Integration Tutorial SE02

Schneider Electric Modicon M580 and HART for Water & Wastewater Industry

Table of Contents

1 Document Information

1.1 Purpose and Scope

This document provides a step by step description on how to integrate HART devices with the Schneider Electric Modicon M580 system. All content of this document is jointly developed, reviewed and approved by Schneider Electric and Endress+Hauser as a common deliverable of Open Integration.

1.2 Document History

This is version 1.00.00 of this document. Version history:

Version	Released	Description
1.00.00	2017-10	Initial version

1.3 Related Documents

Please refer to related documents as listed below:

Document	Description
SD01936S/04/EN/01.17	Reference Topology SE02
SD01938S/04/EN/01.17	Integration Test Summary SE02
SD01939S/04/EN/01.17	List of Tested Devices and Versions SE02

2 Pre-Requisites

Readers of this document should be familiar with related documents as listed in chapter 1.3 and basics on how to work with the Schneider Electric Modicon M580 System and HART in general. Please refer to recommended literature as listed in chapter 2.1.

2.1 Recommended Literature

2.1.1 Schneider Electric

Document	Description
EIO0000001854.02	Modicon M580 Remote I/O Modules (Hardware)
35006238.13	Modicon M580 and Premium/Atrium using Unity Pro
EAV16400.01	Modicon eX80 BME AHI 0812 HART Analog Input Module & BME AHO 0412 HART Analog Output Module User Guide

2.1.2 Endress+Hauser

Document	Description
BA00065S	FieldCare Project Tutorial

2.2 Operable Control System

This document assumes an operable Schneider Electric Modicon M580 System as defined by Reference Topology SE02. Please refer to the manuals listed in chapter 2.1.1 for an explanation on how to use hard- and software provided by Schneider Electric.

2.3 Operable Asset Management System

This document assumes an operable Endress+Hauser PAM System as defined by Reference Topology SE02. Please refer to manuals listed in chapter 2.1.2 for installing of software provided by Endress+Hauser.

2.4 Operable Field Devices

This document assumes an operable selection of Endress+Hauser HART devices, as defined by Reference Topology SE02. Each field device is powered if needed and adequately connected to the Schneider Electric M580 System. If required, please refer to individual device manuals for further advice.

3 Basic Integration

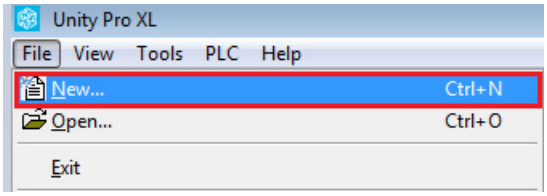
This chapter describes the main workflow for integration of HART devices into the Schneider Electric M580 System by means of Universal Commands. As a result, the 4-20 mA/HART communication is running. HART process values and status information is available within the control strategy of the system for further processing.

3.1 System Configuration

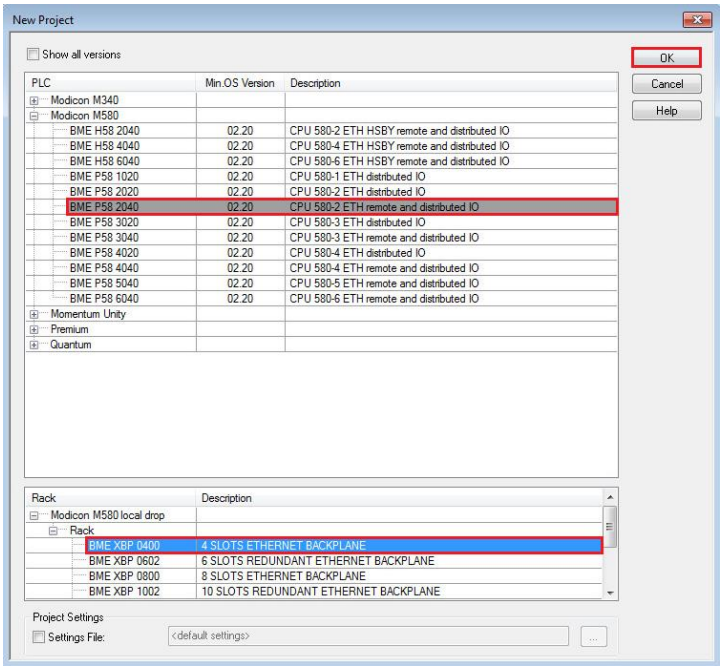
3.1.1 New Project

- Start the software Unity Pro XL:

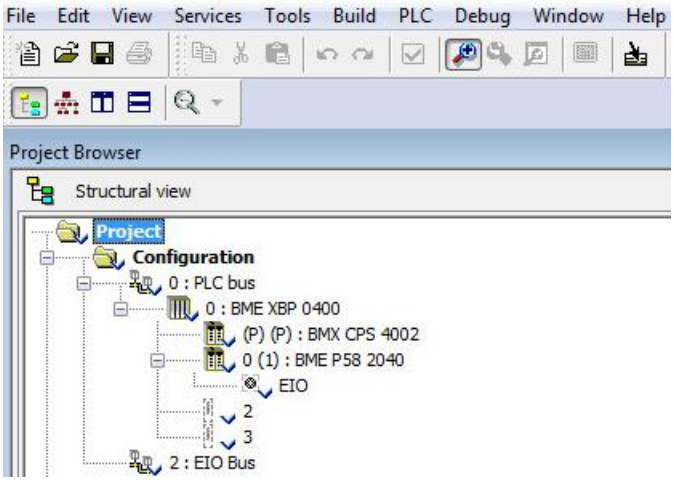
- Create a new project with the menu "File→ New…":



- Select the PLC and Rack type. Click on the button "OK".
  In this example, the PLC BMEP582040 is mounted on a Rack BMEXBP0400:



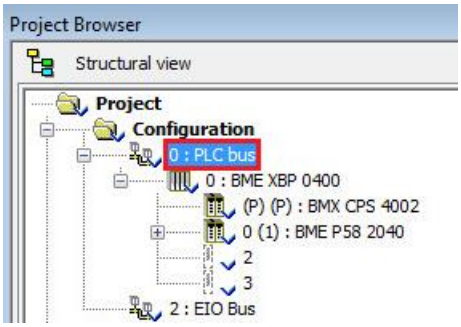
- Created Project structure:



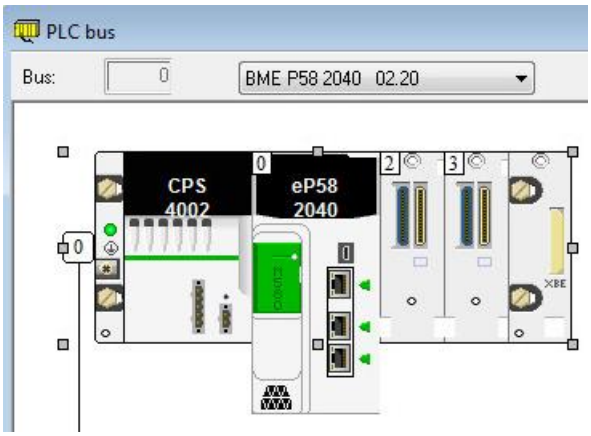
3.1.2 Hardware Configuration

3.1.2.1 M580 PLC

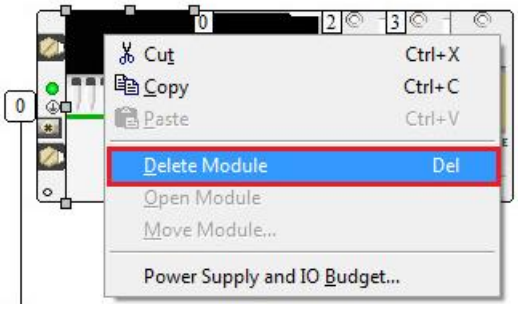
- Double-click on the field "0: PLC bus" in the Project Browser view:



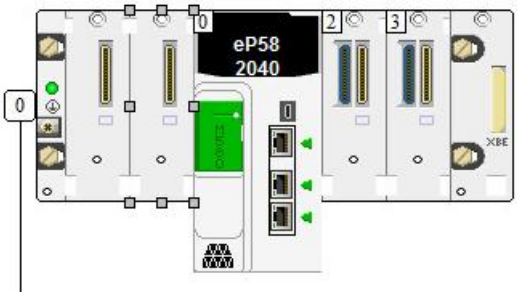
- The PLC module is inserted automatically with the power supply CPS4002:



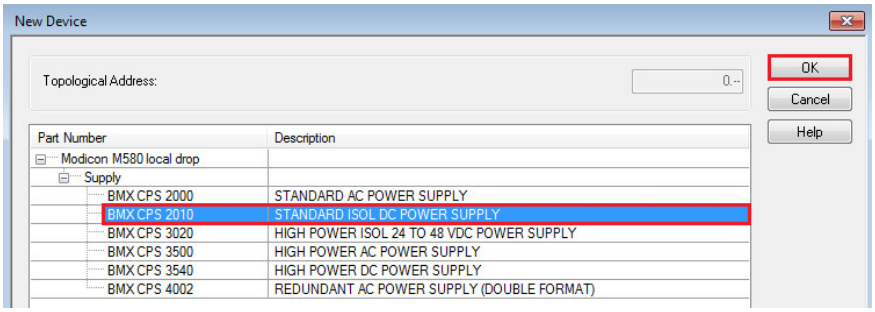
- In our example, the used Power Supply is the CPS2010 module. Delete the current one by right-clicking on the symbol CPS2000 and select the menu “Delete Module”:



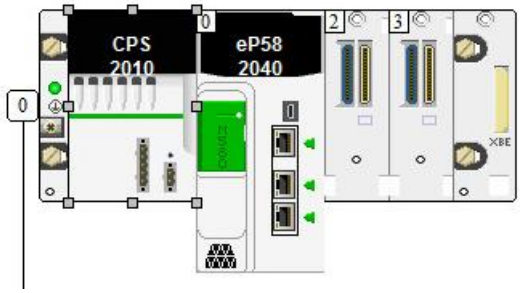
- The power supply module is now deleted:



- Double-click on the empty module and select the correct power supply module. In this case, it is the module BMXCPS2010. Click on the button “OK“:



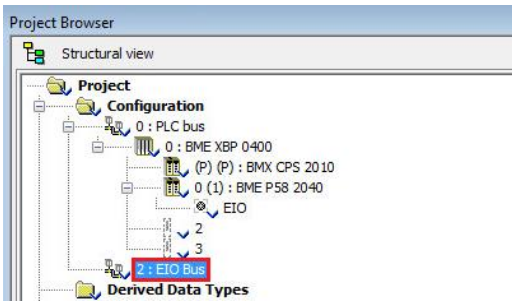
- M580 PLC rack:



3.1.2.2 X80 Remote IO

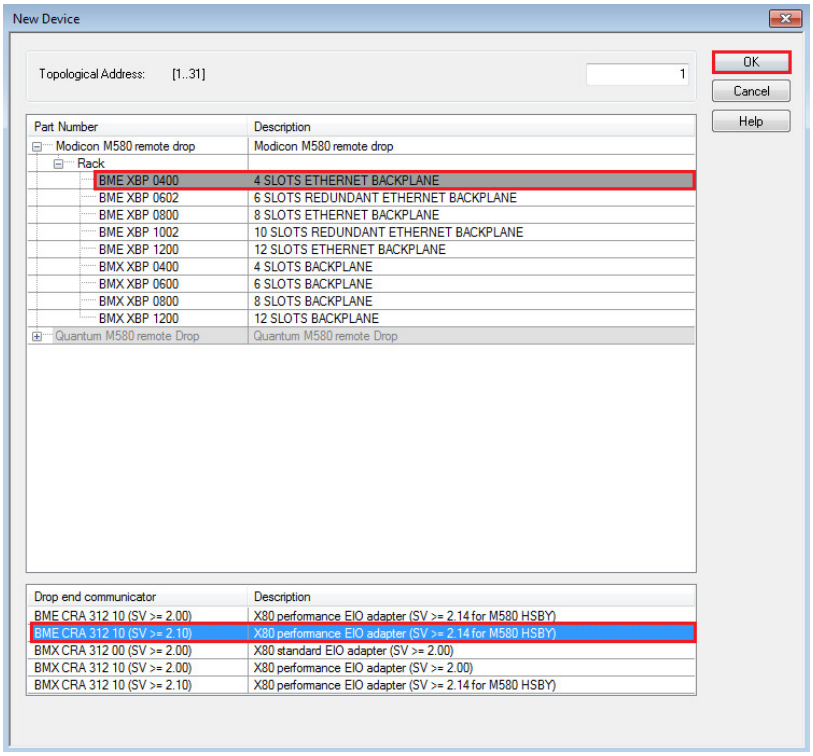
3.1.2.2.1 EIO Adapter CRA31210

- Double-click on the field "2: EIO bus" in the Project Browser view:



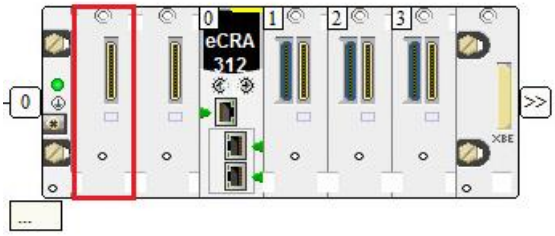
- This opens the following window. Double-click on the white square:

- Select the correct backplane and X80 EIO Adapter and click on the button "OK".
  In this example, the BMEXBP0400 backplane and the X80 EIO adapter BMECRA31210 (SV>=2.10)" are selected:

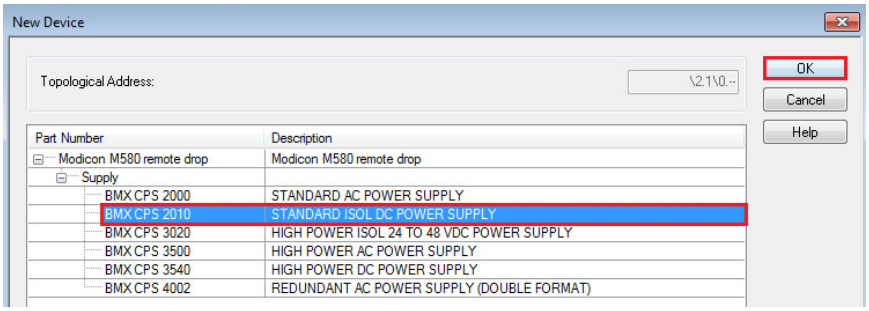


- Inserted backplane and EIO Drop adapter:

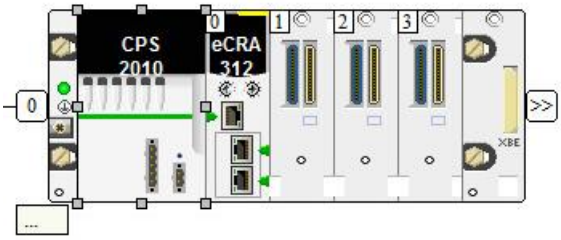
- Double-click on the first empty slot to insert the power supply:



- Select the power supply module BMXCPS2010 and click on the button “OK”:



- Inserted Power Supply module:



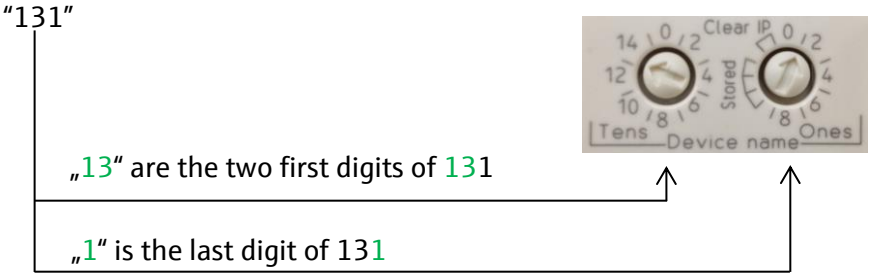
EIO adapter CRA31210 Device Name

- The device name of the EIO adapter BMECRA31210 must be configured on the card itself thanks to the rotary switches as well as in Unity Pro in the card parameters.

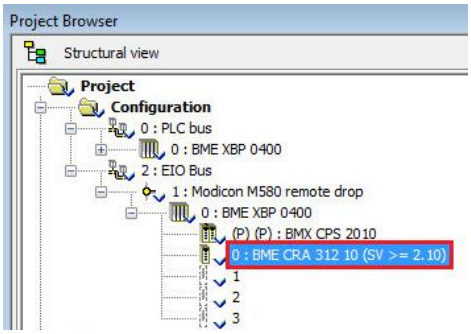
  In this example, the address is set to **1** on the BMECRA31210 EIO adapter thanks to the rotary switches. This corresponds to the device name "BMECRA_**001**":

  Both rotary switches allow the user to set addresses from **0** to **159**, which correspond to device name BMECRA_**000** to BMECRA_**159**.

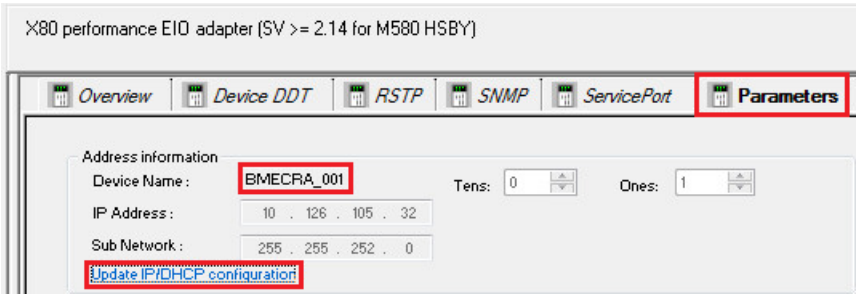
- Rotary switches configuration example for device name "BMECRA_131**"**:



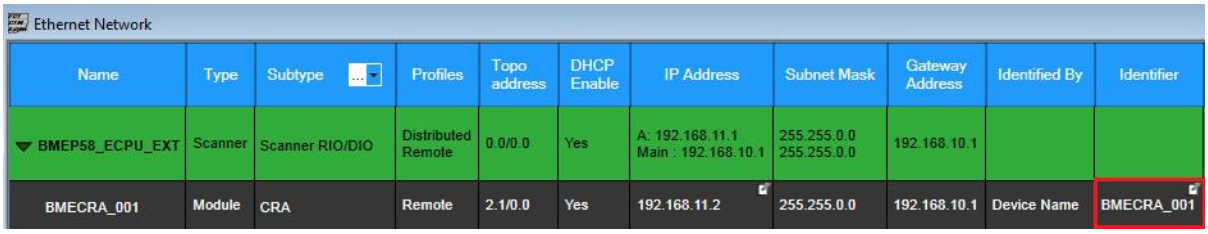
- In the Unity Pro Project Browser, double-click on the field "0 : BMECRA31210 (SV>=2.10)" to check the device name:



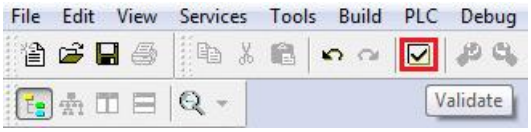
- In this example, the device name is BMECRA_001. Click on the link "Update IP/DHCP configuration" for changing the device name if needed:



- If needed, change **the 3 digits** of the device name Identifier. In this example, EIO adapter device name is **BMECRA_001**:



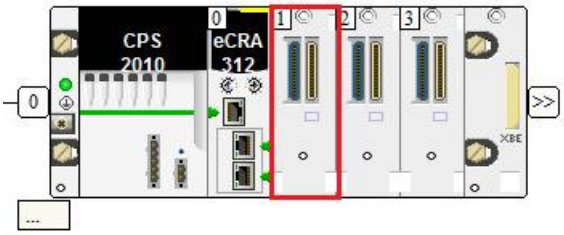
- Save the configuration by clicking on the symbol "Validate" in the tool bar:



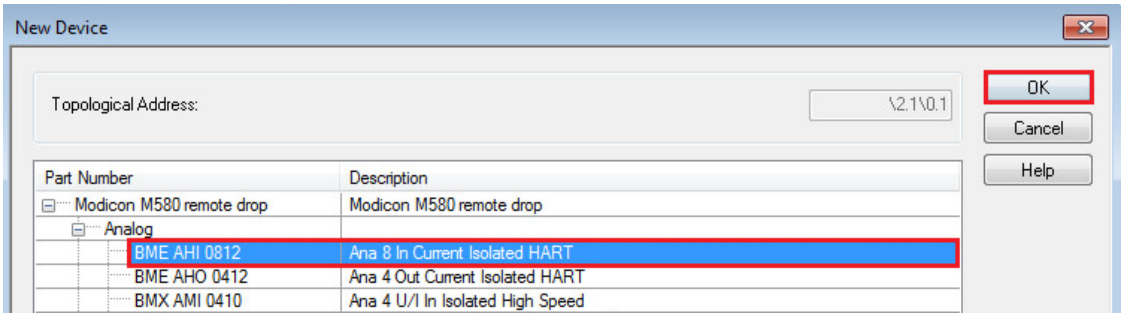
→For the next steps, the EIO adapter device name is **BMECRA_001**.

3.1.2.2.2 HART Analog Input Module AHI0812

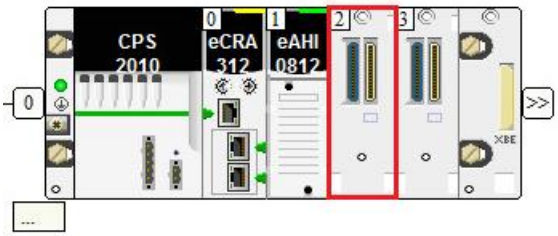
- Double-click on Slot 1 to insert a HART analog input module for this example.



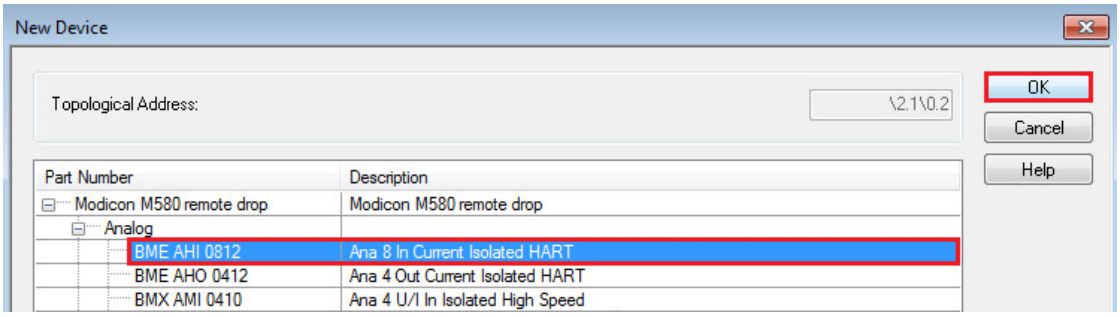
- Select the HART analog input module BMEAHI0812 and click on the button “OK”:



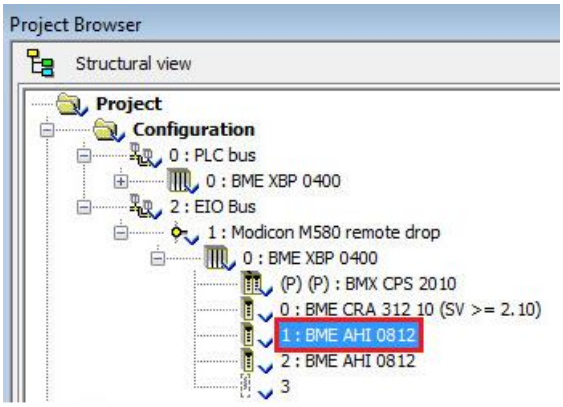
- Double-click on Slot 2 to insert another HART analog input module for this example:



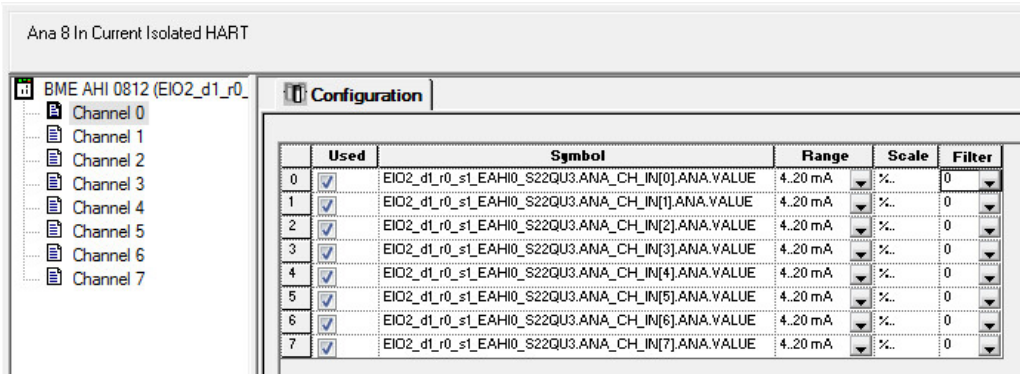
- Select the HART analog input module BMEAHI0812 and click on the button “OK”:



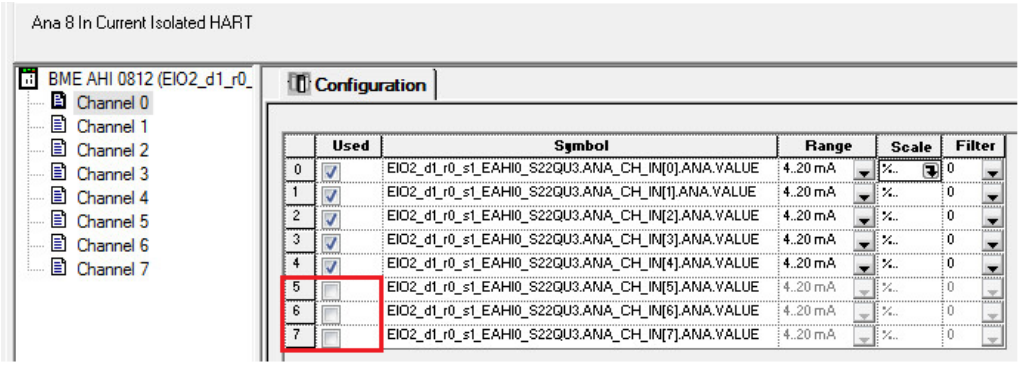
- Double-click on the analog input module AHI0812:



- This displays the 8 configured analog inputs channels:

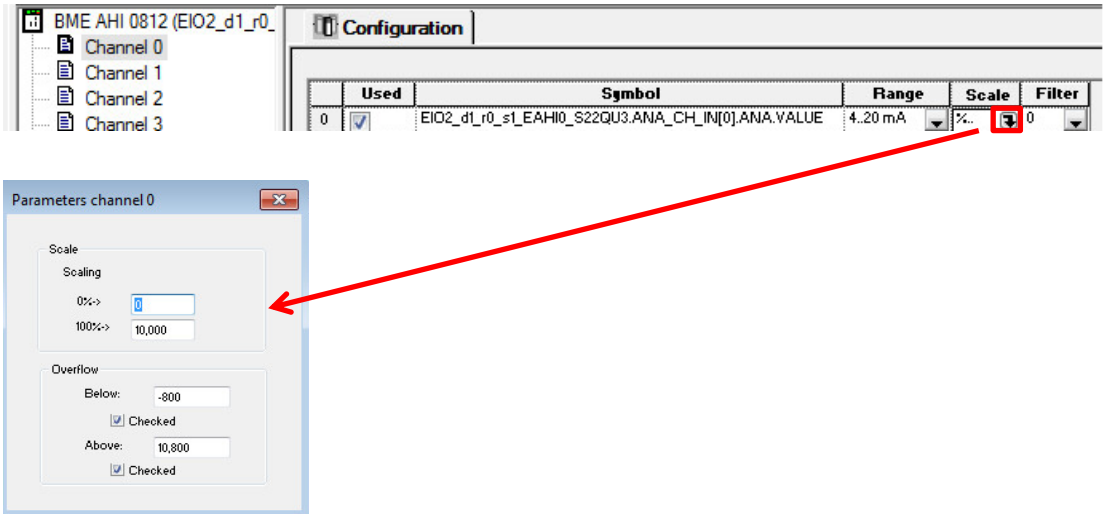


- Channels can be activated or deactivated by clicking in the corresponding channel checkbox. Channels 5, 6 and 7 have been deactivated in this example:



- Default settings are used in this example. The “Scale” configuration is 4.20mA from 0 to 10000 (0% to 100%) and the “Filter” option is set to 0 (No Filtering).

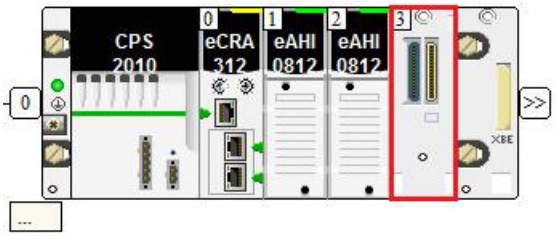
- Scaling ranges can be changed by clicking on the arrow in the “Scale” column:



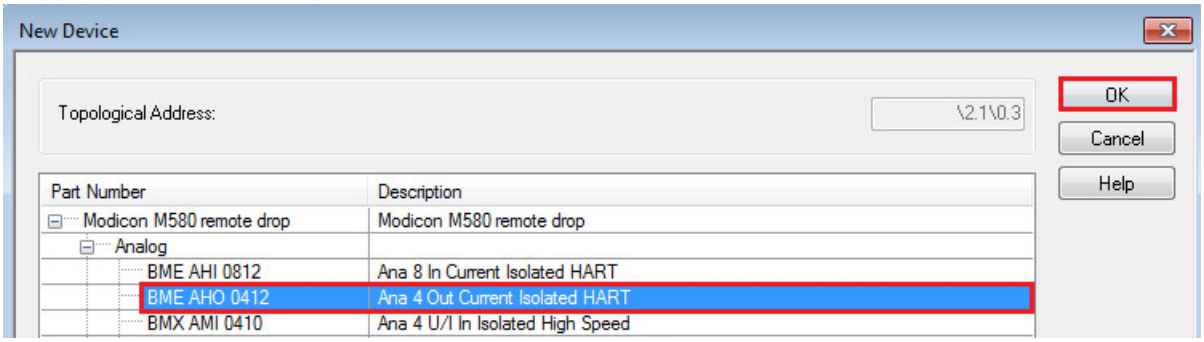
- Please refer to the user manual of the AHI0812 card for further information.

3.1.2.2.3 HART Analog Output Module AHO0412

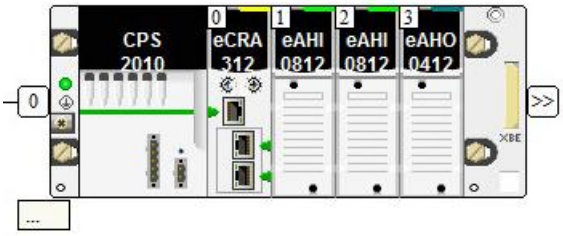
- Double-click on Slot 3 to insert a HART analog output module for this example:



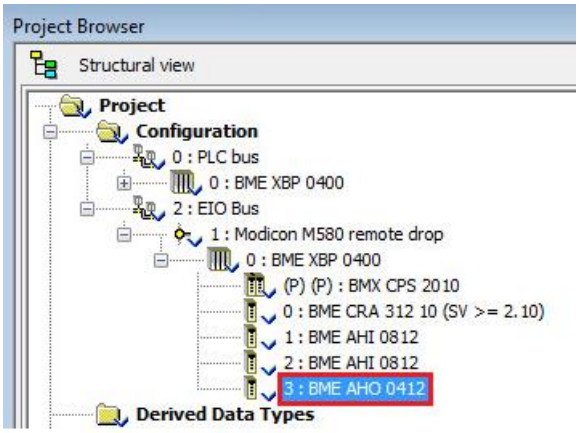
- Select the HART analog output module BMEAHO0412 and click on the button “OK”:



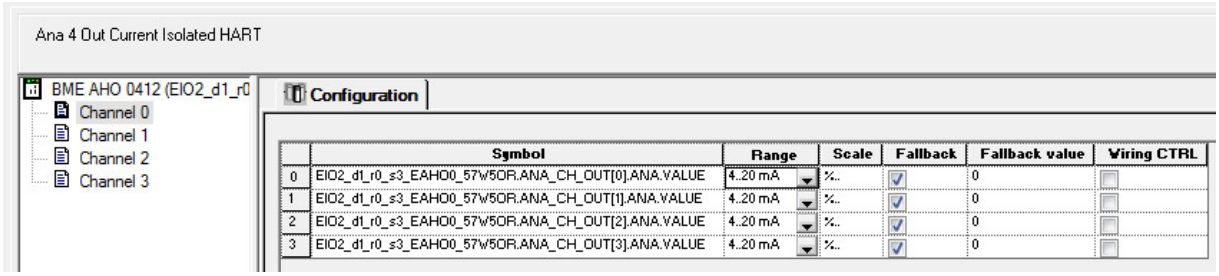
- Inserted module for this example:



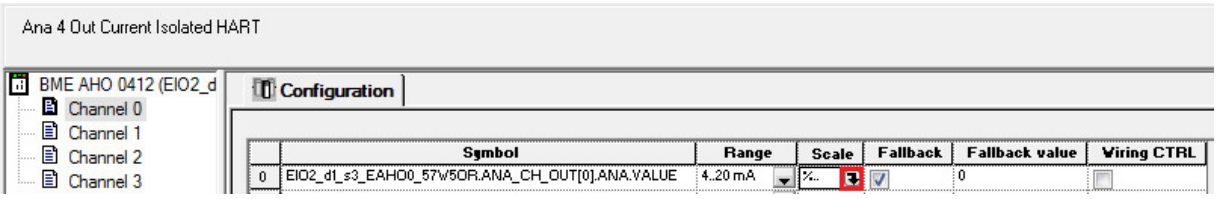
- Double-click on the analog output module AHO0412:



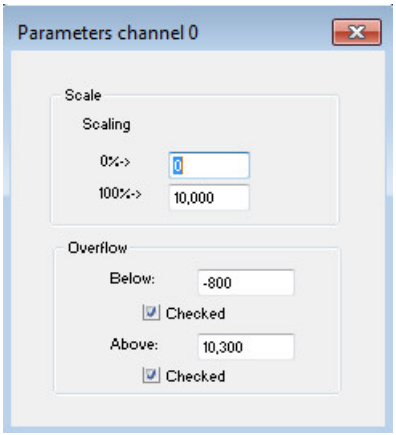
- This displays the 4 configured analog outputs channels:



- Click on the arrow in the "Scale" column:



- This displays the default scaling settings (0 to 10000):



- Other settings in this example:
  - "Fallback" and "Fallback value" are used: that means the actuator receives the "Fallback value" when the PLC is in STOP. If the "Fallback" option is not selected, the actuator maintains its position.
  - "Wiring CTRL" option: this option checks for a broken wire.

- Please refer to the AHO0412 user manual for further information.

3.1.2.3 HART Device Connection

- The following table displays how the Endress+Hauser devices of the SE02 Topology are connected on the Schneider Electric environment:

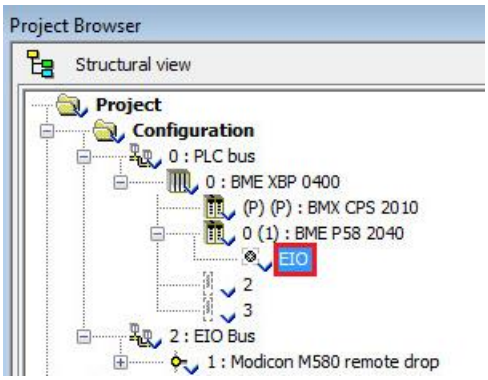
Device	Power Supply	AHI0812 Card 1	ABE-7CPA03 Block 1 Terminals			AHI0812 Card 2	ABE-7CPA03 Block 2 Terminals			AHO0412	ABE-7CPA21 Block 1 Terminals		
Promag10	External	A0	+	IC0	202								
			-	0V0	203								
Promag50	External	A1	+	IC1	102								
			-	0V1	103								
Cerabar M	Loop	A2	+	IS2	204								
			-	IC2	206								
Prosonic S	External	A3	+	IC3	106								
			-	0V3	107								
Liquiline CM442	External	A4	+	IC4	210								
			-	0V4	211								
Promag 400	External					A0	+	IC0	202				
							-	0V0	203				
iTEMP TMT82	Loop					A1	+	IS1	100				
							-	IC1	102				
Cerabar S	Loop					A2	+	IS2	204				
							-	IC2	206				
Prosonic M	Loop					A3	+	IS3	104				
							-	IC3	106				
Micropilot	Loop					A4	+	IS4	208				
							-	IC4	210				
AUMATIC	External									A0	+	CH0	100
											-	COM0	200

- Please refer to the AHI0812/AHO0412 user manual for the power supply and wiring concept.

