

1
2
3
4

DFI302



Fieldbus Universal Bridge





Specifications and information are subject to change without notice.
Up-to-date address information is available on our website.

web: www.smar.com/contactus.asp

AVOIDING ELECTROSTATIC DISCHARGES



ATTENTION

Electrostatic discharges may damage semiconductor electronic components in printed circuit boards. They usually occur when touching components or connector pins from modules and racks, without wearing the appropriate equipment to prevent discharges. It is recommended to take the following precautions:

- Before handling modules and racks, remove the electrostatic charge from your body by wearing a proper wristband or touching grounded devices;
- Avoid touching electronic components or connector pins from racks and modules.

INTRODUCTION

The DFI302 is a powerful multifunction hardware component integral to the modular **SYSTEM302** that includes the most up-to-date hardware and software necessary to manage, monitor, control, maintain and operate your plant. The DFI302 throughout the plant are completely self-contained and perform most of the functions required by a system; therefore very few additional components are required.

Below, there are some DFI302 features:

- Integral part of **SYSTEM302**
- Single integrated unit with functions of interfacing, linking device, bridge, controller, gateway, fieldbus power supply and distributed I/O subsystem
- Tight integration with intelligent devices and software from multiple manufacturers due to the use of open standards, such as: FOUNDATION™ fieldbus and OPC
- Connectivity with equipment through conventional I/O modules and Modbus through RS-232 or Ethernet communications
- Full redundancy and fault isolation for high safety and uninterrupted operation
- Cleanest and most cost effective architecture
- High information throughput from plant floor to the corporate network

Structure of DFI302 manual

For further information about the DFI302 automation platform more quickly, the user manual can be obtained separately by subjects, directly on the site www.smar.com as parts A through L. They are:

A – Installation and basic configuration

This part has all general and basic information about installation of controllers, racks, I/O modules, IP configuration, OPC servers, firmware update, project types, and general information about adding function blocks and flexible function blocks.

This part also has troubleshooting information, and the SRF (Service Request Form).

It has the following sections:

- Overview
- Installing
- Setting up
- Configuring the OPC servers
- Configuring strategies
- Adding function blocks
- Adding logic by using flexible function blocks
- Adding I/O modules
- Adding racks
- Troubleshooting
- Appendix (SRF)
- Appendix B - SNMP Diagnostics
- Appendix C – DF63-RTU

B – Technical Specifications

This part has all technical specification of the DFI302 hardware components. They are:

Controllers – DF51, DF62, DF63, DF73, DF79, DF81, DF89, DF95, DF97, and DF100
 Ethernet cables, serial cables and cables for racks' interconnection
 Power supplies – DF50, DF49/53, DF52/60, DF56, and DF87
 Intrinsic safety barriers – DF47-12 and DF47-17
 Interfaces – DF58 and DF61

It has the following sections:

- Technical specification for controllers
- Cable specification
- Adding power supplies

- Adding interfaces

C – Modbus

This part has information to integrate systems that use Modbus protocol to the DFI302 automation platform. It has the following section:

- Adding Modbus

D – DF51 Platform

This part has information to create control strategies and redundant systems that use the DF51 controller.

This part also has information about redundant system architecture and its configuration, hot standby redundancy and LAS redundancy. It has the following sections:

- Creating a fieldbus strategy by using DF51
- Adding redundancy to DF51 controller

E – DF65 Platform

This part has information about the DF65 logic coprocessor which is connected to DF51 (controller of DFI302 automation platform) to improve its capacity for discrete control. The ladder logic and the coprocessor function blocks help and improve the Fieldbus control system.

It has the following section:

- Adding logic configuration by using coprocessors modules

F – Configuring strategies with HSE/FF controllers

This part has information to create control strategies that use the DF62 or DF63 controllers.

It has the following section:

- Creating a FOUNDATION fieldbus strategy by using the DF62/DF63

G - Configuring strategies with HSE/PROFIBUS controllers

This part has information to create control strategies that use the DF73, DF95 or DF97 controllers.

The DF73 is the HSE/Profibus-DP controller with 2 Ethernet 100 Mbps ports, and 1 Profibus DP channel.

The DF95 is the HSE/Profibus controller with 2 Ethernet 100 Mbps ports, 1 serial port, 2 Profibus PA ports, and 1 Profibus DP channel.

The DF97 is the HSE/Profibus controller with 2 Ethernet 100 Mbps ports, 1 serial port, 4 Profibus PA ports, and 1 Profibus DP channel.

It has the following section:

- Creating a Profibus configuration by using DF73, DF95 or DF97

H - Configuring strategies with HSE/DeviceNet controllers

This part has information to create control strategies that use the DF79 controller.

The DF79 is the HSE/DeviceNet controller with 2 Ethernet 100 Mbps ports and 1 DeviceNet channel.

It has the following section:

- Creating a DeviceNet configuration by using DF79

I - Configuring strategies with HSE/AS-i controllers

This part has information to create control strategies that use the DF81 controller.

The DF81 is the HSE/AS-i controller with 2 Ethernet 100 Mbps ports, and 2 AS-i channels.

It has the following section:

- Creating an AS-i configuration by using DF81

J - Configuring strategies with HSE/Modbus controllers

This part has information to create control strategies that use the DF89 controller.

The DF89 is the HSE/Modbus controller with 2 Ethernet 100 Mbps ports and 1 RS-485/RS-232 Modbus RTU port.

It has the following section:

- Creating a Modbus configuration by using the DF89

K - Configuring strategies with HSE/WirelessHART controllers

This part has information to create control strategies that use the DF100 controller.

The DF100 is the HSE/WirelessHART™ controller with 2 Ethernet 100 Mbps ports, 1 RS-485 port and 1 WirelessHART channel.

It has the following section:

- Creating a FOUNDATION fieldbus strategy by using the DF100

L – Redundancy

To meet the requirements for fault tolerance, system availability and safety of the industrial process, the DFI302 HSE controllers (DF62/DF63/DF73/DF75/DF89) work with a Hot Standby redundancy strategy.

This part has information about installation and configuration of redundant systems which use those controllers.

It has the following sections:

- Adding redundancy to DFI302 HSE controllers
- Adding redundancy with redundant I/O modules R-Series

M – Conventional and redundant input/output modules

There are many types of input/output modules available for DFI302 automation platform, designed to fit a broad range of applications in the automation and process control industry.

The available types are:

- Discrete inputs and outputs
- Combined discrete inputs and outputs
- Pulse inputs
- Analog inputs and outputs
- HART inputs and outputs
- Redundant inputs and outputs

The Digital and Analog Input/Output Modules of DFI302 manual has all information about those modules. This part of manual can be obtained on the Smar's website (www.smar.com), downloading the file MESDAME.pdf.

N - Interfaces for panels

With the Smar's panel interfaces is possible to eliminate the hard work of making cables, fixing washers and mounting terminal blocks. Just fit the interfaces in the DIN rail and connect the cable. It's easy and fast!

The interfaces have many features which will fit your application. They were designed for the Smar's I/O modules.

The Interfaces for Panels manual has all necessary information. This part of manual can be obtained on the Smar's website (www.smar.com), downloading the file ITFPANELME.pdf.

TABLE OF CONTENTS

SECTION 1 - OVERVIEW.....	1.1
AVAILABLE MODULES FOR THE DFI302 SYSTEM	1.2
MAIN FEATURES.....	1.6
DISTRIBUTED ARCHITECTURE.....	1.6
HIGH RELIABILITY	1.6
CONFIGURATION.....	1.7
SUPERVISION	1.7
SYSTEM INTEGRATION.....	1.7
REDUNDANCY.....	1.8
EXPANDABLE	1.8
SECTION 2 - INSTALLING.....	2.1
RACKS, CABLES AND ACCESSORIES OF DFI302 SYSTEM	2.1
INSTALLING THE SYSTEM'S BASE WITH DF92 AND DF93 RACKS	2.2
INSTALLING RACKS - DF92 AND DF93	2.3
INSTALLING THE EXPANSION FLAT CABLES - DF101, DF102, DF103, DF104 AND DF105.....	2.5
FLAT CABLES PROTECTOR (CONNECTOR CAP)	2.6
INSTALLING THE IMB TERMINATOR - DF2 OR DF96	2.7
EXPANDING THE SYSTEM'S POWER - DF90 AND DF91.....	2.8
DIAGNOSTIC RESOURCES.....	2.11
INSTALLING THE SYSTEM'S BASE WITH DF1A AND DF78	2.13
INSTALLING A RACK IN THE DIN RAIL	2.14
ADDING RACKS	2.14
TIPS FOR ASSEMBLING.....	2.15
IMPROVING SIGNAL GROUND OF DFI302 (DF1A AND DF78 RACKS)	2.15
NON-ADJACENT RACKS	2.16
ADJACENT RACKS	2.16
INSTALLING MODULES IN THE RACK	2.17
INSTALLING THE HARDWARE	2.18
USING THE DF51 CONTROLLER.....	2.18
USING THE DF62/DF63 CONTROLLER	2.20
USING THE DF73 CONTROLLER.....	2.22
USING THE DF75 CONTROLLER.....	2.25
USING THE DF79 CONTROLLER.....	2.27
USING THE DF81 CONTROLLER.....	2.29
USING THE DF89 CONTROLLER.....	2.33
USING THE DF95 CONTROLLER.....	2.35
USING THE DF97 CONTROLLER.....	2.37
USING THE DF100 CONTROLLER.....	2.39
DIMENSIONAL DRAWINGS OF DF1A RACK AND MODULES	2.57
DIMENSIONAL DRAWINGS OF DF93 AND MODULES	2.58
DIMENSIONAL DRAWING OF DF100	2.59
INSTALLING THE STUDIO302	2.60
GETTING LICENSE FOR DFI302 SERVERS	2.60
SECTION 3 - SETTING UP.....	3.1
DFI OLE SERVER AND HSE OLE SERVER SETTINGS	3.1
CONNECTING THE DFI302 IN THE SUBNET	3.1
FOR NETWORK WITH DHCP	3.1
FOR NETWORK WITHOUT DHCP	3.1
UPDATING THE FIRMWARE.....	3.8
DFI DOWNLOAD CLASSIC	3.8
BATCH DOWNLOAD	3.11
CHANGING IP ADDRESS.....	3.16
CHANGING IP CONTROLLER	3.16
CONFIGURING THE RTC.....	3.22
SECTION 4 - CONFIGURING THE OPC SERVERS.....	4.1
CONFIGURING THE OPC SERVERS	4.1
INTRODUCTION	4.1

CLIENT / SERVER ARCHITECTURE VIA OPC	4.1
OPC COMPLIANT	4.1
OLE FOR FIELDBUS CONFIGURATION (OFC)	4.1
OPC – OLE FOR PROCESS CONTROL	4.1
OVERVIEW	4.2
LOCAL SERVERS AND REMOTE SERVERS	4.2
MINIMUM DCOM SETTINGS	4.2
RUNNING DCOMCONFIG.EXE	4.3
CONFIGURING THE GENERAL/DEFAULTS OPTIONS	4.3
CONFIGURING THE OPCENUM PARAMETERS	4.5
CONFIGURING OPC SERVER PARAMETERS (HSE OR DFI)	4.9
CONFIGURING LOCAL SECURITY POLICIES	4.9
CONFIGURING WINDOWS FIREWALL	4.13
DFI OLE SERVER DETAILS	4.13
HSE OLE SERVER DETAILS	4.13
A&E OPC SERVER DETAILS	4.13
HDA OPC SERVER DETAILS	4.14
HSE DEVICE DEFINITION	4.14
INFORMATION FOR FIREWALL CONFIGURATION	4.14
SMAROLESERVER.INI CONFIGURATION	4.15
TOPOLOGY UPLOAD	4.16
SYSTEM302 SERVERMANAGER APPLICATION	4.16
OPTIMIZING DF51 ACCESS TO SUBNETS	4.17
ENABLING THE SYNCHRONISM BY SNTP	4.18
ENABLING THE SYNCHRONISM BY SNTP IN THE DF51	4.18
ENABLING THE SYNCHRONISM BY SNTP IN THE DF62, DF63, DF73, DF75, DF79, DF81, DF89, DF95, DF97 AND DF100 CONTROLLERS	4.21
SECTION 5 - CONFIGURING STRATEGIES	5.1
INTRODUCTION	5.1
AREA TYPES	5.2
AREA	5.2
HSE AREA	5.3
PREDEFINED AREA	5.4
STRATEGY TEMPLATE	5.5
DEVICE TEMPLATE	5.5
BRIDGE TEMPLATE	5.5
CONTROLLER TEMPLATE	5.6
SECTION 6 - ADDING FUNCTION BLOCKS	6.1
INTRODUCTION	6.1
CREATING A NEW BLOCK	6.1
ATTACHING NEW BLOCK	6.3
SECTION 7 - ADDING LOGIC BY USING FLEXIBLE FUNCTION BLOCKS (FFB 1131)	7.1
INTRODUCTION	7.1
AREA WITH FFB 1131	7.2
ARRANGING THE SYSCON WINDOWS	7.4
DEFINING THE FFB PARAMETERS	7.4
SECTION 8 - ADDING I/O MODULES	8.1
INTRODUCTION	8.1
STEPS TO CONFIGURE I/O MODULES	8.4
RES – RESOURCE BLOCK	8.5
HCT – HARDWARE CONFIGURATION TRANSDUCER	8.5
TEMP – TEMPERATURE TRANSDUCER	8.6
TBH – RIO HART TRANSDUCER BLOCK	8.8
FUNCTION BLOCK CREATION	8.12
CHANNEL CONFIGURATION	8.12
MODULE SPECIFICATION STANDARD	8.14
SECTION 9 - ADDING RACKS	9.1
DF1A – RACK WITH 4 SLOTS	9.1

DESCRIPTION	9.1
TECHNICAL SPECIFICATIONS	9.1
DF78 - RACK WITH 4 SLOTS FOR REDUNDANT CPUS	9.2
DESCRIPTION	9.2
TECHNICAL SPECIFICATIONS	9.2
DF93 - RACK WITH 4 SLOTS (WITH DIAGNOSTIC)	9.3
DESCRIPTION	9.3
TECHNICAL SPECIFICATIONS	9.3
DF92 - RACK WITH 4 SLOTS FOR REDUNDANT CPUS (WITH DIAGNOSTIC SUPPORT)	9.5
DESCRIPTION	9.5
TECHNICAL SPECIFICATIONS	9.5
SECTION 10 - TROUBLESHOOTING	10.1
WHEN TO USE THE PROCEDURES OF FACTORY INIT/RESET	10.3
SECTION 11 - TECHNICAL SPECIFICATIONS FOR THE CONTROLLERS	11.1
DFI302 SPECIFICATIONS	11.1
DF51 SPECIFICATIONS	11.2
PART NUMBER	11.2
DESCRIPTION	11.2
TECHNICAL SPECIFICATIONS	11.2
USING THE FAULT INDICATION	11.3
JUMPERS ON BOARD	11.4
FIELDBUS LIMITS	11.4
SUPERVISION LIMITS	11.5
MODBUS LIMITS	11.5
DF62 SPECIFICATIONS	11.6
PART NUMBER	11.6
DESCRIPTION	11.6
CHARACTERISTICS AND CONTROLLER LIMITS	11.6
CONTINUOUS CONTROL WITH FOUNDATION™ FIELDBUS	11.7
DISCRETE CONTROL	11.7
FLEXIBLE FUNCTION BLOCK USAGE	11.7
FIRMWARE VERSION AND DEVICE REVISION	11.7
TECHNICAL SPECIFICATIONS	11.8
ELECTRICAL CERTIFICATION	11.9
INDICATION LEDS	11.11
DF63 SPECIFICATIONS	11.12
PART NUMBER	11.12
DESCRIPTION	11.12
CHARACTERISTICS AND CONTROLLER LIMITS	11.12
CONTINUOUS CONTROL WITH FOUNDATION™ FIELDBUS	11.13
DISCRETE CONTROL	11.13
FLEXIBLE FUNCTION BLOCK USAGE	11.14
FIRMWARE VERSION AND DEVICE REVISION	11.14
TECHNICAL SPECIFICATIONS	11.14
ELECTRICAL CERTIFICATION	11.17
FIRMWARE UPDATE WITH FBTOOLS	11.18
INDICATION LEDS	11.19
DF73 SPECIFICATIONS	11.20
PART NUMBER	11.20
DESCRIPTION	11.20
CHARACTERISTICS AND MODULE LIMITS	11.20
CONTINUOUS CONTROL WITH PROFIBUS	11.21
DISCRETE CONTROL	11.21
FIRMWARE VERSION AND DEVICE REVISION	11.21
TECHNICAL SPECIFICATIONS	11.22
ELECTRICAL CERTIFICATION	11.24
INDICATION LEDS	11.25
DF75 SPECIFICATIONS	11.27
PART NUMBER	11.27
DESCRIPTION	11.27
CHARACTERISTICS AND CONTROLLER LIMITS	11.27
CONTINUOUS CONTROL WITH FOUNDATION™ FIELDBUS	11.28
DISCRETE CONTROL	11.28

FLEXIBLE FUNCTION BLOCK USAGE	11.28
FIRMWARE VERSION AND DEVICE REVISION	11.28
TECHNICAL SPECIFICATIONS	11.29
ELECTRICAL CERTIFICATION	11.30
INDICATION LEDS	11.32
DF79 SPECIFICATIONS	11.33
PART NUMBER	11.33
DESCRIPTION	11.33
CHARACTERISTICS AND MODULE LIMITS	11.33
CONTINUOUS CONTROL WITH DEVICENET	11.34
DISCRETE CONTROL	11.34
FIRMWARE VERSION AND DEVICE REVISION	11.34
TECHNICAL SPECIFICATIONS	11.35
ELECTRICAL CERTIFICATION	11.37
INDICATION LEDS	11.38
DF81 SPECIFICATIONS	11.39
PART NUMBER	11.39
DESCRIPTION	11.39
CHARACTERISTICS AND MODULE LIMITS	11.39
CONTINUOUS CONTROL WITH AS-I	11.40
DISCRETE CONTROL	11.40
FLEXIBLE FUNCTION BLOCK USAGE	11.41
FIRMWARE VERSION AND DEVICE REVISION	11.41
TECHNICAL SPECIFICATIONS	11.41
ELECTRICAL CERTIFICATION	11.43
INDICATION LEDS	11.45
DF89 SPECIFICATIONS	11.46
PART NUMBER	11.46
DESCRIPTION	11.46
CHARACTERISTICS AND CONTROLLER LIMITS	11.46
CONTINUOUS CONTROL WITH FOUNDATION™ FIELDBUS	11.47
DISCRETE CONTROL	11.47
FLEXIBLE FUNCTION BLOCK USAGE	11.48
FIRMWARE VERSION AND DEVICE REVISION	11.48
TECHNICAL SPECIFICATIONS	11.48
ELECTRICAL CERTIFICATION	11.50
INDICATION LEDS	11.51
DF95 SPECIFICATIONS	11.52
PART NUMBER	11.52
DESCRIPTION	11.52
CHARACTERISTICS AND MODULE LIMITS	11.52
CONTINUOUS CONTROL WITH PROFIBUS	11.53
DISCRETE CONTROL	11.53
FIRMWARE VERSION AND DEVICE REVISION	11.53
TECHNICAL SPECIFICATIONS	11.54
M12 CONNECTOR	11.56
INDICATION LEDS	11.57
DF97 SPECIFICATIONS	11.59
PART NUMBER	11.59
DESCRIPTION	11.59
CHARACTERISTICS AND MODULE LIMITS	11.59
CONTINUOUS CONTROL WITH PROFIBUS	11.60
DISCRETE CONTROL	11.60
FIRMWARE VERSION AND DEVICE REVISION	11.60
TECHNICAL SPECIFICATIONS	11.61
M12 CONNECTOR	11.63
INDICATION LEDS	11.64
DF100 SPECIFICATIONS	11.66
DESCRIPTION	11.66
GENERAL CHARACTERISTICS AND LIMITS	11.67
CONTINUOUS CONTROL WITH FOUNDATION™ FIELDBUS	11.67
HSE WIO FUNCTION BLOCKS AND TRANSDUCERS	11.67
FIRMWARE VERSION AND DEVICE REVISION	11.68
TECHNICAL SPECIFICATIONS	11.68
INDICATION LEDS	11.70
LEDS RELATED TO WIRELESSHART MANAGER	11.71

HARDWARE CONFIGURATION.....	11.71
SECTION 12 - CABLE SPECIFICATIONS	12.1
ETHERNET CABLE SPECIFICATIONS	12.1
DF54/DF55.....	12.1
SERIAL CABLE SPECIFICATIONS	12.2
DF59.....	12.2
DF68.....	12.3
DF82.....	12.4
DF83.....	12.4
CABLES FOR RACKS INTERCONNECTION AND POWER DISTRIBUTION	12.5
EXPANSION FLAT CABLES FOR SYSTEMS BASED ON DF92 AND DF93.....	12.5
FLAT CABLES PROTECTOR (CONNECTOR CAP)	12.5
DF90 CABLE	12.6
SECTION 13 - ADDING POWER SUPPLIES.....	13.1
INTRODUCTION	13.1
DF50 – POWER SUPPLY MODULE FOR BACKPLANE (REDUNDANT)	13.2
DESCRIPTION	13.2
INSTALLATION AND CONFIGURATION	13.2
TECHNICAL SPECIFICATIONS	13.3
DF56 – POWER SUPPLY FOR BACKPLANE (REDUNDANT)	13.5
DESCRIPTION	13.5
INSTALLATION AND CONFIGURATION	13.5
TECHNICAL SPECIFICATIONS	13.6
DF87 – POWER SUPPLY FOR BACKPLANE (5 A, REDUNDANT, WITH DIAGNOSTIC)	13.8
DESCRIPTION	13.8
INSTALLATION AND CONFIGURATION	13.8
TECHNICAL SPECIFICATIONS	13.9
DIAGNOSTICS LEDS	13.10
CALCULATING THE POWER CONSUMPTION	13.12
POWER SUPPLIES AND RACKS POSITIONING	13.13
FOR SYSTEMS BASED ON DF92 AND DF93 RACKS WITH DF90 AND DF91	13.13
FOR SYSTEMS BASED ON DF78 AND DF1A RACKS (LEGACY SYSTEMS).....	13.15
FAULT DIAGNOSTIC IN IMB	13.16
CONDITIONS THAT MAY CAUSE IMB FAULT STATE:	13.16
DF52 / DF60 – POWER SUPPLY FOR FIELDBUS	13.17
DESCRIPTION	13.17
TECHNICAL SPECIFICATIONS	13.18
DF53/DF98 – POWER SUPPLY IMPEDANCE FOR FIELDBUS	13.20
DESCRIPTION	13.20
TECHNICAL SPECIFICATIONS	13.21
INSTALLATION.....	13.22
MAINTENANCE AND TROUBLESHOOTING	13.22
DF47-12 AND DF47-17 – INTRINSIC SAFETY BARRIER FOR FIELDBUS	13.23
DESCRIPTION	13.23
INSTALLATION.....	13.23
TECHNICAL SPECIFICATIONS	13.25
CERTIFICATION INFORMATION	13.26
HAZARDOUS LOCATIONS APPROVALS.....	13.29
IDENTIFICATION LABELS AND CONTROL DRAWINGS	13.32
SECTION 14 - ADDING INTERFACES.....	14.1
INTRODUCTION	14.1
DF58 – RS-232/RS-485 INTERFACE	14.2
DESCRIPTION	14.2
INTERFACE SETTINGS	14.2
RS-232 MODE: HALF-DUPLEX/FULL-DUPLEX.....	14.2
RS-485 BUS TERMINATOR: ON/OFF	14.2
CONNECTORS.....	14.3
TECHNICAL SPECIFICATIONS	14.3
DF61 – ETHERNET SWITCH 10/100 MBPS	14.4

SECTION 15 - ADDING MODBUS.....	15.1
INTRODUCTION	15.1
STEPS TO CONFIGURE MODBUS.....	15.3
MBCF - PARAMETERS DESCRIPTION.....	15.4
MBCS – MODBUS CONTROL SLAVE.....	15.7
PARAMETERS DESCRIPTION	15.8
MBSS – MODBUS SUPERVISION SLAVE.....	15.11
PARAMETERS DESCRIPTION	15.12
DATA TYPE AND SUPPORTED STRUCTURES BY MBSS.....	15.14
MBCM – MODBUS CONTROL MASTER.....	15.15
PARAMETERS DESCRIPTION	15.16
MBSM – MODBUS SUPERVISION MASTER.....	15.21
PARAMETERS DESCRIPTION	15.22
MODBUS SLAVE ADDRESSES	15.27
MODBUS COMMANDS	15.29
SCALING CONVERSION	15.30
REDUNDANCY AND MODBUS	15.31
USING MODBUS IN CONTROLLERS DF73, DF75, DF79, DF81, DF89, DF95, DF97 AND DF63-MBS	15.33
PARAMETERS DESCRIPTION	15.34
TROUBLESHOOTING.....	15.35
 SECTION 16 - CREATING A FIELDBUS STRATEGY BY USING DF51.....	 16.1
INTRODUCTION	16.1
PROJ_DF51.....	16.1
STARTING THE AREA	16.2
PHYSICAL PLANT PROJECT.....	16.3
ARRANGING FIELDBUS WINDOWS	16.4
ADDING BRIDGES	16.5
ADDING FIELDBUS DEVICES	16.6
ADDING FUNCTION BLOCKS.....	16.7
CREATING NEW PROCESS CELLS.....	16.9
CREATING A CONTROL MODULE (FBAPPLICATION)	16.10
INSERTING BLOCKS IN THE CONTROL MODULE	16.11
CONFIGURING THE CONTROL STRATEGY	16.12
ADDING BLOCKS TO THE STRATEGY WINDOW	16.12
LINKING THE BLOCKS	16.13
FUNCTION BLOCK CHARACTERIZATION.....	16.14
OPTIMIZING THE SUPERVISION	16.18
BACKGROUND TIME	16.19
MVC (MULTIPLE VARIABLE CONTAINERS).....	16.20
SUPERVISION TIME	16.21
UPDATE TIME	16.21
OPC UPDATE RATE.....	16.22
 SECTION 17 - ADDING REDUNDANCY TO THE DF51 CONTROLLER.....	 17.1
INTRODUCTION	17.1
HOT STANDBY REDUNDANCY	17.1
LINK ACTIVE SCHEDULER (LAS) REDUNDANCY	17.2
REDUNDANT SYSTEM ARCHITECTURE	17.2
SYSTEM PRE-REQUIREMENTS.....	17.3
CONFIGURING HOT STANDBY REDUNDANCY	17.3
FIRST TIME CONFIGURATION PROCEDURE	17.6
CHANGING THE CONFIGURATION	17.6
REPLACING A MODULE WITH FAILURE	17.6
FIXING THE SYSTEM WHEN THE H1 CABLE BREAKS.....	17.7
FIRMWARE UPDATE WITHOUT PROCESS INTERRUPTION.....	17.7
ADDING REDUNDANCY TO A SYSTEM IN OPERATION.....	17.7
CONFIGURING LAS REDUNDANCY	17.8
FIRST TIME CONFIGURATION PROCEDURE	17.8
REPLACING AN ACTIVE MODULE WITH FAILURE	17.9
REPLACING A BACKUP MODULE WITH FAILURE	17.9
PLACING THE SYSTEM INTO OPERATION AFTER A GENERAL POWER FAILURE.....	17.10
FIXING THE SYSTEM WHEN THE H1 CABLE BREAKS.....	17.10
FIRMWARE UPDATE WITHOUT PROCESS INTERRUPTION.....	17.10

