Application Note

FESTO

FMCP on ETHERNET/IP

This document will go through a Step by Step description of what must be done to implement a FMCP system using Ethernet ip protocol.

FMCP

****

Title ..... FMCP on ETHERNET /IP
Version 1.2
Document no. .....********
Original ..... en
Author .....

Last saved 20.04.2022

Copyright Notice

Legal Notice

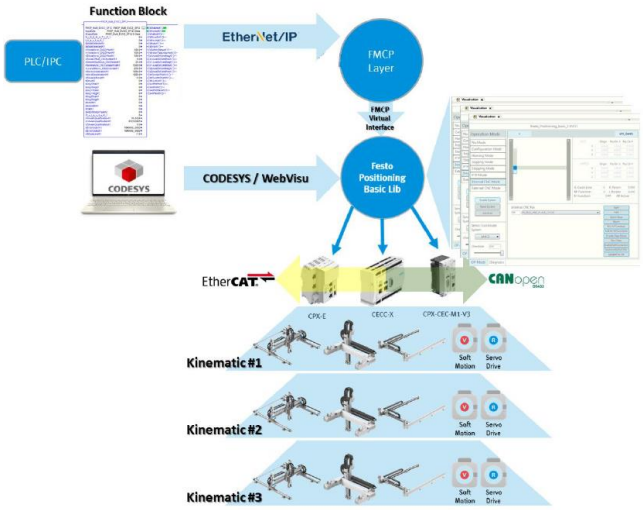
Internet: http://www.festo.com
E-Mail: *******

Contents

FMCP ON ETHERNET/IP

1 INTRO

This document is a reference to understand and implement an FMCP (Festo Motion Control Platform) system, using Ethernet IP protocol.



The FMCP works is based on Festo Positioning Library and useful to save time programming 1 to 3 systems or Kinematics, could be XY, T and H arrangements. The main Idea is to have Jogging, Stepping, PTP (Point to Point), Pick and place and CNC functionalities all in one function block.

The FMCP works with the control of a Master PLC trough a fieldbus in these case ethernet/IP (Could be Modbus, Profinet, OPC UA as an examples) controlling the Festo PLC. Spare I/O and valves can be added and used as well.

The client will receive a kit with a cabinet and axis as required.

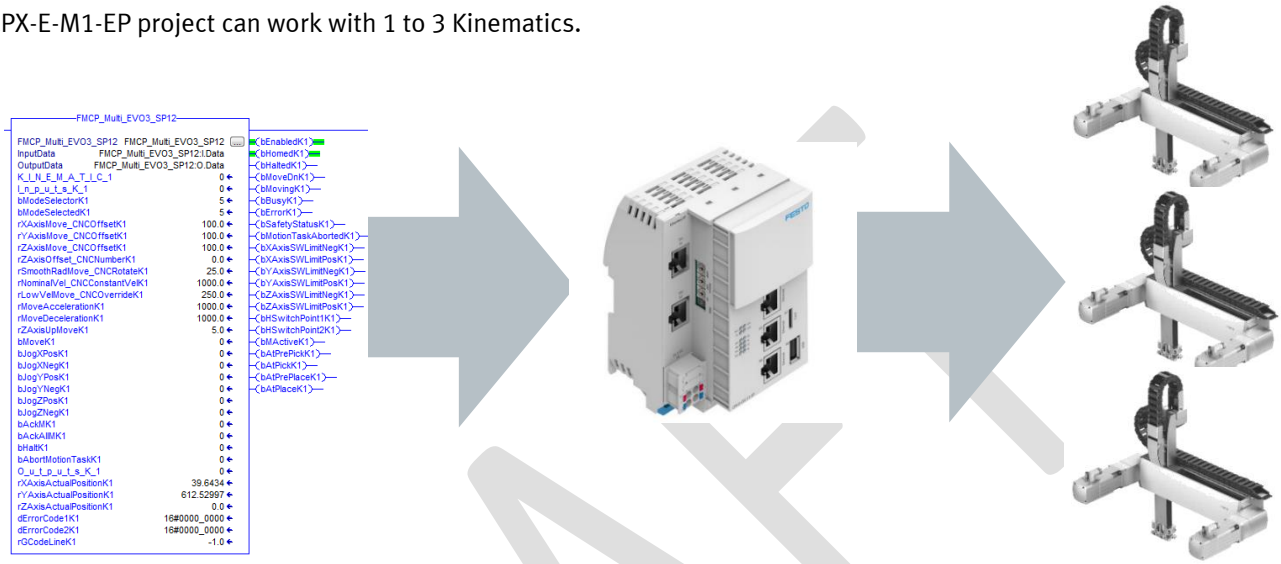
The main idea is a programming sequence will run on the Rockwell plc using an ADD ON, (Function block). And thus, trough the mapping will communicate with Festo plc.

The Festo PLC will run with a fit codesys project, and drives will need configuration. Some values need to be set on FAS or FCT and some on Codesys.

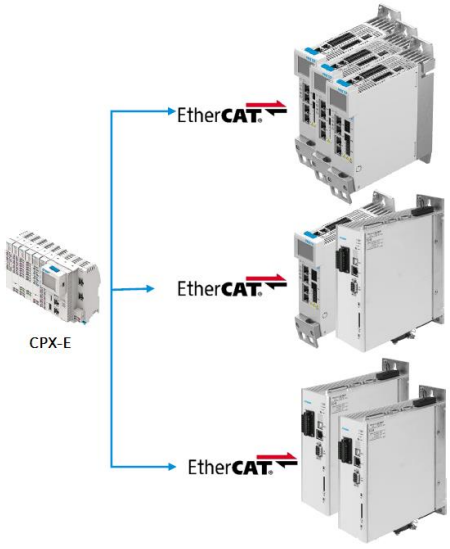
1.1 CPX-E-M1-EP PROJECT

Different kinematic can be managed, a different combinations can be made in the same kinematic (Virtual motors can be added).

CPX-E-M1-EP project can work with 1 to 3 Kinematics.

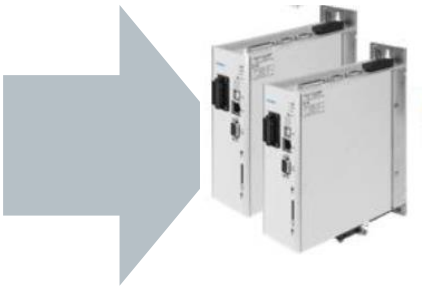


Possible combinations of CMMT or CMMP (M3).

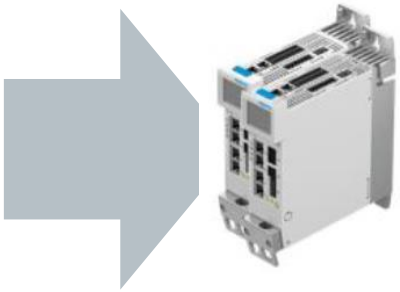


2 DRIVES CONFIGURATION

Depending the Festo drive, FAS or FCT will be required, different software, plug ins and firmware archives will be needed. A combination of communications is possible.



Festo Automation Suite
Program for configuration and programming of Festo Devices



-FESTO CONFIGURATION TOOL SOFTWARE + PLUG IN + FIRMWARE ARCHIVES FOR CMMP-M3

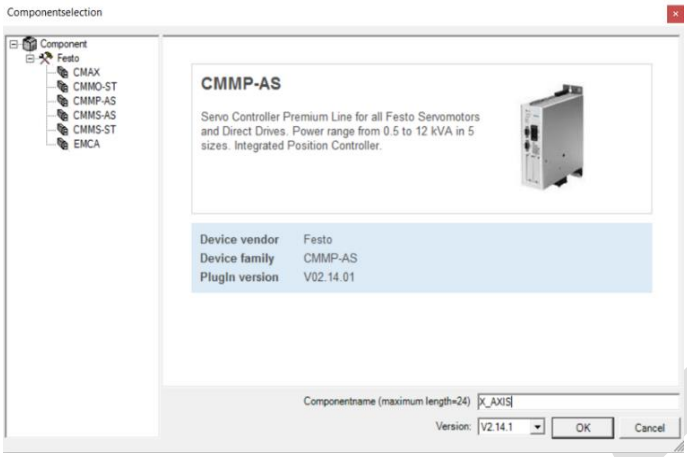
-FESTO AUTOMATION SUITE + PLUG IN + FIRMWARE ARCHVIES FOR CMMT-AS AND CMMT-ST

Go to Festo support and downloads in order to get the required software:

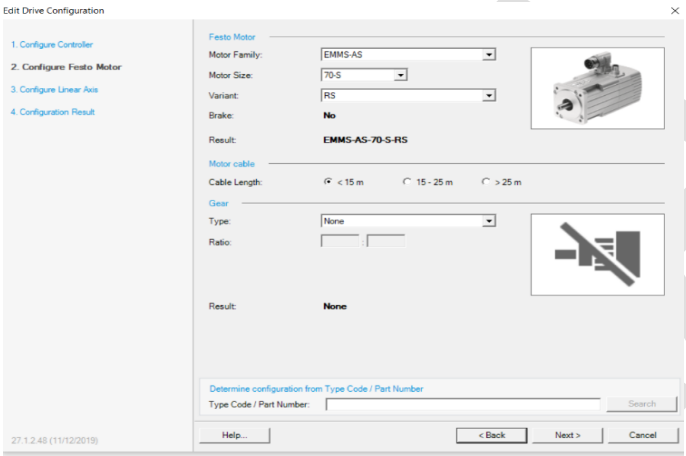
https://www.festo.com/net/es-ve_ve/SupportPortal/default.aspx

Here we will describe the bare minimum configuration on the software as an example, this configuration could vary depending on the parts and the application characteristics. Get the number parts or models of the drives, motors, and axis that will be operating.

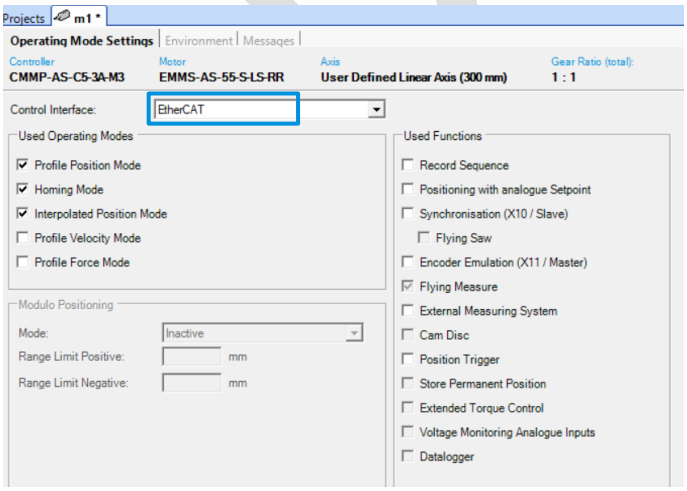
2.1 FCT DRIVE CONFIGURATION FOR CMMP-M3



Create a new project, later choose de plug in you installed, after that assign a component name, later click on create a new drive configuration.



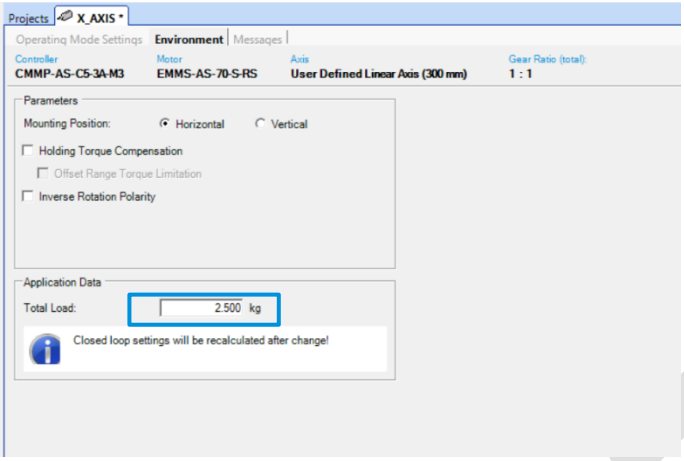
Introduce the model of the drive with its complements (security cam, etherecat board as examples.) model as well, later the motor model, gear if apply, and the length of the cable and finally the Festo axis. Click finish.



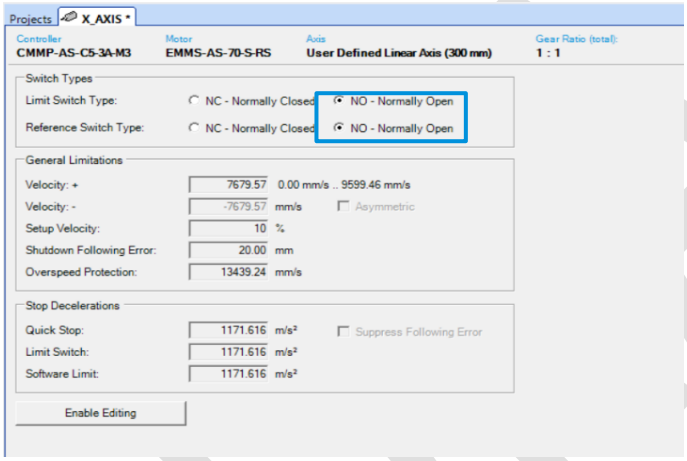
Go to Application mode data, Operation mode settings, and select Ethercat on control interface.