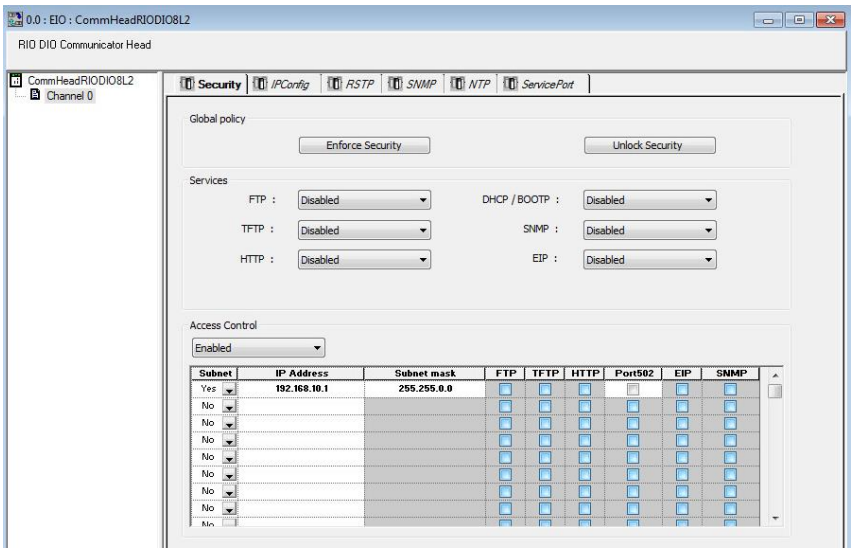
3.1.3 Network Configuration

3.1.3.1 M580 PLC IP Address

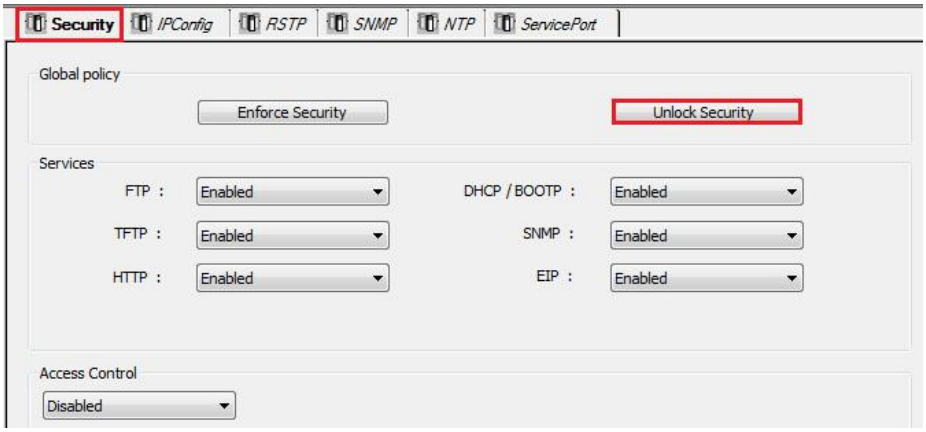
- In the Project Browser, double-click on the field “EIO”:



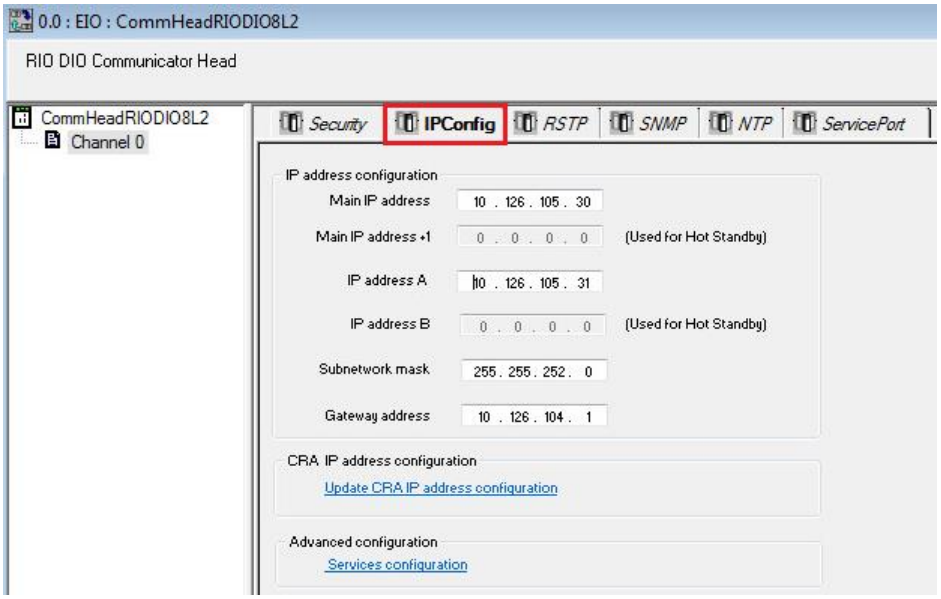
- Following window is displayed:



- In the Tab “Security”, click on the button “Unlock Security”. This enables the options “Services”:



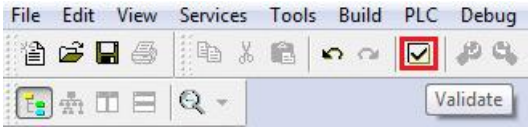
- Select the Tag “IPConfig” and configure the IP addresses of the PLC according to the connected network:



In this example:

  - The main PLC IP address is 10.126.105.30
  - The PLC IP address A is 10.126.105.31
  - The subnet mask is 255.255.252.0
  - The default gateway IP address is 10.126.104.1

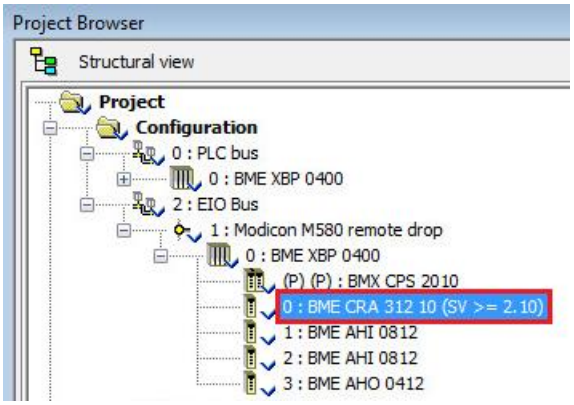
- Save the configuration by clicking on the symbol “Validate” in the tool bar:



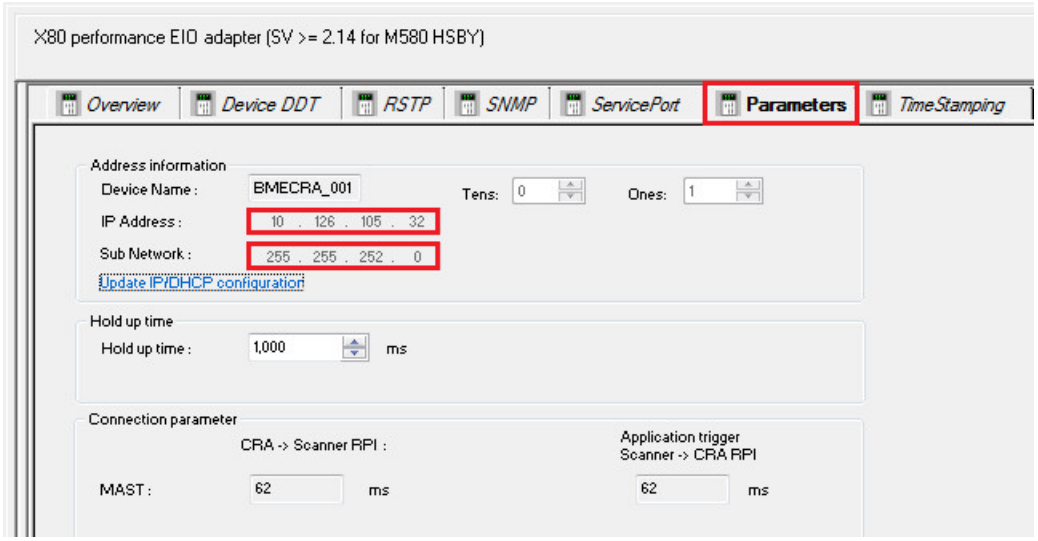
3.1.3.2 EIO Adapter CRA31210 IP Address and Device Name

A default IP address is automatically set for the EIO adapter according to the PLC and network configuration.

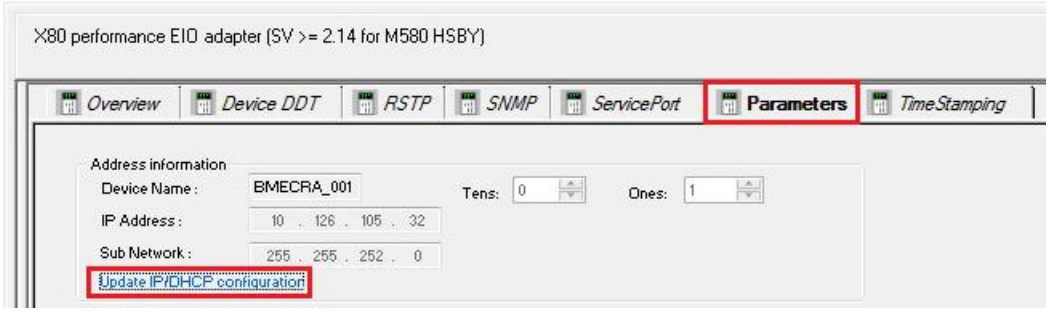
- In the Project view, double-click on the EIO adapter card:



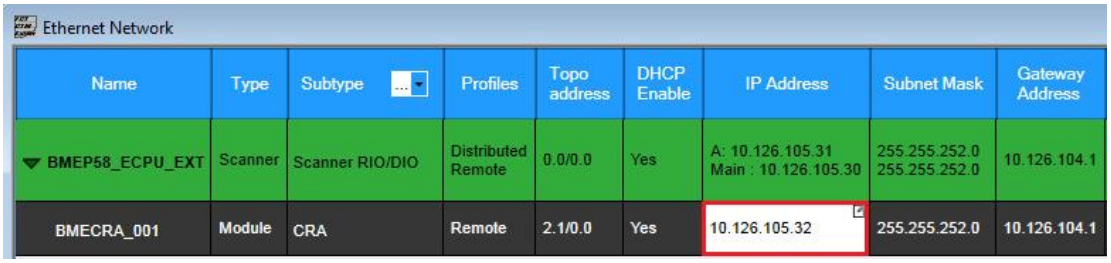
- Check the configured IP address of the EIO adapter in the tab “Parameters”:



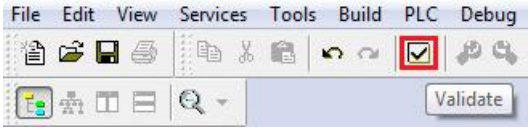
- If needed, the EIO adapter IP address can be changed.
  Click on the link “Update IP/DHCP configuration”:



- Click on the IP address field for changing the IP address:



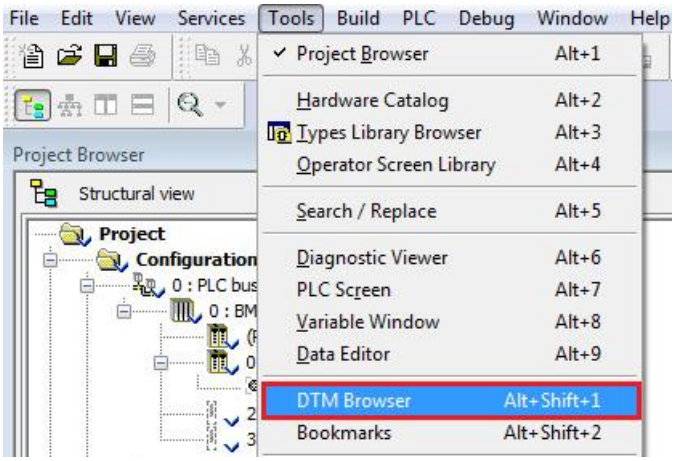
- Save the configuration by clicking on the symbol "Validate" in the tool bar:



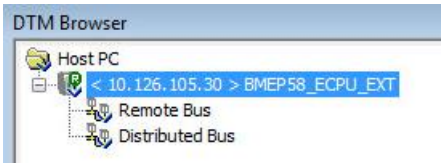
→For the next steps, the EIO adapter IP address is set to the IP address 10.126.105.32.

3.1.3.3 HART Analog Input AHI0812 Module 1 IP Address and Identifier

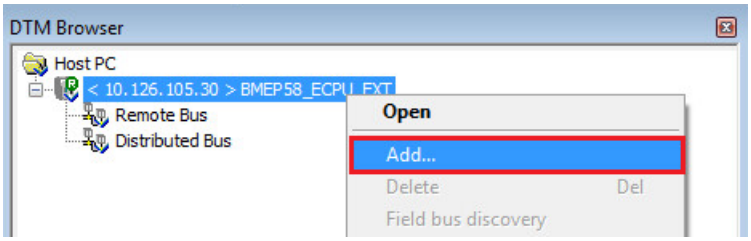
- Click on the menu "Tools→DTM Browser" in the Tool bar:



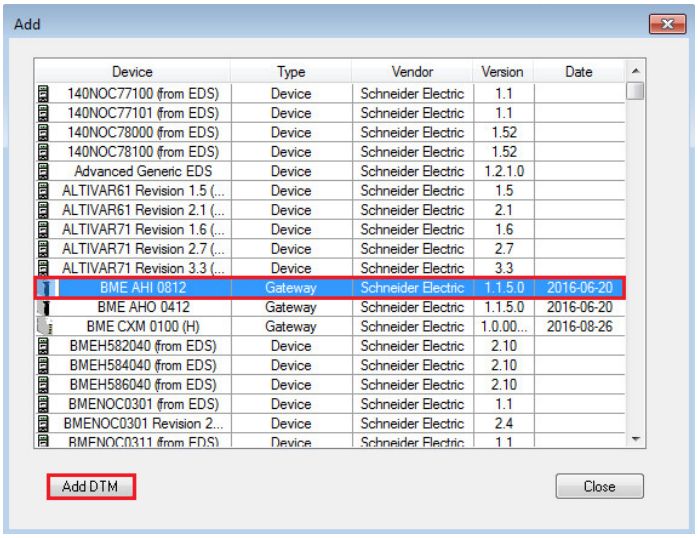
- The DTM Browser displays already the configured PLC IP address, 10.126.105.30 in this example:



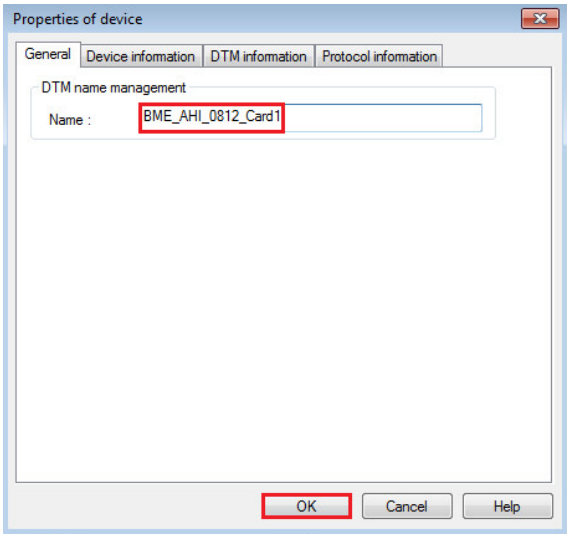
- Right-click on the DTM “BMEP58_ECPU” and select the menu “Add...”:



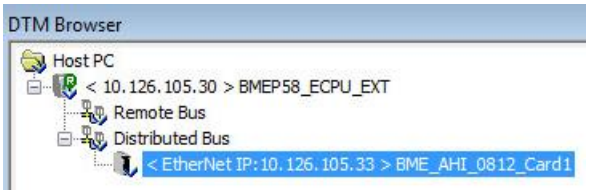
- Select the device “BMEAHI0812” and click on the button “add DTM”:



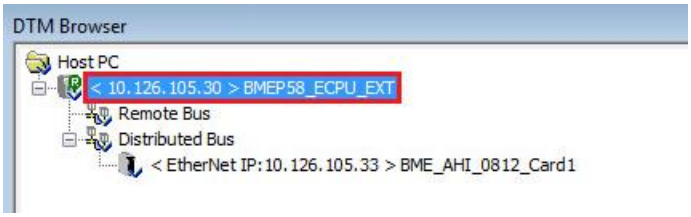
- Enter a DTM name for the project and click on the button “OK”. In this example the project DTM name is “BME_AHI_0812_Card1”:



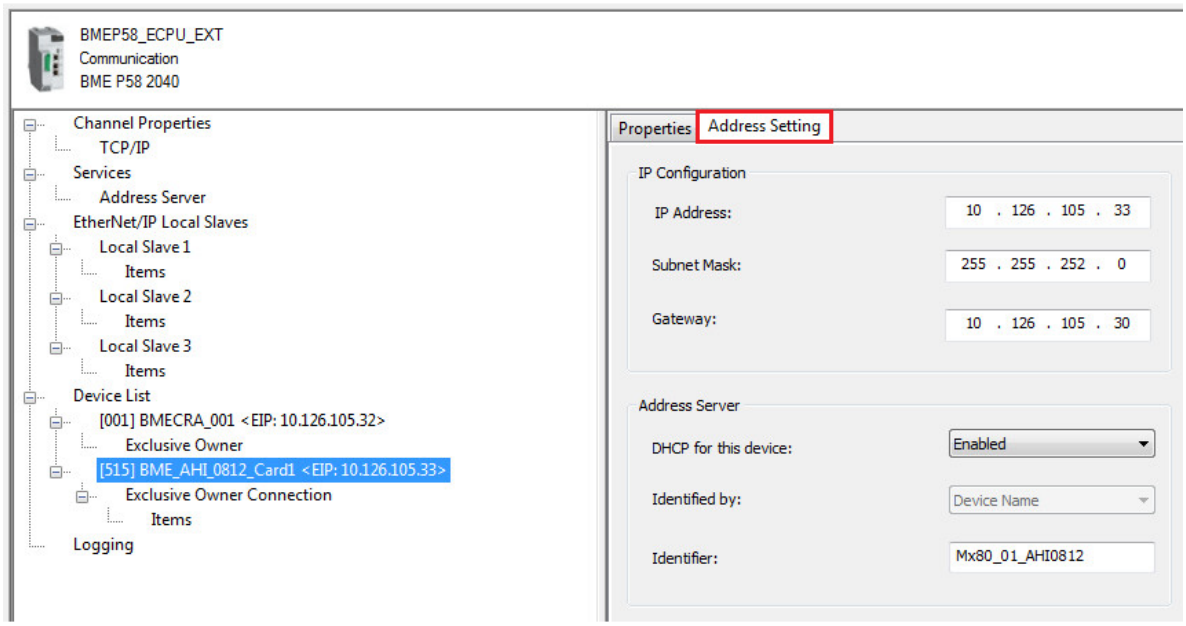
- The DTM is inserted in the project view with a generated IP address, which is 10.12.105.33 in this example:



- Double-Click on the DTM "BMEP58_ECPU":



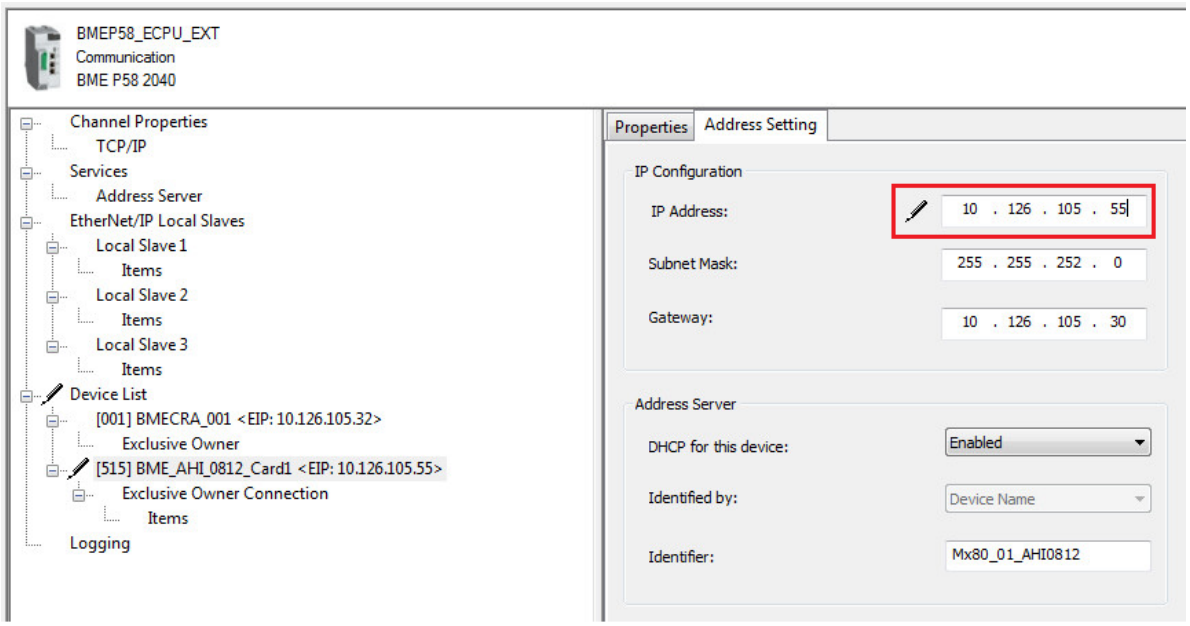
- Select the card "BME_AHI_0812_Card1" in the menu "Device List" and click on the tab "Address Setting":



Two parameters must be configured in the tab "Address Setting": the **IP address** and the **Identifier**.

IP address Configuration

- Configure the new IP Address. In this example, the new IP address is 10.126.105.55:



Identifier Configuration

- The Identifier naming is composed of three parts, which are the Rack ID, Slot Number and Device Name as explained in the AHI0812 user manual :

Parameter	Description

Rack ID	A 4-character field that identifies the rack used for the module:
	• Mx80: a main local rack
	• M58A: primary rack in a Hot Standby network design
	• M58B: standby rack in a Hot Standby network design
	• Cxxx: remote I/O rack, where:
	  • C indicates CRA
	  • xxx represents the rack number, an integer from 000...159

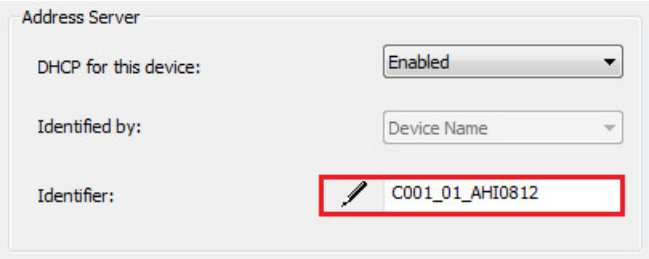
Slot Number	A field that identifies the position of the module in the rack.
Device Name	Use the following module names for the purpose of generating a **Device Name**: • the string “AHI0812” (not including quotation marks) for the BME AHI 0812 module • the string “AHO0412” (not including quotation marks) for the BME AHO 0412 module

Sample device name identifiers could be:
- Mx80_02_AHI0812 for a BME AHI 0812 module located at slot 2 of a main rack.
- M58A_03_AHI0812 for a BME AHI 0812 module located at slot 3 of a primary Hot Standby rack
- M58B_04_AHO0412 for a BME AHO 0412 module located at slot 4 of a standby Hot Standby rack
- C001_05_AHO0412 for a BME AHO 0412 module located at rack 1, slot 5 of a remote I/O rack

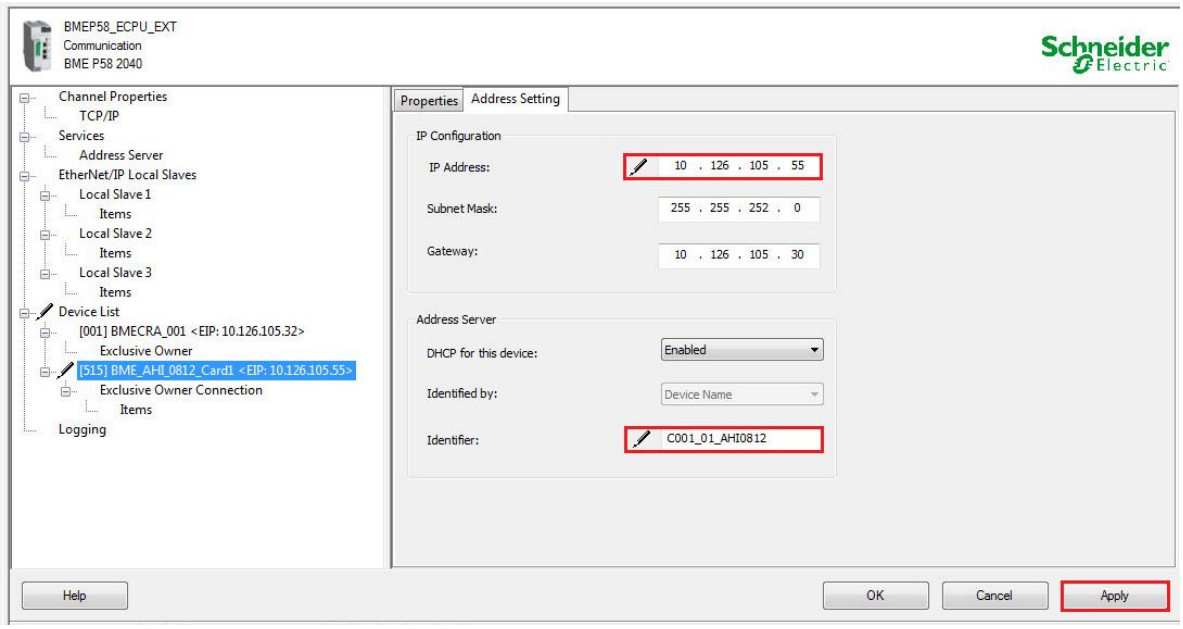
- In this example, the **default Identifier** is “Mx80_01_AHI0812”:

This must be changed because the card is located on the X80 Remote IO Rack.

- The correct Identifier is “C001_01_AHI0812” (RackID=1, SlotNr=1, DeviceName=AHI0812):

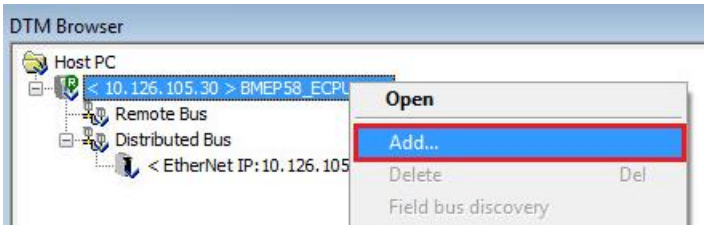


- Click on the button “Apply” when both options are configured and close the window:

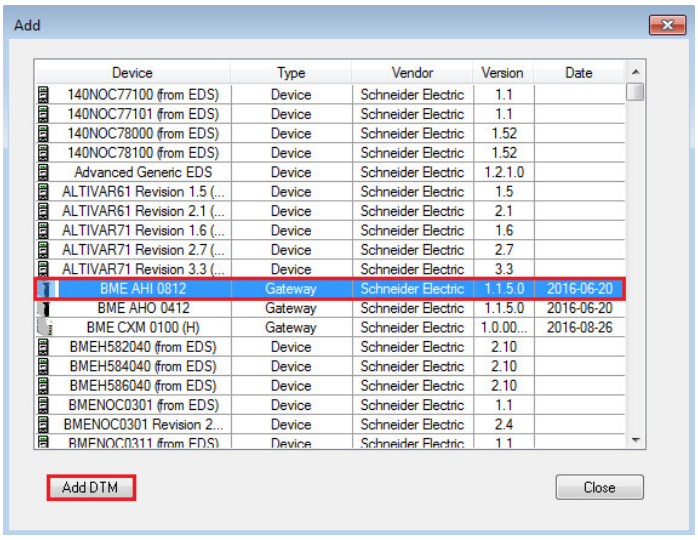


3.1.3.4 HART Analog Input AHI0812 Module 2 Address and Identifier

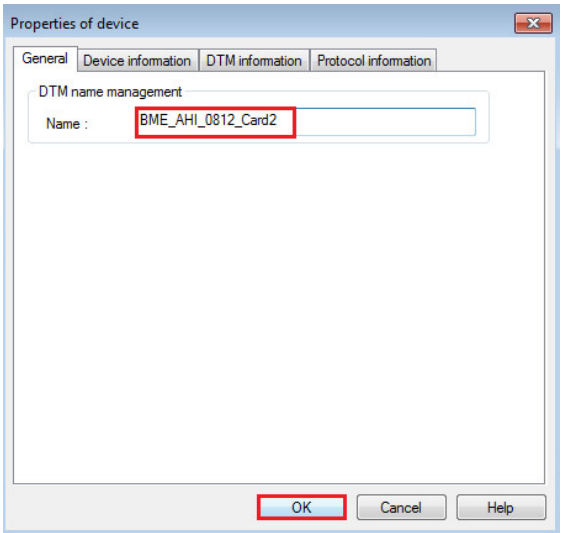
- In the DTM Browser view, right-click on the DTM “BMEP58_ECPU” and select the menu “Add...”:



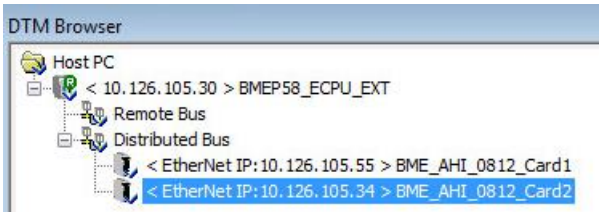
- Select the device “BMEAHI0812” and click on the button “add DTM”:



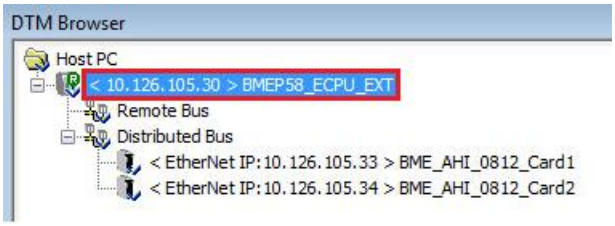
- Enter a DTM name for the project and click on the button “OK”. In this example the project DTM name is “BME_AHI_0812_Card2”:



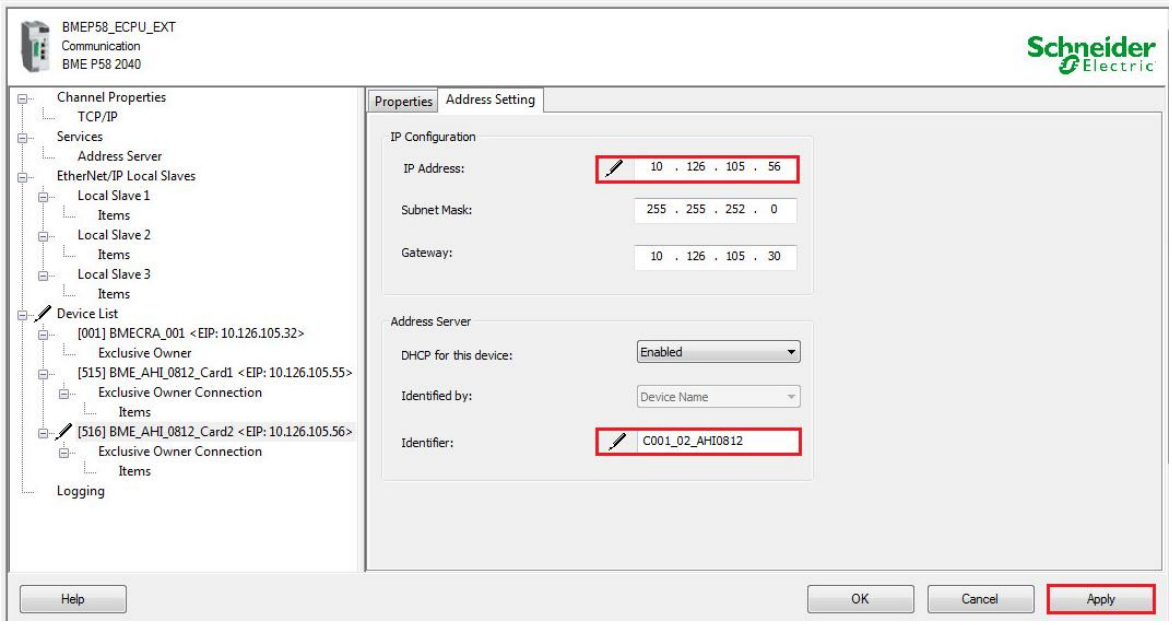
- The DTM is inserted in the project view with a generated IP address, which is 10.12.105.34 in this example:



- Double-Click on the DTM "BMEP58_ECPU":



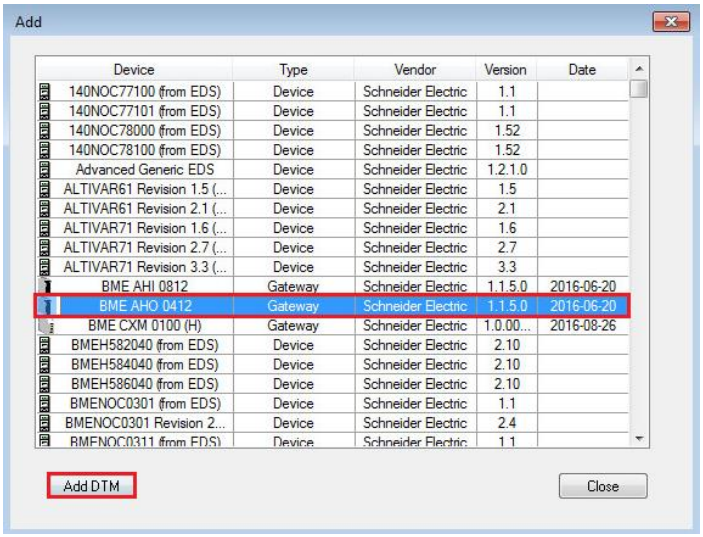
- Select the card "BME_AHI_0812_Card2" in the menu "Device List" and click on the tab "Address Setting" to update the IP address settings and the Identifier as done for the first HART analog input card in Chapter 3.1.3.3. Click on the button "Apply" to save:



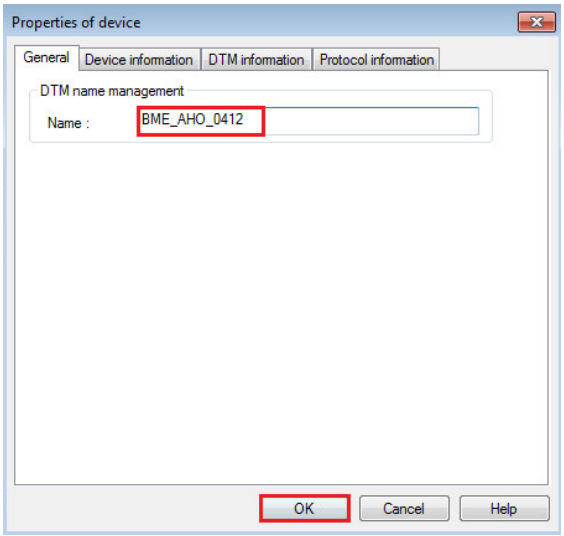
3.1.3.5 HART Analog Output AHO0412Module Address and Identifier

- In the DTM Browser view, right-click on the DTM “BMEP58_ECPU” and select the menu “Add...”:

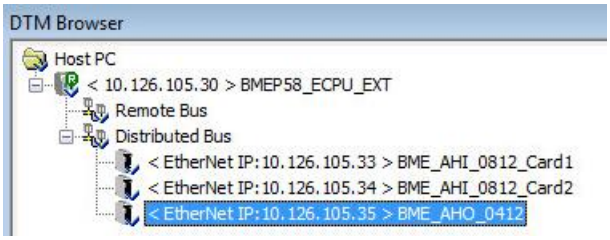
- Select the device “BMEAHI0812” and click on the button “add DTM”:



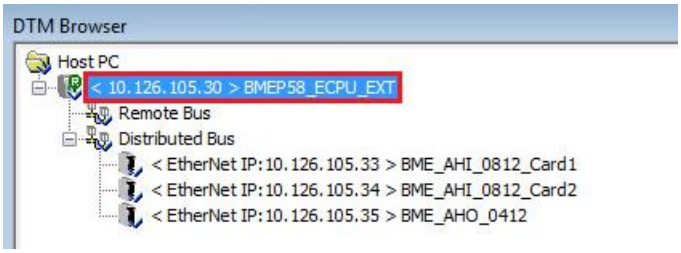
- Enter a DTM name for the project and click on the button “OK”. In this example the project DTM name is “BME_AHO_0412”:



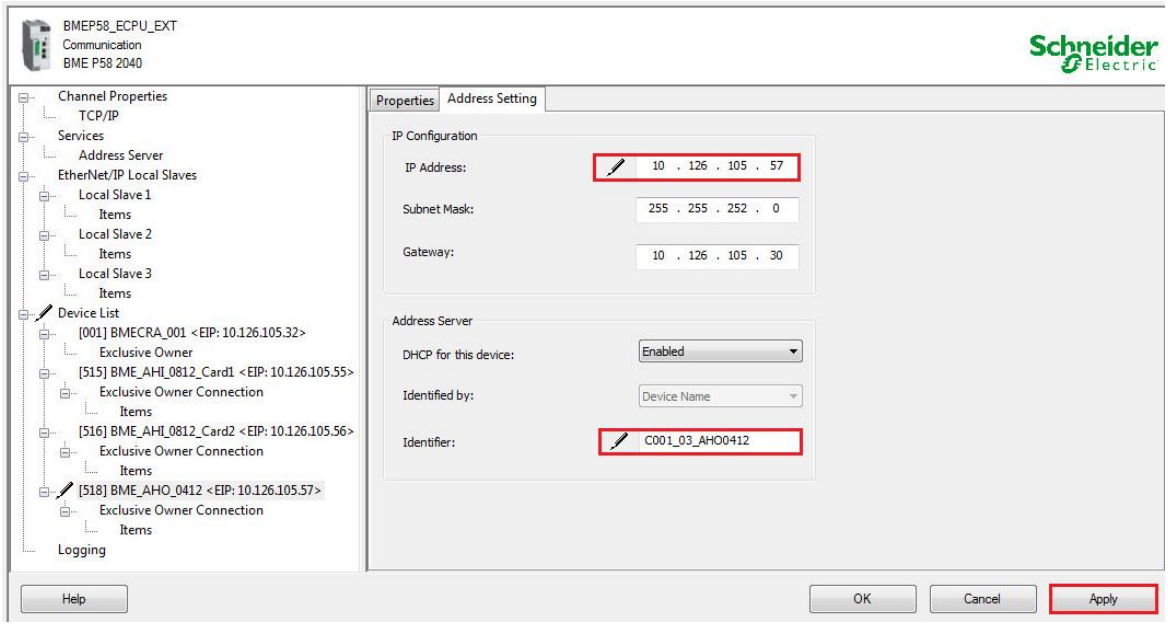
- The DTM is inserted in the project view with a generated IP address, which is 10.12.105.35 in this example:



- Double-Click on the DTM "BMEP58_ECPU":



- Select the card "BME_AHO_0412" in the menu "Device List" and click on the tab "Address Setting" to update the IP address settings and the Identifier as done for the first HART analog input card in Chapter 3.1.3.3. Click on the button "Apply" to save:



3.1.4 Connection to PLC

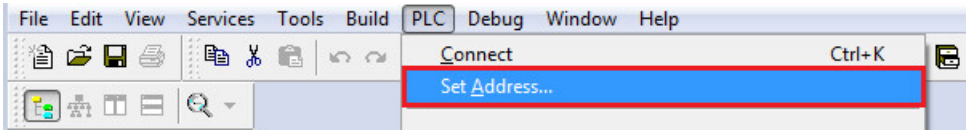
The first download will have to be done with the USB interface because no IP addresses are set in the PLC.

In this example, the PLC IP address is at first downloaded via USB. Then, it is the Ethernet connection which is used.

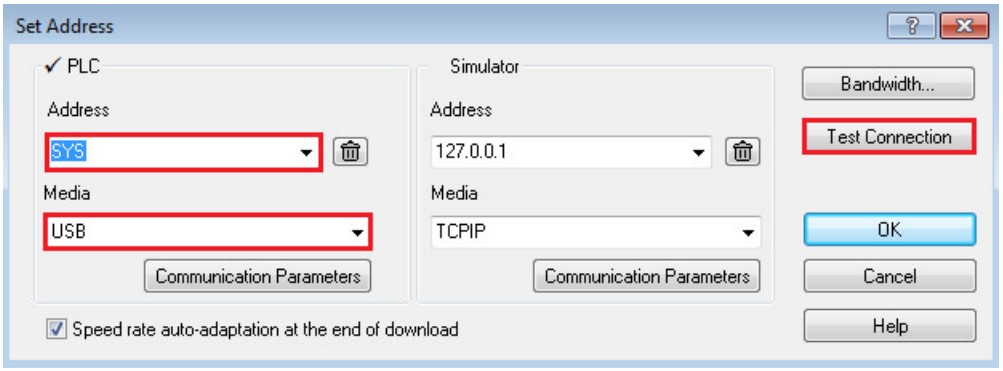
3.1.4.1 Connection via USB Interface

- Connect the USB cable from the PLC USB port to the engineering station one.

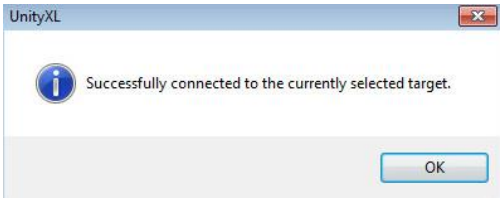
- In the tool bar, click on the menu "PLC→Set Address":



- Select the Address type "SYS" and the Media type "USB":



- Test if the connection is established by clicking on the button "Test Connection". If successful, following message is displayed:



Click on the button "OK".

