Engineering Note

IND360 APPLICATIONS

PROFINET PLC

METTLER TOLEDO

Contents

1 Overview

This Engineering Note demonstrates the integration of IND360 applications with a PROFINET PLC. Visit www.mt.com /ind-ind360-downloads to download the necessary files and documents. The IND360 terminal supports tank / vessel, fill-dose, dynamic weighing and rate control weighing applications.

Latest Firmware

- IND360 Firmware Release History Chart

Latest Firmware

- IND360 Analog Firmware V1.00.0013 (mtb)
- IND360 POWERCELL® Firmware V1.00.0013 (mtb)
- IND360 Precision Firmware V1.00.0013 (mtb)
- IND360legacy Analog Firmware V1.00.0003 (mtb)

PLC Device Description Files

- IND360 Profibus DP Device Description File (GSD)
- IND360 PROFINET Device Description File (zip)
- IND360 EtherNet/IP Device Description File (eds)
- Rockwell Custom AOP V 1.22.05 (EtherNet/IP)

PLC Sample Code

Please read the **engineering notes or Readme files** included in the package to understand the program content before use.

IND360base Weighing Terminals

Analog

- IND360base Analog Profibus Sample Code V1_00 (zip)
- IND360base Analog PROFINET Sample Code V1_01 (zip)
- IND360base Analog EtherNet/IP Sample Code V1_00 (zip)

IND360tank/vessel Weighing Terminals

Analog

- IND360tank/vessel Analog Profibus Sample Code V1_00 (zip)
- IND360tank/vessel Analog PROFINET Sample Code V1_00 (zip)
- IND360tank/vessel Analog EtherNet/IP Sample Code V1_00 (zip)

Figure 1-1: IND360 download page

Note: This sample code is based on the following configuration:

Siemens TIA Portal:	V14 SP1
SAI data format:	**2-Block format** (default); 8-Block format
Device Name:	(to be assigned by PLC)
IP Address:	(to be assigned by PLC)
GSDML file :	GSDML-V2.35-MT-IND360-20200527.xml

It is recommended to integrate one IND360 into the PLC PROFINET network and go through the sample codes to understand the functionality of each Function Block.

2 Setup of Project Development Environment

2.1. Hardware Integration

There are distinct Ethernet ports for service (web configuration) and operation (Industrial Ethernet) on the IND360. Connect the PLC to one of the IND360 Industrial Ethernet network ports.

2.2. Open the Sample Code

To open and use this sample code "IND360_PRNT_APPLICATIONS_V2_00.ap14", Siemens TIA Portal version 14 SP1 or higher is required. All the necessary GSDML files will be installed automatically when the sample code is opened.

2.3. Use the Correct Project

Four projects are included in one sample code, each using a different IND360 application and SAI data format:

- "IND360 Fill Dose SAI2Block" uses S7-1200 series PLC with IND360 Fill Dose Application enabled and communicates in SAI 2-Block data format.
- "IND360 Fill Dose SAI8Block" uses S7-1200 series PLC with IND360 Fill Dose Application enabled and communicates in SAI 8-Block data format.
- "IND360 Tank Vessel SAI2Block" uses S7-1200 series PLC with IND360 Tank Vessel Application enabled and communicates in SAI 2-Block data format.
- "IND360 Tank Vessel SAI8Block" uses S7-1200 series PLC with IND360 Tank Vessel Application enabled and communicates in SAI 8-Block data format.
- "IND360 Dynamic SAI8Block" uses S7-1200 series PLC with IND360 Dynamic Weighing Application enabled and communicates in SAI 8-Block data format.

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

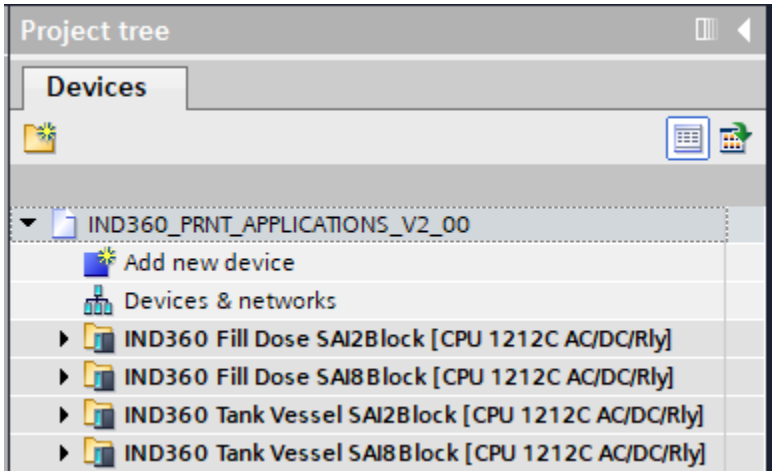

Figure 2-2: Five Projects in the Sample Code

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

Compile and download the project into the controller.

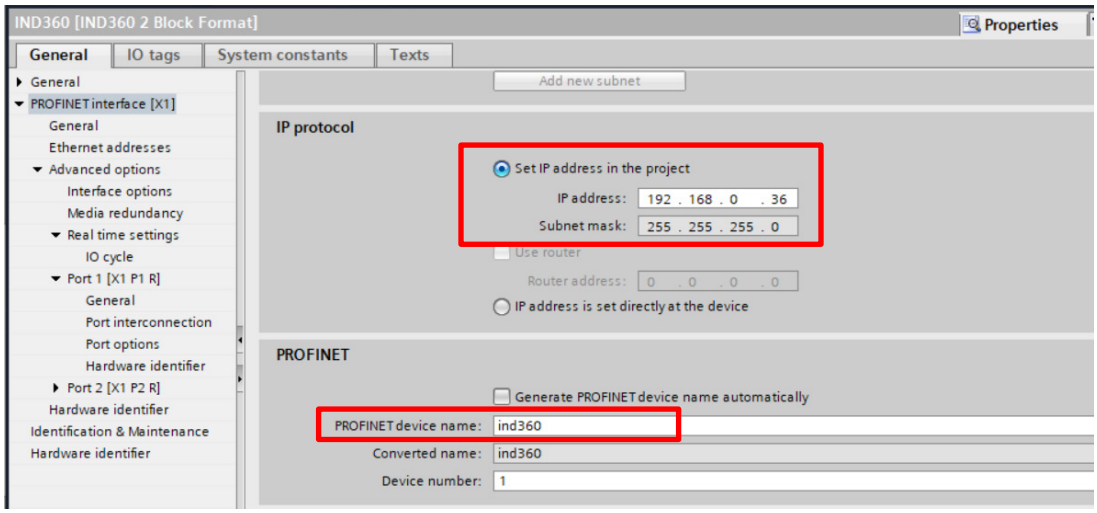

Figure 2-1: IND360 POWERCELL Device Properties – Ethernet Addresses

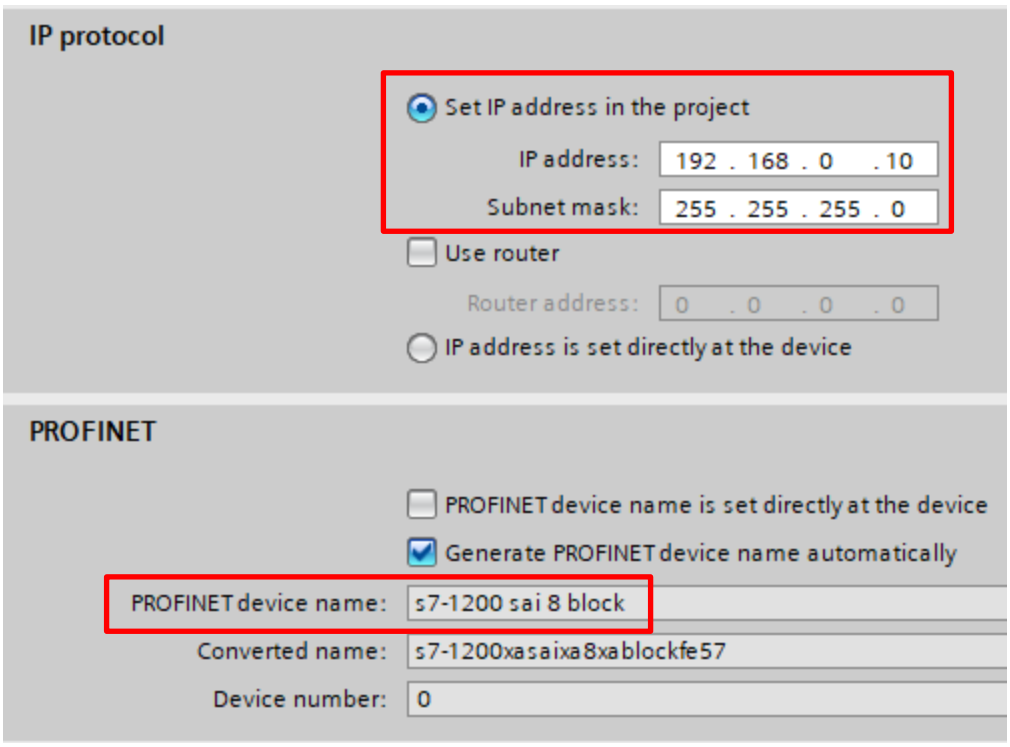

Figure 2-5: PLC Device Properties – Ethernet Addresses

Select the "MT_IND_Application" program and click the "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 **SAI Reference Guide for Transmitters and Terminals** and the **Standard Automation Interface User's Guide**, which are available on the IND360 download page.