.

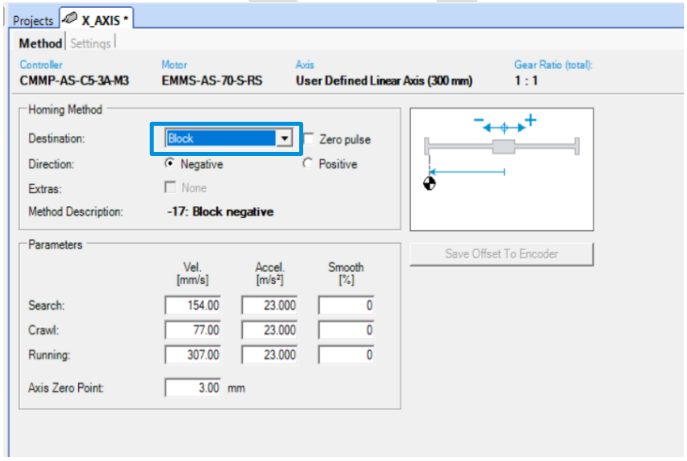
Remember the CMMP-AS..M3 will be use an CAMC-EC board



Click on Next, and on environment type the Calculated load and click enter (The software will calculate the closed loop settings with these value).

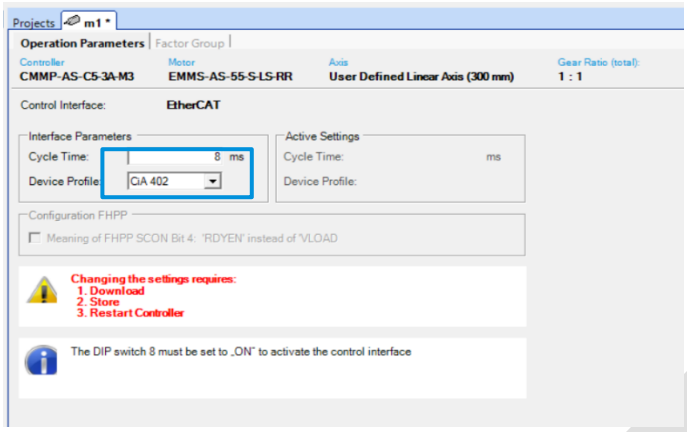


If you are not going to use limit or reference switch select NO.



Later go to Homing, Method, in the example we used Block (-17 Block Negative description). This method could be modified through the visualization as well. In this window you can modify the values of the speed, acceleration and smooth of the homing movement.

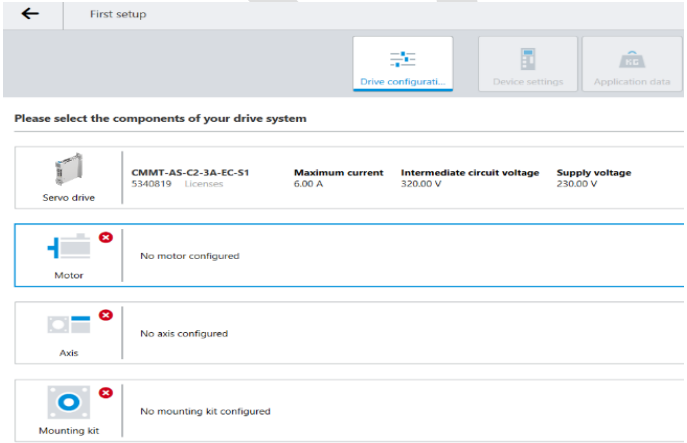
Finally go to Fieldbus, Operation parameters, and assign the address of the axis to be controlled. In cycle time type 8 ms (this time must be synchronized with the controller device.) And device profile CIA402.



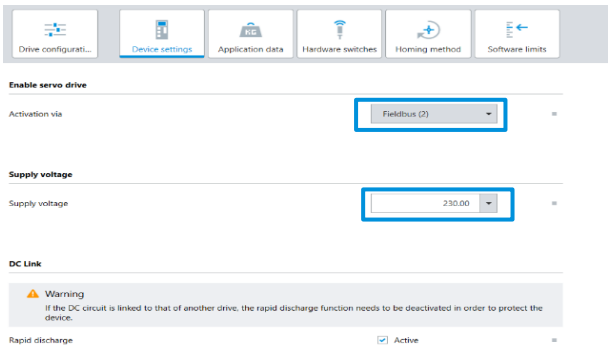
Go online with the controller and click on download and store. It is recommended enable and make some test movements (starting at low speeds) to check the connections and closed loop are ok. Same procedure must be done to all the axis available on your project.

2.2 FAS DRIVE CONFIGURATION

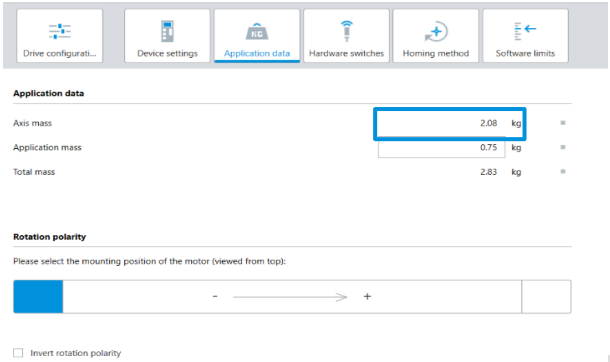
Open FAS (Festo Automation Suite) and click on new project
Go to device Repository and type the number part of the drive you will use. Then click on add device.
A drive will appear depending on the application you can add the drives needed on this step.



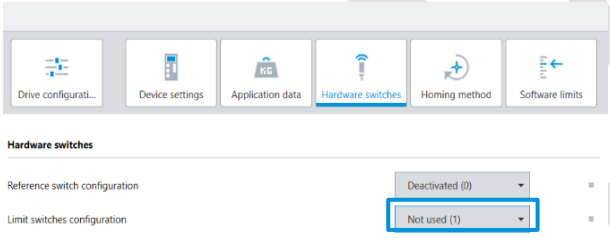
Later click on start first set up, and type the motor, axis, mounting kit and gearbox if applied. Later click on next.



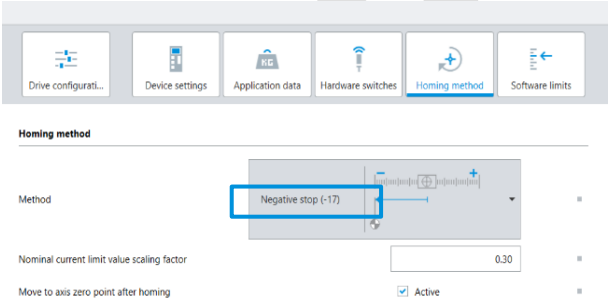
After on device settings choose the “Fieldbus option” and the supply Voltage.



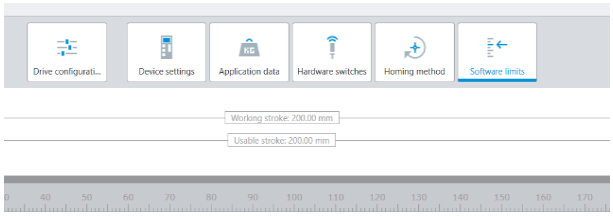
Later, on application data the application mass. And click on next.

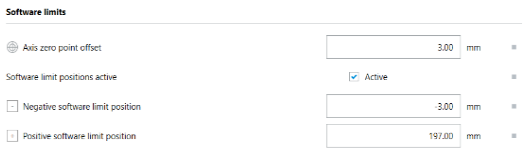


After choose if NO or NC reference switches will be used. If not click next.

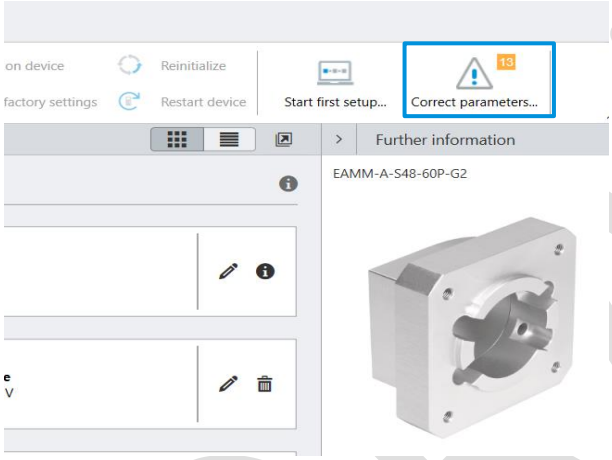


Then choose the homing method and direction of the axis will move. The current limits could be changed in the case of negative stop method.

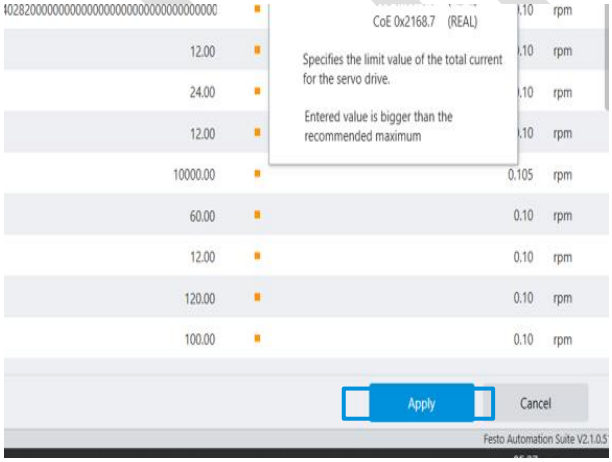




Software limits can be modified. Click on finish. After the software will calculate the closed loop parameters.



On this step the software will probably show some warnings



Select the Correct Parameters and click Apply button to review the warnings. Normally the user simply accepts all recommendations and applies to the application.

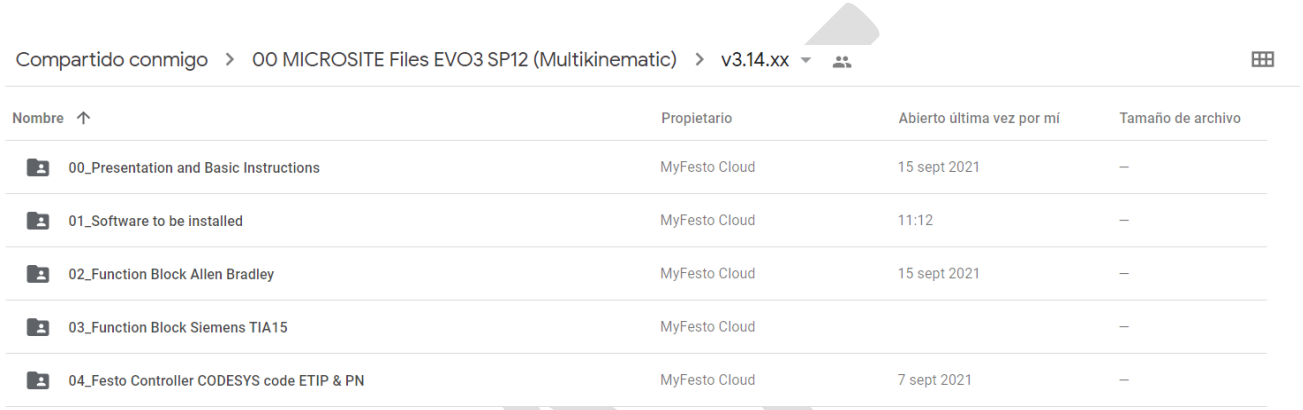
You must do the same with all the axis available on the system, the configuration could vary depending on the elements and payloads.

3 CPX-E PROJECT ARCHIVE FILE

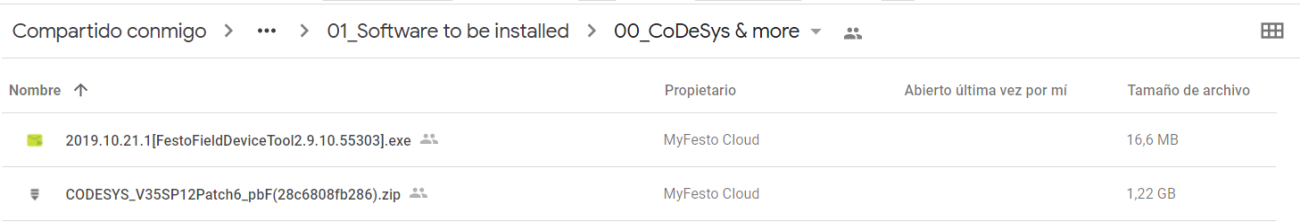
Go to https://sites.google.com/view/fmcp/software-and-more in the section Software and more download you must choose the right procjectarchive to the controller you are going to use.

