

### **CMMT servo drive commissioning with Allen-Bradley PLC through EtherNet/IP network**

This Application Note Gives Detailed Procedure to Integrate CMMT Servo Drive With Allen-Bradley Controller in EtherNet/IP Communication Network. Also Described to Control the Servo Motor in Different Control Operational Modes.

CMMT

Title .....CMMT servo drive commissioning with Allen-Bradley PLC through EtherNet/IP network  
Version ..... 1.60  
Document no. .... 100291  
Original .....en  
Author .....Festo  
  
Last saved ..... 20.05.2026

## Copyright Notice

This documentation is the intellectual property of Festo SE & Co. KG, which also has the exclusive copyright. Any modification of the content, duplication or reprinting of this documentation as well as distribution to third parties can only be made with the express consent of Festo SE & Co. KG.

Festo SE & Co KG reserves the right to make modifications to this document in whole or in part. All brand and product names are trademarks or registered trademarks of their respective owners.

## 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 SE & Co. KG.

Festo SE & 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 SE & Co. KG, which can be found at [www.festo.com](http://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](http://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.

# Table of Contents

|          |                                                                                       |           |
|----------|---------------------------------------------------------------------------------------|-----------|
| <b>1</b> | <b>CMMT Drive in Position Control Mode .....</b>                                      | <b>8</b>  |
| 1.1      | Introduction of Position Control AOI ( PTP_Drives_Festo_EIP ) .....                   | 8         |
| 1.2      | Hardware & Software Requirements .....                                                | 9         |
| 1.3      | Communication Architecture .....                                                      | 9         |
| 1.4      | Hardware Over View of CMMT Servo Drive .....                                          | 10        |
| <b>2</b> | <b>Hardware Configuration for Position Control Mode .....</b>                         | <b>11</b> |
| 2.1      | IP Address Configuration of Allen-Bradly PLC Using USB Serial Connection.....         | 11        |
| 2.2      | IP Address Configuration of Allen-Bradley PLC using EtherNet Connection .....         | 12        |
| 2.3      | IP Address Configuration of CMMT Drive Parametrizing Port - X18 ( Using FAS ) .....   | 15        |
| 2.4      | CMMT Servo Drive Configuration for Position Control Modes .....                       | 16        |
| 2.4.1    | Linear Axis Configuration .....                                                       | 16        |
| 2.4.2    | Rotary Axis Configuration .....                                                       | 23        |
| 2.5      | CMMT Drive Parameterisation for Integrate with PLC in Position Control Mode .....     | 30        |
| <b>3</b> | <b>Software Configurations for CMMT Position Control Mode .....</b>                   | <b>35</b> |
| 3.1      | Install CMMT Drive EDS File in Studio-5000 .....                                      | 35        |
| 3.2      | Add CMMT EDS File in Studio-5000 Ethernet Tree for Hardware Interface .....           | 39        |
| 3.3      | Add Generic EtherNet Module in Studio-5000 EtherNet Tree for Hardware Interface ..... | 41        |
| 3.4      | Import PTP_DRIVES_FESTO_EIP AOI in Studio – 5000 .....                                | 42        |
| 3.5      | Add PTP_DRIVES_FESTO_EIP AOI in PLC Program .....                                     | 44        |
| <b>4</b> | <b>PTP_DRIVES_FESTO_EIP AOI / Position Control .....</b>                              | <b>50</b> |
| 4.1      | Introduction of PTP_DRIVES_FESTO_EIP AOI.....                                         | 50        |
| 4.2      | IO Interface Area Tag Details .....                                                   | 51        |
| 4.3      | Control Area Tag Details .....                                                        | 51        |
| 4.3.1    | ConfigEPos Bit Description .....                                                      | 53        |
| 4.4      | Monitor Area Tag Details .....                                                        | 54        |
| 4.4.1    | Epos ZSW1 Bit Details.....                                                            | 55        |
| 4.4.2    | Epos ZSW2 Bit Details.....                                                            | 55        |
| 4.4.3    | Diagnostic Messages .....                                                             | 56        |
| <b>5</b> | <b>PTP_DRIVES_FESTO_EIP AOI Functions.....</b>                                        | <b>57</b> |
| 5.1      | Axis Enabling & Stopping .....                                                        | 57        |
| 5.2      | Axis Software & Hardware Position Limits Enabling .....                               | 57        |
| 5.3      | Axis Actual Speed Monitoring.....                                                     | 58        |
| 5.4      | AOI Mode of Operation .....                                                           | 58        |
| 5.4.1    | Relative Positioning Mode [ModePos – 1].....                                          | 58        |

|          |                                                                                                         |           |
|----------|---------------------------------------------------------------------------------------------------------|-----------|
| 5.4.2    | Absolute Positioning Task [ModePos – 2] .....                                                           | 59        |
| 5.4.3    | Setup Mode [ModePos – 3] .....                                                                          | 60        |
| 5.4.4    | Home/Reference Point using Festo Automation Suite Configuration [ModePos – 4] .....                     | 60        |
| 5.4.5    | Home to Current Position [ModePos – 5] .....                                                            | 61        |
| 5.4.6    | Traversing/Record Selection Mode [ModePos – 6] .....                                                    | 61        |
| 5.4.7    | Jog Mode [ModePos – 7] .....                                                                            | 62        |
| 5.4.8    | Inching/Incremental Jog Mode [ModePos – 8] .....                                                        | 62        |
| <b>6</b> | <b>CMMT Drive Parameter Read/Write .....</b>                                                            | <b>63</b> |
| 6.1      | Single Parameter Read/Write .....                                                                       | 63        |
| 6.2      | Multi Parameter Read/Write .....                                                                        | 66        |
| 6.2.1    | Introduction of PTP_DRIVES_PARA_MULTI AOI .....                                                         | 67        |
| 6.2.2    | Configure the PTP_DRIVES_PARA_MULTI AOI IO Tags .....                                                   | 67        |
| 6.2.3    | PTP_Drives_PARA_Multi AOI Read/Write MSG Instruction Configuration .....                                | 72        |
| 6.2.4    | PTP_DRIVES_PARA_MULTI AOI Control Area Tag Details .....                                                | 74        |
| 6.2.5    | PTP_DRIVES_PARA_MULTI AOI Monitor Area Tag Details .....                                                | 74        |
| 6.2.6    | PTP_DRIVES_PARA_MULTI AOI MSG Interface Tag Details .....                                               | 74        |
| 6.2.7    | Example for Read Parameters Using PTP_DRIVES_PARA_MULTI AOI .....                                       | 75        |
| 6.2.8    | Example for Write Parameters Using PTP_DRIVES_PARA_MULTI AOI .....                                      | 76        |
| <b>7</b> | <b>CMMT Drive Velocity Control Mode.....</b>                                                            | <b>77</b> |
| 7.1      | Hardware Configuration for Velocity Control Mode .....                                                  | 77        |
| 7.1.1    | CMMT Servo Drive Configuration for Velocity Control Mode .....                                          | 77        |
| 7.1.2    | CMMT Drive Parameterisation for Integrate with PLC in Velocity Control Mode .....                       | 84        |
| 7.2      | Software Configuration for CMMT Velocity Control Mode .....                                             | 89        |
| 7.2.1    | Add CMMT EDS File in Studio-5000 Ethernet Tree for Hardware Interface .....                             | 89        |
| 7.2.2    | Import Tel001_Velocity_Drives_Festo_EIP & Tel102_Velocity_Drives_Festo_EIP AOI's in Studio – 5000 ..... | 89        |
| 7.2.3    | Add “Tel001_Velocity_Drives_Festo_EIP” AOI in PLC Program.....                                          | 90        |
| 7.2.4    | Add “Tel102_Velocity_Drives_Festo_EIP” AOI in PLC Program.....                                          | 94        |
| <b>8</b> | <b>“Tel001_Velocity_Drives_Festo_EIP” AOI / Basic Velocity Control .....</b>                            | <b>98</b> |
| 8.1      | “Tel001_Velocity_Drives_Festo_EIP” AOI Introduction .....                                               | 98        |
| 8.2      | IO Interface Area Tag Details .....                                                                     | 98        |
| 8.3      | Control Area Tag Details .....                                                                          | 98        |
| 8.4      | Monitor Area Tag Details .....                                                                          | 99        |
| 8.4.1    | Diagnostic Messages / Function Block Faults .....                                                       | 100       |

|           |                                                                                |            |
|-----------|--------------------------------------------------------------------------------|------------|
| 8.5       | Quick Procedure to Run the CMMT Drive in Basic Velocity Control AOI .....      | 100        |
| <b>9</b>  | <b>“Tel102_Velocity_Drives_Festo_EIP” AOI / Advance Velocity Control .....</b> | <b>103</b> |
| 9.1       | “Tel102_Velocity_Drives_Festo_EIP” AOI Introduction .....                      | 103        |
| 9.2       | IO Interface Area Tag Details .....                                            | 103        |
| 9.3       | Control Area Tag Details .....                                                 | 103        |
| 9.4       | Monitor Area Tag Details .....                                                 | 104        |
| 9.4.1     | Diagnostic Messages / Function Block Faults .....                              | 104        |
| 9.5       | Quick Procedure to Run the CMMT Drive in Advance Velocity Control AOI .....    | 105        |
| <b>10</b> | <b>“ExtendedProcessData_Festo_EIP” AOI .....</b>                               | <b>108</b> |
| 10.1      | Extended Process Data (EPD) Introduction .....                                 | 108        |
| 10.2      | Configure Extended Process Data in Festo Automation Suite .....                | 109        |
| 10.2.1    | Read Extended Data Configuration in Festo Automation Suite .....               | 109        |
| 10.2.2    | Write Extended Data Configuration in Festo Automation Suite .....              | 110        |
| 10.3      | Extended Process Data Read/Write using a Generic Module .....                  | 111        |
| 10.3.1    | Create MSG Instruction for EPD Read Parameters .....                           | 111        |
| 10.3.2    | Create MSG Instruction for EDP Write Parameters .....                          | 115        |
| 10.3.3    | Import EPD AOI & Configure the IO Tags .....                                   | 119        |
| 10.3.4    | Add “ExtendedProcessData_Festo_EIP” AOI in PLC Program .....                   | 120        |
| 10.4      | Extended Process Data Read/Write using an EDS File .....                       | 123        |
| 10.4.1    | Configure EPD in CMMT Derive Ethernet/IP Module .....                          | 123        |
| 10.4.2    | Import EPD AOI & Configure IO Tags .....                                       | 124        |
| 10.4.3    | Add “ExtendedProcessData_Festo_EIP” AOI in PLC Program .....                   | 126        |
| 10.5      | “ExtendedProcessData_Festo_EIP” AOI Input & Output Tag Details .....           | 129        |
| 10.5.1    | Interface Tag Details .....                                                    | 129        |
| 10.5.2    | Input Tag Details .....                                                        | 129        |
| 10.5.3    | Output Tag Details .....                                                       | 130        |
| 10.6      | Configuration of the EPD Input and EPD Output Data .....                       | 131        |
| 10.7      | Diagnostics .....                                                              | 131        |
| 10.8      | Application Example .....                                                      | 132        |
| 10.9      | Extended Process Data 64Bit AOI Read/Write using an EDS File .....             | 133        |
| 10.9.1    | Import EPD 64Bit AOI & Configuring IO Tags .....                               | 133        |
| 10.9.2    | Adding “ExtendedProcessData_Festo_EIP_64Bit_AOI” in PLC Program .....          | 135        |
| 10.10     | “ExtendedProcessData_Festo_EIP_64Bit” AOI Input & Output Tag Details .....     | 137        |
| 10.10.1   | Interface Tag Details .....                                                    | 137        |
| 10.10.2   | Input Tag Details .....                                                        | 137        |
| 10.10.3   | Output Tag Details .....                                                       | 137        |

## Table of contents

|                                                                                                                                |            |
|--------------------------------------------------------------------------------------------------------------------------------|------------|
| 10.11 Configuration of the EPD Input and EPD Output Data .....                                                                 | 138        |
| 10.12 Diagnostics.....                                                                                                         | 139        |
| 10.13 Application Example .....                                                                                                | 140        |
| <b>11 “CMMT Full Parameter AOI”/ Full Parameter of CMMT Drive .....</b>                                                        | <b>141</b> |
| 11.1 “CMMT Full Parameter” AOI Introduction .....                                                                              | 141        |
| 11.2 Configure FAS .....                                                                                                       | 141        |
| 11.2.1 Creating Rockwell file L5X.....                                                                                         | 141        |
| 11.3 Import CMMT Full Parameter AOI .....                                                                                      | 144        |
| 11.4 Replacing pre-existing AOI by Overwriting .....                                                                           | 146        |
| 11.4.1 Step to modify IP address of CMMT-xx in MyClientInterface: .....                                                        | 148        |
| 11.5 Add “CMMT_FullParam” AOI in PLC Program .....                                                                             | 149        |
| 11.6 CMMT Full Parameter AOI Tag’s Configuration .....                                                                         | 150        |
| 11.6.1 Step to add IP address of CMMT-xx in MyClientInterface:.....                                                            | 151        |
| 11.7 CMMT_FullParam MSG Instruction Configuration.....                                                                         | 154        |
| 11.7.1 Create_Message Instruction Configuration .....                                                                          | 154        |
| 11.7.2 Connect_Message Instruction Configuration .....                                                                         | 155        |
| 11.7.3 Read_Message Instruction Configuration .....                                                                            | 156        |
| 11.7.4 Write_Message Instruction Configuration.....                                                                            | 157        |
| 11.7.5 Delete_Message Instruction Configuration .....                                                                          | 159        |
| 11.7.6 Communication Path.....                                                                                                 | 160        |
| 11.7.7 CMMT_FullParam AOI Tag Details .....                                                                                    | 163        |
| 11.8 Diagnostics.....                                                                                                          | 164        |
| 11.8.1 ErrorID.....                                                                                                            | 164        |
| 11.8.2 ExtErrorID.....                                                                                                         | 165        |
| 11.8.3 Example .....                                                                                                           | 166        |
| 11.9 Copy all default values of parameters and local tags whose values were modified to all tags of the instruction type ..... | 167        |
| 11.10 Application Example .....                                                                                                | 168        |
| <b>12 “Festo_RecordTable_EIP” AOI .....</b>                                                                                    | <b>171</b> |
| 12.1 Festo_RecordTable_EIP AOI Introduction .....                                                                              | 171        |
| 12.1.1 IO Interface Area Tag Details.....                                                                                      | 171        |
| 12.1.2 MSG Interface Tag Details.....                                                                                          | 171        |
| 12.1.3 Control Area Tag Details.....                                                                                           | 171        |
| 12.1.4 Monitor Area Tag Details.....                                                                                           | 172        |
| 12.1.5 Diagnostic Messages / Function Block Faults .....                                                                       | 172        |
| 12.2 Festo_MM_RecordStruct .....                                                                                               | 173        |
| 12.3 Import Festo_RecordTable_EIP AOI .....                                                                                    | 180        |
| 12.4 Add and Configure the “Festo_RecordTable_EIP” AOI IO Tags .....                                                           | 181        |
| 12.4.1 Festo_RecordTable_EIP AOI Read/Write MSG Instruction Configuration .....                                                | 185        |

|           |                                                                      |            |
|-----------|----------------------------------------------------------------------|------------|
| 12.5      | Application Example .....                                            | 187        |
| 12.5.1    | Record_Type .....                                                    | 187        |
| 12.5.2    | Example for Read Record Table Using Festo_RecordTable_EIP AOI .....  | 188        |
| 12.5.3    | Example for Write Record Table Using Festo_RecordTable_EIP AOI ..... | 189        |
| <b>13</b> | <b>CMMT Full Param SD AOI .....</b>                                  | <b>191</b> |
| 13.1      | Introduction of CMMT Full Param SD AOI .....                         | 191        |
| 13.2      | Configure FAS .....                                                  | 191        |
| 13.2.1    | Creating Rockwell file L5X .....                                     | 191        |
| 13.3      | Import CMMT Full Parameter SD AOI .....                              | 195        |
| 13.4      | Adding “CMMT_FullParam_SD” AOI in PLC Program .....                  | 196        |
| 13.5      | CMMT FullParam_SD AOI Tag’s Configuration .....                      | 197        |
| 13.6      | CMMT_FullParam_SD AOI MSG Instruction Configuration .....            | 200        |
| 13.6.1    | SDReadFileMSG Instruction Configuration .....                        | 200        |
| 13.6.2    | SDFileOpenMSG Instruction Configuration .....                        | 201        |
| 13.6.3    | SDSetFilePointerMSG Instruction Configuration .....                  | 203        |
| 13.6.4    | SDCloseFileMSG Instruction Configuration .....                       | 204        |
| 13.6.5    | Create_Message Instruction Configuration .....                       | 205        |
| 13.6.6    | Connect_Message Instruction Configuration .....                      | 206        |
| 13.6.7    | Read_Message Instruction Configuration .....                         | 207        |
| 13.6.8    | Write_Message Instruction Configuration .....                        | 209        |
| 13.6.9    | Delete_Message Instruction Configuration .....                       | 210        |
| 13.6.10   | Communication Path .....                                             | 211        |
| 13.7      | CMMT_FullParam AOI Tag Details .....                                 | 215        |
| 13.7.1    | CMMT_Full_Param_SD In-Out Variables .....                            | 215        |
| 13.7.2    | CMMT_Full_Param_SD Input Variables .....                             | 215        |
| 13.7.3    | CMMT_Full_Param_SD Output Variables .....                            | 215        |
| 13.8      | Diagnostics .....                                                    | 216        |
| 13.8.1    | ErrorID .....                                                        | 216        |
| 13.9      | Sample Code For CMMT_Full_Param_SD AOI .....                         | 217        |
| 13.9.1    | Reading the .l5x File and Writing Data to the CMMT Drive .....       | 219        |

# 1 CMMT Drive in Position Control Mode

## 1.1 Introduction of Position Control AOI ( PTP\_Drives\_Festo\_EIP )

The main use of this application note is to describe the communication between CMMT Servo Drive and Allen-Bradley controller through EtherNet/IP network. The below function block is used to give the Position control command, Start command, Stop command, Jogging, Incremental jogging, Set homing reference & Move the axis to set homing reference point.



### Note

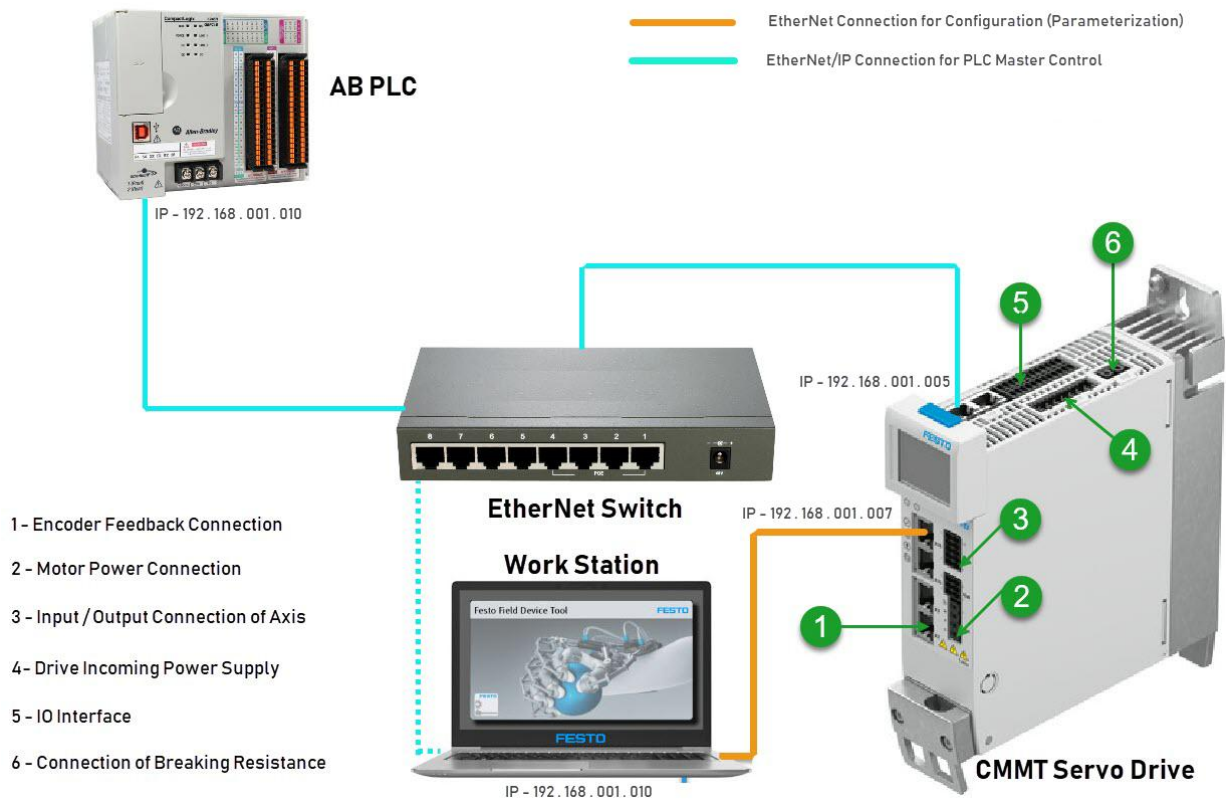
- Refer [Chapter - 10.1](#) for Extended Process Data

| PTP_Drives_Festo_EIP  |                                     | Axis_X_FB ... |                      |
|-----------------------|-------------------------------------|---------------|----------------------|
| PTP_Drives_Festo_EIP  |                                     | 0             | (AxisEnabled)        |
| I_O_I_n_t_e_r_f_a_c_e |                                     |               |                      |
| DRV_Status            | CMMT_AS_Axis_X:I1.Data              |               | (AxisPosOk)          |
| DRV_Control           | CMMT_AS_Axis_X:O1.Data              |               |                      |
| C_o_n_t_r_o_l_A_r_e_a |                                     | 0             | (AxisSpFixed)        |
| ModePos               |                                     | 2             |                      |
| EnableAxis            |                                     | 0             | (AxisAckSetpoint)    |
| CancelTraversing      |                                     | 1             |                      |
| IntermediateStop      |                                     | 1             | (ClampTorqueReached) |
| Positive              |                                     | 0             | (AxisRef)            |
| Negative              |                                     | 0             |                      |
| Jog1                  |                                     | 0             | (AxisWarn)           |
| Jog2                  |                                     | 0             | (AxisError)          |
| AckError              |                                     | 0             | (Lockout)            |
| ReleaseBrake          |                                     | 0             |                      |
| TravelToFixStop       |                                     | 0             | (Error)              |
| ExecuteMode           |                                     | 0             |                      |
| Position              |                                     | 0             |                      |
| Velocity              |                                     | 0             |                      |
| OverV                 |                                     | 100           |                      |
| OverAcc               |                                     | 100           |                      |
| OverDec               |                                     | 100           |                      |
| ConfigEPos            |                                     | 15            |                      |
| BaseSpeedValue        |                                     | 3000.0        |                      |
| ConnectionFaulted     | CMMT_AS_Axis_X:I1.ConnectionFaulted | 0             |                      |
| M_o_n_i_t_o_r_A_r_e_a |                                     | 0             |                      |
| ActVelocity           |                                     | 0.0           |                      |
| ActPosition           |                                     | 0             |                      |
| ActMode               |                                     | 0             |                      |
| EposZSW1              |                                     | 0             |                      |
| EposZSW2              |                                     | 0             |                      |
| ActWarn               |                                     | 0             |                      |
| ActFault              |                                     | 0             |                      |
| Status                |                                     | 16#0000       |                      |

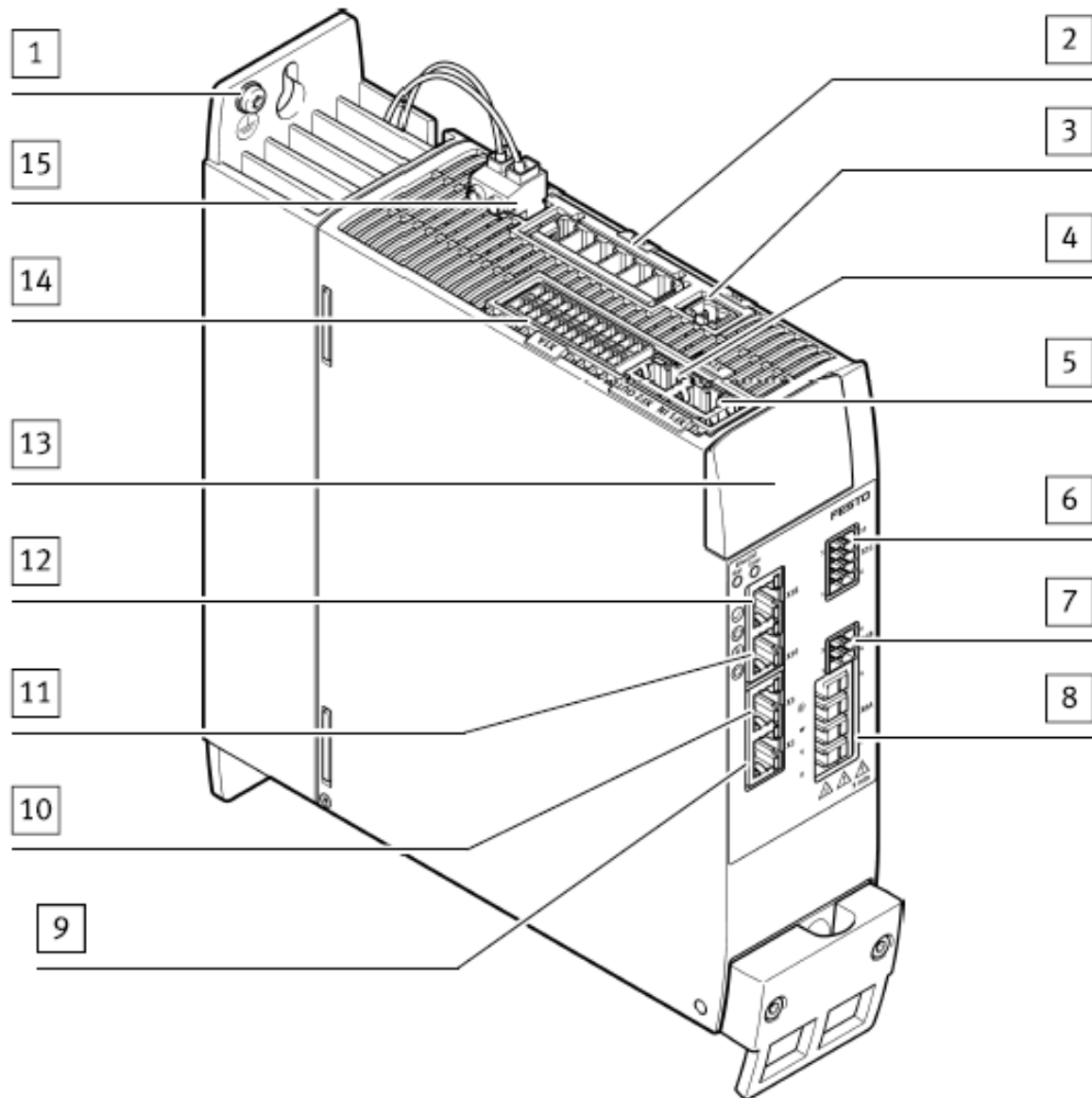
## 1.2 Hardware & Software Requirements

| Device Name                            | Version (or) Part Number            | Device Type |
|----------------------------------------|-------------------------------------|-------------|
| CMMT Drive                             | V36.12.0                            | Hardware    |
| Servo Motor                            | Based on Application                | Hardware    |
| Motor Power Cable                      | Based on Application                | Hardware    |
| Motor Encoder Cable                    | Based on Application                | Hardware    |
| Servo Drive Connection Terminal Blocks | Based on Application                | Hardware    |
| EtherNet Cable                         | NEBC-R3G4-ES-1-S-R3G4-ET            | Hardware    |
| Allen-Bradley PLC                      | Version Free / Studio 5000 Platform | Hardware    |
| CMMT_EIP_Motion Library                | Latest                              | Software    |
| Festo Automation Suite                 | V2.11.0.657 & Above                 | Software    |
| Rockwell Studio 5000                   | Version Free                        | Software    |

## 1.3 Communication Architecture



## 1.4 Hardware Overview of CMMT Servo Drive



- |                                                   |                                                                |
|---------------------------------------------------|----------------------------------------------------------------|
| 1 PE connection, housing                          | 9 [X2] Encoder connection 1                                    |
| 2 [X9A] Mains and intermediate circuit connection | 10 [X3] Encoder connection 2                                   |
| 3 [X9C] Logic voltage                             | 11 [X10] Device synchronisation                                |
| 4 [XF2 OUT] RTE interface port 2                  | 12 [X18] Standard Ethernet                                     |
| 5 [XF1 IN] RTE interface port 1                   | 13 [X5] Connection for operating unit (behind the blind plate) |
| 6 [X1C] Inputs/outputs for the axis               | 14 [X1A] I/O interface                                         |
| 7 [X6B] Motor auxiliary connection                | 15 [X9B] Connection for braking resistor                       |
| 8 [X6A] Motor phase connection                    |                                                                |

## 2 Hardware Configuration for Position Control Mode

This chapter describes about the hardware configurations of PLC & CMMT servo drive.

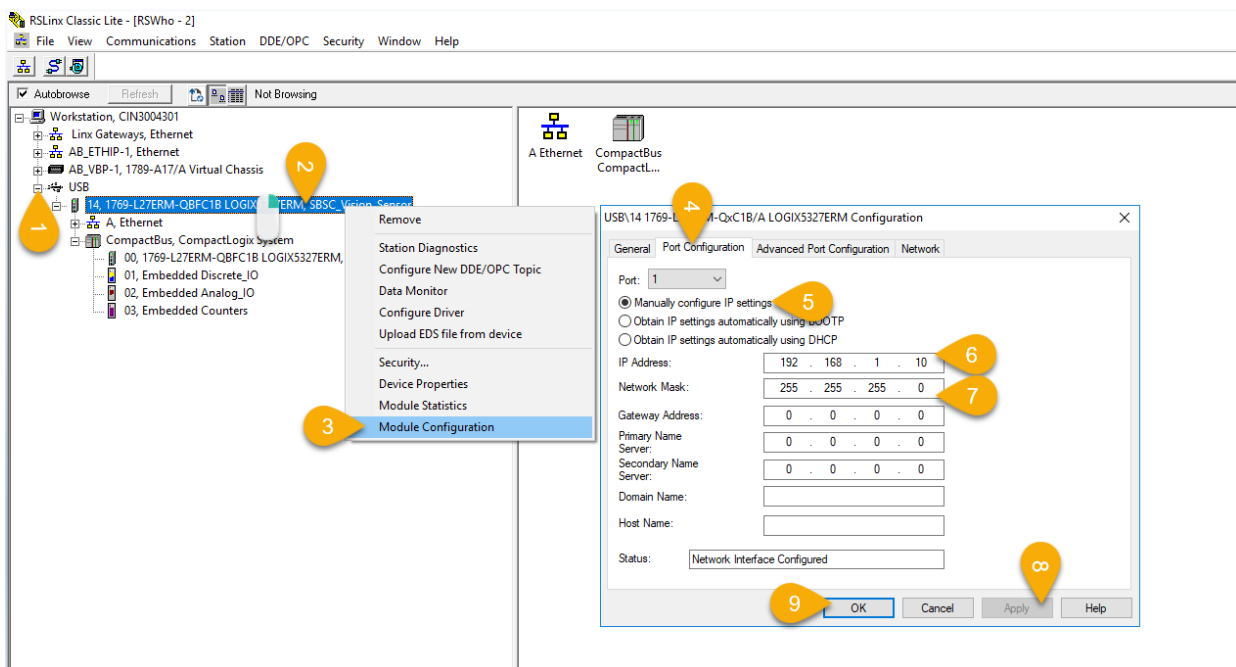
### 2.1 IP Address Configuration of Allen-Bradley PLC Using USB Serial Connection

Before making the IP address configuration in Allen-Bradley controller make sure user has connected USB type-B serial cable between PLC & PC. Once connection is done, Open RSLinx software from windows **Start > Rockwell Automation Software > RSLinx**.



#### Note

- This chapter explains IP address configuration of Allen-Bradley PLC via USB Serial cable. If USB cable is not available then the user can try to configure IP address via BootP-DHCP Tool by using MAC address with Ethernet cable as Shown in [Chapter 2.2](#).



1. Expand the “**USB driver**” folder by clicking plus symbol of the folder.
2. Select the right controller and right click.
3. Select “**Module Configuration**” from right click menu.
4. Select “**Port Configuration**” tab from newly opened configuration window.
5. Select “**Manually configure IP settings**” to enable static IP mode in controller.
6. Enter the new “**IP address**” to the controller.
7. Enter the “**Subnet mask**” to the controller.
8. Click “**Apply**” to confirm the configuration.
9. Click “**OK**” to close the configuration window.



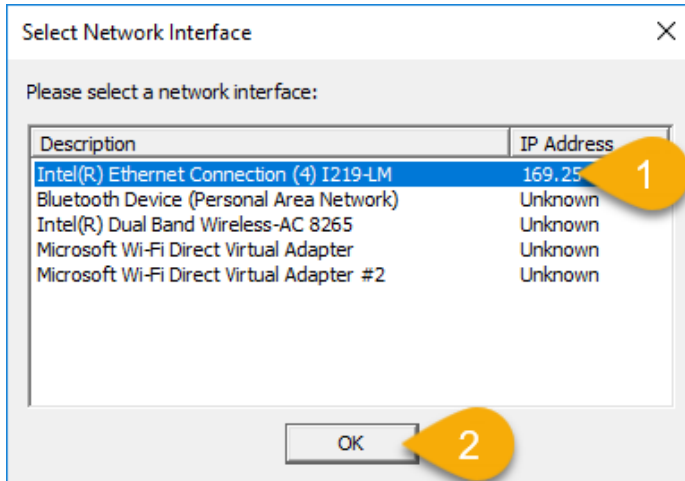
#### Note

- After IP address configuration power restart the controller to ensure static IP mode is enabled & to ensure correct IP address is assigned.

## 2.2 IP Address Configuration of Allen-Bradley PLC using EtherNet Connection

This step gives detailed explanation about IP address configuration of Allen-Bradley PLC through BootP-DHCP Tool by using LAN cable. This method of IP configuration will be used when BootP or DHCP option are enabled in PLC communication settings. Note the MAC ID of the PLC before starting IP configuration process. The MAC ID of this sample code PLC is E4:90:69:BE:BF:5C.

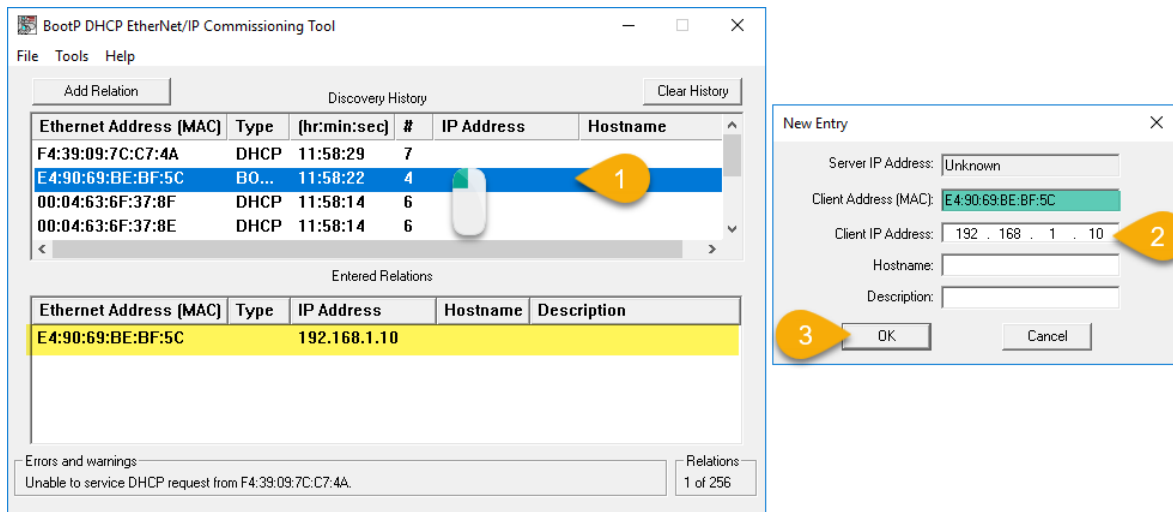
### Step -1



Open IP Configuration software from “**Windows**” **Start** > **Rockwell Software Tool** > **BootP-DHCP Tool**.

1. Select Network Interface Adapter card. Here selection was done for PC local LAN connection.
2. Select “**OK**” to conform the selection.

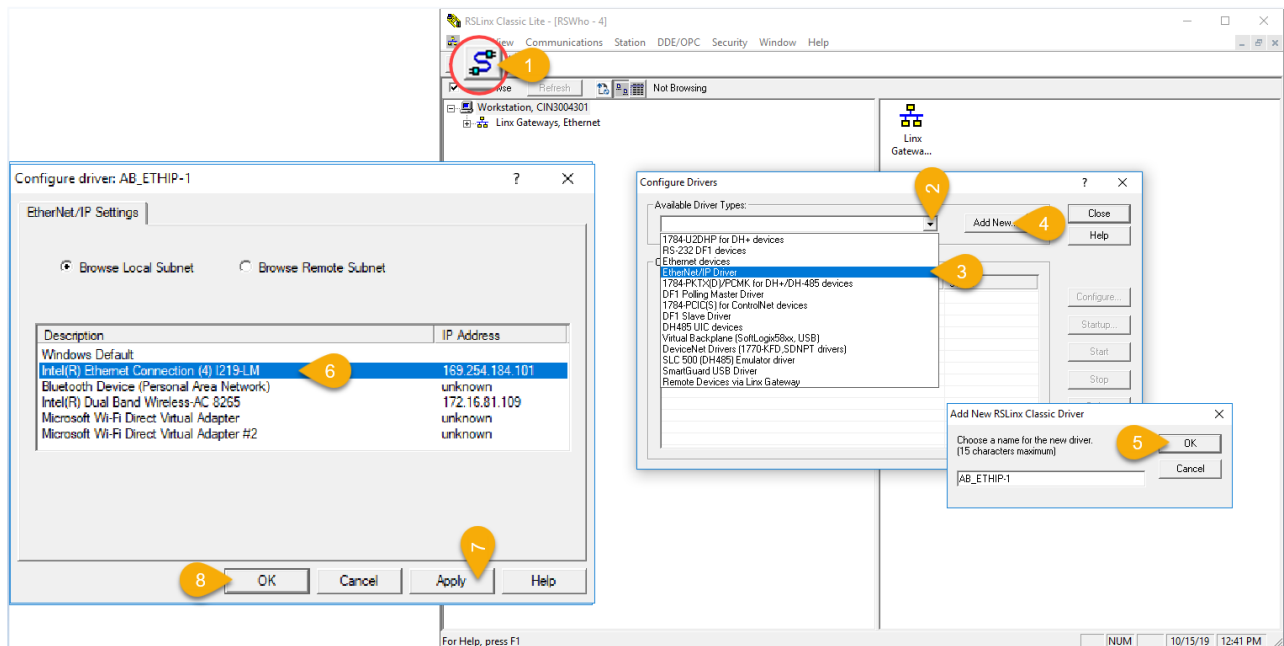
### Step - 2



1. Select “**MAC address**” of the user PLC which was noted initially.
2. Enter the new IP address of the PLC. PLC IP address series must be same series as CMMT drive IP.
3. Select “**OK**” button to confirm the new IP address.

**Note**

- Once IP address is conformed as highlighted in yellow color new IP address and its respective MAC ID will be updated in the Entered Relations window.
- **At this point only IP address is assigned for PLC, but still the PLC communication setting is in BootP mode. It leads PLC to automatically erase its IP address during power restart. So, Step-4 is important to avoid this situation.**

**Step – 3**

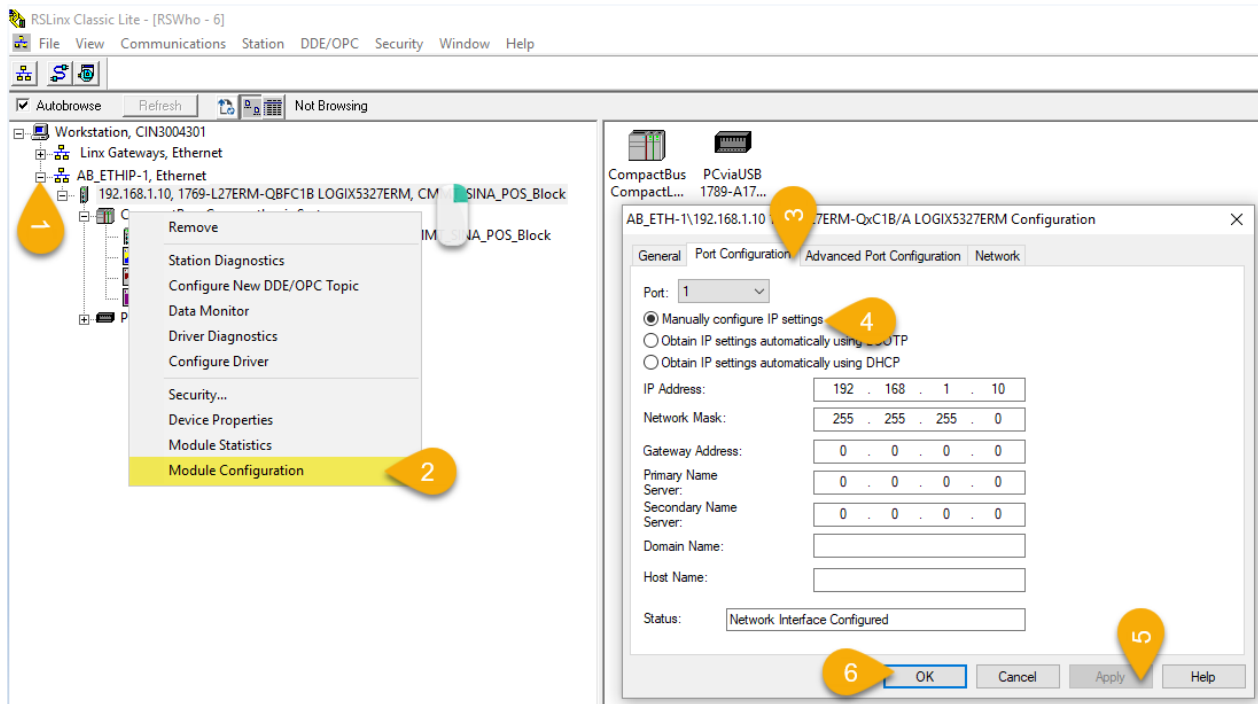
It's important to configure a drive in RSLinx software to establish the connection with PLC. Open the RSLinx software from **"Windows" Start > Rockwell Software > RSLinx**.

1. Select **"Configure Drives"** symbol from tool bar.
2. Click Drives **drop-down menu** from configure driver's window.
3. Select **"EtherNet/IP Drive"** from drop-down list.
4. Click **"Add New"** button from configure driver's window.
5. Click **"OK"** button with default name.
6. Select **EtherNet interface card**. Here selection is done for local LAN port by selecting **"Intel(R) Ethernet Connection (4) I219-LM"**.
7. Click **"Apply"** button to confirm the configuration.
8. Click **"OK"** button to confirm and close the configuration.

**Note**

- Make sure the IP series of user workstation must be in same series as like PLC IP address series.

### Step – 4



This step will describe to change PLC EtherNet/IP port setting from BOOTP mode to Static IP mode.

1. Open the drive folder which created in previous step by clicking on plus symbol.
2. Select PLC Icon and right click to select **“Module Configuration”** option.
3. Select **“Port configuration”** tab from newly opened configuration window.
4. Select **“Manual configuration IP settings”** to enable static IP mode.
5. Click **“Apply”** button to confirm the changes.
6. Click **“OK”** to confirm and close the pop-up window.

PLC communication settings are done at this stage. Ensure once static IP mode is activated by restarting PLC.

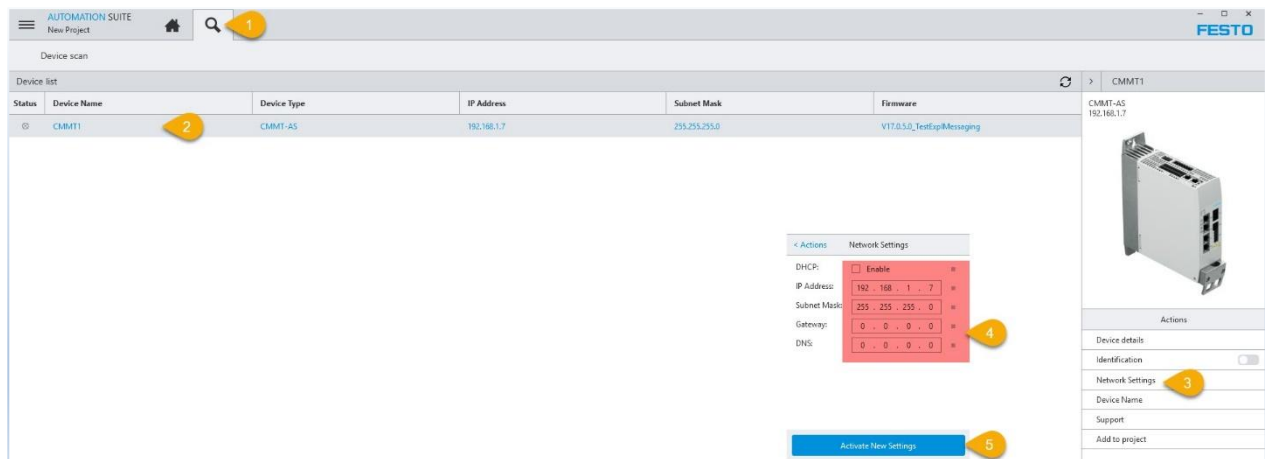
## 2.3 IP Address Configuration of CMMT Drive Parametrizing Port - X18 ( Using FAS )

This chapter describes about IP address configuration of CMMT drive parametrizing port using Festo Automation Suite.



### Note

- Before starting CMMT drive IP address configuration process make sure the hardware connection between PC & servo drive is done as per communication architecture shown in [chapter 1.3](#). In the [chapter 1.3](#) yellow color connection is only for parametrization and after parametrization to establish connection with PLC, connect the PC with Ethernet switch as shown dotted blue color line.



1. Click “**Device Scan**” button from tool bar.
2. Select correct device by its name.
3. Select “**Network Setting**” from action window.
4. From network setting window **Uncheck the DHCP enable, Enter the new address** as per local network as highlighted in red colour.
5. Finally click “**Activate New Setting**” to update the changes.

## 2.4 CMMT Servo Drive Configuration for Position Control Modes

### 2.4.1 Linear Axis Configuration

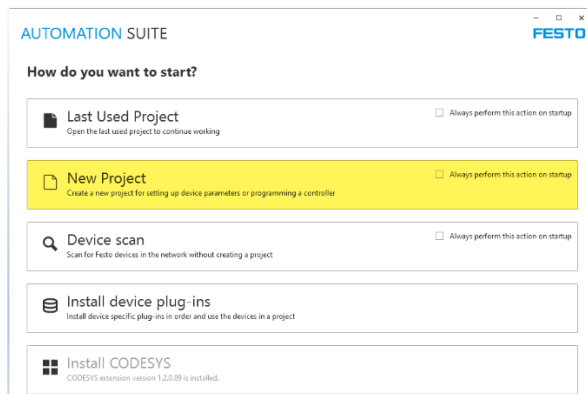


#### Note

- If user looking for Rotary Axis configuration skip this chapter and jump to [chapter – 2.4.2](#).

This chapter describe about parameterization of CMMT drive for linear axis to move 100 mm stock length with 0.50 m/s as base speed & current position as homing mode.

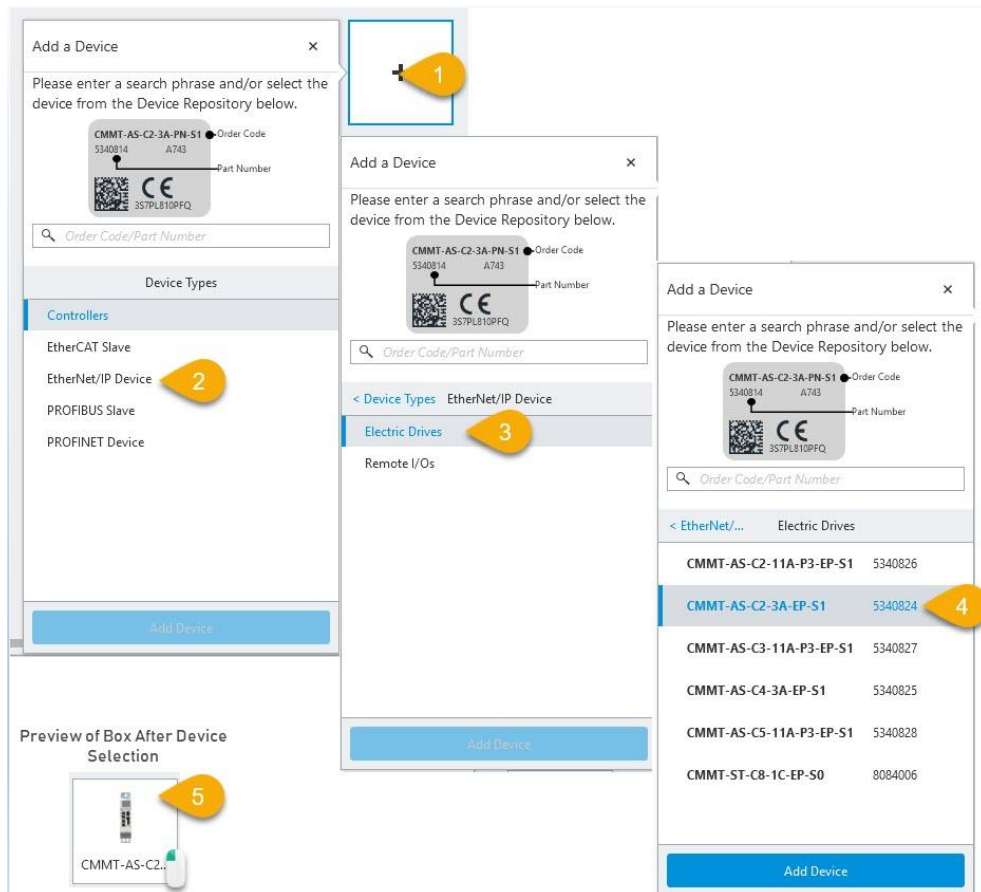
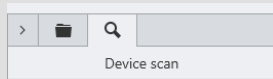
#### **Step – 1**



Open the Automation Suite software from Windows **Start > Festo Software > AUTOMATION SUITE** and select **“New Project”** from start-up menu.

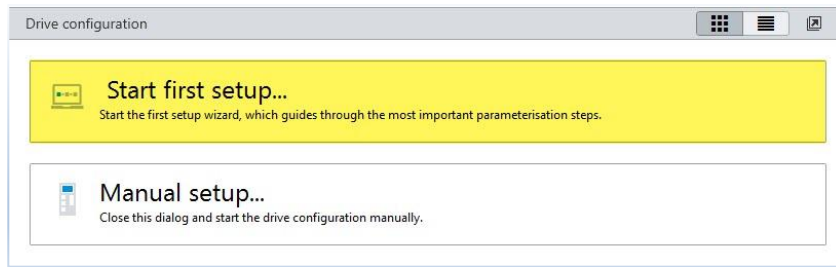
**Step – 2****Note**

- Follow this step if drive is not in online. If the drive is connected with the PC as shown in [chapter 1.3](#) simply click Device scan from home ( Next to device Repository ) and online devices will be visible in available devices list. Select the correct device based on Its configured name & IP address then double click.



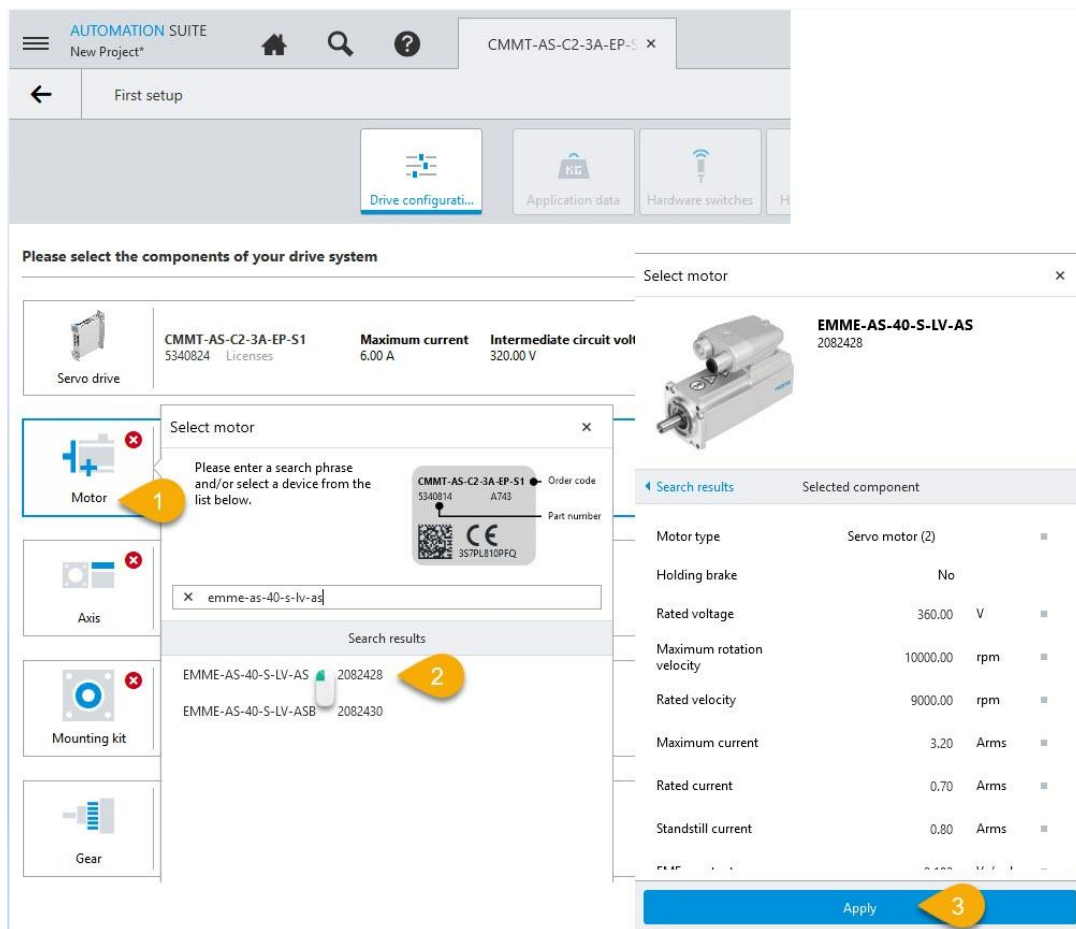
1. Select any empty device box and click add button.
2. Select “**EtherNet/IP Device**” from Add a Device menu.
3. Select “**Electric Drives**” from device type menu.
4. **Select correct drive part number** from electric drives list. As per this sample application CMMT-AS-C2-3A-EP-S1 is selected.
5. Double click the newly added drive to initialize configuration parameters.

## Step – 3



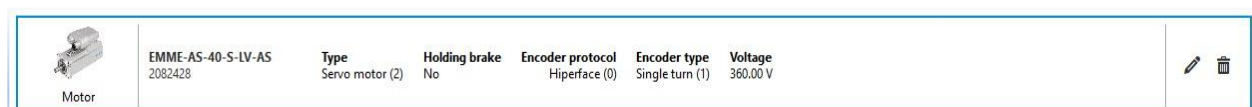
Select **“Start first setup”** from drive configuration menu.

## Step – 4



1. Click **“Motor”** from components of drive system menu for specific motor model.
2. **Select correct motor** based on its **part number** and double click. As per this sample application “EMME-AS-40-S-LV-AS” motor is used.
3. Cross verify the motor parameters & click apply button to ensure motor configuration.

View of motor tab after configuration



### Step – 5

**Please select the components of your drive system**

---



Servo drive

CMMT-AS-C4-3A-EP-S1  
5340825 Licenses

Maximum current  
12.00 A

Intermediate circuit voltage  
320.00 V

---



Motor

| EMME-AS-40-S-LV-AS | Type | Holding brake | Encoder protocol interface (0) | Encoder type | Voltage |
|--------------------|------|---------------|--------------------------------|--------------|---------|
| Select axis        |      |               |                                |              |         |

Please enter a search phrase and/or select a device from the list below.

Order code

Part number

CMMT-AS-C4-3A-EP-S1

5340814 A743



357PL810PFQ

- User defined linear axis
- User defined rotative axis
- DGE-8-%%-ZR 193739
- DGE-8-%%-ZR-HD12-GK 193739
- DGE-8-%%-ZR-HD8-GK 193739

---



Axis

Select axis

Please enter a search phrase and/or select a device from the list below.

Order code/part number

CMMT-AS-C4-3A-EP-S1

5340814 A743



357PL810PFQ

Select axis

User defined linear axis



| Search results    | Selected component              |
|-------------------|---------------------------------|
| Current user unit | Metric [m, m/s, ...] (6)        |
| Motion            | Linear                          |
| Unlimited axis    | <input type="checkbox"/> Active |
| Working stroke    | 100.00 mm                       |
| Feed constant     | 10.00 mm/r                      |
| Design axis       | Single axis (0)                 |

Apply

This step describes about the axis type configuration. As specified initially here axis configuration is done for 100 mm in linear stock length. This configuration will change based on user application type.

**Step – 6**

**Select mounting kit**

Please enter a search phrase and/or select a device from the list below.

Order code: CMMT-AS-C2-3A-EP-S1  
Part number: 5340814 A743  
357PL810PFQ

Search results

| Search results            |         |
|---------------------------|---------|
| User defined mounting kit |         |
| EAMM-A-A19-40A            | 559798  |
| EAMM-A-A19-42A            | 558895  |
| EAMM-A-A22-55A            | 559799  |
| EAMM-A-A22-57A            | 558897  |
| EAMM-A-A22-70A            | 558898  |
| EAMM-A-D100-100A          | 3356796 |
| EAMM-A-D100-100A-S1       | 3356931 |

**User defined mounting kit**

Selected component

Type: Inertia Coupling

Axial: 10 kgm<sup>2</sup>

Apply

Back Next Finish

Gear ratio (total): 1:1

New Project: CMMT-AS-C2-3A-EP-S1 (Plug-in: CMMT-AS Plug-in V1.2.2.4) Festo Automation Suite V1.2.1.1

This step illustrates about mounting kit configuration. This configuration will vary depending on user application type. At the end of this step click next button to move to next level configuration.

**Step – 7**

**Application data**

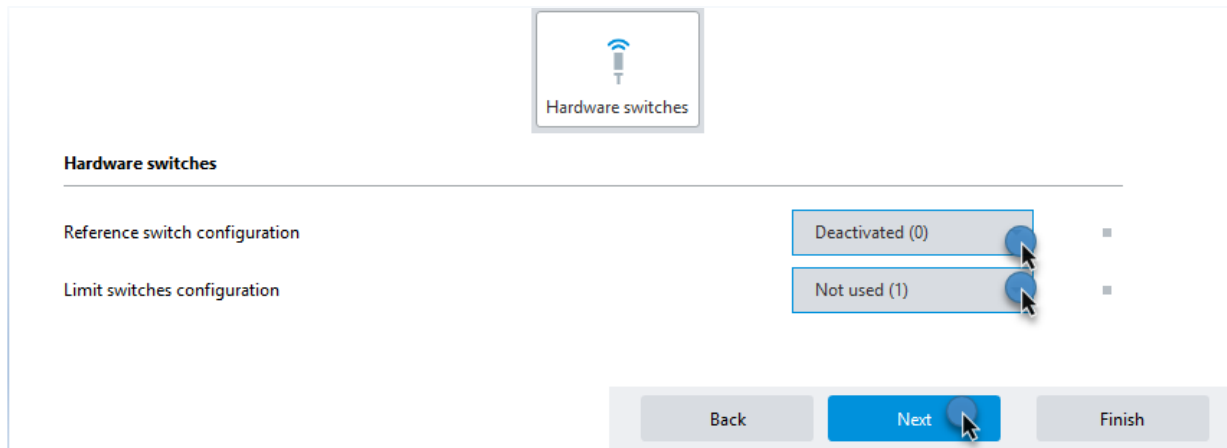
Axis mass: 0.00 kg

Application mass: 10.00 kg

Total mass: 10.00 kg

Back Next Finish

Specify inertia value of user application. This parameter value will vary depend on application type. Click next button for next level configuration.

**Step -8**


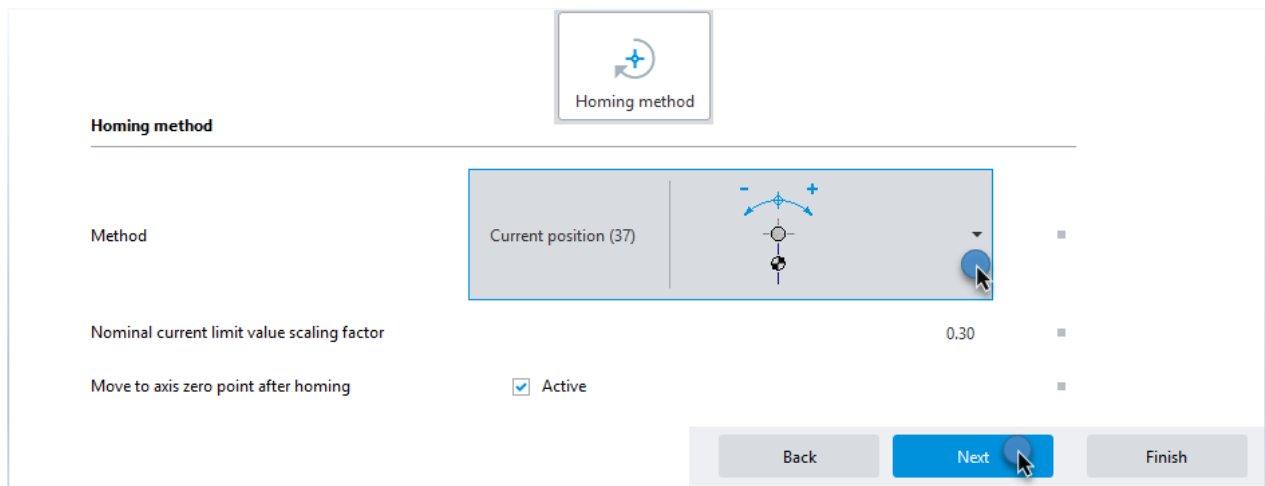
**Hardware switches**

Reference switch configuration: Deactivated (0)

Limit switches configuration: Not used (1)

Back Next Finish

Select the homing & maximum limit switch connection type. Click next button for next level configuration.

**Step – 9**


**Homing method**

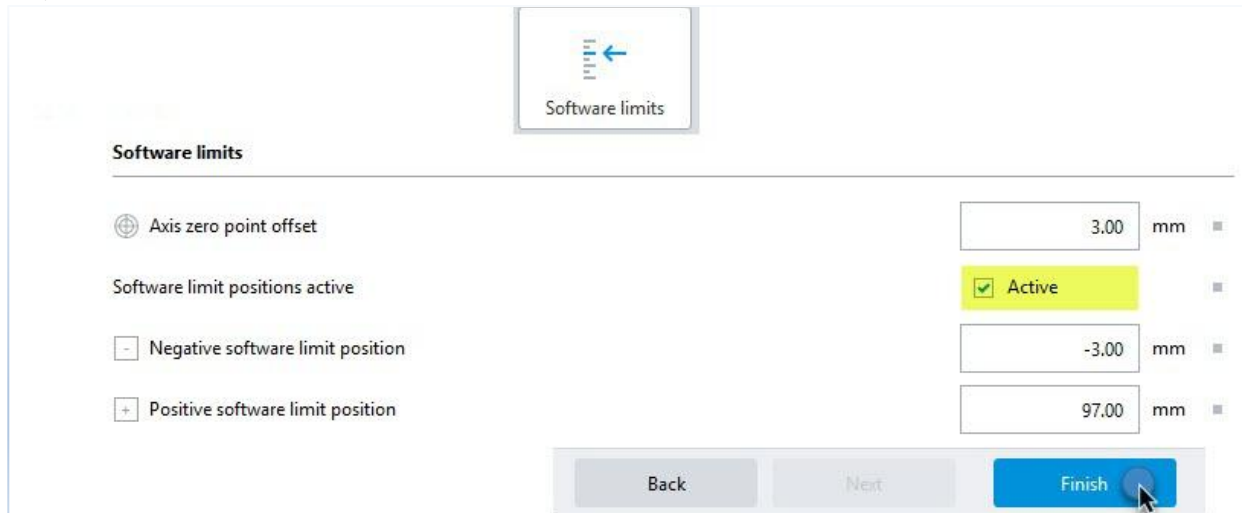
Method: Current position (37)

Nominal current limit value scaling factor: 0.30

Move to axis zero point after homing: ☒ Active

Back Next Finish

Select homing method. As per this sample application “**Current Position**” mode is selected. But user want to select homing mode depending on application requirement. Click next button for next level of configuration.

**Step – 10**

**Software limits**

Axis zero point offset 3.00 mm

Software limit positions active ☒ Active

Negative software limit position -3.00 mm

Positive software limit position 97.00 mm

Back Next Finish

Here user can enable the maximum & minimum values of software limits as per application requirements. Finally click **“Finish”** button to complete the configuration.

**Step – 11**

At this stage all the fundamental configurations of drive system are done. In case of any correction in parameter is required software tool will indicate number of **“correction parameters”** ( Right most top corner in FAS tool ) as shown in the below picture. So, to correct parameter values, click the **“Correction parameters”** button and click the apply button to apply system generated recommended values.



### 2.4.2 Rotary Axis Configuration

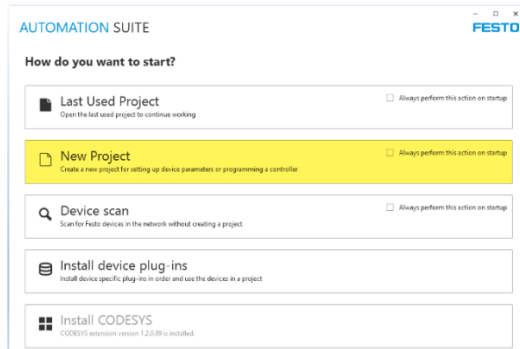
This chapter describe about parameterization of CMMT drive for rotary axis to move 1000 revolution forward and reverse with 1000 RPM base speed & current position as homing mode.



#### Note

- If user looking for Linear Axis configuration skip this chapter and jump to [chapter – 2.4.1](#).

#### Step – 1



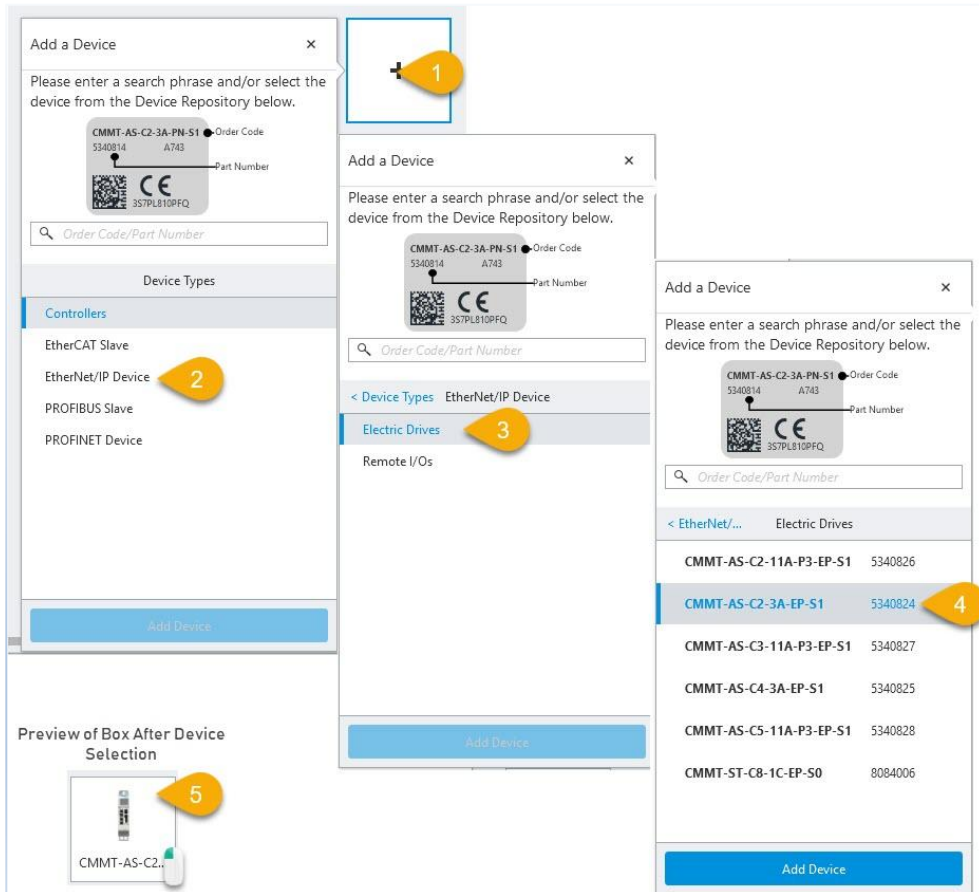
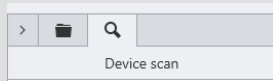
Open the Automation Suite software from Windows **Start > Festo Software > AUTOMATION SUIT** and select **“New Project”** from start-up menu.

## Step – 2

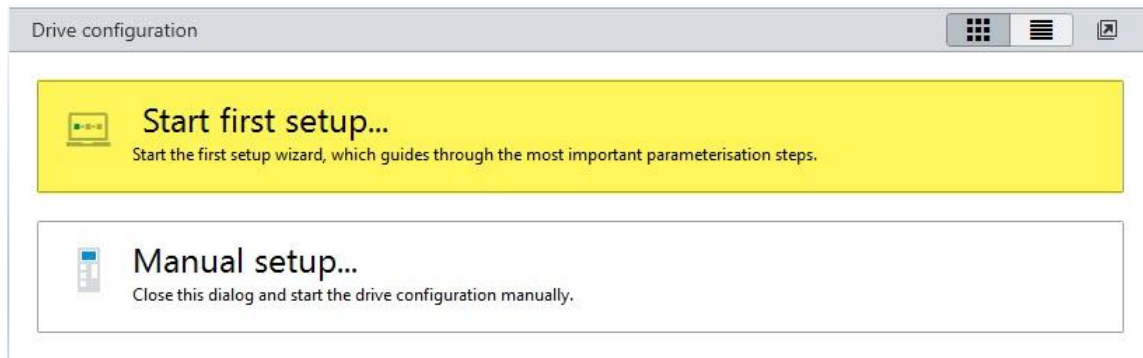


### Note

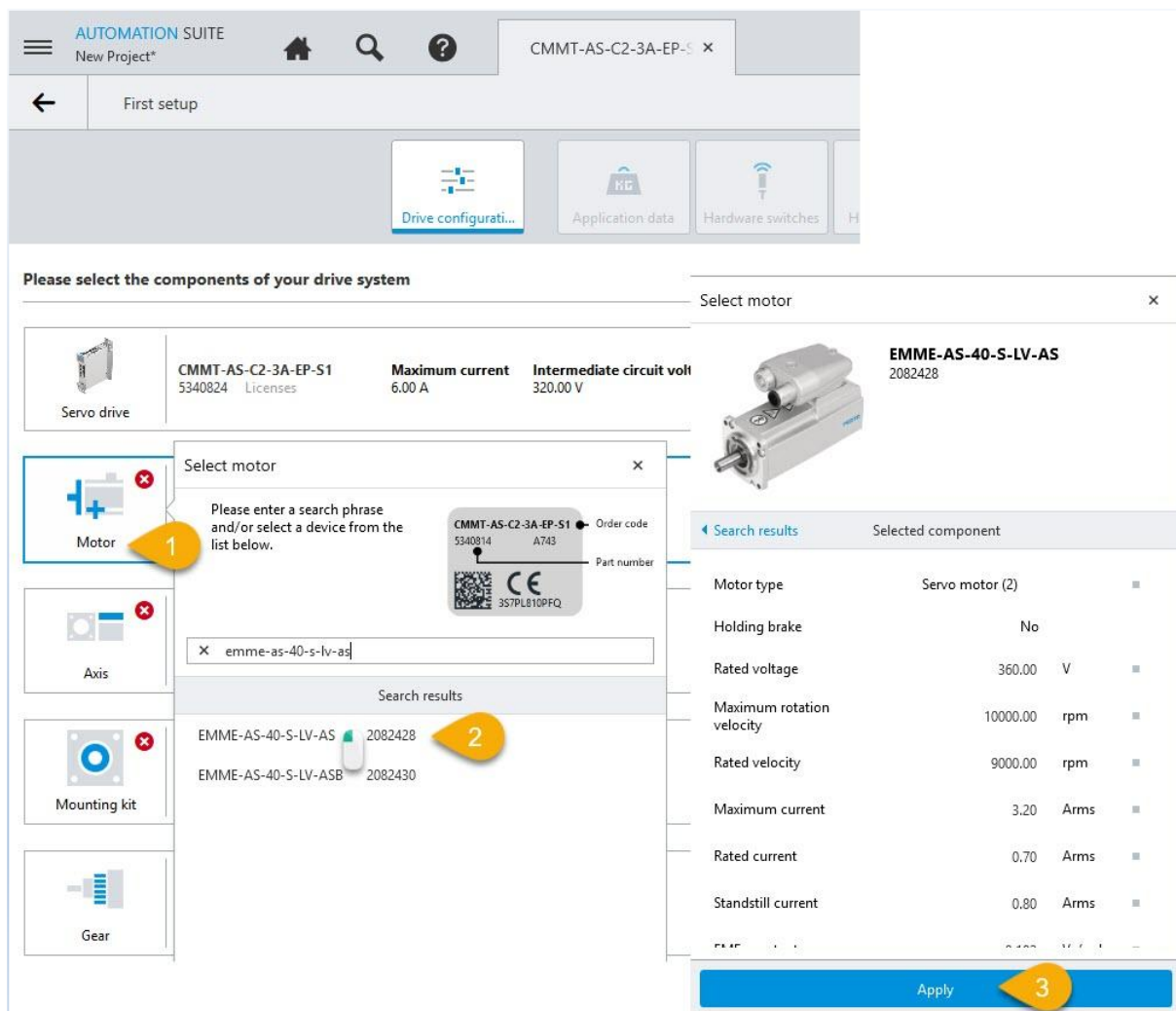
- Follow this step if drive is not in online. If the drive is connected with the PC as shown in [chapter 1.3](#) simply click Device scan from home ( Next to device Repository ) and online devices will be visible in available devices list. Select the correct device based on Its configured name & IP address then double click.



1. Select any empty device box and click add button.
2. Select “**EtherNet/IP Device**” from Add a Device menu.
3. Select “**Electric Drives**” from device type menu.
4. **Select correct drive part number** from electric drives list. As per this sample application CMMT-AS-C2-3A-EP-S1 is selected.
5. Double click the newly added drive to initialize configuration parameters.

**Step – 3**

Select “**Start first setup**” from drive configuration menu.

**Step – 4**

1. Click “**Motor**” from components of drive system menu for specific motor model.
2. **Select correct motor** based on its **part number** and double click. As per this sample application “EMME-AS-40-S-LV-AS” motor is used.
3. Cross verify the motor parameters & click apply button to ensure motor configuration.

View of motor tab after configuration

|                                                                                            |                               |                         |                     |                                   |                                 |                     |                                                                                                                                                                            |
|--------------------------------------------------------------------------------------------|-------------------------------|-------------------------|---------------------|-----------------------------------|---------------------------------|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <br>Motor | EMME-AS-40-S-LV-AS<br>2082428 | Type<br>Servo motor (2) | Holding brake<br>No | Encoder protocol<br>HiPerface (0) | Encoder type<br>Single turn (1) | Voltage<br>360.00 V | <br> |
|--------------------------------------------------------------------------------------------|-------------------------------|-------------------------|---------------------|-----------------------------------|---------------------------------|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Step – 5

Please select the components of your drive system

  
Servo drive

CMMT-AS-C2-3A-EP-S1  
5340824 Licenses

Maximum current  
6.00 A

Intermediate circuit voltage  
320.00 V

  
Motor

Select axis

Please enter a search phrase and/or select a device from the list below.



Order code  
5340814 A743

Part number  
357PL810PFQ

Order code/part number

Search results

User defined linear axis

User defined rotative axis

DGE-8-%%-ZR 193739

DGE-8-%%-ZR-HD12-GK 193739

DGE-8-%%-ZR-HD8-GK 193739

  
Axis

  
Mounting kit

  
Gear

User defined rotative axis



Search results

Selected component

Current user unit  
Rev [rev, rpm, ...] (3)

Motion  
Unlimited axis

Position range  
1000 r

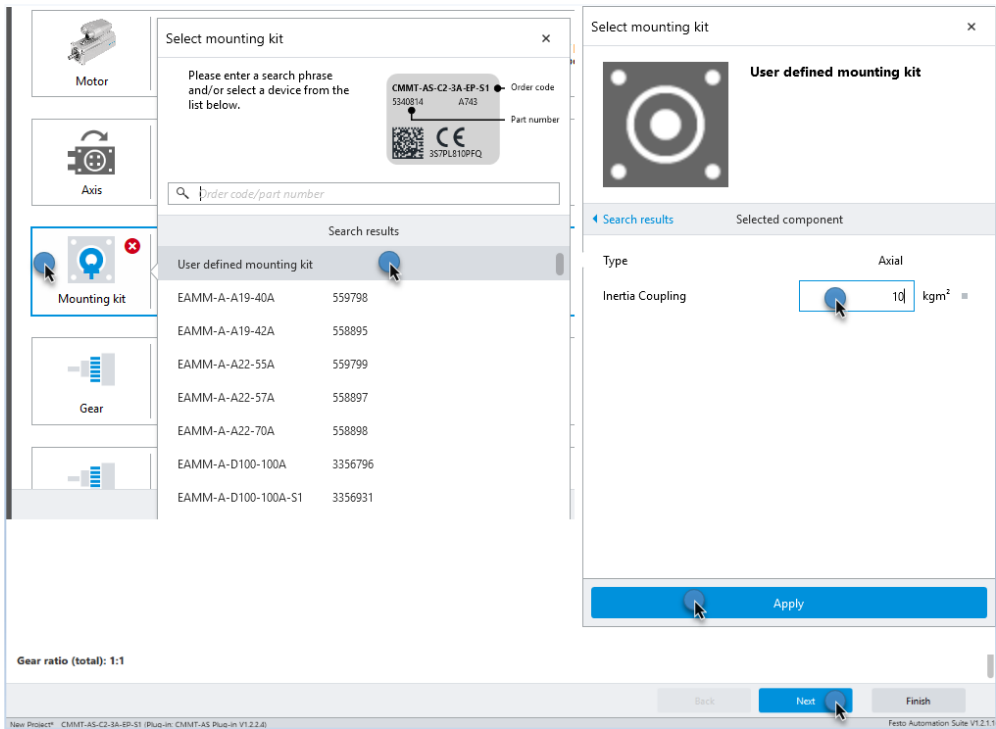
Design axis  
Single axis (0)

Rotative  
☐ Active

Apply

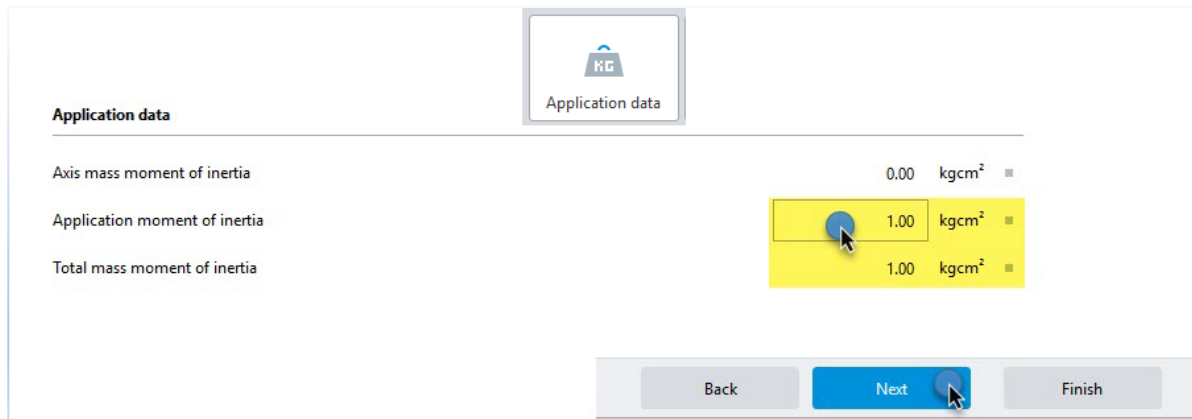
This step describes about the axis type configuration. As specified initially here axis configuration is done for 1000 rotation in forward & reverse. This configuration will change based on user application type.

**Step – 6**



This step illustrates about mounting kit configuration. This configuration will vary depending on user application type. At the end of this step click next button to move to next level configuration.

### Step – 7



**Application data**

Axis mass moment of inertia 0.00 kgcm<sup>2</sup>

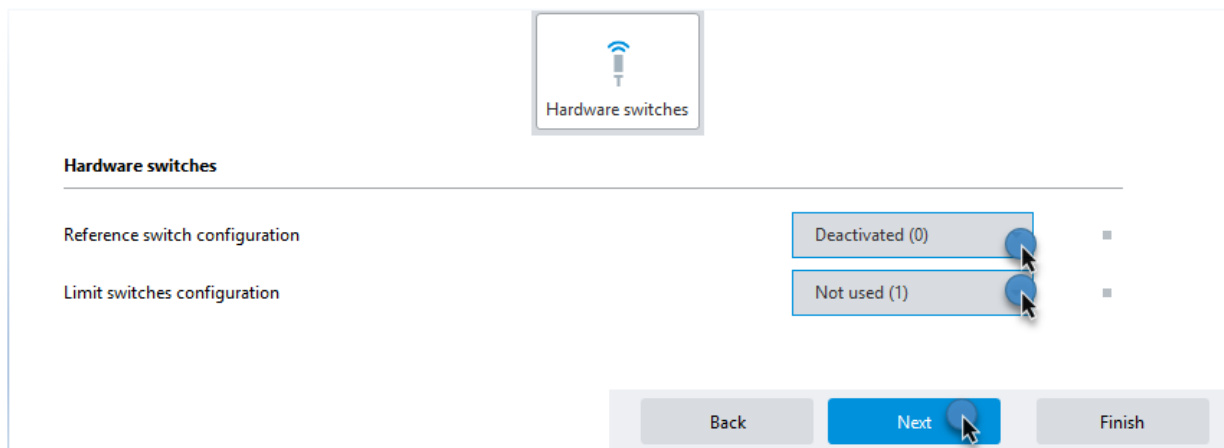
Application moment of inertia 1.00 kgcm<sup>2</sup>

Total mass moment of inertia 1.00 kgcm<sup>2</sup>

Back Next Finish

Specify inertia value of user application. This parameter value will vary depend on application type. Click next button for next level configuration.

