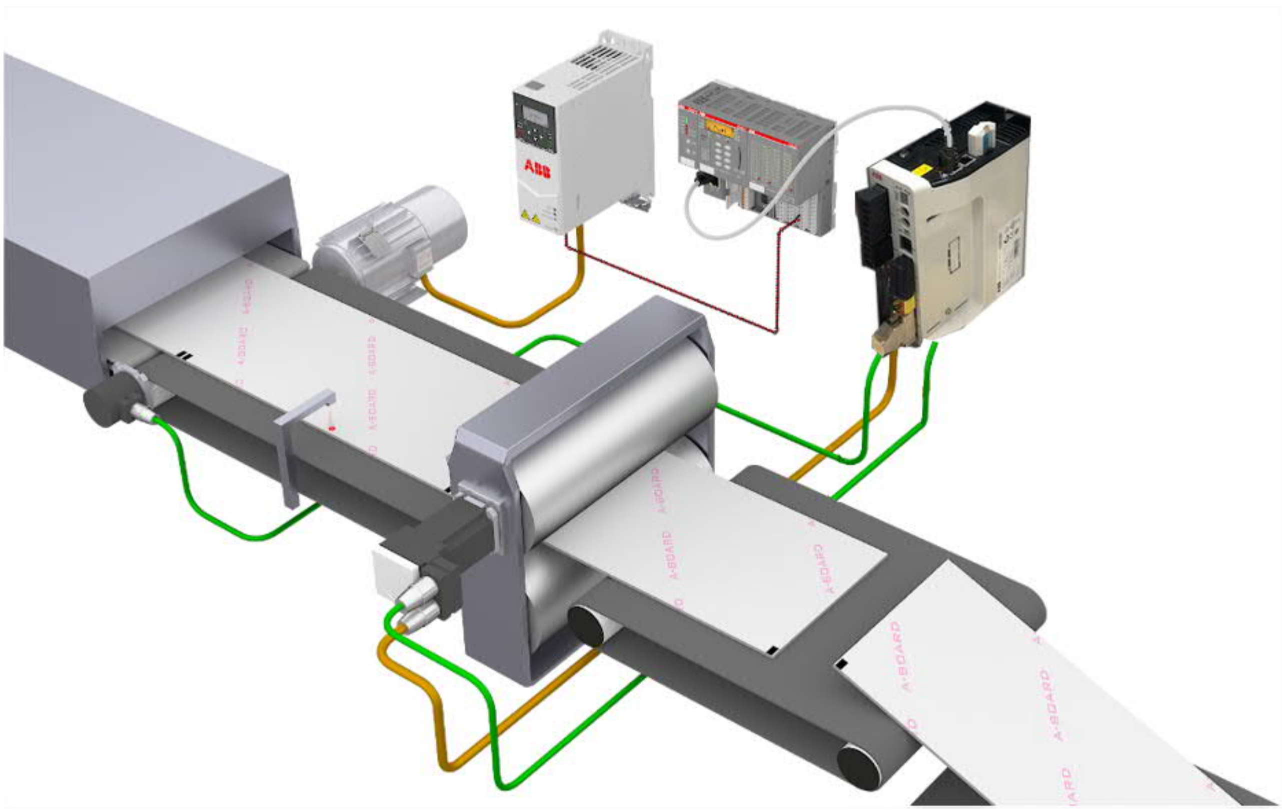
AN00253

Rev B (EN)

ABB motion servo drive products integrate seamlessly with the AC500 PLC and PS552-MC-E motion library, allowing applications such as rotary knives to be realised with ease

Introduction

A typical configuration for a rotary knife application is illustrated below.



An ABB AC500 PLC is used to control the whole system. The material flow is started / stopped via an ABB ACS380 invertor controlled by the PLC (e.g. via digital I/O). An encoder coupled to the material transfer system is connected to the ABB servo drive (MicroFlex e190 in this example, but this could equally be a MotiFlex e180 for larger systems). The MicroFlex e190 additional encoder input (or encoder splitter option card OPT-MF-200) allows for simple and convenient connection of the ABB servo motor’s digital encoder (e.g. Smartabs when using a motor from the ESM motor range) as well as the material encoder.