SECTION 18 - ADDING LOGIC CONFIGURATION USING CO-PROCESSORS MODULES	18.1
INTRODUCTION	18.1
DF65 CONFIGURATION	18.1
SERIAL COMMUNICATION SETTINGS	18.2
PHYSICAL LAYER AND TIMEOUT	18.2
CHANGING DF65 COMMUNICATION SETTINGS	18.3
LOGIC CONFIGURATION DOWNLOAD	18.3
CONFIGURING DF51 MODBUS BLOCKS	18.4
SUPERVISING DF65 CO-PROCESSOR DATA USING MBSM BLOCK	18.4
DATA EXCHANGE BETWEEN DF65 CO-PROCESSOR AND DF51 USING MBCM BLOCK	18.4
EXAMPLE OF COMMUNICATION BETWEEN DF51 AND DF65 WITH LADDER LOGIC	18.5
HOW TO CONFIGURE THE COMMUNICATION AND DATA EXCHANGE BETWEEN THE DF65 AND THE DF51	18.6
DF65 – CO-PROCESSOR MODULE	18.7
DESCRIPTION	18.7
TECHNICAL SPECIFICATIONS	18.8
COMMUNICATION CHANNELS	18.9
RESTRICTIONS:	18.9
DEVICE COMMUNICATION BAUD RATE AND DEVICE ADDRESS	18.11
OPERATION MODES	18.11
DF65 WITH THREE MODBUS RTU CHANNELS	18.12
DF65 MASTER IN A DF66 SYSTEM	18.12
FACTORY INIT	18.12
DF66 – REMOTE I/O COMMUNICATION INTERFACE	18.13
DESCRIPTION	18.13
ADDING A DF66 UNIT	18.13
REMOTE I/O ARCHITECTURE	18.14
BAUD RATE AND ADDRESS SETTING	18.14
DF65R/DF65ER – REDUNDANT CO-PROCESSOR MODULE	18.15
INTRODUCTION	18.15
TERMINOLOGY AND INITIAL DESCRIPTIONS	18.15
ARCHITECTURE	18.17
POWER UP PROCEDURES	18.18
COMMUNICATION WITH REMOTE INPUT AND OUTPUT (RIO) MODULES	18.20
LEDS FOR STATUS INDICATION	18.20
SECTION 19 - CREATING A FOUNDATION FIELDBUS STRATEGY BY USING DF62/DF63	19.1
INTRODUCTION	19.1
PROJ_DF62	19.1
STARTING THE AREA	19.2
PHYSICAL PLANT PROJECT	19.3
ARRANGING THE FIELDBUS WINDOWS	19.4
ADDING THE BRIDGE	19.4
ADDING FIELDBUS DEVICES	19.6
ADDING FUNCTION BLOCKS	19.8
CREATING NEW PROCESS CELLS	19.10
CREATING A CONTROL MODULE	19.11
INSERTING BLOCKS IN THE CONTROL MODULE	19.13
CONFIGURING THE CONTROL STRATEGY	19.14
ADDING BLOCKS TO THE STRATEGY WINDOW	19.14
LINKING THE BLOCKS	19.15
FUNCTION BLOCK CHARACTERIZATION	19.16
MACROCYCLE FOR THE H1 CHANNEL	19.21
BACKGROUND TIME	19.21
SECTION 20 - CREATING A PROFIBUS CONFIGURATION BY USING DF73, DF95 OR DF97	20.1
INTRODUCTION	20.1
PROJ_DF73	20.1
STARTING THE AREA	20.4
PHYSICAL PLANT PROJECT	20.5
ARRANGING THE FIELDBUS WINDOWS	20.6
INSERTING THE CONTROLLER	20.6
ADDING PROFIBUS DEVICES	20.8
INSERTING SLAVE DEVICES THAT ARE NO PRESENT IN THE "AVAILABLE SLAVES" LIST	20.10
SETTING PROFIBUS DEVICES	20.10

MAPPING PROFIBUS IO POINTS TO BE USED IN LADDER	20.18
MAPPING PROFIBUS IO POINTS TO BE USED IN FUNCTION BLOCKS	20.23
ADDING OTHER FUNCTION BLOCKS	20.26
CREATING NEW PROCESS CELLS	20.27
CREATING A CONTROL MODULE	20.29
INSERTING BLOCKS TO THE CONTROL MODULE	20.30
CONFIGURING THE CONTROL STRATEGY	20.30
ADDING BLOCKS IN THE STRATEGY WINDOW	20.32
LINKING THE BLOCKS	20.32
BLOCK CHARACTERIZATION	20.33
COMMISSIONING AND CONFIGURATION DOWNLOAD TO THE CONTROLLER	20.37
SYSTEM302 MAINTENANCE PROCEDURE	20.39
PROFIBUS NETWORK TIME PARAMETERS	20.40
DEFAULT VALUES OF THE PROFIBUS STANDARD	20.42
RECOMMENDED VALUES	20.43
REQUIRED VALUES WHEN USING THIRD PARTY DEVICES	20.43
NETWORK DIAGNOSTICS	20.44
NETWORK DIAGNOSTIC USING THE NETWORK CONFIGURATOR	20.44
EXTENDED DIAGNOSTIC IN THE NETWORK CONFIGURATOR	20.47
SECTION 21 - CREATING A DEVICENET CONFIGURATION BY USING DF79	21.1
INTRODUCTION	21.1
PROJ_DF79	21.1
STARTING THE AREA	21.2
PHYSICAL PLANT PROJECT	21.3
ARRANGING THE FIELDBUS WINDOWS	21.4
INSERTING THE CONTROLLER	21.5
ADDING DEVICENET DEVICES	21.6
INSERTING SLAVE DEVICES THAT ARE NO PRESENT IN THE "AVAILABLE SLAVES" LIST	21.8
SETTING DEVICENET DEVICES	21.9
MAPPING DEVICENET IO POINTS TO BE USED IN LADDER	21.14
MAPPING DEVICENET IO POINTS TO BE USED IN FUNCTION BLOCKS	21.18
ADDING OTHER FUNCTION BLOCKS	21.22
CREATING NEW AREAS	21.23
CREATING A CONTROL MODULE	21.24
INSERTING BLOCKS TO THE CONTROL MODULE	21.25
CONFIGURING THE CONTROL STRATEGY	21.26
ADDING BLOCKS IN THE STRATEGY WINDOW	21.27
LINKING THE BLOCKS	21.27
BLOCK CHARACTERIZATION	21.29
COMMISSIONING AND CONFIGURATION DOWNLOAD TO THE CONTROLLER	21.31
SYSTEM302 MAINTENANCE PROCEDURE	21.33
DEVICENET ADVANCED TOPICS	21.34
I/O MESSAGES CONFIGURATION IN THE NETWORK CONFIGURATOR	21.34
NETWORK CONFIGURATOR ONLINE COMMUNICATION WITH THE CONTROLLER	21.35
EXPLICIT MESSAGE CONFIGURATION	21.37
ONLINE READING/WRITING OF SLAVE'S PARAMETERS	21.38
CHANGING THE ADDRESS VIA SOFTWARE	21.39
ENABLING AND DISABLING A DEVICE IN THE CONFIGURATION	21.40
NETWORK DIAGNOSTICS	21.42
NETWORK DIAGNOSTIC USING THE NETWORK CONFIGURATOR	21.42
COMMON ERRORS OF COMMISSIONING	21.51
THE AUTO DETECTED DEVICES	21.53
DEVICES DIAGNOSTIC THROUGH THEIR LEDS	21.54
DIAGNOSTIC THROUGH TRANSDUCER BLOCK	21.55
DEVICENET CONTROLLER SPECIFIC BLOCKS	21.58
DEVICENET COMMUNICATION TRANSDUCER	21.58
SECTION 22 - CREATING AN AS-I CONFIGURATION BY USING DF81	22.1
INTRODUCTION	22.1
PROJ_DF81	22.1
STARTING THE AREA	22.2
PHYSICAL PLANT PROJECT	22.3
ARRANGING THE FIELDBUS WINDOWS	22.4
INSERTING THE CONTROLLER	22.4

ADDING AS-I DEVICES	22.6
INSERTING DEVICES IN THE TOPOLOGY	22.8
PERFORMING THE UPLOAD OF DETECTED DEVICE	22.10
INSERTING NEW AS-I DEVICES IN STANDARD LIST	22.12
CONFIGURING THE AS-I DEVICES	22.12
MAPPING AS-I I/O POINTS TO BE USED IN LADDER	22.14
NETWORK DIAGNOSTICS	22.18
NETWORK DIAGNOSTIC USING THE NETWORK CONFIGURATOR	22.18
NETWORK DIAGNOSTIC USING THE COMMUNICATION TRANSDUCER BLOCK	22.20
NETWORK DIAGNOSTIC USING THE LEDS' CONTROLLER	22.21
SPECIFIC BLOCKS OF AS-I CONTROLLER	22.21
AS-I COMMUNICATION TRANSDUCER	22.21
SECTION 23 - CREATING A MODBUS CONFIGURATION BY USING DF89	23.1
INTRODUCTION	23.1
CREATING A BASIC AREA	23.2
DF89 CONFIGURATION AS MODBUS SERIAL MASTER	23.6
MODBUS NETWORK CONFIGURATION	23.7
CONFIGURING MODBUS DEVICES	23.8
HOW TO EDIT THE DEVICES PROPERTIES	23.9
ADDING COMMANDS TO THE DEVICE	23.9
MAPPING THE MODBUS POINTS FOR THE SYSTEM302 LOGIC	23.11
DF89 CONFIGURATION AS MODBUS TCP MASTER	23.15
CONFIGURING THE MODBUS DEVICES	23.15
EDITING THE DEVICES PROPERTIES AND COMMANDS	23.16
MAKING LOGIC WITH THE MODBUS I/O POINTS	23.17
MAPPING THE MODBUS POINTS TO BE USED ON LADDER	23.17
MAPPING THE MODBUS POINTS TO BE USED IN FUNCTION BLOCKS	23.19
COMMISSIONING AND CONFIGURATION DOWNLOAD TO THE CONTROLLER	23.20
DF89 CONFIGURATION AS MODBUS SLAVE	23.22
DF89 CONFIGURATION AS BYPASS MODBUS ONLY	23.23
ADVANCED MODBUS TOPICS	23.24
TYPES OF DATA SUPPORTED	23.25
SUPERVISION AND CONTROL USING DF89	23.26
SCAN TIME BETWEEN COMMANDS	23.27
OPTIMIZATION OF COMMANDS	23.27
ENABLING AND DISABLING A DEVICE IN THE CONFIGURATION	23.27
NETWORK CONFIGURATOR ONLINE COMMUNICATION WITH THE CONTROLLER	23.28
SYSTEM302 MAINTENANCE PROCEDURE	23.29
SMAR NETWORK CONFIGURATOR PARAMETERS	23.30
SERIAL CHANNEL PARAMETERS	23.30
TCP CHANNEL PARAMETERS	23.30
SERIAL DEVICE PARAMETERS	23.31
TCP EQUIPAMENT PARAMETERS	23.31
PARAMETERS OF DEVICE COMMANDS (SERIAL OR TCP)	23.32
SPECIFIC BLOCKS OF DF89 CONTROLLER	23.33
MODBUS COMMUNICATION TRANSDUCER	23.33
MODBUS SLAVE CONFIGURATION	23.35
REDUNDANCY AND MODBUS	23.36
SECTION 24 - CREATING A FOUNDATION FIELDBUS STRATEGY BY USING THE DF100	24.1
INTRODUCTION	24.1
PROJ_DF100	24.1
STARTING AN AREA	24.2
PHYSICAL PLANT PROJECT	24.3
ARRANGING THE FIELDBUS WINDOWS	24.4
ADDING THE CONTROLLER	24.4
ADDING HSE WIO TRANSDUCER AND FUNCTION BLOCKS	24.6
ADDING CONVENTIONAL FUNCTION BLOCKS	24.13
CREATING NEW PROCESS CELLS	24.14
CREATING A CONTROL MODULE	24.15
INSERTING BLOCKS IN THE CONTROL MODULE	24.16
CONFIGURING THE CONTROL STRATEGY	24.17
ADDING BLOCKS TO THE STRATEGY WINDOW	24.18

LINKING THE BLOCKS	24.18
FUNCTION BLOCK CHARACTERIZATION.....	24.20
DF100 CONVENTIONAL FUNCTION BLOCKS.....	24.22
DF100 SPECIFIC BLOCKS	24.23
MODBUS PROTOCOL SUPPORT.....	24.24
SUPPORTED CHARACTERISTICS	24.24
NATIVE MAPPING.....	24.24
MODBUS COMBINED SCENARIO	24.28
SECTION 25 - ADDING REDUNDANCY TO THE DFI302 HSE CONTROLLERS.....	25.1
INTRODUCTION	25.1
HOT STANDBY REDUNDANCY.....	25.1
PREPARING A REDUNDANT SYSTEM	25.2
ETHERNET NETWORK ARCHITECTURES.....	25.2
CONFIGURING THE SERVER MANAGER AND SYSCON.....	25.5
SYNCHRONISM CHANNEL.....	25.6
FOUNDATION FIELDBUS H1 CHANNELS	25.7
ACCESSING THE I/O BUS	25.7
HOT STANDBY REDUNDANCY WORKING	25.8
STARTING UP THE REDUNDANCY	25.8
OPERATIONAL TRANSPARENCY.....	25.8
SWITCH OVER CONDITIONS.....	25.9
STANDBY LED BEHAVIOR	25.11
PROCEDURES FOR HOT STANDBY REDUNDANCY	25.12
CONFIGURING FOR THE FIRST TIME A REDUNDANT SYSTEM	25.12
CHANGING THE CONFIGURATION.....	25.13
REPLACING A CONTROLLER WITH FAILURE.....	25.13
ADDING REDUNDANT CONTROLLERS IN A NON-REDUNDANT SYSTEM	25.13
FIRMWARE UPDATE WITHOUT PROCESS INTERRUPTION.....	25.13
TROUBLESHOOTING.....	25.14
SECTION 26 - ADDING REDUNDANCY WITH REDUNDANT I/O MODULES.....	26.1
INTRODUCTION	26.1
R-SERIES ORDERING CODES.....	26.1
R-SERIES IO REDUNDANT SYSTEM OVERVIEW	26.2
SWITCHOVER SCENARIOS	26.3
ADDING THE R-SERIES I/O MODULES TO A REDUNDANT SYSTEM	26.4
STARTING THE AREA	26.4
CREATING A LOGIC FROM THE FFB FUNCTION BLOCK.....	26.6
CONFIGURING THE HARDWARE ON LOGICVIEW FOR FFB	26.7
DIAGNOSTICS VIA SNMP.....	26.9
APPENDIX A - SRF – SERVICE REQUEST FORM	A.1
APPENDIX B - SNMP DIAGNOSTICS.....	B.1
INTRODUCTION	B.1
WORKING PRINCIPLE	B.1
INTEGRATION AND NETWORK SECURITY	B.2
OVERVIEW OF DIAGNOSTIC FEATURES FOR DFI302 CONTROLLERS.....	B.3
DIAGNOSTIC GROUPS COMMON TO ALL CONTROLLERS.....	B.3
GENERAL INFORMATION OF THE CONTROLLERS	B.3
DIAGNOSTIC/STATUS OF ETHERNET PORTS.....	B.3
DIAGNOSTIC/STATUS OF SERIAL PORT.....	B.4
SPECIFIC DIAGNOSTIC GROUPS	B.4
DIAGNOSTIC/STATUS OF THE CONTROLLER REDUNDANCY (DF62, DF63, DF73, DF75, DF89, DF95 AND DF97)	B.4
DIAGNOSTIC/STATUS OF THE FOUNDATION H1 PORTS (DF62 AND DF63)	B.5
DIAGNOSTIC OF REDUNDANT I/O MODULES SYSTEM (DF62, DF63 AND DF75).....	B.5
DETAILS OF STATUS BITS FOR I/O REDUNDANT MODULES	B.6
DIAGNOSTIC/STATUS OF THE DF87 POWER SUPPLY AND DF93 RACKS (DF62, DF63 AND DF75)	B.8
SNMP ITEMS - ADDITIONAL INFORMATION.....	B.8
BATTERY STATUS.....	B.9
REDUNDANCY	B.9
FIELDBUS	B.9

APPENDIX C - DF63-RTU C.1

INTRODUCTION C.1

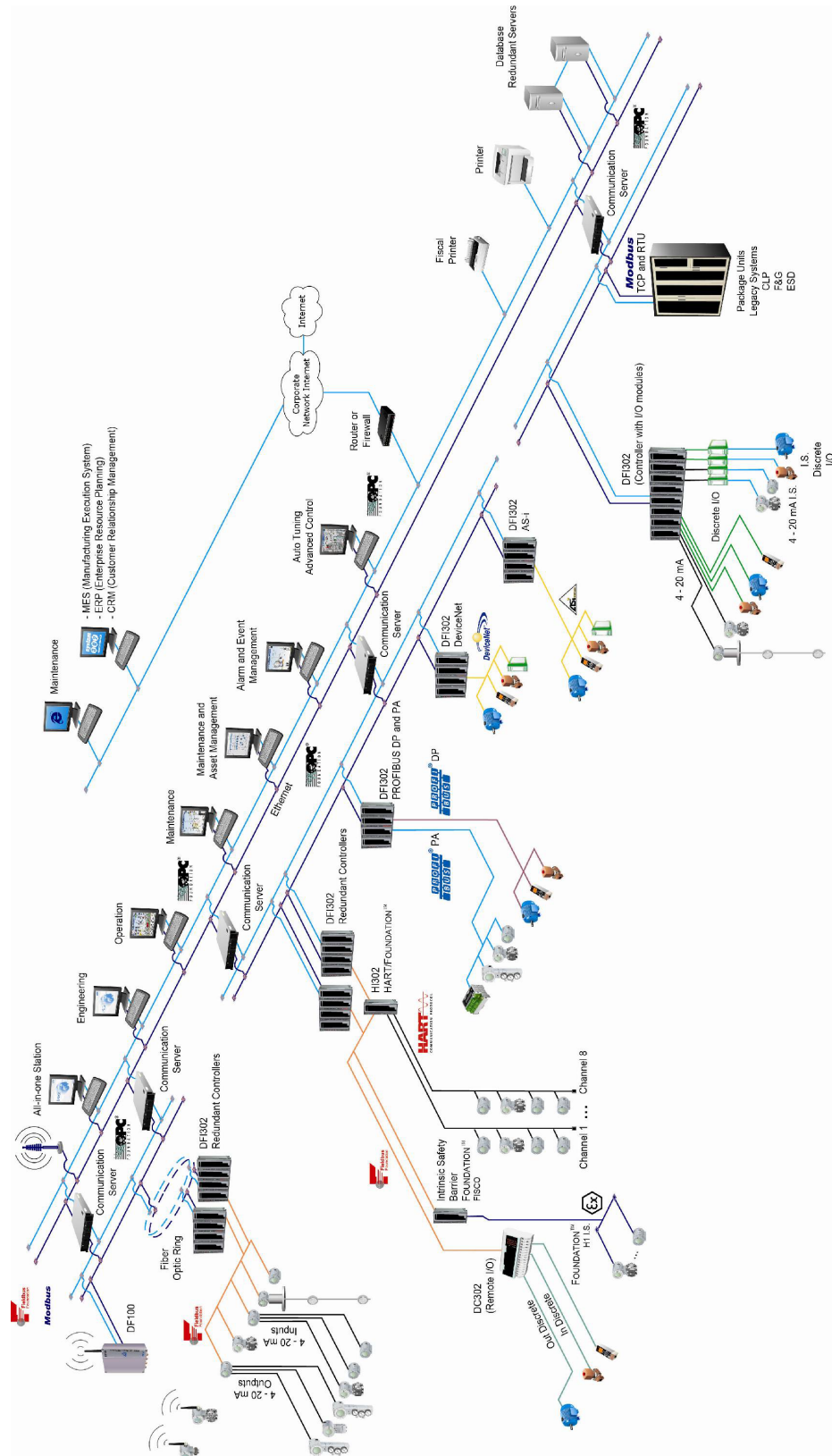
CONFIGURING RTUS..... C.1

DNP3 BLOCKS..... C.3

 DNPCF – CONFIGURATION BLOCKC.3

 DNPCS – DNP CONTROL SLAVEC.10

OVERVIEW



Available Modules for the DFI302 System

CONTROLLERS	
Code	Description
DF51	CPU DFI 1x10 Mbps, 4xH1
DF62	HSE/ FOUNDATION fieldbus Controller
DF63	HSE/ FOUNDATION fieldbus Controller
DF65	Logic Coprocessor
DF65R	Redundant logic Coprocessor
DF65E	Logic Coprocessor - 52 Kbytes
DF65ER	Redundant Logic Coprocessor - 52 Kbytes
DF73	HSE/Profibus-DP Controller
DF75	HSE Controller
DF79	HSE/ DeviceNet Controller
DF81	HSE/AS-i Controller
DF89	HSE/Modbus Controller
DF95	HSE/Profibus Controller with 2 Profibus PA ports e 1 Profibus DP channel
DF97	HSE/Profibus Controller with 4 Profibus PA ports e 1 Profibus DP channel
DF99 ¹	HSE/WirelessHART Controller for rack
DF100	HSE/WirelessHART Controller

I/O MÓDULOS	
Code	Description
DF11	2 Groups of 8 Digital Inputs 24 Vdc - Sink
DF12	2 Groups of 8 Digital Inputs 48 Vdc - Sink
DF13	2 Groups of 8 Digital Inputs 60 Vdc - Sink
DF14	2 Groups of 8 Digital Inputs 125 Vdc - Sink
DF15	2 Groups of 8 Digital Inputs 24 Vdc - Source
DF16	2 Groups of 4 Digital Inputs 120 Vac
DF17	2 Groups of 4 Digital Inputs 240 Vac
DF18	2 Groups of 8 Digital Inputs 120 Vac
DF19	2 Groups of 8 Digital Inputs 240 Vac
DF20	1 Group of 8 Push-Button Switches
DF21	1 Group of 16 Digital Outputs 24 Vdc - Sink
DF22	2 Groups of 8 Digital Outputs 24 Vdc - Source
DF23	2 Groups of 4 Digital Outputs 120/240 Vac - Triac
DF24	2 Groups of 8 Digital Outputs 120/240 Vac - Triac
DF25	2 Groups of 4 NO Relay Outputs
DF26	2 Groups of 4 NC Relay Outputs
DF27	1 Group of 4 NO and 4 NC Relay Outputs
DF28	2 Groups of 8 NO Relay Outputs without RC Protection
DF29	2 Groups of 4 NO Relay Outputs without RC protection
DF30	2 Groups of 4 NC Relay Outputs without RC protection
DF31	1 Group of 4 NO and 4 NC Relay Outputs without RC protection
DF32	1 Group of 8 24 Vdc Inputs and 1 Group of 4 NO Relay
DF33	1 Group of 8 48 Vdc Inputs and 1 Group of 4 NO Relay
DF34	1 Group of 8 60 Vdc Inputs and 1 Group of 4 NO Relay
DF35	1 Group of 8 24 Vdc Inputs and 1 Group of 4 NC Relay
DF36	1 Group of 8 48 Vdc Inputs and 1 Group of 4 NC Relay
DF37	1 Group of 8 60 Vdc Inputs and 1 Group of 4 NC Relay
DF38	1 Group of 8 24 Vdc Inputs and 1 Group of 2 NO and 2 NC Relay
DF39	1 Group of 8 48 Vdc Inputs and 1 Group of 2 NO and 2 NC Relay
DF40	1 Group of 8 60 Vdc Inputs and 1 Group of 2 NO and 2 NC Relay
DF41	2 Groups of 8 Low Frequency (0 - 100 Hz) 24 Vdc Pulse Inputs
DF42	2 Groups of 8 High Frequency (0 - 10 KHz) 24 Vdc Pulse Inputs
DF44	1 Group of 8 Voltage/Current Analogue Inputs with Internal Shunt Resistors
DF45	1 Group of 8 Low Signal Analogue Inputs for TC, RTD, mV and Ohm

¹ The DF99 is a HSE / WirelessHART controller specially designed for rack mounting. Unlike the DF100, it can access digital I/O. For more information, please contact us.