### Step -8



**Hardware switches**

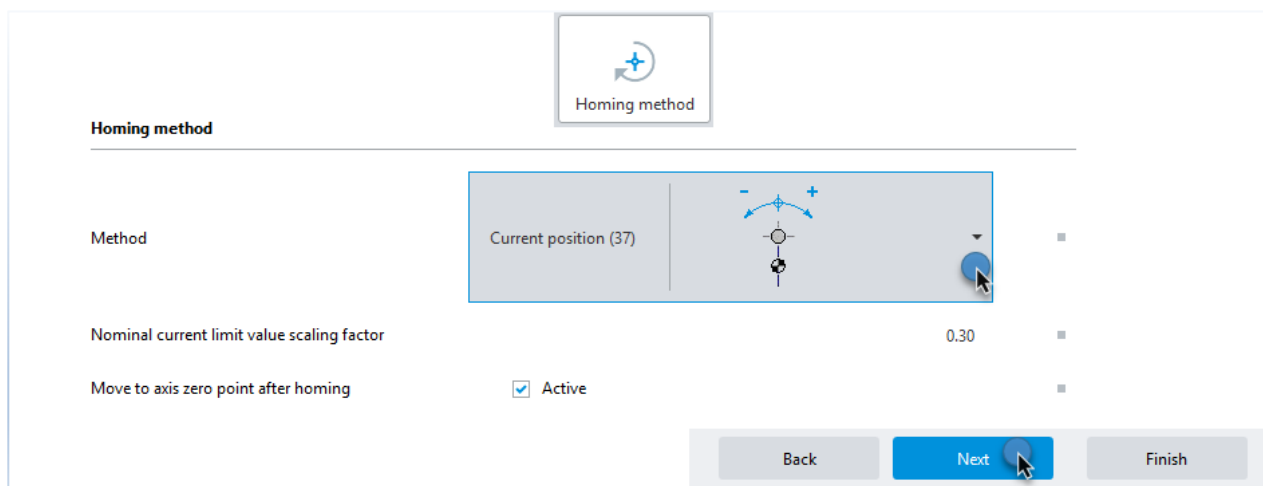
Reference switch configuration Deactivated (0)

Limit switches configuration Not used (1)

Back Next Finish

Select the homing & maximum limit switch connection type. Click next button for next level configuration.

### Step – 9



**Homing method**

Method Current position (37)

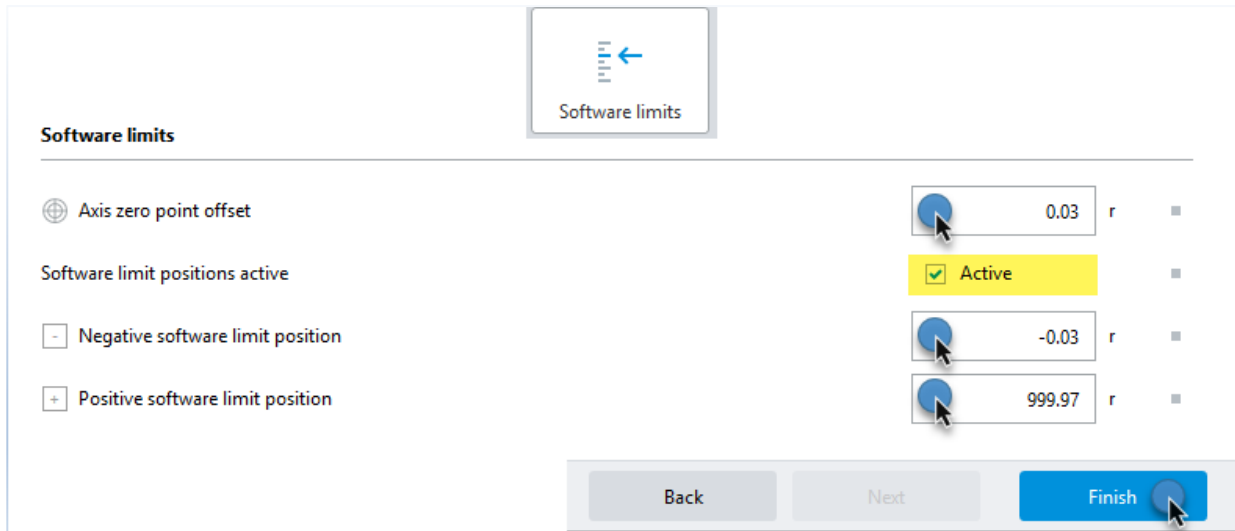
Nominal current limit value scaling factor 0.30

Move to axis zero point after homing ☒ Active

Back Next Finish

Select homing method. As per this sample application “**Current Position**” mode is selected. But user want to select homing mode depending on application requirement. Click next button for next level of configuration.

### Step – 10



**Software limits**

Axis zero point offset: 0.03 r

Software limit positions active: ☒ Active

Negative software limit position: -0.03 r

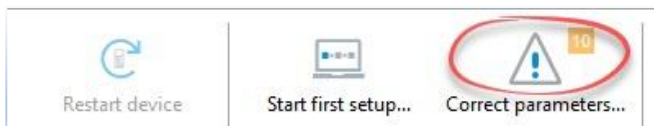
Positive software limit position: 999.97 r

Back Next Finish

Here user can enable the maximum & minimum values of software limits as per application requirements. Finally click “**Finish**” button to complete the configuration.

### Step – 11

At this stage all the fundamental configurations of drive system are done. If parameter corrections are required, the software tool will indicate the number of “**correction parameters**” ( Right most top corner in FAS tool ) as shown in the below picture. So, to correct parameter values, click the “**Correction parameters**” button and click the apply button to apply system generated recommended values.



## 2.5 CMMT Drive Parameterisation for Integrate with PLC in Position Control Mode

This chapter will describe about parameterisation of CMMT drive to establish communication with PLC through Telegram 111. Also, it describes some additional parameters like Factor group, Reference values, EtherNet/IP interface, User defined limits.

### Step – 1



#### Note

- Follow **step – 1** when drive is available for online connection. If drive is not available simply skip this step and jump to next step.

The screenshot displays the software interface for CMMT drive parameterisation. The main window has three tabs: **PARAMETERISATION**, **CONTROL**, and **DIA**. The **PARAMETERISATION** tab is active, showing the drive status as **Disconnected** with the path **192.150.3.7**. A yellow callout '1' points to the 'Path' field. A **Connect** button is visible. Below this, the **Status When Drive is in On-line** section shows the drive as **Connected** with the path **192.168.1.7** and a **Disconnect** button. A yellow callout '2' points to the **Use IP Address for Device Communication** button at the bottom right. A **Device Communication** dialog box is open, prompting for the device's IP address or selection from a list. The IP address **192.168.1.7** is entered in the text field. The **Available Devices** table lists the following:

| Status | Identify                 | Device Name | Device Type | IP Address  |
|--------|--------------------------|-------------|-------------|-------------|
| ⊗      | <input type="checkbox"/> | CMMT1       | CMMT-AS     | 192.168.1.7 |

- Click the IP address **“Path”** from tool bar.
- Select the correct drive based on IP address & select **“Use IP Address for Device Communication”** button to assign path address.
- Click on **“Connect”** button to establish communication & status of the drive changes from Disconnected to Connected.

**Step – 2**

| Factor group      |                                                                                   |                                   |
|-------------------|-----------------------------------------------------------------------------------|-----------------------------------|
| Current user unit |                                                                                   | Metric [m, m/s, ...] (6) ■        |
| Position          |  | <input type="text" value="-6"/> ■ |
| Velocity          |  | <input type="text" value="-3"/> ■ |

Enter the Position & Velocity command scale factor value of fieldbus. User can reach the Factor group in Automation Suite from **PARAMETERISATION > Fieldbus > Factor group**.

Example - 1 PLC scale value is calculated below for move the linear axis 10 mm in 1 m/s speed.

Position Factor (-6) =  $10^{-6} = 0.000001 = 1\mu\text{m}$  Then For 10 mm = 1000000 Increment value in PLC

Velocity Factor (-3) =  $10^{-3} = 0.001 = 1\text{mm/s}$  Then for 1 m/s = 1000 Increment value in PLC

Example - 2 PLC scale value is calculated below for move the rotary axis 5 Rev in 100 RPM speed.

Position Factor (-6) =  $10^{-6} = 0.000001 \text{ Rev}$  Then For 5 Rev = 5000000 Increment value in PLC

Velocity Factor (-3) =  $10^{-3} = 0.001 \text{ RPM}$  Then for 100 RPM = 100000 Increment value in PLC

**Note**

- For more details about fieldbus scaling refer chapter name “**Scaling of internal units for the fieldbus**” in user manual.
- Different user unit can't be set for Position, Velocity, Acceleration & Jerk.
- The maximum value limit for 32 bit DINT data type variable is 2,147,483,647. So, plan scaling factor within the limit.

**Step – 3**

| Reference values              |                                                                                   |                                      |                    |
|-------------------------------|-----------------------------------------------------------------------------------|--------------------------------------|--------------------|
| Base value speed              |  | <input type="text" value="0.50"/>    | m/s ■              |
| Base value speed (controller) |                                                                                   | <input type="text" value="3000.00"/> | rpm ■              |
| Base value acceleration       |                                                                                   | <input type="text" value="1.00"/>    | m/s <sup>2</sup> ■ |
| Base value deceleration       |                                                                                   | <input type="text" value="1.00"/>    | m/s <sup>2</sup> ■ |

Enter the actual Base speed value, Acceleration & Deceleration values of the application in the respective numeric fields.

**Note**

- Base speed value does not affect the speed of the drive, but it will effect on display value of the drive.
- The AOI tag “**BaseSpeedValue**” tag must have same value as the FAS parameter value that is (FAS = P1.11280701.0.0).

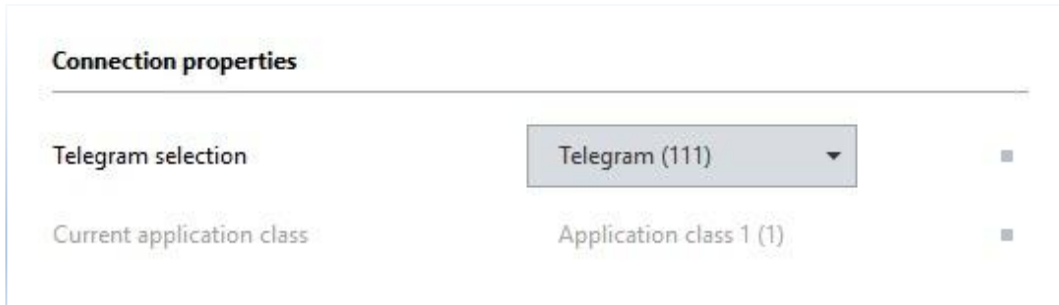
**Step – 4**

| EtherNet/IP interface |                                                  |   |
|-----------------------|--------------------------------------------------|---|
| <b>Configuration</b>  |                                                  |   |
| Activate DHCP         | <input type="checkbox"/> Active                  | ■ |
| IP address            | <input type="text" value="192 . 168 . 1 . 5"/>   | ■ |
| Subnet mask           | <input type="text" value="255 . 255 . 255 . 0"/> | ■ |
| Gateway address       | <input type="text" value="0 . 0 . 0 . 0"/>       | ■ |

Enter the IP address & Subnet mask for EtherNet/IP interface. This communication interface is used to establish connection with PLC to receive control command & to send signal feedbacks.

**Note**

- IP address series of EtherNet/IP interface must match with PLC IP address series to establish connection.

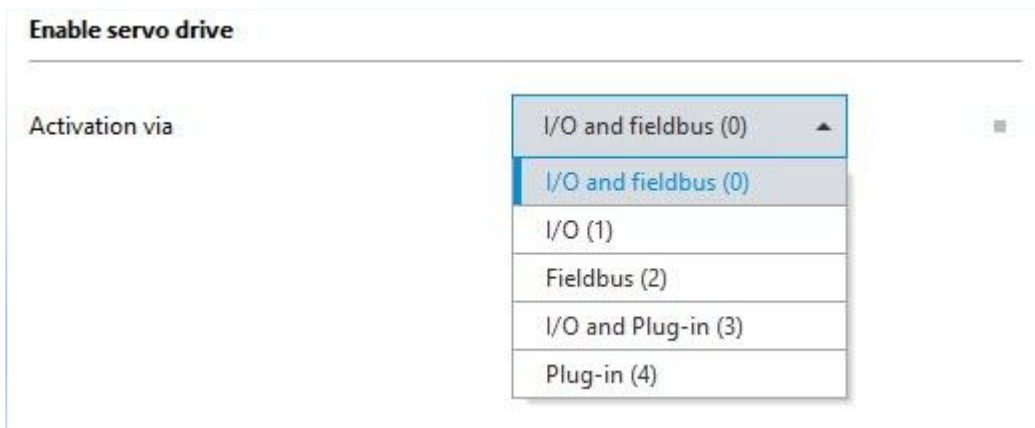
**Step – 5**


**Connection properties**

Telegram selection: Telegram (111)

Current application class: Application class 1 (1)

Select the connection properties as “**Telegram(111)**” from **PARAMETERISATION > Fieldbus > Connection properties**.

**Step – 6**


**Enable servo drive**

Activation via: I/O and fieldbus (0)

I/O and fieldbus (0)

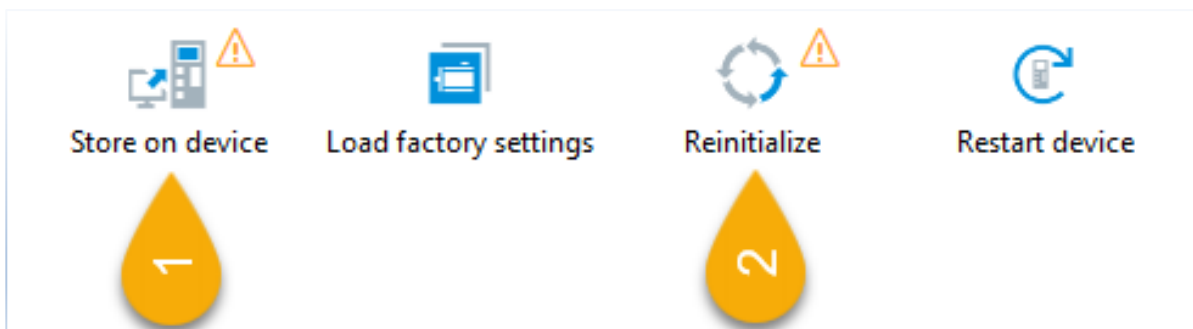
I/O (1)

Fieldbus (2)

I/O and Plug-in (3)

Plug-in (4)

Select the medium to enable the servo drive. The selection must be either “**I/O and fieldbus (0)**” or “**Fieldbus**” to control the drive through fieldbus.

**Step – 7**


Store on device

Load factory settings

Reinitialize

Restart device

1

2

1. Click the “**Store on device**” button to store all the parameters on drive memory.
2. Click the “**Reinitialize**” the button to initialize the drive.



**Note**

- The triangle warning symbol on both buttons indicates action required.
- All the modified parameters will automatically erase from drive during power restart If modified parameters are not stored in the drive.
- Its mandatory to reinitialize the drive for some parameter modification. If drive is not reinitialized the drive performance remains same as like previous configuration.

### 3 Software Configurations for CMMT Position Control Mode

This chapter describe about software configurations like AOI import in Studio 5000, CMMT EDS file import, Adding Generic Ethernet Module, Adding PTP\_DRIVES\_FESTO\_EIP AOI (Add-On Instruction) in program & IN OUT Tags configuration of AOI.

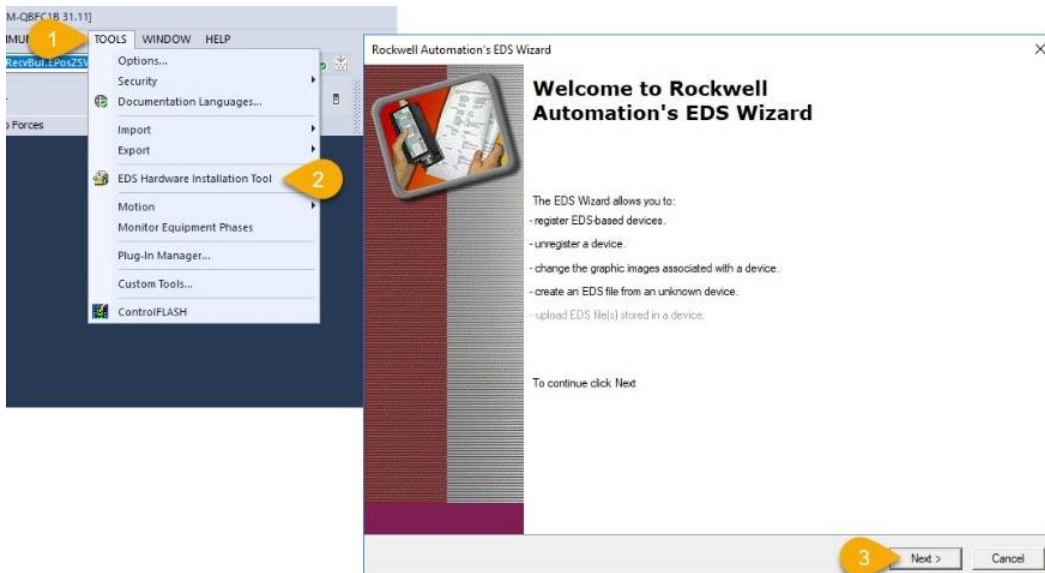


#### Note

- CMMT servo drive EDS file is available with this application note folder & Also user can able to download it from Festo Support Portal website.
- Skip [Chapter - 3.1](#) & [Chapter - 3.2](#) if the Generic Ethernet module is used as hardware interface module instead of EDS module and proceed to [Chapter - 3.3](#)

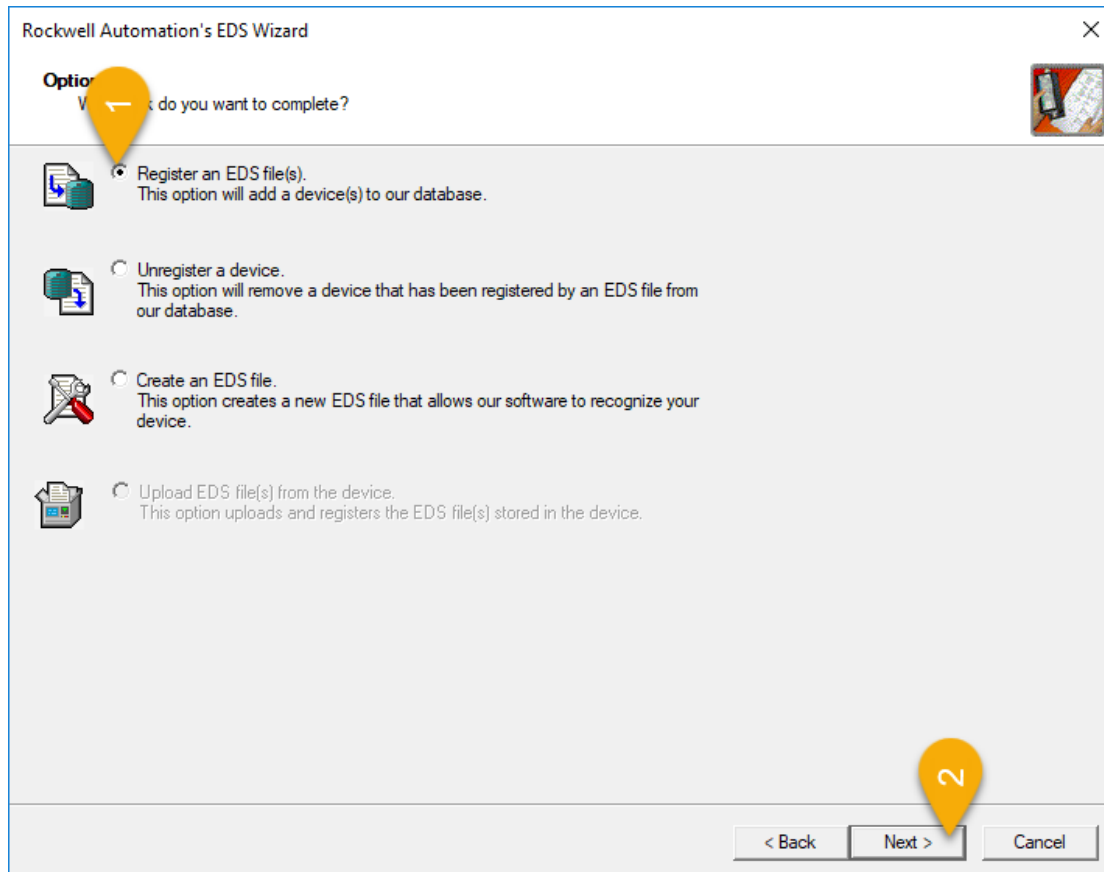
#### 3.1 Install CMMT Drive EDS File in Studio-5000

##### Step – 1

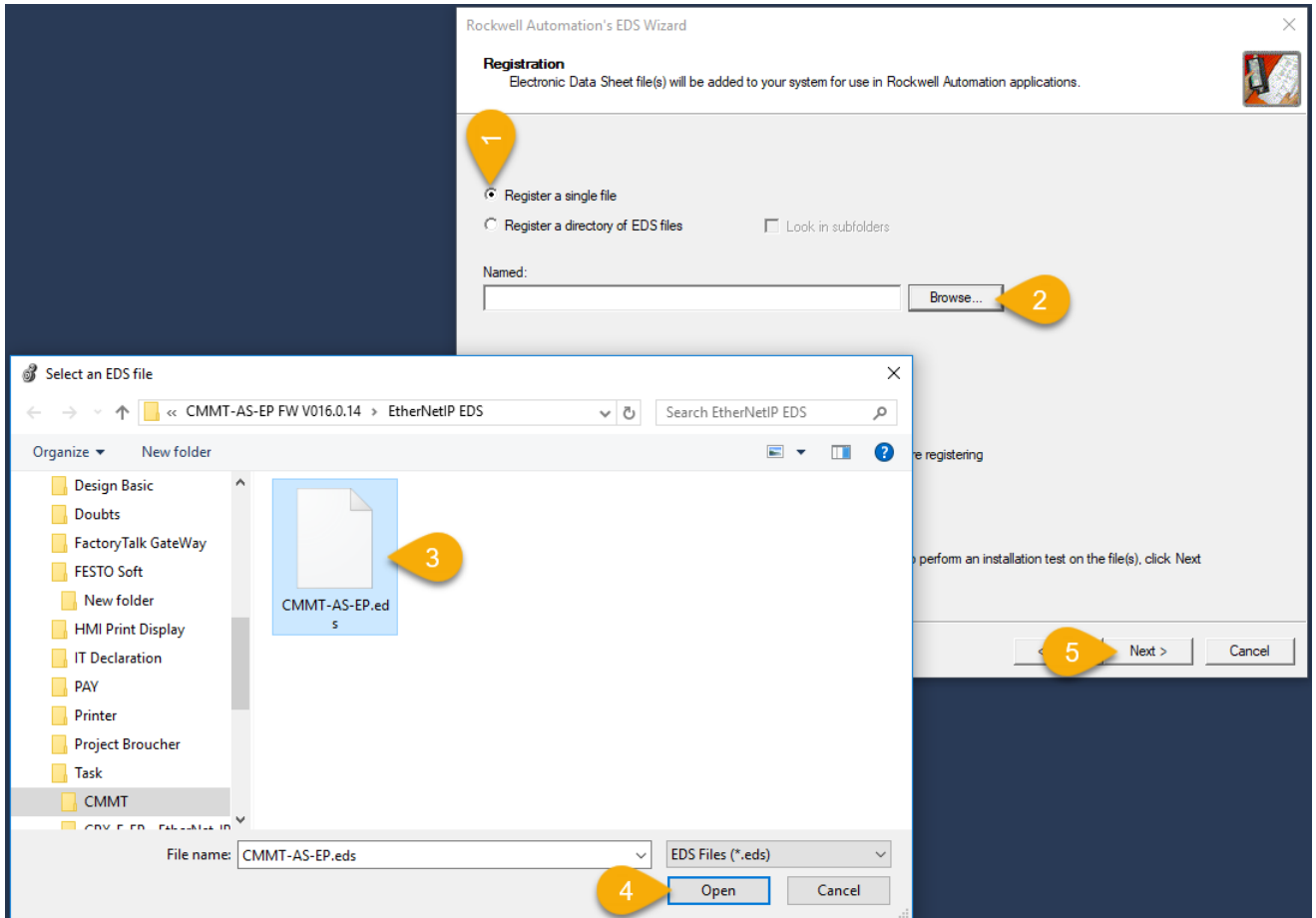


1. Click the **“Tools”** from tool bar.
2. Select **“EDS Hardware Installation Tool”** from tools drop-down menu.
3. Click **“Next”** from EDS installation wizard.

## Step – 2



1. Select “**Register an EDS file(s)**” from the listed options.
2. Select “**Next**” for next level.

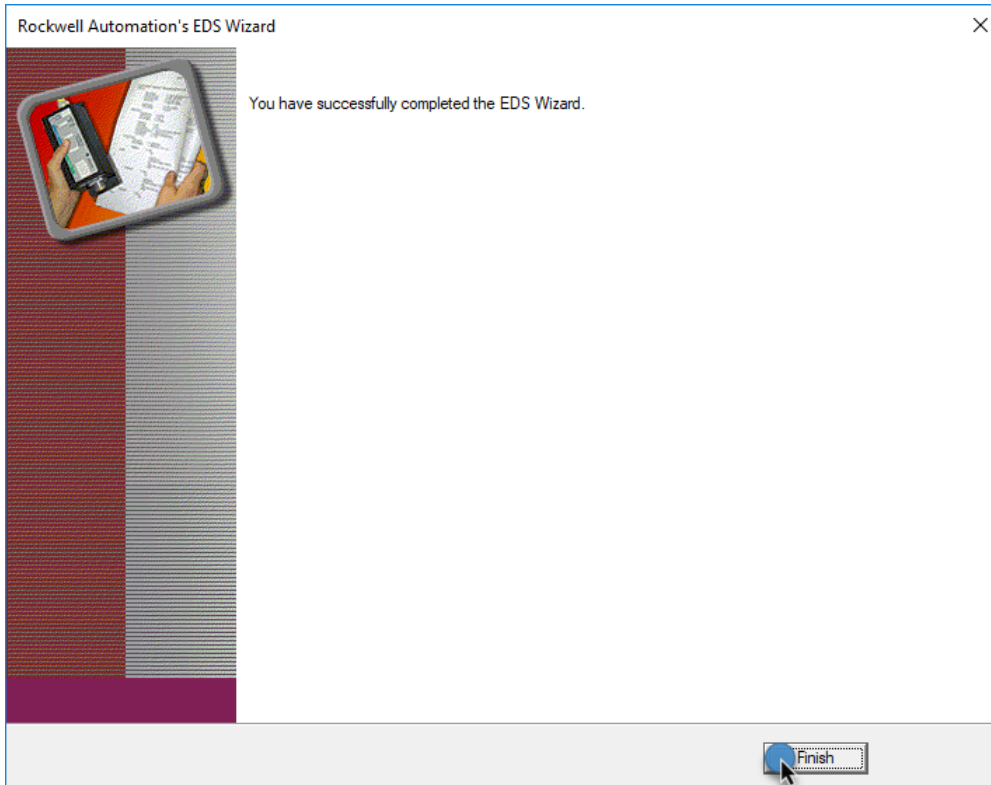
**Step – 3**

1. Select **“Register a single file”** from registration window.
2. Click **“Browse”** button to select EDS backup file.
3. Navigate to the EDS backup file saved location and select it.
4. Click **“Open”** button form EDS file selection window.
5. Click **“Next”** button from Registration window.

**Note**

- Simply click next button for upcoming configuration wizards and move to final window as shown in next step.

**Step – 4**



Click Next button to complete the installation of EDS file installation.

### 3.2 Add CMMT EDS File in Studio-5000 Ethernet Tree for Hardware Interface

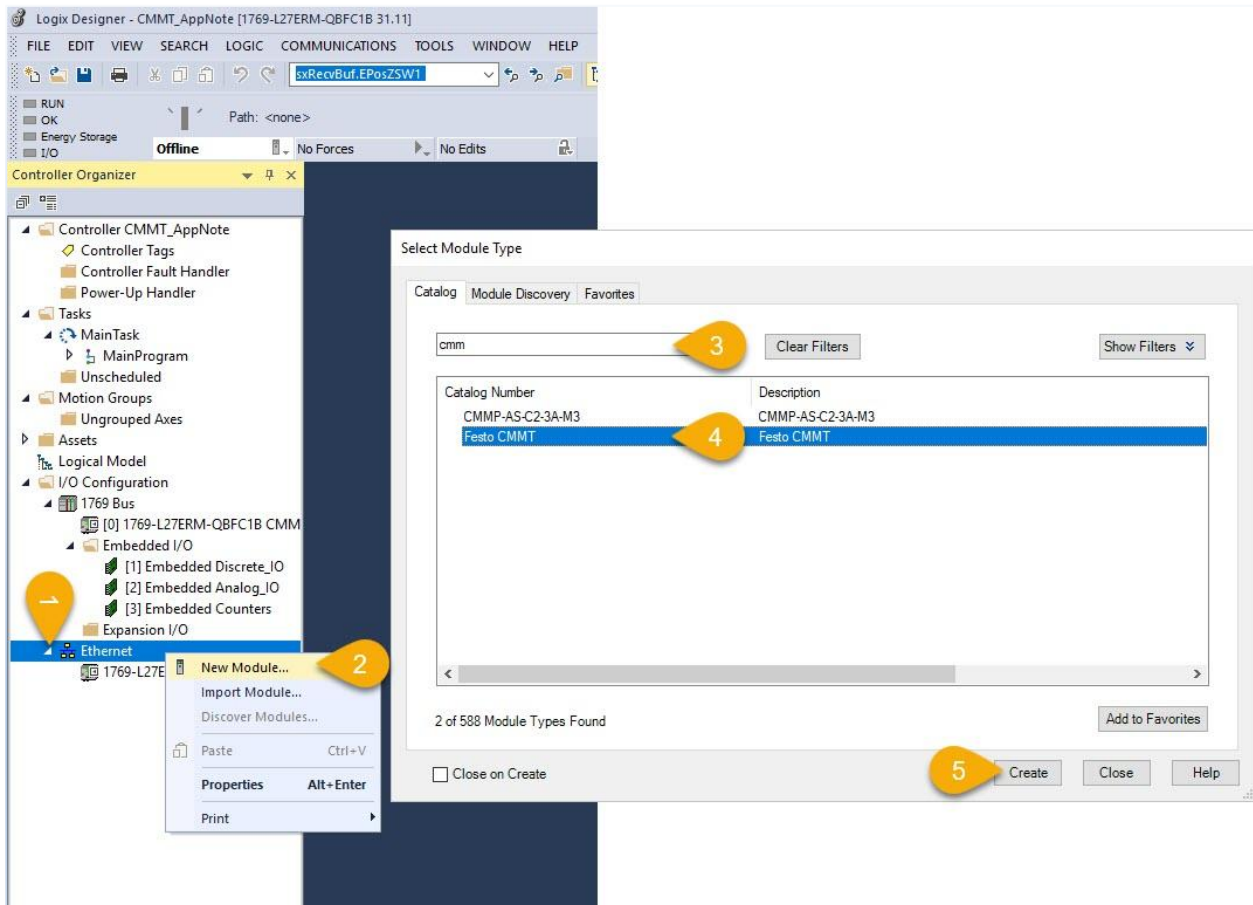
This chapter will describe about adding CMMT drive communication module in Studio - 5000 tool.



#### Note

- Skip this chapter when Generic Ethernet Module used as Hardware interface module and proceed to [Chapter 3.3](#)
- Refer [Chapter – 10.4](#) for configure Extended Process Data in EDS interface file.

#### Step – 1

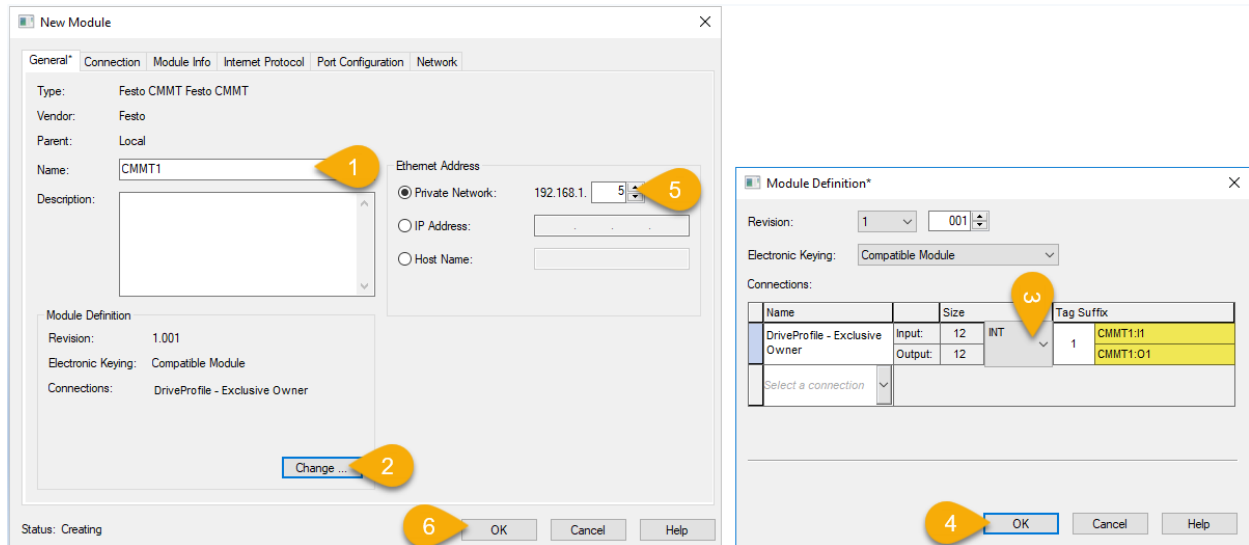


1. Select “**Ethernet**” folder from **controller organizer menu > I/O Configuration** and right click.
2. Select “**New Module**” from menu.
3. Type as **CMMT** in search tab.
4. Select “**Festo CMMT**” from search result.
5. Click “**Create**” to create CMMT interface module.



#### Note

- When using Generic Ethernet module instead of EDS module type as Generic in search tab & select “**ETHERNET-MODULE**” from search result.

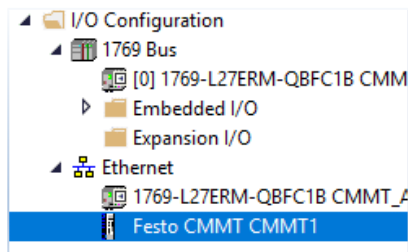
**Step – 2**

1. Enter the name of the module.
2. Click the **“Change”** button from module definition frame.
3. Select **“INT”** data type from module definition window.
4. Select **“OK”** button to confirm the configuration.
5. Enter the IP address of the drive ( **Address of Interface Port – XF1** ).
6. Click **“OK”** button to confirm the configuration.

**Note**

- Select **“Private Network”** option of ethernet address when IP series starts as **192.168.1.xx** & for different IP series select **“IP Address”** option.
- Names of the Input & Output tags will be displayed in controller tags list as highlighted in yellow color.

After completion of EDS file configuration the new module will be available in ethernet tree as shown below picture.

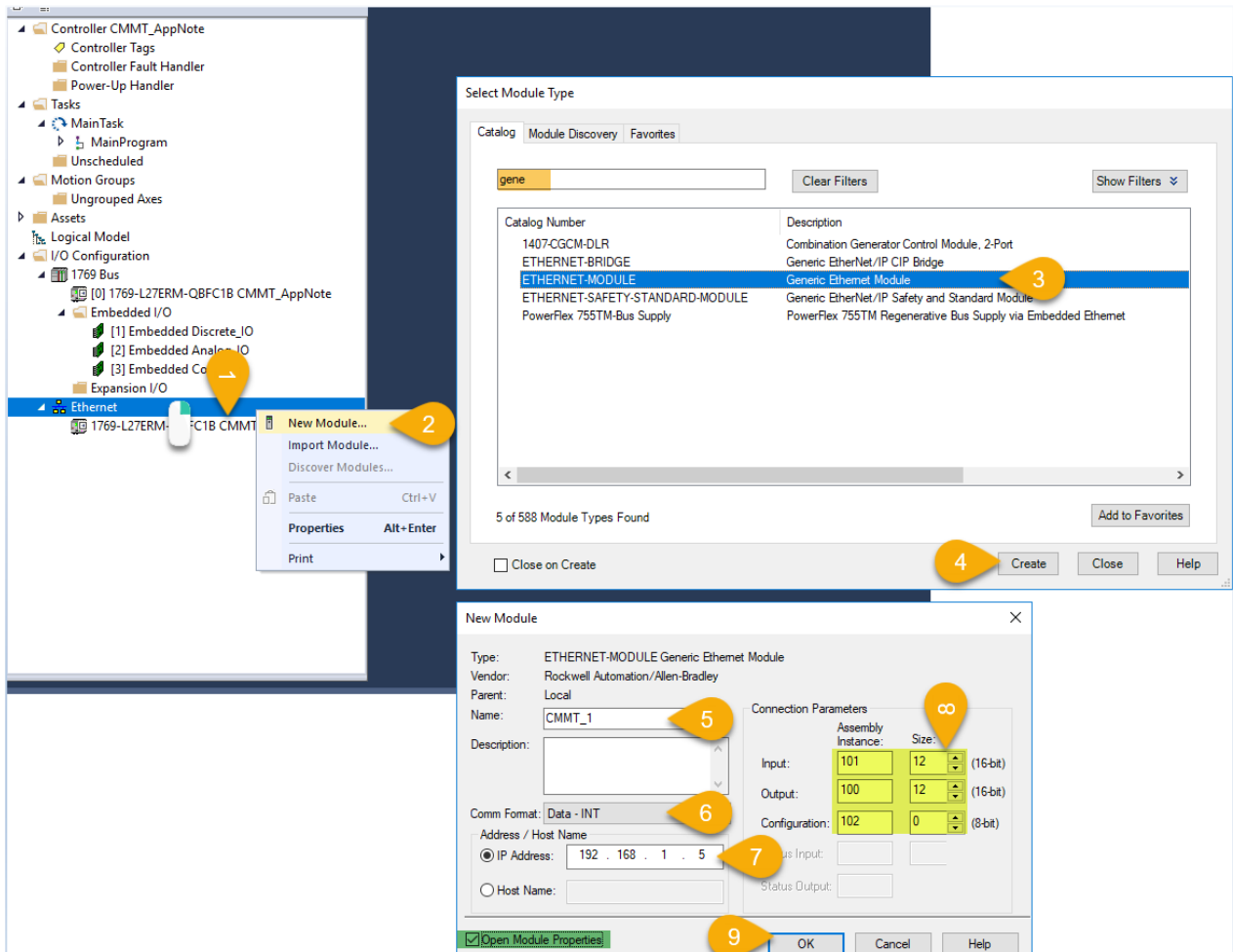


### 3.3 Add Generic EtherNet Module in Studio-5000 EtherNet Tree for Hardware Interface



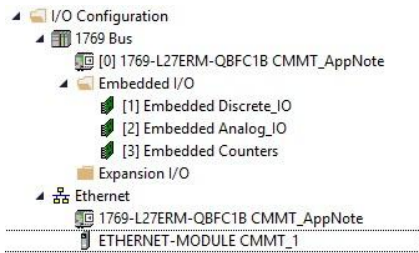
#### Note

- Skip this chapter when EDS used as Hardware interface module and proceed to [Chapter 3.2](#)
- Refer [Chapter - 10.3](#) for Extended Process Data configuration details in Generic Module.



1. Select the **"Ethernet"** tree from controller organizer window
2. Select **"New Module"** option from right click menu.
3. Type the module name in search box as highlighted in orange colour & select the **"ETHERNET-MODULE"**.
4. Click **"Create"** button from select module type window.
5. Enter the name of the module in **"Name"** filed in New Module window.
6. Select the data format as **"Data – INT"**.
7. Enter the **"IP address"** of CMMT drive (**Address of Interface Port – XF1**).
8. Enter the **Assembly Instance & Size values** as highlighted in yellow colour.
9. Click **"OK"** to confirm the configuration. Once the OK button is click, Module property window will open automatically because "Open Module Properties" is selected default as highlighted in green colour.

After completion of Generic ethernet module configuration the new module will be available in ethernet tree as shown below picture.



### 3.4 Import PTP\_DRIVES\_FESTO\_EIP AOI in Studio – 5000

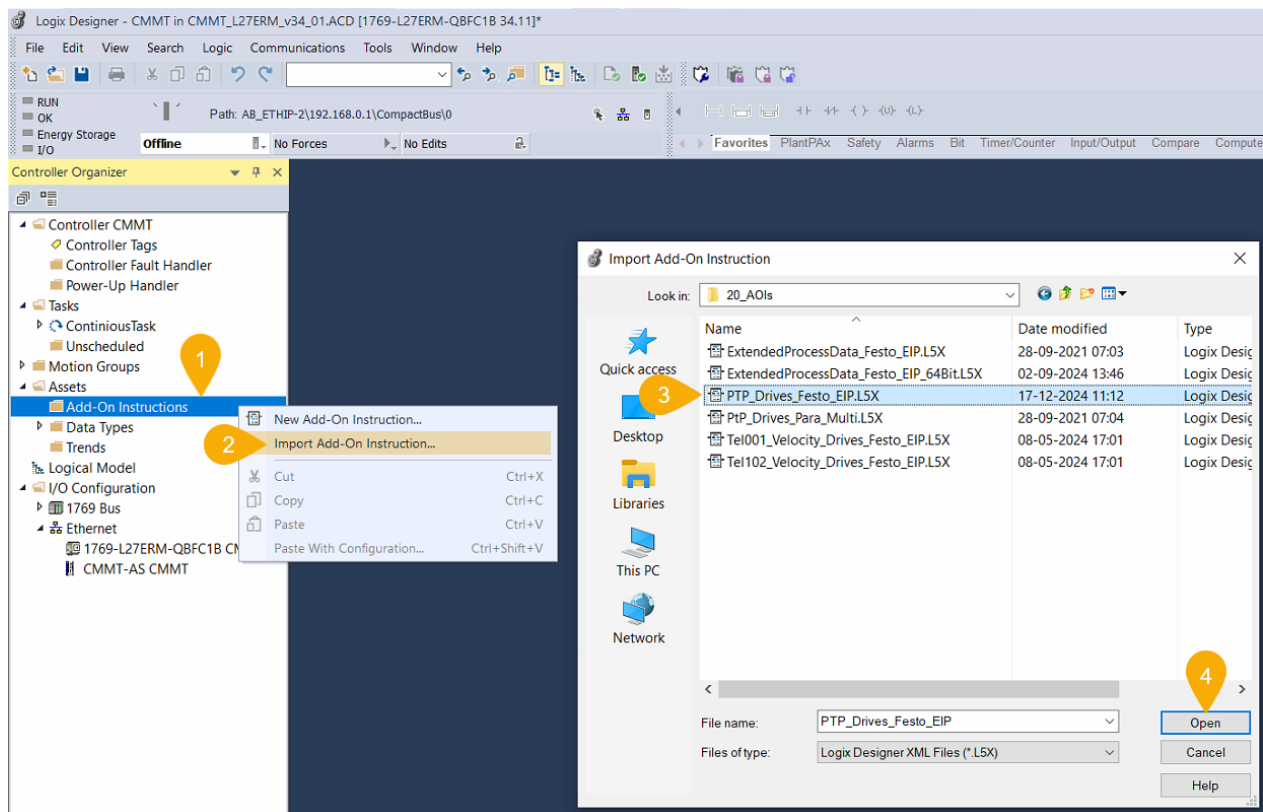
This chapter will describe about importing the PTP\_DRIVES\_FESTO\_EIP AOI file inside the PLC program.



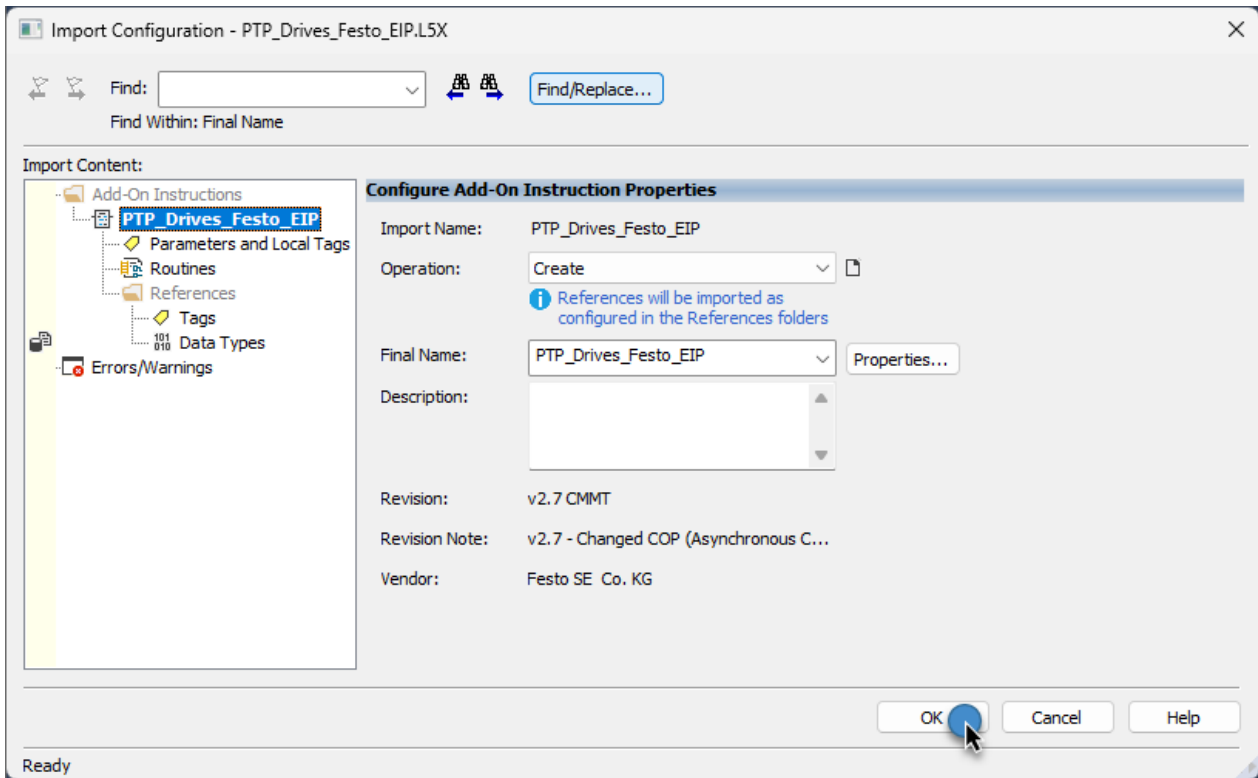
#### Note

- The user can find this AOI file in same folder of this document. The user can also download from FESTO Support portal.

#### Step – 1

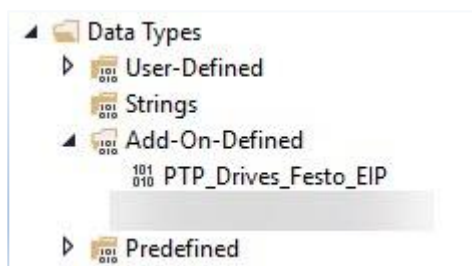


1. Select the **“Add-On Instructions”** folder from **controller organizer > Assets** and right click.
2. Select **“Import Add-On Instruction”** from right click menu.
3. Browse to the **PTP\_DRIVES\_FESTO\_EIP.L5X** export file saved location in workstation and select it.
4. Click **“Open”** button in import add-on instruction window.

**Step – 2**

Click **“OK”** button in import configuration window.

After successful import of AOI new PTP\_DRIVES\_FESTO\_EIP will be displayed in Add-On Instructions folder as shown in below picture.



### 3.5 Add PTP\_DRIVES\_FESTO\_EIP AOI in PLC Program



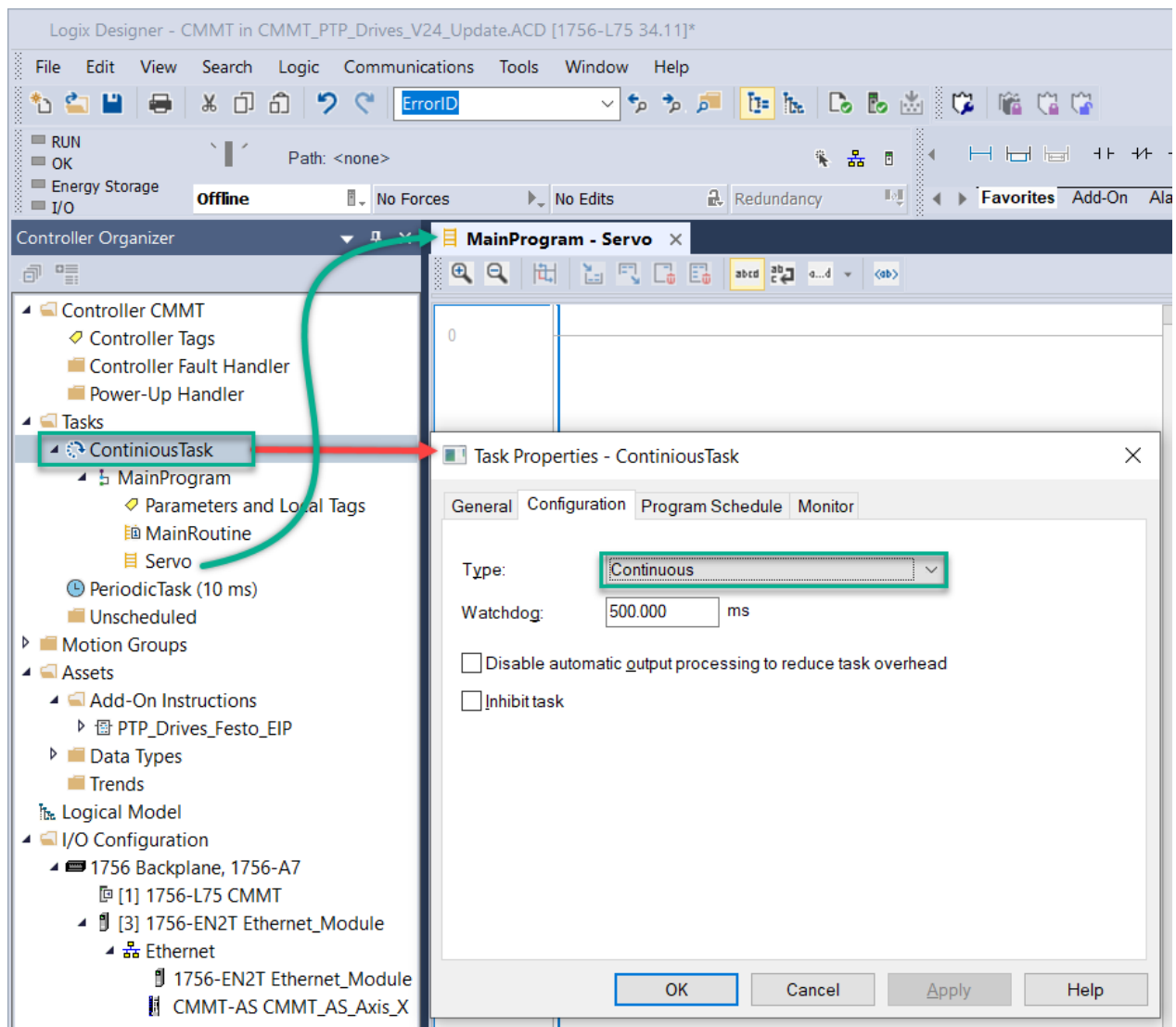
#### Note

- The Add-On Instruction must be called cyclically with a separate instance for every servo drive integrated in the user program (every axis). Simultaneous use of multiple instances of the same device is not permitted.
- Any AOI's need to be in a continuous task called cyclically. ([Refer Chapter 3.5 Step – 1](#))

This chapter describes about adding AOI in program & assigning input & output tags.

#### Step – 1

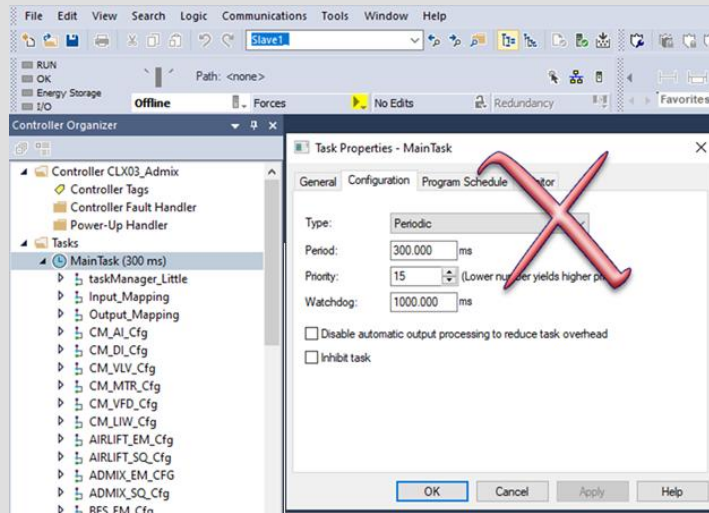
The CORRECT WAY is to have the AOI (Add-On Instruction) programmed within a continuous task like shown here:



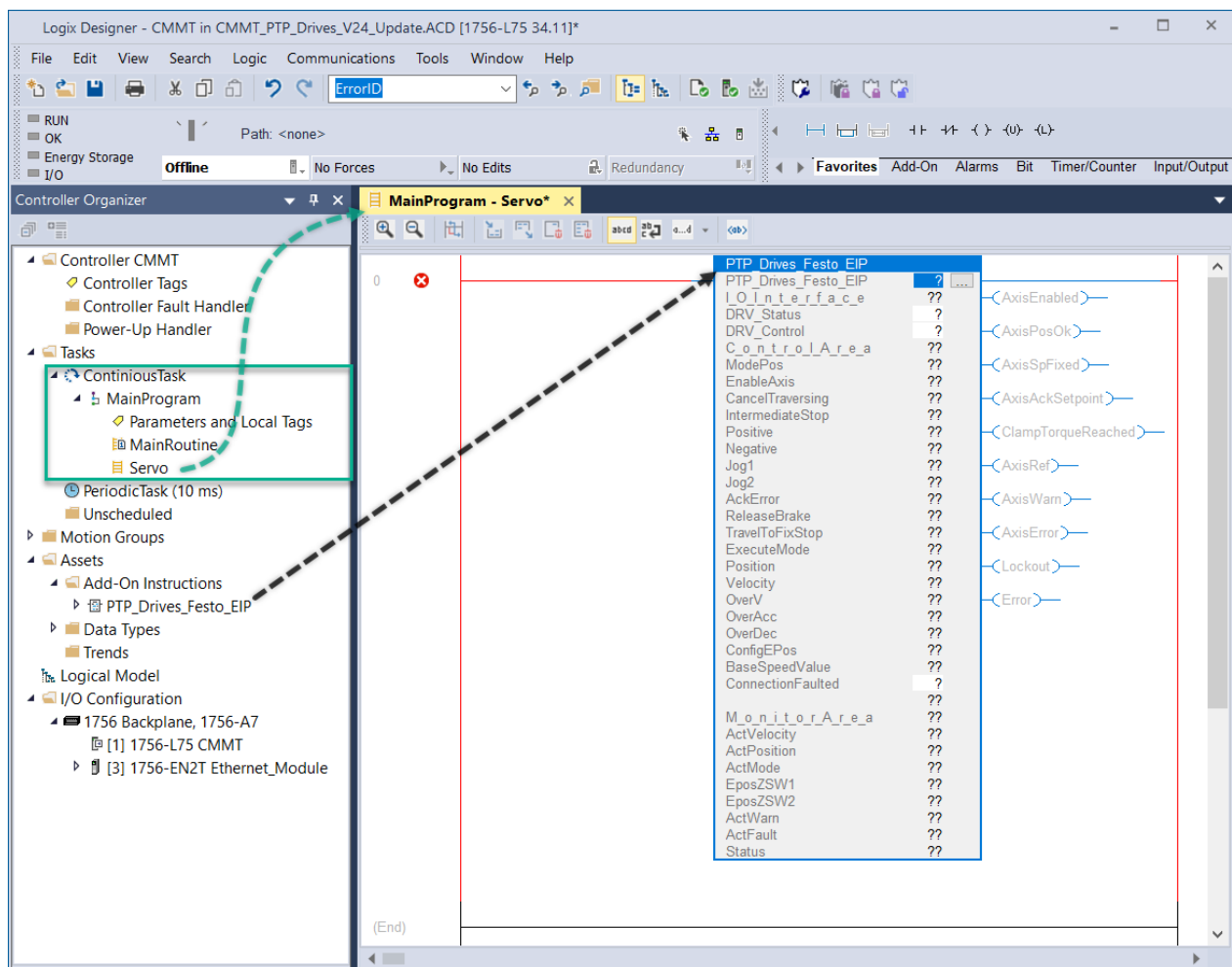


### Warning

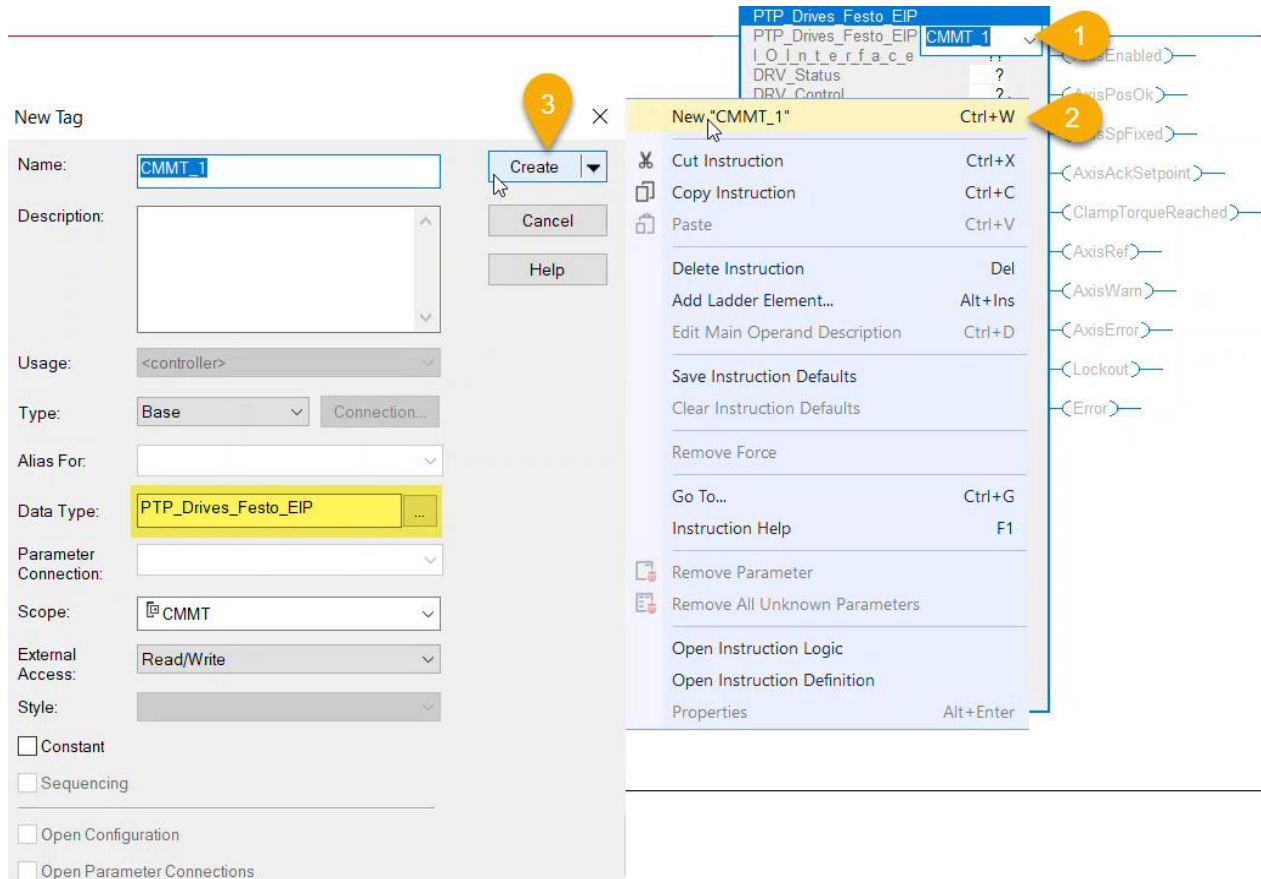
- The **WRONG WAY** is to have the AOI (Add-On Instruction) programmed within a periodic task like shown below.



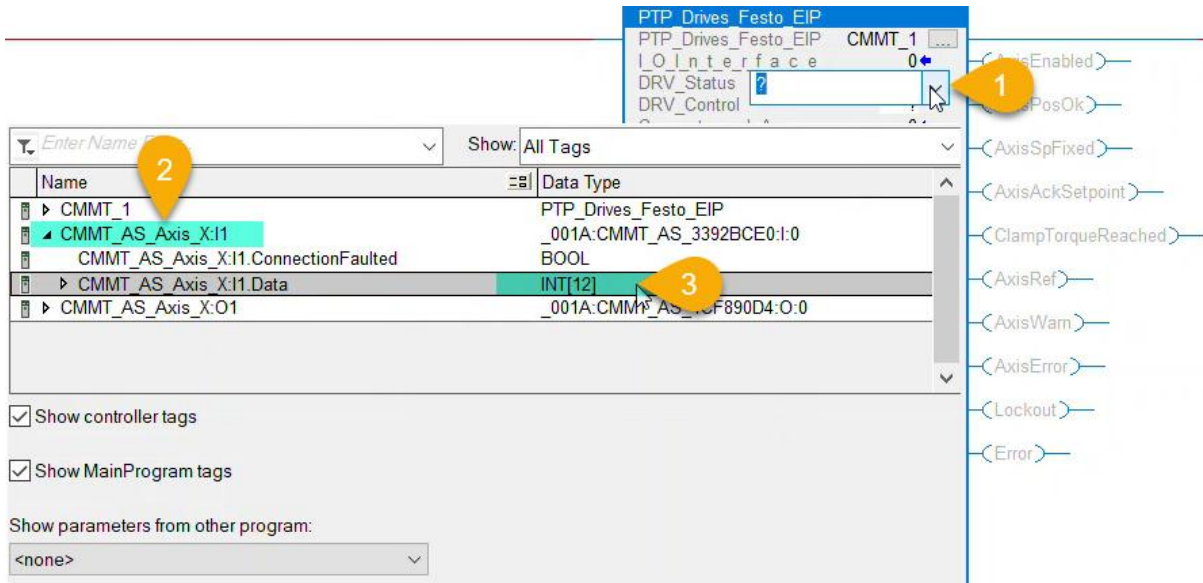
### Step – 2



Drag and drop the AOI from Add-On Instruction folder to the empty ladder rung as shown in above image.

**Step – 3**

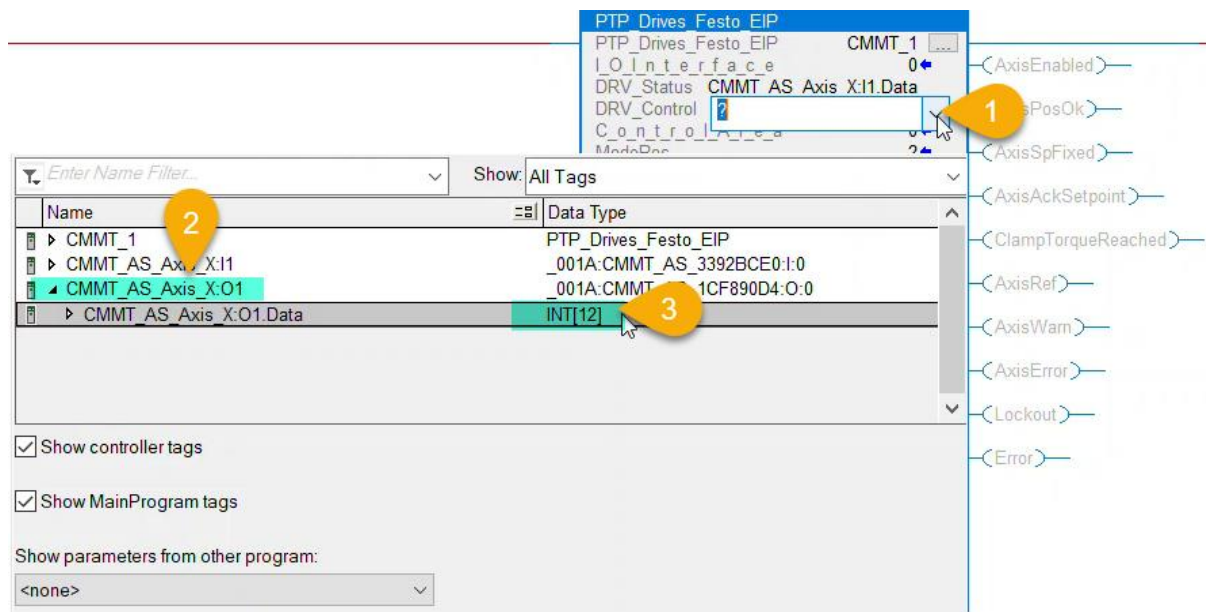
1. Enter the name of AOI & press enter.
2. Right click the entered name & select the first option “**New ( with tag name )**”.
3. Click “**Create**” button on new tag window to confirm the new tag creation. The yellow highlight indicates tag scope & select global or local scope depend on user program requirement.

**Step – 4**

1. Click the drop-down arrow of the “**DRV\_Status**” tag.
2. Select the input tag of the CMMT communication module and click the folder explorer.
3. Select the **INT array with 12 length** from CMMT input structure tag.

**Note**

- The yellow color highlights the data type of the CMMT input tag when EDS file is used as Interface module & green color highlights actual data type of DRV\_Status interface tag. However, user must select the tag “**INT Array with 12 length**” in both the cases (EDS & Generic).

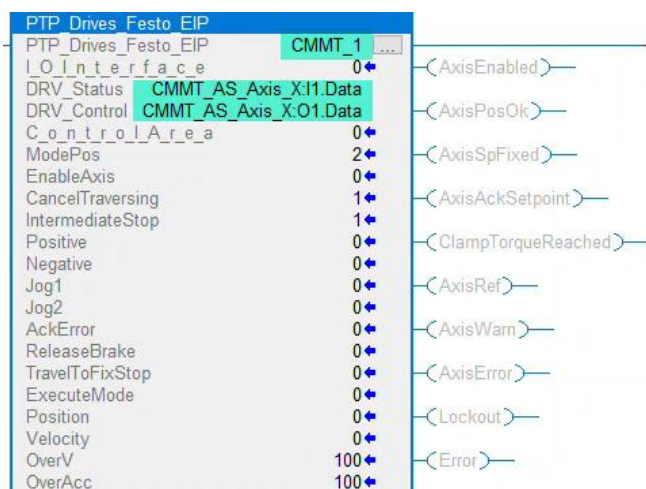
**Step – 5**

1. Click the drop-down arrow of the “**DRV\_Control**” tag.
2. Select the output tag of the CMMT communication module and click the folder explore.
3. Select the **INT array with 12 length** from CMMT input structure tag.

**Note**

- The yellow color highlights data type of the CMMT output tag when EDS file is used as Interface module & green color highlights actual data type of DRV\_Control interface tag. However, user must select the tag “**INT Array with 12 length**” in both the cases (EDS & Generic).

After completion of Steps 3 to 5 the preview of AOI looks like below.





## 4 PTP\_DRIVES\_FESTO\_EIP AOI / Position Control

This chapter describes about PTP\_DRIVES\_FESTO\_EIP AOI Input, Output tags details & AOI behavior in different operating modes.

### 4.1 Introduction of PTP\_DRIVES\_FESTO\_EIP AOI

|                      |                                     |         |        |
|----------------------|-------------------------------------|---------|--------|
| PTP Drives Festo EIP |                                     |         |        |
| PTP Drives Festo EIP |                                     |         | CMMT_1 |
| I/O Interface        |                                     |         | 0      |
| DRV_Status           | CMMT_AS_Axis_X:I1.Data              |         |        |
| DRV_Control          | CMMT_AS_Axis_X:O1.Data              |         |        |
| Control Area         |                                     |         | 0      |
| ModePos              |                                     | 2       |        |
| EnableAxis           |                                     | 0       |        |
| CancelTraversing     |                                     | 1       |        |
| IntermediateStop     |                                     | 1       |        |
| Positive             |                                     | 0       |        |
| Negative             |                                     | 0       |        |
| Jog1                 |                                     | 0       |        |
| Jog2                 |                                     | 0       |        |
| AckError             |                                     | 0       |        |
| ReleaseBrake         |                                     | 0       |        |
| TravelToFixStop      |                                     | 0       |        |
| ExecuteMode          |                                     | 0       |        |
| Position             |                                     | 0       |        |
| Velocity             |                                     | 0       |        |
| OverV                |                                     | 100     |        |
| OverAcc              |                                     | 100     |        |
| OverDec              |                                     | 100     |        |
| ConfigEPos           |                                     | 15      |        |
| BaseSpeedValue       |                                     | 3000.0  |        |
| ConnectionFaulted    | CMMT_AS_Axis_X:I1.ConnectionFaulted |         |        |
|                      |                                     | 0       |        |
| Monitor Area         |                                     |         | 0      |
| ActVelocity          |                                     | 0.0     |        |
| ActPosition          |                                     | 0       |        |
| ActMode              |                                     | 0       |        |
| EposZSW1             |                                     | 0       |        |
| EposZSW2             |                                     | 0       |        |
| ActWarn              |                                     | 0       |        |
| ActFault             |                                     | 0       |        |
| Status               |                                     | 16#0000 |        |

PTP\_DRIVES\_FESTO\_EIP AOI input & output tags are segregated as three different segments which are Hardware Interface tags, Control Area tags & Monitor area tags which are highlighted in orange, blue & yellow colors respectively in the above figure.

Also, this AOI is capable to control the drive in following modes

- Relative Position Mode
- Absolute Position Mode
- Go to Home/Reference using Festo Automation Suite Configuration
- Set Home to Current Position
- Traversing/Record Selection Mode
- Jog Mode
- Inching/Incremental Jog Mode.

## 4.2 IO Interface Area Tag Details

| Interface Area Tags | Data Type        | Function Description              |
|---------------------|------------------|-----------------------------------|
| DRV_Status          | Array of INT[12] | Drive feedback hardware interface |
| DRV_Control         | Array of INT[12] | Drive control hardware interface  |

## 4.3 Control Area Tag Details

| Control Area Tags | Data Type | Function Description                                                                                                                                                                      |
|-------------------|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ModePos           | INT       | Operating mode selection command<br>1=MDI Relative Pos.<br>2=MDI Absolute Pos.<br>3= Setup Mode<br>4=Home FAS Config.<br>5=Home Current Pos.<br>6=Record Select<br>7=Jogging<br>8=Inching |
| EnableAxis        | BOOL      | Enable the axis command                                                                                                                                                                   |
| CancelTraversing  | BOOL      | Cancel travel task ( 0 – Cancel Travel )<br>Ramp Stop @ 100% Set Deceleration<br>Task is Rejected                                                                                         |
| IntermediateStop  | BOOL      | Interrupt travel task ( 0 – Interrupt Travel )<br>Ramp Stop @ Current Deceleration<br>Task is NOT Rejected                                                                                |
| Positive          | BOOL      | Travel positive direction ( 1 – Positive Selected )                                                                                                                                       |
| Negative          | BOOL      | Travel negative direction ( 1 – Negative Selected )                                                                                                                                       |
| Jog1              | BOOL      | Jog profile – 1 activate command ( 1 – Active )<br>Jog Negative Direction                                                                                                                 |
| Jog2              | BOOL      | Jog profile – 2 activate command ( 1 – Active )<br>Jog Positive Direction                                                                                                                 |
| AckError          | BOOL      | Clear error command ( 1 – Clear error )                                                                                                                                                   |
| Release Brake     | BOOL      | Release Servo Motor Brake ( 1 – Release )                                                                                                                                                 |
| TravelToFixStop   | BOOL      | Enable Travel to Fix Stop mode ( 1 – Enable )                                                                                                                                             |
| ExecuteMode       | BOOL      | Execute the task ( 1 – Execute )                                                                                                                                                          |

| Control Area Tags | Data Type | Function Description                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-------------------|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Position          | DINT      | Position Target Setpoint or Record table selection in traversing block mode<br>Position Target Value<br>$\times$<br>FAS=P1.7841.0.0 Factor Group for Position = $10^{-6}$<br>=<br>FAS P1.11280605.0.0 Target Position MDI<br>Example:<br>Position Target Setpoint=<br>$2500000 \times 10^{-6} = 2500.000 \text{ mm}$                                                                                                                                                         |
| Velocity          | DINT      | Velocity Target Setpoint<br>Velocity in User Units While using MDI (Manual Data Input) Mode<br>Velocity Target Setpoint<br>$\times$<br>FAS=P1.7842.0.0 Factor Group for Velocity = $10^{-3}$<br>$\times$<br>OverV<br>FAS=P1.1309.0.0 Velocity Override<br>=<br>FAS P1.11280605.0.0 Profile Velocity MDI<br>Example1:<br>Velocity Target Setpoint = $50 \times 10^{-3} = 0.05 \text{ m/s}$<br>Example2:<br>Velocity Target Setpoint = $250 \times 10^{-3} = 0.25 \text{ m/s}$ |
| OverV             | INT       | Velocity override in % ( 0 to 199 % )<br>OverV Value = 0-100%<br>FAS=P1.1309.0.0 Velocity Override                                                                                                                                                                                                                                                                                                                                                                           |
| OverAcc           | INT       | Acceleration override in % ( 0 to 100 % )<br>OverAcc Target Value = 0-100%<br>$\times$<br>FAS P1.11280702.0.0 Base Value Acceleration<br>=<br>FAS P1.11280606.0.0 Acceleration MDI                                                                                                                                                                                                                                                                                           |
| OverDec           | INT       | Deceleration override in % ( 0 to 100 % )<br>OverDec Target Value = 0-100%<br>$\times$<br>FAS P1.11280703.0.0 Base Value Deceleration<br>=<br>FAS P1.11280607.0.0 Deceleration MDI                                                                                                                                                                                                                                                                                           |

| Control Area Tags | Data Type | Function Description                                                                                    |
|-------------------|-----------|---------------------------------------------------------------------------------------------------------|
| ConfigEPos        | DINT      | Configuration command ( Bit – Granular )<br>See section 4.3.1 for further details                       |
| BaseSpeedValue    | REAL      | Base speed value for actual speed display. Enter the same value given for parameter ( P1.11280701.0.0 ) |
| ConnectionFaulted | BOOL      | Communication disconnected acknowledgement for AOI                                                      |

#### 4.3.1 ConfigEPos Bit Description

| ConfigEPos Bits      | Function Description                                                                                        |
|----------------------|-------------------------------------------------------------------------------------------------------------|
| Bit - 0              | OFF2 ( Pulse Enable )                                                                                       |
| Bit - 1              | OFF3 ( Pulse Enable )                                                                                       |
| Bit - 2              | Software Limits Enable FAS=P1.112414140.0.0                                                                 |
| Bit - 3              | Hardware Limits Enable FAS=P1.112414150.0.0                                                                 |
| Bit - 4              | Probe Edge Evaluation                                                                                       |
| Bit - 5              | Select Probe                                                                                                |
| Bit - 6              | External Block Change (via BUS)                                                                             |
| Bit – 7              | -FW Pending - Signal Source Reference Mark                                                                  |
| Bit - 8              | -FW Pending - Continuous Setpoint Transfer MDI                                                              |
| Bit - 9              | - FW Pending - DDS Bit 0                                                                                    |
| Bit - 10             | - FW Pending - DDS Bit 1                                                                                    |
| Bit - 11             | - FW Pending - DDS Bit 2                                                                                    |
| Bit - 12             | - FW Pending - DDS Bit 3                                                                                    |
| Bit - 13             | - FW Pending - DDS Bit 4                                                                                    |
| Bit - 14             | - FW Pending - Parking Axis Selection                                                                       |
| Bit - 15             | Motor Brake 0=Close 1=Open                                                                                  |
| Bit – 16 to Bit - 29 | Reserved                                                                                                    |
| Bit - 30             | STW2.8 Travel to Fixed Stop ( User can also use the “ <b>TravelToFixStop</b> “ AOI input for this function. |
| Bit - 31             | Reserved                                                                                                    |

#### 4.4 Monitor Area Tag Details

| Monitor Area Tags  | Data Type | Function Description                                                                                                                                                                   |
|--------------------|-----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| AxisEnabled        | BOOL      | Axis enabled feedback ( 1 – Enabled )<br>Drive Ready and Switched On/Enabled                                                                                                           |
| AxisPosOk          | BOOL      | Axis reached target position ( 1 – Reached ) &<br>Motion Complete                                                                                                                      |
| AxisSpFixed        | BOOL      | Setpoint is stationary feedback ( 1 – SP stationary )                                                                                                                                  |
| AxisAckSetpoint    | BOOL      | Acknowledge setpoint ( 1 – Ack )                                                                                                                                                       |
| AxisRef            | BOOL      | Reference/Home is Completed ( 1 – Home Done )                                                                                                                                          |
| ClampTorqueReached | BOOL      | Axis reached Clamping Torque ( 1 – Reached )                                                                                                                                           |
| AxisWarn           | BOOL      | Drive has warning message ( 1 – Warning Active )                                                                                                                                       |
| AxisError          | BOOL      | Drive has error message ( 1 – Error Active )                                                                                                                                           |
| Lockout            | BOOL      | Switching ON inhibited ( 1 – ON Inhibited )                                                                                                                                            |
| ActVelocity        | REAL      | Actual velocity feedback ( Feedback value scaled by<br>base speed value )                                                                                                              |
| ActPosition        | DINT      | Axis actual position feedback                                                                                                                                                          |
| ActMode            | INT       | Actual mode selection feedback<br>1=MDI Relative Pos.<br>2=MDI Absolute Pos.<br>3=Setup Mode<br>4=Home FAS Config.<br>5=Home Current Pos.<br>6=Record Select<br>7=Jogging<br>8=Inching |
| EposZSW1           | INT       | Status of Epos ZSW1 ( Bit – Granular )<br>See section 4.4.1 for further details                                                                                                        |
| EposZSW2           | INT       | Status of Epos ZSW2 ( Bit – Granular )<br>See section 4.4.2 for further details                                                                                                        |
| ActWarn            | INT       | Actual warning number                                                                                                                                                                  |
| ActFault           | INT       | Actual fault number                                                                                                                                                                    |
| Error              | BOOL      | AOI Fault Active                                                                                                                                                                       |
| Status             | INT       | AOI Status Code: For more details refer <a href="#">chapter 4.4.3.</a>                                                                                                                 |

#### 4.4.1 Epos ZSW1 Bit Details

| EPos ZSW1 Bits | Function Description                        |
|----------------|---------------------------------------------|
| Bit - 0        | Position Record Selected Bit 0              |
| Bit - 1        | Position Record Selected Bit 1              |
| Bit - 2        | Position Record Selected Bit 2              |
| Bit - 3        | Position Record Selected Bit 3              |
| Bit - 4        | Position Record Selected Bit 4              |
| Bit - 5        | Position Record Selected Bit 5              |
| Bit - 6        | Position Record Selected Bit 6              |
| Bit - 7        | Reserved                                    |
| Bit - 8        | Negative Hardware Limit Switch is Activated |
| Bit - 9        | Positive Hardware Limit Switch is Activated |
| Bit - 10       | Jogging Task is Active                      |
| Bit - 11       | Homing Task is Active                       |
| Bit - 12       | FW Pending - Flying Reference Active        |
| Bit - 13       | Position Record Task is Active              |
| Bit - 14       | FW Pending - MDI Setup Active               |
| Bit - 15       | MDI Mode is Active                          |

#### 4.4.2 Epos ZSW2 Bit Details

| EPos ZSW2 Bits | Function Description                         |
|----------------|----------------------------------------------|
| Bit - 0        | Tracking Mode Active                         |
| Bit - 1        | Velocity Limit Active                        |
| Bit - 2        | Ready to Accept new Setpoint                 |
| Bit - 3        | FW Pending – Print Mark Outside Outer Window |
| Bit - 4        | Axis is Moving Positive                      |
| Bit - 5        | Axis is Moving Negative                      |
| Bit - 6        | Negative Software Limit Switch is Activated  |
| Bit - 7        | Positive Software Limit Switch is Activated  |
| Bit - 8        | Actual Position <= CAM Position 1            |
| Bit - 9        | Actual Position <= CAM Position 2            |
| Bit - 10       | Direct Output 1 Via Traversing Block         |

| <b>EPos ZSW2 Bits</b> | <b>Function Description</b>          |
|-----------------------|--------------------------------------|
| Bit - 11              | Direct Output 2 Via Traversing Block |
| Bit - 12              | Fixed STOP Reached                   |
| Bit - 13              | Fixed STOP Clamping Torque Reached   |
| Bit - 14              | Travel to Fixed STOP Active          |
| Bit - 15              | Traversing Task Active               |

#### 4.4.3 Diagnostic Messages

The function block gives multiple types of diagnostic messages. The following chapter will describes how to handle & solve function block faults.

| <b>Status Error Number</b> | <b>Case</b>                                       | <b>Remedy</b>                                                                                                                                                                                                                          |
|----------------------------|---------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 16#7002                    | No Error                                          | --                                                                                                                                                                                                                                     |
| 16#8202                    | Incorrect Operating Mode Selected                 | Set ModePos tag value between 1 to 8                                                                                                                                                                                                   |
| 16#8203                    | Incorrect parameterization of the override inputs | Check the setting of override inputs                                                                                                                                                                                                   |
| 16#8204                    | Invalid traversing block number                   | Enter the traversing block number from 0 to 127                                                                                                                                                                                        |
| 16#8401                    | Alarm message(s) in the drive                     | Check the error code in ActFaulted tag                                                                                                                                                                                                 |
| 16#8402                    | Switching ON Inhibited                            | The switch-on lock is active. The drive is in the status S1 Switching On Inhibited.<br>Switching on is only possible with the following command sequence: OFF (OFF1) and no coasting (no OFF2) and no fast stop (no OFF3) and then ON. |
| 16#8403                    | FW Pending - AxisRef could NOT be Started         | Check for pending alarms / faults in the drive                                                                                                                                                                                         |
| 16#8600                    | Communication Disconnected                        | Check the communication connections/ setting between drive and PLC                                                                                                                                                                     |

## 5 PTP\_DRIVES\_FESTO\_EIP AOI Functions

This chapter describes about the configuration of PTP\_Drives\_FESTO\_EIP AOI for different operation modes, Enabling & Stopping the axis, Enabling hardware & software position limits as well as Actual speed monitoring.

### 5.1 Axis Enabling & Stopping

For axis enabling following tags must be in below configuration

- OFF2 & OFF3 of tag “**ConfigEPos**” bit 0 & 1 must be TRUE.
- “**AxisError**” & “**Lockout**” feedback tags must be FALSE to ensure drive is in ready state.
- “**ModePos**” value must be between 1 to 8.
- “**CancelTraversing**”(Reject traversing task) & “**IntermediateStop**”(intermediate stop) tags must be TRUE.
- “**EnableAxis**” must be TRUE & “**AxisEnabled**” feedback will goes to TRUE after enable.
- Toggle “**AckError**” control bit to clear any error in AOI.