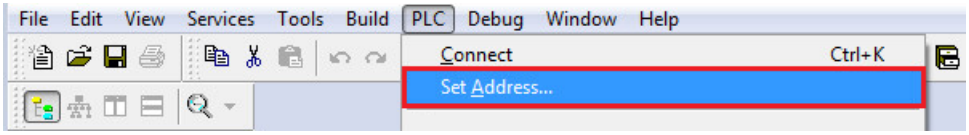
- Click on the button "OK" to close the window "Set Address".

3.1.4.2 IP settings Configuration Download via USB

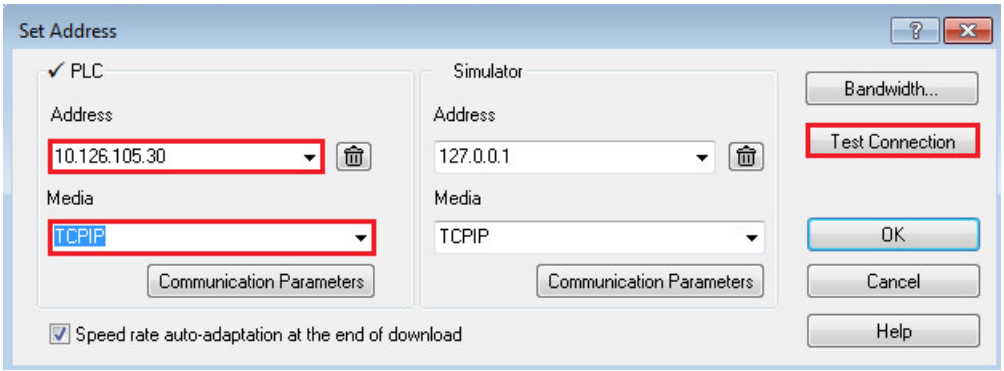
- Download the IP configuration in the PLC:
  →Refer to chapter 0 to proceed.

3.1.4.3 Connection via Ethernet

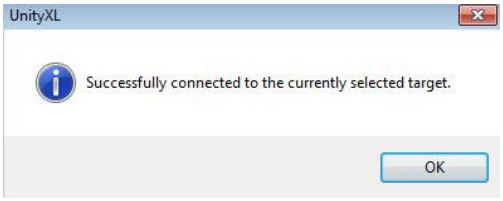
- In the tool bar, click on the menu "PLC→Set Address":



- Enter the PLC IP address as Address type and select the Media type "TCPIP". In this example, the PLC IP address is 10.126.105.30:



- Test if the connection is established by clicking on the button "Test Connection". If successful, following message is displayed:



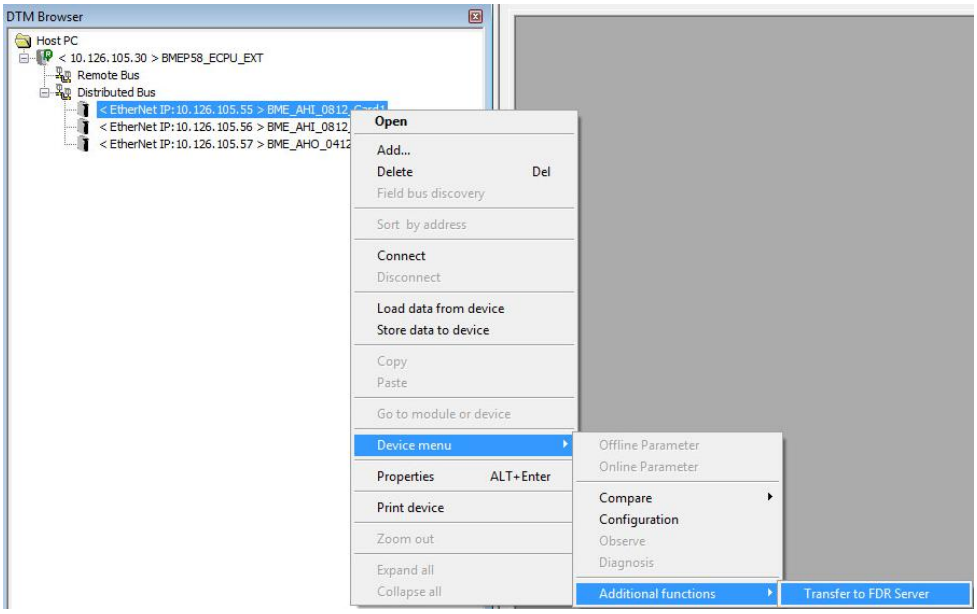
Click on the button "OK".

- Click on the button "OK" to close the window "Set Address".

3.1.4.4 FDR Server Transfer

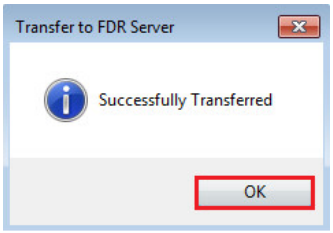
This step is needed to set the IP addresses of the HART cards configured in the DTM Browser.

- In the DTM Browser view, right-click on the DTM “BME_AHI_0812_Card1”and select the option “Device menu→Additional Functions→Transfer to FDR Server”:

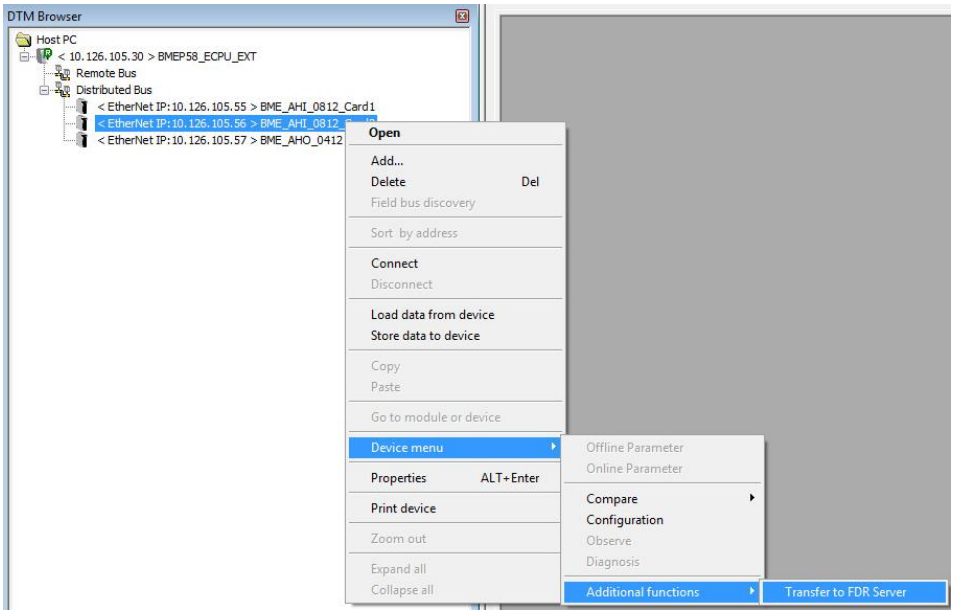


- Click on the button “Yes” to proceed:

- Transfer was successful. Click on the “OK” to close the window:

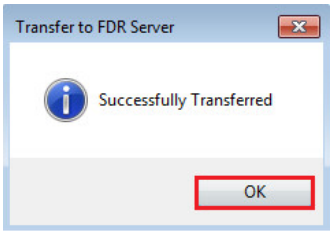


- In the DTM Browser view, right-click on the DTM “BME_AHI_0812_Card2”and select the option “Device menu→Additional Functions→Transfer to FDR Server”:

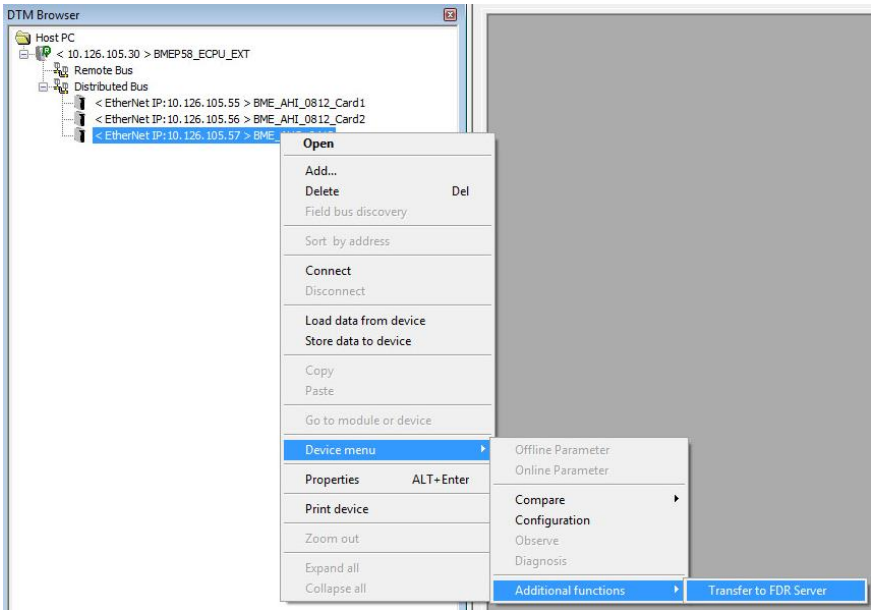


- Click on the button “Yes” to proceed:

- Transfer was successful. Click on the “OK” to close the window:

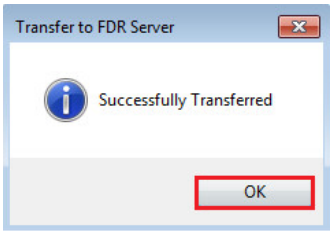


- In the DTM Browser view, right-click on the DTM “BME_AHI_0812_Card2”and select the option “Device menu→Additional Functions→Transfer to FDR Server”:



- Click on the button “Yes” to proceed:

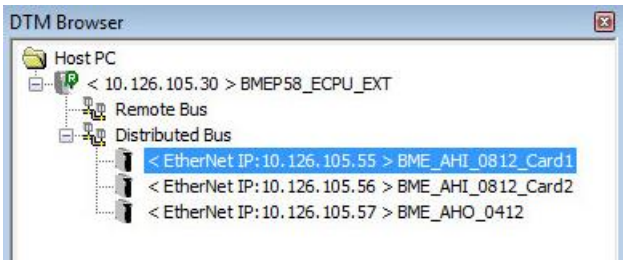
- Transfer was successful. Click on the “OK” to close the window:



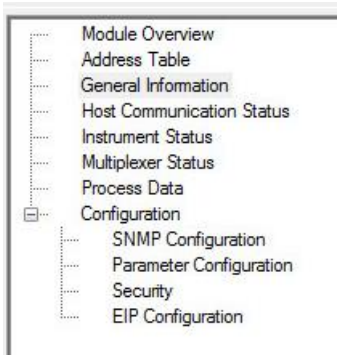
3.1.5 HART Cards Configuration

3.1.5.1 HART Analog Input Cards Configuration

- In the DTM Browser, double-click on the DTM “BME_AHI_0812_Card1”:



- This opens the DTM offline configuration window:

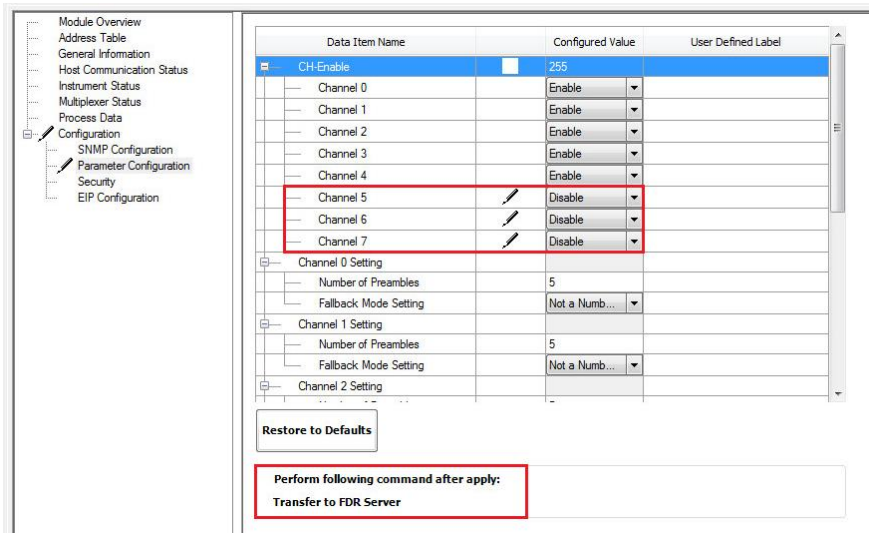


- Different parameters can be modified in the menus “Module Overview”, “General Information”, “Multiplexer Status” and “Configuration”.

- Default parameters are used for the menus “Module Overview”, “General Information”, “Multiplexer Status”.

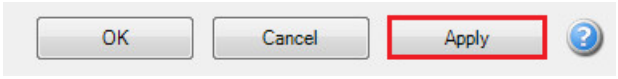
  → Please refer to the AHI0812 user manual for further details.

- Menu “Parameter Configuration” Update:
  Expand the menu Configuration and select the menu “Parameter Configuration”.
  In this example, HART Channels 5, 6 and 7 are disabled (Channels 5, 6 and 7 have been disabled in Chapter 3.1.2.2.2):

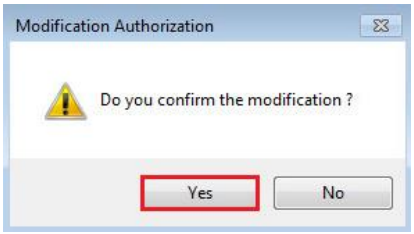


- Check always the “Perform following command after apply” message. E.g., changes in the “Parameter Configuration” tab require again a “Transfer to FDR Server” as done in Chapter 3.1.4.4.

- Click on the button “Apply” to save the modification:



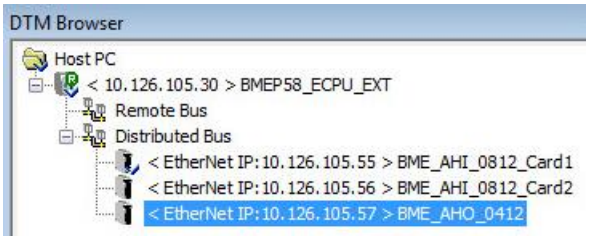
- If asked for Modification Authorization, click on the button “Yes”:



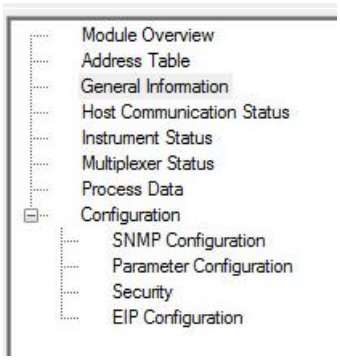
- After that, perform the requested “Transfer to FDR Server” as done in Chapter 3.1.4.4.

3.1.5.2 HART Analog Output Cards Configuration

- In the DTM Browser, double-click on the DTM “BME_AHO_0412”:



- This opens the DTM offline configuration window:



- As for the AHI0812 card, different parameters can be modified in the menus “Module Overview”, “General Information”, “Multiplexer Status” and “Configuration”.

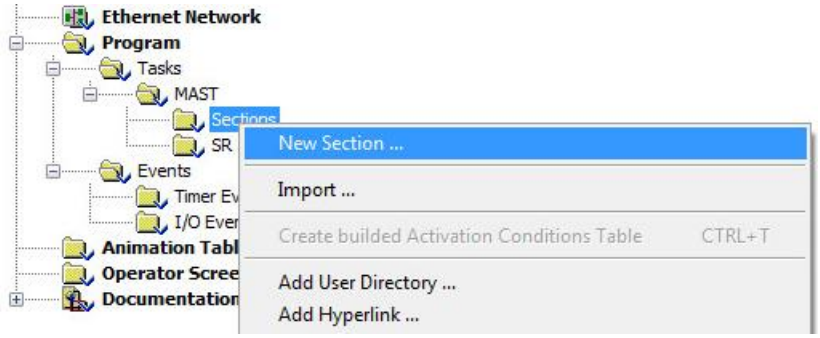
- Default parameters are used for the menus “Module Overview”, “General Information”, “Multiplexer Status”.

  → Please refer to the AHO0412 user manual for further details.

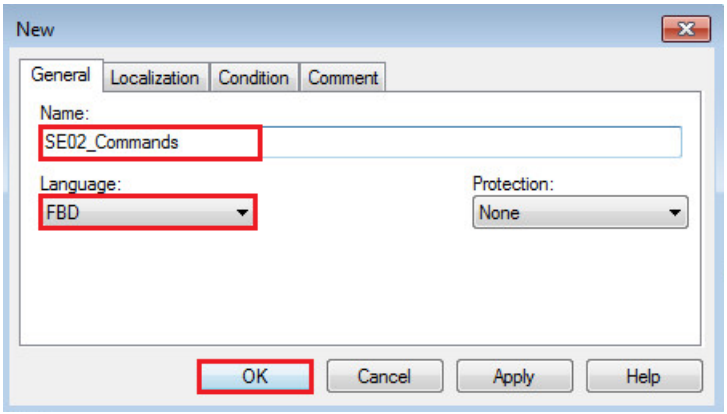
3.2 Mapping of Process Values and Status to Control Strategy

3.2.1 New Program

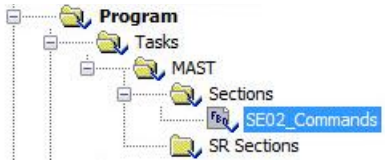
- In the Project Browser view, right-click on “Sections” and select the menu “New Section”:



- Enter a Name for the new Section and select a language. Then click on the button “OK”. In this example, the section name is “SE02_Commands” and the language is “FBD”:



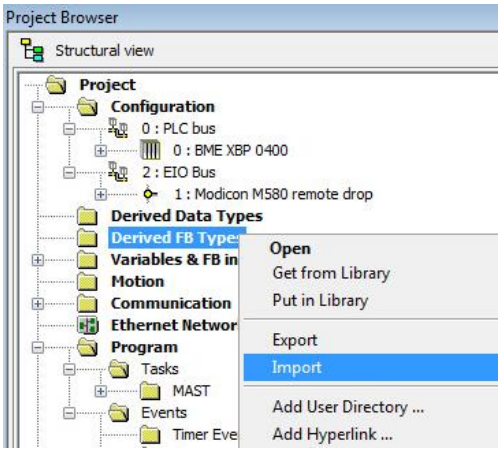
- New section is created:



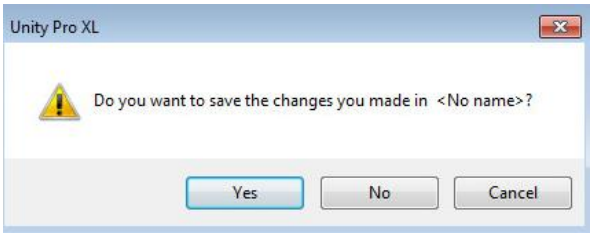
3.2.2 eX80 HART Generic DFB Installation

The library “ex80_hart_generic_dfb.xdb” contains Universal and Common Practice HART Commands function blocks.

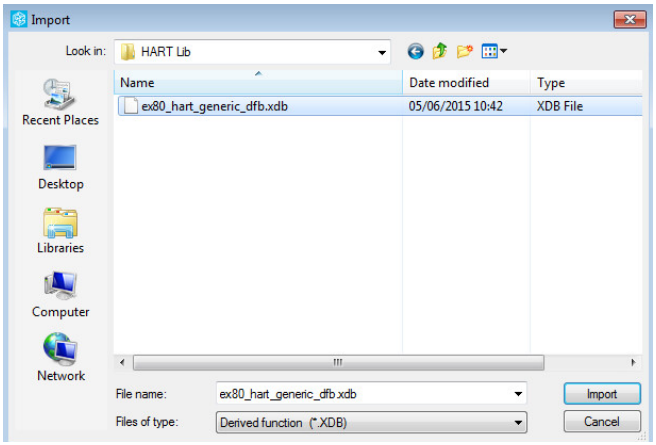
- Right-click on the menu “Derived FB Types” and select the menu “Import”:



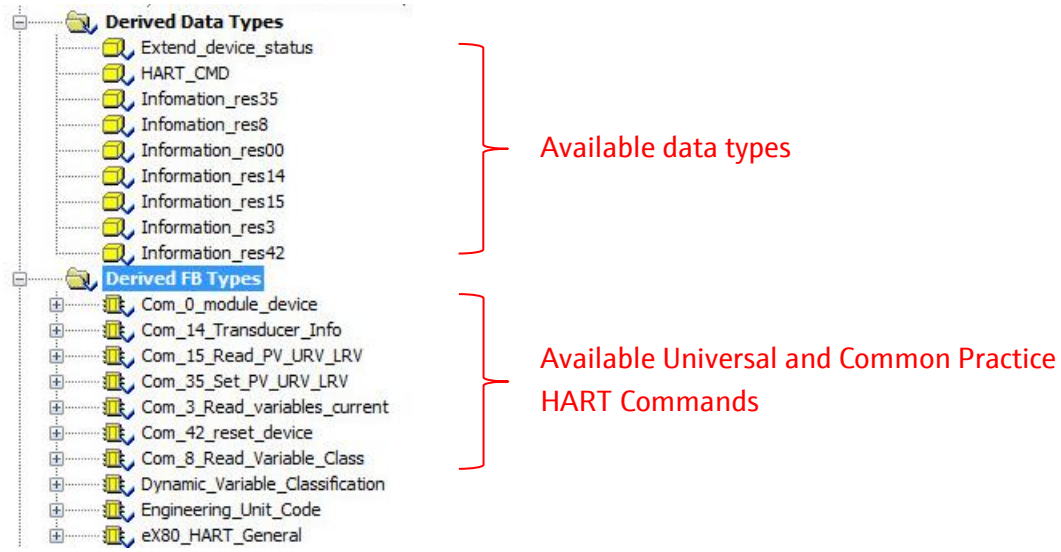
- Project can be saved before the library import:

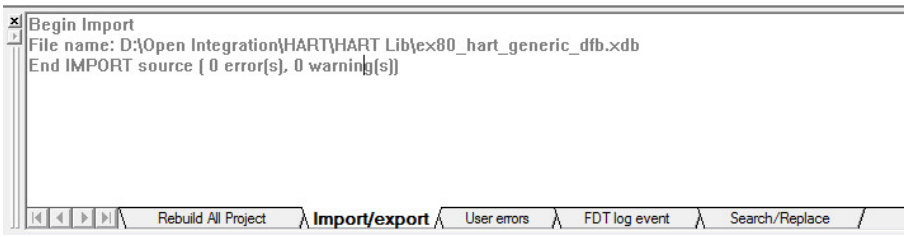


- Select the “ex80_hart_generic_dfb.xdb” file and click on the button “Import”:



- Imported data:

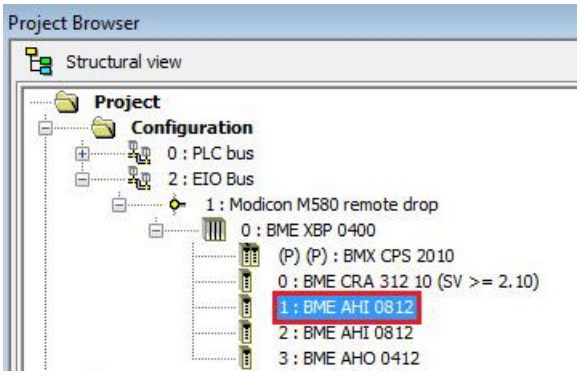




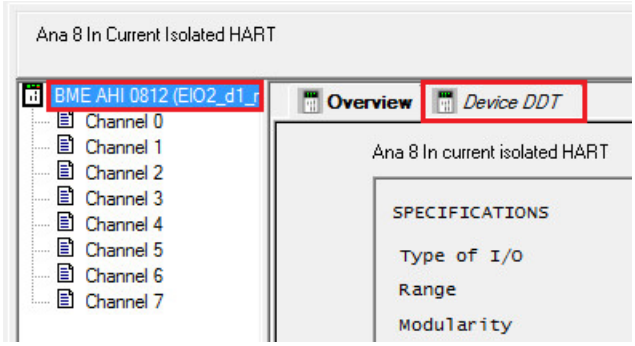
3.2.3 4...20mA Inputs/Outputs

The 4...20mA signal is part of the card data structure (Device DDT). A device DDT is created for each new module.

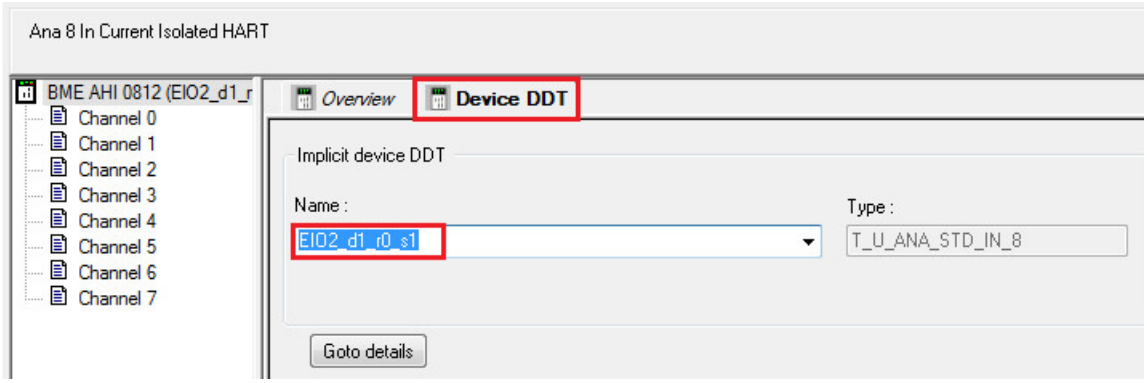
- In the Project view, double-click on the analog input module AHI0812:



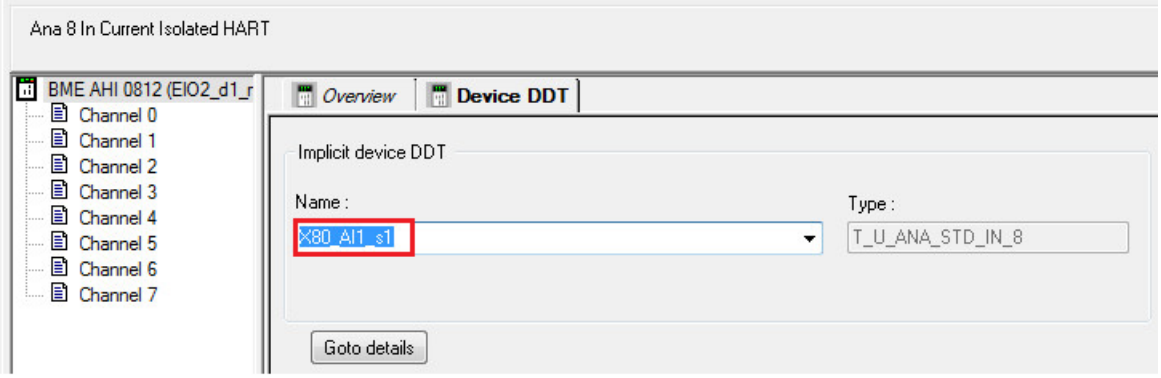
- Select the field "BME AHI 0812" and click on the tab "Device DDT":



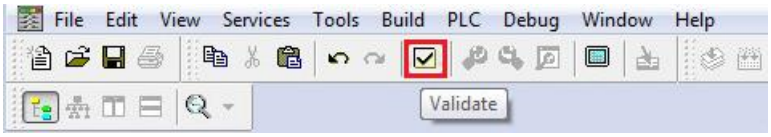
- This displays the device DDT variable name attached to this card:



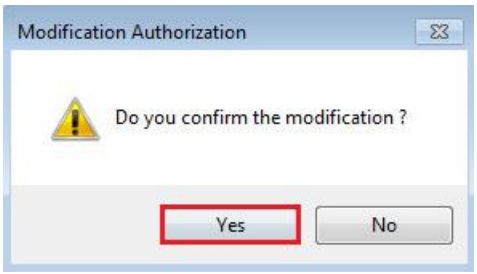
- Update the name if needed, for example “X80_AI1_s1”:



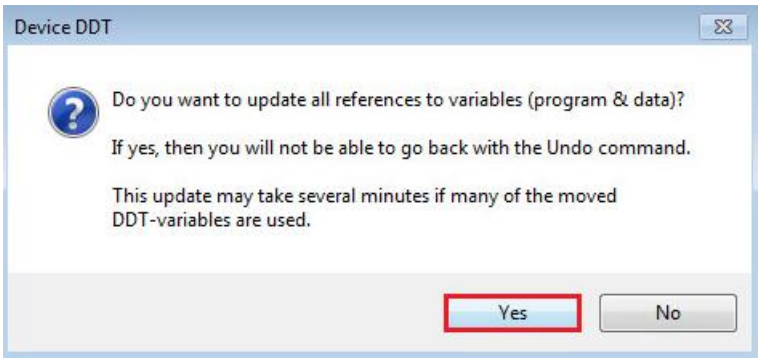
- Save the modification by clicking on the shortcut button “Validate”:



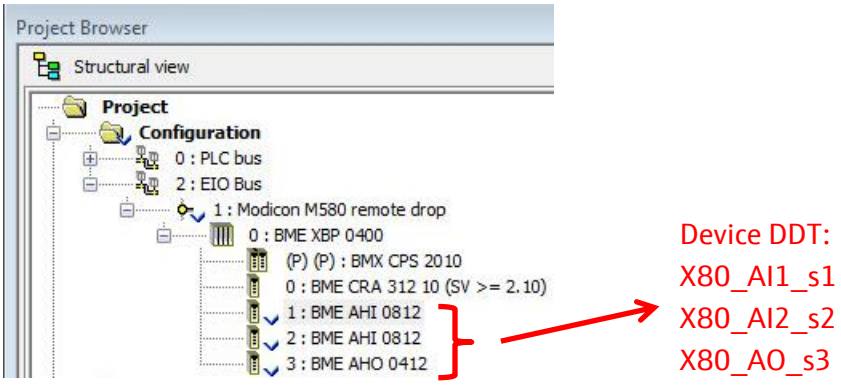
- Confirm the modification by clicking on the button “Yes”:



- Click on the button “Yes” to proceed:



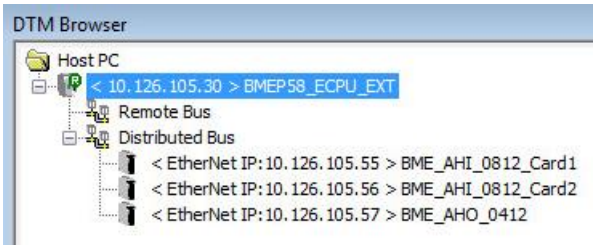
- Repeat the previous steps for the other cards of the X80 Remote IO:



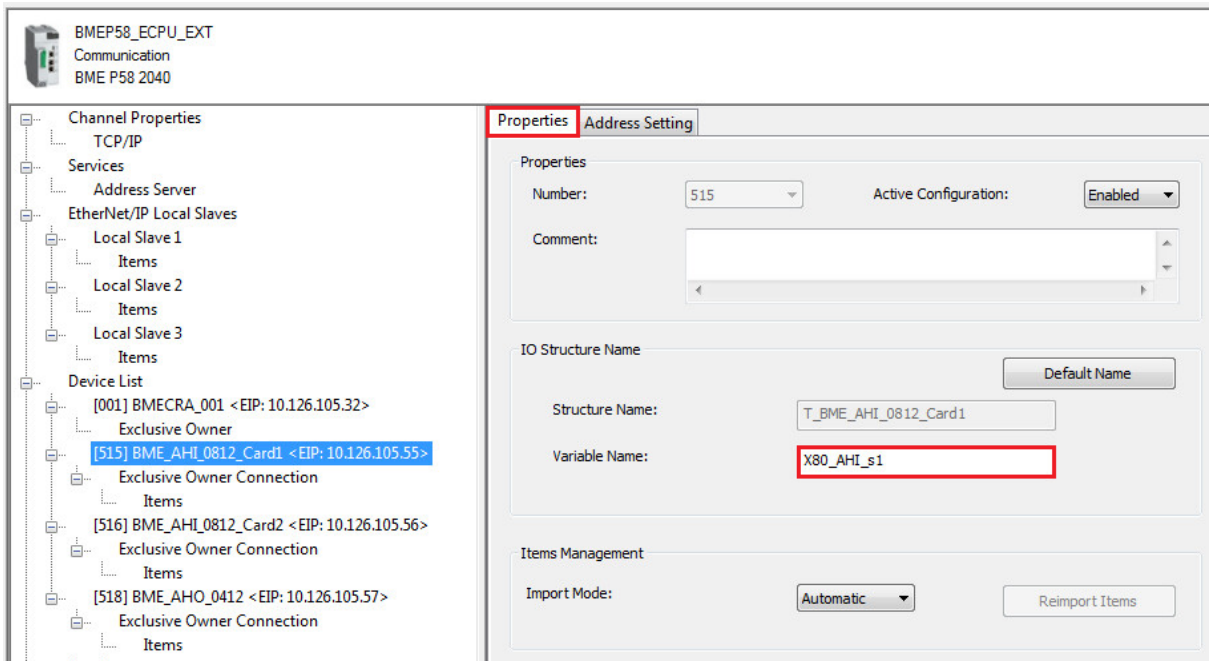
3.2.4 HART Inputs/Outputs

HART data can be accessed for each card via a variable defined in the M580 Master DTM.

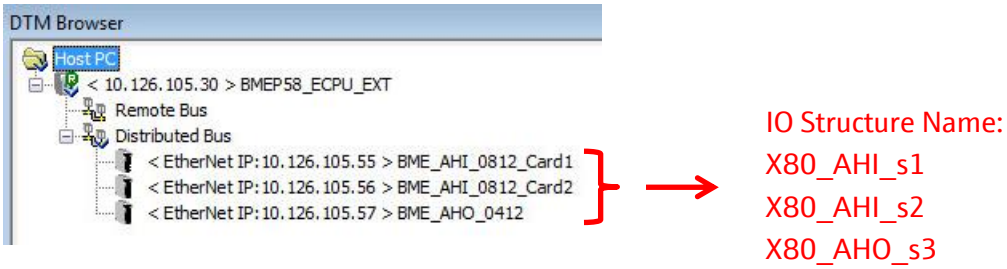
- Double-click on the field "BMEP58_ECPU_EXT":



- Select the first analog input card and update the Variable Name if needed, for example "X80_AHI_s1":



- Repeat the previous steps for the other cards of the X80 Remote IO:



3.2.5 Universal HART Commands

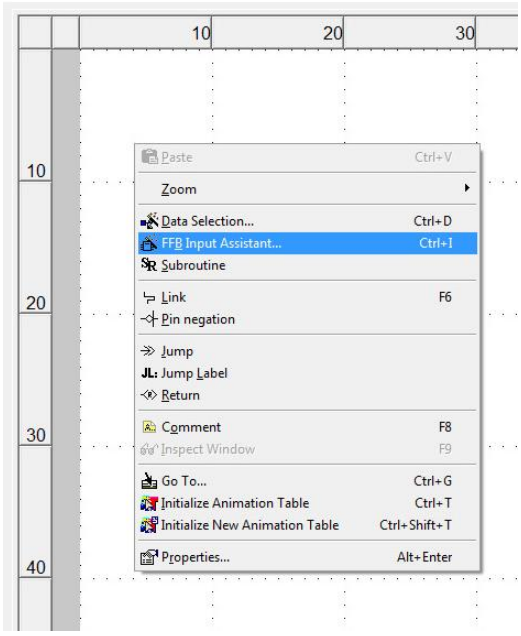
All universal HART Command function blocks are used in a similar manner. Each HART Command function block requires an “Interface Input” and an “Interface Output” signal as well as a BOOL input variable to activate the function block.

The following chapter shows how to configure these parameters for Command 0. All this is applicable for all other Commands later on.

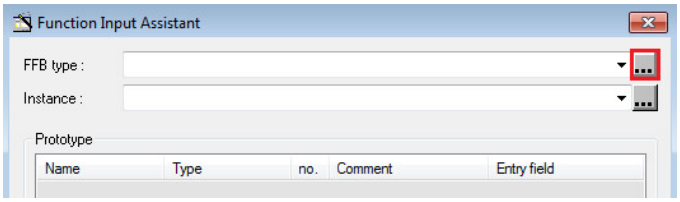
3.2.5.1 HART CMD 0: Read unique Identifier

- HART CMD0 is the first function which needs to be configured in order to get the Module ID and Device ID parameters. These parameters are needed as Input for all other function blocks.