FMCP EVO3 SP12 Multikinematic file

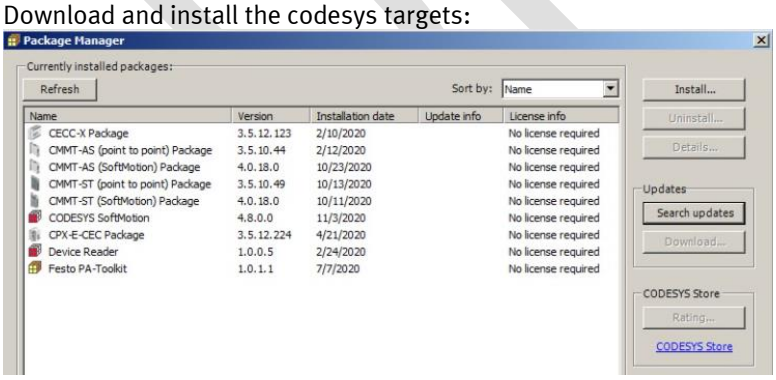
Click on FMCP EVO3 SP12 MultyKinematic files then a Google drive will open. Later click on V3.14.xx in this step If you don't have the Codesys and package for CPX-E installed, proceed to open the folder 01_Software to be installed.



Download and install the codesys

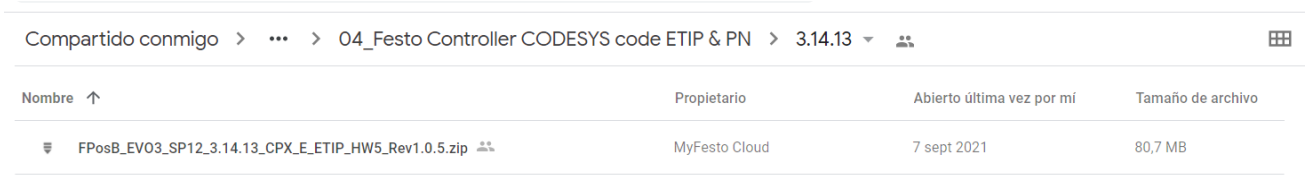


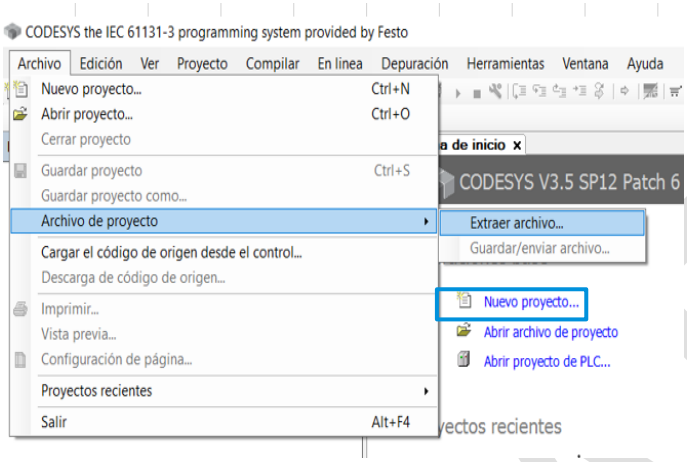
Download and install the codesys targets:



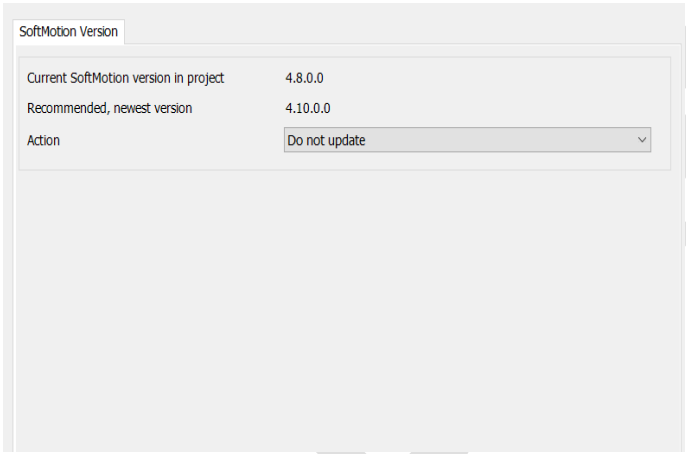
Required targets and their version (You will find them on 01Softare to be installed folder).

Download the FPosB_EVO3_SP12.3.14.13_CPX_E_ETIP_HW5_Rev1.0.5.zip archive, this is the basic codesys project to start to work with:



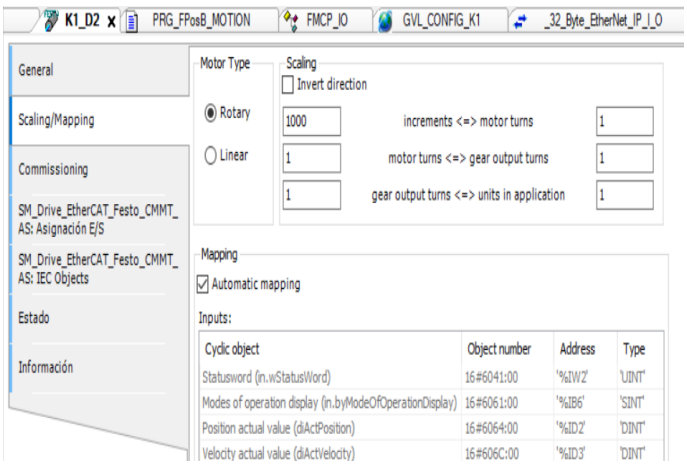


Open CODESYS, click on Project Archive, Extract, then choose .zip file you just download. (To extract the file, you must set “All files” option.)
Wait until the project complete extract. It must compile without errors.



On project environment window use only tested versions. (Refers to required targets and their version.)

ETHERCAT SLAVE SETTINGS

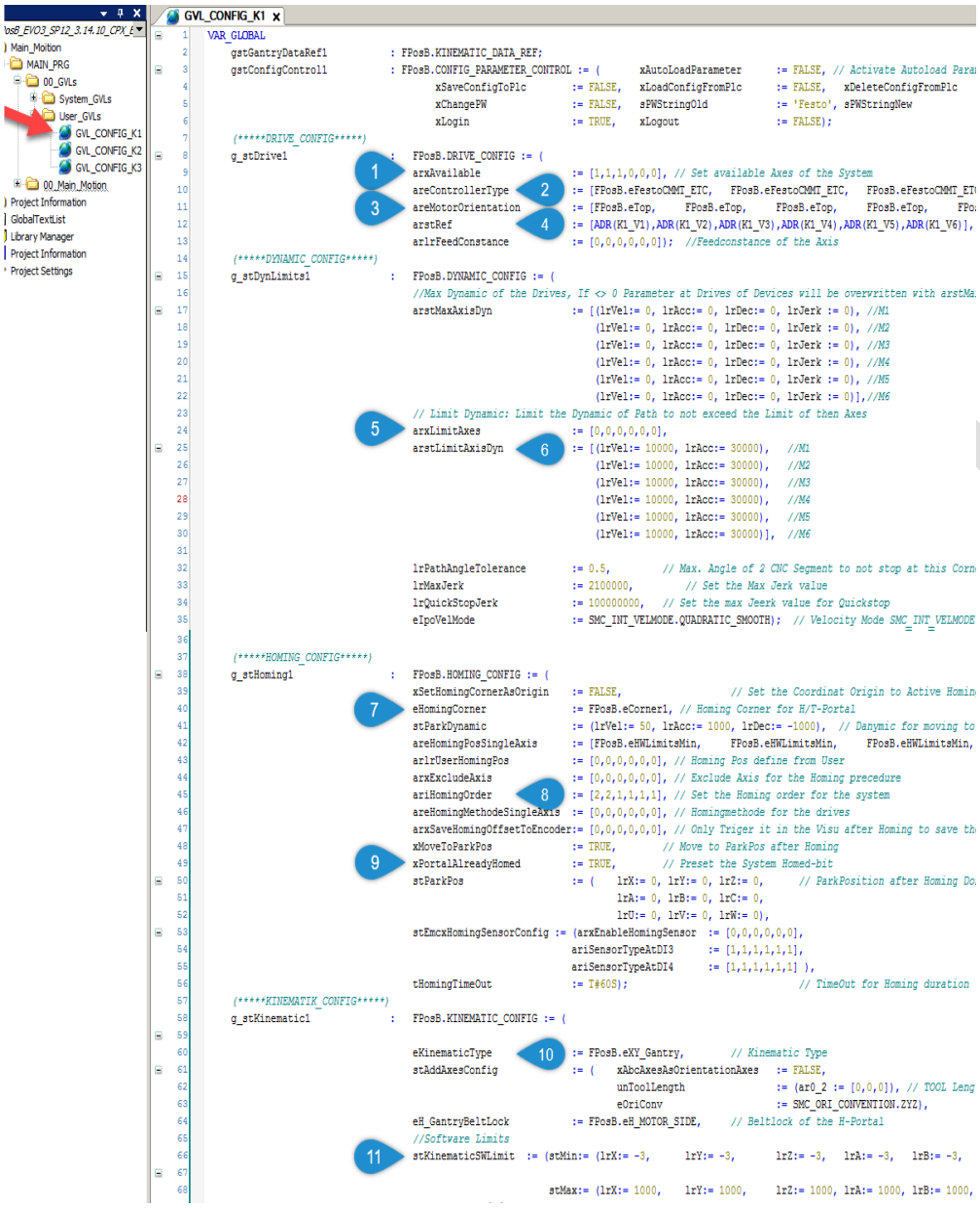


The Scaling mapping section on the cmmt ethercat slaves does not need any modifications. These only is required with **CMMP** with ethercat board

3.1 GLV_CONFIG SETTINGS

GVL_CONFIG (K1..K3) is a Global Variable List that contains system parameters which will be read by the motion controller to determine which system it is it will control. Most parameters here are specified in order X, Y, Z, A, B, C.

Full description of each parameter is described in the Application Notes for the Festo Positioning Basic Library downloadable from the Support Portal.



The first step in the FMCP + FPosLib implementations is to determine and setup the following tags:

1. Available drives per kinematic - [arxAvailable]: a. FMCP: 1, 2, 3.
   b. Virtual: 1, 2, 3, 4, 5 or 6.

2. Type of drive for real axis - [areControllerType]:
a. CMMP-CAN
b. CMMS-CAN
c. EMCA
d. CMMP-ETC
e. CMMT-ETC

3. Motor orientation when H or T type selected - [areMotorOrientation].

4. Address for real and virtual drive used - [arstRef]. (In this point.

5. (Optional) - Axis dynamic limitation - [arxLimitAxes].

6. (Optional) - Axis dynamic limitation - [arstLimitAxisDyn].

7. (Optional) - Homing corner when H or T type selected - [eHomingCorner].

8. (Optional) Homing order - [ariHomingOrder].

9. (Optional) - Portal already homed – [xPortalAlreadyHomed].

10. Kinematics type - [eKinematicType]: a. XY (2D, 3D)
b. H
c. T

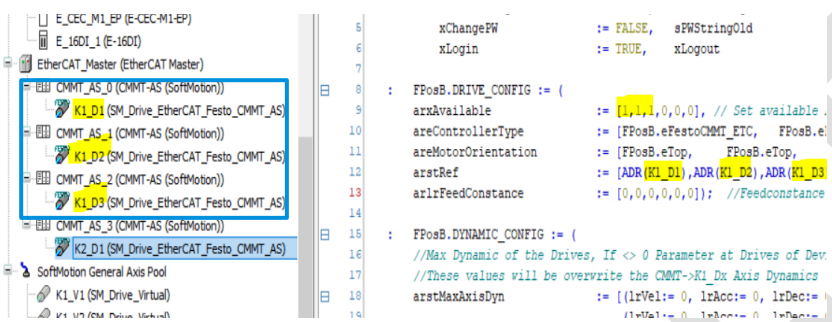
11. Software limits for each axis – [stKinematicSWLimit

3.2 ADDING DRIVES.

The FMCP is equipped with 3 preconfigured kinematics. Is possible to configure the entire project with only virtual drives or any possible combination with real and virtual devices. For that reason, it is possible adding, or changing drives on the project. After adding the drive follow these steps:

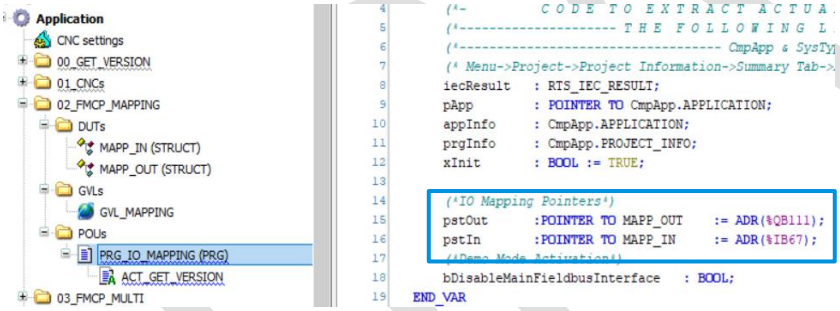
1. Change the name for the REAL drive based on your application, **this name is very important** as it will be used in the **GVL_CONFIG** to reference real drives.
2. Confirm the location for the virtual axis, names are already defined in the SoftMotion General Axis Pool.