### 5.2 Axis Software & Hardware Position Limits Enabling

“**ConfigEPOS tag bits 2 & 3**” are used to enable the software & hardware position limits respectively

- “**ConfigEPOS.2 = TRUE**” this means software position limit is active (Festo Automation Suite Parameter P1.112414140.0.0)
- “**ConfigEPos.3 = TRUE**” this means hardware position limit is active (Festo Automation Suite Parameter P1.112414150.0.0)

When software position limit is active please pay attention if the below parameters are configured correctly as per application requirement.

| Axis configuration                                                                                                   |                                                                                                                       |
|----------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------|
| Reversing the direction of rotation                                                                                  |  <input type="checkbox"/> Active |
|  Axis zero point offset           | <input type="text" value="0.03"/> r                                                                                   |
| Software limit positions active                                                                                      | <input checked="" type="checkbox"/> Active                                                                            |
|  Negative software limit position | <input type="text" value="-0.03"/> r                                                                                  |
|  Positive software limit position | <input type="text" value="999.97"/> r                                                                                 |

### 5.3 Axis Actual Speed Monitoring

User can monitor the actual speed of the axis in the variable “**ActVelocity**”. The actual speed of the axis is scaled from 0 to 100% depending on base speed value of parameter number ( P1.11280701.0.0 ). So, it is mandatory to feed the same value in input tag “**BaseSpeedValue**” of AOI as like parameter number ( P1.11280701.0.0 ).



#### Note

- Make sure the AOI input tag “**BaseSpeedValue**” & Festo Automation Suite parameter number ( P1.11280701.0.0 ) values are same. AOI will display wrong velocity value If both values are not equal. So, enter the same value as given in [Chapter-2.5 Step – 3](#).

### 5.4 AOI Mode of Operation

The PTP\_DRIVES\_FESTO\_EIP AOI will support following modes

- **ModePos – 1:** Relative Positioning
- **ModePos – 2:** Absolute Positioning
- **ModePos – 3:** Setup Mode
- **ModePos – 4:** Home/Reference using Festo Automation Suite Configuration
- **ModePos – 5:** Home to Current Position
- **ModePos – 6:** Traversing/Record Select Mode
- **ModePos – 7:** Jog
- **ModePos – 8 :** Inching/Incremental Jog Mode

#### 5.4.1 Relative Positioning Mode [ModePos – 1]

1. Configuration for Relative Positioning Mode
  - Set the “**ModePos**” value to 1.
  - Check “**AxisRef**” status tag is True. If not true user could also use other modes like mode 4 or 5 to set the reference point for an axis.
  - Enable axis by toggling “**EnableAxis**” command tag.
  - “**CancelTraversing**” and “**IntermediateStop**” command tags must be in true state.
  - Enter the percentage of Override acceleration, Override deceleration & Override velocity in “**Over-Acc**”, “**OverDec**” & “**OverV**” respective command tags.
  - The status tags “**AxisEnabled**” & “**AxisRef**” must be true and status tag “**Error**” must be false.
2. Sequence Command
  - Enter the Position command value in “**Position**” tag & Velocity command value in “**Velocity**” tag. To move axis in forward direction, give position command in positive value & for moving axis in reverse direction give position command in negative value.
  - Toggle the “**ExecuteMode**” command tag to start motion task for configured parameter. Rising edge of ExecuteMode tag will start the command. So, for every new position command tag want to toggle. Check the feedback tag “**ActMode**” has the same value as ModePos.
  - User will receive execute command accepted handshake signal from “**AxisAckSetpoint**” status tag.

- If task is completed successfully, it is acknowledge by rising edge of “**AxisPosOk**” status variable. Also if task completed unsuccessfully with an error “**AxisError**” status variable will indicate it by going false to true.
- “**AxisSpFixed**” Status variable indicates the axis mobility status . If the axis is standby the “**Axis-SpFixed**” status variable will be high.



#### Note

- Refer the document “**CMMT-xx-EP PTP\_Drives\_Festo\_EIP Direct Mode 1 EtherNetIP Timing Sequence**” document available with this download for execution command sequence.
- The existing running task command can replaced by new task command only on mode 1 and 2.

### 5.4.2 Absolute Positioning Task [ModePos – 2]

#### 1. Configuration for Absolute Positioning Mode

- Set the “**ModePos**” value to 2 .
- Check “**AxisRef**” status tag is True. If not true user could also use other modes like mode 4 or 5 to set the reference point for an axis.
- Enable axis by toggling “**EnableAxis**” command tag .
- “**CancelTraversing**” and “**IntermediateStop**” command tags must be in true state.
- Enter the percentage of Override acceleration, Override deceleration & Override velocity in “**Over-Acc**”, “**OverDec**” & “**OverV**” respective command tags.
- The status tags “**AxisEnabled**” & “**AxisRef**” must be true. Also status tag “**Error**” must be false.

#### 2. Sequence Command

- Enter the Position command value in “**Position**” tag & Velocity command value in “**Velocity**” tag. Give the absolute position value. The drive will decide forward & reverse direction based on its current position.
- Toggle the “**ExecuteMode**” command tag to start motion task for configured parameter. Rising edge of ExecuteMode tag will take for task start command. So for every new position command tag want to toggle. Check the feedback tag “**ActMode**” has the same value as ModePos.
- User will receive execute command accepted handshake signal from “**AxisAckSetpoint**” status tag.
- If commanded task is completed successfully, it is acknowledge by rising edge of “**AxisPosOk**” status variable. Also if task completed unsuccessfully with an error “**AxisError**” status variable will indicate it by going false to true.
- “**AxisSpFixed**” Status variable indicates the axis mobility status . If the axis is standby the “**Axis-SpFixed**” status variable will be high.



#### Note

- Refer the document “**CMMT-xx-EP PTP\_Drives\_Festo\_EIP Direct Mode 2 EtherNet/IP Timing Sequence**” document available with this download for execution command sequence.
- The existing running task command can replaced by new task command only on mode 1 and 2.

### 5.4.3 Setup Mode [ModePos – 3]

In the setup mode, the drive is traversed at a constant speed by means of position control. There's no need to specify a target position. It must be noted that continuous motion (e.g. conveyor belt, fan etc.) cannot be implemented in this mode.

1. Configuration for Setup Mode
  - Set the **"ModePos"** value to 3.
  - Check **"AxisRef"** status tag is True. If not true user could also use other modes like mode 4 or 5 to set the reference point for an axis.
  - **"CancelTraversing"** and **"IntermediateStop"** command tags must be in true state.
  - Enter the percentage of Override acceleration, Override deceleration & Override velocity in **"OverAcc"**, **"OverDec"** & **"OverV"** respective command tags.
  - Select the axis moving direction by the inputs **"Positive"** or **"Negative"**.
  - Enable axis by toggling **"EnableAxis"** command tag.
2. Sequence Command
  - Motion dynamics are defined via the **"Position"**, **"Velocity"**, **"OverV"**, **"OverAcc"** and **"OverDec"** inputs.
  - Motion is started by means of a positive edge at **"ExecuteMode"**. The motion command is acknowledged via the **"AxisAckSetpoint"** output. A handshake mechanism can be programmed in this way. Check the feedback tag **"ActMode"** has the same value as ModePos.
  - If a drive error has occurred during motion, this is indicated via the **"AxisError"** output. Any error occurring at the function block is indicated via the "Error output".

### 5.4.4 Home/Reference Point using Festo Automation Suite Configuration [ModePos – 4]

The reference point approach mode used to moves axis to home position.

1. Configuration for Reference Point Approach Mode
  - Set the **"ModePos"** value to 4 . The status tags **"AxisEnabled"** must be true & **"Error"** must be in false to ensure no error.
  - Axis must be in standstill state.
2. Sequence Command
  - All the required parameters for homing will be considered from Automation Suite configuration in "Homing Parameters" like Homing mode, velocity, acceleration & deceleration.
  - Toggle the **"ExecuteMode"** to start task. Check the feedback tag **"ActMode"** has the same value as ModePos.
  - **"AxisRef"** will become true once the axis finds reference point. If an error occurs during traveling motion, The output signal "Error" will become true .

#### 5.4.5 Home to Current Position [ModePos – 5]

This mode marks current position as reference point. This will be used when homing mode is selected as current position(37).

1. Configuration for Set Reference Point Mode
  - Set the “**ModePos**” value to 5. The status tags “**AxisEnabled**” must be true & “**Error**” must be in false to ensure no error.
  - Axis must be in standstill state.
2. Sequence Command
  - Toggle the “**ExecuteMode**” to start task. Check the feedback tag “**ActMode**” has the same value as ModePos.
  - “**AxisRef**” will become true after execute command. If an error occurs, The output signal “**Error**” will be true.

#### 5.4.6 Traversing/Record Selection Mode [ModePos – 6]

Traversing Block mode is implemented via the “Recode Table” of drive function. This mode retrieves parameters like Velocity, Position, Acceleration & Deceleration values from record table saved in the CMMT drive. It enables the creation of automatic programs, velocity mode, force mode, travel to fixed stop and outputs to be set and reset.

1. Configuration for Traversing Block Mode
  - Set the “**ModePos**” value to 6.
  - Check “**AxisRef**” status tag is True. If not true user could also use other modes like mode 4 or 5 to set the reference point for an axis.
  - Enable axis by toggling “**EnableAxis**” command tag .
  - “**CancelTraversing**” and “**IntermediateStop**” command tags must be in true state.
  - Enter the percentage of Override acceleration, Override deceleration & Override velocity in “**Over-Acc**”, “**OverDec**” & “**OverV**” respective command tags.
  - The status tags “**AxisEnabled**” & “**AxisRef**” must be true. Also status tag “**Error**” must be false.
2. Sequence Command
  - Select the record table by entering the table number in “**Position**” command tag. The value must be between 0 to 127.
  - Toggle the “**ExecuteMode**” command tag to start motion task for configured parameter. Rising edge of ExecuteMode tag will start the command. So for every new position command tag want to toggle. Check the feedback tag “**ActMode**” has the same value as ModePos.
  - User will receive execute command accepted handshake signal from “**AxisAckSetpoint**” status tag.
  - If task is completed successfully, it is acknowledge by rising edge of “**AxisPosOk**” status variable. Also if task completed unsuccessfully with an error “**AxisError**” status variable will indicate it by going false to true.
  - “**AxisSpFixed**” Status variable indicates the axis mobility status . If the axis is standby the “**Axis-SpFixed**” status variable will be high.

**Note**

- Refer the document “**CMMT-xx-EP PTP\_Drives\_Festo\_EIP Record Mode 6 EtherNetIP Timing Sequence**” document available with this download for execution command sequence.

#### 5.4.7 Jog Mode [ModePos – 7]

1. Configuration for Jog Mode

- Set the “**ModePos**” value to 7.
- Enable axis by toggling “**EnableAxis**” command tag.
- Axis must be in standstill .
- Velocity, Acceleration & Jerk values are taken from Automation Suite configuration under jog mode tab.

2. Sequence Command

- “**Jog1**” & “**Jog2**” tags are used for Positive & Negative direction commands respectively.
- The “ExecuteMode” signal is not required for jog mode. Check the feedback tag “**ActMode**” has the same value as ModePos.

#### 5.4.8 Inching/Incremental Jog Mode [ModePos – 8]

Incremental jog mode will not work like ordinary jog mode. It will rotate the axis step by step forward or reverse based on direction selection, But ordinary jog mode rotates the axis continuously. All the configuration and command configurations are same as ModePos – 7 but only different is ModePos must select as 8 to choose incremental jog mode.

**Note**

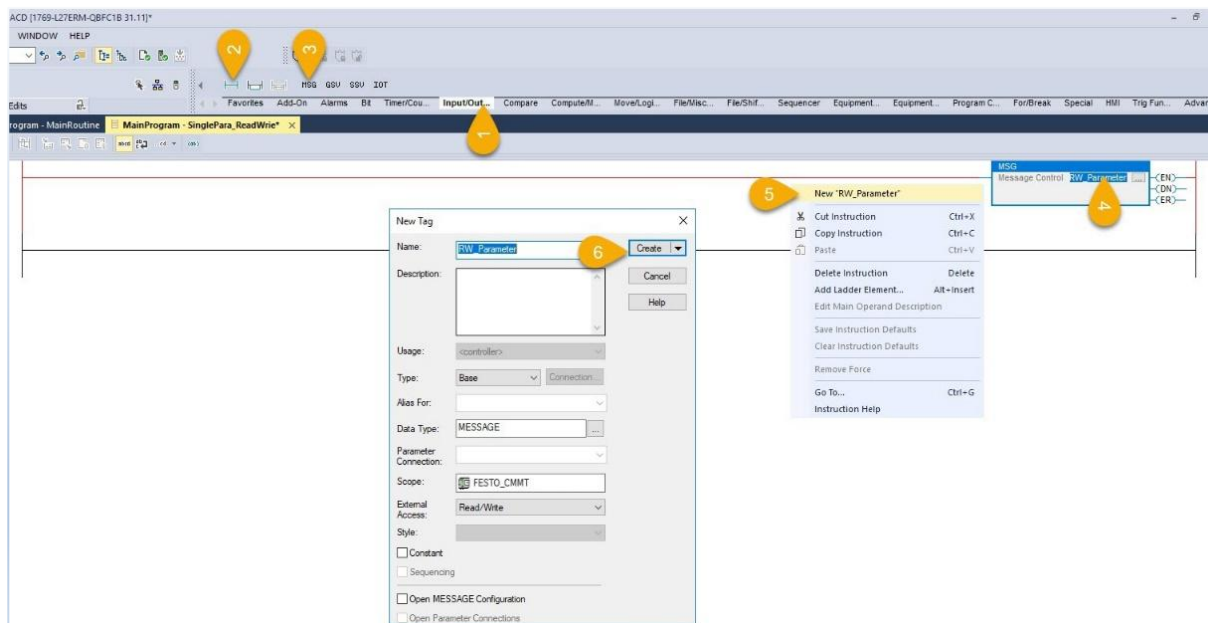
- The inching distance of positive & negative direction will be based on values set in parameters Relative Position Jog1 ( P1.214530.0.0) & Relative Position Jog2 ( P1.214538.0.0) respectively.

## 6 CMMT Drive Parameter Read/Write

This chapter describes about read & write a single & multi parameter in FESTO CMMT drive from Allen-Bradley PLC through Ethernet/IP network. Also detail explanation is given for Message instruction configuration in Studio-5000.

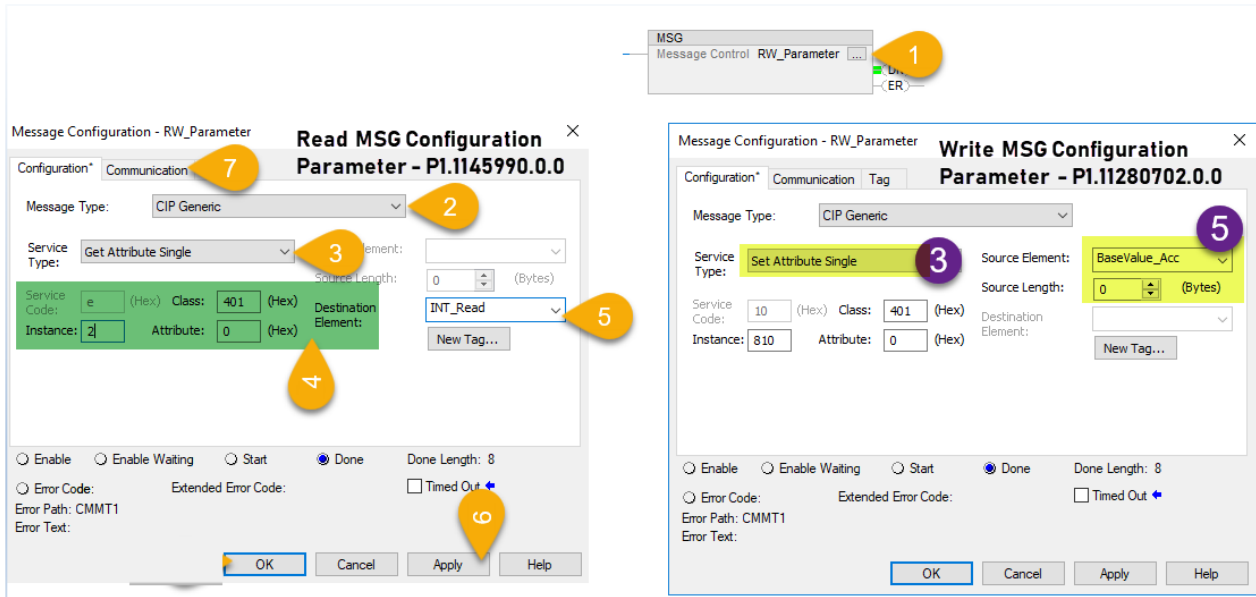
### 6.1 Single Parameter Read/Write

#### Step – 1



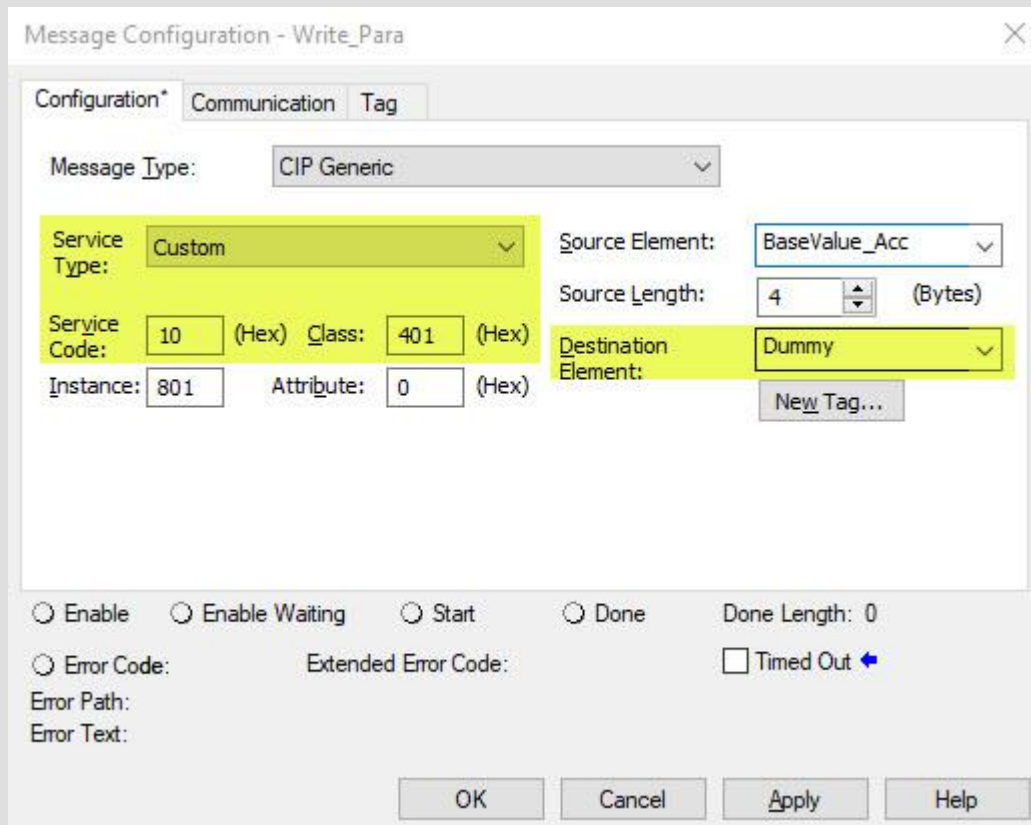
1. Go to **"Input/Output"** instruction tab in instruction pallet
2. Select create new rung button, Now user can see the new rung in red color as shown in above picture.
3. Select **"MSG"** button to create new message instruction in rung.
4. Enter the tag name of the message instruction & right click.
5. Select **"New"** option from the right click menu.
6. Select **"Create"** button from new tag menu.

## Step – 2



### Note

- If CMMT drive firmware is V17.05.0 then user must use **“Custom”** service type in write MSG instruction configuration with following setting **Service Code: 10, Class: 401, Instance: PNU Value, Attribute: PNU Value** & dummy 32 DINT destination element tag.



- Follow Rockwell knowledgebase **ID:27896** for more details about write MSG instruction configuration for CMMT drive firmware V17.05.0.

1. Click the **“browse”** button of the MSG instruction.
2. Set the **“Message Type”** as **“CIP Generic”**.

- Set the **“Service Type”** as **“Get Attribute Single”** for Read Data & set **“Set Attribute Single”** for write data.
- Ether the **Class (Always 401)**, Instance & Attribute value of the parameter. This value is common for both read & write message instruction. **Refer the PNUs Reference List table (Page No: 926) from CMMT drive user manual** to discover the Instance & Attribute values of each parameter. Below picture illustrate the identification tip.

| PNU                                | Name           | Data type  | Access | Parameter               |
|------------------------------------|----------------|------------|--------|-------------------------|
| <b>Profile specific parameters</b> |                |            |        |                         |
| 847.0 ... 63                       | Warning number | Unsigned16 | ro     | P1.11280042.0-.0 ... 63 |

Whole Number Part of PNU Value is the "Instance" Value

Fractional Part of PNU Value is Attribute Value of MSG Instruction. If Attribute Value is an Array like (0..63) Enter the Array Element Number as

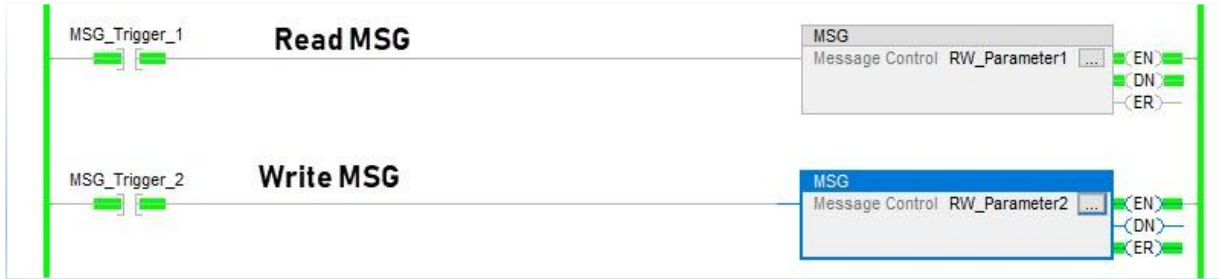
- Assign the variables for read and write. For **reading a parameter**, variable must be assigned in **destination elements** of the MSG instruction with its corresponding data type and length & For **writing a parameter**, variable must be assign in **source elements** of the MSG instruction with its corresponding data type & length. Also specify the length of the write elements in **“Source Length”** tab of the MSG instruction.
- Click **‘Apply’** button to confirm the configuration.
- Click the **“Communication”** tab to configure the communication path of the MSG instruction.

### Step – 3

- Click **“Browse”** button and select the communication interface module of CMMT drive.
- Click **“Apply”** button to confirm the communication path.

3. Click **“OK”** button to confirm & close the MSG instruction configuration window.

#### Step – 4



Here is a sample code to trigger MSG instruction to read and write the parameters from CMMT drive.

After trigger command if MSG instruction is executed successfully then Done bit (DN) of the MSG will be True and Error bit (ER) will be False as like Read MSG rung. When MSG unsuccessful the Error bit (ER) will become true & Done bit (DN) become false as like Write MSG rung.



#### Note

- Parameter Read/Write logic works with CMMT drive Firmware **Version V17.05.0** and above.

## 6.2 Multi Parameter Read/Write

This chapter describes about the multiple parameter read or write using PTP\_DRIVES\_PARA\_MULTI AOI. This chapter also explains about the AOI configuration and MSG instruction configuration.



#### Note

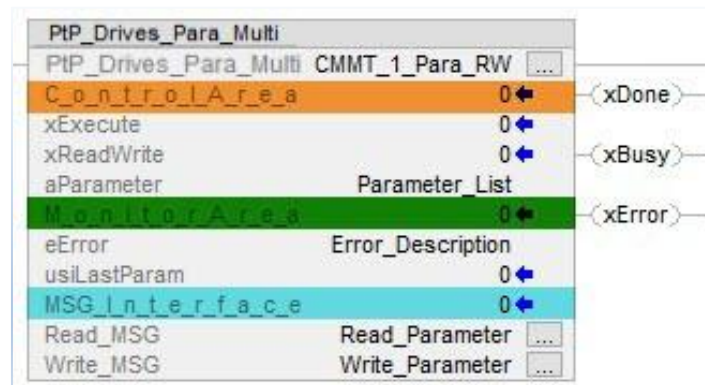
- PTP\_DRIVES\_PARA\_MULTI AOI is used to read or write parameters up to 32 bits data size. If the user wants to read or write the parameters like 64 bits data size, Use the Single parameter Read/Write sample code as like in [chapter 6.1](#).
- User is able to process either read or write function up to 100 parameters with this AOI.



#### Note

- Parameter Read/Write logic works with CMMT drive Firmware **Version V17.05.0** and above.

### 6.2.1 Introduction of PTP\_DRIVES\_PARA\_MULTI AOI



PTP\_DRIVES\_PARA\_MULTI AOI tags are classified as Control Area, Monitor Area & MSG Interface Area tags as highlighted Orange, Green & Blue colors respectively in the above picture.

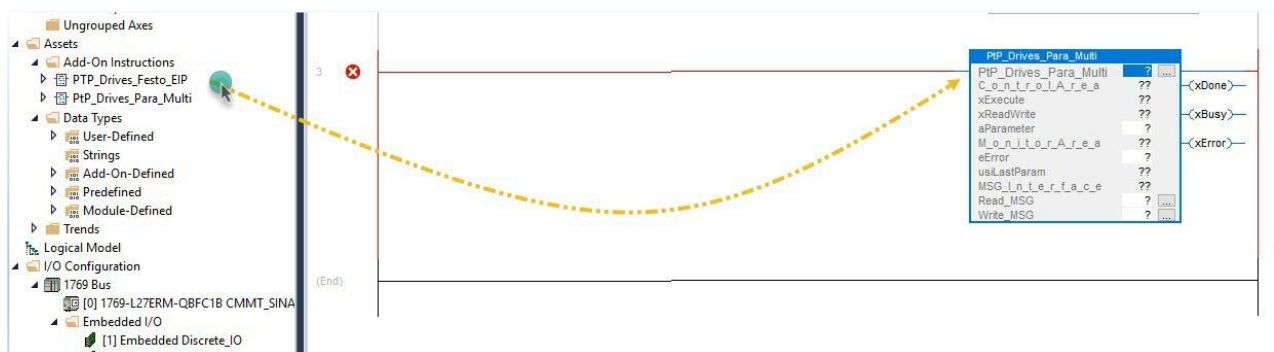
### 6.2.2 Configure the PTP\_DRIVES\_PARA\_MULTI AOI IO Tags



#### Note

- Before configuring the IN & Out tags of the AOI, Import the PTP\_DRIVES\_PARA\_MULTI.L5x import file into the project as shown in [Chapter 3.4](#).
- The Add-On Instruction must be called cyclically with a separate instance for every servo drive integrated in the user program (every axis). Simultaneous use of multiple instances of the same device is not permitted.
- Any AOI's need to be in a continuous task called cyclically. ([Refer Chapter 3.5 Step – 1](#))

#### Step – 1



Drag & drop the imported PTP\_DRIVES\_PARA\_MULTI AOI in an empty rung from Organizer > Assets > Add-On Instructions.

## Step – 2

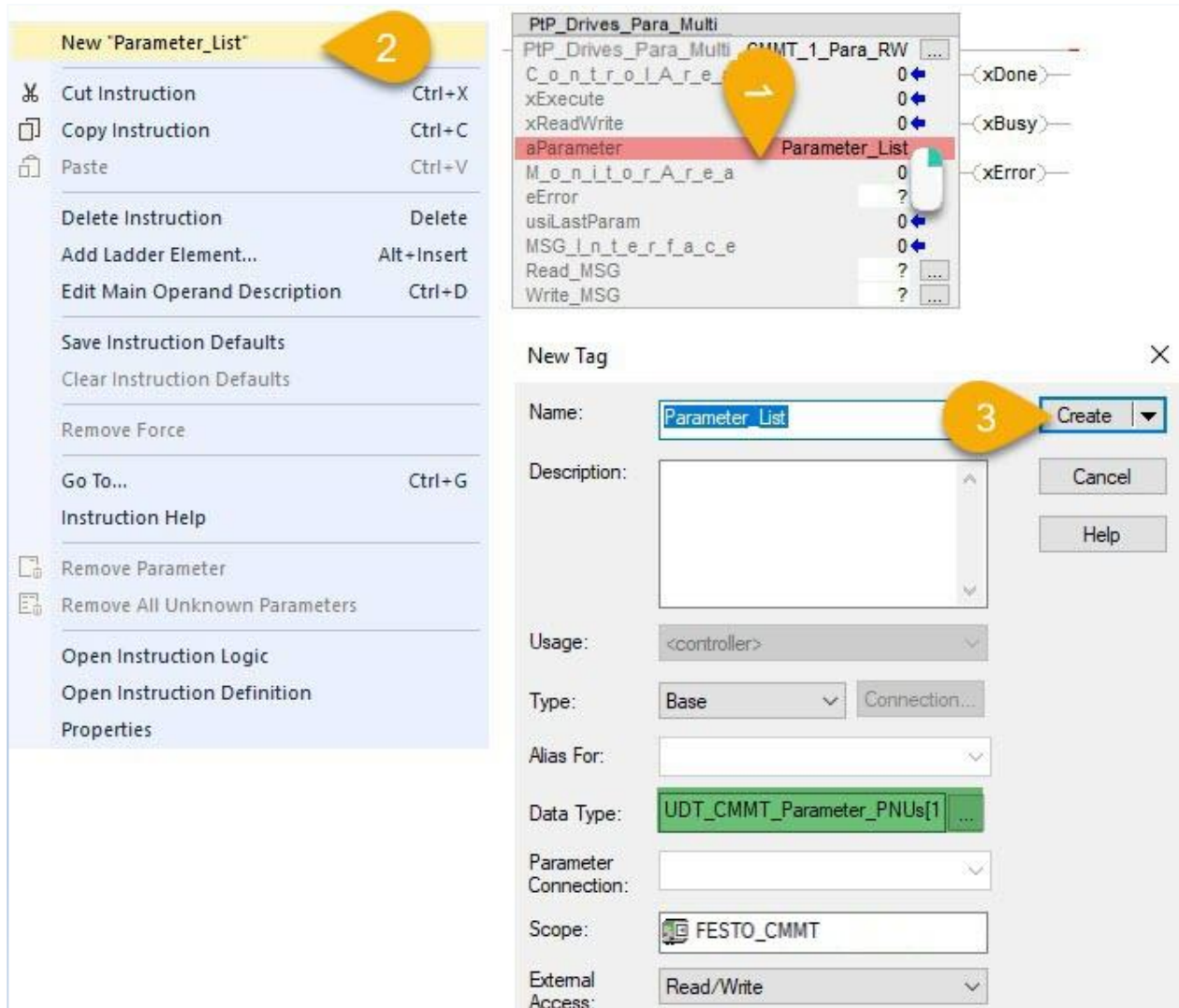
The screenshot shows the 'New Tag' dialog box with the following fields and values:

- Name:** CMMT\_1\_Para\_RW
- Description:** (Empty)
- Usage:** <controller>
- Type:** Base
- Alias For:** (Empty)
- Data Type:** SINA\_PARA\_FESTO\_EIP
- Parameter Connection:** (Empty)
- Scope:** FESTO\_CMMT
- External Access:** Read/Write

The 'Create' button is highlighted with a yellow callout '3'. The 'PtP\_Drives\_Para\_Multi' table is visible in the background, with 'CMMT\_1\_Para\_RW' highlighted in the 'Name' column. A yellow callout '2' points to the 'New Tag' button in the right-click menu.

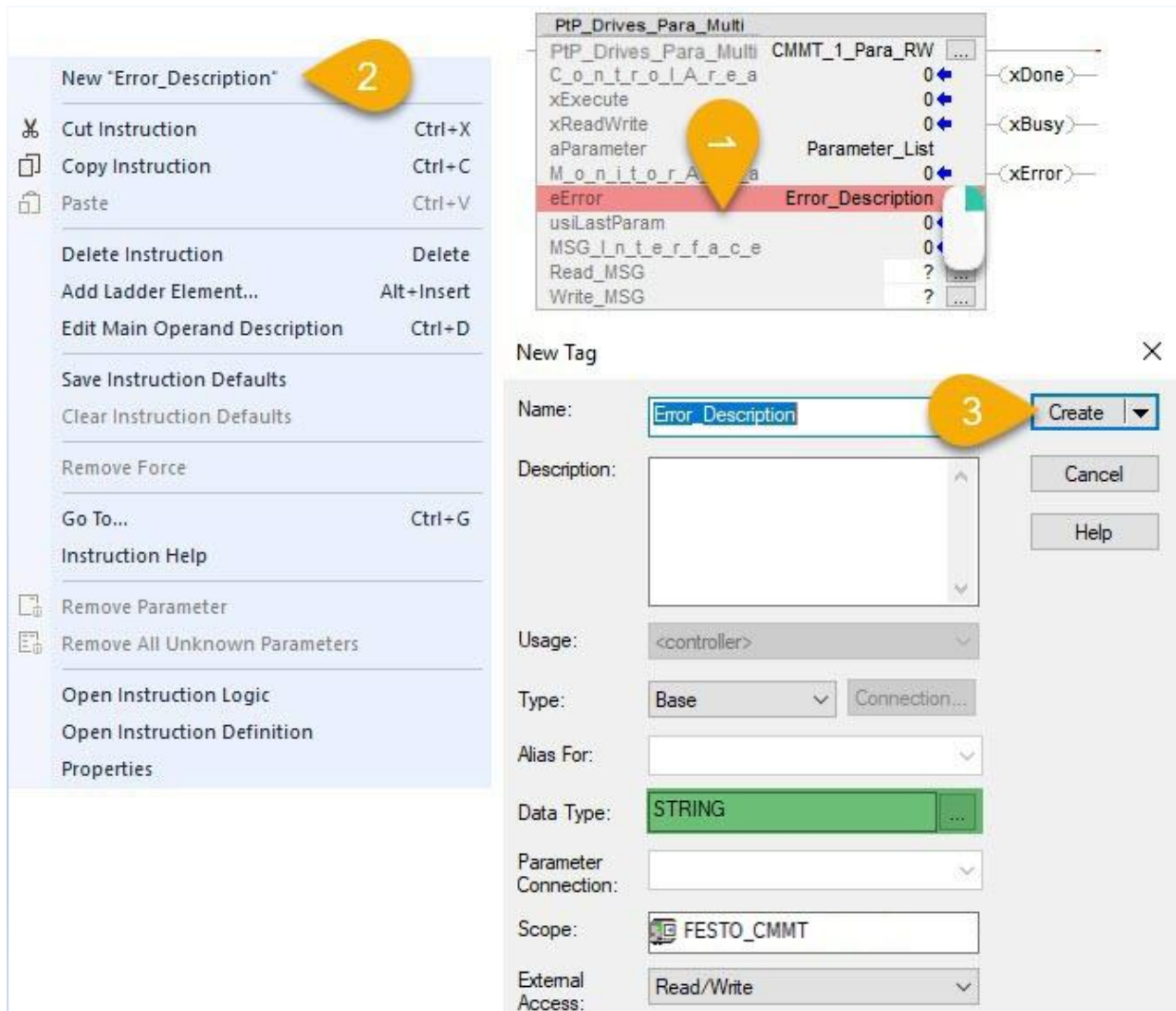
1. Enter the name of the AOI in highlighted location & Right click it.
2. Select “New” with tag name option from right click menu.
3. Click “Create” option from new tag window.

### Step – 3



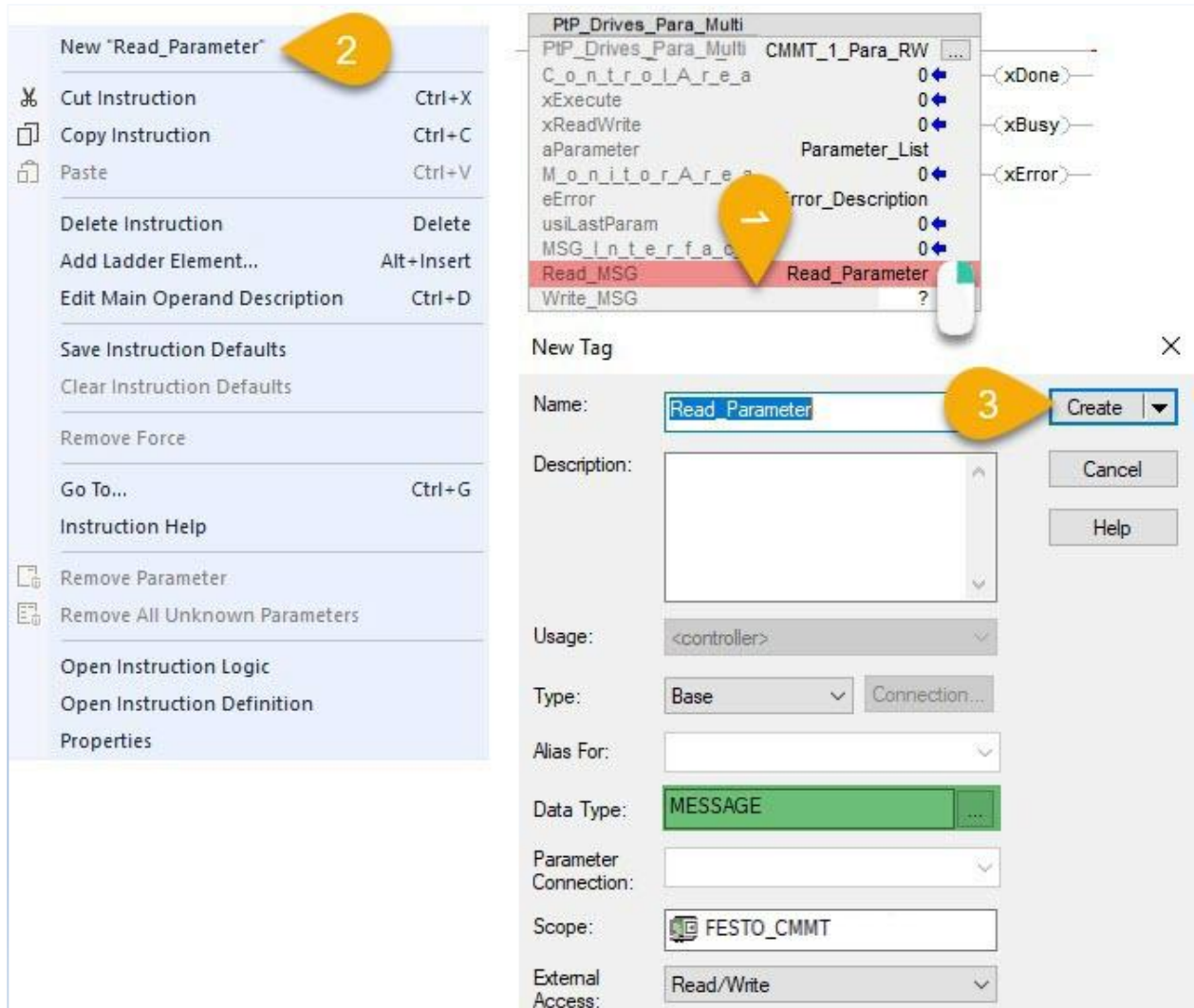
1. Enter the name of the interface tag for “**aParameter**” and right click.
2. Select “New” option with entered tag name from right click menu.
3. Click “**Create**” button on new tag window & ensure Data Type is “**UDT\_CMMT\_Parameter\_PNUs[100]**” as highlighted in green color.

#### Step – 4



1. Enter the name of the interface tag for “**eError**” and right click.
2. Select “New” option with entered tag name from right click menu.
3. Click “**Create**” button on new tag window & ensure Data Type is “**STRING**” as highlighted in green color.

### Step – 5



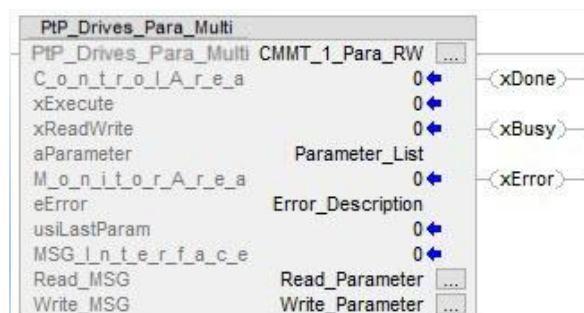
1. Enter the name of the interface tag for “**Read\_MSG**” and right click.
2. Select “New” option with entered tag name from right click menu.
3. Click “**Create**” button on new tag window & ensure Data Type is “**MESSAGE**” as highlighted in green colour.



#### Note

- Follow the above **step – 5** to create “**Write\_MSG**” interface tag.

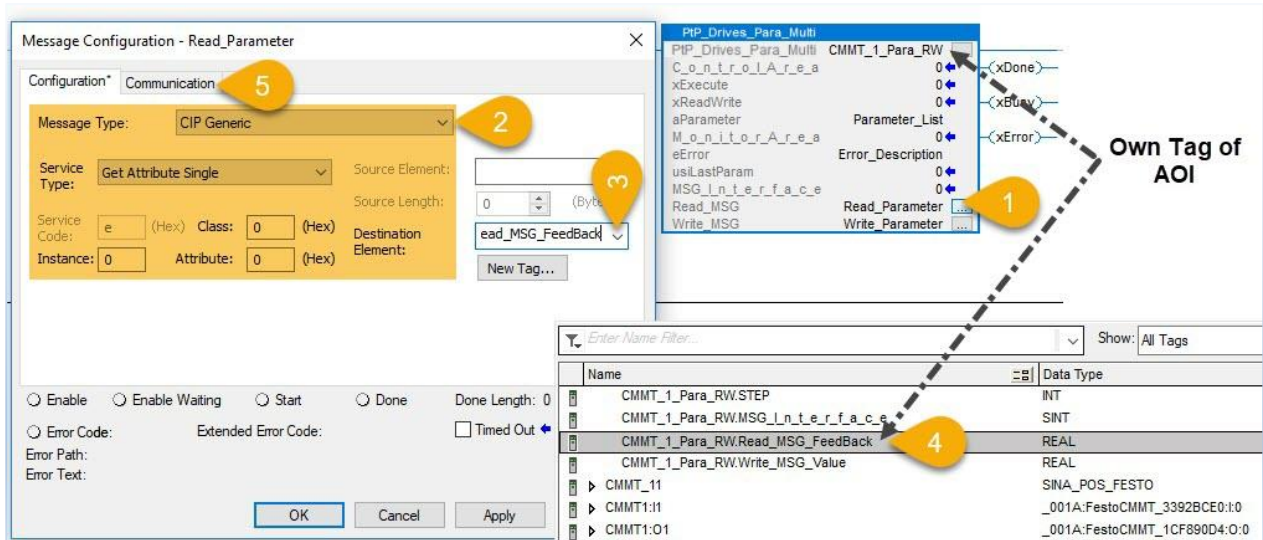
After all interface tags configuration PTP\_DRIVES\_PARA\_MULTI AOI looks like below



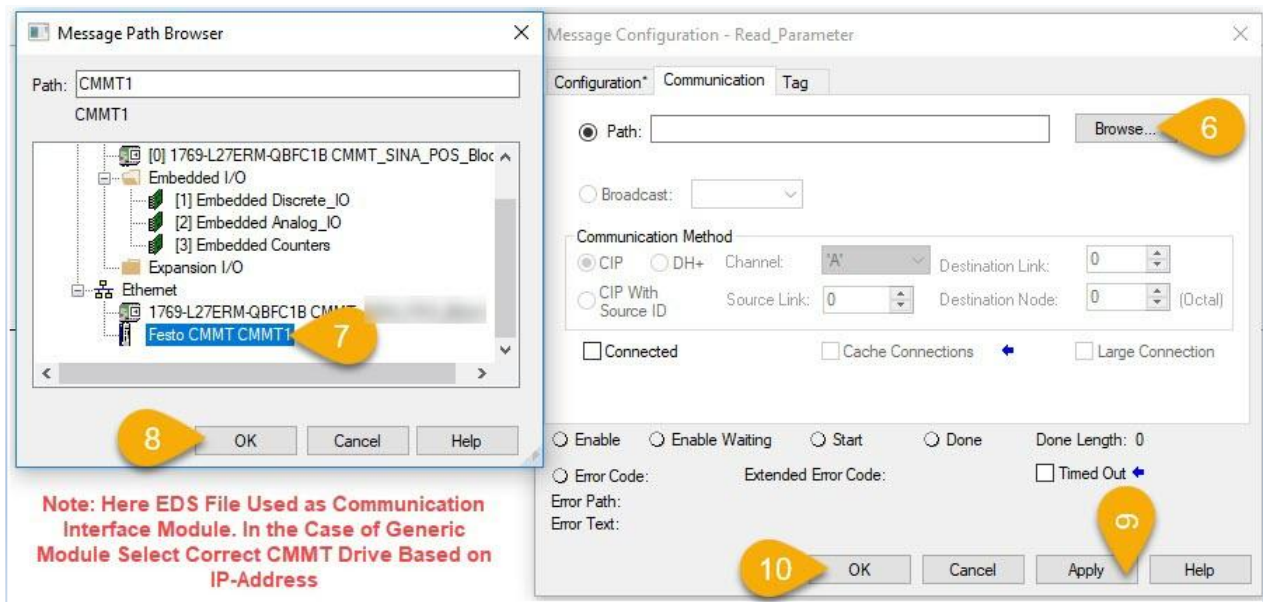
### 6.2.3 PTP\_Drives\_PARA\_Multi AOI Read/Write MSG Instruction Configuration

This chapter explain about read & write message instruction configuration.

#### Read MSG Instruction Configuration



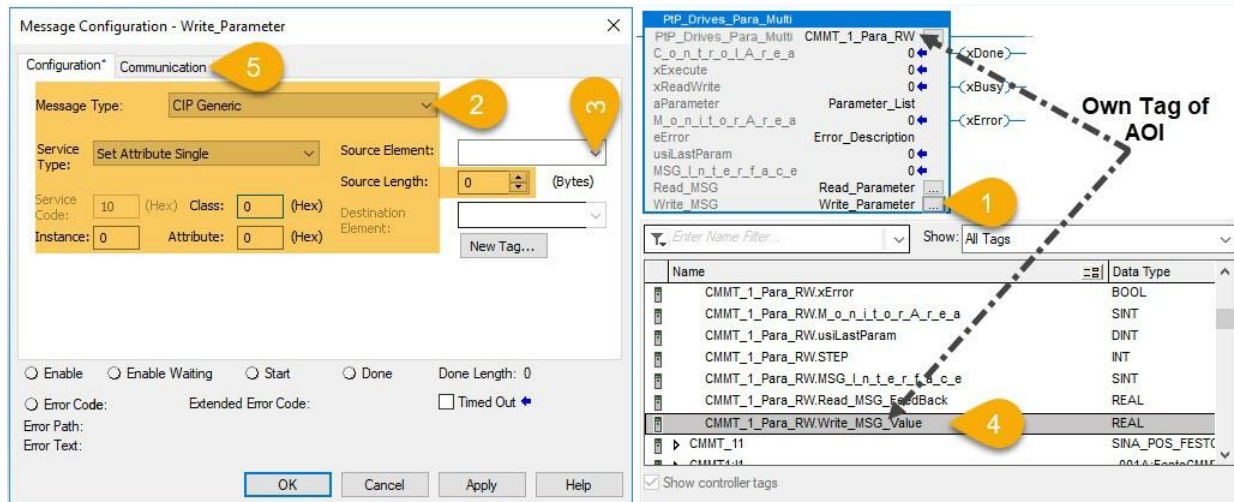
1. Click browse button of “**Read\_Parameter**” interface tag.
2. Set the yellow highlighted parameters as it is in above picture.
3. Click drop-down button of “**Destination Element**” to browse destination tag.
4. Browse the tag “**Read\_MSG\_FeedBack**” from its own AOI list & select it as destination tag.
5. Click communication tab to assign destination device.



**Note: Here EDS File Used as Communication Interface Module. In the Case of Generic Module Select Correct CMMT Drive Based on IP-Address**

6. Click “**Browse**” button from communication tab.
7. Browse the correct CMMT drive based on IP address in Message Path Browser.
8. Click “**OK**” button to confirm the communication module selection.
9. Click “**Apply**” button to confirm the configuration.
10. Click “**OK**” button to confirm & close the MSG instruction configuration.

## Write MSG Instruction Configuration

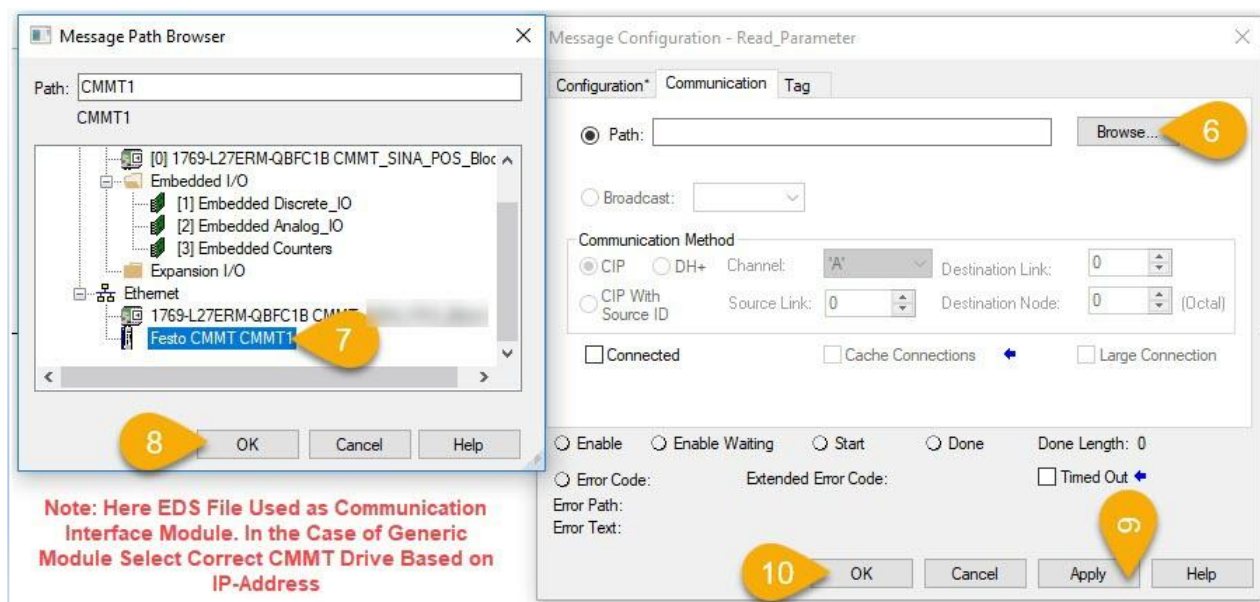


1. Click Browse button of “**Write\_Parameter**” interface tag.
2. Set the yellow highlighted parameters as it is in above picture.
3. Click drop-down button of “**Source Element**” to browse destination tag.
4. Browse the tag “**Write\_MSG\_Value**” from its own AOI tag list & select it as source tag.
5. Click communication tab to assign destination device.



### Note

- If CMMT drive firmware is V17.05.0 then user needs to follow the Rockwell TechNote ID:27896 for above configuration. The same TechNote PDF file has been attached with this application note folder in the name of “**Rockwell TechNote 27896**”.



6. Click “**Browse**” button from communication tab.
7. Browse the correct CMMT drive based on IP address in Message Path Browser
8. Click “**OK**” button to confirm the communication module selection
9. Click “**Apply**” button to confirm the configuration.
10. Click “**OK**” button to confirm & close the MSG instruction configuration.

**6.2.4 PTP\_DRIVES\_PARA\_MULTI AOI Control Area Tag Details**

| Control Area Tags                                                                                                                      | Data Type                                                                                                                           | Function Description                                                                                                                                                                                                                                                                                                          |
|----------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| xExecute                                                                                                                               | BOOL                                                                                                                                | Execute the task ( 1 – Execute )                                                                                                                                                                                                                                                                                              |
| xReadWrite                                                                                                                             | BOOL                                                                                                                                | Read or Write Selection<br>0 – Read<br>1 – Write                                                                                                                                                                                                                                                                              |
| aParameter <ul style="list-style-type: none"> <li>• uiParameter</li> <li>• uiIndex</li> <li>• pValue</li> <li>• uiValueSize</li> </ul> | UDT_CMMT_Parameter_PNUs[100] <ul style="list-style-type: none"> <li>• INT</li> <li>• INT</li> <li>• REAL</li> <li>• SINT</li> </ul> | User Defined DataType for Parametrization <ul style="list-style-type: none"> <li>• Parameter number</li> <li>• Index position of the parameter</li> <li>• Feedback value in read function &amp; Set value in write function</li> <li>• Data size of the parameter ( Only applicable for write parameter function )</li> </ul> |

**6.2.5 PTP\_DRIVES\_PARA\_MULTI AOI Monitor Area Tag Details**

| Area Tags    | Data Type | Function Description                                |
|--------------|-----------|-----------------------------------------------------|
| eError       | STRING    | Error message description                           |
| usiLastParam | DINT      | Last processed parameter                            |
| xDone        | BOOL      | Read or Write parameter list processed successfully |
| xBusy        | BOOL      | Read or Write function under process                |
| xError       | BOOL      | Read or Write function completed with unsuccessful  |

**6.2.6 PTP\_DRIVES\_PARA\_MULTI AOI MSG Interface Tag Details**

| Area Tags | Data Type | Function Description            |
|-----------|-----------|---------------------------------|
| Read_MSG  | MESSAGE   | Parameter read MSG instruction  |
| Write_MSG | MESSAGE   | Parameter write MSG instruction |

### 6.2.7 Example for Read Parameters Using PTP\_DRIVES\_PARA\_MULTI AOI



#### Note

- Before starting the read parameter configuration process download the CMMT user manual.

Here in this example we are going to read parameters “Debug variable index 0” & “Debug variable index 1” whose parameter number are P0.190.0.0 & P0.191.0.0 respectively. Below are the simple steps to configure these parameters.

- Ensure **xReadWrite** bit is in false state. ( Note: If **xReadWrite** bit is in true state, then it will be considered as write function )
- Open “**aParameter**” interface structure and fill the values. Parameters PNU values used to fetch data from the drive look like below.

| PNU    | Name                   | Data type     | Access | Parameter  |
|--------|------------------------|---------------|--------|------------|
| 2040.0 | Debug variable index 0 | FloatingPoint | rw     | P0.190.0.0 |
| 2041.0 | Debug variable index 1 | FloatingPoint | rw     | P0.191.0.0 |

| Name                          | Value | Style   | Data Type                    |
|-------------------------------|-------|---------|------------------------------|
| Parameter_List                |       | {...}   | UDT_CMMT_Parameter_PNUs[100] |
| Parameter_List[0]             |       | {...}   | UDT_CMMT_Parameter_PNUs      |
| Parameter_List[0].uiParameter | 2040  | Decimal | INT                          |
| Parameter_List[0].uiIndex     | 0     | Decimal | INT                          |
| Parameter_List[0].pValue      | 0.0   | Float   | REAL                         |
| Parameter_List[0].uiValueSize | 0     | Decimal | SINT                         |
| Parameter_List[1]             |       | {...}   | UDT_CMMT_Parameter_PNUs      |
| Parameter_List[1].uiParameter | 2041  | Decimal | INT                          |
| Parameter_List[1].uiIndex     | 0     | Decimal | INT                          |
| Parameter_List[1].pValue      | 0.0   | Float   | REAL                         |
| Parameter_List[1].uiValueSize | 0     | Decimal | SINT                         |

Feedback "pValue" will Update After Successful Execution  
 "uiValueSize" Only Used for Write Function

- Toggle the “**xExecute**” bit to trigger the read function. While the read function is active “**xBusy**” status will be true. If the entire parameter read process has completed successfully, the “**xDone**” bit will be high or If any error then “**xError**” bit will be high and error description will be on “**eError**” tag.
- After successful execution, Feedback values will be in “**pValue**” tag of each branch of parameter.



#### Note

- When ‘**uiParameter**’ tag is equal to zero, Then Program will consider list of parameter is completed. So, make sure there is no non-filled value in parameter list.
- Some parameters PNU value will be like this **2013. 0..2**. It means the length of the parameter is 3 array. So, user need to specify the correct element number in “**uiIndex**” tag. For an example if user wants to read second element of the parameter 2013, then he must enter the value 1 in “**uiIndex**” & 2013 in “**uiParameter**” tag.
- AOI considers the execute command when “**xExecute**” changes from OFF to ON.

### 6.2.8 Example for Write Parameters Using PTP\_DRIVES\_PARA\_MULTI AOI

Write function parameterization is like the read function. Below are some of the differences in write parameterization. Apart from below points, the remaining others are the same or like the read procedure.

- Ensure “**xReadWrite**” bit is in true state before execution.
- Enter the set value in “**pValue**” tag of each parameter element and also enter the data size of the parameter in “**uiValueSize**” (in Bytes). For **SINT – 1, INT – 2, REAL – 4 & DINT – 4**.

## 7 CMMT Drive Velocity Control Mode

The velocity control mode of CMMT drive has been designed with two different AOI's. They are named as “Tel001\_Velocity\_Drives\_Festo\_EIP” & “Tel102\_Velocity\_Drives\_Festo\_EIP”. The “Tel001\_Velocity\_Drives\_Festo\_EIP” AOI controls only speed of a servo motor with reference to its given speed setpoint but in “Tel102\_Velocity\_Drives\_Festo\_EIP” taking advantage by controlling both speed and torque of a servo motor.



### Note

- Refer [Chapter - 1.2](#) for Hardware & Software Requirements, [Chapter – 1.3](#) for Communication Architecture & [Chapter – 1.4](#) for CMMT Drive Hardware Overview.

### 7.1 Hardware Configuration for Velocity Control Mode



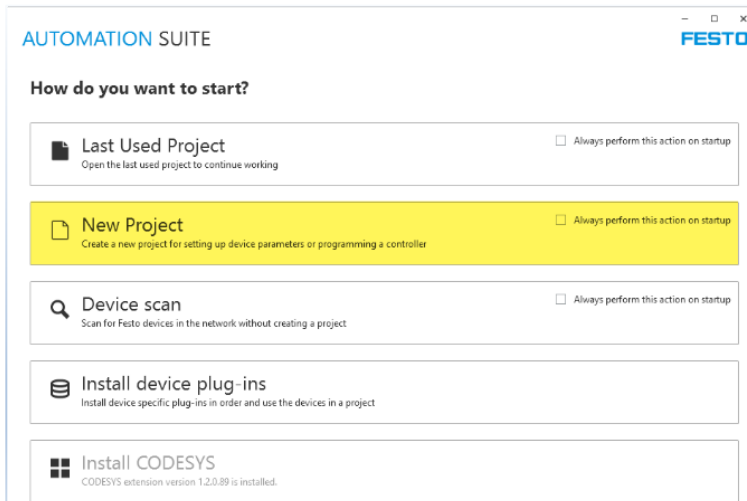
### Note

- Refer entire [Chapter – 2](#) for IP address configuration of Allen-Bradley PLC & CMMT servo drive.
- This chapter Hardware configuration procedures are similar for both Basic & Advance Velocity control modes.

#### 7.1.1 CMMT Servo Drive Configuration for Velocity Control Mode

This chapter describe about parameterization of CMMT drive for rotary axis in set RPM & current position as homing mode.

#### Step – 1



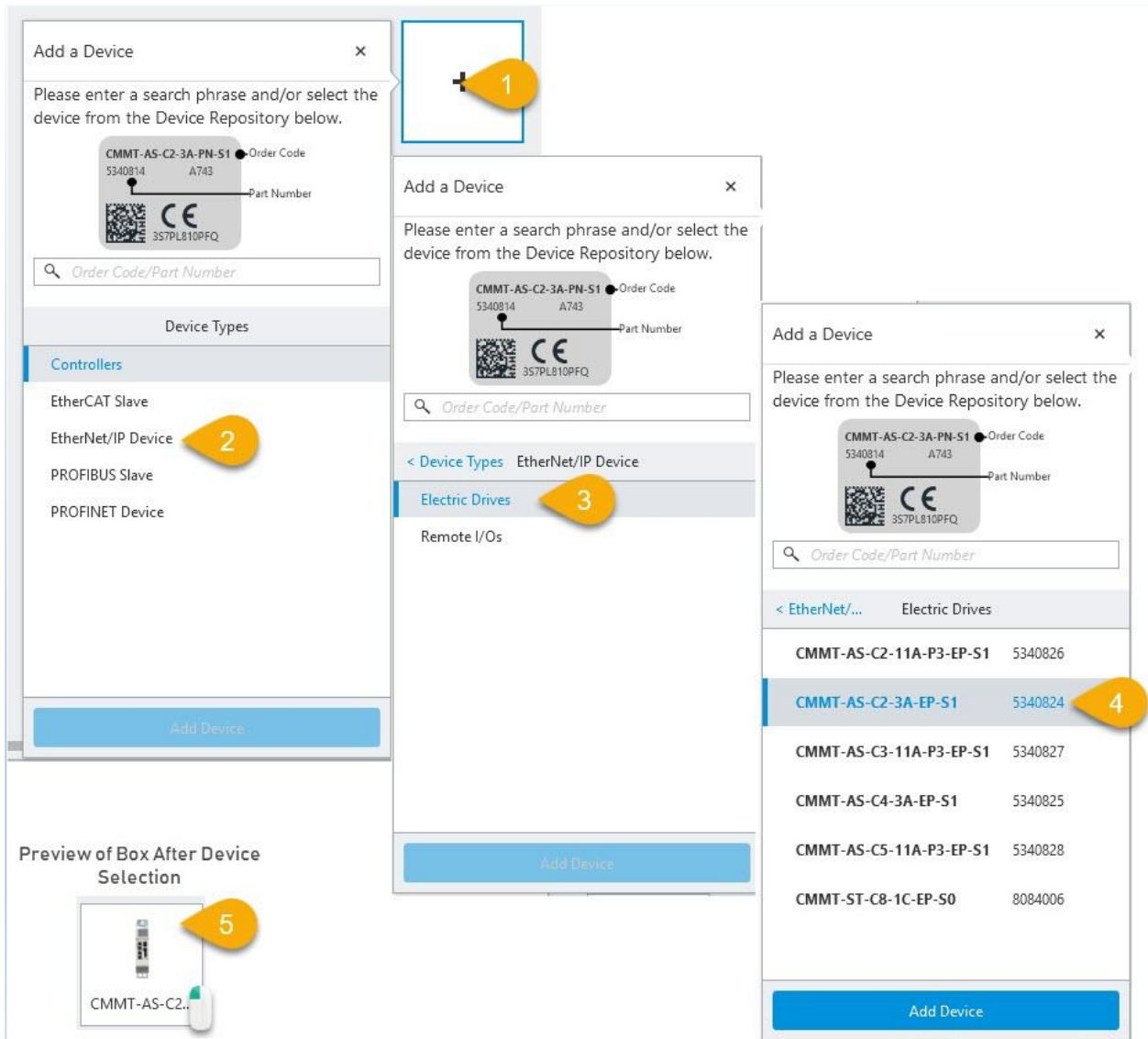
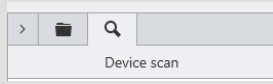
Open the Automation Suite software from Windows **Start > Festo Software > AUTOMATION SUITE** and select “New Project” from start-up menu.

## Step – 2

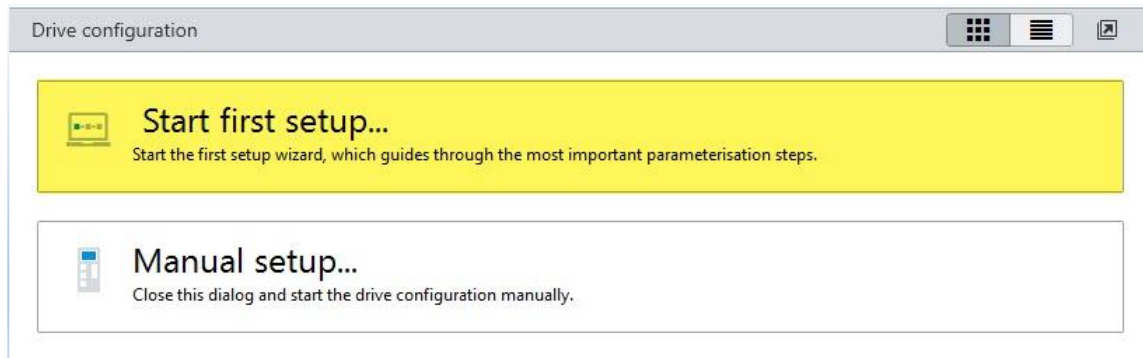


### Note

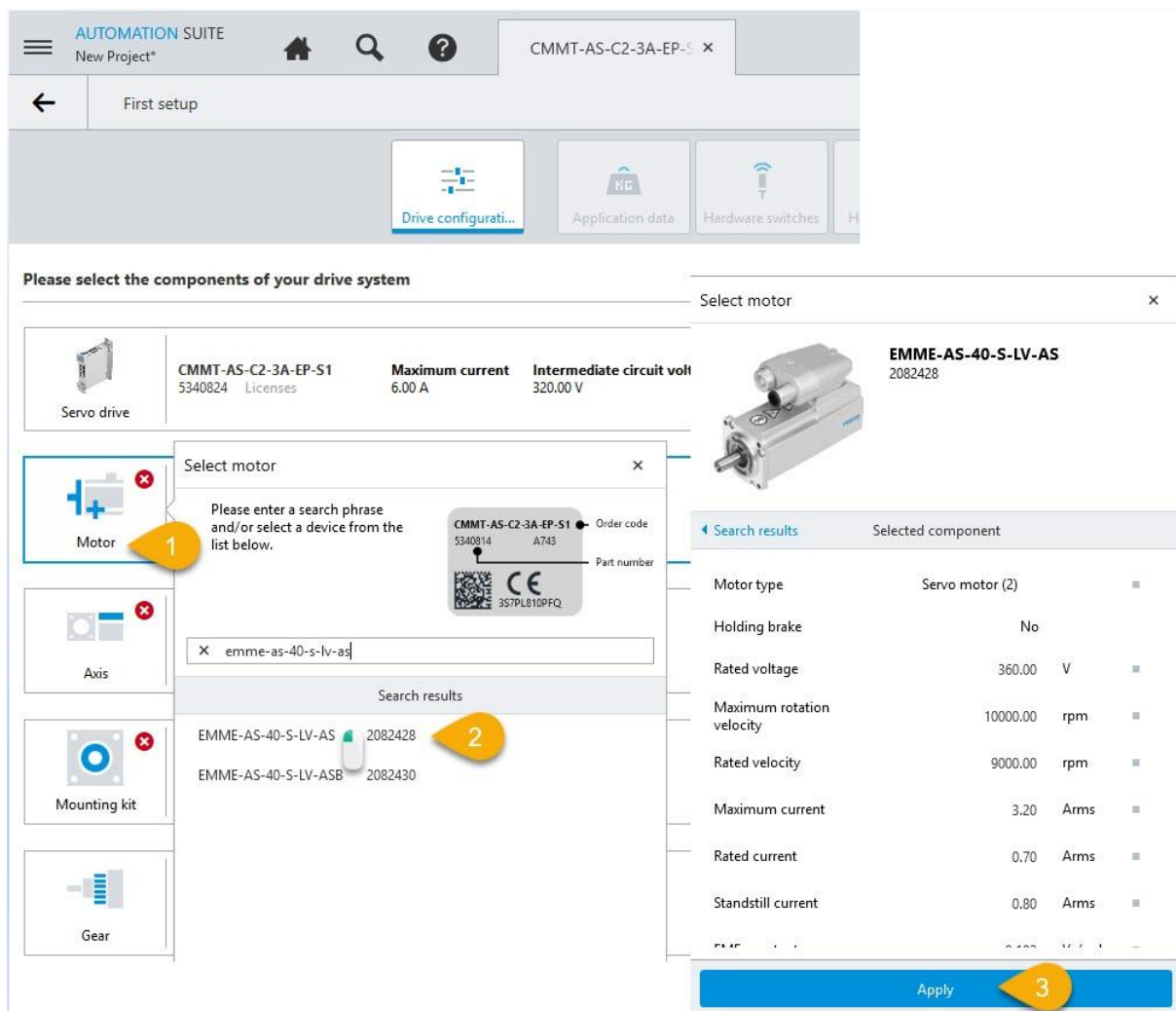
- Follow this step if drive is not in online. If the drive is connected with the PC as shown in [Chapter 2.3](#), simply click Device scan from home ( Next to device Repository ) and online devices will be visible in available devices list. Select the correct device based on Its configured name & IP address then double click.



- Select any empty device box and click add button.
- Select “**EtherNet/IP Device**” from Add a Device menu.
- Select “**Electric Drives**” from device type menu.
- Select correct drive part number** from electric drives list . As per this sample application CMMT-AS-C2-3A-EP-S1 is selected.
- Double click the newly added drive to initialize configuration parameters.

**Step – 3**

Select “**Start first setup**” from drive configuration menu.

**Step – 4**

1. Click “**Motor**” from components of drive system menu for specific motor model.
2. **Select correct motor** based on its **part number** and double click. As per this sample application “EMME-AS-40-S-LV-AS” motor is used.
3. Cross verify the motor parameters & click apply button to ensure motor configuration.

## View of motor tab after configuration

|                                                                                   |                               |                         |                     |                                   |                                 |                     |                                                                                                                                                                         |
|-----------------------------------------------------------------------------------|-------------------------------|-------------------------|---------------------|-----------------------------------|---------------------------------|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  | EMME-AS-40-S-LV-AS<br>2082428 | Type<br>Servo motor (2) | Holding brake<br>No | Encoder protocol<br>HiPerface (0) | Encoder type<br>Single turn (1) | Voltage<br>360.00 V |   |
| Motor                                                                             |                               |                         |                     |                                   |                                 |                     |                                                                                                                                                                         |

## Step – 5

Please select the components of your drive system



Servo drive

**CMMT-AS-C2-3A-EP-S1**  
5340824 Licenses

**Maximum current**  
6.00 A

**Intermediate circuit v**  
320.00 V



Motor

**EMME-AS-40-S-LV-AS**  
2082428

Type  
Servo motor (2)

Holding brake  
No

Encoder  
HiPer



Axis

Please enter a search phrase and/or select a device from the list below.



Order code  
5340814 A743

Part number  
357PL810PFQ

Order code/part number

Search results

User defined linear axis

User defined rotative axis 

DGE-8-%%-ZR 193739



Mounting kit



Axis



User defined rotative axis

Search results

Selected component

Current user unit  
Rev [rev, rpm, ...] (3)

Motion  
Rotative

Unlimited axis ☒ Active

Design axis  
Single axis (0)

Apply

This step describes about the axis type configuration. As specified initially here axis configuration is done for rotate the axis in velocity mode for set RPM.



### Note

- Make sure the “**Unlimited axis**” is activated as highlighted in yellow color. This ensures the axis does not have position limits.

**Step – 6**

**Select mounting kit**

Please enter a search phrase and/or select a device from the list below.

Search results

| Search results            |         |
|---------------------------|---------|
| User defined mounting kit |         |
| EAMM-A-A19-40A            | 559798  |
| EAMM-A-A19-42A            | 558895  |
| EAMM-A-A22-55A            | 559799  |
| EAMM-A-A22-57A            | 558897  |
| EAMM-A-A22-70A            | 558898  |
| EAMM-A-D100-100A          | 3356796 |
| EAMM-A-D100-100A-S1       | 3356931 |

**User defined mounting kit**

Selected component

Type: Inertia Coupling

Axial: 10 kgm<sup>2</sup>

Apply

Back Next Finish

Gear ratio (total): 1:1

New Project: CMMT-AS-C2-3A-EP-S1 (Plug-in: CMMT-AS Plug-in V1.2.2.4) Festo Automation Suite V1.2.1.5

This step illustrates about mounting kit configuration. This configuration will vary depending on user application type. At the end of this step click next button to move to next level configuration.

**Step – 7**

**Application data**

Axis mass moment of inertia: 0.00 kgcm<sup>2</sup>

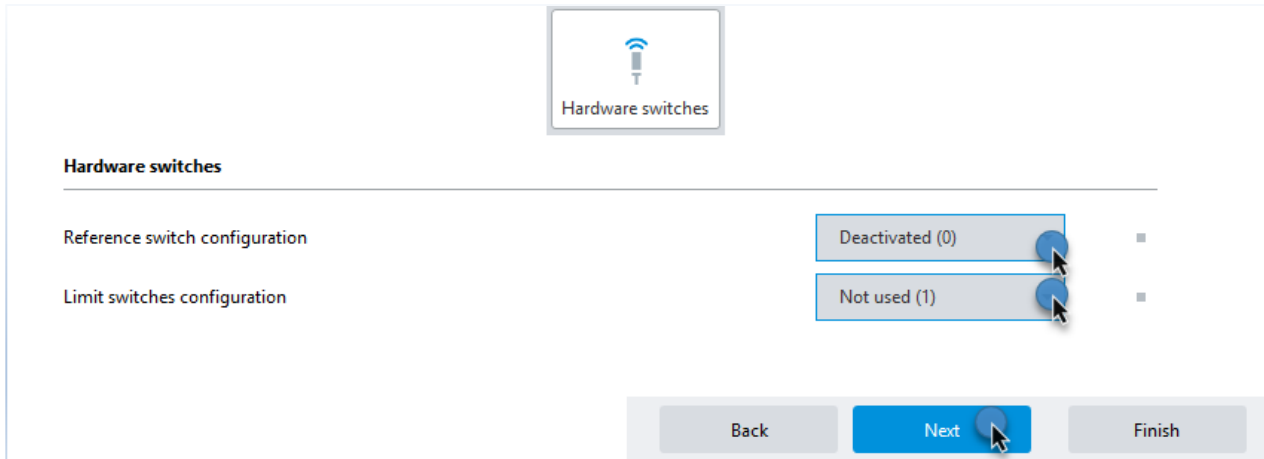
Application moment of inertia: 1.00 kgcm<sup>2</sup>

Total mass moment of inertia: 1.00 kgcm<sup>2</sup>

Back Next Finish

Specify inertia value of user application. This parameter value will vary depend on application type. Click next button for next level configuration.

### Step -8



**Hardware switches**

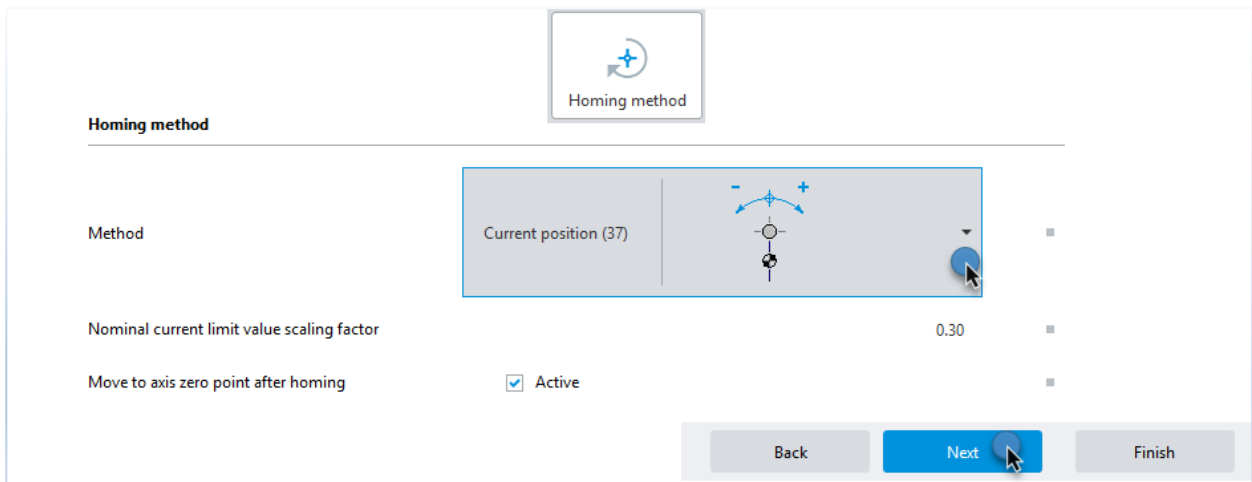
Reference switch configuration: Deactivated (0)

Limit switches configuration: Not used (1)

Back Next Finish

Select the reference & limit switch configuration types. This sample application has both disabled. This is the reason “Current Position (37)” homing mode selected for this sample application. Click next button for next level configuration.

### Step – 9



**Homing method**

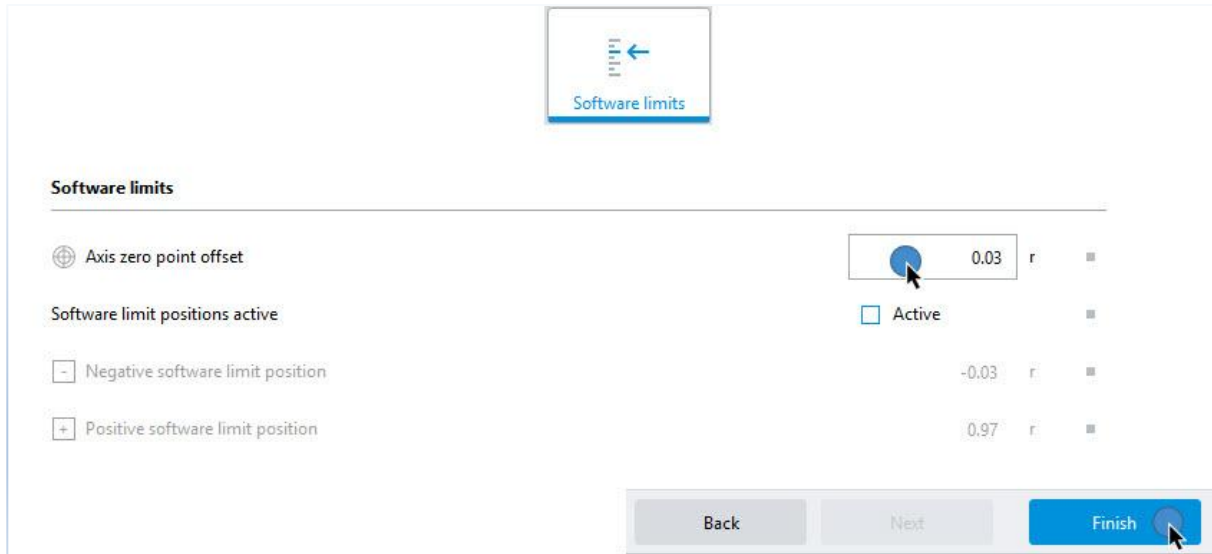
Method: Current position (37)

Nominal current limit value scaling factor: 0.30

Move to axis zero point after homing: ☒ Active

Back Next Finish

Select homing method. As per this sample application “**Current Position (37)**” mode is selected. But user want to select homing mode depending on application requirement. Click next button for next level of configuration.

**Step – 10**


**Software limits**

|                                  |                                 |   |  |
|----------------------------------|---------------------------------|---|--|
| Axis zero point offset           | 0.03                            | r |  |
| Software limit positions active  | <input type="checkbox"/> Active |   |  |
| Negative software limit position | -0.03                           | r |  |
| Positive software limit position | 0.97                            | r |  |

Back Next **Finish**

Here user can select offset position of home reference point. This parameter is not valid for velocity mode, so simply leave default mode & click **“Finish”** button to complete the configuration.

**Note**

- Don't activate the “Software Limit Positions Active” option.

**Step – 11**

At this stage all the fundamental configurations of drive system are done. In case of any correction in parameter is required software tool will indicate number of **“Correct parameters”** ( Right most top corner in FAS tool ) as shown in the below picture. So to correct parameter value click the **“Correct parameters”** button and click apply button to apply system generated recommended values.



### 7.1.2 CMMT Drive Parameterisation for Integrate with PLC in Velocity Control Mode

This chapter will describe about parameterisation of CMMT drive to establish communication with PLC through Telegram-1 & Telegram-102. Also it describes some additional parameters like Factor group, Reference values, Ethernet/IP interface.



#### Note

- This chapter explains parameterisation for both Basic Speed Control & Advance Speed Control modes .

#### Step – 1



#### Note

- Follow **step – 1** when drive is available for online connection. If drive is not available simply skip this step and jump to next step.

The screenshot displays the 'PARAMETERISATION' tab of the CMMT drive software. The main window shows the drive 'CMMT1' (CMMT-AS-C2-3A-EP-S1) with its path '192.150.3.7' and a 'Disconnected' status. A 'Connect' button is visible. A yellow callout bubble with the number '3' points to the 'Drive communication' tab. Below the main window, a section titled 'Status When Drive is in On-line' shows the drive 'CMMT1' with its path '192.168.1.7' and a 'Connected' status. A 'Disconnect' button is also present. A yellow callout bubble with the number '2' points to a blue button labeled 'Use IP Address for Device Communication' at the bottom right. On the right side, a 'Device Communication' dialog box is open, showing the IP address '192.168.1.7' and a table of available devices.

| Status                              | Identify                 | Device Name | Device Type | IP Address  |
|-------------------------------------|--------------------------|-------------|-------------|-------------|
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | CMMT1       | CMMT-AS     | 192.168.1.7 |

1. Click the IP address “**Path**” from tool bar.
2. Select the correct drive based on IP address & select “**Use IP Address for Device Communication**” button to assign path address.
3. Click on “**Connect**” button to establish communication & status of the drive changes from Disconnected to Connected.

**Step – 2**

| Factor group      |                                 |   |
|-------------------|---------------------------------|---|
| Current user unit | Rev [rev, rpm, ...] (3)         | ■ |
| Position          | <input type="text" value="-6"/> | ■ |
| Velocity          | <input type="text" value="-3"/> | ■ |

Enter the Velocity command scale factor value of fieldbus as highlighted in yellow color field. Here scale factor is set for 1000. User can reach the Factor group in Automation Suite from **PARAMETERISATION > Fieldbus > Factor group**.

Example - 1 PLC scale value is calculated below for 1 RPM speed.

Velocity Factor (-3) =  $10^{-3} = 0.001 \text{ RPM}$  Then for 1 RPM = 1000 Increment value in PLC

Example - 2 PLC scale value is calculated below for 100 RPM speed.

Velocity Factor (-3) =  $10^{-3} = 0.001 \text{ RPM}$  Then for 100 RPM = 100000 Increment value in PLC

**Note**

- For more details about fieldbus scaling refer chapter name “**Scaling of internal units for the fieldbus**” in user manual.
- Different user unit can't be set for Velocity, Acceleration & Jerk.
- The maximum value limit for 32 bit DINT data type variable is 2,147,483,647. So, plan scaling factor within the limit.

**Step – 3****Dynamic Values**

Coast stop

|                                                                                                        |                                        |                    |
|--------------------------------------------------------------------------------------------------------|----------------------------------------|--------------------|
| Acceleration                                                                                           | <input type="text" value="3000.00"/>   | r/min/s            |
| Deceleration                                                                                           | <input type="text" value="3000.00"/>   | r/min/s            |
| Jerk                                                                                                   | <input type="text" value="300000.00"/> | rpm/s <sup>2</sup> |
| Deceleration (system stop AC1/AC3)                                                                     | <input type="text" value="600.00"/>    | r/min/s            |
| Jerk (system stop AC1/AC3)                                                                             | <input type="text" value="600000.00"/> | rpm/s <sup>2</sup> |
| Deceleration (stop ramp)                                                                               | <input type="text" value="12000.00"/>  | r/min/s            |
| Jerk (stop ramp)                                                                                       | <input type="text" value="600000.00"/> | rpm/s <sup>2</sup> |
| Base value acceleration                                                                                | <input type="text" value="600.00"/>    | r/min/s            |
| Base value deceleration                                                                                | <input type="text" value="600.00"/>    | r/min/s            |
| Base value velocity  | <input type="text" value="3000.00"/>   | r/min              |
| Base value velocity (PLC, motorside)                                                                   | <input type="text" value="3000.00"/>   | r/min              |

Enter the actual Base value velocity, Acceleration & Deceleration values of the application in the respective numeric fields. The base value velocity specifies the base value for all Application Classes. The base value in user units (drive side without gearbox) is multiplied by the normalized value in the process data and then results in the internal velocity setpoint.

