Engineering Note

IND360 Precision

PROFIBUS PLC

METTLER TOLEDO

Contents

1. Overview

This Engineering Note is based on integration of Mettler Toledo's Compact Automation Transmitter/ Terminal IND360 Precision with a Profibus PLC. Go to www.mt.com/ind-ind360-downloads to download all the necessary files and documents.

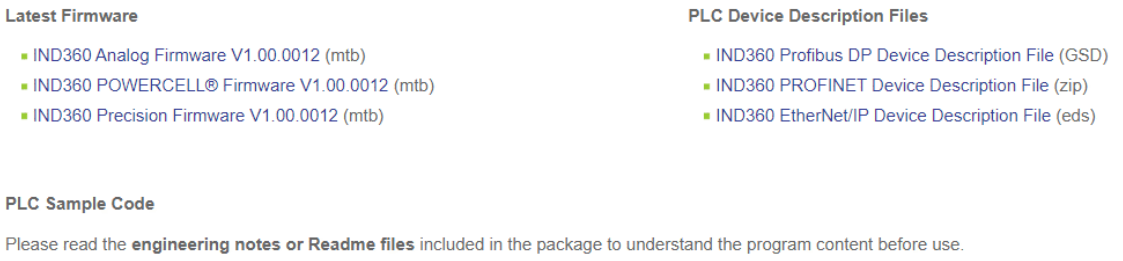


Figure 1-1： IND360 download page

Note: The configuration used in this sample code is based on the default settings：

Siemens TIA Portal V14 SP1

SAI data format: 2-Block format

GSD file： MT1153.GSD

IND360 device firmware version： IND360_Precision_V1.00.0012.mtb

It is recommended to integrate one IND360 Precision into the PLC Profibus network and go through the sample code to understand the functionality of each Function Block. To add more IND360 Precision into the Profibus network, follow the steps listed in Chapter 6. Steps to Add New IND360s.

2. Setup of Project Development Environment

2.1. Hardware Integration

Connect the PLC and IND360 with the appropriate PROFIBUS DP cable.

2.2. Open the Sample Code

To open and use this sample code "IND360_PRECISION_PBDP.ap14", you need to use Siemens TIA Portal version 14 SP1 or higher. All the required GSD files will be installed automatically when opening the sample code.

2.3. Switching Project Languages

Under Tools -> Project Languages -> Editing Language, choose the preferred language for your project. Selections are English (United States) and Chinese (People's Republic of China).

Figure 2-1： Switching Project Languages between English and Chinese

2.4. Select the correct controller model

There are two projects included in one sample code, each project uses different Siemens PLC model:

1. "S7-1200" uses S7-1200 series PLC with IND360 Precision Profibus;
2. "S7-1500" uses S7-1500 series PLC with IND360 Precision Profibus

Choose the most relevant project according to your PLC type to download into the PLC.

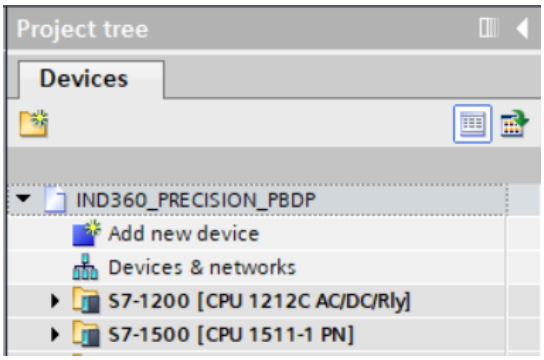


Figure 2-2: Three projects included in one sample code

To change the PLC model: Go to Device Configuration under the project folder, right click on the current controller, select "Change Device" and choose the new controller as well as its firmware version.

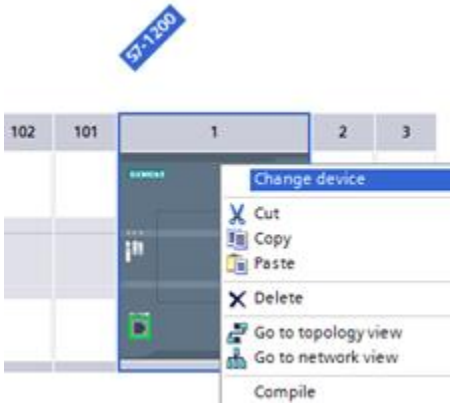


Figure 2-3: Change the controller type

Compile and download the project into the controller.

Under the Devices & networks, the IND360 Precision Profibus Node Address is set as "3". Make sure the same node address is also set up at the physical unit. As for the PLC, its Profibus Node Address is "2" and IP Address is "192.168.0.10".

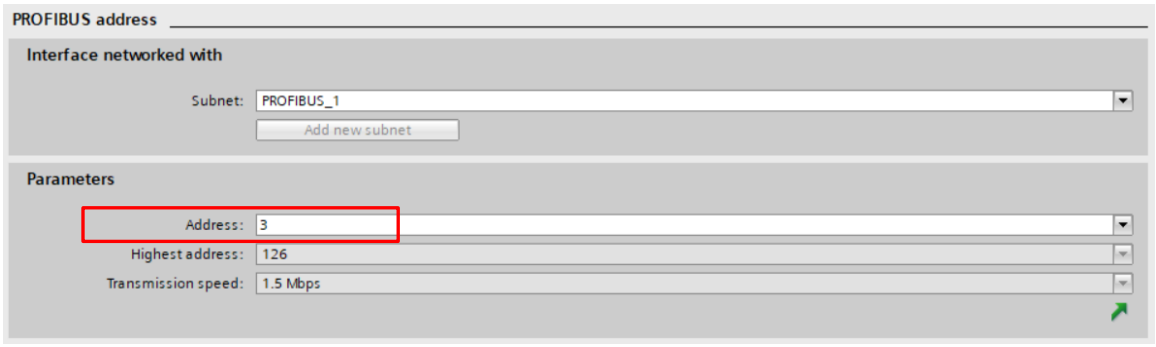


Figure 2-4: IND360 Device Properties

Figure 2-5: PLC Device Properties

Select the "MT_IND_Application" program, click on "Go Online" button to start using the sample code.