This combination can be done in any way, all drives real, all drives virtual or combined.

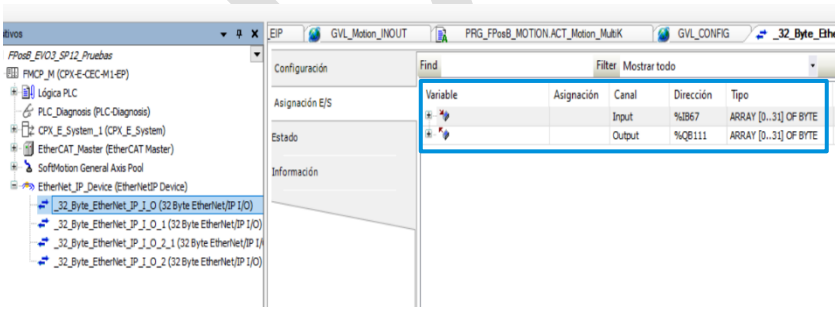


On this example One real drive was add to the project for a second Kinematic usage. To add a drive right click on Ehtercat Master and “Add Device” and choose the cmmt with the tested version. If that would be the case the name would be assign on “arstRef” of GVL_CONFIG_K2.

The “K1_D1” Refers to Kinematic 1, Drive1, “K1_V1” Refers to Kinematic 1 virtual 1.



The addresses of PRG_IO_MAPPING must match with the Ethernet IP I/O bytes. If Drives or modules of the CPX-change these addresses will be changing as well.



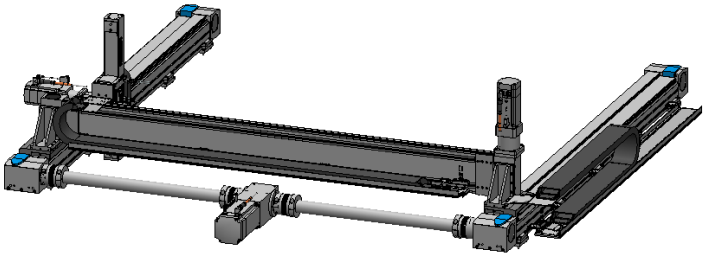
Starting addresses of ethernet ip.

A similar situation is when a CPX-E module is added, these addresses would change as well.

3.3 eXY, eT AND eH KINEMATICS SETTINGS

eYZ SETTINGS:

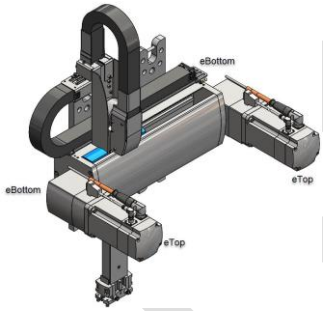
```
FPosB.DRIVE_CONFIG := (
arxAvailable                := [1,1,1,0,0,0], // Set available Axes of the System
areControllerType           := [FPosB.eFestoCMMT_ETC,   FPosB.eFestoCMMT_ETC,   FPosB.eFestoCMMT_ETC,   FPos
areMotorOrientation         := [FPosB.eTop,     FPosB.eTop,     FPosB.eTop,     FPosB.eTop,     FPosB.eTop,
arstRef                     := [ADR(K1_D1),ADR(K1_D2),ADR(K1_D3),ADR(K1_V4),ADR(K1_V5),ADR(K1_V6)],     // A
arlrFeedConstance           := [0,0,0,0,0,0]);  //Feedconstance of the Axis
```



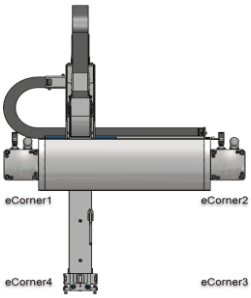
eXY Gantry

eT SETTINGS:

```
FPosB.DRIVE_CONFIG := (
arxAvailable                := [0,1,1,0,0,0], // Set available Axes of the System
areControllerType           := [FPosB.eFestoCMMT_ETC,   FPosB.eFestoCMMT_ETC,   FPosB.eFestoCMMT_ETC,   FPos
areMotorOrientation         := [FPosB.eTop,     FPosB.eTop,     FPosB.eTop,     FPosB.eTop,     FPosB.eTop,
arstRef                     := [ADR(K1_V1),ADR(K1_D1),ADR(K1_D2),ADR(K1_V4),ADR(K1_V5),ADR(K1_V6)],     // A
arlrFeedConstance           := [0,0,0,0,0,0]);  //Feedconstance of the Axis
```



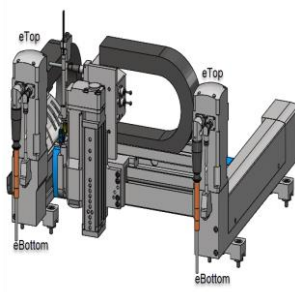
Orientiation for eT Gantry



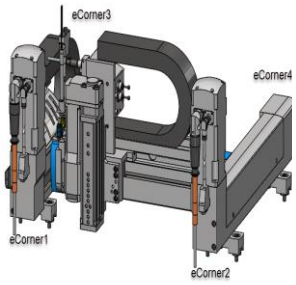
Homing corner for T gantry

eH SETTINGS:

```
FPosB.DRIVE_CONFIG := (
arxAvailable                := [1,1,1,0,0,0], // Set available Axes of the System
areControllerType           := [FPosB.eFestoCMMT_ETC,   FPosB.eFestoCMMT_ETC,   FPosB.eFestoCMMT_ETC,   FPos
areMotorOrientation         := [FPosB.eTop,     FPosB.eTop,     FPosB.eTop,     FPosB.eTop,     FPosB.eTop,
arstRef                     := [ADR(K1_D1),ADR(K1_D2),ADR(K1_D3),ADR(K1_V4),ADR(K1_V5),ADR(K1_V6)],     // A
arlrFeedConstance           := [0,0,0,0,0,0]);  //Feedconstance of the Axis
```



Orientiation for H gnatry

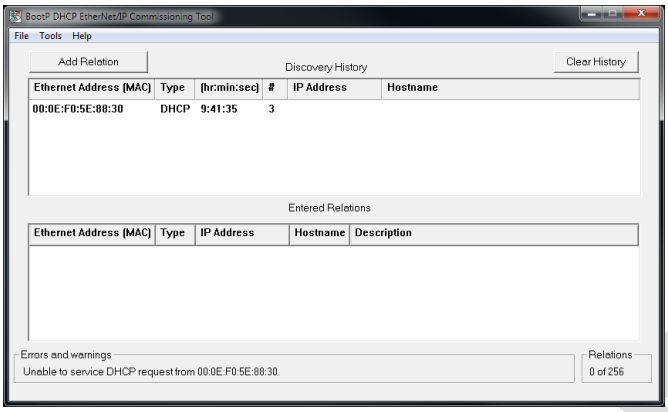


Homing corners for H gantry

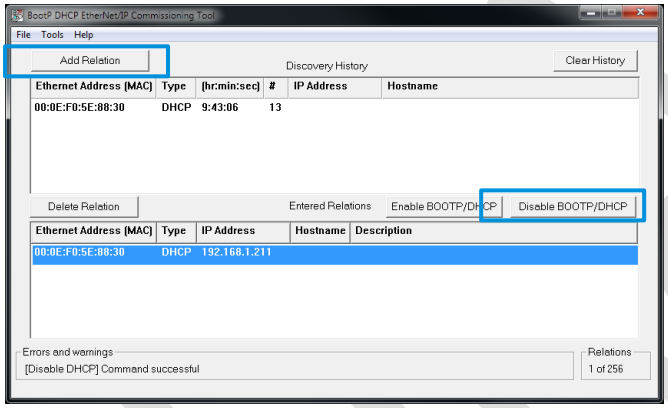
3.4 ASSIGNING AN IP XF1 PORT

The cpx-e has on configuration port (ETH1) this port can be used to firmware update, monitoring, and programing activities, at these point a user needs to configure the IP Address of the Port XF1 (EtherNet/IP) to have it connected to the Rockwell PLC

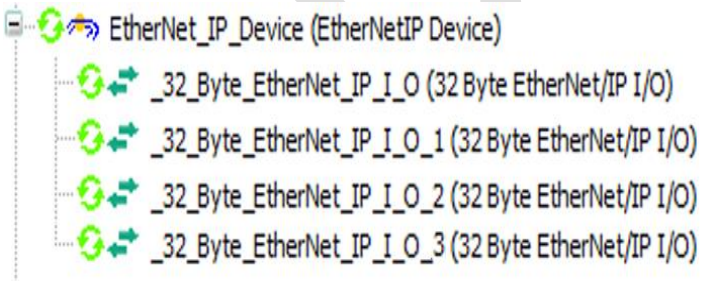
Using BOOT/IP



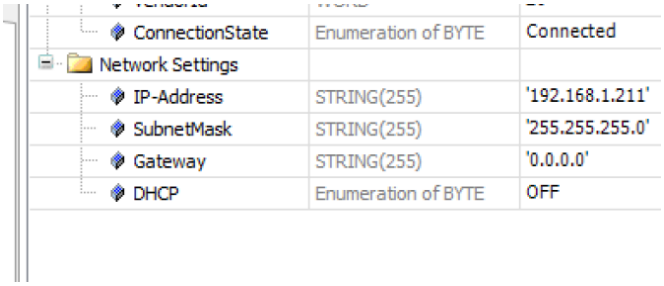
Start BootP software and wait for scan to complete



Add relation and Disable DHCP



Open the codesys project and double click on Ethernet_IP_Device,

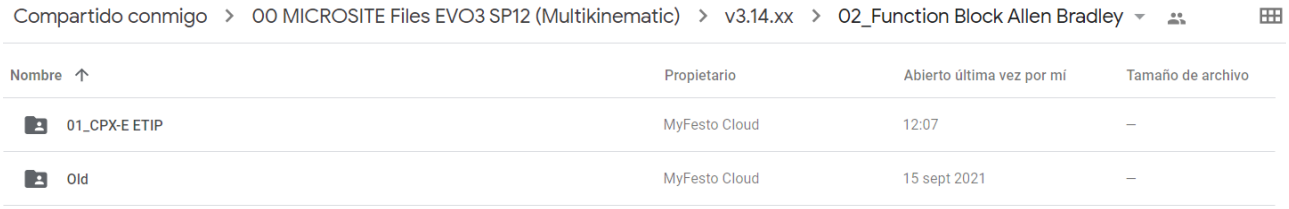


On the Parameters tab, Expand the "Network Settings “parameter heading.
Configure the IP Address/Subnet/gateway and
Turn OFF the DHCP

4 DOWNLOAD ROCKWELL ARCHIVES.

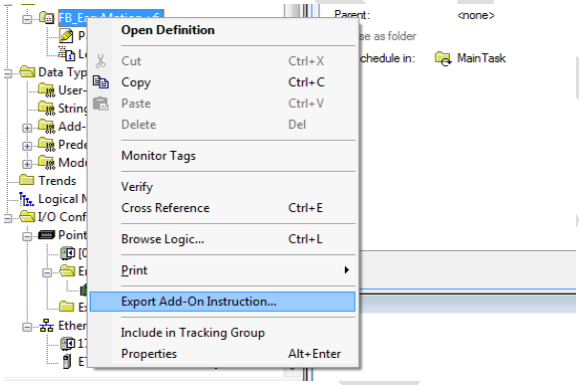
CPX-E MULTI KINEMATIC EXAMPLE

Go to the microsite, and click on FMCP MULTICINEMATIC and go to the 02_Funtion block Allen Bradley folder and choose 01_CPX-E ETIP:

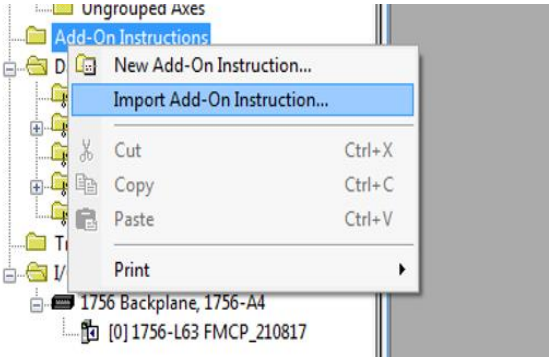


Later click on FPosB_EVO3_SP12_3_14_10_CPX_E_ETIP.ACD file and download.