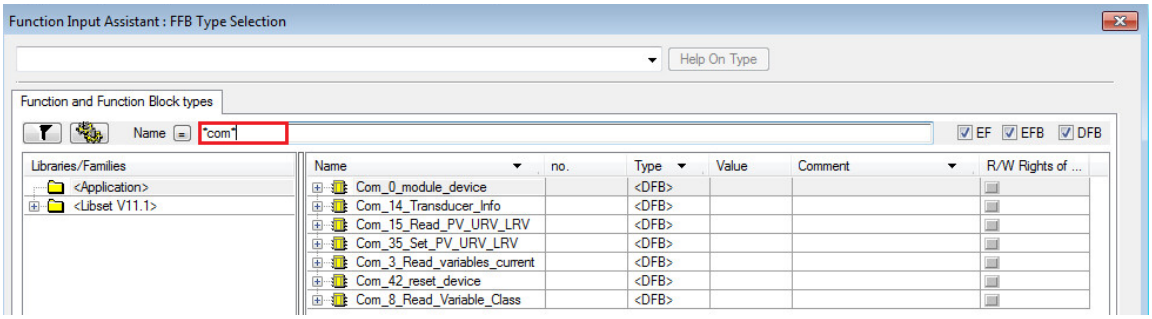
- In the “SE02_Commands” program page, right-click in the page and select the option “FBB Input Assistant”:



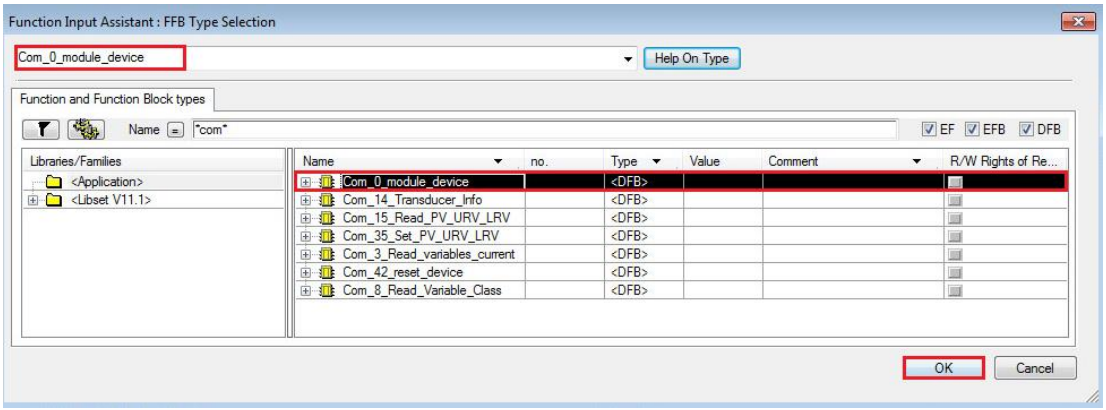
- Click on the “Browse” shortcut:



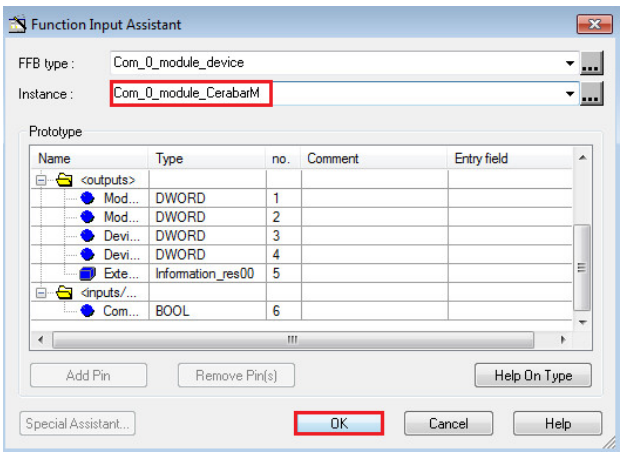
- Look for the HART CMD0 function block by searching with “*com*”. This displays available function blocks:



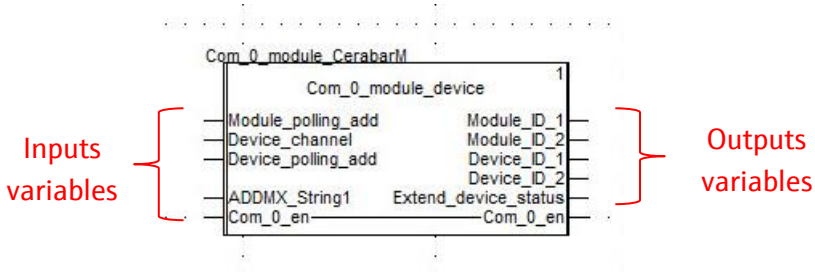
- Select the function block “Com_0_module_device” and click on the button “OK”:



- Change the Instance name, “Com_0_module_CerabarM” in this example and click on the button “OK”:

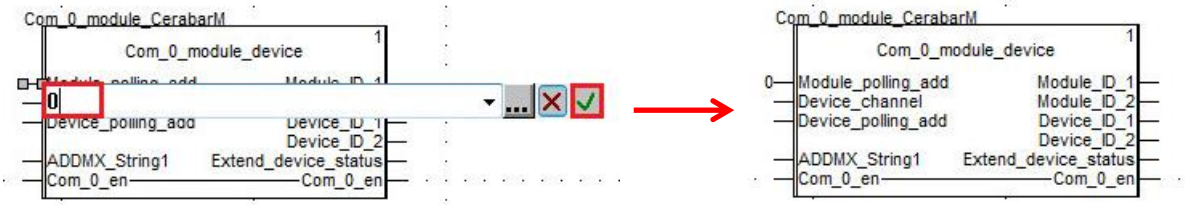


- Click on the page for inserting the function block:



Input variables assignment

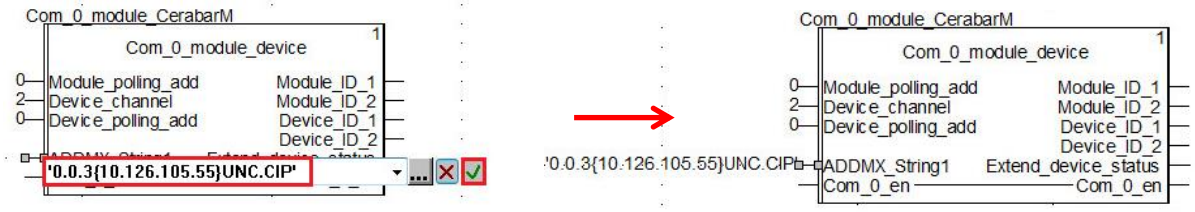
- Module Polling Address
  Click on the wire, enter the module polling address, "0" for this example and validate:



- Device Channel
  Click on the wire, enter the channel number, "3" for this example and validate:

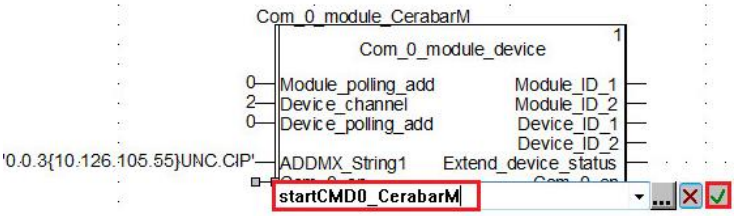
- Device Polling Address
  Click on the wire, enter the device polling address, "0" for this example and validate:

- ADDMX_String1
  Click on the wire, enter the value "0.0.3{10.126.105.55}UNC.CIP" for this example and validate:

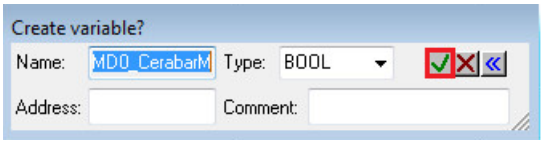


  The parameter "ADDMX_String1", '0.0.3{10.126.105.55}UNC.CIP', is made of six parts 'A.B.C{D}E.F':
  - 'A' contains the rack number of the communication module. This is application specific.
  - 'B' contains the slot number of the communication module. This is application specific.
  - 'C' contains the communication channel: This is always "3".
  - 'D' contains the IP address of the HART module AHI0812/AHO0412.
  - 'E' contains the message type. This always 'UNC' (Unconnected message).
  - 'F' contains the protocol type. This is always 'CIP'.

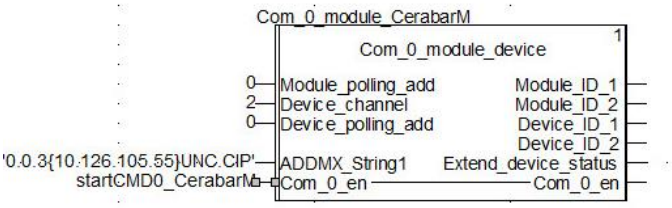
- Com_0_en
  Click on the wire, enter the start bit, "startCMD0_CerabarM" for this example and validate:



- If the variable does not exist, a menu is asking for creating this variable with the correct type. Just validate:

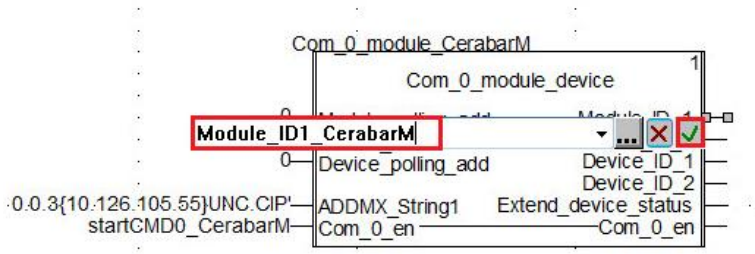


- Configured inputs:

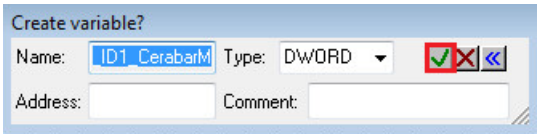


Output variables assignment

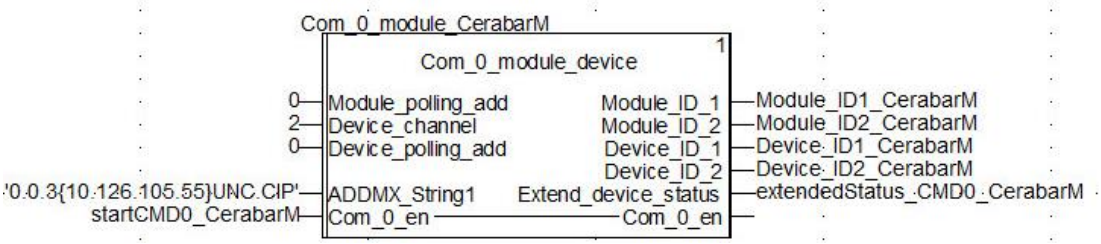
- Module_ID_1:
  Click on the wire and enter for example the variable “Module_ID1_CerabarM”:



- If the variable does not exist, a menu is asking for creating this variable with the correct type. Just validate:



- Assign all other outputs variables as done for “Module_ID_1”:

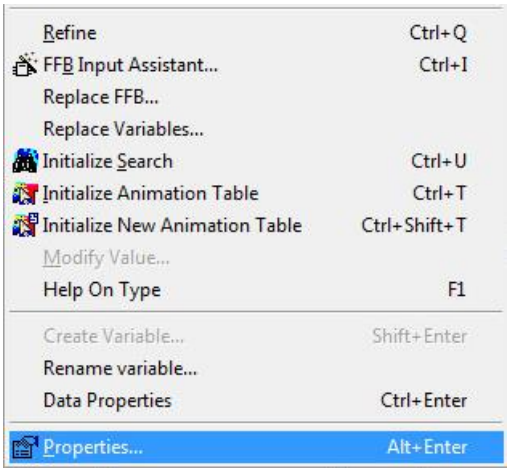


Function Block activation/deactivation

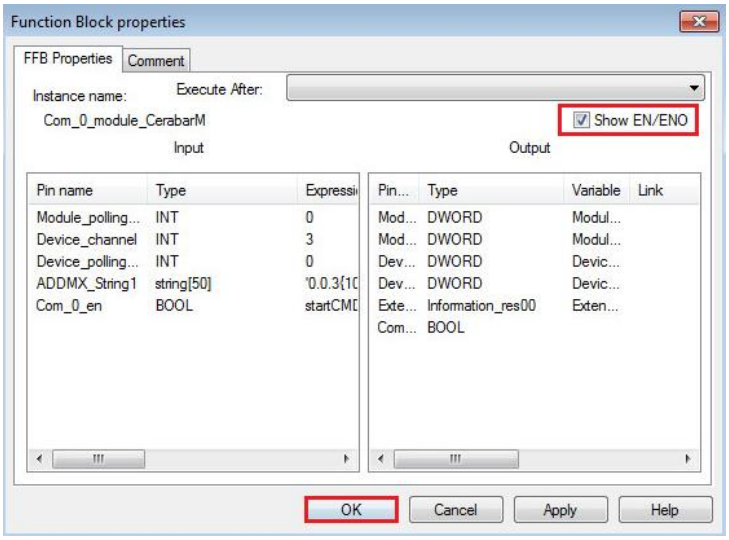
- The function block is activated per default. However, an additional variable has to be set in order to trigger the activation/deactivation of the function block. This is recommended for using deviceDTMs either in Unity Pro or in FieldCare because only one HART master can be active with the same devices at the same time.

- Therefore, before using the HART Commands function blocks with a device, make sure that the corresponding deviceDTM is disconnected as wells as the HART module comDTM of the corresponding card on which is connected the device.

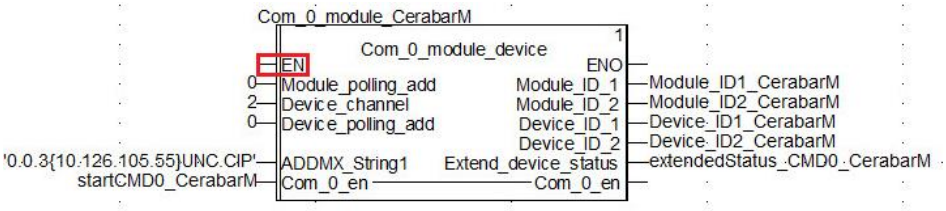
- Right-click on the CMD0 function block and select the option “Properties”:



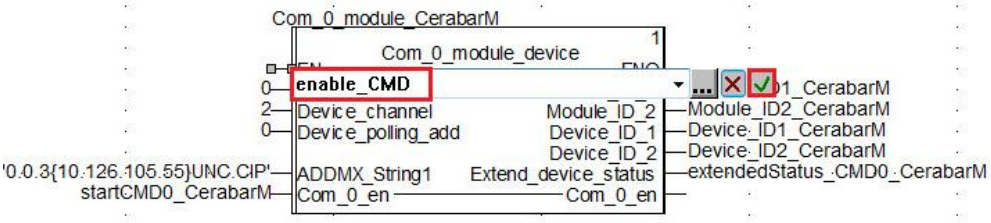
- Select the option “Show EN/ENO” and click on the button “OK”:



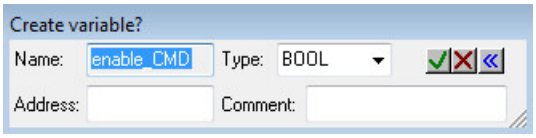
- Inserted additional Inputs:



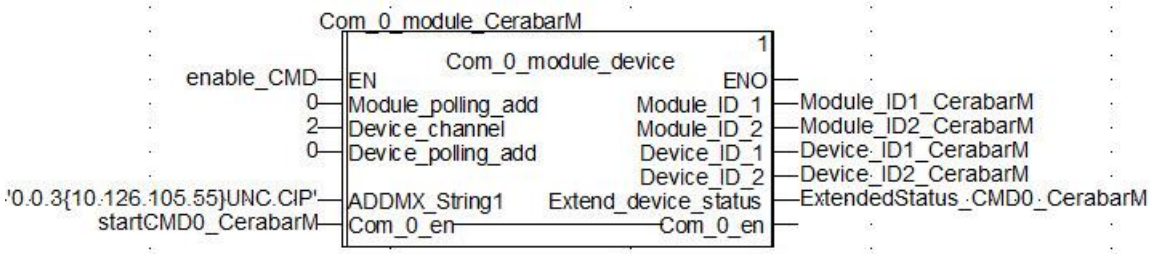
- Click on the wire and assign the corresponding variable:



- If the variable does not exist, a menu is asking for creating this variable with the correct type. Just validate:

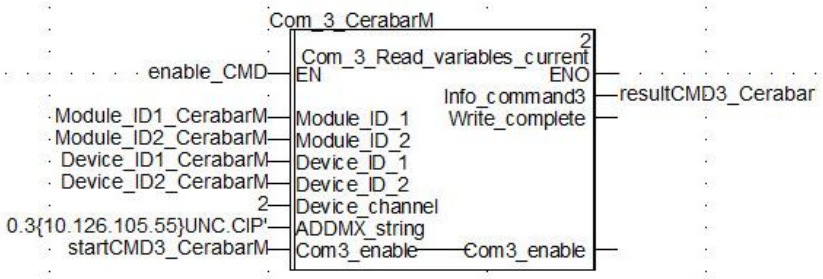


- Configured CMD0 Function Block:



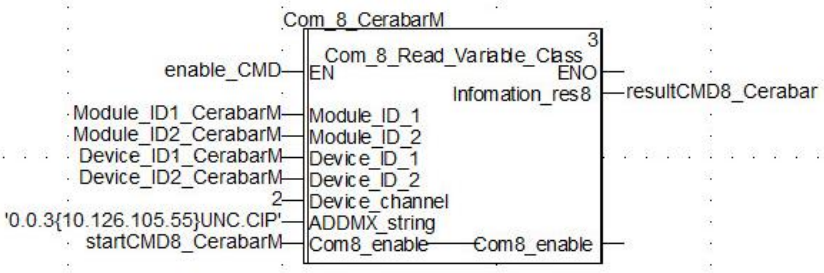
3.2.5.2 HART CMD 3: Read dynamic variables and loop current

- HART CMD3 function block:



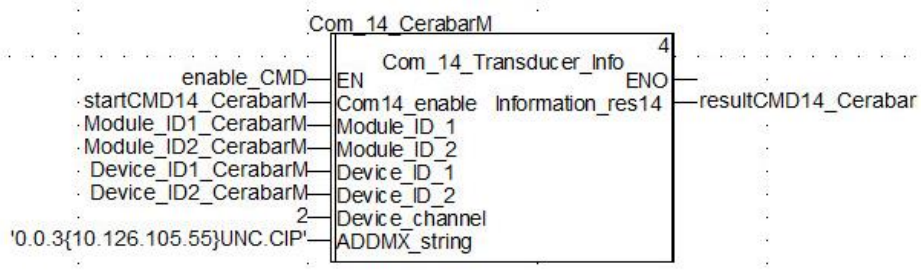
3.2.5.3 HART CMD 8: Read dynamic variable classifications (from HART6)

- HART CMD8 function block:



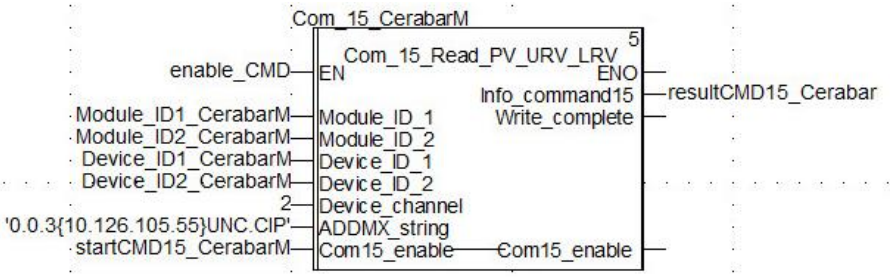
3.2.5.4 HART CMD 14: Read primary variable sensor information

- HART CMD14 function block:



3.2.5.5 HART CMD 15: Read primary variable output information

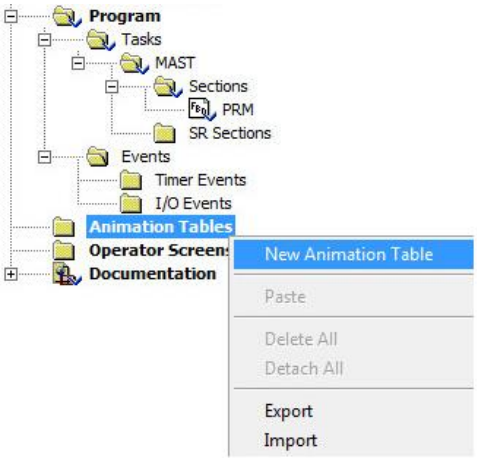
- HART CMD15 function block:



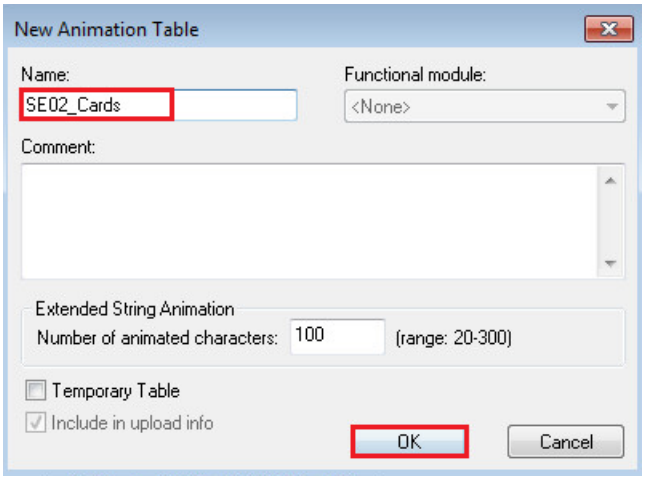
3.2.6 Animation Tables Configuration

Animation tables are used to display variables values in online mode. The following part explains how to configure an animation table for displaying relevant data.

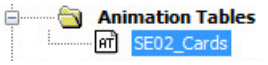
- In the Project Browser, right-click on the menu “Animation Tables” and select the menu “New Animation Table”:



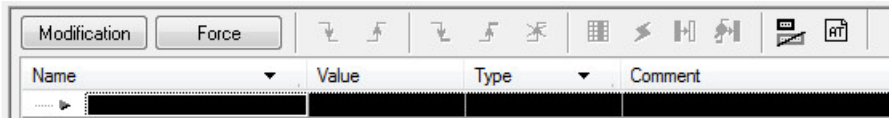
- Enter a name for the animation table and click on the button “OK”:



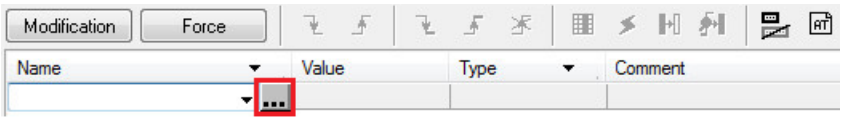
- The created animation table is added in the Project Browser:



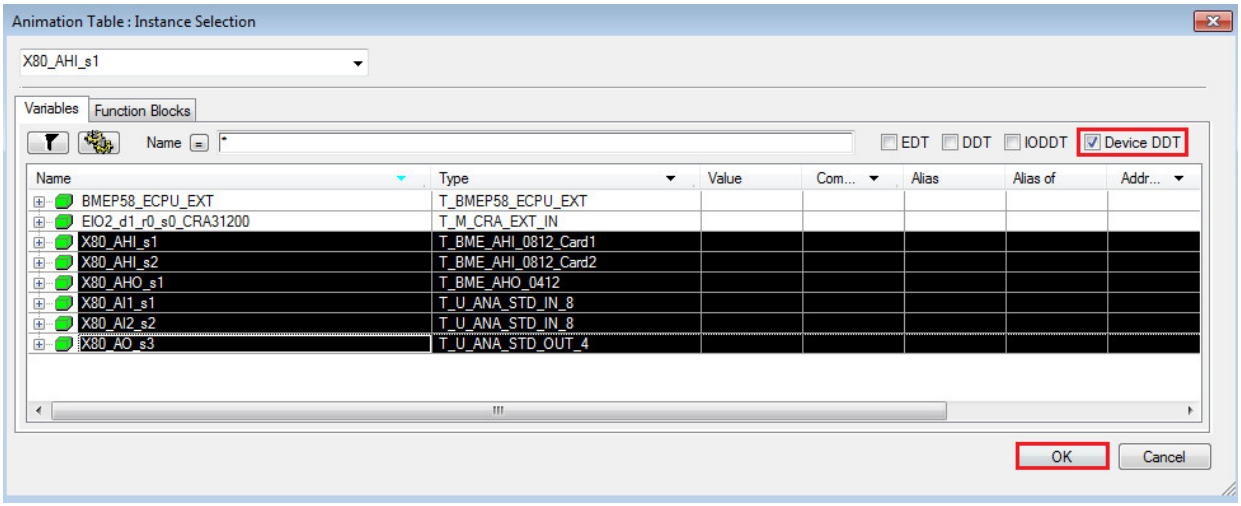
- Double-click on the animation table “SE02_Cards”.
  Following window is displayed:



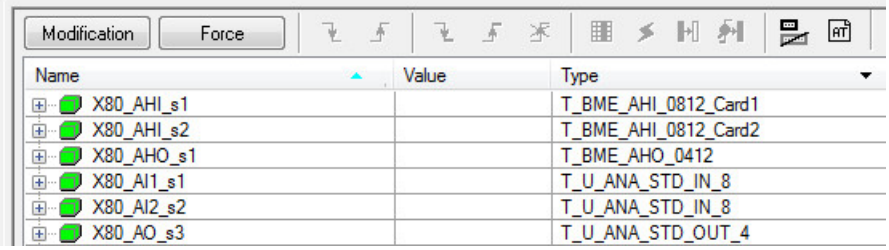
- Double-click in the field “Name “in order to display the button “…”:



- Select for example following variables in the animation table and click on the button “OK”:

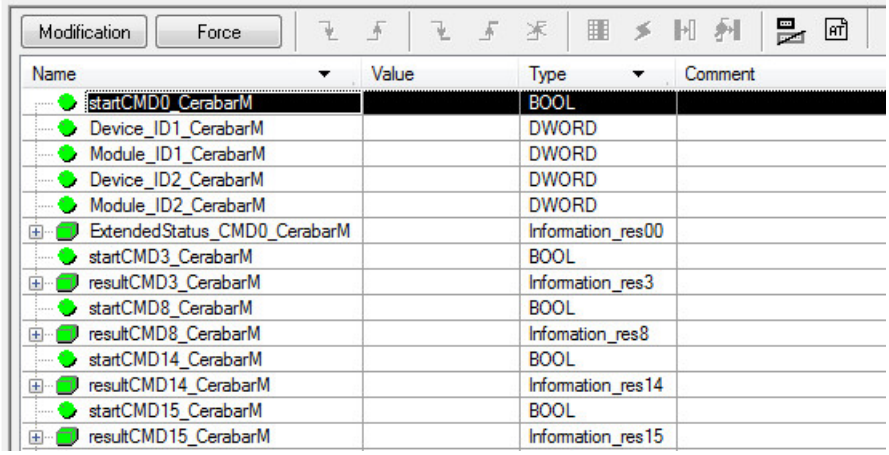


- Selected variables are added in the animation table:



- Another animation table can be configured for displaying HART Commands variables. In this example, the Animation Table is “SE02_HART_UniversalCommands”:

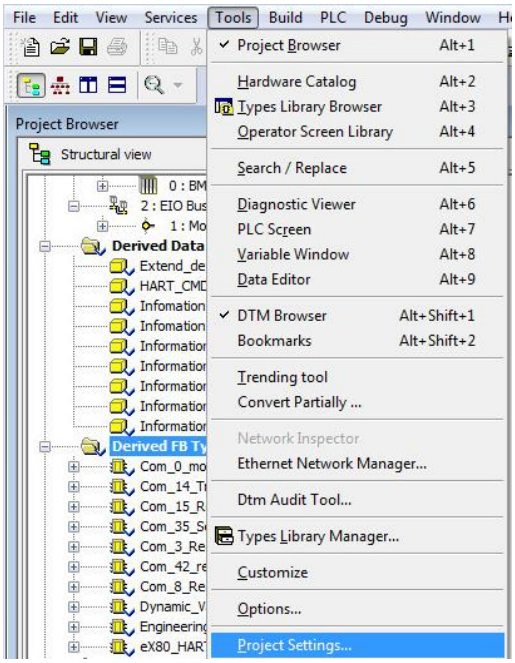
- Corresponding animation table contains function blocks start bit and output values for the Universal Hart Commands:



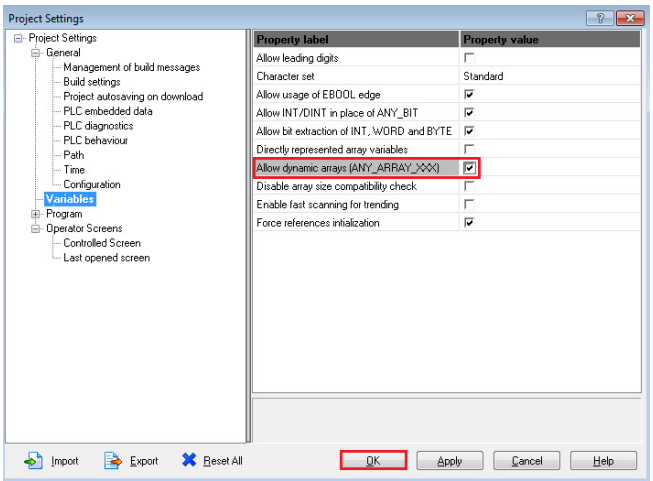
3.3 Commissioning of the Control Project

3.3.1 Project Settings

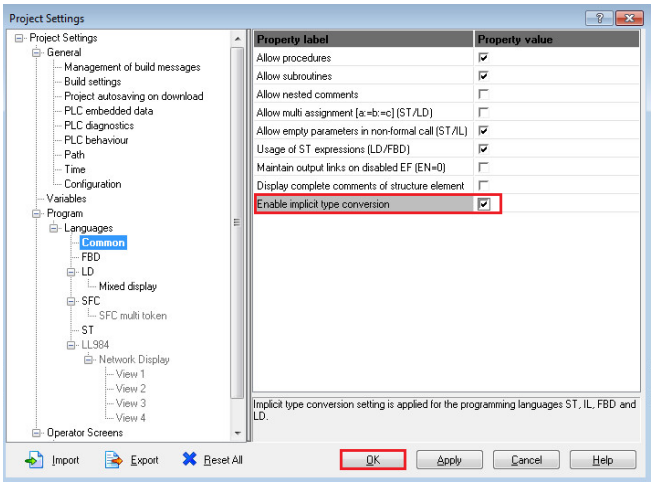
- Click on the menu “Tools→Project Settings”:



- Select the menu “Variables” and then the option “Allow dynamic arrays”. Then, click on the button “OK” to save:

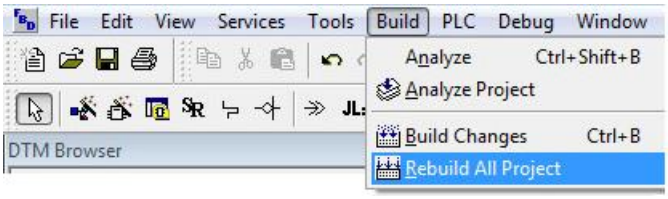


- Select the menu “Common” and then the option “Enable Implicit Data Conversion”. Click on the button “OK” to save:

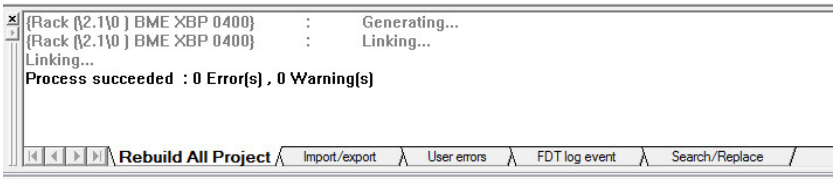


3.3.2 Project Compilation

- Select the menu “Build→Rebuild All Project”:

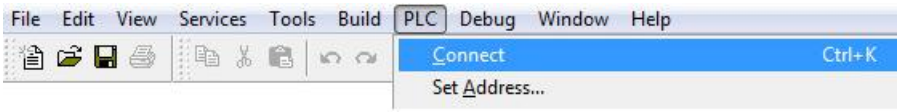


- Compilation is succesful:

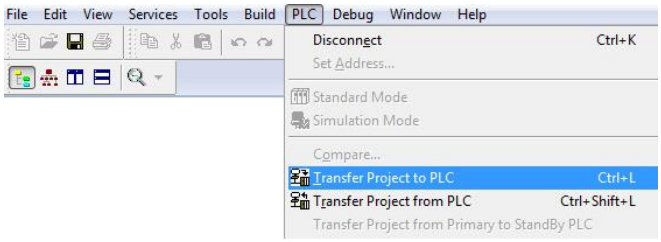


3.3.3 Project Download in PLC

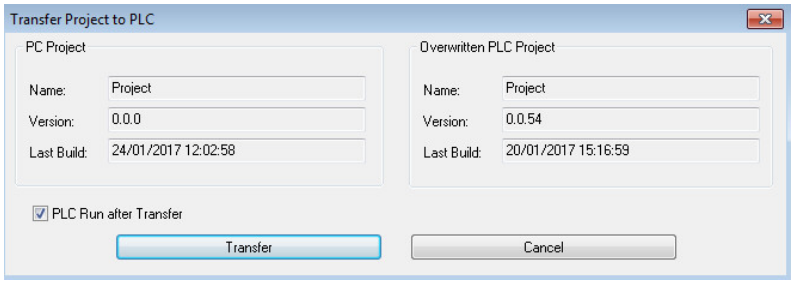
- Select the menu “PLC→Connect” in the tool bar:



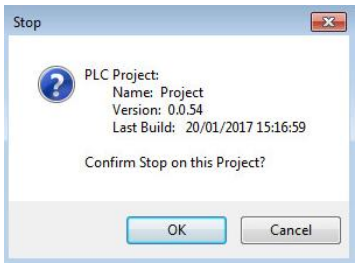
- Once connected, select the menu “PLC→Transfer Project to PLC”:



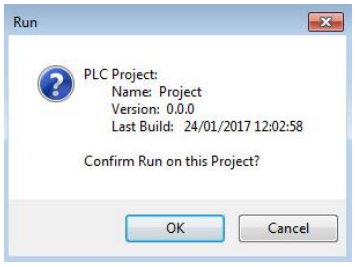
- Following Message Box is displayed. Select the option “PLC Run after Transfer” if needed:



- Following Message Box is displayed. Confirm by clicking on the button “OK:



- Following Message Box is displayed. Confirm by clicking on the button “OK”:



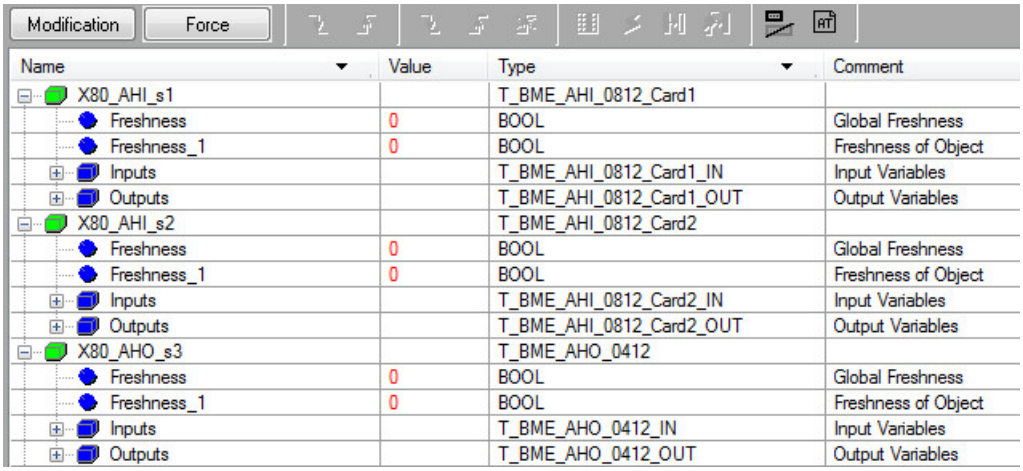
- The PLC is in run mode.

3.3.4 Modules Freshness Verification

- After a program download or PLC reboot, the PLC acts as a DHCP server and distributes the IP addresses to the HART modules. Make sure that no other DHCP server is active on the same time on the network. Otherwise, the HART modules cannot be initialized with their IP addresses and all HART modules statuses LEDs are “OFF” except the “Error” LEDs which are blinking.

- Always check the HART modules Freshness bits after a program download or reboot. The corresponding IO structure name variable containing the Freshness bits is defined in chapter 3.2.4:

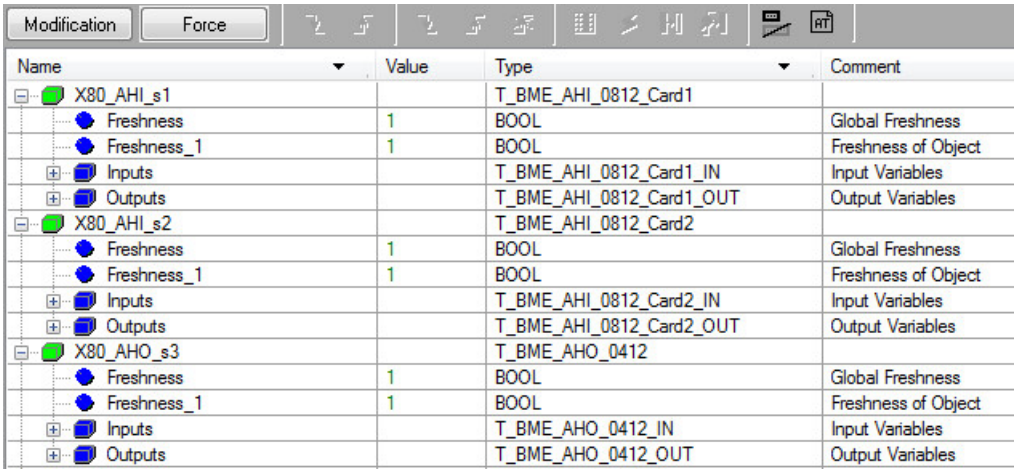
- Initialization with Errors (Freshness = Freshness_1 = 0):



Name	Value	Type	Comment
X80_AHI_s1		T_BME_AHI_0812_Card1	
Freshness	0	BOOL	Global Freshness
Freshness_1	0	BOOL	Freshness of Object
Inputs		T_BME_AHI_0812_Card1_IN	Input Variables
Outputs		T_BME_AHI_0812_Card1_OUT	Output Variables
X80_AHI_s2		T_BME_AHI_0812_Card2	
Freshness	0	BOOL	Global Freshness
Freshness_1	0	BOOL	Freshness of Object
Inputs		T_BME_AHI_0812_Card2_IN	Input Variables
Outputs		T_BME_AHI_0812_Card2_OUT	Output Variables
X80_AHO_s3		T_BME_AHO_0412	
Freshness	0	BOOL	Global Freshness
Freshness_1	0	BOOL	Freshness of Object
Inputs		T_BME_AHO_0412_IN	Input Variables
Outputs		T_BME_AHO_0412_OUT	Output Variables

In this example, another DHCP server was active and IO modules have not been successfully initialized. To fix this, disconnect the PLC from the supervisory network and wait few seconds for the HART modules initialization. Then reconnect the PLC to the supervisory network again.

- Good Initialization (Freshness = Freshness_1 = 1):



Name	Value	Type	Comment
X80_AHI_s1		T_BME_AHI_0812_Card1	
Freshness	1	BOOL	Global Freshness
Freshness_1	1	BOOL	Freshness of Object
Inputs		T_BME_AHI_0812_Card1_IN	Input Variables
Outputs		T_BME_AHI_0812_Card1_OUT	Output Variables
X80_AHI_s2		T_BME_AHI_0812_Card2	
Freshness	1	BOOL	Global Freshness
Freshness_1	1	BOOL	Freshness of Object
Inputs		T_BME_AHI_0812_Card2_IN	Input Variables
Outputs		T_BME_AHI_0812_Card2_OUT	Output Variables
X80_AHO_s3		T_BME_AHO_0412	
Freshness	1	BOOL	Global Freshness
Freshness_1	1	BOOL	Freshness of Object
Inputs		T_BME_AHO_0412_IN	Input Variables
Outputs		T_BME_AHO_0412_OUT	Output Variables

In this example, the initialization was successful and all modules are ready for use.