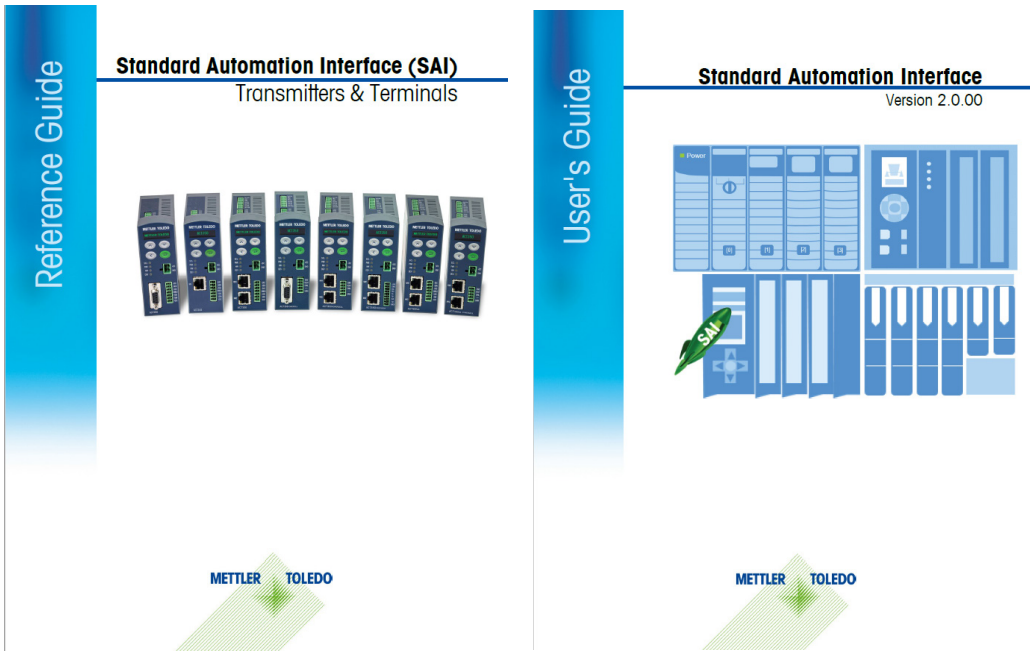

Figure 3-3-1 : SAI Reference Guide for Transmitters & Terminals and SAI User's Guide on the IND360 Download Page

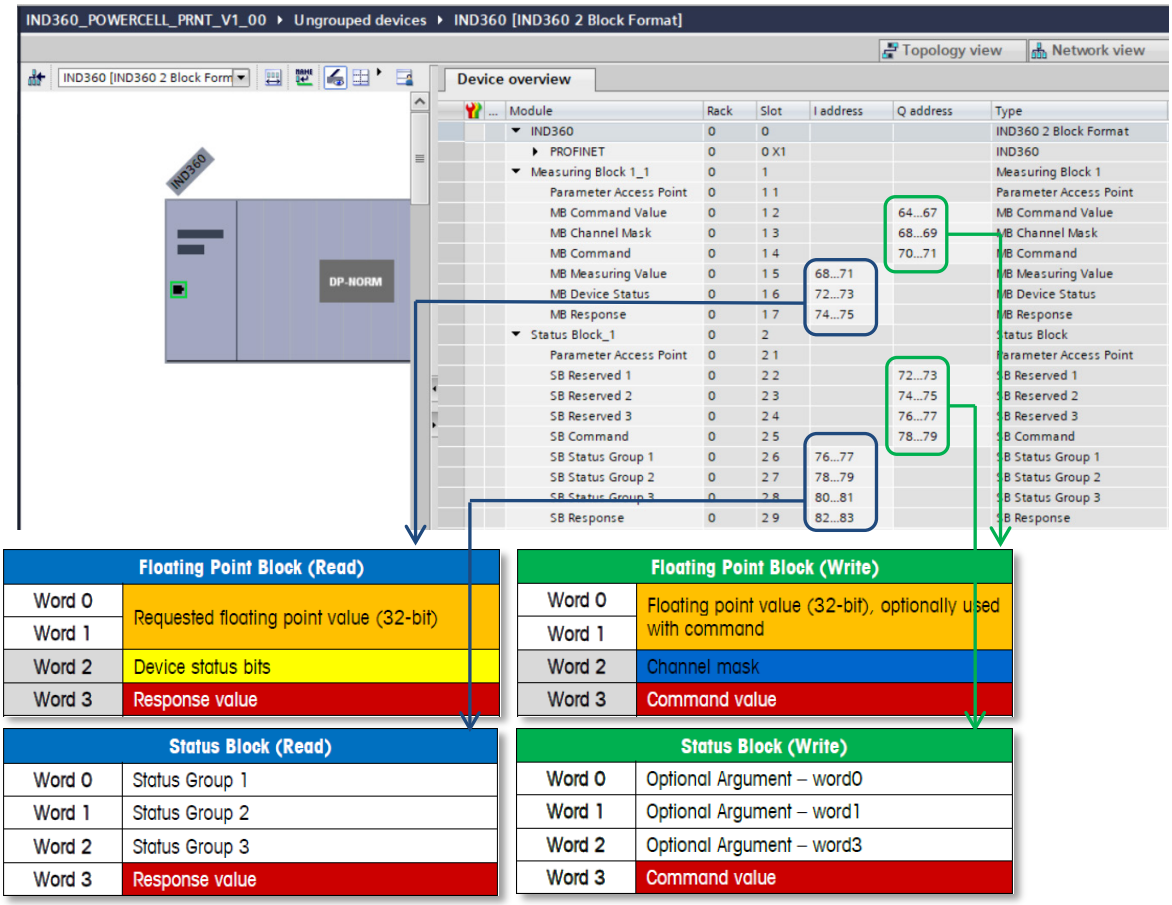

Figure 3-2 : 2-Block SAI Data Structure as Shown in the Device Overview

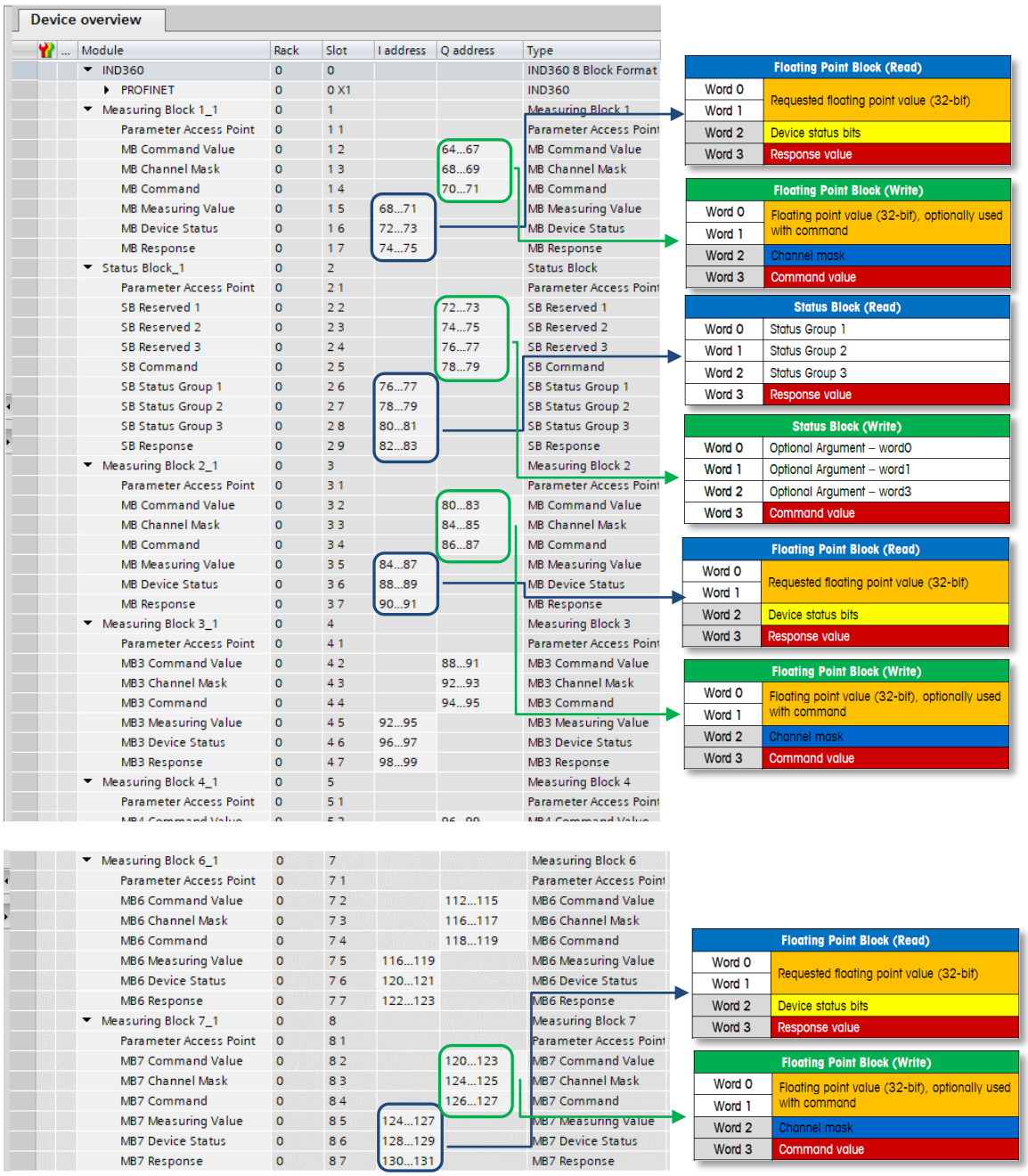

Figure 3-3 : 8-Block SAI Data Structure as Shown in the Device Overview

The I and Q addresses above will be used as input parameters in section 4, **Application Function Blocks**.