**Information**

- As a recommendation, the base value should be half of the maximum velocity in order to use the full value range 0..200% from the control.
- Base value velocity does not affect the speed of the drive, but it will effect on display value of the drive.
- The AOI tag “**BaseValueVelocity**” tag must have same value as the FAS parameter value that is (FAS = P1.11280701.0.0).

**Step – 4**

| EtherNet/IP interface |                                 |
|-----------------------|---------------------------------|
| <b>Configuration</b>  |                                 |
| Activate DHCP         | <input type="checkbox"/> Active |
| IP address            | 192 . 168 . 1 . 5               |
| Subnet mask           | 255 . 255 . 255 . 0             |
| Gateway address       | 0 . 0 . 0 . 0                   |

Ether the IP address & Subnet mask for EtherNet/IP interface. This communication interface is used to establish connection with PLC to receive control command & to send signal feedbacks.

**Note**

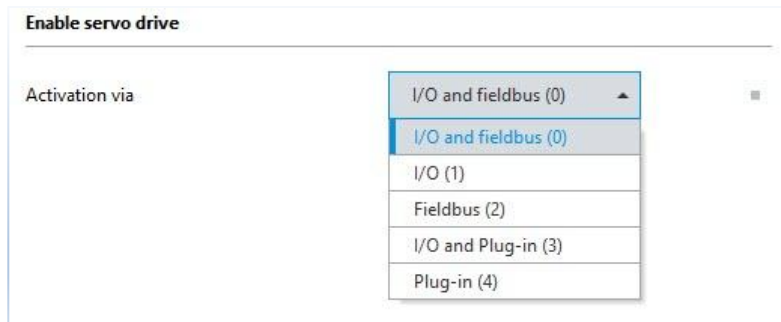
- IP address series of EtherNet/IP interface must match with PLC IP address series to establish connection.

**Step – 5**

| Telegram - EtherNet/IP - ModbusTCP |                         |
|------------------------------------|-------------------------|
| Tel001_Velocity_Drives_Festo_EIP   |                         |
| Telegram selection                 | Telegram (1)            |
| Actual application class           | Application class 1 (1) |
| Telegram - EtherNet/IP - ModbusTCP |                         |
| Tel102_Velocity_Drives_Festo_EIP   |                         |
| Telegram selection                 | Telegram (102)          |
| Actual application class           | Application class 1 (1) |

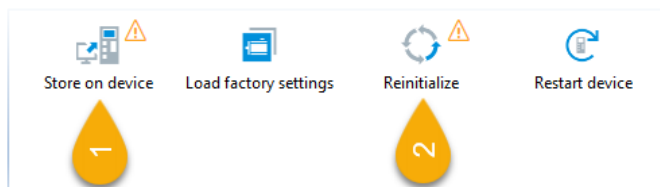
Select the connection properties as “**Telegram(1)**” for “Tel001\_Velocity\_Drives\_Festo\_EIP” AOI as highlighted in yellow color & select “**Telegram (102)**” for “Tel102\_Velocity\_Drives\_Festo\_EIP” AOI as highlighted in red color. User can reach Connection properties from **PARAMETERISATION > Fieldbus > Connection properties**.

### Step – 6



Select the medium to enable the servo drive. The selection must be either **“I/O and fieldbus (0)”** or **“Fieldbus”** to control the drive through fieldbus.

### Step – 7



1. Click the **“Store on device”** button to store all the parameters on drive memory.
2. Click the **“Reinitialize”** the button to initialize the drive.



#### Note

- The triangle warning symbol on both buttons indicates action required.
- All the modified parameters will automatically erase from drive during power restart If modified parameters are not stored in the drive.
- Its mandatory to reinitialize the drive for some parameter modification. If drive is not reinitialized the drive performance remains same as like previous configuration.

## 7.2 Software Configuration for CMMT Velocity Control Mode

This chapter describes about software configuration for Basic & Advance Velocity Control Modes like AOI's import & Ethernet/IP interface module creation.

### 7.2.1 Add CMMT EDS File in Studio-5000 Ethernet Tree for Hardware Interface



#### Note

- Follow the [chapter 3.2](#) for adding the EDS file into the project.
- When using EDS file as communication interface, IO size will automatically accept for Telegrams 1, 102 & 111.

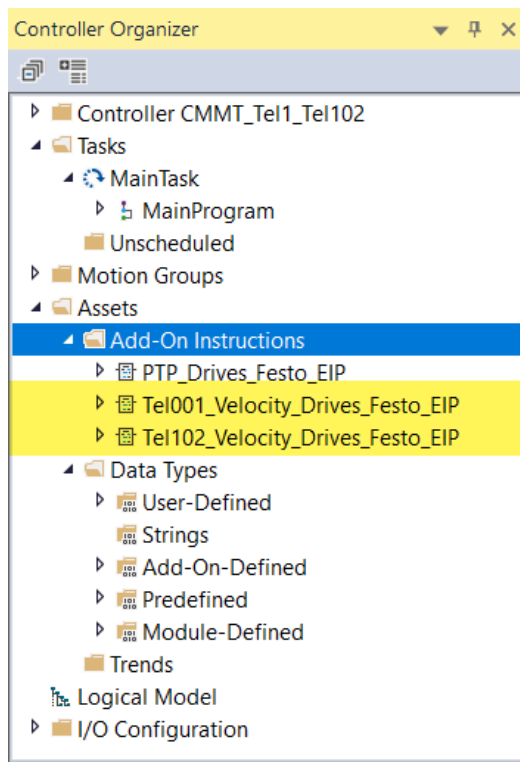
### 7.2.2 Import Tel001\_Velocity\_Drives\_Festo\_EIP & Tel102\_Velocity\_Drives\_Festo\_EIP AOI's in Studio – 5000



#### Note

- Follow the [chapter – 3.4](#) for import Basic & Advance Velocity control AOI's in Studio – 5000 project file. There AOI importation is done for control block of CMMT but users have to select velocity blocks instead of control block.

After successful importation of velocity AOI's. It will be available in Add-On Instruction folder as highlighted in blow picture.



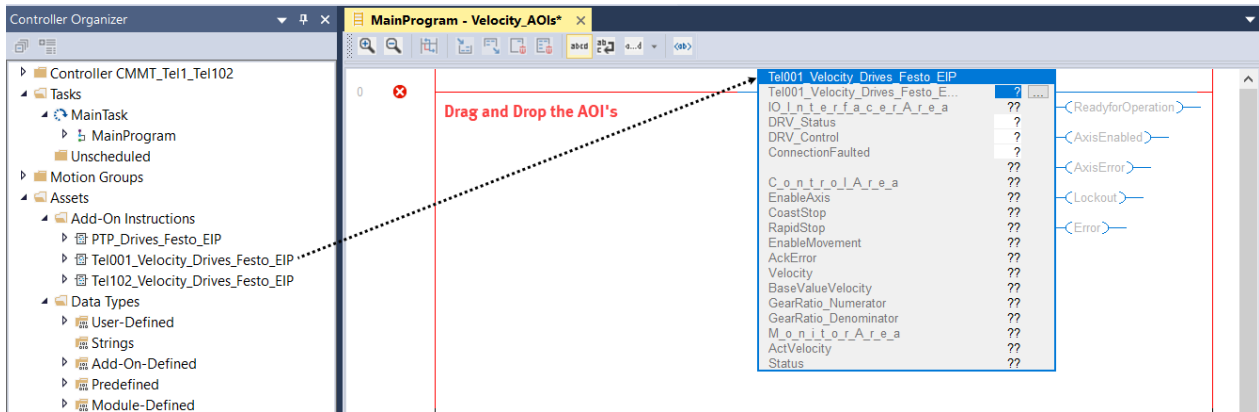
### 7.2.3 Add “Tel001\_Velocity\_Drives\_Festo\_EIP” AOI in PLC Program



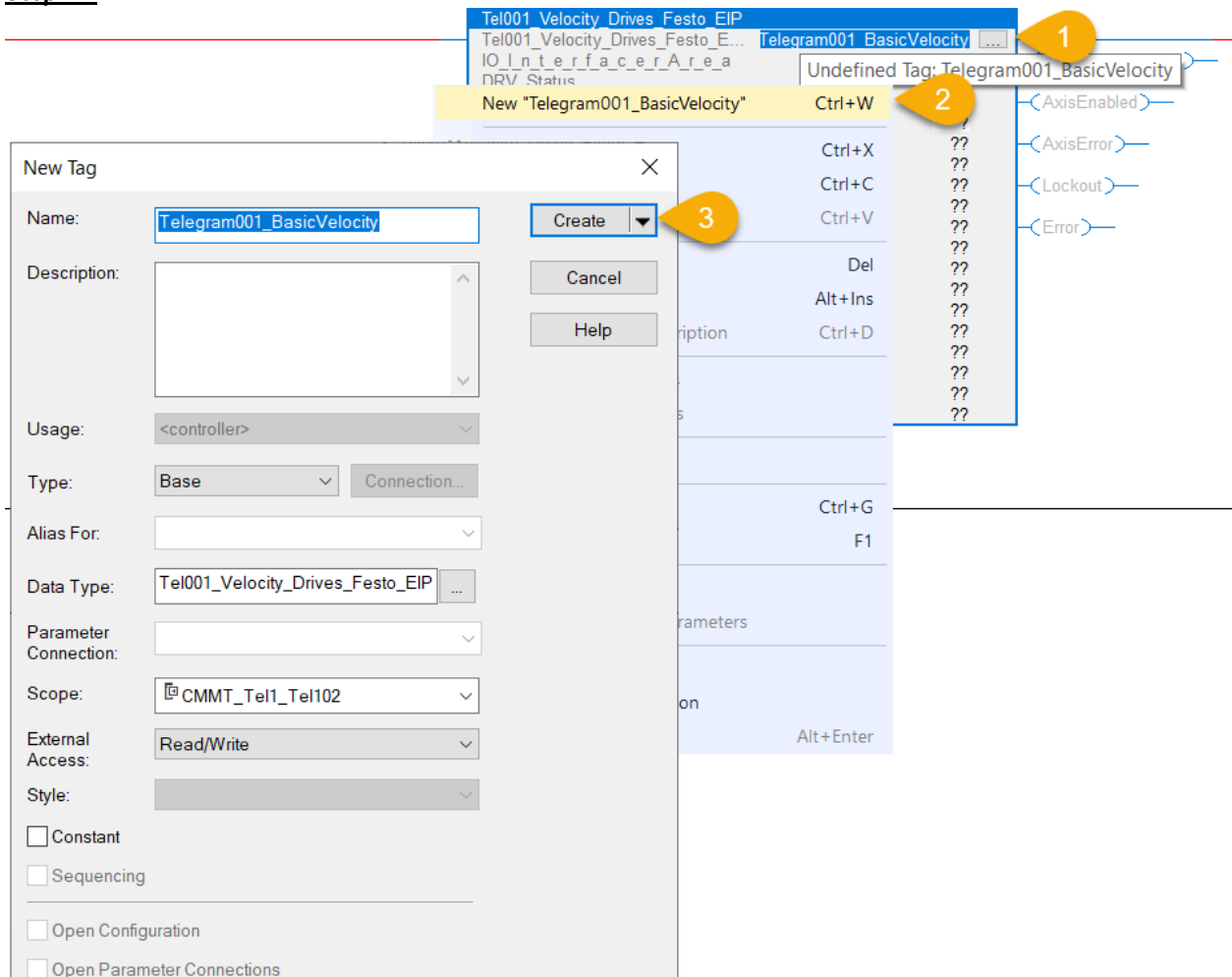
#### Note

- The Add-On Instruction must be called cyclically with a separate instance for every servo drive integrated in the user program (every axis). Simultaneous use of multiple instances of the same device is not permitted.
- Any AOI's need to be in a continuous task called cyclically. ([Refer Chapter 3.5 Step – 1](#))

#### Step – 1

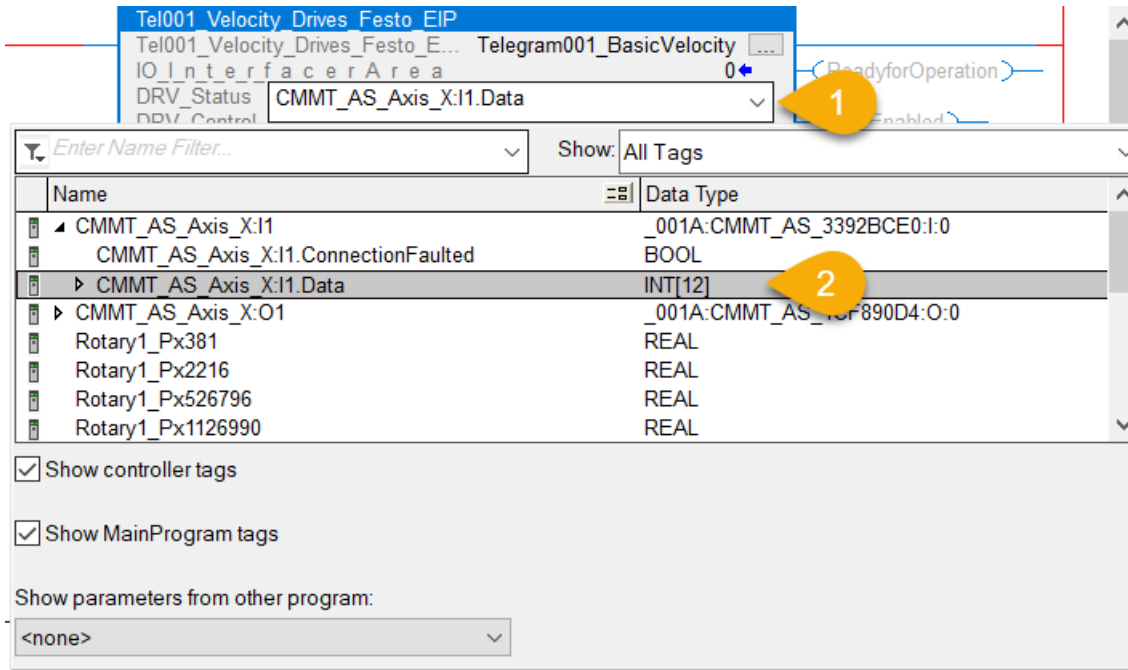


Drag and drop the “Tel001\_Velocity\_Drives\_Festo\_EIP” AOI file from Add-On Instruction folder to empty rung of project.

**Step – 2**

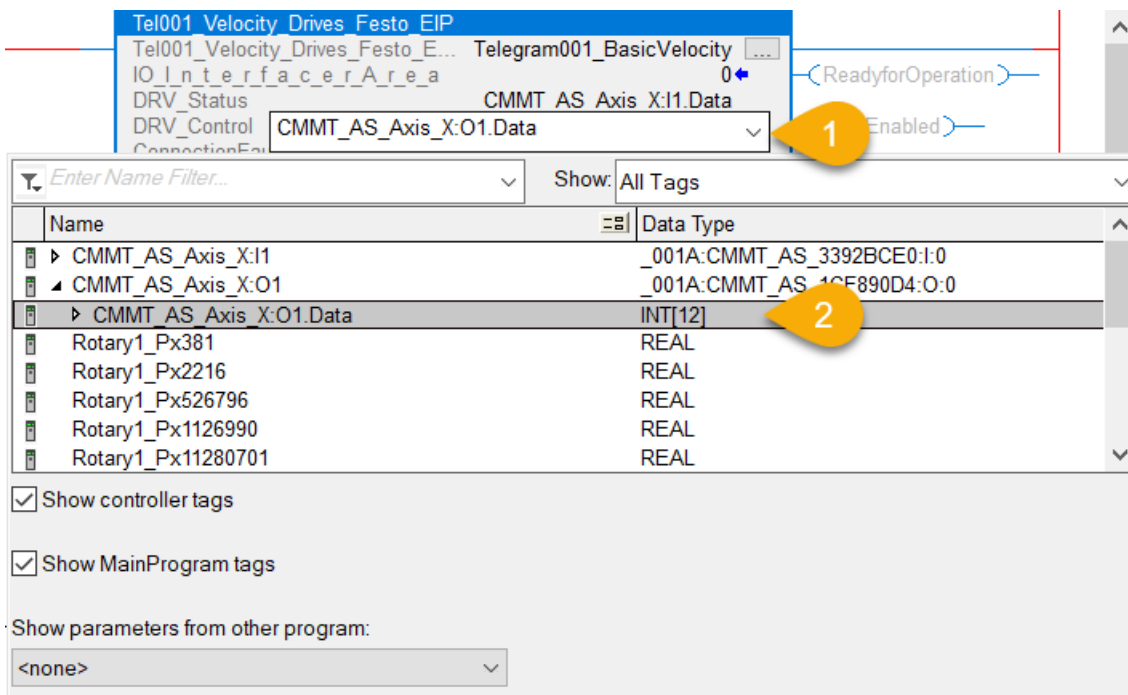
1. Enter the tag name for the AOI & right click.
2. Select the new option with name of the tag from right click menu.
3. Click **"Create"** button from new tag window.

### Step – 3

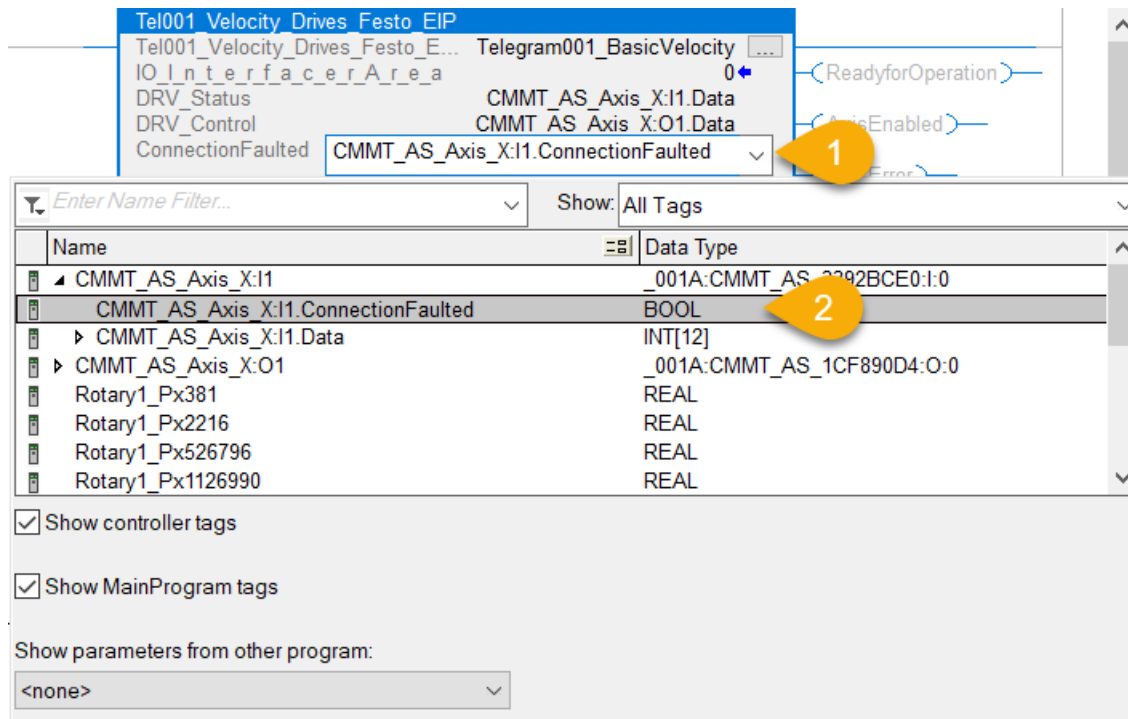


1. Select drop-down of “**DRV\_Status**” from IO interface tags.
2. Select input data structure of CMMT Ethernet interface module which created in [chapter-7.2.1](#).

### Step – 4

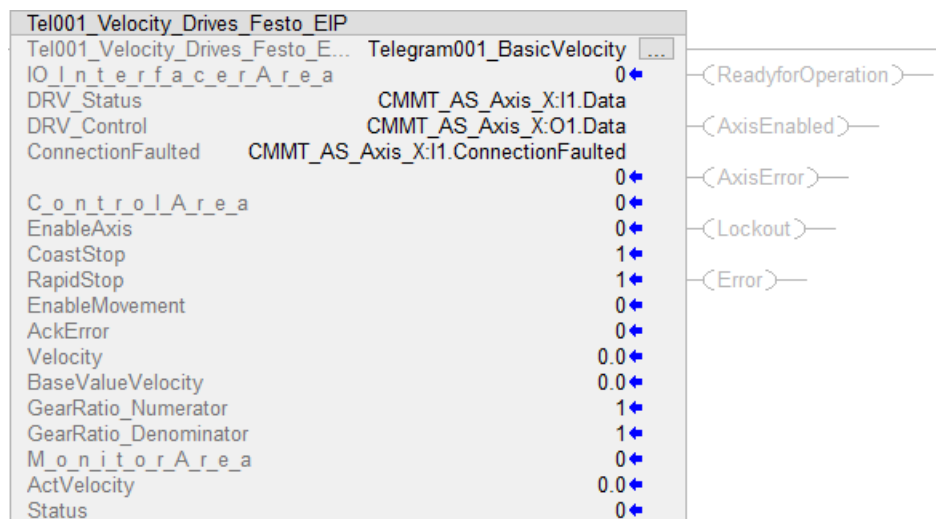


1. Select drop-down of “**DRV\_Control**” from IO interface tags.
2. Select output data structure of CMMT Ethernet interface module which created in [chapter-7.2.1](#).

**Step – 5**

1. Select drop-down of “**ConnectionFaulted**” from IO interface tags.
2. Select input data structure of CMMT Ethernet interface module which created in [chapter-7.2.1](#).

AOI's look as shown in below picture, once all the configurations are done.



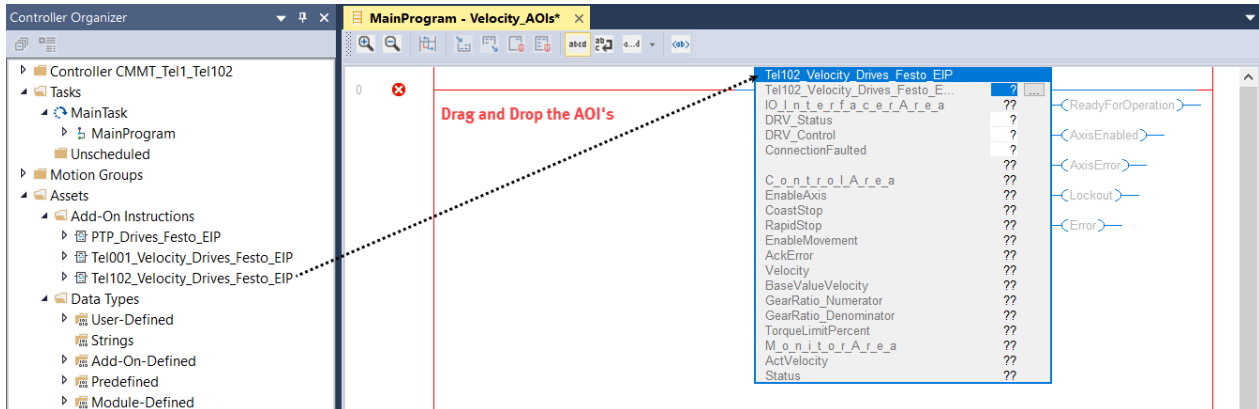
#### 7.2.4 Add “Tel102\_Velocity\_Drives\_Festo\_EIP” AOI in PLC Program



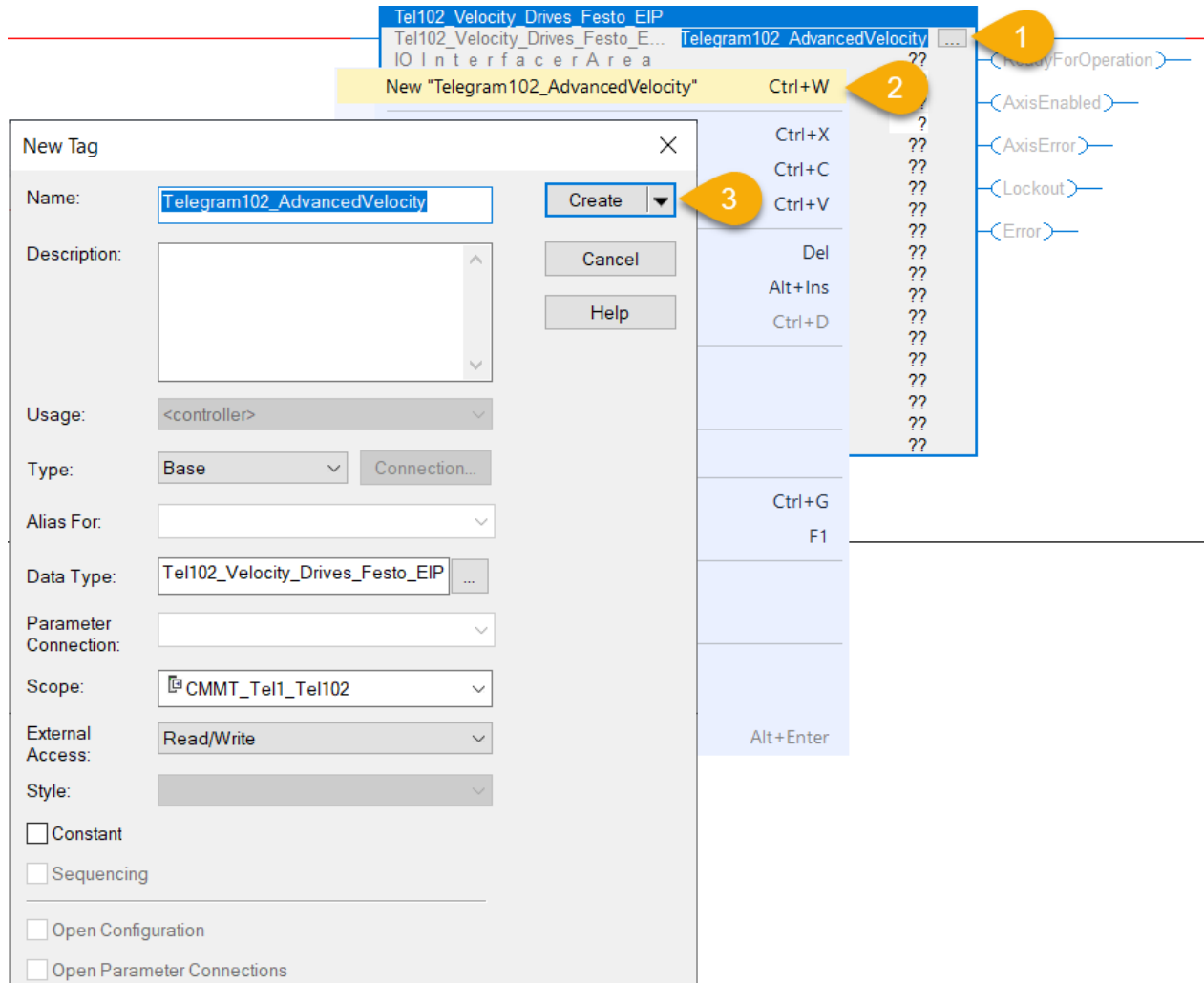
##### Note

- The Add-On Instruction must be called cyclically with a separate instance for every servo drive integrated in the user program (every axis). Simultaneous use of multiple instances of the same device is not permitted.
- Any AOI's need to be in a continuous task called cyclically. ([Refer Chapter 3.5 Step – 1](#))

##### Step – 1

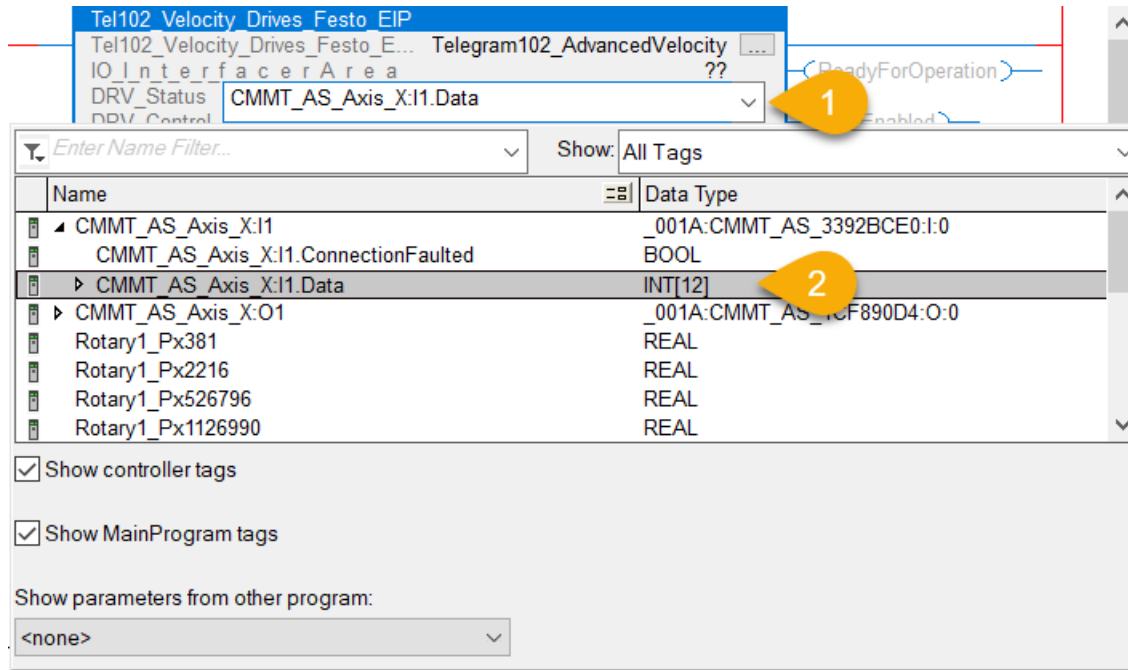


Drag and drop the “Tel102\_Velocity\_Drives\_Festo\_EIP” AOI file from Add-On Instruction folder to empty rung of project.

**Step – 2**

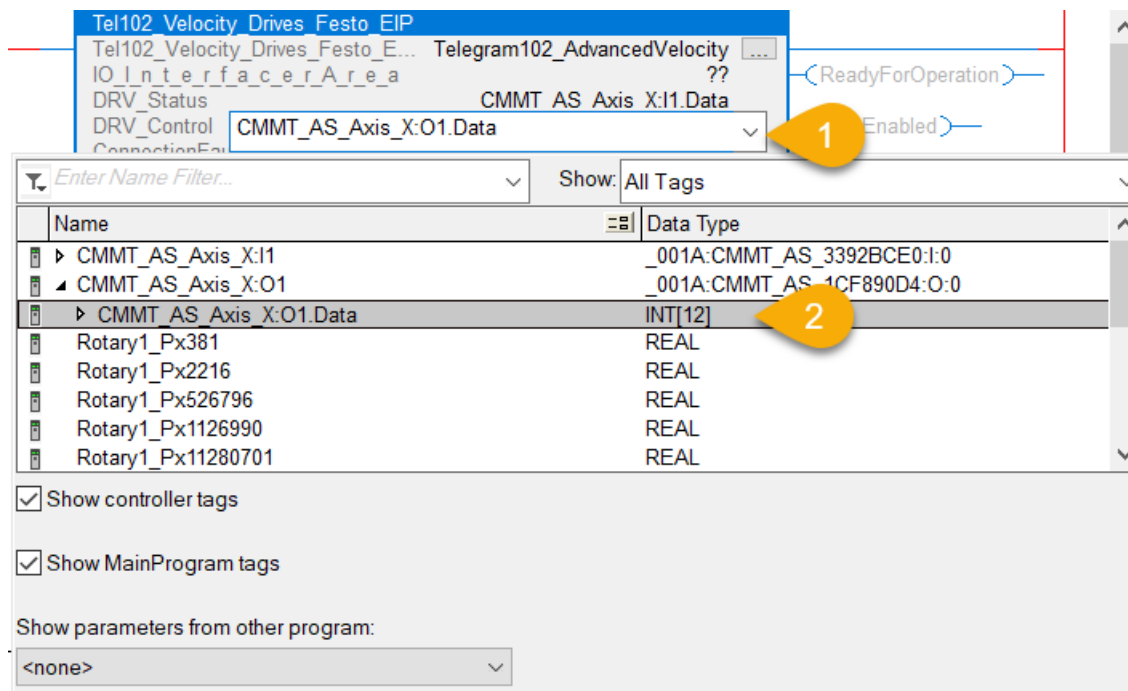
1. Either the tag name for the AOI & right click.
2. Select the new option with name of the tag from right click menu.
3. Click **“Create”** button from new tag window.

### Step – 3

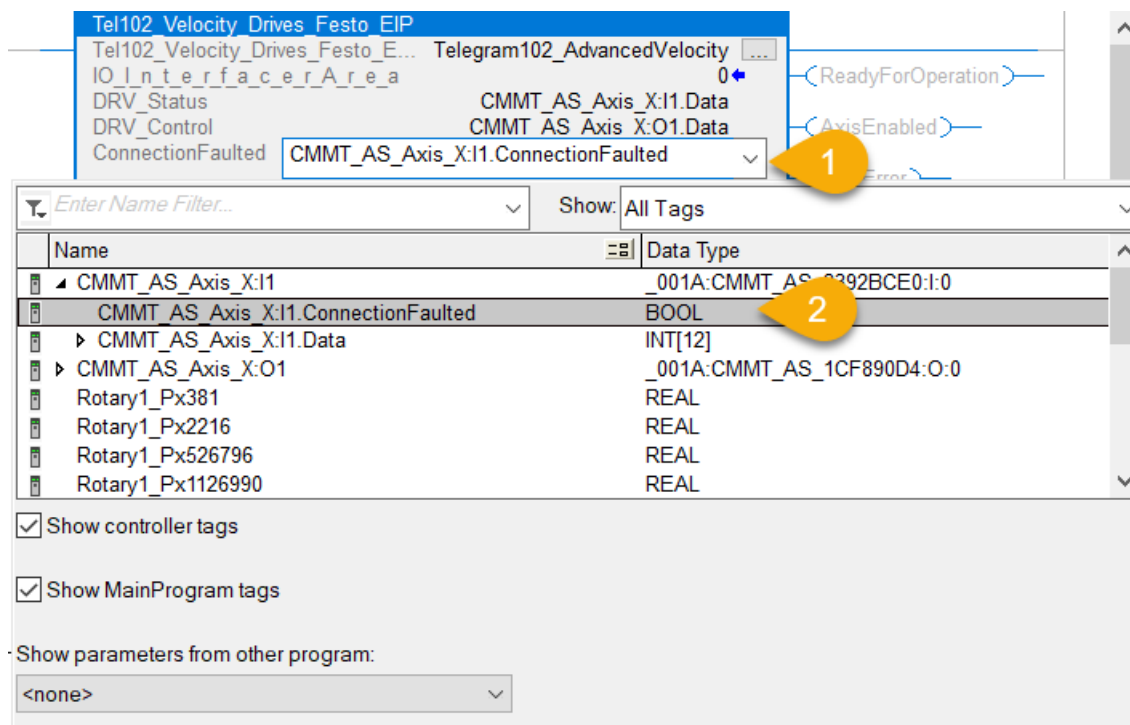


1. Select drop-down of “**DRV\_Status**” from IO interface tags.
2. Select input data structure of CMMT Ethernet interface module which created in [chapter-7.2.1](#)

### Step – 4

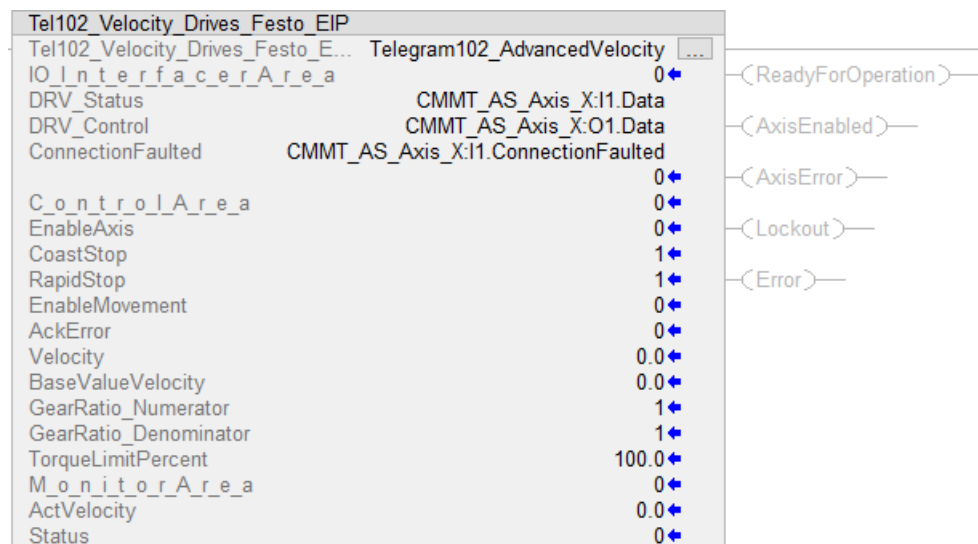


1. Select drop-down of “**DRV\_Control**” from IO interface tags.
2. Select output data structure of CMMT Ethernet interface module which created in [chapter-7.2.1](#).

**Step – 5**

1. Select drop-down of “**ConnectionFaulted**” from IO interface tags.
2. Select input data structure of CMMT Ethernet interface module which created in [chapter-7.2.1](#).

AOI's look as shown in below picture, once all the configurations are done.



## 8 “Tel001\_Velocity\_Drives\_Festo\_EIP” AOI / Basic Velocity Control

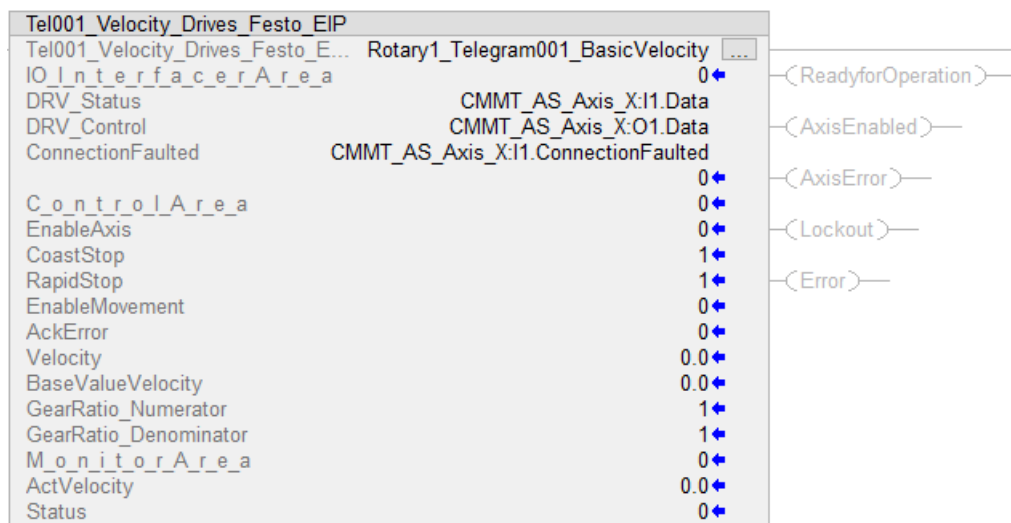
### 8.1 “Tel001\_Velocity\_Drives\_Festo\_EIP” AOI Introduction

The drive is controlled by a main setpoint (Velocity Setpoint) via Telegram 1, the velocity is controlled completely in the drive.

The “Tel001\_Velocity\_Drives\_Festo\_EIP” AOI is used to control the CMMT drive in velocity mode

. This AOI controls the drive by toggling appropriate control bits & speed has been controlled by setting RPM value in specified control word. This AOI controls the enabling of the axis, coast stop, rapid stop & drive fault reset.

“Tel001\_Velocity\_Drives\_Festo\_EIP” AOI is also designed to update the feedback status like Drive Enabled , Drive Ready, Drive Fault is Active, Drive in Inhibited status & Actual axis RPM feedback.



### 8.2 IO Interface Area Tag Details

| Interface Area Tags | Data Type             | Function Description              |
|---------------------|-----------------------|-----------------------------------|
| DRV_Status          | ARRAY OF INT [ 0.. 1] | Ethernet Module Status Variables  |
| DRV_Control         | ARRAY OF INT [ 0.. 1] | Ethernet Module Control Variables |

### 8.3 Control Area Tag Details

| Control Area Tags | Data Type | Meaning                                                                       |
|-------------------|-----------|-------------------------------------------------------------------------------|
| ConnectionFaulted | BOOL      | Communication disconnected acknowledgement for AOI                            |
| EnableAxis        | BOOL      | Drive Enable ( ON/OFF1 )<br>1 = Enable Drive<br>0 = Ramp Down & Disable drive |
| CoastStop         | BOOL      | Drive coast (OFF 2)                                                           |

|                       |      |                                                                                                                                            |
|-----------------------|------|--------------------------------------------------------------------------------------------------------------------------------------------|
|                       |      | – 1: no coasting<br>– 0: coasting. Power output stage is deactivated (OFF2). The drive coasts to a stop.                                   |
| RapidStop             | BOOL | Fast stop (OFF 3)<br>– 1: no fast stop<br>– 0: brake drive to standstill with fast stop and then deactivate the power output stage (OFF3). |
| EnableMovement        | BOOL | Enable rotational speed setpoint value<br>1 = Enable                                                                                       |
| AckError              | BOOL | Reset Active Faults<br>1 = Reset Faults                                                                                                    |
| Velocity              | REAL | Speed Setpoint as per Actual user Unit.                                                                                                    |
| BaseValueVelocity     | REAL | Base Value Velocity. User can find the value from the parameter P1.11280701.0.0                                                            |
| GearRatio_Numerator   | INT  | Determine the numerator of the overall gear ratio(drive side). User can find the value from the parameter P1.1242.0.0                      |
| GearRatio_Denominator | INT  | Determine the numerator of the overall gear ratio(drive side). User can find the value from the parameter P1.1243.0.0                      |

## 8.4 Monitor Area Tag Details

| Output signal     | Type | Meaning                                                                 |
|-------------------|------|-------------------------------------------------------------------------|
| ReadyforOperation | BOOL | Ready to State                                                          |
| AxisEnabled       | BOOL | Pulses are Enabled                                                      |
| AxisError         | BOOL | Fault is Active                                                         |
| Lockout           | BOOL | Drive Inhibited                                                         |
| Error             | BOOL | AOI Fault Active                                                        |
| ActVelocity       | REAL | Actual velocity as per Actual user Unit.                                |
| Status            | INT  | AOI Status Code: For more details refer <a href="#">chapter 8.4.1</a> . |

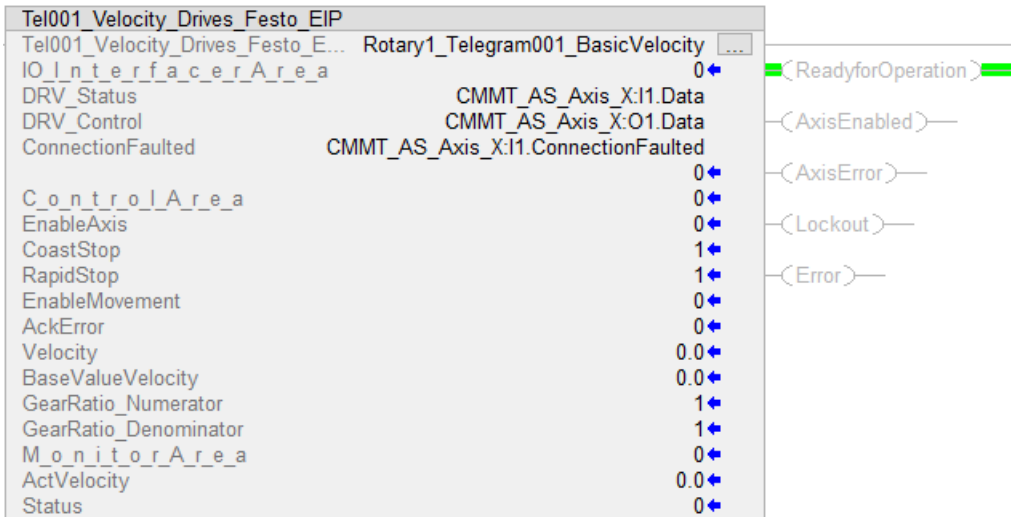
#### 8.4.1 Diagnostic Messages / Function Block Faults

The function block gives multiple types of diagnostic messages. The following chapter will describes how to handle & solve them.

| Status Error Number | Case                          | Remedy                                                                                                                                                                                                                                 |
|---------------------|-------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 16#7002             | No Error                      | --                                                                                                                                                                                                                                     |
| 16#8401             | Alarm message(s) in the drive | Check the error code in ActFaulted tag                                                                                                                                                                                                 |
| 16#8402             | Switching ON Inhibited        | The switch-on lock is active. The drive is in the status S1 Switching On Inhibited.<br>Switching on is only possible with the following command sequence: OFF (OFF1) and no coasting (no OFF2) and no fast stop (no OFF3) and then ON. |
| 16#8600             | Communication Disconnected    | Check the communication connections/ setting between drive and PLC                                                                                                                                                                     |

#### 8.5 Quick Procedure to Run the CMMT Drive in Basic Velocity Control AOI

##### AOI View at First Time Online



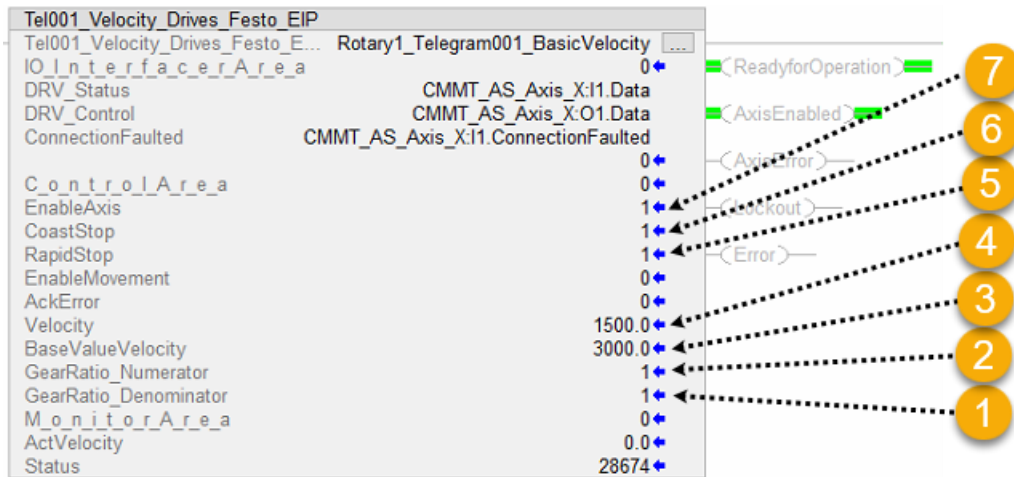
Once the project is ready & communication established with CMMT drive the “Tel001\_Velocity\_Drives\_Festo\_EIP” AOI looks like above figure. The “ReadyforOperation” status tag indicates the AOI ready state.



##### Note

- “ReadyforOperation” status tag indicates drive is in ready state
- Drive will run when “ReadyforOperation” status tags are true.

##### AOI View at Parameters Configuration Completed



1. Enter the value of the GearRatio\_Numerator. User can find the value from the parameter P1.1242.0.0
2. Enter the value of the GearRatio\_Denominator. User can find the value from the parameter P1.1243.0.0



#### Information

- The set velocity is always based on the motor velocity. If the user has a gear mounted, the gear factor is not included in the calculation. Therefore, the gear factor needs to be added in the AOI to achieve the set velocity at the drive side.

3. Enter the value of the Base Value Velocity. User can find the value from the parameter P1.11280701.0.0
4. Enter the velocity value within a range that is twice the range of the Base value velocity.



#### Information

- Users can enter a velocity value that is twice the BaseValueVelocity. If a value higher than twice the BaseValueVelocity is entered, the actual velocity will not exceed twice the BaseValueVelocity.
- User can also vary the setpoint value during drive in run condition.

5. Ensure the “**RapidStop**” is in true condition if “**RapidStop**” is in false state then drive will stop the axis with deceleration time.
6. Ensure the “**CoastStop**” is in true condition if “**CoastStop**” is in false state then immediate stop (Emergency Stop) is active.
7. When “**EnableAxis**” is true then AOI will enable the drive. At this stage “**AxisEnabled**” status tags will become true.

### AOI View After Run Command

| Area              | Tag                              | Value                               |
|-------------------|----------------------------------|-------------------------------------|
| IO_Interface_Area | Tel001_Velocity_Drives_Festo_EIP | Rotary1_Telegram001_BasicVelocity   |
|                   | IO_Interface_Area                | 0                                   |
|                   | DRV_Status                       | CMMT_AS_Axis_X:I1.Data              |
|                   | DRV_Control                      | CMMT_AS_Axis_X:O1.Data              |
| Control_Area      | ConnectionFaulted                | CMMT_AS_Axis_X:I1.ConnectionFaulted |
|                   | Control_Area                     | 0                                   |
|                   | EnableAxis                       | 1                                   |
|                   | CoastStop                        | 1                                   |
| Monitor_Area      | RapidStop                        | 1                                   |
|                   | EnableMovement                   | 1                                   |
|                   | AckError                         | 0                                   |
|                   | Velocity                         | 1500.0                              |
|                   | BaseValueVelocity                | 3000.0                              |
|                   | GearRatio_Numerator              | 1                                   |
|                   | GearRatio_Denominator            | 1                                   |
|                   | Monitor_Area                     | 0                                   |
|                   | ActVelocity                      | 1501.648                            |
|                   | Status                           | 28674                               |

At this stage drive is ready to execute the run command provide by user.

1. Change the “**EnableMovement**” tag from false to true for provide the run command to the drive.



#### Note

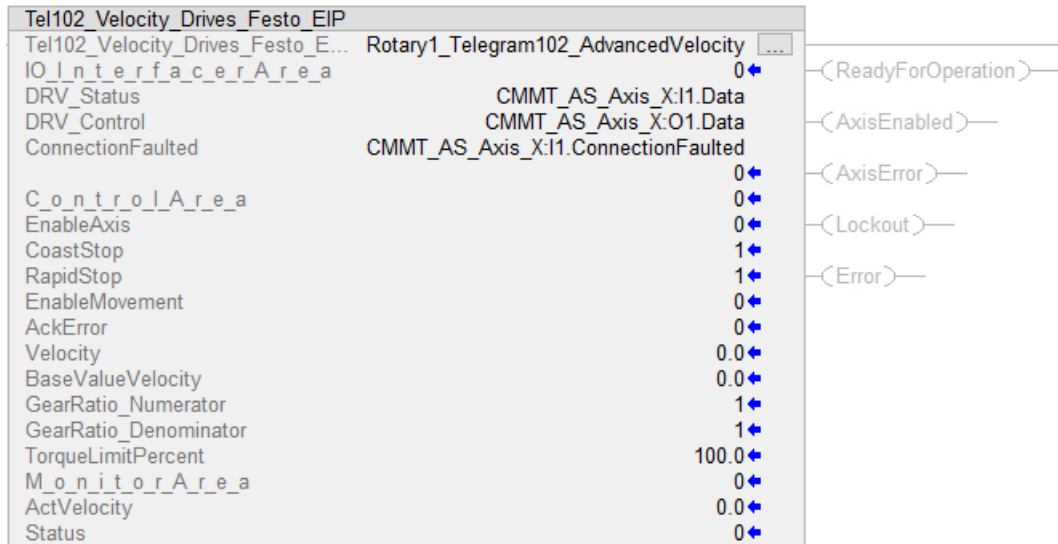
- Motor actual speed will update in the “**ActVelocity**” status tag as highlighted in yellow color.

## 9 “Tel102\_Velocity\_Drives\_Festo\_EIP” AOI / Advance Velocity Control

### 9.1 “Tel102\_Velocity\_Drives\_Festo\_EIP” AOI Introduction

The drive is controlled by a main setpoint (Velocity Setpoint) via Telegram 102, the velocity is controlled completely in the drive.

The “Tel102\_Velocity\_Drives\_Festo\_EIP” AOI is used to control the CMMT drive in velocity mode with specified torque limit. This is the major different between advanced & basic velocity modes.



### 9.2 IO Interface Area Tag Details

| Interface Area Tags | Data Type              | Function Description              |
|---------------------|------------------------|-----------------------------------|
| DRV_Status          | ARRAY OF INT [ 0.. 10] | Ethernet Module Status Variables  |
| DRV_Control         | ARRAY OF INT [ 0.. 6]  | Ethernet Module Control Variables |

### 9.3 Control Area Tag Details

| Control Area Tags | Data Type | Function Description                                                                                                            |
|-------------------|-----------|---------------------------------------------------------------------------------------------------------------------------------|
| ConnectionFaulted | BOOL      | Communication disconnected acknowledgement for AOI                                                                              |
| EnableAxis        | BOOL      | Drive Enable ( ON/OFF1 )<br>1 = Enable Drive<br>0 = Ramp Down & Disable drive                                                   |
| CoastStop         | BOOL      | Drive coast (OFF 2)<br>– 1: no coasting<br>– 0: coasting. Power output stage is deactivated (OFF2). The drive coasts to a stop. |

|                       |      |                                                                                                                                            |
|-----------------------|------|--------------------------------------------------------------------------------------------------------------------------------------------|
| RapidStop             | BOOL | Fast stop (OFF 3)<br>– 1: no fast stop<br>– 0: brake drive to standstill with fast stop and then deactivate the power output stage (OFF3). |
| EnableMovement        | BOOL | Enable rotational speed setpoint value<br>1 = Enable                                                                                       |
| AckError              | BOOL | Reset Active Faults<br>1 = Reset Faults                                                                                                    |
| Velocity              | REAL | Speed Setpoint as per Actual user Unit.                                                                                                    |
| BaseValueVelocity     | REAL | Base Value Velocity. User can find the value from the parameter P1.11280701.0.0                                                            |
| GearRatio_Numerator   | INT  | Determine the numerator of the overall gear ratio(drive side). User can find the value from the parameter P1.1242.0.0                      |
| GearRatio_Denominator | INT  | Determine the denominator of the overall gear ratio(drive side). User can find the value from the parameter P1.1243.0.0                    |
| TorqueLimitPercent    | REAL | Torque Limit Setpoint in Percentage<br>0 = No Torque Limit, 100 = Full Torque Limit                                                        |

## 9.4 Monitor Area Tag Details

| Monitor Area Tags | Data Type | Function Description                                                    |
|-------------------|-----------|-------------------------------------------------------------------------|
| ReadyforOperation | BOOL      | Ready to Start                                                          |
| AxisEnabled       | BOOL      | Pulses are Enabled                                                      |
| AxisError         | BOOL      | Fault is Active                                                         |
| Error             | BOOL      | AOI Fault Active                                                        |
| ActVelocity       | REAL      | Actual velocity as per Actual user Unit.                                |
| Status            | INT       | AOI Status Code: For more details refer <a href="#">chapter 9.4.1</a> . |

### 9.4.1 Diagnostic Messages / Function Block Faults

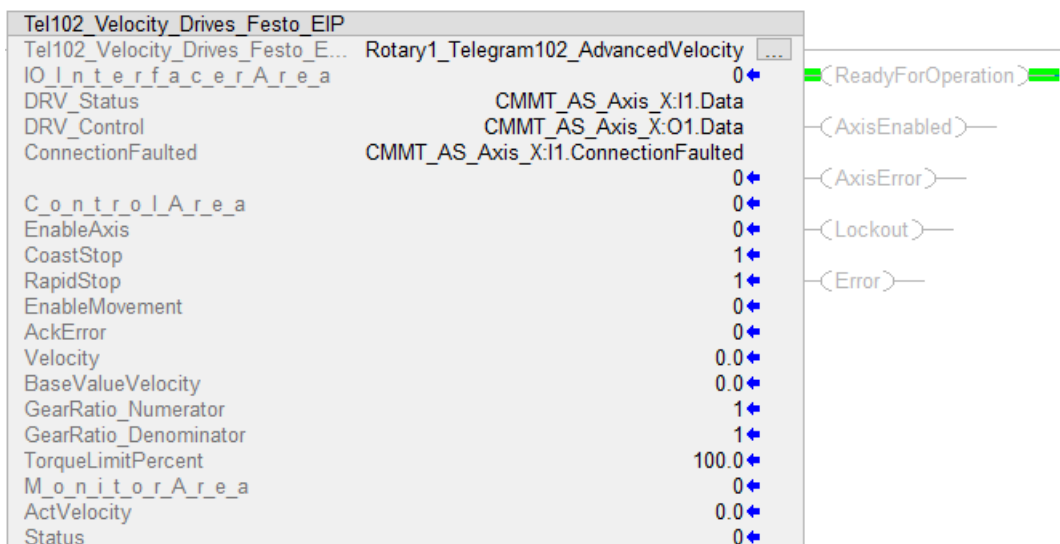
The function block gives multiple types of diagnostic messages. The following chapter will describes how to handle & solve them.

| Status Error Number | Case                                                        | Remedy                                                                            |
|---------------------|-------------------------------------------------------------|-----------------------------------------------------------------------------------|
| 16#7002             | No Error                                                    | --                                                                                |
| 16#8203             | Incorrect parameterization of the TorqueLimitPercent inputs | Check the setting of TorqueLimitPercent inputs value within the Range (0% - 100%) |
| 16#8401             | Alarm message(s) in the drive                               | Check the error code in ActFaulted tag                                            |

| Status Error Number | Case                       | Remedy                                                                                                                                                                                                                                     |
|---------------------|----------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 16#8402             | Switching ON Inhibited     | The switch-on lock is active. The drive is in the status S1 Switching On Inhibited.<br><br>Switching on is only possible with the following command sequence: OFF (OFF1) and no coasting (no OFF2) and no fast stop (no OFF3) and then ON. |
| 16#8600             | Communication Disconnected | Check the communication connections/ setting between drive and PLC                                                                                                                                                                         |

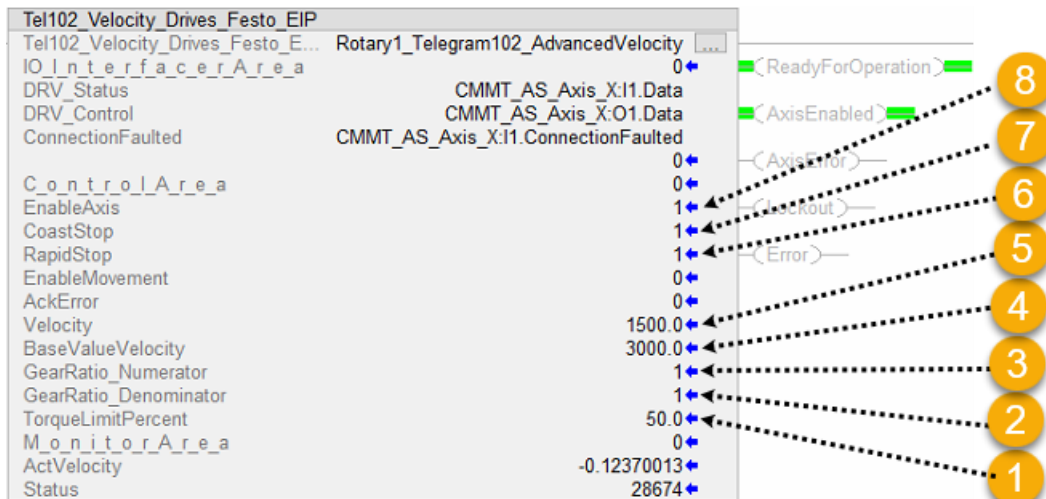
## 9.5 Quick Procedure to Run the CMMT Drive in Advance Velocity Control AOI

### AOI View at First Time Online



Once the project is ready & communication established with CMMT drive the “Tel102\_Velocity\_Drives\_Festo\_EIP” AOI looks like above figure. The “ReadyforOperation” status tag indicates the AOI ready state.

### AOI View at Parameters Configuration Completed



1. Enter the value of torque limit. User can also torque limit during drive in running condition.
2. Enter the value of the GearRatio\_Numerator. User can find the value from the parameter P1.1242.0.0
3. Enter the value of the GearRatio\_Denominator. User can find the value from the parameter P1.1243.0.0



#### Information

- The set velocity is always based on the motor velocity. If the user has a gear mounted, the gear factor is not included in the calculation. Therefore, the gear factor needs to be added in the AOI to achieve the set velocity at the drive side.

4. Enter the value of the Base Value Velocity. User can find the value from the parameter P1.11280701.0.0
5. Enter the velocity value within a range that is twice the range of the Base value velocity.

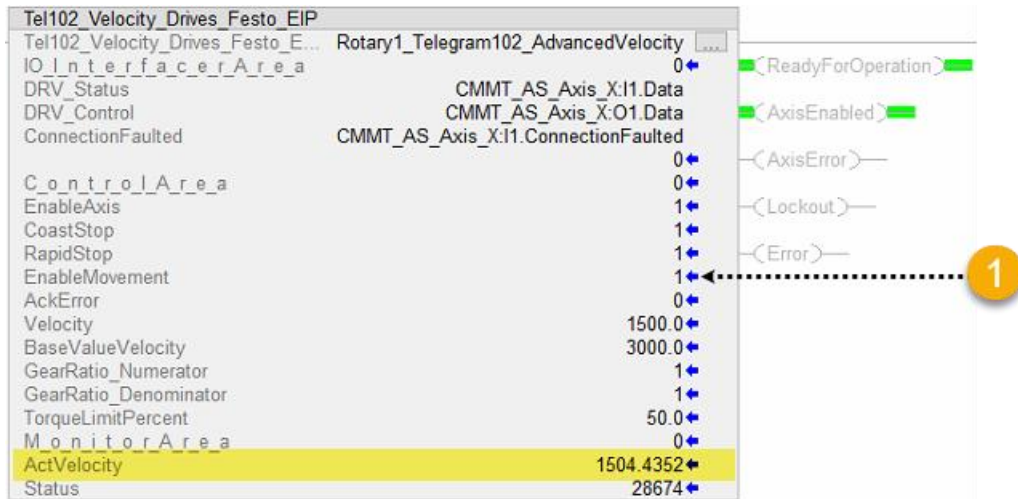


#### Information

- Users can enter a velocity value that is twice the BaseValueVelocity. If a value higher than twice the BaseValueVelocity is entered, the actual velocity will not exceed twice the BaseValueVelocity.
- User can also vary the setpoint value during drive in run condition.

6. Ensure the “**RapidStop**” is in true condition if “**RapidStop**” is in false state then drive will stop the axis with deceleration time.
7. Ensure the “**CoastStop**” is in true condition if “**CoastStop**” is in false state then immediate stop (Emergency Stop) is active.
8. When “**EnableAxis**” is true then AOI will enable the drive. At this stage “**AxisEnabled**” status tags will become true.

### AOI View After Run Command



At this stage drive is ready to execute the run command provide by user.

1. Change the “**EnableMovement**” tag from false to true for provide the run command to the drive.

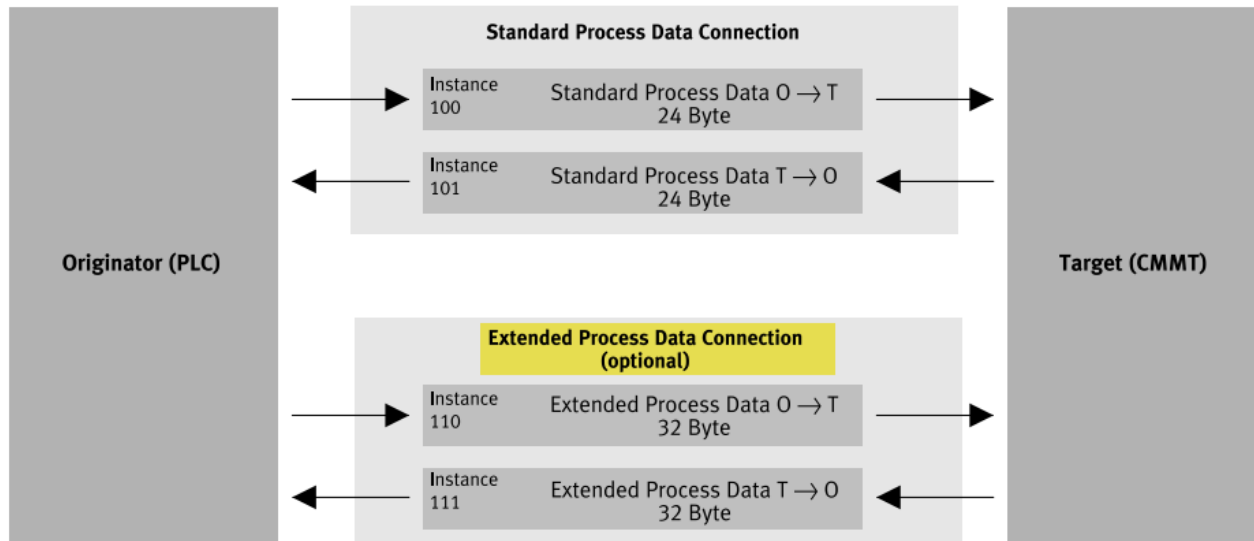


**Note**

- Motor actual speed will update in the “**ActVelocity**” status tags as highlighted in yellow color.

## 10 “ExtendedProcessData\_Festo\_EIP” AOI

### 10.1 Extended Process Data (EPD) Introduction



The Extended Process Data (EPD) is an optional telegram to exchange additional parameters from CMMT drive. The EPD is capable to read & write up to 8 parameters or 32 Bytes of IO's from the CMMT drive. The EPD telegram is named as telegram 910 & IO values can access through the instances 110 & 111.



#### Note

- RSLogix – 5000 V20.0 below software does not allow the 3<sup>rd</sup> party EDS files, so here the examples are given for EPD configuration with ETHERNET GENERIC MODULE in [Chapter -10.3](#).
- User can find the sample code for Extended Process Data in this library directory  
“CMMT\_EIP\_MotionLib Vx.x > 50\_Examples >  
**CMMT\_Tel910\_EPD\_v1\_3\_GEN\_Class3\_MSG\_L32Ev16\_05.ACD**” &  
**“CMMT\_Tel910\_EPD\_v1\_3\_EDS\_L30ERMV20\_04”**

## 10.2 Configure Extended Process Data in Festo Automation Suite

### 10.2.1 Read Extended Data Configuration in Festo Automation Suite

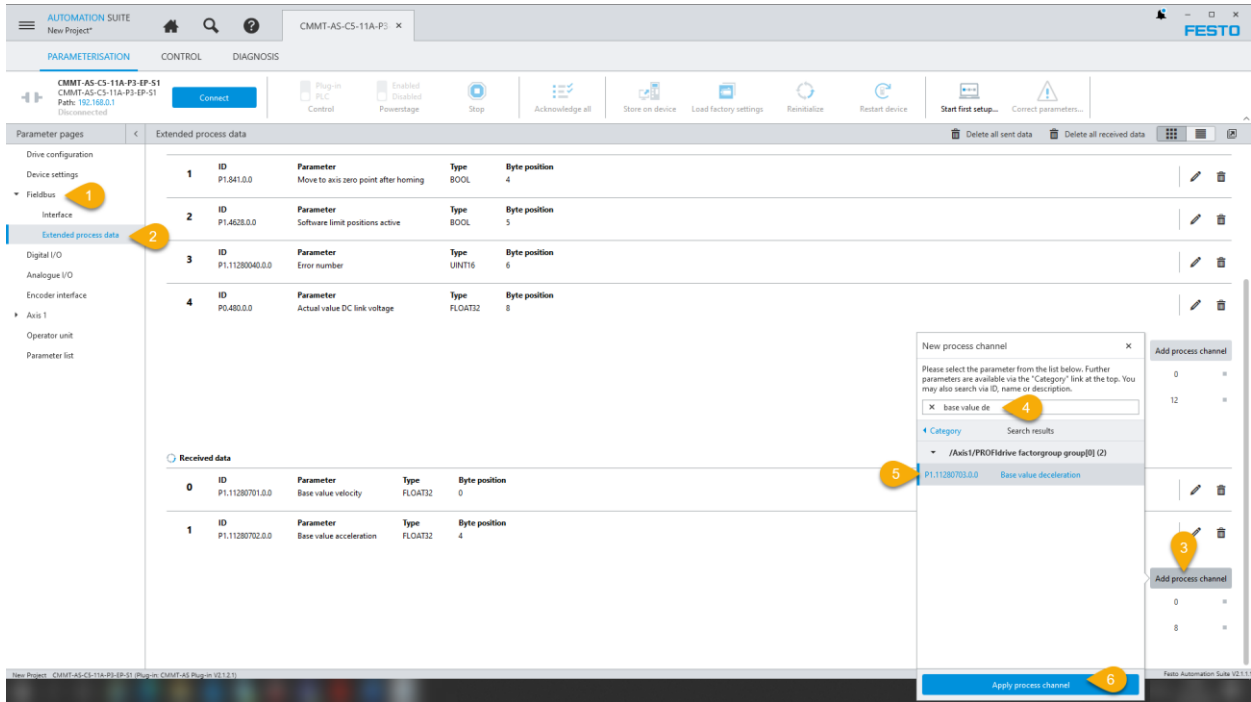
The screenshot shows the Festo Automation Suite V2.1.1 interface. The top bar indicates the current project is 'CMMT-AS-CS-11A-P3'. The left sidebar has a 'Fieldbus' tab selected, and within it, the 'Extended process data' sub-tab is active. The main workspace shows a table of 'Sent data' parameters:

| ID | Parameter                                          | Type    | Byte position |
|----|----------------------------------------------------|---------|---------------|
| 0  | P1.7118.0.0<br>Current nominal current             | FLOAT32 | 0             |
| 1  | P1.841.0.0<br>Move to axis zero point after homing | BOOL    | 4             |
| 2  | P1.4628.0.0<br>Software limit positions active     | BOOL    | 5             |
| 3  | P1.11280040.0.0<br>Error number                    | UINT16  | 6             |

A 'New process channel' dialog box is open, showing a search for 'actual value DC'. The search results list includes '/System/DC link manager group[0] (23)' and 'P0.480.0.0 Actual value DC link voltage'. The 'Apply process channel' button is at the bottom right.

1. Open “**Fieldbus**” tab from parameterization menu.
2. Select “**Extended process data**” from Fieldbus list.
3. Select “**Add Process Channel**” from “**Send data**” type.
4. Search the parameter to add by its name or parameter number.
5. Select parameter from the list.
6. Click “**Apply process channel**” button to add the selected parameter.

## 10.2.2 Write Extended Data Configuration in Festo Automation Suite



1. Open “**Fieldbus**” tab from parameterization menu.
2. Select “**Extended process data**” from Fieldbus list.
3. Select “**Add Process Channel**” from “**Received data**” type.
4. Search the parameter to add by its name or parameter number.
5. Select parameter from the list.
6. Click “**Apply process channel**” button to add the selected parameter.



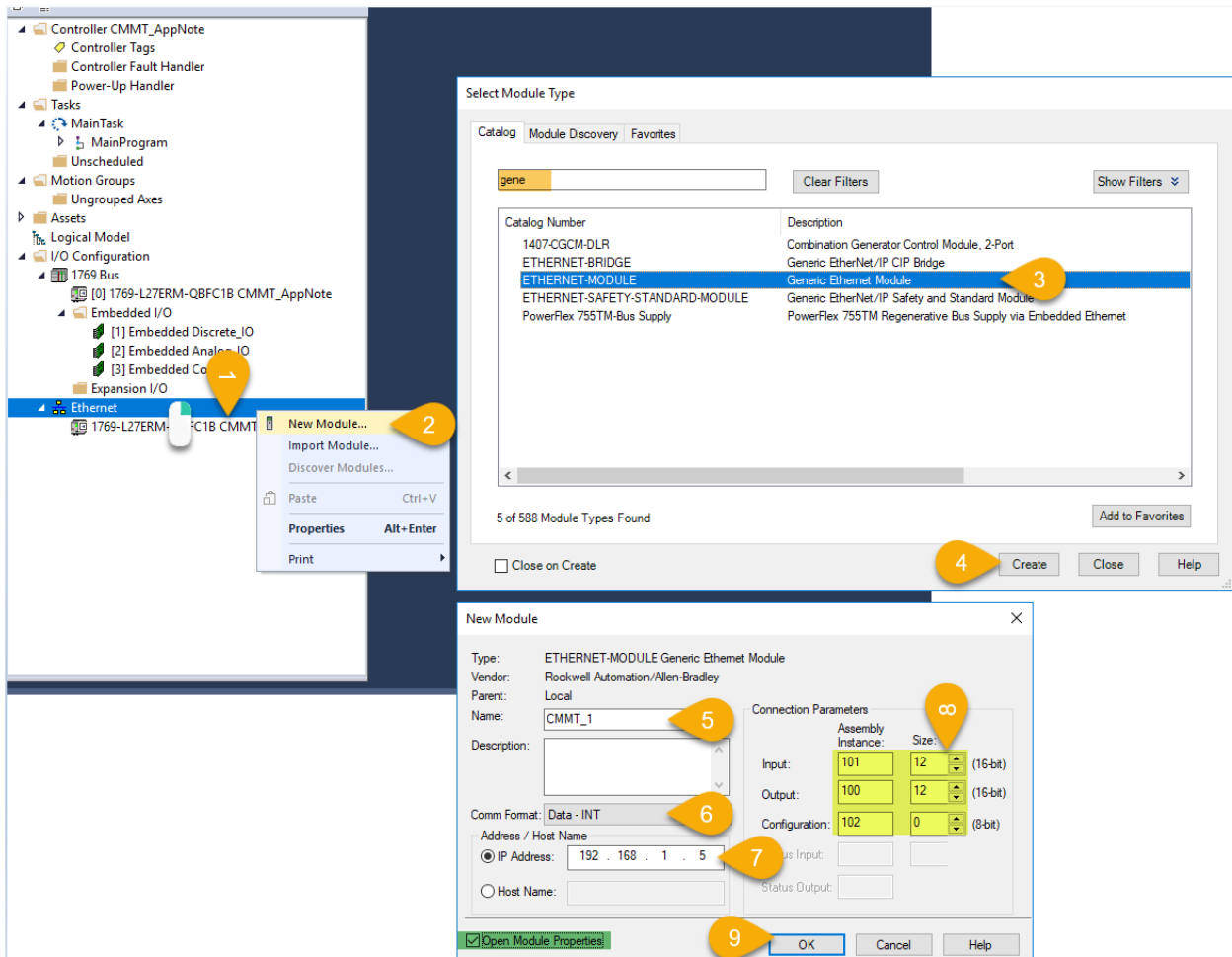
### Note

- Once all the parameters configured successfully download the data to the servo controller.

## 10.3 Extended Process Data Read/Write using a Generic Module

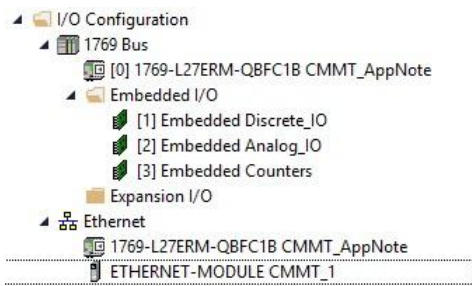
### 10.3.1 Create MSG Instruction for EPD Read Parameters

#### Step – 1

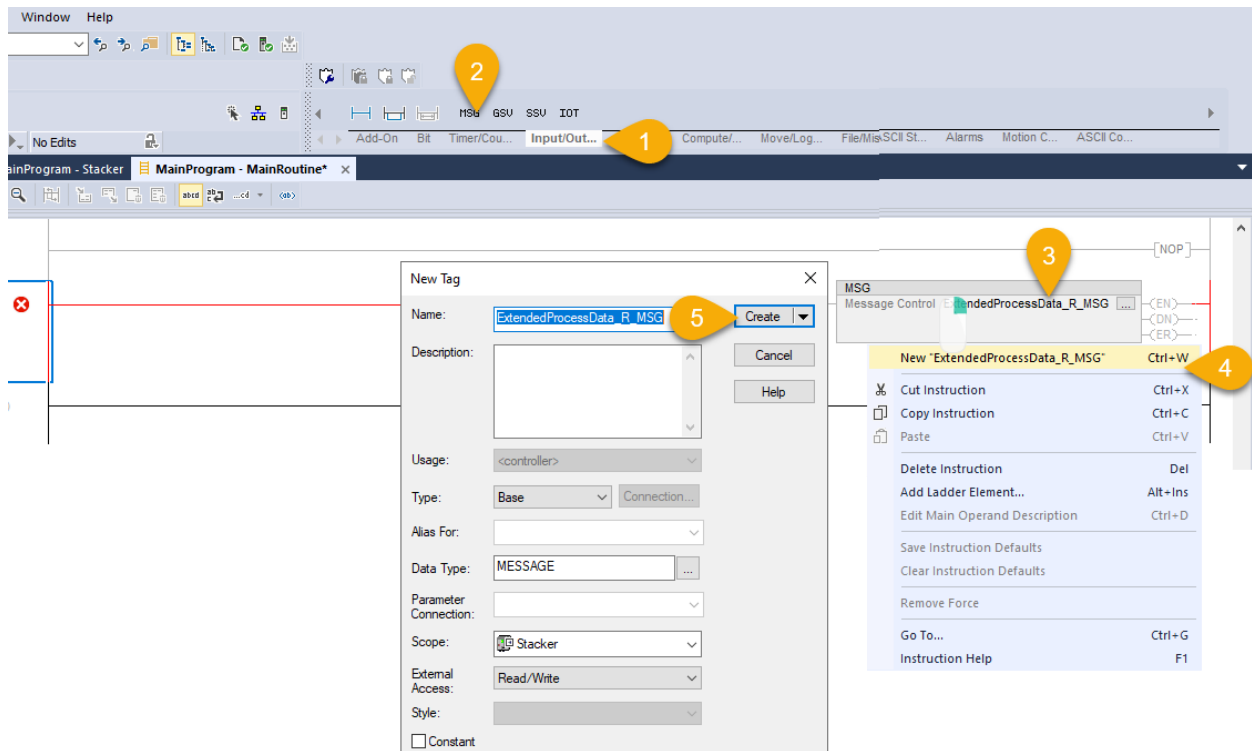


1. Select the **“Ethernet”** tree from controller organizer window
2. Select **“New Module”** option from right click menu.
3. Type the module name in search box as highlighted in orange color & select the **“ETHERNET-MODULE”**.
4. Click **“Create”** button from select module type window.
5. Enter the name of the module in **“Name”** filed in New Module window.
6. Select the data format as **“Data – INT”**.
7. Enter the **“IP address”** of CMMT drive ( **Address of Interface Port – XF1** ).
8. Enter the **Assembly Instance & Size values** as highlighted in yellow color.
9. Click **“OK”** to confirm the configuration. Once the OK button is click, Module property window will open automatically because **“Open Module Properties”** is selected default as highlighted in green color.

After completion of Generic ethernet module configuration the new module will be available in ethernet tree as shown below picture.

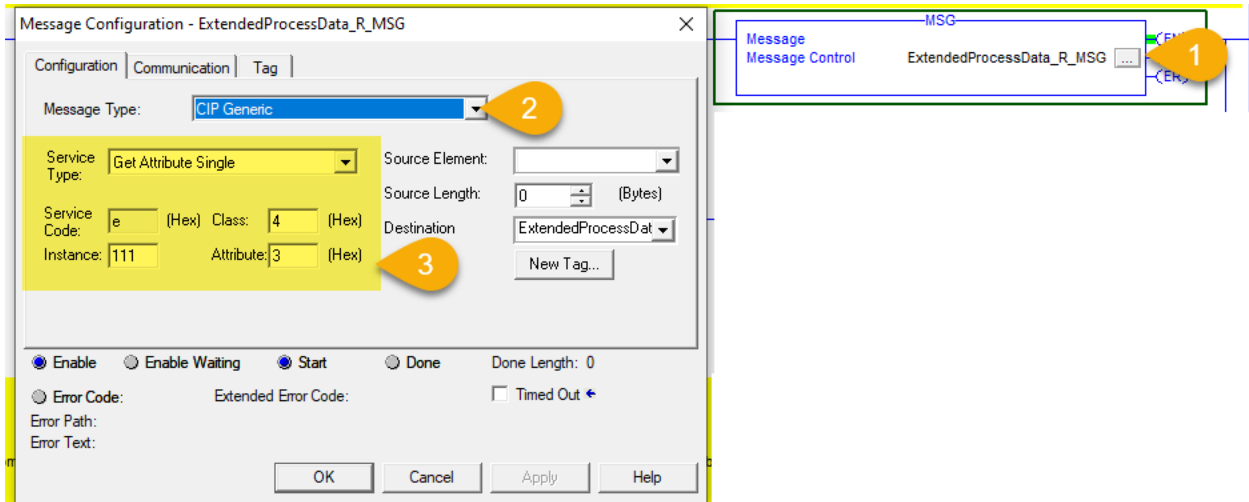


## Step – 2



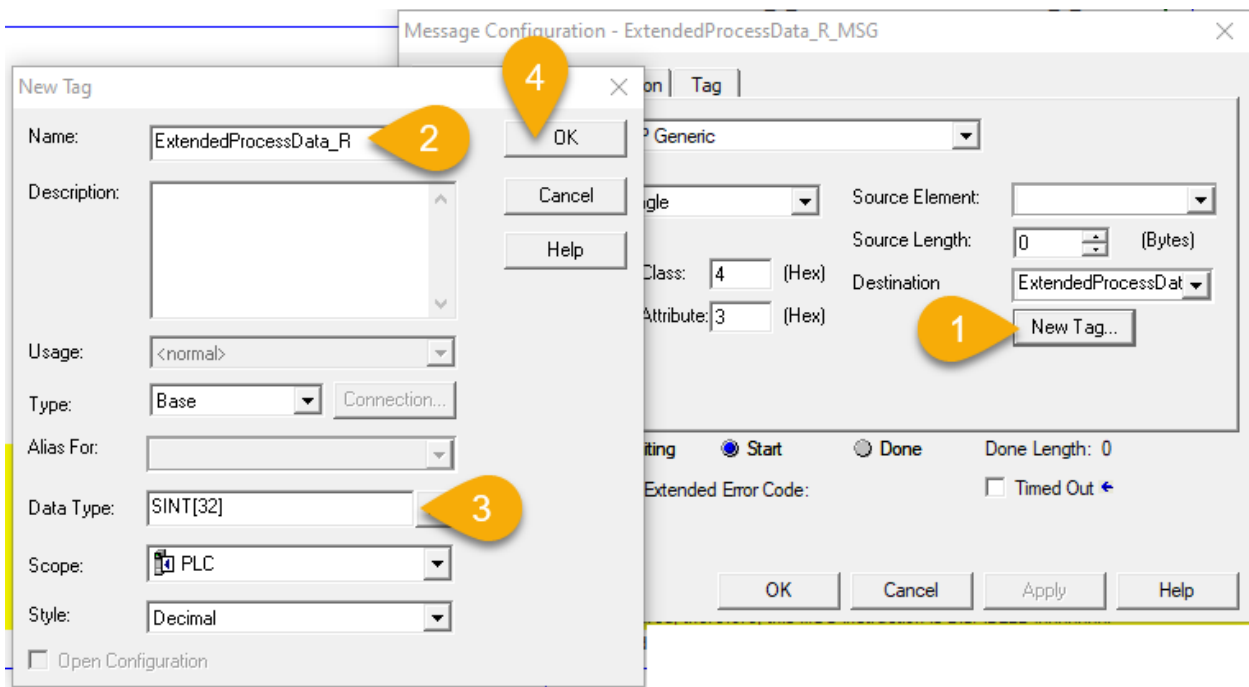
1. Select “**Input/Output**” instruction set from instruction pallet.
2. Select “**MSG**” instruction.
3. Enter the tag name for the MSG instruction & right click.
4. Select “**New**” option with tag name in right click menu.
5. Click “**Create**” option in new tag window.

