Balluff IO-Link Profibus Setup Guide

Configurator DP/PLC/HMI

Version

2.02

MEAU – Engineering Group

Content

Document Management Information

Date	Version No.	Document Name	Revision	Revised By
8/27/2015	1			MEAU-EG
1/15/2016	1.02			MEAU-EG
1/23/2016	1.03			MEAU-EG
6/29/2016	2.00		Additional devices, more process data FB and feature add to previous revisions	MEAU-EG
7/25/2016	2.01		Ultrasonic Device Type Error Corrected	MEAU-EG
8/11/2016	2.02		Corrected scaling for BNI0041; added condition for antenna ready	MEAU-EG

1 OVERVIEW

1.1 Overview of the Balluff IOL Profibus Setup Guide

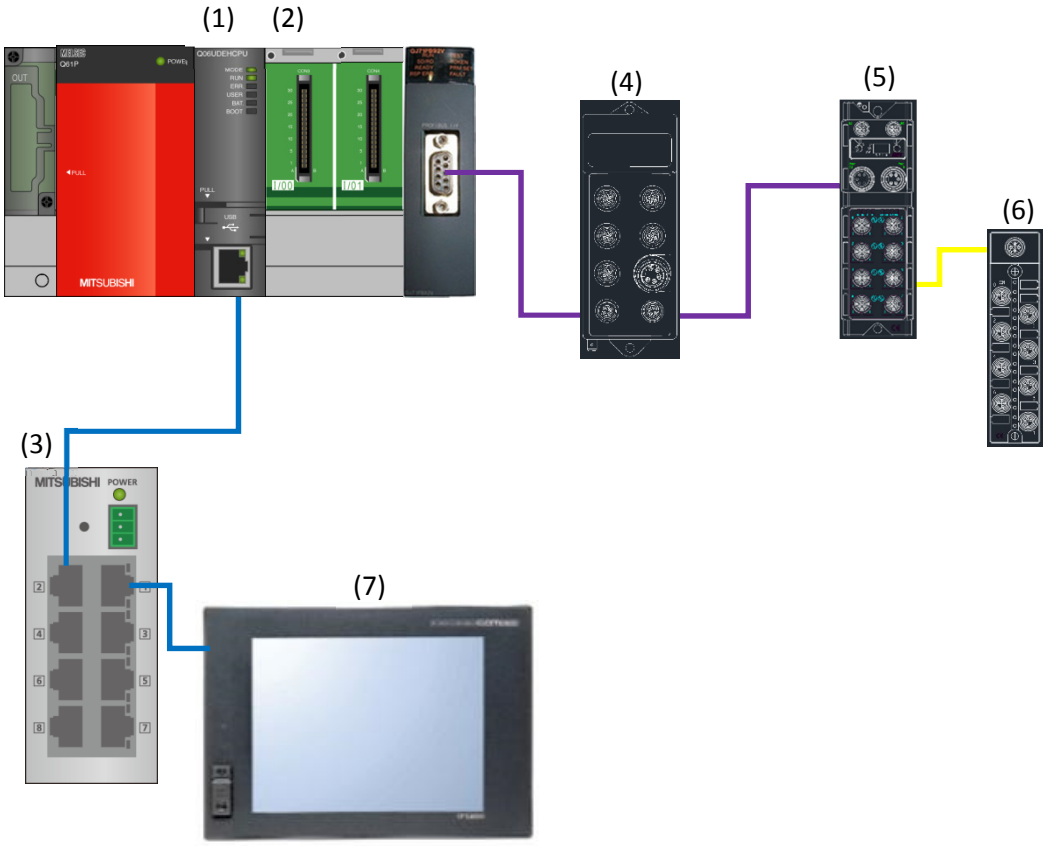
This guide shows a user how to setup the IOLink_Diag_DP Template Project for proper use of Balluff IO-link diagnostics, parameterization of IO-link devices, and formatting of the process data from Balluff IO-link devices.

This guide is divided into four main areas:

1. **Overview:** this section which gives general feeling for the software requirements and software package.
2. **Configurator DP Setup:** For proper Profibus/IO-Link configuration/for the QJ71PB92V
3. **GXWorks2 FB Setup:** For customization of the Profibus/IO-Link function blocks (FB) in the GXWorks 2 template project.
4. **GTDesigner3 Elements:** Description of Profibus/IO-Link HMI display screens and proper usage and element readouts.

1.2 Example of System Configuration

For Q series, the following configuration is used:



No.	Module	Description
1	Q series programmable controller	Use the base unit, power supply module, and Q series Built-in Ethernet port CPU module.
2	QJ71PB92V	Profibus Master Module
3	Ethernet Switch	Managed switch for systems-level Ethernet communications (may be separated into multiple switches if needed)
4	Profibus Node Device without IOL	Balluff Profibus Node with NO IOL communication capability

5	Profibus Node Device with IOL	Balluff Profibus Node with IOL communication capability
6	IOL Hub Device	Balluff IOL Device with Diagnostics
7	GOT 1000	HMI interface

1.3 Software and Project Requirements

1.3.1 Programming Software Requirements

To properly open, configure, edit and save project data for IO-Link diagnostics, the required software with minimum revision level is listed below:

Software Title	Version	Description
MELSOFT Navigator	2.12N+	Integrated project for storing and configuring both the PLC and HMI projects. System labels are used to allow common labels between PLC and HMI with this software. IO-Link indicators on the HMI use system labels extensively.
GX Works2	1.540	PLC software needed for all PLC-related configuration and programming. Used in conjunction with MELSOFT Navigator. Numerous FBs require configuration for IO-Link Diagnostics to work properly.
GT Designer3 (GOT 1000 version)	1.136S+	HMI software needed for all GOT-related configuration and programming. Used in conjunction with MELSOFT Navigator. IO-Link diagnostics and Parameter read/write screens reside here.
GX Configurator DP	7.12N+	Profibus configuration software to set up all Profibus devices on a network. Used in conjunction with GX Works 2 to map I/O and diagnostic data to/from PLC. Specific IO-Link mapping and Profibus Node master device mapping is needed for proper diagnostics display.

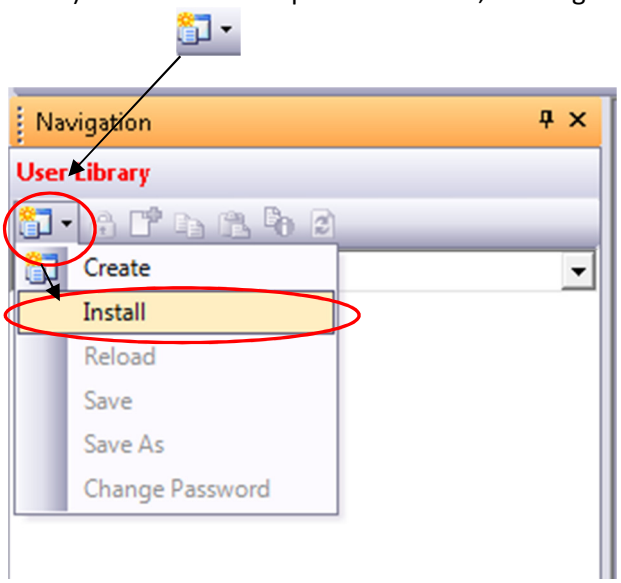
1.4 Importing User Libraries

This section explains how to import any User Library into GX Works2.

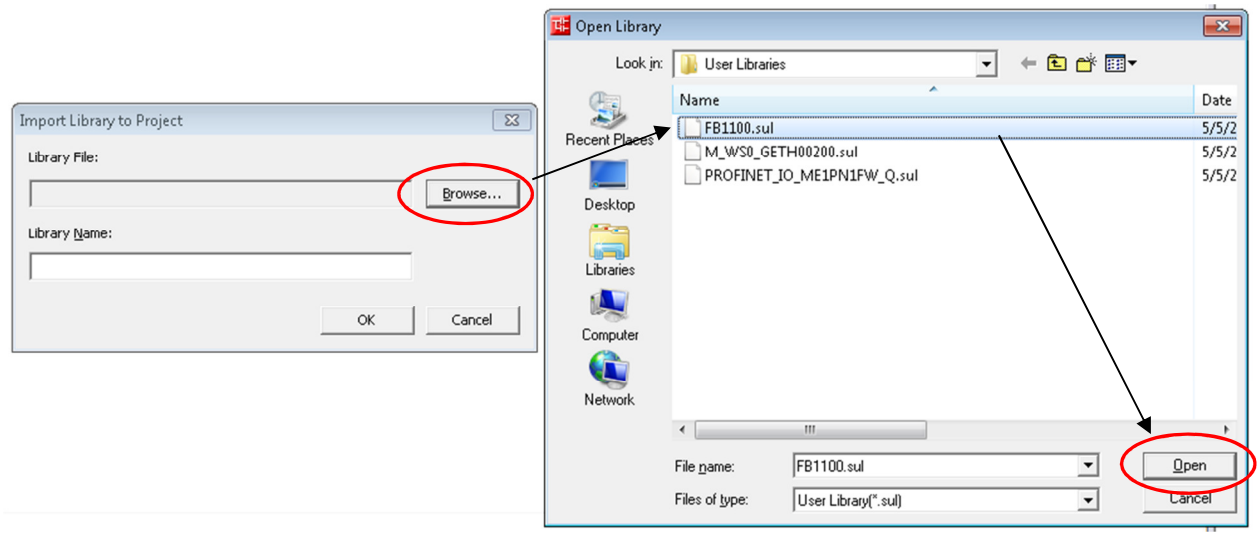
There are five key libraries for this project

1. DP_IOLDiag_v200
    a. This library contains structures and function blocks for IO-Link Diagnostics and maintenance
2. Balluff_IOL_CFG_V201
    a. This library contains structures and function blocks related to the configuration of Balluff hardware
3. Balluff_IOL_ProcessData_v202
    a. This library contains structures and function blocks related to utilizing process data for connected IOL Devices
4. RFID_Balluff_BIS_v304
    a. This library contains structure and FB in regards to RFID access for the BIS-M, BIS-V, and Io-link RFID
5. DPV1_accessory_v120
    a. This library contains Acyclic Profibus functions and IO Link Device Call

1) Click on the “User Library” tab on the GX Works 2 Navigation Pane.
2) Click the arrow part of this icon, to bring a drop-down menu. Select the “Install” option:

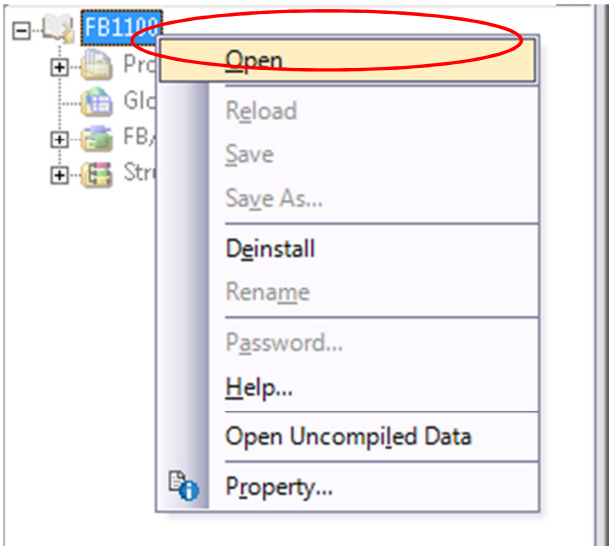


3) From the “Import Library to Project” window, click the Browse button, find the .sul library file, and click Open:


Import Library to Project
Library File:
Browse...
Library Name:
OK
Cancel
Open Library
Look in:
User Libraries
Recent Places
Desktop
Libraries
Computer
Network
Name
Date
FB1100.sul
5/5/2
M_WS0_GETH00200.sul
5/5/2
PROFINET_IO_ME1PN1FW_Q.sul
5/5/2
File name:
FB1100.sul
Files of type:
User Library(*.sul)
Open
Cancel

4) Click the “OK” button. Click “OK” again when prompted that the library has been installed.

The library elements will now show up as a directory tree in the Navigation Pane. Right click on the top folder and select “Open” to view/edit the library elements:



2 CONFIGURATOR DP PROFIBUS SETUP

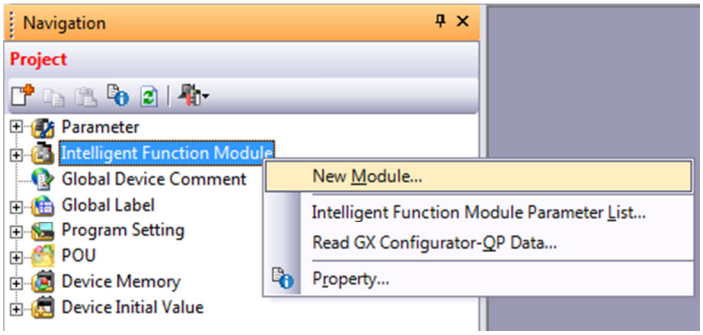
Configurator DP is used to set up the Profibus network configuration in the QJ71PB92V module. This section will explain how to set up a new configuration and populate the network with one Profibus node device with IO-Link capability. It will also show how to configure all the necessary diagnostic and process data needed for each node and IO-Link device.

IMPORTANT! GXWorks 2 will be used to do the actual Profibus setup in this procedure. When the GX Configurator DP software is installed, it is embedded in GXWorks 2. After the Profibus configuration is completed and saved, the configuration can be written to the module from the Online Data Operation Window.

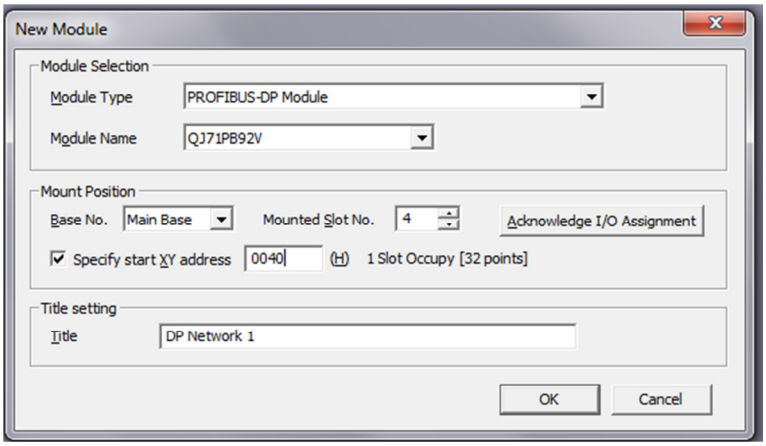
Important! Before any configuration can be done on a Profibus node device, a GSD file specific to that device must be imported into the Configurator DP software. GSD files for target Balluff Profibus devices can be found on the Balluff website (http://www.balluff.com)

2.1 Creating a new Profibus Configuration

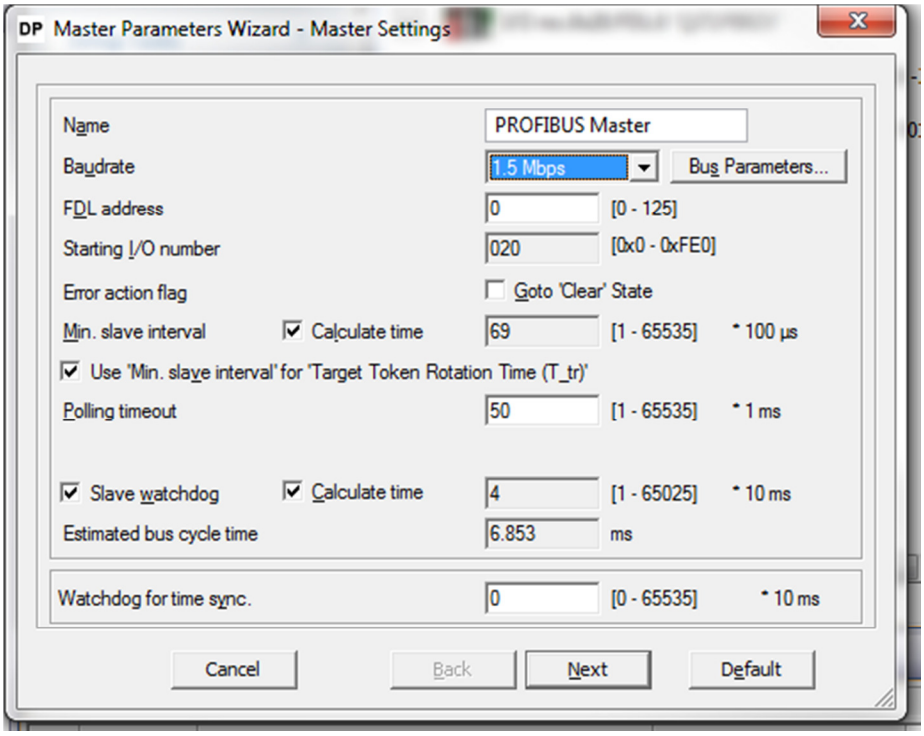
1. Open the target **GXWorks 2** PLC project used for your application.
2. On the main navigation side bar, select the **Project** tab.
3. Right click on the **Intelligent Module Function** item and select **New Module**:



4. On the **New Module** window, fill in the hardware details that match your module and slot configuration and click **OK**:



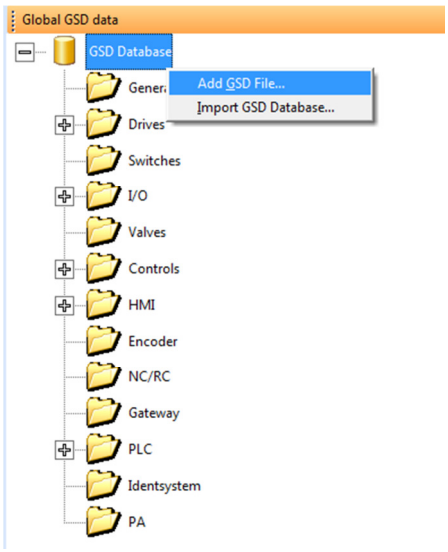
5. On the **Setup Tasks** area, click the **Master Settings…** option.
6. Set the Master device according to the screenshot shown on the next page:



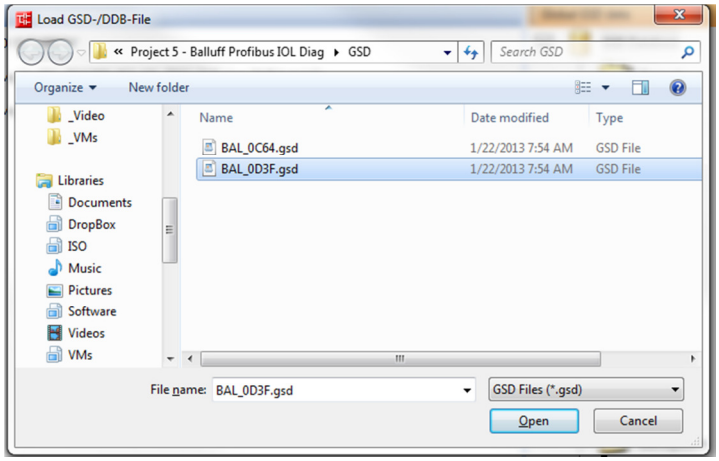
NOTE: These are default settings only. The target project may use custom settings that may require changes to these settings.

2.2 Importing GSD file to GX Configurator DP

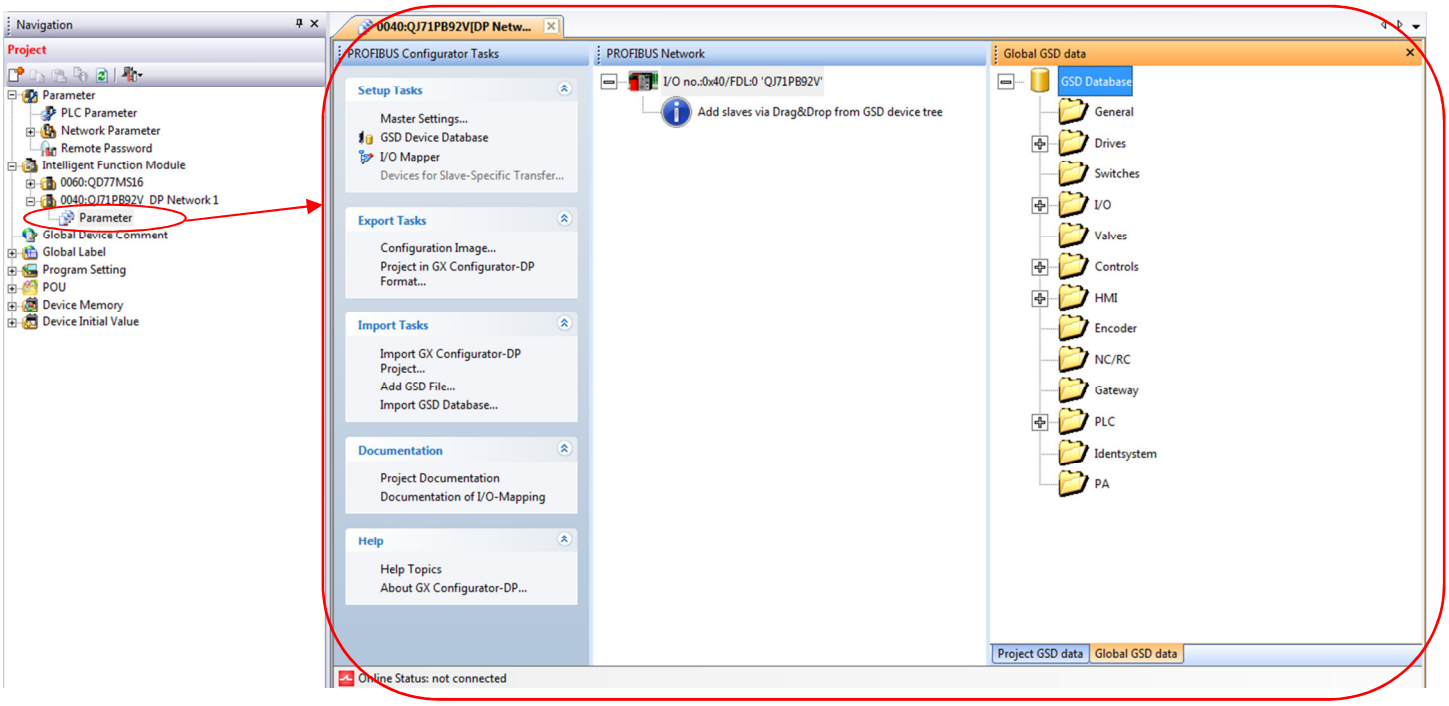
1. Open **GX Configurator DP or the GX Works 2 project that contains the Profibus configuration.**
2. From the **Global GSD** data window, right click on the GSD Database container and select **Add GSD File…**



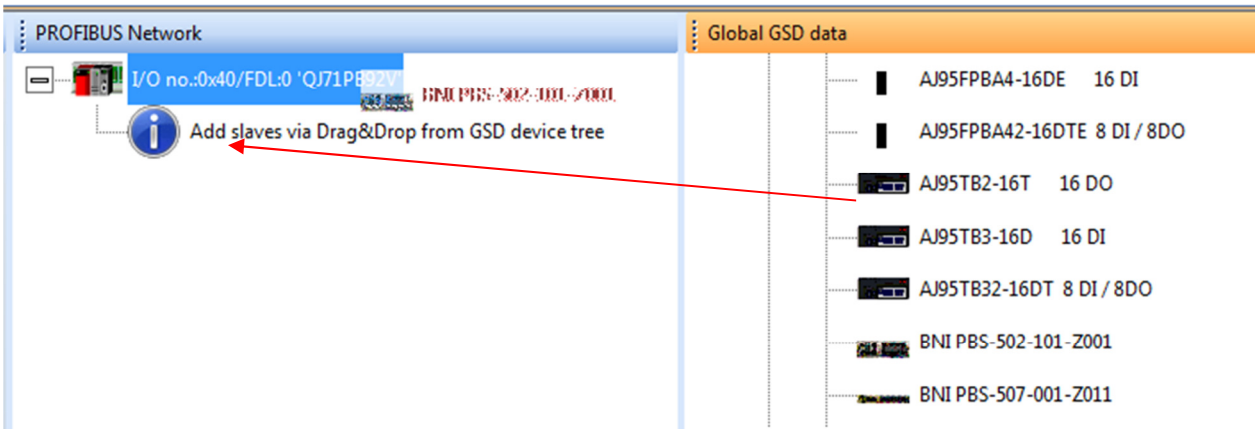
3. From the Load GSD-/DDB window, navigate to the target GSD file and click **Open**:



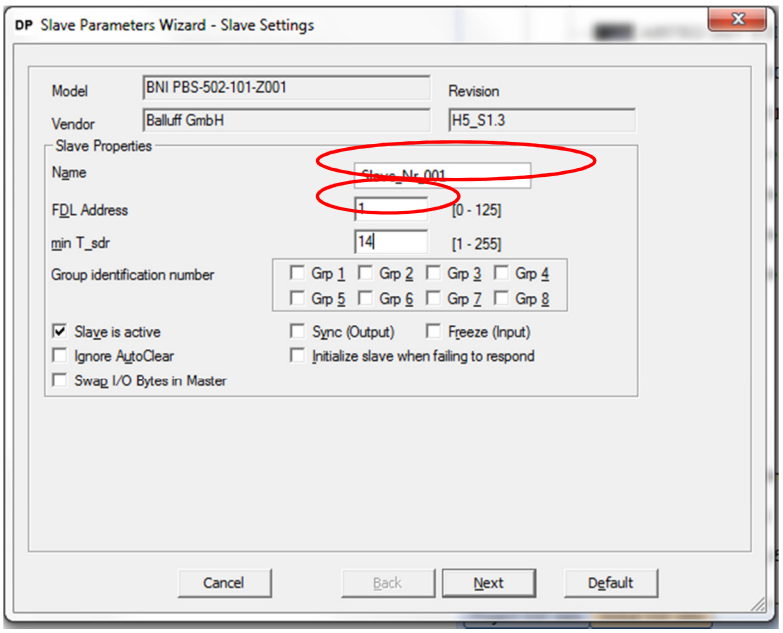
4. Click **Yes** to confirm addition of new device.



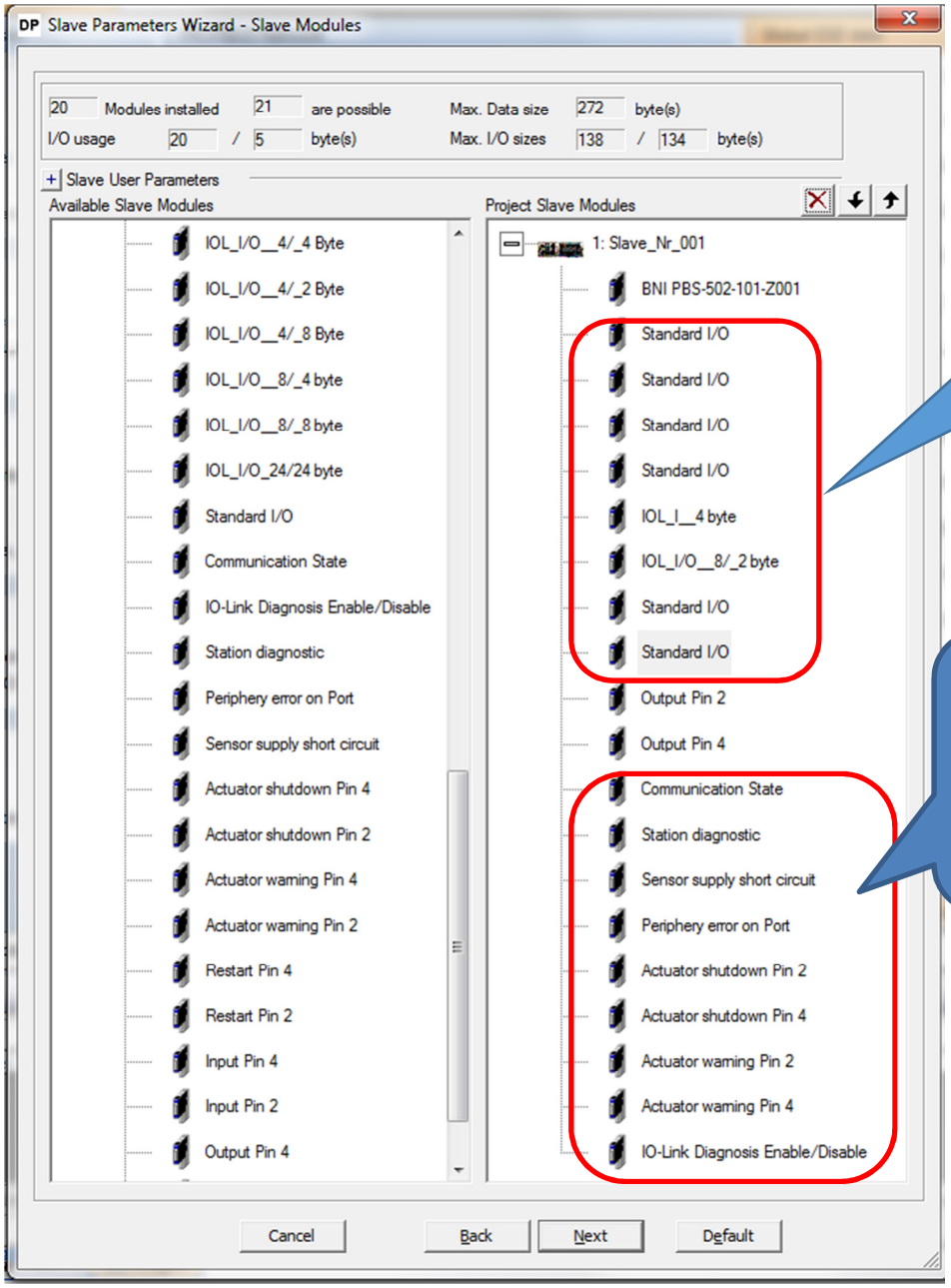
2.3 Add a Profibus Node to the Project Configuration

1. From the **Global GSD data** window, click and drag each Profibus module type on the target network to the **PROFIBUS Network** window:
   NOTE: If the node device is not already in the Global GSD library, a GSD file for that module will need to be imported. Refer to section **2.2 - Importing GSD file to GX Configurator DP** after acquiring the proper GSD file.
2. 