Rotary Knife example

Available to accompany this application note is an example Automation Builder v1.2 project. This project has been written for a PM591 processor fitted with CM579-ETHCAT coupler and a MicroFlex e190 drive connected via EtherCAT (but is easily converted to suit other configurations – note that a PM585 or any PM59x processor must be used for EtherCAT motion applications).

The project uses version 3.2.0 of the PS552-MC-E PLCopen motion control library so you will need a licensed version of this installed. The project references the MicroFlex e190 as a device included in Automation Builder via installation of the Mint servo drives package v1.2.4.0, so it may also be necessary to install this package via Tools>Installation Manager… before opening the example projects if you are not using a version of Automation Builder that ships with this package already installed. Please refer to the Automation Builder help system and application note AN00205 for further information about installing packages if needed. This application note includes the relevant Mint servo drives package file for convenience.

Please contact CN-motionsupport@cn.abb.com if you would like to be sent a copy of this example Automation Builder project.

An AC500 PLC controls the motion of the Knife roll, initially homing this based just on the axis' current position (in a real application the home may be performed via a home sensor connected to one of the drive's digital inputs – see application note AN00220 for further details about EtherCAT homing methods). During the homing process the system ensures the material is cut (and the waste material removed) so that the machine first starts with the lead edge of the material at the cut point.
The home position is adjusted so the knife starts its motion from the midpoint between knives (any number of knives are possible, this might typically vary from 1 to 4 depending on product cut length).
When the knife starts the velocity/position profile is controlled in such a way that the very first cut length is correct. From this point onwards the knife profile is adjusted to ensure consistent cut length regardless of machine speed and system settings.

Controls are provided to allow the user to adjust (at any time):

- Cut length
- Knife sync angle (the angle over which the knife surface speed matches the material speed)
- Cut ratio (the speed ratio between the knife surface speed and the material linear speed)

The system also allows the number of knives and home position to be adjusted, but these only take effect when the system is homed and have no effect during a running cycle.

The example PLC application assumes the following mechanical/system properties:

- Knife motor uses encoder with resolution of 131072 counts per rev
- Knife diameter of 100mm
- Knife gear ratio (from motor to roll) of 10:1
- Material encoder resolution of 20000 counts per rev (quadrature)
- Material encoder driven directly from roll with 50mm diameter

These properties are all defined by constants declared as part of the Global_Variables module within the PLC application code and are easily adjusted to suit any application.

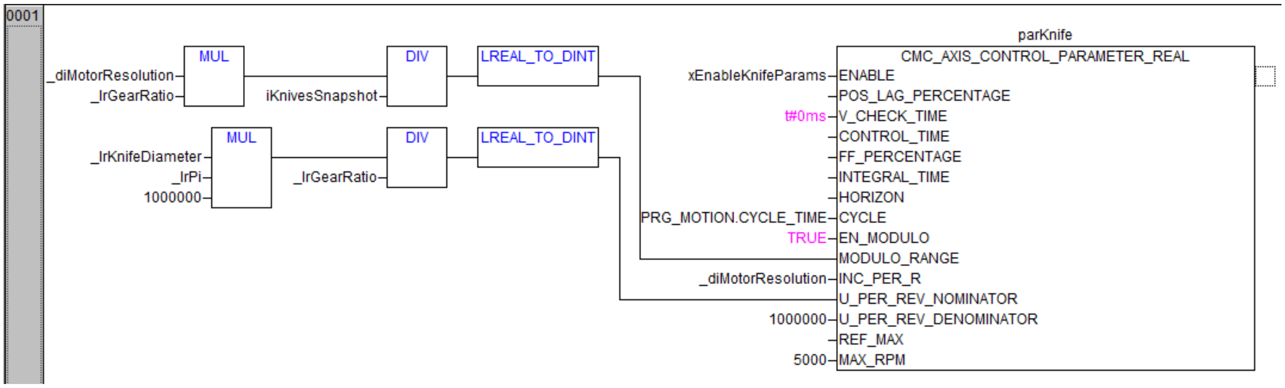
A master encoder should be wired to the MicroFlex e190's secondary encoder input (via the optional encoder splitter module).