### Step – 3



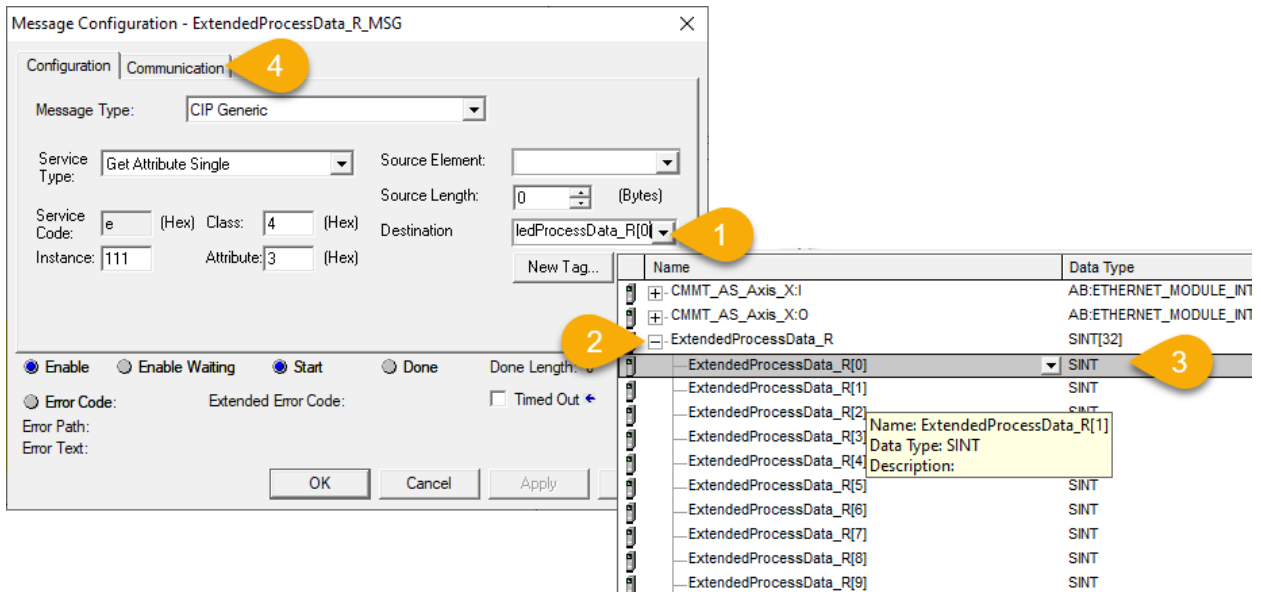
1. Click the “**View Configuration Dialog**” button of newly created MSG instruction.
2. Select “**CIP Generic**” from Message Type drop-down menu.
3. Enter the values for **Service Type, Service Code, Class, Instance & Attribute** as highlighted in yellow color.

### Step – 4



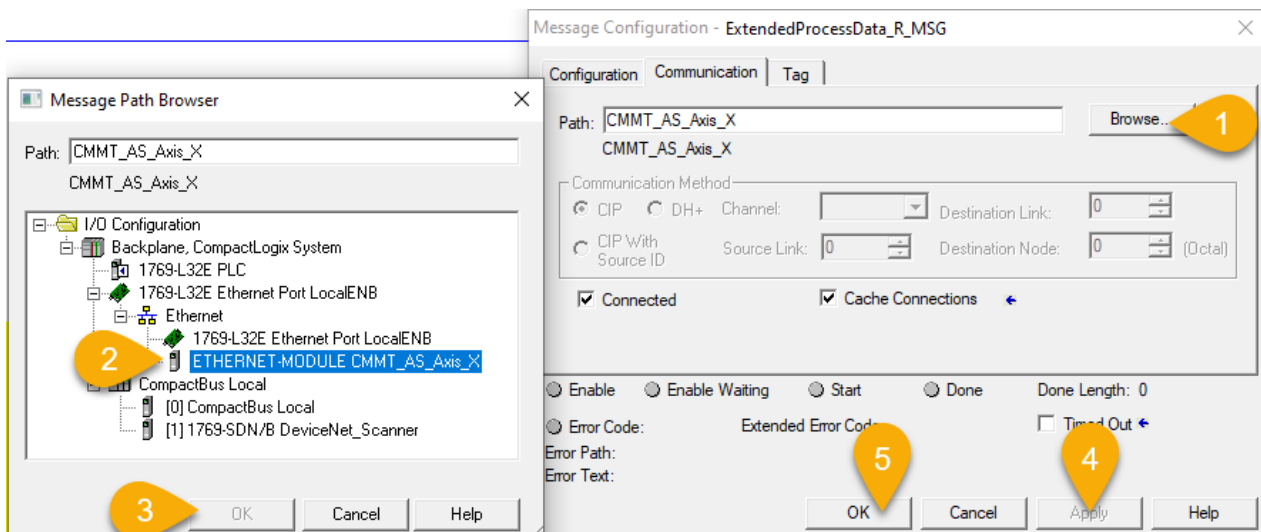
1. Select “**New Tag**” option under the destination element field.
2. Enter the name of the tag in “**Name**” field from New Tag window.
3. Assign the SINT[32] for tag “**Data Type**”.
4. Click “**OK**” button to confirm the configuration.

### Step – 5



1. Select drop-down menu of destination tag element.
2. Browse new tag created in previous step & explore the array by clicking plus button.
3. Select first element of the array.
4. Click **“Communication”** tab from Message configuration window.

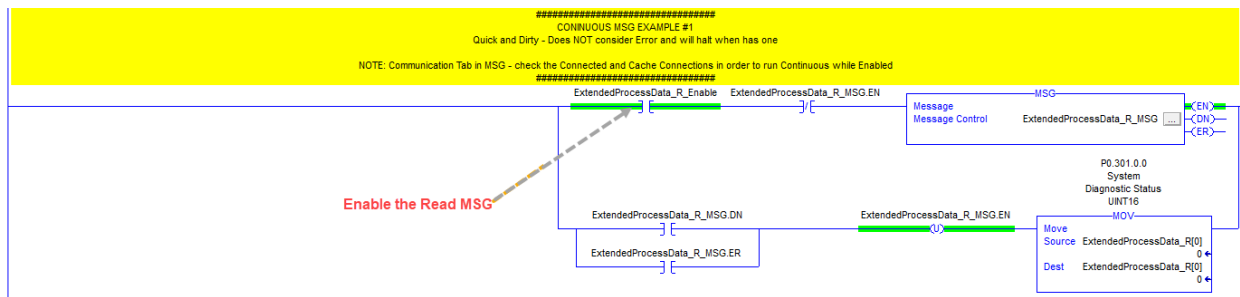
### Step -6



1. Click **“Browse”** button of communication path.
2. Select the Ethernet module created in step – 1.
3. Click **“OK”** button to confirm the path router selection.
4. Click **“Apply”** to confirm the MSG instruction configuration.
5. Click **“OK”** confirm & close the MSG instruction configuration window.

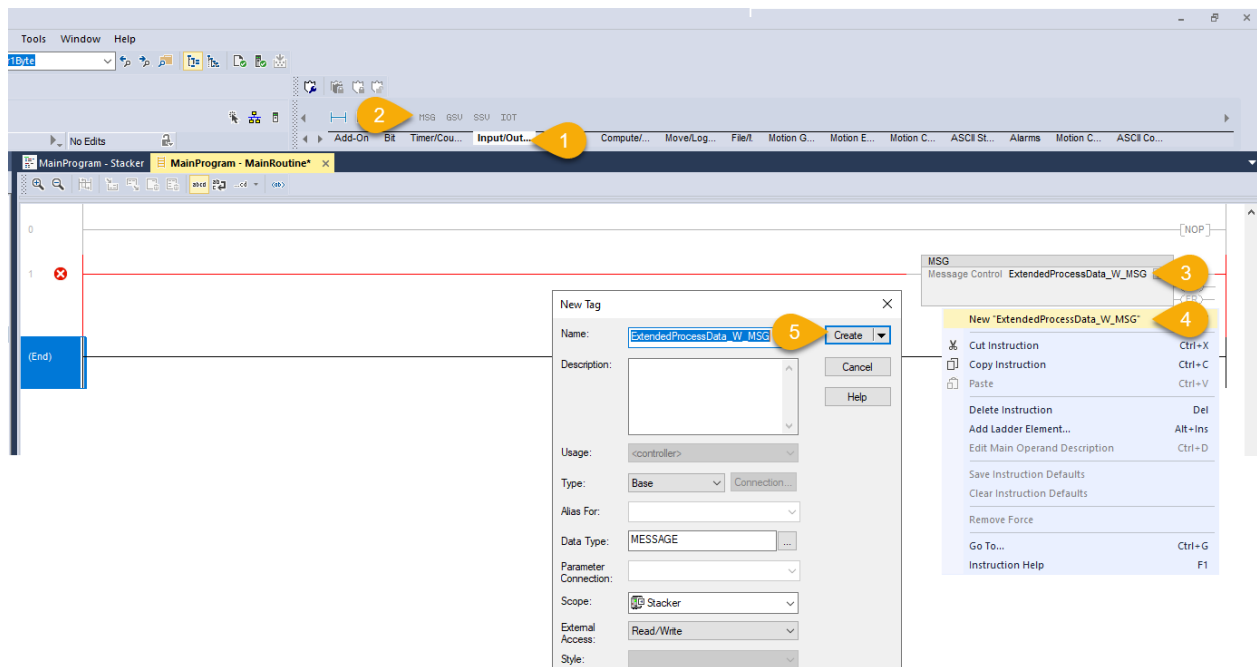
## Step – 7

Sample code for continually execute Read MSG instruction.



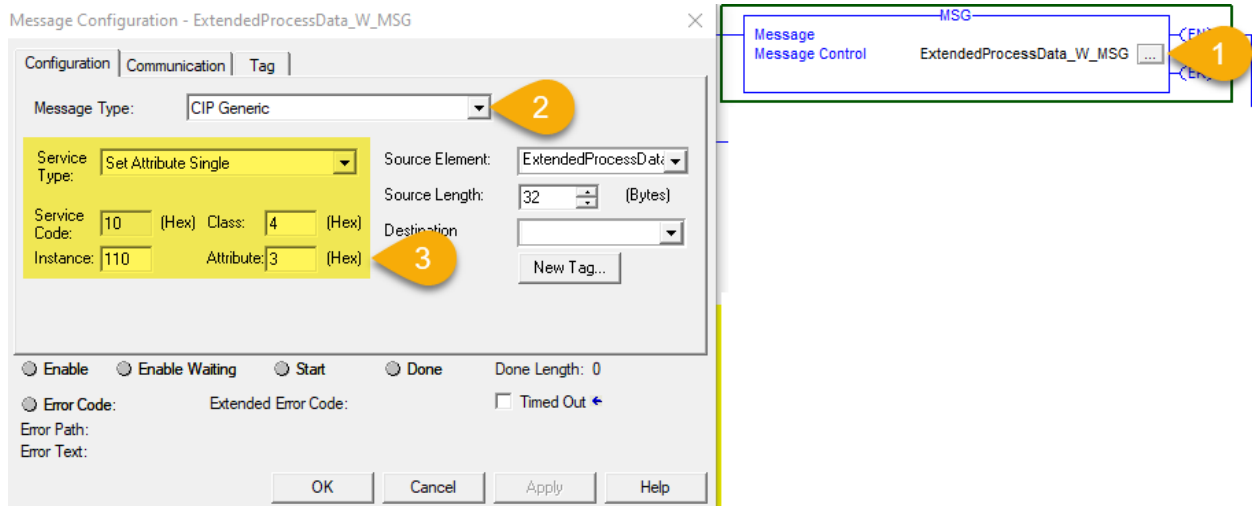
## 10.3.2 Create MSG Instruction for EDP Write Parameters

### Step – 1



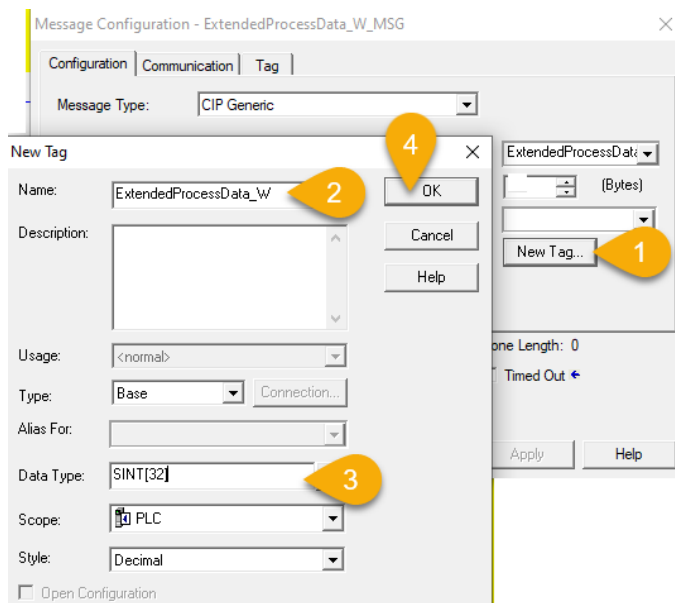
1. Select “**Input/Output**” instruction set from instruction pallet.
2. Select “**MSG**” instruction.
3. Enter the tag name for the MSG instruction & right click.
4. Select “**New**” option with tag name in right click menu.
5. Click “**Create**” option in new tag window.

## Step – 2



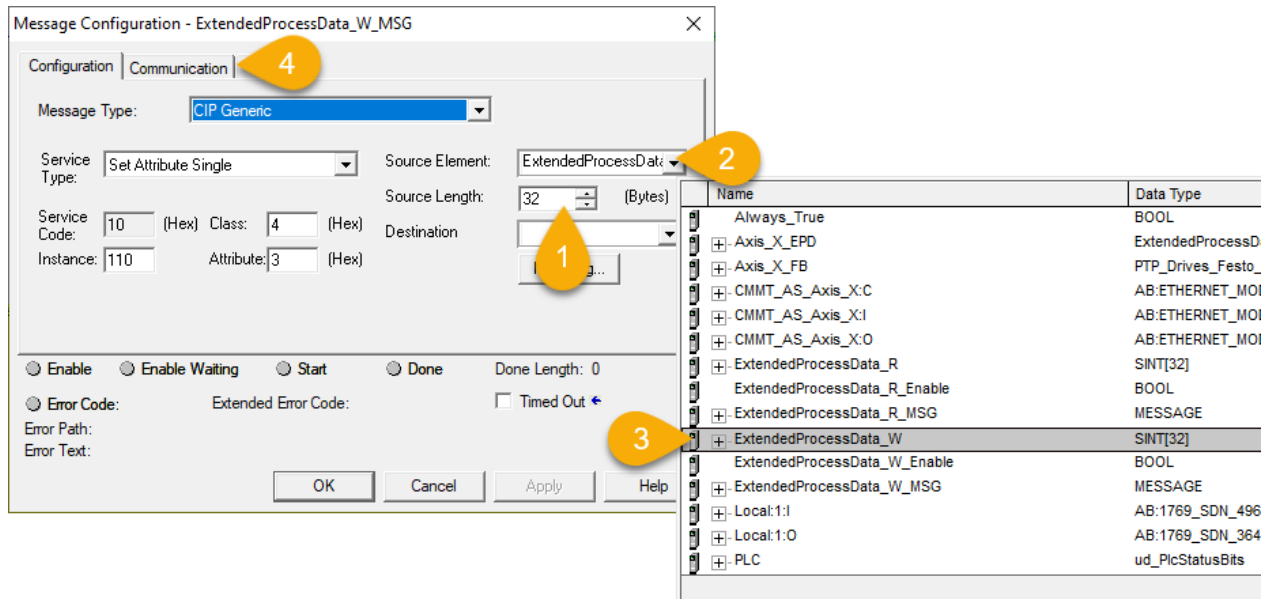
1. Click the “**View Configuration Dialog**” button of newly created MSG instruction.
2. Select “**CIP Generic**” from Message Type drop-down menu.
3. Enter the values for **Service Type, Service Code, Class, Instance & Attribute** as highlighted in yellow color.

## Step – 3



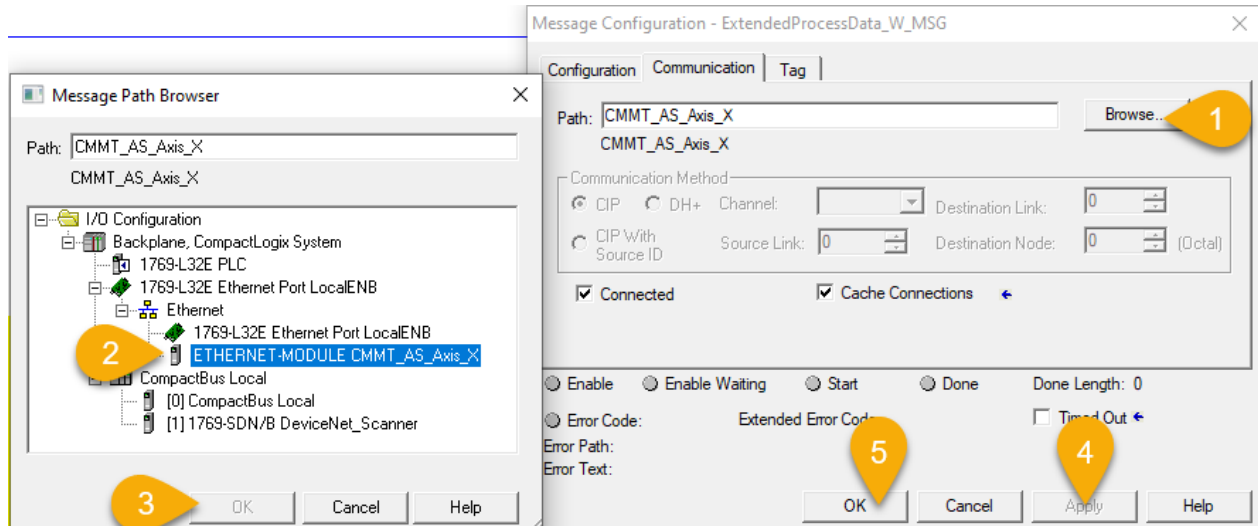
1. Select “**New Tag**” option under the destination element field.
2. Enter the name of the tag in “**Name**” field from New Tag window.
3. Assign the SINT[32] for tag “**Data Type**”.
4. Click “**OK**” button to confirm the configuration.

#### Step – 4



1. Enter the Source Length value as “32” Bytes.
2. Select drop-down menu of source tag element.
3. Browse new tag created in previous step & Select the tag.
4. Click “**Communication**” tab from Message configuration window.

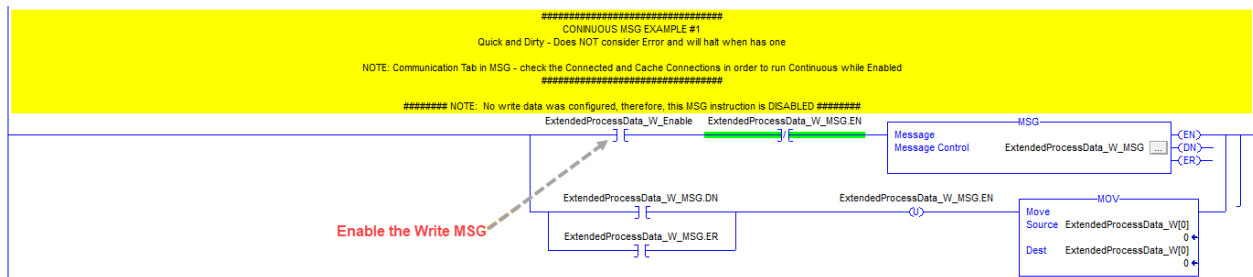
#### Step – 5



1. Click “**Browse**” button of communication path.
2. Select the Ethernet module created in step – 1.
3. Click “**OK**” button to confirm the path router selection.
4. Click “**Apply**” to confirm the MSG instruction configuration.
5. Click “**OK**” confirm & close the MSG instruction configuration window.

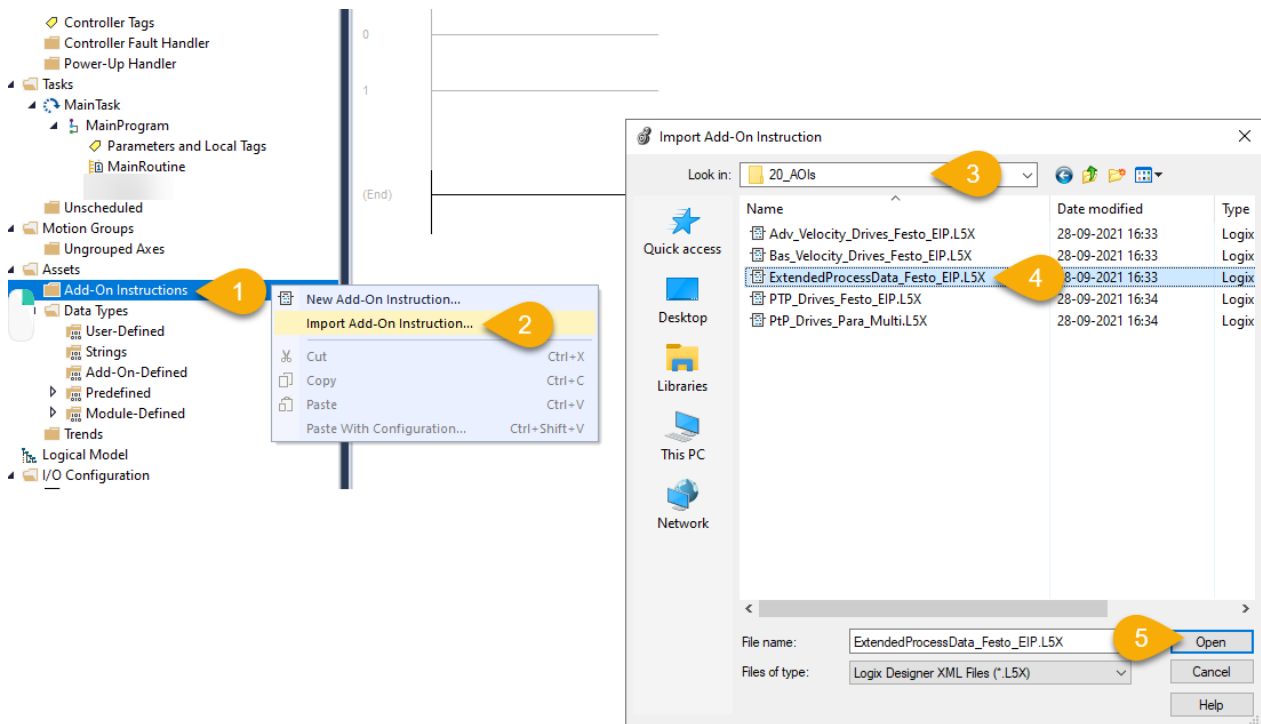
## Step – 6

Sample code for continually execute Write MSG instruction.



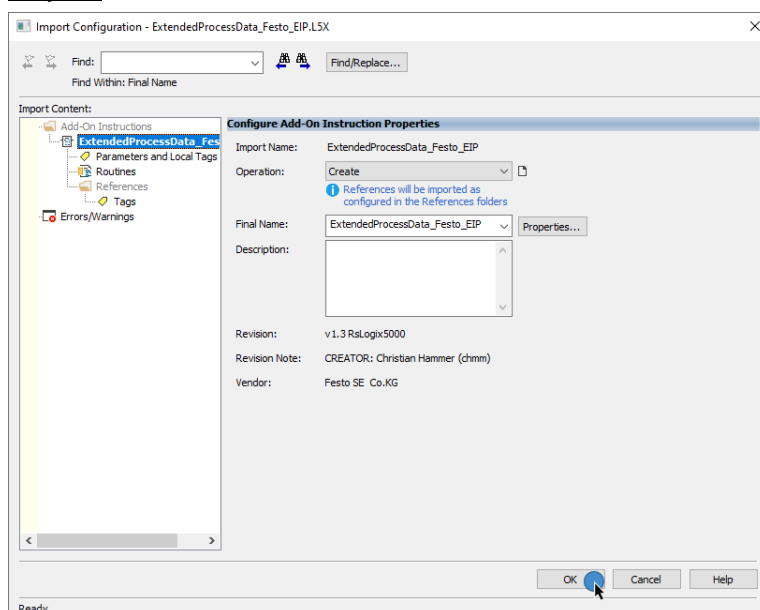
### 10.3.3 Import EPD AOI & Configure the IO Tags

#### Step – 1



1. Select “Add-On Instructions” from controller organizer menu & Right click.
2. Select “Import Add-On Instruction” from right click menu
3. Browse the “ExtendedProcessData\_Festo\_EIP.L5X” file folder location. All AOI files will be in “20\_AOI’s” folder in Library directory.
4. Select “ExtendedProcessData\_Festo\_EIP.L5X” file.
5. Click “Open” button in Import Add-On Instruction window.

#### Step – 2



Click “OK” button.

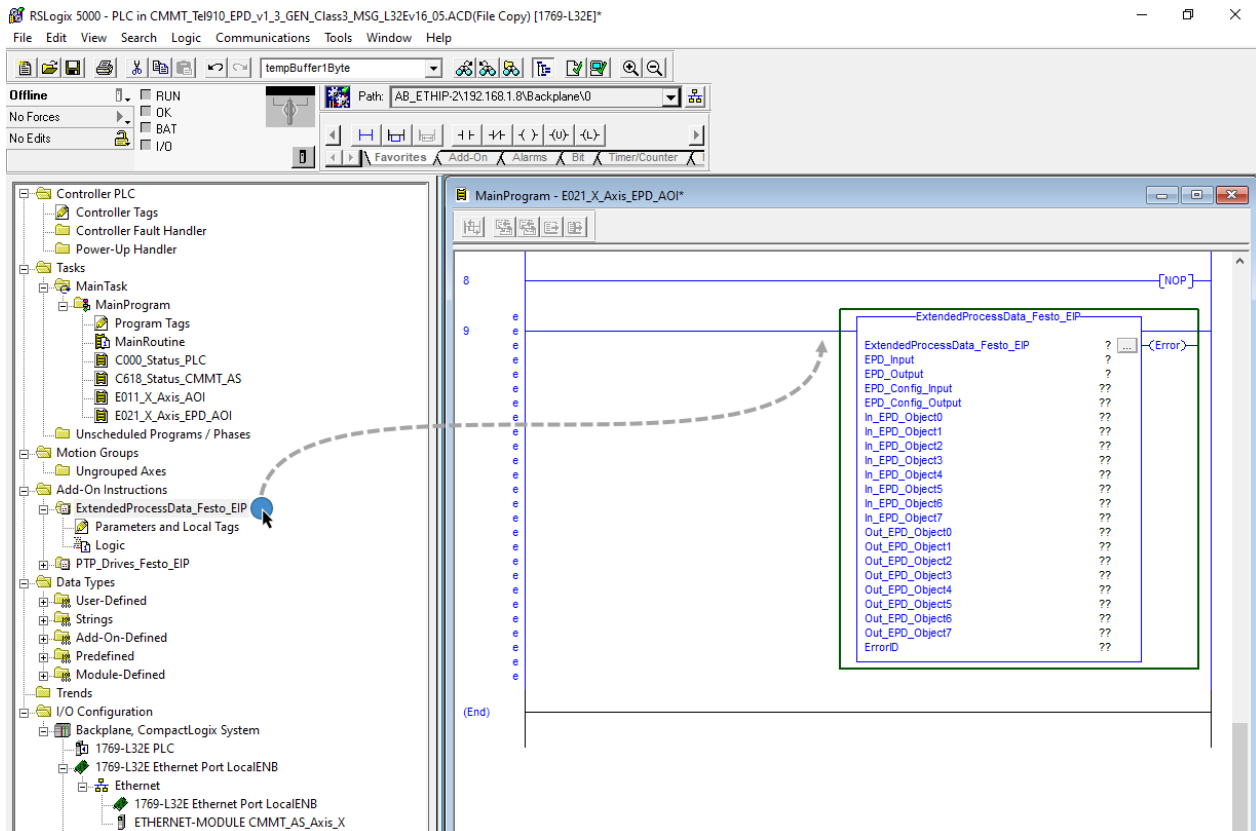
### 10.3.4 Add “ExtendedProcessData\_Festo\_EIP” AOI in PLC Program



#### Note

- The Add-On Instruction must be called cyclically with a separate instance for every servo drive integrated in the user program (every axis). Simultaneous use of multiple instances of the same device is not permitted.
- Any AOI's need to be in a continuous task called cyclically. ([Refer Chapter 3.5 Step – 1](#))

#### Step – 1



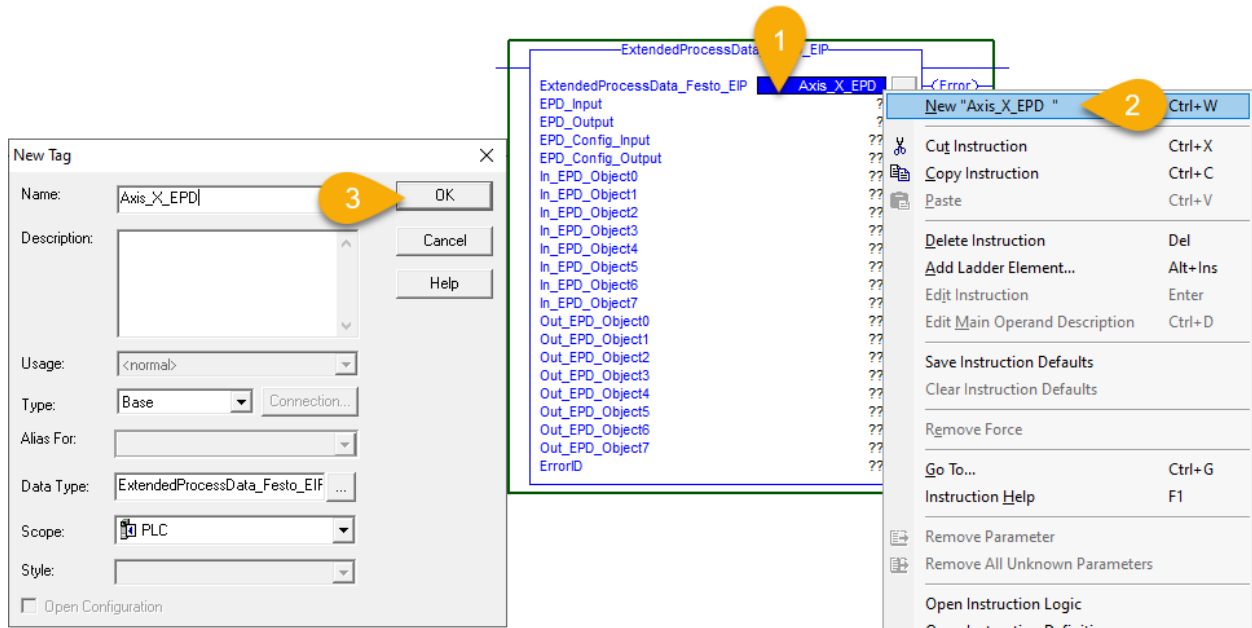
Drag and drop the “**ExtendedProcessData\_Festo\_EIP**” AOI file from Add-On Instructions folder as shown above.



#### Note

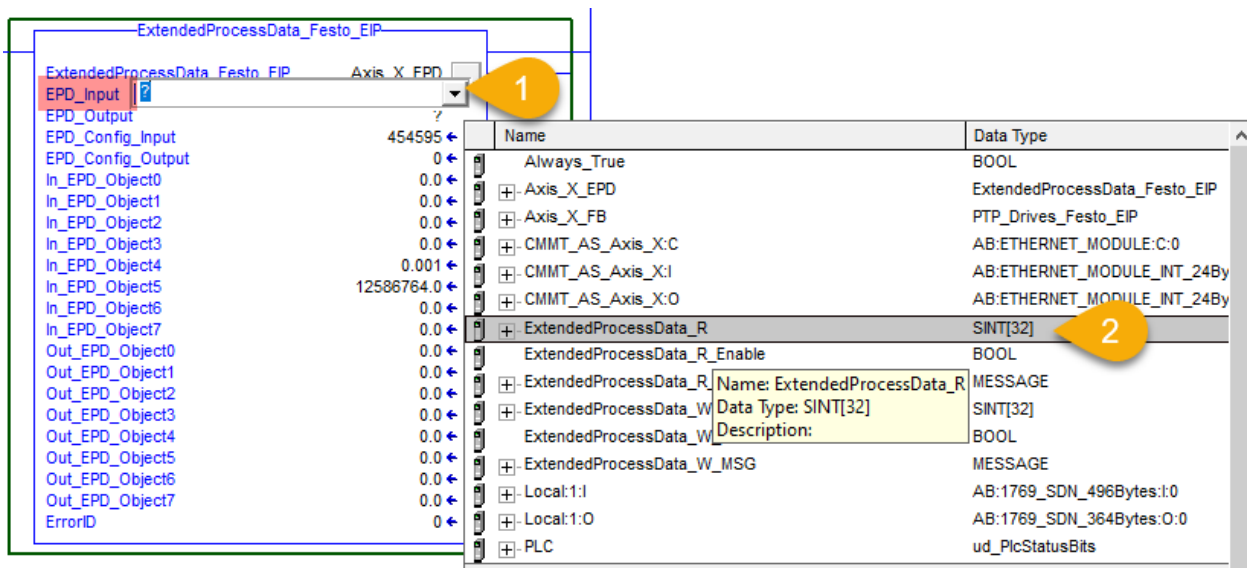
- This Add-On Instruction (AOI) has been designed to support as many Rockwell controllers as possible. Therefore, 64-bit data types have not been used. This Add-On Instruction only supports parameters with a maximum length of 32 bits (4 bytes).

## Step – 2



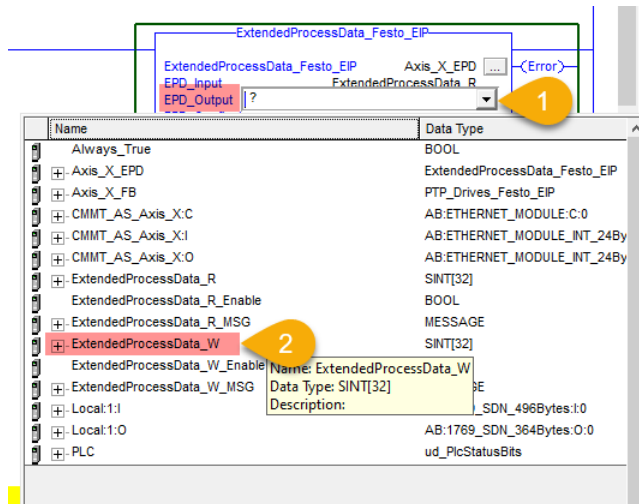
1. Enter the Tag name for the “**ExtendedProcessData\_Festo\_EIP**” AOI & right click.
2. Select “**New**” option with tag name from right click menu.
3. Click “**OK**” button from New Tag window.

## Step – 3



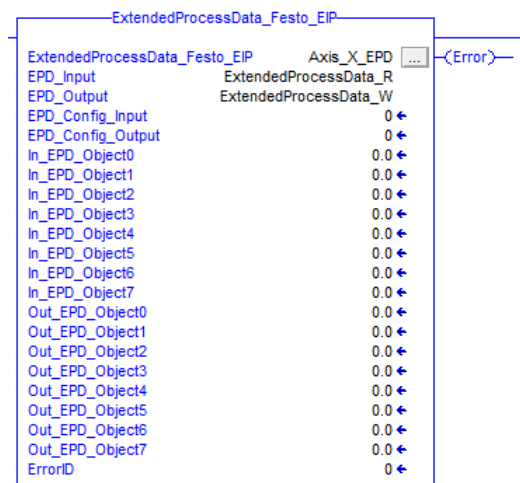
1. Select the drop-down menu of “**EPD\_Input**”.
2. Select the tag created in [Chapter – 10.3.1 Step – 4](#) It will be SINT[32].

#### Step – 4



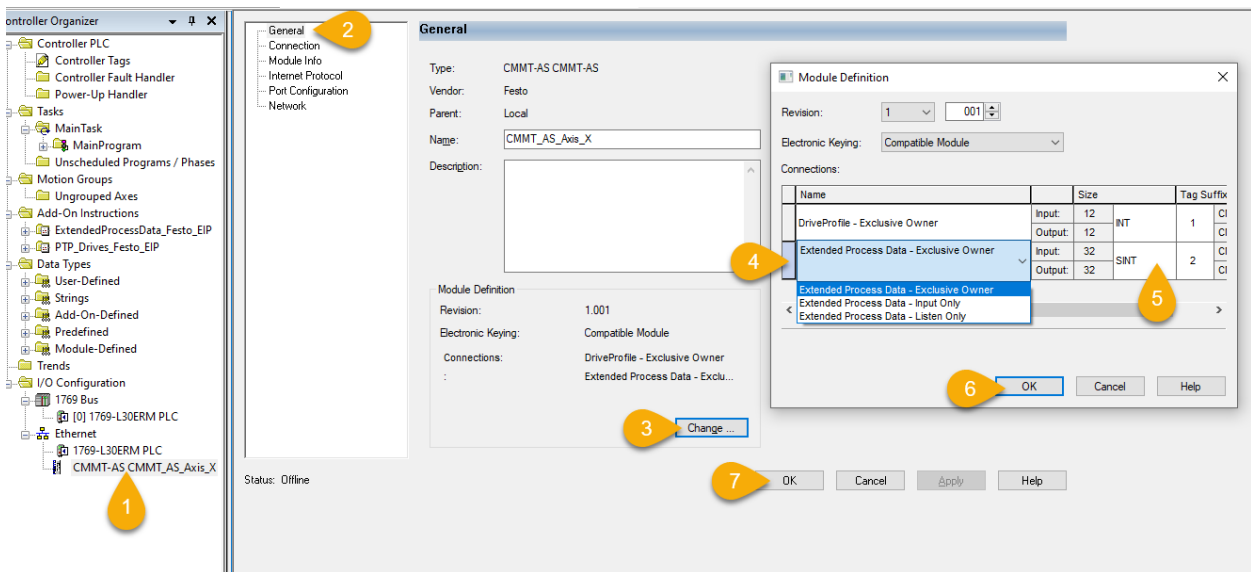
1. Select the drop-down menu of “EPD\_Output”.
2. Select the tag created in [Chapter – 10.3.2 step – 4](#). It will be SINT[32].

Once the above steps are completed successfully the AOI looks like below.



## 10.4 Extended Process Data Read/Write using an EDS File

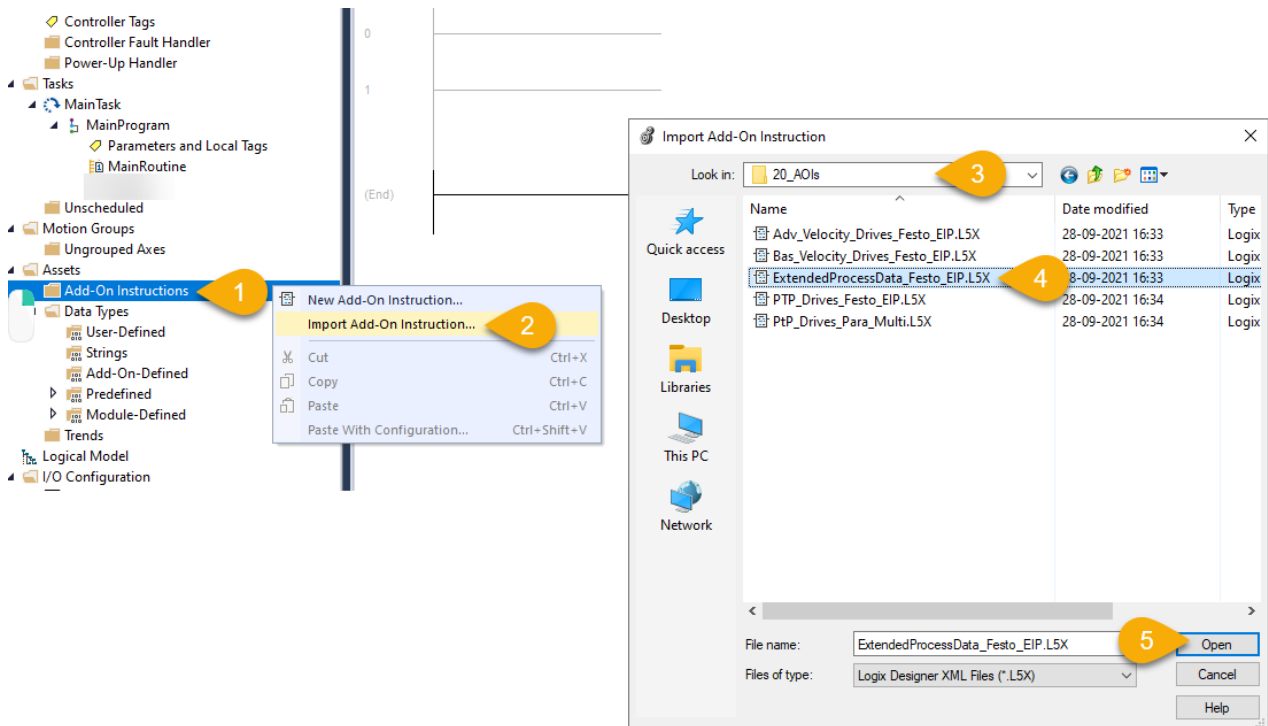
### 10.4.1 Configure EPD in CMMT Drive Ethernet/IP Module



1. Double click the CMMT interface module.
2. Select “**General**” Module.
3. Select Module Definition “**Change**” button.
4. Select “**Extended Process Data – Exclusive Owner**” option from list.
5. Select “**SINT**” Data type.
6. Click “**OK**” button of Module Definition.
7. Click “**OK**” button of EDS configuration window.

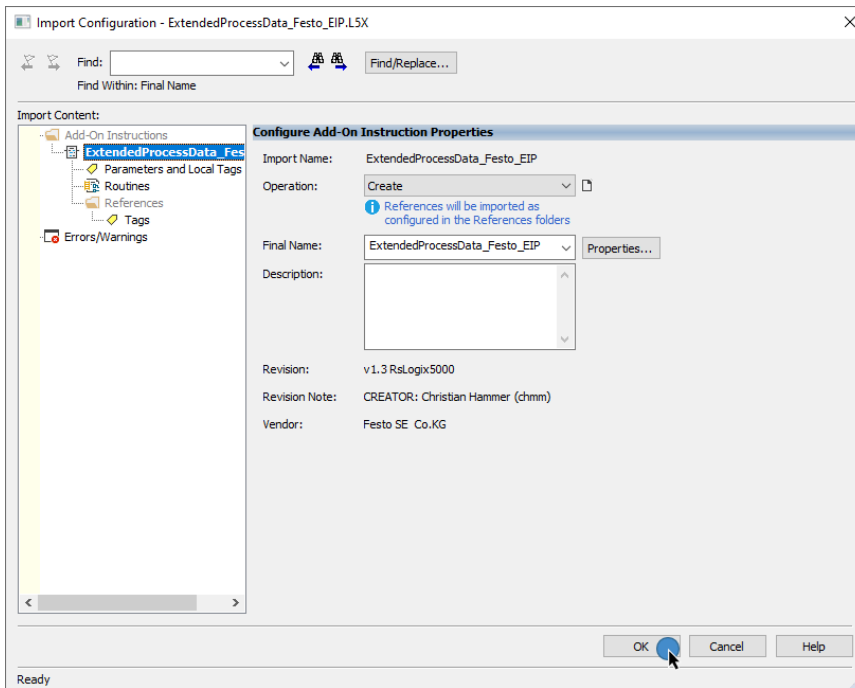
## 10.4.2 Import EPD AOI & Configure IO Tags

### Step – 1



1. Select “**Add-On Instructions**” from controller organizer menu & Right click.
2. Select “**Import Add-On Instruction**” from right click menu
3. Browse the “**ExtendedProcessData\_Festo\_EIP.L5X**” file folder location. All AOI files will be in “**20\_AOI's**” folder in Library directory.
4. Select “**ExtendedProcessData\_Festo\_EIP.L5X**” file.
5. Click “**Open**” button in Import Add-On Instruction window.

## Step – 2



Click “OK” button.

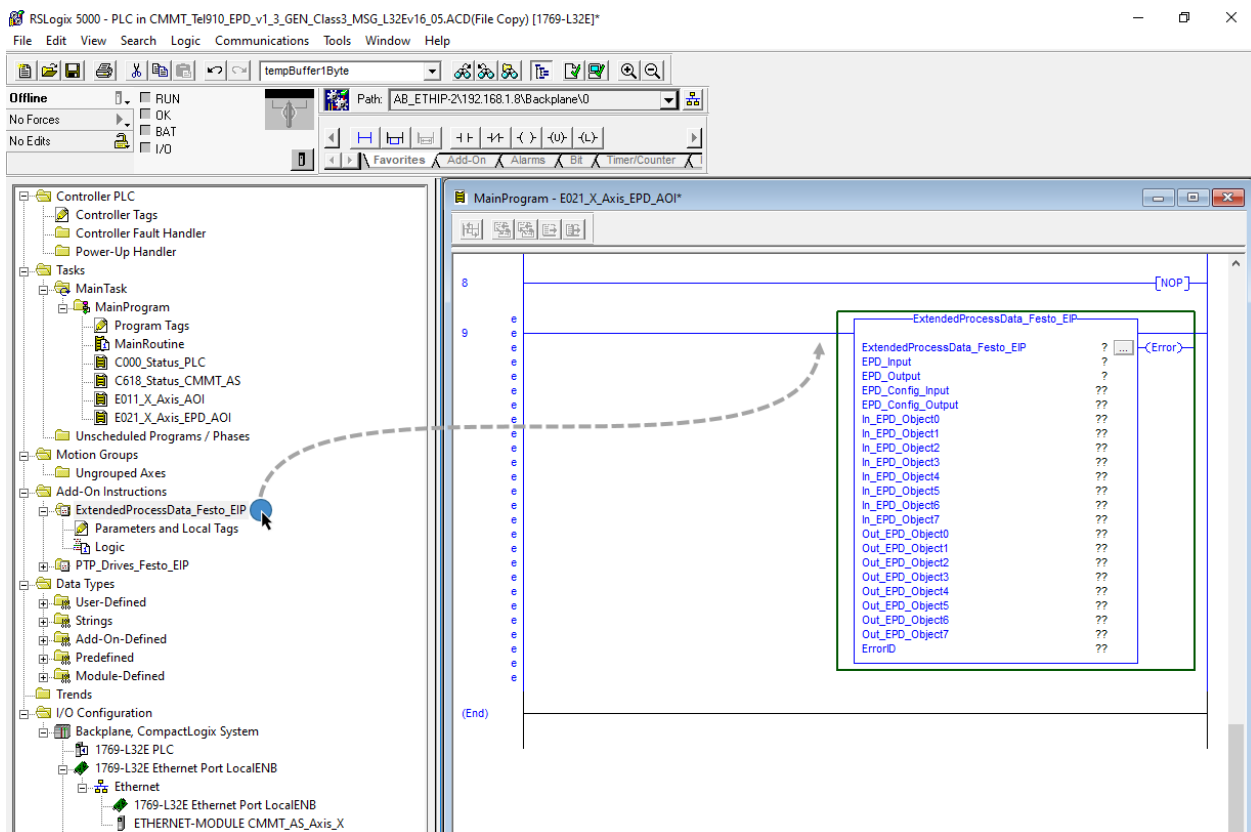
### 10.4.3 Add “ExtendedProcessData\_Festo\_EIP” AOI in PLC Program



#### Note

- The Add-On Instruction must be called cyclically with a separate instance for every servo drive integrated in the user program (every axis). Simultaneous use of multiple instances of the same device is not permitted.
- Any AOI's need to be in a continuous task called cyclically. ([Refer Chapter 3.5 Step – 1](#))

#### Step – 1



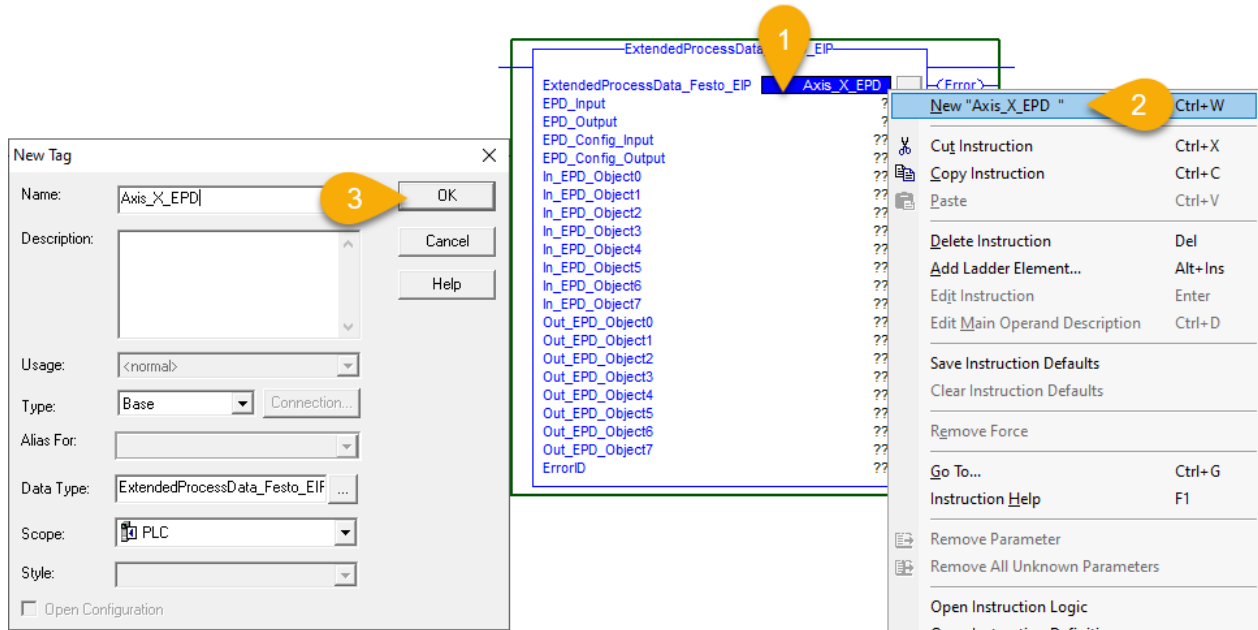
Drag and drop the “**ExtendedProcessData\_Festo\_EIP**” AOI file from Add-On Instructions folder as shown above.



#### Note

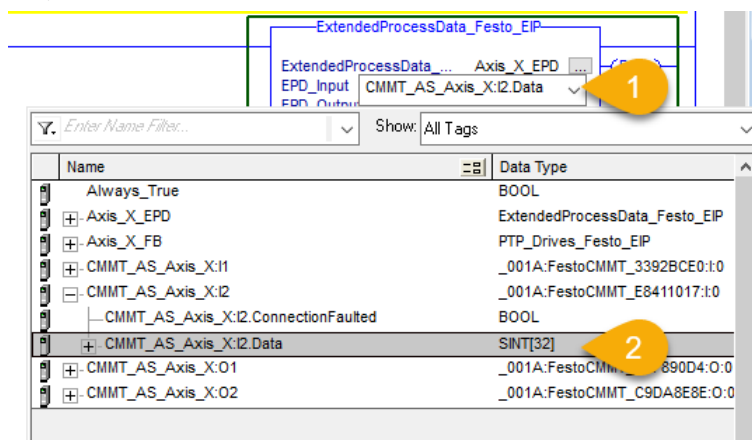
- This Add-On Instruction (AOI) has been designed to support as many Rockwell controllers as possible. Therefore, 64-bit data types have not been used. This Add-On Instruction only supports parameters with a maximum length of 32 bits (4 bytes).

## Step – 2



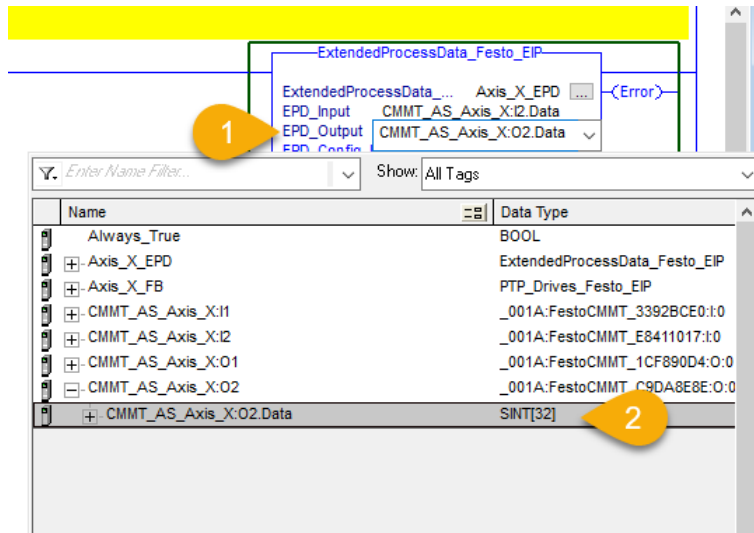
1. Enter the Tag name for the “**ExtendedProcessData\_Festo\_EIP**” AOI & right click.
2. Select “**New**” option with tag name from right click menu.
3. Click “**OK**” button from New Tag window.

## Step – 3



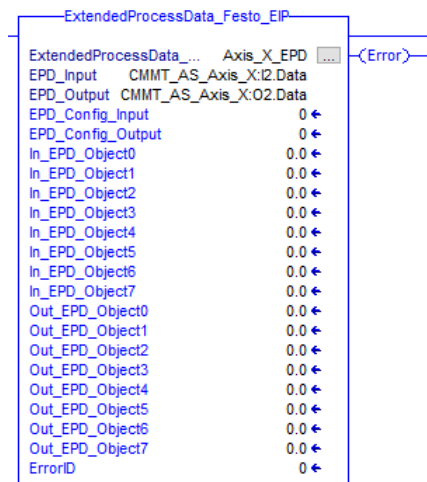
1. Select the drop-down menu of “**EPD\_Input**”.
2. Select the CMMT second element input tag as shown above. Which created in [chapter – 10.4.1](#).

#### Step – 4



1. Select the drop-down menu of “EPD\_Output”.
2. Select the CMMT second element output tag as shown above. Which created in [chapter – 10.4.1](#).

Once the above steps are completed successfully the AOI looks like below.



## 10.5 “ExtendedProcessData\_Festo\_EIP” AOI Input & Output Tag Details

### 10.5.1 Interface Tag Details

| Interface Area Tags | Data Type               | Function Description                                                                                        |
|---------------------|-------------------------|-------------------------------------------------------------------------------------------------------------|
| EPD_Input           | ARRAY OF SINT [ 0.. 32] | EPD Input data from servo drive ( Input must be linked to the controller tag of the closed-loop controller) |
| EPD_Output          | ARRAY OF SINT [ 0.. 32] | EPD Output data to servo drive ( Output must be linked to the controller tag of the closed-loop controller) |

### 10.5.2 Input Tag Details

| Input Tags        | Data Type | Function Description                                                                          |
|-------------------|-----------|-----------------------------------------------------------------------------------------------|
| EPD_Config_Input  | DINT      | Configuration of the EPD input data ( See Chapter-10.2 to configure EPD Input & Output Data)  |
| EPD_Config_Output | DINT      | Configuration of the EPD output data ( See Chapter-10.2 to configure EPD Input & Output Data) |
| Out_EPD_Object0   | REAL      | EPD output object 0 (output data to actuator controller)                                      |
| Out_EPD_Object1   | REAL      | EPD output object 1 (output data to actuator controller)                                      |
| Out_EPD_Object2   | REAL      | EPD output object 2 (output data to actuator controller)                                      |
| Out_EPD_Object3   | REAL      | EPD output object 3 (output data to actuator controller)                                      |
| Out_EPD_Object4   | REAL      | EPD output object 4 (output data to actuator controller)                                      |
| Out_EPD_Object5   | REAL      | EPD output object 5 (output data to actuator controller)                                      |
| Out_EPD_Object6   | REAL      | EPD output object 6 (output data to actuator controller)                                      |
| Out_EPD_Object7   | REAL      | EPD output object 7 (output data to actuator controller)                                      |

### 10.5.3 Output Tag Details

| Output Tags    | Data Type | Function Description                                                                                                                                                                  |
|----------------|-----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| In_EPD_Object0 | REAL      | EPD input object 0 (input data from the actuator controller)                                                                                                                          |
| In_EPD_Object1 | REAL      | EPD input object 1 (input data from the actuator controller)                                                                                                                          |
| In_EPD_Object2 | REAL      | EPD input object 2 (input data from the actuator controller)                                                                                                                          |
| In_EPD_Object3 | REAL      | EPD input object 3 (input data from the actuator controller)                                                                                                                          |
| In_EPD_Object4 | REAL      | EPD input object 4 (input data from the actuator controller)                                                                                                                          |
| In_EPD_Object5 | REAL      | EPD input object 5 (input data from the actuator controller)                                                                                                                          |
| In_EPD_Object6 | REAL      | EPD input object 6 (input data from the actuator controller)                                                                                                                          |
| In_EPD_Object7 | REAL      | EPD input object 7 (input data from the actuator controller)                                                                                                                          |
| Error          | BOOL      | A malfunction has occurred during the processing <ul style="list-style-type: none"> <li>• TRUE – Malfunction active (See ErrorID output)</li> <li>• FALSE – No Malfunction</li> </ul> |
| ErrorID        | INT       | Error Number. See <a href="#">Chapter – 10.7</a>                                                                                                                                      |

## 10.6 Configuration of the EPD Input and EPD Output Data

A length must be specified in order to inform the EPD function block of the correct data length of the individual parameters. This is done via “EPD\_Config\_Input” and “EPD\_Config\_Output” input variables based on numerical input. Every number in this case represents a data type. The configuration must be read from left to right. The first number on the left position represents the size of the first parameter (object 0) in the Festo Automation Suite ( See [chapter - 10.8](#) application example). The following table shows the allocation between number and data type.

| Number Value | Data Type Rockwell | Data Type Festo Automation Suite | Size   |
|--------------|--------------------|----------------------------------|--------|
| 0            | Not Permissible    |                                  | -      |
| 1            | SINT               | BOOL                             | 1 Byte |
| 2            | USINT              | BOOL                             | 1 Byte |
| 3            | INT                | SINT16                           | 2 Byte |
| 4            | UINT               | UINT16                           | 2 Byte |
| 5            | DINT               | SINT32                           | 4 Byte |
| 6            | Not Permissible    |                                  | -      |
| 7            | Not Permissible    |                                  | -      |
| 8            | Not Permissible    |                                  | -      |
| 9            | REAL               | FLOAT32                          | 4 Byte |

## 10.7 Diagnostics

The following table shows the possible error number and the associated error description

| ID (Dec) | ID (Hex) | Description                                                           |
|----------|----------|-----------------------------------------------------------------------|
| 10       | 0x0A     | “EPD_Config_Input” Input variable is too long ( More than 8 digits )  |
| 11       | 0x0B     | “EPD_Config_Output” Input variable is too long ( More than 8 digits ) |
| 20       | 0x14     | Number “0” is not permitted in “EPD_Config_Input” Input variable      |
| 21       | 0x15     | Number “0” is not permitted in “EPD_Config_Output” Input variable     |
| 30       | 0x1E     | Specify unsupported data type in “EPD_Config_Input” input variable    |
| 31       | 0x1F     | Specify unsupported data type in “EPD_Config_Output” input variable   |
| 40       | 0x28     | Specify unknown data type in “EPD_Config_Input” input variable        |
| 41       | 0x29     | Specify unknown data type in “EPD_Config_Output” input variable       |

10.8 Application Example

The following diagrams show the configuration of the EPD in the Festo Automation Suite and the application of the function block in Rockwell Studio 5000.

PARAMETERIZATION CONTROL DIAGNOSIS

CMMT-ST-C8-1C-MP-S0  
CMMT-ST-C8-1C-MP-S0  
Path: 192.168.7.130  
Connected

Disconnect Plug-in PLC Control Enabled Disabled Powerstage Stop Acknowledge All Store on Device Load Factory Settings Reinitialize Restart Device

Parameter Pages < Extended Process Data Delete All Sent Data Delete All Received Data

Drive Configuration  
Device Settings  
Energy Recovery  
Application Data  
Fieldbus  
Profiles  
CIA 402  
PROFIdrive  
Factor Group  
Telegram  
AC4 (PROFINET)  
Extended Pro...

Digital I/O  
Encoder Interface  
Axis 1  
Parameter List

Status

Extended process data (EtherNet/IP) ☒ Active

Extended process data active ☒ Active EPD\_Config\_Input

Sent Data

| ID | Parameter                                          | Type | Byte position | Function Block ID |
|----|----------------------------------------------------|------|---------------|-------------------|
| 0  | P1.7118.0.0<br>Actual nominal current              | REAL | 0             | 9                 |
| 1  | P1.841.0.0<br>Move to axis zero point after homing | BOOL | 4             | 1                 |
| 2  | P1.4628.0.0<br>Software limit positions active     | BOOL | 5             | 1                 |
| 3  | P1.11280040.0.0<br>Error number                    | UINT | 6             | 4                 |
| 4  | P0.480.0.0<br>Actual value DC link voltage         | REAL | 8             | 9                 |

Add Process Channel

Number of bytes Tx 22

Number of bytes Tx (Offline) 12

PLC input configuration 91149

Received Data

| ID | Parameter                                  | Type | Byte position | Function Block ID |
|----|--------------------------------------------|------|---------------|-------------------|
| 0  | P1.11280701.0.0<br>Base value velocity     | REAL | 0             | 9                 |
| 1  | P1.11280702.0.0<br>Base value acceleration | REAL | 4             | 9                 |
| 2  | P1.11280703.0.0<br>Base value deceleration | REAL | 8             | 9                 |

Add Process Channel

Number of bytes Rx 28

Number of bytes Rx (Offline) 12

PLC output configuration 999

mapping of the  
ExtendedProcessData  
objects

ExtendedProcessData\_Festo\_EIP

Festo\_ExtendedProces... fb\_EP

EPD\_Input CMMT\_AS\_EP:I2.Data

EPD\_Output CMMT\_AS\_EP:O2.Data

EPD\_Config\_Input 91149

EPD\_Config\_Output 999

In\_EP\_Object0 1.4

In\_EP\_Object1 1.0

In\_EP\_Object2 0.0

In\_EP\_Object3 229.0

In\_EP\_Object4 335.6958

In\_EP\_Object5 0.0

In\_EP\_Object6 0.0

In\_EP\_Object7 0.0

Out\_EP\_Object0 123.0

Out\_EP\_Object1 456.0

Out\_EP\_Object2 789.0

Out\_EP\_Object3 0.0

Out\_EP\_Object4 0.0

Out\_EP\_Object5 0.0

Out\_EP\_Object6 0.0

Out\_EP\_Object7 0.0

ErrorID 0

Error

“ExtendedProcessData\_Festo\_EIP” configuration in Rockwell Studio-5000.

## 10.9 Extended Process Data 64Bit AOI Read/Write using an EDS File



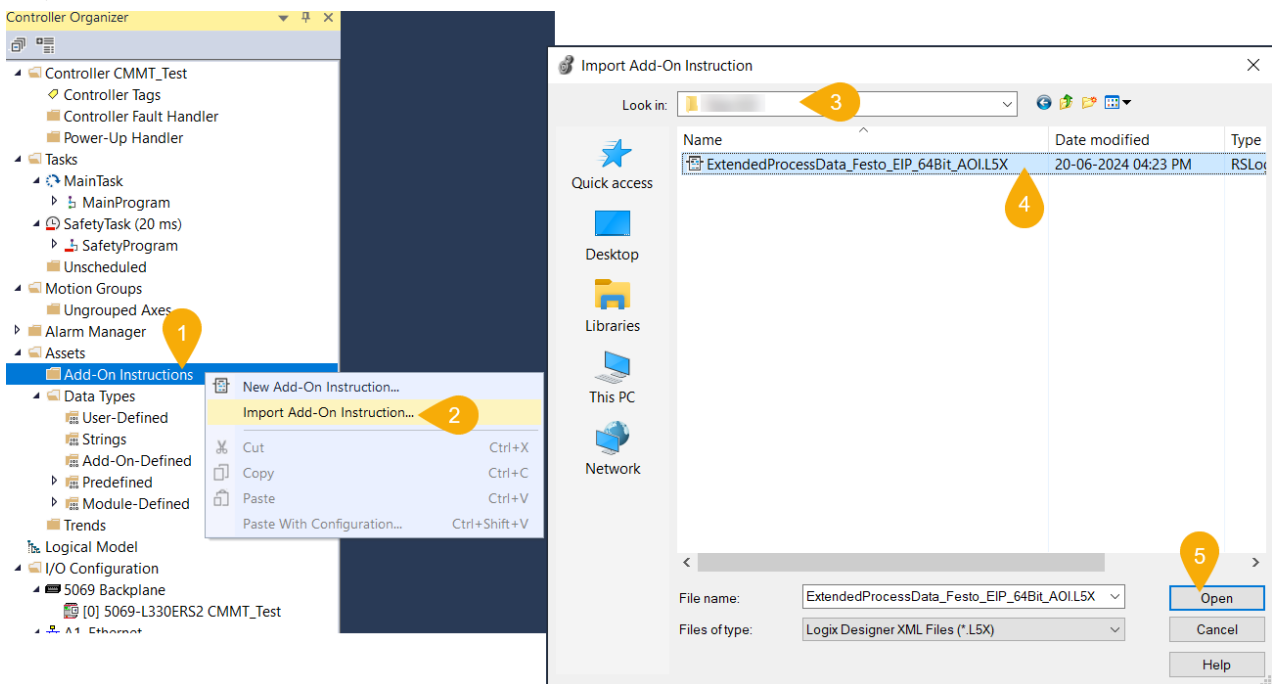
### Note

- This Add-On Instruction (AOI) has been designed to support 64bit Rockwell controllers for read and write the CMMT LINT data. All Input and Outputs data are configured as LREAL data type, so From Studio V33 will be supported for the Rockwell 64bit Controllers.

- Repeat the same procedure from [Chapter-10.4.1](#) if user configuring EDS file for EPD.

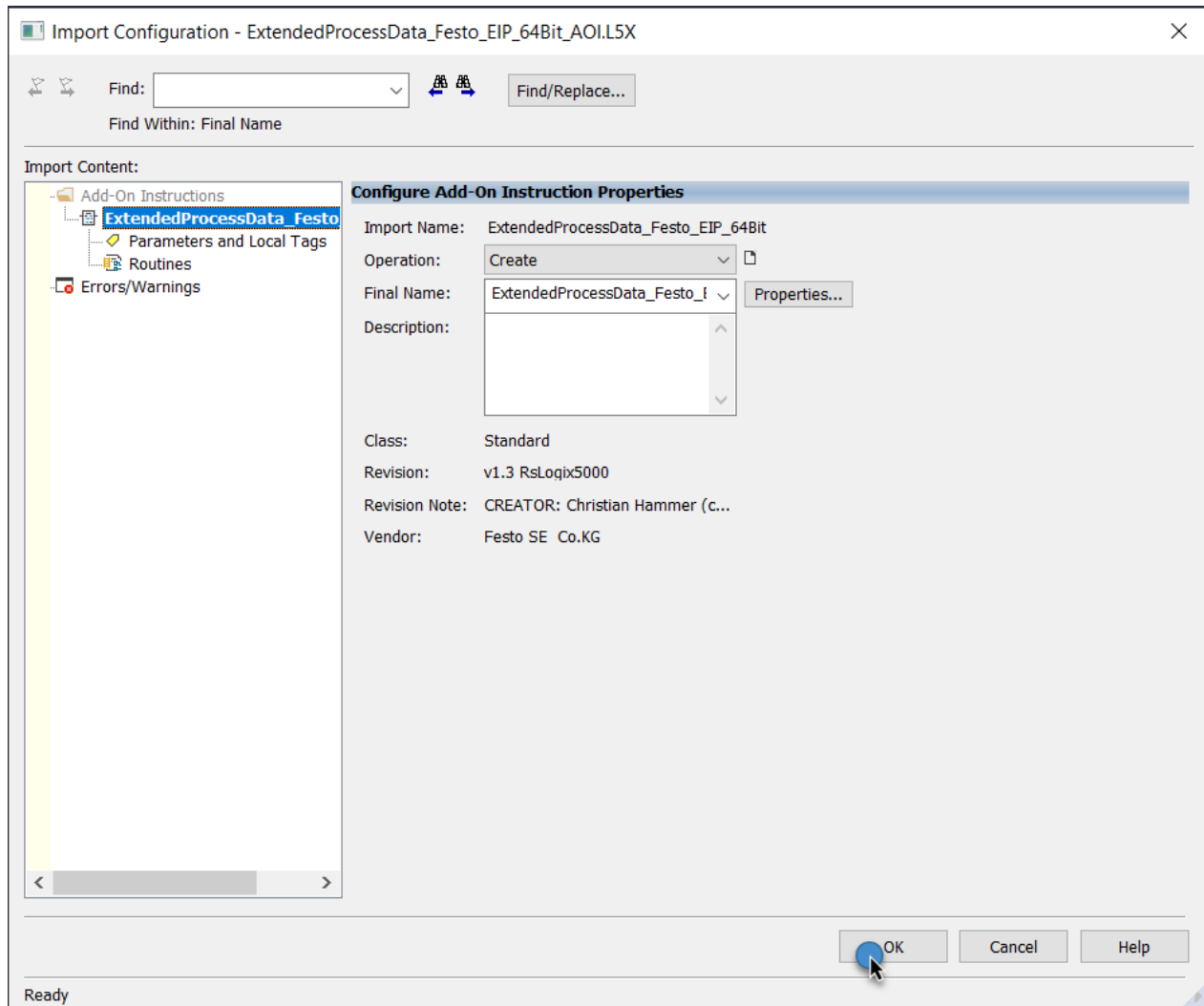
### 10.9.1 Import EPD 64Bit AOI & Configuring IO Tags

#### Step – 1



1. Select “**Add-On Instructions**” from controller organizer menu & Right click.
2. Select “**Import Add-On Instruction**” from right click menu
3. Browse the “**ExtendedProcessData\_Festo\_EIP\_64Bit\_AOI.L5X**” file folder location.
4. Select “**ExtendedProcessData\_Festo\_EIP\_64Bit\_AOI.L5X**” file.
5. Click “**Open**” button in Import Add-On Instruction window.

## Step – 2



- Click **OK** to complete the file Import.

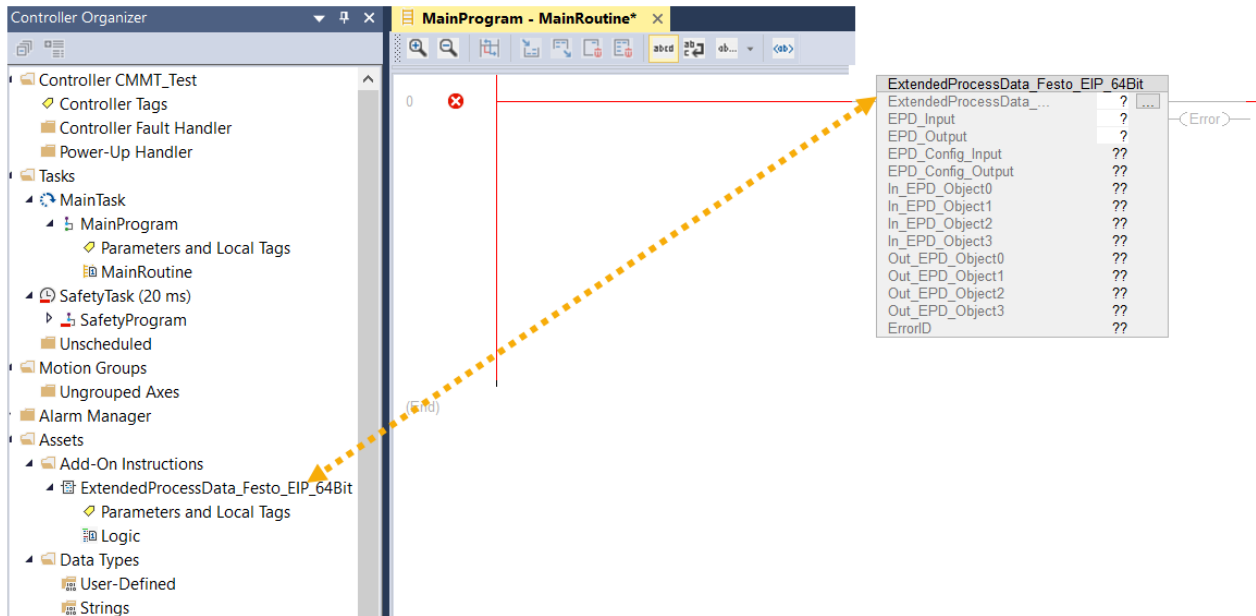
## 10.9.2 Adding “ExtendedProcessData\_Festo\_EIP\_64Bit\_AOI” in PLC Program



### Note

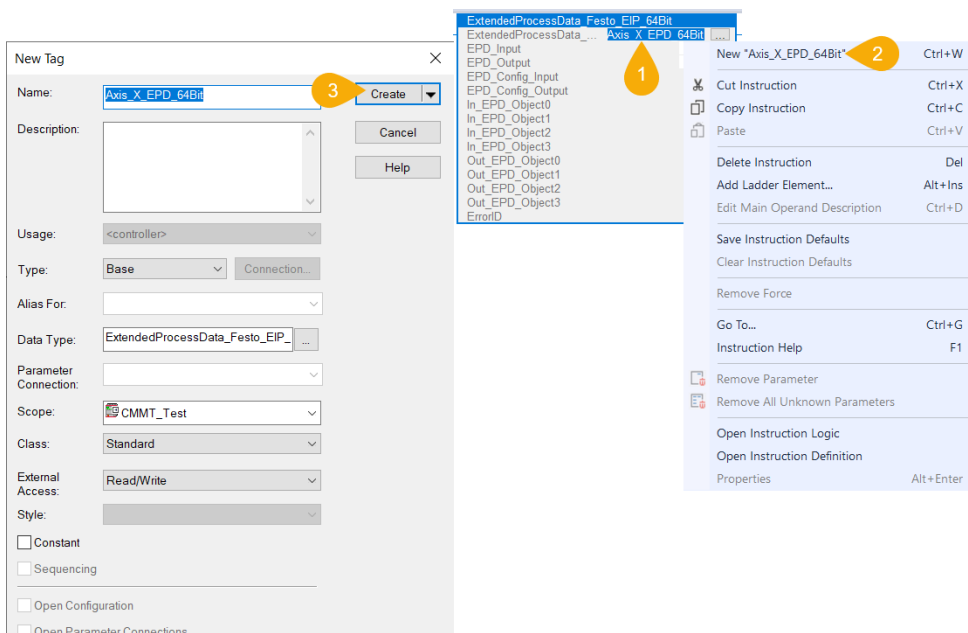
- The Add-On Instruction must be called cyclically with a separate instance for every servo drive integrated in the user program (every axis). Simultaneous use of multiple instances of the same device is not permitted.
- Any AOI's need to be in a continuous task called cyclically. ([Refer Chapter 3.5 Step – 1](#))

### Step – 1



Drag and drop the “**ExtendedProcessData\_Festo\_EIP\_64Bit\_AOI**” file from Add-On Instructions folder as shown above.

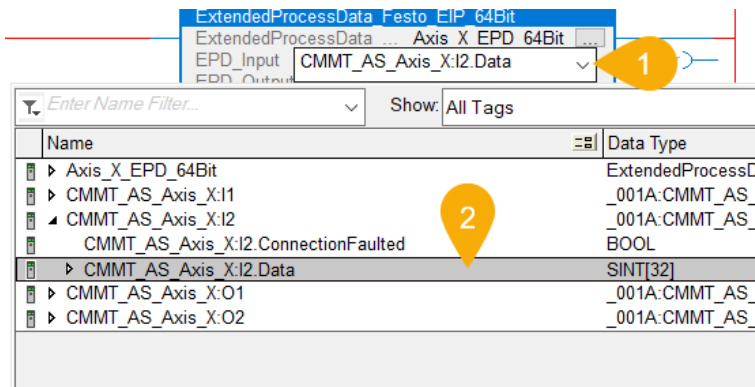
### Step – 2



1. Enter the Tag name for the “**ExtendedProcessData\_Festo\_EIP\_64Bit**” AOI & right click.

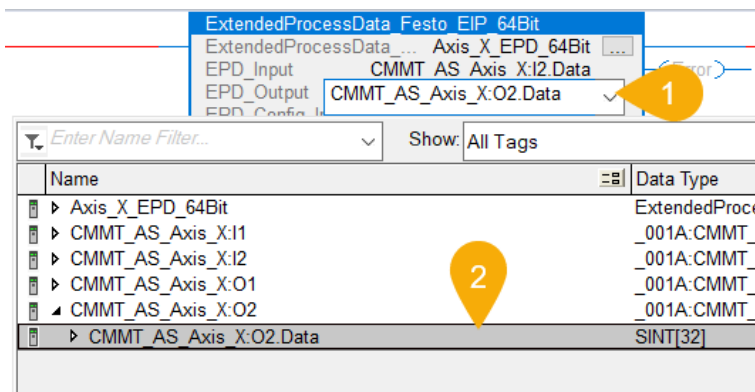
2. Select “**New**” option with tag name from right click menu.
3. Click “**OK**” button from New Tag window.

### Step – 3



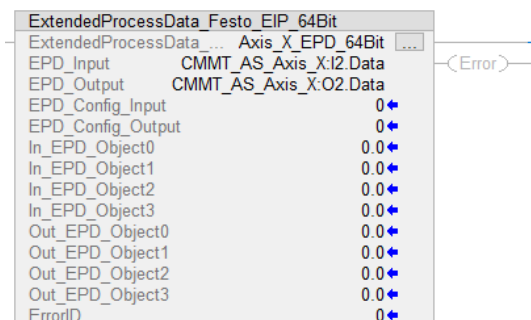
1. Select the drop-down menu of “**EPD\_Input**”.
2. Select the CMMT second element input tag as shown above. Which created in [chapter – 10.4.1](#).

### Step – 4



1. Select the drop-down menu of “**EPD\_Output**”.
2. Select the CMMT second element output tag as shown above. Which created in [chapter – 10.4.1](#).

Once the above steps are completed successfully the AOI looks like below.



## 10.10 “ExtendedProcessData\_Festo\_EIP\_64Bit” AOI Input & Output Tag Details

### 10.10.1 Interface Tag Details

| Interface Area Tags | Data Type               | Function Description                                                                                        |
|---------------------|-------------------------|-------------------------------------------------------------------------------------------------------------|
| EPD_Input           | ARRAY OF SINT [ 0.. 32] | EPD Input data from servo drive ( Input must be linked to the controller tag of the closed-loop controller) |
| EPD_Output          | ARRAY OF SINT [ 0.. 32] | EPD Output data to servo drive ( Output must be linked to the controller tag of the closed-loop controller) |

### 10.10.2 Input Tag Details

| Input Tags        | Data Type | Function Description                                                                          |
|-------------------|-----------|-----------------------------------------------------------------------------------------------|
| EPD_Config_Input  | DINT      | Configuration of the EPD input data ( See Chapter-10.2 to configure EPD Input & Output Data)  |
| EPD_Config_Output | DINT      | Configuration of the EPD output data ( See Chapter-10.2 to configure EPD Input & Output Data) |
| Out_EPD_Object0   | LREAL     | EPD output object 0 (output data to actuator controller)                                      |
| Out_EPD_Object1   | LREAL     | EPD output object 1 (output data to actuator controller)                                      |
| Out_EPD_Object2   | LREAL     | EPD output object 2 (output data to actuator controller)                                      |
| Out_EPD_Object3   | LREAL     | EPD output object 3 (output data to actuator controller)                                      |

### 10.10.3 Output Tag Details

| Output Tags    | Data Type | Function Description                                         |
|----------------|-----------|--------------------------------------------------------------|
| In_EPD_Object0 | LREAL     | EPD input object 0 (input data from the actuator controller) |
| In_EPD_Object1 | LREAL     | EPD input object 1 (input data from the actuator controller) |
| In_EPD_Object2 | LREAL     | EPD input object 2 (input data from the actuator controller) |
| In_EPD_Object3 | LREAL     | EPD input object 3 (input data from the actuator controller) |

| Output Tags | Data Type | Function Description                                                                                                                                                            |
|-------------|-----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Error       | BOOL      | A malfunction has occurred during the processing <ul style="list-style-type: none"> <li>TRUE – Malfuction active (See ErrorID output)</li> <li>FALSE – No Malfuction</li> </ul> |
| ErrorID     | INT       | Error Number. See <a href="#">Chapter – 10.7</a>                                                                                                                                |

### 10.11 Configuration of the EPD Input and EPD Output Data

A length must be specified in order to inform the EPD function block of the correct data length of the individual parameters. This is done via “EPD\_Config\_Input” and “EPD\_Config\_Output” input variables based on numerical input. Every number in this case represents a data type. The configuration must be read from left to right. The first number on the left position represents the size of the first parameter (object 0) in the Festo Automation Suite ( See [chapter - 10.13](#) application example). The following table shows the allocation between number and data type.

| Number Value | Data Type Rockwell | Size   |
|--------------|--------------------|--------|
| 0            | Not Permissible    | -      |
| 1            | SINT               | 1 Byte |
| 2            | USINT              | 1 Byte |
| 3            | INT                | 2 Byte |
| 4            | UINT               | 2 Byte |
| 5            | DINT               | 4 Byte |
| 6            | UDINT              | 4 Byte |
| 7            | LINT               | 8 Byte |
| 8            | ULINT              | 8Byte  |
| 9            | REAL               | 4 Byte |

## 10.12 Diagnostics

The following table shows the possible error number and the associated error description

| ID (Dec) | ID (Hex) | Description                                                           |
|----------|----------|-----------------------------------------------------------------------|
| 10       | 0x0A     | “EPD_Config_Input“ Input variable is too long ( More than 4 digits )  |
| 11       | 0x0B     | “EPD_Config_Output“ Input variable is too long ( More than 4 digits ) |
| 20       | 0x14     | Number “0“ is not permitted in “EPD_Config_Input“ Input variable      |
| 21       | 0x15     | Number “0“ is not permitted in “EPD_Config_Output“ Input variable     |
| 30       | 0x1E     | Specify unsupported data type in “EPD_Config_Input“ input variable    |
| 31       | 0x1F     | Specify unsupported data type in “EPD_Config_Output“ input variable   |
| 40       | 0x28     | Specify unknown data type in “EPD_Config_Input“ input variable        |
| 41       | 0x29     | Specify unknown data type in “EPD_Config_Output“ input variable       |

## 10.13 Application Example

The following diagrams show the configuration of the EPD in the Festo Automation Suite and the application of the function block in Rockwell Studio 5000.