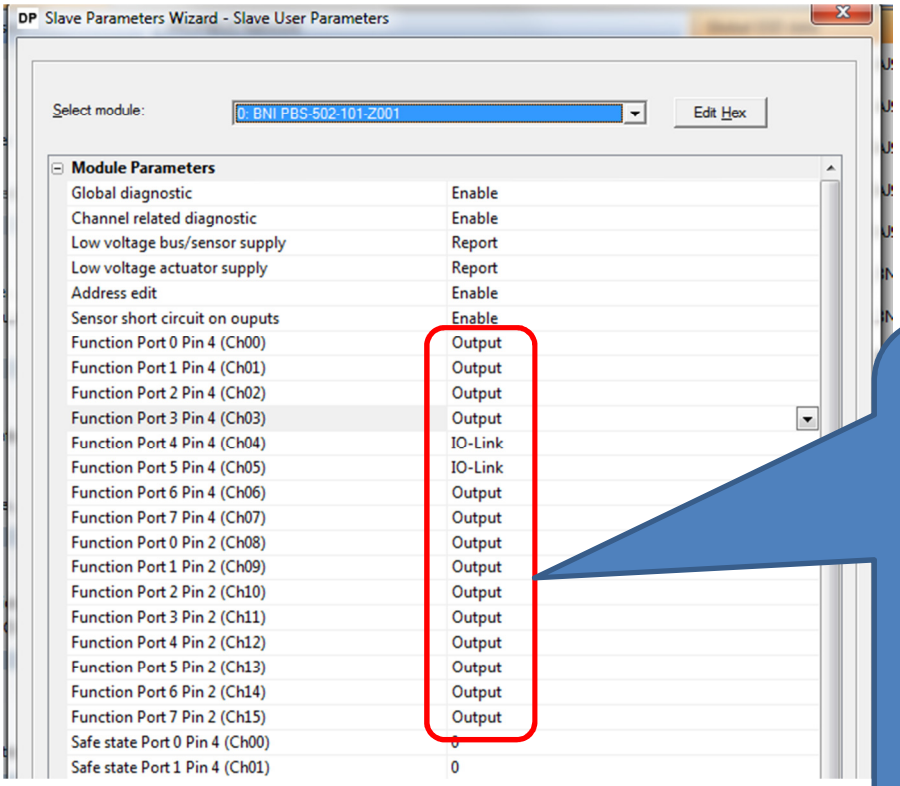
3. On the resulting **Slave Parameters Wizard** window, start by setting the **Name** and **FDL Address** of the target node device. All other settings should be left to default (see next page):



4. Click the **Next** button.
5. The next window allows drag and drop “Slave Modules” that allow mapping for IO Link and node-specific diagnostic data.



6. Click the **Next** button. On the next window, set the ports and pins for IO-Link and/or standard I/O. See the next page for more details:



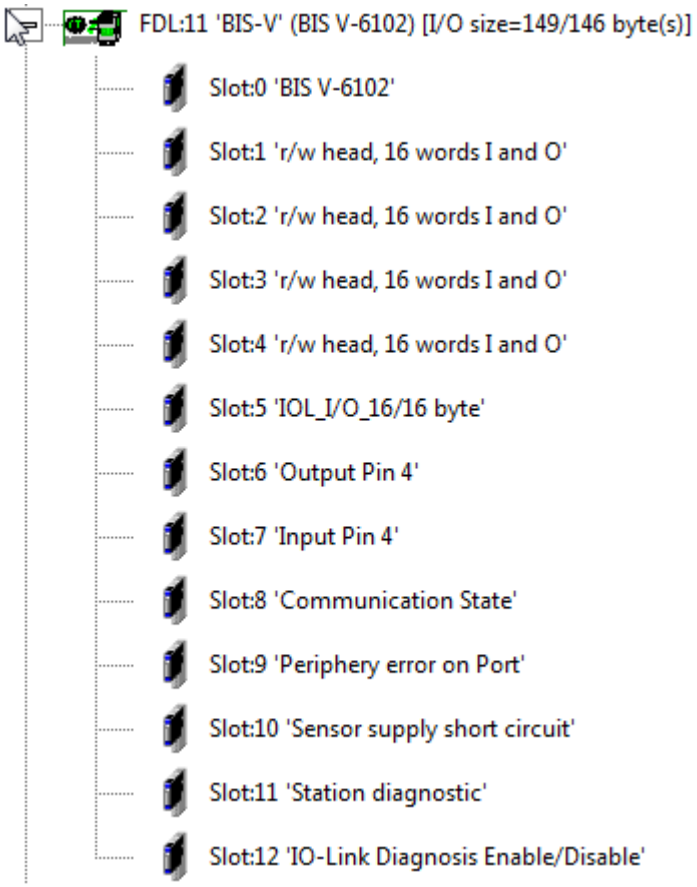
Each Port on the device has a Pin 2 and Pin 4. Pin 4 can be switched to IO Link while other Ports and pins can use standard input and/or outputs, depending on device type. Refer to the device's manual for more information on possible configurations

7. Click the **Next** button, then click the **Finish** button.

Repeat this procedure for each node device in the target Profibus network.

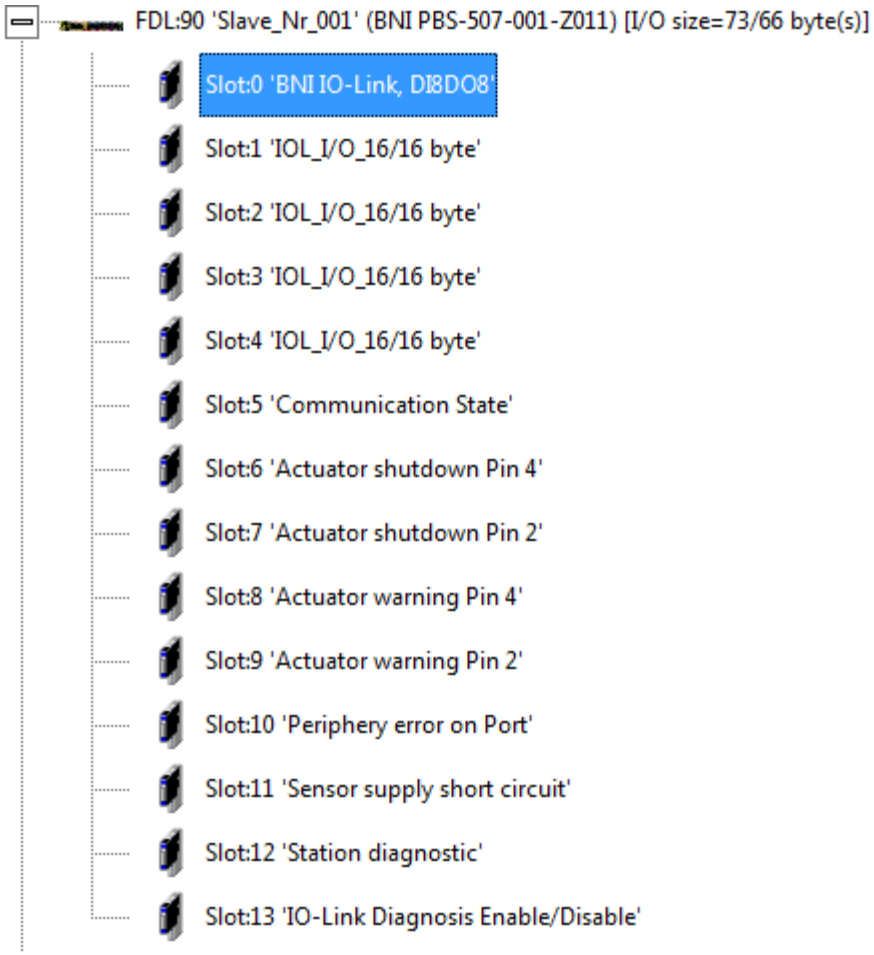
2.3.1 BIS013W Slot Configuration

For the Master function blocks to function properly Slot 8 through Slot 12 must be configured in the order shown:



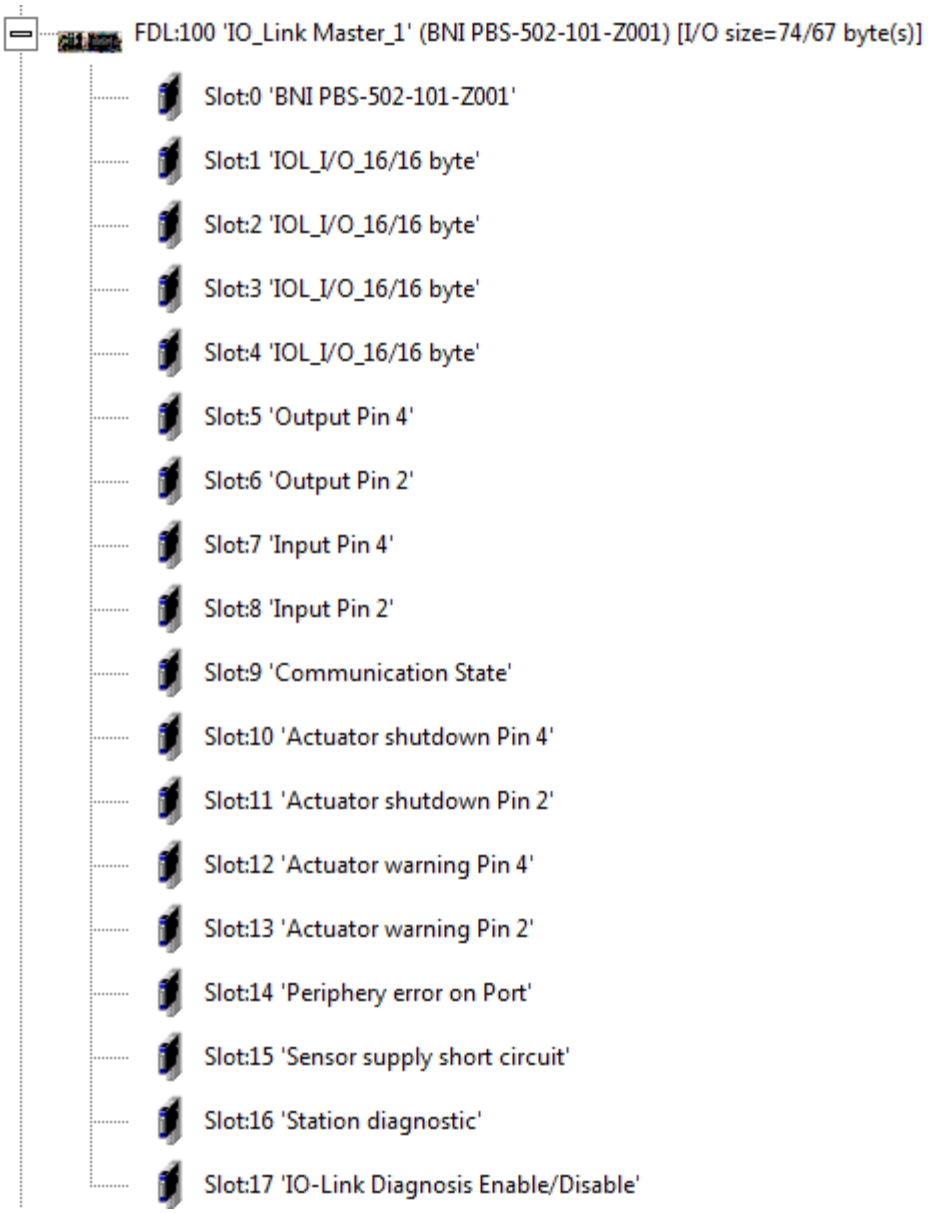
2.3.2 BNI003P Slot Configuration

For the Master function block to operate properly slot 5 through 13 must be configured in the order shown:

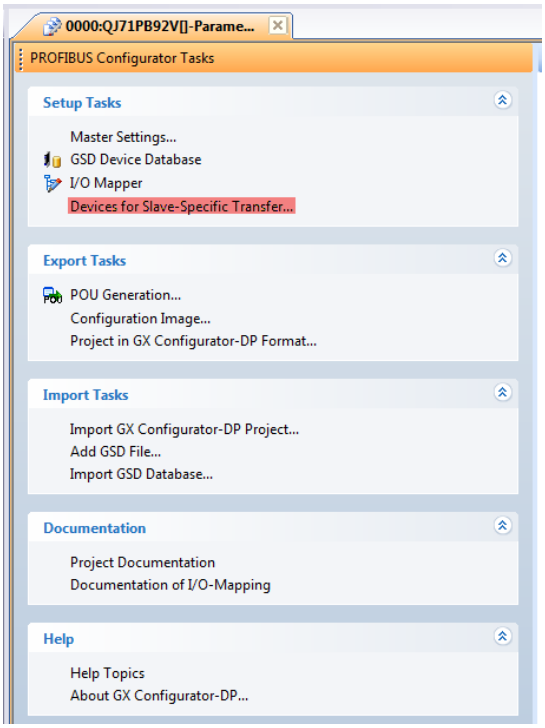


2.3.3 BNI005R Slot Configuration

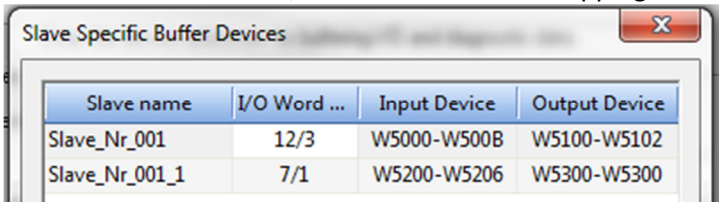
For the Master function block to operate properly slot 9 through 16 must be configured in the order shown:



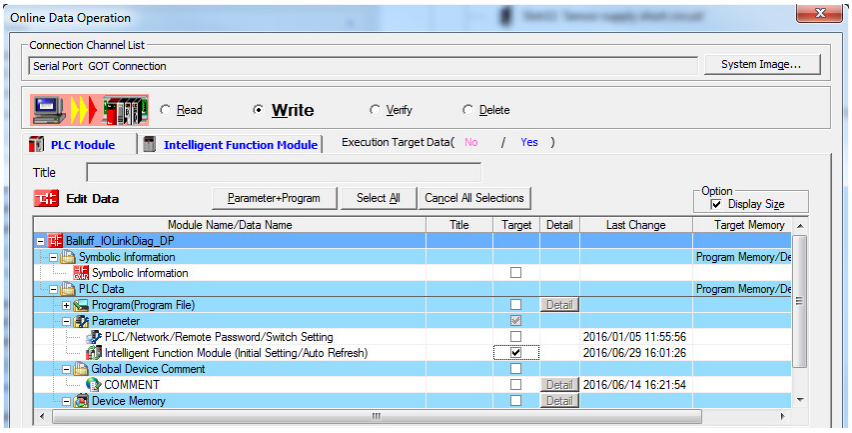
2.4 Setting Auto-Refresh Ranges



1. On the Slave Specific Buffer Devices window, set the slave device mapping to the desired ranges:



2. These settings are updated when the intelligent function module parameters are written to the PLC

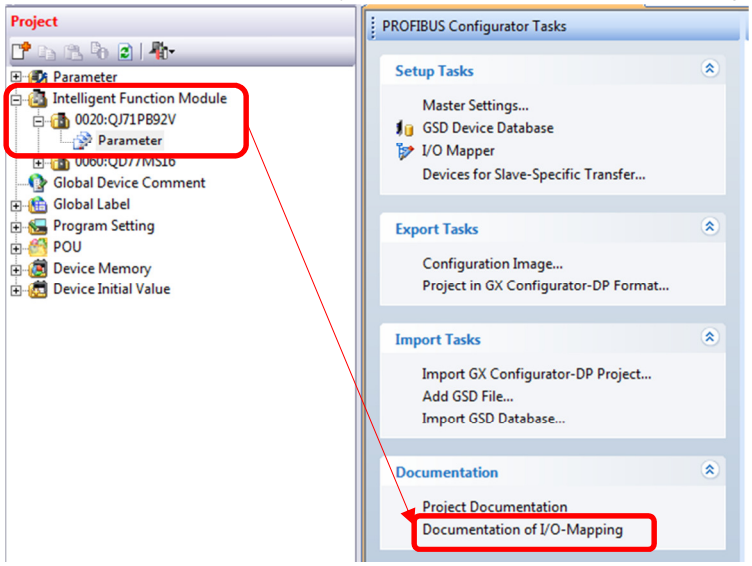


2.5 Referencing I/O/Diagnostic Data Mapping

When a Profibus network configuration is completed, the actual device mapping for each slot of each node and hub can be referenced within GX Works 2 or the GX Configurator DP software.

Most times, the GXWorks 2 PLC template project will automatically handle all node and hub data for diagnostic and parameter display/manipulation but there may be a need to reference this material for added features.

1. In GX Works 2, go to the **Project** tab of the Navigation window. Inside the **Intelligent Module** folder, expand the **QJ71PB92V** module and double-click on **Parameter**.
2. On the Documentation area, click on the **Documentation of I/O-Mapping** button (see below):



NOTE: In GX Developer DP, with a project loaded, go to the **PROFIBUS Configuration Tasks** pane and select **Documentation of I/O-Mapping** from the **Documentation** area.

3. The resulting report will load on your default web browser on your PC.
4. The report is broken up into 2 major areas:
    a. A master slot table of all slave nodes on the ProfiNet network with hyperlink shortcuts to more details
    b. Details tables for each slot configured for each slave node on the ProfiNet network.

Example with two ProfiNet node devices (No 14 and 15)

Project: QJ71PB92V_0001.dp2

FDL Addr.	Name	Model	Modules

Slot	Model	Global Var.
0	BNI PBS-502-101-Z001	-
1	IOL_I__4 byte	vHA20SLV14MOD1
2	IOL_I/O__8/_2 byte	vHA20SLV14MOD2
3	Standard I/O	-
4	IOL_I__2 byte	vHA20SLV14MOD4
5	Input Pin 4	vHA20SLV14MOD5
6	Input Pin 2	vHA20SLV14MOD6
7	Output Pin 4	vHA20SLV14MOD7
8	Output Pin 2	vHA20SLV14MOD8
9	Communication State	vHA20SLV14MOD9
10	Station diagnostic	vHA20SLV14MOD10
11	Sensor supply short circuit	vHA20SLV14MOD11
12	Periphery error on Port	vHA20SLV14MOD12
13	Actuator shutdown Pin 2	vHA20SLV14MOD13
14	Actuator shutdown Pin 4	vHA20SLV14MOD14
15	Actuator warning Pin 2	vHA20SLV14MOD15
16	Actuator warning Pin 4	vHA20SLV14MOD16
17	IO-Link Diagnosis Enable/Disable	vHA20SLV14MOD17

14 | Slave_Nr_001 | BNI PBS-502-101-Z001

Slot	Model	Global Var.
0	BNI IO-Link, DI8DO8	vHA20SLV15MOD0
1	IOL_I__4 byte	vHA20SLV15MOD1
2	Standard I/O	-
3	Standard I/O	-
4	Standard I/O	-
5	Communication State	vHA20SLV15MOD5
6	Station diagnostic	vHA20SLV15MOD6
7	Sensor supply short circuit	vHA20SLV15MOD7
8	Periphery error on Port	vHA20SLV15MOD8
9	Actuator shutdown Pin 2	vHA20SLV15MOD9
10	Actuator shutdown Pin 4	vHA20SLV15MOD10
11	Actuator warning Pin 2	vHA20SLV15MOD11
12	Actuator warning Pin 4	vHA20SLV15MOD12
13	IO-Link Diagnosis Enable/Disable	vHA20SLV15MOD13

15 | Slave_Nr_001_1 | BNI PBS-507-001-Z011

Basic node device info as configured in the General Configuration of the module

Slot Data with clickable hyperlinks as configured in the Module Configuration of the module

Element Identifier	Element Type	Class	User MIT-Address	Global Var. Identifier	Buffer MIT-Address
inputs	ARRAY [0..1] OF WORD	Input	-	-	W5000 (%MW1.20480)

Slave_Nr_001.Module Slot 2 : vHA20SLV14MOD2

Element Identifier	Element Type	Class	User MIT-Address	Global Var. Identifier	Buffer MIT-Address
inputs	ARRAY [0..3] OF WORD	Input	-	-	W5002 (%MW1.20482)
outputs	WORD	Output	-	-	W5100 (%MW1.20736)

Slave_Nr_001.Module Slot 4 : vHA20SLV14MOD4

Element Identifier	Element Type	Class	User MIT-Address	Global Var. Identifier	Buffer MIT-Address
inputs	WORD	Input	-	-	W5006 (%MW1.20486)

Slave_Nr_001.Module Slot 5 : vHA20SLV14MOD5

Element Identifier	Element Type	Class	User MIT-Address	Global Var. Identifier	Buffer MIT-Address
inputs	ARRAY [0..7] OF BOOL	Input	-	-	W5007.0 (%MX1.20487.0)

Slave_Nr_001.Module Slot 6 : vHA20SLV14MOD6

Element Identifier	Element Type	Class	User MIT-Address	Global Var. Identifier	Buffer MIT-Address
inputs	ARRAY [0..7] OF BOOL	Input	-	-	W5007.8 (%MX1.20487.8)

Slave_Nr_001.Module Slot 7 : vHA20SLV14MOD7

Element Identifier	Element Type	Class	User MIT-Address	Global Var. Identifier	Buffer MIT-Address
outputs	ARRAY [0..7] OF BOOL	Output	-	-	W5101.0 (%MX1.20737.0)

Slot Data for the first IO-Link Hub device for node device 15. Actual breakdown of I/O signals and diagnostic indicators can be referenced in that device's User Guide.

			Slot	Model	Global Var.
15	Slave_Nr_001_1	BNI PBS-507-001-Z011	0	BNI IO-Link, DI8DO8	vHA20SLV15MOD0
			1	IOL_I__4 byte	vHA20SLV15MOD1
			2	Standard I/O	-
			3	Standard I/O	-
			4	Standard I/O	-
			5	Communication State	vHA20SLV15MOD5
			6	Station diagnostic	vHA20SLV15MOD6
			7	Sensor supply short circuit	vHA20SLV15MOD7

Clicking on one of these hyperlinks will navigate to the appropriate details section of the report

Slave_Nr_001_1.Module Slot 1 : vHA20SLV15MOD1

Element Identifier	Element Type	Class	User MIT-Address	Global Var. Identifier	Buffer MIT-Address
inputs	ARRAY [0..1] OF WORD	Input	-	-	W5200.8 (%MX1.20992.8)

Slave_Nr_001_1.Module Slot 5 : vHA20SLV15MOD5

Element Identifier	Element Type	Class	User MIT-Address	Global Var. Identifier	Buffer MIT-Address
inputs	ARRAY [0..7] OF BOOL	Input	-	-	W5202.8 (%MX1.20994.8)

Slave Nr 001 1.Module Slot 6 : vHA20SLV15MOD6

Number of byes used. In this case, 4 are used since a two-element array of words is configured

Starting device address as this slot data as configured in the **Slave Specific Transfer** settings .

NOTE: Sometimes mapping for these "slots" may start on the high byte of the target word. If so, the start address will be at bit 8 of the word as shown in this example.

2.6 Saving and Downloading Configuration to Profibus Master Module

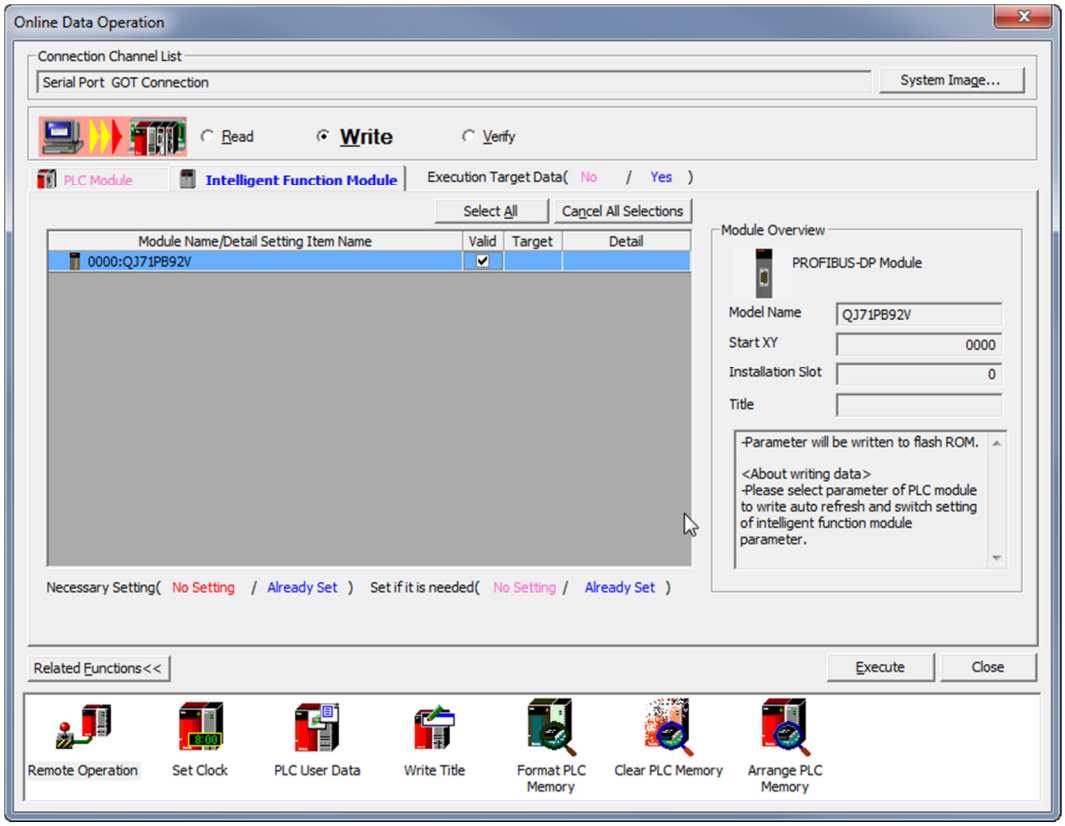
After all the preceding steps have been complete:

- New Configurator created with Master settings
- GSD files imported (if needed)
- Node and Hub devices mapped
- Slave-specific device mapping applied

Then, the final configuration can be saved and downloaded to the Profibus Master module

1. From GX Work 2, the **Online>Write To PLC** has an **Intelligent module** tab that allows transfer of parameters to the QJ71PB92V module.

NOTE: After writing the module's parameters, always reset your Main CPU processor for changes to take effect.



3 GX WORKS 2 PLC SETUP

From the template project, much of the Profibus and IO-Link Diagnostics functionality is configured to allow for dynamic changes to the setup. However, some verification and changes are required for proper Profibus operation on a target system.