Scale factors
The scale factor allows an axis to be scaled into engineering units for ease of use and can take into account any mechanical linkages such as gearing. The scale factor is applied to all motion variables for an axis (speed, acceleration, move distance, etc.). For this example we wish to scale our knife axis into linear mm of travel around the knife circumference, so for the example mechanical setup our scale factor can be worked out as follows:

We are driving our knife roll via a 10:1 speed reduction gearbox…

- 10 revolutions of the motor will produce one revolution of the knife
- One knife revolution = 100 * Pi mm of linear travel around the circumference
- One motor revolution = 100 * Pi / 10 = 10 * Pi mm of linear travel around the circumference
- One motor revolution = 131072 encoder counts

With this information we can program our PLCopen CMC_AXIS_CONTROL_PARAMETER_REAL function block for the knife axis as shown below…

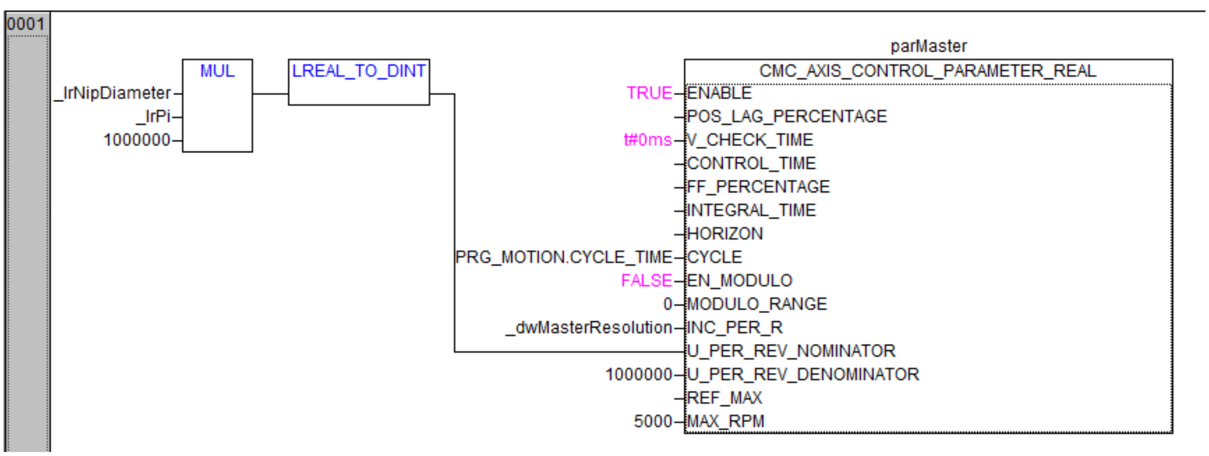


CYCLE is set to a constant with a value of 2ms to suit the EtherCAT cycle time set in our Automation Builder device configuration.
EN_MODULO is set to true because the axis continually runs in one direction and wraps every rev of the knife.
MODULO_RANGE is set to the number of counts in one cycle of the knife (where a cycle is the distance between knives).
INC_PER_R is set to match the number of encoder counts in one revolution of our motor.
U_PER_REV_NOMINATOR and U_PER_REV_DENOMINATOR are set so that between them we arrive at our value of 10 * Pi units per motor rev (i.e. 100 * Pi * 1000000 / (10 * 1000000) = 10 * Pi).

As we are using an encoder connected to the servo drive to represent material travel, this too must be used to create an axis position and must also be scaled….

- 1 rev of the material encoder = 50 * Pi mm of linear travel of the material
- 1 encoder revolution = 20000 counts

With this information we can program our PLCopen CMC_AXIS_CONTROL_PARAMETER_REAL function block for the master axis as shown below…



CYCLE is set to a constant with a value of 2ms to suit the EtherCAT cycle time set in our Automation Builder device configuration.
EN_MODULO is set to false because although the axis continually runs in one direction the application code always shifts the axis position backwards after every cut so it never effectively gets any larger than the maximum cut length.
MODULO_RANGE is therefore set to zero.
INC_PER_R is set to match the number of encoder counts in one revolution of our material encoder (20000).
U_PER_REV_NOMINATOR and U_PER_REV_DENOMINATOR are set so that between them we arrive at our value of 50 * Pi units per encoder rev (i.e. 50 * Pi * 1000000 / 1000000 = 50 * Pi).