4 Application Function Blocks

Please refer to the **IND360 Application User's Guides** for complete information on the various weighing application operating principle. Download these User's Guides from www.mt.com/ind360-downloads.

Figure 4-4-1 : the IND360 Application User's Guide on the IND360 Download Page

4.1. Tank Vessel Application Control

This function block is used to read and write the upper and lower limits, receive alarms, and start and stop the application. In this sample code, this function block is assigned with IO addresses from the Measuring Block 1 (referred as Measuring Block 1_1 in the TIA Portal Device Overview). If PLC communication is set up as SAI 8 Block format, this function block's IO addresses can also be assigned to any other Measuring Block.

Before executing this function block, make sure the tank vessel application is enabled in the PAC Management setup menu.

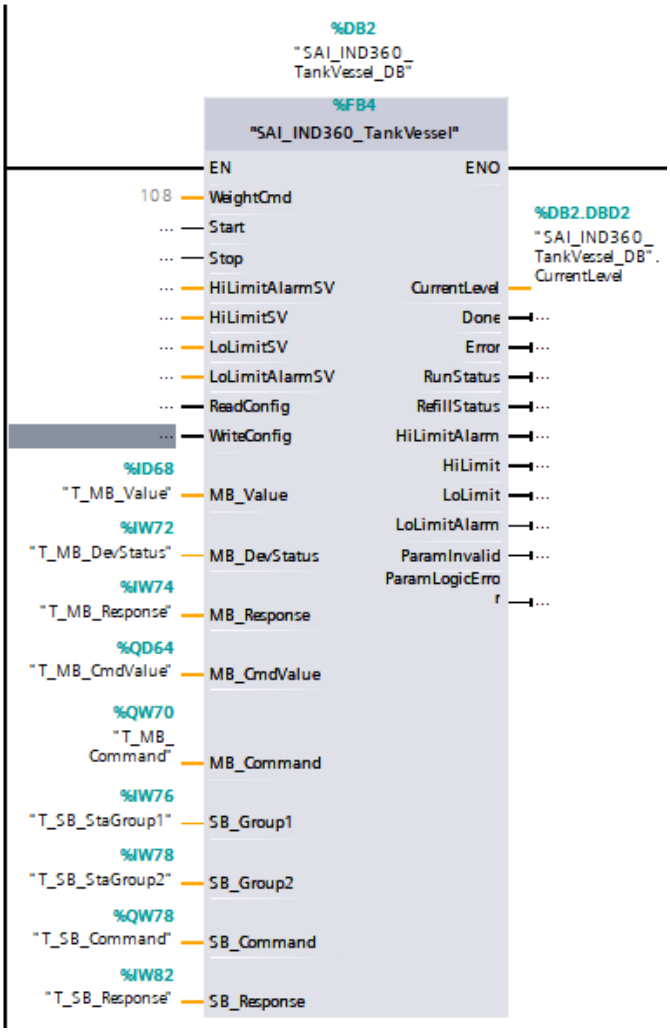

Figure 4-2: SAI_IND360_TankVessel Function Block

Table 4-1: SAI_IND360_TankVessel Function Block Parameters

Input Parameters	Data Type	Values	Description
WeightCmd	Word	107	Report material level in gross weight
		108 (default)	**Report material level in percentage**
		0,1	Report the scale default (gross), gross weight
		2	Report the scale tare weight value
		3	Report the scale net weight value
Start	Bool		Trigger this bit to start the automatic level monitoring and control by the IND360. Upon successful execution of this command, the input bit will be reset and the output DONE bit will be turned ON.
Stop	Bool		Trigger this bit to stop the automatic level monitoring and control by the IND360. Upon successful execution of this command, the input bit will be reset and the output DONE bit will be turned ON.
HiLimitAlarmSV	REAL	e.g. "90.0"	This is the highest extreme material level at which the IND360 triggers an alarm. There is a potential risk of over-filling (e.g. due to malfunction of refill pump). This setting, along with the other limit and alarm settings will be written to the IND360 when the WriteConfig bit is triggered. Similarly, triggering the ReadConfig bit will update all the limits and alarms to the PLC DataBlock.
HiLimit	REAL	e.g. "80.0"	When the material level exceeds this limit, the REFILL signal turns OFF, and remains OFF until the level falls below LoLimit again. This setting, along with the other limit and alarm settings will be written to the IND360 when the WriteConfig bit is triggered. Similarly, triggering the ReadConfig bit will update all the limits and alarms to the PLC DataBlock.
LoLimit	REAL	e.g. "50.0"	When the material level falls below this limit, the REFILL signal turns ON, and remains ON until the level exceeds the HiLimit. This setting, along with the other limit and alarm settings will be written to the IND360 when the WriteConfig bit is triggered. Similarly, triggering the ReadConfig bit will update all the limits and alarms to the PLC DataBlock.
LoLimitAlarmSV	REAL	e.g. "10.0"	This is the lowest extreme material level at which the material level is considered critically low. This setting, along with the other limit and alarm settings will be written to the IND360 when the WriteConfig bit is triggered. Similarly, triggering the ReadConfig bit will update all the limits and alarms to the PLC DataBlock.
MB_Value	Real		Refer to Device Overview, input address of MB Measuring Value.
MB_DevStatus	Word		Refer to Device Overview, input address of MB Device Status.
MB_Response	Word		Refer to Device Overview, input address of MB Response.

Input Parameters	Data Type	Values	Description
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.
SB_Group1	Word		Refer to Device Overview, input address of SB Status Group 1.
SB_Group2	Word		Refer to Device Overview, input address of SB Status Group 2.
SB_Command	Word		Refer to Device Overview, output address of SB Command.
SB_Response	Word		Refer to Device Overview, input address of SB Response.
Output Parameters	**Data Type**	**Values**	**Description**
CurrentLevel	Real		Real-time weight value. This can be the material level expressed in percent or gross weight, depending on the input WeightCmd.
Done	Bool	0	Start / Stop / ReadConfig / WriteConfig command in process, or failed.
		1	Start / Stop / ReadConfig / WriteConfig command successful.
Error	Bool	0	Start / Stop / ReadConfig / WriteConfig command in process, or success.
		1	Start / Stop / ReadConfig / WriteConfig command failed.
RunStatus	Bool	0	Tank / Vessel level monitoring and control application not started.
		1	Tank / Vessel level monitoring and control application running.
RefillStatus	Bool	0	Refill signal is OFF.
		1	Refill signal is ON.
HiLimitAlarm	Bool	0	Application alarm is OFF.
		1	Application alarm is ON. Check the Refill line.
HiLimit	Bool	0	Upper Limit is OFF.
		1	Upper Limit is ON.
LoLimit	Bool	0	Lower Limit is OFF.
		1	Lower Limit is ON.
LoLimitAlarm	Bool	0	Application alarm is OFF.
		1	Application alarm is ON. Check the Refill line.

4.2. Tank Vessel Alibi Record Access

This Function Block is used to execute ePrint to register the latest transaction into the alibi log. Also it provides user interface to read one alibi record at a time.

To read out a certain record number from the alibi memory:

1. Write in the 'RecordNumber' to specify which alibi transaction entry to be read.

2. Trigger the 'Set_TransactionNum' bit to set the alibi record number to be read out.

3. Trigger the 'Get_Alibi' bit to read out the alibi transaction record specified by "RecordNum". Alibi record is read out as an array of 44 bytes, written into "Alibi_Array" data block.

To read the latest entry in the alibi memory, execute step 3 without steps 1 and 2.

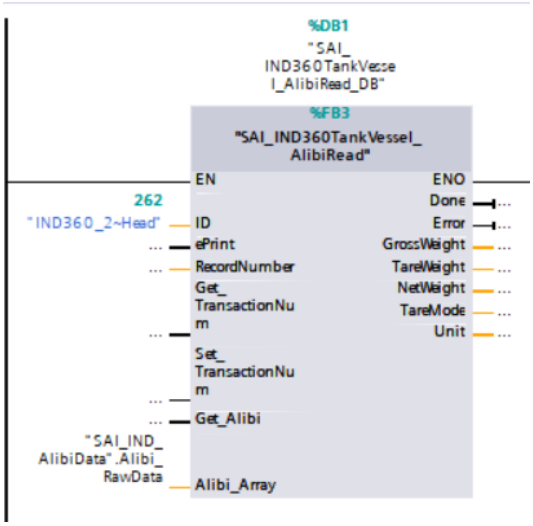

Figure 4-3: SAI_IND360TankVessel_AlibiRead Function Block

Table 4-2: SAI_IND360TankVessel_AlibiRead Function Block Parameters

Input Parameters	Data Type	Values	Description
ID	HW_IO	Example: "IND360~Head"	The ID parameter selects 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**.
ePrint	Bool	0 ,1	Trigger this bit to register the latest weighing transaction into the alibi log memory. The transaction number will be incremented.
RecordNumber	DINT	"3"	The alibi transaction record number which is to be read out from the weighing indicator. Specify the record number first before setting and reading an alibi record.