The template project is designed for a Q06UDEH processor, with a Profibus Master module(QJ71PB92V) in Slot 0

In subsequent sections we will cover:

1. PLC Parameters and required programming
2. IO-Link device configuration function blocks
3. IO-Link Master Configuration
4. IO-Link Diagnostics required programming
5. IO-Link Device Process Data function blocks
6. RFID required programming

3.1 PLC Parameters and required Programming

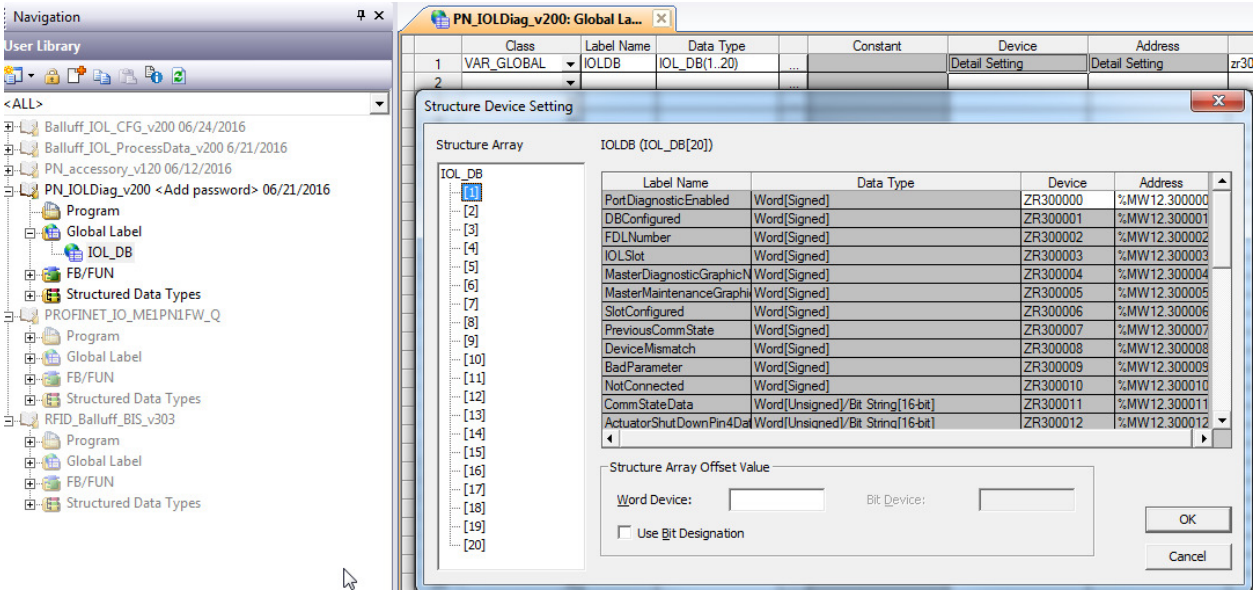
Whether starting with the template project or importing or copy/pasting elements into an existing project, there are common PLC parameters required for correct operation.

PLC file registers are used in our example template, the maximum non-extended size of a Q06UDEH, 384 K points is defined.

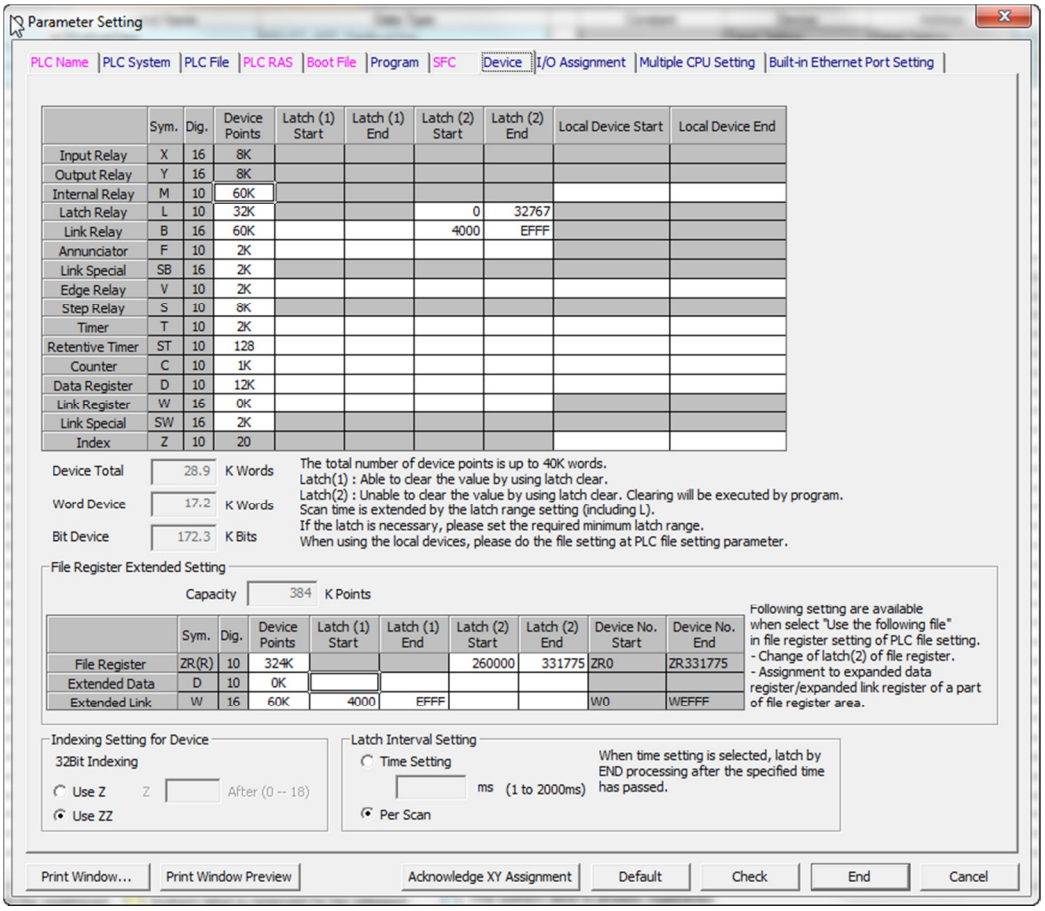
This project uses retentive registers for storing data through a power cycle.

IO-link Data (IOLDB) needs to be stored in retentive word memory consisting of 9340 words. The template project stores this data within ZR300000-ZR309339.

This can be configured from the Global Label settings of the DP_IOLDiag library

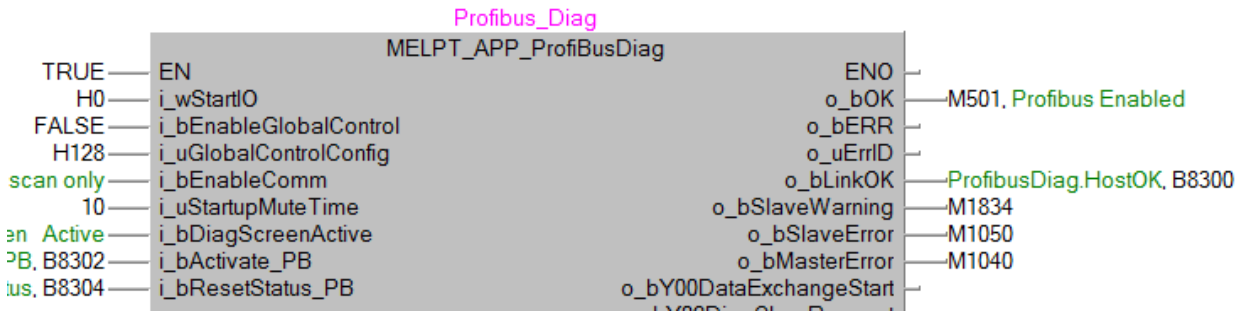


Profibus Diagnostics needs to store the deactivated nodes (128 bits) in retentive bit memory. The template project uses B8300-B83FF to store this retained data.



Proper operation of IO-Link diagnostics requires the use and operation of the MELPT_APP_ProfibusDiag function block. The template version uses version 3.03.

3.1.1 Configuration of MELPT_APP_ProfibusDiag



Purpose: Turns on the cyclic communication functionality of the DP controller. Determines Fieldbus Status for individual nodes, allows the ability to reserve/activate a node. Retrieves channel diagnostics. Provides alarm summary bits

User Configuration: Profibus controller location, attaching alarm outputs to HMI alarm memory.

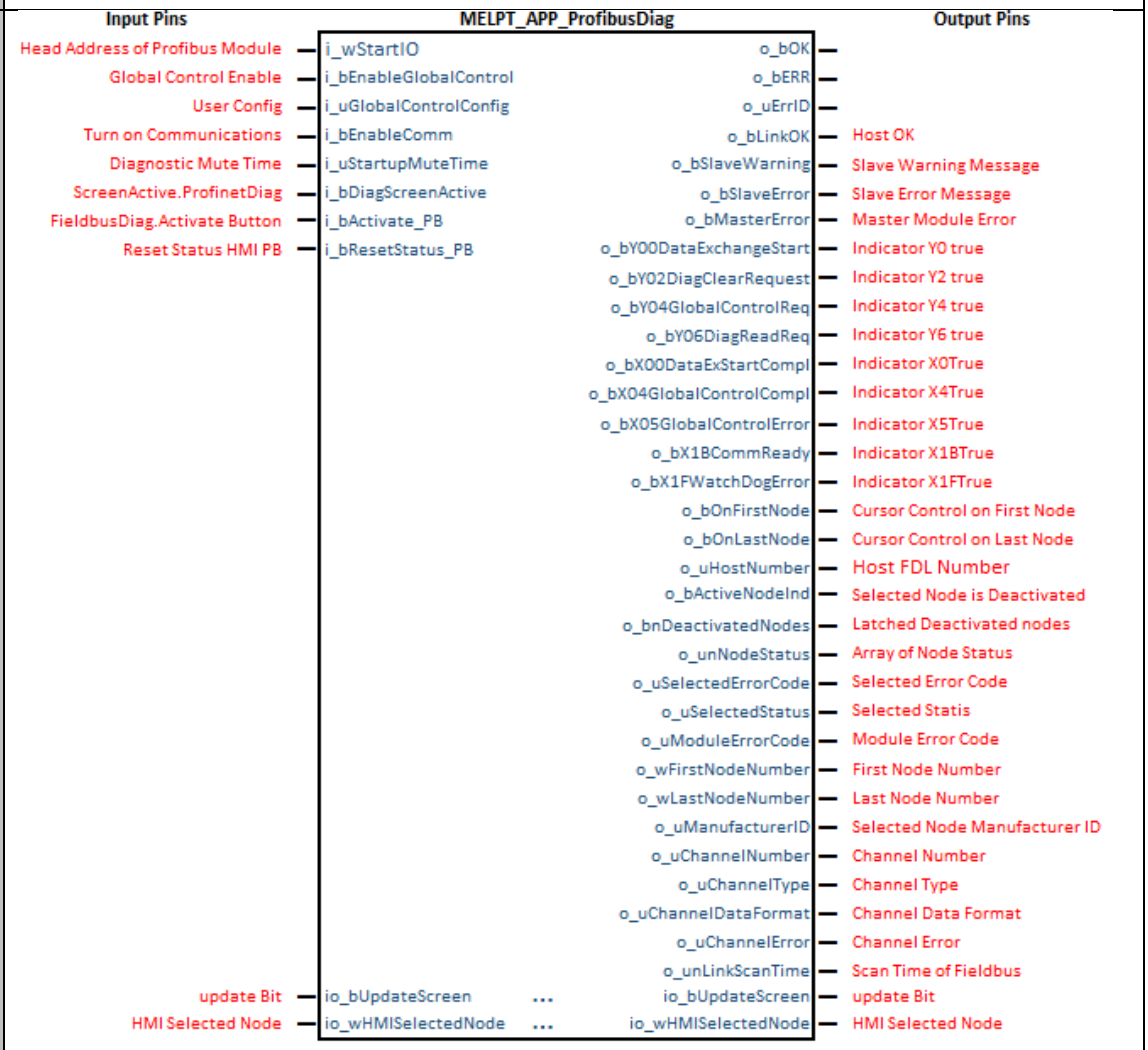
This function block is used to monitor overall node health for the fieldbus and provide detailed diagnostics for the selected fieldbus node.

Note: If a diagnostic is occurring when the io-link diagnostic is disabled. An electrical change has to occur once it is enabled for the diagnostic to be reported to the fieldbus diagnostic programming.

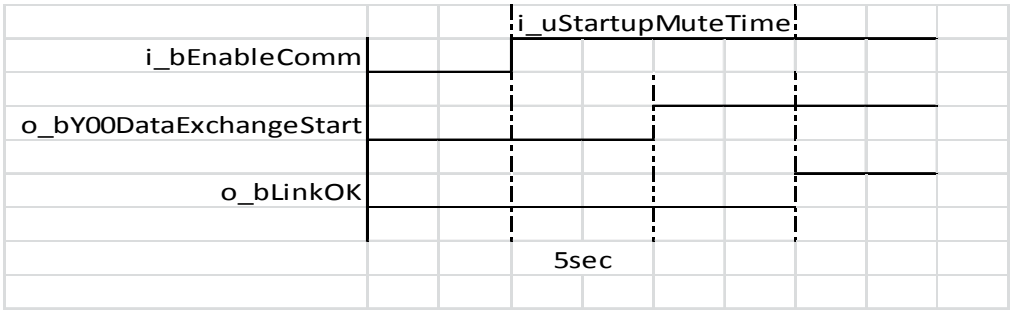
3.1.2 PLC-MELPT_APP_ProfibusDiag

This program calls the function block MELPT_APP_ProfibusDiag. This function block is used to monitor overall node health for the fieldbus and provide detailed diagnostics for the selected fieldbus node.

FB Name
MELPT_APP_ProfibusDiag

Item	Description
Function overview	This function block monitors Profibus node health, starts the fieldbus, and allows a method to reserve/activate nodes
Symbol	

<table>
<tr><th>Item</th><th>Description</th></tr>
<tr><td>Applicable Hardware</td><td>
<table>
<tr><th>Series</th><th>Model</th><th>Serial Restriction</th></tr>
<tr><td>MELSEC-Q series</td><td>Universal Model</td><td>None</td></tr>
<tr><td>MELSEC-Q Series intelligent</td><td>QJ71PB92V</td><td>None</td></tr>
<tr><td>GOT 1000 series</td><td>GT16(800*600) or Higher</td><td>None</td></tr>
</table>
</td></tr>
<tr><td>Applicable Software</td><td>
<table>
<tr><th>Series</th><th>Version</th></tr>
<tr><td>GX Works 2</td><td>1.536</td></tr>
<tr><td>GT Designer 3</td><td>1.136</td></tr>
<tr><td>Configurator-DP</td><td>7.12</td></tr>
</table>
</td></tr>
<tr><td>Programming language</td><td>Structured Ladder/FBD</td></tr>
<tr><td>Number of Ladder Steps</td><td>QnU: 767 steps
*The number of steps of the FB program depends on the CPU Model that is used and input and output definition</td></tr>
<tr><td>Device Memory Used</td><td>936 bits
1280 words
3 timer</td></tr>
<tr><td>Compiling method</td><td>Macro type;</td></tr>
<tr><td>Execution type</td><td>Real-time Execution</td></tr>
<tr><td>Dependences</td><td>This FB requires the function MELPT_APP_IntModuleCheck</td></tr>
<tr><td>Function description</td><td>This function block enabled the Profibus fieldbus, monitors node health, sets alarms for node issues, interrogates an individual node for channel diagnostics, and manages reserved nodes</td></tr>
</table>

Item	Description
Restrictions and precautions	This function block uses Z9, z8, z7, z16, z18
Timing chart	

FB Error Code

Error Code	Description
0	No Error
H100 (256)	Profibus Module not detected at i_wStartIO

Labels

■ Input labels

User Input	Symbol Name	Var_Input name	Data Type	Setting range	Description
X	Head Address of Profibus Module	i_wStartIO	Word[Signed]	0 to 7E0	Head Address Profibus Module in HEX
X	Global Control Enable	i_bEnableGlobalControl	Bit		On Enables Global Control function
X	User Config	i_uGlobalControlConfig	Word[Unsigned]		Configuration for Un\G2081

User Input	Symbol Name	Var_Input name	Data Type	Setting range	Description
X	Turn on Communications	i_bEnableComm	Bit		Y0
X	Diagnostic Mute Time	i_uStartupMuteTime	Word[Unsigned]		Mute Timer for status
	ScreenActive.ProfinetDiag	i_bDiagScreenActive	Bit		On Profibus Diag Screen
	FieldbusDiag.Activate Button	i_bActivate_PB	Bit		Activate Node
	Reset Status HMI PB	i_bResetStatus_PB	Bit		Reset Status

■ Input/Output labels

User Input	Name	Var_In_Out name	Data Type	Setting range	Description
	update Bit	io_bUpdateScreen	Bit		Refresh Diag Screen
	HMI Selected Node	io_wHMISelectedNode	Word[Signed]		HMI numeric Input of FDL address

■ Output labels

Name	Var_Output name	Data Type	Description
FB Executes Normally	o_bOK	Bit	When TRUE, indicates processing has completed normally
FB Execution Abnormal	o_bERR	Bit	When TRUE, indicates an Error has occurred
FB Error Code	o_uErrID	Word[Unsigned]	FB Error Code Output
Host OK	o_bLinkOK	Bit	Status of Data Link
Slave Warning Message	o_bSlaveWarning	Bit	A slave currently has diagnostic message

Name	Var_Output name	Data Type	Description
Slave Error Message	o_bSlaveError	Bit	A slave currently is disconnected
Master Module Error	o_bMasterError	Bit	Master has local Diag Info
Indicator Y0 true	o_bY00DataExchangeStart	Bit	y0
Indicator Y2 true	o_bY02DiagClearRequest	Bit	y2
Indicator Y4 true	o_bY04GlobalControlReq	Bit	y4
Indicator Y6 true	o_bY06DiagReadReq	Bit	y6
Indicator X0True	o_bX00DataExStartCompl	Bit	Profibus Network Card I/O X0
Indicator X4True	o_bX04GlobalControlCompl	Bit	Xn4
Indicator X5True	o_bX05GlobalControlError	Bit	Xn5
Indicator X1BTrue	o_bX1BCommReady	Bit	Communication Ready
Indicator X1FTrue	o_bX1FWatchDogError	Bit	Profibus Network Card I/O X1F
Cursor Control on First Node	o_bOnFirstNode	Bit	HMI on first Node for cursor control
Cursor Control on Last Node	o_bOnLastNode	Bit	HMI on Last node for cursor control
Host FDL Number	o_uHostNumber	Word[Unsigned]	FDL of Host Station
Selected Node is Deactivated	o_bActiveNodeInd	Bit	PB control
Latched Deactivated nodes	o_bnDeactivatedNodes	Bit(0..127)	Temporary Reserved Nodes

Name	Var_Output name	Data Type	Description
Array of Node Status	o_unNodeStatus	Word[Unsigned](0..127)	Profibus HMI Ind. Node With Diagnostic
Selected Error Code	o_uSelectedErrorCode	Word[Unsigned]	Node Status Array
Selected Statis	o_uSelectedStatus	Word[Unsigned]	Selected node Status
Module Error Code	o_uModuleErrorCode	Word[Unsigned]	Profibus HMI Module Fault Code
First Node Number	o_wFirstNodeNumber	Word[Signed]	First Node Number
Last Node Number	o_wLastNodeNumber	Word[Signed]	Last Node Number on Machine
Selected Node Manufacturer ID	o_uManufacturerID	Word[Unsigned]	Manufacturer ID
Channel Number	o_uChannelNumber	Word[Unsigned]	Channel number of selected Error
Channel Type	o_uChannelType	Word[Unsigned]	Channel Type of selected Error
Channel Data Format	o_uChannelDataFormat	Word[Unsigned]	channel Data format of Selected Error
Channel Error	o_uChannelError	Word[Unsigned]	Channel Error of Selected Error
Scan Time of Fieldbus	o_unLinkScanTime	Word[Unsigned](0..2)	Profibus Link Scan Time

FB Version Upgrade History	
Version	Description
1.01	Initial Release
1.02	Corrected addressing issue with FDL's
1.10	Replaced first scan bit with EN bit
1.12	Modified Screen Active; Designate Hex for Start_IO Modified FB variables to use the standard specified in BCN-89000-0823-D
1.13	Added global control function for sync
1.20	Added FB Status Output; Modified Logic for o_s_HMI_ProfiDiagFaultCode
2.00A	Adopted BCN-89000-0969 Implemented Intelligent Module Check, Added Superimpose Offset; Added Link Time Output
3.00	Rewrite, added channel diagnostics, reserved nodes, and link reboot logic

FB Version Upgrade History	
Version	Description
3.03	Changed Programming to reduce program size, added MC/MCR

SDT Usage

1. MELPT_APP_FieldbusDiag

System Label: ProfibusDiag		
Member	Type	Usage
HostOK	Bit	Profibus Card OK
Update	Bit	Set by HMI, requires refresh of channel diagnostic
ActivateDeactivate_PB	Bit	HMI PB to change activated status
ActiveNode_Ind	Bit	HMI Indicator of activation status
ResetStatus	Bit	Reset Status PB From HMI. Seta Status to zero
OnFirstNode	Bit	Set by Function block to limit HMI PB from decrementing too low
OnLastNode	Bit	Set by Function block to limit HMI PB from incrementing too high
WriteCheck	Bit(0..7)	
DeactivatedNodes	Bit(0..127)	Stored Values of Deactivated nodes
IMSupported	Bit(0..15)	
LocalNetworkNumber	Word[Unsigned]	
LocalStationNumber	Word[Unsigned]	FDL of Master
FieldbusModuleCommentNumber	Word[Unsigned]	
FieldbusModuleFault	Word[Unsigned]	Profinet Module Error

System Label: ProfibusDiag		
Member	Type	Usage
FirstConfiguredNode	Word[Signed]	Node Number of first configured node
LastConfiguredNode	Word[Signed]	Node Number of Last configured node
SelectedNode	Word[Signed]	HMI Selected node, this is FB limited by First and Last Configured Node
SelectedStatus	Word[Unsigned]	Status of Selected Node
SelectedDeviceDiagnostic	Word[Unsigned]	Error Array for Selected Node
ManufacturerID	Word[Unsigned]	Retrieved during acyclic requests manufacturer code
DeviceID	Word[Unsigned]	
ChannelStatus	Word[Unsigned]	
ChannelNumber	Word[Unsigned]	Retrieved during acyclic requests channel number under error
ChannelType	Word[Unsigned]	Retrieved during acyclic requests channel diagnostics channel type
ChannelDataFormat	Word[Unsigned]	Retrieved during acyclic requests channel diagnostics data format
ChannelError	Word[Unsigned]	Retrieved during acyclic requests channel error
ExtendedChannelError	Word[Unsigned]	
Slot	Word[Unsigned]	
SlotStatus	Word[Unsigned]	
SubSlot	Word[Unsigned]	
SubSlotStatus	Word[Unsigned]	
SlotIDent	Double Word[Unsigned]	

System Label: ProfibusDiag		
Member	Type	Usage
SubSlotIDent	Double Word[Unsigned]	
ChasisData	Word[Unsigned](0..15)	
MacAddress	Word[Unsigned](0..2)	
IPAddress	Word[Unsigned](0..1)	
SubnetMask	Word[Unsigned](0..1)	
GatewayAddress	Word[Unsigned](0..1)	
OrderNumber	Word[Unsigned](0..9)	
PlantID	Word[Unsigned](0..15)	
LocationName	Word[Unsigned](0..10)	
InstallationDate	Word[Unsigned](0..7)	
Description	Word[Unsigned](0..23)	
LinkTime	Word[Unsigned](0..2)	Profibus Minimum, maximum and current cycle time in milliseconds
NodeStatus	Word[Unsigned](0..127)	Status of all nodes determined by the FB

3.1.3 Necessary System Labels

The HMI project is common between Profibus and Profinet implementations of this project

ProfibusDiag(for Profibus diagnostics display)

ProfibusDiag(for Profibus diagnostics display)

GOTRFID (for manual RFID interface)

IOLinkDiag (For IO-link diagnostics features)

1 instance of IOlinkPressureSwitch for each pressure switch graphic needed.

A PLC device register is assigned to each system label, this prevents synching issues with the PLC project when additional programming and labels are added to the project.

3.2 IO-Link device configuration function blocks

Purpose: Provide System information and default parameter values to the PLC so device specific information can be displayed properly, and the device can be parameterized.

User Configuration: The function block must be in the initial or active scan, EN must be TRUE, Device Number must be unique and between 1-10, Diagnostic Graphic Number must be between 1 and 32767. A unique device description should be provided.

The Balluff IOL_CFG library contains multiple function blocks containing device information needed for proper program execution. There are function blocks for IOL-link devices for the following families.

Analog

IO Hubs

Positioning

Pressure

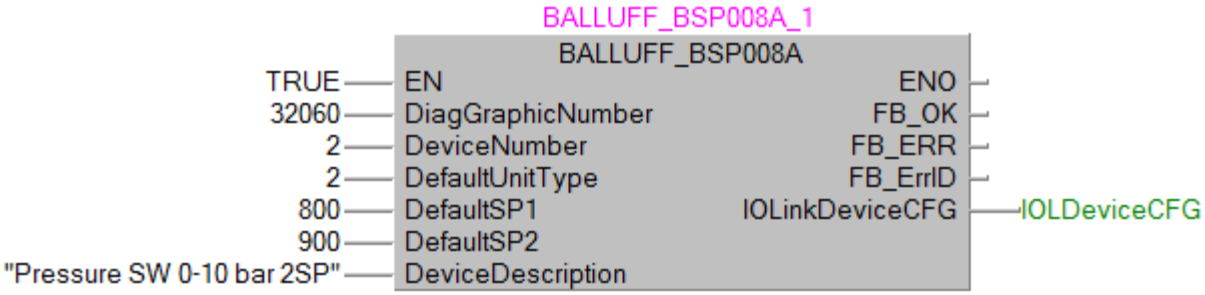
RFID FB

SmartLights

Valve Interface

Up to 10 devices total can be selected and placed into the initial scan of the PLC. These function blocks only have to be executed once the data contained within the function block is transferred to the data structure array IOLCFG

Configuration function block setup



3.2.1 FB Inputs

The EN pin must be set to true for the information to be available, set to false if the configuration is not desired.

DiagGraphicNumber

The HMI window number for this device, this is a value between 1 and 32767 corresponding to the diagnostic window displayed on the HMI for the specific device.

DeviceNumber

A value 1-10 that corresponds to the array element within IOLDeviceCFG, where this data will be stored. This number needs to be unique between all the configuration function blocks.

DefaultUnitType

This element only exists for Pressure Switches

Enter one of the following values:

0=bar

1=mbar

2=PSI

3=MPa

Default SP1 & Default SP2

The default set points when first configuring the device, this value is a raw device value

Device Description

This is the text that will be displayed on the HMI Maintenance Screen. There is a 24 character maximum.

3.2.2 FB Outputs

When the device number is outside the range of 1-10 FB_ERR is true
The ERRID will be 256(H100)

When the graphic number is outside the allowable range of 1 to 32767. FB_ERR is True.
The ERRID will be 257(H101)

For Pressure Switches an additional FB_ERR can occur when the value of DefaultUnitType is less than zero, or greater than three.
The ERRID will be 258(H102)

When ERRID is non-zero FB_ERR is TRUE.
When ERRID is zero FB_OK is TRUE.

The output of all CFG function blocks is IOLDeviceCFG which is array of CFG structures. Each individual block stores data in 1 element of this structure array.

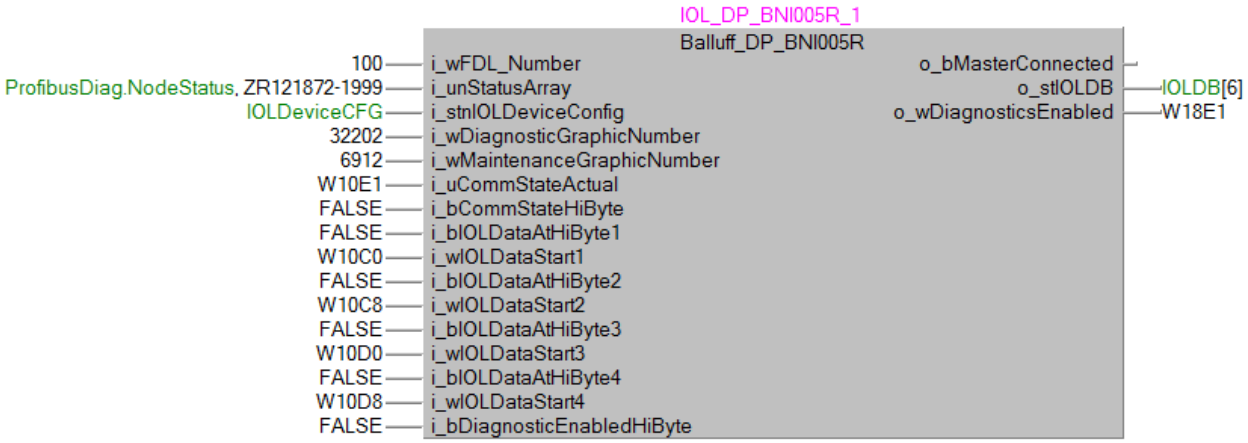
3.3 IO-Link Master Configuration

Three function blocks exist within the Balluff IOL_CFG library, an instance for each IO-link master on your PLC network needs to be added to the scan, the maximum number of IO-Link Masters is twenty.