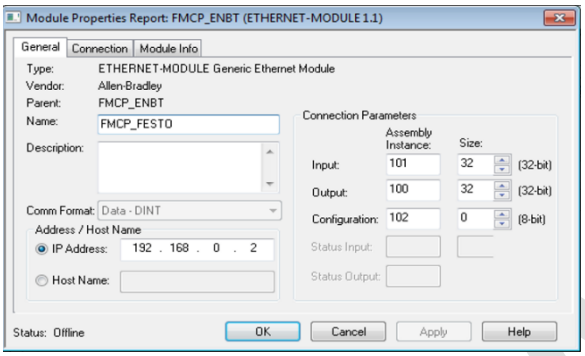
This .ACD program contains the function block and a generic ip/module for reference. The FMCP_Multi_EVO3_CPX_E_SP12_CPX_E add on can be exported and later imported on your program. Open de FPosB_EVO3_SP12_3_14_10_CPX_E_ETIP.ACD on the main routine.



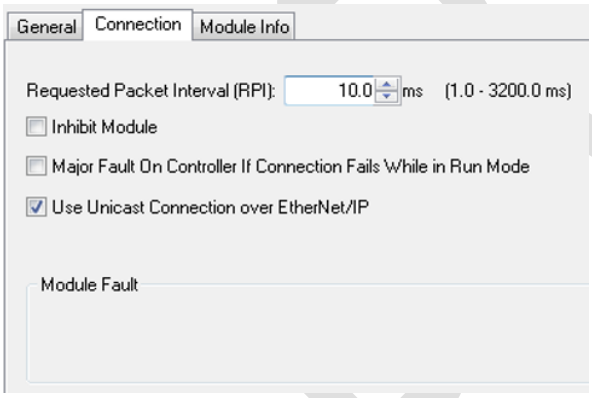
Go to the add on Folders and right click. Later Export Add-On Instruction. Choose a location for the function block (L5X file).



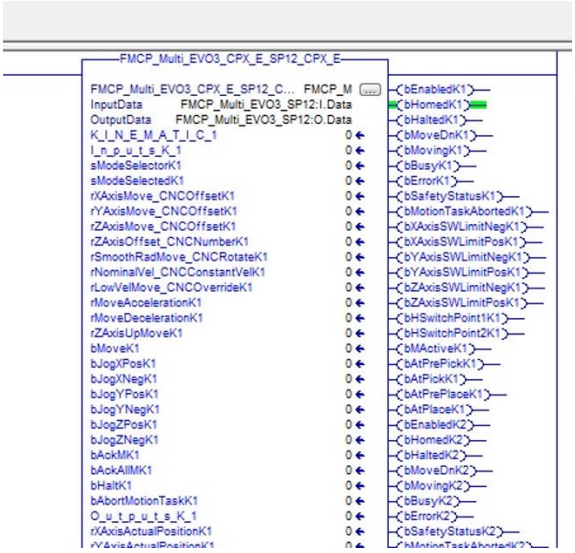
Create or open the project where you will add the Add on instruction. Go to Add on instruction folders and right click and Import the file you just export. Now you can add the block in to the routine.



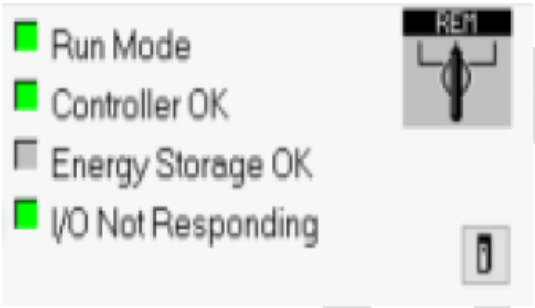
Add a Generic ethernet module. Assign a name, IP (Same of the ethernet ip adapter on codesys), for Inputs Assembly Instance 101, Size 32. On Outputs Assembly instance 100, Size 32. On Configuration Assembly Instance 102, Size 0., data type DINT



Default Connection Properties can be left as default, adjustments might be needed depending on the network (RPI, Unicast).



Input Data: Variable that should be linked to the input data of the Ethernet ip Module.
Output Data: Variable that should be linked to the output data of the Ethernet ip Module.
Include the Add on on your routine, create the tag for the function block and reference to the generic ethernet module I/O´s. Compile and download the project. If the communication is ok now the system is ready to move.jj

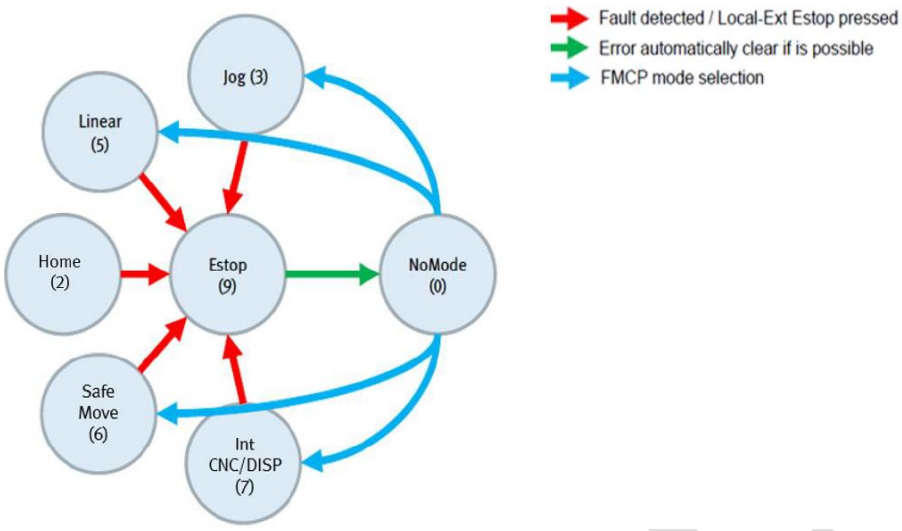


if you have a flashing green LED on I/O Not Responding on the Studio 5000 you can verify some parameters (Generic ethernet instances, size or IP´s as examples).

Compile and download to Rockwell plc. You can verify the communication on codesys as well.

5 FLOW CHART AND OPERATION MODES.

The FMCP available functions or states of operation are described as follows:



0.-No mode: On this state the system will clear faults (If exists) And will enter on config mode if has not been performed and later will return to no mode.

9.-Estop: These operation mode disable any movement, restart any mode and clean errors.

2.-Home: homing routine of the system.

3.-Jog: The axis can be jogged (one at the time) on this mode.

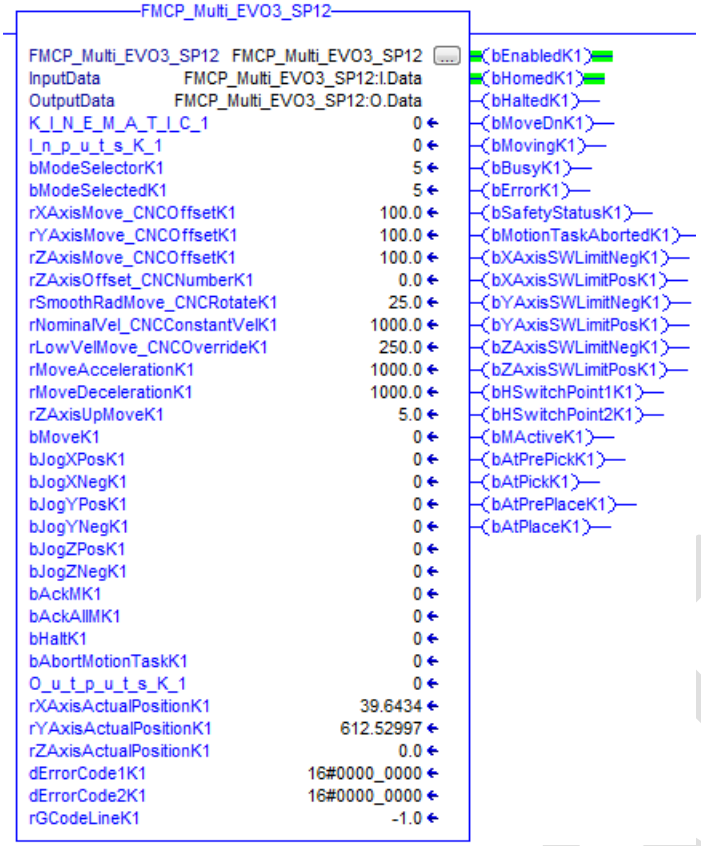
5.-Linear: Point to point movements can be done on this mode.

6.-Safe Move: Allows interpolated pick and place movements (inverted U), saving height in every motion to protect oat (End of arm tooling).

7.-Dedicated function for dispensing at constant speed.

- H function to trigger external signals
- M function to pause during the path.

5.1 ADD ON DESCRIPTION



FB_FMCP_Multi_EVO3_CPX_SP12_CPX_E: Instance of the AOI, any name can be used here.

InputData: Variable which should be linked to the Input Data of the Ethernet/IP Module created in the project.

OutputData: Variable which should be linked to the Output Data of the Ethernet/IP Module created in the project.

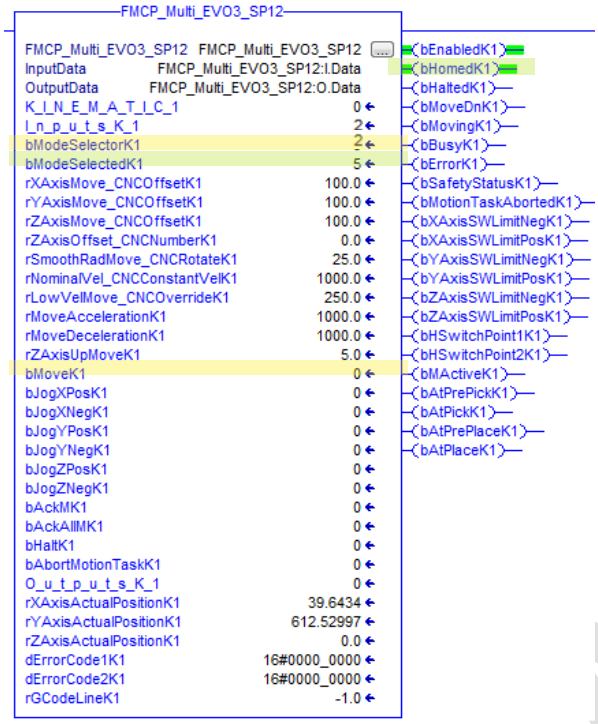
INPUTS

sModeSelectorK1..K3: User can select the mode to work with. See AOI Operation Modes section.
sModeSelected K1..K3: Actual Operation Mode that is running.
RAxisMove _CNCOffsetK1...K3(For X,Y,Z Axis): If working PTP or Safe Move Axis target position, if working with CNC Mode CNC offset for the 0 absolute position.
rZaxisOffset_CNCNumberK1..K3: Offset Z axis on safe mode / CNC number program (Loaded on codesys).
rSmoothRadMove_CNCRotateK1..K3(For X,Y,Z Axis):. Radius, which is used to smoothen the path between Z- and X/Y- movement (for all positioning). The radius is limited by the shortest distance of a part of the whole movement. / Rotation of the CNC path mode 7.
rNominalVel_CNCConstantVelK1..K3: Target speed mm/sec /Constant velocity CNC path mode 7.
rMoveAccelerationK1..K3:Target acceleration mm/sec2.
rMoveDeccelerationK1..K3: Target deceleration mm/sec2.
rZaxisUpMoveK1..K3: Up-position of the z-axis, while moving between positions
bMoveK1..K3: Trigger signal for Point to Point, Home, Safe move, and CNC.
bJogPosX..Z(K1..K3): Axis jogging positive direction signal.
bJogNegX..Z(K1..K3): Axis jogging negative direction signal.
bAckMK1..K3: M signal acknowledgment on CNC routines.
bAckAlMK1..K3: M signal M acknowledgment, for continues pick and place movements.
bHaltK1..K3: Halt current motion task.
bAbortMotionTaskK1..K3: Abort current motion task
rXActualPositionX..Z(K1..K3): Actual position of axis.

OUTPUTS
dErrorCode1K1..K3: Error code number 1.
dErrorCode2K1...K3: Error code number 2.
rGCodeLineKi1..K3: Current line of g code running.
bEnabledK1..K3: Kinematic is enabled.
bHomedK1..K3: Kinematic Already homed.
bHalted K1..K3: Kinematic stopped.
bModeDnK1..K3: Motion task done.
bMovingK1..K3: Kinematic is moving.
bBusyK1..K3: Kinematic is Busy.
bErrorK1..K3: Failure present.
bMotionTaskAbortedK1..K3: Motion task aborted.
bX(..Z)AxisSwLimitNegK1..K3: Axis reach its configured negative limits.
bX(..Z)AxisSwLimitPosK1..K3: Axis reach its configured positive limits.
bHSwitchPoint1K1..K3: Switch point 1 H function.
bHSwitchPoint2K1..K3: Switch point 2 H function.
bMactiveK1..K3: M function signal active.
bAtPrePickK1..K3: Pre pick signal position safe mode.
bAtPickK1..K3: Pick signal position safe mode.
bAtPrePlaceK1..K3: Pre place signal position safe mode.
bArPlaceK1..K3: Place signal position safe mode.