Input Parameters	Data Type	Values	Description
Get_TransactionNum	Bool	0 ,1	Trigger this bit to get the number of the previously read alibi record. The alibi record number will be updated in the "RecordNum" field. This bit will be toggled off by the FB after its execution status has been reported (DONE / ERROR).
Set_TransactionNum	Bool	0 ,1	Trigger this bit to set the alibi record number which is to be read out from the weighing indicator. Ensure that the "RecordNum" has been specified before this step. This bit will be toggled off by the FB after its execution status has been reported (DONE / ERROR).
Get_Alibi	Bool	0 ,1	Trigger this bit to read out the alibi transaction record specified by "RecordNum". Alibi record is read out as an array of 44 bytes, written into "Alibi_Array" data block. This bit will be toggled off by the FB after reporting its execution status (DONE / ERROR).

Alibi_Array	Array of Bytes [0..55]		The array of 44 bytes used to store the raw data of the alibi record being read from the weighing indicator. This array is structured as follows:
			Struct(44 Bytes)
			{
			Byte[20]: Date & time; / /ASCII String =>Byte0-19
			Long: Transaction Number =>Byte20-23
			Float32: Gross Weight =>Byte24-27
			Float32: Net Weight =>Byte28-31
			Float32: Tare Weight =>Byte32-35
			Byte: Tare Mode =>Byte36
			Byte: Unit =>Byte37
			Byte[6]: Not Used =>Byte38-43
			}

Output Parameters	**Data Type**	**Values**	**Description**
Done	Bool	0	Get / Set_TransactionNum or Get_Alibi command is not completed
		1	Get / Set_TransactionNum or Get_Alibi command is completed
Error	Bool	0	Get / Set_TransactionNum or Get_Alibi command gives no error
		1	Get / Set_TransactionNum or Get_Alibi command is completed, with error
GrossWeight	REAL		This is the gross weight field extracted from the latest alibi record read. This variable is in double precision floating point format
TareWeight	REAL		This is the tare weight field extracted from the latest alibi record read. This variable is in double precision floating point format

Input Parameters	Data Type	Values	Description
NetWeight	REAL		This is the net weight field extracted from the latest alibi record read. This variable is in double precision floating point format
Unit	SINT		The weight unit used according to the weighing indicator's setting. For more details, refer to the Secondary Scale Status unit bits section in the **SAI Reference Manual**. A combination of 4 bits represents the weight unit [b4 /b3 /b2 /b1]. For example, [0 /0 /0 /0] is unit 'g'; [0 /0 /0 /1] is unit 'kg'; [0 /0 /1 /0] is unit 'lb'.

4.3. Fill Dose Application Control

This is the function block used to read /write the fill /dose application parameters, fill /dose target settings, receive status / alarms, and start /stop the fill /dose application.

In this sample code (both projects: SAI 2-Block and SAI 8-Block), this function block occupies the IO addresses from the Measuring Block 1 (referred as Measuring Block 1_1 in the TIA Portal Device Overview). If the PLC communication is set up as SAI 8 Block format, these function block IO addresses can also be assigned to any other Measuring Block.

Before executing this function block, make sure the fill /dose application is enabled under the IND360 PAC Management in the device setup menu.

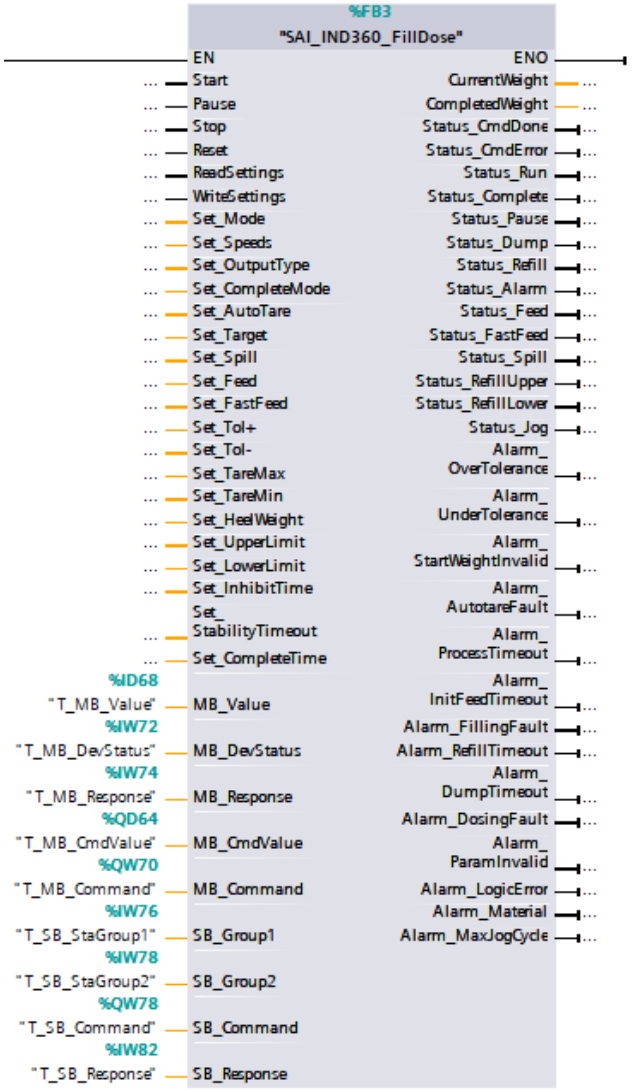

Figure 4-2: SAI_IND360_FillDose Function Block

Table 4-1: SAI_IND360_FillDose Function Block Parameters

Input Parameters	Data Type	Values	Description
Start	Bool	0, 1	Trigger this bit to start one cycle of fill / dose / fill-dump / refill-dump. Upon successful execution of this command, the input bit will be reset.
Pause	Bool	0, 1	Trigger this bit to pause the current cycle.
Stop	Bool	0, 1	Trigger this bit to stop the current cycle.
Reset	Bool	0, 1	Trigger this bit to transition to Idle state (safe state, all I /O turned off, captured tare weight reset and configuration on HMI and web interface is possible).
ReadSettings	Bool	0, 1	Trigger this bit to read all the settings from the terminal.

WriteSettings	Bool	0, 1	Trigger this bit to write all the changed settings to the terminal.
Set_Mode	Real	0, 1, 2, 3	Operation mode: 0 – Fill Dump; 1 – Refill Dump; 2 – Fill; 3 – Dose
Set_Speeds	Real	0, 1, 2	Filling / Dosing number of speeds: 0 – One speed; 1 – Two speeds
Set_OutputType	Real	0, 1	Output type: 0 – Concurrent; 1 – Independent
Set_CompleteMode	Real	0, 1	Fill / Dump mode only. Determines whether the **complete** signal is triggered by the heel weight or by a set time. 0 – Weight Mode; 1 – Time Mode
Set_AutoTare	Real	0, 1	Fill or Fill / Dump mode: Automatic Tare occurs once process begins if the current weight is between the container tare max and min values, else a Smart5 yellow alarm will be raised.
Set_Target	Real	0…scale capacity	Set target weight for a weigh-in cycle. Note that the target should be greater than the sum of Spill, Feed and Fast Feed.
Set_Spill	Real	0…scale capacity	Spill sets the weight of material that will be added to (in a weigh-in cycle) or removed (in a weigh-out cycle) from the scale after all feeds are turned off.
Set_Feed	Real	0…scale capacity	Feed sets the weight of material that will be fed in a slow speed rate.
Set_FastFeed	Real	0…scale capacity	Fast Feed sets the weight of material that will be fed in a fast speed rate.
Set_Tol+	Real	0…scale capacity	+ Tolerance sets the maximum permissible amount above the target weight.
Set_Tol-	Real	0…scale capacity	- Tolerance sets the minimum permissible amount below the target weight.
Input Parameters	**Data Type**	**Values**	**Description**
Set_TareMax	Real	0…scale capacity	Container Tare Max sets the maximum weight of an empty container. Used when auto tare function is enabled. The empty container needs to be in this range for the fill operation to be started.
Set_TareMin	Real	0…scale capacity	Container Tare Min sets the minimum weight of an empty container. Used when auto tare function is enabled. The empty container needs to be in this range for the fill operation to be started.
Set_HeelWeight	Real	0…scale capacity	Indicates the amount of material that might remain in the weigh vessel when it is considered empty – for example, material stuck to the sides of a hopper.