The three function blocks included are:

- Balluff_DP_BIS012E
- Balluff_DP_BNI003P
- Balluff_DP_BNI005R

- **Purpose:** this function block determines how a master is configured, retrieves and organizes diagnostic data for the master also retrieves process data for any IO-link devices and stores appropriately.
- **User Configuration:** The user must properly populate the following pins:
    - FDL Number
    - Diagnostic Graphic Number
    - Maintenance Graphic Number
    - IOLDB element number
    - Diagnostics Enabled auto-refresh address
    - Starting address for Communication State
    - Starting Address for each IO-link Slot
    - Whether any element is started in the high byte of the word



3.3.1 FB Inputs

NodeNumber

This is a value between 0 and 126 and corresponds to the value set to the specific node in configurator DP

StatusArray

This is the node status word array that is output from MELPT_APP_ProfibusDiag

DiagnosticGraphicNumber

This is the HMI Window Number that displays the master diagnostics

Maintenance Graphic Number

This is the HMI window Number that displays the maintenance data

CommStateActual

Place the autorefresh address of the communication state slot here. This information can be retrieved from configurator-dp

			Slot	Model	Globa
100	IO_Link Master_1	BNI PBS-502-101-Z001	0	BNI PBS-502-101-Z001	-
			1	IOL_I/O_16/16 byte	vHA0SLV100MOD1
			2	IOL_I/O_16/16 byte	vHA0SLV100MOD2
			3	IOL_I/O_16/16 byte	vHA0SLV100MOD3
			4	IOL_I/O_16/16 byte	vHA0SLV100MOD4
			5	Output Pin 4	vHA0SLV100MOD5
			6	Output Pin 2	vHA0SLV100MOD6
			7	Input Pin 4	vHA0SLV100MOD7
			8	Input Pin 2	vHA0SLV100MOD8
			9	Communication State	vHA0SLV100MOD9
			10	Actuator shutdown Pin 4	vHA0SLV100MOD10
			11	Actuator shutdown Pin 2	vHA0SLV100MOD11
			12	Actuator warning Pin 4	vHA0SLV100MOD12
			13	Actuator warning Pin 2	vHA0SLV100MOD13
			14	Periphery error on Port	vHA0SLV100MOD14
			15	Sensor supply short circuit	vHA0SLV100MOD15
			16	Station diagnostic	vHA0SLV100MOD16
			17	IO-Link Diagnosis Enable/Disable	vHA0SLV100MOD17

IO_Link Master_1.Module Slot 9 : vHA0SLV100MOD9

Element Identifier	Element Type	Class	User MIT-Address	Global Var. Identifier	Buffer MIT-Address
inputs	ARRAY [0..7] OF BOOL	Input	-	-	W10E1.0 (%MX1.4321.0)

Since the slot configuration is fixed, the next 8 bytes are captured.

CommStateHiByte

If the data starts at WXXX.8 set this bool to TRUE

Programming inside the function block will offset appropriately.

IOLDataStart

Place the starting address for the IO-link slot. 4 bytes are captured for this location and stored within IOLDB

IOLdata at Hi Byte

If the data starts at WXXX.8 set this bool to TRUE

Programming inside the function block will offset appropriately.

3.3.2 FB Outputs

O_bMasterConnected

This value is True if the Master is detected and communication (function block processing)

IOLDB

Attach the IOLDB element where you want to store the settings for this master. The element number must be unique and a value between 1 and 20.

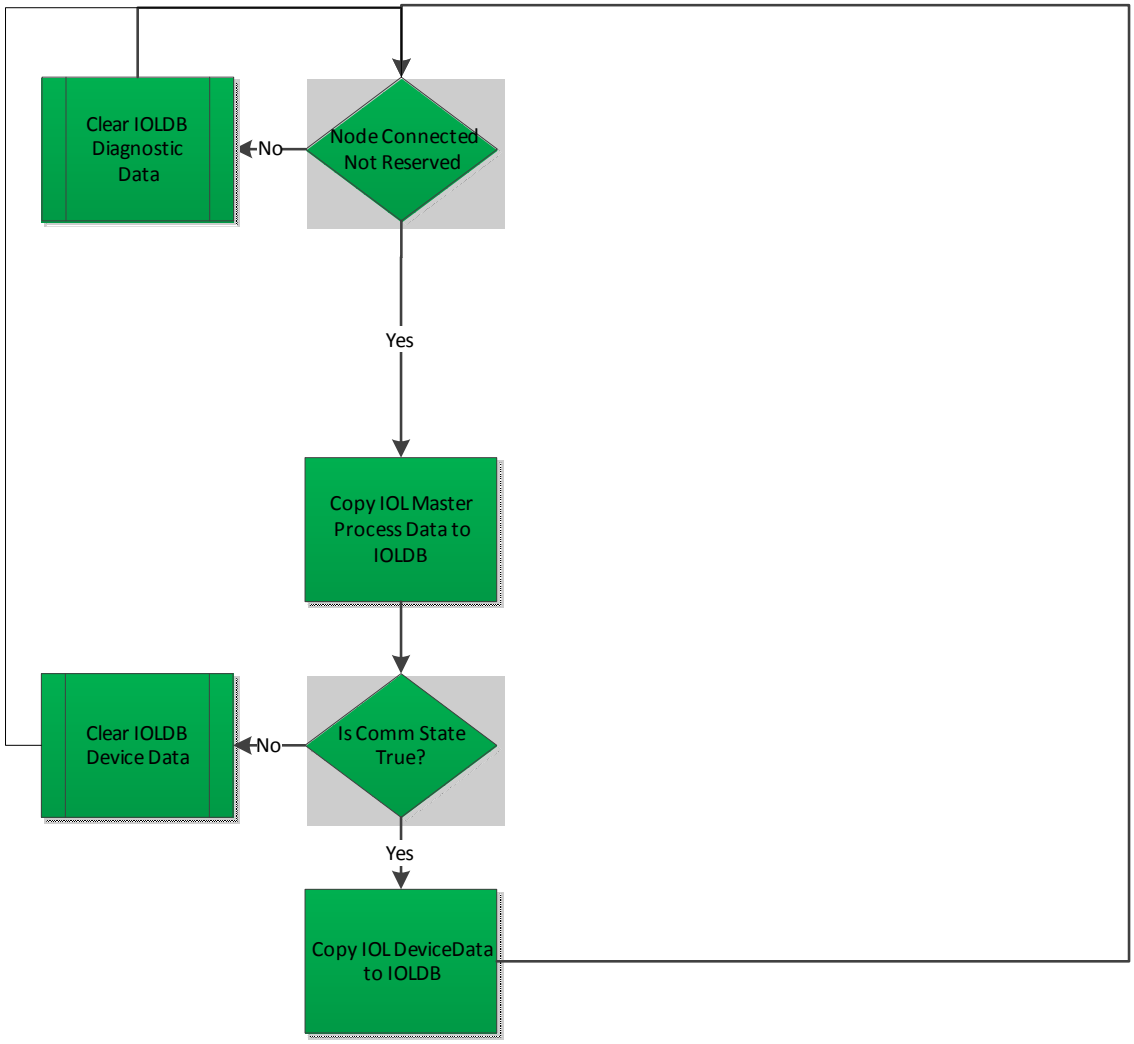
DiagnosticsEnabled

This is the output from IOLDB to the IO-link Master. This signal can be used to mask channel diagnostics on the fieldbus related to IO-link.

The Auto refresh address of the fieldbus must be attached to this pin. The auto refresh register can be obtained by following the instructions in 2.5

Note: If a diagnostic is occurring when the io-link diagnostic is disabled. An electrical change has to occur once it is enabled for the diagnostic to be reported to the fieldbus diagnostic programming.

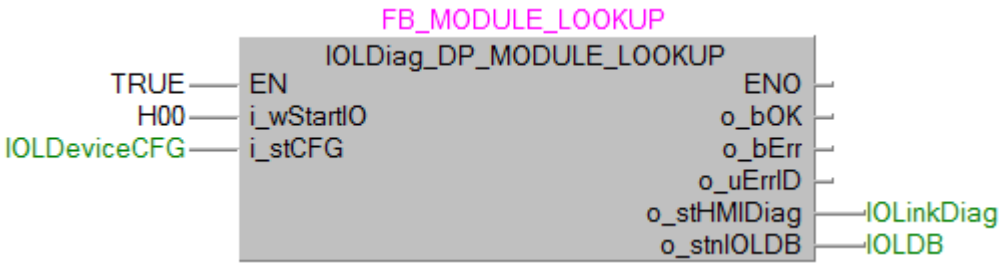
3.3.3 Program Flow



3.4 IO-Link Diagnostics required programming

Two function blocks are required to provide the functionality of this project that are not Device related.

3.4.1 IOLDiag_DP_module_Lookup



Purpose: Interrogates all elements of IOLDB, determines when a new IO-link connection is made, sends the stored parameters to the IO-Link Device.

User Configuration: Make sure the FB is scanned and required libraries are installed.

This function block is in charge of acyclic communication with an IO-Link device, and the writing of parameters to the device.

This function block will check the comm state for every IO-Link port. If communication has not been established the Manufacturer and Device ID are retrieved and compared to the available configurations if a match is made, then the configuration is checked for the same type. If a second match is made, parameters are written to the device if the configuration has them.

This function block requires the user library to be installed:

DP_accessory_v120

3.4.1.1 Inputs

Set the EN pin to TRUE

StartIO

This is hex address of the QJ71PB92V slot

CFG

The configuration data IOLDeviceCFG is connected to i_stCFg, this structure was populated in section 3.2 by the initial scan sequence

3.4.1.2 Outputs

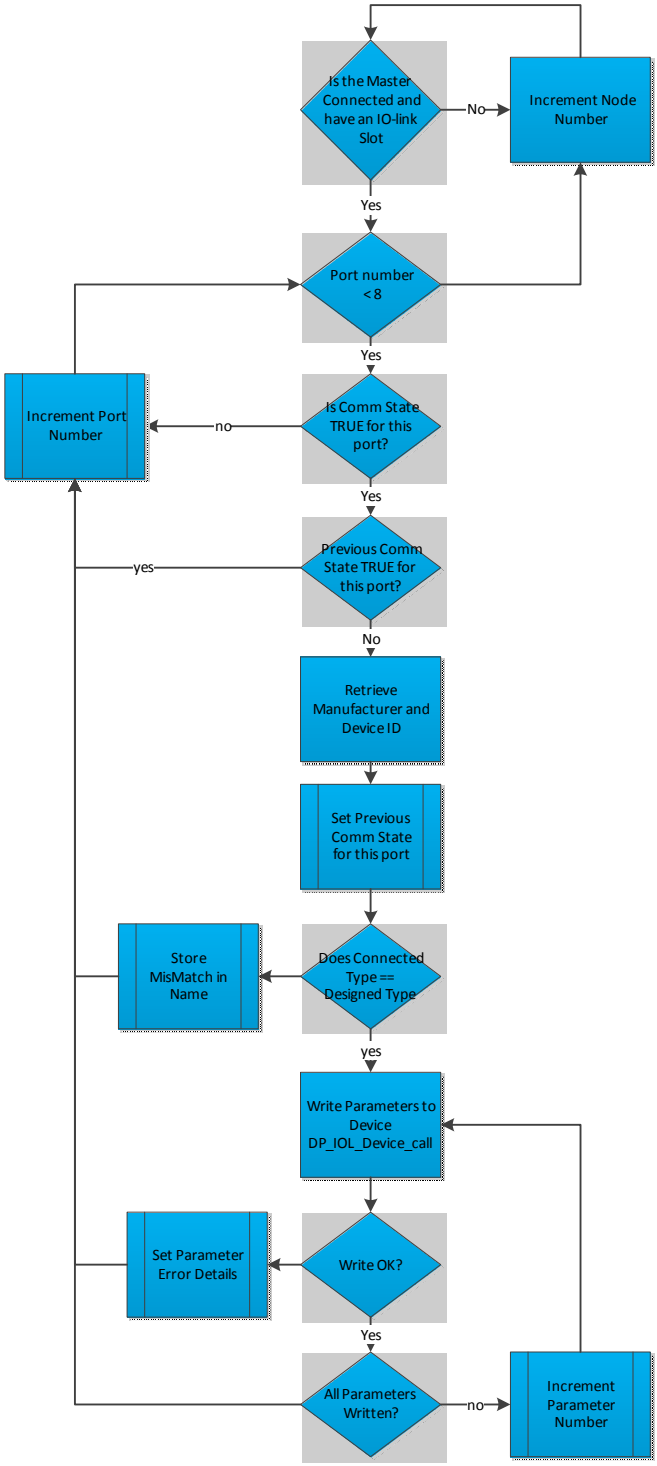
HMI Diag

The HMI System variables are retrieved and compared to the Label IOLinkDiag needs to be connected to the output HMI Diag

IOLDB

The Global Label IOLDB is attached to the output IOLDB

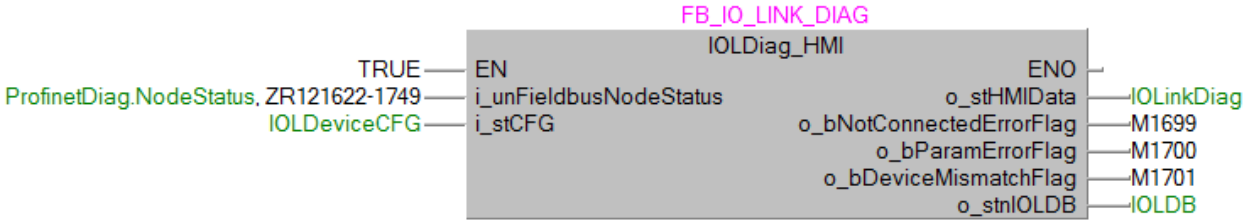
3.4.1.3 IOLDiag_DP_Module_Lookup FlowChart



3.4.2 IOLDiag_HMI

Purpose: This function blocks takes all the stored data from the master function blocks, determines error status and presents selected information to the HMI. In addition it accepts input from the HMI to modify the data structures.

User Configuration: Attach Flag Output Pins to the Alarming System of the HMI



3.4.2.1 FB Inputs

Set the EN pin to TRUE

CFG

The configuration data IOLDeviceCFG is connected to i_stCFg, this structure was populated in section 3.2 by the initial scan sequence

FieldbusNodeStatus

This is the node status word array that is output from MELPT_APP_ProfibusDiag

3.4.2.2 FB Outputs

HMIData

The data formatted to and from the HMI. IOLinkData.

IOLDB

The Global Label IOLDB is attached to the output IOLDB. This is a collection of all configured masters and if they are connected the diagnostic process data associated with them.

NotConnectedErrorFlag

This Boolean is connected to the HMI monitoring area for alarm messages to be displayed by the HMI alarm banner; It is true when a port is enabled, and configured but a device is not connected

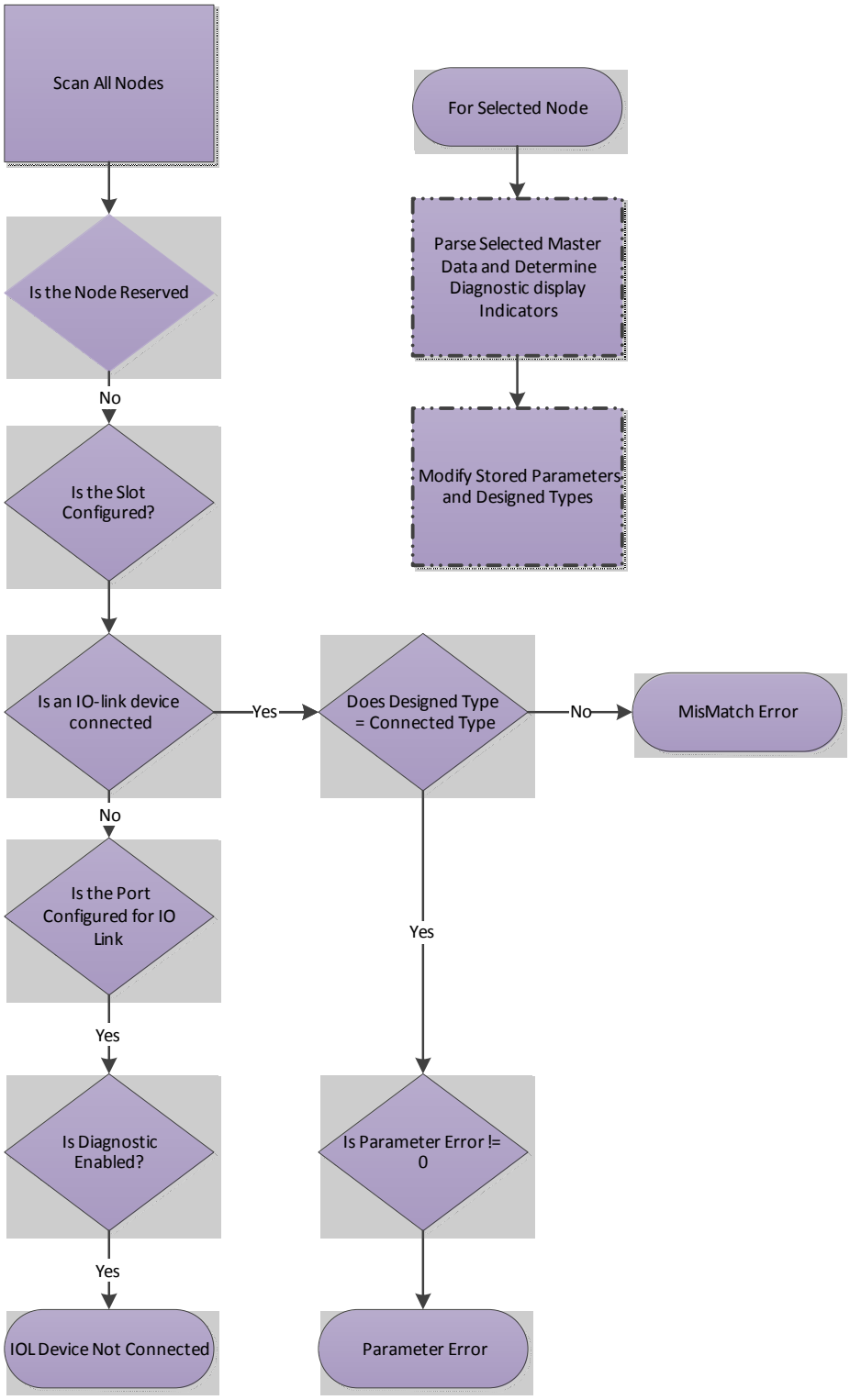
ParamErrorFlag

This Boolean is connected to the HMI monitoring area for alarm messages to be displayed by the HMI alarm banner; It is true when a port has an error message

DeviceMisMatchFlag

This Boolean is connected to the HMI monitoring area for alarm messages to be displayed by the HMI alarm banner. It is true when a connected device did not match the manufacturer and device ID configured on the HMI for that port.

3.4.2.3 IOLDiag_HMI FlowChart



3.5 IO-Link Device Process Data Function Blocks

There are eleven different types of process data function blocks available within the Balluff_IOL_Processdata

This section goes over how each type is used, and how to configure within your project.

3.5.1 Balluff Pressure Switch: Balluff_PSW_Processdata

Purpose: For each Balluff pressure switch used in a configuration, one instance of the **Balluff_PSW_Processdata** FB is recommended. For each instance of this FB, a new system label is required as well.

User Configuration:

Adding a new System label for each pressure switch that will be configured (Note: the target project must be part of a Melsoft Navigator project for the system label to function properly)

Setting the correct Port Number

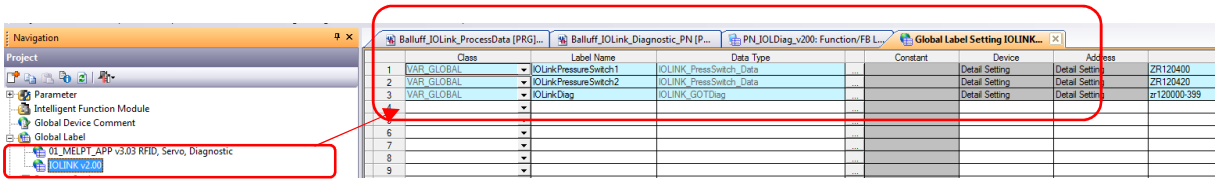
Setting the correct MasterData(IOLDB) Instance

Setting the warning percentage range

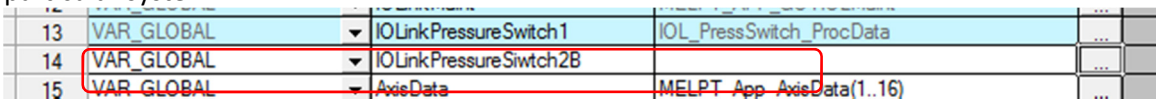
Tying flag outputs to the HMI alarming section

3.5.1.1 System Label

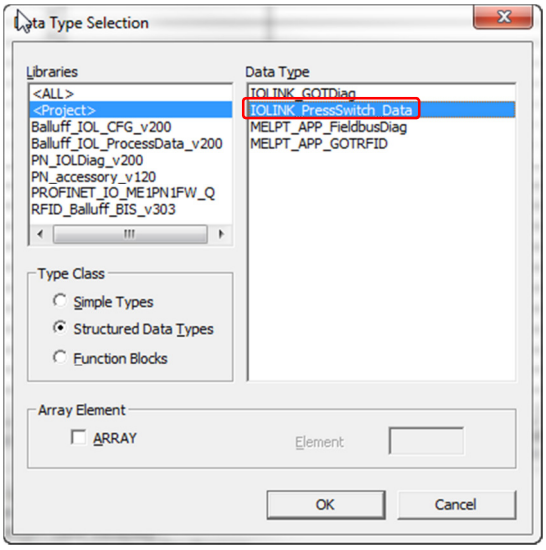
1. Open the **IO-Link** label group in the **Global Label** section of the target project:



2. On a blank row create a new tag with a **Class** of **VAR_GLOBAL** and any **Label Name** that fits your particular system:

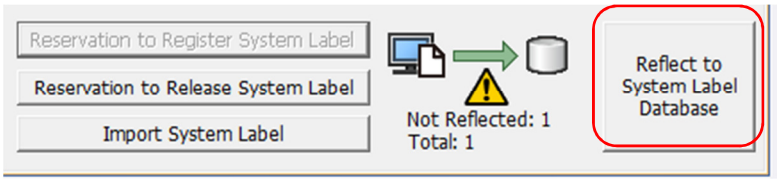


3. For the **Data Type**, click the **Ellipse** Button. On the **Data Type Selection** screen, switch the **Type Class** to **Structured Data Types** and select **IOLink_PressSwitch_Data** as the **Data Type**:

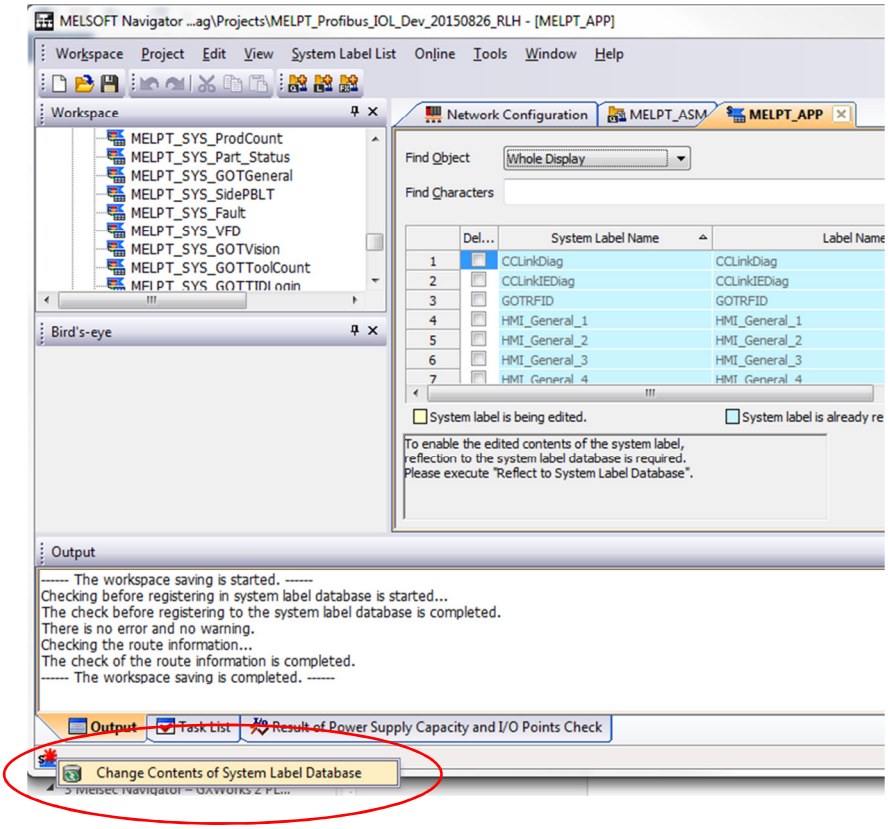


4. Click **OK**.
5. On the lower right of the Global Label list window, click the **Reservation to Register System Label** button:

6. The **Reflect to System Label Database** will enable. Click it to register the label to Melsoft navigator:



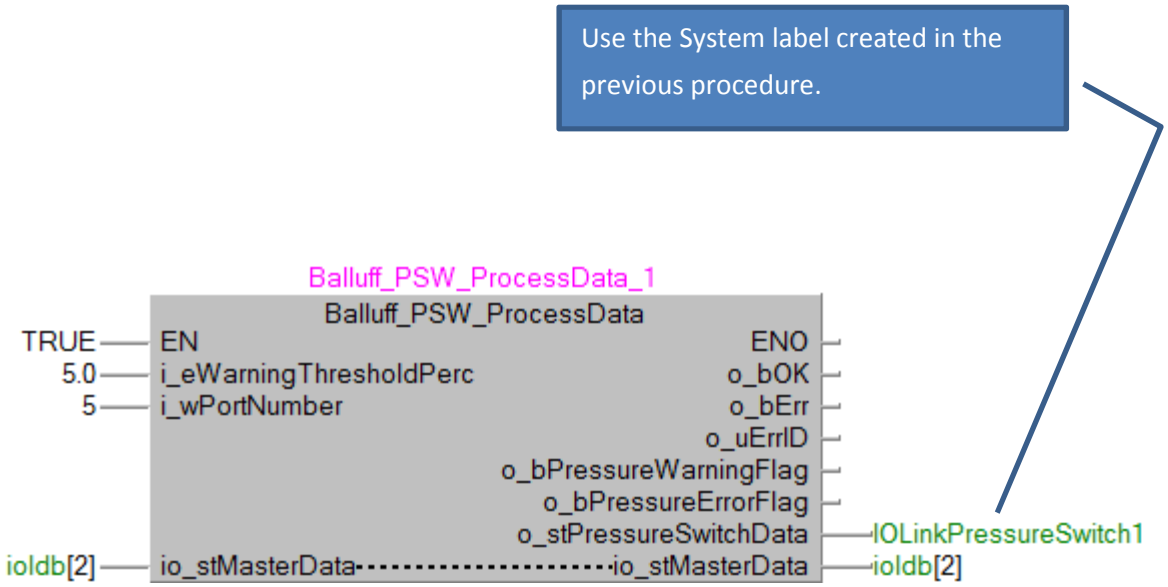
7. Click **Yes** when prompted.
8. Click the **Register** button on the **Check before registering in system label database** window
9. Open the Melsoft Navigator project for this PLC project.
10. On the lower left of the Navigator window, an icon will be flashing red.
11. Right click it and click the **Change Contents of System Label Database** that appears:



12. Click the **Import** button on the **Change Contents of System Label Database** window.
13. Open the GT Designer HMI project associate with this Navigator project.
14. Repeat steps 10-12 listed above in the HMI project.
15. Download the updated GT Designer 3 project to the HMI.