Principle of operation

Much like application note AN00226 (written for Mint applications) the rotary knife linear velocity is synchronized to the material speed at the point of cut only. To achieve this, and to allow the user to adjust all necessary application parameters (cut length, sync angle, cut ratio) at any time, the axis velocity is generated by cam table data for each cycle of the knife (where a cycle is the distance between each knife blade).

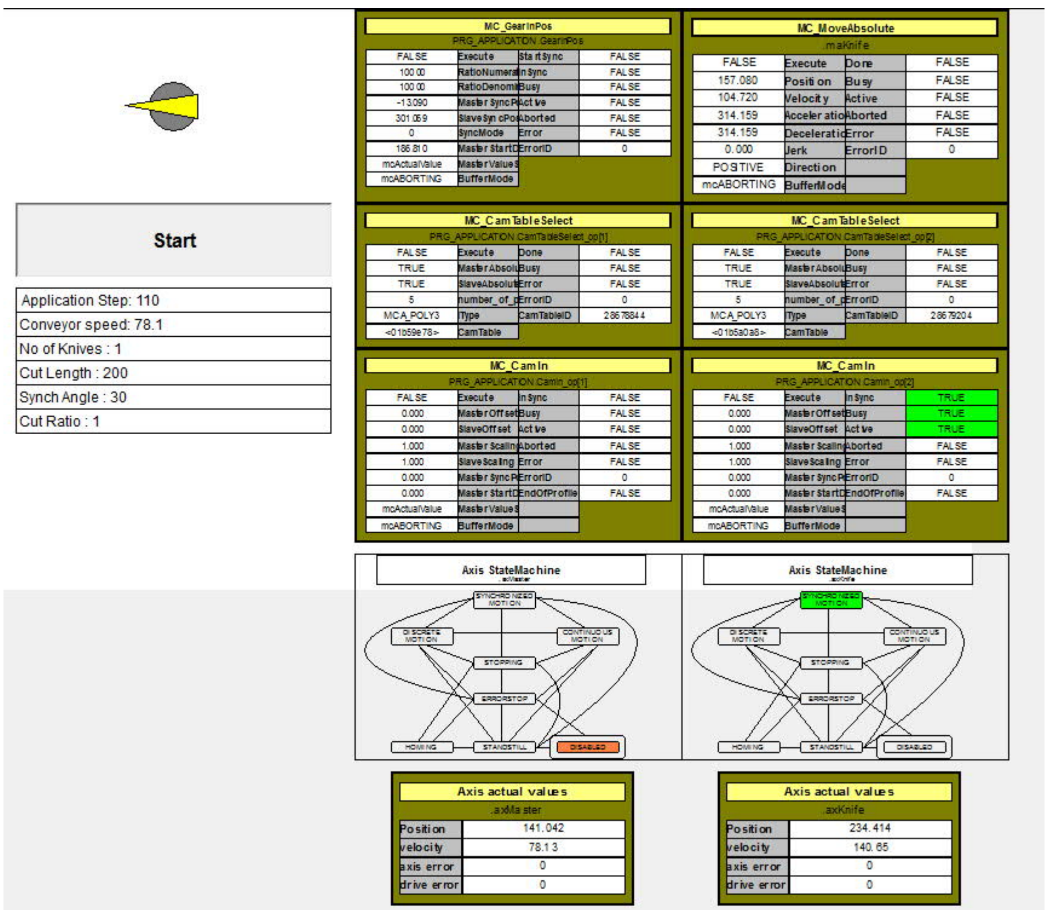
Two cam tables (and hence two MC_CAMIN and two MC_CAMTABLESELECT function blocks) are used to allow the cutting parameters to be changed “on the run”. The code switches between the two sets of cam data every cycle, at a point in the cycle where the master and slave axis positions are identical regardless of whether the cutting parameters have changed or not (i.e. at the point of cutting the material).