**Parameterization** CONTROL DIAGNOSIS

CMMT-AS-C2-3A-MP-S1  
CMMT-AS-C2-3A-MP-S1  
Path: 192.168.1.81  
Connected

Disconnect Plug-in PLC Control Enabled Disabled Powerstage Stop Acknowledge All Store on Device Load Factory Settings Reinitialize Restart Device

Parameter Pages < Extended Process Data Delete All Sent Data Delete All Received Data

**Status**

Extended process data (EtherNet/IP) ☒ Active

Extended process data active ☒ Active

**Sent Data**

| ID | Parameter                   | Type | Byte position | Function Block ID |
|----|-----------------------------|------|---------------|-------------------|
| 0  | Referencing method          | DINT | 0             | 5                 |
| 1  | Base value velocity         | REAL | 4             | 9                 |
| 2  | Axis zero point offset      | LINT | 8             | 7                 |
| 3  | Activation Save zero offset | BOOL | 16            | 1                 |

**Received Data**

| ID | Parameter                        | Type | Byte position | Function Block ID |
|----|----------------------------------|------|---------------|-------------------|
| 0  | Negative software limit position | LINT | 0             | 7                 |
| 1  | Positive software limit position | LINT | 8             | 7                 |
| 2  | Clamping torque                  | REAL | 16            | 9                 |
| 3  | Homing timeout                   | REAL | 20            | 9                 |

Number of bytes Tx 17

Number of bytes Tx (Offline) 17

PLC input configuration 5971

Number of bytes Rx 24

Number of bytes Rx (Offline) 24

PLC output configuration 7799

EPD\_Config\_Input

EPD\_Config\_Output

| ExtendedProcessData_Festo_EIP_64Bit |                     |
|-------------------------------------|---------------------|
| ExtendedProcessData_...             | ExtendProcessData64 |
| EPD_Input                           | CMMT_AS:I2.Data     |
| EPD_Output                          | CMMT_AS:O2.Data     |
| EPD_Config_Input                    | 5971                |
| EPD_Config_Output                   | 7799                |
| In_EPD_Object0                      | 37.0                |
| In_EPD_Object1                      | 3000.0              |
| In_EPD_Object2                      | 30000000.0          |
| In_EPD_Object3                      | 1.0                 |
| Out_EPD_Object0                     | -350000000000.0     |
| Out_EPD_Object1                     | 350000000000.0      |
| Out_EPD_Object2                     | 1.0                 |
| Out_EPD_Object3                     | 60.0                |
| ErrorID                             | 0                   |

“ExtendedProcessData\_Festo\_EIP\_64Bit” configuration in Rockwell Studio-5000.

## 11 “CMMT Full Parameter AOI” / Full Parameter of CMMT Drive



### Note

Required Stable Version of Festo Automation Suite(FAS) **V2.6.0.380** and above and the Plug-ins.

### 11.1 “CMMT Full Parameter” AOI Introduction

This function block is generated from the Festo Automation Suite(FAS). The “CMMT Full Parameter” AOI is used to parameterize the new CMMT drive or existing CMMT Drive to restore previous parameter.

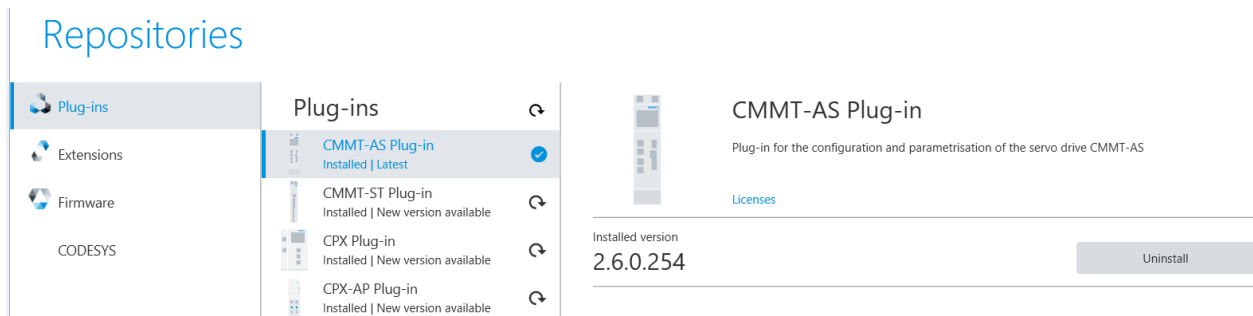
### 11.2 Configure FAS

#### Step – 1

Download latest FAS from support portal **V2.6.0.380** and above and Start with the installation of FAS.

#### Step – 2

After Installation done, continue with the download of **Plug-ins(CMMT-AS) V2.6.0.380** and above.



### Note

Repositories (requires Internet connection)

#### 11.2.1 Creating Rockwell file L5X

##### Step – 1

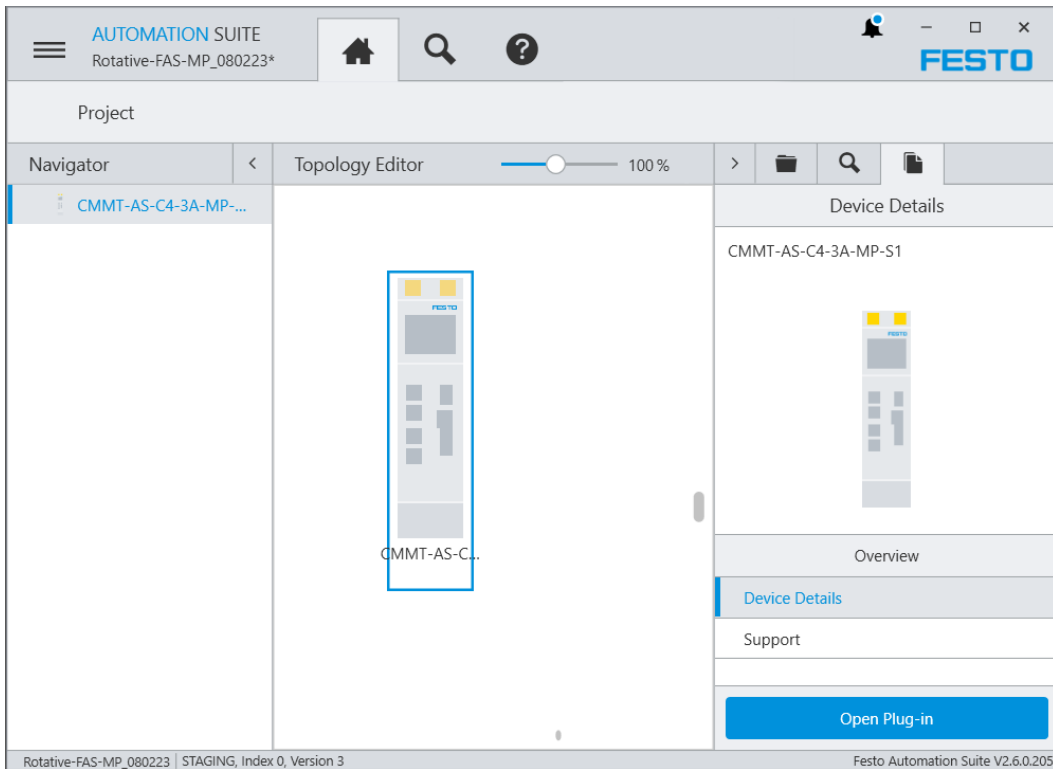
To create Rockwell file L5X, open the FAS project.



### Note

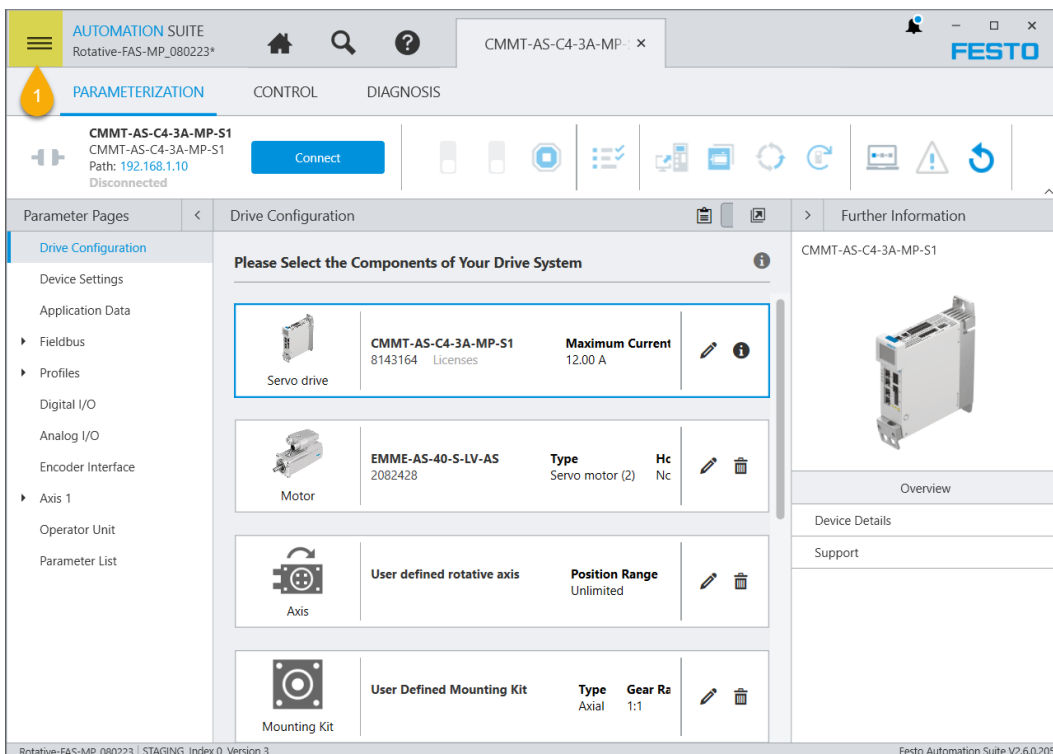
- Either Create New Project in FAS as per user settings (Refer [Chapter 2.4](#) and Refer [Chapter 2.5](#) for Integration with PLC), or Open previous saved project if available.
- In-case if user does not have previous project of Automation Suite then, read the settings from Device(CMMT) to FAS.

## Step – 2



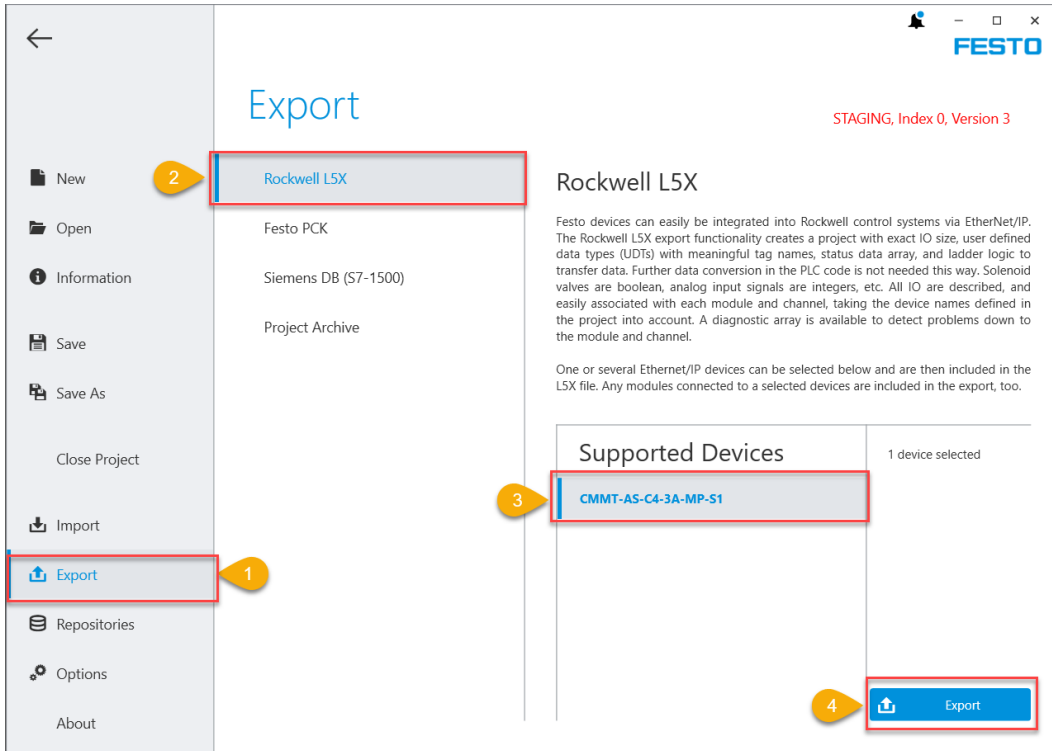
Double click on the project.

## Step – 3



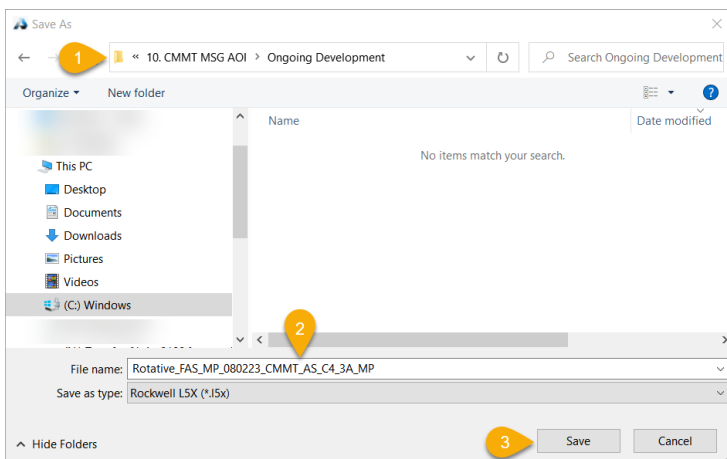
1. Click on the Menu button.

#### Step – 4



1. Click on Export
2. Select Rockwell L5X
3. Select CMMT-xx configuration which exist
4. Click on “Export”

#### Step – 5



1. Choose the location to save the Project
2. Enter the user desired **File name**.
3. Click on “Save”

### 11.3 Import CMMT Full Parameter AOI

This chapter will describe about importing the CMMT\_FullParam AOI file inside the PLC program.

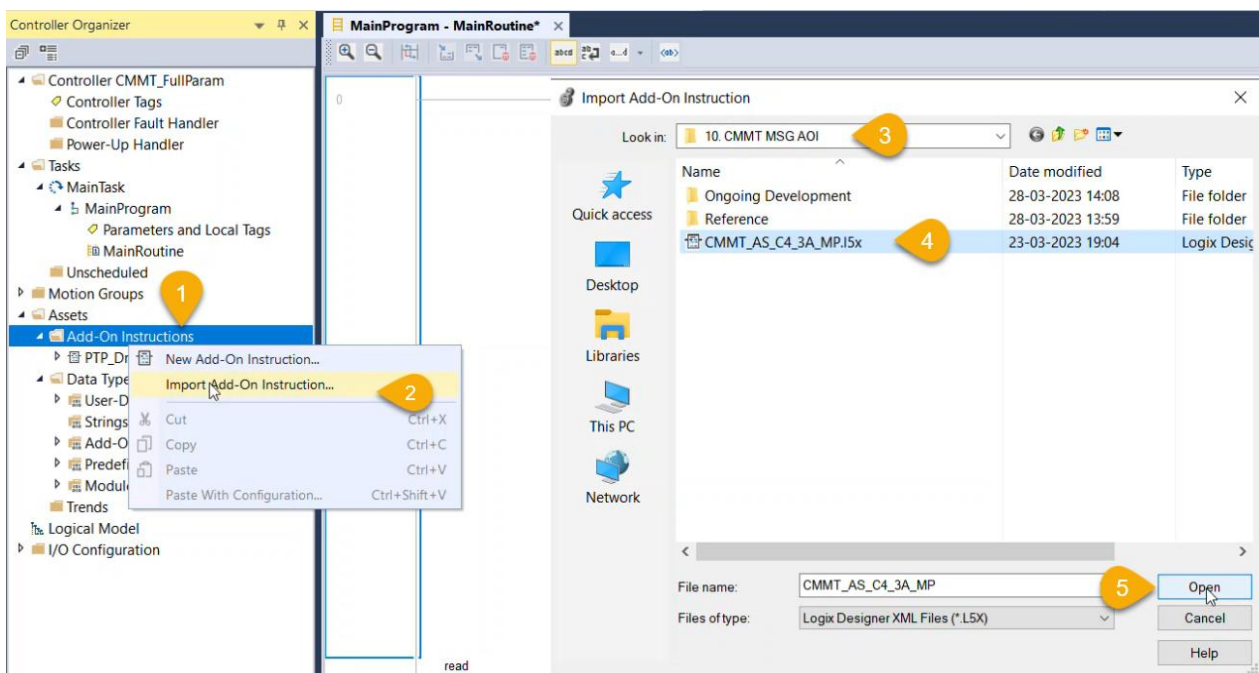


#### Note

There are hardware limitations with certain Allen Bradley controllers or EtherNet/IP masters. The AOI (Add-On Instruction) requires Socket Services which are ONLY available/supported with these Controllers/Modules listed here:

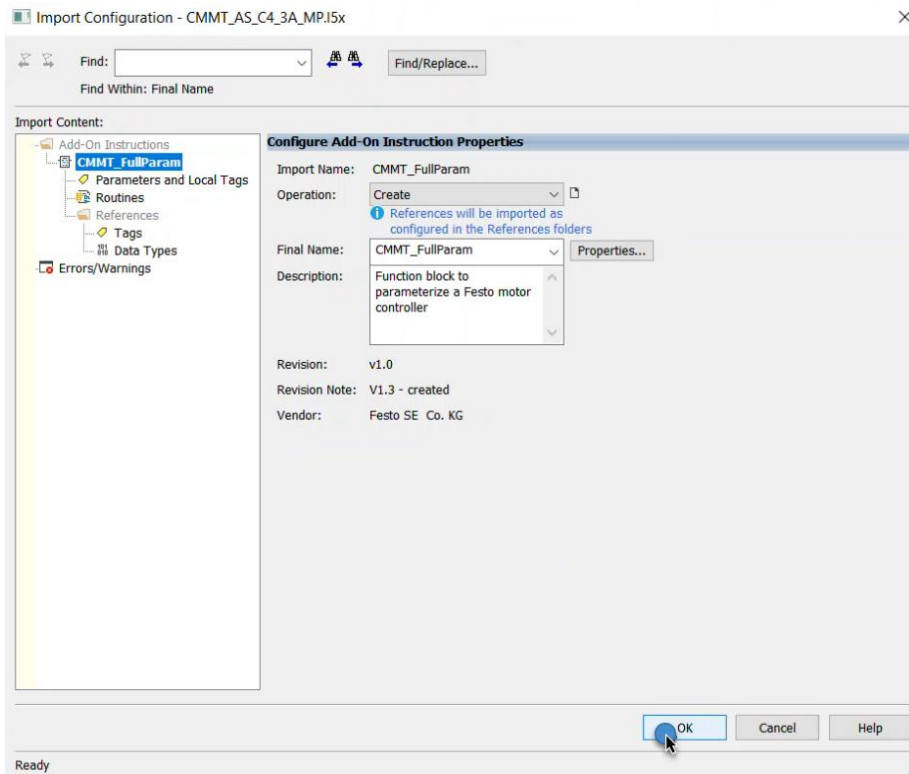
- ControlLogix® 5580 and GuardLogix® 5580 controllers
- CompactLogix™ 5380 and Compact GuardLogix 5380 controllers
- CompactLogix 5370 and Compact GuardLogix 5370 controllers
- CompactLogix 5480 controllers
- 1756-EN2x, 1756-EN3TR, and 1756-EN4TR EtherNet/IP communication modules
- 1756-EWEB and 1768-EWEB EtherNet/IP web server modules

#### Step – 1



1. Select “**Add-On Instructions**” from controller organizer menu & Right click.
2. Select “**Import Add-On Instruction**” from right click menu
3. Browse folder location to open the Project
4. Select “**CMMT\_As---.L5X**” file.
5. Click “**Open**” button in Import Add-On Instruction window.

## Step – 2



Click “OK” button.

## 11.4 Replacing pre-existing AOI by Overwriting

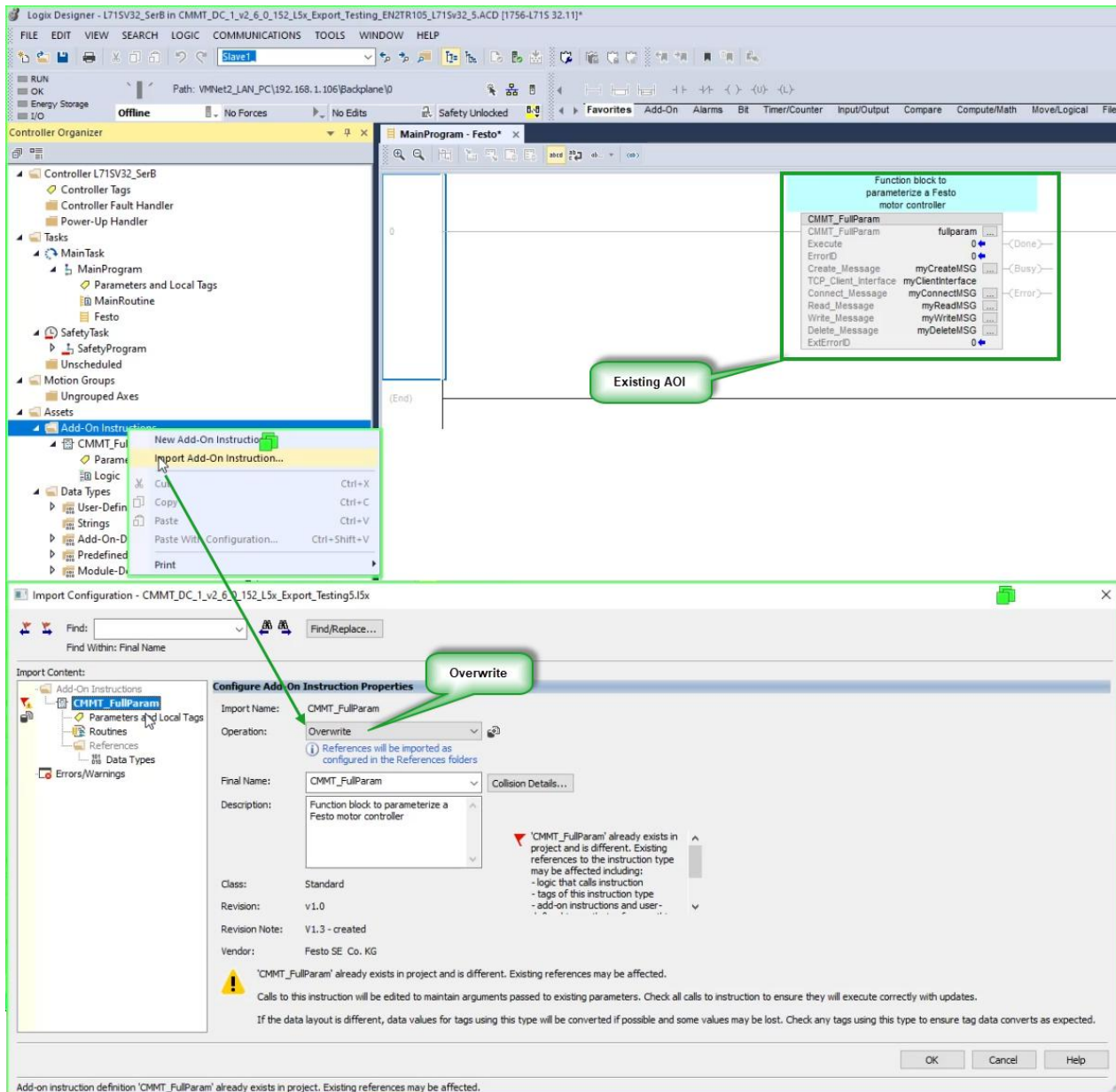


### Note

- In the event that this is the very first time importing AOI into a PLC programme, skip this [chapter 11.4](#) and jump to next [chapter 11.5](#)

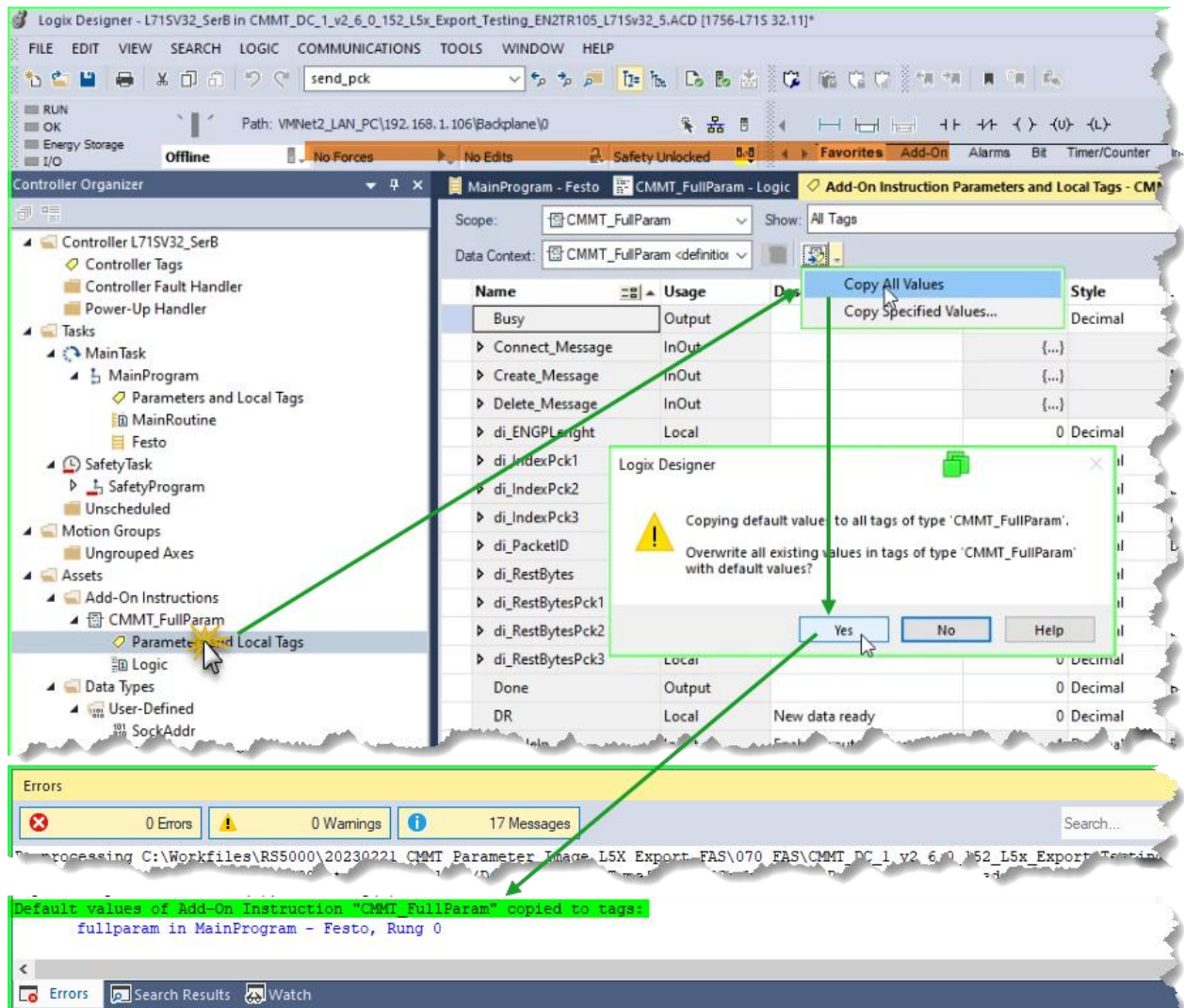
### ISSUE:

A project has a pre-existing AOI and the intention is to overwrite UPDATED VALUES with another import. The user selected Overwrite, however, existing invocations of the instruction are NOT UPDATED.



## SOLUTION:

After the Overwrite import is complete, the user needs to double click the “Parameters and Local Tags” as shown below to provoke an alternative interface popup and then Copy All Values and Accept this task. This will update all existing invocations of the instruction as shown here:



The Rockwell help file describes this in the following manner.

### Copy definition default values

Displays options to copy all default parameter and local tag values to all instance tags or display a dialog you can use to copy specific default values to specific instance tags (including copying specific default values to all instance tags or copying all default values to specific instance tags).

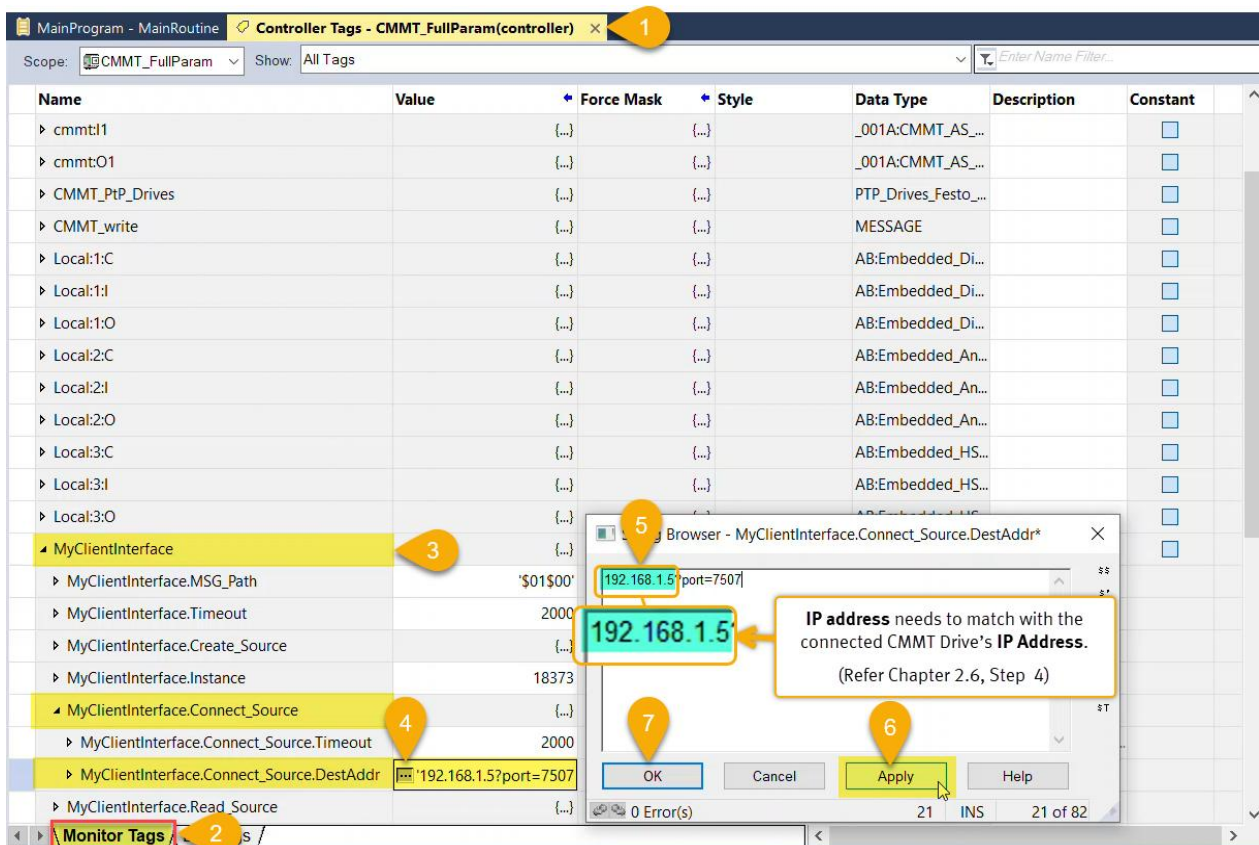
#### 11.4.1 Step to modify IP address of CMMT-xx in MyClientInterface:



##### Note

- Cross check the IP address entered in “MyClientInterface.Connect\_Source.DestAddr” is matching with the connected CMMT Drive’s IP Address, if IP Address is mismatching, then enter the correct IP address of connected Drive’s.
- Syntax and Port no will remain same in all situations, only **IP address** needs to match with the connected CMMT Drive’s **IP Address**. ([Refer Chapter 2.5, Step – 4](#))
- Incase if user misses to enter correct IP address of connected Drive’s, the AOI will malfunction.

Image shown below for updating IP address.



1. Open **Controller Tags** window.
2. Select **Monitor Tags**
3. Expand **MyClientInterface** Tag and expand **MyClientInterface.Connect\_Source**
4. Select **MyClientInterface.Connect\_Source.DestAddr** and click on the three dot to modify the IP address.
5. Enter the correct IP address of connected CMMT Drive’s.  
(Syntax and Port no will remain same in all situations, only **IP address** needs to match with the connected CMMT Drive’s **IP Address**, [Refer Chapter 2.5, Step – 4](#))
6. Click **“Apply”** button to confirm the configuration.
7. Click **“OK”** button to confirm.

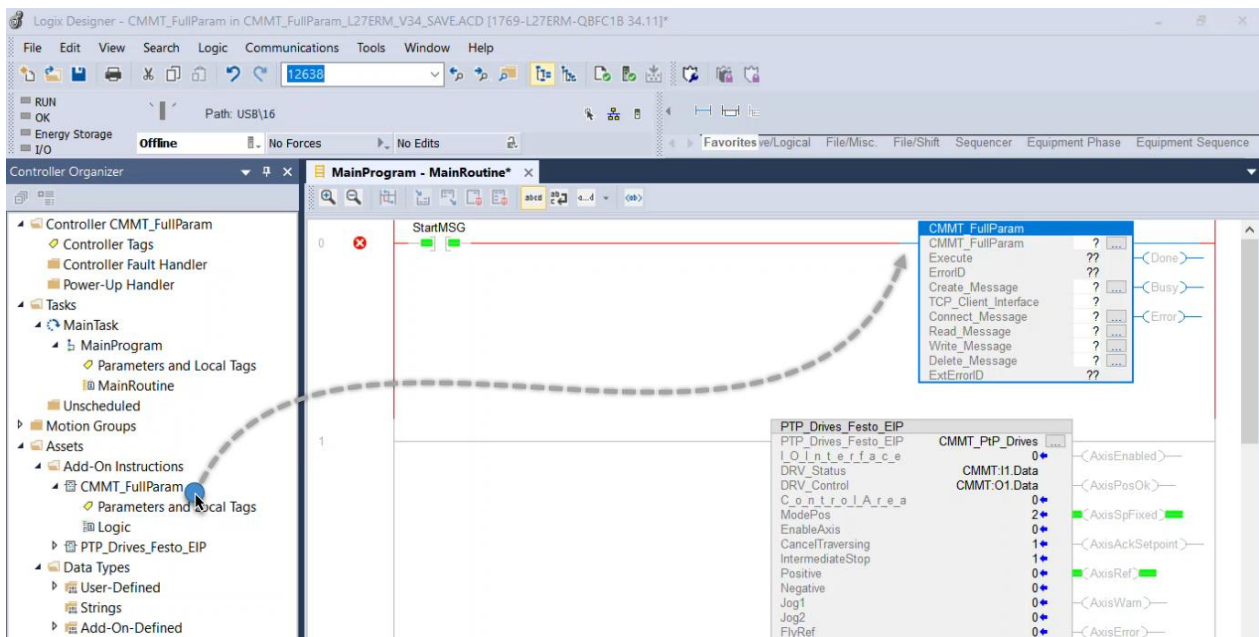
## 11.5 Add “CMMT\_FullParam” AOI in PLC Program



### Note

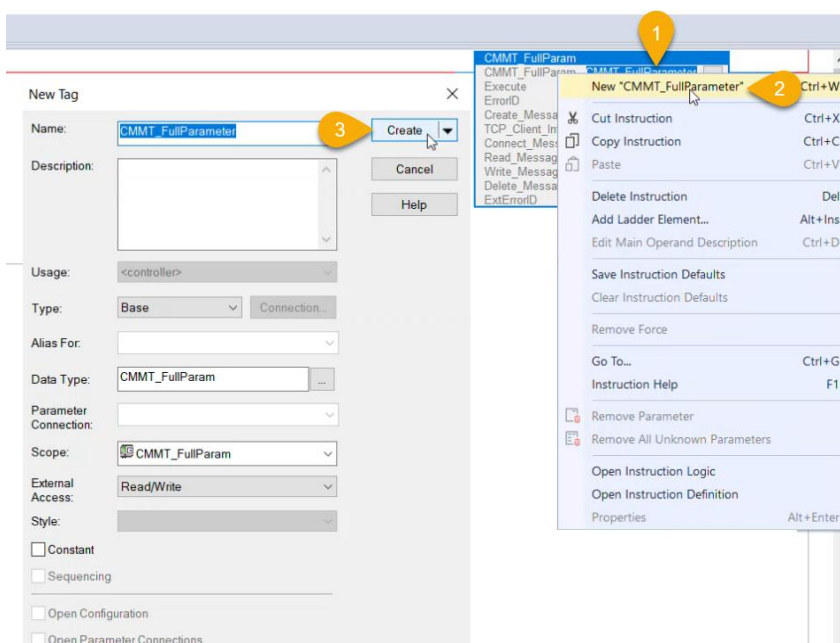
- The Add-On Instruction must be called cyclically with a separate instance for every servo drive integrated in the user program (every axis). Simultaneous use of multiple instances of the same device is not permitted.
- Any AOI's need to be in a continuous task called cyclically. ([Refer Chapter 3.5 Step – 1](#))

### Step – 1



Drag and drop the “CMMT\_FullParam” AOI file from Add-On Instructions folder as shown above.

### Step – 2

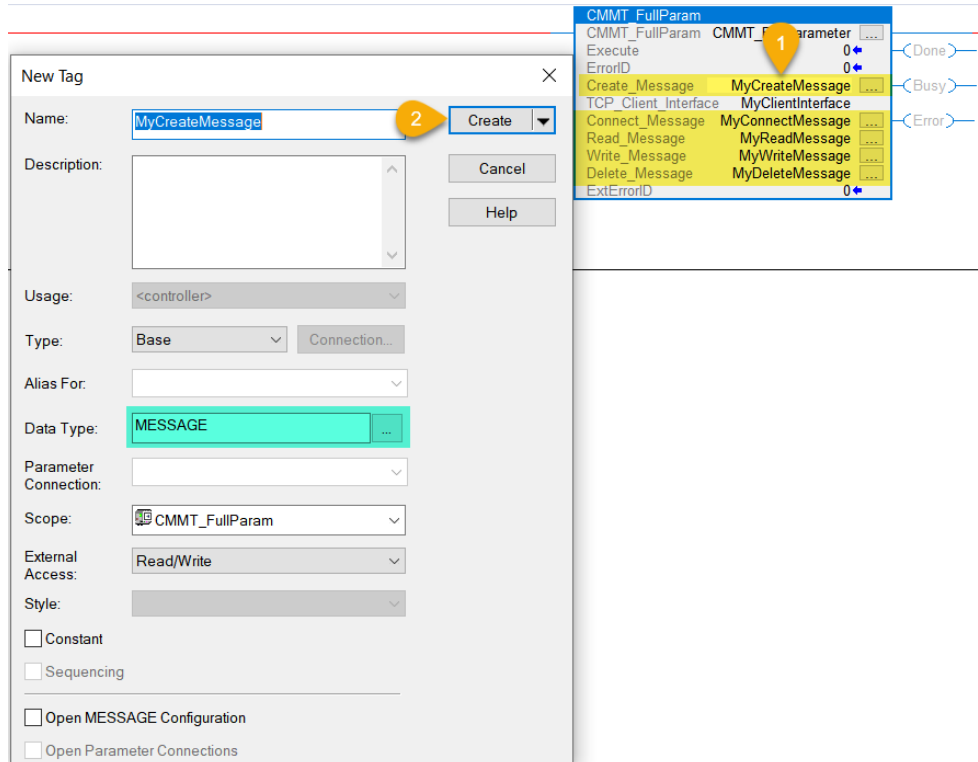


1. Enter the Tag name for the “CMMT\_FullParam” AOI & right click.

2. Select “**New**” option with tag name from right click menu.
3. Click “**Create**” button from New Tag window.

## 11.6 CMMT Full Parameter AOI Tag’s Configuration

### Step – 1



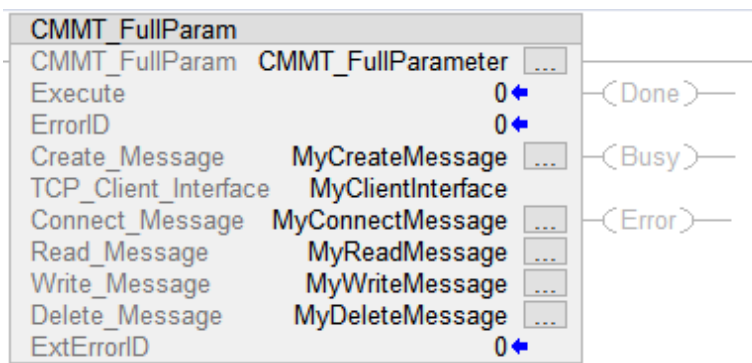
1. Enter the Tag name for the “**MyCreateMessage**” & right click. Select “**New**” option with tag name from right click menu.
- Click “**Create**” button on new tag window & ensure Data Type is “**MESSAGE**” as highlighted in green color.



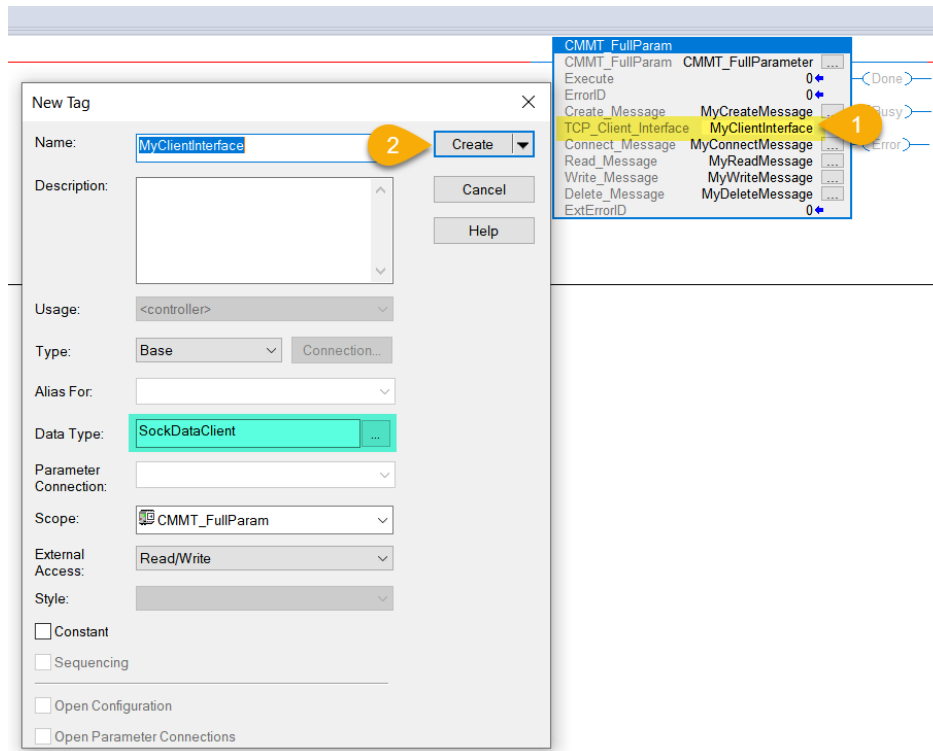
#### Note

- Follow the above step (step – 1) to create “**Connect\_Message, Read\_Message, Write\_Message, Delete\_Message**” interface tag.

Once the above steps are completed successfully the AOI looks like below.



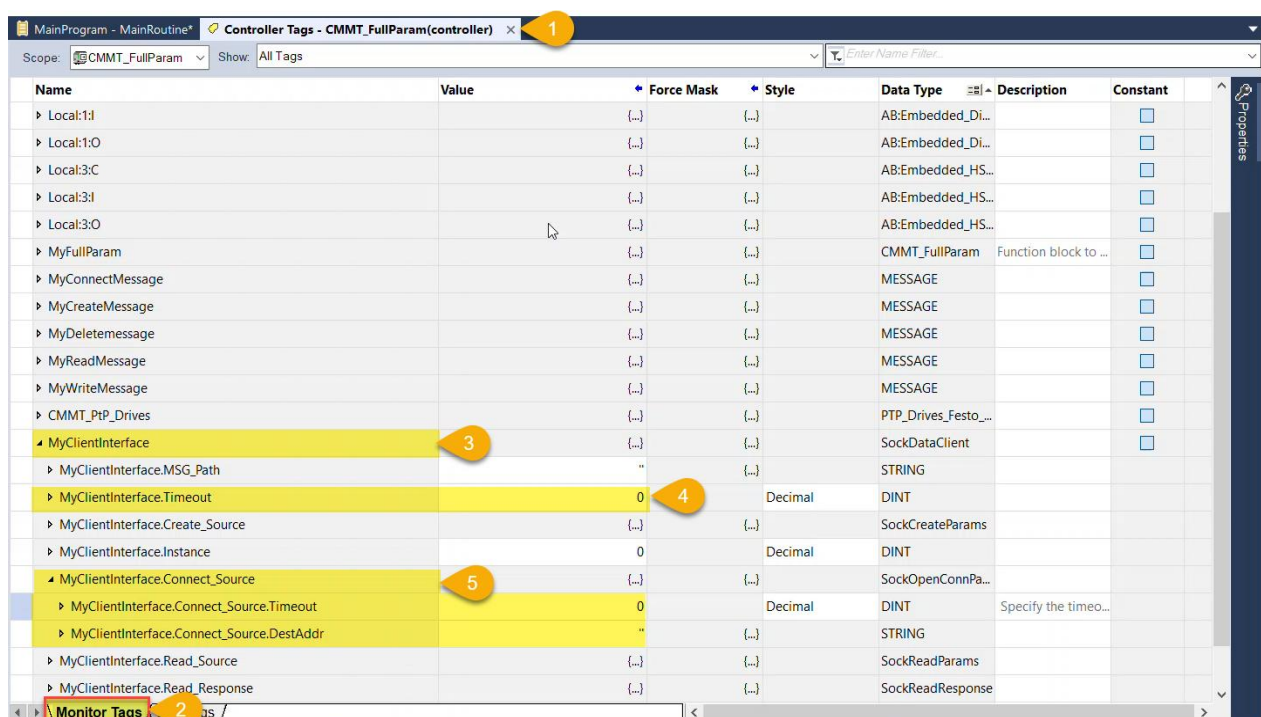
## Step – 2



1. Enter the Tag name for the “**MyClientInterface**” & right click.  
Select “**New**” option with tag name from right click menu.
2. Click “**Create**” button on new tag window & ensure Data Type is “**SockDataClient**” as highlighted in green color.

### 11.6.1 Step to add IP address of CMMT-xx in MyClientInterface:

## Step – 1



1. Open **Controller Tags** window.
2. Select **Monitor Tags**
3. Expand **MyClientInterface**
4. Initial value of **MyClientInterface.Timeout = 0**
5. Expand **MyClientInterface.Connect\_Source**  
Initial value of **MyClientInterface.Connect\_Source.Timeout = 0**  
Destination Address will not be defined in **MyClientInterface.Connect\_Source.DestAddr**

## Step – 2



### Note

- Syntax and Port no will remain same in all situations, only IP address needs to match with the connected CMMT Drive’s IP Address. ([Refer Chapter 2.5, Step – 4](#))
- Incase if user misses to enter correct IP address of connected Drive’s, the AOI will malfunction.

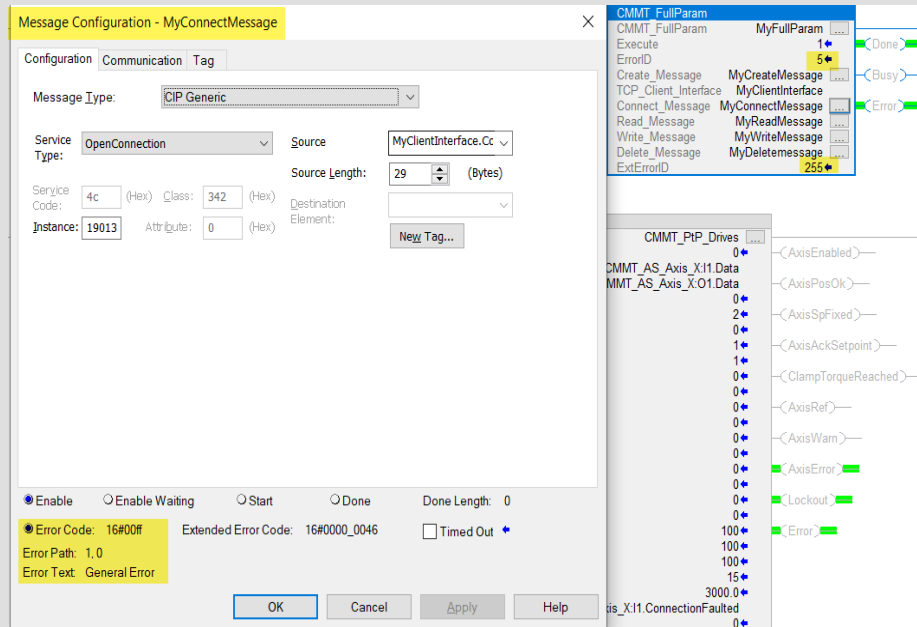
| Name                                      | Value | Force Mask | Style | Data Type            | Description           | Constant                 |
|-------------------------------------------|-------|------------|-------|----------------------|-----------------------|--------------------------|
| Local:1:I                                 |       | {...}      | {...} | AB:Embedded_Di...    |                       | <input type="checkbox"/> |
| Local:1:O                                 |       | {...}      | {...} | AB:Embedded_Di...    |                       | <input type="checkbox"/> |
| Local:3:C                                 |       | {...}      | {...} | AB:Embedded_HS...    |                       | <input type="checkbox"/> |
| Local:3:I                                 |       | {...}      | {...} | AB:Embedded_HS...    |                       | <input type="checkbox"/> |
| Local:3:O                                 |       | {...}      | {...} | AB:Embedded_HS...    |                       | <input type="checkbox"/> |
| MyFullParam                               |       | {...}      | {...} | CMMT_FullParam       | Function block to ... | <input type="checkbox"/> |
| MyConnectMessage                          |       | {...}      | {...} | MESSAGE              |                       | <input type="checkbox"/> |
| MyCreateMessage                           |       | {...}      | {...} | MESSAGE              |                       | <input type="checkbox"/> |
| MyDeletemessage                           |       | {...}      | {...} | MESSAGE              |                       | <input type="checkbox"/> |
| MyReadMessage                             |       | {...}      | {...} | MESSAGE              |                       | <input type="checkbox"/> |
| MyWriteMessage                            |       | {...}      | {...} | MESSAGE              |                       | <input type="checkbox"/> |
| CMMT_PtP_Drives                           |       | {...}      | {...} | PTP_Drives_Festo_... |                       | <input type="checkbox"/> |
| MyClientInterface                         |       | {...}      | {...} |                      |                       | <input type="checkbox"/> |
| MyClientInterface.MSG_Path                |       |            |       |                      |                       |                          |
| MyClientInterface.Timeout                 | 2000  |            |       |                      |                       |                          |
| MyClientInterface.Create_Source           | {...} |            |       |                      |                       |                          |
| MyClientInterface.Instance                | 0     |            |       |                      |                       |                          |
| MyClientInterface.Connect_Source          | {...} |            |       |                      |                       |                          |
| MyClientInterface.Connect_Source.Timeout  | 2000  |            |       |                      |                       |                          |
| MyClientInterface.Connect_Source.DestAddr |       |            |       |                      |                       |                          |
| MyClientInterface.Read_Source             | {...} |            |       |                      |                       |                          |
| MyClientInterface.Read_Response           | {...} |            |       |                      |                       |                          |

1. Expand **MyClientInterface**
2. Enter Timeout value **2000** in **MyClientInterface.Timeout**
3. Expand **MyClientInterface.Connect\_Source**
4. Enter Timeout value **2000** in **MyClientInterface.Connect\_Source.Timeout**
5. Select **MyClientInterface.Connect\_Source.DestAddr** and click on the three dot to modify the IP address.
6. Enter the same syntax and port no in all situations expect IP address of connected CMMT Drive’s as shown in above image, Example: **192.168.1.5?port=7507**  
(IP address needs to match with the connected CMMT Drive’s IP Address [Refer Chapter 2.5, Step – 4](#))
7. Click **“Apply”** button to confirm the configuration.
8. Click **“OK”** button to confirm.



### Warning

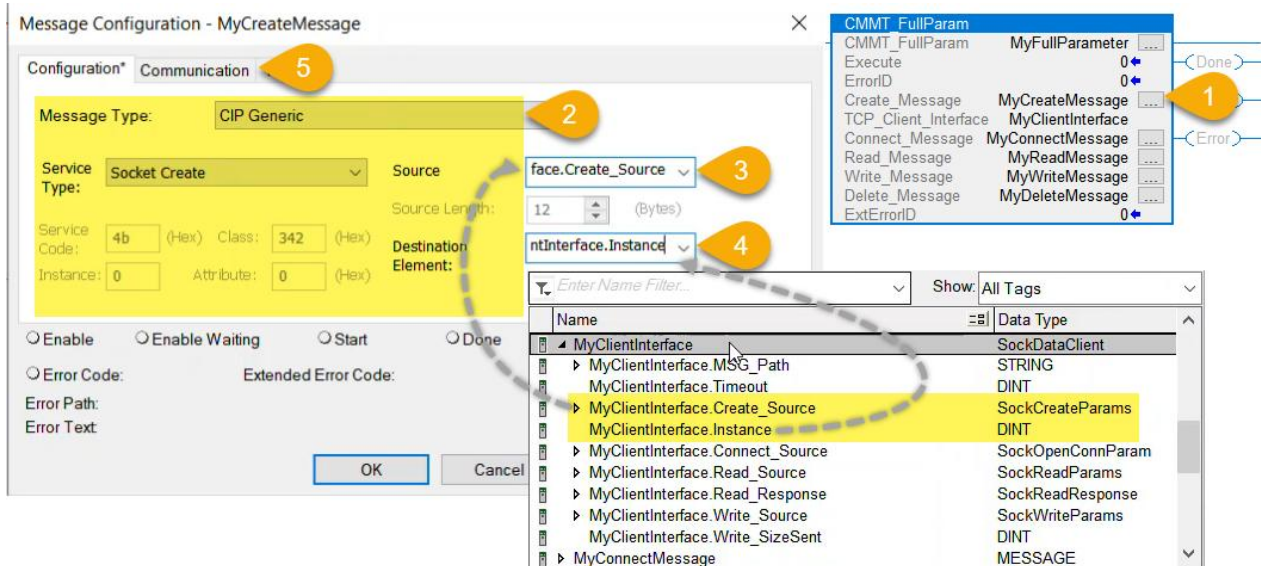
- If Incorrect IP address is assigned in CMMT\_FullParam AOI, then AOI will show error as shown below.



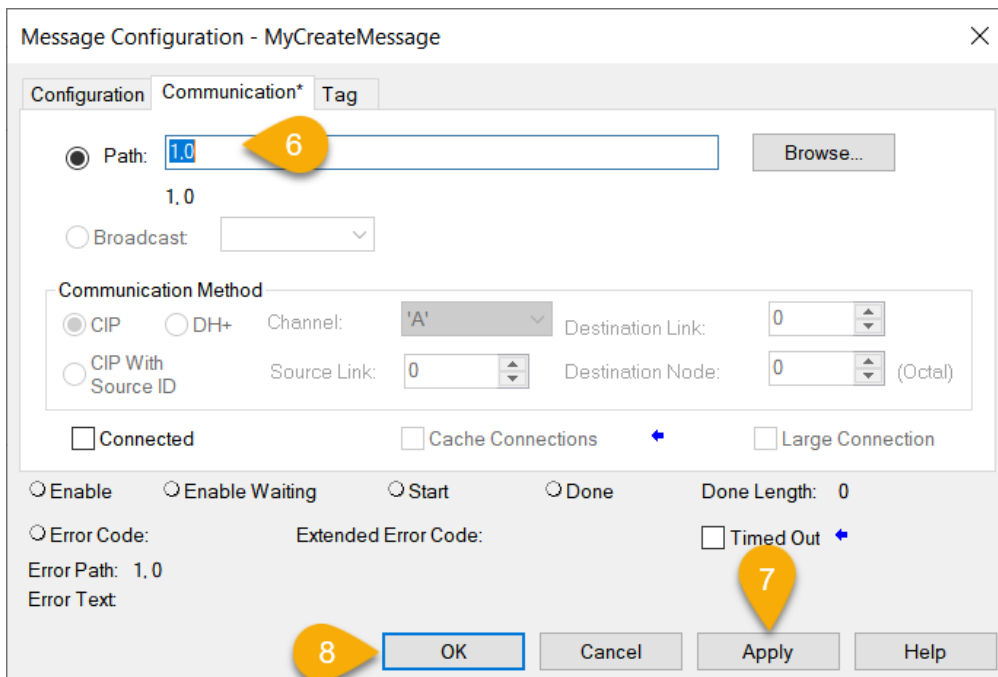
## 11.7 CMMT\_FullParam MSG Instruction Configuration

This chapter explain about message instruction configuration.

### 11.7.1 Create\_Message Instruction Configuration

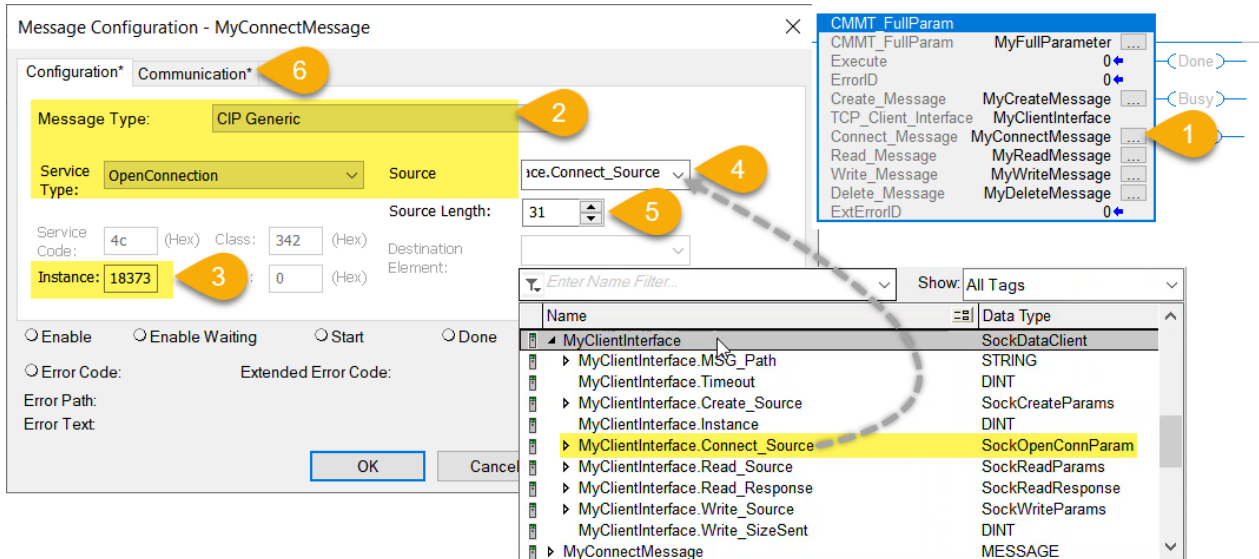


1. Click browse button of “**MyCreateMessage**” interface tag.
2. Set the yellow highlighted parameters as it is in above picture.
3. Click drop-down button of “**Source**” to browse destination tag.  
Browse the tag “**MyClientInterface.Create\_Source**” from tag list & select it as Source tag.
4. Click drop-down button of “**Destination Element**” to browse destination tag.  
Browse the tag “**MyClientInterface.Instance**” from tag list & select it as Destination tag.
5. Click communication tab to assign destination device.



6. Enter “**1, slot number**” in path from communication tab. (Example: 1, 0 (Refer [Chapter 11.7.6 Communication Path](#)))
7. Click “**Apply**” button to confirm the configuration.
8. Click “**OK**” button to confirm & close the MSG instruction configuration.

### 11.7.2 Connect\_Message Instruction Configuration



1. Click browse button of “**MyConnectMessage**” interface tag.
2. Set the yellow highlighted parameters as it is in above picture.
3. Enter random number in “**Instance**”. (Example: 18373)
4. Click drop-down button of “**Source**” to browse destination tag.  
Browse the tag “**MyClientInterface.Connect\_Source**” from tag list & select it as Source tag.
5. Enter “**31**” in Source Length.
6. Click communication tab to assign destination device.

7. Enter “**1, slot number**” in path from communication tab. (Example: 1, 0 (Refer [Chapter 11.7.6 Communication Path](#)))
8. Click “**Apply**” button to confirm the configuration.
9. Click “**OK**” button to confirm & close the MSG instruction configuration.

### 11.7.3 Read\_Message Instruction Configuration

1. Click browse button of “**MyReadMessage**” interface tag.
2. Set the yellow highlighted parameters as it is in above picture.
3. Enter random number in “**Instance**”. (Example: 18373)
4. Click drop-down button of “**Source**” to browse destination tag.  
Browse the tag “**MyClientInterface.Read\_Source**” from tag list & select it as Source tag.

- Click drop-down button of “**Destination Element**” to browse destination tag.  
Browse the tag “**MyClientInterface.Read\_Response**” from tag list & select it as Destination tag.
- Click communication tab to assign destination device.

Message Configuration - MyReadMessage

Configuration\* Communication\* Tag

☒ Path: 1.0 Browse...

☐ Broadcast

Communication Method

☒ CIP ☐ DH+ Channel: 'A' Destination Link: 0

☐ CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)

☐ Connected ☐ Cache Connections ☐ Large Connection

☐ Enable ☐ Enable Waiting ☐ Start ☐ Done Done Length: 0

☐ Error Code: Extended Error Code: ☐ Timed Out

Error Path: Error Text:

OK Cancel Apply Help

- Enter “**1, slot number**” in path from communication tab. (Example: 1, 0 (Refer [Chapter 11.7.6 Communication Path](#)))
- Click “**Apply**” button to confirm the configuration.
- Click “**OK**” button to confirm & close the MSG instruction configuration.

#### 11.7.4 Write\_Message Instruction Configuration

Message Configuration - MyWriteMessage

Configuration\* Communication\* 6

Message Type: CIP Generic 2

Service Type: WriteSocket Source: MyClientInterface.Write\_Source 3

Service Code: 4e (Hex) Class: 342 (Hex) Source Length: 46 4

Instance: 18373 Attribute: 0 (Hex) Destination Element: MyClientInterface.Write\_SizeSent 5

☐ Enable ☐ Enable Waiting ☐ Start ☐ Done

☐ Error Code: Extended Error Code:

Error Path: Error Text:

OK Cancel

CMMT\_FullParam MyFullParameter

|                      |                   |     |
|----------------------|-------------------|-----|
| CMMT_FullParam       | MyFullParameter   | ... |
| Execute              | 0                 | ... |
| ErrorID              | 0                 | ... |
| Create_Message       | MyCreateMessage   | ... |
| TCP_Client_Interface | MyClientInterface | ... |
| Connect_Message      | MyConnectMessage  | ... |
| Read_Message         | MyReadMessage     | ... |
| Write_Message        | MyWriteMessage    | ... |
| Delete_Message       | MyDeleteMessage   | ... |
| ExtErrorID           | 0                 | ... |

Enter Name Filter... Show: All Tags

| Name                             | Data Type           |
|----------------------------------|---------------------|
| MyClientInterface                | SocketDataClient    |
| MyClientInterface.MSG_Path       | STRING              |
| MyClientInterface.Timeout        | DINT                |
| MyClientInterface.Create_Source  | SocketCreateParams  |
| MyClientInterface.Instance       | DINT                |
| MyClientInterface.Connect_Source | SocketOpenConnParam |
| MyClientInterface.Read_Source    | SocketReadParams    |
| MyClientInterface.Read_Response  | SocketReadResponse  |
| MyClientInterface.Write_Source   | SocketWriteParams   |
| MyClientInterface.Write_SizeSent | DINT                |
| MyConnectMessage                 | MESSAGE             |

- Click browse button of “**MyWriteMessage**” interface tag.
- Set the yellow highlighted parameters as it is in above picture.

3. Click drop-down button of “**Source**” to browse destination tag.  
Browse the tag “**MyClientInterface.Write\_Source**” from tag list & select it as Source tag.
4. Enter “**46**” in Source Length.
5. Click drop-down button of “**Destination Element**” to browse destination tag.  
Browse the tag “**MyClientInterface.Write\_SizeSent**” from tag list & select it as Destination tag.
6. Click communication tab to assign destination device.

Message Configuration - MyWriteMessage

Configuration\* Communication\* Tag

☒ Path: 1.0 Browse...

☐ Broadcast: 1.0

Communication Method

☒ CIP ☐ DH+ Channel: A Destination Link: 0

☐ CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)

☐ Connected ☐ Cache Connections ☐ Large Connection

☐ Enable ☐ Enable Waiting ☐ Start ☐ Done Done Length: 0

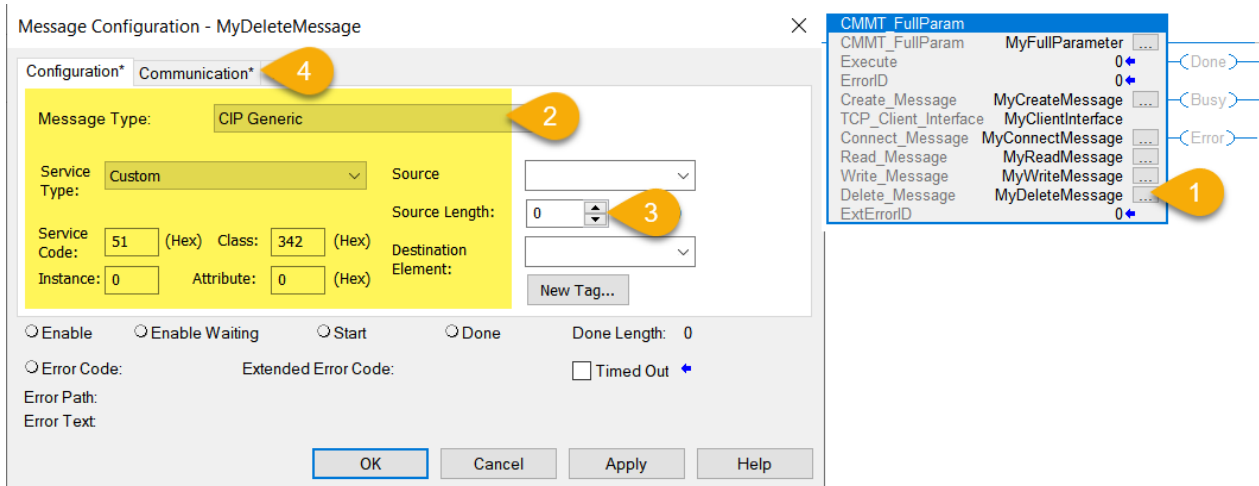
☐ Error Code: Extended Error Code: ☐ Timed Out

Error Path:  
Error Text:

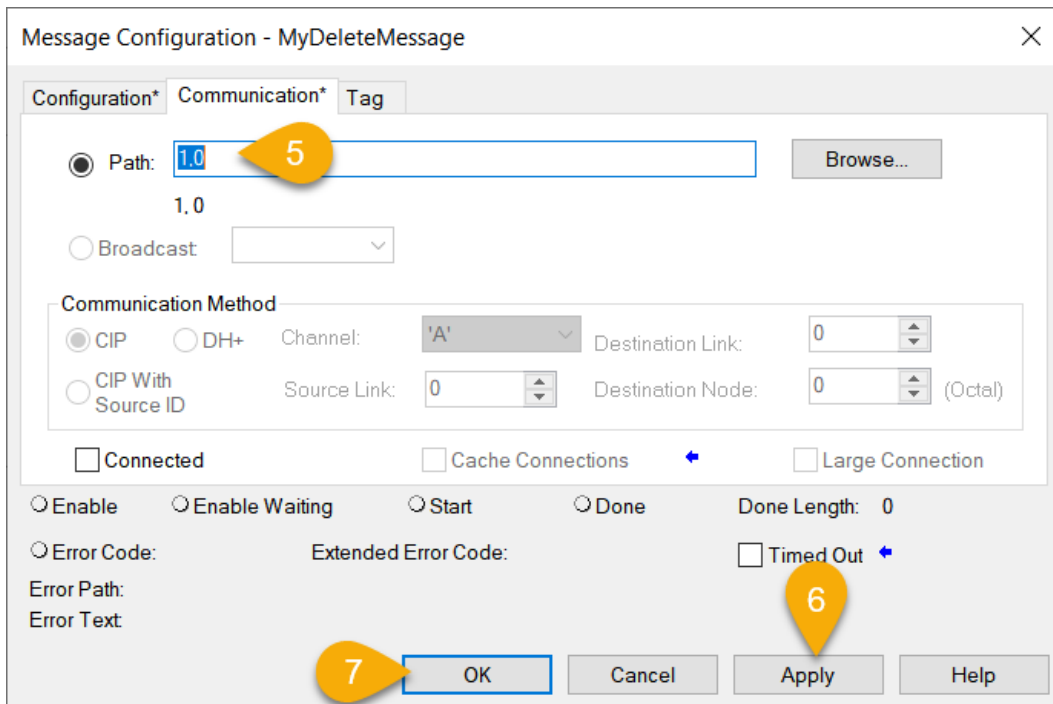
OK Cancel Apply Help

7. Enter “**1, slot number**” in path from communication tab. (Example: 1, 0 (Refer [Chapter 11.7.6 Communication Path](#)))
8. Click “**Apply**” button to confirm the configuration.
9. Click “**OK**” button to confirm & close the MSG instruction configuration.

### 11.7.5 Delete\_Message Instruction Configuration



1. Click browse button of “**MyDeleteMessage**” interface tag.
2. Set the yellow highlighted parameters as it is in above picture.
3. Enter “0” in Source Length.
4. Click communication tab to assign destination device.



5. Enter “1, slot number” in path from communication tab. (Example: 1, 0 (Refer [Chapter 11.7.6 Communication Path](#)))
6. Click “**Apply**” button to confirm the configuration.
7. Click “**OK**” button to confirm & close the MSG instruction configuration.

### 11.7.6 Communication Path



#### Note

There are hardware limitations with certain Allen Bradley controllers or EtherNet/IP masters. The AOI (Add-On Instruction) requires Socket Services which are ONLY available/supported with these Controllers/Modules listed here:

- ControlLogix® 5580 and GuardLogix® 5580 controllers
- CompactLogix™ 5380 and Compact GuardLogix 5380 controllers
- CompactLogix 5370 and Compact GuardLogix 5370 controllers
- CompactLogix 5480 controllers
- 1756-EN2x, 1756-EN3TR, and 1756-EN4TR EtherNet/IP communication modules
- 1756-EWEB and 1768-EWEB EtherNet/IP web server modules

#### CompactLogix controllers

For CompactLogix controllers, the path is 1 and the slot is 0.

Message Configuration - MyCreateMessage

Configuration\* Communication\* Tag

☒ Path: 1.0

☐ Broadcast:

Communication Method

☒ CIP ☐ DH+ Channel: 'A' Destination Link: 0

☐ CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)

☐ Connected ☐ Cache Connections ☐ Large Connection

☐ Enable ☐ Enable Waiting ☐ Start ☐ Done Done Length: 0

☐ Error Code: Extended Error Code: ☐ Timed Out

Error Path:  
Error Text:

OR

For CompactLogix controllers with version 28.11 or later, you can also use a path of 'THIS'.

Message Configuration - MyCreateMessage

Configuration Communication Tag

☒ Path: THIS

☐ Broadcast:

Communication Method

☒ CIP ☐ DH+ Channel: 'A' Destination Link: 0

☐ CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)

☐ Connected ☐ Cache Connections ☐ Large Connection

☐ Enable ☐ Enable Waiting ☐ Start ☐ Done Done Length: 0

☐ Error Code: Extended Error Code: ☐ Timed Out

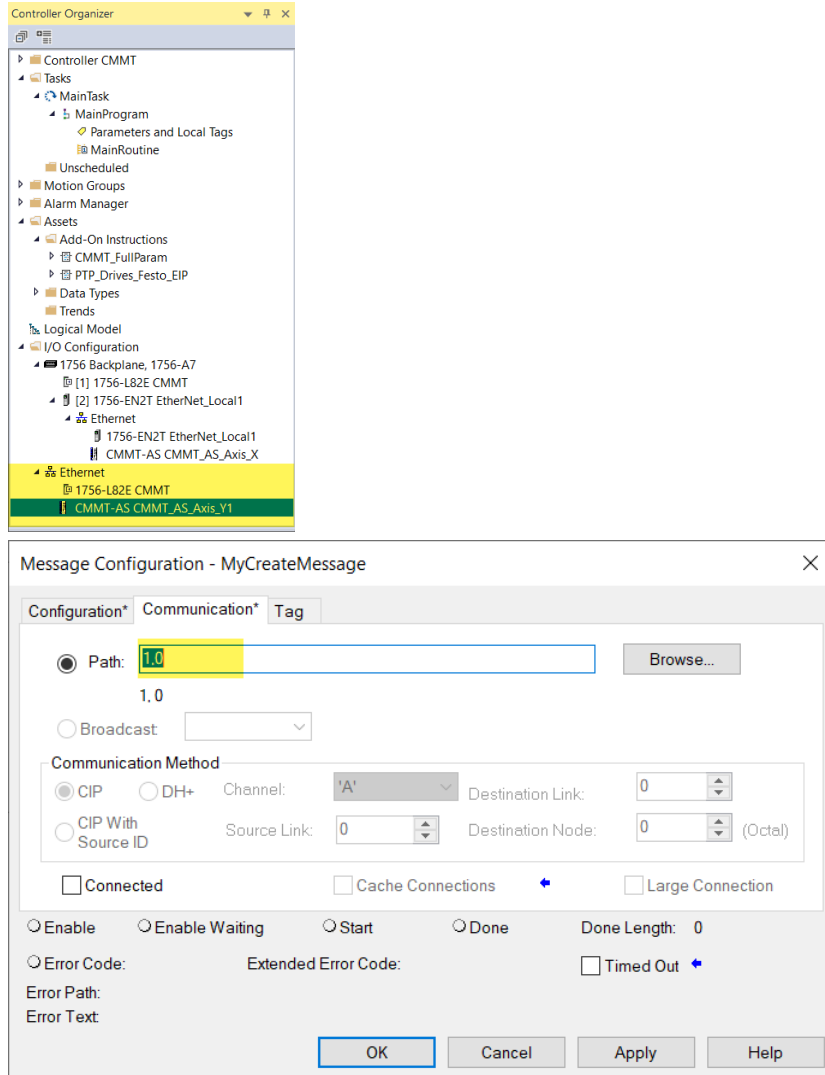
Error Path: THIS  
Error Text:

## ControlLogix controllers

For ControlLogix 5580 controllers and communication modules, the path is 1, x. Where x is the slot number of the communication module, or the slot number of a ControlLogix 5580 controller if the front Ethernet port is used.

### Method – 1

If CMMT module added under Controller Ethernet Port then, follow Method – 1



OR

For ControlLogix 5580 controllers with version 28.11 or later, you can also use a path of 'THIS'.

## “CMMT Full Parameter AOI”/ Full Parameter of CMMT Drive

Message Configuration - MyCreateMessage

Configuration Communication Tag

Path: THIS

Broadcast: THIS

Communication Method

CIP DH+ Channel: A Destination Link: 0

CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)

Connected Cache Connections Large Connection

Enable Enable Waiting Start Done Done Length: 0

Error Code: Extended Error Code: Timed Out

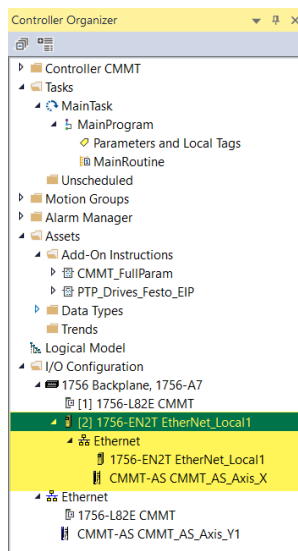
Error Path: THIS

Error Text:

OK Cancel Apply Help

### Method – 2

If CMMT module added under Ethernet Module then, follow Method – 2



The path is 1, the slot number of the communication module is 2

Message Configuration - MyCreateMessage

Configuration Communication\* Tag

Path: 1:2

Broadcast: EtherNet\_Local1

Communication Method

CIP DH+ Channel: A Destination Link: 0

CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)

Connected Cache Connections Large Connection

Enable Enable Waiting Start Done Done Length: 0

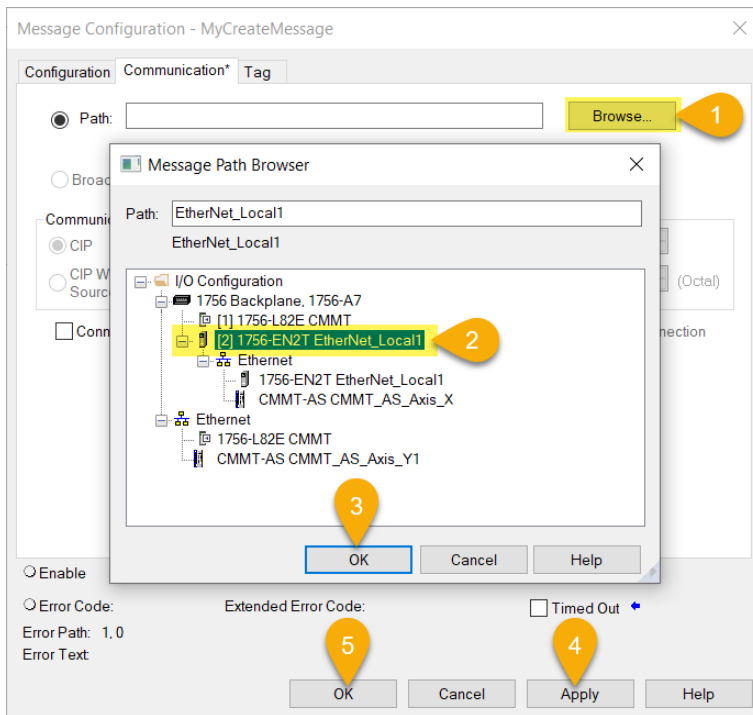
Error Code: Extended Error Code: Timed Out

Error Path: EtherNet\_Local1

Error Text:

OK Cancel Apply Help

Or



1. Click on Browse... button
2. Select the Ethernet Module (1756-EN2T)
3. Click “OK” button to confirm
4. Click “Apply” button to confirm the configuration.
5. Click “OK” button to confirm.