Adding a new Function Block for a pressure switch device:

1. Add an instance of the **Balluff_PSW_ProcessData** FB to the SCAN program. Configure these Input parameters and Output for **each** instance of the function block (see next page):



3.5.1.2 Inputs

Input Name	Input Parameter Setting
i_eWarningThresholdPerc	Setpoint 1 is subtracted from Setpoint 2. This value is multiplied by the input percentage. The result is added to Set point 1, this creates the lower warning range. The result is also subtracted from setpoint2, this creates the upper warning range.
I_wPortNumber	This is a number 0-7 corresponding to the port the IO-link device is connected to on the master.
io_stMasterData	The data block for the master where this particular device is connected. This function block adjusts the return point parameters to one less than the set points

3.5.1.3 Outputs

Output Name	Output Description/Setting
o_bOK	This is True when the function block executes normally
o_bErr	This is True when the function block aborted execution due to error

o_uErrID	This is the Error code, when the function block aborted execution 256(0x100) Invalid Port Number 257(0x101) Invalid Device Type Detected 258(0x102) Device Mismatch 259(0x103) Device Not Connected
o_bPressureWarningFlag	When the Actual Value is within one of the warning ranges this output is true. It should be tied to the HMI alarming
o_bPressureErrorFlag	When the Actual Value is below Setpoint 1, or above SetPoint 2, this output is true. It should be tied to the HMI alarming.
o_stPressureSwitchData	This data structure contains all data for the HMI pertaining to this device

3.5.2 Balluff_Smartlight_ProcessData

Purpose: For each SmartLight used in a configuration, one instance of the **Balluff_SmartLight_Processdata** FB is recommended. For each instance of this FB, a new control label structure instance will be needed.

User Configuration:

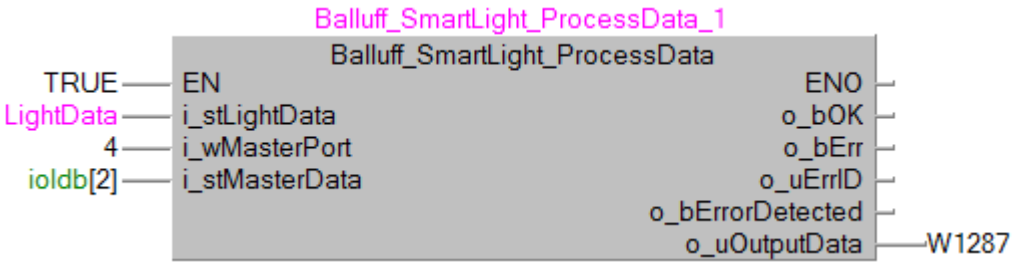
Setting the correct Port Number

Setting the correct MasterData(IOLDB) Instance

Controlling the Data for the LightData Structure

Tying the Error Output to the HMI alarming system.

Tying the Output Data to the correct auto-refresh register.



3.5.2.1 Inputs

Input Name	Input Parameter Setting
i_stLightData	Data Structure used to control the output to the light
I_wMasterPort	This is a number 0-7 corresponding to the port the IO-link device is connected to on the master.
i_stMasterData	The data block for the master where this particular device is connected.

3.5.2.2 Outputs

Output Name	Output Description/Setting
o_bOK	This is True when the function block executes normally

o_bErr	This is True when the function block aborted execution due to error
o_uErrID	This is the Error code, when the function block aborted execution 256(0x100) Invalid Port Number 257(0x101) Invalid Device Type Detected 258(0x102) Invalid Mode Selected (more than one mode or no mode is selected)
o_bErrorDetected	The input data has an error code.
o_uOutputData	This is the starting register for the output telegram to the smartlight.

3.5.2.3 IOL_SmartLight_Data

1. IOL_SmartLight_Data

Label: LightData		
Member	Type	Usage
SegmentMode_Enabled	BOOL	True to Enable Segment Mode
SegMode_Segment1Color	WORD	0=off 1=green 2=red 3=yellow 4=blue 5=orange 6=user defined by parameter 7=white
SegMode_Segment1Blink	BOOL	TRUE=Blink
SegMode_Segment1Blink50	BOOL	When Blink ON Blink 50%

Label: LightData		
Member	Type	Usage
SegMode_Segment2Color	WORD	0=off 1=green 2=red 3=yellow 4=blue 5=orange 6=user defined by parameter 7=white
SegMode_Segment2Blink	BOOL	TRUE=Blink
SegMode_Segment2Blink50	BOOL	When Blink ON Blink 50%
SegMode_Segment3Color	WORD	0=off 1=green 2=red 3=yellow 4=blue 5=orange 6=user defined by parameter 7=white
SegMode_Segment3Blink	BOOL	TRUE=Blink
SegMode_Segment3Blink50	BOOL	When Blink ON Blink 50%
SegMode_Segment4Color	WORD	0=off 1=green 2=red 3=yellow 4=blue 5=orange 6=user defined by parameter 7=white

Label: LightData		
Member	Type	Usage
SegMode_Segment4Blink	BOOL	TRUE=Blink
SegMode_Segment4Blink50	BOOL	When Blink ON Blink 50%
SegMode_Segment5Color	WORD	0=off 1=green 2=red 3=yellow 4=blue 5=orange 6=user defined by parameter 7=white
SegMode_Segment5Blink	BOOL	TRUE=Blink
SegMode_Segment5Blink50	BOOL	When Blink ON Blink 50%
SegMode_NumberOfSegments	WORD	Number of Segments
SegMode_BlinkingFrequency	WORD	1=0.5hz 2=1hz 3=2HZ 4=5hz 5=10hz
LevelMode_Enabled	BOOL	True to Enable Level Mode

Label: LightData		
Member	Type	Usage
LvlMode_Segment1Color	WORD	0=off 1=green 2=red 3=yellow 4=blue 5=orange 6=user defined by parameter 7=white
LvlMode_Segment1Dominance	BOOL	When true lower segments take this color
LvlMode_Segment2Color	WORD	0=off 1=green 2=red 3=yellow 4=blue 5=orange 6=user defined by parameter 7=white
LvlMode_Segment2Dominance	BOOL	When true lower segments take this color
LvlMode_Segment3Color	WORD	0=off 1=green 2=red 3=yellow 4=blue 5=orange 6=user defined by parameter 7=white

Label: LightData		
Member	Type	Usage
LvlMode_Segment3Dominance	BOOL	When true lower segments take this color
LvlMode_Segment4Color	WORD	0=off 1=green 2=red 3=yellow 4=blue 5=orange 6=user defined by parameter 7=white
LvlMode_Segment4Dominance	BOOL	When true lower segments take this color
LvlMode_Segment5Color	WORD	0=off 1=green 2=red 3=yellow 4=blue 5=orange 6=user defined by parameter 7=white
LvlMode_Segment5Dominance	BOOL	When true lower segments take this color
LvlMode_LevelType	BOOL	True = Top Down
LvlMode_LevelValue	WORD	Value between 0-255
RunLight_Enabled	BOOL	TRue to Enable Run Light Mode

Label: LightData		
Member	Type	Usage
RunLghtMode_BackGroundColor	WORD	0=off 1=green 2=red 3=yellow 4=blue 5=orange 6=user defined by parameter 7=white
RunLghtMode_RunningColor	WORD	0=off 1=green 2=red 3=yellow 4=blue 5=orange 6=user defined by parameter 7=white
RunLghtMode_NumberOfSegments	WORD	Value 1-5
RunLghtMode_RunningSpeed	WORD	Value 1-5
BuzzerState	BOOL	Buzzer On
BuzzerType	WORD	0=Continous 1 1hz chopped 5 hz chopped 3=3 short beep with 2 sec pause

3.5.3 Balluff_BNI0041_ProcessData

Purpose:

This function block scales the input from the device and will output a value 4.0ms-20ma

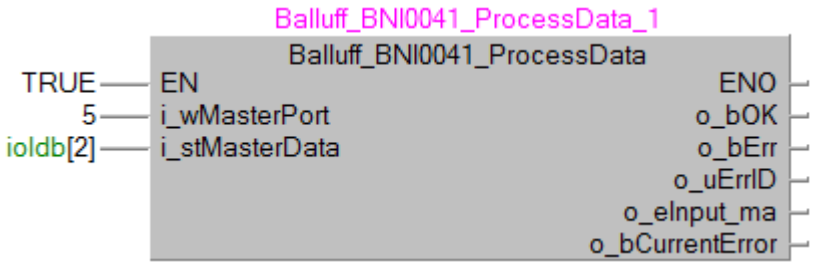
For each Balluff Analog current input(BNI0041) used in a configuration, one instance of the **Balluff_BNI0041_Processdata** FB is recommended.

User Configuration:

Setting the correct Port Number

Setting the correct MasterData(IOLDB) Instance

Tying Current Error outputs to the HMI alarm monitoring



3.5.3.1 Inputs

Input Name	Input Parameter Setting
EN	Must be True for the function block to be executed
I_wMasterPort	This is a number 0-7 corresponding to the port the IO-link device is connected to on the master.
i_stMasterData	The data block for the master where this particular device is connected.

3.5.3.2 Outputs

Output Name	Output Description/Setting
o_bOK	This is True when the function block executes normally

o_bErr	This is True when the function block aborted execution due to error
o_uErrID	This is the Error code, when the function block aborted execution 256(0x100) Invalid Port Number 257(0x101) Device Mismatch 258(0x102) Device Not Connected
o_bErrorDetected	If the input to the device is below 4ma, the device will set an error output, that bool is reported here.
o_eInput_ma	The input data is scaled, a Value of 4.0 to 20.0 is output from this function block, if an error is occurring the output is 0.0

3.5.4 Balluff_BNI0042_ProcessData

Purpose:

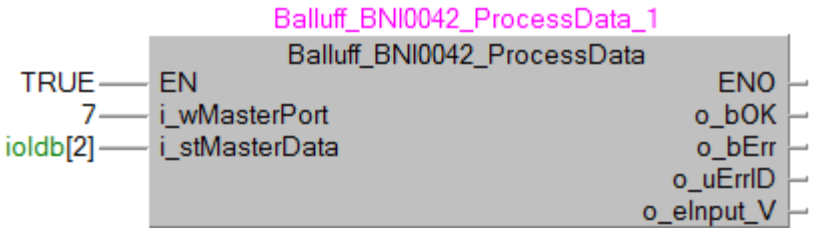
This function block scales the input from the device and will output a value 4.0ms-20ma

For each Balluff Analog Voltage input used in a configuration, one instance of the **Balluff_BNI0042_Processdata** FB is recommended.

User Configuration:

Setting the correct Port Number

Setting the correct MasterData(IOLDB) Instance



3.5.4.1 Inputs

Input Name	Input Parameter Setting
EN	Must be True for the function block to be executed
I_wMasterPort	This is a number 0-7 corresponding to the port the IO-link device is connected to on the master.
i_stMasterData	The data block for the master where this particular device is connected.

3.5.4.2 Outputs

Output Name	Output Description/Setting
o_bOK	This is True when the function block executes normally
o_bErr	This is True when the function block aborted execution due to error

o_uErrID	This is the Error code, when the function block aborted execution 256(0x100) Invalid Port Number 257(0x101) Device Mismatch 258(0x102) Device Not Connected
o_eInput_V	The input value is scaled and the function block will output a value between 0.0 and 10.0V

3.5.5 Balluff_AnalogOut_ProcessData

Purpose:

This function block receives a user input between 0.0 and 100.0 the value is scaled and output to the device where the voltage or current is output.

For each Balluff Analog Output BNI004C or BNI004E used in a configuration, one instance of the **Balluff_AnalogOut_Processdata** FB is recommended.

User Configuration:

Setting the correct Port Number

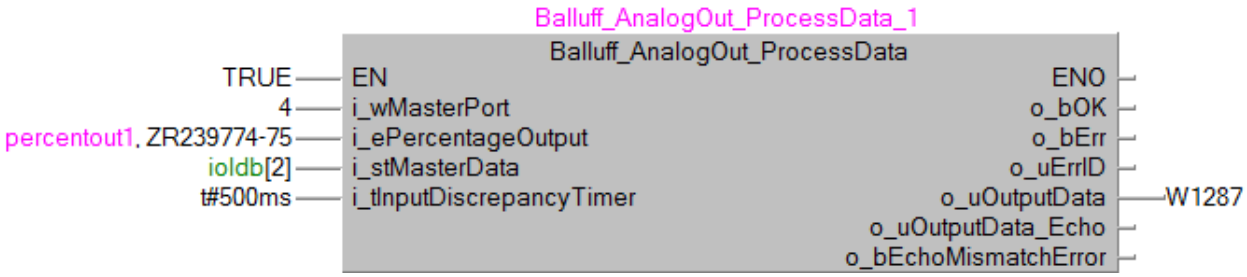
Setting the correct MasterData(IOLDB) Instance

Set a discrepancy timer value

Control the Percentage Output Value

Tie the auto-refresh register to the OutputData

Tie the EchoMismatchError to the HMI alarming



3.5.5.1 Inputs

Input Name	Input Parameter Setting
EN	Must be True for the function block to be executed
I_wMasterPort	This is a number 0-7 corresponding to the port the IO-link device is connected to on the master.
I_ePercentageOutput	A value between 0.0 and 100.0 corresponding to the percentage to be output.

I_stMasterData	The data block for the master where this particular device is connected.
I_tInputDiscrepencyTimer	The device echo's the value being output, internally the two are compared, if they do not match for the time input length, the echo mismatch error is TRUE

3.5.5.2 Outputs

Output Name	Output Description/Setting
o_bOK	This is True when the function block executes normally
o_bErr	This is True when the function block aborted execution due to error
o_uErrID	This is the Error code, when the function block aborted execution 256(0x100) Invalid Port Number 257(0x101) Device Mismatch 258(0x102) Device Not Connected
o_uOutputData	The percentage scaled and formatted for the device. Connect the auto-refresh register to this pin
o_uOutputDataEcho	The Data returned from the device
o_bEchoMisMatchError	When the Echo does not match the output data for the input discrepancy time.

3.5.6 Balluff_Ultrasonic_ProcessData

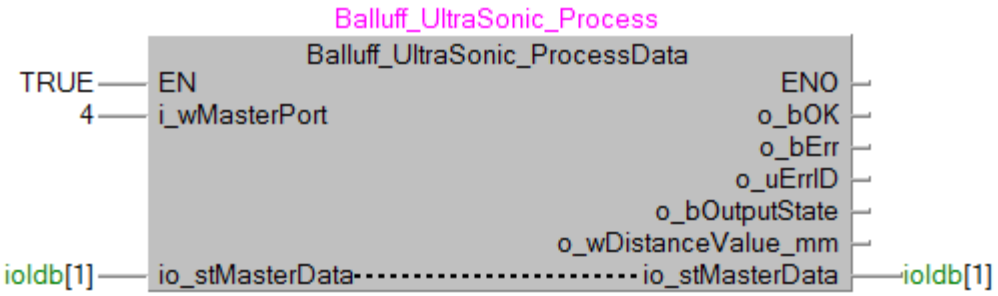
Purpose:

For each Balluff ultrasonic used in a configuration, one instance of the **Balluff_UltraSonic_Processdata** FB is recommended.

User Configuration:

Setting the correct Port Number

Setting the correct MasterData(IOLDB) Instance



3.5.6.1 Inputs

Input Name	Input Parameter Setting
EN	Must be True for the function block to be executed
I_wMasterPort	This is a number 0-7 corresponding to the port the IO-link device is connected to on the master.
io_stMasterData	The data block for the master where this particular device is connected. This function block will adjust the parameters to be within certain limits of the set points.

3.5.6.2 Outputs

Output Name	Output Description/Setting
o_bOK	This is True when the function block executes normally
o_bErr	This is True when the function block aborted execution due to error

o_uErrID	This is the Error code, when the function block aborted execution 256(0x100) Invalid Port Number 257(0x101) Device Mismatch 258(0x102) Device Not Connected
o_bOutputState	The Boolean State of the Switched Output
o_wDistanceValue_mm	The Data returned from the device scaled to mm

3.5.7 Balluff_BAW002F_ProcessData

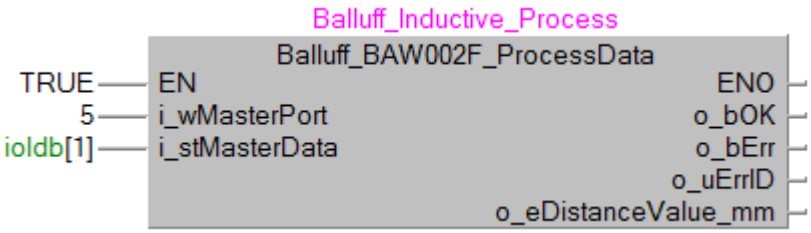
Purpose:

For each Balluff BAW002F sensor used in a configuration, one instance of the **Balluff_BAW002F_Processdata** FB is recommended.

User Configuration:

Setting the correct Port Number

Setting the correct MasterData(IOLDB) Instance



3.5.7.1 Inputs

Input Name	Input Parameter Setting
EN	Must be True for the function block to be executed
I_wMasterPort	This is a number 0-7 corresponding to the port the IO-link device is connected to on the master.
i_stMasterData	The data block for the master where this particular device is connected. This function block will adjust the parameters to be within certain limits of the set points.

3.5.7.2 Outputs

Output Name	Output Description/Setting
o_bOK	This is True when the function block executes normally
o_bErr	This is True when the function block aborted execution due to error

o_uErrID	This is the Error code, when the function block aborted execution 256(0x100) Invalid Port Number 257(0x101) Device Mismatch
o_eDistanceValue_mm	The Data returned from the device scaled between 1.0 and 5.0 mm

3.5.8 Balluff_BIP0004_ProcessData

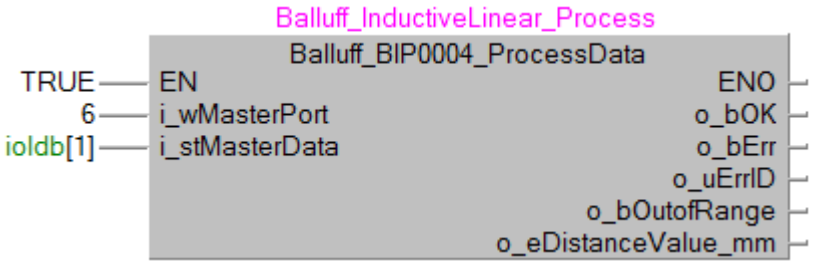
Purpose:

For each Balluff BIP0004 sensor used in a configuration, one instance of the **Balluff_BIP0004_Processdata** FB is recommended.

User Configuration:

Setting the correct Port Number

Setting the correct MasterData(IOLDB) Instance



3.5.8.1 Inputs

Input Name	Input Parameter Setting
EN	Must be True for the function block to be executed
I_wMasterPort	This is a number 0-7 corresponding to the port the IO-link device is connected to on the master.
i_stMasterData	The data block for the master where this particular device is connected. This function block will adjust the parameters to be within certain limits of the set points.

3.5.8.2 Outputs

Output Name	Output Description/Setting
o_bOK	This is True when the function block executes normally
o_bErr	This is True when the function block aborted execution due to error

o_uErrID	This is the Error code, when the function block aborted execution 256(0x100) Invalid Port Number 257(0x101) Device Mismatch
o_eDistanceValue_mm	The Data returned from the device scaled between 0.0 and 40.0 mm

3.5.9 Balluff_BOD0012_ProcessData

Purpose:

For each Balluff BOD0012 sensor used in a configuration, one instance of the **Balluff_BOD0012_Processdata** FB is recommended.

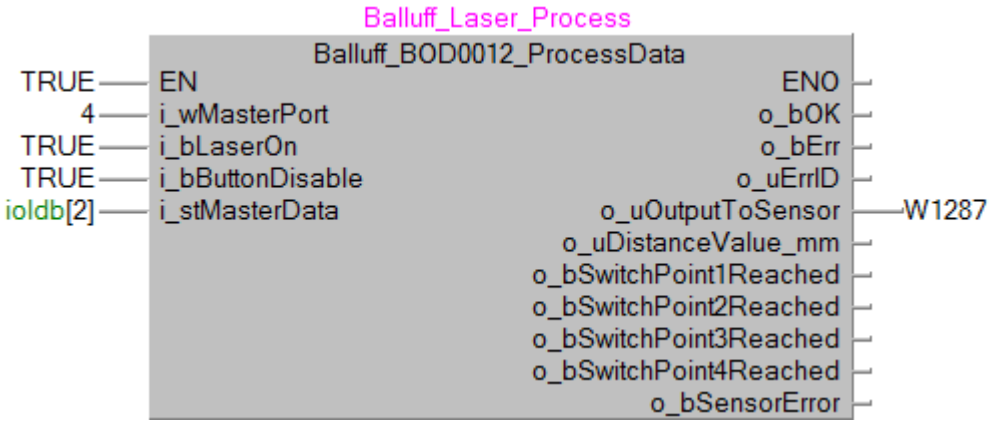
User Configuration:

Setting the correct Port Number

Setting the correct MasterData(IOLDB) Instance

Setting the output to sensor auto-refresh register

Turn in on the Laser to retrieve result



3.5.9.1 Inputs

Input Name	Input Parameter Setting
EN	Must be True for the function block to be executed
I_wMasterPort	This is a number 0-7 corresponding to the port the IO-link device is connected to on the master.
I_bLaserOn	Turns on Measurement
I_bButtonDisable	Disables user modification via sensor pushbutton

i_stMasterData	The data block for the master where this particular device is connected. This function block will adjust the parameters to be within certain limits of the set points.

3.5.9.2 Outputs

Output Name	Output Description/Setting
o_bOK	This is True when the function block executes normally
o_bErr	This is True when the function block aborted execution due to error
o_uErrID	This is the Error code, when the function block aborted execution 256(0x100) Invalid Port Number 257(0x101) Device Mismatch
O_uOutputToSensor	This is the output to the auto refresh register of the io-link device
o_uDistanceValue_mm	The Data returned from the device scaled between 0 and 6000
o_bSwitchPoint1Reached	Output when the distance exceeds switch point 1 parameter
o_bSwitchPoint2Reached	Output when the distance exceeds switch point 2 parameter
o_bSwitchPoint3Reached	Output when the distance exceeds switch point 3 parameter
o_bSwitchPoint4Reached	Output when the distance exceeds switch point 4 parameter
o_bSensorError	Output when the sensor reports error

3.5.10 Balluff_Valve16_ProcessData

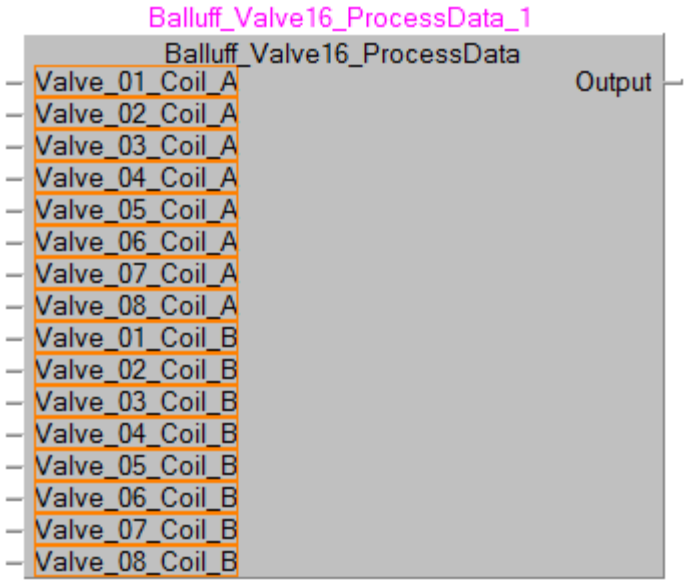
Purpose:

For each 16 Valve Interface used in a configuration, one instance of the **Balluff_Valve16_Processdata** FB is recommended.

User Configuration:

Setting the output to valve interface auto-refresh register

Setting control bits for the individual valves



3.5.10.1 Inputs

Input Name	Input Parameter Setting
Valve_XX_Coil_A/B	Boolean Condition to output to an individual valve

3.5.10.2 Outputs

Output Name	Output Description/Setting
Output	This is the output to the auto refresh register of the io-link device

3.5.11Balluff_Valve24_ProcessData

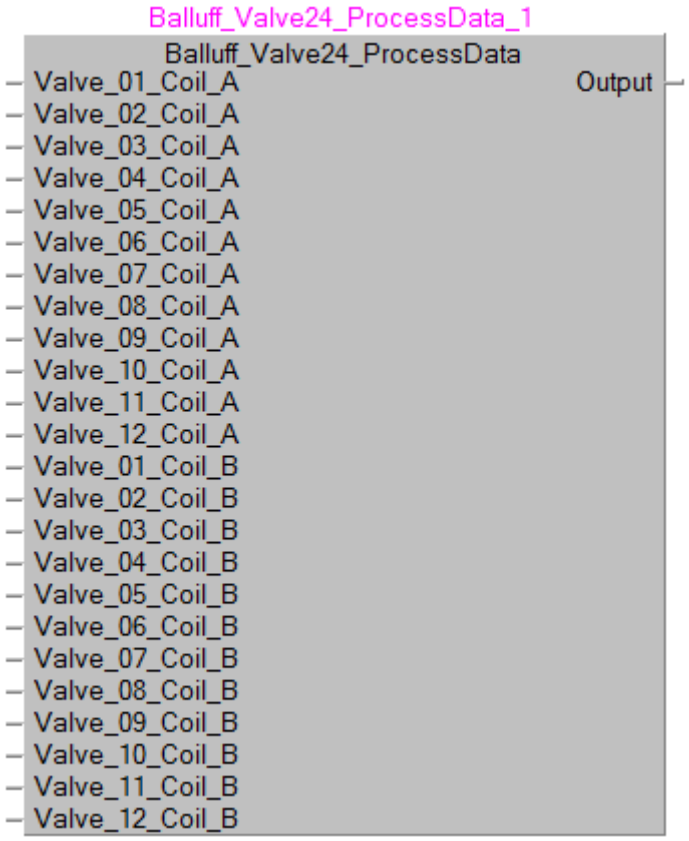
Purpose:

For each 24 Valve Interface used in a configuration, one instance of the **Balluff_Valve24_Processdata** FB is recommended.

User Configuration:

Setting the output to valve interface auto-refresh register

Setting control bits for the individual valves



3.5.11.1 Inputs

Input Name	Input Parameter Setting
Valve_XX_Coil_A/B	Boolean Condition to output to an individual valve

3.5.11.2 Outputs

Output Name	Output Description/Setting
Output	This is the output to the auto refresh register of the io-link device

3.6 RFIDIF_Balluff_IOLink_256b / RFIDIF_BalluffBIS_V_M_2K

Purpose:

IO-link antennas can be used with the RFIDIF_Balluff_IOLink_256b function block. This function block allows for a continuous read/write of up to 256 bytes for one antenna.

When using the Balluff BIS-V or –M system.

You use the following function block:

RFIDIF_BalluffBIS_V_M_2K

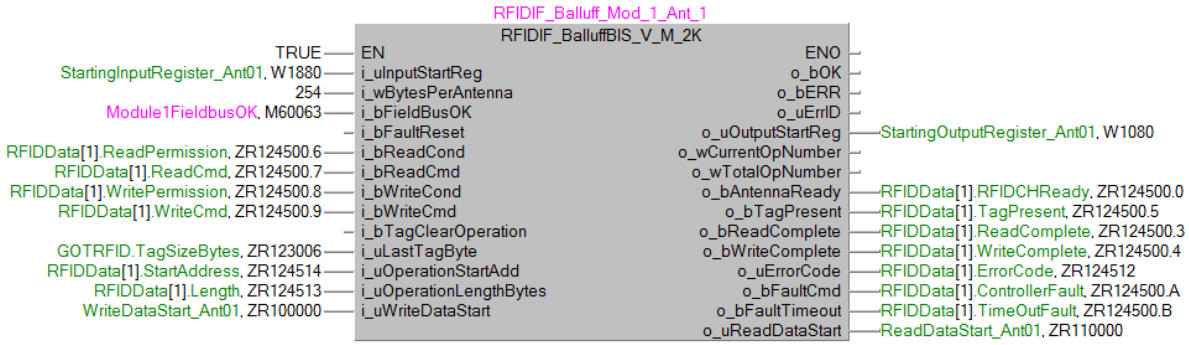
This function block allows for a continuous read/write of up to 2048 bytes for one antenna.

User Configuration:

The user must place the correct Starting Input and Output auto-refresh registers

The user must configure the proper number of bytes per antenna (10 or 32 for IO-Link) Slot limited for BIS-V/M

RFID data is placed into a data structure with up to 16 elements. There is also a GOT RFID FB that control manual signals for this function block.