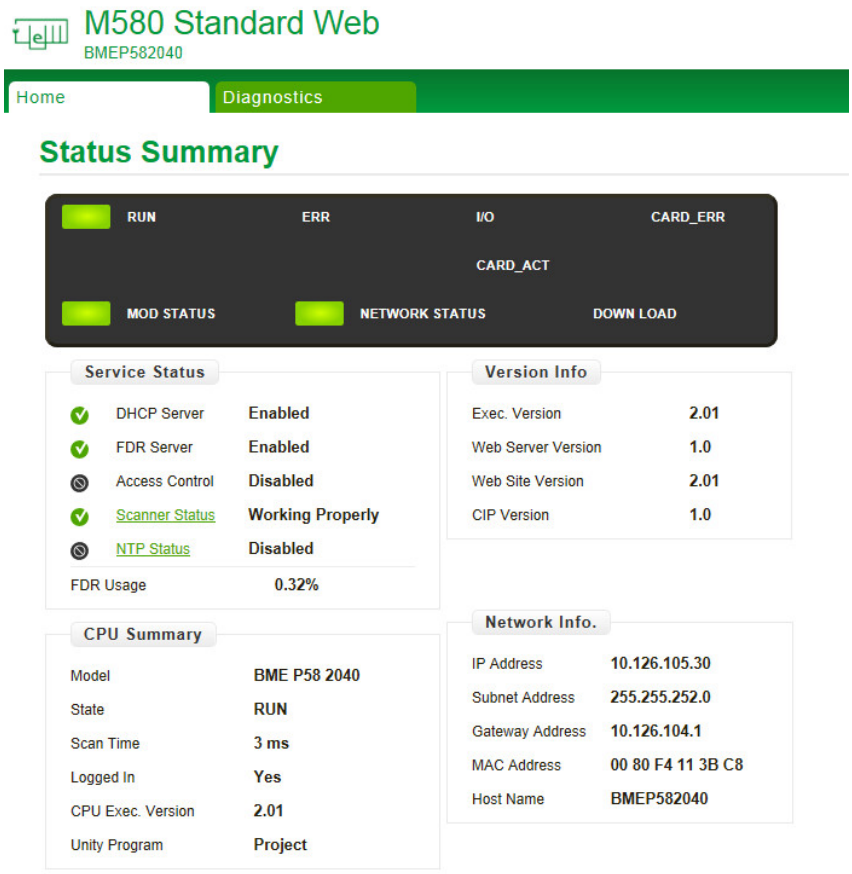
3.4 Monitoring of Process Values and Status Information

3.4.1 Diagnostics via Web Browser

- Open a web browser and enter the IP PLC IP address.
  In this example, the PLC IP address is 10.126.105.30:

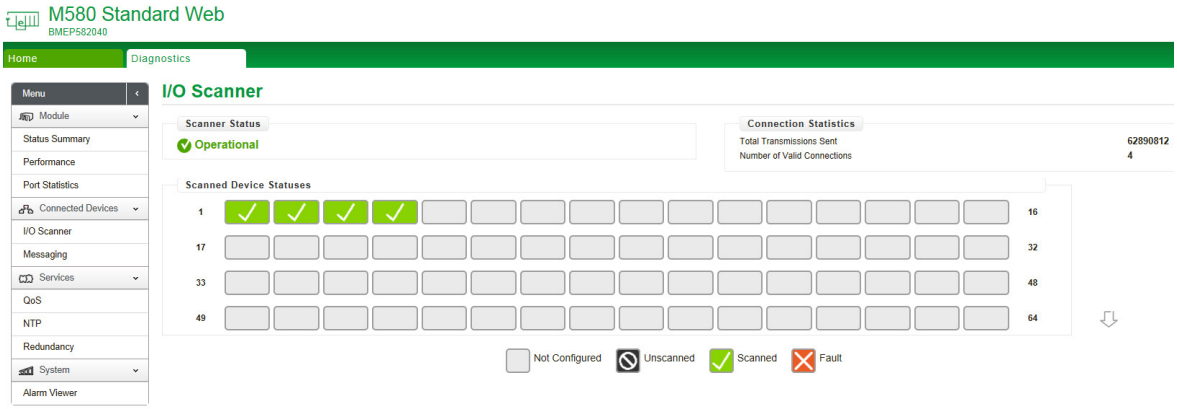
3.4.1.1 PLC & Network Diagnostics

- The M580 Standard Web page is displayed. This page Tag shows:
  - PLC status.
  - Version Info.
  - Network configuration.



3.4.1.2 I/O Scanner

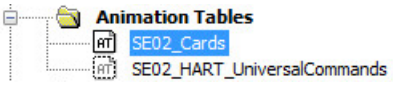
- Click on the Tag “Diagnostics” and select the menu “I/O Scanner”.
  In this example, four devices have been scanned:
  - The CRA module with IP address 10.126.105.32.
  - The AHI0812 module with IP address 10.126.105.55.
  - The AHI0812 module with IP address 10.126.105.56.
  - The AHO0412 module with IP address 10.126.105.57.



3.4.2 Animation Tables Monitoring

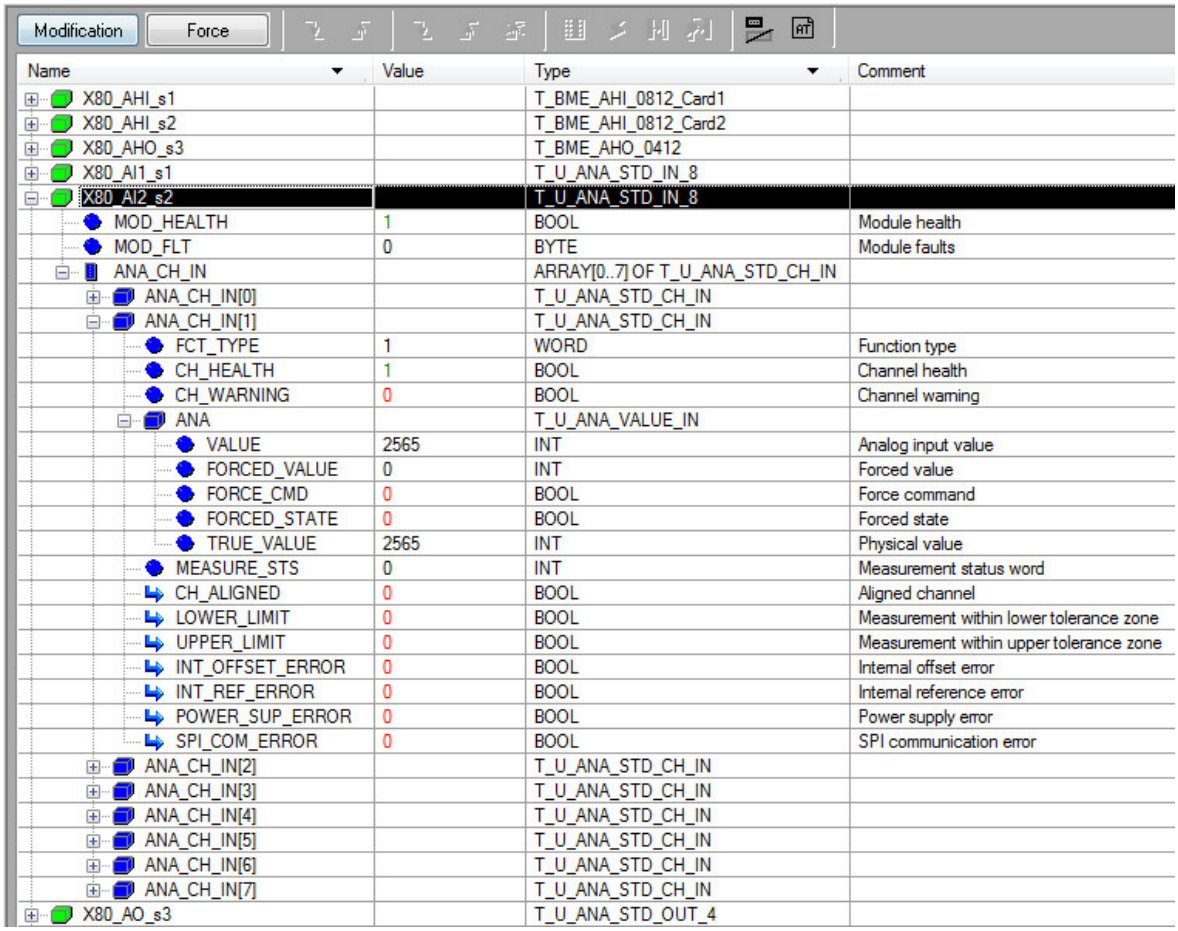
- Connect the PLC by clicking on the shortcut button:

- Double-click on the animation table “SE02_Cards”:



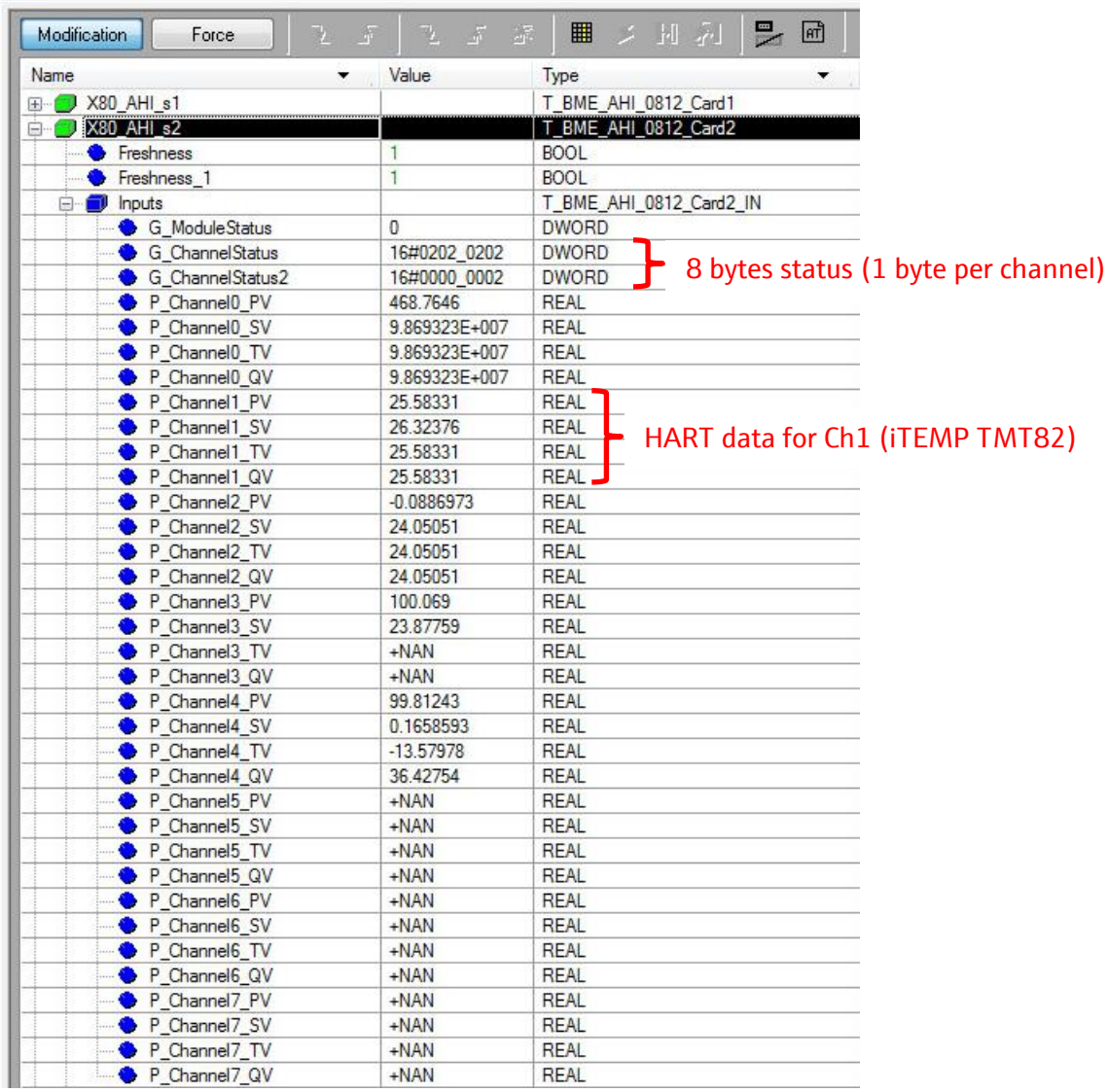
3.4.2.1 Analog Input Data

- Expand for example the data structure “X80_AI_s2 for displaying all specific channel data for the second configured HART analog input module:



Name	Value	Type	Comment
X80_AHI_s1		T_BME_AHI_0812_Card1	
X80_AHI_s2		T_BME_AHI_0812_Card2	
X80_AHO_s3		T_BME_AHO_0412	
X80_AI1_s1		T_U_ANA_STD_IN_8	
X80_AI2_s2		T_U_ANA_STD_IN_8	
MOD_HEALTH	1	BOOL	Module health
MOD_FLT	0	BYTE	Module faults
ANA_CH_IN		ARRAY[0..7] OF T_U_ANA_STD_CH_IN	
ANA_CH_IN[0]		T_U_ANA_STD_CH_IN	
ANA_CH_IN[1]		T_U_ANA_STD_CH_IN	
FCT_TYPE	1	WORD	Function type
CH_HEALTH	1	BOOL	Channel health
CH_WARNING	0	BOOL	Channel warning
ANA		T_U_ANA_VALUE_IN	
VALUE	2565	INT	Analog input value
FORCED_VALUE	0	INT	Forced value
FORCE_CMD	0	BOOL	Force command
FORCED_STATE	0	BOOL	Forced state
TRUE_VALUE	2565	INT	Physical value
MEASURE_STS	0	INT	Measurement status word
CH_ALIGNED	0	BOOL	Aligned channel
LOWER_LIMIT	0	BOOL	Measurement within lower tolerance zone
UPPER_LIMIT	0	BOOL	Measurement within upper tolerance zone
INT_OFFSET_ERROR	0	BOOL	Internal offset error
INT_REF_ERROR	0	BOOL	Internal reference error
POWER_SUP_ERROR	0	BOOL	Power supply error
SPI_COM_ERROR	0	BOOL	SPI communication error
ANA_CH_IN[2]		T_U_ANA_STD_CH_IN	
ANA_CH_IN[3]		T_U_ANA_STD_CH_IN	
ANA_CH_IN[4]		T_U_ANA_STD_CH_IN	
ANA_CH_IN[5]		T_U_ANA_STD_CH_IN	
ANA_CH_IN[6]		T_U_ANA_STD_CH_IN	
ANA_CH_IN[7]		T_U_ANA_STD_CH_IN	
X80_AO_s3		T_U_ANA_STD_OUT_4	

- Expand for example the data structure "X80_AHI_s2 for displaying all HART data per channel for the second configured HART analog input module:



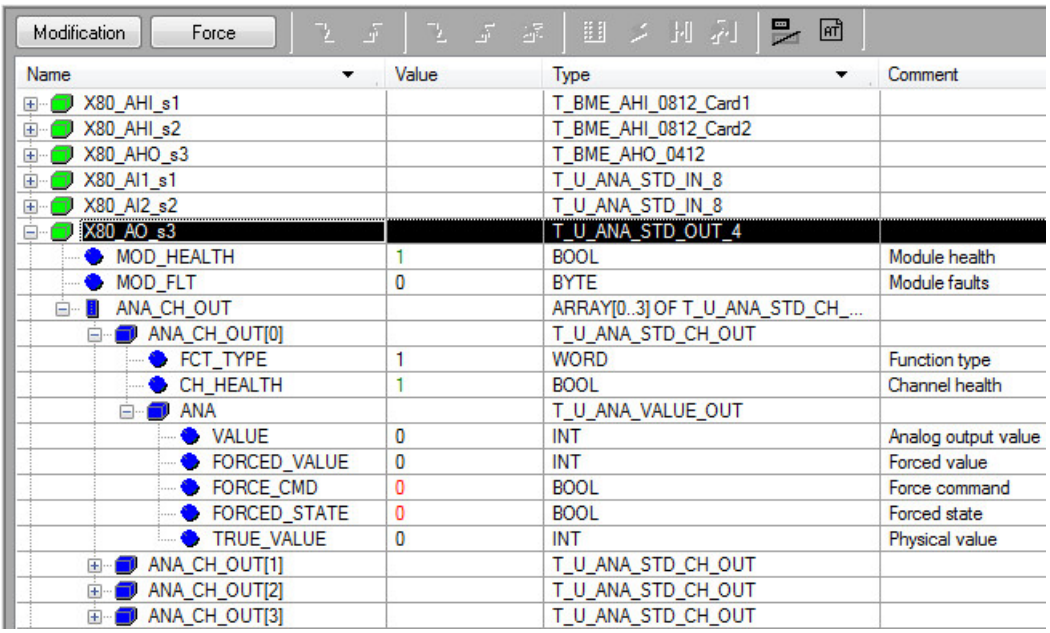
Name	Value	Type
X80_AHI_s1		T_BME_AHI_0812_Card1
X80_AHI_s2		T_BME_AHI_0812_Card2
Freshness	1	BOOL
Freshness_1	1	BOOL
Inputs		T_BME_AHI_0812_Card2_IN
G_ModuleStatus	0	DWORD
G_ChannelStatus	16#0202_0202	DWORD
G_ChannelStatus2	16#0000_0002	DWORD
P_Channel0_PV	468.7646	REAL
P_Channel0_SV	9.869323E+007	REAL
P_Channel0_TV	9.869323E+007	REAL
P_Channel0_QV	9.869323E+007	REAL
P_Channel1_PV	25.58331	REAL
P_Channel1_SV	26.32376	REAL
P_Channel1_TV	25.58331	REAL
P_Channel1_QV	25.58331	REAL
P_Channel2_PV	-0.0886973	REAL
P_Channel2_SV	24.05051	REAL
P_Channel2_TV	24.05051	REAL
P_Channel2_QV	24.05051	REAL
P_Channel3_PV	100.069	REAL
P_Channel3_SV	23.87759	REAL
P_Channel3_TV	+NAN	REAL
P_Channel3_QV	+NAN	REAL
P_Channel4_PV	99.81243	REAL
P_Channel4_SV	0.1658593	REAL
P_Channel4_TV	-13.57978	REAL
P_Channel4_QV	36.42754	REAL
P_Channel5_PV	+NAN	REAL
P_Channel5_SV	+NAN	REAL
P_Channel5_TV	+NAN	REAL
P_Channel5_QV	+NAN	REAL
P_Channel6_PV	+NAN	REAL
P_Channel6_SV	+NAN	REAL
P_Channel6_TV	+NAN	REAL
P_Channel6_QV	+NAN	REAL
P_Channel7_PV	+NAN	REAL
P_Channel7_SV	+NAN	REAL
P_Channel7_TV	+NAN	REAL
P_Channel7_QV	+NAN	REAL

- Each channel has its own status. In this example, all enabled channels (Ch0, Ch1, Ch2, Ch3 and Ch4) have the status value "0x02", this means that the channel is connected to the HART device. All disabled channels (Ch5, Ch6 and Ch7) have the status value "0x00".

3.4.2.2 Analog Output Data

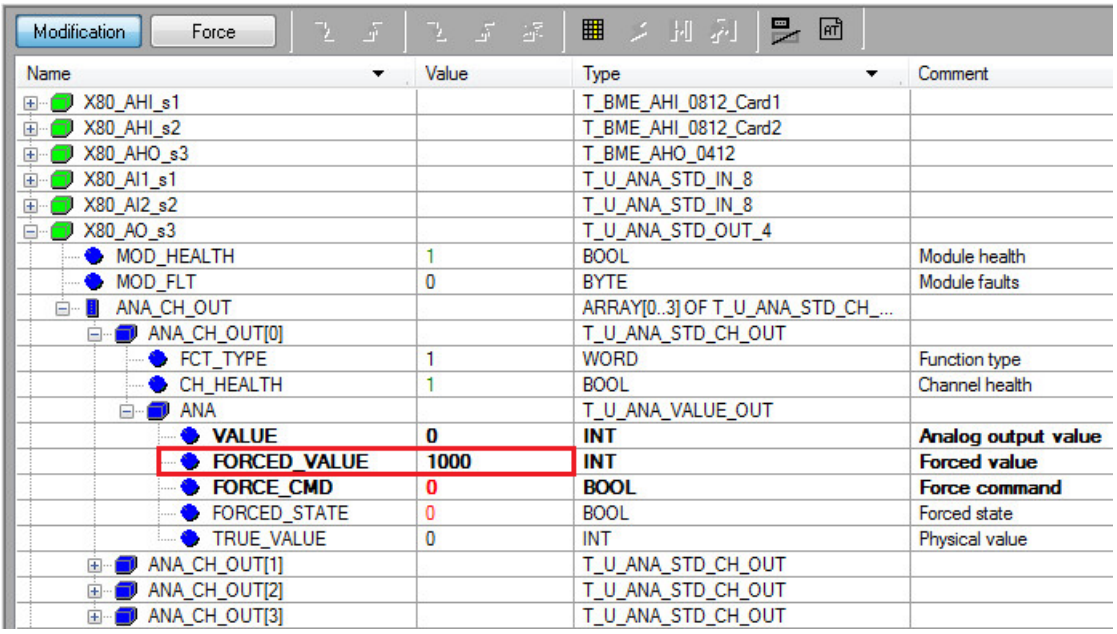
In this example, an AUMA actuator is connected on Ch0 of the analog output card. The target is to send a 4...20mA command to the AUMA actuator and to read back the feedback via the HART data.

- Expand for example the data structure "X80_AO_s3" for displaying all HART data per channel for the configured HART analog output module:



Name	Value	Type	Comment
X80_AHI_s1		T_BME_AHI_0812_Card1	
X80_AHI_s2		T_BME_AHI_0812_Card2	
X80_AHO_s3		T_BME_AHO_0412	
X80_AI1_s1		T_U_ANA_STD_IN_8	
X80_AI2_s2		T_U_ANA_STD_IN_8	
X80_AO_s3		T_U_ANA_STD_OUT_4	
MOD_HEALTH	1	BOOL	Module health
MOD_FLT	0	BYTE	Module faults
ANA_CH_OUT		ARRAY[0..3] OF T_U_ANA_STD_CH_...	
ANA_CH_OUT[0]		T_U_ANA_STD_CH_OUT	
FCT_TYPE	1	WORD	Function type
CH_HEALTH	1	BOOL	Channel health
ANA		T_U_ANA_VALUE_OUT	
VALUE	0	INT	Analog output value
FORCED_VALUE	0	INT	Forced value
FORCE_CMD	0	BOOL	Force command
FORCED_STATE	0	BOOL	Forced state
TRUE_VALUE	0	INT	Physical value
ANA_CH_OUT[1]		T_U_ANA_STD_CH_OUT	
ANA_CH_OUT[2]		T_U_ANA_STD_CH_OUT	
ANA_CH_OUT[3]		T_U_ANA_STD_CH_OUT	

- Click on the button "Modification" and enter a set point value in the field "FORCED_VALUE". This set point value is of course depending on the defined scaling range. In this example, the set point is 10% (Scaling Range is 0 to 100%):



Name	Value	Type	Comment
X80_AHI_s1		T_BME_AHI_0812_Card1	
X80_AHI_s2		T_BME_AHI_0812_Card2	
X80_AHO_s3		T_BME_AHO_0412	
X80_AI1_s1		T_U_ANA_STD_IN_8	
X80_AI2_s2		T_U_ANA_STD_IN_8	
X80_AO_s3		T_U_ANA_STD_OUT_4	
MOD_HEALTH	1	BOOL	Module health
MOD_FLT	0	BYTE	Module faults
ANA_CH_OUT		ARRAY[0..3] OF T_U_ANA_STD_CH_...	
ANA_CH_OUT[0]		T_U_ANA_STD_CH_OUT	
FCT_TYPE	1	WORD	Function type
CH_HEALTH	1	BOOL	Channel health
ANA		T_U_ANA_VALUE_OUT	
VALUE	**0**	**INT**	**Analog output value**
FORCED_VALUE	**1000**	**INT**	**Forced value**
FORCE_CMD	**0**	**BOOL**	**Force command**
FORCED_STATE	0	BOOL	Forced state
TRUE_VALUE	0	INT	Physical value
ANA_CH_OUT[1]		T_U_ANA_STD_CH_OUT	
ANA_CH_OUT[2]		T_U_ANA_STD_CH_OUT	
ANA_CH_OUT[3]		T_U_ANA_STD_CH_OUT	

- Send the set point by setting the bit "FORCE_CMD" to TRUE:

Modification | Force

Name	Value	Type	Comment
X80_AHI_s1		T_BME_AHI_0812_Card1	
X80_AHI_s2		T_BME_AHI_0812_Card2	
X80_AHO_s3		T_BME_AHO_0412	
X80_AI1_s1		T_U_ANA_STD_IN_8	
X80_AI2_s2		T_U_ANA_STD_IN_8	
X80_AO_s3		T_U_ANA_STD_OUT_4	
MOD_HEALTH	1	BOOL	Module health
MOD_FLT	0	BYTE	Module faults
ANA_CH_OUT		ARRAY[0..3] OF T_U_ANA_STD_CH_...	
ANA_CH_OUT[0]		T_U_ANA_STD_CH_OUT	
FCT_TYPE	1	WORD	Function type
CH_HEALTH	1	BOOL	Channel health
ANA		T_U_ANA_VALUE_OUT	
VALUE	**1000**	**INT**	**Analog output value**
FORCED_VALUE	**1000**	**INT**	**Forced value**
FORCE_CMD	1	BOOL	Force command
FORCED_STATE	1	BOOL	Forced state
TRUE_VALUE	1000	INT	Physical value
ANA_CH_OUT[1]		T_U_ANA_STD_CH_OUT	
ANA_CH_OUT[2]		T_U_ANA_STD_CH_OUT	
ANA_CH_OUT[3]		T_U_ANA_STD_CH_OUT	

- Expand for example the data structure "X80_AHO_s3" for displaying all specific channel data for the configured HART analog output module on which is connected the AUMA actuator:

Modification | Force

Name	Value	Type	Comment
X80_AHI_s1		T_BME_AHI_0812_Card1	
X80_AHI_s2		T_BME_AHI_0812_Card2	
X80_AHO_s3		T_BME_AHO_0412	
Freshness	1	BOOL	Global Freshness
Freshness_1	1	BOOL	Freshness of Object
Inputs		T_BME_AHO_0412_IN	Input Variables
G_ModuleStatus	3	DWORD	
G_ChannelStatus	16#0505_0502	DWORD	4 bytes status (1 byte per channel)
P_Channel0_PV	9.9	REAL	
P_Channel0_SV	10.7	REAL	HART data for Ch0 (AUMA Actuator)
P_Channel0_TV	0.0	REAL	
P_Channel0_QV	0.0	REAL	
P_Channel1_PV	+NAN	REAL	
P_Channel1_SV	+NAN	REAL	
P_Channel1_TV	+NAN	REAL	
P_Channel1_QV	+NAN	REAL	
P_Channel2_PV	+NAN	REAL	
P_Channel2_SV	+NAN	REAL	
P_Channel2_TV	+NAN	REAL	
P_Channel2_QV	+NAN	REAL	
P_Channel3_PV	+NAN	REAL	
P_Channel3_SV	+NAN	REAL	
P_Channel3_TV	+NAN	REAL	
P_Channel3_QV	+NAN	REAL	
Outputs		T_BME_AHO_0412_OUT	Output Variables
X80_AI1_s1		T_U_ANA_STD_IN_8	
X80_AI2_s2		T_U_ANA_STD_IN_8	
X80_AO_s3		T_U_ANA_STD_OUT_4	

- Each channel has its own status. In this example, all channels are enabled but only Ch0 is connected (ChannelStatus=0x02), this means that the channel is connected to the HART device. The other channels are enabled but no devices are connected (ChannelStatus=0x05).

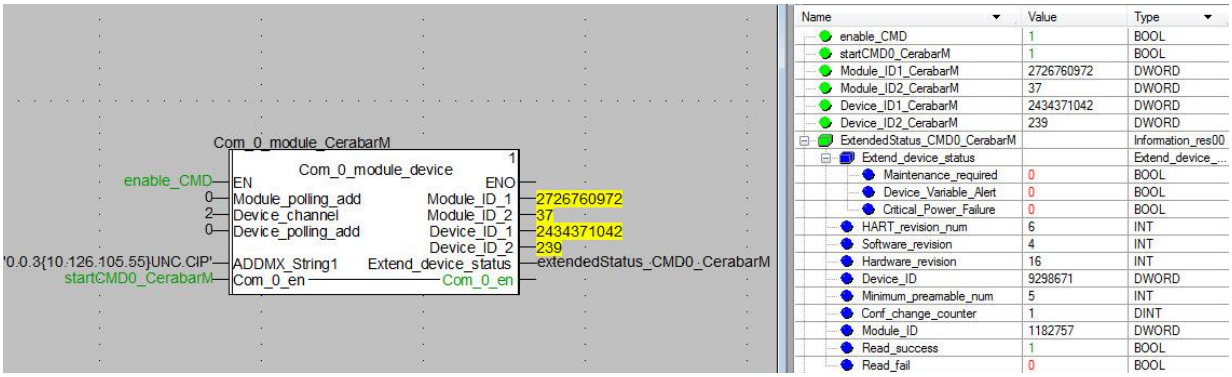
  In this example, the PV value “9.9” corresponds to the feedback value set point and the SV value “10.7%”corresponds to the real position of the actuator.

3.4.3 Monitoring of HART Process Variables and Commands

Sending HART Commands from the logic can be successfully executed if no other online connections are running in parallel on the same HART card, as via DTM in the DTM Browser or in FieldCare.

3.4.3.1 HART CMD 0: Read unique Identifier

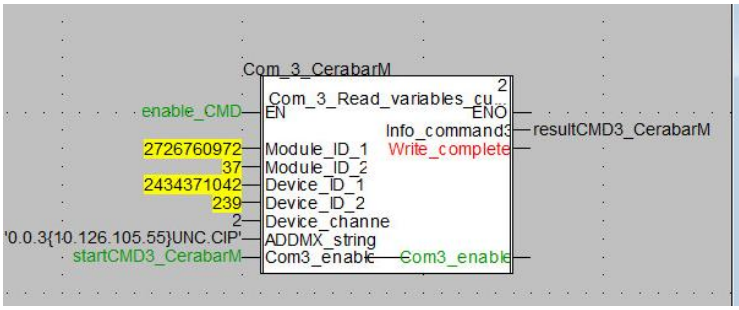
- Function block HART CMD0 in online mode:
  The start bit “startCMD0_CerabarM” must be triggered twice:



Name	Value	Type
enable_CMD	1	BOOL
startCMD0_CerabarM	1	BOOL
Module_ID1_CerabarM	2726760972	DWORD
Module_ID2_CerabarM	37	DWORD
Device_ID1_CerabarM	2434371042	DWORD
Device_ID2_CerabarM	239	DWORD
ExtendedStatus_CMD0_CerabarM		Information_res00
Extend_device_status		Extend_device_...
Maintenance_required	0	BOOL
Device_Variable_Alert	0	BOOL
Critical_Power_Failure	0	BOOL
HART_revision_num	6	INT
Software_revision	4	INT
Hardware_revision	16	INT
Device_ID	9298671	DWORD
Minimum_preamable_num	5	INT
Conf_change_counter	1	DINT
Module_ID	1182757	DWORD
Read_success	1	BOOL
Read_fail	0	BOOL

3.4.3.2 HART CMD 3: Read dynamic variables and loop current

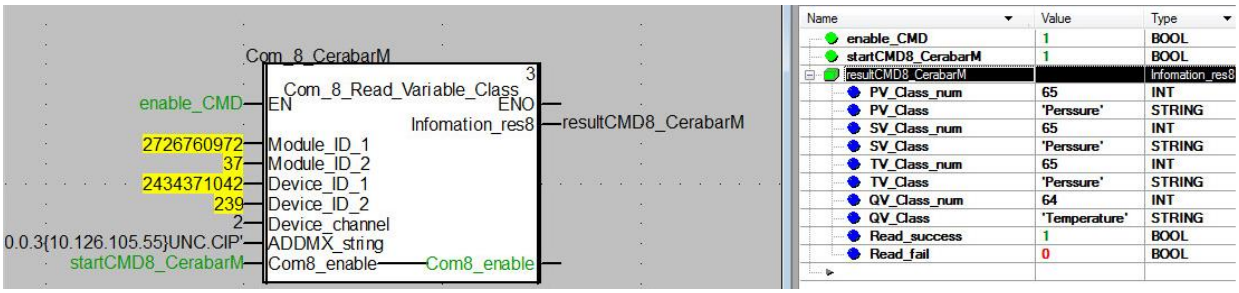
- Function block HART CMD3 in online mode:
  The start bit “startCMD3_CerabarM” is set to TRUE and must be set to FALSE manually:



Name	Value	Type
enable_CMD	1	BOOL
startCMD3_CerabarM	1	BOOL
resultCMD3_CerabarM		Information_res3
PV_loopcurrent	3.980576	REAL
PV_unit	'Bars'	STRING
PV_value	-0.002427971	REAL
SV_unit	'Bars'	STRING
SV_value	-0.002435188	REAL
TV_unit	'Bars'	STRING
TV_value	-0.0003711804	REAL
QV_unit	'Deg C'	STRING
QV_value	25.16937	REAL
Read_success	1	BOOL
Read_fail	0	BOOL

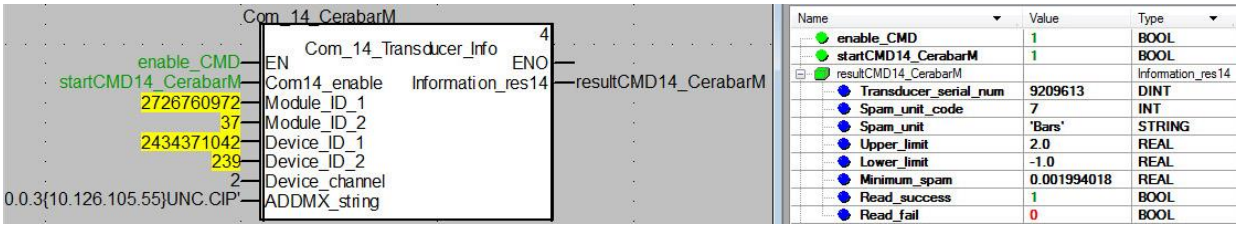
3.4.3.3 HART CMD 8: Read dynamic variable classifications (from HART6)

- Function block HART CMD8 in online mode:
  The start bit "startCMD8_CerabarM" is set to TRUE and must be set to FALSE manually:



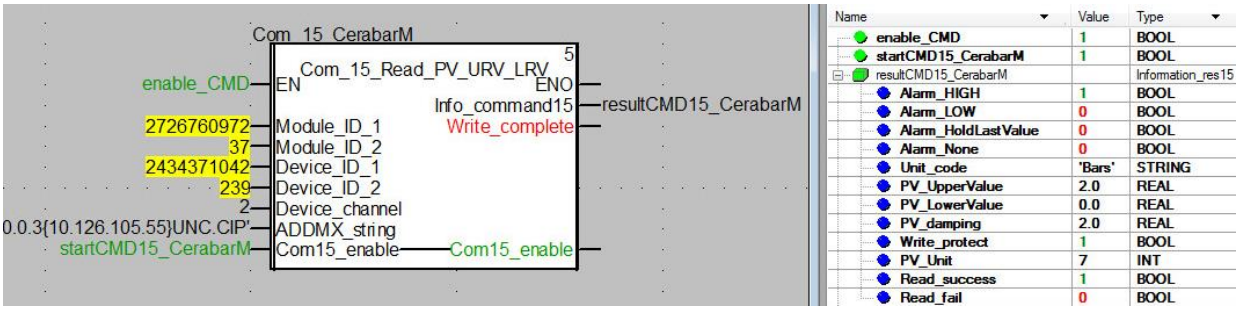
3.4.3.4 HART CMD 14: Read primary variable sensor information

- Function block HART CMD14 in online mode:
  The start bit "startCMD14_CerabarM" is set to TRUE and must be set to FALSE manually:



3.4.3.5 HART CMD 15: Read primary variable output information

- Function block HART CMD15 in online mode:
  The start bit "startCMD15_CerabarM" is set to TRUE and must be set to FALSE manually:



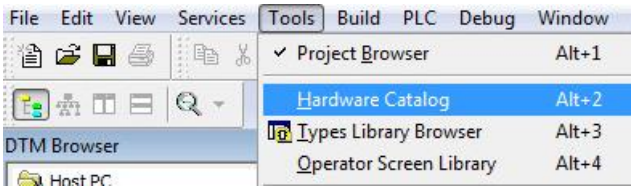
4 Advanced Integration

The Advanced Integration consists in using Endress+Hauser deviceDTMs in Unity Pro DTM Browser.

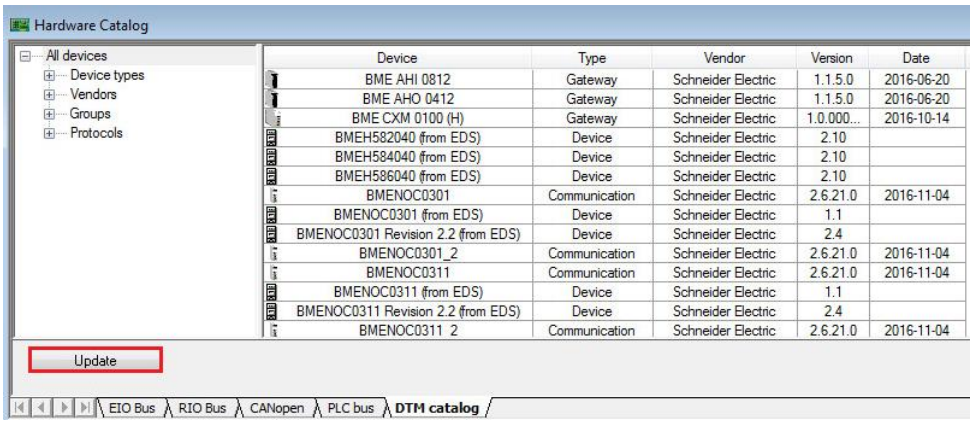
4.1 Device DTMs Library

- Install the Endress+Hauser HART DTM Library V2.44.00.