Set_UpperLimit	Real	0…scale capacity	Defines the upper limit for the automatic refill to stop. The refill signal will stay high until the upper limit is reached.
Set_LowerLimit	Real	0…scale capacity	If a Dose is started and the material in the tank weighs less than Lower Limit, the refill signal will be triggered and stay high until the weight of material in the tank reaches the Upper Limit.
Set_InhibitTime	Real	0~9.99s	Inhibit Time sets the amount of time that the comparator will have to wait before it is allowed to compare. This is used to ignore peaks in the weight signal when switching between feed speeds.
Set_StabilityTimeout	Real	0~9.99s	Optional parameter. Stability Timeout is the maximum length of time the program will wait for a stable weight reading after material dispensing has stopped. A value of 0 disables the timeout and will allow the IND360 to wait indefinitely until stability.
Set_CompleteTime	Real	0~9.99s	Fill / Dump mode only when complete mode is set to "Time mode". Complete Time sets a fixed duration for the dump process to be executed.
MB_Value	Real	e.g. %ID68	Refer to Device Overview, input address of MB Measuring Value.
MB_DevStatus	Word	e.g. %IW72	Refer to Device Overview, input address of MB Device Status.
MB_Response	Word	e.g. %IW74	Refer to Device Overview, input address of MB Response.
MB_CmdValue	Real	e.g. %QD64	Refer to Device Overview, output address of MB Command Value.
MB_Command	Word	e.g. %QW70	Refer to Device Overview, output address of MB Command.
Input Parameters	**Data Type**	**Values**	**Description**
SB_Group1	Word	e.g. %IW76	Refer to Device Overview, input address of SB Status Group 1.
SB_Group2	Word	e.g. %IW78	Refer to Device Overview, input address of SB Status Group 2.
SB_Command	Word	e.g. %QW78	Refer to Device Overview, output address of SB Command.
SB_Response	Word	e.g. %IW82	Refer to Device Overview, input address of SB Response.
Output Parameters	**Data Type**	**Values**	**Description**
CurrentWeight	Real	0… capacity	Real-time weight reading of container on scale.
CompleteWeight	Real	0… capacity	The amount of material filled or dispensed in the last cycle.
Status_CmdDone	Bool	0, 1	Signals that the PLC configuration or control commands to the indicator are being executed successfully.

Status_CmdError	Bool	0, 1	Signals that the PLC configuration or control commands to the indicator are erroneous.
Status_Run	Bool	0, 1	The filling /dosing cycle has been started.
Status_Complete	Bool	0, 1	The cycle has been completed.
Status_Pause	Bool	0, 1	The cycle is paused.
Status_Dump	Bool	0, 1	Dump sequence is running.
Status_Refill	Bool	0, 1	Refill sequence is running.
Status_Alarm	Bool	0, 1	An alarm is active.
Status_Feed	Bool	0, 1	Current speed is feed.
Status_FastFeed	Bool	0, 1	Current speed is fast feed.
Status_Spill	Bool	0, 1	Spill sequence is running.
Status_RefillUpper	Bool	0, 1	Refill hits the upper limit.
Status_RefillLower	Bool	0, 1	Lower refill limit reached.
Status_Jog	Bool	0, 1	Jog is being executed.
Alarm_Over Tolerance	Bool	0, 1	The final weight is over the predefined positive tolerance.
Alarm_Under Tolerance	Bool	0, 1	The final weight is below the predefined negative tolerance.
Alarm_StartWeightInvalid	Bool	0, 1	Unable to start due to incorrect starting weight (e.g. start weight above target).
Alarm_AutotareFault	Bool	0, 1	Auto tare failed due to container out of tare range.
Alarm_ProcessTimeout	Bool	0, 1	Maximum allowed time (configurable) for fill /dose cycle exceeded.
Alarm_InitFeedTimeout	Bool	0, 1	Initial feed timeout: weight value is not changing for predefined time after process is started.
Alarm_FillingFault	Bool	0, 1	Fault happened during filling.
Output Parameters	**Data Type**	**Values**	**Description**
Alarm_RefillTimeout	Bool	0, 1	The Refill Timeout determines the maximum acceptable amount of time for the refill to complete.
Alarm_DumpTimeout	Bool	0, 1	The Dump Timeout determines the maximum acceptable amount of time for the dump to complete.
Alarm_DosingFault	Bool	0, 1	Fault happened during dosing.
Alarm_Param Invalid	Bool	0, 1	Invalid parameter set-up.
Alarm_LogicError	Bool	0, 1	Logic error.
Alarm_Material	Bool	0, 1	Insufficient material.
Alarm_MaxJogCycle	Bool	0, 1	There is a maximum number of jog cycles for automatic jog. To protect the equipment, this alert is triggered when the maximum number of cycles has been exceeded.

4.4. Fill Dose Alibi Record Access

This Function Block is used to execute ePrint to register the latest transaction into the alibi log. It also provides a user interface to read one alibi record at a time.

To read out a certain transaction from the alibi memory:

1. Write in the 'TransactionNumber' to specify which alibi transaction entry to be read.

2. Trigger the 'Set_TransactionNum' bit to set the alibi record number to be read out.

3. Trigger the 'Get_Alibi' bit to read out the alibi transaction record specified by "RecordNum". Alibi record is read out as an array of 76 bytes, written into "Alibi_Array" data block.

To read the latest entry in the alibi memory, execute step 3 without step 1 and 2.

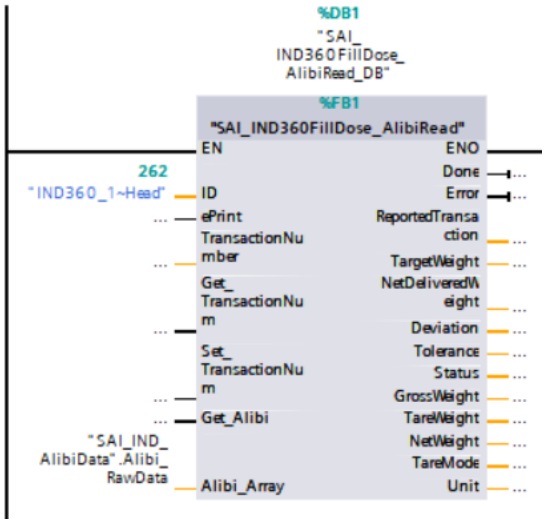

Figure 4-3: SAI_IND360FillDose_AlibiRead Function Block

Table 4-2: SAI_IND360FillDose_AlibiRead Function Block Parameters

Input Parameters	Data Type	Values	Description
ID	HW_IO	Example: "IND360~Head"	The ID parameter is used 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**.
ePrint	Bool	0 ,1	Trigger this bit to register the latest weighing transaction into the alibi log memory. The transaction number will be incremented.
TransactionNumber	DINT	″3″	The alibi transaction record number which is to be read out from the weighing indicator. Specify the record number first before setting and reading alibi record.

Get_TransactionNum	Bool	0 ,1	Trigger this bit to get the previously read alibi record number. The alibi record number will be updated at the "RecordNum" field. This bit will be toggled off by the FB after reporting its execution status (DONE / ERROR).
Set_TransactionNum	Bool	0 ,1	Trigger this bit to set the alibi record number which is to be read out from the weighing indicator. Ensure that the "RecordNum" has been specify before this step. This bit will be toggled off by the FB after reporting its execution status (DONE / ERROR).
Get_Alibi	Bool	0 ,1	Trigger this bit to read out the alibi transaction record specified by "RecordNum". Alibi record is read out as an array of 76 bytes, written into "Alibi_Array" data block. This bit will be toggled off by the FB after reporting its execution status (DONE / ERROR).

Alibi_Array	Array of Byte [0..55]		The array of 76 bytes to store the raw data of alibi record being read from the weighing indicator. This array is structured as follows:
			struct(76 Bytes)
			{
			Byte[20] Start Date & time; / /ASCII String =>Byte0-19
			Byte[20] Finish Date & time; / /ASCII String =>Byte20-39
			Long Transaction Number =>Byte40-43
			Float32 Target Weight =>Byte44-47
			Float32 Net Delivered Weight =>Byte48-51
			Float32 Deviation =>Byte52-55
			Byte Tolerance / /1: Under, 2: In, 3: Over =>Byte56
			Byte Status / /0: Unstable, 1: Stable, 2: Stopped =>Byte57
			Byte[2] Not Used =>Byte58-59
			Float32 Gross Weight =>Byte60-63
			Float32 Net Weight =>Byte64-67
			Float32 Tare Weight =>Byte68-71
			Byte Tare Type; / / Tare type:0,tare;1;preset tare =>Byte72
			Byte Unit =>Byte73
			Byte[2] Not Used =>Byte74-75
			}

Output Parameters	Data Type	Values	Description
Done	Bool	0	Get / Set_TransactionNum or Get_Alibi command is not completed.
		1	Get / Set_TransactionNum or Get_Alibi command is completed.
Error	Bool	0	Get / Set_TransactionNum or Get_Alibi command gives no error.
		1	Get / Set_TransactionNum or Get_Alibi command is completed, with error.
ReportedTransaction	LINT		Transaction number which is read.
TargetWeight	REAL		Filling / dosing target weight.
NetDeliveredWeight	REAL		Final filled / dosed net weight, or the result of current fill / dose cycle
Deviation	REAL		The difference in weight between the target weight and the net delivered weight
Tolerance	SINT		The filling / dosing weight tolerance: 1: Under, 2: In, 3: Over

Status	SINT		Process status: 0: Unstable, 1: Stable, 2: Stopped
GrossWeight	REAL		The gross weight field extracted from the latest alibi record read. This variable is in double precision floating point format.
TareWeight	REAL		The tare weight field extracted from the latest alibi record read. This variable is in double precision floating point format.
NetWeight	REAL		This is the net weight field extracted from the latest alibi record read. This variable is in double precision floating point format.
TareType	SINT		0: tare;1: preset tare
Unit	SINT		The weight unit used according to the weighing indicator's setting. For more details, refer to the Secondary Scale Status unit bits, in the **SAI Reference Manual**. A combination of 4 bits represents the weight unit [b4 /b3 /b2 /b1]. For example, [0 /0 /0 /0] is unit 'g'; [0 /0 /0 /1] is unit 'kg'; [0 /0 /1 /0] is unit 'lb'.