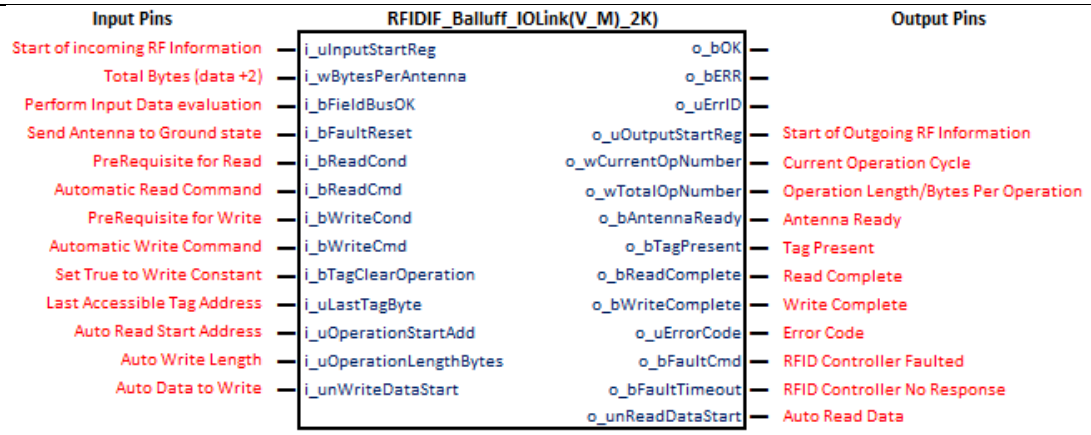


The RFID_Balluff User library contains programs, function blocks and SDT for use with the Balluff BIS-V, Balluff BIS-M, or Balluff IO-link RFID

If you are using the Balluff BIS-M RFID system the maximum size of an operation is 2000 bytes.

If you are using an IO-link antenna RFIDIF_Balluff_IOLink_256b must be used.

FB Name
RFIDIF_BalluffBIS_V_M_2K

Item	Description
Function overview	This Function block handles manual and automatic RFID functions for 1 antenna of the system. Commands supported are: Read, Write, and Write Constant Value.
Symbol	
Applicable Hardware	

Series	Model	Serial Restriction
MELSEC-Q series	Universal Model PLC	None
GOT 1000 series	GT16(800*600) or Higher	None
Melsec Q Series	Applicable Fieldbus Module	
Balluff BIS-V RF Controller	BIS012E, BIS013W	
Balluff BIS-M RF Controller	BIS00L7, BIS00EW	
IO-link Antenna	BIS00LH	FW 1.8

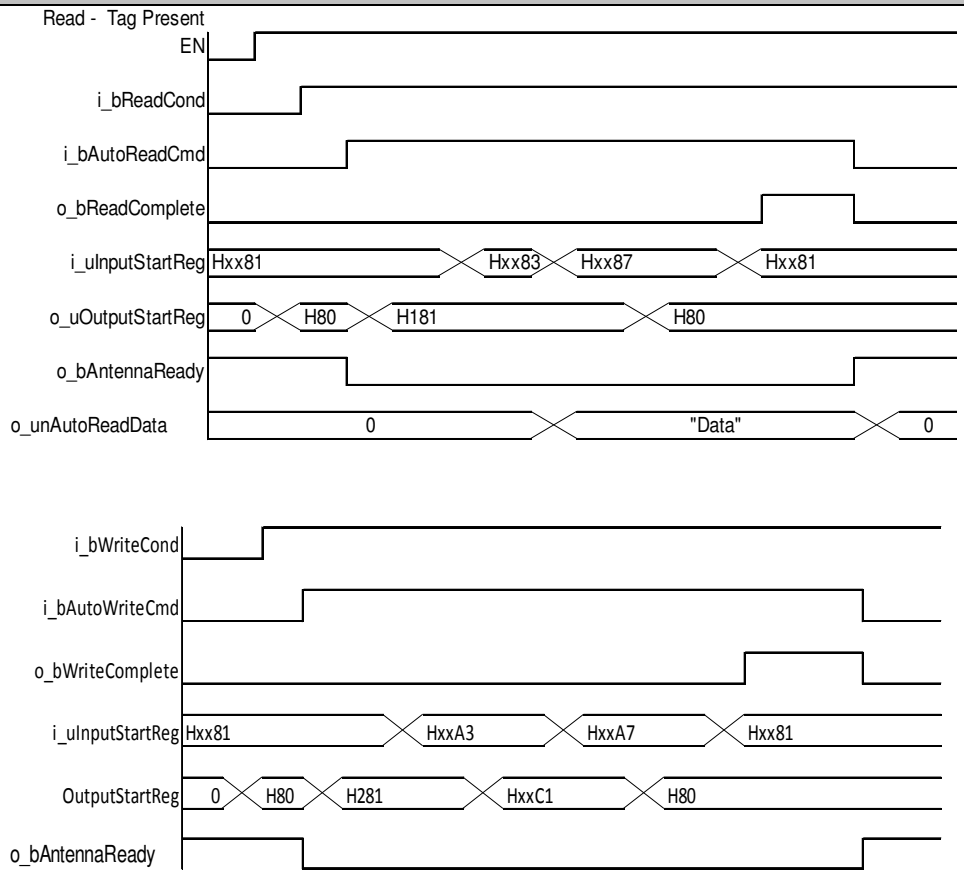
<table>
<tr><th>Item</th><th>Description</th></tr>
<tr><td>Applicable
Software</td><td>
<table>
<tr><th>Series</th><th>Version</th></tr>
<tr><td>GX Works 2</td><td>1.540</td></tr>
<tr><td>GT Designer 3</td><td>1.136</td></tr>
<tr><td>Fieldbus Configurator</td><td></td></tr>
</table>
</td></tr>
<tr><td>Programming language</td><td>Structured Ladder/FBD</td></tr>
<tr><td>Number of Ladder Steps</td><td>QnU: 582 steps BIS_V_M

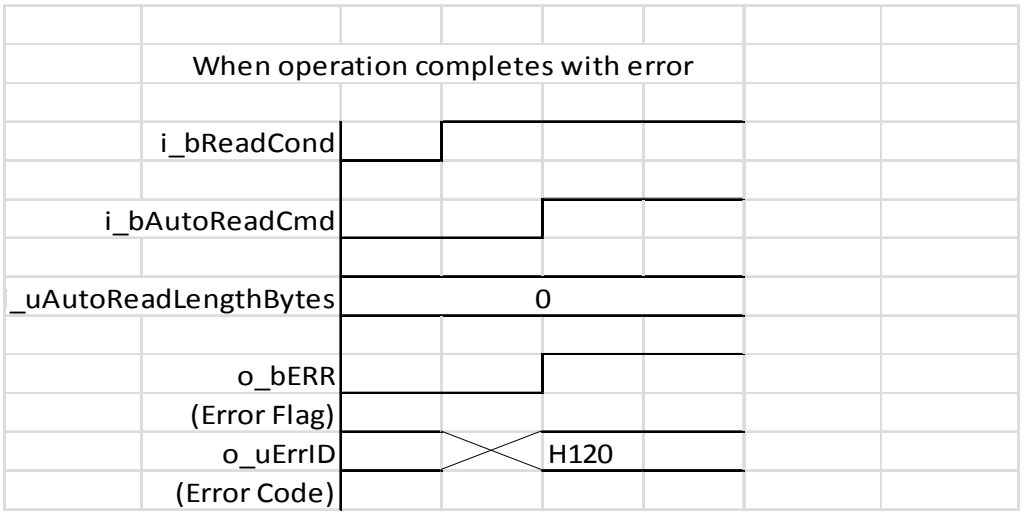
QnU: 691 steps: IO-link

*The number of steps of the FB program depends on the CPU Model that is used and input and output definition</td></tr>
<tr><td>Device Memory Used</td><td></td></tr>
<tr><td>Compiling method</td><td>Macro type;</td></tr>
<tr><td>Execution type</td><td>Real-time Execution</td></tr>
<tr><td>Dependences</td><td>The RFID controller needs to have 2-byte headers enabled; BIS-M must have synchronization enabled.</td></tr>
<tr><td>Function description</td><td>This function blocks separates status data from the antenna interrogated and places into HMI labels.

It also moves HMI input data to the manual related members of the RFID data structure of the selected antenna.</td></tr>
</table>

Item	Description
Restrictions and precautions	

Item	Description
Timing chart	



Item	Description

FB Error Code

Error Code	Description
0	No Error.
H100(256)	Bytes Per Antenna is less than 2
H101(257)	Bytes per antenna > 1024/2048
H110(272)	Manual Length = 0
H111(273)	Manual Length Greater than 16
H112(274)	Manual Operation Exceeds Tag Capacity
H120(288)	Auto Read Length = 0
H121(289)	Auto Read Length >1024/2048
H122(290)	Read Operation Exceeds Tag Capacity
H130(304)	Auto Write Length = 0
H131(305)	Auto Write Length >1024/2048
H132(306)	Write Operation Exceeds Tag Capacity
H140(320)	Fieldbus OK not true

Labels

■nput labels

User Input	Symbol Name	Var_Input name	Data Type	Setting range	Description

User Input	Symbol Name	Var_Input name	Data Type	Setting range	Description
X	Start of incoming RF Information	i_uInputStartReg	Word[Unsigned]		Input from Fieldbus
X	Total Bytes (data +2)	i_wBytesPerAntenna	Word[Signed]	1 to 1024/2048	to fieldbus maximum
X	Perform Input Data evaluation	i_bFieldBusOK	Bit		Fieldbus Ok signal
	Send Antenna to Ground state	i_bFaultReset	Bit		Fault Reset Signal
X	Prerequisite for Read	i_bReadCond	Bit		Input Conditions for Read
X	Read Command	i_bReadCmd	Bit		Read Request
X	Prerequisite for Write	i_bWriteCond	Bit		Input Conditions for Write
X	Write Command	i_bWriteCmd	Bit		Write Begin
X	Set True to Write Constant	i_bTagClearOperation	Bit		If this is true the next write will be a tag clear operation
	Last Accessible Tag Address	i_uLastTagByte	Word[Unsigned]		Tag Size
X	Operation Start Address	i_uOperationStartAdd	Word[Unsigned]	0 to i_uTagByteCapacity	Starting tag Address

User Input	Symbol Name	Var_Input name	Data Type	Setting range	Description
X	Operation Length	i_uOperationLengthBytes	Word[Unsigned]	1 to 2048/2000/256	Length in Bytes
X	Data to Write	i_uWriteDataStart	Word[Unsigned]		Packed Data for Writing to the Tag

■Output labels

User Output	Symbol Name	Var_Output name	Data Type	Description
	FB Executed Normally	o_bOK	Bit	When TRUE, indicates processing has completed normally
	FB Execution Aborted	o_bERR	Bit	When TRUE, indicates an Error has occurred
	FB Error Code	o_uErrID	Word[Unsigned]	FB Error Code Output
X	Start of Outgoing RF Information	o_uOutputStartReg	Word[Unsigned]	Output to fieldbus
	Current Operation Cycle	o_wCurrentODPumber	Word[Signed]	Current Operation Number
	Operation Length/Bytes Per Operation	o_wTotalODPumber	Word[Signed]	Total Operations to Complete
	Antenna Ready	o_bAntennaReady	Bit	RFID Ready For new Command Signal
	Tag Present	o_bTagPresent	Bit	Tag Presence Signal

User Output	Symbol Name	Var_Output name	Data Type	Description
	Read Complete	o_bReadComplete	Bit	Read Complete signal
	Write Complete	o_bWriteComplete	Bit	Write Complete signal
	Error Code	o_uErrorCode	Word[Unsigned]	Error Code
	RFID Controller Faulted	o_bFaultCmd	Bit	On When Controller Error Bit TRUE
	RFID Controller No Response	o_bFaultTimeout	Bit	Command execution time > 20 seconds
X	Read Data	o_uReadDataStart	Word[Unsigned]	Received Data(available at Read Complete

FB Version Upgrade History	
Version	Description
1.01	Initial Release
1.02	Clear Manual Read memory on status Change
1.12	Changed the Manual Read Structure Modified FB variables to use the standard specified in BCN-89000-0823-D
1.13	Added Clear Tag Function
1.20	Added FB Status Outputs
2.00A	Adopted BCN-89000-0969; Corrected Issue with Tag Clear WTOB; Removed 5 Read Write Option, added ob_suRFIDReady; Move to flexible sized input and output
2.10	Continuous Read/write supported
3.00	Tag Clear is integrated into Write

3.01	Label Name Updates
3.02	Removed Fault Timer Input
3.03	Added 256 byte function limit added to packet offset; Introduced IO-link version

SDT Usage

RFIDData

Global Label: RFIDData		
Member	Type	Usage
RFIDCHReady	Bit	Populated by this FB when a new command can be accepted
ManualReadCondTrue	Bit	Populated by associated program code to allow a manual read command
ManualWriteCondTrue	Bit	Populated by associated program code to allow a manual write command
ReadComplete	Bit	Populated by this FB when done and held true until input condition is released
WriteComplete	Bit	Populated by this FB when done and held true until input condition is released
TagPresent	Bit	Populated by this FB
ReadPermission	Bit	User Populated needed by this function block to know when to accept A Read Command
ReadCmd	Bit	User Populated to execute an automatic Read Command; Associated code will populate for HMI driven

Global Label: RFIDData		
Member	Type	Usage
WritePermission	Bit	User Populated needed by this function block to know when to accept A Write Command
WriteCmd	Bit	User Populated to execute an automatic Read Command; Associated program code will populate for HMI driven
ControllerFault	Bit	Attached to FB output pins Command Fault
TimeOutFault	Bit	Attached to FB output pins for Timeout or no response
ErrorCode	Word[Unsigned]	Error Code received from RF controller
Length	Word[Unsigned]	User Populated for automatic control, Associated program code will populate for HMI driven
StartAddress	Word[Unsigned]	User Populated for automatic control, Associated program code will populate for HMI driven

MELPT_APP_GOTRFID

System Label: GOTRFID		
Member	Type	Usage
StationDataRequest_PB	Bit	
StationDataRequestPre_PB	Bit	
ManualRead_PB	Bit	
ManualWrite1_PB	Bit	
ManualWrite16_PB	Bit	
OnLastAntenna	Bit	

System Label: GOTRFID		
Member	Type	Usage
OnLastAddress	Bit	
TagPresent	Bit	
CHReady	Bit	
ReadCondTrue	Bit	
WriteCondTrue	Bit	
ManualReadRequest	Bit(1..16)	Antenna Element used to Trigger Manual Read
ManualWriteRequest	Bit(1..16)	Antenna Element used to Trigger Manual Write
SelectedAntennaNumber	Word[Signed]	
SelectedStatus	Word[Unsigned]	
TagSizeBytes	Word[Unsigned]	Used as input check to prevent operation over the limits of the tag
RFIDErrorCommentNumber	Word[Unsigned]	
ErrorCodes	Word[Unsigned](1..16)	
WriteData1Byte	Word[Unsigned]	
StartAddress	Word[Unsigned]	
WriteStartAddress	Word[Unsigned]	
HMIData	Word[Unsigned](0..15)	

Application Example

To utilize this function block the following inputs and outputs need to be adapted to the application:

I_uInputStartReg

This is the starting register for the fieldbus input for this antenna. You can find this information in the auto-refresh range for the fieldbus used.

For example we are to configure antenna #1 on Profibus this is slot 1 on the device the starting register is W2001

I_wBytesPerAntenna

This is the length of the fieldbus input, this variable is symmetrical. If you define 254 bytes for input it will also output 254 bytes, this information is contained within the fieldbus setup of the node.

I_bFaultReset

Turning this bit to true will send an output from the PLC to the antenna to drive the antenna to ground state.

I_bReadCond

Conditions to allow Read operation, must be true to allow read operation

I_bWriteCond

Conditions to allow Write operation, must be true to allow Write or Tag Clear operation

I_bTagClearOperation

Set to TRUE to make your write operation a tag clear operation, the data used to clear the tag is the first byte of i_unAutoWriteData

O_uOutputStartReg

This is the starting register for the PLC output to this antenna. You can find this information in the auto-

refresh range for the fieldbus used.

For example we are to configure antenna #1 on Profibus this is slot 1 on the device the starting register is W1001. The function block outputs data for the number of bytes list on i_wBytesPerAntenna

O_bFaultCMD

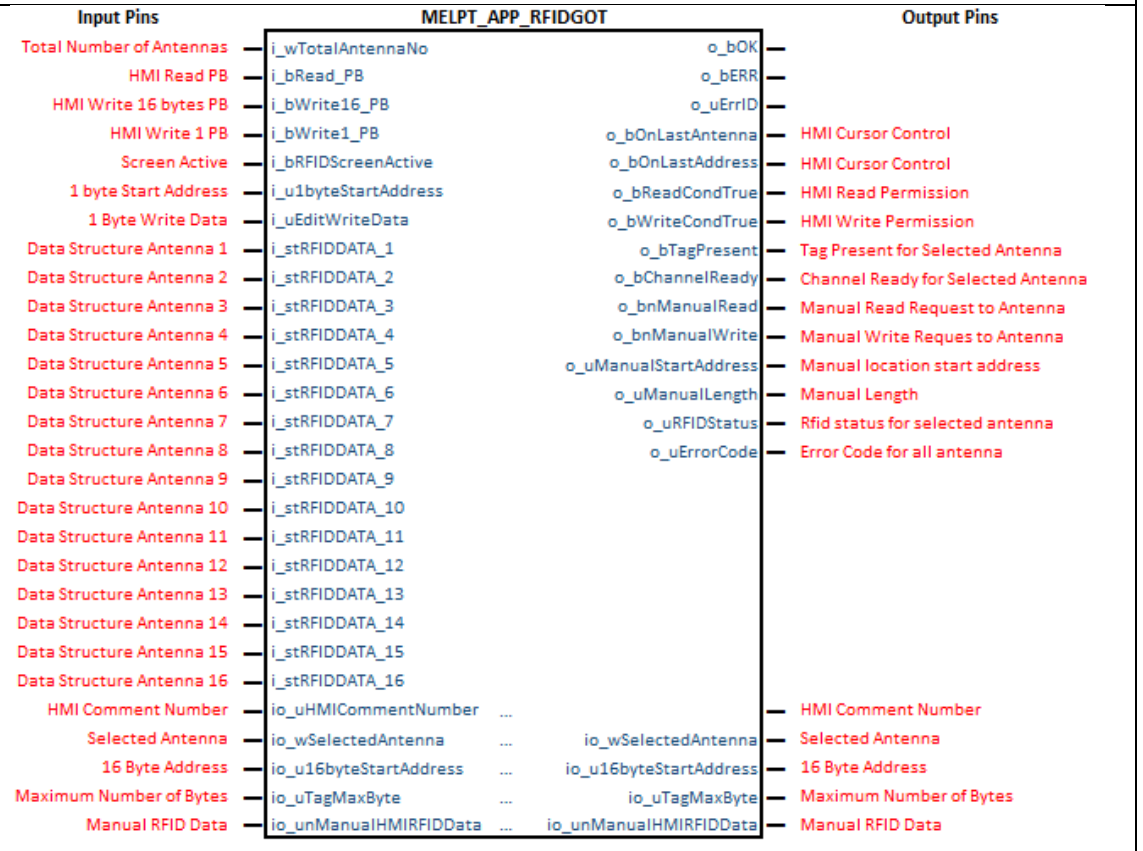
This is true when the controller has returned an error code investigate o_uErrorCode for ErrorCode

O_bFaultTimeout

3.6.1 MELPT_APP_RFIDGOT

This program calls the function block MELPT_APP_RFIDGOT_1KB, _2KB, or _256b. Only one of these programs is designed to be in the scan at any one time and is intended to be used with the data structures needed by the RFID system in use

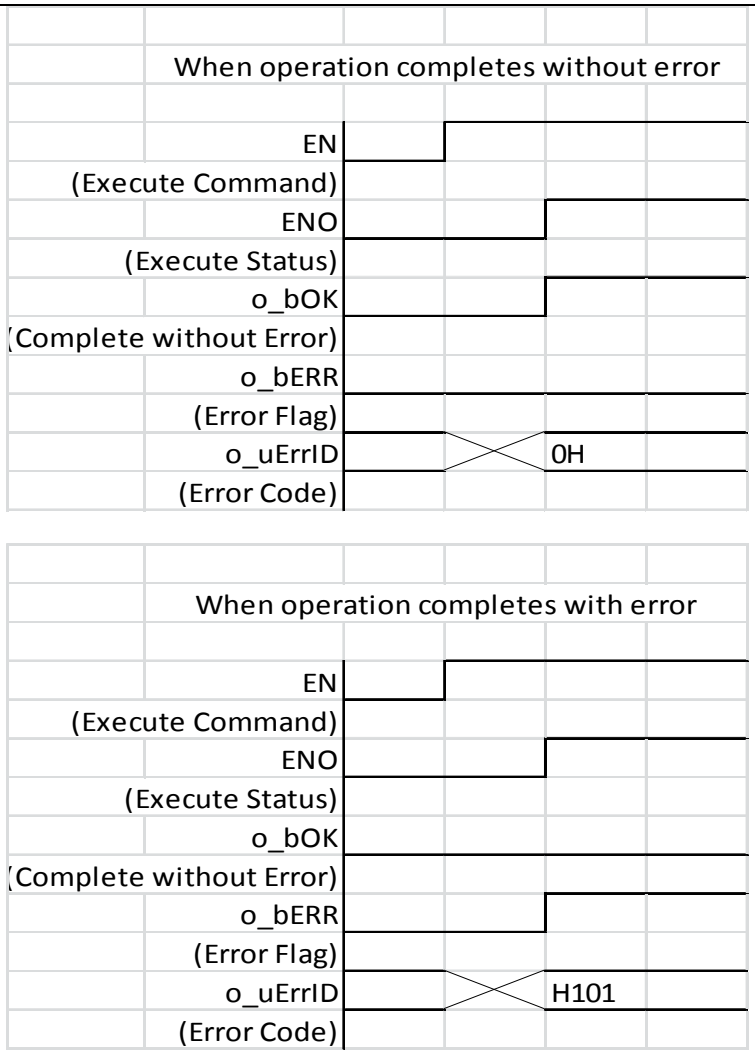
FB Name
MELPT_APP_RFIDGOT

Item	Description
Function overview	This function block interfaces GOT related RFID functions to the selected RFID IF FB. 1 instance per GOT screen, this function block can interface with up to 16 antenna data structures.
Symbol	

<table>
<tr><td>Item</td><td>Description</td></tr>
<tr><td>Applicable Hardware</td><td>
<table>
<tr><th>Series</th><th>Model</th><th>Serial Restriction</th></tr>
<tr><td>MELSEC-Q series</td><td>Universal Model PLC</td><td>None</td></tr>
<tr><td>GOT 1000 series</td><td>GT16(800*600) or Higher</td><td>None</td></tr>
</table>
</td></tr>
<tr><td>Applicable Software</td><td>
<table>
<tr><th>Series</th><th>Version</th></tr>
<tr><td>GX Works 2</td><td>1.536</td></tr>
<tr><td>GT Designer 3</td><td>1.136</td></tr>
</table>
</td></tr>
<tr><td>Programming language</td><td>Structured Ladder/FBD</td></tr>
<tr><td>Number of Ladder Steps</td><td>QnU: 1286 steps
*The number of steps of the FB program depends on the CPU Model that is used and input and output definition</td></tr>
<tr><td>Device Memory Used</td><td>MELPT_APP_RFIDGOT

61 bits

98 words</td></tr>
<tr><td>Compiling method</td><td>Macro type;</td></tr>
<tr><td>Execution type</td><td>Real-time Execution</td></tr>
<tr><td>Dependences</td><td></td></tr>
<tr><td>Function description</td><td>This function block multiplexes up to 16 antenna worth of information for display on the HMI. The status of the selected antenna is determined, and Manual RFID Read and Write Requests are generated from this FB.</td></tr>
</table>

Item	Description
Restrictions and precautions	
Timing chart	

FB Error Code

Error Code	Description

Error Code	Description
0	No Error.
H101(257)	Tag Max Byte is equal to zero
H102(258)	Total Antenna Number is greater than 16
H103(259)	Total Antenna number is less than or equal to zero

Labels

■ Input labels

User Input	Symbol Name	Var_Input name	Data Type	Setting range	Description
X	Total Number of Antennas	i_wTotalAntennaNo	Word[Signed]	1 to 16	No. of Antennas used
	HMI Read PB	i_bRead_PB	Bit		HMI PB to read tag
	HMI Write 16 bytes PB	i_bWrite16_PB	Bit		HMI PB to write to tag 16 bytes
	HMI Write 1 PB	i_bWrite1_PB	Bit		HMI PB to write to tag 1 byte
	Screen Active	i_bRFIDScreenActive	Bit		Screen Active Bit
	RFID IF Data Structure	i_stRFIDData_X	RFIDData_1KB RFIDData_2KB RFIDData_256b		RFID Data for one antenna
	1 byte Start Address	i_u1byteStartAddress	Word[Unsigned]		1 Byte Write Start Address
	1 Byte Write Data	i_uEditWriteData	Word[Unsigned]		The data to be written to the tag for 1-byte write

■ Input/Output labels

User Input	Name	Var_In_Out name	Data Type	Setting range	Description

User Input	Name	Var_In_Out name	Data Type	Setting range	Description
x	HMI Comment Number	Io_uHMICommentNumber	Word[Unsigned]	1-255	Comment File for the Advanced user Alarm.
	Selected Antenna	io_wSelectedAntenna	Word[Signed]		Selected Antenna
	16 Byte Address	io_u16byteStartAddress	Word[Unsigned]		16 Byte Start Address
X	Maximum Number of Bytes	io_uTagMaxByte	Word[Unsigned]	0-32k	Tag Size in Bytes
	Manual RFID Data	io_unManualHMIRFIDData	Word[Unsigned](0..15)		16-byte write data or data read from tag

■ Output labels

Name	Var_Output name	Data Type	Description
FB Executed Normal	o_bOK	Bit	When TRUE, indicates processing has completed normally
FB Execution abnormal	o_bERR	Bit	When TRUE, indicates an Error has occurred
FB Error Code	o_uErrID	Word[Unsigned]	FB Error Code Output
HMI Cursor Control	o_bOnLastAntenna	Bit	HMI Indicator for PB Control
HMI Cursor Control	o_bOnLastAddress	Bit	HMI Indicator for PB Control
HMI Read Permission	o_bReadCondTrue	Bit	HMI Indicator for PB Control
HMI Write Permission	o_bWriteCondTrue	Bit	HMI Indicator for PB Control
Tag Present for Selected Antenna	o_bTagPresent	Bit	Tag present Ind. On HMI
Channel Ready for Selected Antenna	o_bChannelReady	Bit	Selected Antenna can accept new command
Manual Read Request to Antenna	o_bnManualRead	Bit(1..16)	Manual Read to RFID FB

Name	Var_Output name	Data Type	Description
Manual Write Request to Antenna	o_bnManualWrite	Bit(1..16)	Manual Write to RFID FB
Manual location start address	o_uManualStartAddress	Word[Unsigned]	Manual Start Address
Manual Length	o_uManualLength	Word[Unsigned]	Manual Operation Length
RFID status for selected antenna	o_uRFIDStatus	Word[Unsigned]	Status of RFID HMI Ind.
Error Code for all antenna	o_uErrorCode	Word[Unsigned](1..16)	Array of Error Codes to Display on HMI

FB Version Upgrade History

Version	Description
1.10	Initial Release
1.12	Label Update
1.20	FB Status Outputs added
2.00A	Adopted BCN-89000-0969 added intelligent module check, changes made to screen display and use of Superimpose Offset; Issue with Detailed Diagnostic Output corrected
2.10	Start address changed to double word
3.00	Modified for version 3.00 structures, start address back to single word
3.03	Modified for 16 independent inputs instead of array of structures; memory improvement

SDT Usage

1. RFIDData

Global Label: RFIDData		
Member	Type	Usage
RFIDCHReady	Bit	Selected Antenna Data forwarded to HMI structure
ManualReadCondTrue	Bit	Selected Antenna Data forwarded to HMI structure
ManualWriteCondTrue	Bit	Selected Antenna Data forwarded to HMI structure
ReadComplete	Bit	Selected Antenna Data forwarded to HMI structure
WriteComplete	Bit	Selected Antenna Data forwarded to HMI structure
TagPresent	Bit	Selected Antenna Data forwarded to HMI structure
ReadPermission	Bit	Selected Antenna Data forwarded to HMI structure
ReadCmd	Bit	
WritePermission	Bit	Selected Antenna Data forwarded to HMI structure
WriteCmd	Bit	
ErrorCode	Word[Unsigned]	All Data forwarded to HMI
Length	Word[Unsigned]	Modified by this FB and sent to individual RF function block
StartAddress	Word[Unsigned]	Modified by this FB and sent to individual RF function block