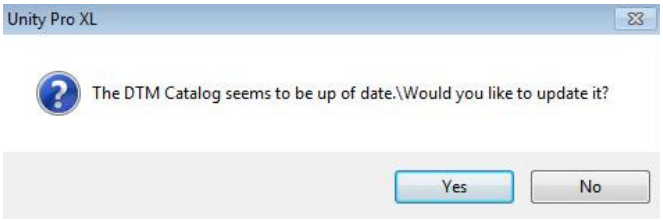
- Open the Hardware Catalog by clicking on the menu “Tools→Hardware Catalog”:



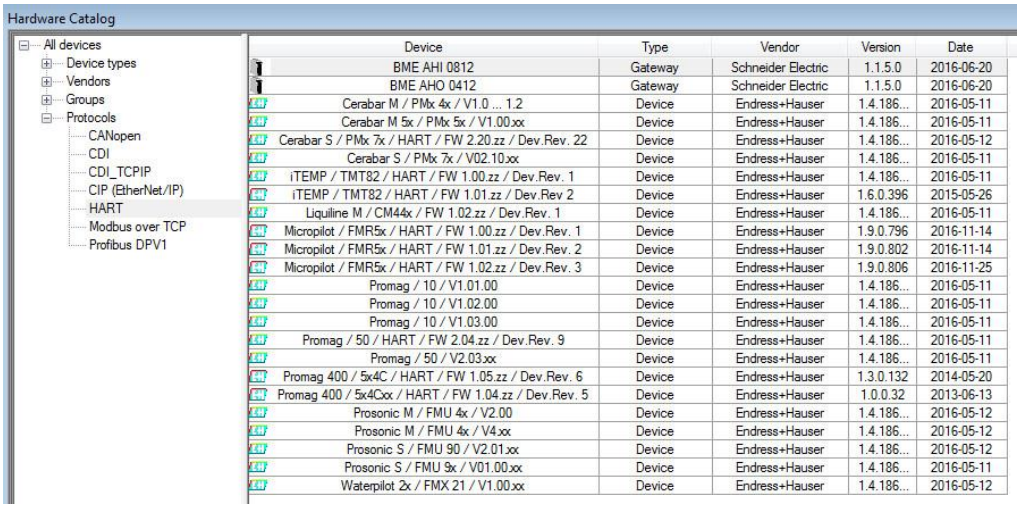
- Click on the button “Update” to update the DTM catalog database:



- Following Message Box is displayed. Click on the button “Yes”:



- Endress+Hauser HART deviceDTMs are now installed:



Device	Type	Vendor	Version	Date
BME AHI 0812	Gateway	Schneider Electric	1.1.5.0	2016-06-20
BME AHO 0412	Gateway	Schneider Electric	1.1.5.0	2016-06-20
Cerabar M / PMx 4x / V1.0 ... 1.2	Device	Endress+Hauser	1.4.186...	2016-05-11
Cerabar M 5x / PMx 5x / V1.00.xx	Device	Endress+Hauser	1.4.186...	2016-05-11
Cerabar S / PMx 7x / HART / FW 2.20.zz / Dev.Rev. 22	Device	Endress+Hauser	1.4.186...	2016-05-12
Cerabar S / PMx 7x / V02.10.xx	Device	Endress+Hauser	1.4.186...	2016-05-11
iTEMP / TMT82 / HART / FW 1.00.zz / Dev.Rev. 1	Device	Endress+Hauser	1.4.186...	2016-05-11
iTEMP / TMT82 / HART / FW 1.01.zz / Dev.Rev 2	Device	Endress+Hauser	1.6.0.396	2015-05-26
Liquiline M / CM44x / FW 1.02.zz / Dev.Rev. 1	Device	Endress+Hauser	1.4.186...	2016-05-11
Micropilot / FMR5x / HART / FW 1.00.zz / Dev.Rev. 1	Device	Endress+Hauser	1.9.0.796	2016-11-14
Micropilot / FMR5x / HART / FW 1.01.zz / Dev.Rev. 2	Device	Endress+Hauser	1.9.0.802	2016-11-14
Micropilot / FMR5x / HART / FW 1.02.zz / Dev.Rev. 3	Device	Endress+Hauser	1.9.0.806	2016-11-25
Promag / 10 / V1.01.00	Device	Endress+Hauser	1.4.186...	2016-05-11
Promag / 10 / V1.02.00	Device	Endress+Hauser	1.4.186...	2016-05-11
Promag / 10 / V1.03.00	Device	Endress+Hauser	1.4.186...	2016-05-11
Promag / 50 / HART / FW 2.04.zz / Dev.Rev. 9	Device	Endress+Hauser	1.4.186...	2016-05-11
Promag / 50 / V2.03.xx	Device	Endress+Hauser	1.4.186...	2016-05-11
Promag 400 / 5x4C / HART / FW 1.05.zz / Dev.Rev. 6	Device	Endress+Hauser	1.3.0.132	2014-05-20
Promag 400 / 5x4Cxx / HART / FW 1.04.zz / Dev.Rev. 5	Device	Endress+Hauser	1.0.0.32	2013-06-13
Prosonic M / FMU 4x / V2.00	Device	Endress+Hauser	1.4.186...	2016-05-12
Prosonic M / FMU 4x / V4.xx	Device	Endress+Hauser	1.4.186...	2016-05-12
Prosonic S / FMU 90 / V2.01.xx	Device	Endress+Hauser	1.4.186...	2016-05-12
Prosonic S / FMU 9x / V01.00.xx	Device	Endress+Hauser	1.4.186...	2016-05-11
Waterpilot 2x / FMX 21 / V1.00.xx	Device	Endress+Hauser	1.4.186...	2016-05-12

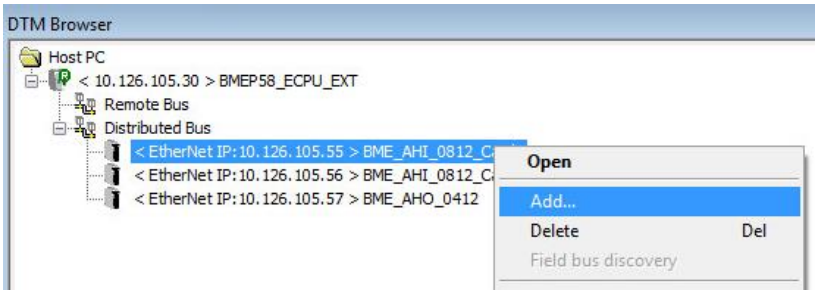
- All imported device DTMs are reasonably assigned to predefined folders :
  - Device types → Devices
  - Vendors → Endress+Hauser
  - Groups →DTM specific
  - Groups → Electromechanical Analyser
  - Groups → Flow
  - Groups → Level
  - Groups → Pressure
  - Groups → Temperature
  - Protocols → HART

4.2 Field Device DTM

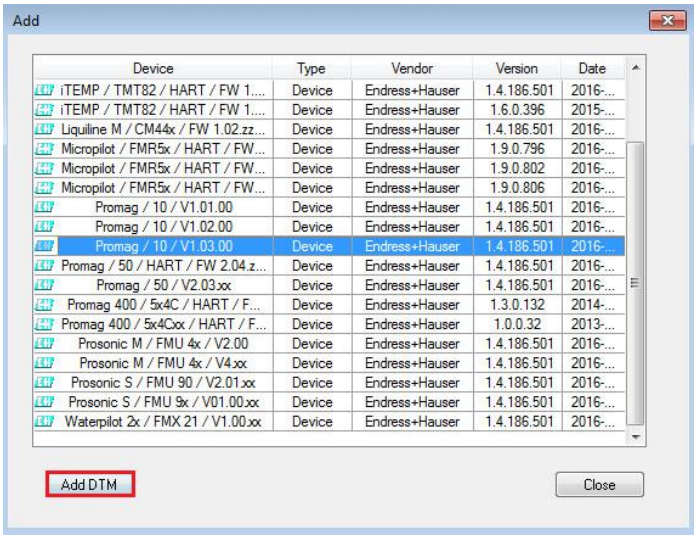
A new device can be added manually or by using the function “Fieldbus discovery”.

4.2.1 New Field Device added Manually

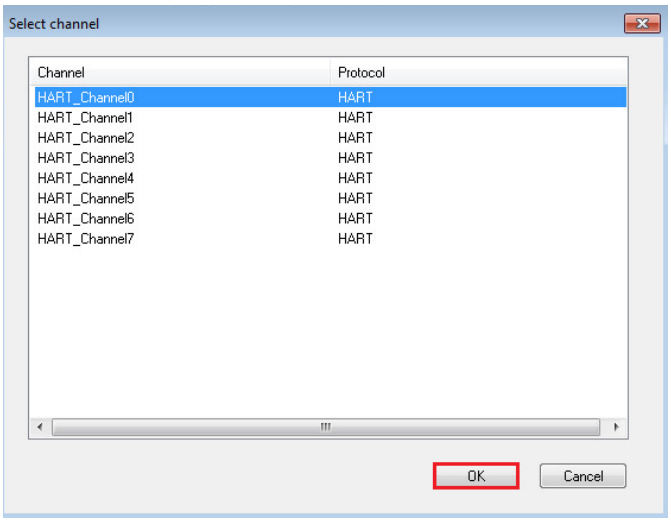
- In the DTM Browser, right-click on the DTM “BME_AHI_0812_Card1” and select the menu “Add…”:



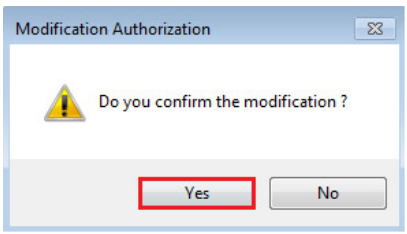
- Select the DTM “Promag / 10 / V1.03.00” for this example and click on the button “AddDTM”:



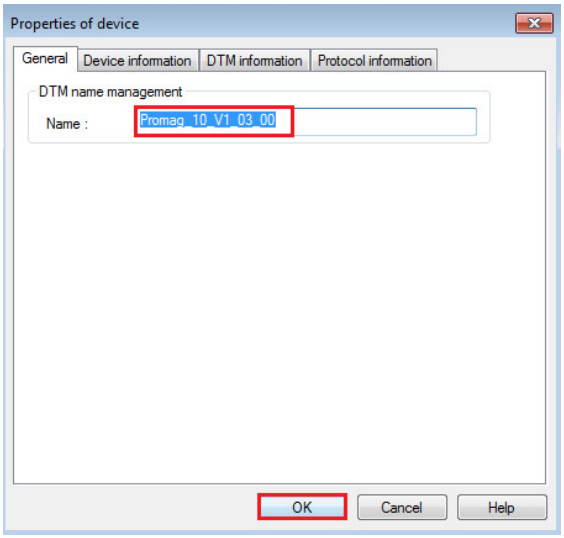
- Select the channel on which is connected the device, “HART_Channel0” for this example and click on the button “OK”:



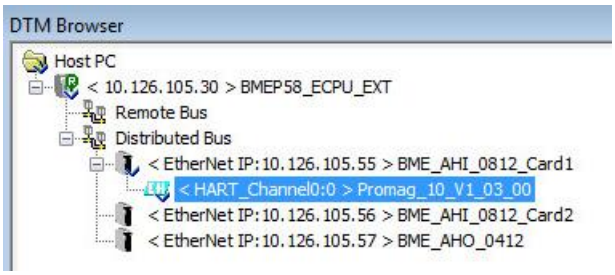
- Confirm by clicking on the button “Yes”:



- Default DTM name is displayed. Change it if needed and click on the button “OK”.
  In this example the project DTM name is the default one:

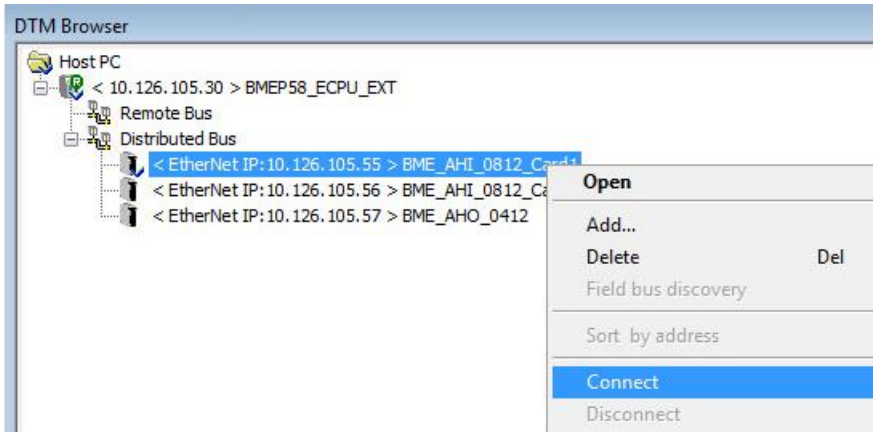


- The deviceDTM is successfully inserted in the DTM Browser:

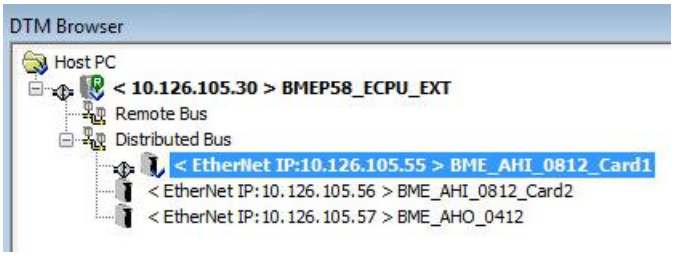


4.2.2 New Field Device added with the FieldBus Scanner

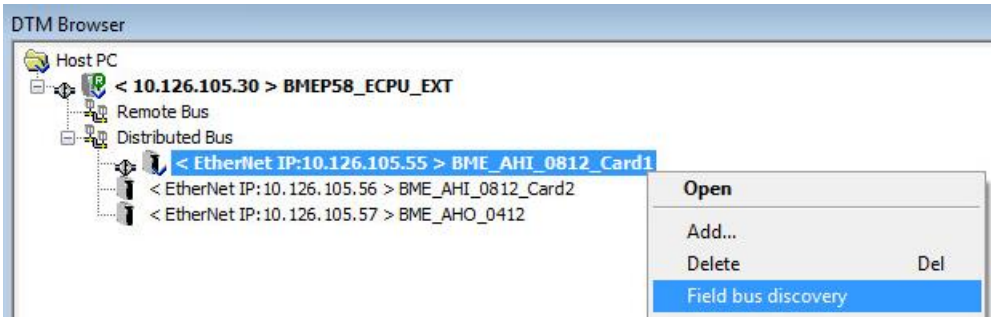
- Right-click on the DTM “BME_AHI_0812_Card1” and select the option “Connect”:



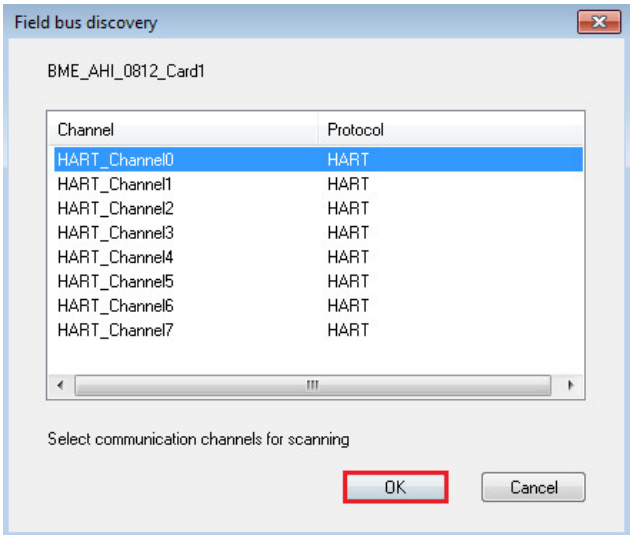
- Symbol indicates that the deviceDTM is successfully connected:



- Right-click on the deviceDTM and select the menu “Field bus discovery”:

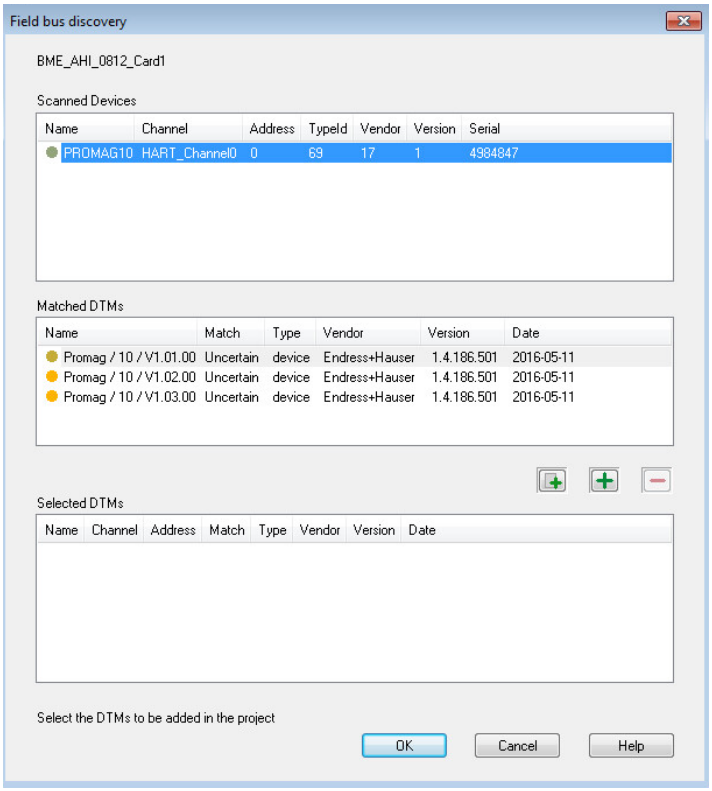


- Select the channel which has to be scanned and click on the button “OK”.
  Channel 0 is selected in this example:

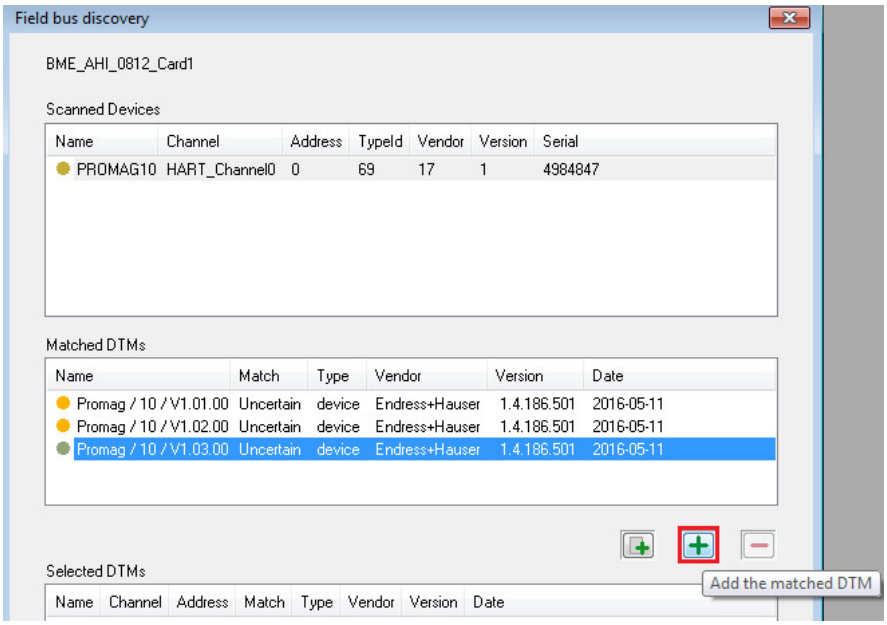


All channels cannot be selected for scanning all devices connected on the card.

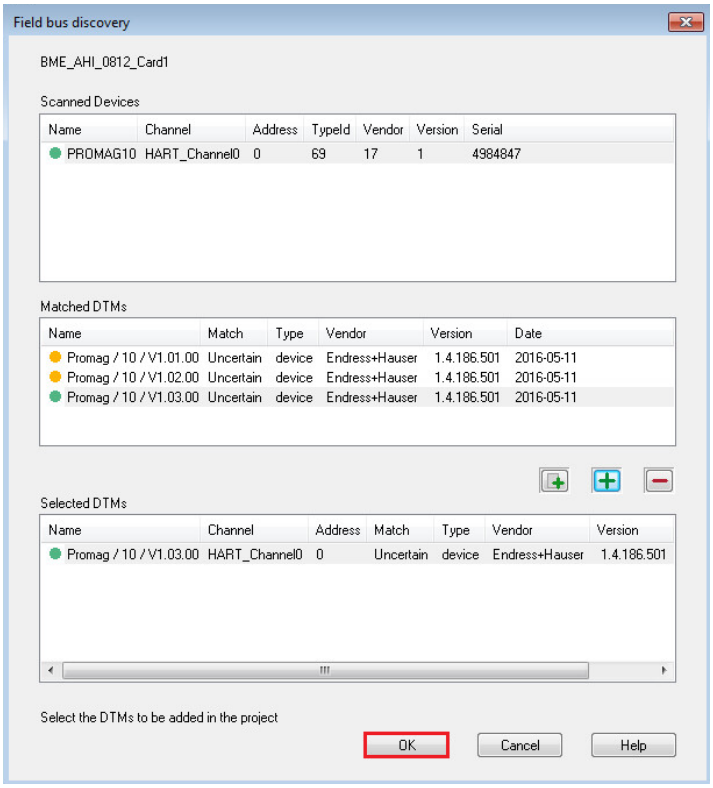
- Following window appears. The scanner founds the Promag10 and displays the matched DTMs:



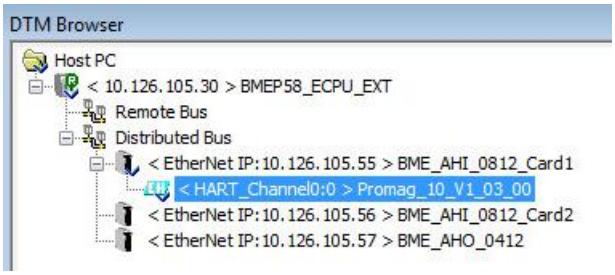
- The correct deviceDTM revision must be selected manually.
  In this example, the deviceDTM “Promag / 10 /V1.03.00” is selected. Click on the shortcut button “Add the matched DTM”:



- Click on the button “OK” to close the window:

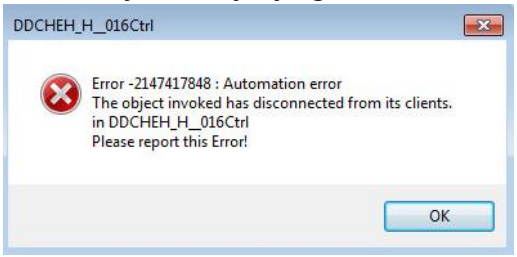


- The deviceDTM is successfully inserted in the DTM Browser:



4.3 Data Execution Prevention Option

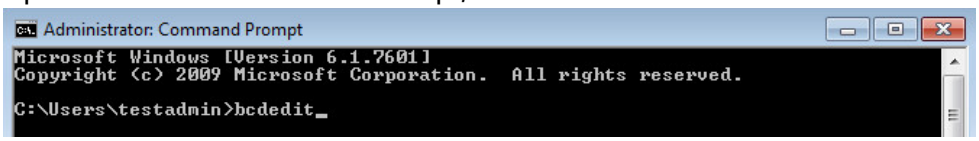
- Errors may occur by trying to connect some Endress+Hauser deviceDTMs in Unity Pro 11.1 Frame:



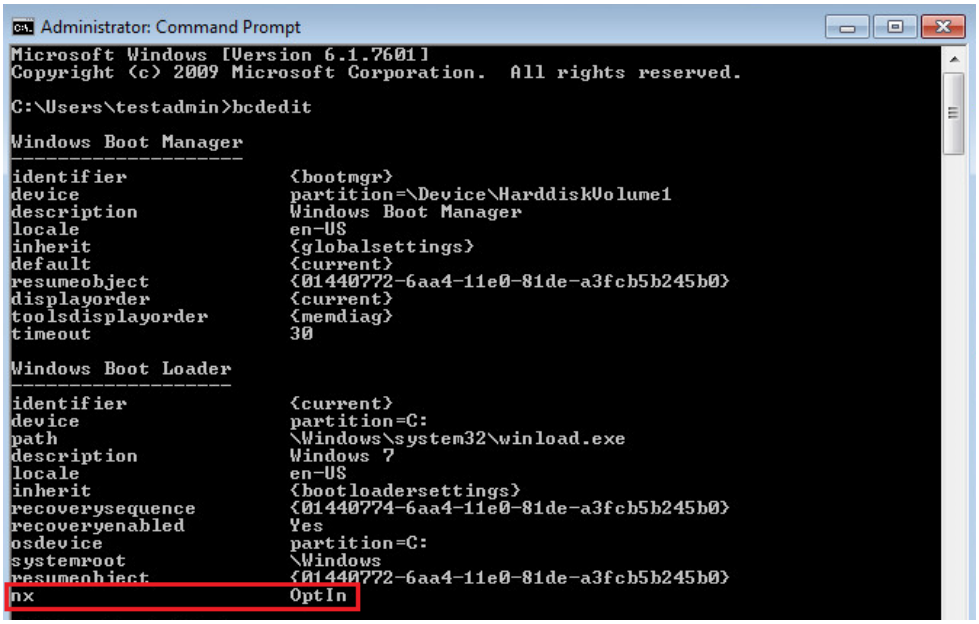
The workaround to avoid this issue consists in disabling the Windows Data Execution Prevention (DEP) option on the engineering station, on which is installed Unity Pro.

Steps to proceed:

- Open the MS DOS Command Prompt, write the command “bcdedit” and click on ”Enter”:

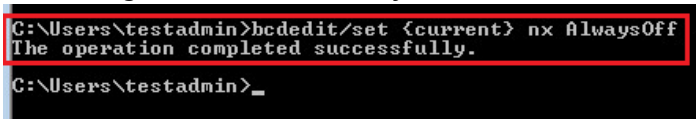


- This displays the Windows Boot Manager and Loader settings:



- Check the parameter “nx”.
  If its state is “OptIn”, write following command and click on “Enter”:

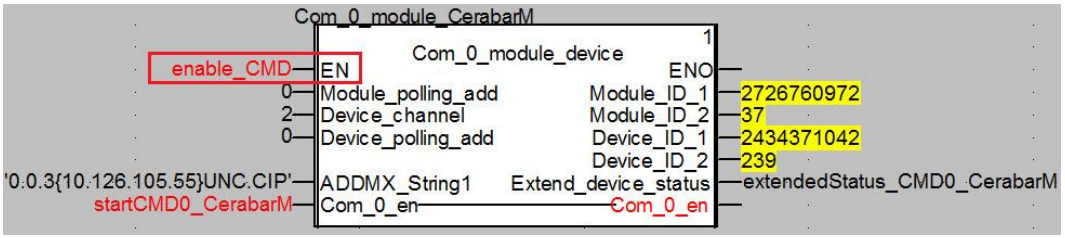
- This changes the state to “AlwaysOff”:



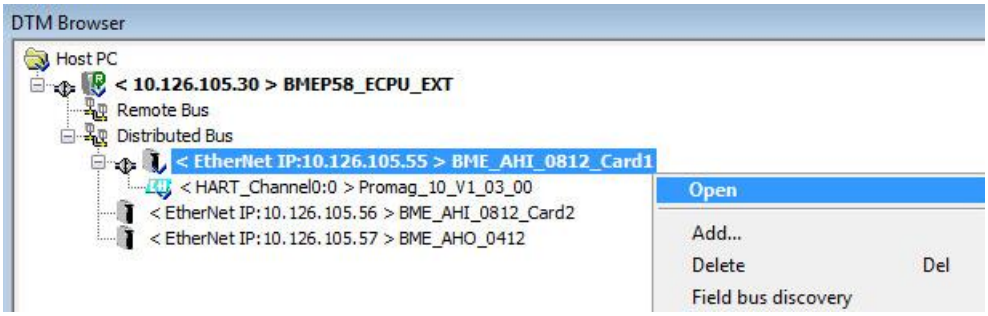
- Reboot the Engineering station.

4.4 HART Analog Input Module Online Connection

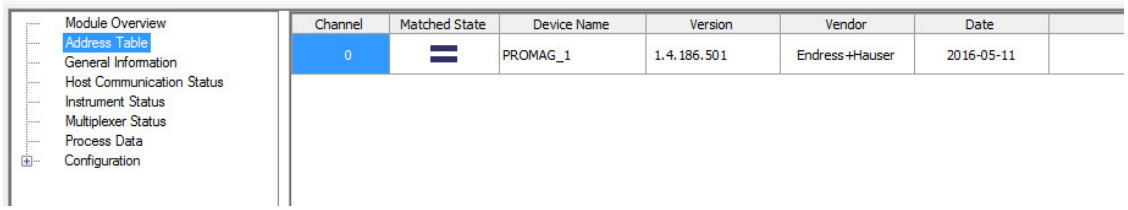
- Before connecting the HART module comDTM and deviceDTMs, make sure that all HART function blocks are deactivated in the program. Please refer to chapter 3.2.5.1.



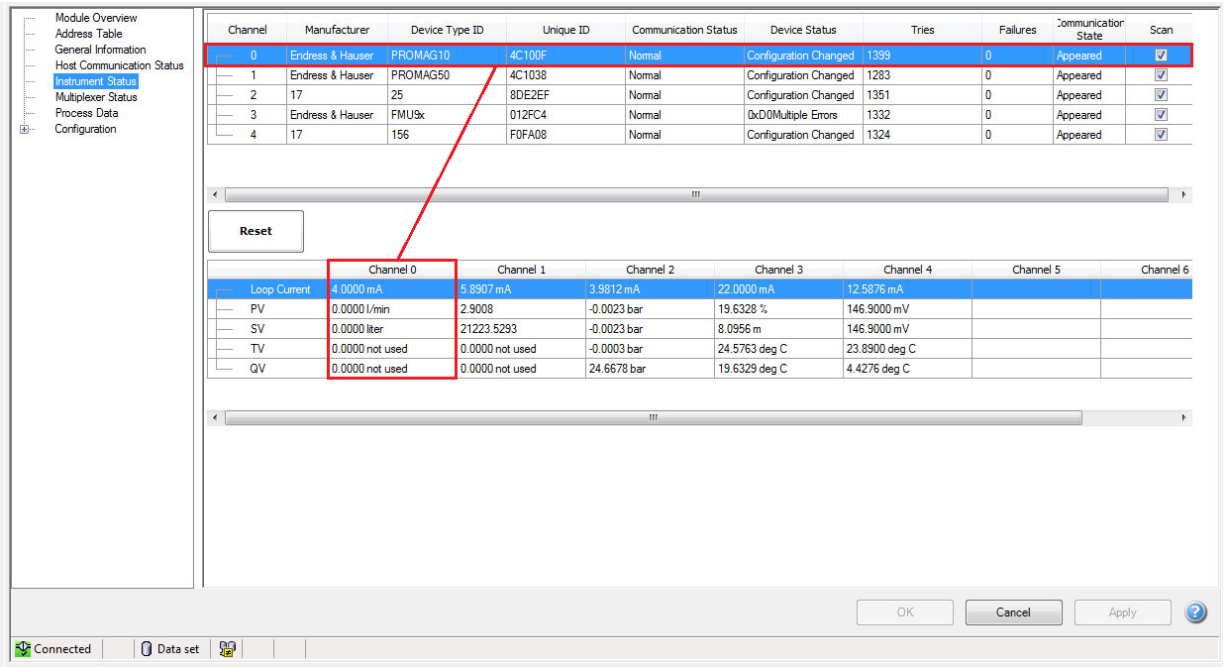
- Right-click on the DTM “BME_AHI_0812” and select the menu “Open”:



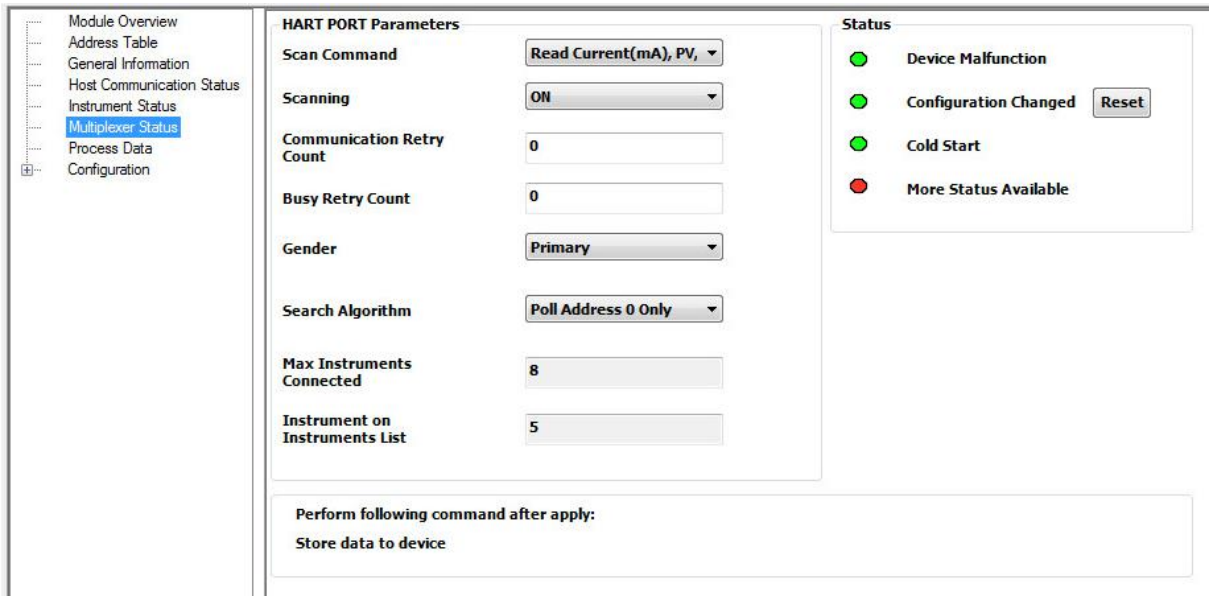
- Select the menu “Address Table”. This menu displays the devices whose deviceDTM is configured in the DTM Browser. If needed, click on the button “Rescan” to refresh.
  In this example, the “Matched State” is equal. That means the device ID and vendor ID in the project match the online values of the device:



- Select the menu “Instrument Status”. This menu displays all connected devices of the card as well as the HART Data if these one are enabled (See Chapter 3.1.5.1) and if the Multiplexer Scanning option is enabled:

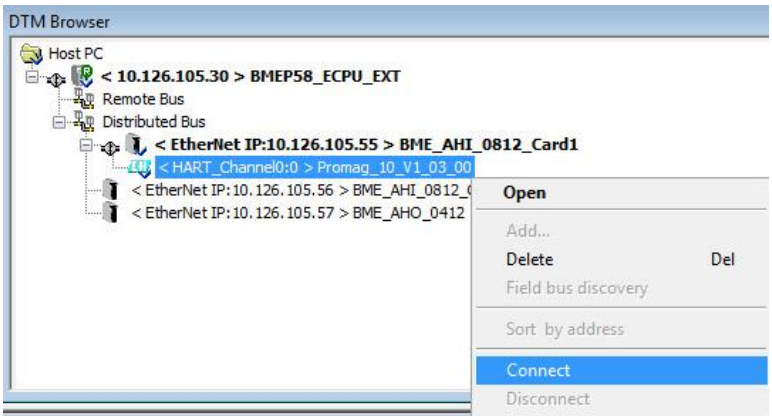


- Select the menu “Instrument Status”. This menu allows the user to configure the HART PORT Parameters. In this example, the option “Scanning” is enabled:

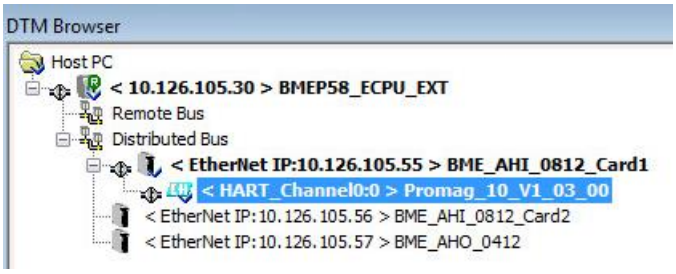


4.5 DeviceDTM Online Connection

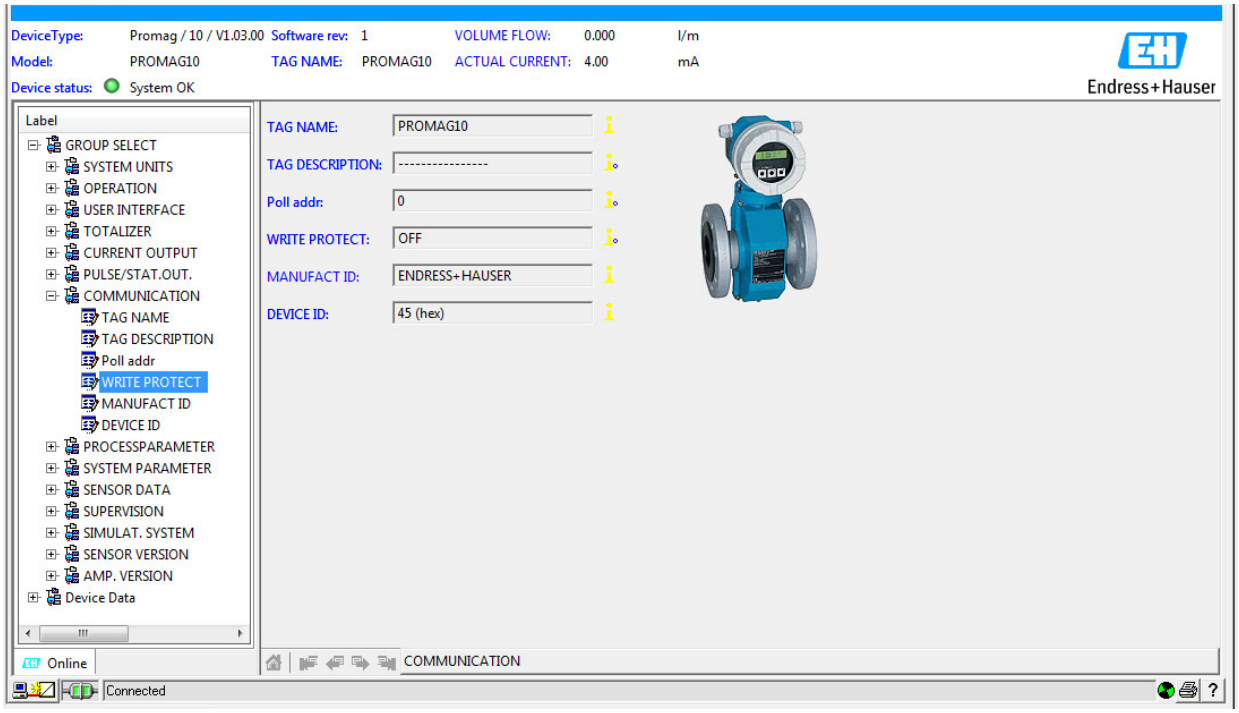
- Right-click on the deviceDTM “Promag_10_V1_03_00” and select the menu “Connect”:



- The DeviceDTM “Promag_10_V1_03_00” is now connected:



- Double-click on the deviceDTM “Promag_10_V1_03_00”:
  The Online connection is established. Device parameters can be accessed:



5 Specific Integration

This chapter explains how to handle device specific HART commands over Ethernet IP from the control strategy.

To evaluate and document the workflow, we have implemented this by example for reading and resetting the totalizer of a Promag50 device. Further device specific HART Commands may be implemented based on this concept.

The Promag50 is connected to the HART analog input card AHI0812 Channel 1 of the X80 Remote IO Station.

5.1 Principle

Sending HART Commands over Ethernet IP is possible by using the function block "DATA_EXCH".

Two Device Specific function blocks for the Promag50 have been developed by using the "DATA_EXCH" function block in combination with the Universal HART CMD0 function block "Com_0_module_device" of the library "ex80_hart_generic_dfb.xdb":

The function block "readTotalizer1_Promag50" is used to read the Totalizer1 value.

The function block "resetTotalizer1_Promag50 is used to reset the Totalizer1 value.

5.1.1 General Workflow

The sequence for sending/receiving Device Specific HART Commands over Ethernet IP has four main steps in our example:

- Step 1: Get module and device ID
  - Command 0 is sent on the HART module's channel on which the device is connected in order to get the module ID and the device ID.

- Step 2: Request telegram configuration
  - Request telegram of the "DATA_EXCH" function block is prepared (see chapter 5.1.2).

- Step 3: Function block "DATA_EXCH" enabling
  - The function block "DATA_EXCH" can be enabled.

- Step 4: Response telegram decoding
  - Received data can be decoded (see chapter 5.1.3).

5.1.2 Request Telegram

The request telegram of the function block “DATA_EXCH” sent from the M580 PLC to the HART device is composed of one Ethernet IP part and two HART parts, one specific for communicating with the HART Module (via Command 155) and another one specific for communicating with the HART device (via Command 174/Command175).