4.5. Dynamic Weighing Application Control

This is the function block used to read the IND360dynamic application status / alarms and start /stop the application.

In the sample code, this function block is occupying 8 blocks of IO addresses (32 words) starting from Measuring Block 1 (referred as Measuring Block 1_1 in the TIA Portal Device Overview). In order to use this function block, the IND360 PLC data has be configured as "SAI 8 Block" format.

Before executing this function block, make sure the dynamic weighing application is enabled under the IND360 PAC Management in the device setup menu.

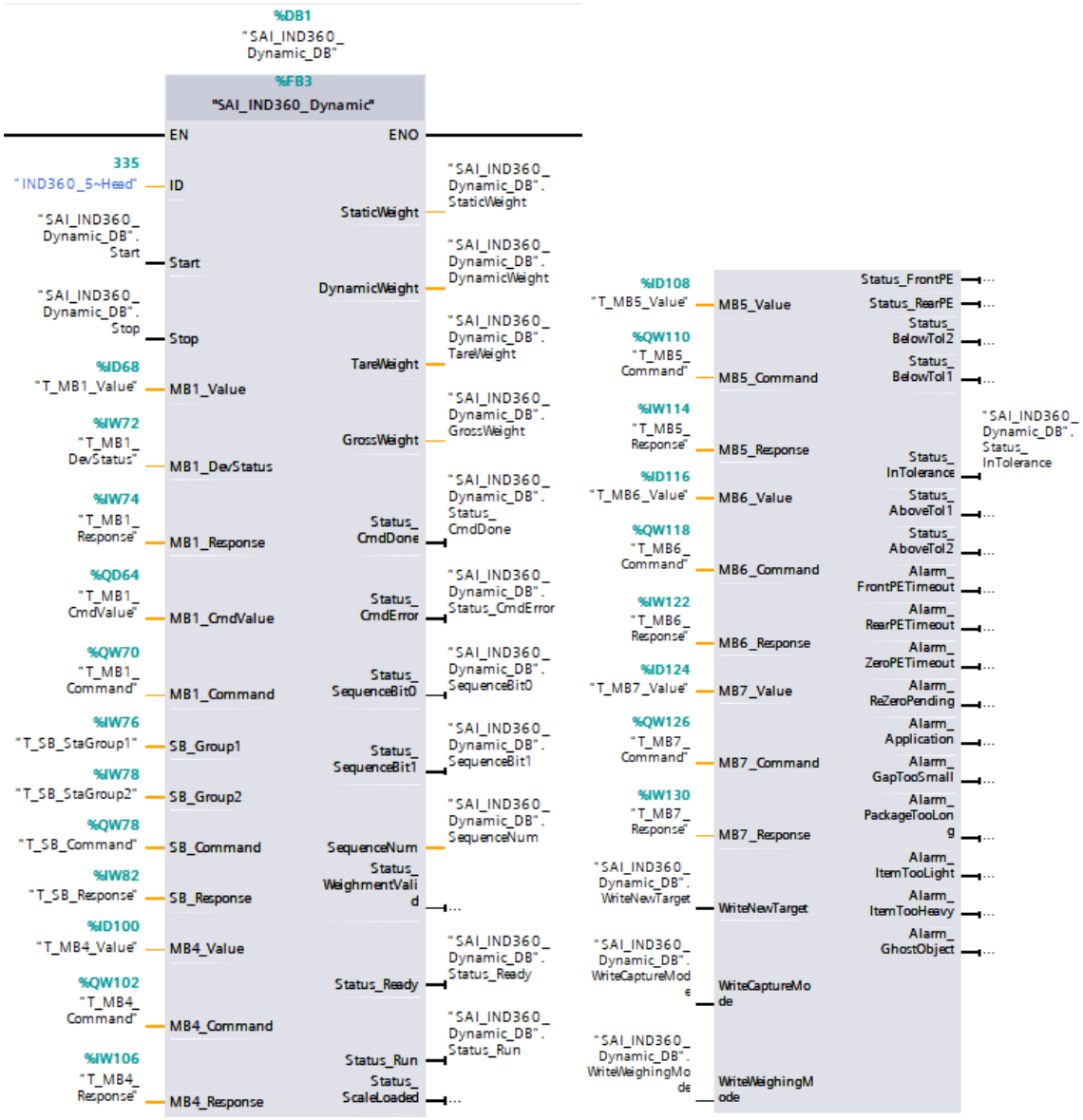

Figure 4-4: SAI_IND360_Dynamic Function Block

Table 4-3: SAI_IND360_Dynamic Function Block Parameters

Input Parameters	Data Type	Values	Description
Start	Bool	0, 1	Trigger this bit to start the dynamic weighing application. Upon successful execution of this command, the input bit will be reset.
Stop	Bool	0, 1	The dynamic weighing application will continue to run until this stop bit is triggered. Once stopping is complete, this bit will be reset.
MB1_Value	Real	e.g. %ID68	Refer to Device Overview, input address of MB Measuring Value.
MB1_DevStatus	Word	e.g. %IW72	Refer to Device Overview, input address of MB Device Status.
Input Parameters	**Data Type**	**Values**	**Description**
MB1_Response	Word	e.g. %IW74	Refer to Device Overview, input address of MB Response.
MB1_CmdValue	Real	e.g. %QD64	Refer to Device Overview, output address of MB Command Value.
MB1_Command	Word	e.g. %QW70	Refer to Device Overview, output address of MB Command.
SB_Group1	Word	e.g. %IW76	Refer to Device Overview, input address of SB Status Group 1.
SB_Group2	Word	e.g. %IW78	Refer to Device Overview, input address of SB Status Group 2.
SB_Command	Word	e.g. %QW78	Refer to Device Overview, output address of SB Command.
SB_Response	Word	e.g. %IW82	Refer to Device Overview, input address of SB Response.
MB4_Value … MB7_Value	Real	e.g. %ID100	Refer to Device Overview, input address of MB Measuring Value.
MB4_DevStatus … MB7_DevStatus	Word	e.g. %IW112	Refer to Device Overview, input address of MB Device Status.
MB4_Response … MB7_Response	Word	e.g. %IW114	Refer to Device Overview, input address of MB Response.
MB4_CmdValue … MB7_CmdValue	Real	e.g. %QD96	Refer to Device Overview, output address of MB Command Value.
MB4_Command … MB7_Command	Word	e.g. %QW102	Refer to Device Overview, output address of MB Command.
WriteNewTarget	Bool	0, 1	Trigger this bit to write the Classifications command (Operating mode, Target weight, Tolerances). Refer to SAI_IND360DYN_Target_DB "Classification" Struct to change relevant parameters. Use this function to mainly change target weight, tolerances, and type.
WriteCaptureMode	Bool	0, 1	Trigger this bit to write the Weight Capture command (Trigger Source, Photoeye setup, Photoeye position, Weighing mode, Trigger point). Refer to SAI_IND360DYN_Target_DB

			"Weight Capture" Struct to change relevant parameters. Use this function to mainly change trigger source, such as via Automation interface, Digital input or Weight trigger.
WriteWeighingMode	Bool	0, 1	Trigger this bit to write the General settings command (Power up delay, Mode, Unit). SAI_IND360DYN_Target_DB "Weighing Mode" Struct to change relevant parameters. Use this function to mainly change mode of weighing, such as In-Motion or Static, Check-weighing or Catch-weighing.
Output Parameters	**Data Type**	**Values**	**Description**
StaticWeight	Real	0… capacity	The static net weight of package on the scale.
DynamicWeight	Real	0… capacity	The captured package dynamic weight when dynamic weighing operation is running.
TareWeight	Bool	0… capacity	The tare weight after a tare operation.
GrossWeight	Bool	0… capacity	The total weight on the scale.
Status_CmdDone	Bool	0, 1	Signals that the PLC configuration or control commands to the indicator are being executed successfully.
Status_CmdError	Bool	0, 1	Signals that the PLC configuration or control commands to the indicator are erroneous.
Status_SequenceBit0	Bool	0, 1	Sequence bit 0, together with sequence bit 1 will increment from 00, 01, 10 and 11 every time after a new package dynamic weight is captured. SequenceNum increments in the pattern of 0,1,2,3,0,1… as packages being dynamically weighed.
Status_SequenceBit1	Bool	0, 1	
SequenceNum	Bool	0, 1	
Status_WeighmentValid	Bool	0, 1	The captured dynamic weight is good to use.
Status_Ready	Bool	0, 1	Ready to run dynamic weighing operation.
Output Parameters	**Data Type**	**Values**	**Description**
Status_Run	Bool	0, 1	Dynamic weighing mode is running.
Status_ScaleLoaded	Bool	0, 1	A package is present on the conveyor.
Status_FrontPE	Bool	0, 1	Front photo eye has been triggered.
Status_RearPE	Bool	0, 1	Rear photo eye has been triggered.
Status_BelowTol2	Bool	0, 1	The captured weight is below tolerance 2.
Status_BelowTol1	Bool	0, 1	The captured weight is below tolerance 1; above tolerance 2.
Status_InTolerance	Bool	0, 1	The captured weight is within predefined tolerances.