DF46	1 Group of 4 Voltage/Current Analogue Outputs
DF57	1 Group of 8 Voltage/Current Differential Analogue Inputs with Internal Shunt Resistors
DF67	2 Groups of 8 High Frequency (0 - 10 KHz) AC Pulse Inputs
DF69	2 Groups of 8 NO Relay Outputs
DF71	2 Groups of 4 NO Relay Outputs without RC protection - Max 10 mA
DF72	2 Groups of 4 NC Relay Outputs without RC protection - Max 10 mA
DF116	8 analog inputs with HART master interface (4-20 mA)
DF117	8 analog outputs with HART master interface (4-20 mA)

CABLES	
Code	Description
DF3	Flat cable to connect 2 racks – length 6.5 cm
DF4A	Flat cable to connect 2 racks – length 65 cm
DF5A	Flat cable to connect 2 racks – length 81,5 cm
DF6A	Flat cable to connect 2 racks – length 98 cm
DF7A	Flat cable to connect 2 racks – length 110 cm
DF54	Twisted pair cable 100 Base-TX
DF55	Twisted pair cable 100 Base-TX – cross cable – length 2m
DF59	Cable RJ12 used to connect DF51 and DF58
DF68	Cable to connect redundant CPUs
DF76	Cable to connect coprocessors
DF82	Synchronism cable to connect redundant controllers – length 500 mm
DF83	Synchronism cable to connect redundant controllers – length 1800 mm
DF90	IMB power cable
DF101	Flat cable to connect racks by left side – length 70 cm
DF102	Flat cable to connect racks by right side – length 65 cm
DF103	Flat cable to connect racks by right side – length 81 cm
DF104	Flat cable to connect racks by right side – length 98 cm
DF105	Flat cable to connect racks by right side – length 115 cm

POWER SUPPLIES	
Code	Description
DF50	Power Supply
DF52	Power Supply for Fieldbus – 90 to 264 Vac
DF56	Power Supply for Backplane – 20 to 30 Vdc Input
DF60	Power Supply for Fieldbus - 20-30 Vdc Input
DF87	Power Supply for Backplane 20-30 Vdc, 5 A, redundant, with diagnostic

BARRIERS AND IMPEDANCES FOR POWER SUPPLY	
Code	Description
DF47-12 DF47-17	Intrinsic Safety Barrier for FOUNDATION fieldbus
DF53	FOUNDATION fieldbus Power Supply Impedance with 4 ports
DF98	High Current Power Supply Impedance for Fieldbus

INTERFACES	
Code	Description
DF58	EIA-232/ EIA-485 Interface
DF61	Ethernet Switch 10/100 Mbps
DF66	Remote I/O Communication Interface for DF65
DF66E	Remote I/O Communication Interface for DF65E

RACKS AND ACCESSORIES	
Code	Description
DF0	Blind module to fill empty slots
DF1A	Rack with 4 slots – support to shielded flat cable
DF2	Terminator for last the rack – right side
DF9	Support for a single module
DF78	Rack with 4 slots – It supports Hot Swap of CPUs and redundant I/O access

DF84	IMB Soft Starter
DF91	Lateral adapter
DF92	Rack with 4 slots for redundant CPUs, hot swap and diagnostic support
DF93	Rack with 4 slots, with diagnostic
DF96	Terminator for the last rack – left side

INTERFACES FOR I/O MODULES AND THEIR ACCESSORIES*	
Code	Description
ITF - 005AC1	Interface for 16 points of 120 Vac input compatible with DF15.
ITF - 005AC2	Interface for 16 points of 240 Vac input compatible with DF15.
ITF - 001	Interface for 16 points of 24 Vdc input compatible with DF11.
ITF - 005DC	Interface for 16 points of 24 Vdc input compatible with DF15.
ITF - 101	Interface for 16 points digital output for relay with NA and NC contact compatible DF21.
ITF - 101FAC	Interface for 16 points digital output for relay with NA and NC contact with fuse for AC load compatible with DF21.
ITF - 101FDC	Interface for 16 points digital output for relay with NA and NC contact with fuse for DC load compatible with DF21.
ITF - 102	Interface for 16 points digital output for relay with NA and NC contact compatible DF22.
ITF - 102FAC	Interface for 16 points digital output for relay with NA and NC contact with fuse for AC load compatible with DF22.
ITF - 102FDC	Interface for 16 points digital output relay for relay with NA and NC contact with fuse for DC load compatible with DF22.
ITF - 120AC	Interface for 8 points digital output for AC load relay compatible with DF25.
ITF - 120DC	Interface for 8 points digital output for DC load relay compatible with DF25.
ITF - 123-7	Interface for 16 points digital output for relay with NA and NC contact compatible for AC load with DF69.
ITF - 1237FAC	Interface for 16 points digital output for relay with NA and NC contact with fuse for AC load compatible with DF28 or DF69.
ITF - 1237FDC	Interface for 16 points digital output for relay with NA and NC contact with fuse for DC load compatible with DF28 or DF69.
ITF - 304	Interface for 16 point AC pulse input compatible with DF67.
ITF - 401	Interface for 8 point analog input compatible with DF44 and DF57.
ITF - 402	Interface for 8 point analog input compatible with DF45.
ITF - 501	Interface for 8 point analog output without fuse compatible with DF46.
ITF - QDA-AC	Power distribution interface for 10 points 110/240 Vac @ 2A per point.
ITF - QDA-DC	Power distribution interface for 10 points 24 Vdc @ 2A per point.
ITF - C-10	Connection cable between DFI302 modules and ITF interfaces - 1.0 m.
ITF - C-15	Connection cable between DFI302 modules and ITF interfaces - 1.5 m.
ITF - C-20	Connection cable between DFI302 modules and ITF interfaces - 2.0 m.
ITF - C-25	Connection cable between DFI302 modules and ITF interfaces - 2.5 m.
ITF - C-30	Connection cable between DFI302 modules and ITF interfaces - 3.0m.
ITF - C-35	Connection cable between DFI302 modules and ITF interfaces - 3.5 m.
ITF - C-40	Connection cable between DFI302 modules and ITF interfaces - 4.0 m.
ITF - C-45	Connection cable between DFI302 modules and ITF interfaces - 4.5 m.
ITF - C-50	Connection cable between DFI302 modules and ITF interfaces - 5.0 m.

R-SERIES	
RACKS AND ACCESSORIES	
Code	Description
DF106	Master Rack - 6 slots for I/O redundancy
DF110 -1	Slave Rack - 10 slots for I/O redundancy - Terminal blocks
DF110 -2	Slave Rack - 10 slots for I/O redundancy – Interface cabling
DF109	Thin stub cable (0,40m)
DF119	Thick cable (1,0m) for DF106-DF109 or DF106-DF110
DF0-R	Blind module to fill empty slots
ITF-CR-10 ITF-CR-15 ITF-CR-20 ITF-CR-25 ITF-CR-30 ITF-CR-35 ITF-CR-40 ITF-CR-45 ITF-CR-50	Interface cabling (1 m to 5 m)
ITF-DIG	Passive Interface Panel for 16 Digital Input and/or Output Module - DC
ITF-AN-IOR	Interface Panel for 8 Analog Input and/or Output Module
SCANNERS	
DF107	Master Scanner for I/O Redundancy
DF108	Slave Scanner for I/O Redundancy
I/O MODULES	
DF111	1 Group of 16 Redundant Digital Inputs 24 Vdc - Source
DF112	1 Group of 16 Redundant Digital Outputs 24 Vdc - Sink
DF113	1 Group of 8 Redundant Current Analog Inputs
DF114	1 Group of 8 Redundant Current Analog Outputs

Main Features

DFI302 is the most flexible multiprotocol system of the market.

DFI302 (Fieldbus Universal Bridge) is a key-interfacing element in the distributed architecture of Field Control Systems. It combines powerful communication features with direct I/O access and advanced control for discrete and continuous applications. Due to its modular structure, it can be placed inside panels in the control room or sealed boxes in hazardous areas. Highly expandable, it is targeted to applications ranging from small standalone systems to large and complex plants.

DFI302 is a modular multifunctional device with a DIN-rail mounted backplane into which all components are installed, including main and fieldbus power supplies (DF50 and DF52, respectively), Controllers (DF51, DF62, DF63, DF73, etc), and line impedance (DF53) modules. They are plugged-in through industrial grade connectors and by using a robust metal screw. A conventional or redundant I/O subsystem with modules for analog and discrete inputs and outputs can be connected (optional). Modularity is the keyword for **DFI302** flexibility. Besides, since all modules including the fieldbus power supply subsystem are plugged into the same backplane, the **DFI302** comes up as a single integrated unit.

Electric wiring for power and H1 channels are done through plug-in connectors making removal and insertion easy and reliable.

DFI302 power supply module is plugged straight to the Backplane board, extinguishing any additional bulk power supplies. Their LED indicators show normal and failure modes, making troubleshooting easier to be detected, especially in a wide plant system. An external fuse located on the incoming line side provides the user easy replacement without removal of the power supply module or disconnecting any wiring.

Note that:

- One Backplane is required for each 4 modules.
- Flat Cable is required for interconnecting one to other backplanes.
- One terminator is required for each **DFI302**.
- Each **DFI302** requires at least a power supply for backplane and one CPU module.
- Additional power supply for fieldbus can be required.
- The license for the DFI OLE Server is available in different levels, depending on the number of supervision blocks.

Distributed architecture

The modular concept of the **DFI302** makes it the perfect building block of the system topology. Any topology can be created with the **DFI302**. All system set up and maintenance can be easily done with high efficiency and interoperability.

The distribution of the control task among field devices and multiple **DFI302** systems increases the overall performance and reliability.

The system supports

- Modbus Gateway
- Ethernet Gateway
- Profibus Gateway
- DeviceNet Gateway
- AS-i Gateway
- WirelessHART Gateway
- H1 Power Supply
- H1 Safety Barrier
- Conventional I/O
- Redundant I/O

High Reliability

The distributed and embedded architecture of the **DFI302** warrants high reliability even if in adverse industrial environments: no HDDs, no fans, and no mechanical moving parts. At software execution

level, the internal processes (communication, function blocks, supervision, etc.) are controlled by a prioritized multitasking operating system guaranteeing determinism and real-time operation.

Configuration

The **DFI302** is completely configurable through Foundation fieldbus function blocks. They allow that the whole system (DFI302 and field devices) to be set up by single software.

The function block language is ideal for process control, since it represents all the process functions well known by the automation professionals - process control, logic interlocking, alarms, recipes, calculations and equations. Everything can be configured in one single environment.

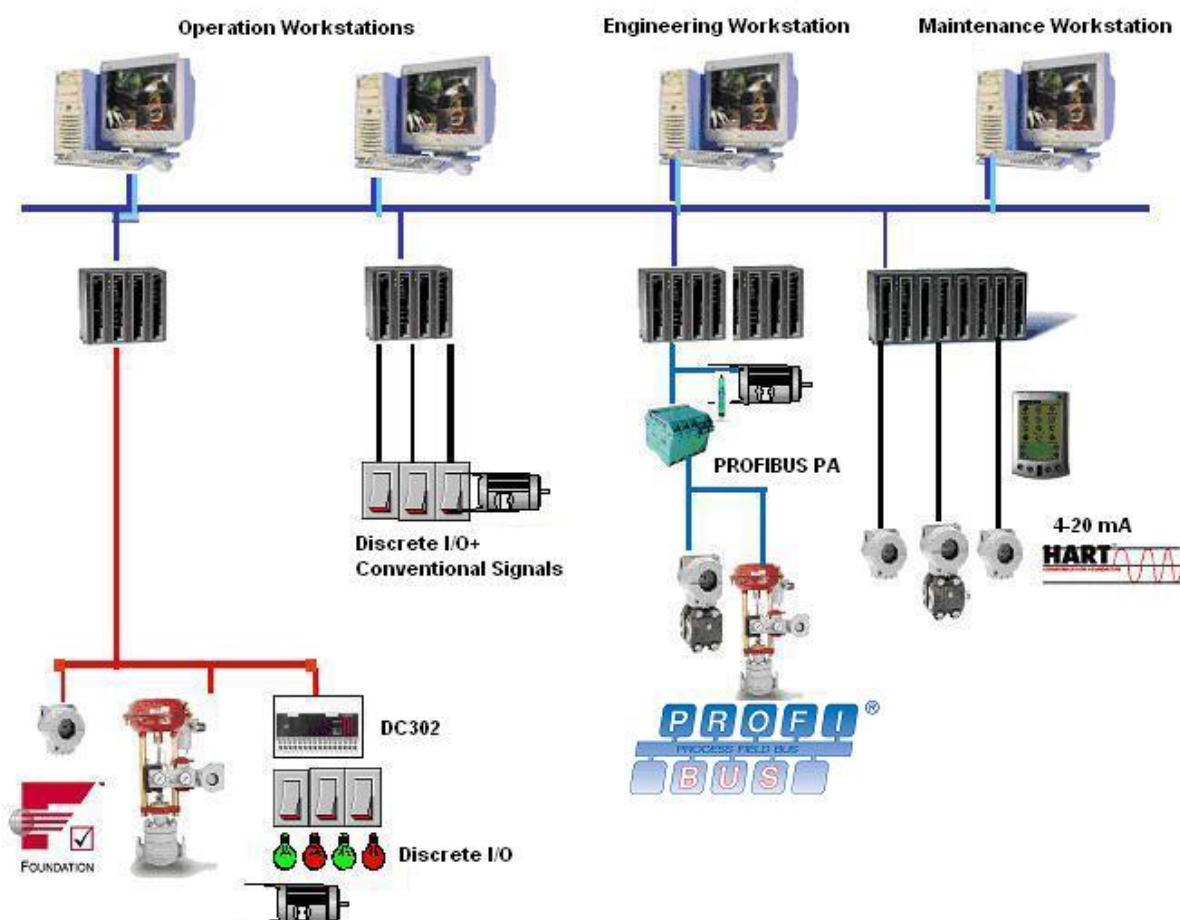
Supervision

The **DFI302** is developed with the most updated technology. The use of leading technologies as OPC (OLE for Process Control) makes the **DFI302** the most flexible Fieldbus interface in the market.

The OPC server allows the **DFI302** to be connected to any HMI package. The only requirement is to have an HMI compatible with OPC. **DFI302** connects with most HMI in the market, customizing our System302 to your needs and knowledge.

System Integration

Advanced communication features found in the **DFI302** grant high system integration:



Redundancy

DFI302 supports hot-standby redundancy in several levels:

- OLE Server
- LAS (Link Active Scheduler)
- Ethernet
- Function Blocks
- H1 Links
- Modbus Gateway

Expandable

Each controller can directly access I/O points distributed among local I/O modules. Exploring fieldbus features such as interoperability, bridge, and Ethernet communication, **DFI302** system becomes a virtually unbounded solution for automation industry.

INSTALLING

Racks, cables and accessories of DFI302 system

MODEL	DESCRIPTION
DF0	Blind module to fill empty slots
DF1A	Rack with 4 slots – support to shielded flat cable
DF2	Terminator for the last rack – right side
DF3	Flat cable to connect 2 racks – length 6.5 cm
DF4A	Flat cable to connect 2 racks – length 65 cm
DF5A	Flat cable to connect 2 racks – length 81,5 cm
DF6A	Flat cable to connect 2 racks – length 98 cm
DF7A	Flat cable to connect 2 racks – length 110 cm
DF9	Support for a single module
DF54	Twisted pair cable 100 Base-TX
DF55	Twisted pair cable 100 Base-TX – cross cable – length 2m
DF59	Cable RJ12 used to connect DF51 and DF58
DF68	Cable to connect redundant CPUs
DF76	Cable to connect coprocessors
DF78	Rack with 4 slots – It supports Hot Swap of CPUs and redundant I/O access
DF82	Synchronism cable to connect redundant controllers – length 500 mm
DF83	Synchronism cable to connect redundant controllers – length 1800 mm
DF84	IMB Soft Starter
DF90	IMB power cable
DF91	Lateral adapter
DF92	Rack with 4 slots for redundant CPUs, hot swap and diagnostic support
DF93	Rack with 4 slots, with diagnostic
DF96	Terminator for the last rack – left side
DF101	Flat cable to connect racks by left side – length 70 cm
DF102	Flat cable to connect racks by right side – length 65 cm
DF103	Flat cable to connect racks by right side – length 81 cm
DF104	Flat cable to connect racks by right side – length 98 cm
DF105	Flat cable to connect racks by right side – length 115 cm

Installing the system's base with DF92 and DF93 racks

In the following figure is shown the DF93 rack with its components.

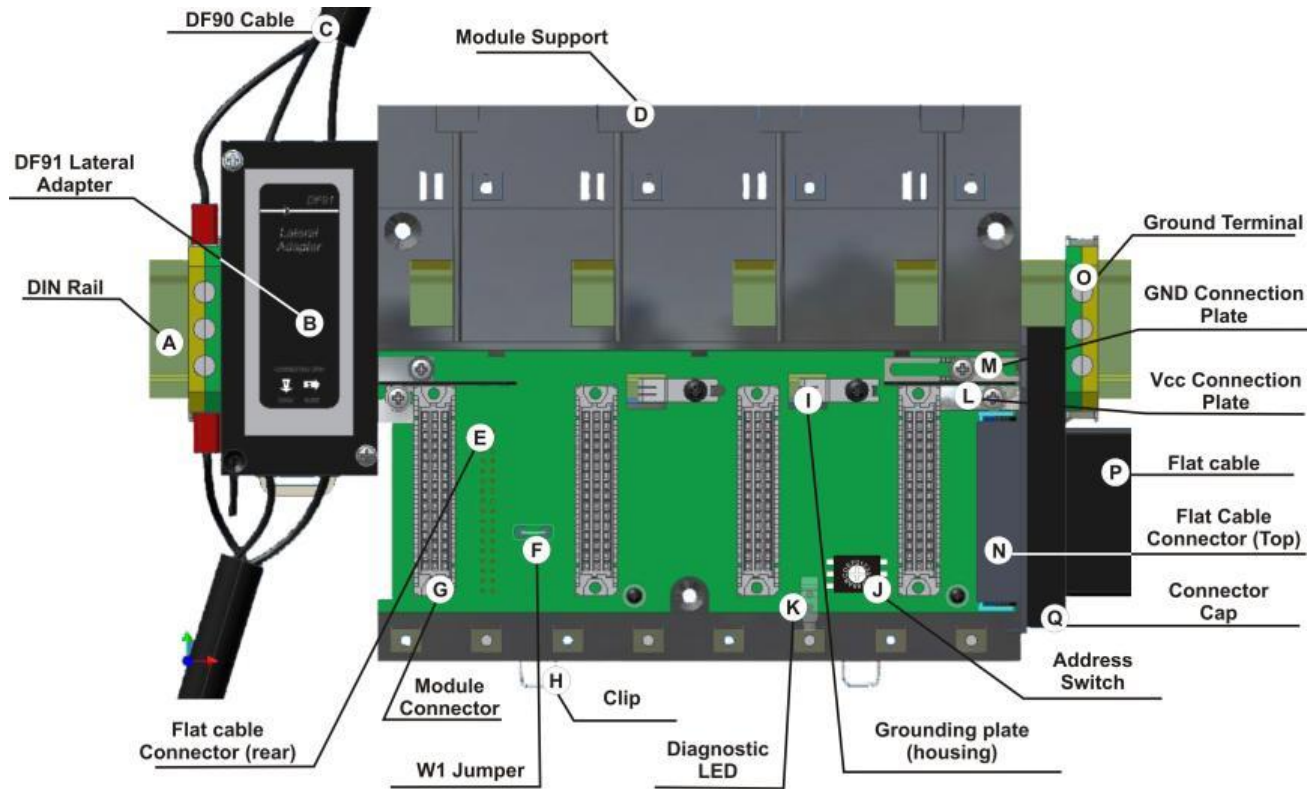


Figure 2. 1 - DF93 components

A – DIN rail- Base for rack connection. It should be tightly fixed to the place where the rack is being mounted.

B – Lateral adapter DF91 – It allows the connection of DF90 cables to rack.

C – DF90 cable– Cable for IMB power transmission. In this cable is the Vcc and GND of IMB and it has to be connected in the rack's left side.

D – Module support - Module holder located in the top of the rack.

E – Flat Cable Connector (rear) – It allows that two racks are interconnected by flat cable (P). When there is more than one rack in a same DIN rail, the user should proceed as described in the "Connection between adjacent racks" topic.

F –W1 Jumper – To disconnect the rack from the power of the previous rack, W1 must be cut, together with the Vcc connection plate (L) of the previous rack. This condition is necessary if a new power supply is inserted from this rack.

G – Module connector – Connector to attach the module's bottom part to the rack.

H – Clips – The metal clips, located in the rack's bottom part, allow attaching the rack to the DIN rail. They must be pulled before fitting the rack on DIN rail, and then, pushed for pieces fixation.

I – Grounding plate (housing)

J – Address switch – When there is more than one rack in same data bus, the addressing switch allows different addresses to each rack.

K – LED for diagnostic – It is used for diagnostic of the rack's voltage.

L – Vcc connection plate – Vcc terminal (for power transmission).

M – GND connection plate - GND terminal (for power transmission).

N – Flat Cable Connector (top)– It allows that two racks are interconnected by flat cable (P). When there is more than one rack in a same DIN rail, the user should proceed as described in the "Connection between adjacent racks" topic.

O – Ground terminal – It is used to ground the flat cables shield.

P – Flat Cable – Cable used to interconnect the data bus among racks.

Q – Connector cap – To meet the EMC requeriments a protector against ESD must be installed in the flat cables connections, at right.

Installing Racks - DF92 and DF93

The DF92 is used by redundant controllers, and it must be the first rack of IMB. The other racks must be DF93.

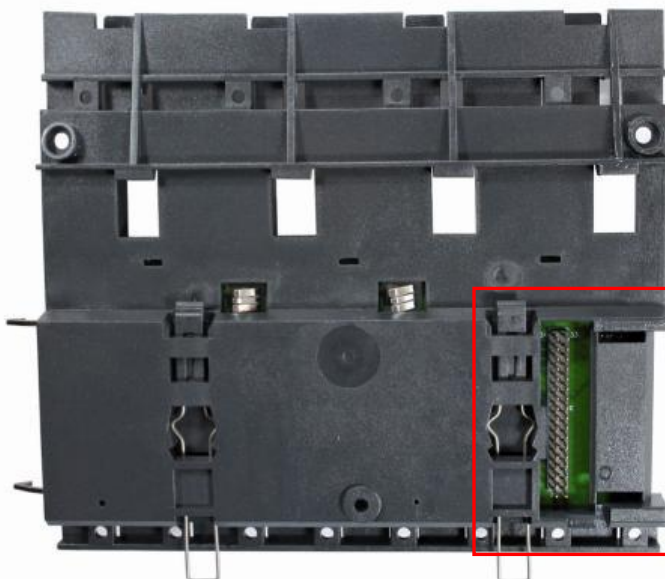


Figure 2. 2 - Rear connector of DF93 rack

IMPORTANT

Remember to leave a space in the DIN rail to install the DF91 and the grounding terminal at rack's left side.

Installing racks in the DIN rail

IMPORTANT

Before installing the rack on DIN rail, connect the flat cable to rear's connector (E) if you will connect this rack to another at left. After connected to the DIN rail is not possible place the flat cable on the rear's rack without remove it.

1. Use a screwdriver (or your fingers) to pull the clips down.
2. Place the back of the rack on the top of the DIN rail edge.
3. Accomodate the rack on the DIN rail and push the clips up. You will hear a click sound when they lock properly.
4. Set the correct address for the DF93 rack using its rotating switch (J). The DF92 rack does not have address switch.

Connection between adjacent racks

1. The adjacent cards to the joining part, between the racks, must be removed allowing access to this operation (racks's third slot, at left and slot 0 of rack, at right).
2. Connect the two racks using DF3 flat cable. This flat cable should already be connected to the connector on the rear's rack at right. And then, connect it to the top connector (N) of the rack at left.
3. Connect the two racks to the power connectors (L and M), moving them with a screwdriver and fixing with screws. Loose the screws only the sufficient avoiding them from falling when making the connection. See the next figure.

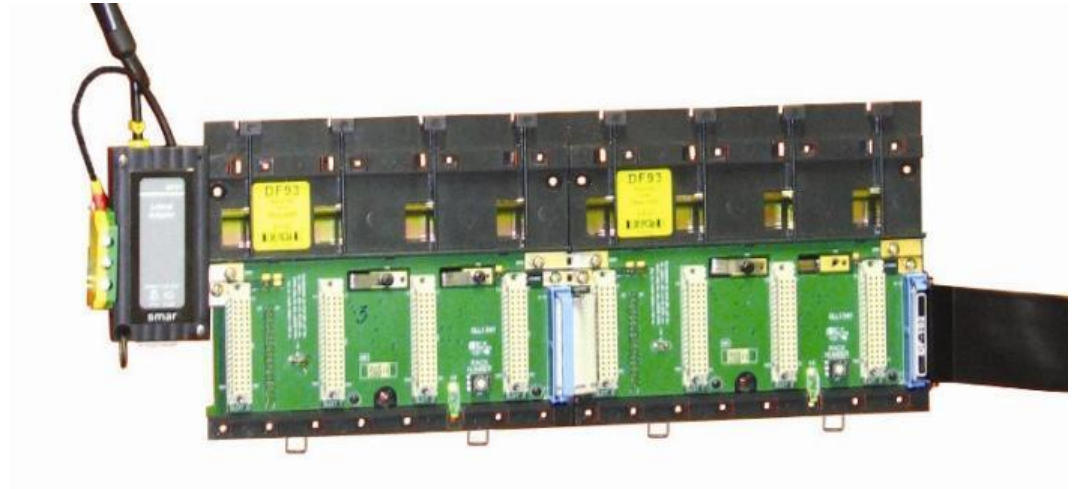


Figure 2. 3 - Connection between adjacent racks

Using the DF91

For further details about DF91 installation, refer to "Expanding the system's power supply –DF90 and DF91" topic.

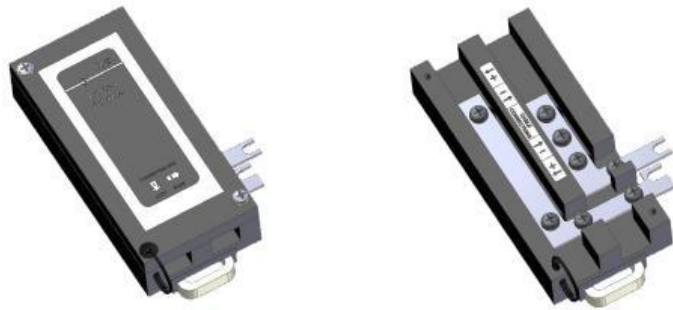


Figure 2. 4 - DF91 details

Left side ESD protection

If the power supply side connector on the left side of the rack (DF92 or DF93) is disconnected, it should be capped with the left side ESD protection for compatibility with the EMC standards. This situation can occur in the left-most rack in systems with a single row of racks or systems with individual racks.

The installation is done screwing the protection in the connection terminals on the left side of the rack. See the following figure.



Figure 2.5 – Left side ESD protection installed on the rack

This protection is provided along with the DF2 terminator.

Disconnecting racks

1. The adjacent cards to the joining part, between the racks, must be removed allowing access to this operation.
2. Remove the flat cable of top connector (N) of the adjacent rack, at left.
3. Remove the power connections (L and M) of both sides of the rack that will be disconnected. For that, with a screwdriver, release the screws (only the sufficient) and move the connection plates to left until they are completely withdrawn, thus the rack is free to be removed.
4. If the DF91 (B) is connected to rack that will be removed, remove it until the rack to be free.
5. Remove the rear connector (E) after removing the rack from DIN rail.

IMPORTANT

Although any application using DF93 as the first rack can use DF84 (IMB soft starter), it is only necessary when the controller (DF62, DF63, DF73, DF75, DF79, DF81, DF95 and DF97) executes local logic with discrete output cards. This rule applies only in the DF1A and DF93 racks where the controller is installed.

When using DF92 rack, DF84 is not necessary, the stabilization feature is already built into the rack's board.

Installing the expansion flat cables - DF101, DF102, DF103, DF104 and DF105.

These flat cables are used when the DF1302 is expanded in more than one row of racks, i.e., in different DIN rail segments, one below the other.

DF101 - Flat cable to connect racks by left side

It is installed in the rack's rear connectors (E) of the left extremity of each row of racks, interconnecting the rows 2-3, 4-5 and 6-7 (if they exist).

To ground the flat cables shield, use the ground terminal (O) next to flat cables connection. The available terminal, next to each DF91 (B), can be used.

DF102, DF103, DF104 and DF105 - Flat cable to connect racks by right side

They are installed on the upper connectors (N) of the right extremity rack of each row of racks, interconnecting the rows 1-2, 3-4 and 5-6 (if they exist).