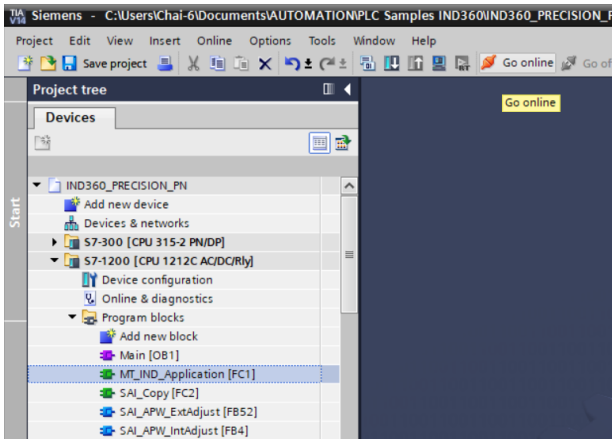


Figure 2-6: go online with MT_IND_Application

3. SAI Data Structure in Device Overview

In the Device Overview, the SAI input and output data structure has been assigned with the respective I and Q addresses as shown below. For more details on SAI data structure, please refer to the Standard Automation Interface – User Guide and Reference Guide English (pdf), which are downloadable from the IND360 Download Page.

Full User Manual

- IND360base Weighing Terminal Full Users Guide (pdf)
- IND360tank/vessel Weighing Terminal Full Users Guide (pdf)

- Standard Automation Interface - Reference Guide - IND360 English (pdf)
- Standard Automation Interface - User Guide - IND360 English (pdf)

Figure 3-1: the SAI User Manual on the IND360 Download page

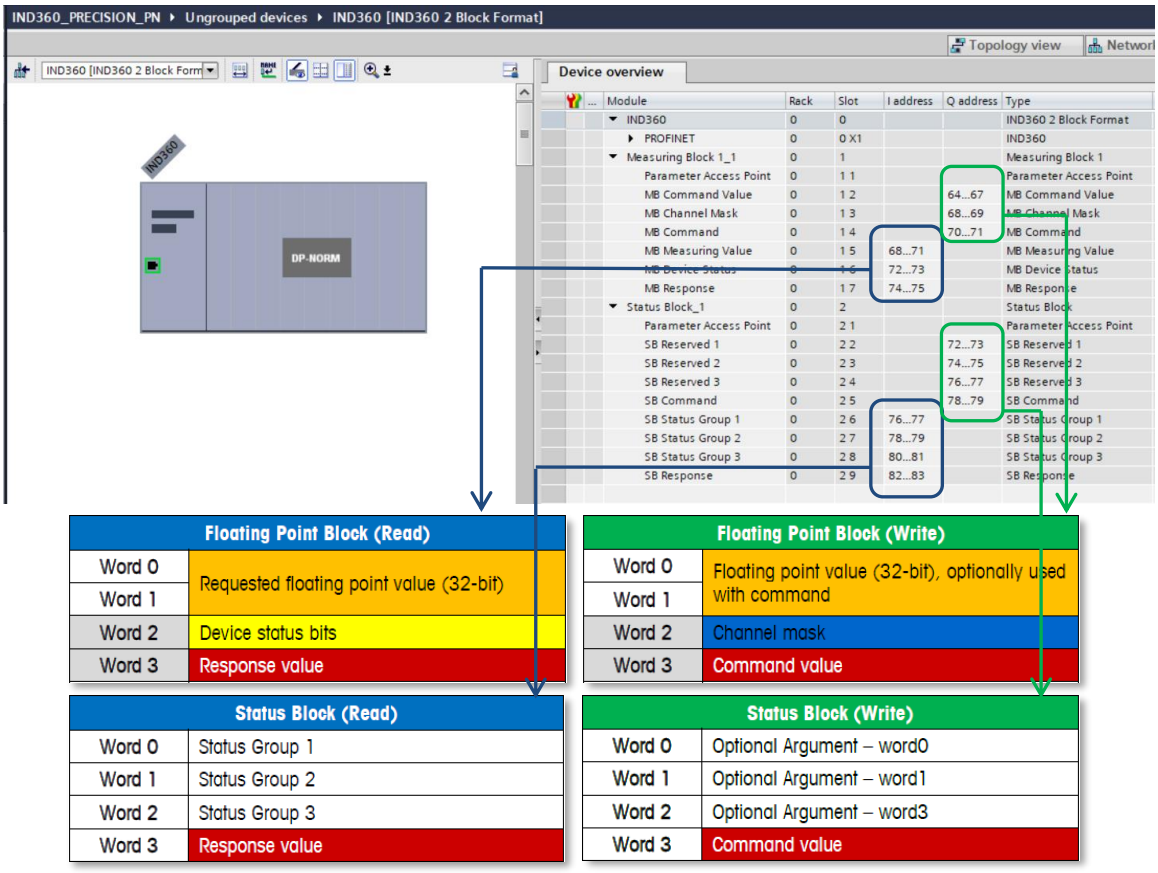


Figure 3-2: SAI Data Structure as shown in the Device Overview

These I and Q addresses shown above will be used as input parameters in 4. Function Blocks

4. Function Blocks

About the "ID" input parameter for all the acyclic communication function blocks:

For all the function blocks which involve acyclic communication between the PLC and the weighing transmitter, the "ID" input parameter is required. Examples of function block with acyclic communication are internal adjustment and external adjustment.

For S7-1200 and S7-1500 PLCs, the ID is the Hardware Identifier pointing to the device's Rack 0/ Slot 1. This also can be found in the PROFIBUS Device view.