Request Telegram		
Part 1 Header Ethernet IP	Part 2 HART Command to HART Module	Part 3 HART Command to HART Device

Following device specific data need to be implemented in the Request Telegram of the function block “DATA_EXCH” (Part3):

- Specific HART Data for Command 174
  Read Totalizer 1:
  `Request data (3 bytes): 0x07 0x6E 0x01`

- Specific HART Data for Command 175
  Reset Totalizer 1:
  `Request data (5 bytes): 0x07 0x6B 0x01 0xFB 0x01`

5.1.2.1 Complete Request Telegram for Device Specific Command 174

The following table displays the 40 bytes request telegram for the device specific Command 174 in order to read the value of Totalizer1. This telegram is specific for the used Module and device in our example.

Request Frame	Offset	Value	Comment
Part1 Header Ethernet IP	Byte 0	0x4B	Explicit Message Service
	Byte 1	0x03	Request path size
	Byte 2	0x21	Request path
	Byte 3	0x00	
	Byte 4	0x10	
	Byte 5	0x04	
	Byte 6	0x24	
	Byte 7	0x01	
Part2 HART Command to HART Modulelin	Byte 8	0xFF	Preambles
	Byte 9	0xFF	
	Byte 10	0xFF	
	Byte 11	0xFF	
	Byte 12	0xFF	
	Byte 13	0x82	Delimiter
	Byte 14	0xA2	Module ID
	Byte 15	0x87	
	Byte 16	0x12	
	Byte 17	0x0C	
	Byte 18	0x25	
	Byte 19	0x9B	Command 155
	Byte 20	0x12	Data length (from Byte 21 to 38)
	Byte 21	0x01	HART Channel Number
Part3 HART Command to Device	Byte 22	0xFF	Preambles
	Byte 23	0xFF	
	Byte 24	0xFF	
	Byte 25	0xFF	
	Byte 26	0xFF	
	Byte 27	0x82	Delimiter
	Byte 28	0x91	Device ID
	Byte 29	0x41	
	Byte 30	0x4C	
	Byte 31	0x10	
	Byte 32	0x38	
	Byte 33	0xAE	Device Specific Command 174
	Byte 34	0x03	Device Specific data length (from Byte 35 to 37)
	Byte 35	0x07	Device Specific request data
	Byte 36	0x6E	
	Byte 37	0x01	
	Byte 38	0xF3	Checksum Command 174 (from Byte 27 to 37)
	Byte 39	0xEB	Checksum Command 155 (from Byte 13 to 38)

5.1.2.2 Complete Request Telegram for Device Specific Command 175

The following table displays the 42 bytes request telegram for the device specific Command 175 in order to reset the value of Totalizer1. This telegram is specific for the used Module and device in our example.

Request Frame	Offset	Value	Comment
Part1 Header Ethernet IP	Byte 0	0x4B	Explicit Message Service
	Byte 1	0x03	Request path size
	Byte 2	0x21	Request path
	Byte 3	0x00	
	Byte 4	0x10	
	Byte 5	0x04	
	Byte 6	0x24	
	Byte 7	0x01	
Part2 HART Command to HART Module	Byte 8	0xFF	Preambles
	Byte 9	0xFF	
	Byte 10	0xFF	
	Byte 11	0xFF	
	Byte 12	0xFF	
	Byte 13	0x82	Delimiter
	Byte 14	0xA2	Module ID
	Byte 15	0x87	
	Byte 16	0x12	
	Byte 17	0x0C	
	Byte 18	0x25	
	Byte 19	0x9B	Command 155
	Byte 20	0x12	Data length (from Byte 21 to 38)
	Byte 21	0x01	HART Channel Number
Part3 HART Command to Device	Byte 22	0xFF	Preambles
	Byte 23	0xFF	
	Byte 24	0xFF	
	Byte 25	0xFF	
	Byte 26	0xFF	
	Byte 27	0x82	Delimiter
	Byte 28	0x91	Device ID
	Byte 29	0x51	
	Byte 30	0xA2	
	Byte 31	0x2F	
	Byte 32	0x74	
	Byte 33	0xAF	Device Specific Command 175
	Byte 34	0x05	Device Specific data length (from Byte 35 to 37)
	Byte 35	0x07	Device Specific request data
	Byte 36	0x6B	
	Byte 37	0x01	
	Byte 38	0xFB	
	Byte 39	0x01	
	Byte 40	0x86	Checksum Command 175 (from Byte 27 to 39)
	Byte 42	0xEC	Checksum Command 155 (from Byte 27 to 40)

5.1.3 Response Telegram

The response telegram of the function block “DATA_EXCH” received from the HART device on the M580 PLC is composed of one Ethernet IP part and two HART parts, one specific for with the HART Module (via Command 155) and another one specific for the HART device (via Command 174/Command175).

Response Telegram		
Part 1 Header Ethernet IP	Part 2 HART Command to HART Module	Part 3 HART Command to HART Device

Following device specific data are received in the Response Telegram of the function block “DATA_EXCH” (Part3):

- Specific HART Data for Command 174
  Read Totalizer 1:
  `Response data (8 bytes): 0x07 0x6E 0x01 0xFB + 4 bytes data`

- Specific HART Data for Command 175
  Reset Totalizer 1:
  `Response data (5 Bytes): 0x07 0x6B 0x01 0xFB 0x01`

5.1.3.1 Complete Response Telegram for Device Specific Command 174

The following table displays a 40 bytes response telegram for the device specific Command 174. This telegram is specific for the used Module and device in our example.

Request Frame	Offset	Value	Comment
Part1 Header Ethernet IP	Byte 0	0xCB	Explicit Message Service
	Byte 1	0x00	
	Byte 2	0x00	
	Byte 3	0x00	
Part2 HART Command to HART Module	Byte 4	0xFF	Preambles
	Byte 5	0xFF	
	Byte 6	0xFF	
	Byte 7	0xFF	
	Byte 8	0xFF	
	Byte 9	0x86	Delimiter
	Byte 10	0xA2	Module ID
	Byte 11	0x87	
	Byte 12	0x12	
	Byte 13	0x0C	
	Byte 14	0x25	
	Byte 15	0x9B	Command 155
	Byte 16	0x16	Data length (from Byte 17 to 38)
	Byte 17	0x00	Communication Status (HART Specification 99)
	Byte 18	0x50	Device Status (HART Specification 99)
	Byte 19	0x01	HART Channel Number
Part3 HART Command to Device	Byte 20	0x86	Delimiter
	Byte 21	0x91	Device ID
	Byte 22	0x41	
	Byte 23	0x4C	
	Byte 24	0x10	
	Byte 25	0x38	
	Byte 26	0xAE	Device Specific Command 174
	Byte 27	0x0A	Device Specific data length(from Byte 28 to 37)
	Byte 28	0x00	Communication Status (HART Specification 99)
	Byte 29	0x40	Device Status (HART Specification 99)
	Byte 30	0x07	HART Specific Response
	Byte 31	0x6E	
	Byte 32	0x01	
	Byte 33	0xFB	
	Byte 34	0x41	Totalizer 1 value
	Byte 35	0xBD	
	Byte 36	0x37	
	Byte 37	0xF4	
	Byte 38	0x2E	Checksum Command 174 (from Byte 27 to 37)
	Byte 39	0x63	Checksum Command 155 (from Byte 13 to 38)

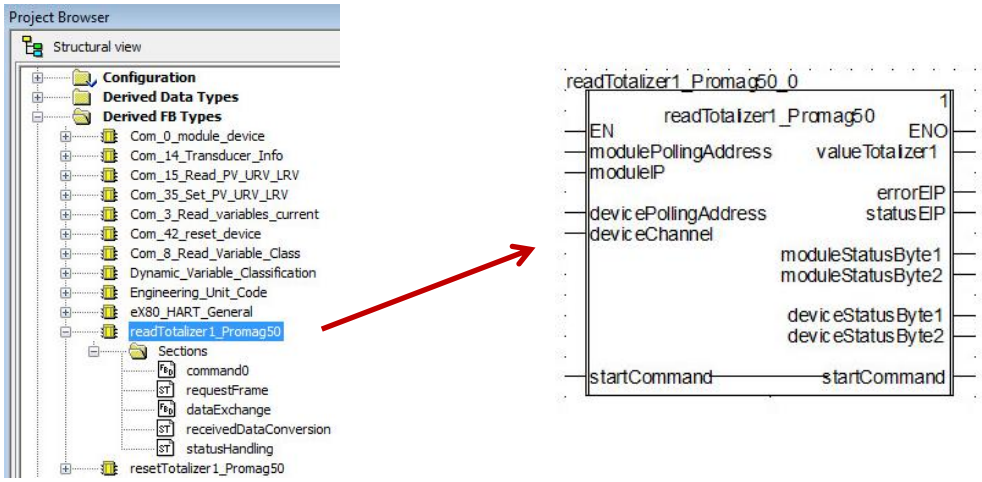
5.1.3.2 Complete Response Telegram for Device Specific Command 175

The following table displays the 37 bytes response telegram for the device specific Command 175. This telegram is specific for the used Module and device in our example.

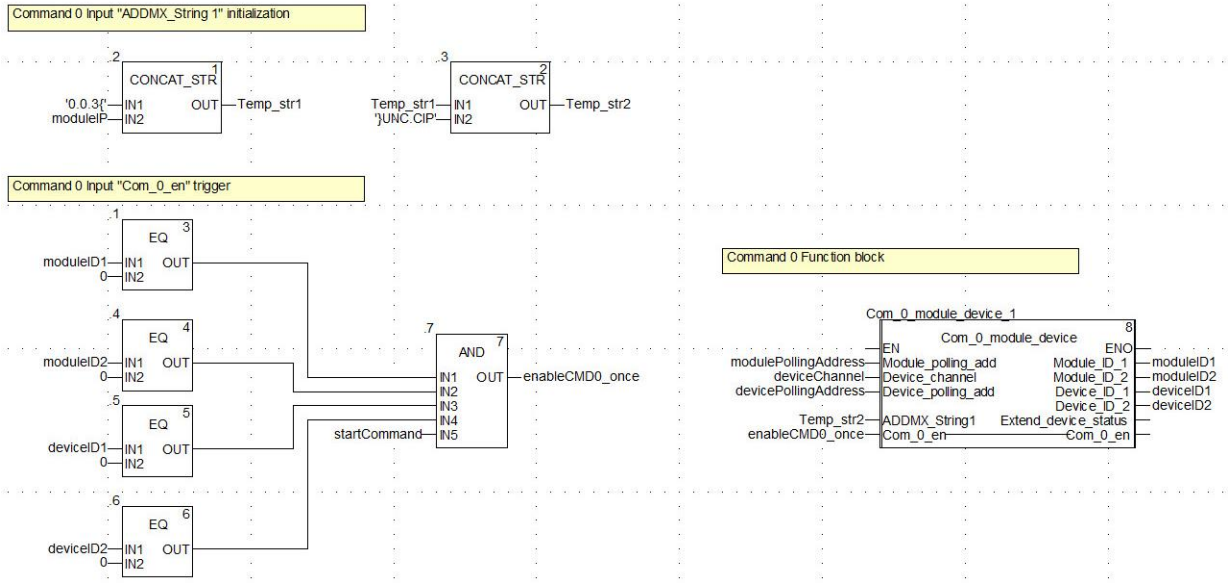
Request Frame	Offset	Value	Comment
Part1 Header Ethernet IP	Byte 0	0xCB	Explicit Message Service
	Byte 1	0x00	
	Byte 2	0x00	
	Byte 3	0x00	
Part2 HART Command to HART Module	Byte 4	0xFF	Preambles
	Byte 5	0xFF	
	Byte 6	0xFF	
	Byte 7	0xFF	
	Byte 8	0xFF	
	Byte 9	0x86	Delimiter
	Byte 10	0xA2	Module ID
	Byte 11	0x87	
	Byte 12	0x12	
	Byte 13	0x0C	
	Byte 14	0x25	
	Byte 15	0x9B	Command 155
	Byte 16	0x13	Data length (from Byte 17 to 35)
	Byte 17	0x00	Communication Status (HART Specification 99)
	Byte 18	0x50	Device Status (HART Specification 99)
	Byte 19	0x01	HART Channel Number
Part3 HART Command to Device	Byte 20	0x86	Delimiter
	Byte 21	0x91	Device ID
	Byte 22	0x41	
	Byte 23	0x4C	
	Byte 24	0x10	
	Byte 25	0x38	
	Byte 26	0xAF	Device Specific Command 175
	Byte 27	0x07	Device Specific data length(from Byte 28 to 34)
	Byte 28	0x00	Communication Status (HART Specification 99)
	Byte 29	0x40	Device Status (HART Specification 99)
	Byte 30	0x07	HART Specific Response
	Byte 31	0x6B	
	Byte 32	0x01	
	Byte 33	0xFB	
	Byte 34	0x01	
	Byte 35	0x4D	Checksum Command 175 (from Byte 20 to 34)
	Byte 36	0x41	Checksum Command 155 (from Byte 9 to 35)

5.2 Function Block “readTotalizer1_Promag50”

- The function block has been created in the “Derived FB Types” library and is composed of 5 sections:



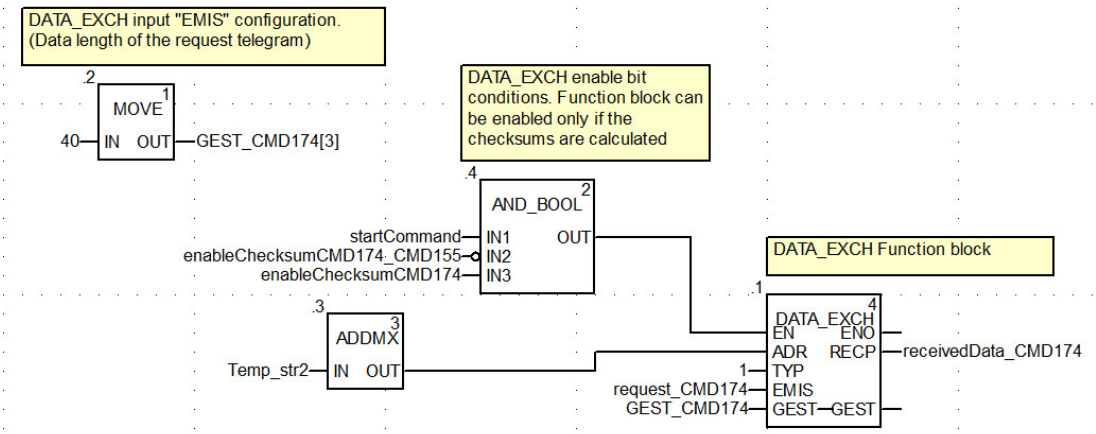
- Section “command0”
  This part is handling the HART command 0 function block:
  - The first part is configuring the Command 0 input “ADDMX_String1”.
  - The second part is handling the Command 0 input “Com_0_en” in order to execute the function block only one time.
  - The third part is the Command 0 function block configuration.



- Section “requestFrame”
  This part is handling the input request table of the function Block “DATA_EXCH” as described in Chapter 5.1.2.1:

```
(*Command 174: Request Telegram "Read Totaliser1 Value"*)
request_CMD174[0]:= 16#034B;
request_CMD174[1]:= 16#0021;
request_CMD174[2]:= 16#0410;
request_CMD174[3]:= 16#0124;
request_CMD174[4]:= 16#FFFF;
request_CMD174[5]:= 16#FFFF;
request_CMD174[6]:= 16#82FF;
request_CMD174[7]:=DWORD_TO_INT(ROR((moduleID1 AND 16#00FF_0000),8)) + DWORD_TO_INT(ROR((moduleID1 AND 16#FF00_0000),24));
request_CMD174[8]:=DWORD_TO_INT(ROL((moduleID1 AND 16#0000_00FF),8)) + DWORD_TO_INT(ROR((moduleID1 AND 16#0000_FF00),8));
request_CMD174[9]:=DWORD_TO_INT((moduleID2 AND 16#0000_00FF)) + 16#9B00;
tempChannel:=SHL(deviceChannel,8);
request_CMD174[10]:= tempChannel+ 16#0012;
request_CMD174[11]:= 16#FFFF;
request_CMD174[12]:= 16#FFFF;
request_CMD174[13]:= 16#82FF;
request_CMD174[14]:=DWORD_TO_INT(ROR((deviceID1 AND 16#FF00_0000),24)) + DWORD_TO_INT(ROR((deviceID1 AND 16#00FF_0000),8));
request_CMD174[15]:=DWORD_TO_INT(ROR((deviceID1 AND 16#0000_FF00),8)) + DWORD_TO_INT(ROL((deviceID1 AND 16#0000_00FF),8));
request_CMD174[16]:=DWORD_TO_INT((deviceID2 AND 16#0000_00FF)) + 16#AE00;
request_CMD174[17]:= 16#0703;
request_CMD174[18]:= 16#016E;

(*Checksum 1 Command 174: Request Telegram "Read Totaliser1 Value"*)
if ((moduleID1>0 AND moduleID2>0 AND deviceID1>0 AND deviceID2>0 ) AND (NOT enableChecksumCMD174))  THEN
   FOR offset:=13 to 17 BY 1 DO
      checksumCMD174:= checksumCMD174 XOR (SHR(request_CMD174[offset],8) XOR request_CMD174[offset+1]);
   END_FOR;
   checksumCMD174:=checksumCMD174 XOR SHR(request_CMD174[offset],8);
   request_CMD174[19]:= checksumCMD174;
   enableChecksumCMD174:= TRUE;
   enableChecksumCMD174_CMD155:= TRUE;
END_IF;

(*Checksum 2 Command 174: Request Telegram "Read Totaliser1 Value"*)
if enableChecksumCMD174_CMD155  THEN
   FOR offset:=6 to 18 BY 1 DO
      checksumCMD174_155:= checksumCMD174_155 XOR (SHR(request_CMD174[offset],8) XOR request_CMD174[offset+1]);
   END_FOR;
   tempRegCMD174:=SHL(checksumCMD174_155,8);
   request_CMD174[19]:=request_CMD174[19] + tempRegCMD174;
   enableChecksumCMD174_CMD155:= FALSE;
END_IF;
```

- Section “dataExchange”:
  This section is handling the configuration of the function block “DATA_EXCH”:



- Section “receivedDataConversion”
  This section is handling the data conversion of the “DATA_EXCH” function block output value:

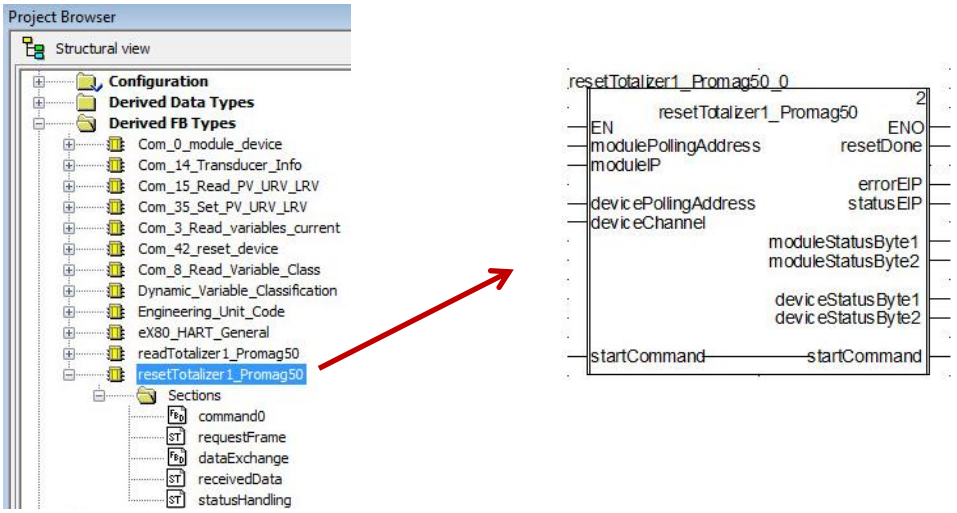
```
(* Conversion Totalizer1 value*)
valueTotalizer1:=DWORD_TO_REAL(ROL(INT_TO_DWORD(receivedData_CMD174[17] AND 16#00FF),24) +
                                ROL(INT_TO_DWORD(receivedData_CMD174[17] AND 16#FF00),8) +
                                ROL(INT_TO_DWORD(receivedData_CMD174[18] AND 16#00FF),8) +
                                ROR(INT_TO_DWORD(receivedData_CMD174[18] AND 16#FF00),8)) ;
```

- Section “statusHandling”
  This section is handling the main errors status of the Ethernet IP communication as well as the HART communication status:

```
(*HART Communication Status Bytes received on Byte 17 and Byte 18*)
moduleStatusByte1:=SHR(receivedData_CMD174[8],8);
moduleStatusByte2:=receivedData_CMD174[9];

(*HART Communication Status Bytes received on Byte 28 and Byte 29*)
deviceStatusByte1:=receivedData_CMD174[14];
deviceStatusByte2:=SHR(receivedData_CMD174[14],8);

(*Error EIP*)
IF(receivedData_CMD174[01]=16#0D80 or receivedData_CMD174[01]=16#1280
or receivedData_CMD174[01]=16#1580 or receivedData_CMD174[01]=16#3080) THEN
   errorEIP:=TRUE;
   statusEIP:=receivedData_CMD174[01];
else
   errorEIP:=FALSE;
   statusEIP:=receivedData_CMD174[01];
END_IF;
```

5.3 Function Block “resetTotalizer1_Promag50”

- The function block has been created in the “Derived FB Types” library and is composed of 5 parts:



- Section “command0”
  This part is based on the same principle as done for the section “command0” of the function block “readTotalizer1_Promag50”.

- Section “requestFrame”
  This part is handling the input request table of the function Block “DATA_EXCH” as described in Chapter 5.1.2.2:

```
(*Command 175: Request Telegram "Reset Totaliser1 Value"*)
request_CMD175[0]:= 16#034B;
request_CMD175[1]:= 16#0021;
request_CMD175[2]:= 16#0410;
request_CMD175[3]:= 16#0124;
request_CMD175[4]:= 16#FFFF;
request_CMD175[5]:= 16#FFFF;
request_CMD175[6]:= 16#82FF;
request_CMD175[7]:=DWORD_TO_INT(ROR((moduleID1 AND 16#00FF_0000),8)) + DWORD_TO_INT(ROR((moduleID1 AND 16#FF00_0000),24));
request_CMD175[8]:=DWORD_TO_INT(ROL((moduleID1 AND 16#0000_00FF),8)) + DWORD_TO_INT(ROR((moduleID1 AND 16#0000_FF00),8));
request_CMD175[9]:=DWORD_TO_INT((moduleID2 AND 16#0000_00FF)) + 16#9B00;
tempChannel:=SHL(deviceChannel,8);
request_CMD175[10]:= tempChannel + 16#0014;
request_CMD175[11]:= 16#FFFF;
request_CMD175[12]:= 16#FFFF;
request_CMD175[13]:= 16#82FF;
request_CMD175[14]:=DWORD_TO_INT(ROR((deviceID1 AND 16#FF00_0000),24)) + DWORD_TO_INT(ROR((deviceID1 AND 16#00FF_0000),8));
request_CMD175[15]:=DWORD_TO_INT(ROR((deviceID1 AND 16#0000_FF00),8)) + DWORD_TO_INT(ROL((deviceID1 AND 16#0000_00FF),8));
request_CMD175[16]:=DWORD_TO_INT((deviceID2 AND 16#0000_00FF)) + 16#AF00;
request_CMD175[17]:= 16#0705;
request_CMD175[18]:= 16#016B;
request_CMD175[19]:= 16#01FB;

(*Checksum 1 Command 175: Request Telegram "Reset Totaliser1 Value"*)
if ((moduleID1>0 AND moduleID2>0 AND deviceID1>0 AND deviceID2>0 ) AND (NOT enableChecksumCMD175))  THEN
   FOR offset:=13 to 18 BY 1 DO
      checksumCMD175:= checksumCMD175 XOR (SHR(request_CMD175[offset],8) XOR request_CMD175[offset+1]);
   END_FOR;
   checksumCMD175:=checksumCMD175 XOR SHR(request_CMD175[offset],8);
   request_CMD175[20]:= checksumCMD175;
   enableChecksumCMD175:= TRUE;
   enableChecksumCMD175_CMD155:= TRUE;
END_IF;

(*Checksum 2 Command 175: Request Telegram "Reset Totaliser1 Value"*)
if enableChecksumCMD175_CMD155  THEN
   FOR offset:=6 to 19 BY 1 DO
      checksumCMD175_155:= checksumCMD175_155 XOR (SHR(request_CMD175[offset],8) XOR request_CMD175[offset+1]);
   END_FOR;
   tempRegCMD175:=SHL(checksumCMD175_155,8);
   request_CMD175[20]:=request_CMD175[20] + tempRegCMD175;
   enableChecksumCMD175_CMD155:= FALSE;
END_IF;
```

- Section “dataExchange”
  This part is based on the same principle as done for the section “dataExchange” of the function block “readTotalizer1_Promag50”.

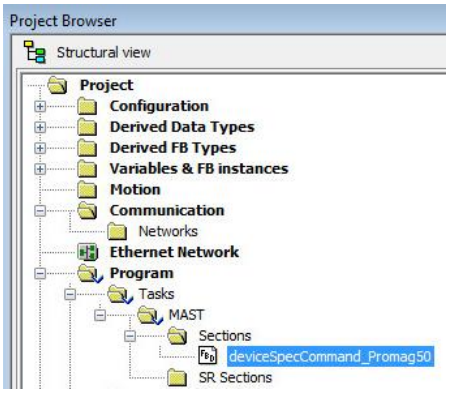
- Section “receivedDataConversion”
  This section is handling the data conversion of the “DATA_EXCH” function block output value:

```
(* Test if the 5 bytes HART specific code are received*)
IF ((receivedData_CMD175[15]= 16#6B07) AND (receivedData_CMD175[16]= 16#FB01) AND
((receivedData_CMD175[17] AND 16#00FF)= 16#0001) AND startCommand) THEN
   resetDone:=TRUE;
   startCommand:=FALSE;
END_IF;

(*Cycle Counter*)
IF resetDone THEN
   cptCycle:= cptCycle + 1;
END_IF;

(*Reset the signal*)
IF (resetDone AND (NOT startCommand) AND cptCycle>5) THEN
   resetDone:=FALSE;
   cptCycle:=0;
END_IF;
```

- Section “statusHandling”
  This part is based on the same principle as done for the section “statusHandling” of the function block “readTotalizer1_Promag50”.

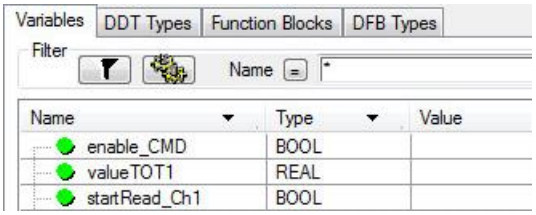
5.4 Program

- In the Project Browser, a new section “deviceSpecificCommand_Promag50” is created for this example:

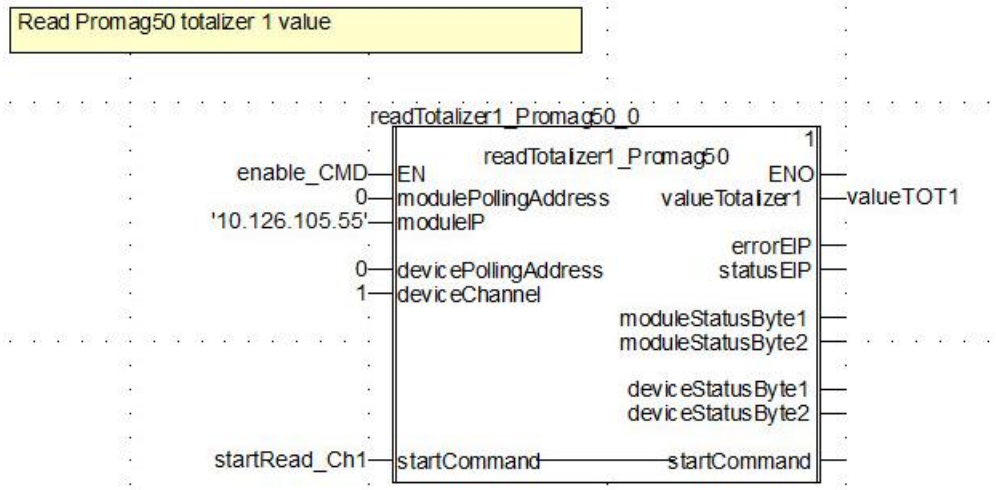


5.4.1 Function Block “readTotalizer1_Promag50” Configuration

- Create following variables:



- Assign all required variables to the function block “readTotalizer1_Promag50”:

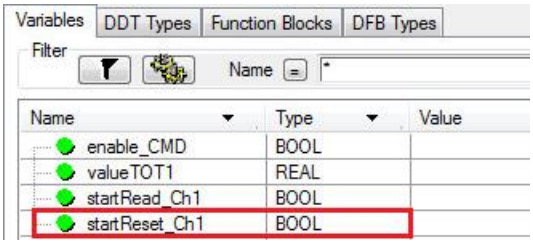


- Mandatory Inputs
  - The parameter “modulePollingAddress” corresponds to the module polling address. In this example, the parameter is set to 0.
  - The parameter “moduleIP” corresponds to the IP address of the HART Module on which is connected the device. In this example, the IP address of the AHI0812 HART module is 10.126.105.55.
  - The parameter “devicePollingAddress” corresponds to the device polling address. In this example, the parameter is set to 0.
  - The parameter “deviceChannel” corresponds to the HART module’s channel on which is connected the device.
  - The parameter “startRead_Ch1” corresponds to the function block start bit.

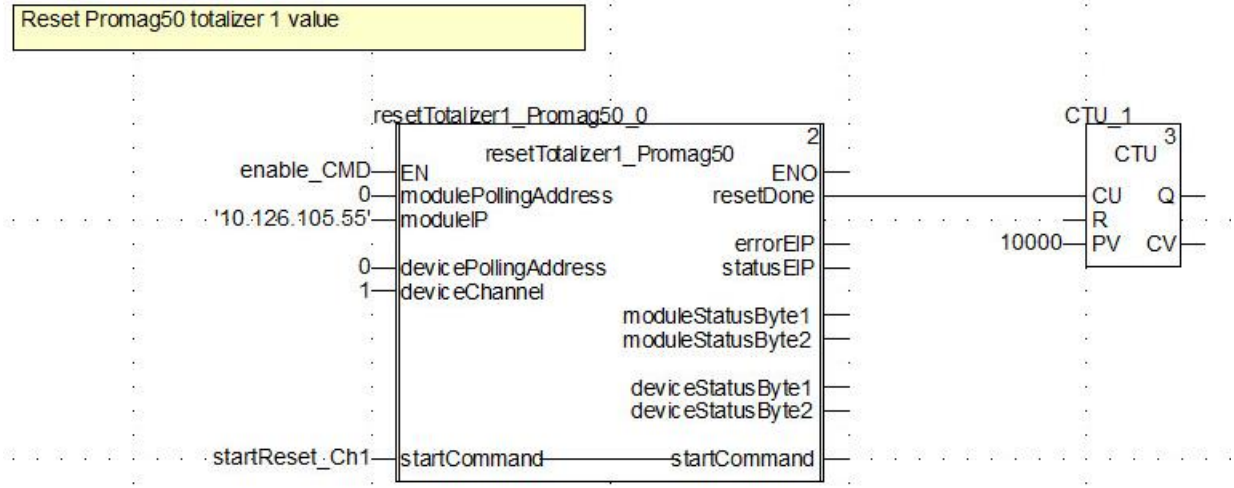
- Outputs
  - The parameter “valueTOT1” corresponds to the received and decoded totalizer1 value in this example.
  - The parameter “errorEIP” is set to TRUE as soon as an Ethernet IP error is detected.
  - The other parameters display status of the EIP and HART communication.

5.4.2 Function block “readTotalizer1_Promag50” Configuration

- Create following Boolean variable “startReset_Ch1”:



- Assign all required variables to the function block “readTotalizer1_Promag50”:



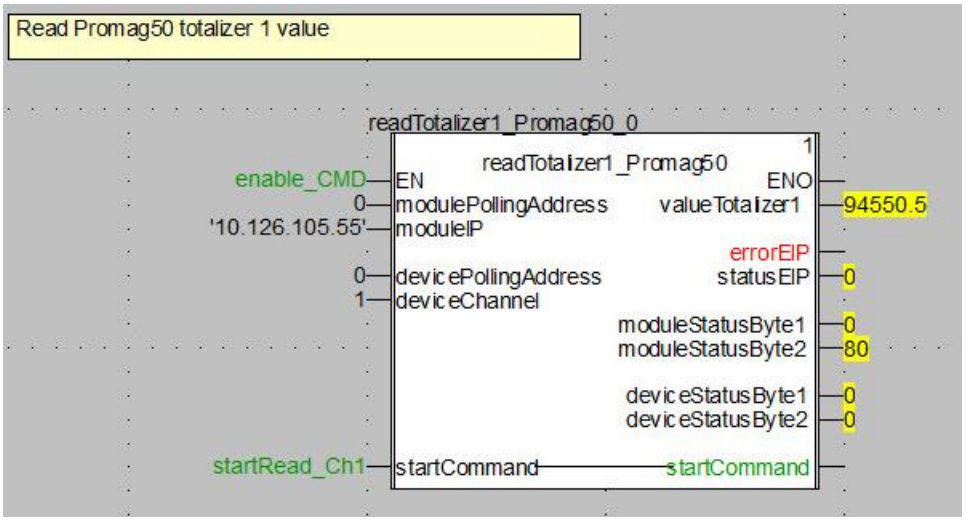
- Mandatory Inputs
  - The parameter “modulePollingAddress” corresponds to the module polling address. In this example, the parameter is set to 0.
  - The parameter “moduleIP” corresponds to the IP address of the HART Module on which is connected the device. In this example, the IP address of the AHI0812 HART module is 10.126.105.55.
  - The parameter “devicePollingAddress” corresponds to the device polling address. In this example, the parameter is set to 0.
  - The parameter “deviceChannel” corresponds to the HART module’s channel on which is connected the device.
  - The parameter “startReset_Ch1” corresponds to the function block start bit.

- Outputs
  - The parameter “resetDone” corresponds to the output status bit and is connected to a counter. If the reset is successful, the counter parameter “CV” is incremented. The preset value “PV” has been initialized to “10000” in this example. As a consequence, if the current counter value “CV” reaches 10000, the output “Q” is set to TRUE.
  - The parameter “errorEIP” is set to TRUE as soon as an Ethernet IP error is detected. The other parameters display status of the EIP and HART communication.

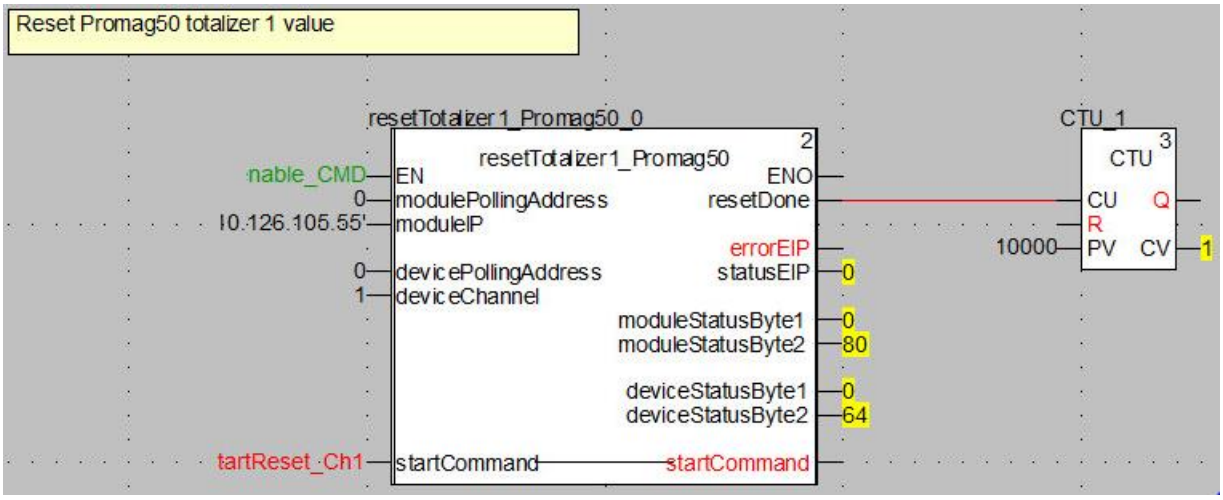
5.5 Online Monitoring

- Before using the function blocks with the Promag50, make sure that its deviceDTM is disconnected as wells as the HART module comDTM of the corresponding card on which the Promag50 is connected.

- The startCommand bit “startRead_Ch1” is set to “TRUE”. The totalizer output value is successfully read:



- The startCommand bit “startReset_Ch1” is set to “TRUE”. The totalizer output value is successfully set to zero. When finished, the counter parameter “CV” is incremented and the startCommand bit “startReset_Ch1” is automatically set to “FALSE”:



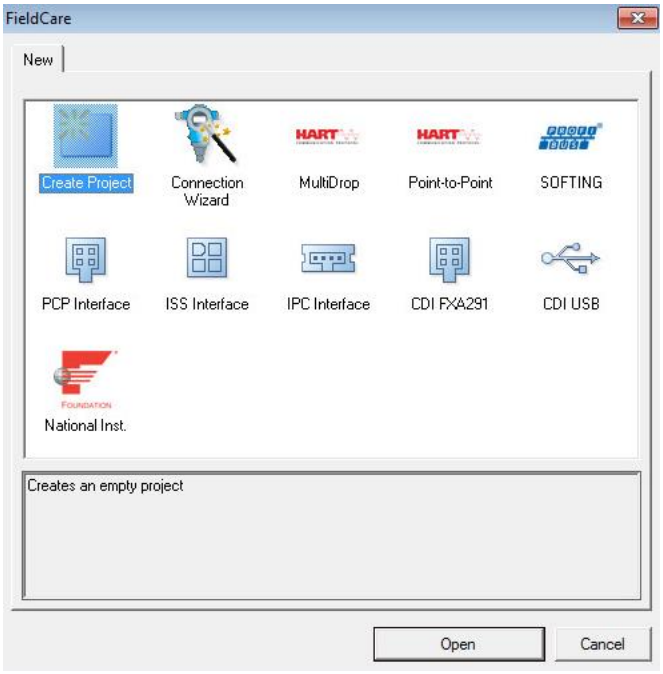
6 Routed Tool Integration

This chapter describes the main workflow for integration of Schneider Electric system components to the Endress+Hauser Plant Asset Management (PAM system) by means of Communication DTMs. As a result, the Endress+Hauser PAM system can access underlying HART devices via Schneider Electric Ethernet backbone for device configuration.