2. MELPT_APP_GOTRFID

System Label: GOTRFID		
Member	Type	Usage
StationDataRequest_PB	Bit	
StationDataRequestPre_PB	Bit	
ManualRead_PB	Bit	GOT Screen Manual Read
ManualWrite1_PB	Bit	GOT Screen Manual Write 1 byte
ManualWrite16_PB	Bit	GOT Screen Manual Write 16byte

System Label: GOTRFID		
Member	Type	Usage
OnLastAntenna	Bit	True when on Last Antenna
OnLastAddress	Bit	True When on Last Address
TagPresent	Bit	Tag Present Ind.
CHReady	Bit	Antenna Ready Indicator
ReadCondTrue	Bit	Read Cond True allow Manual Read Push Button
WriteCondTrue	Bit	Write Cond True FB will accept Manual Write PB if pressed
ManualReadRequest	Bit(1..16)	Manual Read Pulse for Each Antenna, this function block pulses the selected antenna bit
ManualWriteRequest	Bit(1..16)	Manual Write Pulse for Each Antenna, this function block pulses the selected antenna biut
SelectedAntennaNumber	Word[Signed]	Antenna Selection number hmi controlled
SelectedStatus	Word[Unsigned]	Determined by this FB
TagSizeBytes	Word[Unsigned]	This is the last editable byte number of the tag, used for input checking
RFIDErrorCommentNumber	Word[Unsigned]	Populate as part of the RFID driver you are using:
ErrorCodes	Word[Unsigned](1..16)	Error Codes for RFID alarming
WriteData1Byte	Word[Unsigned]	1 Byte of Write Data
StartAddress	Word[Unsigned]	Manual Read and 16 Write Start Add.
WriteStartAddress	Word[Unsigned]	1 byte Write Start Address

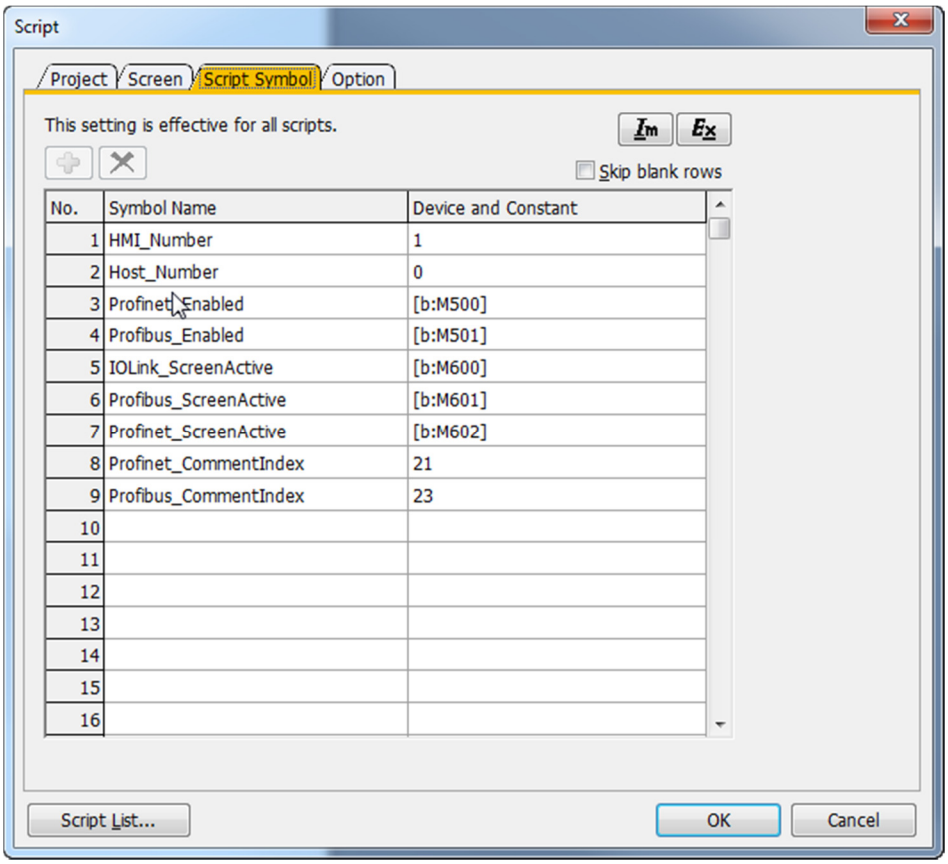
System Label: GOTRFID		
Member	Type	Usage
HMIData	Word[Unsigned](0..15)	Manual Data I/O 16bytes of read/write data
OperationAddress	Word[Unsigned]	The Tag Start Address for HMI requests
OperationLength	Word[Unsigned]	The Length in bytes for HMI requests

4 GTDESIGNER 3 HMI ELEMENTS

Much of the Profibus and IO-Link Diagnostics HMI functionality is pre-configured to allow diagnostics of Profibus and IO-Link Devices and Parameter Read/Write on the Maintenance section. However, some extra configuration of certain screen elements may be required based on network requirements.

The first section will show the Profibus Diagnostics
The second section will cover the IO-Link Diagnostics screens.
The third section will show the Maintenance parameter screens for initial IO-Link device parameter setup and subsequent adjustment of parameters.
The fourth section will cover what is needed to add the pressure switch graphics to an HMI screen.
The fifth section regards the Manual Edit Screen for RFID.

Script Symbols are used, the proper devices on the PLC must be assigned to the Script Symbol



4.1.1 Profibus Diagnostic Displays

Fieldbus Diagnostics will determine fieldbus node health.

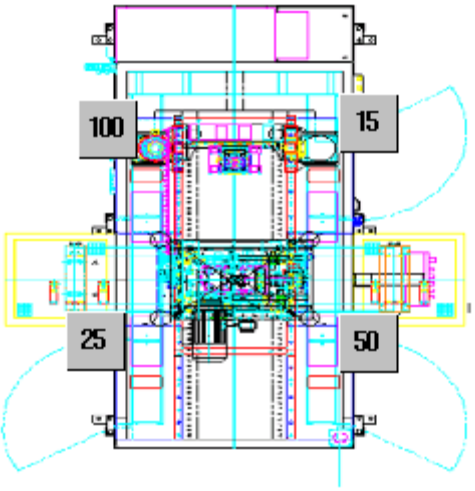
The associated function block will determine node health and categorize the node number into one of the following categories

ProfibusDiag.Nodestatus[0..127]

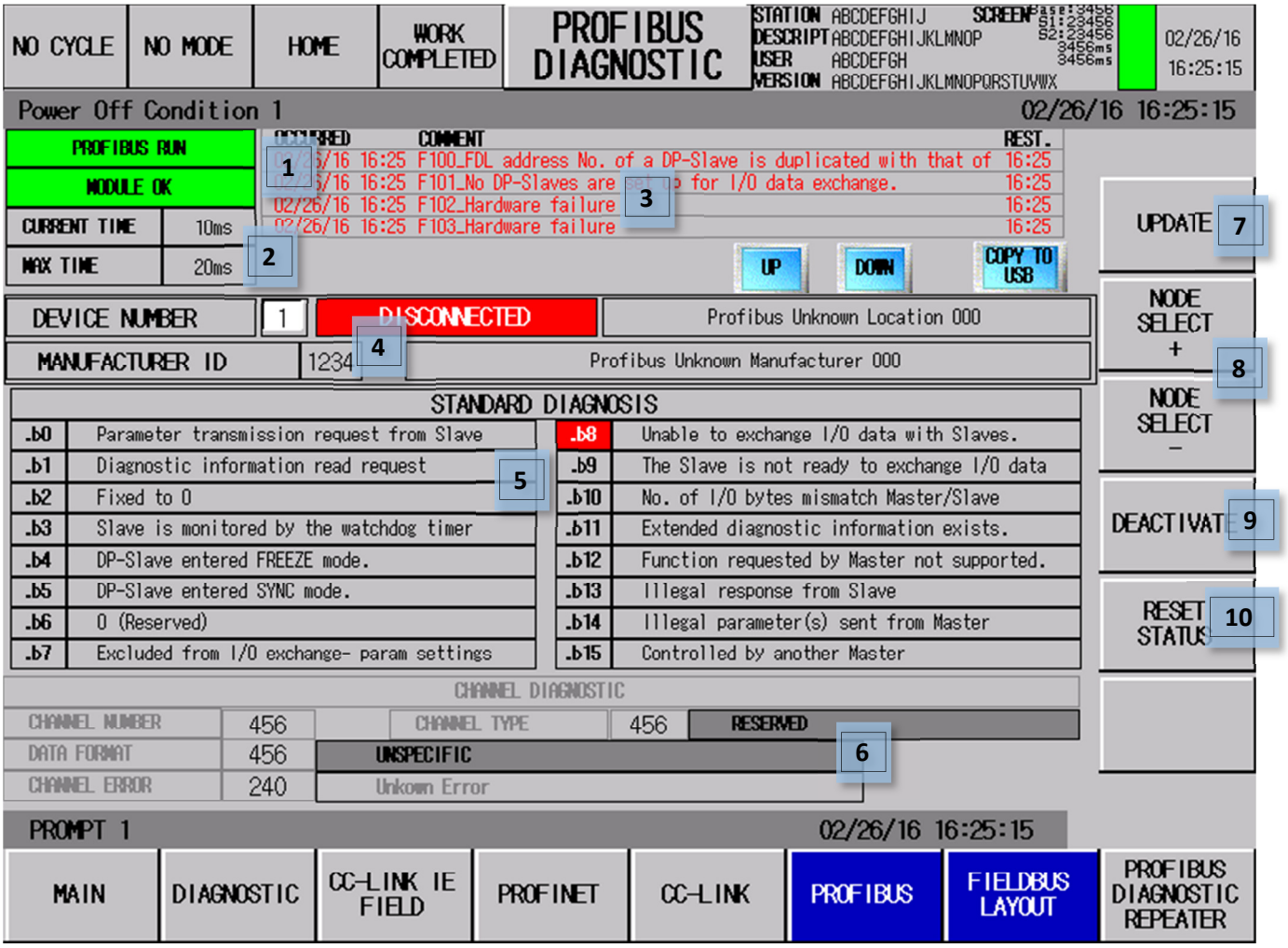
State	Decimal Value	Graphic
Not Configured	0	
Normal	1	
Maintenance	34	
Ok Was Maintained	33	
Defective	68	
Ok Was Defective	65	
Disconnected	136	
Ok Was Disconnected	129	
Deactivated	272	
Ok Was Deactivated	257	

Each Fieldbus Diagnostic Screen can be broken into a layout and detailed diagnostic screen.

The layout is intended to give the operator a general view of machine operation, and where the fieldbus nodes are located.



The detailed diagnostic is intended to provide detailed information about one node.



Screen script #300 monitors the screen you are on and when you are on a specific fieldbus diagnostic screen the specific labels associated with the screen are loaded into the general HMI registers for display.

Each Indicator on the Overview screen is layered with a hidden multifunction switch.

The switch accomplishes 3 tasks

Sets bit GB65111, when this bit is set the screen navigates to the appropriate detailed diagnostic screen

Sets the node number

When on the detailed diagnostic screen the information displayed is for this node number

Sets the particular fieldbus.update signal

This signal is used by the associated FB to know to retrieve information for the node selected

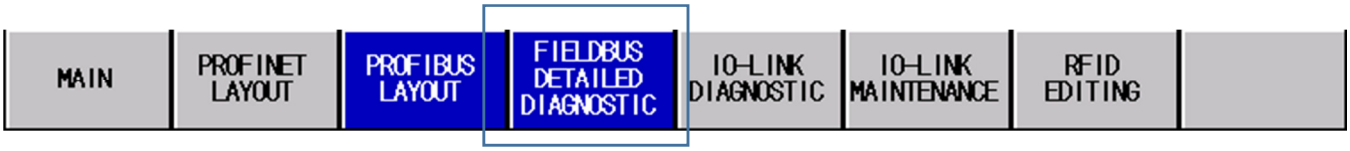
On each layout screen indicators and switches are provided for the maximum allowed nodes per fieldbus, it is recommended that not configured indicators be moved to another window inside the project for future use.

Navigation

The navigation bar has two options for each fieldbus

Layout

Detailed Diagnostic



The fieldbus detailed diagnostic button is a toggle switch to switch between the layout and detailed diagnostic for the selected Fieldbus

HMI-Profibus Detailed Diagnostics

This screen provides detailed information for the selected node of the Profibus Fieldbus

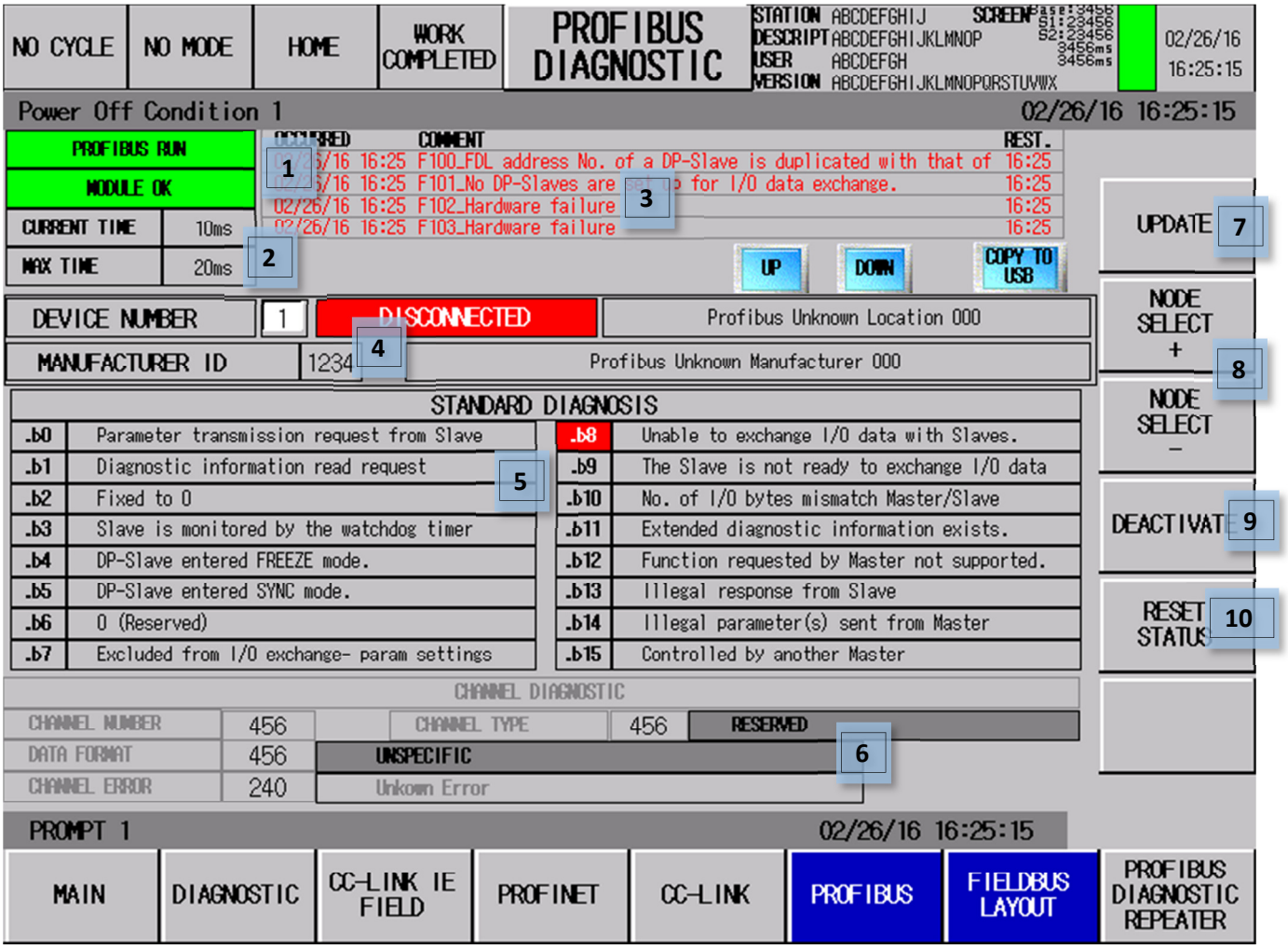


Figure 1-Profibus Diagnostics

Item #	Description	Object	Details

Item #	Description	Object	Details
1	Host Status	Bit Lamp: ProfibusDiag.HostOK	Off: PROFIBUS NOT RUNNING On: PROFIBUS RUN
	Module Faulted Indicator	Bit Lamp: ProfibusDiag.FieldbusModuleFault	Off: MODULE OK On: MODULE ERROR
2	Current Fieldbus Time	Numeric Display: ProfibusDiag.LinkTime[0]	Ms text added
2	Max Fieldbus Time	Numeric Display: ProfibusDiag.LinkTime[2]	Ms text added
3	Alarm Area	User Alarm Display(23)	COPY to USB button Executes Screen script that will copy alarms to E:\backup\Profibus.CSV
4	Selected Node	Numeric Input ProfibusDiag.SelectedNode	Numeric Input limited by the first and last node number detected by FB; Sets ProfibusDiag.Update when a new value is entered
4	Location	Word Comment Display: ProfibusDiag.SelectedNode	Comment File #23 $$ + 10000

<table>
<tr><th>Item #</th><th>Description</th><th>Object</th><th>Details</th></tr>
<tr><td>4</td><td>Manufacturer</td><td>Word Comment Display:
ProfibusDiag.SelectedNode</td><td>Comment File #23
20000 + $$</td></tr>
<tr><td>4</td><td>Selected Status</td><td>Word Lamp:
ProfinetDiag.SelectedStatus</td><td>
<table>
<tr><th>State</th><th>Value</th><th>Graphic</th></tr>
<tr><td>Not Configured</td><td>0</td><td>NOT CONFIGURED</td></tr>
<tr><td>Ok</td><td>1</td><td>OK</td></tr>
<tr><td>Ok Was Maintained</td><td>33</td><td>OK, WAS MAINTAINED</td></tr>
<tr><td>Maintenance</td><td>34</td><td>MAINTENANCE</td></tr>
<tr><td>Ok Was Defective</td><td>65</td><td>OK, WAS DEFECTIVE</td></tr>
<tr><td>Defective</td><td>68</td><td>DEFECTIVE</td></tr>
<tr><td>Ok Was Disconnected</td><td>129</td><td>OK, WAS DISCONNECT</td></tr>
<tr><td>Disconnected</td><td>136</td><td>DISCONNECTED</td></tr>
<tr><td>Ok Was Deactivated</td><td>257</td><td>OK, WAS DEACTIVATE</td></tr>
<tr><td>Deactivated</td><td>272</td><td>DEACTIVATED</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Item #</th><th>Description</th><th>Object</th><th colspan="3">Details</th></tr>
<tr><td>5</td><td>Standard Diagnostic</td><td>Word Comment Display:
Numeric Display:
ProfibusDiag.SelectedDeviceDiagnostic</td><td colspan="3">Comment File #23
STANDARD DIAGNOSIS
<table>
<tr><td>.b0</td><td>Parameter transmission request from Slave</td></tr>
<tr><td>.b1</td><td>Diagnostic information read request</td></tr>
<tr><td>.b2</td><td>Fixed to 0</td></tr>
<tr><td>.b3</td><td>Slave is monitored by the watchdog timer</td></tr>
<tr><td>.b4</td><td>DP-Slave entered FREEZE mode.</td></tr>
<tr><td>.b5</td><td>DP-Slave entered SYNC mode.</td></tr>
<tr><td>.b6</td><td>0 (Reserved)</td></tr>
<tr><td>.b7</td><td>Excluded from I/O exchange- param settings</td></tr>
<tr><td>.b8</td><td>Unable to exchange I/O data with Slaves.</td></tr>
<tr><td>.b9</td><td>The Slave is not ready to exchange I/O data.</td></tr>
<tr><td>.b10</td><td>No. of I/O bytes mismatch Master/Slave</td></tr>
<tr><td>.b11</td><td>Extended diagnostic information exists.</td></tr>
<tr><td>.b12</td><td>Function requested by Master not supported.</td></tr>
<tr><td>.b13</td><td>Illegal response from Slave</td></tr>
<tr><td>.b14</td><td>Illegal parameter(s) sent from Master</td></tr>
<tr><td>.b15</td><td>Controlled by another Master</td></tr>
</table></td></tr>
<tr><td>6</td><td>Channel</td><td>Information retrieved from device</td><td>Channel Number</td><td>Numeric Display</td><td>ProfibusDiag.ChannelNumber</td></tr>
</table>

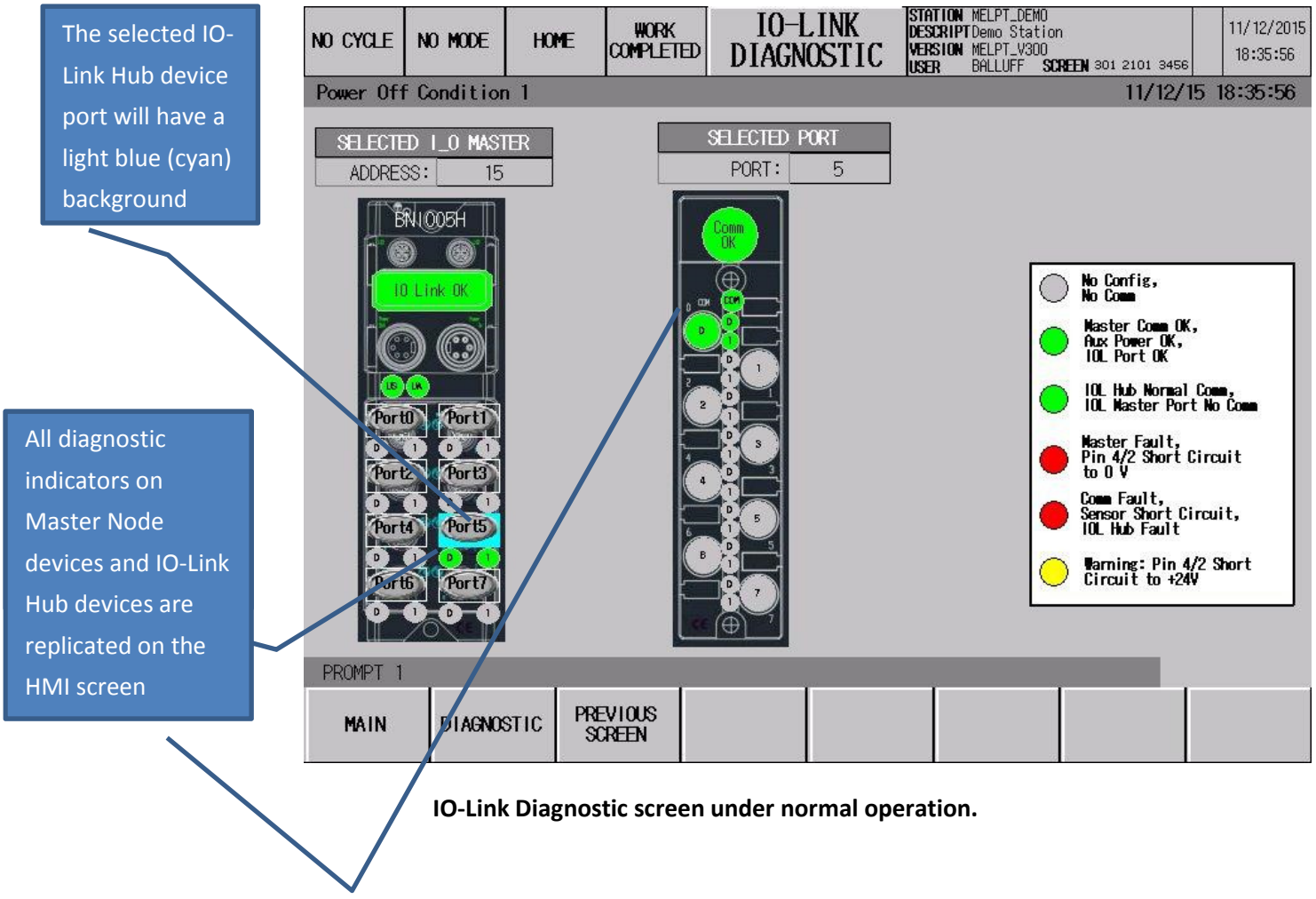
Item #	Description	Object	Details			
	Diagnostics	via acyclic means when the Profibus.update is triggered	Channel Type	ProfibusDiag.ChannelType	0	Reserved
					64	Input
					128	Output
					192	Input/Output
			Data Format	ProfibusDiag.ChannelDataFormat	0	Unspecific
					32	BIT
					64	2 BIT
					96	4 Bit
					128	Byte
					144	Word
					192	2 Word
					224	Reserved
			Channel Error	ProfibusDiag.ChannelError	1	Short Circuit
					2	Undervoltage
					3	Overvoltage
					4	Overload
					5	Overtemperature
					6	Line Break
					7	Upper Limit Value Exceeded
					8	Lower Limit Value Exceeded
					9	Error

Item #	Description	Object	Details			
					16	Parametrization Fault
					17	Power Supply Fault
					18	Fuse Blown / Open
					19	Communication Error
					20	Ground Fault
					21	Reference Point Lost
					22	Process Event Lost
					23	Threshold Warning
					24	Output Disabled
					25	Safety Event
					26	External Fault
					27	Sensor has incorrect Configuration
					28	Manufacturer Specific

Item #	Description	Object	Details			
					29	Primary Variable out of limits
					30	Non-primary Variable out of limits
					31	The Channel Needs Parameters
					32	Reserved
					33	Sensor Supply Short Circuit
					34	Ground fault

4.1.2 IO-Link Diagnostics

The IO-link Diagnostics Screen shows the Profibus Master Node device (on left) and the selected port's IO-Link connected io-link device on the right. See graphic below for more details:

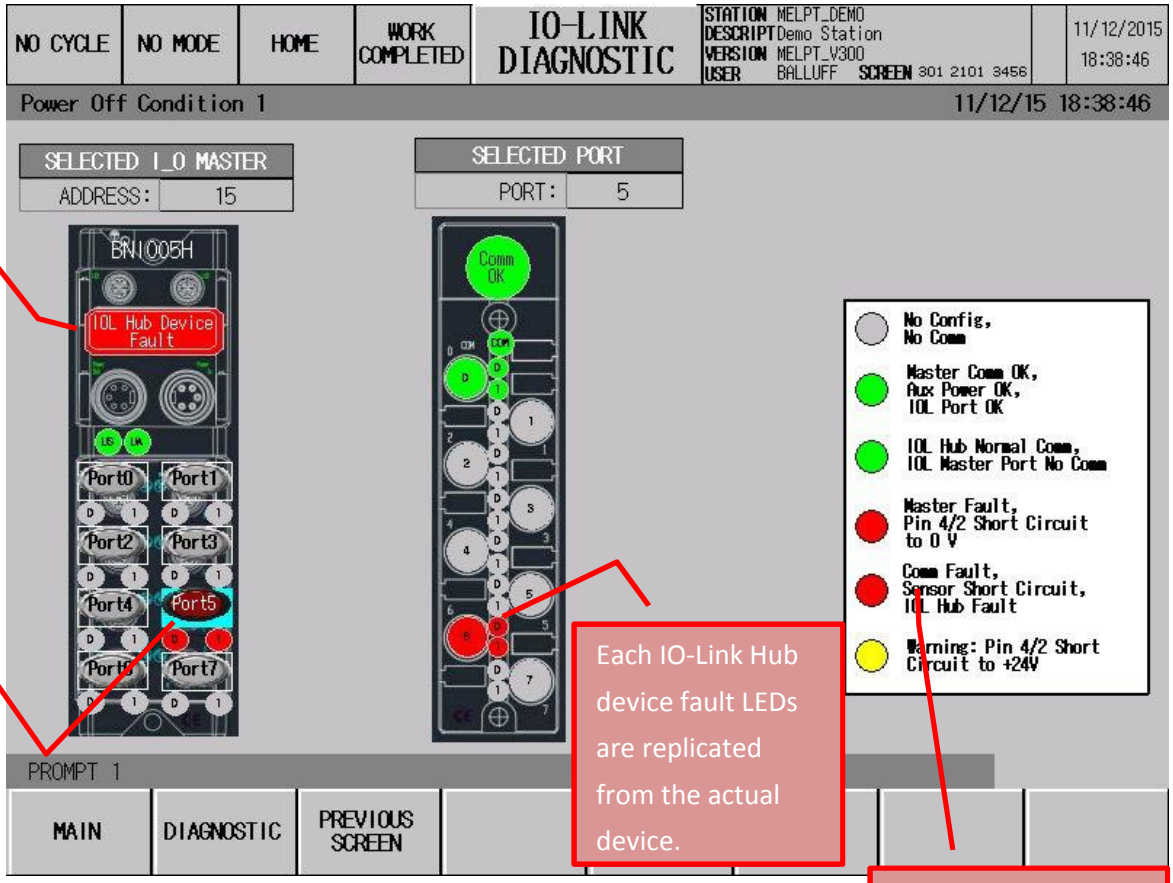


IO-Link Diagnostic screen under normal operation.