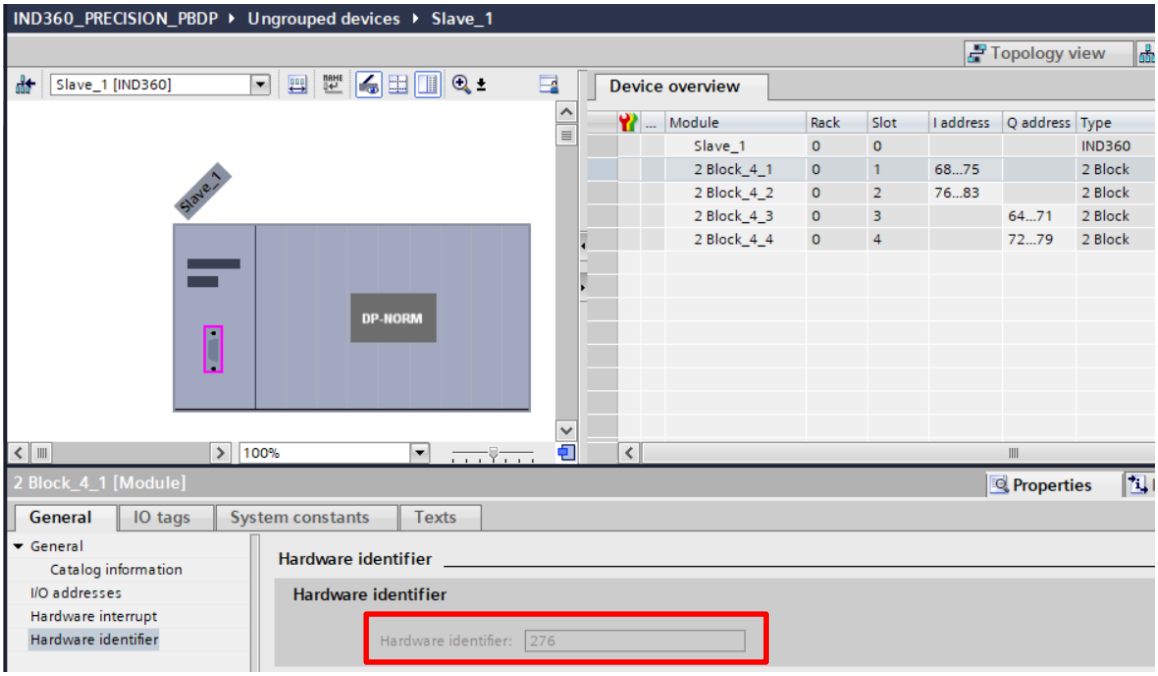


Figure 4-1: the ID parameter for S7-1200 and 1500

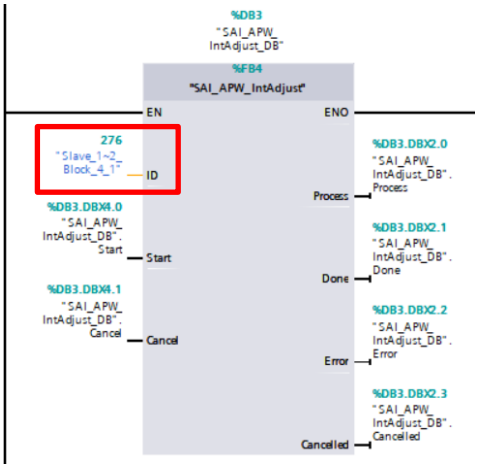


Figure 4-2: the ID parameter for acyclic communication

4.1. Cyclic Weight Data Processing

This function block reads in all the important real-time, cyclical weighing data such as weight value, Data OK bit, Motion bit, Net mode bit and critical alarm bit.

Set the scale command bit one at a time to trigger different commands such as tare stable, zero stable, tare immediate, zero immediate, preset tare and clear tare. A successful execution of a scale command will set the Done bit on, else the Error bit will be set on instead.

The cyclic weight data can be reported automatically right after any scale command. The type of weight data (gross, net, or tare) being reported depends on the setting for WeightCmd. By default, the WeightCmd is decimal "3" and the function block will return a net weight value every time after any scale command such as tare or zero. Similarly, if the WeightCmd parameter is configured as decimal "0" or "1" the function block will then return a gross weight after any scale command.

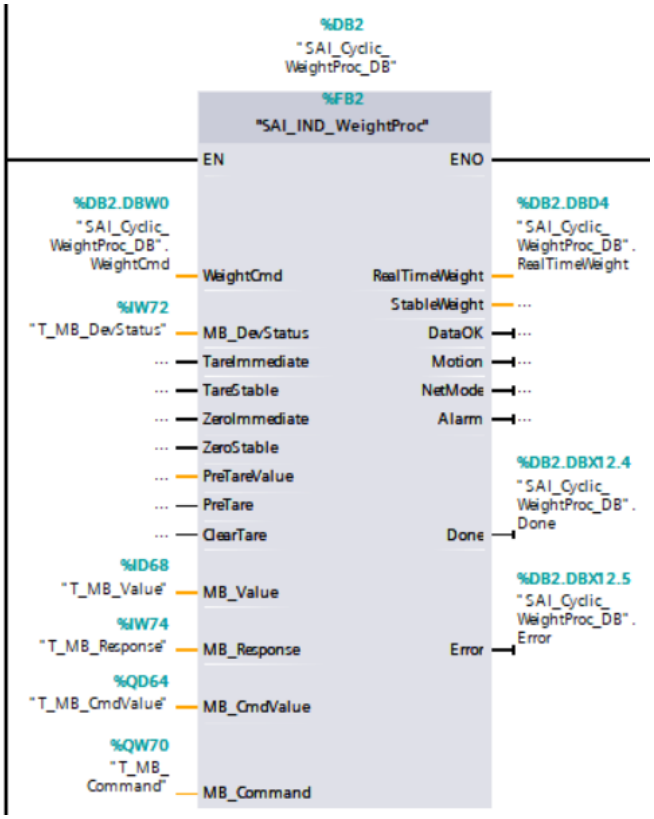


Figure 4-1: SAI_IND_WeightProc Function Block

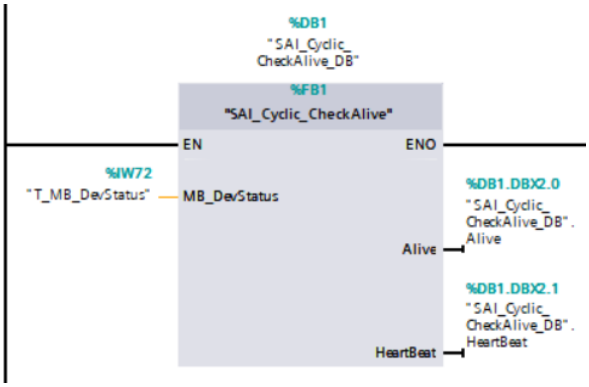
Table 4-1: SAI_IND_WeightProc Function Block Parameters