### 11.7.7 CMMT\_FullParam AOI Tag Details

#### Control Area Tag Details

| Control Area Tags | Data Type | Function Description                               |
|-------------------|-----------|----------------------------------------------------|
| Execute           | BOOL      | Execute the task ( 0 – Unexecute;<br>1 – Execute ) |

#### Monitor Area Tag Details

| Area Tags  | Data Type | Function Description                                  |
|------------|-----------|-------------------------------------------------------|
| ErrorID    | INT       | Error message description                             |
| ExtErrorID | DINT      | Error message from Message Tags                       |
| Done       | BOOL      | Parameterization processed successfully               |
| Busy       | BOOL      | Parameterization function under process               |
| Error      | BOOL      | Parameterization function completed with unsuccessful |

#### MSG Interface Tag Details

Application Note – CMMT servo drive commissioning with Allen-Bradley PLC through EtherNet/IP network – 1.60

| Area Tags       | Data Type | Function Description             |
|-----------------|-----------|----------------------------------|
| Create_Message  | MESSAGE   | Socket Create MSG instruction    |
| Connect_Message | MESSAGE   | Open Connection MSG instruction  |
| Read_Message    | MESSAGE   | Parameter Read MSG instruction   |
| Write_Message   | MESSAGE   | Parameter Write MSG instruction  |
| Delete_Messafe  | MESSAGE   | Parameter Delete MSG instruction |

## 11.8 Diagnostics

The following table shows the possible error number and the associated error description

### 11.8.1 ErrorID

The “ErrorID” serves for Internal Error ID of the CMMT\_FullParam FB.

| ErrorID<br>(Dec) | ErrorID<br>(Hex) | ErrorID Description                      |
|------------------|------------------|------------------------------------------|
| 3                | 0x3              | Error in Delete_Message                  |
| 4                | 0x4              | Error in Create_Message                  |
| 5                | 0x5              | Error in Connect_Message                 |
| 6                | 0x6              | size Login is 0                          |
| 7                | 0x7              | ACK not OK                               |
| 8                | 0x8              | size_openfile is 0                       |
| 9                | 0x9              | Error in Read_Message                    |
| 10               | 0xA              | frame counter out of range               |
| 11               | 0xB              | IndexPck out of range                    |
| 12               | 0xC              | size writefile is 0                      |
| 13               | 0xD              | Error in Write_Message                   |
| 14               | 0xE              | something went wrong with the PCK states |
| 15               | 0xF              | received data size invalid               |
| 16               | 0x10             | size_closefile is 0                      |
| 17               | 0x11             | size_unpackfile is 0                     |

### 11.8.2 ExtErrorID

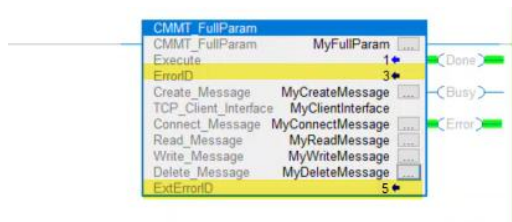
The “ExtErrorID” serves for External Error ID of the Message interface used in CMMT\_FullParam FB.

| Error Code | Error Text: Description                                                                                              | Error Code | Error Text: Description                               |
|------------|----------------------------------------------------------------------------------------------------------------------|------------|-------------------------------------------------------|
| 1          | Connection failure (extended error codes)                                                                            | 21         | Too much data                                         |
| 2          | Insufficient resource                                                                                                | 26         | Bridge request too large                              |
| 3          | Invalid value                                                                                                        | 27         | Bridge response too large                             |
| 4          | IOI syntax error (see extended error codes)                                                                          | 28         | Attribute list shortage                               |
| 5          | Destination unknown, class unsupported, instance undefined or structure element undefined (see extended error codes) | 29         | Invalid attribute list                                |
| 6          | Insufficient packet space                                                                                            | 30         | Embedded service error                                |
| 7          | Connection lost                                                                                                      | 31         | Connection related failure (see extended error codes) |
| 8          | Service unsupported                                                                                                  | 32         | Invalid Parameter                                     |
| 9          | Error in data segment or invalid attribute value                                                                     | 34         | Invalid reply received                                |
| 10         | Attribute list error                                                                                                 | 37         | Key segment error                                     |
| 11         | State already exists                                                                                                 | 38         | Invalid IOI error                                     |
| 12         | Object model conflict                                                                                                | 39         | Unexpected attribute in list                          |
| 13         | Object already exists                                                                                                | 40         | DeviceNet error - invalid member ID                   |
| 14         | Attribute cannot be set                                                                                              | 41         | DeviceNet error - member not settable                 |
| 15         | Permission denied                                                                                                    | 209        | Module not in run state                               |
| 16         | Device state conflict                                                                                                | 251        | Message port not supported                            |
| 17         | Reply will not fit                                                                                                   | 252        | Message unsupported data type                         |
| 18         | Fragment primitive                                                                                                   | 253        | Message uninitialized                                 |

|    |                           |     |                                           |
|----|---------------------------|-----|-------------------------------------------|
| 19 | Insufficient command data | 254 | Message timeout                           |
| 20 | Attribute not supported   | 255 | General error (see extended error codes)z |

### 11.8.3 Example

Refer [Chapter 11.5.1](#) to check ErrorID Description and also Refer [Chapter 11.5.2](#) to check ExtErrorID Description

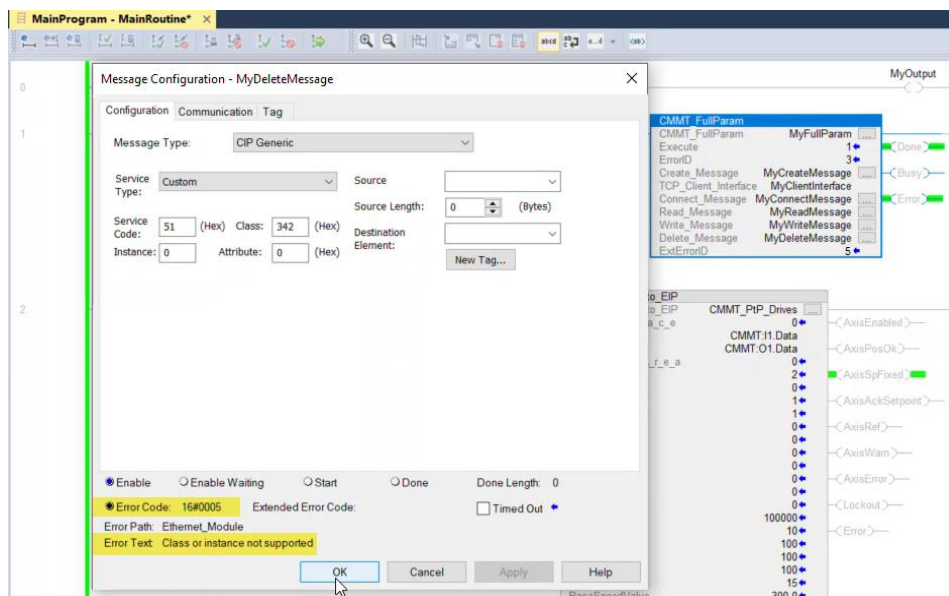


ErrorID

|   |                         |
|---|-------------------------|
| 3 | Error in Delete_Message |
|---|-------------------------|

ExtErrorID

|   |                                                                                                                      |
|---|----------------------------------------------------------------------------------------------------------------------|
| 5 | Destination unknown, class unsupported, instance undefined or structure element undefined (see extended error codes) |
|---|----------------------------------------------------------------------------------------------------------------------|

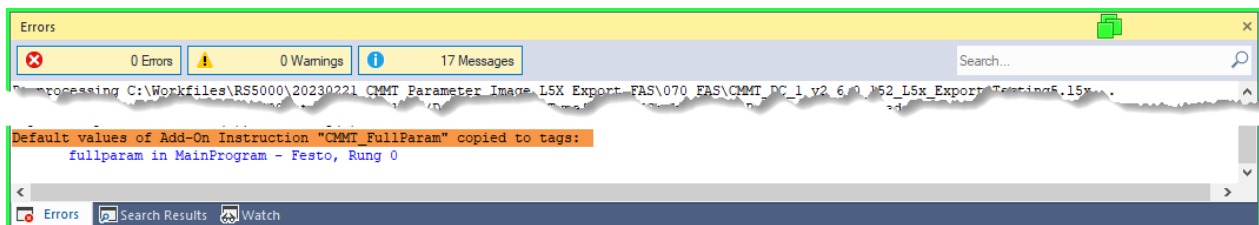
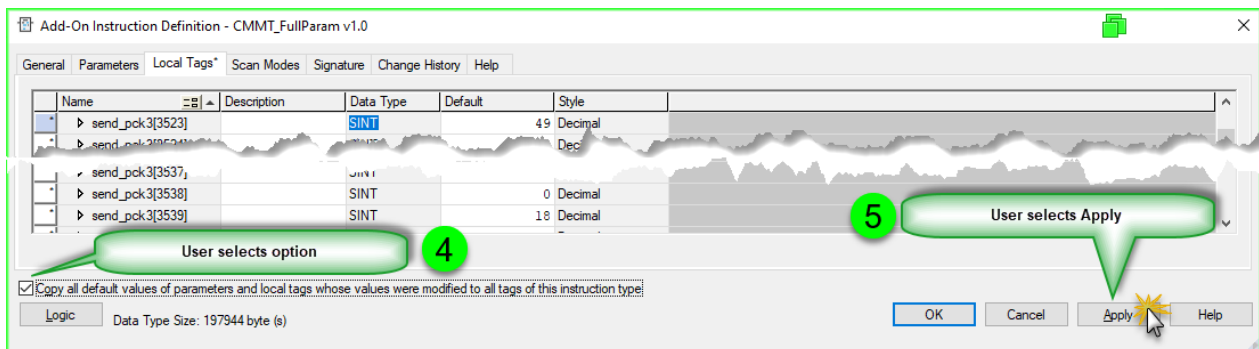
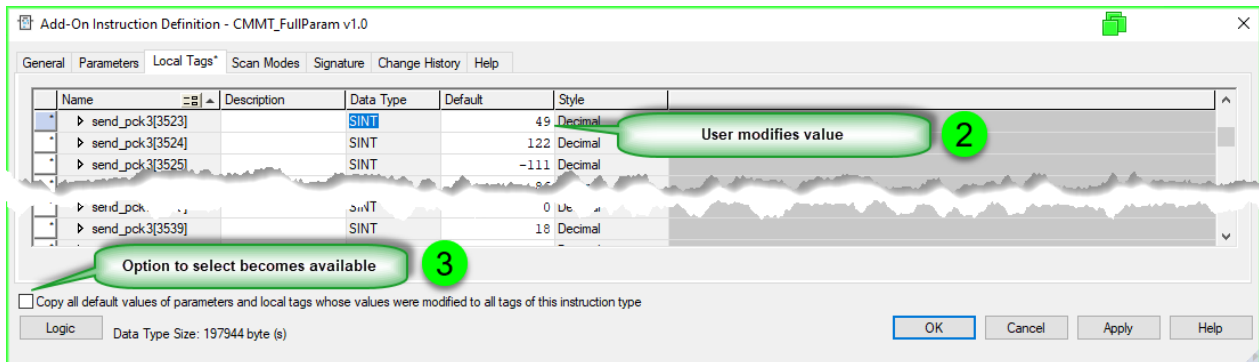
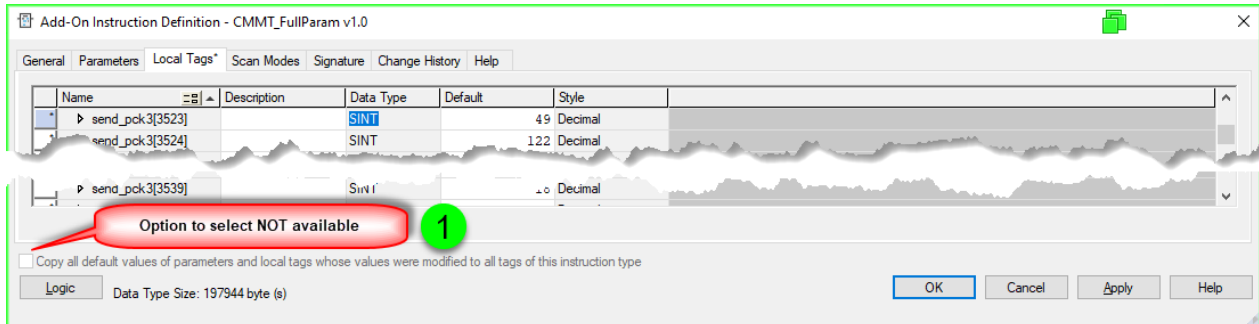


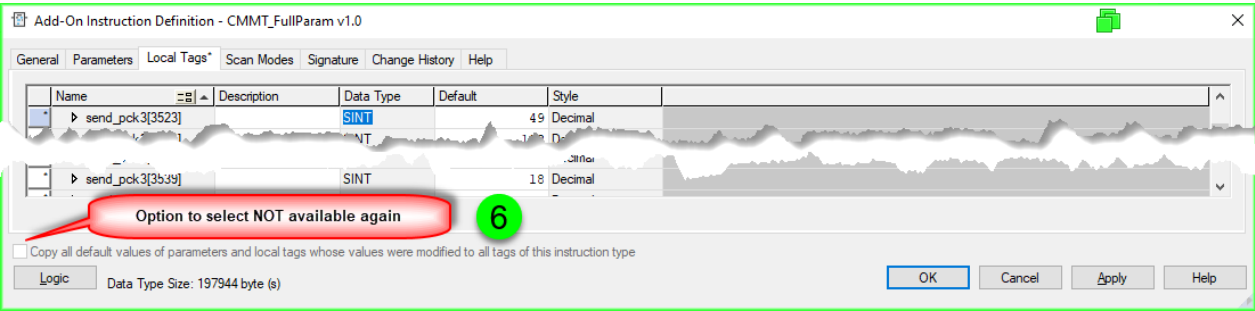
#### Note :

- If following error occurs “class or instance not supported” then update the firmware revision of the AB module to the latest version.

## 11.9 Copy all default values of parameters and local tags whose values were modified to all tags of the instruction type

When modifying values within the "Definition Editor" of an Add-On Instruction, if you want to update existing invocations of the instruction to the new default values, the user will select the [Copy all default values of parameters and local tags whose values were modified to all tags of this instruction type](#) prior to clicking Ok or Apply. An example is shown here:





11.10 Application Example

Here in this example we are going to parameterize “CMMT-xx”. Below are the simple steps to configure.

Step – 1

- Ensure **IP address** of CMMT-xx is assigned in **“MyClientInterface.Connect\_Source.DestAddr”**.

➔

**Note**

- Cross check the IP address entered in “MyClientInterface.Connect\_Source.DestAddr” is matching with the connected CMMT Drive’s IP Address, if IP Address is mismatching then enter the correct IP address of connected Drive

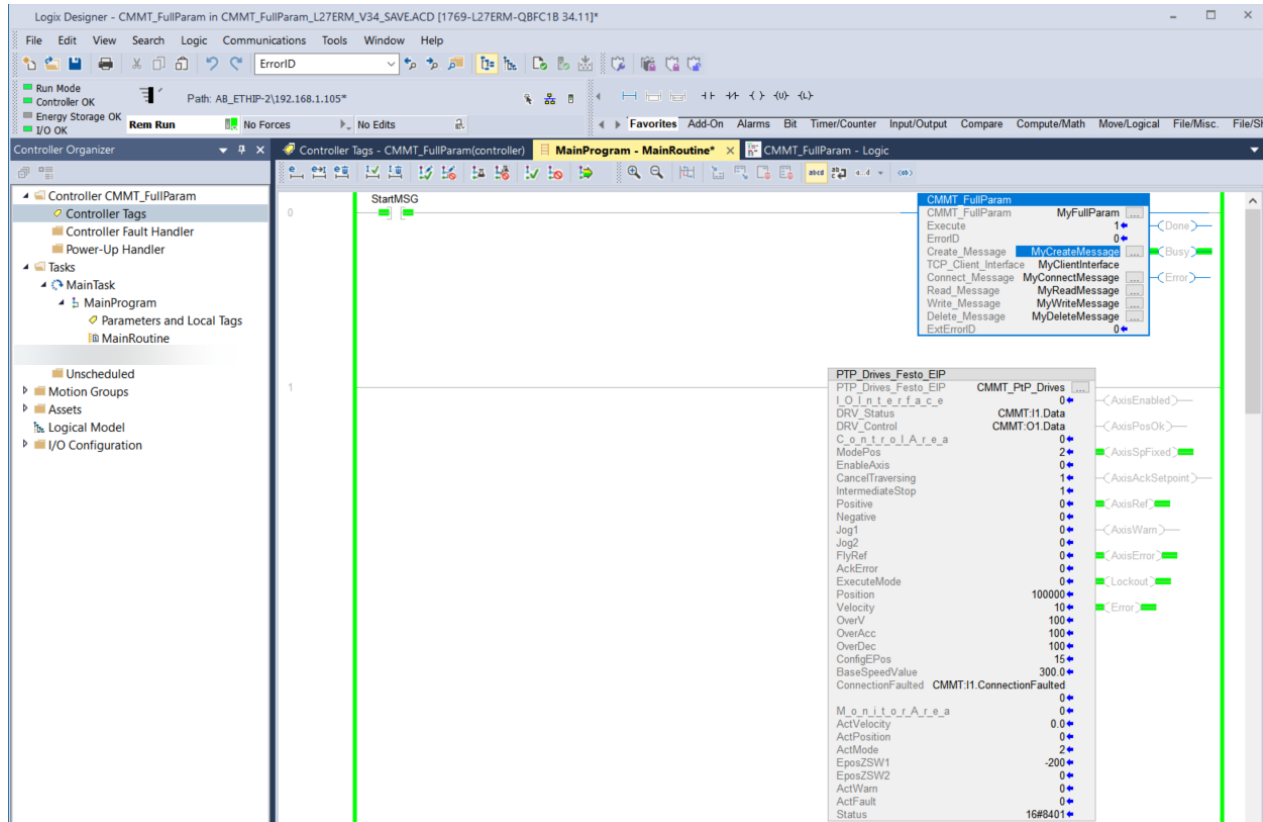
Controller Tags - CMMT\_FullParam(controller) x MainProgram - MainRoutine CMMT\_FullParam - Logic

Scope: CMMT\_FullParam Show: All Tags

| Name                                      | Value                   | Force Mask | Style   | Data Type         |
|-------------------------------------------|-------------------------|------------|---------|-------------------|
| MyClientInterface                         |                         |            |         | SockDataClient    |
| MyClientInterface.MSG_Path                | '\$01\$00'              |            |         | STRING            |
| MyClientInterface.Timeout                 | 2000                    |            | Decimal | DINT              |
| MyClientInterface.Create_Source           |                         |            |         | SockCreateParams  |
| MyClientInterface.Instance                | 29120                   |            | Decimal | DINT              |
| MyClientInterface.Connect_Source          |                         |            |         | SockOpenConnPa... |
| MyClientInterface.Connect_Source.Timeout  | 2000                    |            | Decimal | DINT              |
| MyClientInterface.Connect_Source.DestAddr | '192.168.1.5?port=7507' |            |         | STRING            |
| MyClientInterface.Read_Source             |                         |            |         | SockReadParams    |
| MyClientInterface.Read_Response           |                         |            |         | SockReadResponse  |

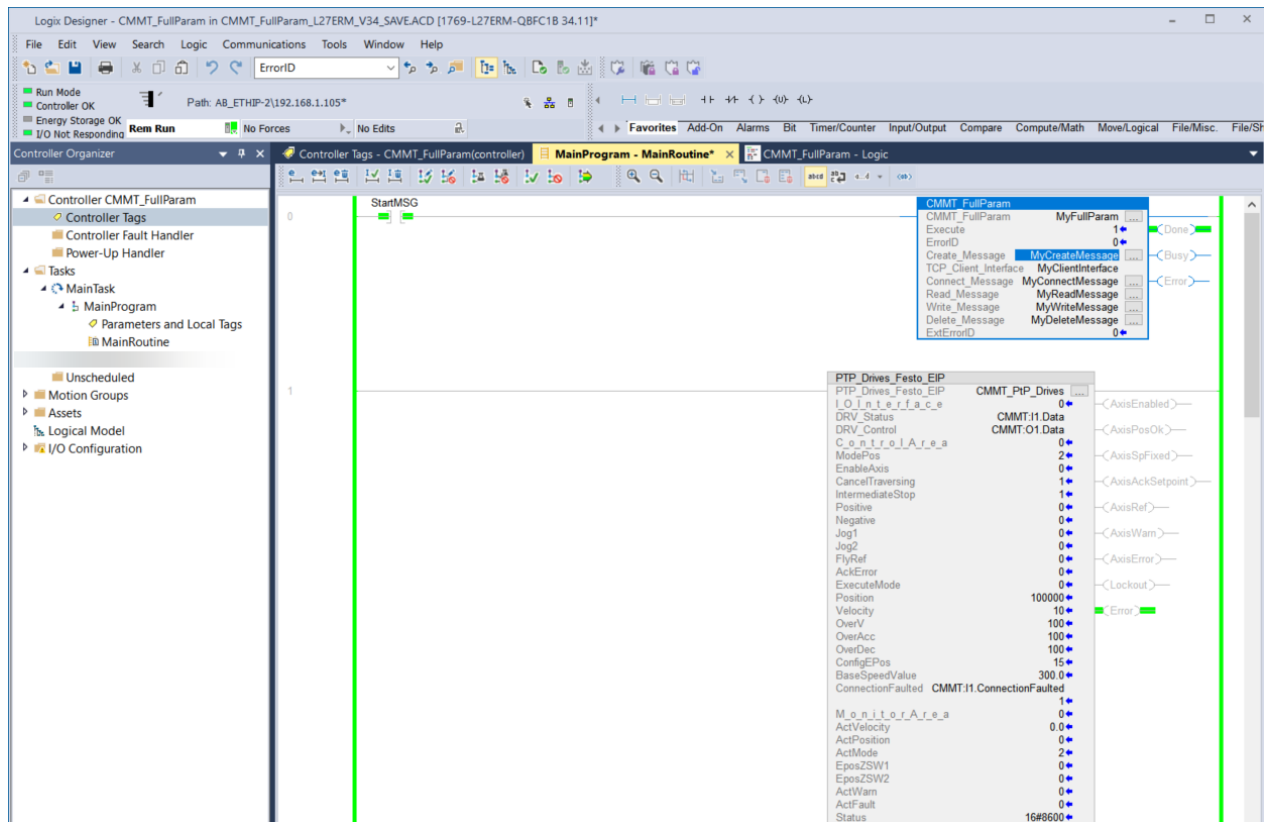
## Step – 2

Toggle the **“Execute”** bit to trigger the parameterization. While the parameterization is active **“Busy”** status will be true. If the entire parameterization has completed successfully, the **“Done”** bit will be high or If any error then **“Error”** bit will be high and error description will be on **“ErrorID”** tag.



After successful execution, CMMT-xx will boot-up automatically.

“CMMT Full Parameter AOI”/ Full Parameter of CMMT Drive



## 12 “Festo\_RecordTable\_EIP” AOI

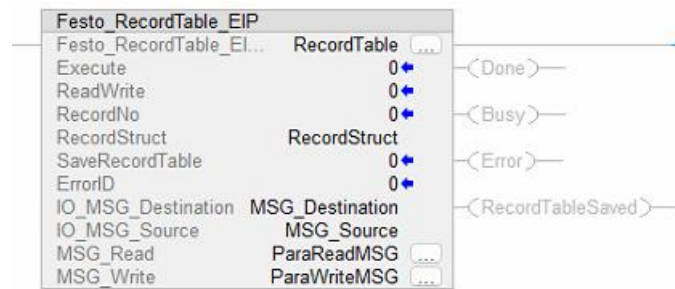


### Note

- This Add-On Instruction (AOI) has been designed to support 64bit Rockwell controllers for read and write the CMMT LINT data. All Input and Outputs data are configured as SINT[8] data type, so From Studio V33 will be supported for the Rockwell 64bit Controllers.

### 12.1 Festo\_RecordTable\_EIP AOI Introduction

This chapter describes about reading and writing from/to internal record table in FESTO CMMT drive from Allen-Bradley PLC through Ethernet/IP network. Also detail explanation is given for RecordStruct and message instruction configuration in Studio-5000.



#### 12.1.1 IO Interface Area Tag Details

| Input signal       | Type    | Meaning                                                              |
|--------------------|---------|----------------------------------------------------------------------|
| RecordStruct       | UDT     | Struct of the record (Festo_MM_X) Refer <a href="#">Chapter 12.2</a> |
| IO_MSG_Destination | SINT[8] | Read Data is stored in Destination Element                           |
| IO_MSG_Source      | SINT[8] | Write Data to be stored in Source Element                            |

#### 12.1.2 MSG Interface Tag Details

| Area Tags | Data Type | Function Description |
|-----------|-----------|----------------------|
| MSG_Read  | Message   | Message Instruction  |
| MSG_Write | Message   | Message Instruction  |

#### 12.1.3 Control Area Tag Details

| Input signal | Type | Meaning                                      |
|--------------|------|----------------------------------------------|
| Execute      | BOOL | 1 = Start parameter read or write operation. |
| ReadWrite    | BOOL | 1 = Write, 0 = Read                          |

| Input signal    | Type  | Meaning                                                                                                                                                  |
|-----------------|-------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| RecordNo        | USINT | Record number in the motor controller                                                                                                                    |
| SaveRecordTable | BOOL  | 1 = Starts after Write Operation, record table will be saved in Drive.<br>0 = Save RecordTable operation will not be performed/ won't be saved in Drive. |

#### 12.1.4 Monitor Area Tag Details

| Output signal    | Type | Meaning                                                                                                            |
|------------------|------|--------------------------------------------------------------------------------------------------------------------|
| Done             | BOOL | Completed successfully.                                                                                            |
| Busy             | BOOL | read or write is under progress.                                                                                   |
| Error            | BOOL | 1 = Error occurred while process. Refer <a href="#">Chapter 12.1.5</a> ; also refer <a href="#">chapter 11.8.2</a> |
| ErrorID          | DINT | Error code generated during process.                                                                               |
| RecordTableSaved | BOOL | Record Table Save successful.                                                                                      |

#### 12.1.5 Diagnostic Messages / Function Block Faults

The function block gives multiple types of diagnostic messages and also describes how to handle & solve them.

| Error Code | Description                       | Remedy                                                 |
|------------|-----------------------------------|--------------------------------------------------------|
| 48         | Record table index out of borders | Check the input RecordNo should be less than value 127 |
| 49         | Unknown record read out           | Check the input RecordNo                               |
| 50         | Unknown record write out          | Check the input Record Table Type                      |

## 12.2 Festo\_MM\_RecordStruct

| Name                   | Data Type | Description                      |
|------------------------|-----------|----------------------------------|
| Record_Type            | USINT     | Record Type Number               |
| StopRamp               | UDT       | Festo_MM_StopRamp                |
| Position               | UDT       | Festo_MM_Position                |
| Velocity               | UDT       | Festo_MM_Velocity                |
| Homing                 | UDT       | Festo_MM_Homing                  |
| Torque                 | UDT       | Festo_MM_Torque                  |
| FlowControl            | UDT       | Festo_MM_FlowControl             |
| AnaloguePosition       | UDT       | Festo_MM_AnaloguePosition        |
| AnalogueVelocity       | UDT       | Festo_MM_AnalogueVelocity        |
| AnalogueTorque         | UDT       | Festo_MM_AnalogueTorque          |
| TorqueWithBrake        | UDT       | Festo_MM_TorqueWithBrake         |
| PositionTrigger        | UDT       | Festo_MM_PositionTrigger         |
| SetDigitalOuput        | UDT       | Festo_MM_SetDigitalOutput        |
| TouchProbe             | UDT       | Festo_MM_TouchProbe              |
| Modulo                 | UDT       | Festo_MM_Modulo                  |
| ChangeParameterSet     | UDT       | Festo_MM_ChangeParameterSet      |
| SetPositionStatusWord2 | UDT       | Festo_MM_SetPositionStatusword2  |
| MovingFixedStop        | UDT       | Festo_MM_MovingFixedStop         |
| MoveFixedStopParameter | UDT       | Festo_MM_MovedFixedStopParameter |
| BrakeTest              | UDT       | Festo_MM_BrakeTest               |
| GearInOut              | UDT       | Festo_MM_GearInOut               |
| SetMasterPosition      | UDT       | Festo_MM_SetmasterPositionGear   |
| TorquePercent          | UDT       | Festo_MM_TorquePercent           |

### Festo\_MM\_StopRamp

| Name         | Data Type | Default Value |
|--------------|-----------|---------------|
| RecordTypeNo | USINT     | 2             |
| Deceleration | REAL      |               |
| Jerk         | REAL      |               |

### Festo\_MM\_Homing

| Name         | Data Type | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|--------------|-----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| RecordTypeNo | USINT     | 3                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| HomingMethod | DINT      | -27: Negative stop/limit switch with run to reference switch<br>-23: Positive stop/limit switch with run to reference switch<br>-18: Positive stop<br>-17: Negative stop<br>-2: Positive stop with zero pulse<br>-1: Negative stop with zero pulse<br>1: Negative hardware limit switch with zero pulse<br>2: Positive hardware limit switch with zero pulse<br>7: Positive reference switch with zero pulse<br>11: Negative reference switch with zero pulse<br>17: Negative hardware limit switch<br>18: Positive hardware limit switch<br>23: Positive reference switch<br>27: Negative reference switch<br>33: Current position with zero pulse in negative direction<br>34: Current position with zero pulse in positive direction<br>37: Current position (default setting) |

### Festo\_MM\_Velocity

| Name                    | Data Type | Default Value |
|-------------------------|-----------|---------------|
| RecordTypeNo            | USINT     | 4             |
| Target_torque           | REAL      |               |
| Torque_rise_ramp        | REAL      |               |
| Limit_speed             | REAL      |               |
| Activation_stroke_limit | BOOL      |               |
| Stroke_limit_negative   | REAL      |               |
| Stroke_limit_positive   | REAL      |               |

### Festo\_MM\_Position

| Name         | Data Type | Default Value                                                                           |
|--------------|-----------|-----------------------------------------------------------------------------------------|
| RecordTypeNo | USINT     | 5                                                                                       |
| Type         | USINT     | 0: absolute positioning<br>1: relative actual position<br>2: relative setpoint position |

|                 |      |                                                                                                                                             |
|-----------------|------|---------------------------------------------------------------------------------------------------------------------------------------------|
|                 |      | 3: relative target position<br>4: relative captured position0<br>5: relative captured position1<br>6: relative actual position unreferenced |
| Target_position | REAL |                                                                                                                                             |
| Velocity        | REAL |                                                                                                                                             |
| Acceleration    | REAL |                                                                                                                                             |
| Deceleration    | REAL |                                                                                                                                             |
| Jerk            | REAL |                                                                                                                                             |
| End_velocity    | REAL |                                                                                                                                             |

#### **Festo\_MM\_Torque**

| Name                    | Data Type | Default Value |
|-------------------------|-----------|---------------|
| RecordTypeNo            | USINT     | 7             |
| Target_torque           | REAL      |               |
| Torque_rise_ramp        | REAL      |               |
| Limit_speed             | REAL      |               |
| Activation_stroke_limit | BOOL      |               |
| Stroke_limit_negative   | REAL      |               |
| Stroke_limit_positive   | REAL      |               |

#### **Festo\_MM\_FlowControl**

| Name         | Data Type | Default Value |
|--------------|-----------|---------------|
| RecordTypeNo | USINT     | 10            |

#### **Festo\_MM\_AnaloguePosition**

| Name         | Data Type | Default Value |
|--------------|-----------|---------------|
| RecordTypeNo | USINT     | 21            |

#### **Festo\_MM\_AnalogueVelocity**

| Name         | Data Type | Default Value |
|--------------|-----------|---------------|
| RecordTypeNo | USINT     | 22            |

### Festo\_MM\_AnalogueTorque

| Name         | Data Type | Default Value |
|--------------|-----------|---------------|
| RecordTypeNo | USINT     | 23            |

### Festo\_MM\_TorqueWithBrake

| Name                    | Data Type | Default Value |
|-------------------------|-----------|---------------|
| RecordTypeNo            | USINT     | 25            |
| Target_torque           | REAL      |               |
| Torque_rise_ramp        | REAL      |               |
| Limit_speed             | REAL      |               |
| Activation_stroke_limit | BOOL      |               |
| Stroke_limit_negative   | REAL      |               |
| Stroke_limit_positive   | REAL      |               |
| HoldingBrake            | USINT     |               |

### Festo\_MM\_PositionTrigger

| Name         | Data Type | Default Value                                                                                                                                                                                               |
|--------------|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| RecordTypeNo | USINT     | 26                                                                                                                                                                                                          |
| Mode         | USINT     | 0: Inactive<br>1: ON (manual)<br>2: OFF (manual)<br>3: Force last status<br>4: Switch-on time (manual)<br>5: Switch-off time (manual)<br>6: Automatic<br>7: Set modulo position<br>8: Reset modulo position |
| Instance     | USINT     |                                                                                                                                                                                                             |
| UpdateMode   | USINT     | 0: Immediately<br>1: At modulo transition                                                                                                                                                                   |

### Festo\_MM\_SetDigitalOutput

| Name         | Data Type | Default Value                                          |
|--------------|-----------|--------------------------------------------------------|
| RecordTypeNo | USINT     | 27                                                     |
| OutputID     | USINT     | 33: Record table output 0<br>34: Record table output 1 |

|        |      |  |
|--------|------|--|
| Status | BOOL |  |
|--------|------|--|

#### Festo\_MM\_TouchProbe

| Name         | Data Type | Default Value                                                                                                                                                                            |
|--------------|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| RecordTypeNo | USINT     | 28                                                                                                                                                                                       |
| Mode         | USINT     | 0: Inactive<br>1: Once<br>2: Once with window<br>3: Cyclic<br>4: Cyclic with window<br>7: Set modulo position<br>8: Once via CiA402<br>9: Cyclic via CiA402<br>10: Reset modulo position |
| Instance     | USINT     |                                                                                                                                                                                          |

#### Festo\_MM\_Modulo

| Name         | Data Type | Default Value                                                                                                                                                                                                                                                                 |
|--------------|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| RecordTypeNo | USINT     | 29                                                                                                                                                                                                                                                                            |
| ModuloMode   | USINT     | 0: Inactive<br>1: Shortest path<br>2: Shortest path with turns<br>3: Within the modulo limits<br>4: Only positive path<br>5: Only positive path with turns<br>6: Only negative path<br>7: Only negative path with turns<br>8 :Set modulo position<br>9: Reset modulo position |

#### Festo\_MM\_ChangeParameterSet

| Name           | Data Type | Default Value |
|----------------|-----------|---------------|
| RecordTypeNo   | USINT     | 30            |
| ParameterSet   | USINT     |               |
| TranistionTime | REAL      |               |

### Festo\_MM\_SetPositionStatusword2

| Name         | Data Type | Default Value |
|--------------|-----------|---------------|
| RecordTypeNo | USINT     | 31            |
| StatusID     | USINT     |               |
| State        | BOOL      |               |

### Festo\_MM\_MovingFixedStop

| Name            | Data Type | Default Value                                                                                                                                                                                                                          |
|-----------------|-----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| RecordTypeNo    | USINT     | 32                                                                                                                                                                                                                                     |
| Type            | USINT     | 0: absolute positioning<br>1: relative actual position<br>2: relative setpoint position<br>3: relative target position<br>4: relative captured position0<br>5: relative captured position1<br>6: relative actual position unreferenced |
| Target_position | REAL      |                                                                                                                                                                                                                                        |
| Velocity        | REAL      |                                                                                                                                                                                                                                        |
| Acceleration    | REAL      |                                                                                                                                                                                                                                        |
| Deceleration    | REAL      |                                                                                                                                                                                                                                        |
| Jerk            | REAL      |                                                                                                                                                                                                                                        |

### Festo\_MM\_MovedFixedStopParameter

| Name                   | Data Type | Default Value |
|------------------------|-----------|---------------|
| RecordTypeNo           | USINT     | 33            |
| Clamping_torque        | REAL      |               |
| Offset_clamping_torque | REAL      |               |
| Stroke_limit_negative  | REAL      |               |
| Stroke_limit_positive  | REAL      |               |

### Festo\_MM\_BrakeTest

| Name         | Data Type | Default Value |
|--------------|-----------|---------------|
| RecordTypeNo | USINT     | 34            |

### Festo\_MM\_GearInOut

| Name         | Data Type | Default Value                                                                                                                           |
|--------------|-----------|-----------------------------------------------------------------------------------------------------------------------------------------|
| RecordTypeNo | USINT     | 35                                                                                                                                      |
| GearInMode   | USINT     | 0: Synchronous velocity<br>1: Synchronous Position Absolute<br>2: Synchronous Position Relative 1<br>3: Synchronous Position Relative 2 |
| GearOutMode  | USINT     | 0: Stop<br>1: Position 1<br>2: Position 2<br>3: Velocity<br>4: Without down synchronization                                             |
| Offset       | REAL      |                                                                                                                                         |

### Festo\_MM\_SetmasterPositionGear

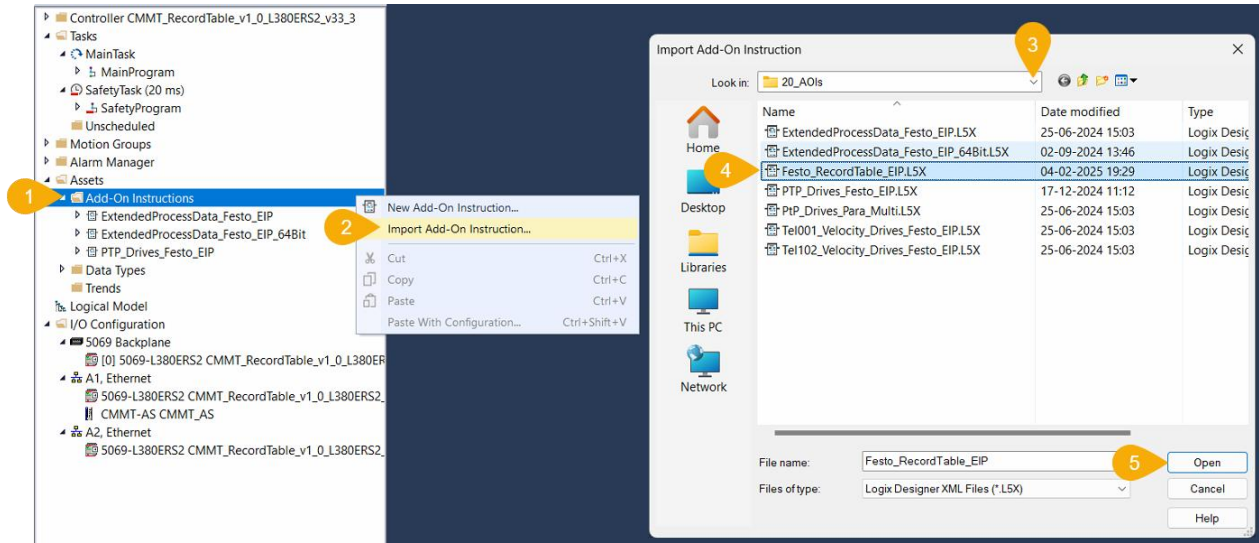
| Name         | Data Type | Default Value |
|--------------|-----------|---------------|
| RecordTypeNo | USINT     | 36            |
| Value        | USINT     |               |

### Festo\_MM\_Torque\_Percent

| Name                    | Data Type | Default Value            |
|-------------------------|-----------|--------------------------|
| RecordTypeNo            | USINT     | 37                       |
| Target_torque_percent   | REAL      | Percentage 0.0 to 100.0% |
| Torque_rise_ramp        | REAL      |                          |
| Limit_speed             | REAL      |                          |
| Activation_stroke_limit | BOOL      |                          |
| Stroke_limit_negative   | REAL      |                          |
| Stroke_limit_positive   | REAL      |                          |

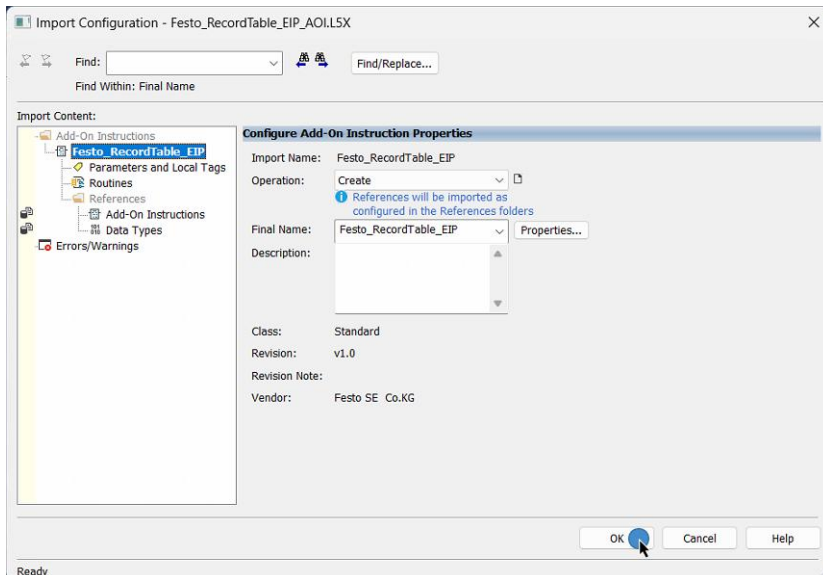
## 12.3 Import Festo\_RecordTable\_EIP AOI

### Step – 1



1. Select “Add-On Instructions” from controller organizer menu & Right click.
2. Select “Import Add-On Instruction” from right click menu
3. Browse the “Festo\_RecordTable\_EIP.L5X” file folder location. All AOI files will be in “20\_AOI’s” folder in Library directory.
4. Select “Festo\_RecordTable\_EIP.L5X” file.
5. Click “Open” button in Import Add-On Instruction window.

### Step – 2



Click “OK” button.

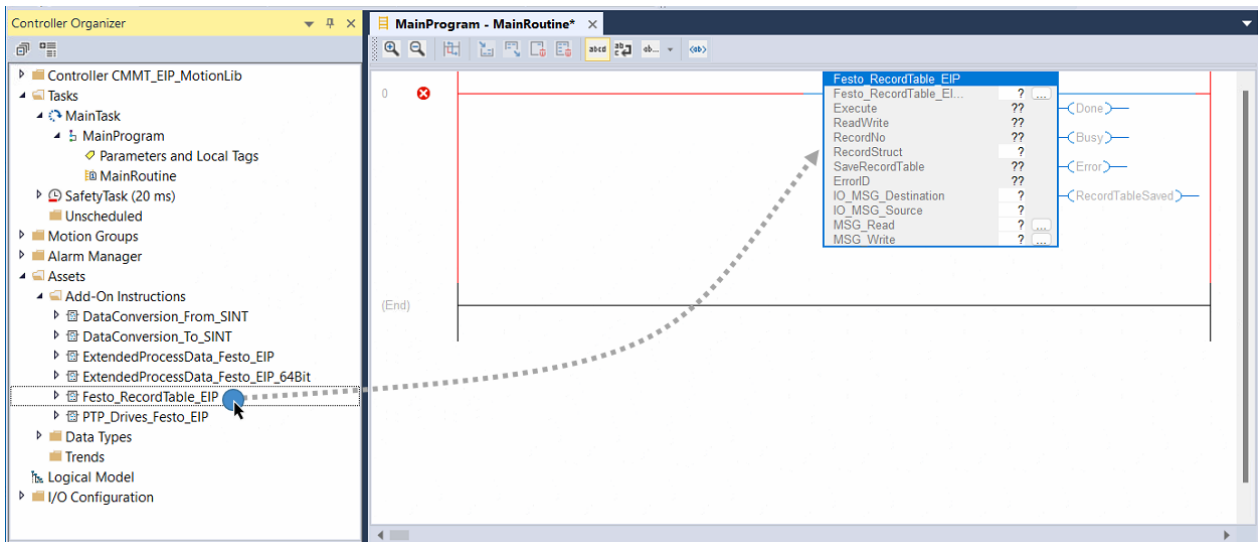
## 12.4 Add and Configure the “Festo\_RecordTable\_EIP” AOI IO Tags



### Note

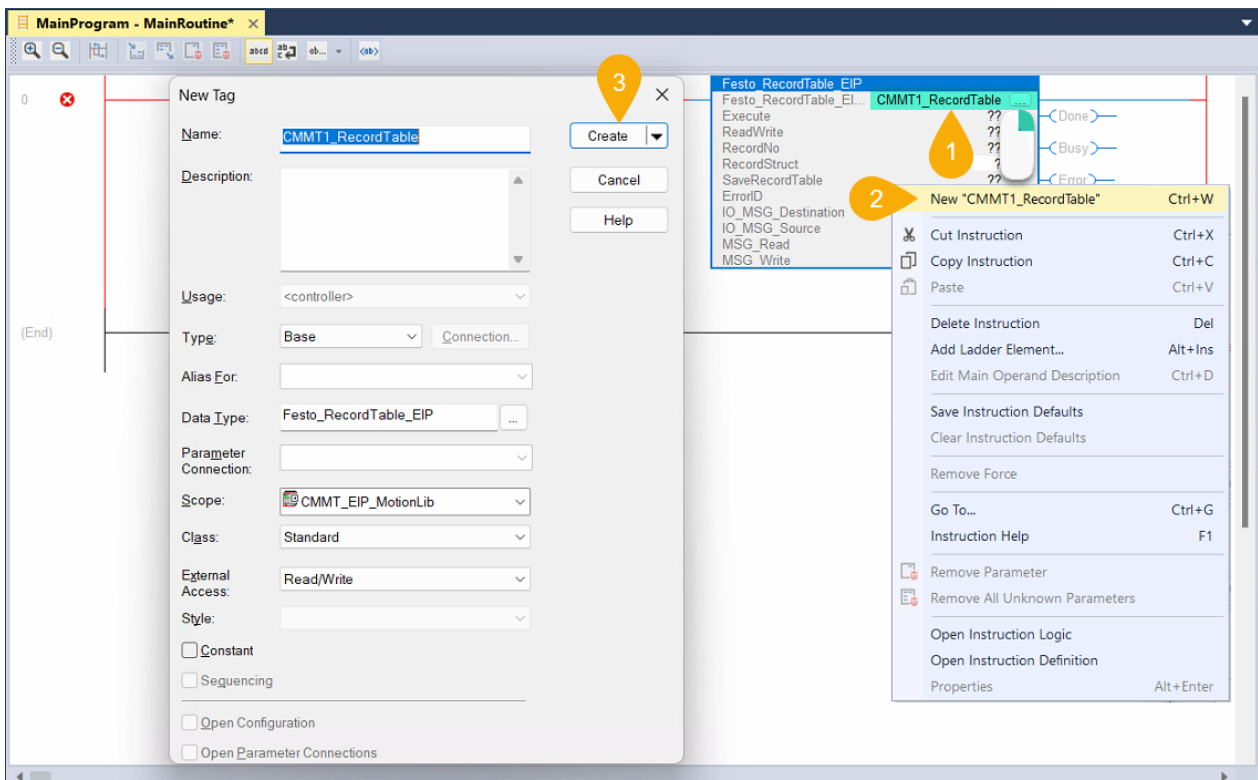
- The Add-On Instruction must be called cyclically with a separate instance for every servo drive integrated in the user program (every axis). Simultaneous use of multiple instances of the same device is not permitted.
- Any AOI's need to be in a continuous task called cyclically. [\(Refer Chapter 3.5 Step – 1\)](#)

### Step – 1



Drag and drop the “Festo\_RecordTable\_EIP” AOI file from Add-On Instructions folder as shown above.

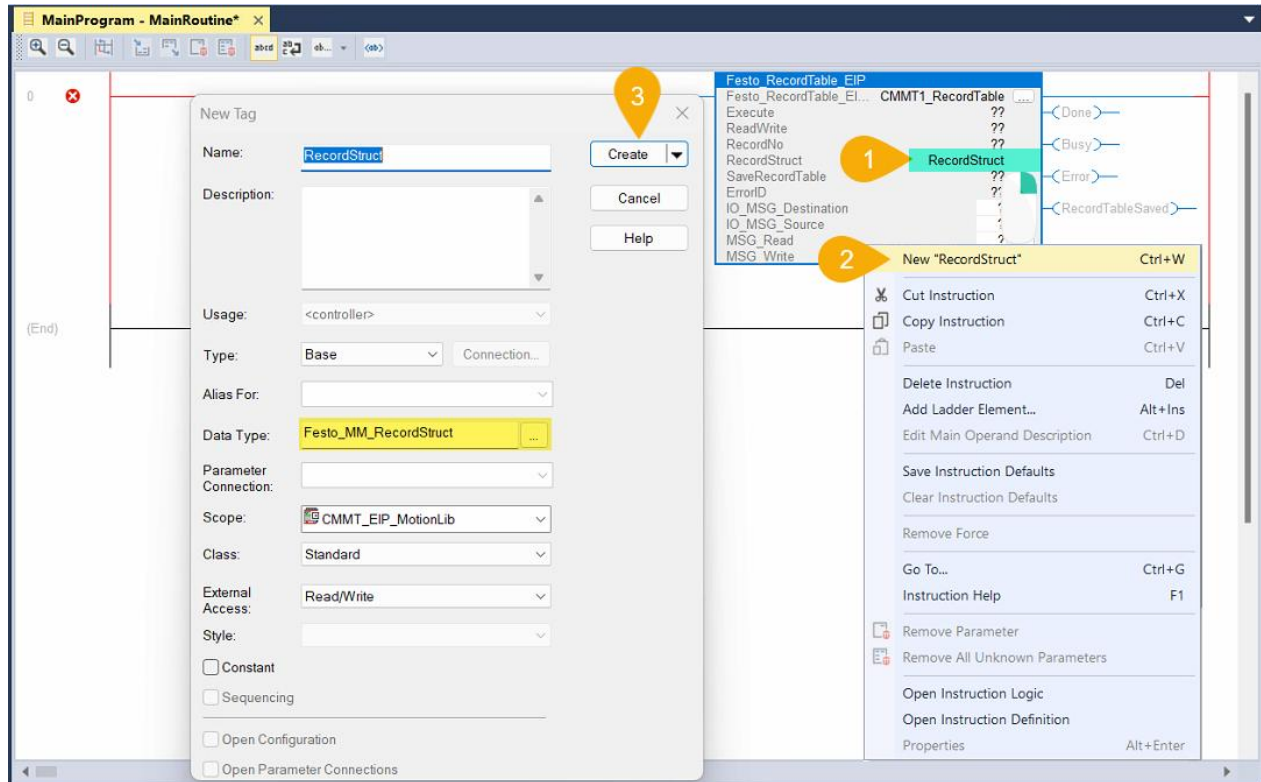
### Step – 2



## “Festo\_RecordTable\_EIP” AOI

1. Enter the name of the AOI in highlighted location & Right click it.
2. Select **“New”** with tag name option from right click menu.
3. Click **“Create”** option from new tag window.

### Step – 3



1. Enter the name of the interface tag for **“RecordStruct”** and right click.
2. Select **“New”** option with entered tag name from right click menu.
3. Click **“Create”** button on new tag window & ensure Data Type is **“Festo\_MM\_RecordStruct”** as highlighted in yellow color.

#### Step – 4

The screenshot illustrates the process of creating a new interface tag. On the left, a right-click context menu is open, with 'New "MSG\_Destination"' as the first option, highlighted by a yellow circle with the number 2. In the background, a tag list window shows 'IO\_MSG\_Destination' and 'MSG\_Destination' with a yellow circle and number 1 pointing to the latter. On the right, the 'New Tag' dialog box is open. The 'Name' field contains 'MSG\_Destination', and the 'Data Type' is set to 'SINT[8]', which is highlighted in green. A yellow circle with the number 3 points to the 'Create' button. Other fields in the dialog include 'Description', 'Usage' (set to '<controller>'), 'Type' (set to 'Base'), 'Alias For', 'Parameter Connection', 'Scope' (set to 'CMMT\_RecordTable\_v1\_0\_L380E'), 'Class' (set to 'Standard'), 'External Access' (set to 'Read/Write'), 'Style' (set to 'Decimal'), and several unchecked checkboxes at the bottom.

1. Enter the name of the interface tag for “**IO\_MSG\_Destination**” and right click.
2. Select “**New**” option with entered tag name from right click menu.
3. Click “**Create**” button on new tag window & ensure Data Type is “**SINT[8]**” as highlighted in green color.



#### Note

- Follow the above **step – 4** to create “**IO\_MSG\_Source**” interface tag.

## Step – 5

The screenshot illustrates the steps to create a new interface tag. On the left, a right-click context menu is shown with the option "New "RT\_MSG\_Read"" highlighted, indicated by a yellow callout bubble with the number 2. In the center, the "Festo\_RecordTable\_EIP" tag properties window is visible, with the "MSG\_Read" property selected, indicated by a yellow callout bubble with the number 1. On the right, the "New Tag" dialog box is open, showing the "Name" field set to "RT\_MSG\_Read" (indicated by a yellow callout bubble with the number 3), the "Data Type" set to "MESSAGE" (highlighted in green), and the "Create" button highlighted.

1. Enter the name of the interface tag for “**Read\_MSG**” and right click.
2. Select “**New**” option with entered tag name from right click menu.
3. Click “**Create**” button on new tag window & ensure Data Type is “**MESSAGE**” as highlighted in green colour.



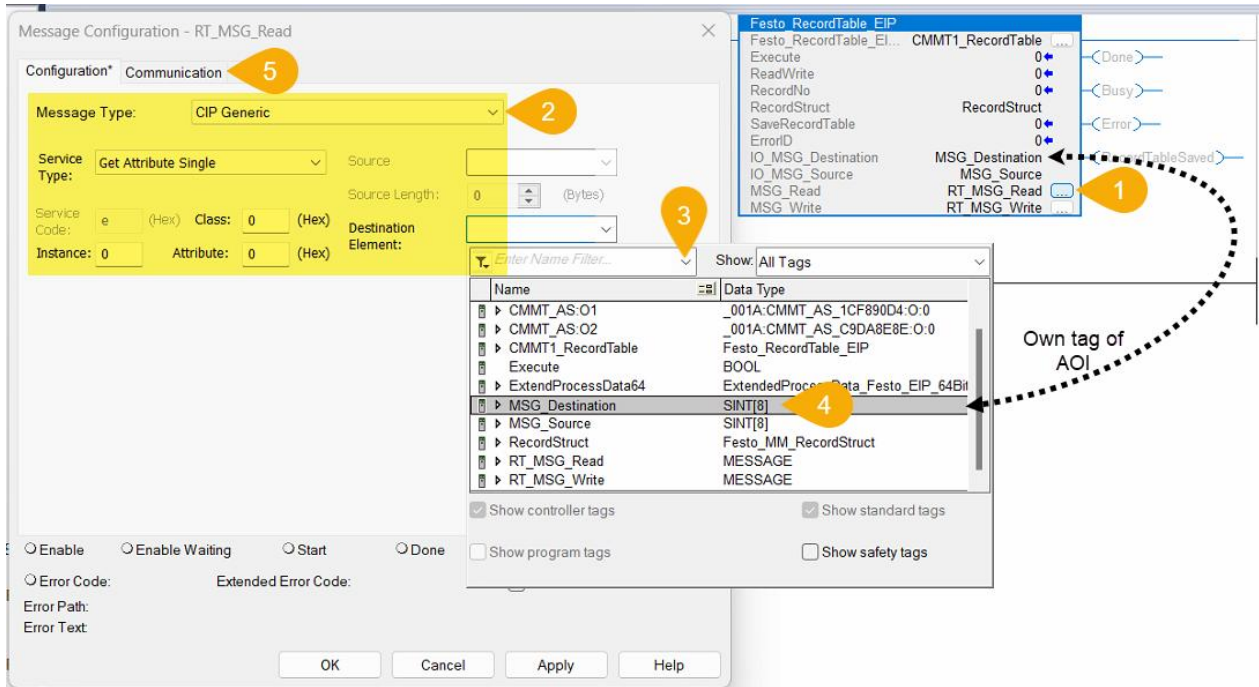
### Note

- Follow the above **step – 5** to create “**Write\_MSG**” interface tag.

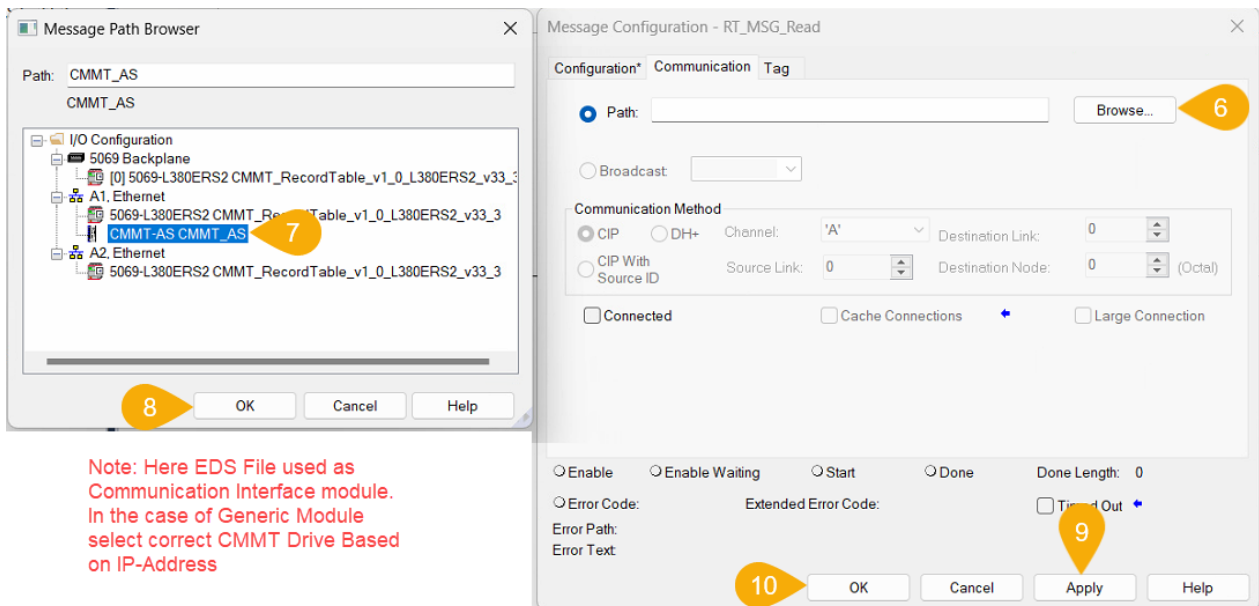
### 12.4.1 Festo\_RecordTable\_EIP AOI Read/Write MSG Instruction Configuration

This chapter explain about read & write message instruction configuration.

#### Read MSG Instruction Configuration



1. Click browse button of “**Read\_Parameter**” interface tag.
2. Set the yellow highlighted parameters as it is in above picture.
3. Click drop-down button of “**Destination Element**” to browse destination tag.
4. Browse the tag “**Read\_MSG\_FeedBack**” from its own AOI tag list & select it as destination tag.
5. Click communication tab to assign destination device.



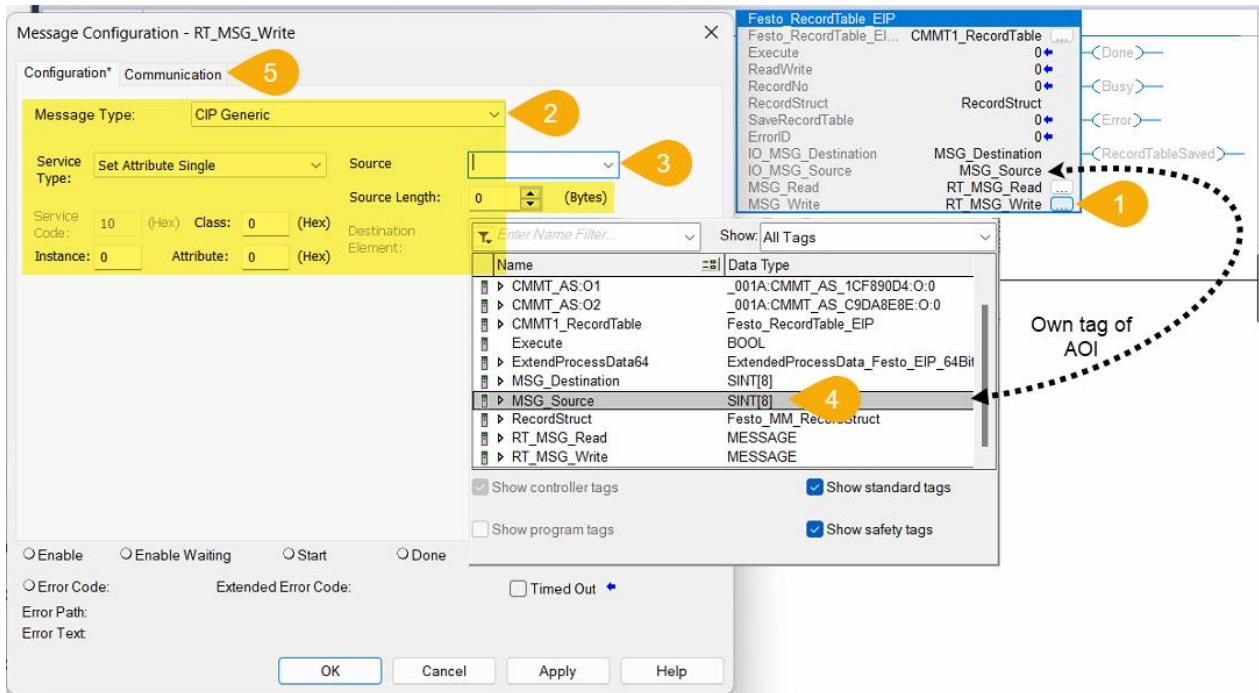
Note: Here EDS File used as Communication Interface module. In the case of Generic Module select correct CMMT Drive Based on IP-Address

6. Click “**Browse**” button from communication tab.
7. Browse the correct CMMT drive based on IP address in Message Path Browser.
8. Click “**OK**” button to confirm the communication module selection.

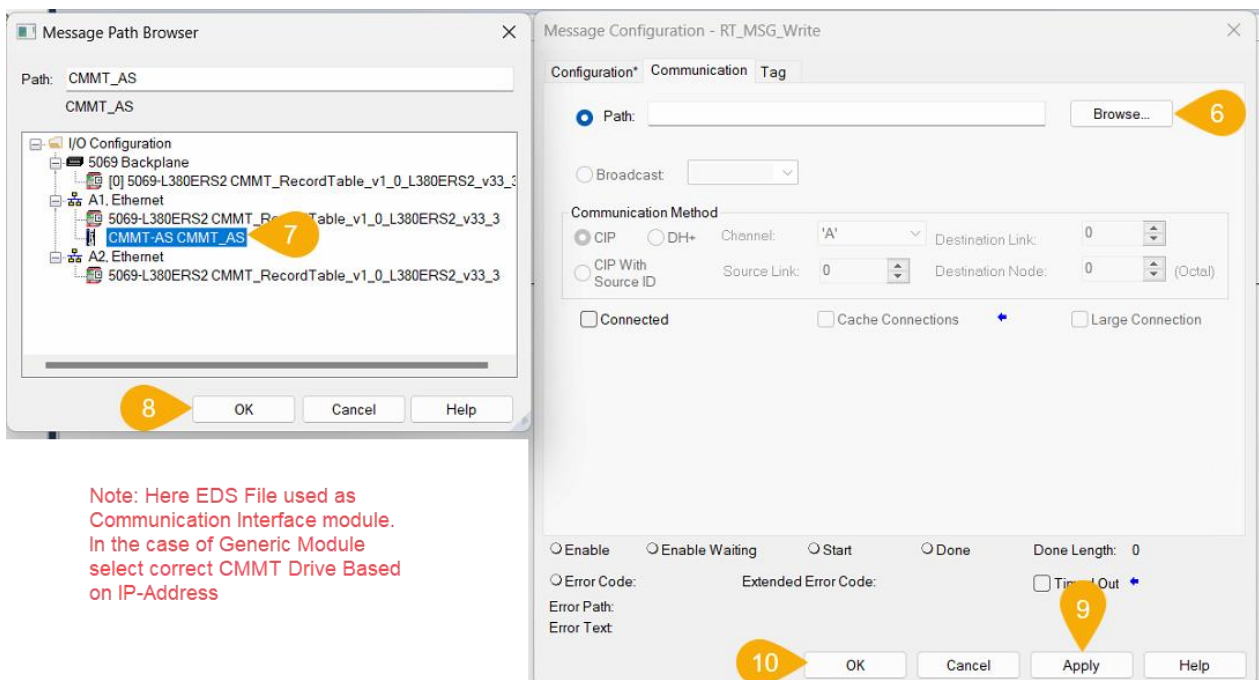
“Festo\_RecordTable\_EIP” AOI

9. Click **“Apply”** button to confirm the configuration.
10. Click **“OK”** button to confirm & close the MSG instruction configuration.

## Write MSG Instruction Configuration



1. Click Browse button of **“Write\_Parameter”** interface tag.
2. Set the yellow highlighted parameters as it is in above picture.
3. Click drop-down button of **“Source Element”** to browse destination tag.
4. Browse the tag **“Write\_MSG\_Value”** from its own AOI tag list & select it as source tag.
5. Click communication tab to assign destination device.



6. Click **“Browse”** button from communication tab.

7. Browse the correct CMMT drive based on IP address in Message Path Browser
8. Click **“OK”** button to confirm the communication module selection
9. Click **“Apply”** button to confirm the configuration.
10. Click **“OK”** button to confirm & close the MSG instruction configuration.

## 12.5 Application Example

### 12.5.1 Record\_Type

The meaning of the Record\_type can be found in the following table.

| Record_Type_No | Record_Type                 |
|----------------|-----------------------------|
| 2              | Stop ramp                   |
| 3              | Homing                      |
| 4              | Velocity                    |
| 5              | Position                    |
| 7              | Torque                      |
| 10             | Flow Control                |
| 21             | Position Analogue           |
| 22             | Velocity Analogue           |
| 23             | Torque Analog               |
| 25             | Torque with brake           |
| 26             | Position trigger            |
| 27             | Set digital output          |
| 28             | Touch probe                 |
| 29             | Modulo                      |
| 30             | Change parameter set        |
| 31             | Set position statusword 2   |
| 32             | Moving fixed stop           |
| 33             | Moving fixed stop parameter |
| 34             | Brake test                  |
| 35             | Gear In/Out                 |
| 36             | Set master position         |
| 37             | Torque Percent              |

### 12.5.2 Example for Read Record Table Using Festo\_RecordTable\_EIP AOI

If a read job is started the read values are written to the assigned “RecordStruct.Record\_Type”. The data type must either correspond to the record type read (e.g. Festo\_MM\_Position; Festo\_MM\_Velocity) to the general struct (Festo\_MM\_RecordStruct). This general struct contains all valid record types. The read data will be written automatically into the suitable substructure.

The screenshot displays the Allen-Bradley Studio 5000 software interface. On the left, the 'MainProgram - MainRoutine' ladder logic is shown with the 'Festo\_RecordTable\_EIP' AOI block. The block's parameters are configured: 'RecordNo' is set to 2 (highlighted in purple), 'Execute' is set to 1, and 'RecordStruct' is assigned to 'RecordStruct'. The 'RecordStruct' variable is expanded in the 'Controller Tags' window on the right, showing the 'RecordStruct.Record\_Type' tag with a value of 4 (highlighted in green). Below the ladder logic, the 'Record Table' configuration window is open, showing a table of record types. The table has columns for 'Record type', 'Deceleration', 'Jerk', 'Target velocity', 'Acceleration', 'Deceleration', 'Jerk', 'Activation stroke limit', 'Stroke limit negative', 'Stroke limit positive', and 'Comment'. The table lists six record types: Stop ramp (2), Homing (3), Velocity (4), Position (5), Torque (7), and Flow control (10). The 'Velocity (4)' record type is highlighted in yellow, and its details are expanded in the table below.

| Record type       | Deceleration                                | Jerk            | Target velocity | Acceleration    | Deceleration    | Jerk            | Activation stroke limit | Stroke limit negative | Stroke limit positive | Comment |
|-------------------|---------------------------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-------------------------|-----------------------|-----------------------|---------|
| Stop ramp (2)     | 3000.00 r/min/s                             | 10000.00 rpm/s² |                 |                 |                 |                 |                         |                       |                       |         |
| Homing (3)        | Actual position (37)                        |                 |                 |                 |                 |                 |                         |                       |                       |         |
| Velocity (4)      | 150.00 r/min                                | 3000.00 r/min/s | 3000.00 r/min/s | 3000.00 r/min/s | 15000.00 rpm/s² |                 |                         | -180.00 rev           | 180.00 rev            |         |
| Position (5)      | Positioning relative to actual position (1) | 1580.00 rev     | 175.00 r/min    | 3000.00 r/min/s | 3000.00 r/min/s | 90000.00 rpm/s² |                         |                       |                       |         |
| Torque (7)        | 2.00 Nm                                     | 0.10 Nm/s       | 150.00 r/min    |                 |                 |                 |                         | -1.00 rev             | 1.00 rev              |         |
| Flow control (10) |                                             |                 |                 |                 |                 |                 |                         |                       |                       |         |

1. Enter the Record Table Number as shown in FAS, highlighted in purple in the **RecordNo** tag.
2. Set Execute = 1 to start reading the Record Table.
3. Go to the Monitor Tags window to view the record table data in **RecordStruct**. Looking at the **RecordStruct.Record\_Type**, the value highlighted in green indicates a specified **Record\_Type**, refer to [Chapter 12.5.1](#).
4. Based on the value shown in **RecordStruct.Record\_Type**, expand the **Record\_Type** to view the specified record table details highlighted in yellow, as shown.
5. Once the **RecordTable** Read Successfully, then the **Done** will be in TRUE.

### 12.5.3 Example for Write Record Table Using Festo\_RecordTable\_EIP AOI

Analogous to the read command a data structure for the write command must be assigned at the “RecordStruct” input. The values to be written for the record set are taken from this struct. Either a record specific struct (e.g. “Festo\_MM\_Position”) to the general structure (“Festo\_MM\_RecordStruct”).

With the “Festo\_MM\_RecordStruct” the associated record type is written depending on the variable “Festo\_MM\_RecordStruct.Record\_type”.

The screenshot displays the Allen-Bradley Studio 5000 software interface. The top window shows the ladder logic for the 'Festo\_RecordTable\_EIP' AOI. Callout 1 points to the 'RecordNo' tag. Callout 2 points to the 'RecordStruct' input. Callout 3 points to the 'RecordStruct.Record\_Type' variable in the variable declaration table. Callout 4 points to the 'ReadWrite' tag. Callout 5 points to the 'SaveRecordTable' tag. Callout 6 points to the 'Enable' tag. Callout 7 points to the 'RecordTableSaved' tag. Callout 8 points to the 'Done' tag.

The variable declaration table on the right lists the following variables:

| Name                                                 | Value | Style   | Data Type                 | Description                            |
|------------------------------------------------------|-------|---------|---------------------------|----------------------------------------|
| RecordStruct.Record_Type                             | 25    | Decimal | USINT                     | S Number of the record type (P1.1810)  |
| RecordStruct.StopRamp                                | (-)   | (-)     | Festo_MM_StopRamp         | S                                      |
| RecordStruct.Homing                                  | (-)   | (-)     | Festo_MM_Homing           | S                                      |
| RecordStruct.Velocity                                | (-)   | (-)     | Festo_MM_Velocity         | S                                      |
| RecordStruct.Position                                | (-)   | (-)     | Festo_MM_Position         | S                                      |
| RecordStruct.Torque                                  | (-)   | (-)     | Festo_MM_Torque           | S                                      |
| RecordStruct.FlowControl                             | (-)   | (-)     | Festo_MM_FlowControl      | S                                      |
| RecordStruct.AnaloguePosition                        | (-)   | (-)     | Festo_MM_AnaloguePosition | S                                      |
| RecordStruct.AnalogueVelocity                        | (-)   | (-)     | Festo_MM_AnalogueVelocity | S                                      |
| RecordStruct.AnalogueTorque                          | (-)   | (-)     | Festo_MM_AnalogueTorque   | S                                      |
| RecordStruct.TorqueWithBrake                         | (-)   | (-)     | Festo_MM_TorqueWithBrake  | S                                      |
| RecordStruct.TorqueWithBrake.RecordTypeNo            | 25    | Decimal | USINT                     | S Number of the record type (P1.1810)  |
| RecordStruct.TorqueWithBrake.TargetTorque            | 1.1   | Float   | REAL                      | S                                      |
| RecordStruct.TorqueWithBrake.Torque_rise_ramp        | 0.1   | Float   | REAL                      | S                                      |
| RecordStruct.TorqueWithBrake.Limit_speed             | 10.0  | Float   | REAL                      | S                                      |
| RecordStruct.TorqueWithBrake.Activation_stroke_limit | 0     | Decimal | BOOL                      | S                                      |
| RecordStruct.TorqueWithBrake.Stroke_limit_negative   | -1.0  | Float   | REAL                      | S                                      |
| RecordStruct.TorqueWithBrake.Stroke_limit_positive   | 1.0   | Float   | REAL                      | S                                      |
| RecordStruct.TorqueWithBrake.HoldingBrake            | 2     | Decimal | USINT                     | S Holding Brake 1 (0); Holding Brake 2 |
| RecordStruct.PositionTrigger                         | (-)   | (-)     | Festo_MM_PositionTrigger  | S                                      |

The parameter configuration table at the bottom shows the following records:

| Record type                      | Deceleration                                       | Jerk                 | Comment                 |
|----------------------------------|----------------------------------------------------|----------------------|-------------------------|
| 1 Stop ramp (2)                  | 3000.00 r/min/s                                    | 10000.00 rpm/s²      |                         |
| 2 Record type                    | Method                                             |                      | Comment                 |
| 3 Homing (3)                     | Actual position (37)                               |                      |                         |
| 3 Record type                    | Target velocity                                    | Acceleration         | Deceleration            |
| 4 Velocity (4)                   | 150.00 r/min                                       | 3000.00 r/min/s      | 3000.00 r/min/s         |
| 4 Record type                    | Position                                           | Target position      | Velocity                |
| 5 Position (5)                   | Positioning relative to actual actual position (1) | 1580.00 rev          | 175.00 r/min            |
| 5 Record type                    | Target torque                                      | Torque rise ramp     | Limit speed             |
| 6 Torque (7)                     | 2.00 Nm                                            | 0.10 Nm/s            | 150.00 r/min            |
| 6 Record type                    | Flow control                                       | Limit value velocity | Activation stroke limit |
| 7 Torque with holding brake (25) | 1.10 Nm                                            | 0.10 Nm/s            | 10.00 r/min             |
| 7 Record type                    | Target torque                                      | Rise ramp torque     | Limit value velocity    |
| 7 Torque with holding brake (25) | 1.10 Nm                                            | 0.10 Nm/s            | 10.00 r/min             |

1. Enter the Record Table Number where the record table is to be created or modify the existing record table, as shown in FAS and highlighted in purple in the **RecordNo** tag.
2. Go to the Monitor Tags window to write the record table data in **RecordStruct**. Enter the specified value for **Record\_Type** (refer to [Chapter 12.5.1](#) in the **RecordStruct.Record\_Type**, highlighted in green as shown).
3. Based on the value written in **RecordStruct.Record\_Type**, expand the specified **Record\_Type** to enter the value accordingly, as highlighted in yellow, as shown.  
Refer to the **note** given below and review [Chapter 12.2](#) to understand that some parameters have selection values. For example, **RecordStruct.Position.Type**.
4. Set **ReadWrite** = 1 to perform write operation.
5. Set **SaveRecordTable** = 1 to perform the save operation; otherwise, the Record Table will not be saved in the Drive.
6. Set **Enable** = 1 to start writing the Record Table.
7. Once the RecordTable saved Successfully, then the **RecordTableSved** will be in TRUE.
8. Once the RecordTable write Successfully, then the **Done** will be in TRUE.



**Note**

- If a user enters a value greater than the maximum allowed, FAS does not display any error, nor does the Record Table. Therefore, users should be aware of the default, minimum, and maximum values before writing to the record table

**Example:** If a user enters a value greater than 6 in **RecordStruct.Position.Type** (which defines the types of position selection), the record table in FAS will consider 0.

## 13 CMMT Full Param SD AOI



### Note

- The project automates the transfer of configuration data, stored on an SD card, to a CMMT drive via a TCP connection, using a PLC. The PLC retrieves data from the SD card, parses it, and transmits it to the CMMT drive.

### 13.1 Introduction of CMMT Full Param SD AOI

This application note provides a step-by-step guide to configure the Function Block '**CMMT\_Full\_Param\_SD.**' Additionally, this AOI is used to parameterize the new CMMT drive or an existing CMMT drive to restore previous parameters using the L5x file generated by FAS (Festo Automation Suite) from the SD card in the Rockwell Controller.

### 13.2 Configure FAS

#### Step – 1

Download latest FAS from support portal **V2.9.0.719** and above and Start with the installation of FAS.

#### Step – 2

After Installation done, continue with the download of **Plug-ins(CMMT-AS) V2.9.0.977** and above.

## Repositories



### Note

Repositories (requires Internet connection)

#### 13.2.1 Creating Rockwell file L5X

#### Step – 1

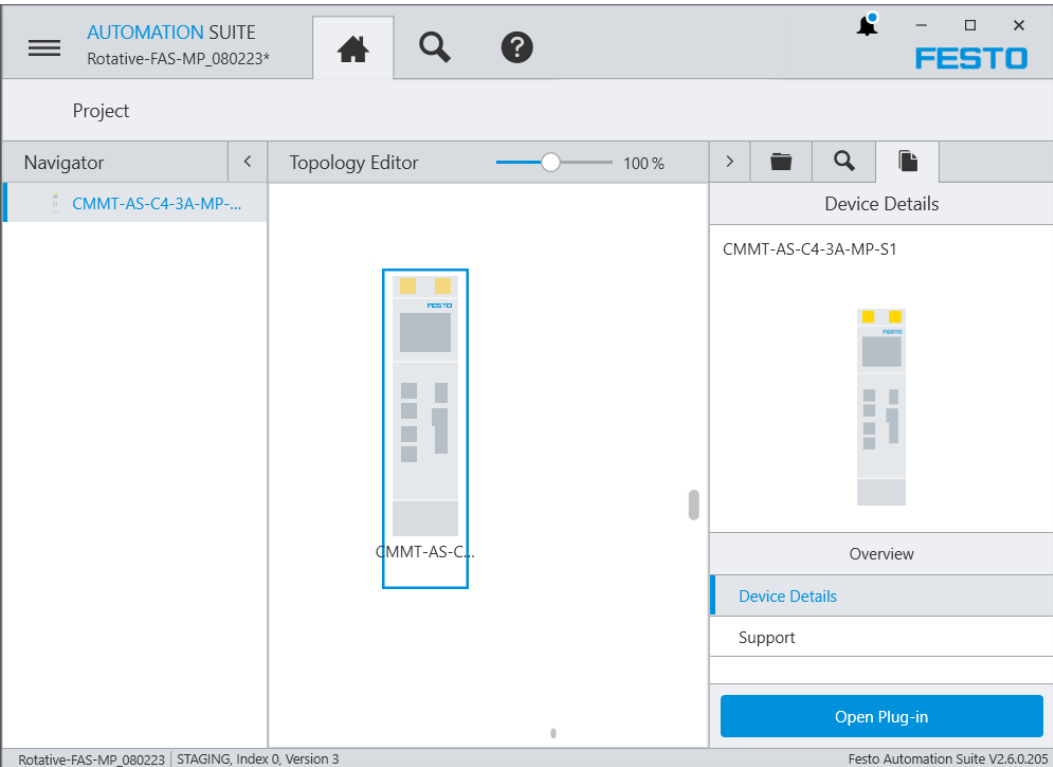
To create Rockwell file L5X, open the FAS project.



### Note

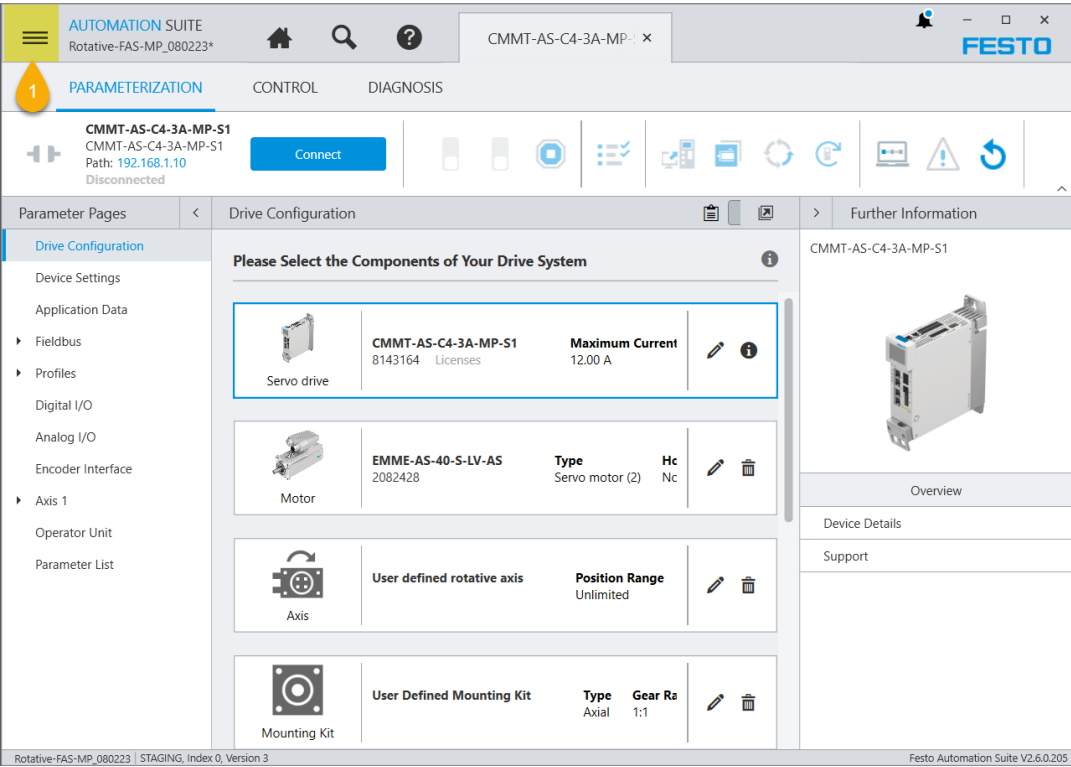
- Either Create New Project in FAS as per user settings (Refer [Chapter 2.4](#) and Refer [Chapter 2.5](#) for Integration with PLC), or Open previous saved project if available.
- In-case if user does not have previous project of Automation Suite then, read the settings from Device(CMMT) to FAS.

Step – 2

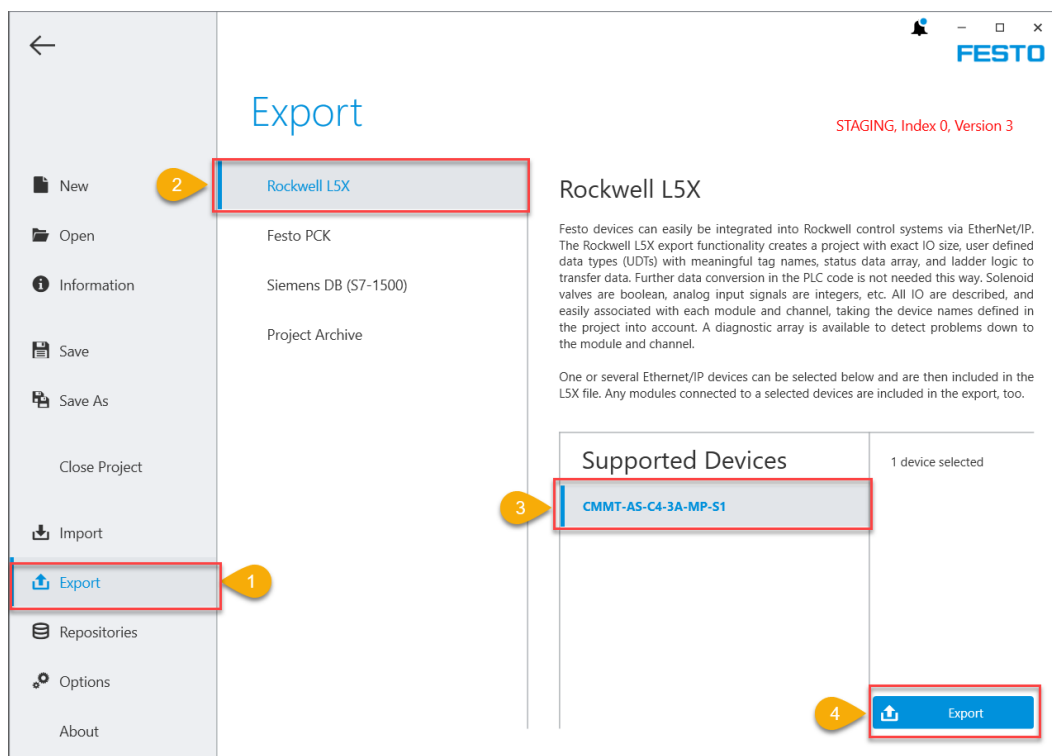


Double click on the project.

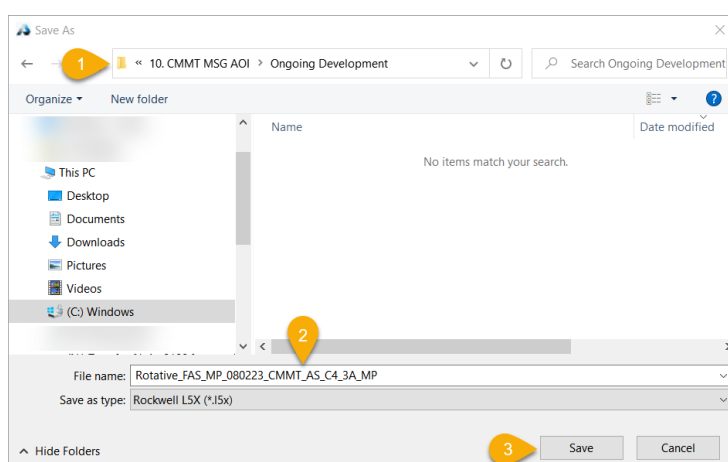
Step – 3



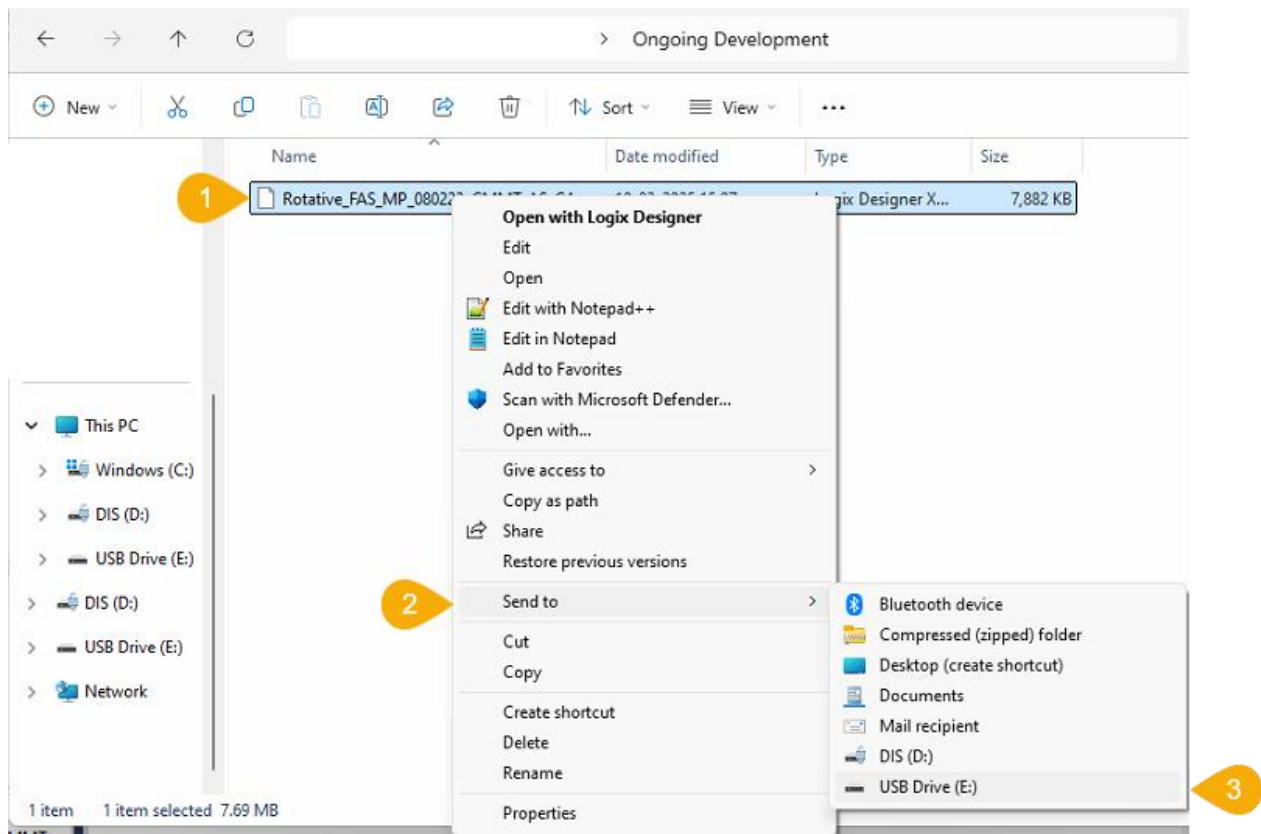
1. Click on the Menu button.