Status_AboveTol1	Bool	0, 1	The captured weight is above tolerance 1; below tolerance 2.
Status_AboveTol2	Bool	0, 1	The captured weight is above tolerance 2.
Alarm_FrontPETimeout	Bool	0, 1	Front photo eye is blocked for a time duration longer than the predefined photoeye timeout setting.
Alarm_RearPETimeout	Bool	0, 1	Rear photo eye is blocked for a time duration longer than the predefined photoeye timeout setting.
Alarm_ZeroPETimeout	Bool	0, 1	Re-zero photo eye is blocked for a time duration longer than 10 seconds.
Alarm_ReZeroPending	Bool	0, 1	Re-zeroing is overdue and needs to be executed. On the display this is indicated with a star.
Alarm_Application	Bool	0, 1	Alarm is presence preventing the application to execute, see Smart5 alarm details.
Alarm_GapTooSmall	Bool	0, 1	Only available in two photo eyes mode: the gap between two consecutive packages is too small, risk of insufficient measuring time.
Alarm_PackageTooLong	Bool	0, 1	Package is too long. Higher measurement uncertainty due to insufficient measurement time.
Alarm_ItemTooLight	Bool	0, 1	The package is too light, causing higher measurement uncertainty.
Alarm_ItemTooHeavy	Bool	0, 1	The package is too heavy, causing higher measurement uncertainty.
Alarm_GhostObject	Bool	0, 1	Only available in two photo eyes mode: package triggered the rear photo eye, however front photo eye was not triggered.

4.6. Dynamic Weighing Alibi Record Access

This Function Block is used to execute an ePrint to register the latest transaction in the alibi log. It also provides a user interface to read one alibi record at a time.

To read out a certain transaction from the alibi memory:

1. Write in the **TransactionNumber** to specify which alibi transaction entry to be read.
2. Trigger the **Set_TransactionNum** bit to set the alibi record number to be read out.
3. Trigger the **Get_Alibi** bit to read out the alibi transaction record specified by **RecordNum**. The alibi record is read out as an array of 24 bytes, written into **Alibi_Array** data block.

To read the latest entry in the alibi memory, execute step 3 without steps 1 and 2.

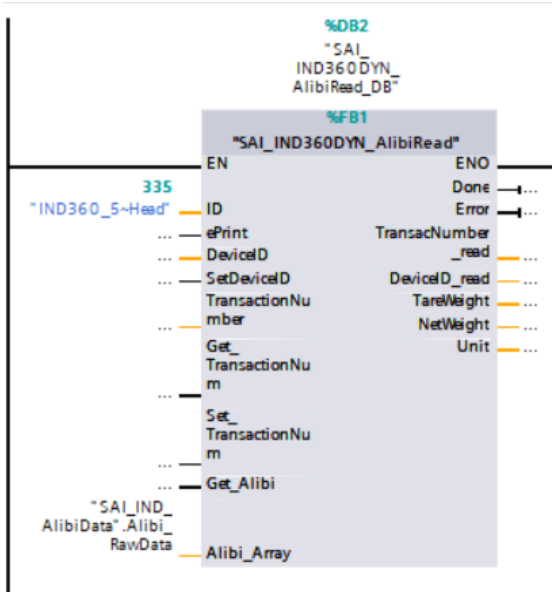

Figure 4-5: SAI_IND360DYN_AlibiRead Function Block

Table 4-4: SAI_IND360DYN_AlibiRead Function Block Parameters

Input Parameters	Data Type	Values	Description
ID	HW_IO	Example: "IND360~Head"	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**.
ePrint	Bool	0 ,1	Trigger this bit to register the latest weighing transaction into the alibi log memory. Transaction number will be incremented.
DeviceID	LINT	″5″	The dynamic checkweighing machine can be assigned a unique device ID.
SetDeviceID	Bool	0, 1	Write the device ID into the IND360 terminal.
TransactionNumber	LINT	″3″	The alibi transaction record number which is to be read out from the weighing indicator. Specify the record number first before setting and reading alibi record.
Get_TransactionNum	Bool	0 ,1	Trigger this bit to get the previously read alibi record number. The alibi record number will be updated at the **RecordNum** field. This bit will be toggled off by the FB after reporting its execution status (DONE / ERROR).

Input Parameters	Data Type	Values	Description
Set_TransactionNum	Bool	0 ,1	Trigger this bit to set the alibi record number which is to be read out from the weighing indicator. Ensure that the **RecordNum** has been specify before this step. This bit will be toggled off by the FB after reporting its execution status (DONE / ERROR).
Get_Alibi	Bool	0 ,1	Trigger this bit to read out the alibi transaction record specified by **RecordNum**. Alibi record is read out as an array of 24 bytes, written into **Alibi_Array** data block. This bit will be toggled off by the FB after reporting its execution status (DONE / ERROR).

Alibi_Array	Array of Byte [0..24]		The array of 24 bytes to store the raw data of alibi record being read from the weighing indicator. This array is structured as follows:
			struct(24 Bytes)
			{
			Long Transaction Number =>Byte0-3
			Long Date & Time; / /UTC timestamp =>Byte4-7
			Byte Device ID =>Byte8
			Byte[3] Not Used =>Byte9-11
			Float32 Rounded Net Weight =>Byte12-15
			Float32 Rounded Tare Weight =>Byte16-19
			Byte Unit Type =>Byte20
			Byte Status =>Byte21
			Byte[2] Not Used =>Byte22-23
			}

Done	Bool	0	Get / Set_TransactionNum or Get_Alibi command is not completed.
		1	Get / Set_TransactionNum or Get_Alibi command is completed.
Error	Bool	0	Get / Set_TransactionNum or Get_Alibi command gives no error.
		1	Get / Set_TransactionNum or Get_Alibi command is completed, with error.
TransactionNumber_ Read	LINT		The transaction number being read out.
DeviceID_Read	LINT		The machine device ID being read out.
TareWeight	REAL		This is the tare weight field extracted from the latest alibi record read. This variable is in double precision floating point format.
NetWeight	REAL		This is the net weight field extracted from the latest alibi record read. This variable is in double precision floating point format.

Input Parameters	Data Type	Values	Description
Unit	SINT		The weight unit used according to the weighing indicator's setting. For more details, refer to the Secondary Scale Status unit section in the **SAI Reference Manual**. A combination of 4 bits represents the weight unit [b4 /b3 /b2 /b1]. For example, [0 /0 /0 /0] is unit 'g'; [0 /0 /0 /1] is unit 'kg'; [0 /0 /1 /0] is unit 'lb'.

4.7. Cyclic Weight Data Processing

This function block is available in the "S7-1200 SAI 8 Block" project. It occupies Measuring Block 2 (referred to as Measuring Block 2_1 in the **TIA Portal Device Overview**) I /O addresses. Its I /O addresses can also be assigned to other Measuring Blocks such as 3, 4 … 7.

This function block reads 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 bits 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 immediately after any scale command. The type of weight data (gross, net, or tare) reported depends on the setting for WeightCmd. By default, the WeightCmd is decimal "3" and the function block will return a net weight value immediately 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.

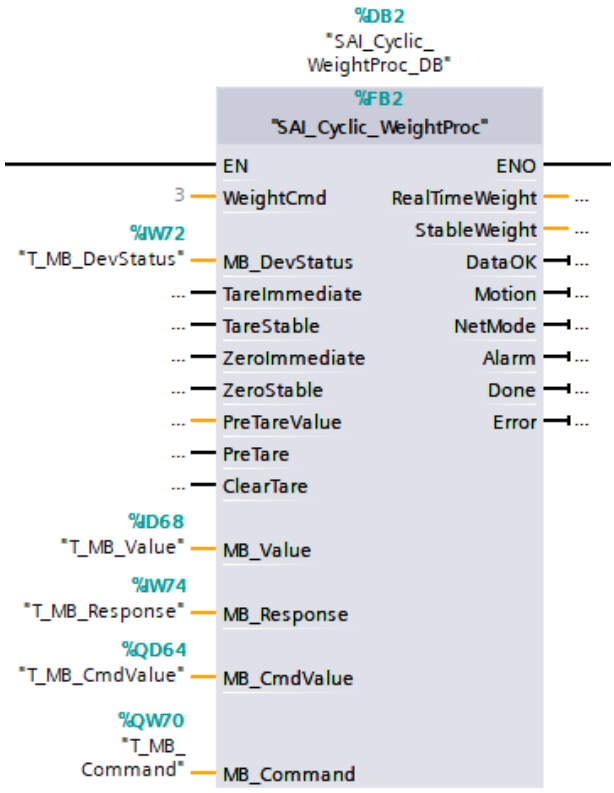

Figure 4-6: SAI_Cyclic_WeightProc Function Block

Table 4-5: SAI_Cyclic_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 an immediate tare command. This tare command does not 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 (+-1d within 0.3 second) for a predefined timeout range (3 seconds by default), failing which, the command will return an error. Upon completion of this command, the input bit will be reset.