See next page for IOL Diagnostic Faulted example.

Errors/Warnings from the Master Node device will have a solid red indicator. IO-Link Hub-related errors/warnings will be blinking

Any IO-Link hub faults will show a blinking red indicator on the Port button. Pressing this will show the faulted device on the right



Each IO-Link Hub device fault LEDs are replicated from the actual device.

This legend can be used to determine the error or warning to aid in troubleshooting

IO-Link Diagnostic screen with IO-Link Hub Fault.

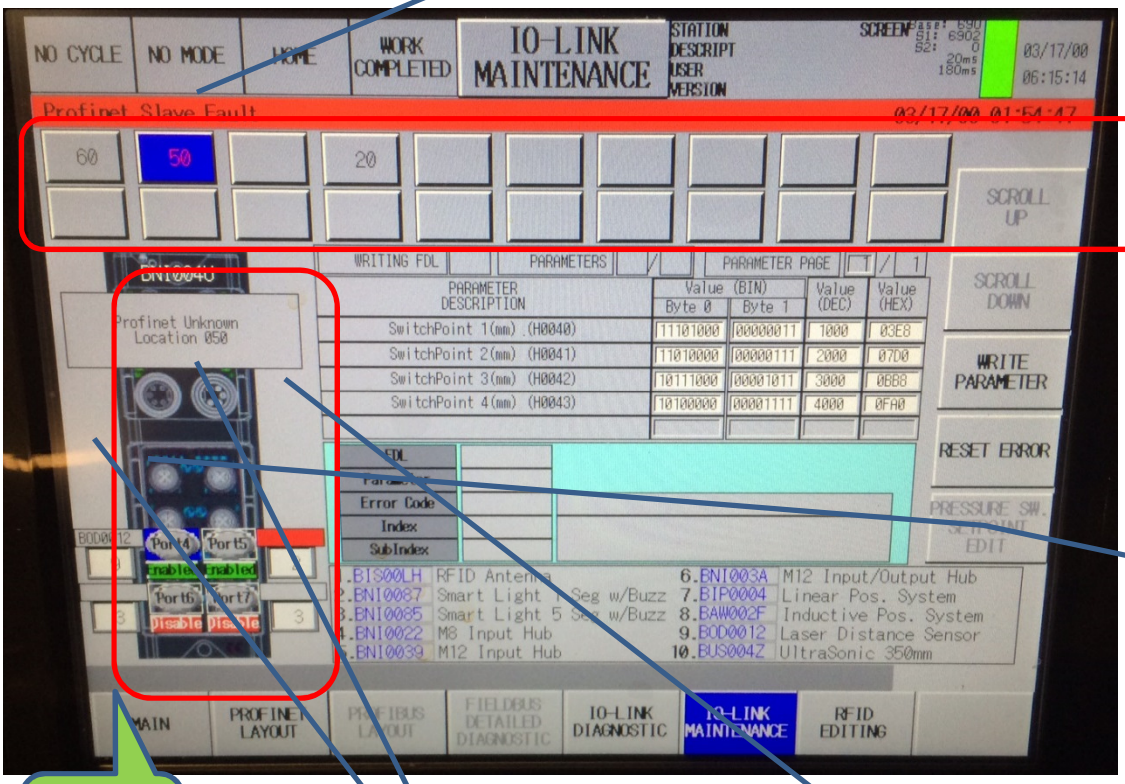
4.1.3 IO-Link Maintenance – Device Parameters Setup Display

In the Maintenance section of the HMI, there is an IO-Link Maintenance screen which allows reading/writing of Parameters of All IO-Link Hubs of all IO-Link-enabled Node devices on a Profibus network.

The parameters for all the configured IOL hubs are stored in permanent memory in the PLC. Whenever a node is disconnected and reconnected or someone performs a Write Parameters function, data stored from that PLC memory location is written to the parameters of the target IO-Link hub device.

For initial commissioning of the Profibus network, all configured IO-Link devices must have Device Type assigned. This is accomplished by placing the device CFG function block in the initial scan. This will allow a check to be performed and annunciate a possible wrong device replacement/installation.

See below for more details:



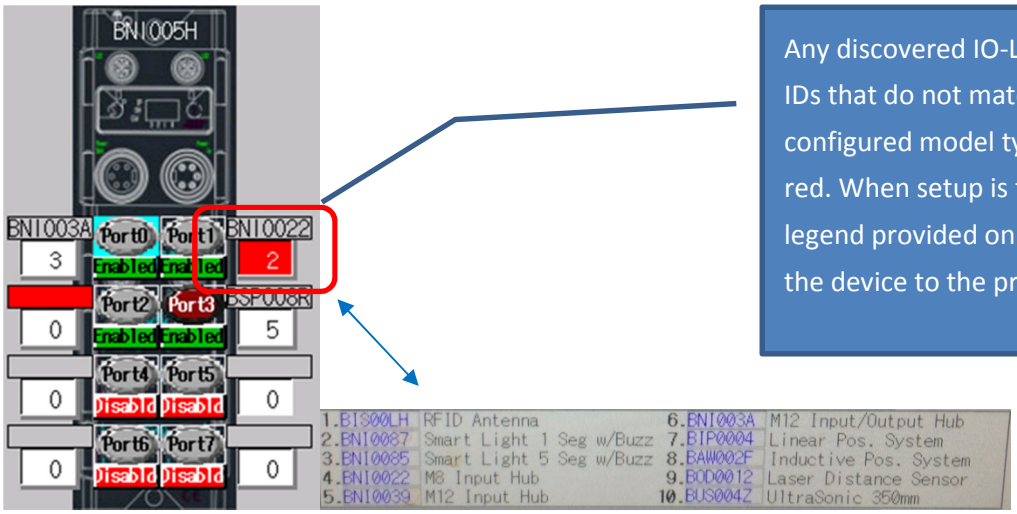
All possible nodes on the ProfiNet network will have an enabled button here. Red-colored buttons represent some error on that node. Pressing the node button will load the node

Selected Port will highlight in blue. Enable/Disable is for IO-Link Diagnostics. Disabling for a port will prevent Errors/Warnings on the IO-Link device from registering as ProfiNet faults

Any port with a connected device type will show the device type name in this box. If communication is lost, the indicator will blink red.

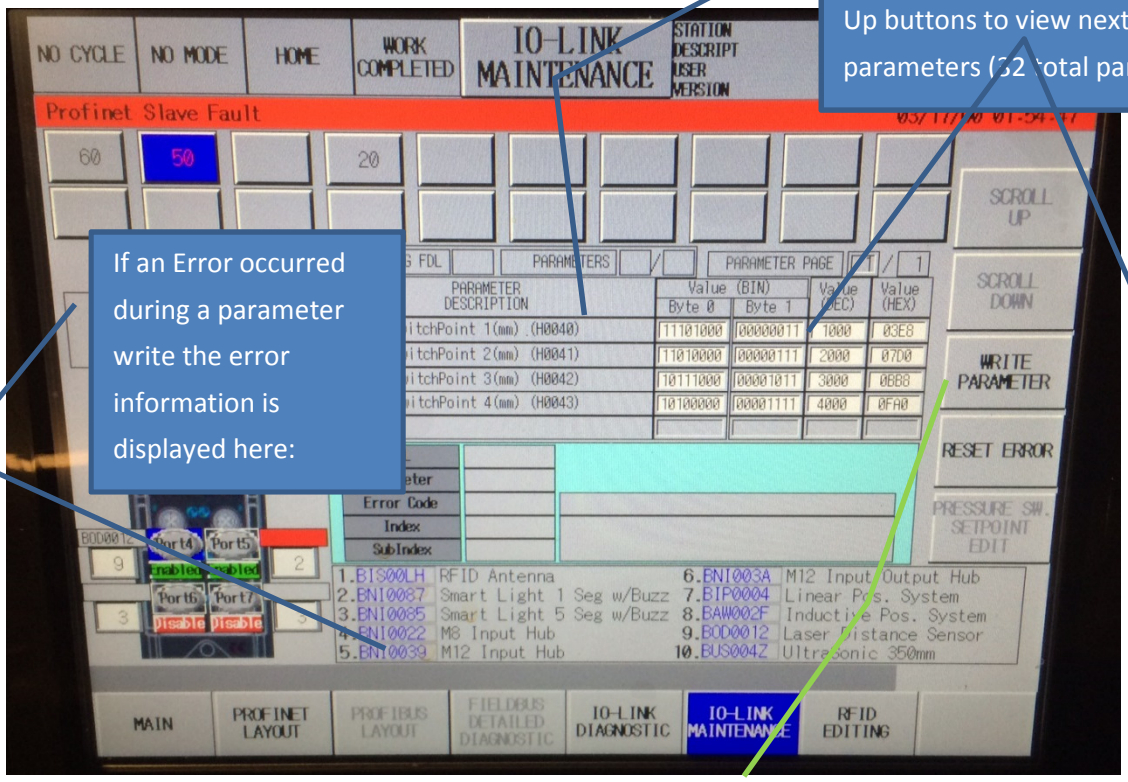
IOL Parameter write error(s) will make the target node button blink red, pressing the button will load up error details to the right of the module graphic

Device mismatches will turn this indicator background red. White=device match.



Any discovered IO-Link device product IDs that do not match the user-configured model type will show in red. When setup is first done, use the legend provided on the screen to set the device to the proper model type.

Actual parameter settings for IO-Link device are listed here. Use the Scroll Down and Scroll Up buttons to view next 5 or previous 5 parameters (32 total parameters possible).



If an Error occurred during a parameter write the error information is displayed here:

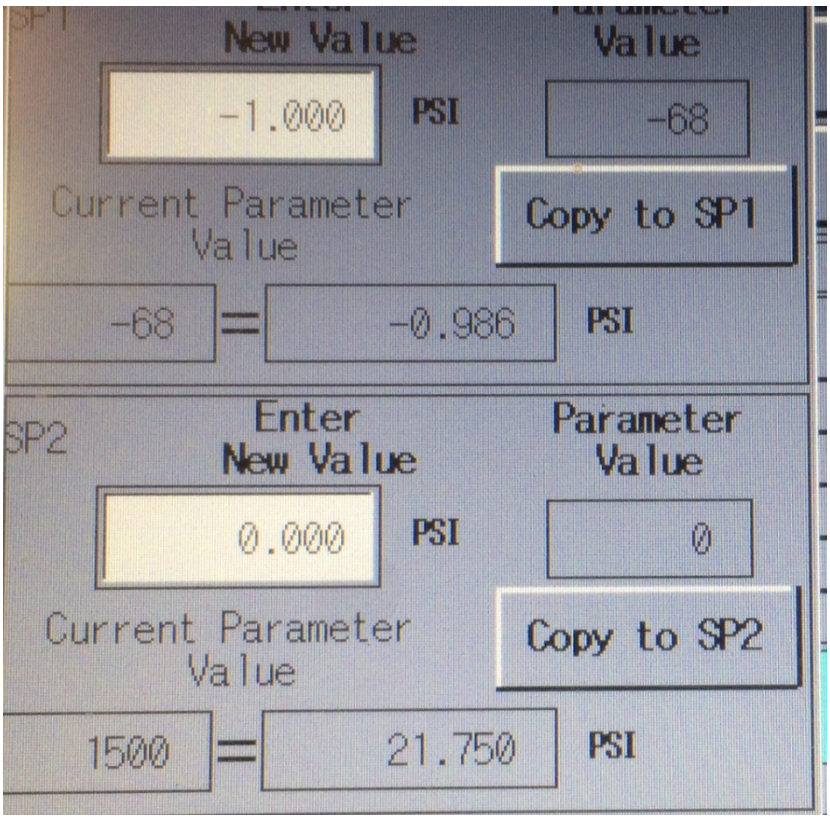
Press Either the Decimal (DEC) value or Hexadecimal (HEX) value for each parameter to input new values, Clicking Enter on the number display when complete.

When the Enter key is pressed for numeric entry the parameters are saved to the PLC. If it is connected, a 10 second timer begins additional edits can be made, to reset the timer. Parameters are written to the device when the timer expires, the device is navigated away from, or the write Parameters Push Button is pressed.

4.1.4 IO-Link Pressure Switch Devices

4.1.4.1 Modifying Set points

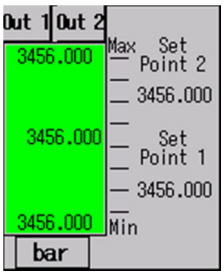
When modifying the setpoint of a Balluff pressure switch, the pressure switch accepts the parameter in bar, with a device multiplier. The operator may not know the conversion to the unit type commonly used. On the maintenance screen when a pressure switch device is selected the Pressure Sw. Setpoint Edit button is available.



This window converts the current parameter value into the current unit type. When a change is to be made the operator will place a new value in the input box, the parameter value is calculated in the display box. Transferring from the window to the parameter list is done with the copy to push button.

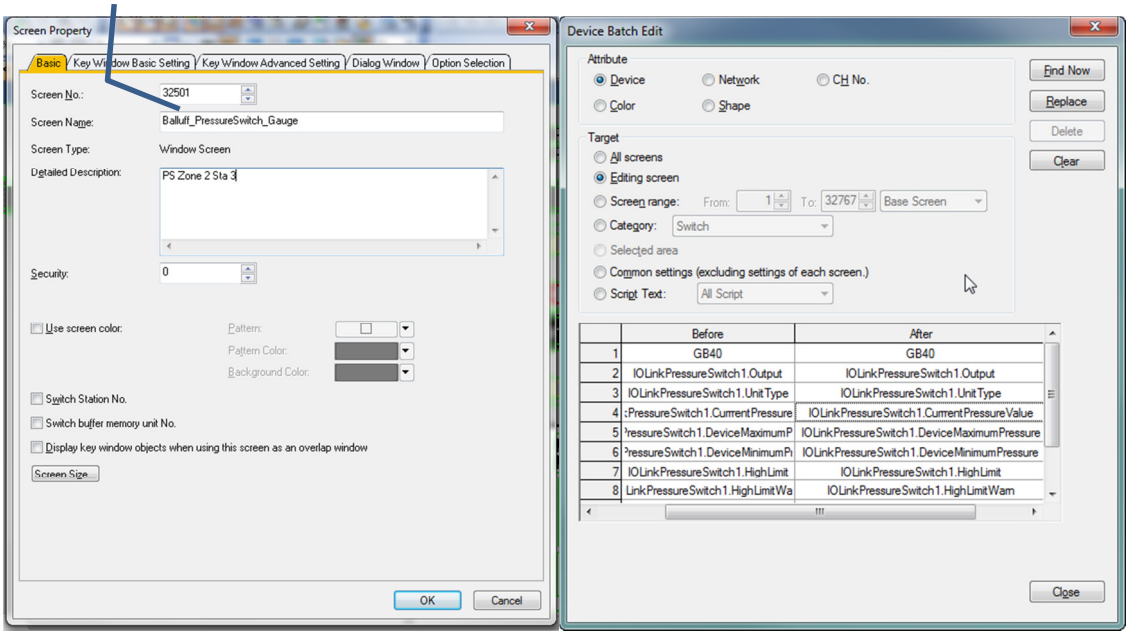
4.1.4.2 Graphics Display

The Balluff pressure switch IO-Link device has a gauge window that can be added as an overlay window on any base HMI screen. For each pressure switch used on the system, a new duplicated gauge window must be created.

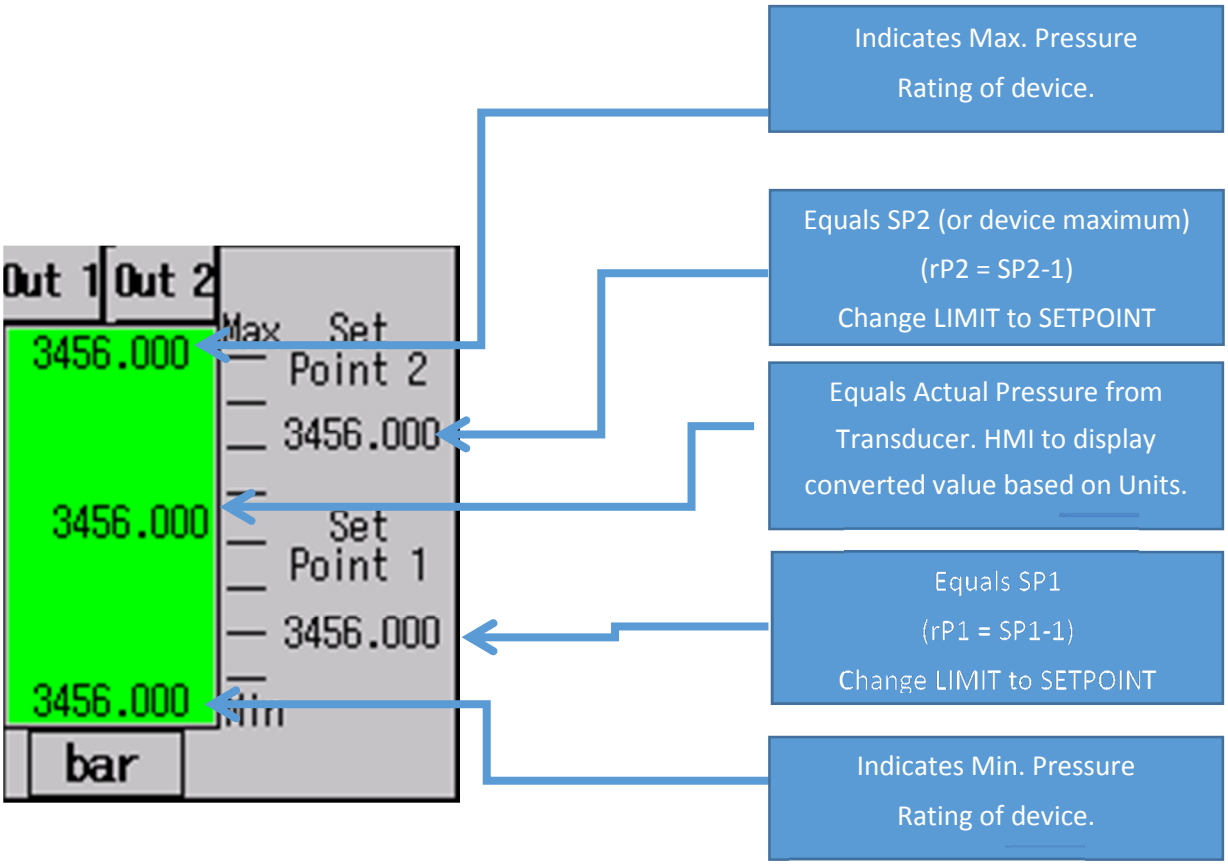


For each new pressure switch gauge needed, window screen 32500 can be copied and pasted to new window screens (see next page):

After the copy/paste of the 32500 window screen, a new window dialog box appears. Adjust the Screen No. to any unused screen number and click OK. The



After Copying the Pressure Device window, use the Device Batch Edit top change HMI variables to the new system label.



BAR Color designations:

RED: Less than SP1

YELLOW: Between SP1 and (((SP2-SP1)x.05)+SP1)

GREEN: Between and (((SP2-SP1)x.05)+SP1) (SP2-((SP2-SP1)x.05))

YELLOW: Between (SP2-((SP2-SP1)x.05)) and SP2

RED: Greater than SP2

4.1.5 RFID Editing Screen

The RFID screen provides a manual interface to the RFID system installed

RFID systems are installed as a user library, the user library contains a program and function block for reading and writing to multiple antennas on a RFID controller.

The HMI Interface is a standard screen with HMI Interface function block. The program included in the user library already interfaces the HMI driven manual components with the standard RFID function block

Up to 16 antennas can be managed by the screen, function block and labels provided with MELPT

Writing to the RF tag is a password protected function

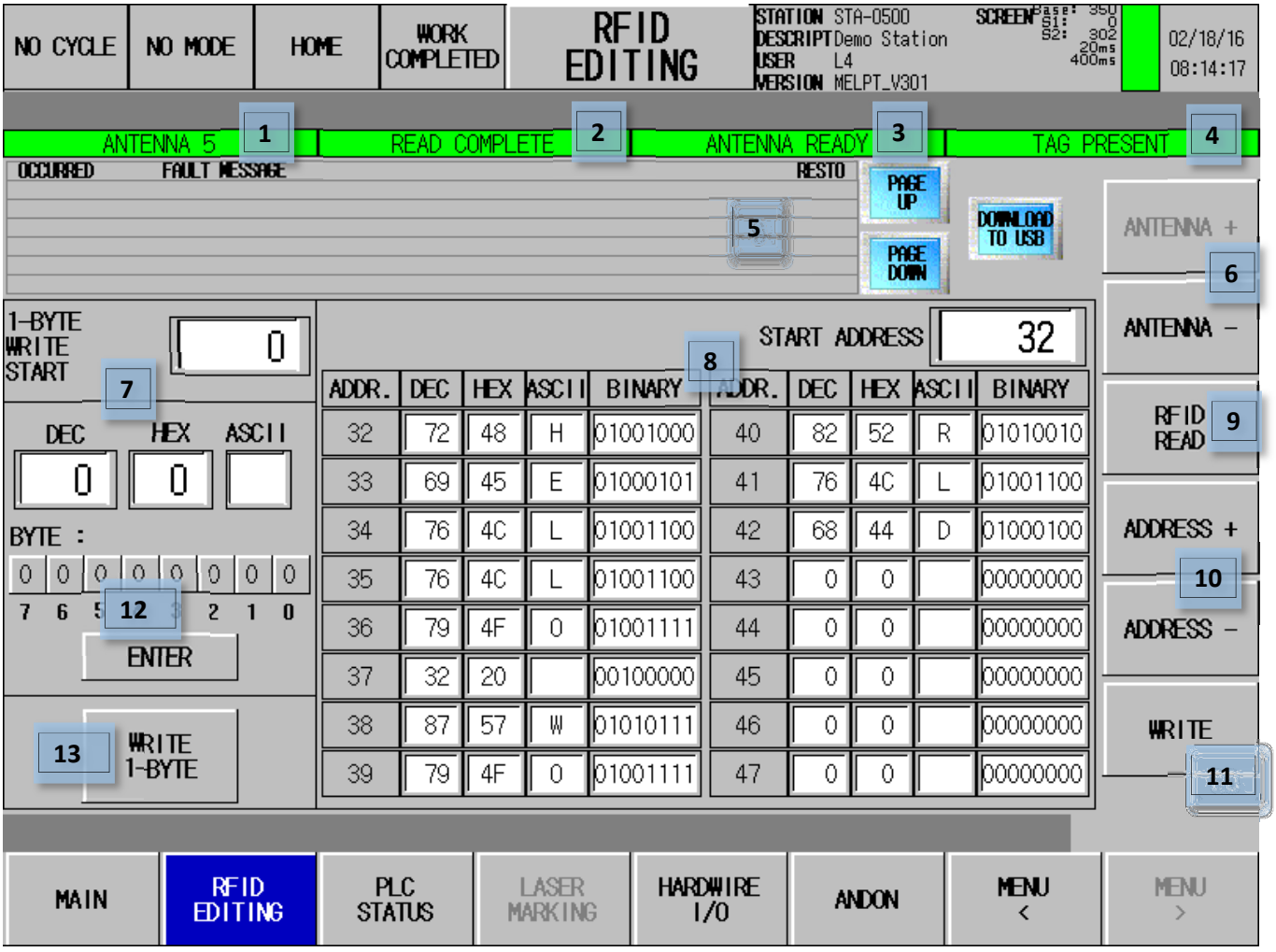


Figure-RFID Editing Screen

<table>
<tr><th>#</th><th>Description</th><th>Object</th><th>Details</th></tr>
<tr><td>1</td><td>Antenna
Number</td><td>Word Comment Display:
GOTRFID.SelectedAntennaNumber
Comment Displayed is $$+50
Comment File:125</td><td>1-16</td></tr>
<tr><td>2</td><td>Operation
Status
Indicator</td><td>Word Comment Display:
GOTRFID.SelectedStatus</td><td>
<table>
<tr><th>Value</th><th>Graphic</th></tr>
<tr><td>None</td><td>INVALID DATA</td></tr>
<tr><td>1</td><td>FAULTED</td></tr>
<tr><td>2</td><td>READ IN PROGRES</td></tr>
<tr><td>3</td><td>READ COMPLETE</td></tr>
<tr><td>4</td><td>WRITE IS PROGRE</td></tr>
<tr><td>5</td><td>WRITE COMPLETE</td></tr>
</table>
</td></tr>
<tr><td>3</td><td>Antenna
Status</td><td>Bit Lamp:
GOTRFID.CHReady</td><td>
<table>
<tr><th>State</th><th>Graphic</th></tr>
<tr><td>Off</td><td>ANTENNA NOT READY</td></tr>
<tr><td>On</td><td>ANTENNA READY</td></tr>
</table>
</td></tr>
<tr><td>4</td><td>Tag
Present</td><td>Word Lamp:
31 == GOTRFID.RFIDErrorCommentNumber
GOTRFID.TagPresent</td><td>
<table>
<tr><th>State</th><th>Graphic</th></tr>
<tr><td>None</td><td>TAG NOT PRESENT</td></tr>
<tr><td>31 == GOTRFID.RFIDErrorCommentNumber</td><td></td></tr>
<tr><td>GOTRFID.TagPresent==0</td><td>TAG NOT PRESENT</td></tr>
<tr><td>GOTRFID.TagPresent==1</td><td>TAG PRESENT</td></tr>
</table>
</td></tr>
</table>

#	Description	Object	Details
5	RFID Alarming	Advanced User Alarm Display(GOTRFID.RFIDErrorCommentNumber)	COPY to USB button Executes Screen script that will copy alarms to E:\backup\Balluff.CSV Or E:\backup\OMRON.CSV Based off value contained within PLC logic needs to place a value in GOTRFID.RFIDErrorCommentNumber
6	Antenna Selection PB	Word Switch:	Numeric Input limited to the maximum number of antenna
7	One Byte Write Address And Data	Accessible when GB65101 and GOTRFID.WriteCondTrue are TRUE Address: Numeric Input GOTRFID.WriteStartAddress Limited to 0<=$W<=GOTRFID.TagSizeBytes	
8	16 Byte Start Address And Data	Numeric Input: GOTRFID.StartAddress Limited to 0<=$W<=GOTRFID.TagSizeBytes	
9	RFID Read PB	Bit Momentary: GOTRFID.ManualRead_PB	GOTRFID.ReadCondTrue must be TRUE

#	Description	Object	Details
10	Address Change PB	Word Switch: GOTRFID.StartAddress	Adds or Subtracts 16 to the Starting Address
11	RFID Write PB	Bit Momentary: GOTRFID.ManualWrite16_PB	GOTRFID.WriteCondTrue and GB65101 must be TRUE
12	Bit Enter	Byte: Bit Alternate GD64510.bit number Enter: Bit Set: GB64511	8 individual switched alternate the bits of a byte. Pressing the Enter Key executes a script that loads the value into GOTRFID.WriteData1Byte
13	Write 1-byte PB	Bit Momentary: GOTRFID.ManualWrite1_PB	

5 APPENIDX-CHANGE HISTORY

10/1/2015

Project Started based off Profinet Version 1.01

- Profibus V1.02

Acyclic communication issues corrected

- Profibus V1.03

Shared HMI design with Profinet package

- Profibus V1.04

Match Profinet Release.

- Profibus V2.00

CFG library is shared with Profinet release

Process library is shared with Profinet release

Added Configuration function blocks for all project book devices.

Created initial program to place configuration into data array.

Created Process Data function blocks for:

BIP0004

Analog

Valve

BAW002F

BOD0012

Smartlights

Standardized slot configuration per master node

Standardized HMI FB between packages

- Profibus V2.01 (7/26/2016)

Ultrasonic sensors were using an incorrect device type value. They are now type 8(.b3)