Input Parameters	Data Type	Values	Description
WeightCmd	Word	0, 1	Report gross weight value
		2	Report tare weight value
		3 (default)	**Report net weight value**
		5	Report gross weight value (with internal resolution)
		6	Report tare weight value (with internal resolution)
		7	Report net weight value (with internal resolution)
MB_DevStatus	Word		Refer to Device Overview, input address of MB Device Status
TareImmediate	Bool		Trigger this bit to perform immediate tare command. This tare command doesn't check for stability criteria. Upon completion of this command, the input bit will be reset.
TareStable	Bool		Trigger this bit to perform stable tare command. This tare command requires the weight value to remain stable within the stability criteria (configurable in web browser or APW-Link) for a predefined timeout range, failing which, the command will return an error. Upon completion of this command, the input bit will be reset.
ZeroImmediate	Bool		Trigger this bit to perform immediate zero command. The zero command can only be executed when the weight value is within the zero range. Else, the command will return an error. Upon completion of this command, the input bit will be reset.
ZeroStable	Bool		Trigger this bit to perform a stable zero command. This zero command requires the weight value to remain stable within the

			stability criteria (configurable in web browser or APW-Link) for a predefined timeout range. Furthermore the weight value has to be in the zero range to trigger this command, failing either condition; the command will return an error. Upon completion of this command, the input bit will be reset.
PreTareValue	Real		The preset tare value which has to be configured before issuing the PreTare command. Valid PreTare value is between scale's zero point up to maximum capacity.
PreTare	Bool		Trigger this bit to perform a preset tare command. The PreTareValue has to be configured prior to issuing this PreTare command. Upon completion of this command, the input bit will be reset.
ClearTare	Bool		Trigger this bit to perform a clear tare command. This command removes the tare and brings the scale into gross mode. Upon completion of this command, the input bit will be reset.
MB_Value	Real		Refer to Device Overview, input address of MB Measuring Value
MB_Response	Word		Refer to Device Overview, input address of MB Response
MB_CmdValue	Real		Refer to Device Overview, output address of MB Command Value
MB_Command	Word		Refer to Device Overview, output address of MB Command
Output Parameters	Data Type	Values	Description
RealTimeWeight	Real		Real-time weight value, can be gross, tare or net weight
StableWeight	Real		Stable weight value, the last real-time weight during Motion = 0
DataOK	Bool	0	This bit gets set to 0 when the device is still operational but the value being reported cannot be guaranteed to be valid. The following conditions cause the Data Okay bit to be set to 0: • Device is powering up • Device is in setup mode • Device is in test mode • Over capacity condition occurs - When the A/D converter is at its limit - Product dependent over capacity that occurs when the device determines it cannot trust the weight • Under capacity condition occurs - When the A/D converter is at its limit - Product dependent under capacity that occurs when the device determines it cannot trust the weight
		1	Weight data is normal, valid
Motion	Bool	0	Weight value is stable
		1	Weight value is in motion
NetMode	Bool	0	Weighing is in gross mode
		1	Weighing is in net mode
Alarm	Bool	0	No alarm
		1	Also called the RedAlert alarm. If this bit is true it is an indication that the control device should stop until the source of the alarm is evaluated and corrected. The control system should use a Field Value command or evaluate the RedAlert status block to determine the nature of the alarm.
Done	Bool	0	Zero, tare or clear tare command is in process, or failed
		1	Zero, tare or clear tare command is successful
Error	Bool	0	Zero, tare or clear tare command is in process, or succeeded
		1	Zero, tare or clear tare command is not completed due to error

4.2. Device Heart Beat Monitoring

This function block monitors the Heart Beat bit of the weighing transmitter and outputs an "Alive" flag.



S7-1200 & S7-1500

Figure 4-2: SAI_Cyclic_CheckAlive Function Block

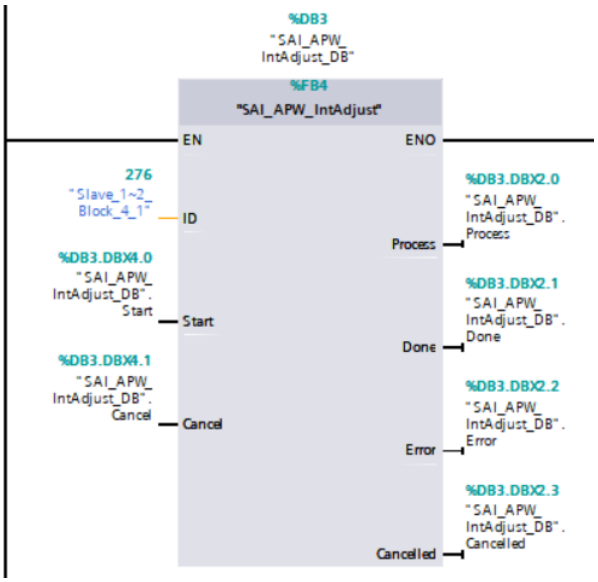
Table 4-2: SAI_Cyclic_CheckAlive Function Block Parameters

Input Parameters	Data Type	Values	Description
MB_DevStatus	Word		Refer to Device Overview, input address of MB Device Status
Output Parameters	Data Type	Values	Description
Alive	Bool	0	Device has lost communication
		1	Device is communicating OK
HeartBeat	Bool		To insure that the device is working as expected and updating data in Words 0, 1 and 2, this heart beat bit is toggled between off and on states. The frequency is dependent on the specific device's ability to cycle this bit. For example, a 1 second heart beat would be sufficient for most applications.

4.3. Internal Adjustment

Automated Precision Weighing "APW" module with internal adjustment feature can be adjusted without applying any external weight.

Trigger the "Start" input bit to start the internal adjustment process. Upon completion of the adjustment process, this "Start" bit will be reset.