Input Parameters	Data Type	Values	Description
ZeroImmediate	Bool		Trigger this bit to perform an immediate zero command. The zero command can only be executed when the weight value is within the zero range (+-2% by default). 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 (+-1d within 0.3 second) for a predefined timeout range (3 seconds by default). 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 Section 3, **Device Overview**, input address of MB Response.
MB_CmdValue	Real		Refer to Section 3, **Device Overview**, output address of MB Command Value.
MB_Command	Word		Refer to Section 3, **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

Output Parameters	Data Type	Values	Description
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 indicates 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 has succeeded
		1	Zero, tare or clear tare command has not completed due to error.

5 Sample Code

5.1. Hardware Configurations

4. Under **Devices & networks -> Network view**, add (or drag over) an IND360 2 Block Format.

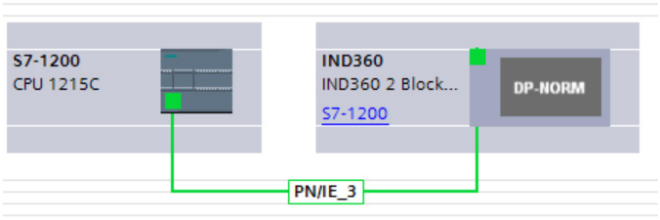

Figure 5-1: Add a PROFINET Device in the Network View

5. Assign the independent PROFINET device name and IP address for the added device.

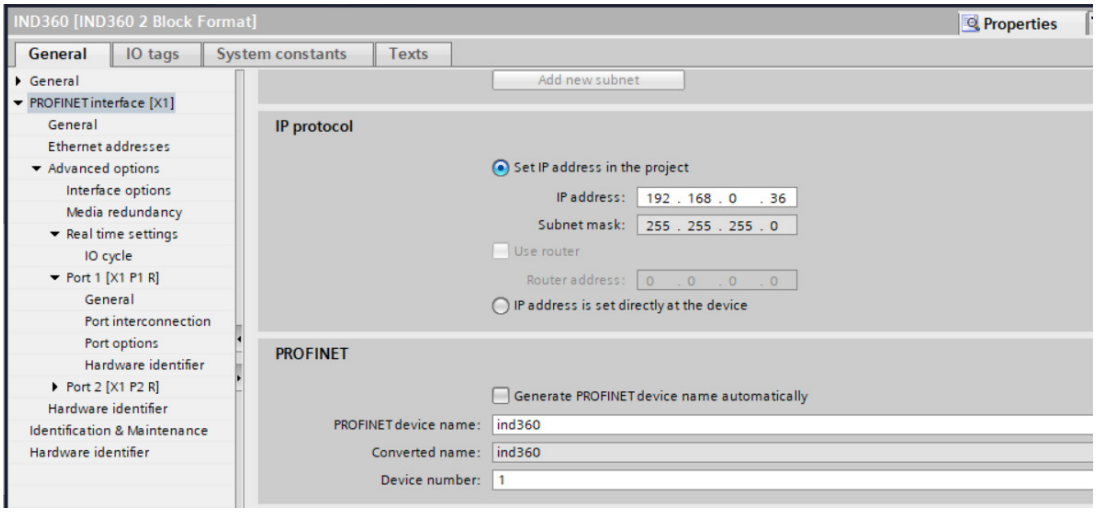

Figure 5-2: PROFINET Device Name and IP Address

6. Under **Devices & networks -> Topology view**, link the PLC to the IND360's network port 1 (left, NW1).

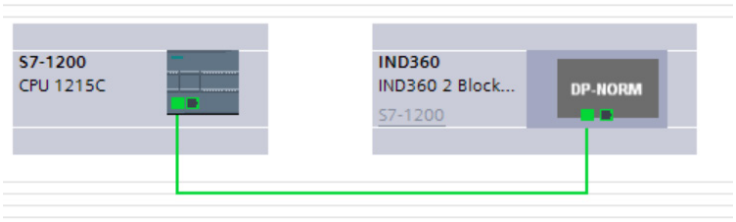

Figure 5-3: Devices & Networks, Topology view

7. The sample code follows the default I and Q addresses assignment as shown below. To minimize modifications to the code, consider sticking to the same I and Q address assignment.

Device overview

Module	Rack	Slot	I address	Q address	Type
IND360	0	0			IND360 2 Block Format
PROFINET	0	0 X1			IND360
Measuring Block 1_1	0	1			Measuring Block 1
Parameter Access Point	0	1 1			Parameter Access Point
MB Command Value	0	1 2		64...67	MB Command Value
MB Channel Mask	0	1 3		68...69	MB Channel Mask
MB Command	0	1 4		70...71	MB Command
MB Measuring Value	0	1 5	68...71		MB Measuring Value
MB Device Status	0	1 6	72...73		MB Device Status
MB Response	0	1 7	74...75		MB Response
Status Block_1	0	2			Status Block
Parameter Access Point	0	2 1			Parameter Access Point
SB Reserved 1	0	2 2		72...73	SB Reserved 1
SB Reserved 2	0	2 3		74...75	SB Reserved 2
SB Reserved 3	0	2 4		76...77	SB Reserved 3
SB Command	0	2 5		78...79	SB Command
SB Status Group 1	0	2 6	76...77		SB Status Group 1
SB Status Group 2	0	2 7	78...79		SB Status Group 2
SB Status Group 3	0	2 8	80...81		SB Status Group 3
SB Response	0	2 9	82...83		SB Response

Figure 5-4: Device I and Q Addresses

5.2. PLC Settings

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

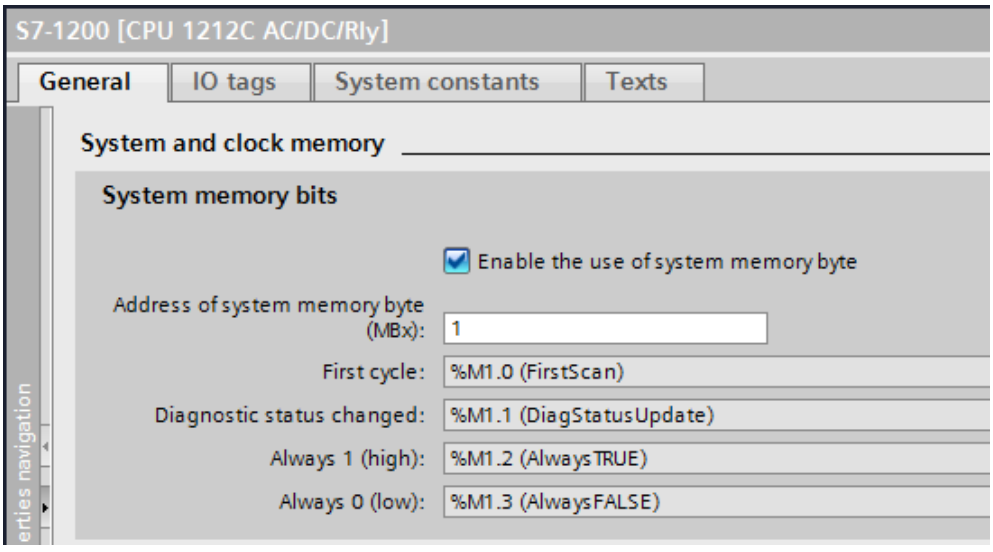

Figure 5-5: Enable System Memory Byte

5.3. Duplicate Programming Files

1. The required program blocks:
   a) MT_IND_Application (FC)
   b) SAI_IND360_TankVessel (FB), SAI_ IND360_TankVessel_DB (DB)
   c) Optional, for SAI 8-Block only: SAI_Cyclic_WeightProc (FB), SAI_Cyclic_WeightProc_DB (DB)
2. Delete the other unused program blocks in MT_IND360_Application.
3. Duplicate the "IND360" under the PLC tags.

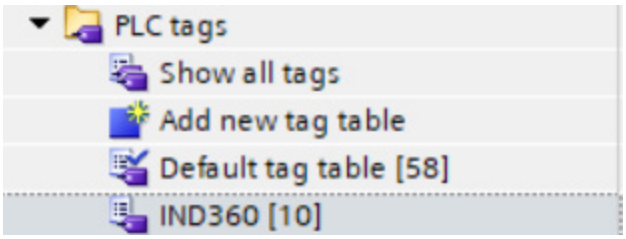

Figure 5-6: Duplicate the PLC Tags

4. Finally, in the Main (OB1) call up the function "MT_IND_Application".

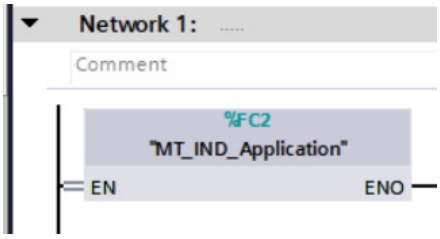

Figure 5-7: Call up "MT_IND_Application" in the Main OB