**Step – 4**

1. Click on Export
2. Select Rockwell L5X
3. Select CMMT-xx configuration which exist
4. Click on “Export”

**Step – 5**

1. Choose the location to save the Project
2. Enter the user desired **File name**.
3. Click on “Save”

**Step – 6**

1. Right-click on the exported file from FAS.
2. Select "Send to."
3. Click on "USB Drive" to transfer the exported configuration file to your memory card.

**Note**

- Insert the controller memory card into your workstation to transfer the L5x file.

### 13.3 Import CMMT Full Parameter SD AOI

This chapter will describe about importing the CMMT\_FullParam\_SD AOI file inside the PLC program.

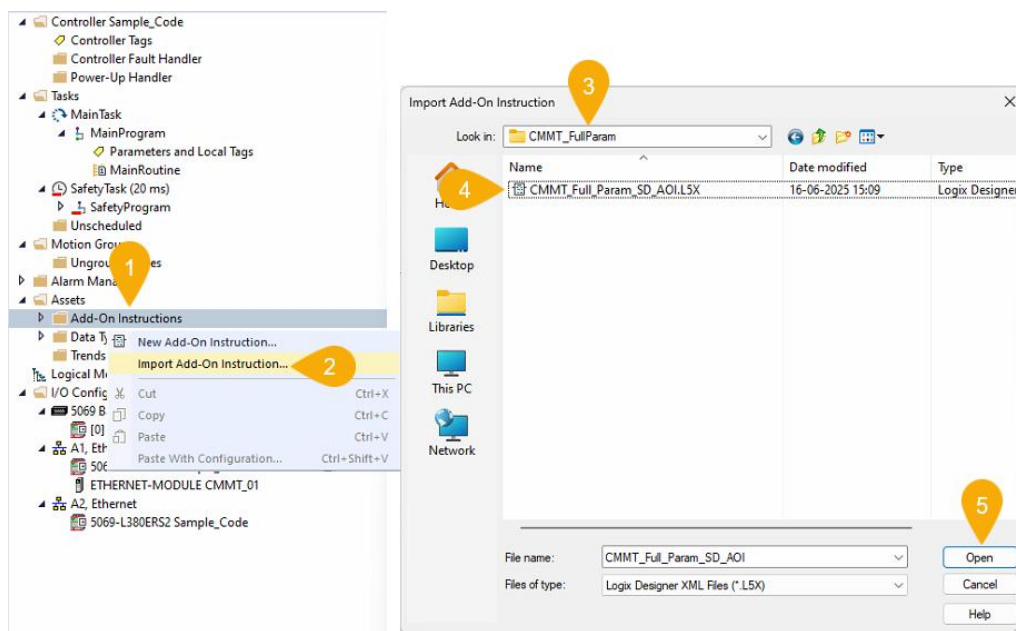


#### Note

There are hardware limitations with certain Allen Bradley controllers or EtherNet/IP masters. The AOI (Add-On Instruction) requires Socket Services which are ONLY available/supported with these Controllers/Modules listed below:

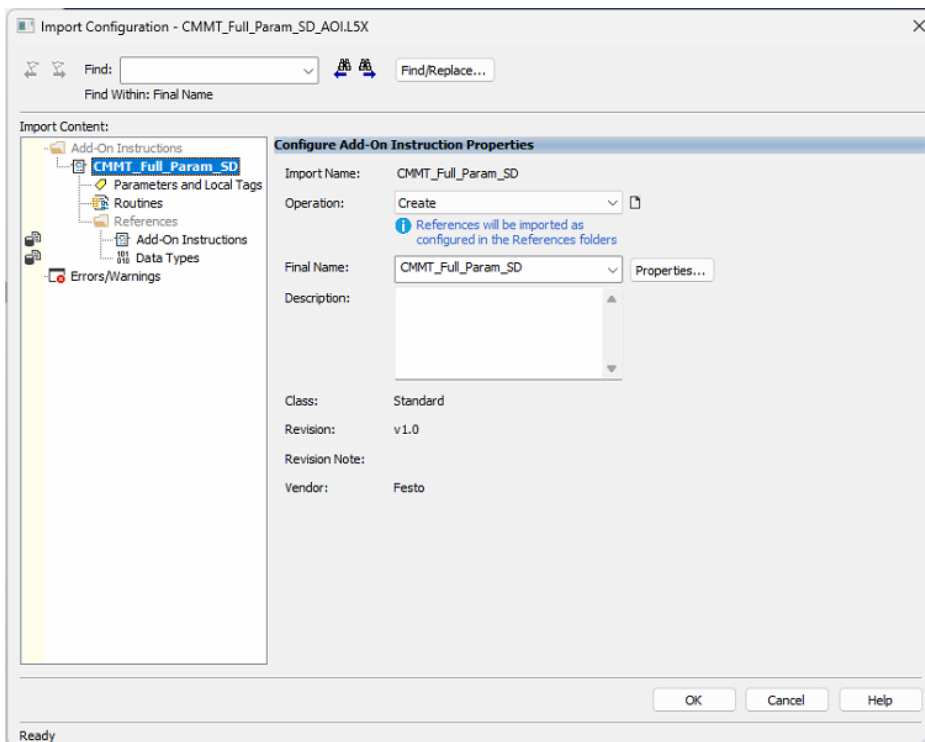
- ControlLogix® 5580 and GuardLogix® 5580 controllers
- CompactLogix™ 5380 and Compact GuardLogix 5380 controllers
- CompactLogix 5370 and Compact GuardLogix 5370 controllers
- CompactLogix 5480 controllers
- 1756-EN2x, 1756-EN3TR, and 1756-EN4TR EtherNet/IP communication modules
- 1756-EWEB and 1768-EWEB EtherNet/IP web server modules

#### Step – 1



1. Select **"Add-On Instructions"** from controller organizer menu & Right click.
2. Select **"Import Add-On Instruction"** from right click menu
3. Browse folder location to open the Project
4. Select **"CMMT\_Full\_Param\_SD.L5X"** file.
5. Click **"Open"** button in Import Add-On Instruction window.

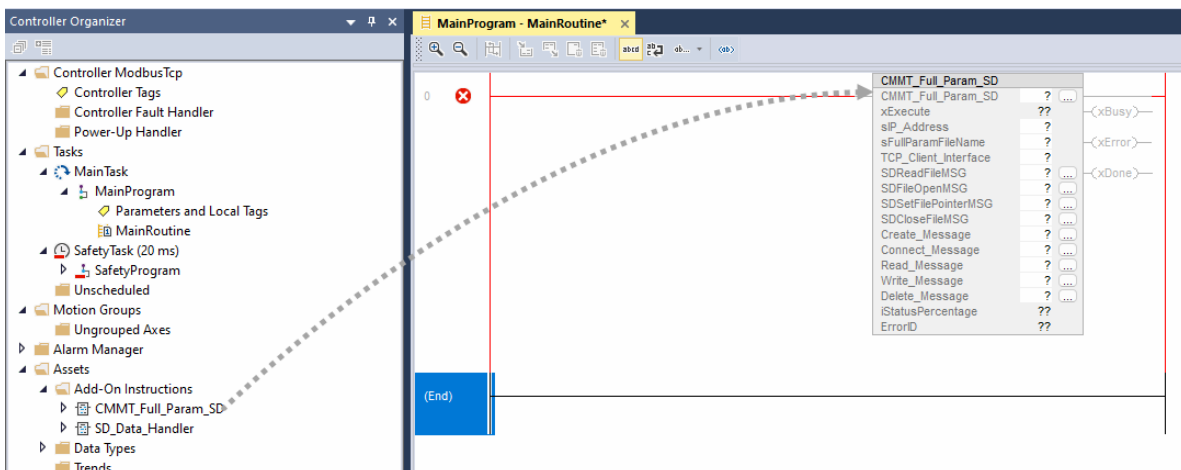
## Step –2



Click “OK” button.

## 13.4 Adding “CMMT\_FullParam\_SD” AOI in PLC Program

### Step –1

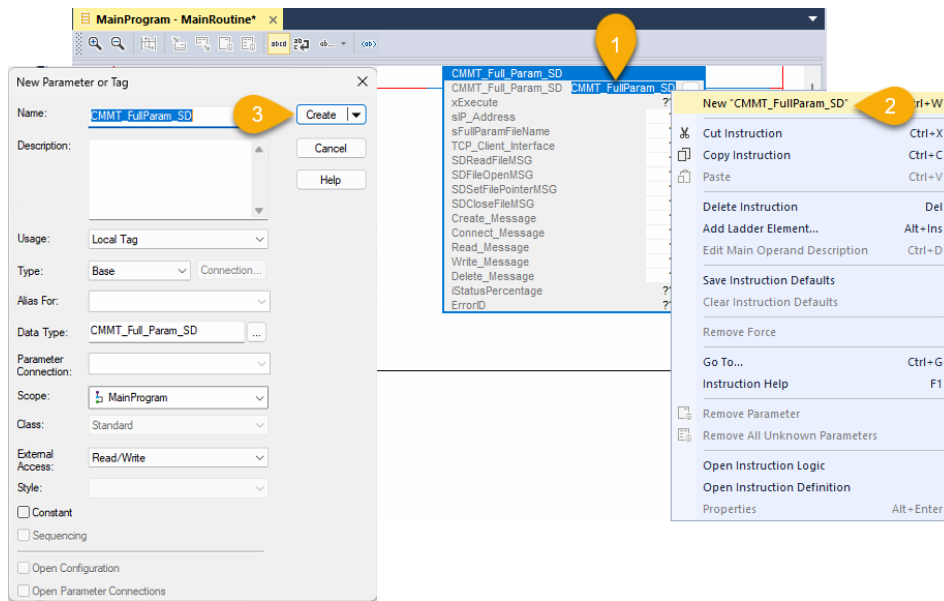


Drag and drop the “CMMT\_Full\_Param\_SD” AOI file from Add-On Instructions folder as shown above.

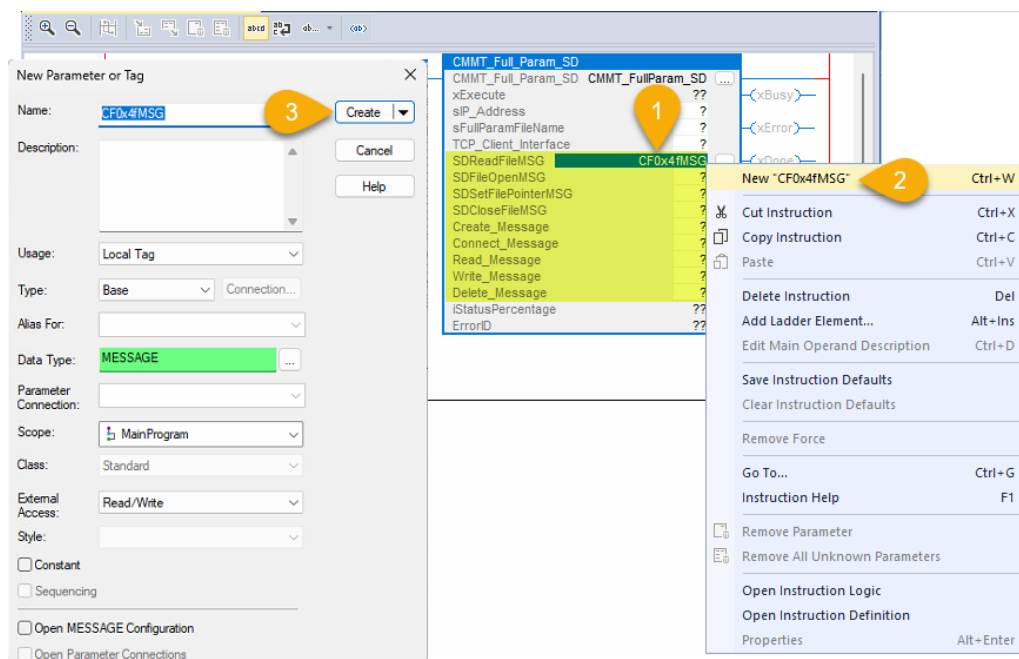


#### Note

- The SD\_Data\_Handler supporting AOI will be imported automatically when the user imports the CMMT\_Full\_Param\_SD AOI file.

**Step –2**

1. Enter the Tag name for the “CMMT\_FullParam\_SD” AOI & right click.
2. Select “New” option with tag name from right click menu.
3. Click “Create” button from New Tag window.

**13.5 CMMT FullParam\_SD AOI Tag’s Configuration****Step –1**

1. Enter the tag name for “SDReadFileMSG” and right-click.
2. Select the “New” option from the right-click menu.

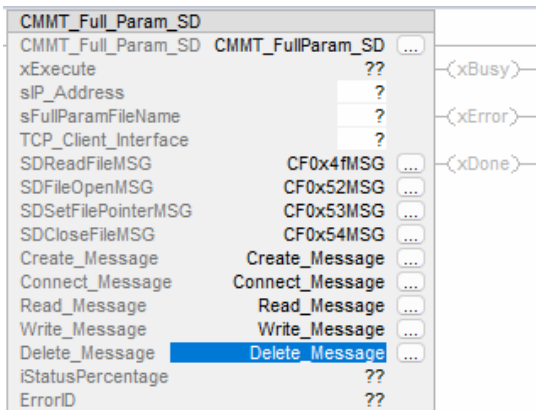
- Click the **“Create”** button in the new tag window and ensure that the data type is set to **“MESSAGE,”** as highlighted in the green colour.



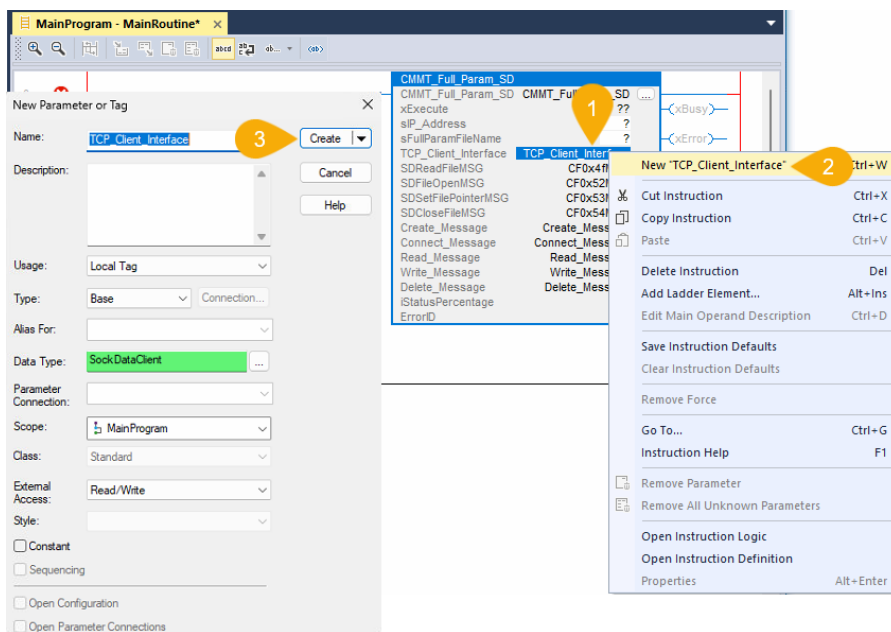
#### Note

- Follow the above step (Step 1) to create the following interface tags **“SDReadFileMSG,SDFileOpenMSG,SDSetFilePointerMSG,SDCloseFileMSG,Create\_Message,Connect\_Message,Read\_Message,Write\_Message,Delete\_Message”**

Once the above steps are completed successfully the AOI looks like below.

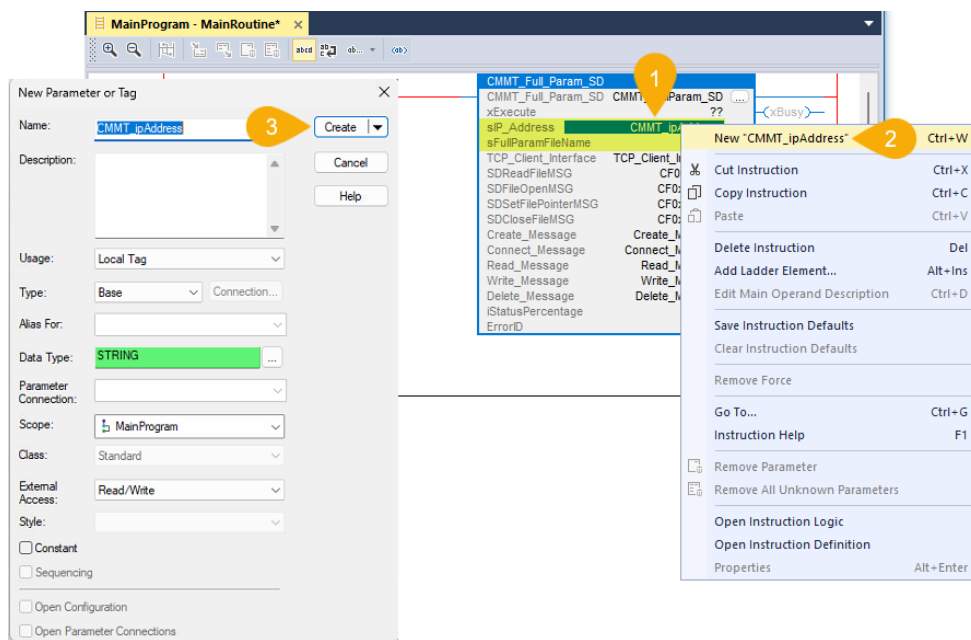


## Step –2



- Enter the Tag name for the **“TCP\_Client\_Interface”** & right click.
- Select **“New”** option with tag name from right click menu.
- Click **“Create”** button on new tag window & ensure Data Type is **“SockDataClient”** as highlighted in green colour.

### Step –3

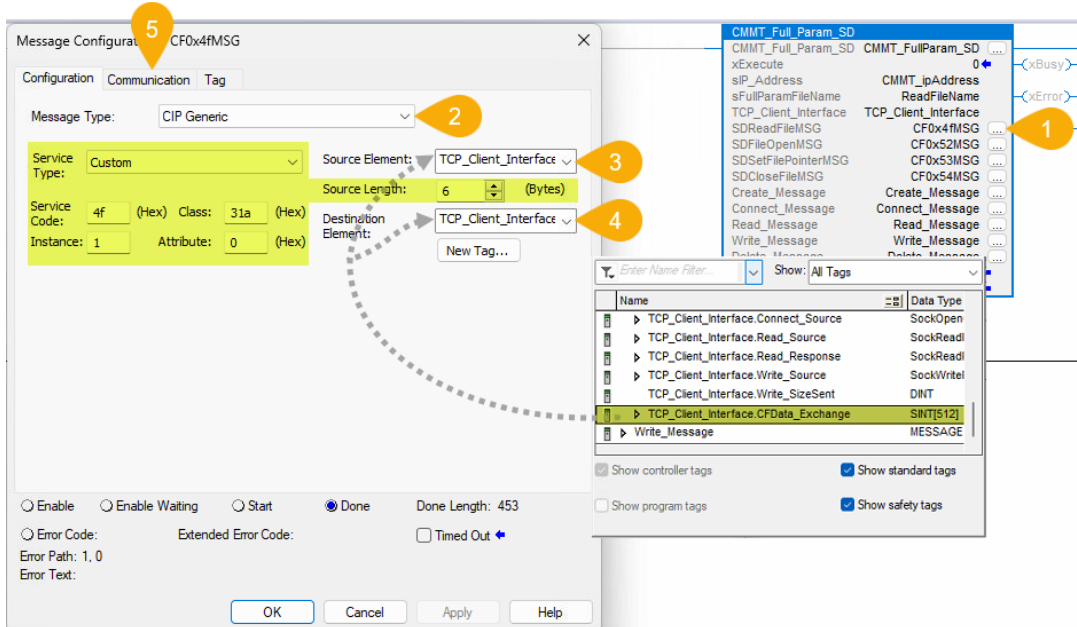


1. Enter the Tag name for the “**sIP\_Address & sFullParamFileName**” & right click.
2. Select “**New**” option with tag name from right click menu.
3. Click “**Create**” button on new tag window & ensure Data Type is “**String**” as highlighted in green color.

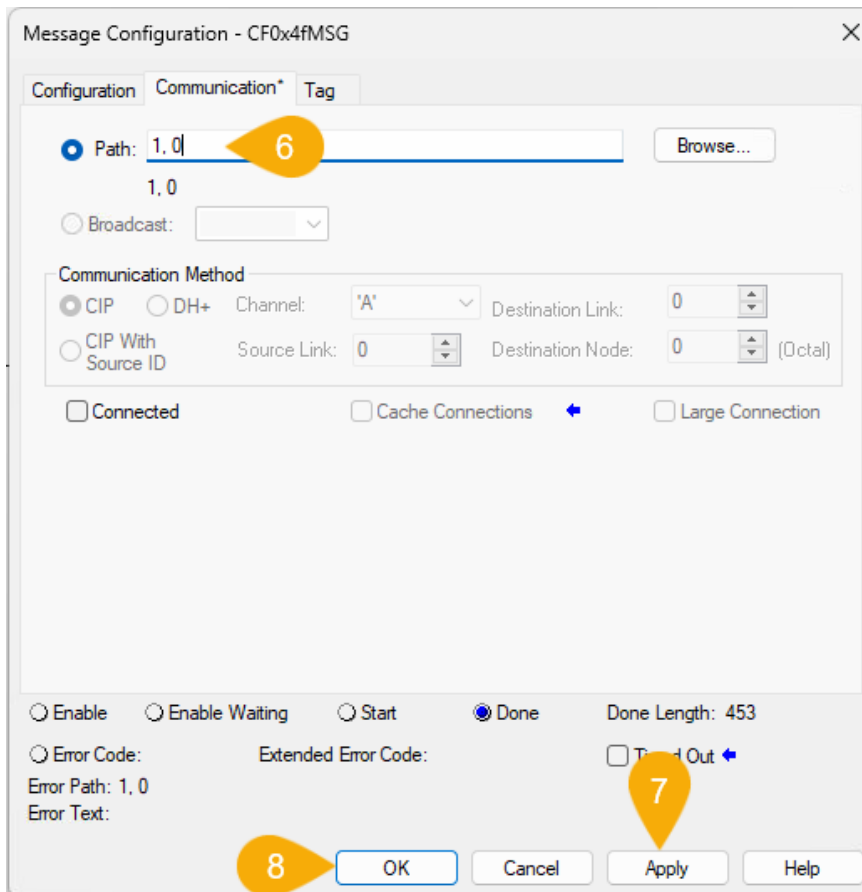
## 13.6 CMMT\_FullParam\_SD AOI MSG Instruction Configuration

This chapter explain about message instruction configuration.

### 13.6.1 SDReadFileMSG Instruction Configuration

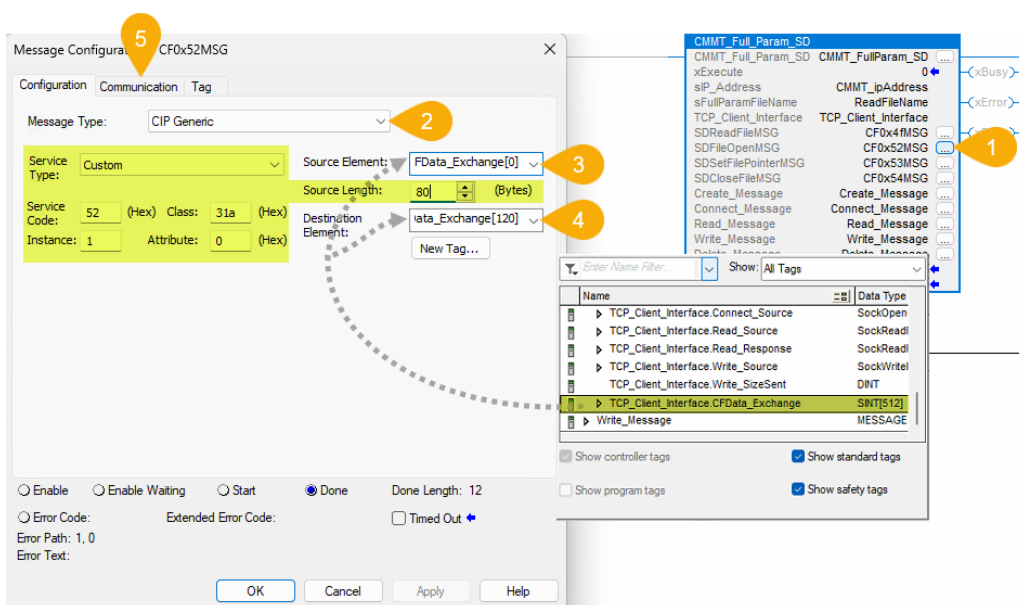


1. Click browse button of “**SDReadFileMSG**” interface tag.
2. Set the yellow highlighted parameters as it is in above picture.
3. Click drop-down button of “**Source Element**” to browse Source tag. Browse the tag “**TCP\_Client\_Interface.CFData\_Exchange[0]**” from tag list & select it as Source tag.
4. Click drop-down button of “**Destination Element**” to browse destination tag. Browse the tag “**TCP\_Client\_Interface.CFData\_Exchange[20]**” from tag list & select it as Destination tag.
5. Click communication tab to assign destination device.



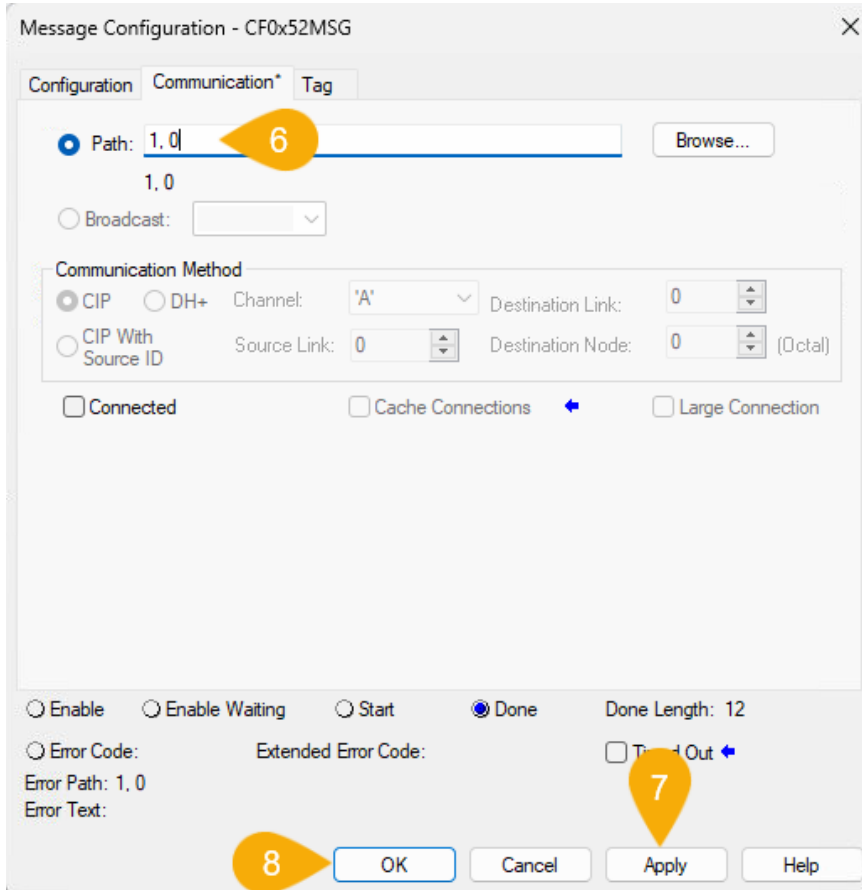
6. Enter “1, slot number” in path from communication tab. (Example: 1, 0 ([Refer Chapter 13.6.10 Communication Path](#)))
7. Click “Apply” button to confirm the configuration.
8. Click “OK” button to confirm & close the MSG instruction configuration.

### 13.6.2 SDFileOpenMSG Instruction Configuration



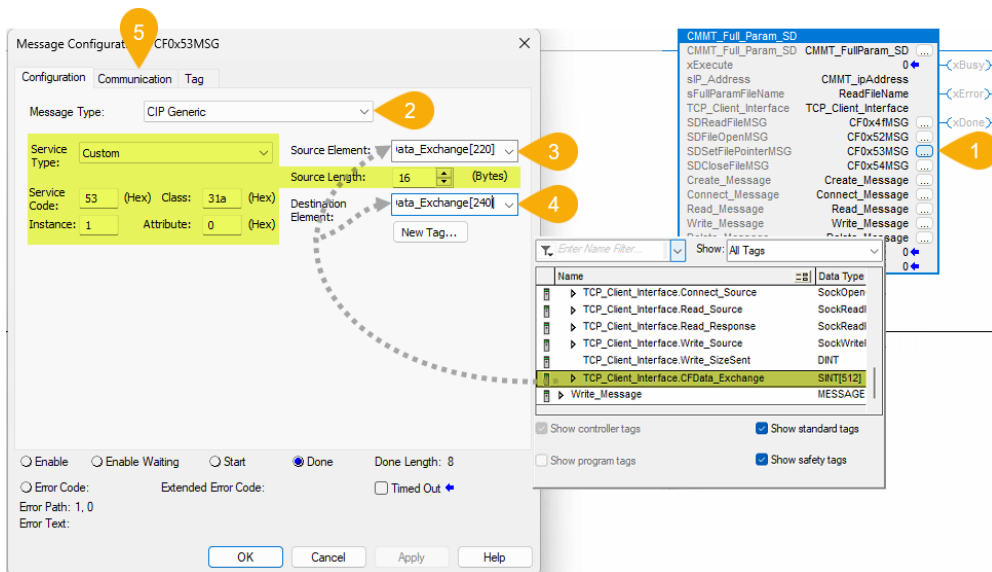
1. Click browse button of “SDFileOpenMSG” interface tag.
2. Set the yellow highlighted parameters as it is in above picture.

3. Click drop-down button of “**Source Element**” to browse Source tag. Browse the tag “**TCP\_Client\_Interface.CFData\_Exchange[0]**” from tag list & select it as Source tag.
4. Click drop-down button of “**Destination Element**” to browse destination tag. Browse the tag “**TCP\_Client\_Interface.CFData\_Exchange[120]**” from tag list & select it as Destination tag.
5. Click communication tab to assign destination device.

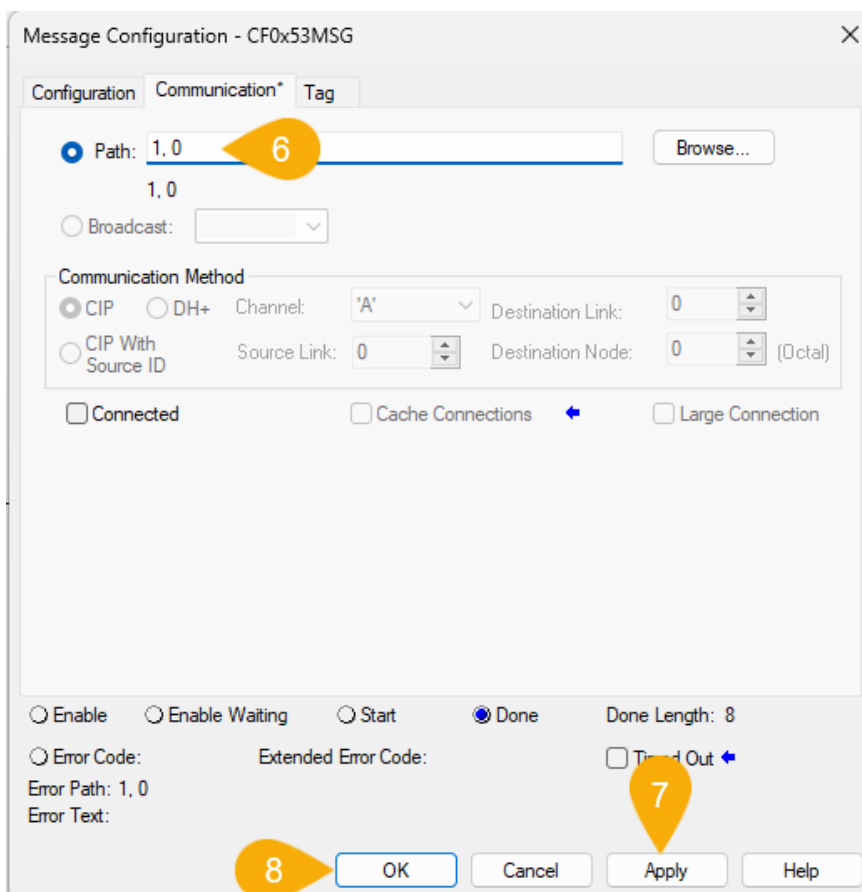


6. Enter “**1, slot number**” in path from communication tab. (Example: 1, 0 ([Refer Chapter 13.6.10 Communication Path](#)))
7. Click “**Apply**” button to confirm the configuration.
8. Click “**OK**” button to confirm & close the MSG instruction configuration.

### 13.6.3 SDSetFilePointerMSG Instruction Configuration

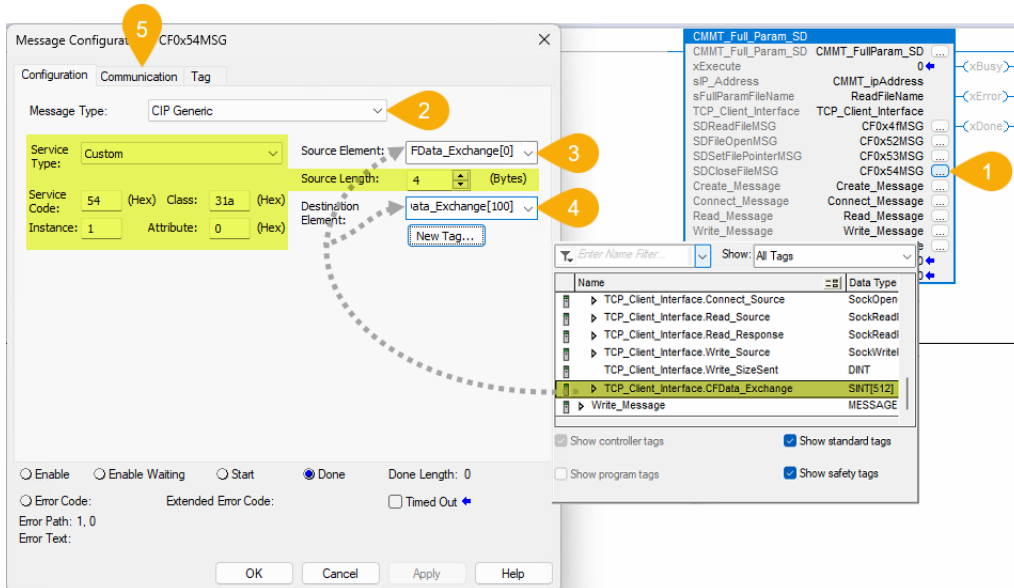


1. Click browse button of “SDSetFilePointerMSG” interface tag.
2. Set the yellow highlighted parameters as it is in above picture.
3. Click drop-down button of “Source Element” to browse Source tag. Browse the tag “TCP\_Client\_Interface.CFData\_Exchange[220]” from tag list & select it as Source tag.
4. Click drop-down button of “Destination Element” to browse destination tag. Browse the tag “TCP\_Client\_Interface.CFData\_Exchange[240]” from tag list & select it as Destination tag.
5. Click communication tab to assign destination device.

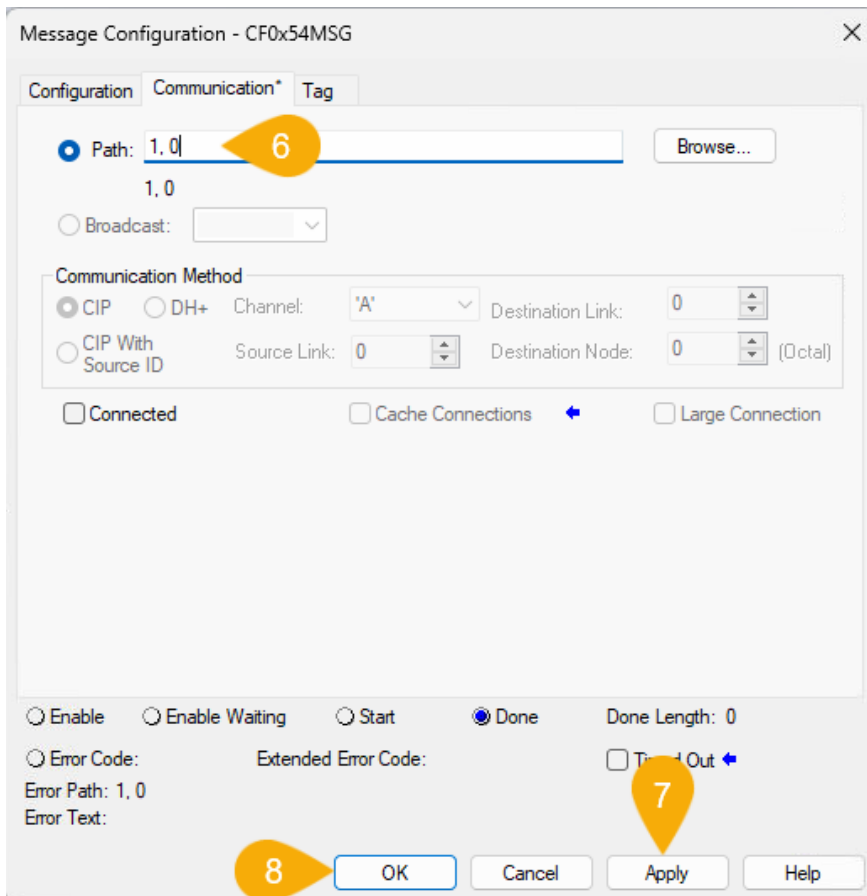


6. Enter “**1, slot number**” in path from communication tab. (Example: 1, 0 ([Refer Chapter 13.6.10 Communication Path](#)))
7. Click “**Apply**” button to confirm the configuration.
8. Click “**OK**” button to confirm & close the MSG instruction configuration.

### 13.6.4 SDCloseFileMSG Instruction Configuration

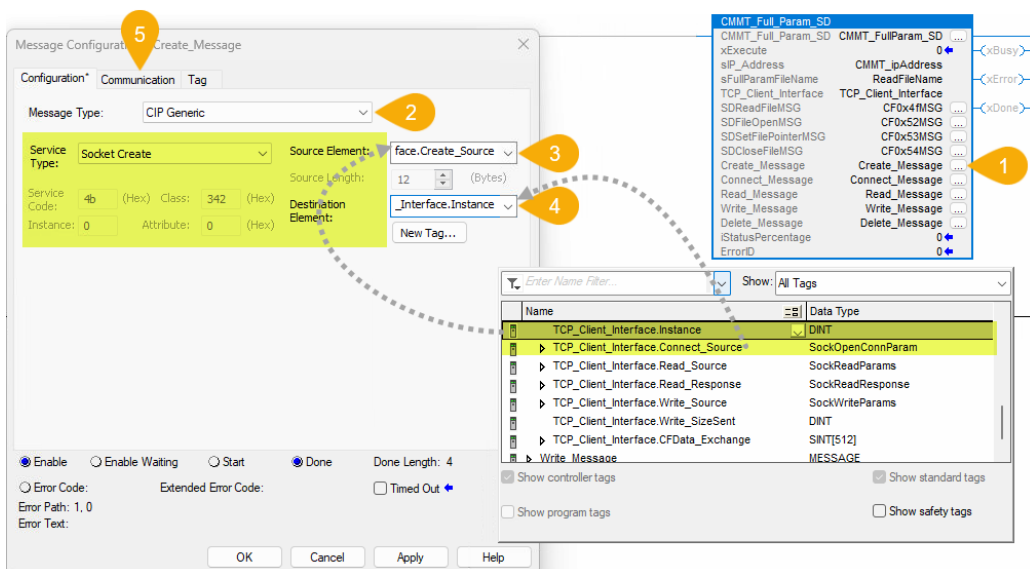


1. Click browse button of “**SDCloseFileMSG**” interface tag.
2. Set the yellow highlighted parameters as it is in above picture.
3. Click drop-down button of “**Source Element**” to browse Source tag. Browse the tag “**TCP\_Client\_Interface.CFData\_Exchange[0]**” from tag list & select it as Source tag.
4. Click drop-down button of “**Destination Element**” to browse destination tag. Browse the tag “**TCP\_Client\_Interface.CFData\_Exchange[100]**” from tag list & select it as Destination tag.
5. Click communication tab to assign destination device.



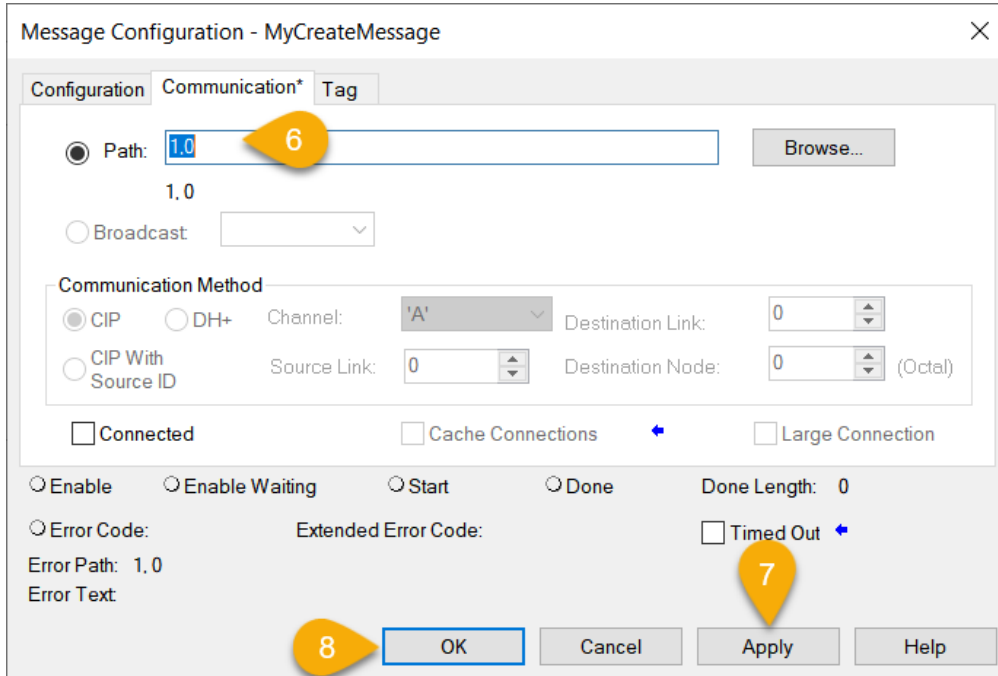
6. Enter “1, slot number” in path from communication tab. (Example: 1, 0 ([Refer Chapter 13.6.10 Communication Path](#)))
7. Click “Apply” button to confirm the configuration.
8. Click “OK” button to confirm & close the MSG instruction configuration.

### 13.6.5 Create\_Message Instruction Configuration



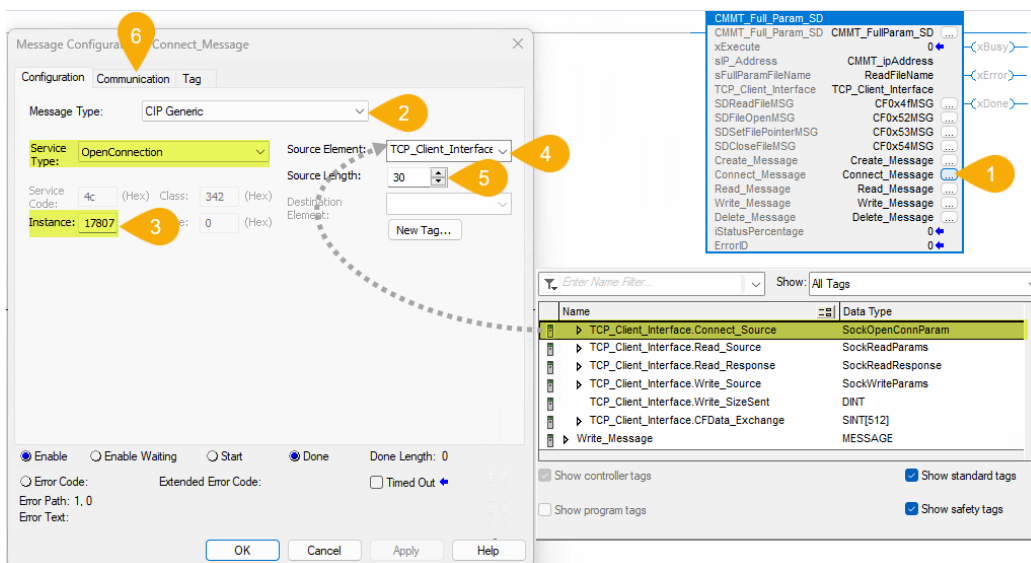
1. Click browse button of “MyCreateMessage” interface tag.
2. Set the yellow highlighted parameters as it is in above picture.

- Click drop-down button of “**Source**” to browse destination tag.  
Browse the tag “**MyClientInterface.Create\_Source**” from tag list & select it as Source tag.
- Click drop-down button of “**Destination Element**” to browse destination tag.  
Browse the tag “**MyClientInterface.Instance**” from tag list & select it as Destination tag.
- Click communication tab to assign destination device.



- Enter “**1, slot number**” in path from communication tab. (Example: 1, 0 ([Refer Chapter 13.6.10 Communication Path](#)))
- Click “**Apply**” button to confirm the configuration.
- Click “**OK**” button to confirm & close the MSG instruction configuration.

### 13.6.6 Connect\_Message Instruction Configuration



- Click browse button of “**MyConnectMessage**” interface tag.
- Set the yellow highlighted parameters as it is in above picture.

3. Enter random number in **"Instance"**. (Example: 17807)
4. Click drop-down button of **"Source"** to browse destination tag.  
Browse the tag **"MyClientInterface.Connect\_Source"** from tag list & select it as Source tag.
5. Enter **"30"** in Source Length.
6. Click communication tab to assign destination device.

7. Enter **"1, slot number"** in path from communication tab. (Example: 1, 0 ([Refer Chapter 13.6.10 Communication Path](#)))
8. Click **"Apply"** button to confirm the configuration.
9. Click **"OK"** button to confirm & close the MSG instruction configuration.

### 13.6.7 Read\_Message Instruction Configuration

1. Click browse button of “**MyReadMessage**” interface tag.
2. Set the yellow highlighted parameters as it is in above picture.
3. Enter random number in “**Instance**”. (Example: 17807)
4. Click drop-down button of “**Source**” to browse destination tag.  
Browse the tag “**MyClientInterface.Read\_Source**” from tag list & select it as Source tag.
5. Click drop-down button of “**Destination Element**” to browse destination tag.  
Browse the tag “**MyClientInterface.Read\_Response**” from tag list & select it as Destination tag.
6. Click communication tab to assign destination device.

Message Configuration - MyReadMessage

Configuration\* Communication\* Tag

☒ Path: 1.0 7 Browse...

1.0

☐ Broadcast ▼

Communication Method

☒ CIP ☐ DH+ Channel: 'A' ▼ Destination Link: 0 ▲▼

☐ CIP With Source ID Source Link: 0 ▲▼ Destination Node: 0 ▲▼ (Octal)

☐ Connected ☐ Cache Connections ☐ Large Connection

☐ Enable ☐ Enable Waiting ☐ Start ☐ Done Done Length: 0

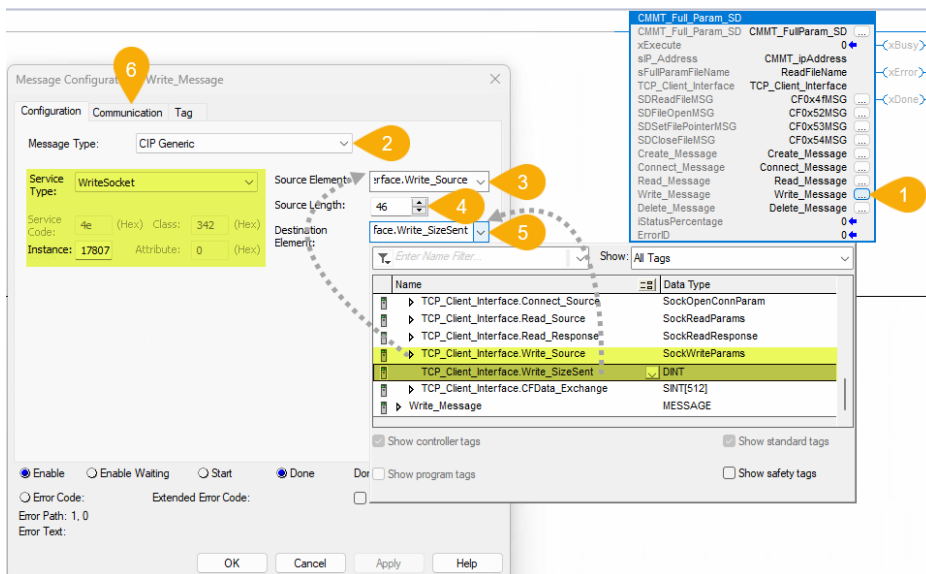
☐ Error Code: Extended Error Code: ☐ Timed Out ▲

Error Path:  
Error Text:

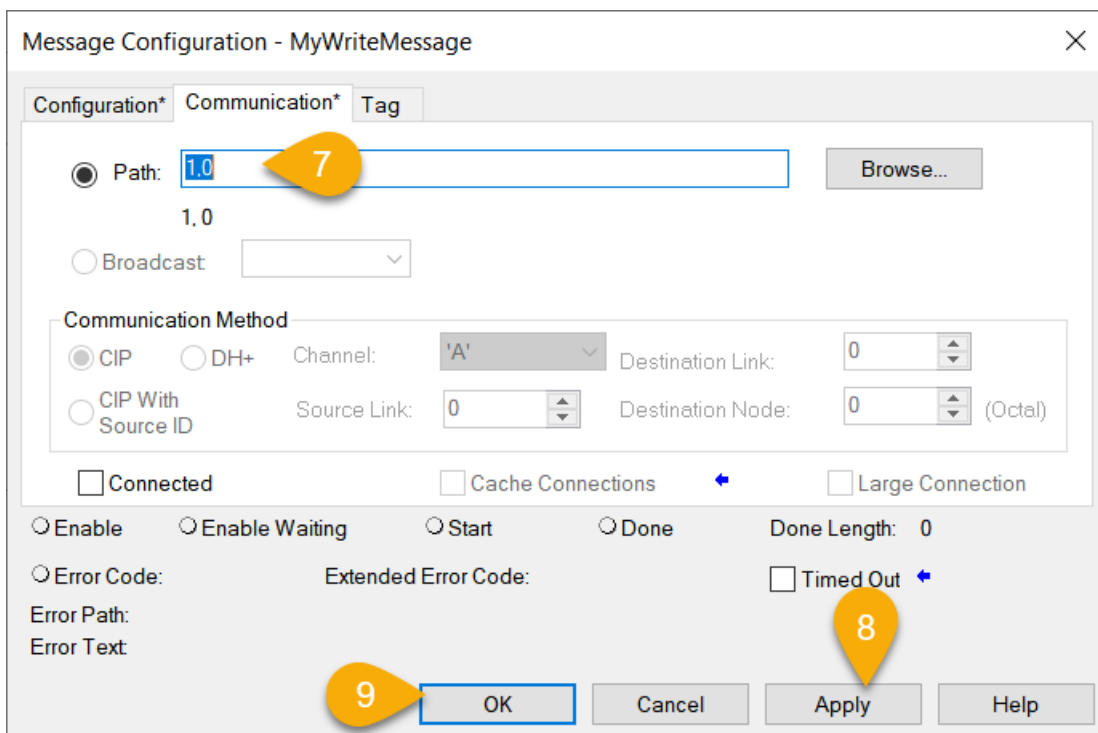
9 OK Cancel Apply Help

7. Enter “**1, slot number**” in path from communication tab. (Example: 1, 0 ([Refer Chapter 13.6.10 Communication Path](#)))
8. Click “**Apply**” button to confirm the configuration.
9. Click “**OK**” button to confirm & close the MSG instruction configuration.

### 13.6.8 Write\_Message Instruction Configuration



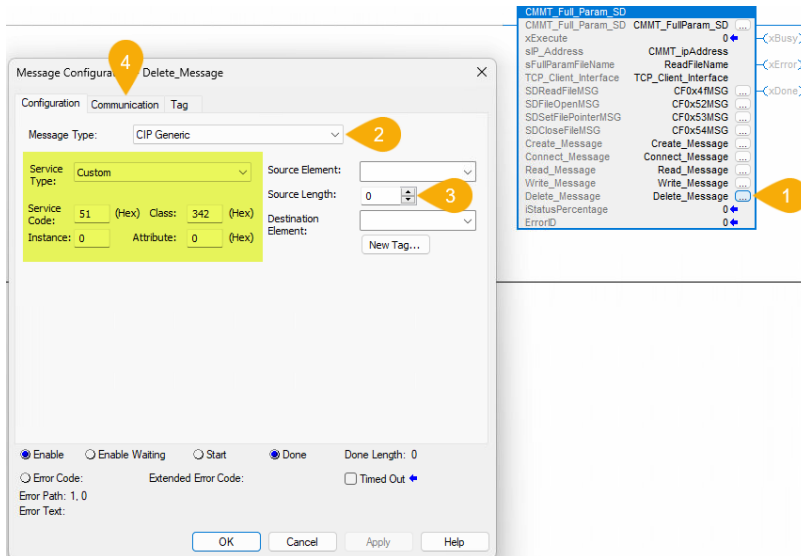
1. Click browse button of “**MyWriteMessage**” interface tag.
2. Set the yellow highlighted parameters as it is in above picture.
3. Click drop-down button of “**Source**” to browse destination tag.  
Browse the tag “**MyClientInterface.Write\_Source**” from tag list & select it as Source tag.
4. Enter “**46**” in Source Length.
5. Click drop-down button of “**Destination Element**” to browse destination tag.  
Browse the tag “**MyClientInterface.Write\_SizeSent**” from tag list & select it as Destination tag.
6. Click communication tab to assign destination device.



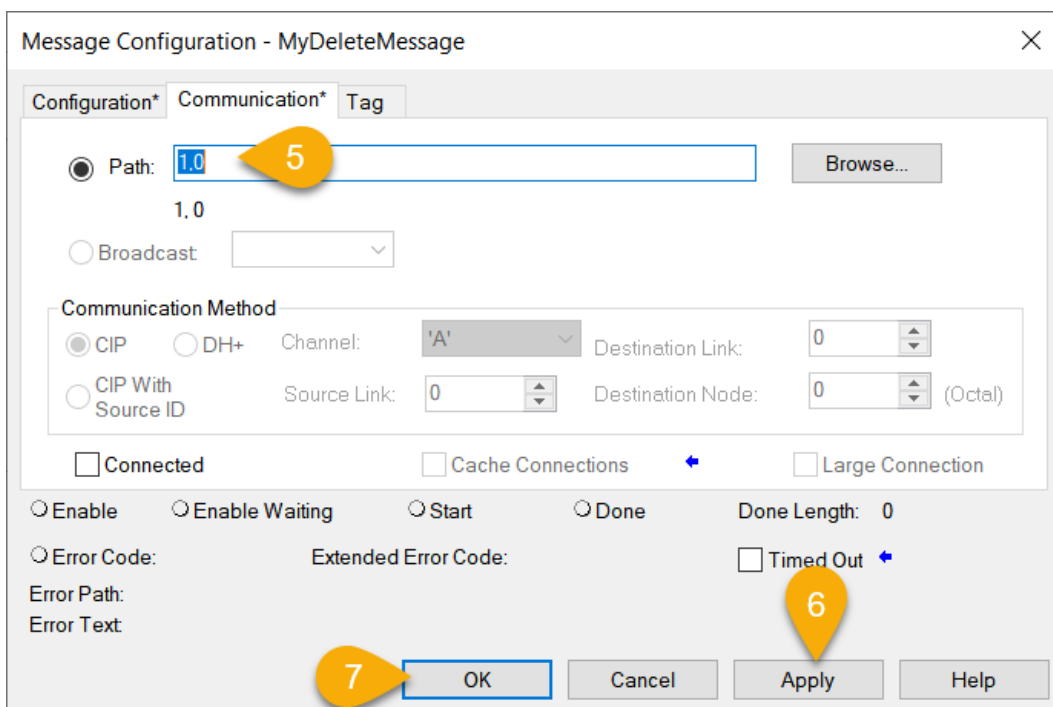
7. Enter “**1, slot number**” in path from communication tab. (Example: 1, 0 ([Refer Chapter 13.6.10 Communication Path](#)))

8. Click **“Apply”** button to confirm the configuration.
9. Click **“OK”** button to confirm & close the MSG instruction configuration.

### 13.6.9 Delete\_Message Instruction Configuration



1. Click browse button of **“MyDeleteMessage”** interface tag.
2. Set the yellow highlighted parameters as it is in above picture.
3. Enter **“0”** in Source Length.
4. Click communication tab to assign destination device.



5. Enter **“1, slot number”** in path from communication tab. (Example: 1, 0 ([Refer Chapter 13.6.10 Communication Path](#)))
6. Click **“Apply”** button to confirm the configuration.
7. Click **“OK”** button to confirm & close the MSG instruction configuration.

### 13.6.10 Communication Path



#### Note

There are hardware limitations with certain Allen Bradley controllers or EtherNet/IP masters. The AOI (Add-On Instruction) requires Socket Services which are ONLY available/supported with these Controllers/Modules listed here:

- ControlLogix® 5580 and GuardLogix® 5580 controllers
- CompactLogix™ 5380 and Compact GuardLogix 5380 controllers
- CompactLogix 5370 and Compact GuardLogix 5370 controllers
- CompactLogix 5480 controllers
- 1756-EN2x, 1756-EN3TR, and 1756-EN4TR EtherNet/IP communication modules
- 1756-EWEB and 1768-EWEB EtherNet/IP web server modules

#### 13.6.10.1 CompactLogix controllers

For CompactLogix controllers, the path is 1 and the slot is 0.

Message Configuration - MyCreateMessage

Configuration\* Communication\* Tag

Path: 1.0

Broadcast: [v]

Communication Method

☒ CIP ☐ DH+ Channel: 'A' Destination Link: 0

☐ CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)

☐ Connected ☐ Cache Connections ☐ Large Connection

☐ Enable ☐ Enable Waiting ☐ Start ☐ Done Done Length: 0

☐ Error Code: Extended Error Code: ☐ Timed Out

Error Path: Error Text:

OK Cancel Apply Help

OR

For CompactLogix controllers with version 28.11 or later, you can also use a path of 'THIS'.

Message Configuration - MyCreateMessage

Configuration Communication Tag

Path: THIS

Broadcast: [v]

Communication Method

☒ CIP ☐ DH+ Channel: 'A' Destination Link: 0

☐ CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)

☐ Connected ☐ Cache Connections ☐ Large Connection

☐ Enable ☐ Enable Waiting ☐ Start ☐ Done Done Length: 0

☐ Error Code: Extended Error Code: ☐ Timed Out

Error Path: THIS Error Text:

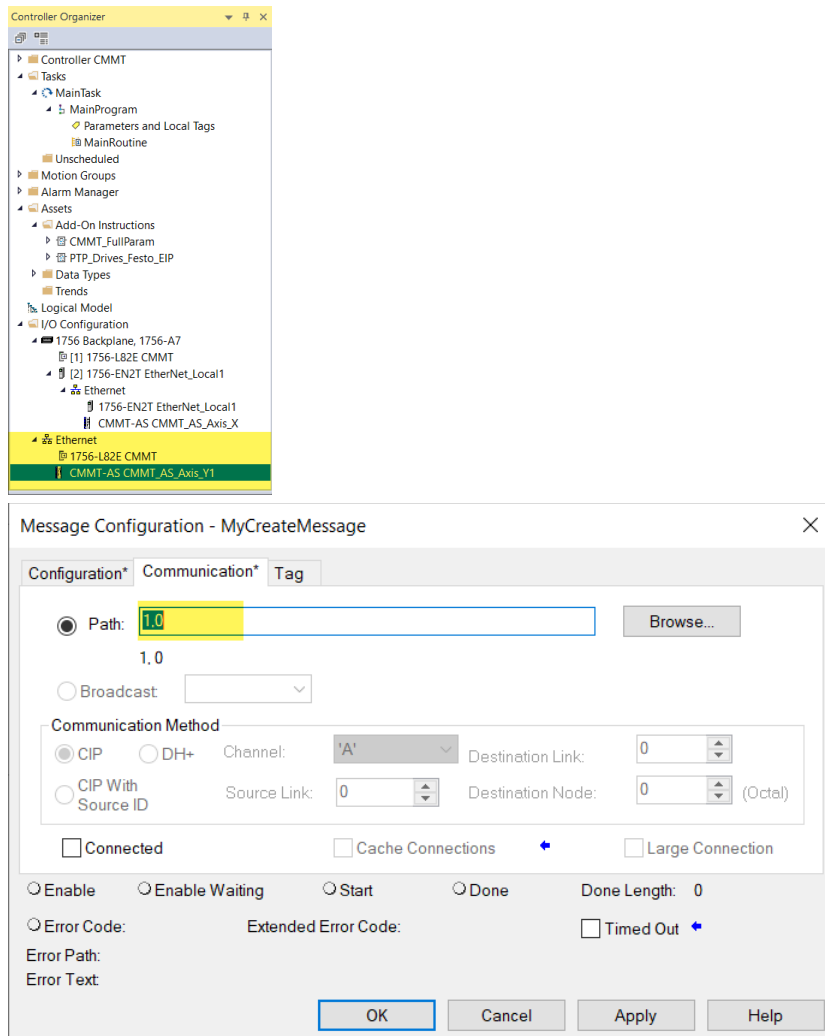
OK Cancel Apply Help

### 13.6.10.2 ControlLogix controllers

For ControlLogix 5580 controllers and communication modules, the path is 1, x. Where x is the slot number of the communication module, or the slot number of a ControlLogix 5580 controller if the front Ethernet port is used.

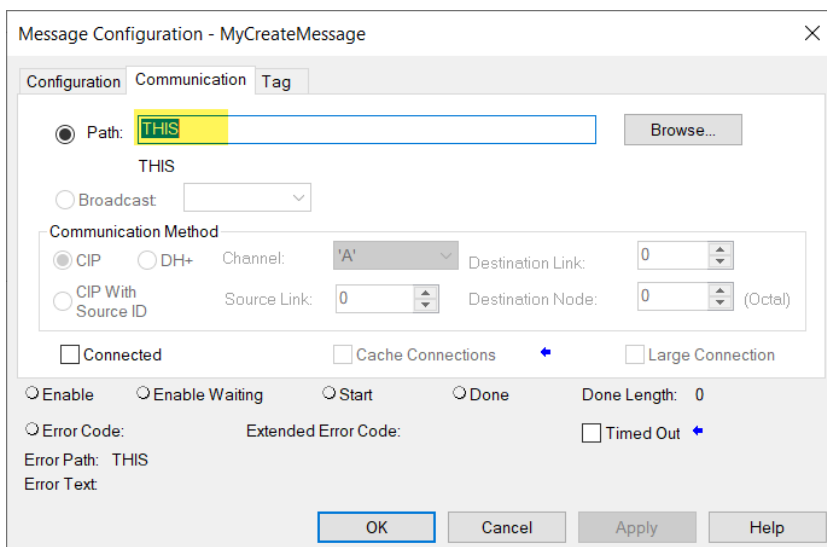
#### Method – 1

If CMMT module added under Controller Ethernet Port then, follow Method – 1



OR

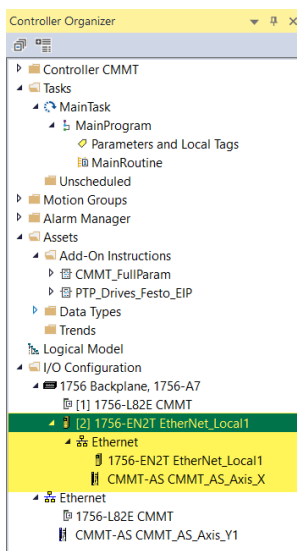
For ControlLogix 5580 controllers with version 28.11 or later, you can also use a path of 'THIS'.



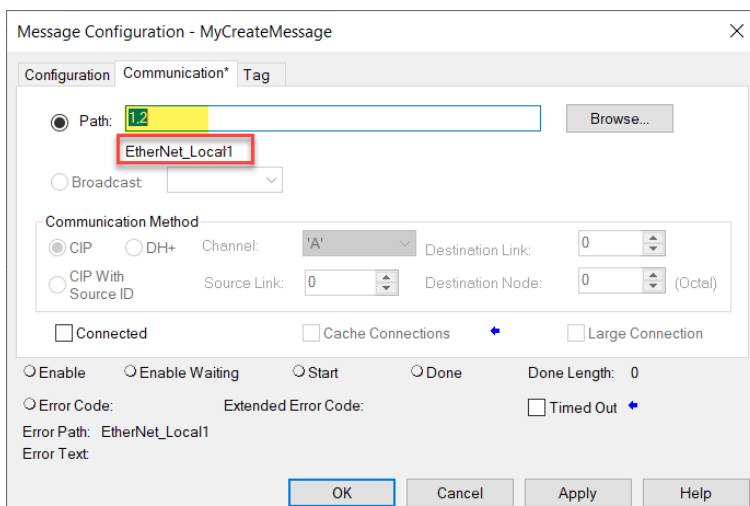
The dialog box 'Message Configuration - MyCreateMessage' has three tabs: Configuration, Communication, and Tag. The Configuration tab is active. It features a 'Path' field with a dropdown menu showing 'THIS' and a 'Browse...' button. Below this is a 'Broadcast' dropdown. The 'Communication Method' section includes radio buttons for 'CIP' (selected), 'DH+', and 'CIP With Source ID'. It also has 'Channel' (set to 'A'), 'Destination Link' (0), 'Source Link' (0), and 'Destination Node' (0) fields. Checkboxes for 'Connected', 'Cache Connections', and 'Large Connection' are present. At the bottom, there are radio buttons for 'Enable', 'Enable Waiting', 'Start', and 'Done', along with a 'Done Length' field set to 0. There are also fields for 'Error Code', 'Extended Error Code', and a 'Timed Out' checkbox. The 'Error Path' is set to 'THIS' and 'Error Text' is empty. Buttons for 'OK', 'Cancel', 'Apply', and 'Help' are at the bottom right.

## Method – 2

If CMMT module added under Ethernet Module then, follow Method – 2

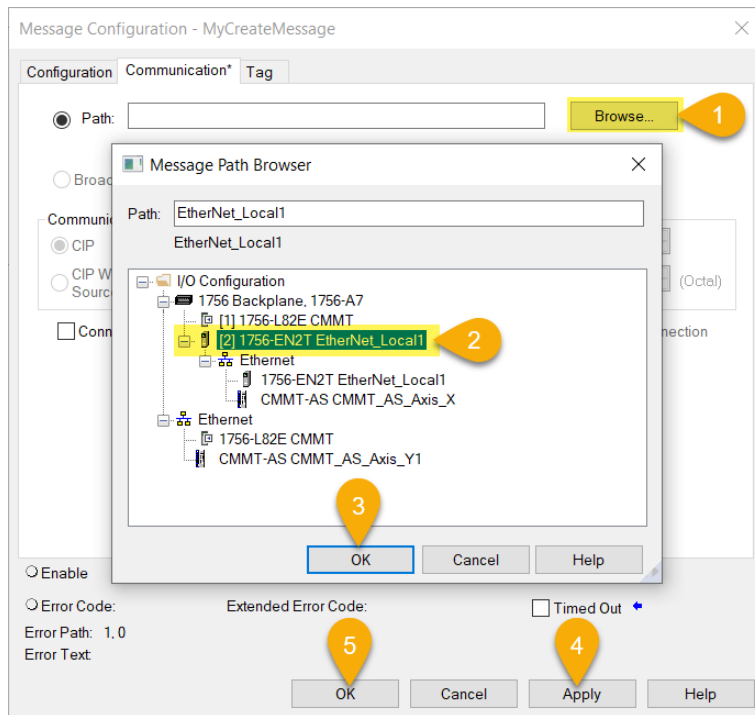


The path is 1, the slot number of the communication module is 2



The 'Message Configuration - MyCreateMessage' dialog box is shown with the 'Communication' tab selected. The 'Path' field now shows '1:2' and a dropdown menu with 'EtherNet\_Local1' selected and highlighted with a red box. The 'Communication Method' section remains the same, with 'CIP' selected. The 'Error Path' is now set to 'EtherNet\_Local1'. The 'OK' button is highlighted.

Or



1. Click on Browse... button
2. Select the Ethernet Module (1756-EN2T)
3. Click **“OK”** button to confirm
4. Click **“Apply”** button to confirm the configuration.
5. Click **“OK”** button to confirm.

## 13.7 CMMT\_FullParam AOI Tag Details

### 13.7.1 CMMT\_Full\_Param\_SD In-Out Variables

| In-Out Variables    | Data type | Description                      |
|---------------------|-----------|----------------------------------|
| SDReadFileMSG       | MESSAGE   | Read File MSG Instruction        |
| SDFileOpenMSG       | MESSAGE   | File Open MSG Instruction        |
| SDSetFilePointerMSG | MESSAGE   | Set Pointer MSG Instruction      |
| SDCloseFileMSG      | MESSAGE   | Close File MSG Instruction       |
| Create_Message      | MESSAGE   | Socket Create MSG instruction    |
| Connect_Message     | MESSAGE   | Open Connection MSG instruction  |
| Read_Message        | MESSAGE   | Parameter Read MSG instruction   |
| Write_Message       | MESSAGE   | Parameter Write MSG instruction  |
| Delete_Messafe      | MESSAGE   | Parameter Delete MSG instruction |

### 13.7.2 CMMT\_Full\_Param\_SD Input Variables

| Input Variable     | Data Type | Description                                               |
|--------------------|-----------|-----------------------------------------------------------|
| xExecute           | BOOL      | 1 = Start parameter read and write operation.             |
| sIP_Address        | STRING    | CMMT IP Address of Port X18 for writing parameters.       |
| sFullParamFileName | STRING    | Full file name for reading, including the .l5x extension. |

### 13.7.3 CMMT\_Full\_Param\_SD Output Variables

| Output Variable   | Data Type | Description                                            |
|-------------------|-----------|--------------------------------------------------------|
| ErrorID           | INT       | Error message description and Message.                 |
| iStatusPercentage | INT       | Current Status in Percentage.                          |
| xDone             | BOOL      | Parameterization processed successfully.               |
| xBusy             | BOOL      | Parameterization function under process.               |
| xError            | BOOL      | Parameterization function completed with unsuccessful. |

## 13.8 Diagnostics

The following table shows the possible error number and the associated error description

### 13.8.1 ErrorID

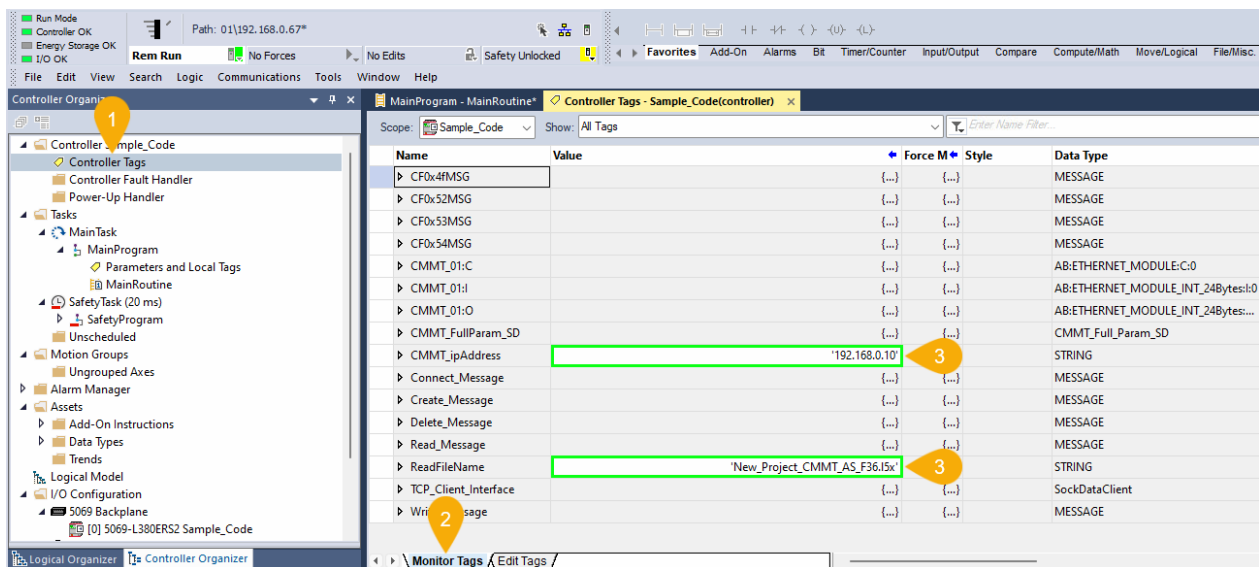
The "ErrorID" is used for internal and message errors of the CMMT\_Full\_Param\_SD AOI.

| ErrorID<br>(Dec) | ErrorID<br>(Hex) | ErrorID Description                                   |
|------------------|------------------|-------------------------------------------------------|
| 3                | 0x3              | Error in Delete_Message                               |
| 4                | 0x4              | Error in Create_Message                               |
| 5                | 0x5              | Error in Connect_Message                              |
| 6                | 0x6              | size Login is 0                                       |
| 7                | 0x7              | ACK not OK                                            |
| 8                | 0x8              | size_openfile is 0                                    |
| 9                | 0x9              | Error in Read_Message                                 |
| 10               | 0xA              | frame counter out of range                            |
| 11               | 0xB              | IndexPck out of range                                 |
| 12               | 0xC              | size writefile is 0                                   |
| 13               | 0xD              | Error in Write_Message                                |
| 14               | 0xE              | something went wrong with the PCK states              |
| 15               | 0xF              | received data size invalid                            |
| 16               | 0x10             | size_closefile is 0                                   |
| 17               | 0x11             | size_unpackfile is 0                                  |
| -12              | 0x0C             | Incorrect File or something is wrong                  |
| 92               | 0x5C             | SD card is not present in the controller.             |
| 1                | 0x1              | Connection failure (extended error codes)             |
| 2                | 0x2              | Insufficient resource                                 |
| 18               | 0x12             | Fragment primitive                                    |
| 19               | 0x13             | Insufficient command data                             |
| 20               | 0x14             | Attribute not supported                               |
| 21               | 0x15             | Too much data                                         |
| 26               | 0x1A             | Bridge request too large                              |
| 27               | 0x1B             | Bridge response too large                             |
| 28               | 0x1C             | Attribute list shortage                               |
| 29               | 0x1D             | Invalid attribute list                                |
| 30               | 0x1E             | Embedded service error                                |
| 31               | 0x1F             | Connection related failure (see extended error codes) |
| 32               | 0x20             | Invalid Parameter                                     |

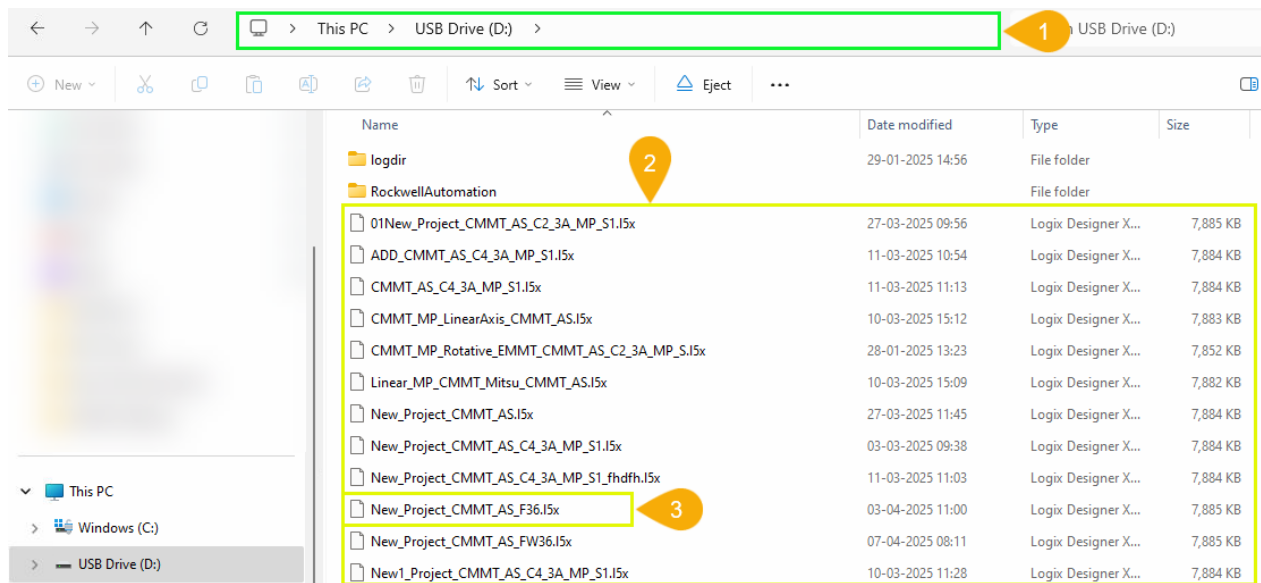
| ErrorID (Dec) | ErrorID (Hex) | ErrorID Description                      |
|---------------|---------------|------------------------------------------|
| 34            | 0x22          | Invalid reply received                   |
| 37            | 0x25          | Key segment error                        |
| 38            | 0x26          | Invalid IOI error                        |
| 39            | 0x27          | Unexpected attribute in list             |
| 40            | 0x28          | DeviceNet error - invalid member ID      |
| 41            | 0x29          | DeviceNet error - member not settable    |
| 209           | 0xD1          | Module not in run state                  |
| 251           | 0xFB          | Message port not supported               |
| 252           | 0xFC          | Message unsupported data type            |
| 253           | 0xFD          | Message uninitialized                    |
| 254           | 0xFE          | Message timeout                          |
| 255           | 0xFF          | General error (see extended error codes) |

### 13.9 Sample Code For CMMT\_Full\_Param\_SD AOI

#### Step-1

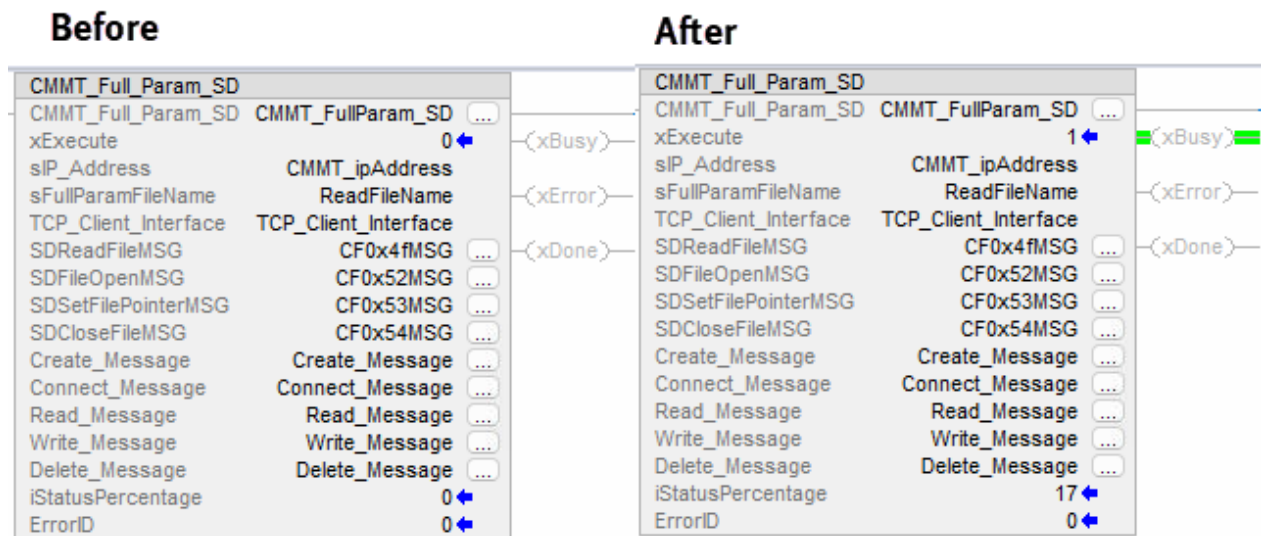


1. Double-click on the **Controller Tags** in the project tree bar.
2. Click on the **Monitor Tags** option to start monitoring control tags.
3. Enter the **CMMT X18 port IP address** in the **CMMT\_ipAddress** field.
4. Enter the full file name from which the user wants to read data in the **ReadFileName** field. The image in Step 2 shows the reading file details.

**Step-2**

1. The directory path for the Industrial SD Card supports the Allen-Bradley Controller.
2. Available .I5x files on the SD Card are shown above.
3. Copy the name of the file and paste it into the **ReadFileName** tag field.

### 13.9.1 Reading the .l5x File and Writing Data to the CMMT Drive



- Toggle the **xExecute** bit to trigger the parameterization. While the parameterization is active, the **xBusy** status will be true. If the entire parameterization completes successfully, the **xDone** bit will be high. If there is any error, the **xError** bit will be high, and the error description will be in the **ErrorID** tag. Additionally, the **iPercentageStatus** tag will show the current status of reading and writing data to CMMT drive.
- After successful execution, the output **xDone** will turn ON, and if the **iPercentageStatus** is 100%, the CMMT-xx will reboot automatically.