S7-1200 & S7-1500

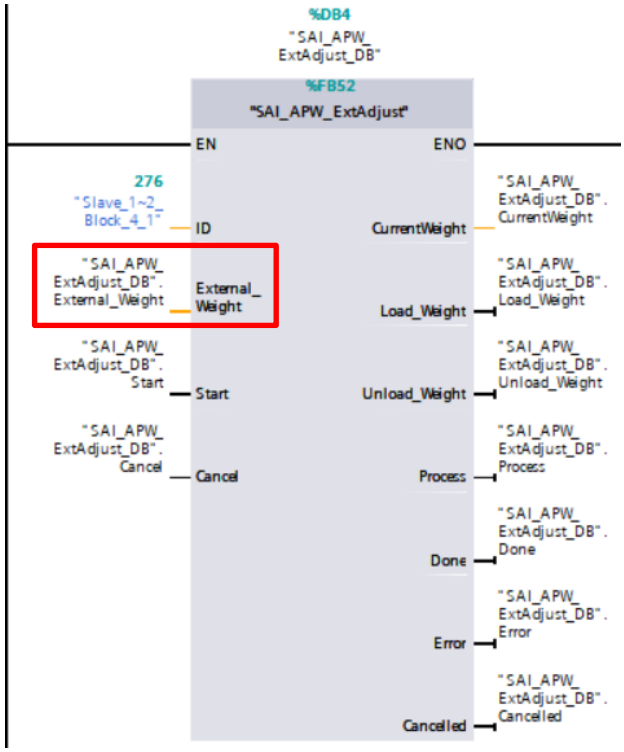
Figure 4-7: SAI_APW_IntAdjust Function Block

Table 4-5: SAI_APW_IntAdjust Function Block Parameters

Input Parameters	Data Type	Values	Description
ID (for S7-1200 and S7-1500)	HW_IO	Example: "Slave_1~2 _Block_4_1"	ID parameter to select the module for which a data record is to be written. Use only the hardware identifier (HW ID) of the module for the ID parameter. In this sample program, the ID parameter of the device can be found under Device Properties > Slot 0 Hardware Identifier.
Start	Bool	1, 0	Trigger this input bit to start the adjustment process.
Cancel	Bool	1, 0	Trigger this input bit to cancel the adjustment process.
Output Parameters	**Data Type**	**Values**	**Description**
Process	Bool	1	Adjustment is started and in process
		0	Adjustment is not started
Done	Bool	1	Adjustment is completed successfully
		0	Adjustment is in process or in error state
Error	Bool	1	Adjustment failed due to error
		0	No error
Cancelled	Bool	1	Adjustment is cancelled successfully
		0	-

4.4. External Adjustment

Use this Function Block to perform scale adjustment with external weight.



S7-1200 & S7-1500

Figure 4-8: SAI_APW_ExtAdjust Function Block

Configure the "External_Weight" according to the adjustment weight to be used. The weight unit is gram by default. Trigger the Start input bit to run the adjustment procedure.

Table 4-7: SAI_APW_ExtAdjust Function Block Parameters

Input Parameters	Data Type	Values	Description
ID (for S7-1200 and S7-1500)	HW_IO	Example: "Slave_1~2 _Block_4_1"	ID parameter to select the module for which a data record is to be written. Use only the hardware identifier (HW ID) of the module for the ID parameter. In this sample program, the ID parameter of the device can be found under Device Properties > Slot 0 Hardware Identifier.
External_Weight	REAL (32 bits)	Example: "200.00"	This is the span weight value for the sensitivity adjustment.
Start	Bool	1, 0	Trigger this input bit to start the adjustment process.
Cancel	Bool	1, 0	Trigger this input bit to cancel/ abort the adjustment process after being started.
Output Parameters	**Data Type**	**Values**	**Description**
CurrentWeight	REAL (32 bits)	Example: "200.00"	The required reference weight here is shown here.
Load_Weight	Bool	1	User has to load the external weight according to the value displayed in CurrentWeight.
		0	No action required from the user
Unload_Weight	Bool	1	User has to unload the external weight

		0	No action required from the user
Process	Bool	1	Adjustment is started and in process
		0	Adjustment is not started
Done	Bool	1	Adjustment is completed successfully
		0	Adjustment is in process or in error state
Error	Bool	1	Adjustment failed due to error
		0	No error
Cancelled	Bool	1	Adjustment is cancelled successfully
		0	No cancellation

5. Sample Code Migration

5.1. Hardware Configurations

1) Under Devices & networks > Network view, add (or drag over) an IND360.

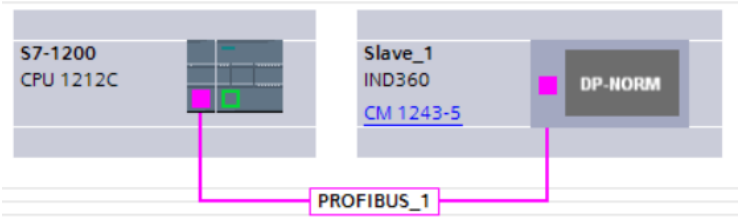


Figure 5-1: Add a Profibus device in the Network view

2) Under Devices & networks > Device view, select the 2 Block data structure from the Hardware Catalog.

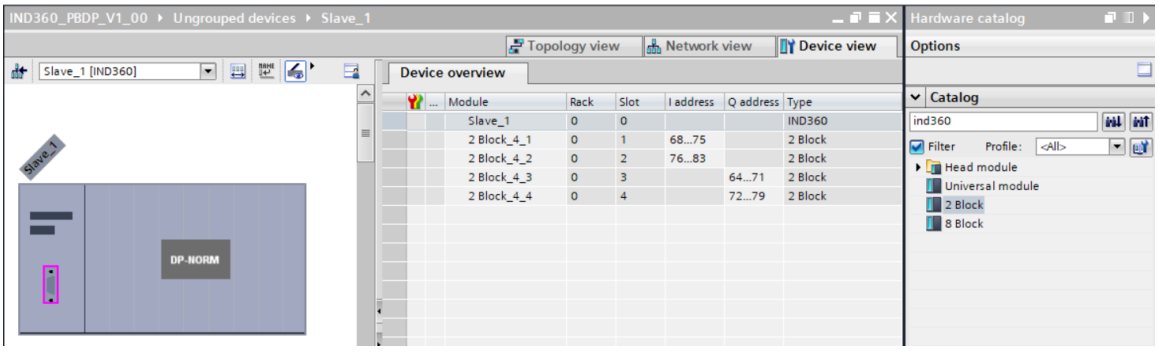


Figure 5-2: Select the 2 Block data structure

3) Assign the Profibus Node Address for the IND360.