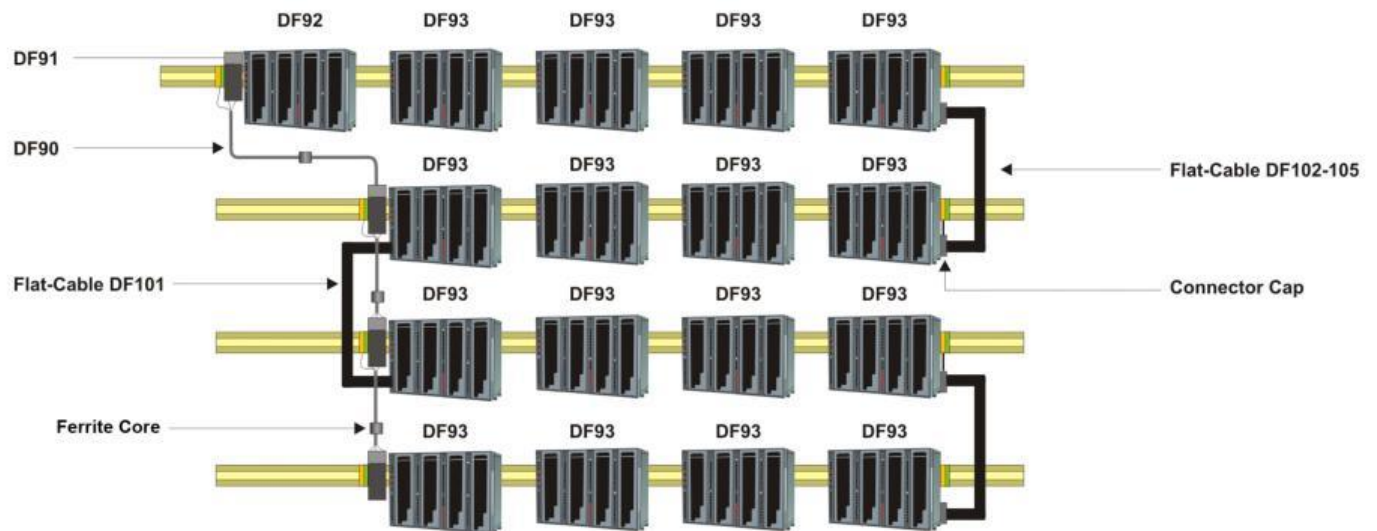


Figure 2. 6 - Illustration - DF101 and DF102-105 Flat cables

To ground the flat cables shield, use the ground terminals (O) next to flat cables connection.

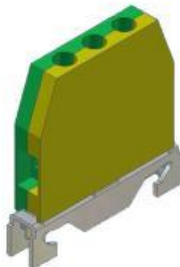


Figure 2. 7 - Ground terminal

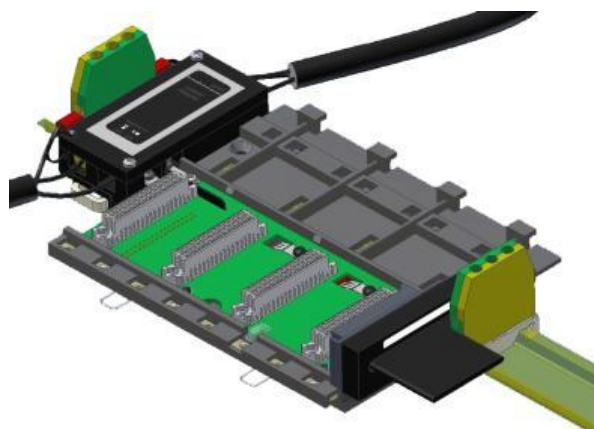


Figure 2. 8 - Ground terminal installed

Flat cables protector (connector cap)

To meet the EMC requirements a protector against ESD has to be installed on the flat cables connection, at right. In the following figure a flat cable protector is shown when it is being installed on the cable connector.

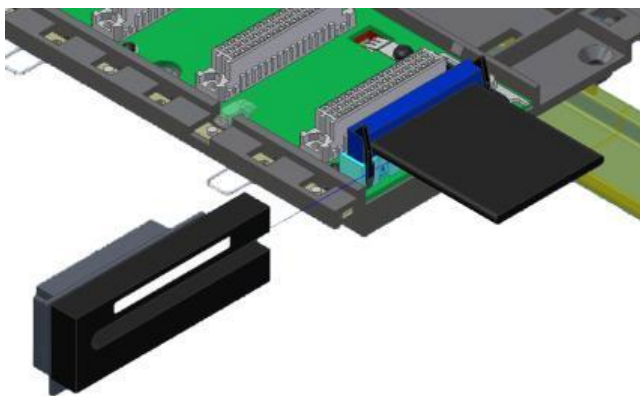


Figure 2.9 - Installing the connector cap

In the following figure is shown a connector cap installed.

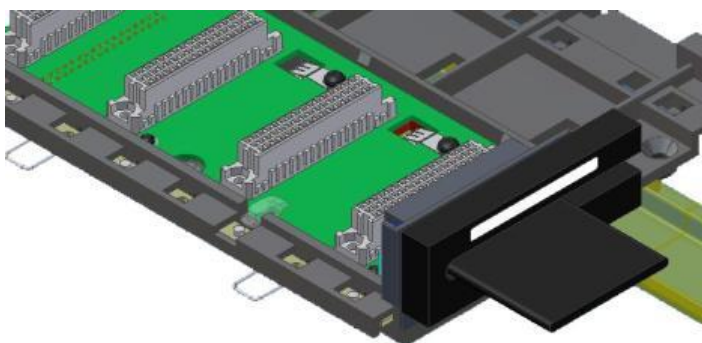


Figure 2.10 - Connector cap installed

Installing the IMB terminator - DF2 or DF96

Only one of these two terminators types (DF2 or DF96) must to be installed at the end of IMB bus. It will depend on which side the last rack is connected to the system.

DF2 – IMB terminator for right side

It is connected to connector N of the last rack, when it is connected to the others by the left side. See the following figure.

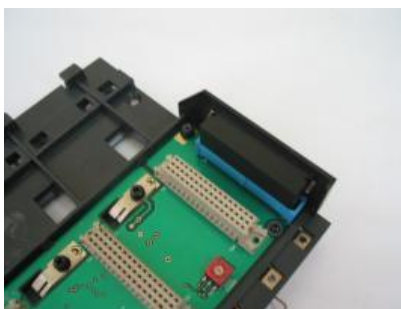


Figure 2.11 - DF2 terminator installed

For further details about its installation refer to DF2 manual.

DF96 – IMB terminator for left side

It is connected to connector E of the last rack, when it is connected to the others by the right side. See the next figure.

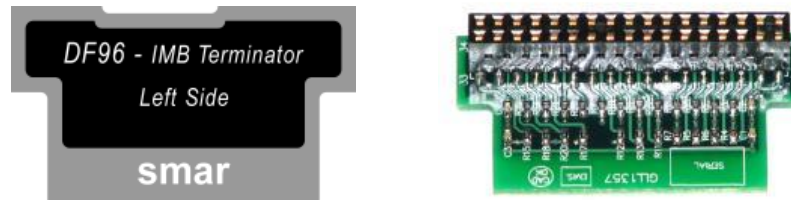


Figure 2.12 - DF96 terminator

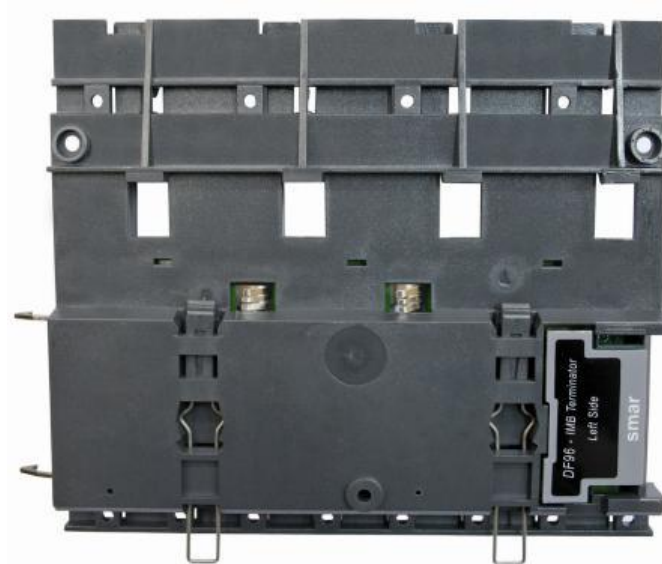


Figure 2.13 - DF96 terminator installed in the DF93 rack

Summarizing, if the last rack has a flat cable connected by left side, use the DF2 terminator. If the last rack has a flat cable connected by right, use DF96 rack.

Both cases depend on the number of row of racks, if it is even or odd.

Expanding the system's power - DF90 and DF91.

This expansion has to be used when the DFI302 is expanded in more than one row of racks, i.e., in different DIN rail segments, one below the other.

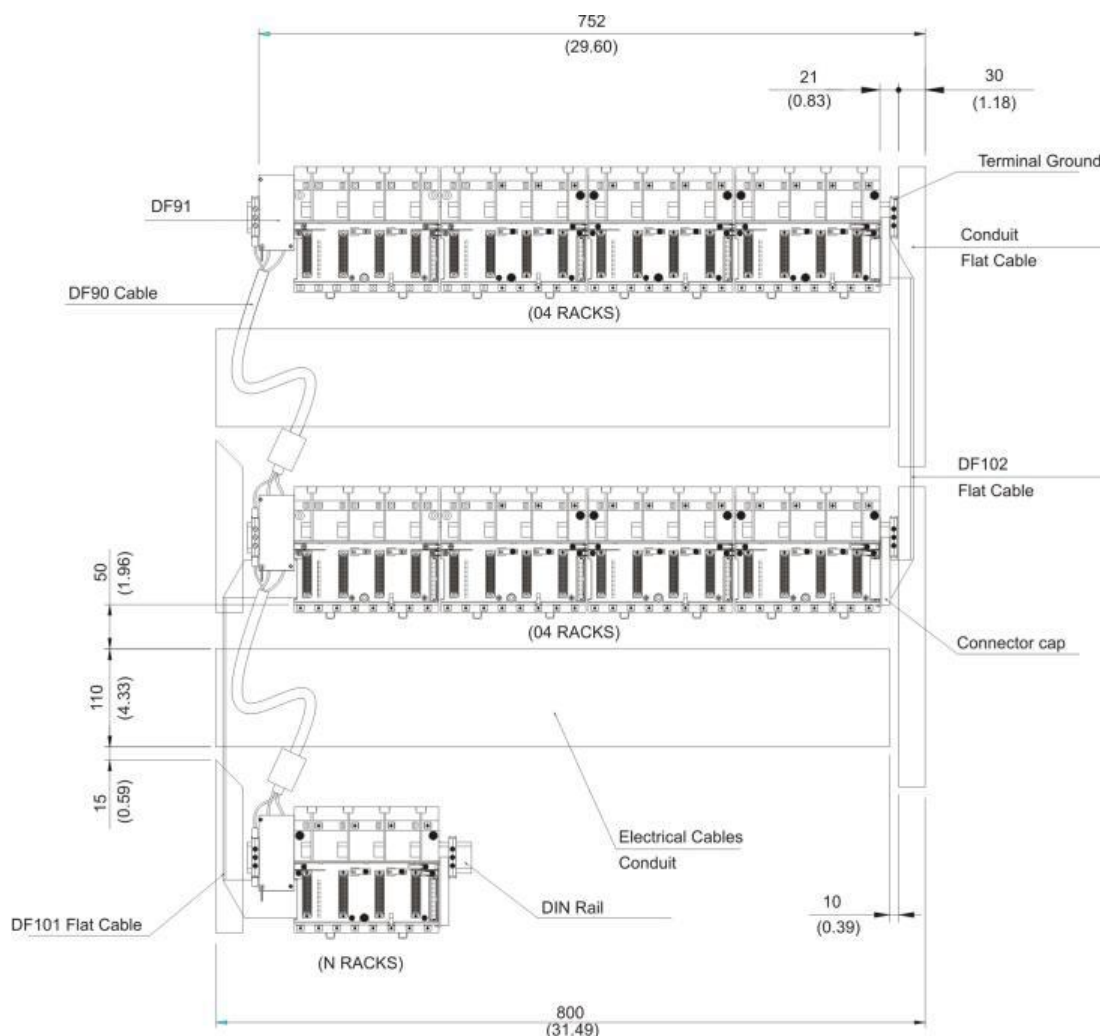


Figure 2. 14 - Example of expanded system

Installing the DF91 in the DIN rail

The DF91 is installed on the rack of the left extremity of each row of racks.

To connect the DF91 to the DIN rail, fix the DF91's rear part in the upper edge of the DIN rail, and then, accommodate the DF91 in the rail, pushing it until you hear a "click" sound.

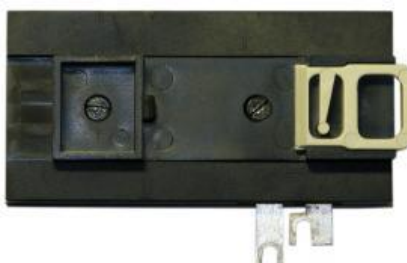


Figure 2. 15 - DF91 rear part

Connecting the DF91 to rack

The first rack's slot needs to be empty allowing access to this operation.

1. Loose the screws (only the sufficient) of the rack's power connector. See the next figure.

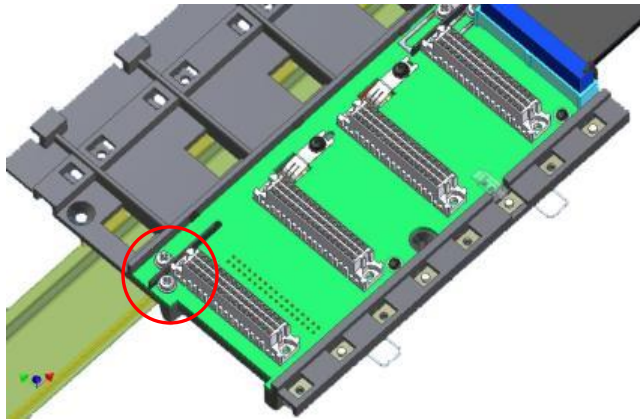


Figure 2. 16 - Details of screws of the rack's power connector

2. Move the DF91 to right up to fix in the screws.
3. Tighten the screws.
4. After connect the DF91 to the rack, install the terminal ground in the left side of DF91, keeping it firm to the rack. This terminal also will be used for grounding of DF90's shield.

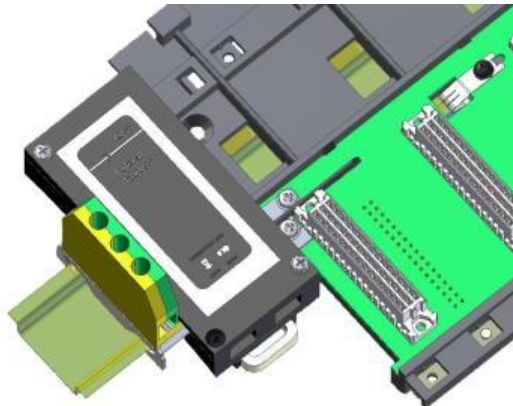


Figure 2. 17 - DF91 connected to rack

Installing DF90



Figure 2. 18 - IMB power cable (DF90)

The cable DF90 must be connected only through DF91, interconnecting two of them. Follow the next steps to execute that procedure.

1. With DF91 already connected to rack, release the cover's screws, and open it;
2. Release the DF91's screws indicated by labels (+) and (-);

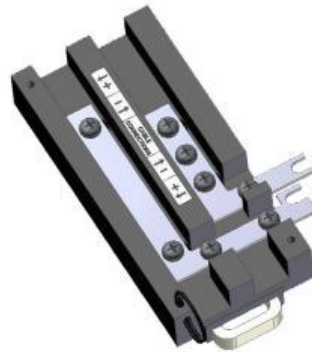


Figure 2.19 - DF91 detail

3. Attach the DF90's terminals with the DF91's screws, obeying the polarity indications;
4. Connect the DF90's shield terminal to the ground terminal next to DF91;

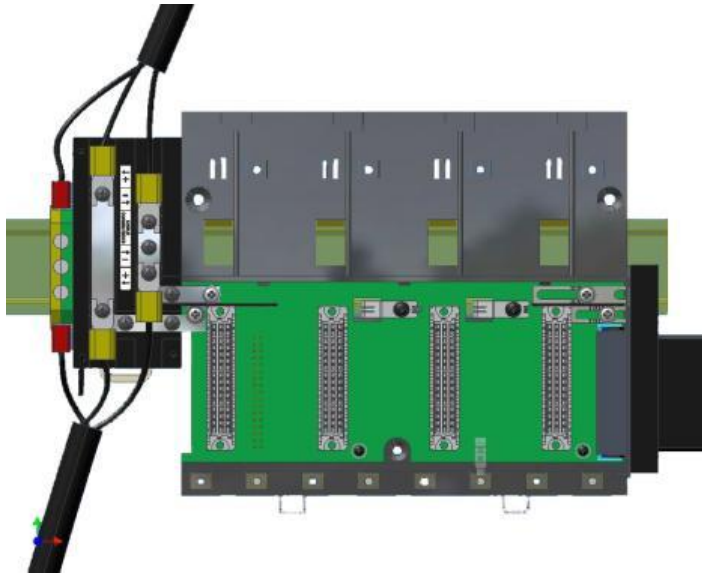


Figure 2.20 - DF91 installed in the rack

5. Close the DF91 cover and tighten the screws.

Disconnecting DF91 from rack

1. The first card of the rack that will be disconnected must be removed allowing access to this operation;
2. Release (only the sufficient) the connector's screws of rack power, where DF91 is connected;
3. Move the DF91 to left (without separate it from rail) until the DF91's connection plates are out of rack's edge;
4. Tighten again the rack's screws if you will not connect them;
5. To remove the DF91, with a screwdriver, unlock it from DIN rail by pulling down the lock at its bottom part and removing that part from the rail.

Diagnostic resources

The DF93 rack has simple resources, but valuable, for voltage diagnostic in the bus. See the following table.

LED	Status
Off	Without voltage or voltage very low
Red	Insufficient voltage
Green	Sufficient voltage

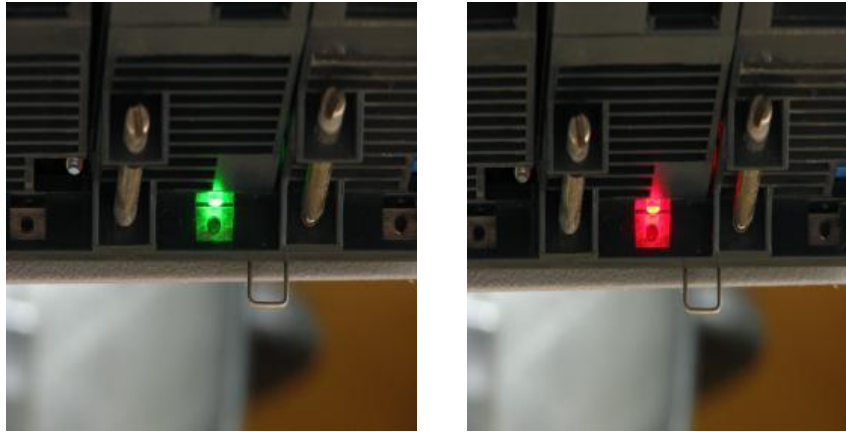


Figure 2. 21 - LEDs for diagnostic in the DF93 rack

Installing the system's base with DF1A and DF78

See below the figures and descriptions of module and rack:

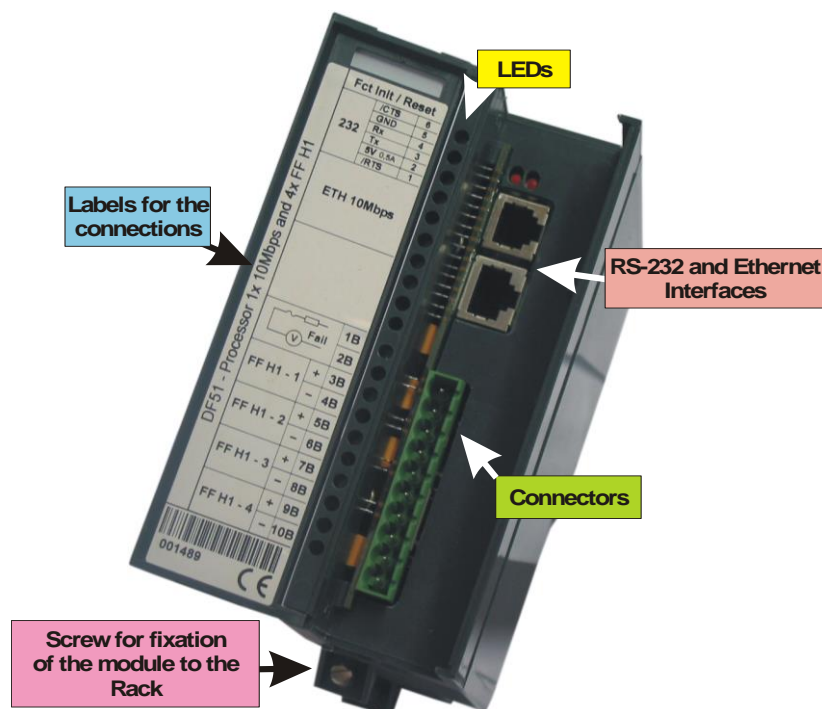


Figure 2.22 - DF51 Module

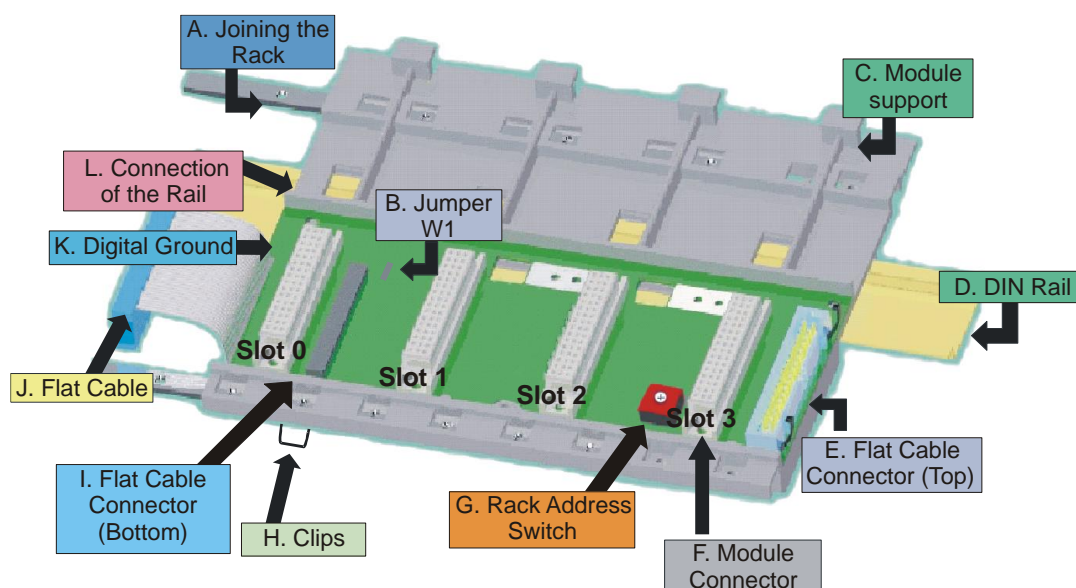


Figure 2.23 - Rack – DF1A

A. **Joining the Rack:** When assembling more than one rack in the same DIN rail, use this metallic piece to interconnect the racks. This connection generates stability to the assembly and makes possible the digital ground connection (K).

B. **Jumper W1:** When connected, it allows the rack to be powered by the previous rack.

- C. **Module support:** Module holder located in the top of the rack.
- D. **DIN Rail:** base rack connection. It should be tightly fixed to the place where the rack is being mounted.
- E. **Flat Cable Connector (Top):** When existing more than one rack in the same DIN rail, they must be hooked up by a flat cable (J) connected to the flat cable connectors (I) and (E).
- F. **Module Connector:** Bottom connection of the module to the rack.
- G. **Rack Address Switch:** When using more than one rack in the DIN rail, the rack address switch allows a distinct address to each rack.
- H. **Clips:** The clips, located above of the rack, allow it to be connected in the DIN rail. It should be pushed down before inserting the rack in the DIN rail and after that pushed up to fix the pieces.
- I. **Flat Cable Connector (Bottom):** When existing more than one rack in the same DIN rail, they must be hooked up by a flat cable (J) connected to the flat cable connectors (I) and (E).
- J. **Flat Cable:** Cable used to connect the data bus between the racks.
- K. **Digital Ground** – When using more than one rack in the same DIN rail, the connection between digital grounds (K) must be reinforced through appropriate metallic piece.
- L. **Connection of the Rail:** Support that brings the connection between the rack and the DIN rail (D).

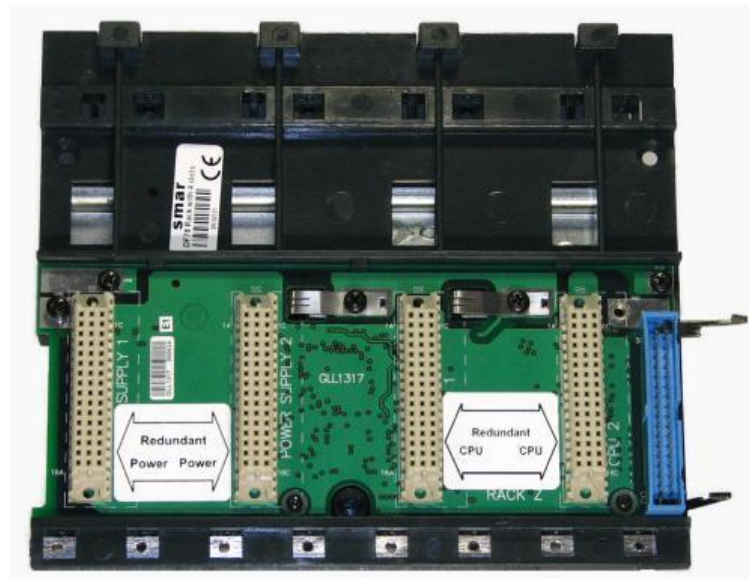


Figure 2. 24 - Rack – DF78

Installing a Rack in the DIN rail

1. In case of only one rack, this fixation can be done as the first step, even before of fixing any module to the rack.
2. Use a screwdriver (or your fingers) to pull the clips (H) down.
3. Place the back of the rack on the top of the DIN rail edge.
4. Accommodate the rack on the DIN rail and push the clips up. You will hear a click sound when they lock properly.
5. Set the correct address for the rack using the rotating switch at the rack.

Adding Racks

1. In case of using more than one rack in the same DIN rail, take a look in the flat cable connections (J) in the top connector of the first rack and in the bottom connector in the second rack, before plugging

the new module in the slot 3 of the first rack;

2. Fix one rack to the other through the joining part of the rack (A). Pass the metal connector of one rack to the other and fix with screws;
3. Connect the digital ground (K), using one metallic connection fixed by screws.
4. Do not forget to place a terminator in the last rack. The terminator should be plugged in the flat cable connector (top) (E);
5. Set the address for the new rack using the rotating switch.

Tips for Assembling

If there is more than one rack in the same system:

- Do the grip in the DIN rail at the end of the assembly.
- Keep free the slot 3 of the rack to connect the other module through the flat cable connector.
- Check the addresses configuration (rack address switch), as well as the jumper W1 and the cable of the bus.
- Remember that to give continuity to the DC power supply to the previous rack, it is necessary to have the jumper W1 connected.
- Make the amendment of racks and strengthens the digital ground of the hardware.

IMPORTANT

Although any application using DF1A as the first rack can use DF84 (IMB soft starter), it is only necessary when the controller (DF62, DF63, DF73, DF75, DF79, DF81, DF95 and DF97) executes local logic with discrete output cards. This rule applies only in the DF1A and DF93 racks where the controller is installed.