5.2 SEQUENCE EXAMPLES FB_FMCP_Multi_EVO3_CPX_SP12_CPX_E

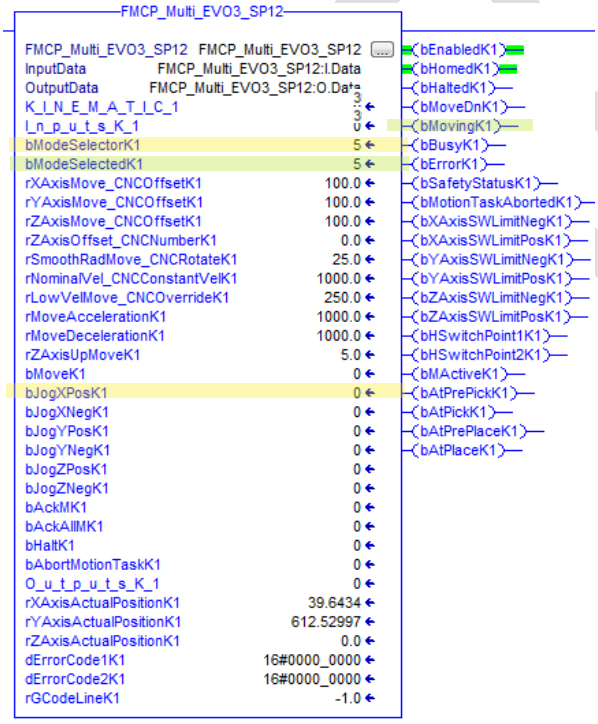
HOMMING MODE(2):



On these example the program change from STOP to HOMMING operation mode, then complete home:

ACTION	TRANSITION
bModeSelector := 9 EStop	bModeSelected == 9
bModeSelector :=0 (Error verification)	bModeSelected == 0
bModeSelector := 2 (Homming)	bModeSelected == 2
bMove := 1	bHomed == 0
bMove := 0	bHomed == 1

JOGGIN MODE(3):

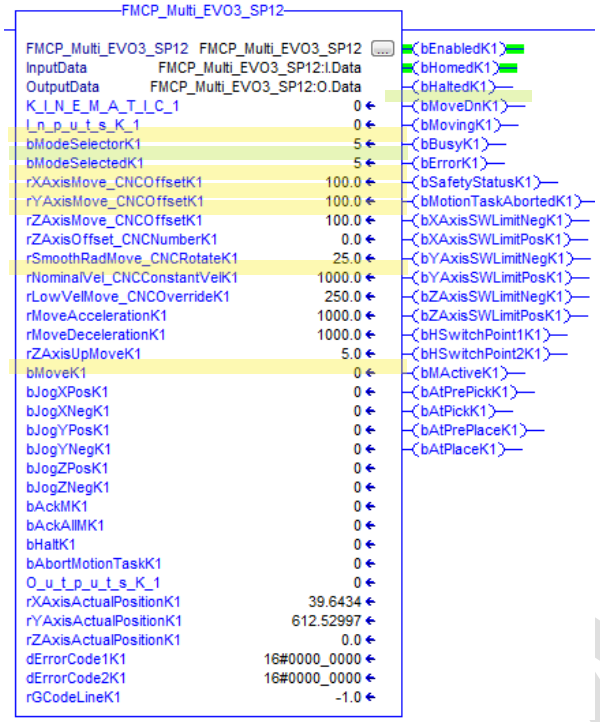


On this example the program is on JOG operation mode and jog X axis (Only can jog one axis at the time).

ACTION	TRANSITION
bModeSelector := 3 (Jogging)	bModeSelected== 3
rLowVelMove_CNCOverrideK1:= Target speed mm/sec bJogXpos :=1	bMovingK1 == 1
bJogXpos :=0	bMovingK1 == 0

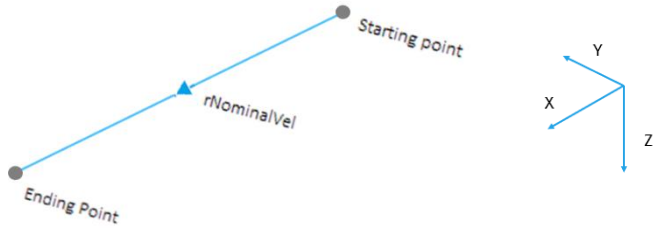
POINT TO POINT MODE (5):

Point to point sequence: On this example the program is on PTP operation mode and complete one point to point movement.



ACTION	TRANSITION
bModeSelector := 5 (Point to Point)	bModeSelected == 5
rNominalVel_CNCConstantVel:=Target speed mm/s RMoveAcceleration/rMoveDeceleration:=Target Acceleration and Deceleration mm/sec^2 rXAxisMove_CNCOffset := Target Destination rYAxisMove_CNCOffset := Target Destination rZAxisMove_CNCOffset := Target Destination bMove :=1	bMoveDn == 1
bMove :=0	bMoveDn == 0

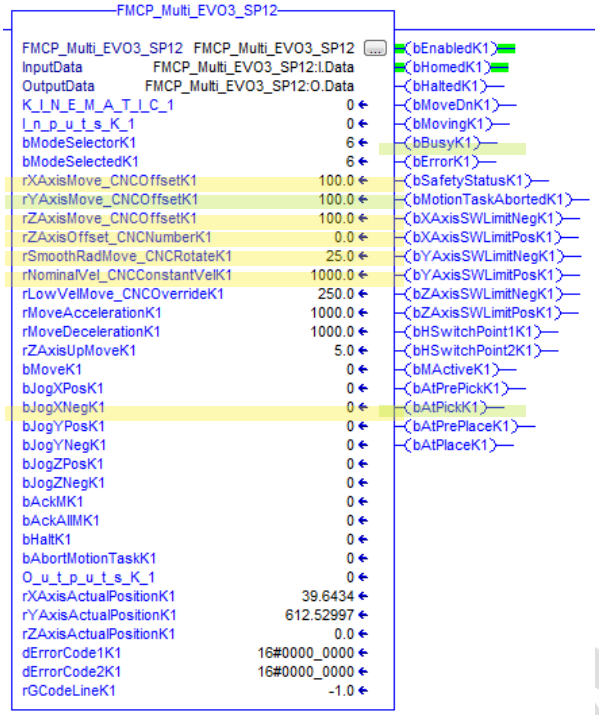
POINT TO POINT DIAGRAM:



End X Point= rXAxisMove
End Y Point = rYAxisMove
End Z Point= rZ rZAxisMove

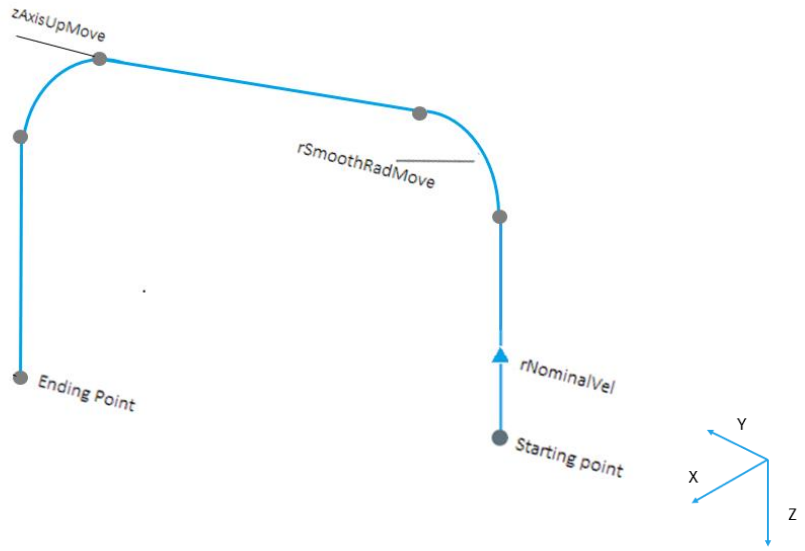
SAFE MOVE WITH NO Z OFFSET (6):

Safe move: On this example the program is on safe move operation mode and complete one interpolated movement.



ACTION	TRANSITION
bModeSelector := 6 (Safe Move)	bModeSelected == 6
rNominalVel_CNCConstantVel:=Target speed mm/s RMoveAcceleration/rMoveDeceleration:=Target Acceleration and Deceleration mm/sec^2 rXAxisMove_CNCOffset := Target Destination rYAxisMove_CNCOffset := Target Destination rZAxisMove_CNCOffset := Target Destination zAxisUpMoveK1:= Target Distance Z AxisUp rSmoothRadMove_CNCRotateK1:= Target smooth radius bMove :=1	bMoveDn == 1
bMove :=0	bMoveDn == 0

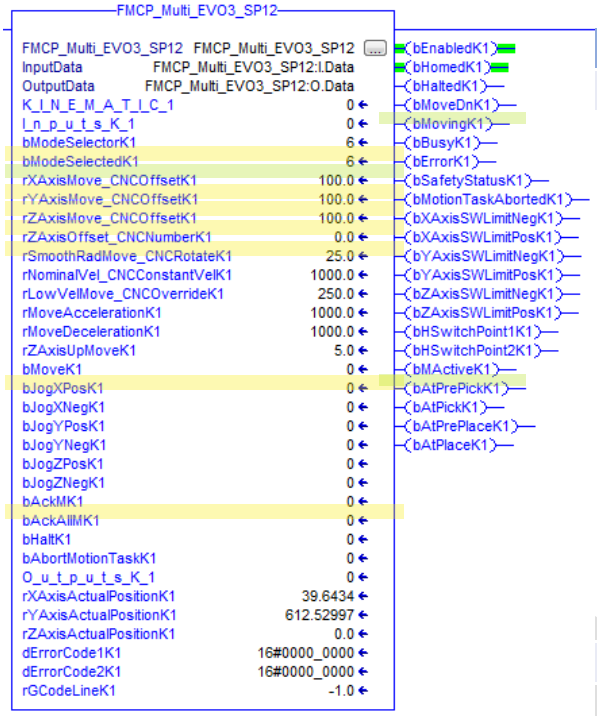
SAFE MOVE DIAGRAM:



End X Point= rXAxisMove
End Y Point = rYAxisMove
End Z Point= rZ rZAxisMove

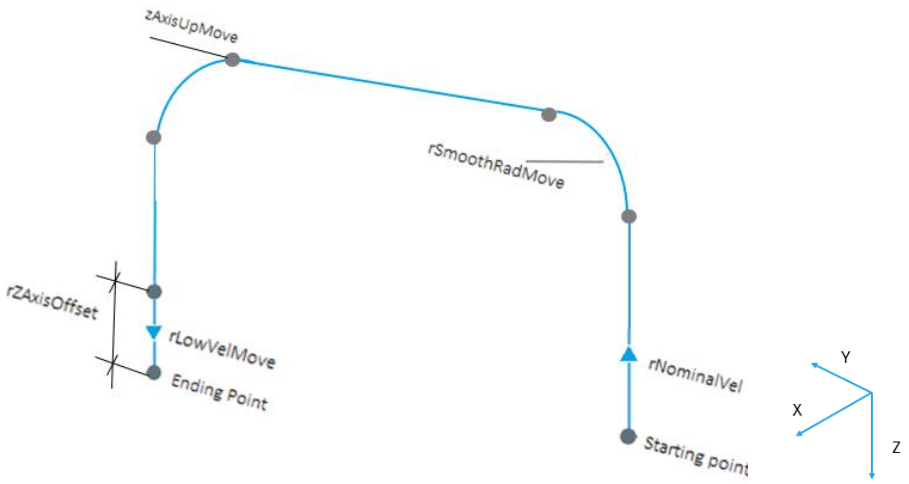
SAFE MOVE WITH Z OFFSET M ACTIVE (6):

Safe move: On this example the program is on safe move operation mode and complete one interpolated movement including low speed during offset Z movement.