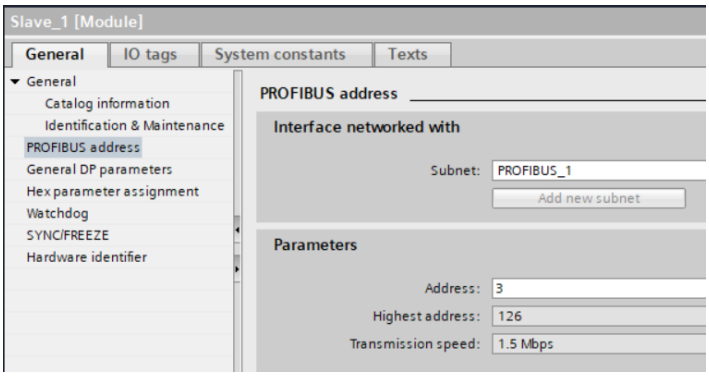
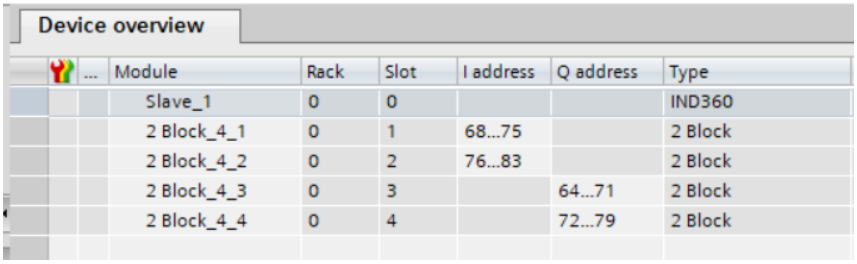


Figure 5-3: Assign the Profibus Node Address

4) The sample code is following the default I and Q addresses assignment as shown below. To minimize the modification to the code, consider sticking to the same I and Q address assignment.



...	Module	Rack	Slot	I address	Q address	Type
	Slave_1	0	0			IND360
	2 Block_4_1	0	1	68...75		2 Block
	2 Block_4_2	0	2	76...83		2 Block
	2 Block_4_3	0	3		64...71	2 Block
	2 Block_4_4	0	4		72...79	2 Block

Figure 5-4: The Profibus Device I and Q Addresses

5.2. PLC Settings

1) Under the PLC device properties -> System and clock memory, tick "Enable the use of system memory byte" (this feature is not available in the S7-300 series PLC).

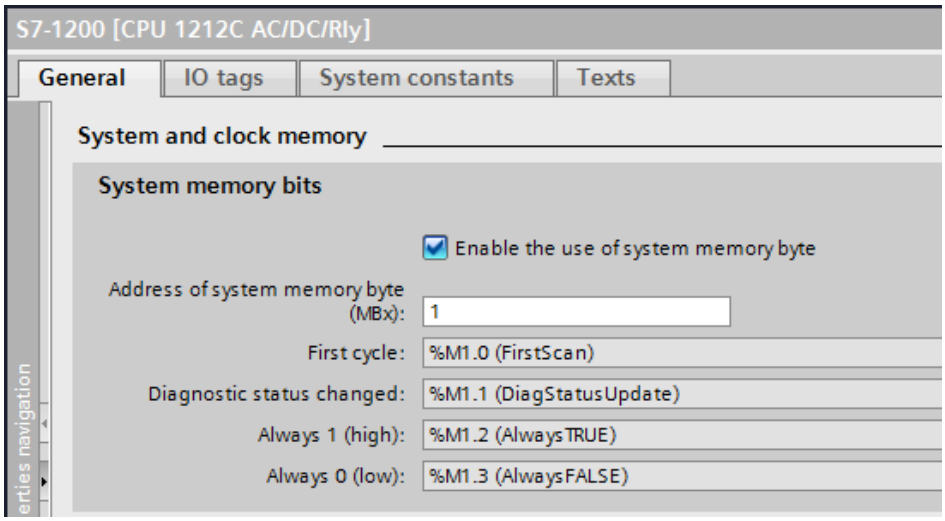


Figure 5-5: Enable system memory byte

5.3. Duplicate Programming Files

1) The required program blocks:

   a) MT_IND_Application(FC)

   b) SAI_Copy(FC)

   c) SAI_IND_WeightProc(FB), SAI_IND_WeightProc_DB

   d) SAI_Cyclic_CheckAlive(FB), SAI_Cyclic_CheckAlive_DB

   The function blocks below are used to perform scale adjustment from the PLC. All variants of IND360 now support scale adjustment via keypad menu, APW-Link and web browser.

   a) SAI_APW_IntAdjust(FB), SAI_APW_IntAdjust_DB

   b) SAI_APW_ExtAdjust(FB), SAI_APW_ExtAdjust_DB

2) Delete the other unused program blocks in MT_IND_Application.

3) Duplicate the "ACT" under the PLC tags.

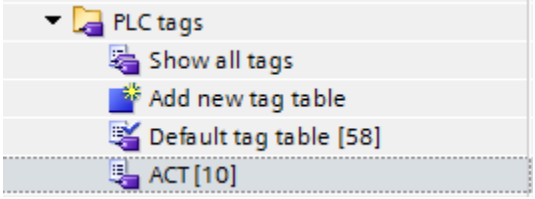


Figure 5-6: Duplicate the PLC tags

4) Duplicate all the PLC data types.

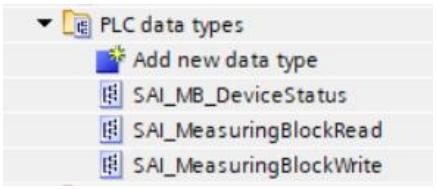


Figure 5-9: Duplicate the PLC data types

5) Lastly, in the Main (OB1) call up the function "MT_IND_Application".