When using DF78 rack, DF84 is not necessary, the stabilization feature is already built into the rack's board.

Improving Signal Ground of DFI302 (DF1A and DF78 Racks)

Besides the fact that the racks of the **DFI302** system are connected by flat cables for signal and power transportation, it is possible to occur some fading in the signal ground for applications that make use of many modules. A solution to keep the signal ground stable and the system more immune to electrical noises is to add an extra wire between racks.

These wires should follow the flat cable path to avoid ground loops. Wire must be strained and have a diameter of at least the AWG18.

For adjacent racks, use the "extending connector" placed on its left side. Obviously, it is possible to have a system with adjacent and non-adjacent racks.

NOTE

The terminator board must be always used in the last rack.

Non-Adjacent Racks

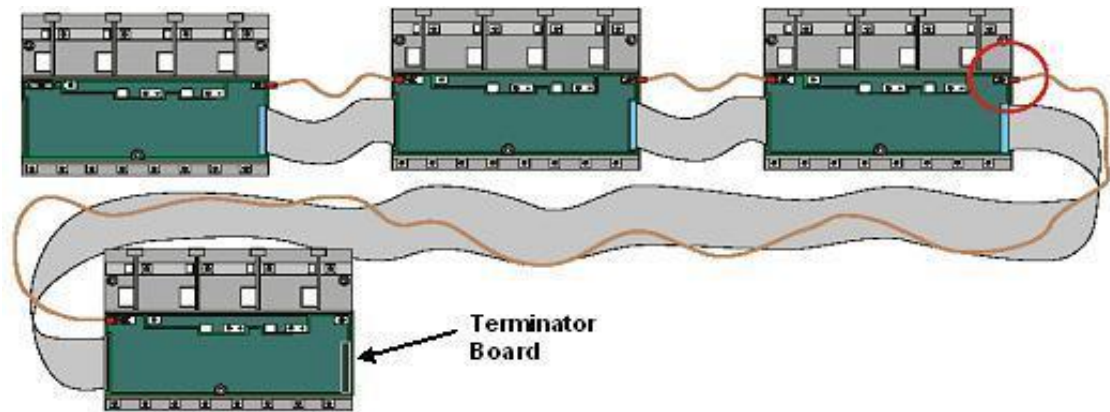


Figure 2. 25 – Improving signal ground

The figure above shows how the signal ground cable is connected between racks.

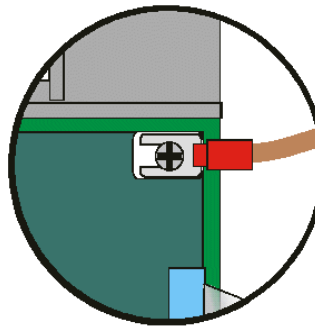


Figure 2. 26 - Detail of the signal ground connection wire

Adjacent Racks

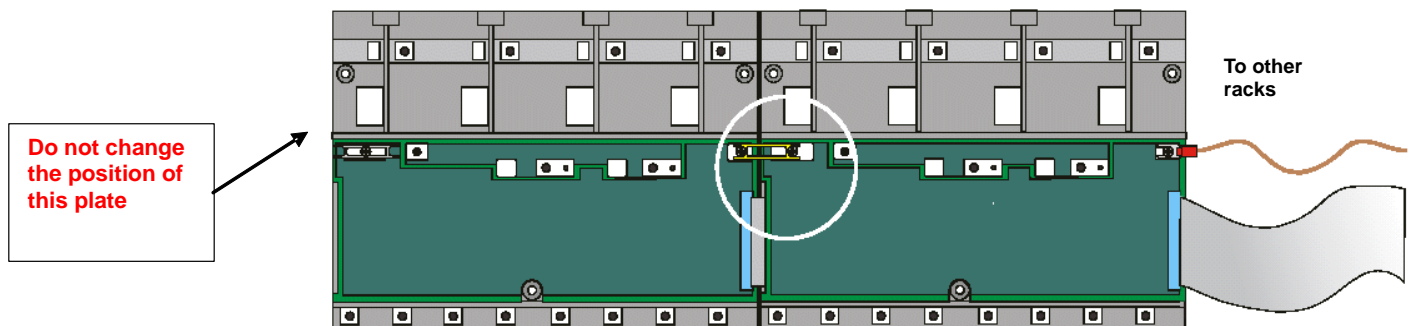


Figure 2. 27 - Connecting adjacent racks

Installing Modules in the Rack

Follow the steps below to install a module in the rack.

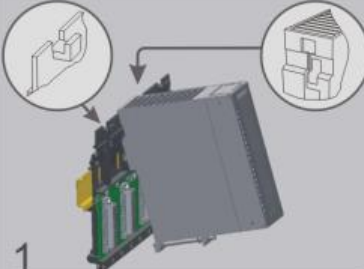
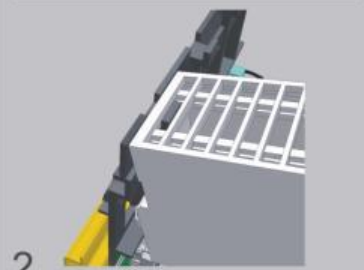
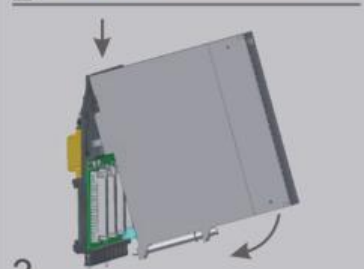
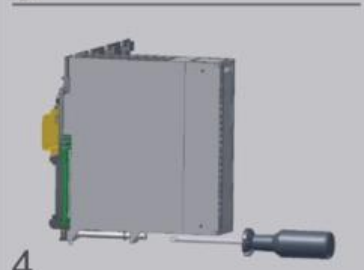
 <p>1</p>	<p>Attach the top of the module (with a 45° inclination) to the module support located on the upper part of the rack.</p>
 <p>2</p>	<p>Mounting detail.</p>
 <p>3</p>	<p>Push the module fixing it to the module connector.</p>
 <p>4</p>	<p>Next, fix the module to the rack using a screwdriver, and fasten the fixation screw at the bottom of the module.</p>

Figure 2. 28 - Installing a module in the rack

Installing the Hardware

DFI302 has LED indicators to show when the communication is active or failing. The modules can be connected and disconnected without turn them off. Using hub/switches the devices can be disconnected without interrupting the process or the control with other nodes.

Using the DF51 Controller

See the details of the modules frontal view:

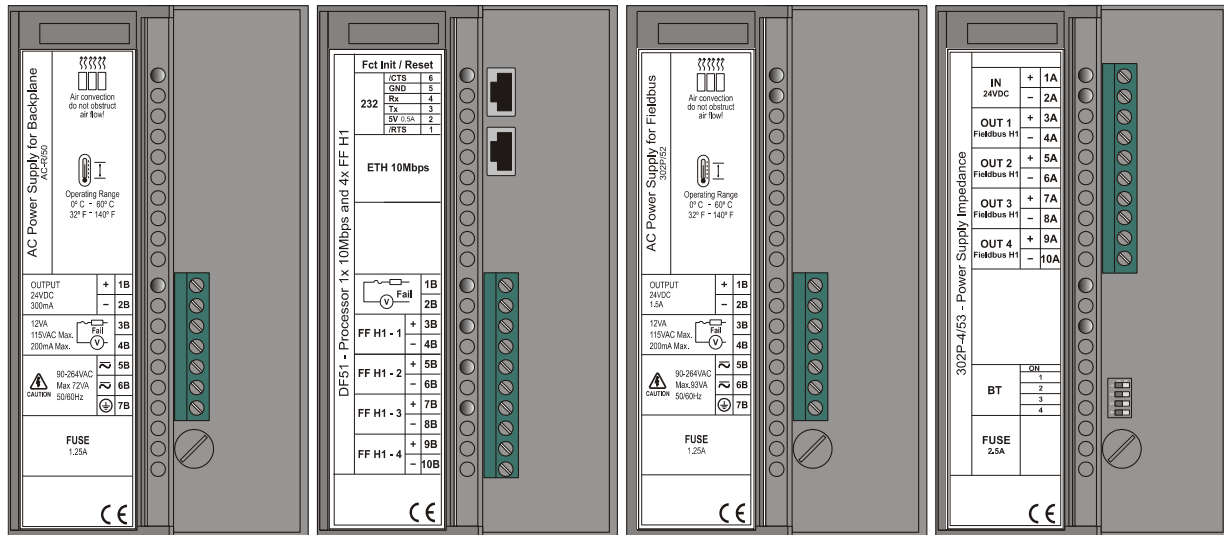


Figure 2. 29 - Basic system of DFI302 with DF51 (frontal view - opened)

For the connection between the DF51 and hub, the DF54 cable (or compatible with CAT5 STP type) should be used. The direct connection between the DF51 and computer can be done using the DF55 cable (or compatible with CAT5 STP cross type). For further information about available cables, refer to the cable specifications section.

Hints to basic installation:

1. Connect the four modules (DF50, DF51, DF52, and DF53) plus the terminator (DF2) in the rack (DF1A or DF93).
2. Connect the AC voltage in the inputs of DF50 and DF52 power supply modules.
3. Connect the DF52 output to the DF53 input.
4. Plug the Ethernet twisted pair cable, connecting DF51 to an Ethernet Hub or Switch.
5. Connect the Fieldbus H1 bus in the DF51 and DF53 FOUNDATION fieldbus H1 ports.
6. If DHCP Server is available **DFI302** IP address is automatically set up, otherwise a fixed IP will be generated (192.168.164.100). This initial fixed address IP can be changed through FBTools (see the "Connecting **DFI302** in your Subnet " topic).

Observe in the following figure:

Detail A shows the electrical connections above mentioned, without the rack view (DF1A or DF93) and the terminator (DF2).

Detail B shows the switches that enable the internal terminator for each Fieldbus H1 channel. In this example, there is only one Fieldbus H1 channel powered on, so the corresponding switch 1 is ON.

Note:

A shielded twisted pair is used for connecting the DF51 to the hub. **DFI302** has simple RJ-45 connectors. It is no necessary any special tool to connect the modules. It is an easy and quick installation.

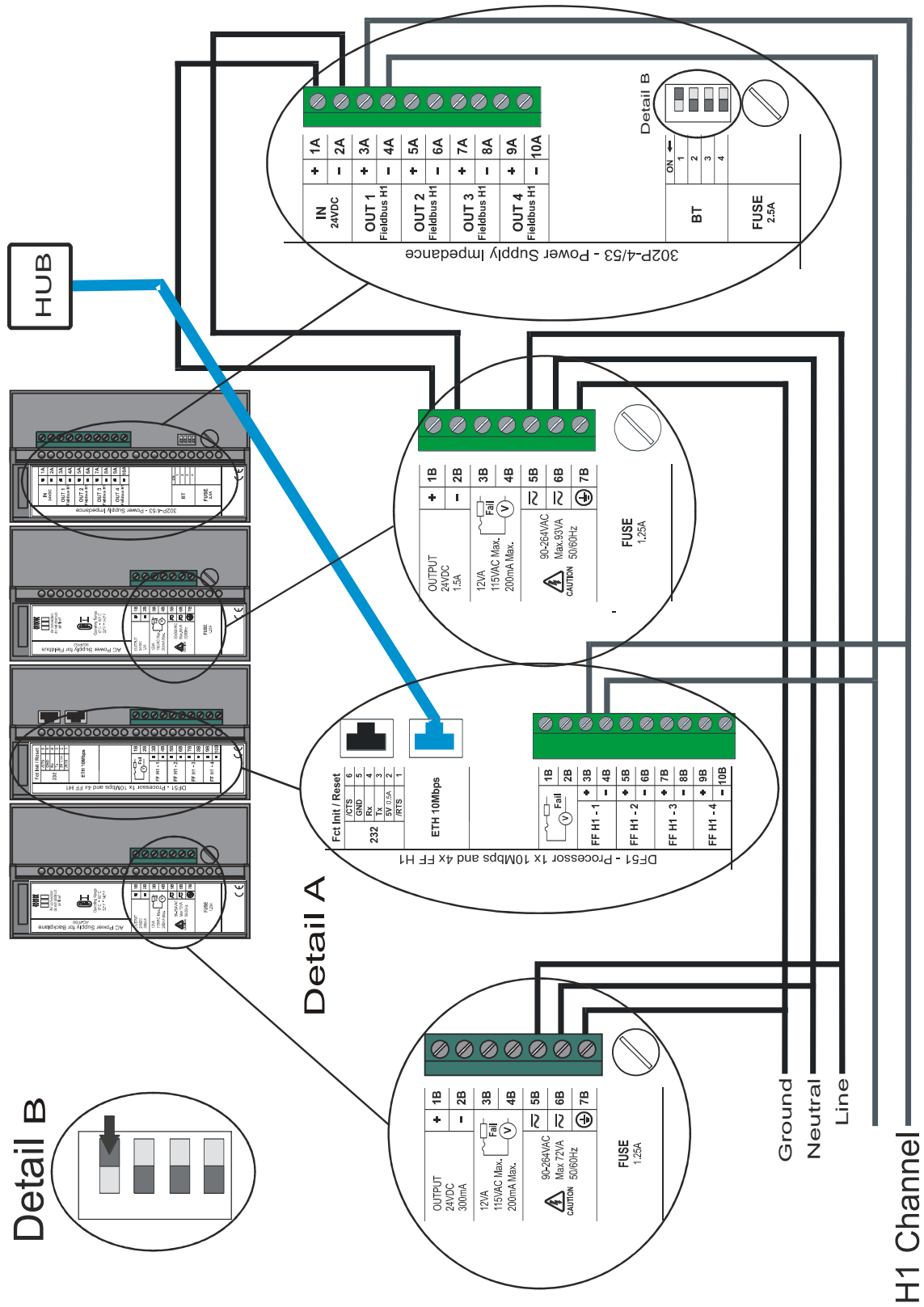


Figure 2. 30 – Connections' details

Using the DF62/DF63 Controller

A typical configuration of the system with the DF62 controller is showed below:

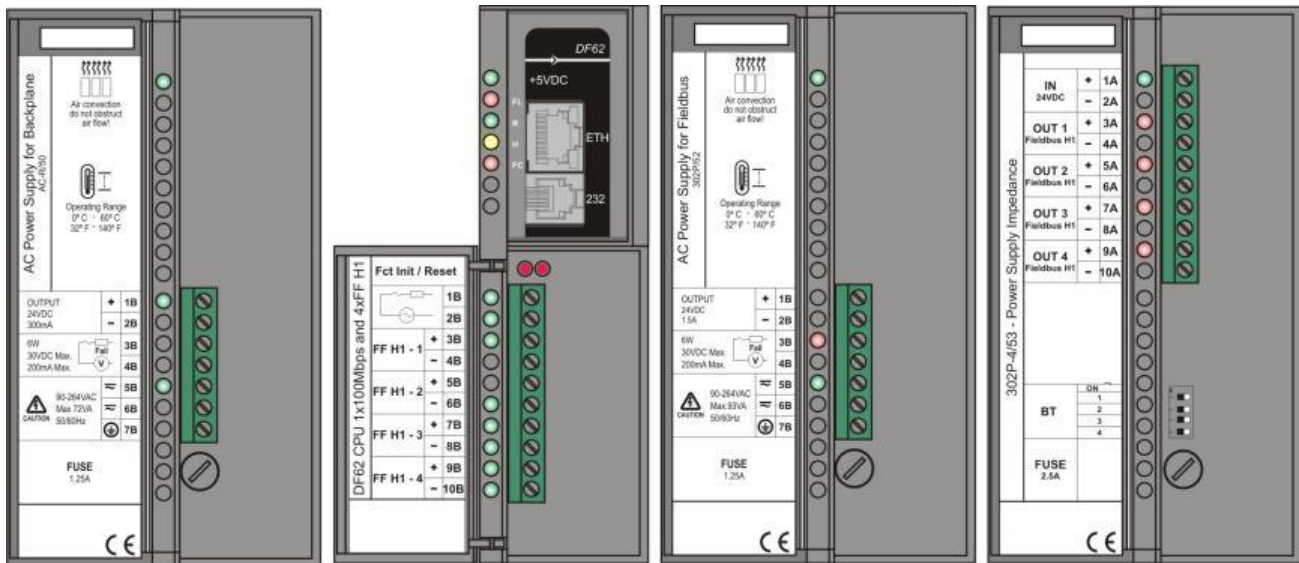
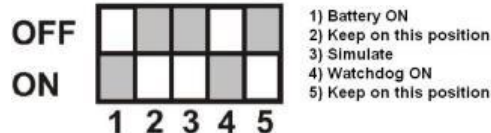


Figure 2.31 - Basic system of DFI302 with DF62 and/or DF63 (frontal view - opened)

Important:

The DF62 has an internal battery that keeps the Real Time Clock (RTC) and its non volatile RAM (NVRAM) when there is lack of external supply. This battery can be either enabled or disabled, depending on the position of the switch 1, in the back part of the DF62. To enable the battery, set the switch to 1 as shown in the following figure:



- In this configuration, when there is lack of energy, the RTC and the NVRAM will be supplied by the battery, allowing the retention of all configuration data. In case of equipment storage, it is recommended that the battery is turned off (switch 1 in position OFF). For further information about the battery, refer to the "Technical specification for controllers" section.

So, before fixing the DF62 module in the rack, be sure the switch 1, which refers to the battery, is in the enabled position.

The Watchdog is a mechanism to detect if an important or high priority task stops in the controller. So, be sure the switch 4, which refers to the Watchdog, is in the ON position.

Steps for the basic installation

- Connect the four modules (DF50, DF62, DF52, and DF53) plus the terminator (DF2) in the rack (DF1A or DF93).
- Connect the AC voltage in the inputs of DF50 and DF52 power supply modules.
- Connect the DF52 output to the DF53 input.
- Plug the Ethernet twisted pair cable, connecting DF62 to an Ethernet Hub or Switch.
- Connect the Fieldbus H1 bus in the DF62 and DF53 FOUNDATION fieldbus H1 ports.
- If DHCP Server is available, **DFI302** IP address is automatically set up, otherwise a fixed IP will be generated (192.168.164.100). This initial fixed address IP can be changed through FBTools (see the "Connecting DFI302 in the Subnet" topic).

Observe in the following figure:

In the next figure, the cable diagram is showed when using the DF62 module.

- Observe that only one H1 segment is being used.
- The DIP switches for the bus are showed in the Detail A.

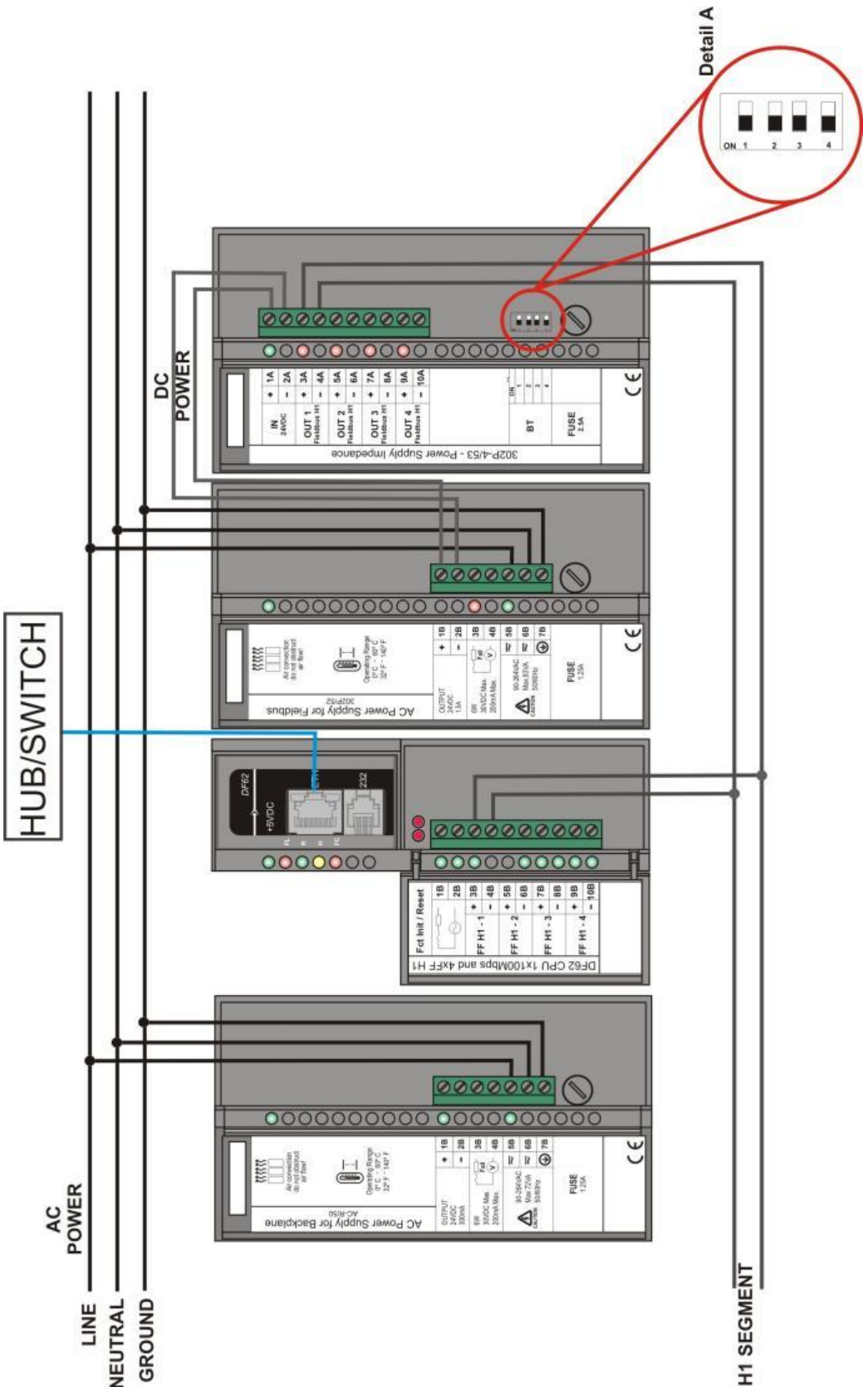


Figure 2. 32 – Cable diagram for DF62

Using the DF73 Controller

A typical system using the DF73 can be composed by:

- DF1A or DF93 – Rack with 4 slots
- DF50 –AC Power Supply
- DF73 – HSE/Profibus DP Controller
- Up to 16 DF1A or DF93 with I/O modules

A redundant system by using the DF73 can be composed of:

- DF78 – Rack with 4 slots (Hot Swap of CPUs and redundant I/O access) or DF92 – Rack with 4 slots (for redundante CPUs, support to Hot Swap and diagnostic);
- DF50 –AC Power Supply;
- DF50 –AC Power Supply ;
- DF73 –HSE Controller;
- DF73 - HSE Controller;
- Plus up to 16 DF1A or DF93 with I/O modules

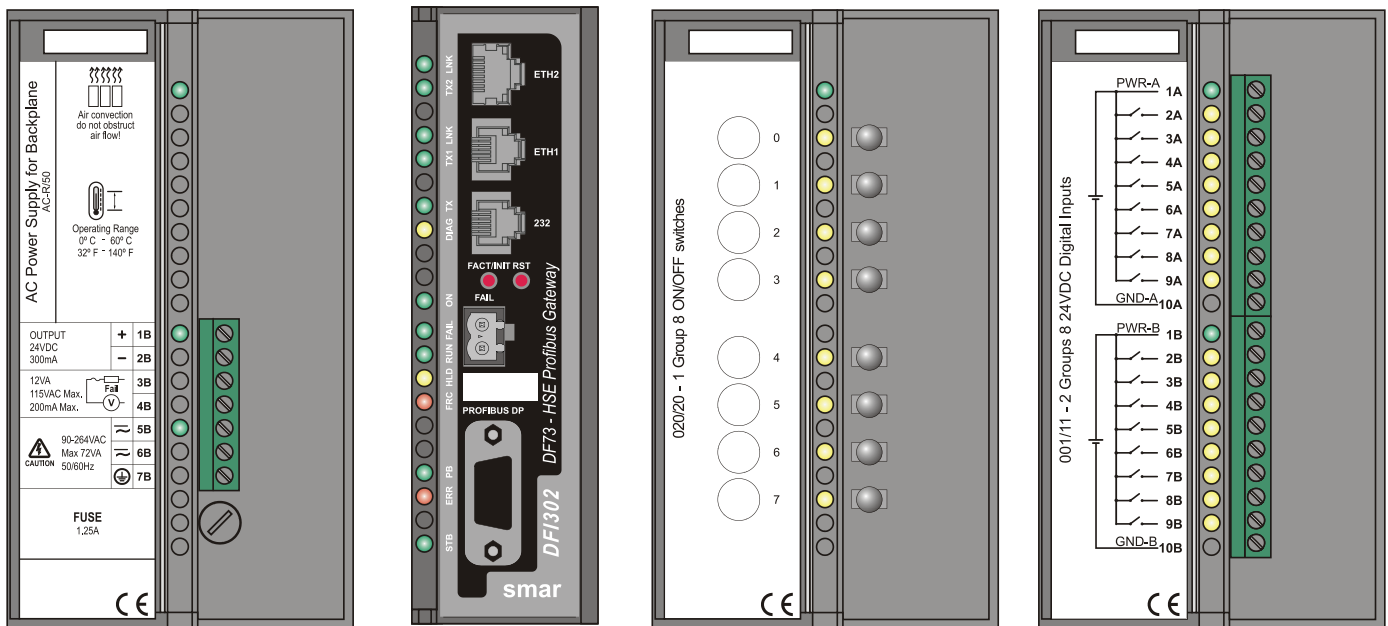
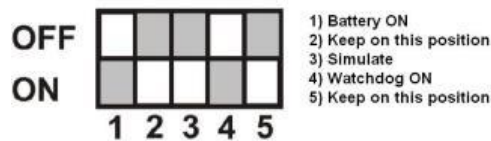


Figure 2.33 - Basic system of DFI302 with DF73

Important:

The DF73 has an internal battery that keeps the Real Time Clock (RTC) and its non volatile RAM (NVRAM) when there is lack of external supply. This battery can be either enabled or disabled, depending on the position of the switch 1, in the back part of the DF73. To enable the battery, set the switch to 1 as shown in the following figure:



- In this configuration, when there is lack of energy, the RTC and the NVRAM will be supplied by the battery, allowing the retention of all configuration data. In case of equipment storage, it is recommended that the battery is turned off (switch 1 in position OFF). For further information about the battery, refer to "Technical specification for controllers" section.

So, before fixing the DF73 module in the rack, be sure the switch 1, which refers to the battery, is in the enabled position.

The Watchdog is a mechanism to detect if an important or high priority task stops in the controller. So, be sure the switch 4, which refers to the Watchdog, is in the ON position.

NOTE:

The terminators used in the DP network must be enabled according to the network topology.