ACTION	TRANSITION
bModeSelector := 6 (Safe Move) rNominalVel_CNCConstantVel:=Target speed mm/s rLowVelMove_CNCOverride:=target low speed RMoveAcceleration/rMoveDeceleration:=Target Acceleration and Deceleration mm/sec^2 rXAxisMove_CNCOffset := Target Destination rYAxisMove_CNCOffset := Target Destination rZAxisMove_CNCOffset := Target Destination rZAxisOffset_CNCNumber:=Offset low speed distance at Z. zAxisUpMoveK1:= Target Distance Z AxisUp rSmoothRadMove_CNCRotateK1:= Target smooth radius bMove :=1	bModeSelected == 6 bmActive == 1
bAckMK1 := 1	bMActive == 0
bAckMK1 := 0	bMoveDn == 1
bMove := 0	bMoveDn == 0

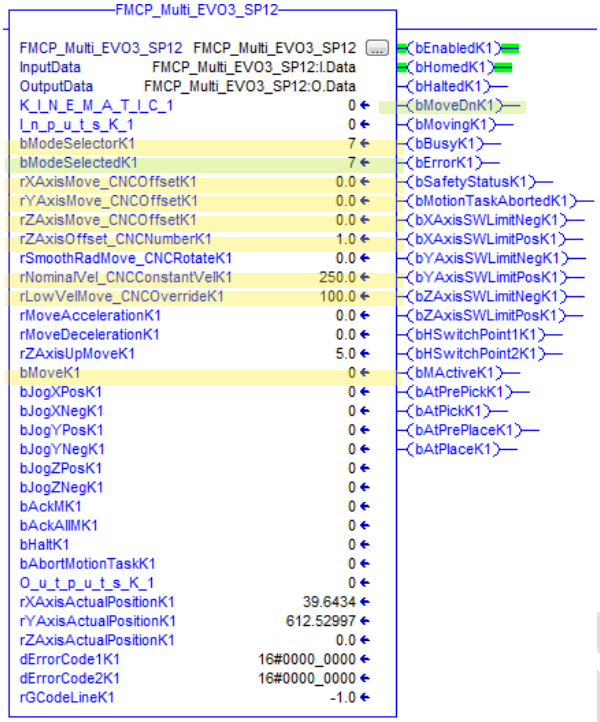
SAFE MOVE DIAGRAM:



EndXPoint=rXAxisMove
End Y Point = rYAxisMove
End Z Point= rZ rZAxisMove

CNC MODE (7):

Dedicated function for dispensing at constant speed, Maximum of 256 CNC programs.

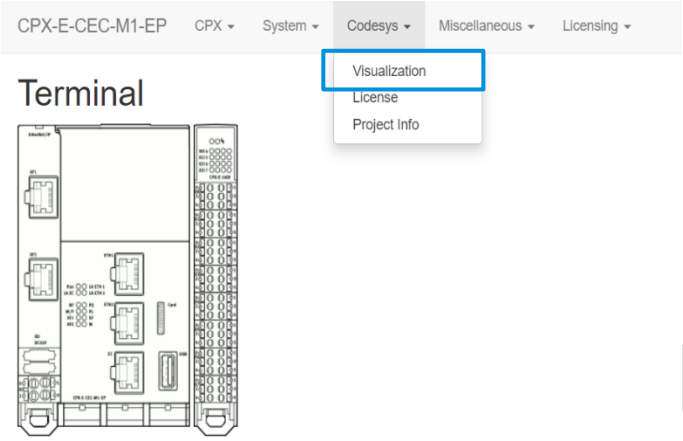


ACTION	TRANSITION
bModeSelector := 7 (CNC Mode, dispensing with CNC customer speed via rZAxisOffset_CNCNumber := CNC# (Previously loaded via CoDeSys CNC Editor)	bModeSelected == 7
rNominalVel_CNCConstantVel := Constant velocity in mm/sec (Updated at the begginging of the cicle) rLowVelMove_CNCOverride := Percentage of the nominal velocity (0 100%, live update during the cycle) rXAxisMove_CNCOffset := CNCOffset in mm from 0 absolute position* rYAxisMove_CNCOffset := CNCOffset in mm from 0 absolute position* rZAxisMove_CNCOffset := CNCOffset in mm from 0 absolute position* rSmoothRadMove_CNCRotate := Rotation angle in degrees for the current CNC* ETIP & PN these values are divided by 10 (+/ 0.1) bMove := 1	bMoveDn == 1
bMove := 0	bMoveDn == 0

H Function expected in CNC as:
•bHSwitchPoint1: H1/H-1 O0
•bHSwitchPoint2: H2/H-2 O0
This function will trigger the outputs CECC 14I/8O
•Bit0/Bit1 for K1, Bit2/Bit3 for K2, Bit4/Bit5 for K3

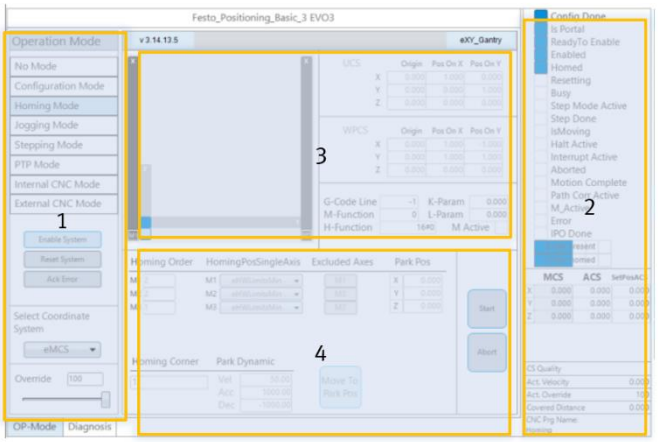
6 WEB VISUALIZATION

When the project has been downloaded to the festo controller, the web visualization can be used for configuring, monitoring, and troubleshooting the system. You can refer to the Application note; *Festo_Positioning_Basic library – system Visualization and Festo_Positioning_Basic library – Change, save, load settings,* for futther information.



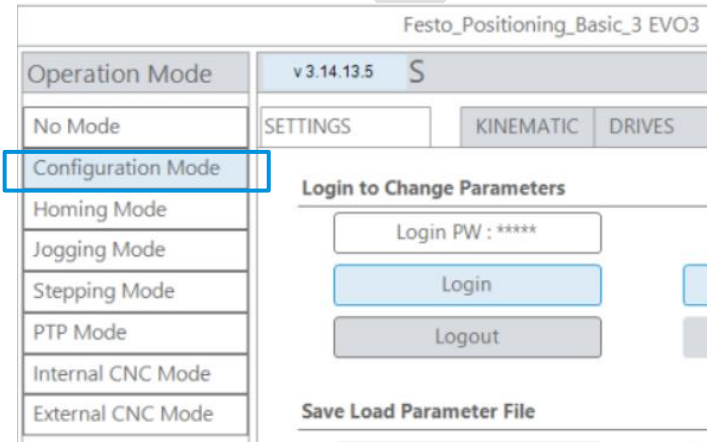
Type the IP on your web browser, later codesys and Visualization.

Main Visualization Areas:

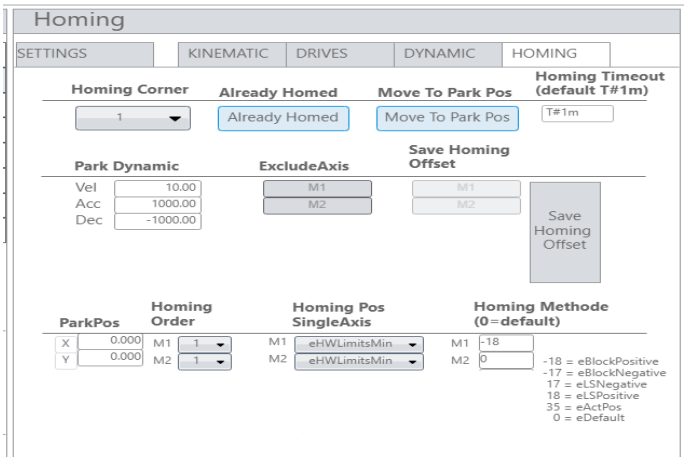


1.-Visualization input area. (clicking you can test the movements and main states of the equipment).
2.-Output range of the visualization. (Signals and reactions of the equipment on this section.).
3.-Kinematics can be visualized.(Movements and positions of the axis, g.code running etc).
4.-Information of the control elements, and control for testing the movements.

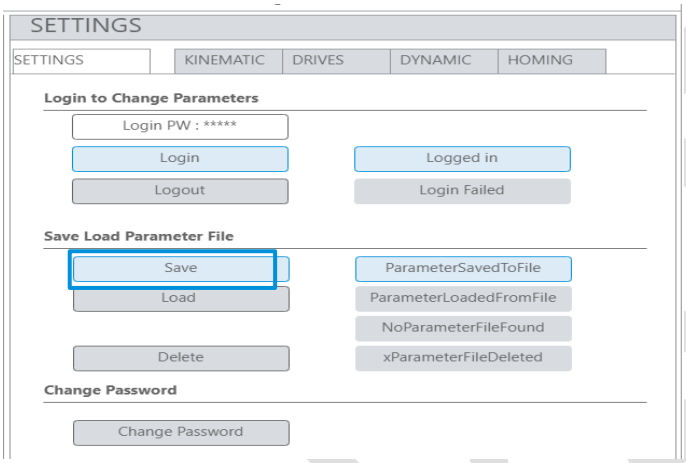
6.1 CHANGING PARAMETERS:



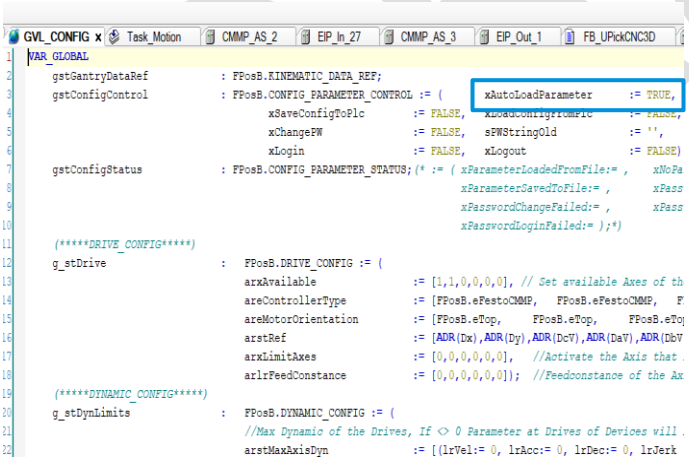
Click on the “Configuration mode”, password: Festo.



Go to the parameters you need to modify. When you´ve done, it can be tested (without power cycle).

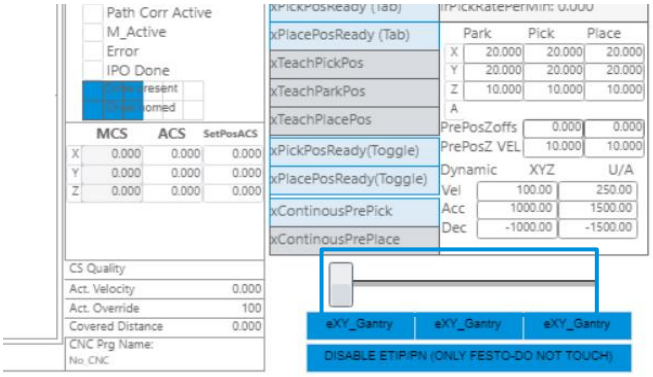


Go to Settings and click on “Save “in this way you would´ve saved the actual configuration on a file.



Go to GLV_CONFIG on codesys code. And set xAutoLoadParameter to TRUE. And download to the plc. In this way the configuration is going to be saved on the file and load after a power cycle.

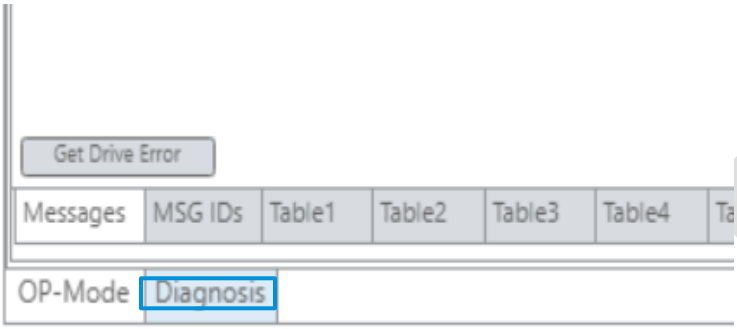
Kinematic selector:



A selector is available to visualize every kinematic and there is a button to unable the ethernet/ip signals for testing purposes.

7 TROUBLESHOOTING

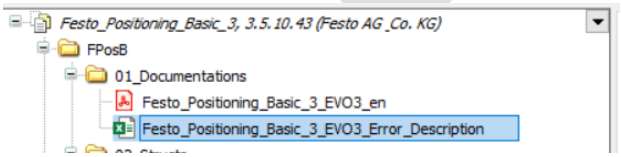
In order to get troubleshooting information from the system, you can reference to the document *Festo_Positioning_Basic library – Messagesystem / Errorhandling*. If an error appears during operation, a description can be get and a possible reaction to solve the problem can be found on these visualization section.



Select “Diagnosis”.

The message table has the following columns

Designation	**Significance**
ACK	If this value is active (1), the respective message must be acknowledged by the user.
MSG ID	Represents the error code which is explained in the error code description.
MSG table	Display with additional table including more detailed information (1-› SMC_ERROR; 2-› CANOPEN_KERNEL_ERROR; 3-› SDO_ERROR)
Add01	Error code for error table 1-3 in accordance with the MSG table
Dev	Number of the participant which has caused the error
Subsystem	A number can be assigned to the kinematics here, so that the master controller receives information indicating which subsystem has caused the error (advisable when several kinematic systems are connected to a single controller).
Reaction	Code in the error reaction table
Cat	Message category
Timestamp	Timestamp for the message



The error description file can be found on the documentation folder in the library section.