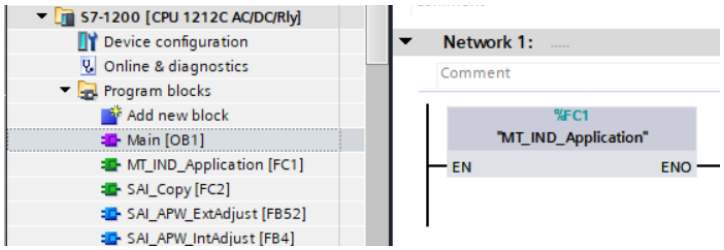


Figure 5-10: Call up "MT_IND_Application" in the Main OB

6. Add New IND360

All the devices in a PROFIBUS subnet must have different Node Address. Only when all the modules in a subnet have different addresses and your actual structure matches that of the network configuration produced, should you load the settings across the network.

Reserve the PROFIBUS address "0" for a main controller or DP master. Allocate a unique PROFIBUS address between 1 and 126 for each DP slave or other DP master in the network.

1) On the IND360, press the ENTER key for 3 seconds to access the device menu. Go to Communication > PROFIBUS DP > Node Address, configure the Node Address.

   The Profibus DP node address can also be configured in the web browser as shown below:

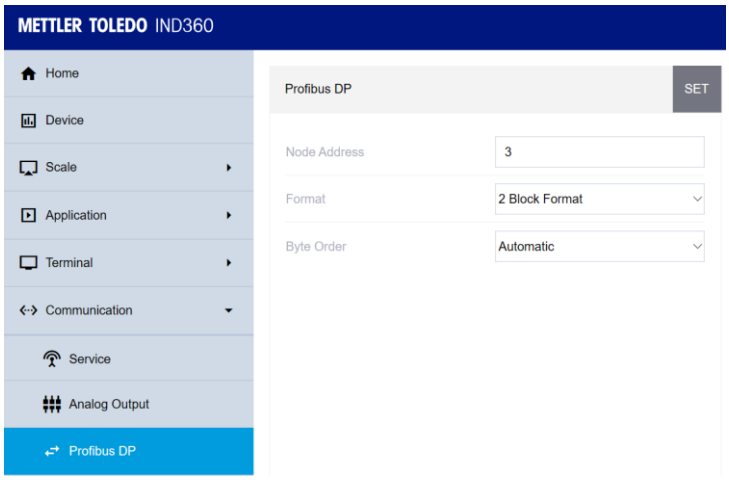


Figure 6-1: Change the PROFIBUS Node Address in Web Browser

2) At Devices & networks, add another IND360 from the Hardware Catalog.

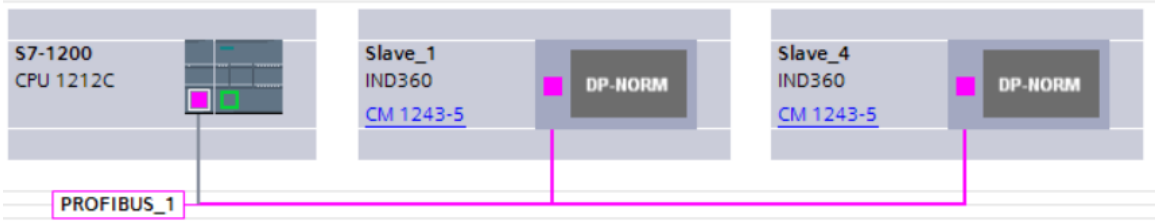


Figure 6-2: Add a new IND360

3) At Devices & networks > Device view, select the 2 Block data structure.

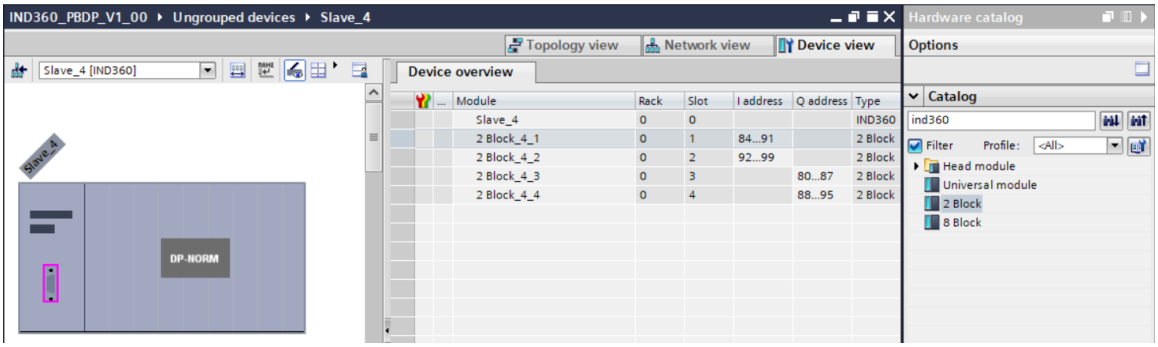


Figure 6-3: Select the 2 Block data structure

4) Assign a different PROFIBUS Node Address in ascending order.

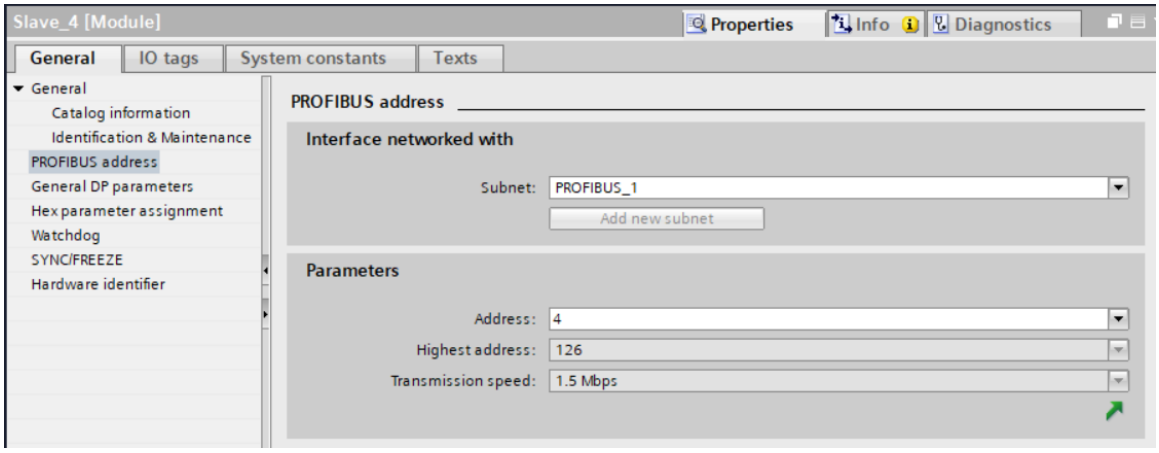


Figure 6-4: Allocate the PROFIBUS Node Address

5) When necessary, edit the automatically allocated I and Q addresses of the PROFIBUS device.