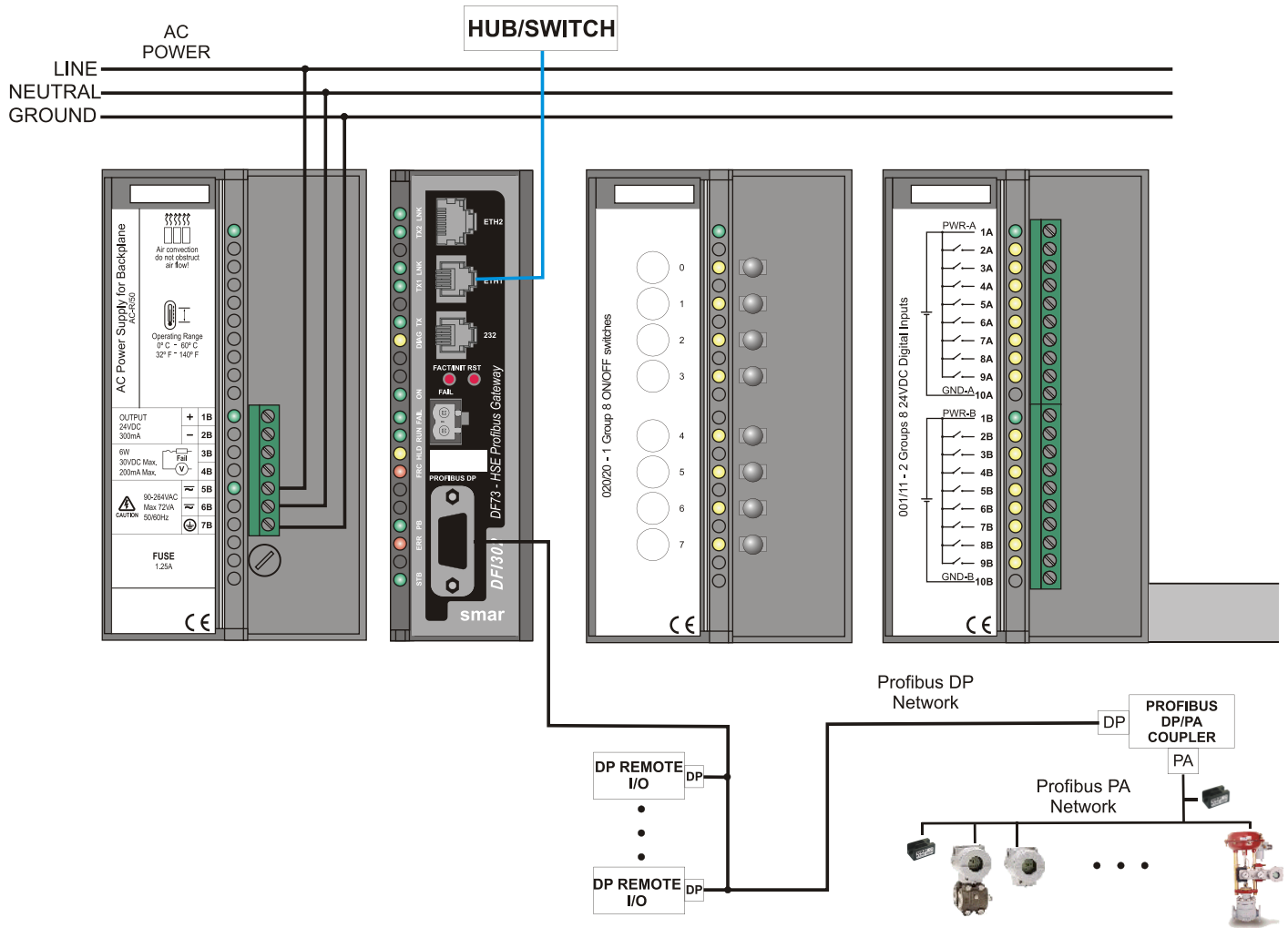


Figure 2. 34 – Cable diagram for DF73



Using the DF75 Controller

A typical system is composed by:

- DF1A or DF93 – Rack with 4 slots;
- DF50 –AC Power Supply;
- DF75 –HSE Controller;
- Up to 16 DF1A or DF93 with I/O modules

A redundant system by using the DF75 can be composed by:

- DF78 – Rack with 4 slots (Hot Swap of CPUs and redundant I/O access) or DF92 – Rack with 4 slots (for redundante CPUs, support to Hot Swap and diagnostic);
- DF50 –AC Power Supply;
- DF50 –AC Power Supply ;
- DF75 –HSE Controller;
- DF75 - HSE Controller;
- Plus up to 16 DF1A or DF93 with I/O modules.

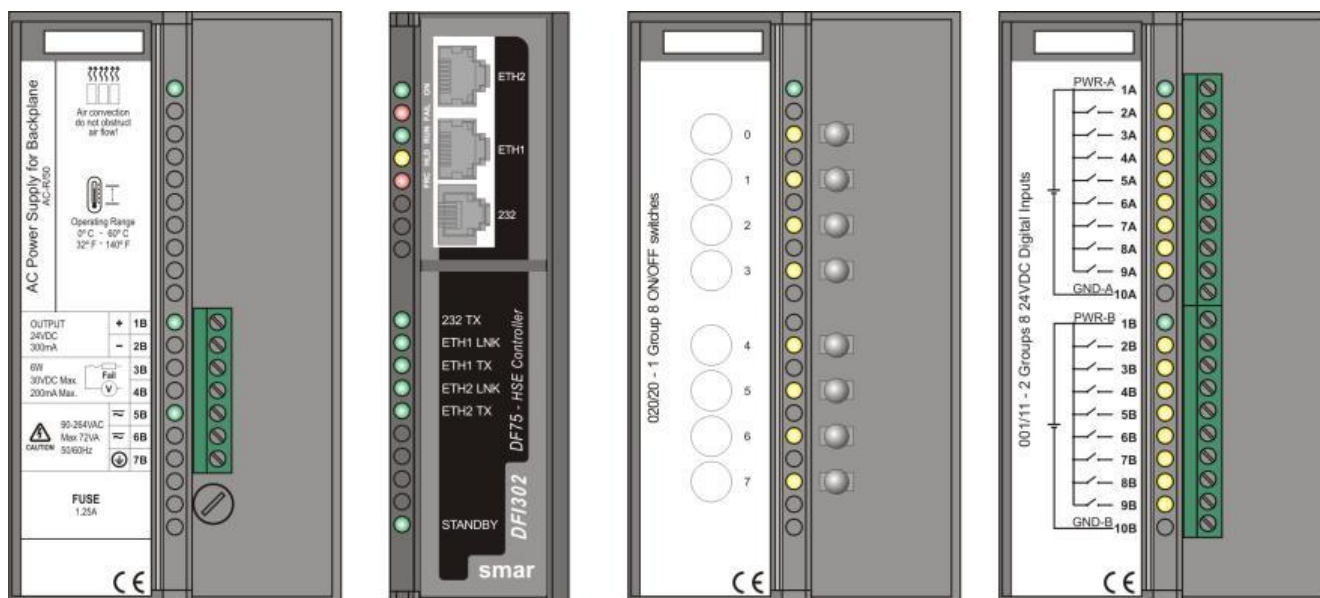
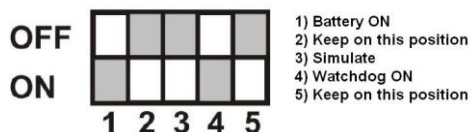


Figure 2.36 - Basic system of DFI302 with DF75

Important:

The DF75 has an internal battery that keeps the Real Time Clock (RTC) and its non volatile RAM (NVRAM) when there is lack of external supply. This battery can be either enabled or disabled, depending on the position of the switch 1, in the back part of the DF75. To enable the battery, set the switch to 1 as shown in the following figure:



In this configuration, when there is lack of energy, the RTC and the NVRAM will be supplied by the battery, allowing the retention of all configuration data. In case of equipment storage, it is recommended that the battery is turned off (switch 1 in position OFF). For further information about the battery, refer to the "Technical specification for controllers" section.

So, before fixing the DF75 module in the rack, be sure the switch 1, which refers to the battery, is in the enabled position.

The Watchdog is a mechanism to detect if an important or high priority task stops in the controller. So, be sure the switch 4, which refers to the Watchdog, is in the ON position.

Observe in the next figures:

- Basic system using the DF75 controller
- Redundant system using the DF75 controller

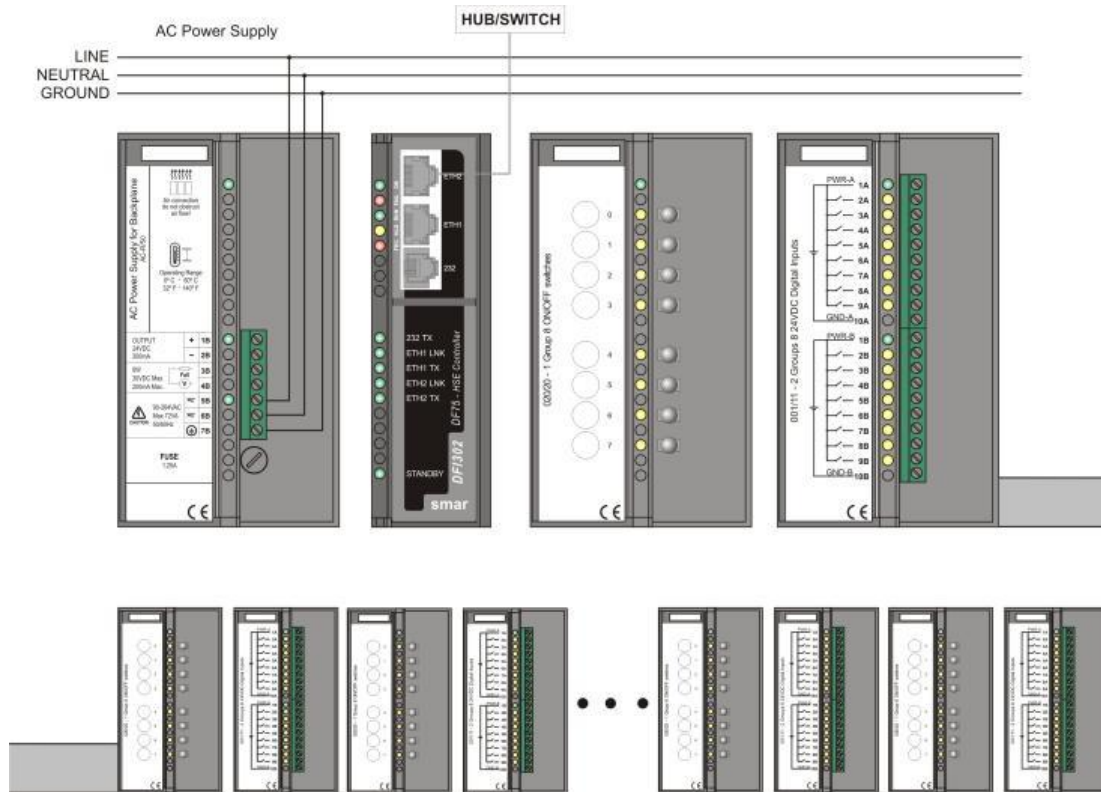


Figure 2.37 - Basic system using the DF75 controller

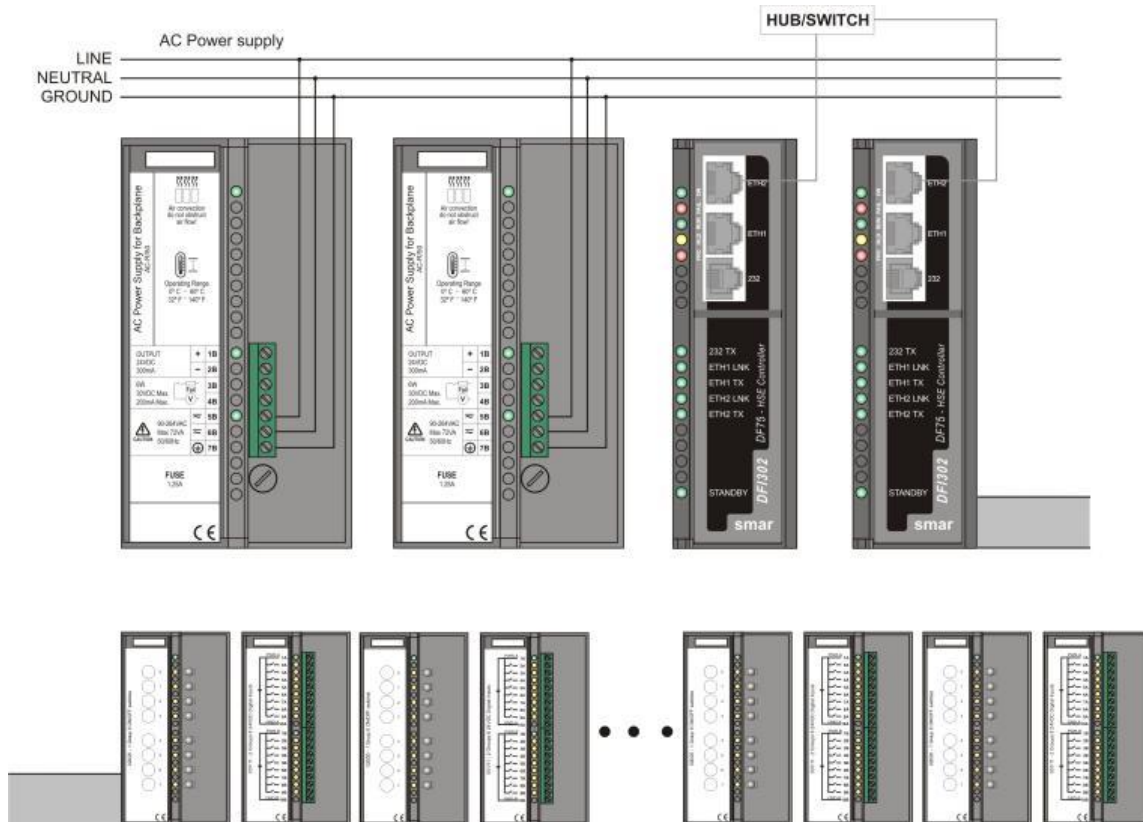


Figure 2.38 - Redundant system using the DF75 controller

Using the DF79 Controller

A typical system using the DF79 can be composed by:

- DF1A or DF93 – Rack with 4 slots
- DF50 –AC Power Supply
- DF79 – HSE/DeviceNet Controller
- Power Supply 24 Vdc, 8A for DeviceNet devices

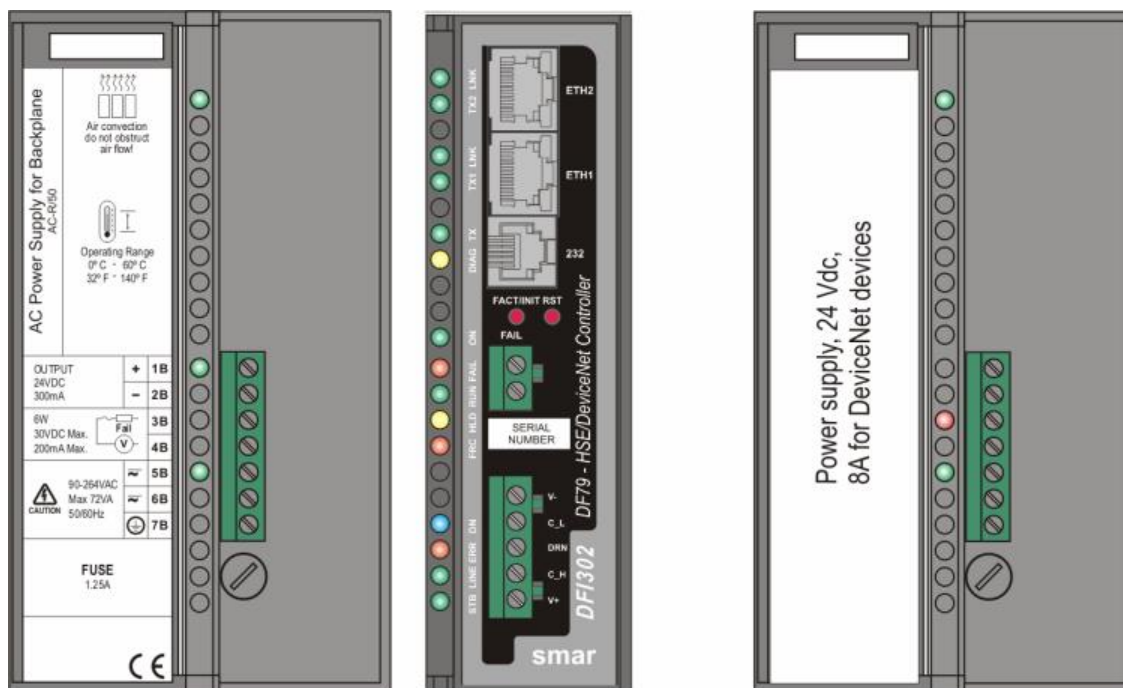
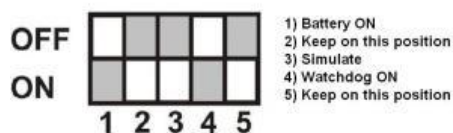


Figure 2.39 - Basic system of DFI302 with DF79

Important:

The DF79 has an internal battery that keeps the Real Time Clock (RTC) and its non volatile RAM (NVRAM) when there is lack of external supply. This battery can be either enabled or disabled, depending on the position of the switch 1, in the back part of the DF79. To enable the battery, set the switch to 1 as shown in the following figure:



In this configuration, when there is lack of energy, the RTC and the NVRAM will be supplied by the battery, allowing the retention of all configuration data. In case of equipment storage, it is recommended that the battery is turned off (switch 1 in position OFF). For further information about the battery, refer to the "Technical specification for controllers" section.

So, before fixing the DF79 module in the rack, be sure the switch 1, which refers to the battery, is in the enabled position.

The Watchdog is a mechanism to detect if an important or high priority task stops in the controller. So, be sure the switch 4, which refers to the Watchdog, is in the ON position.

NOTE:

The terminators used in the DeviceNet network must be enabled according to the network topology.

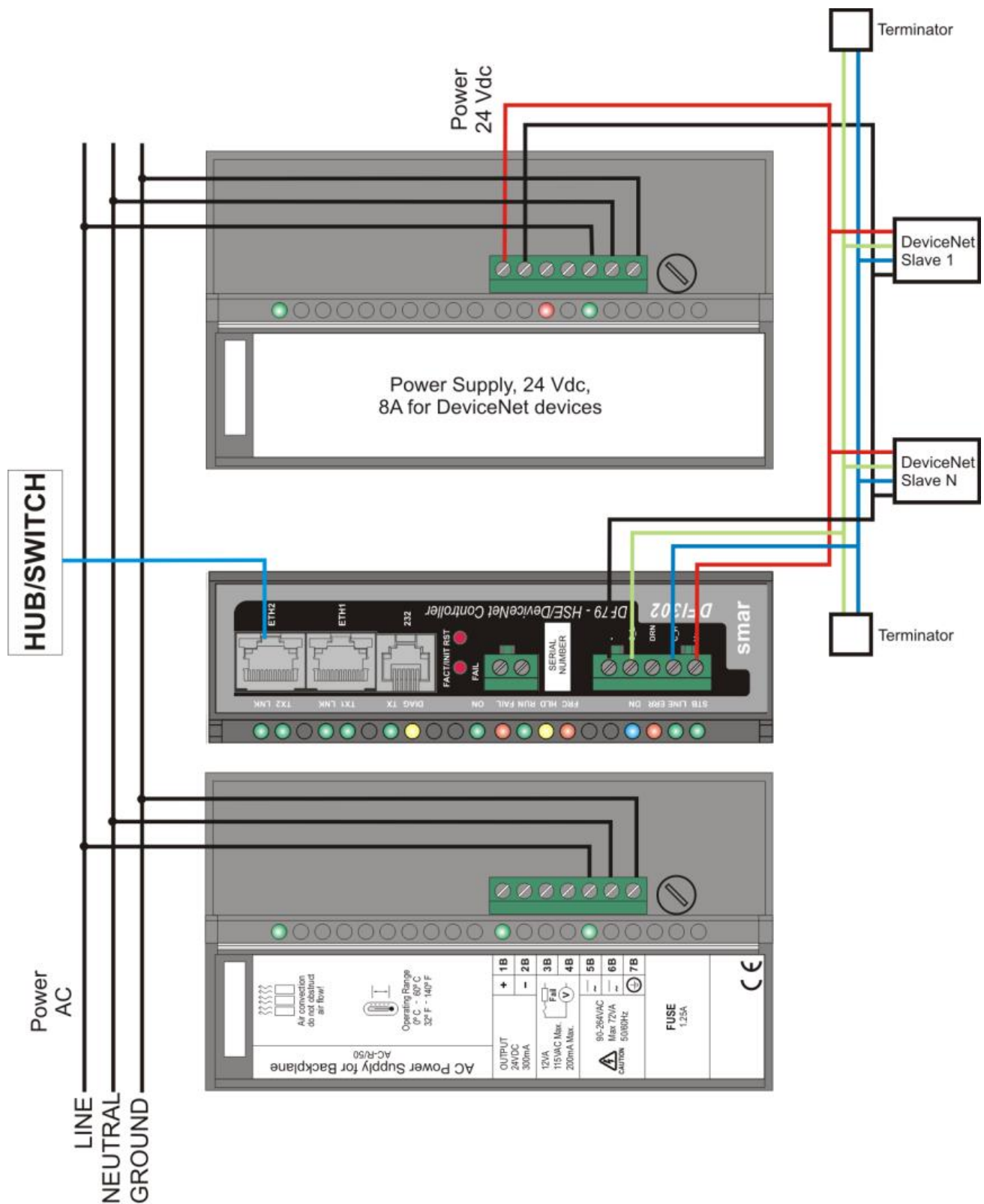


Figure 2. 40 – Cable diagram for DF79

Using the DF81 Controller

A typical system using the DF81 can be composed by:

- DF1A or DF93 – Rack with 4 slots;
- DF50 – AC Power Supply
- DF81 – HSE/AS-i Controller
- DF00 – Empty Module
- AS-i Power Supply 29,5V up to 31,6V (approximately $30.55V \pm 3\%$) and maximum current 8A (one for each channel).

A typical configuration of a system with DF81 controller can be seen in the following figure:

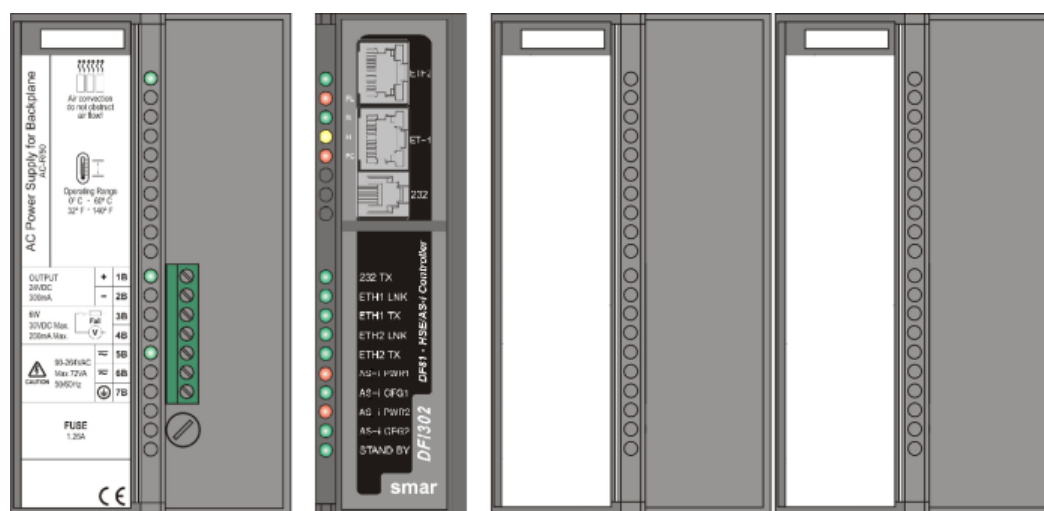
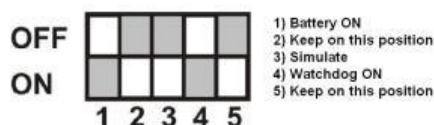


Figure 2. 41 - System of DFI302 with DF81

Important:

The DF81 has an internal battery that keeps the Real Time Clock (RTC) and its non volatile RAM (NVRAM) when there is lack of external supply. This battery can be either enabled or disabled, depending on the position of the switch 1, in the back part of the DF81. To enable the battery, set the switch to 1 as shown in the following figure:



In this configuration, when there is lack of energy, the RTC and the NVRAM will be supplied by the battery, allowing the retention of all configuration data. In case of equipment storage, it is recommended that the battery is turned off (switch 1 in position OFF). For further information about the battery, refer to the "Technical specification for controllers" section.

So, before fixing the DF81 module in the rack, be sure the switch 1, which refers to the battery, is in the enabled position.

The Watchdog is a mechanism to detect if an important or high priority task stops in the controller. So, be sure the switch 4, which refers to the Watchdog, is in the ON position.

Steps for a basic installation:

Execute the following procedures for a basic installation using DF81:

- Connect 4 modules (DF50, DF81, DF00, DF00) and the terminator (DF2) in the rack (DF1A or DF93);
- Connect the power supply in the DF50 input;
- Connect the auxiliary power supply in the channels AS-i CH1 and AS-i CH2, one for each channel;
- Connect the Ethernet cable (twisted pair cable) connecting DF81 to the hub or switch;

- Connect the AS-i bus cable to the channels AS-i CH1 and AS-i CH2 of DF81.

WARNING
NEVER grounded the negative AS-i channel.

- Note there are 2 inputs CH1 and 2 inputs CH2. It can, for example, to connect the cable of the power supply in one of the CH1 input and in the other the AS-i cable (yellow) in the other CH1 input/output for communication and supply of the AS-i slaves.
- DF81 will obtain automatically an IP address from DHCP Server, but if this server is not available, initially have a fixed IP (this initial fixed IP address it may be changed by FBTools (see the "Connecting the DFI302 in the subnet" topic).
- It is important that the IP addresses of the ETH1 and ETH2 ports are different.