Example:

Ack	Hex View MSG ID	MSG Table	(Text/Hex) Add01	(Hex) Add02	Dev	Sub System	Reac-tion	Cat	Timestamp
	ERR_PositioningError	1	SMC_AXIS_NOT_READY_FOR_M	0	3	0	1000	3	DT#2000-05-03-04:54:35
	ERR_PositioningError	1	SMC_AXIS_NOT_READY_FOR_M	0	2	0	1000	3	DT#2000-05-03-04:54:35
	ERR_PositioningError	1	SMC_AXIS_NOT_READY_FOR_M	0	1	0	1000	3	DT#2000-05-03-04:54:35

If click "Hex View" errors will be shown on hex values (these msg id is the same shown on Rockwell ADD ON):

Ack	Hex View MSG ID	MSG Table	(Text/Hex) Add01	(Hex) Add02	Dev	Sub System	Reac-tion	Cat	Timestamp
	16#f0040060	1	16#22	0	3	0	1000	3	DT#2000-05-03-21:57:00
	16#f0040060	1	16#22	0	2	0	1000	3	DT#2000-05-03-21:57:00
	16#f0040060	1	16#22	0	1	0	1000	3	DT#2000-05-03-21:57:00

Opening the excel file on the on ERROR_ID sheet:

F004_0060	4026794080	ERR_PositioningError

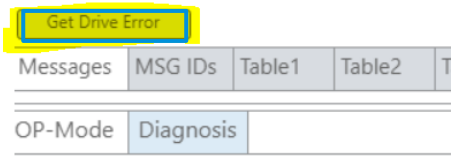
In T1_SMC_ERROR_CODE sheet:

22	34	SMC_AXIS_NOT_READY_FOR_MOTION	Axis in its current state cannot execute a motion command, because the axis doesn't signal currently that it follows the target values.

And in the reaction sheet:

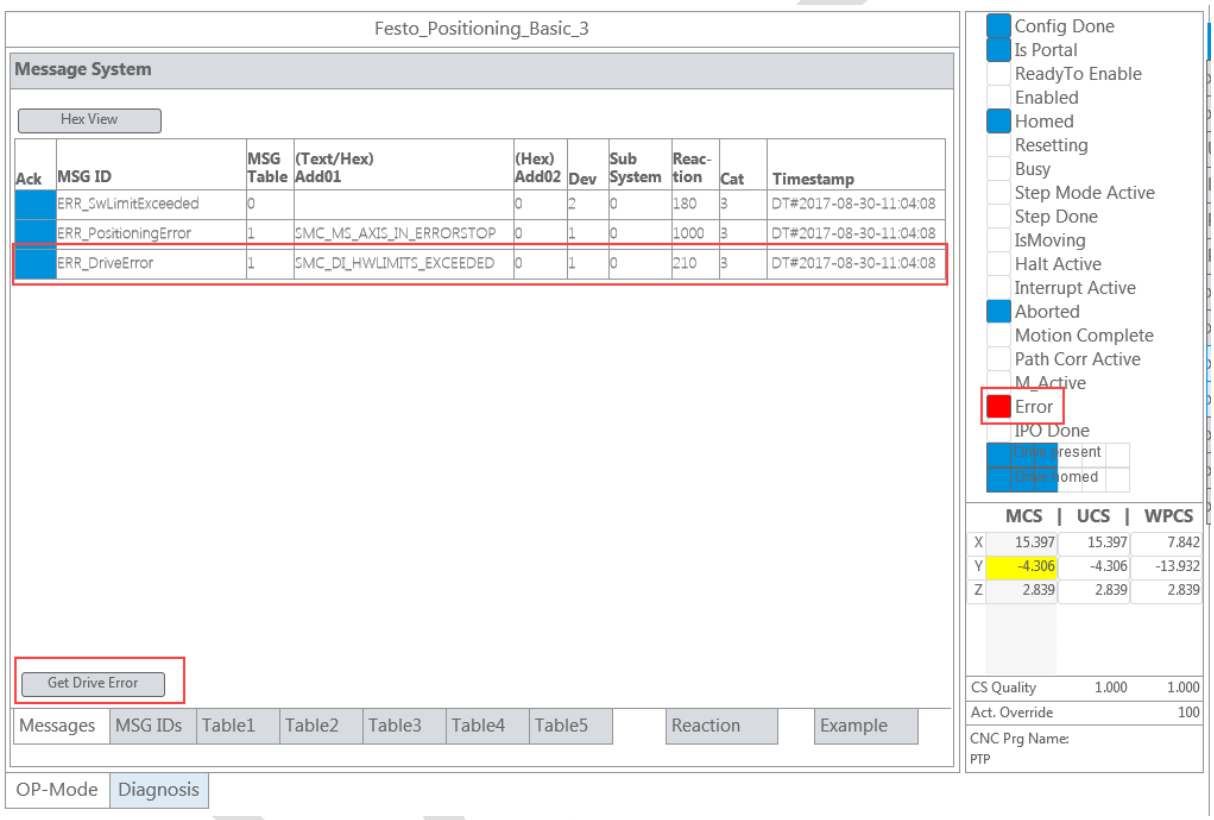
3E8	1000	Check_SMC_ERROR_Table

7.1 GET DRIVE ERROR

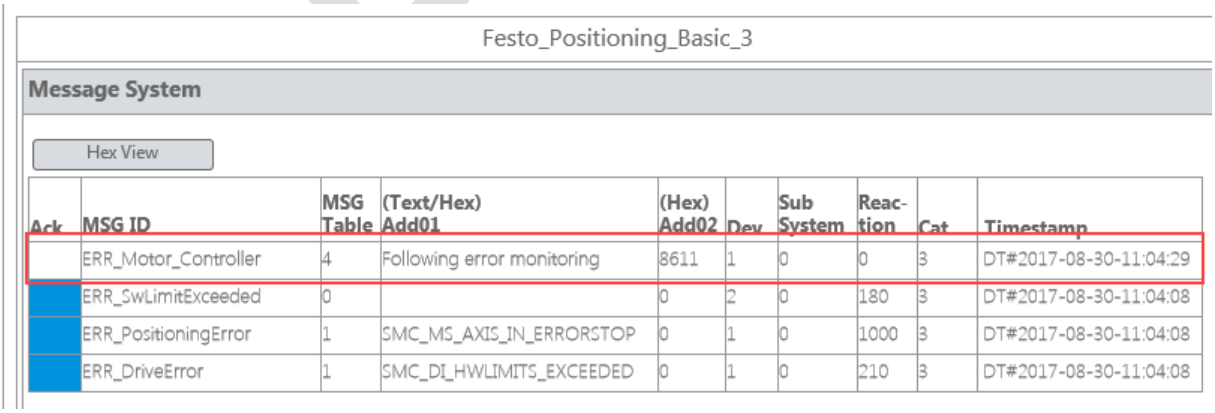


It's possible now to get information about the specific error of each drive.
Here we have a button "Get Drive Error":

Example:



Click "Get Drive Error"



Click on “Hex View”

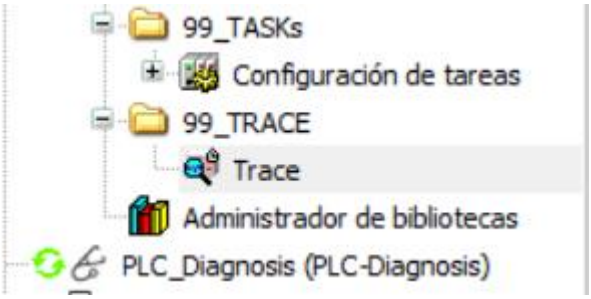
Ack	MSG ID	MSG Table	(Text/Hex) Add01	(Hex) Add02	Dev	Sub System	Reac-tion	Cat	Timestamp
	f0000030	4	8611	8611	1	0	0	3	DT#2017-08-30-11:04:29
	f0020060	0	0	0	2	0	180	3	DT#2017-08-30-11:04:08
	f0040060	1	b7	0	1	0	1000	3	DT#2017-08-30-11:04:08
	f0010040	1	b	0	1	0	210	3	DT#2017-08-30-11:04:08

Ack	MSG ID	MSG Table	(Text/Hex) Add01	(Hex) Add02	Dev	Sub System	Reac-tion	Cat	Timestamp
	f0000030	4	8611	8611	1	0	0	3	DT#2017-08-30-11:04:29
	f0020060	0	0	0	2	0	180	3	DT#2017-08-30-11:04:08
	f0040060	1	b7	0	1	0	1000	3	DT#2017-08-30-11:04:08
	f0010040	1	b	0	1	0	210	3	DT#2017-08-30-11:04:08

The corresponding message to the displayed error on the display of the controller can be found in the description of each drive.

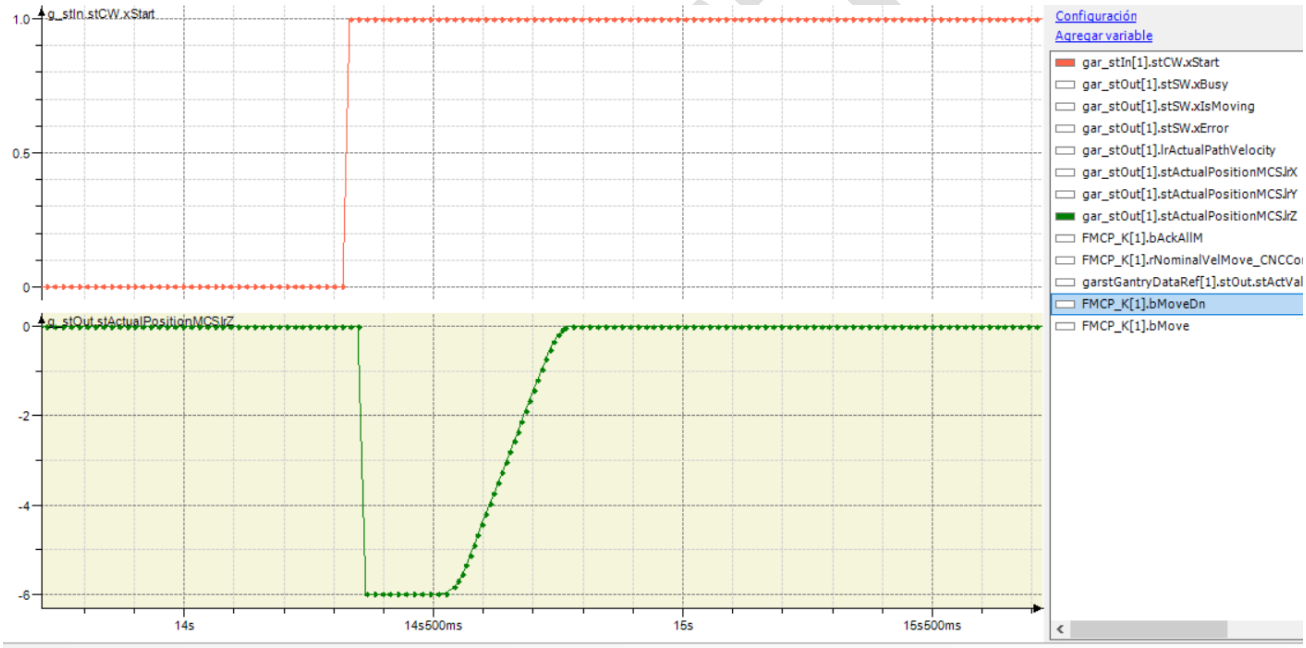
Error group 17		**Following error monitoring**	
no.	Code	Message	Reaction
17-0	8611h	**Following error monitoring**	configurable
		Cause	Comparison threshold for the limit value of the following error exceeded.
		Action	• Enlarge error window. • Parameterise acceleration to be less. • Motor overloaded (current limiter from the I^2t monitoring active?).

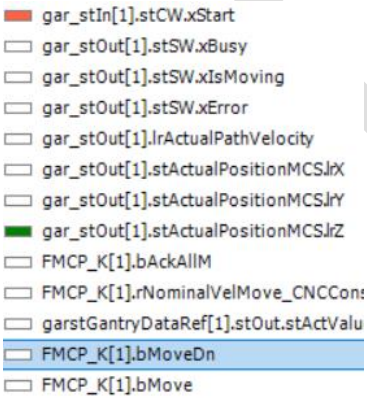
7.2 MONITORING SIGNALS:



If the case is abnormal operation or may be the system is not running and not showing any failure as well, there is a trace section available in the codesys program:

The start and error signals in the program can be monitoring through the time during the master plc execution program.





If you need monitoring more of the recommended signals, click right and “Visible”.