Because the cam data for both the slave and master axes is absolute we have to ensure the axis positions cycle/wrap as necessary. The knife axis position wraps automatically because of the modulo settings for this axis (as described earlier). The master axis however is non-modulo. Instead we ensure this axis cycles up to the required cut length by shifting its position (instantaneously, using MCA_SETPOSITIONCONTINUOUS) backwards at the point the knife cuts the material (i.e. when the master axis position reaches the required cut length).

To start the knife initially the MC_GEARINPOS function block is used to ensure the knife axis travels from the start position to the beginning of the synchronous region in such a way that the first cut is the correct length (this assumes the material was initially cut off during the homing process).

Using the example project

The example project available to accompany this application note includes a simple visualization to control the operation of the rotary knife…

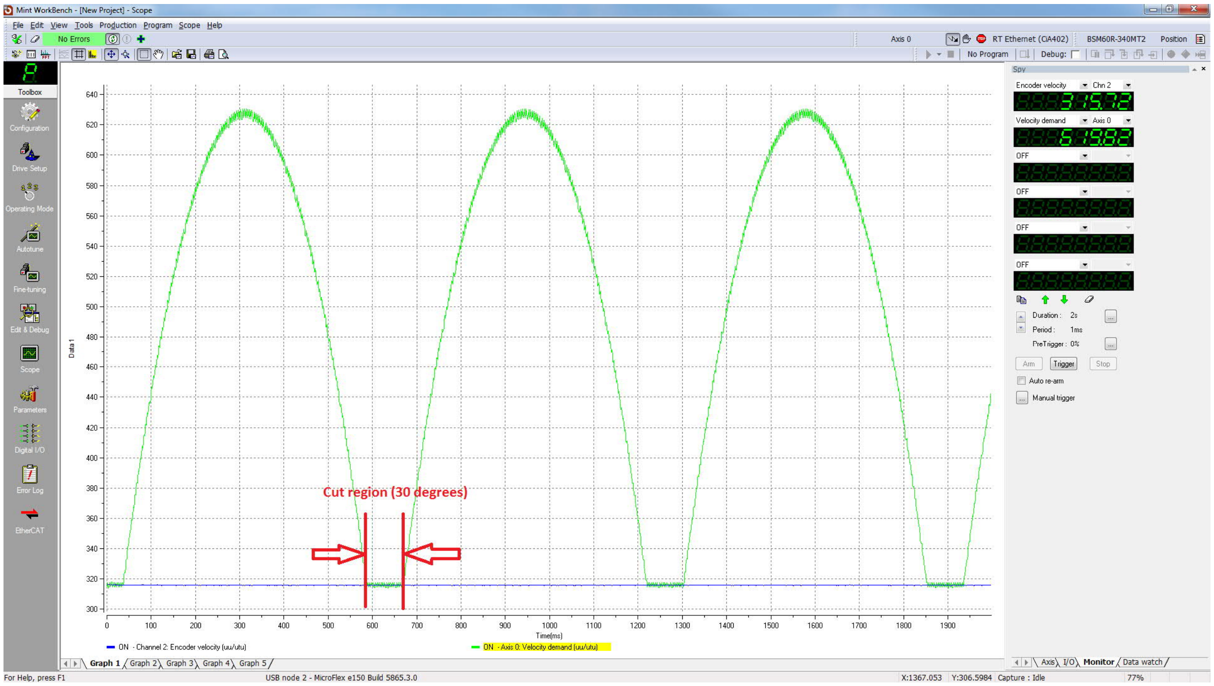


Initially ensure the master encoder is stationary (there is a small visualization to show the current position and speed of both the master and knife axes at the bottom of the screen). The axis should then enable and home automatically when the Start button is pressed (providing there are no errors). To reset an error condition deselect and reselect the Start button if required