See in the figure follow:

In the following figure, the wiring diagram is shown for the use of the DF81:

- Note that only one AS-i segment is being used;
- In the figure the present cable is standardized yellow, in that the brown wire should be connected to the terminator " + " and the blue in the terminator " - "

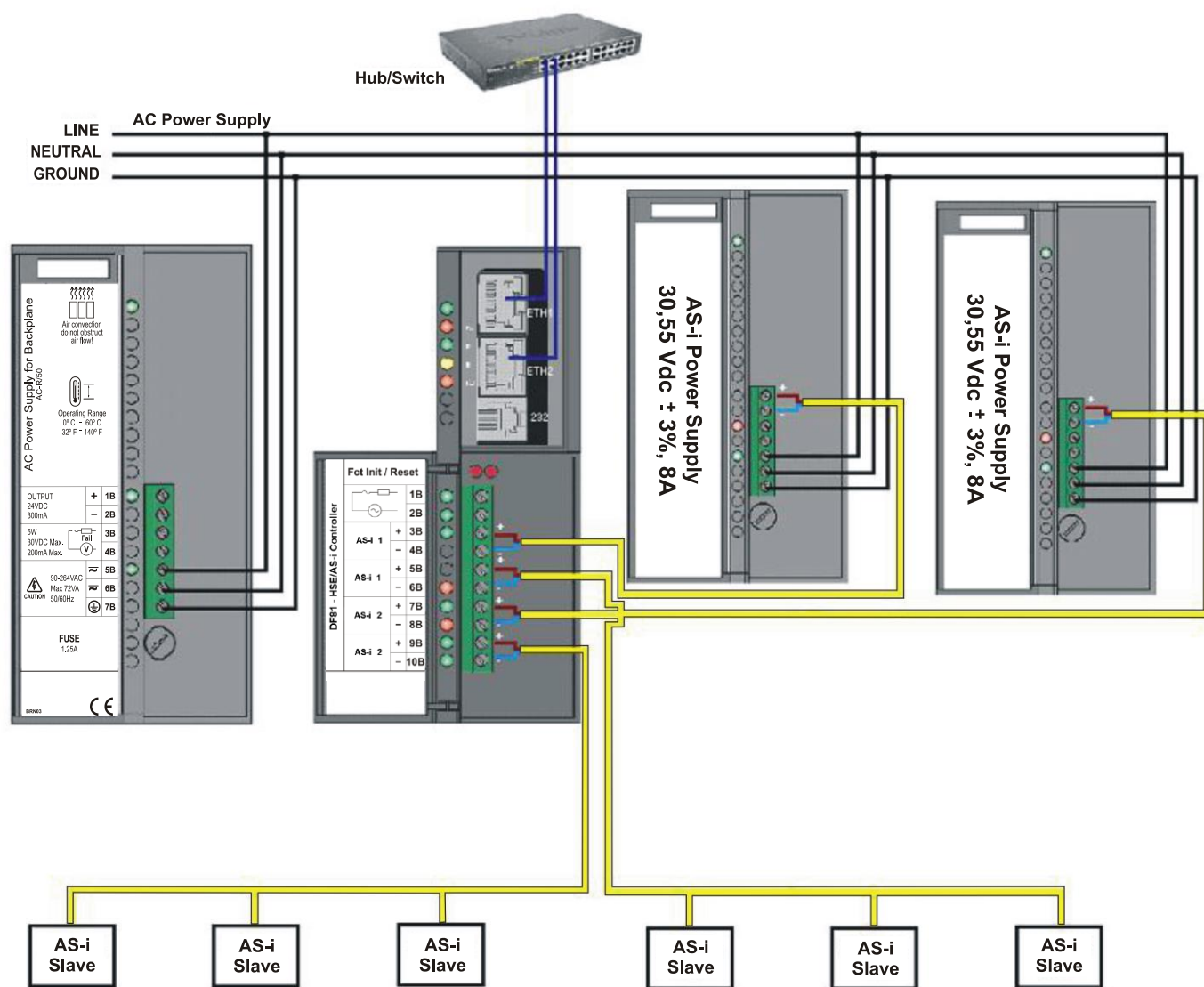
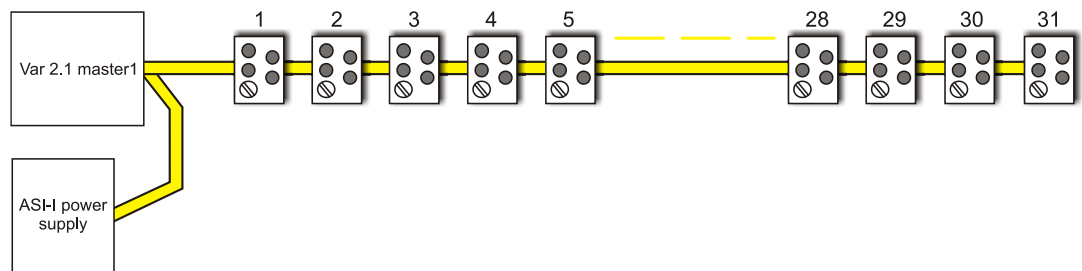


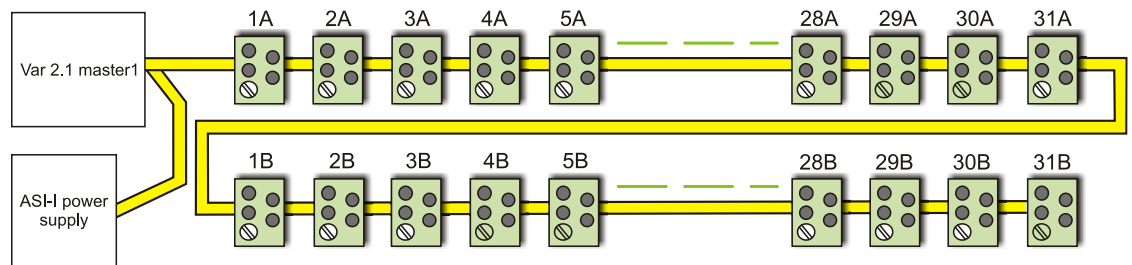
Figure 2. 42 – Cable diagram for DF81

AS-i topology examples:

Example 1: 31 standard slaves



Example 2: 62 A/B slaves



Example 3: Standard slaves, A/B slaves and analog slaves used together

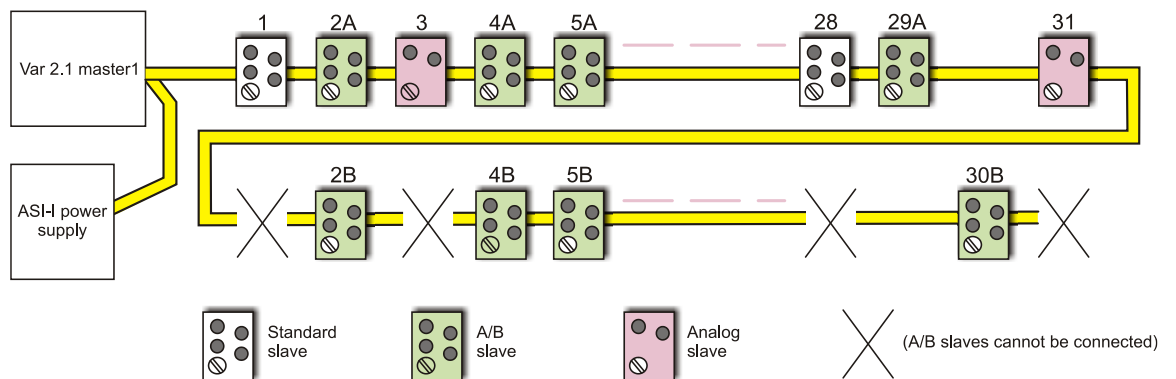


Figure 2. 43 – Bus examples with slaves 2.0 and/or 2.1 versions (include analog). The analog slaves are not supported by DF81

NOTE

The above illustration shows the allocation of slave for easy reference. Actual connections do not require allocation in address order.

Using the DF89 Controller

A typical system is composed by:

- DF1A or DF93 – Rack with 4 slots;
- DF50 –AC Power Supply;
- DF89 –HSE/Modbus Controller;
- Up to 16 DF1A or DF93 with I/O modules

A redundant system by using the DF89 can be composed by:

- DF78 – Rack with 4 slots (Hot Swap of CPUs and redundant I/O access) or DF92 – Rack with 4 slots (for redundante CPUs, support to Hot Swap and diagnostic);
- DF50 –AC Power Supply;
- DF50 –AC Power Supply ;
- DF89 –HSE/Modbus Controller;
- DF89 – HSE/Modbus Controller;
- Plus up to 16 DF1A or DF93 with I/O modules.

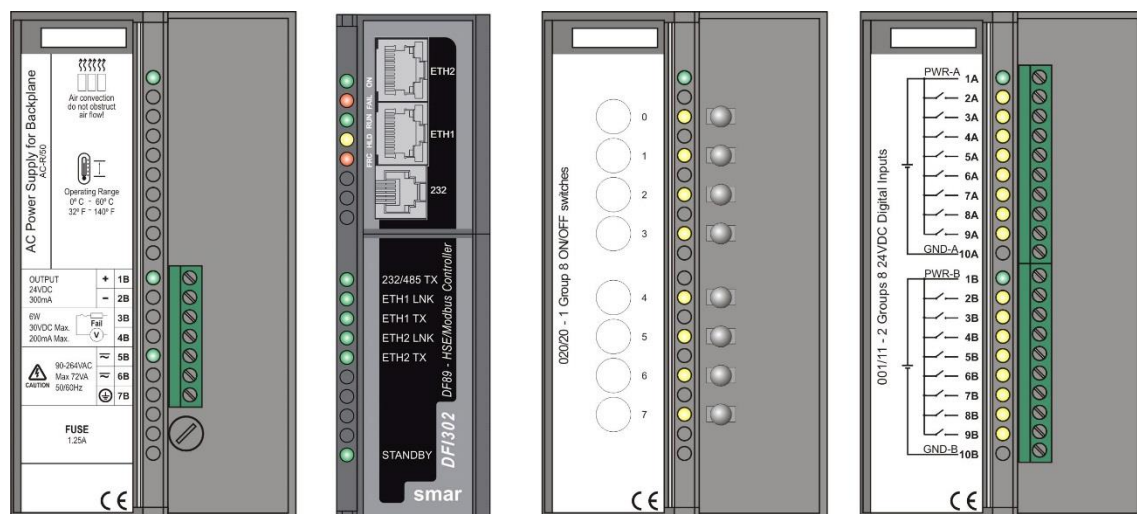
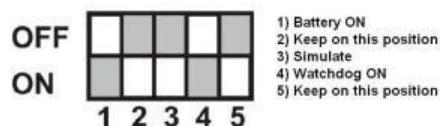


Figure 2.44 - Basic system of DFI302 with DF89

Important:

The DF89 has an internal battery that keeps the Real Time Clock (RTC) and its non volatile RAM (NVRAM) when there is lack of external supply. This battery can be either enabled or disabled, depending on the position of the switch 1, in the back part of the DF89. To enable the battery, set the switch to 1 as shown in the following figure:



In this configuration, when there is lack of energy, the RTC and the NVRAM will be supplied by the battery, allowing the retention of all configuration data. In case of equipment storage, it is recommended that the battery is turned off (switch 1 in position OFF). For further information about the battery, refer to the “Technical specification for controllers” section.

So, before fixing the DF89 module in the rack, be sure the switch 1, which refers to the battery, is in the enabled position.

The Watchdog is a mechanism to detect if an important or high priority task stops in the controller. So, be sure the switch 4, which refers to the Watchdog, is in the ON position.

Observe in the next figures:

- Basic system using the DF89 controller
- Redundant system using the DF89 controller

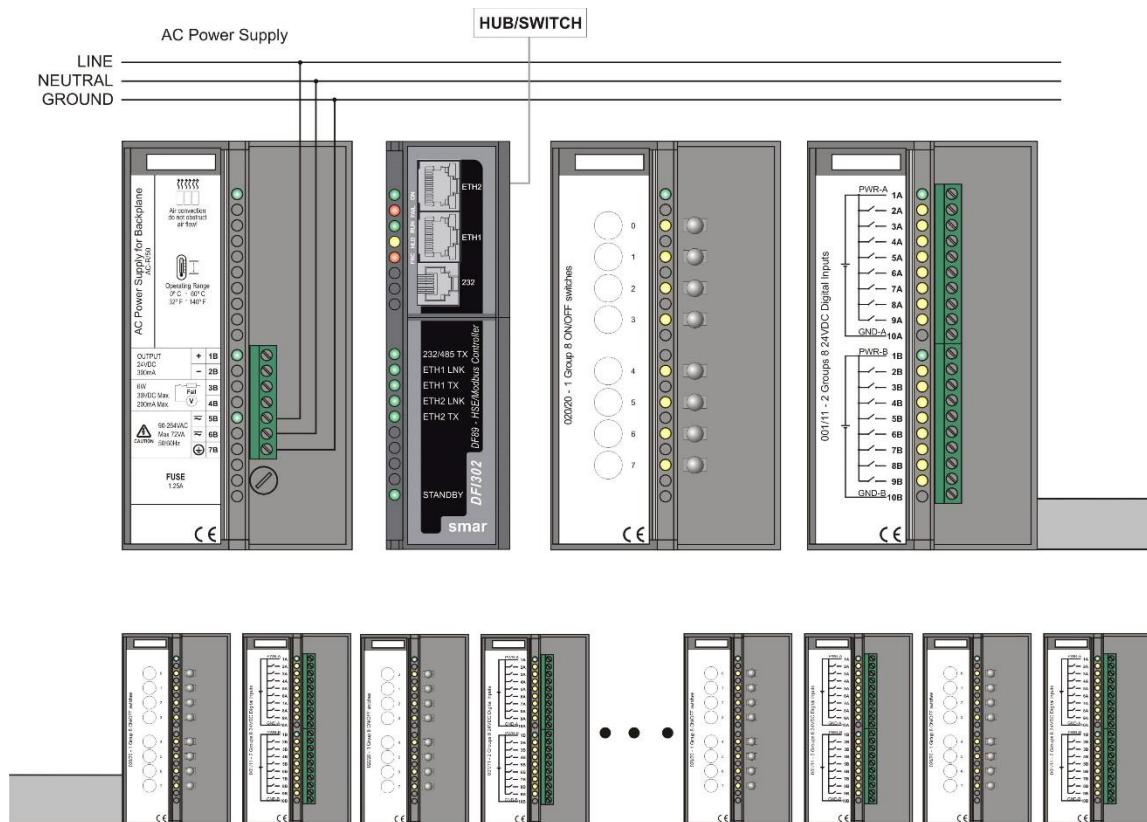


Figure 2. 45 - Basic system using the DF89 controller

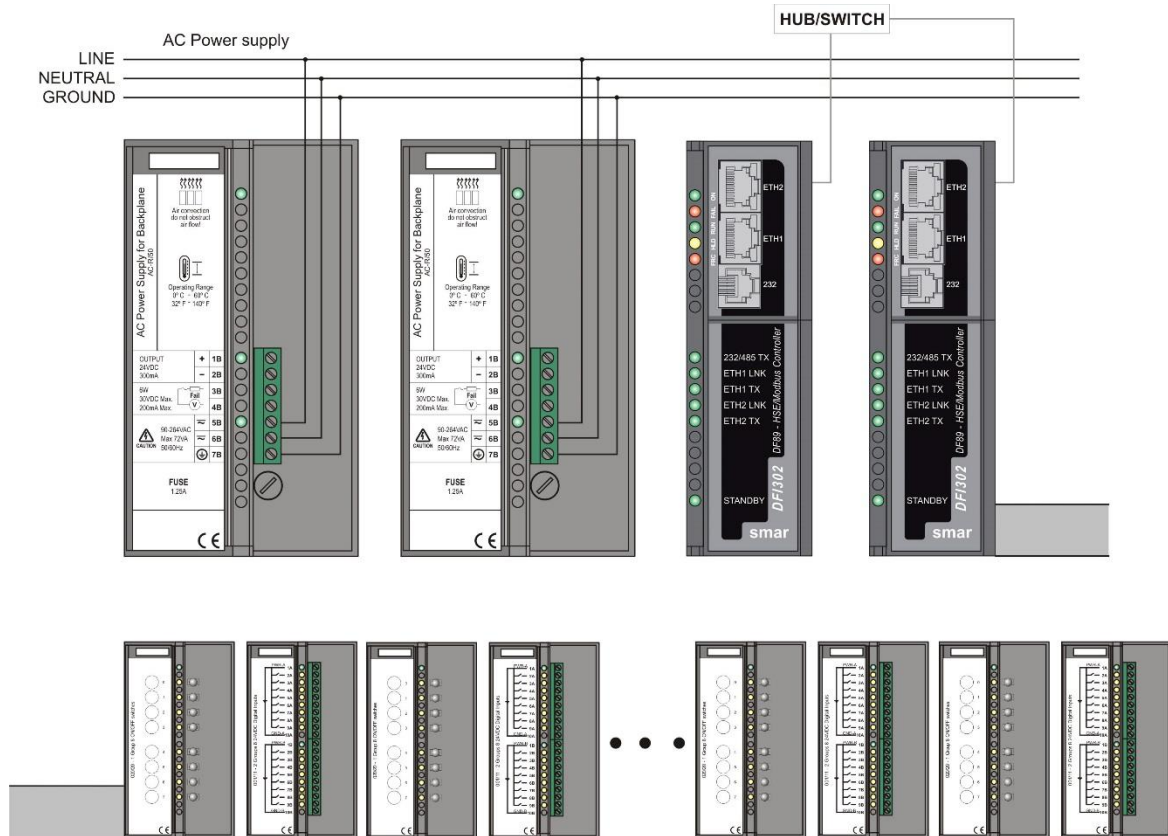


Figure 2. 46 - Redundant system using the DF89 controller

Using the DF95 Controller

A typical system using the DF95 can be composed by:

- DF1A or DF93 – Rack with 4 slots
- DF50 –AC Power Supply
- DF95 – HSE/Profibus DP Controller
- DF52 – Power Supply for Profibus devices
- DF53 – Power Supply Impedance for Fieldbus
- Up to 16 DF1A or DF93 with I/O modules

NOTE

If the DF52 and DF53 modules were not used for powering Profibus devices, the DF0 or I/O modules must be inserted in the empty slots.

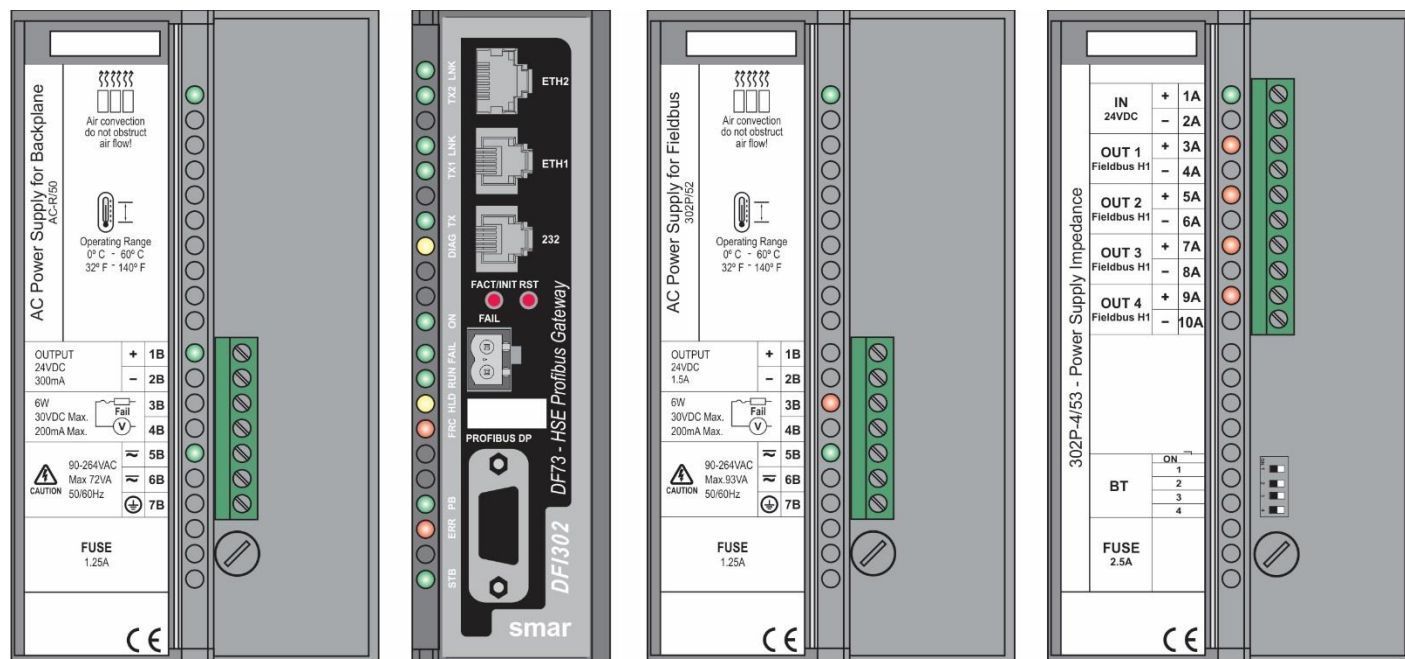
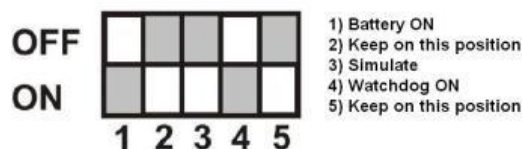


Figure 2. 47 – System of DFI302 with DF95

Important:

The DF95 has an internal battery that keeps the Real Time Clock (RTC) and its non volatile RAM (NVRAM) when there is lack of external supply. This battery can be either enabled or disabled, depending on the position of the switch 1, in the back part of the DF95. To enable the battery, set the switch to 1 as shown in the following figure:



In this configuration, when there is lack of energy, the RTC and the NVRAM will be supplied by the battery, allowing the retention of all configuration data. In case of equipment storage, it is recommended that the battery is turned off (switch 1 in position OFF). For further information about the battery, refer to the “Technical specification for controllers” section.

So, before fixing the DF95 module in the rack, be sure the switch 1, which refers to the battery, is in the enabled position.

The Watchdog is a mechanism to detect if an important or high priority task stops in the controller. So, be sure the switch 4, which refers to the Watchdog, is in the ON position.

NOTE:

If the terminators is positioned at the beginning or end of the Profibus DP network, the terminator must

be put on the ON position.

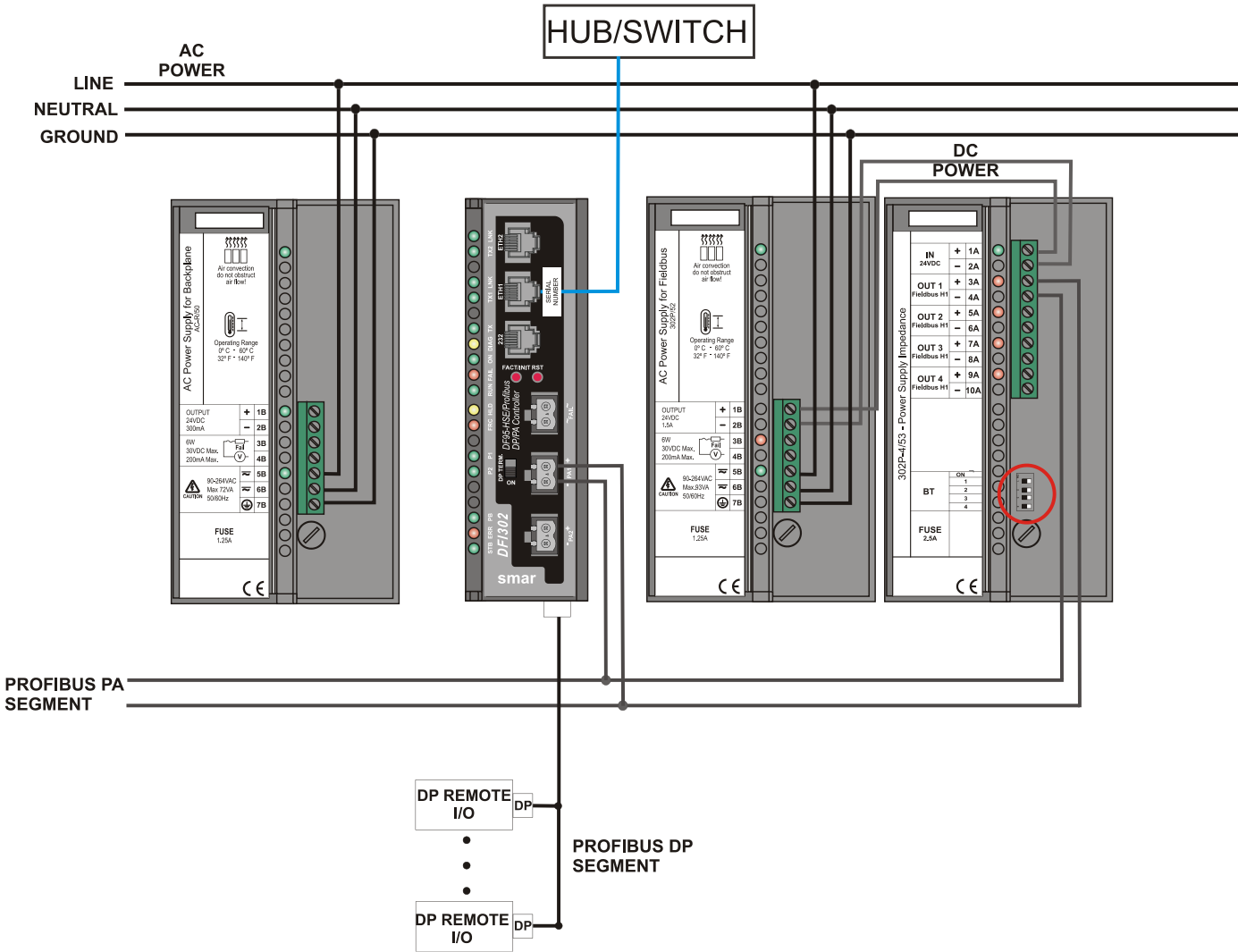


Figure 2. 48 – Cable diagram for DF95

Using the DF97 Controller

A typical system using the DF95 can be composed by:

- DF1A or DF93 – Rack with 4 slots
- DF50 –AC Power Supply
- DF97 – HSE/Profibus DP Controller
- DF52 – Power Supply for Profibus devices
- DF53 – Power Supply Impedance for Fieldbus
- Up to 16 DF1A or DF93 with I/O modules

NOTE

If the DF52 and DF53 modules were not used for powering Profibus devices, the DF0 or I/O modules must be inserted in the empty slots.