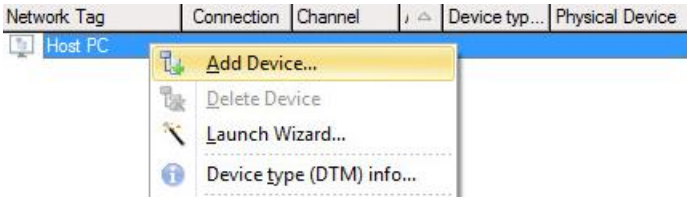
6.1 Schneider Electric "EtherNet/IP Comm Adapter" DTM Configuration

- Start the application FieldCare:

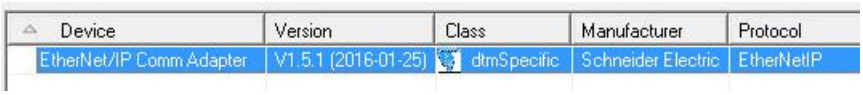
- Create a new project:



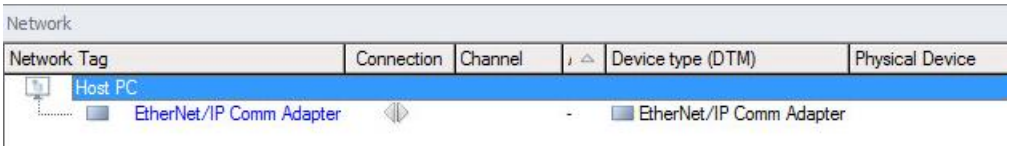
- Right-click on the Network Tag "Host PC" and select the menu "Add Device":



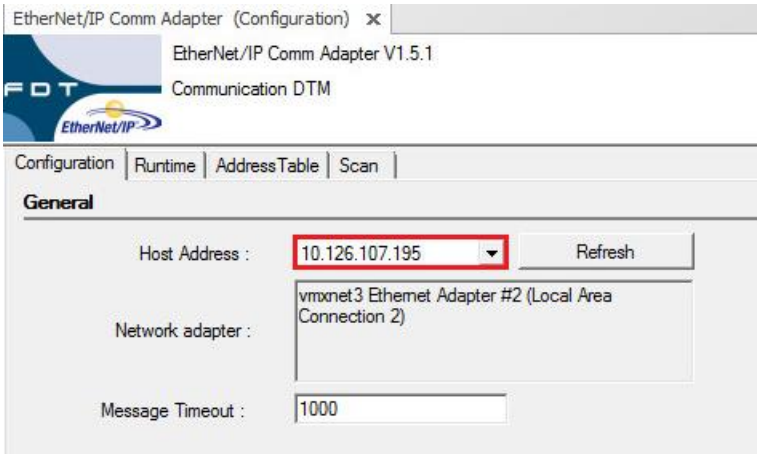
- Select the DTM "EtherNet/IP Comm Adapter" and click on the button "OK":



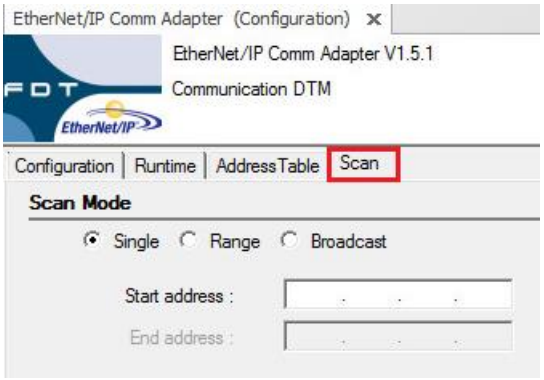
- The DTM "EtherNet/IP Comm Adapter" is implemented in the Network view:



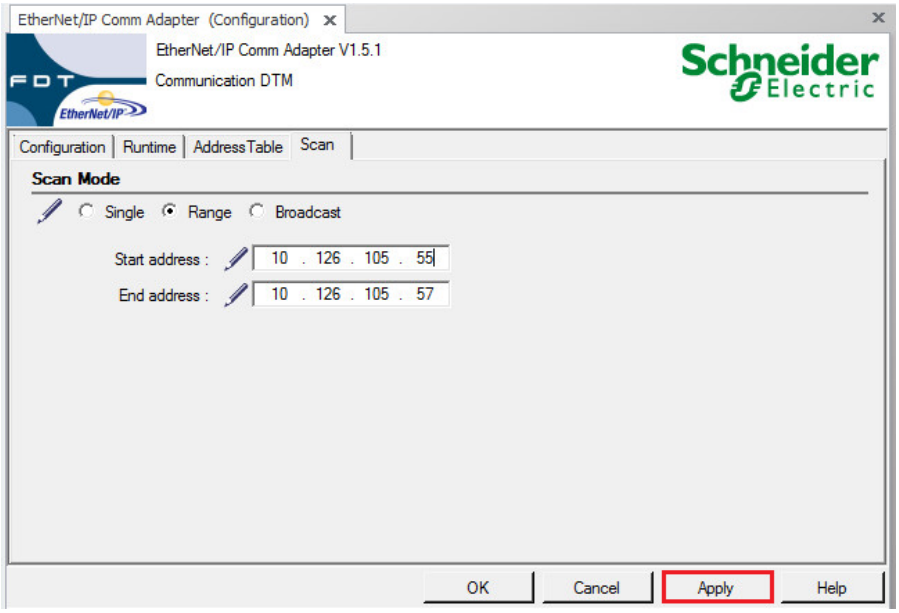
- Double-click on the DTM "EtherNet/IP Comm Adapter". This opens the offline Configuration window. Verify the Host Address IP address:



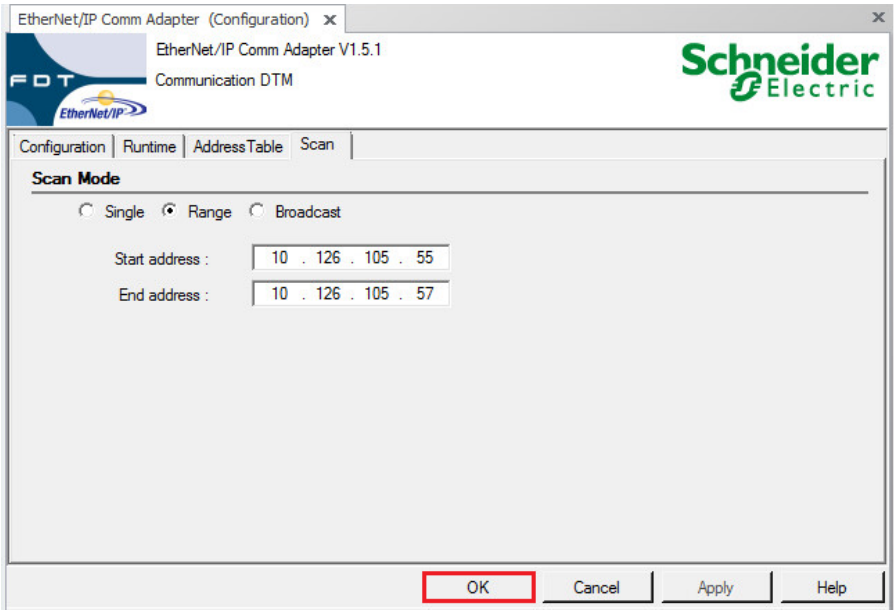
- Select the tab "Scan":



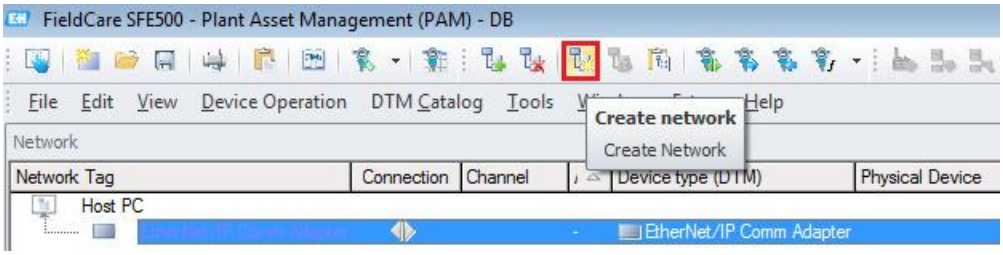
- Select the “Scan Mode” option “Range” and indicate the IP addresses of the HART cards and click on the button “Apply”. In this example, there are 3 cards:
  - 1st card: AHI0812 with IP address 10.126.105.55
  - 2nd card: AHI0812 with IP address 10.126.105.56
  - 3rd card: AHO0412 with IP address 10.126.105.57



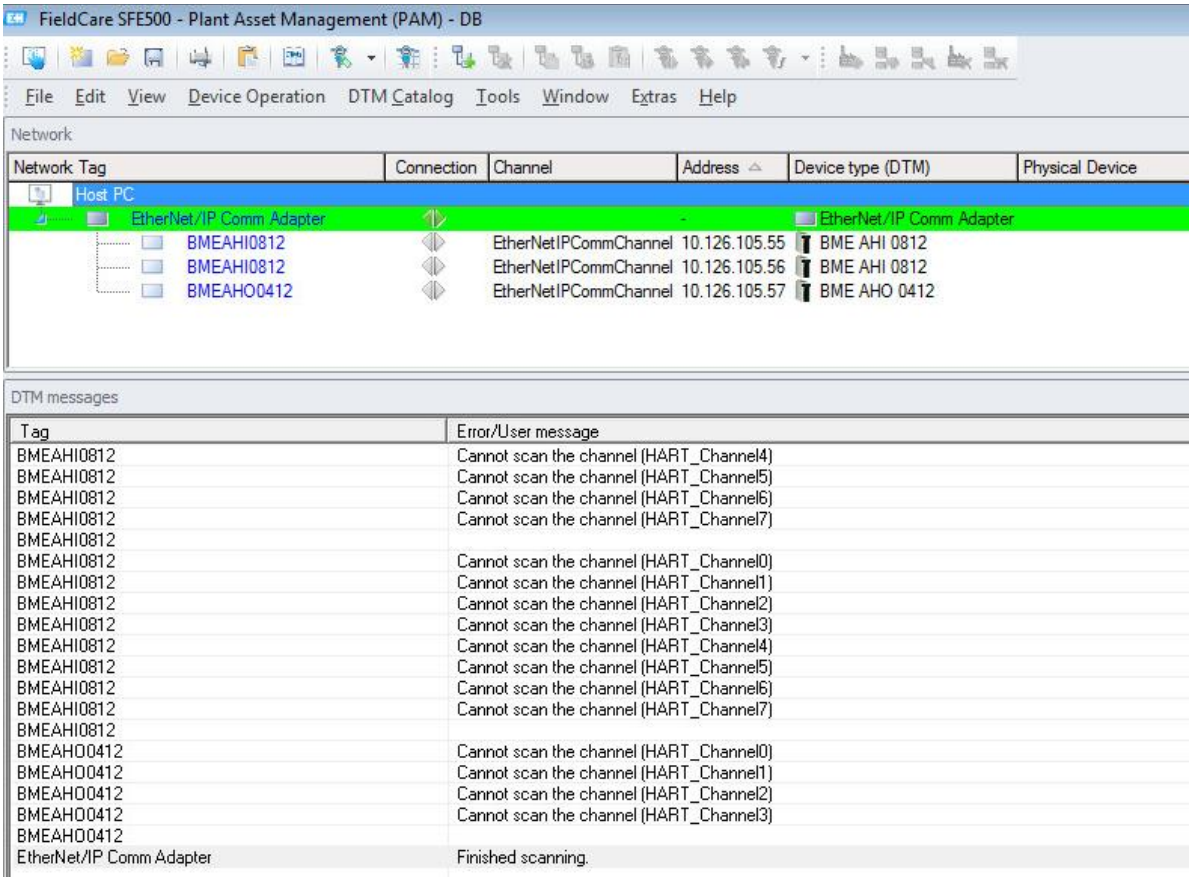
- Configured IP addresses for the “Scan Mode” and click on the button “OK” to close the window:



- In the network view, select the DTM “EtherNet/IP Comm Adapter” and select the shortcut button “Create Network”:

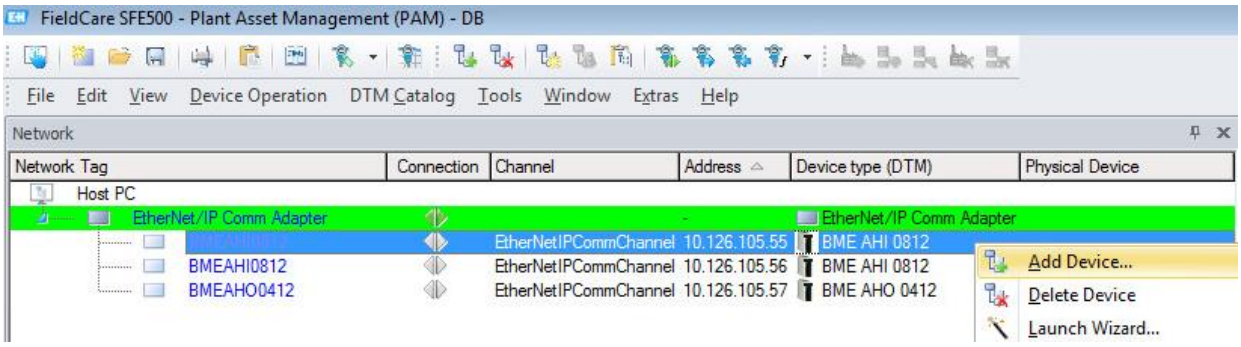


- Schneider Electric HART cards have been found and inserted in the project. However, no Endress+Hauser devices have been scanned. This must be done manually:

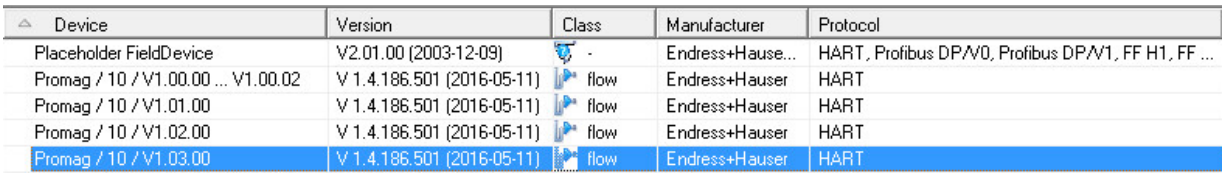


6.2 Endress+Hauser deviceDTM Configuration

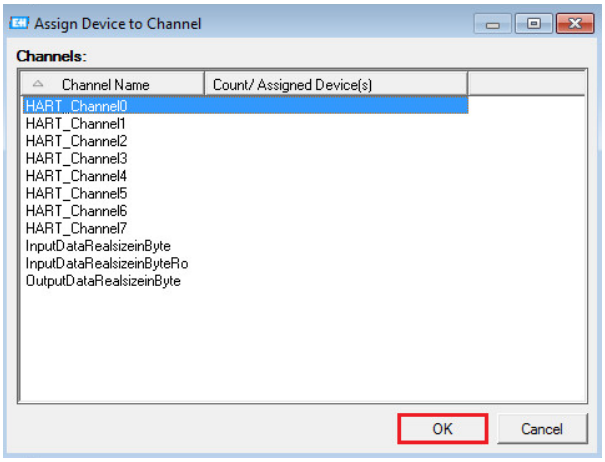
- Right-click on the DTM “BMEAHI0812” and select the menu “Add Device…”:



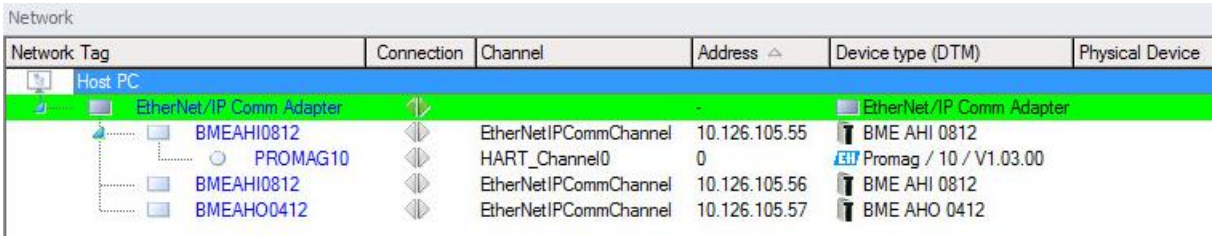
- Select the DTM “Promag/10/V1.03.00” for this example:



- Select the channel number on which the device is wired; channel 0 in this example and click on the button “OK”:



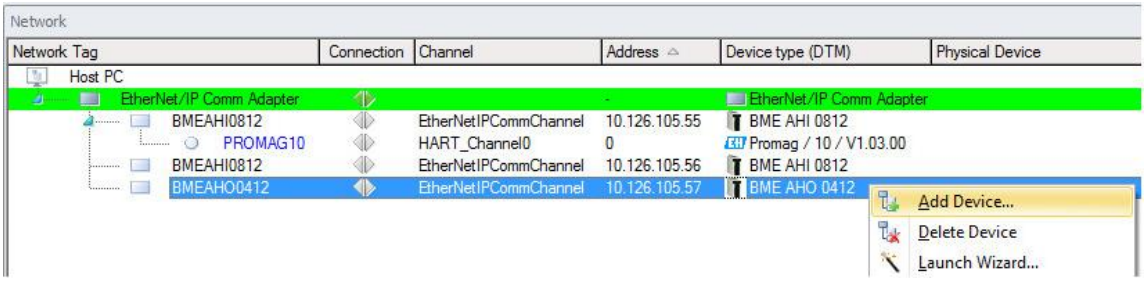
- Device DTM is implemented in the Network view:



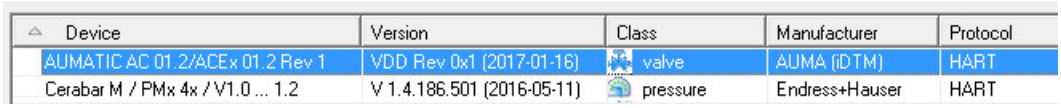
6.3 AUMA Actuator deviceDTM Configuration

The deviceDTM “AUMATIC AC01.2/ACEx 01.2 Rev1” must be added manually in the project.

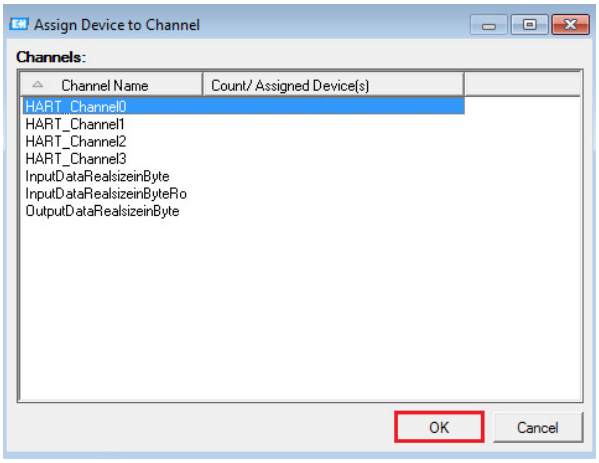
- Right-click on the DTM “BMEAHO0412” and select the menu “Add Device…”:



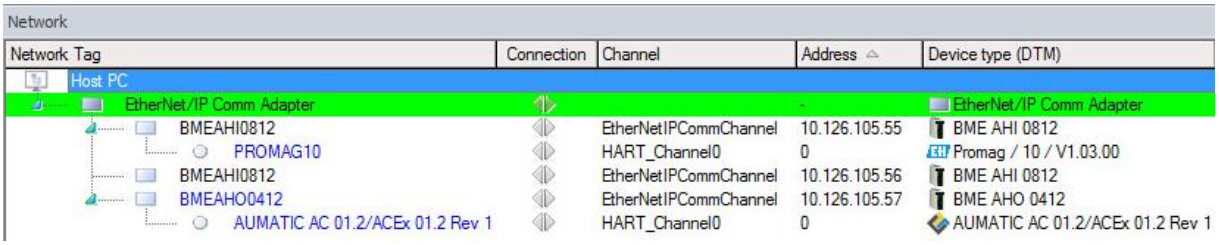
- Select the iDTM “AUMATIC AC01.2/ACEx01.2 Rev1” for this example:



- Select the channel number on which the device is wired; channel 0 in this example and click on the button “OK”:

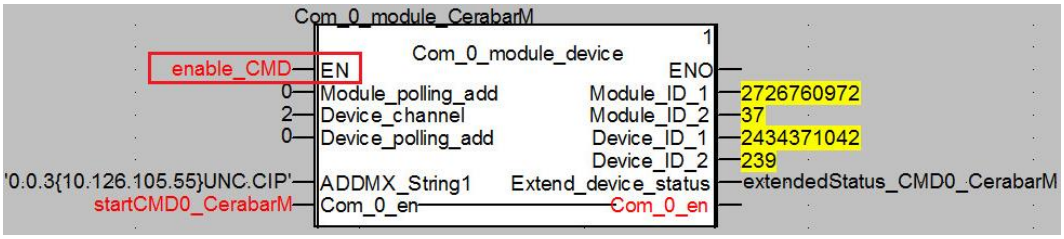


- Device DTM “AUMATIC AC01.2/ACEx01.2 Rev1” is implemented in the Network view:



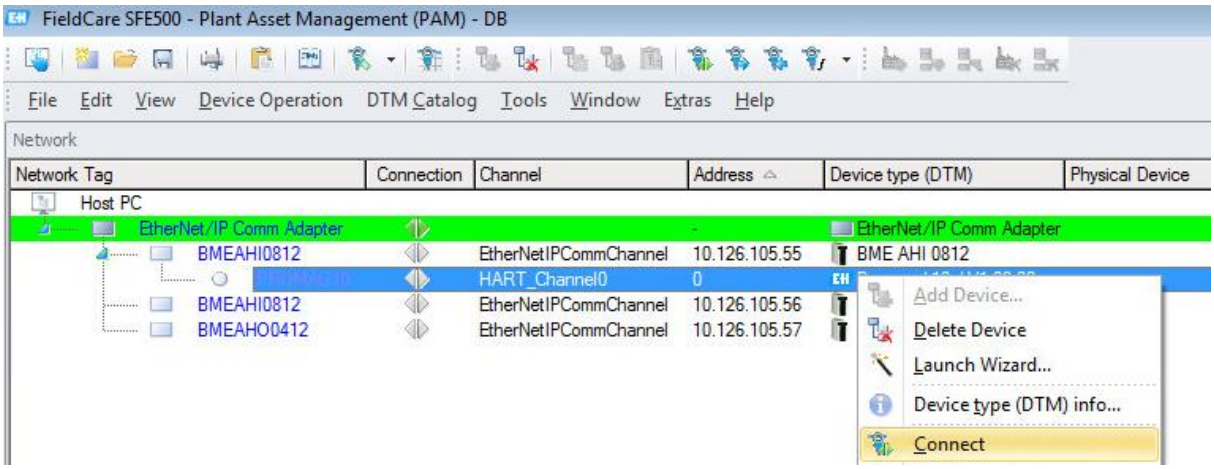
6.4 DeviceDTM Online Connection

- Before connecting the HART module comDTM and deviceDTMs, make sure that all HART function blocks are deactivated in the program. Please refer to chapter 3.2.5.1.

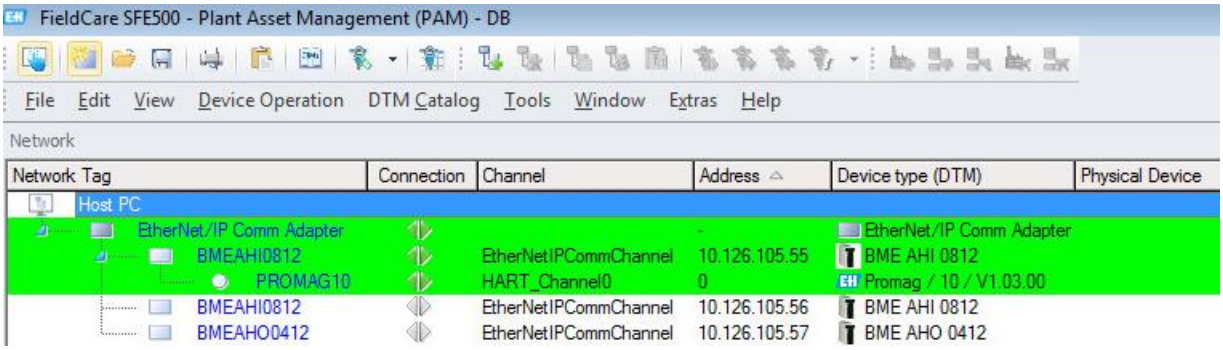


6.4.1 Endress+Hauser PROMAG 10 Online Connection

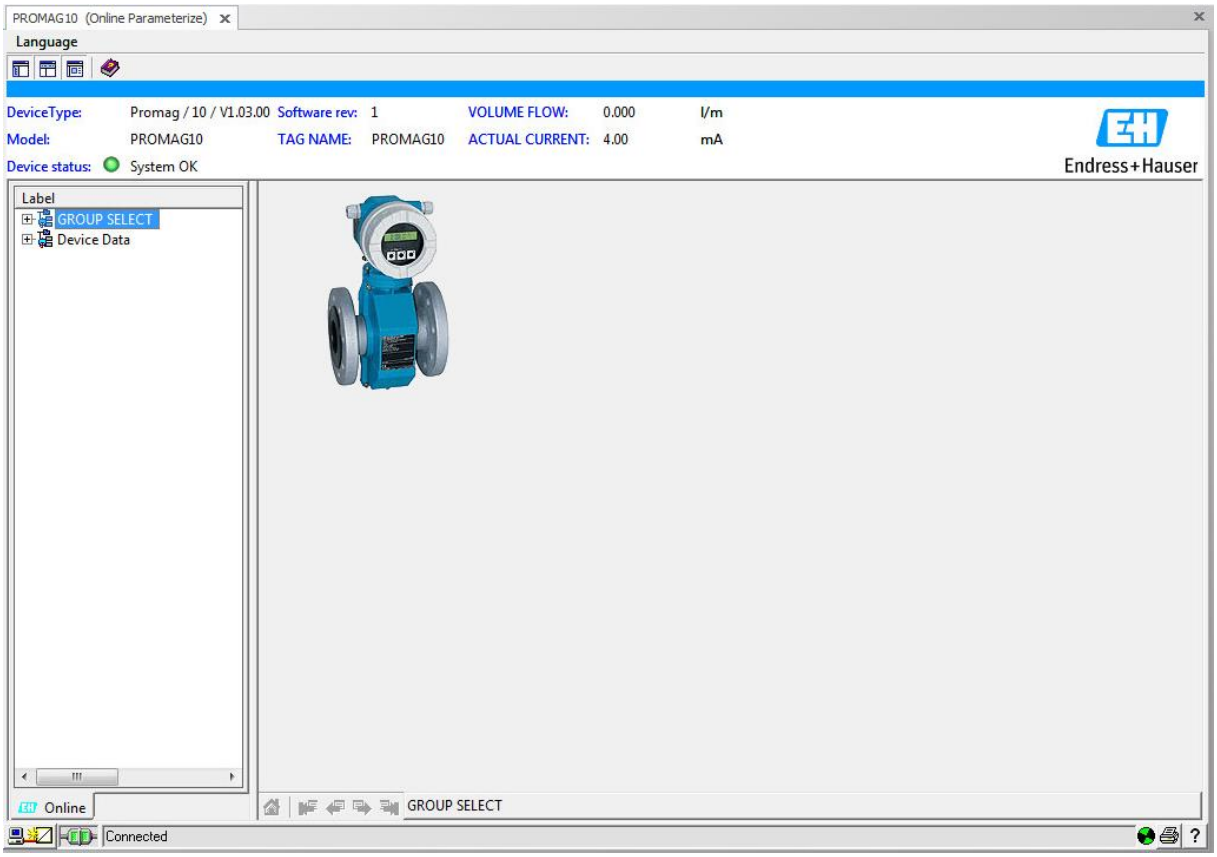
- Right-click on the deviceDTM "Promag/10/V1.03.00" and select the menu "Connect":



- Connected device:

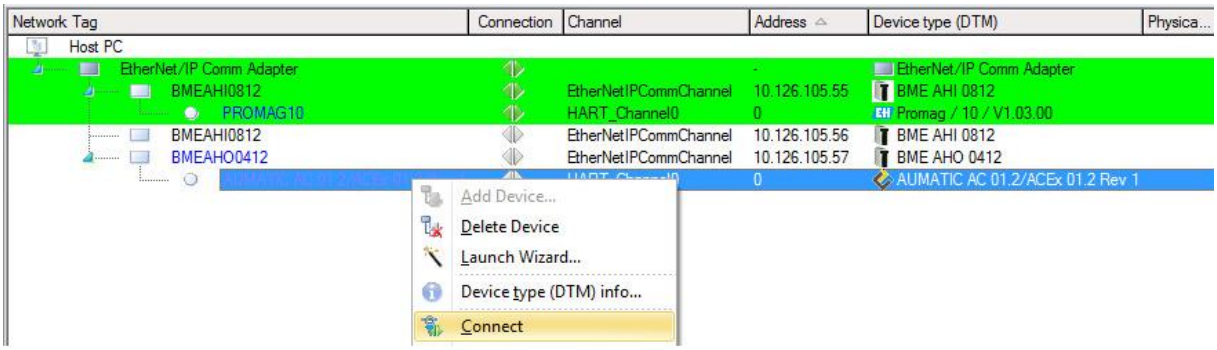


- Double-click on the device DTM “Promag/10/V1.03.00”. This opens the device DTM window:

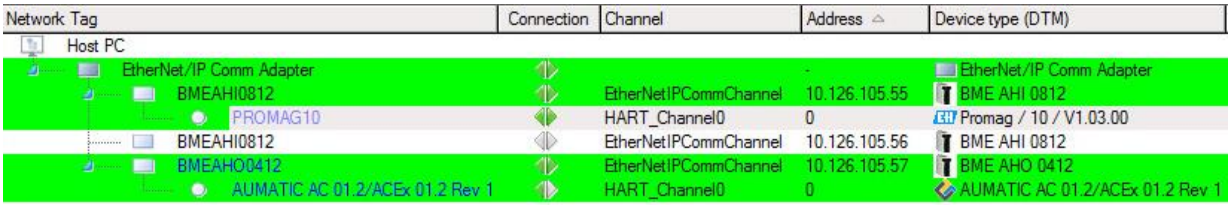


6.4.2 AUMA Actuator Online Connection

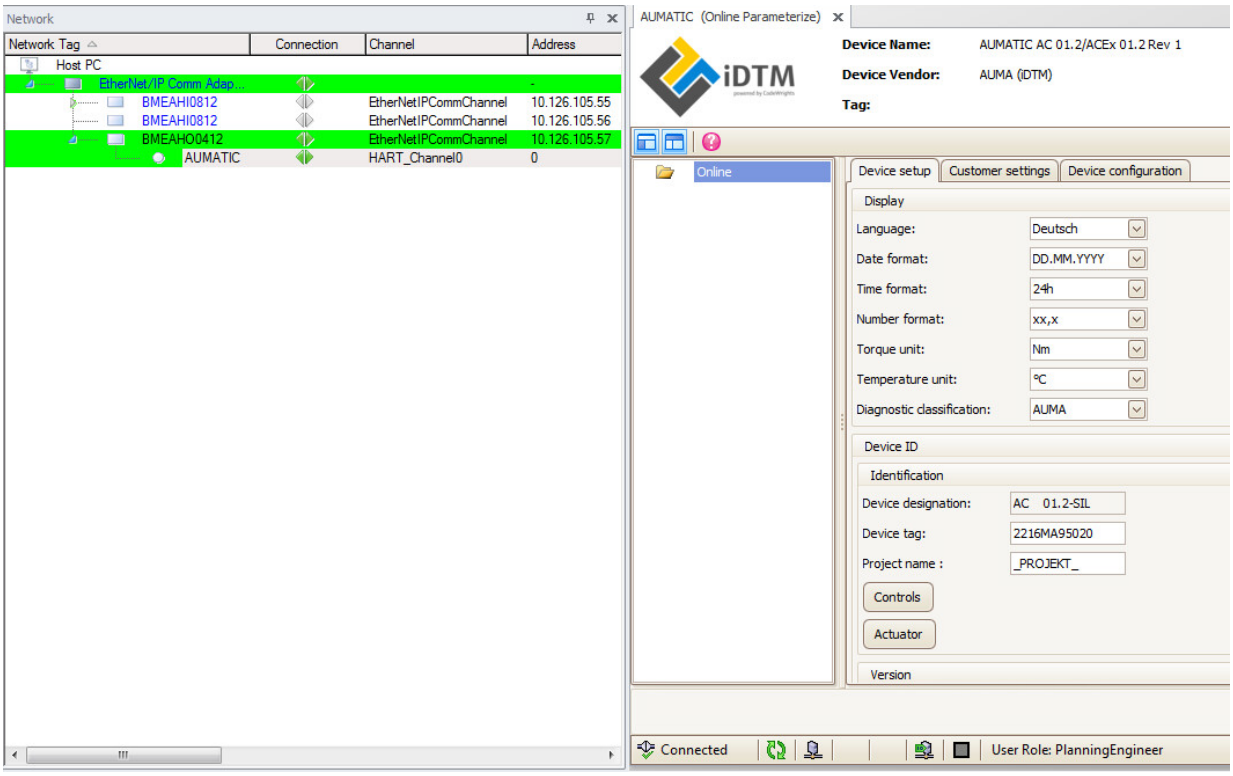
- Right-click on the deviceDTM “AUMATIC AC01.2/ACEx 01.2 Rev1” and select the menu “Connect”:



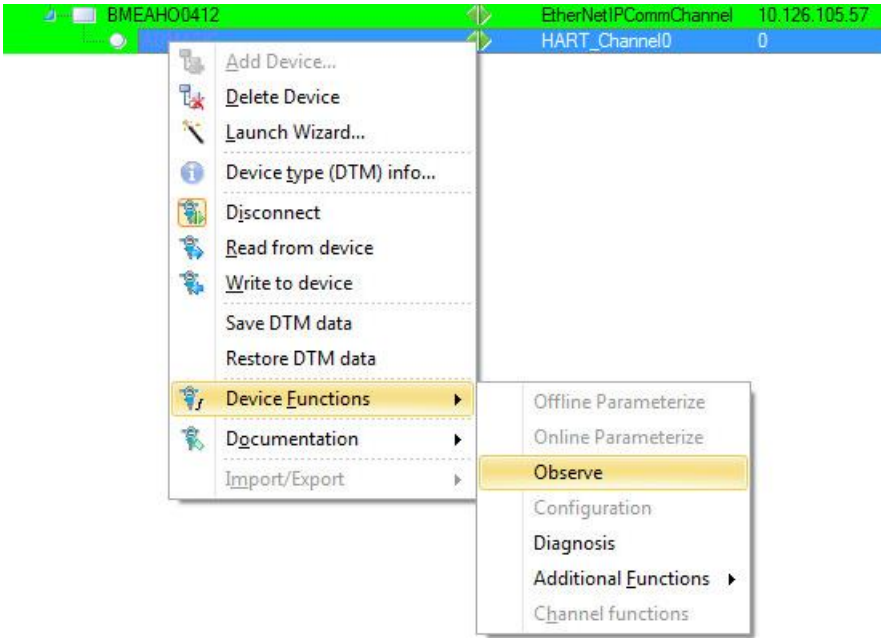
- Connected device:



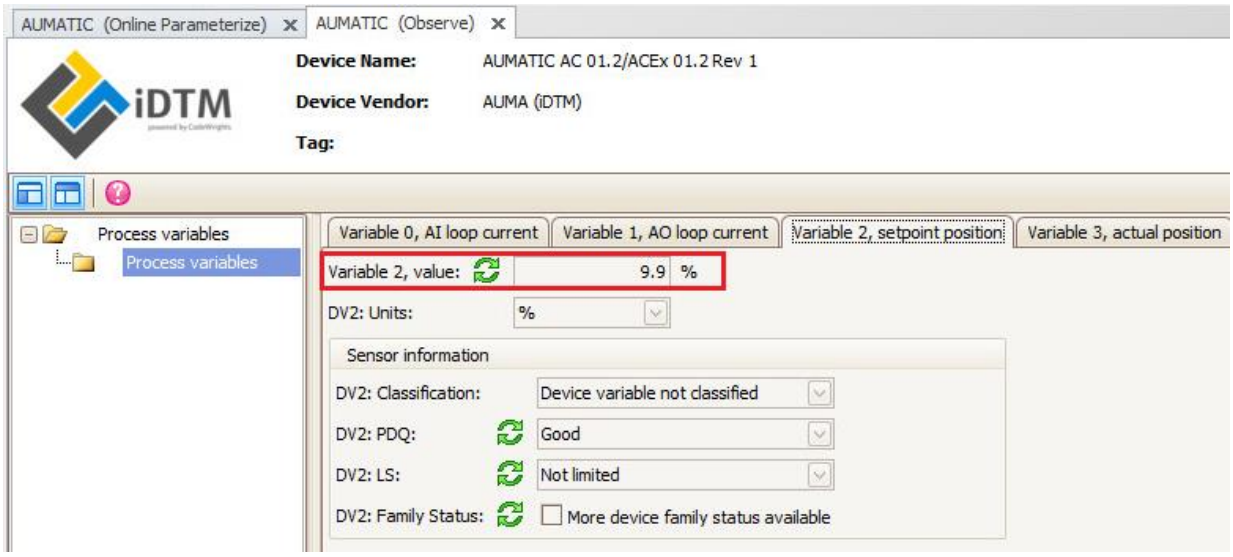
- Double-click on the device DTM “Promag/10/V1.03.00”.
  This opens the device DTM window:



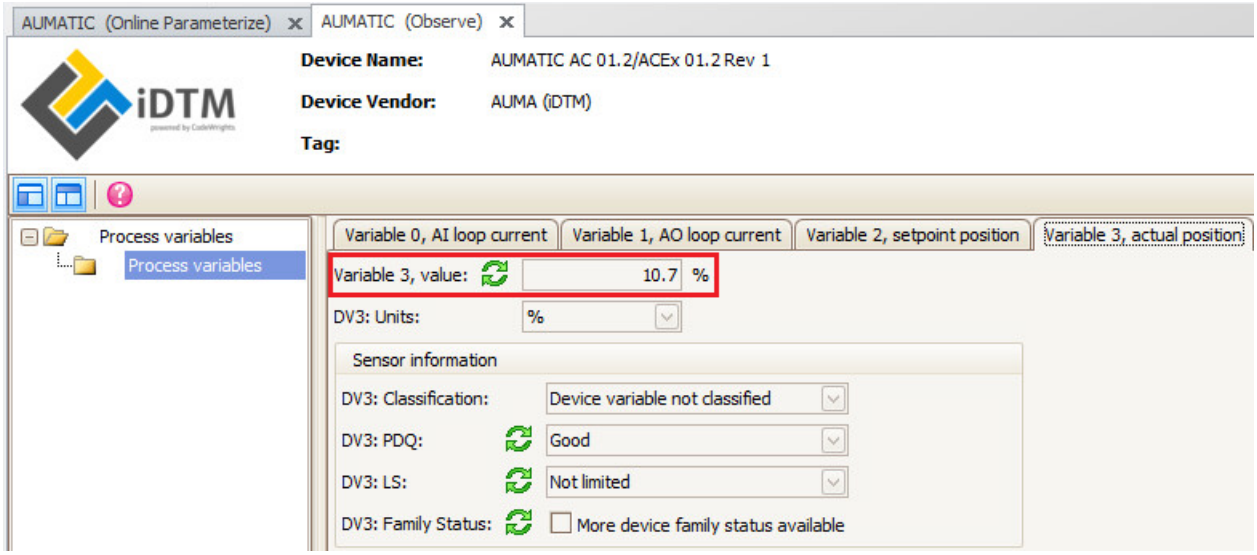
- Right-click on the AUMA deviceDTM and select the option “Observe” to display the process variables:



- Value of the configured set point done in chapter 3.4.2.2:



- Value of the configured feedback. This is the same value as in chapter 3.4.2.2:



www.endress.com/open-integration