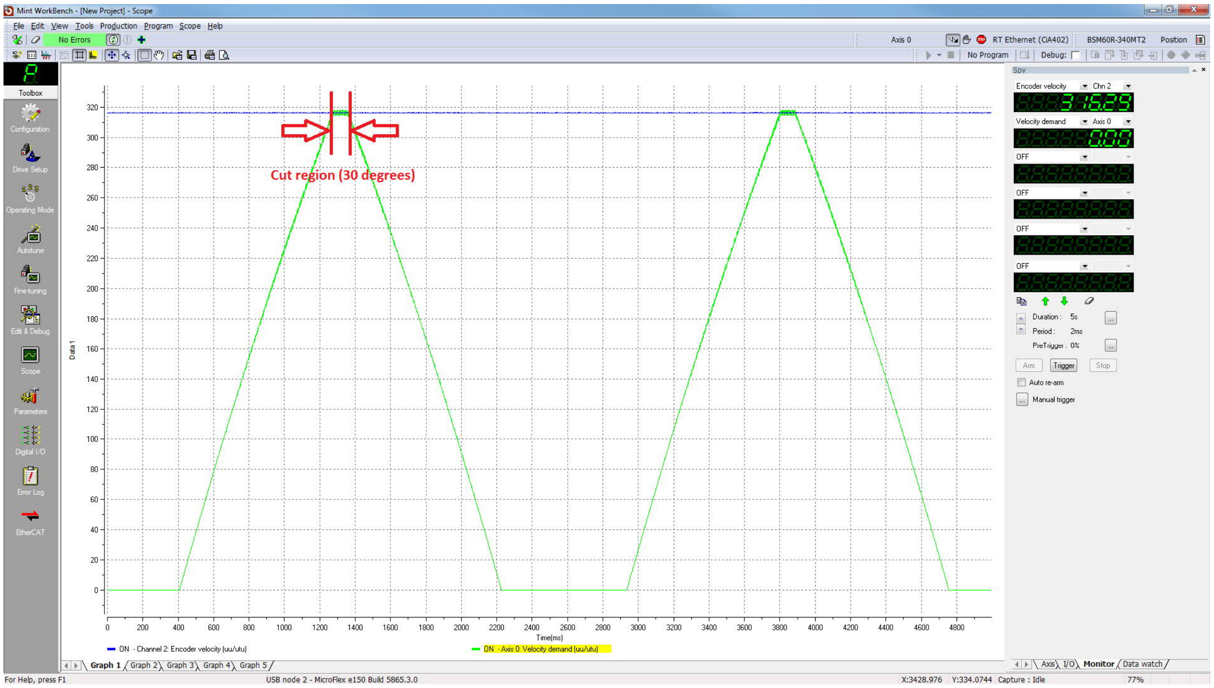
At the top left of the screen is an animation to show the current absolute position of the knife cylinder (which will also show multiple knives if the number of knives is greater than one). The example assumes the home position places the knife roll midway between knife blades (e.g. placing the knife at top dead centre if there is only a single knife blade configured).

Once homed the master encoder can now be started (ensure it runs in a positive direction). The knife will now ramp up to the cut ratio and reach the cut point over one material cut length. From this point onwards it now starts performing the calculated cam profile. For short lengths the knife axis will speed up between cuts, for long lengths it will slow down (eventually stopping completely between knives for very long lengths). Cut Length, Cut Ratio and Sync Angle can all be adjusted whilst the axis is running. To make a new 'Number of knives' take effect the system must be rehomed.

The screenshot below shows a Workbench capture of demand velocity and master encoder (encoder channel 2) velocity with the system running (with the default settings)…



It can be seen that the knife velocity matches the master velocity for a period equal to the specified knife sync angle. With the default settings (i.e. a cut length of 200mm) the knife has to speed up in between cuts. The screenshot below now shows what happens if the cut length is increased to 800mm…



With the longer cut length the knife axis has to slow to a stop, dwell for a short time and then reaccelerate up to the synchronous material speed for the start of the cutting region.

To be sure your Workbench traces match the scaling of the PLC application be sure to enter the necessary scalefactors into the drive parameters.

e.g.
```
SCALEFACTOR(0) = (131072 * 10) / (100 * _pi)
ENCODERSCALE(2) = 20000 / (50 * _pi)
```

If you run the master encoder slowly you will see how the application switches between the two MC_CAMIN function blocks as the master axis position reaches the programmed cut length (and the knife simultaneously reaches the cut point).

Contact us

For more information please contact your
local ABB representative or one of the following:

new.abb.com/motion
new.abb.com/drives
new.abb.com/drives/drivespartners
new.abb.com/PLC