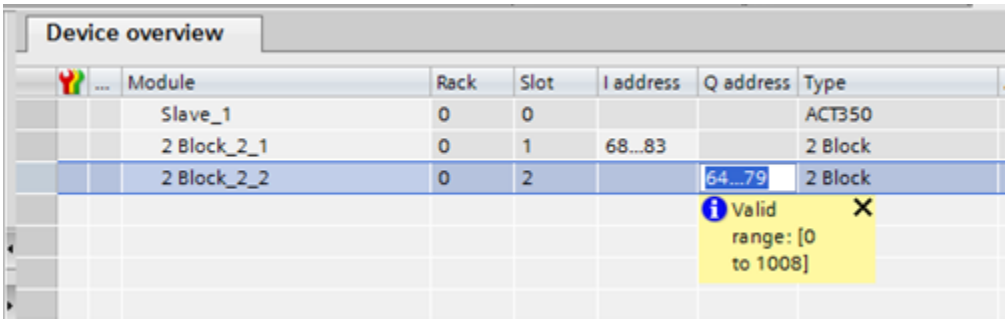


Figure 6-5: Allocate the I and Q addresses for the new device

6) Duplicate the function blocks, and configure all the required input and output parameters. Each function block FB must have an independent data block DB. As shown below, there are two SAI_IND_WeightProc function blocks but both FBs are assigned with different DBs which are SAI_IND_WeightProc_DB (DB2) and SAI_IND_WeightProc_DB_1 (DB5)

A small trick can be used here to add adjacent function block, drag the function block from the Project Tree side window into the destination network.

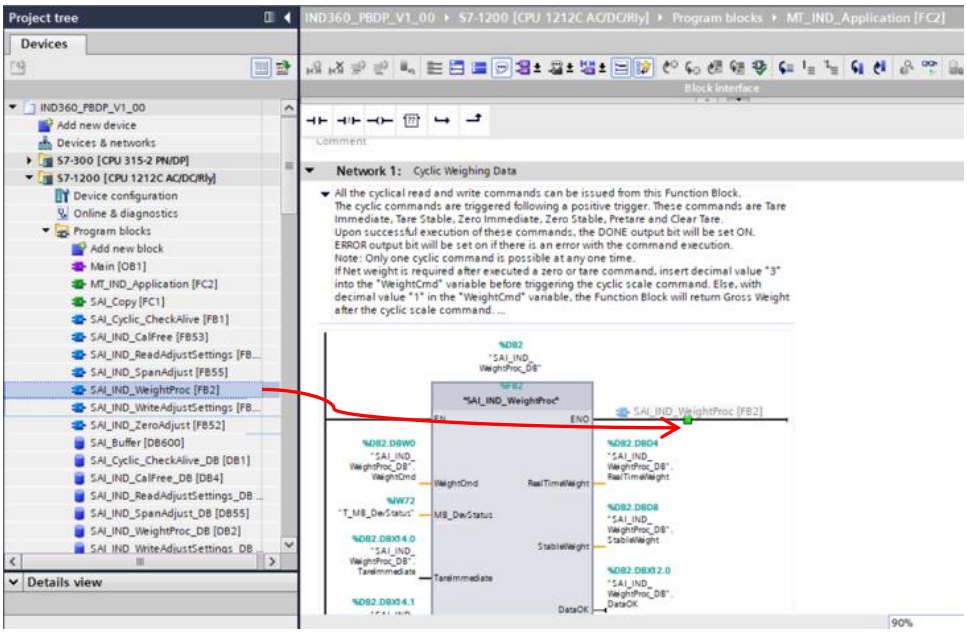


Figure 6-6: Duplicate function block for additional device

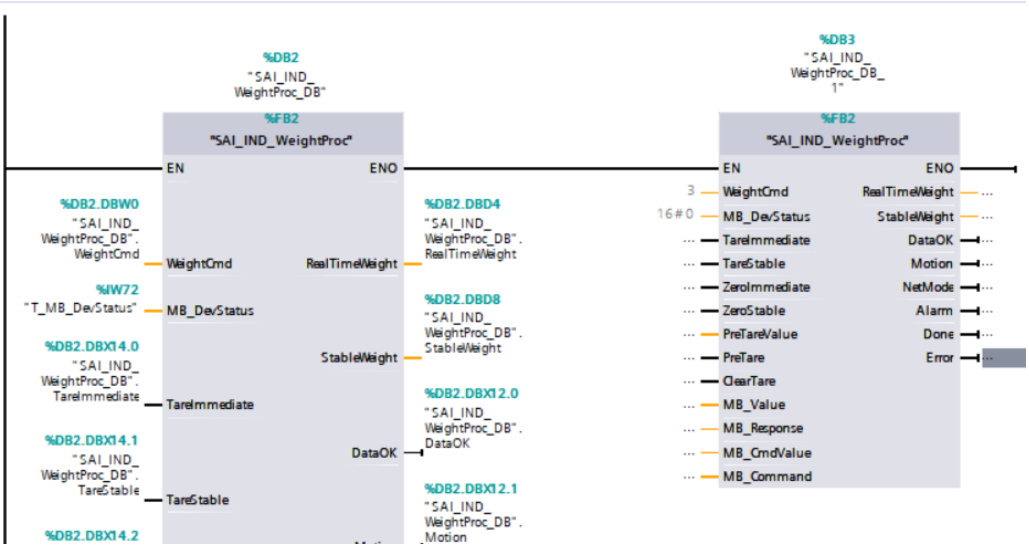


Figure 6-6: Two function blocks of the same type, but different data blocks

7) Repeat steps 1 – 5, until all the new IND360s have been integrated into the PROFIBUS network.