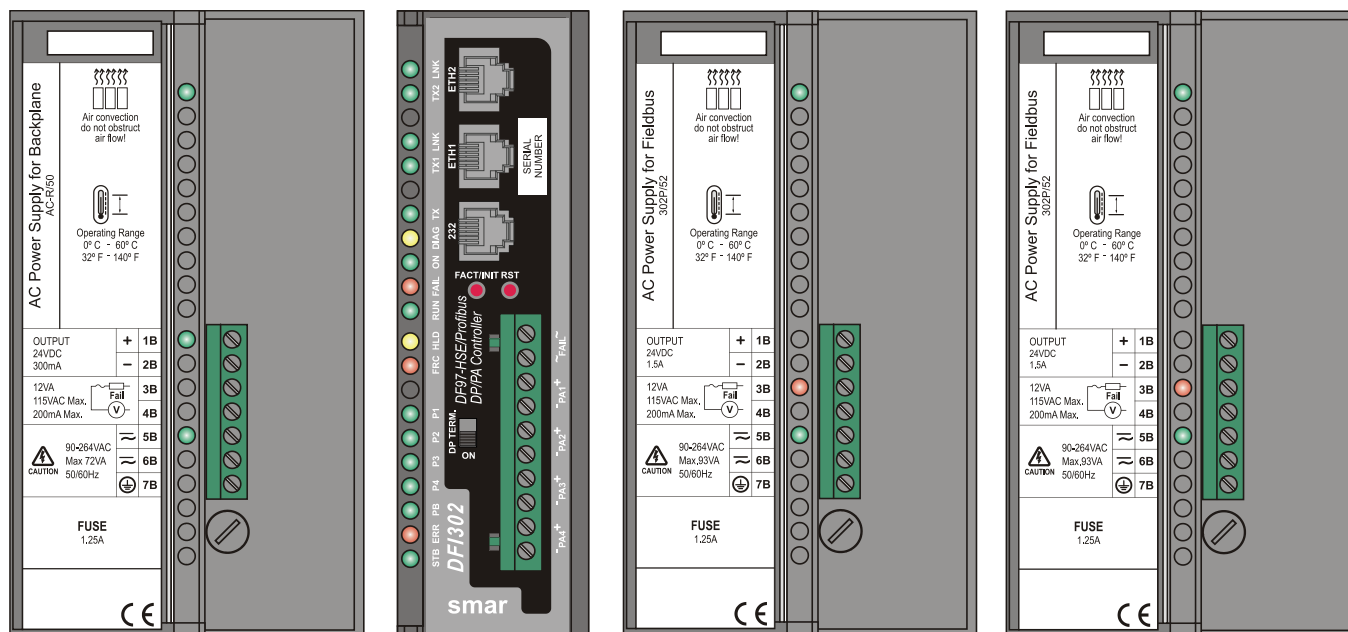
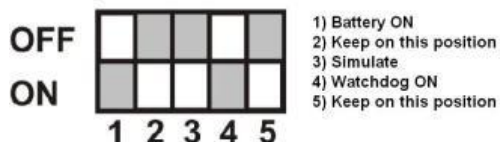


Figure 2.49 – Basic system of DFI302 with DF97

Important:

The DF97 has an internal battery that keeps the Real Time Clock (RTC) and its non volatile RAM (NVRAM) when there is lack of external supply. This battery can be either enabled or disabled, depending on the position of the switch 1, in the back part of the DF97. To enable the battery, set the switch to 1 as shown in the following figure:



In this configuration, when there is lack of energy, the RTC and the NVRAM will be supplied by the battery, allowing the retention of all configuration data. In case of equipment storage, it is recommended that the battery is turned off (switch 1 in position OFF). For further information about the battery, refer to the “Technical specification for controllers” section.

So, before fixing the DF97 module in the rack, be sure the switch 1, which refers to the battery, is in the enabled position.

The Watchdog is a mechanism to detect if an important or high priority task stops in the controller. So, be sure the switch 4, which refers to the Watchdog, is in the ON position.

NOTE:

If the terminators is positioned at the beginning or end of the Profibus DP network, the terminator must be put on the ON position.

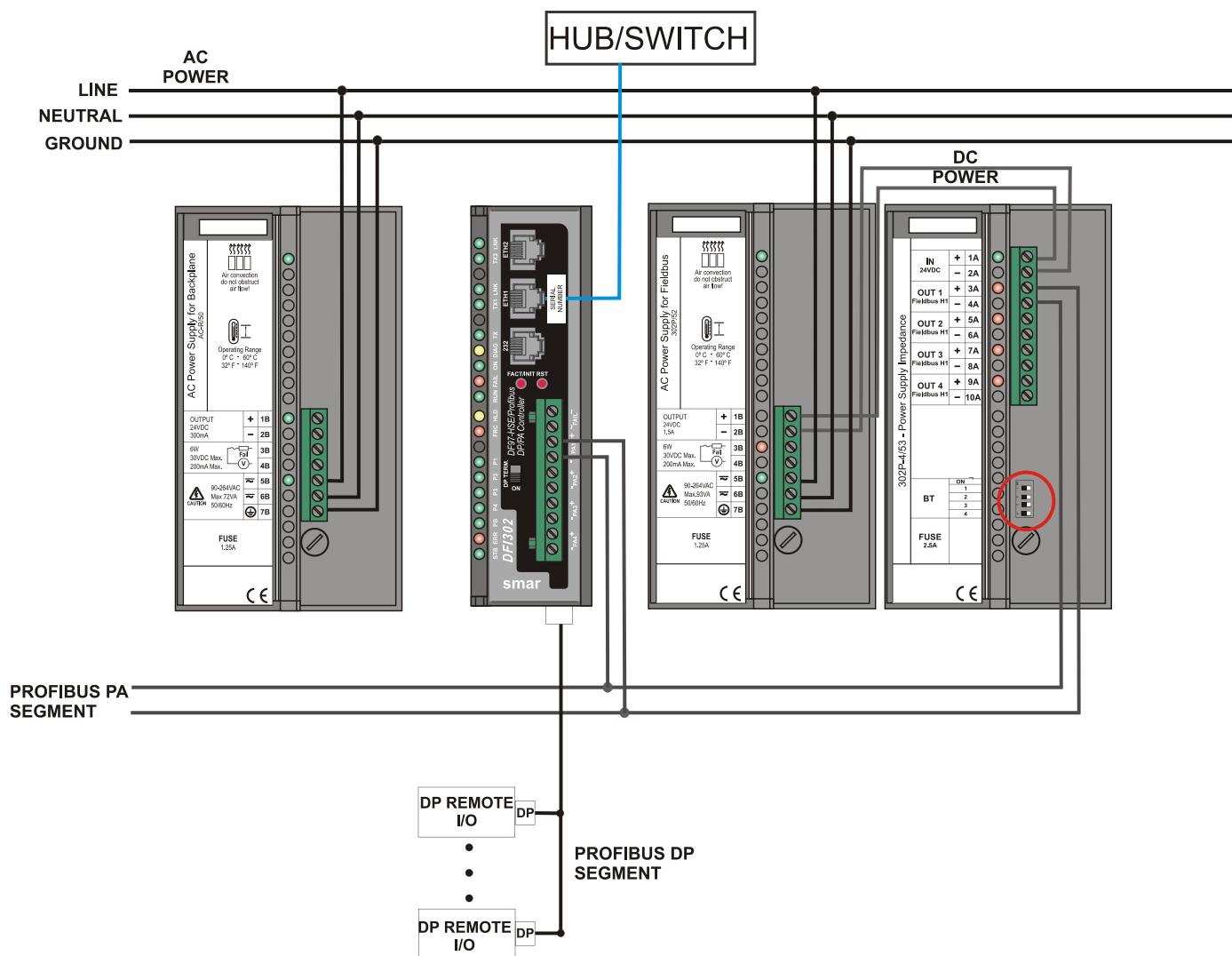


Figure 2. 50 – Cable diagram for DFI302

Using the DF100 Controller

WirelessHART technology overview

The *WirelessHART* technology is based on a wireless mesh network communication protocol used in process automation applications. It adds wireless capabilities to the HART protocol, while maintaining compatibility with existing HART devices, commands and already known and used tools.

WirelessHART network

Basically, a *WirelessHART* network, defined in the HART specifications, consists of a host, a *WirelessHART* Gateway and one or more field devices and/or *WirelessHART* adapters. Together they compose a mesh network where the host and devices can communicate.

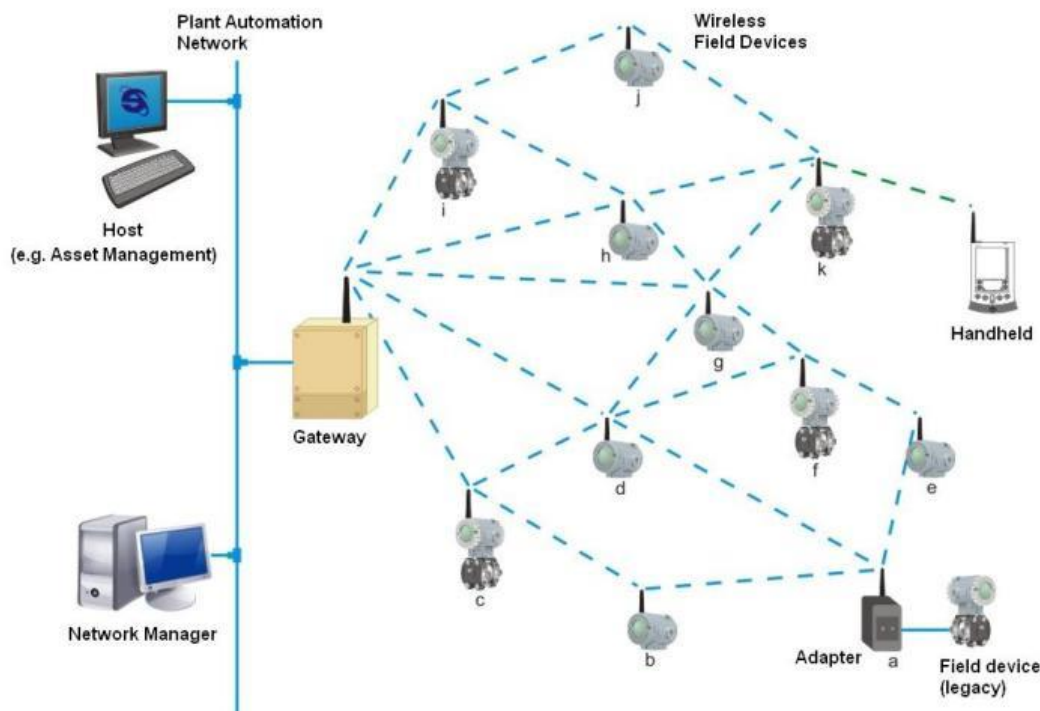


Figure 2. 51 – WirelessHART network

Host

The host, usually connected to the control network, is a workstation in which, e.g., can be installed an Human Machine Interface application, which allows an operator to interact with the process. Through the *WirelessHART* Gateway, the host can gather data from devices connected to the *WirelessHART* network. The host communicates with the *WirelessHART* Gateway using a communication protocol, for example, HSE, H1, Profibus or Modbus.

WirelessHART Gateway

It is a "translator" equipment. Thus it converts data from the host to the *WirelessHART* protocol, used by the devices connected to the *WirelessHART* network, and converts data from the devices to the host. In general, the *WirelessHART* Gateway incorporates the features of Network Manager and Access Point. Roughly, the access point can be understood as the *WirelessHART* radio installed at the gateway to communicate with devices connected to the wireless network.

Network Manager

The Network Manager is an application that can be embedded in the *WirelessHART* Gateway. On a *WirelessHART* network is only allowed to have one Network Manager. Among its responsibilities, the Network Manager distributes network identity (advertisement) publishing its existence, manages and authenticates the addition (joining) of devices to the network. It also distributes individual security keys (static or rotating) to the devices to ensure secure communication between it and the devices. The Network Manager assigns communication band to the devices already connected to the network that requested services to it, as well as manages the routes between the devices on the mesh network.

Specifically about the joining process of a *WirelessHART* device to the network, the Network Manager validates the Network ID and the Join Key attributes which are configured in the *WirelessHART* Gateway and *WirelessHART* devices.

The Network ID identifies a *WirelessHART* network in unique way. It is an **unsigned integer** attribute and must be configured on the *WirelessHART* Gateway and all *WirelessHART* devices. Considering a *WirelessHART* network installed in a plant, the permitted values for the Network ID ranges from 0 (hex 0x0000) to 32767 (0x7FFF hexadecimal).

The Join Key is a security key used to encrypt joining requests from *WirelessHART* devices that receive the advertisement with the Network Id identical to theirs. It may be single or each *WirelessHART* device may be configured with an individual Join Key. In the first case, the *WirelessHART* Gateway and all *WirelessHART* devices must be configured with the same Join Key. In the second case, which provides higher communication security level, (a) must be configured in the *WirelessHART* Gateway a list with individual Join Keys, i.e., a key for each *WirelessHART* device, and (b) you must configure each *WirelessHART* device with its individual Join Key. The Join Key is a hexadecimal string of 16 bytes. There is no restriction to the hexadecimal value of each byte. The table below shows examples of some join keys.

JOIN KEYS	16-BYTES HEXADECIMAL STRING
00000000000000000000000000000000	0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
000000000000000000000000000000302	0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x02
00000000FFFFFFFF0000000000000000	0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
550000000000000000000000000000AA	0x55, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xAA

Examples of Join Keys

WirelessHART device

The *WirelessHART* field device is the device that connects to the process, being able to receive and/or transmit data on the *WirelessHART* network. It is a *WirelessHART* router (repeater) by nature, i.e., it is able to retransmit messages to/from other devices on the *WirelessHART* network.

WirelessHART Adapter

It is a bridge-type device, because it is able to provide data of HART + 4 to 20mA field device, legacy, to the host via *WirelessHART*. The adapter uses HART FSK standard communication, wired, to access data from HART field devices. And the adapter also uses the *WirelessHART* communication to provide data of the field device to the host. The adapter thus enables a HART field device to work on *WirelessHART* network.

We recommend a visit to the HART Communication Foundation website for additional information about the *WirelessHART* protocol such as *WirelessHART* project planning, positioning of devices, commissioning and verification tools, and practices.

Planning an *WirelessHART* network

The planning of a *WirelessHART* network is a task that is very similar to the activities that currently we perform with conventional wired devices. Furthermore, due to the simplicity of a mesh *WirelessHART* network, is exempt, in general, detailed field surveys, which are usually needed when we plan networks based on other wireless technologies.

Basically, a *WirelessHART* network involves planning, design, installation and commissioning phases.

Planning

This phase requires the execution of the steps below:

Scope definition

Clearly define the scope of the network. Answer the question: why do we need the wireless network? To monitor process variables or to implement a non-critical control? The answer to this question will facilitate the understanding between the team members responsible for the network and determine

one or more process units in the plant. For each process unit, allocate a gateway with unique and specific Network ID. Outline the main field devices.

Identify potential sources of interference

Are there radio communications or other wireless networks in the plant? What protocols and frequencies do they use? Use high power? Although unlikely, given the robustness of the radios used by the *WirelessHART* technology, prior knowledge of the answers to these questions may identify potential sources of interference and to indicate the taking of preventive and/or limiting actions even before installation. For example, you can select a frequency channel as unavailable, adding it to the black list of frequencies that is under the *WirelessHART* Network Manager control.

Integration with the host

The gateway connects the *WirelessHART* field devices to the host system. Plan what devices and what data are needed. Also, the stations or applications which will process the data have to be clearly defined. From this set, among the protocols in the system, define which one will be used for integration with the host and with the existing tools for configuring the devices. After defining the protocol for integration, the user has to choose the gateway on the market that best meets your requirements.

Project

In the project phase, it is recommended the adoption of the practices below. Although conservative, these practices ensure robustness and scalability to the network.

- Define the Network ID that will be used for all devices in the process unit;
- Define if the Join Key will be common to all devices or individual and dedicated ;
- Define the policy to be used for the definition of devices (Long) Tags;
- Use a scale drawing of the process unit;
- Place the gateway in a strategic position in the process unit ;
- Plan networks with at least five devices;
- Install at least five devices within the gateway coverage area;
- Ensure that 25 % of the devices are within the gateway coverage area;

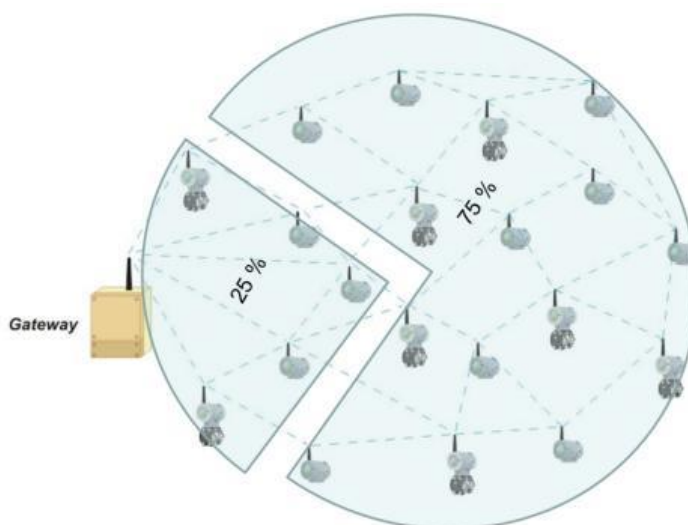


Figure 2. 52 –Gateway coverage area

- Reposition the gateway as needed;
- Check the coverage area of each device;
- Ensure that each device has three neighbors within its coverage area;

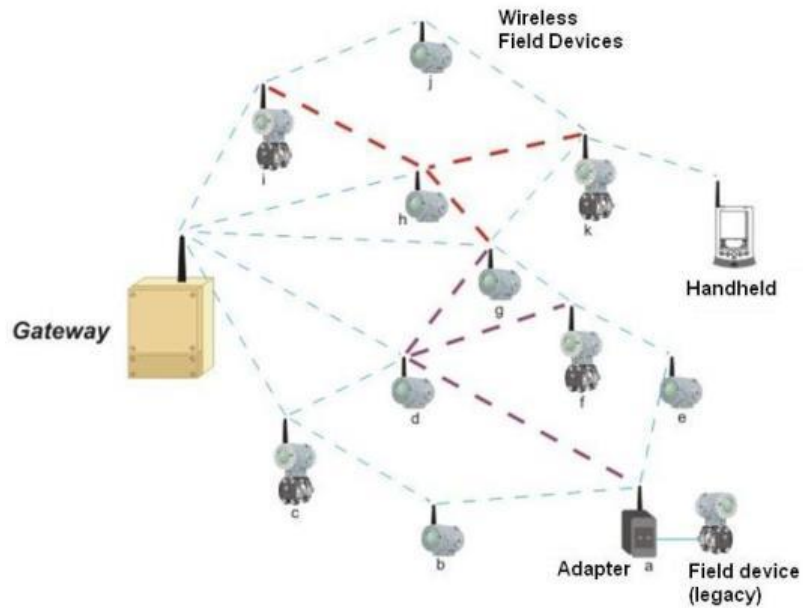


Figure 2.53 – WirelessHART devices vicinity

- Place the repeaters as needed.

Installation

As mentioned before, *WirelessHART* devices should be connected to the process and configured the same way as conventional wired HART devices.

Handheld terminals can be used normally. Just be sure of having it properly uploaded with the latest DD files of the devices. However, it is known that the *WirelessHART* devices have characteristics inherent to the technology. Because of this, it is recommended the adoption of practices mentioned below for positioning the gateway and devices.

- Install the gateway and the devices so that their antennas are vertical;
- Ensure that the antennas are at 0.5 m minimum distance of large obstacles or surfaces;
- Ensure that the antennas of gateway and repeaters are 2 m above most obstacles within their coverage areas;

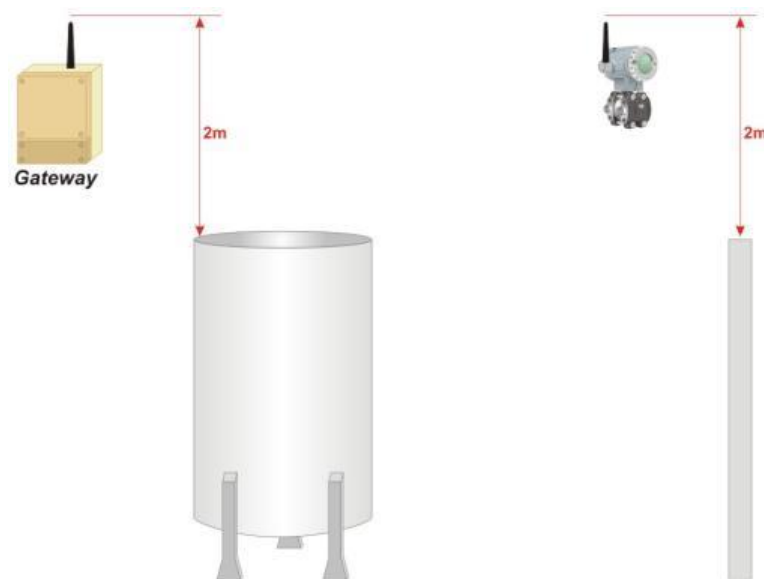


Figure 2.54 – Gateway and repeater 2m above the obstacles

- If there are high devices, does not exceed 45° viewing angles between them;

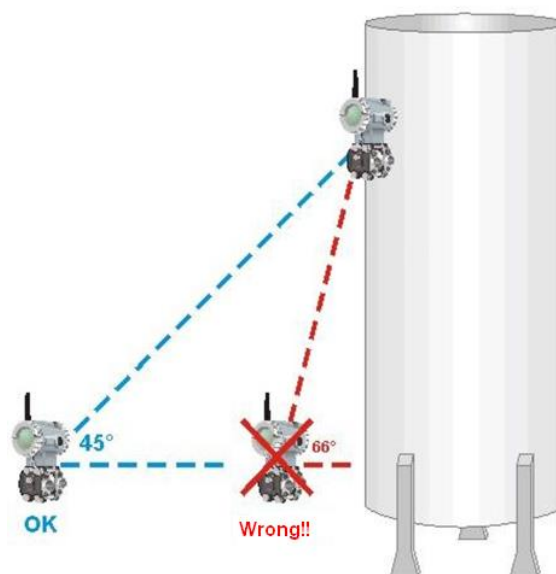


Figure 2. 55 – Device's viewing angle

- Make sure that the gateway is integrated to the host system as planned.

Commissioning

The commissioning of devices and gateway must be considered¹.

WirelessHART devices commissioning

- Ensure that the gateway is installed and powered;
- Install each device individually. Start with those closest to the gateway, i.e., those that will be within the coverage area of the gateway;
- If the device is powered by batteries, check that they have the same characteristics documented in the device's operation manual;
- Power the device up;
- Use a handheld terminal and configure the device according to the application requirements;
- Configure the Long Tag of the device;
- Configure the Network ID;
- Configure the Join Key;
- Define and configure the update rate;
- Command, if necessary, the device connection to the network;
- Follow the device connection to the network, waiting until it reaches the operational state. The monitoring can be done from the device² or gateway;
- Make sure the device is operating to ensure its commissioning. For example, check the value of PV measured and its update rate.

Gateway commissioning

- Make sure that the gateway is available to the host system;
- Check the gateway and make sure it has at least five devices directly connected to it;
- Check if 25 % of the devices are connected directly to the gateway. If necessary, add repeaters;
- The gateway connects the devices to the host system. Thus, check if the data of the devices are coming to the applications that subscribe them.

Basic Characteristics of DF100

The DF100 is an HSE/*WirelessHART* controller. Integrates up to 100 *WirelessHART*³ devices to the SYSTEM302 through recently standardized FF HSE WIO technology and *WirelessHART*. Thus, the DF100 is a controller that provides a *WirelessHART* device data to workstations and controllers connected to corporate and control networks. It is a more powerful Smar controller, which, besides

¹ The steps below assumes that the Network ID and the Join Key(s) are already configured.

² Refer to the device's manual to learn procedures for such verification.

³ The maximum number of *WirelessHART* devices may be less depending on the update rate desired.

the HSE protocol, can be integrated with existing host systems via Modbus TCP and RTU (via RS-485 port). Unlike other controllers of the DFI302 family, the DF100 does not use the rack, nor supports I/O modules. In addition, the DF100 has IP66 rating which allows their installation outdoors, open, whose temperature range varies from -40 °C to +60 °C.

Package Contents

The package of DF100 controller contains the following items:

- 01 DF100 Controller (installed in the IP66 metal housing, with grounding block);
- 02 Cable glands installed in the IP66 metal housing;
- 02 Plugs (installed in the IP66 metal housing, next to the cable glands above mentioned);
- 01 Antenna protective cover;
- 01 Omnidirectional antenna;
- 02 1/4" U clamps (for housing fixation in pipes);
- 04 1/4" pressure washers (for housing fixation in pipes);
- 04 1/4" hex nuts (for housing fixation in pipes);
- 02 Cable glands (spare parts – for Ethernet and RS-485 communication);
- 01 Fuse (spare part);

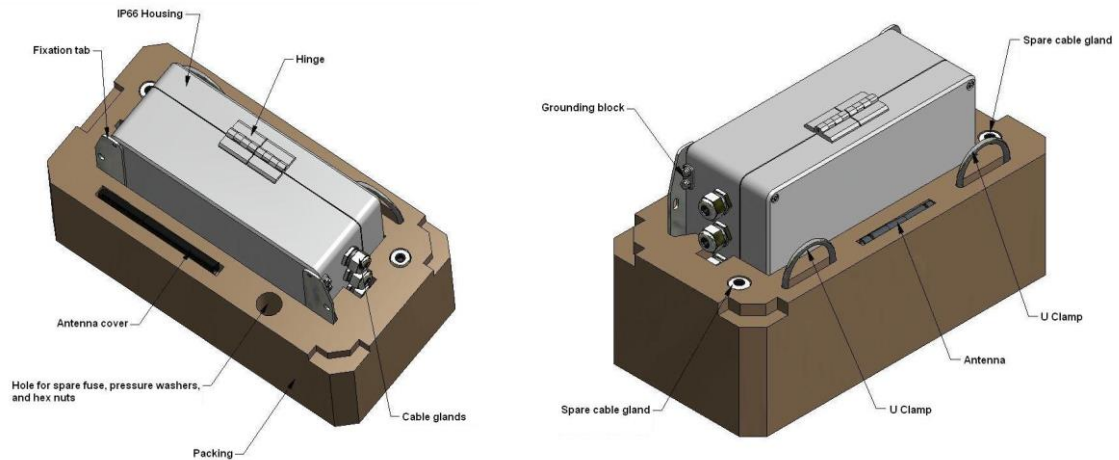


Figure 2.56 – Package detailing

Mounting the Antenna

The DF100 controller received must be mounted before being powered for the first time. Its mounting is very simple and resumes to the mounting of the antenna. Although simple, requires that primary care be taken against static discharges during installation (see Preventing Electrostatic Discharge section at the beginning of this manual).

Once opened the package, remove the IP66 metal housing of the controller, the omnidirectional antenna, and its protective cover. The antenna and the protective cover are embedded into polyurethane packaging that protects the DF100. The following figure shows and identifies each of the above mentioned parties.

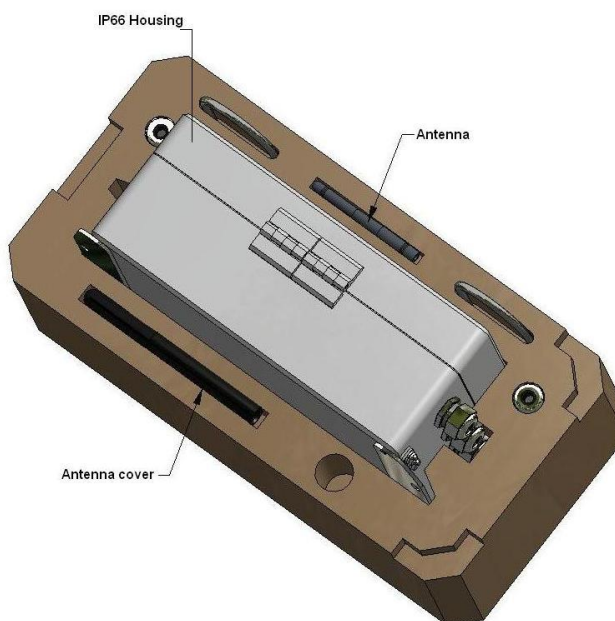


Figure 2. 57 – Parts needed to mount the antenna

Initially, firmly screw the omnidirectional antenna in the SMA (gold) connector existing on the antenna adapter. The antenna adapter is available on the outer IP66 metal housing, in its upper part, right. Once installed the omnidirectional antenna, finally screw the antenna protective cover until the base of the antenna adapter. The figure below shows the steps for mounting the antenna.

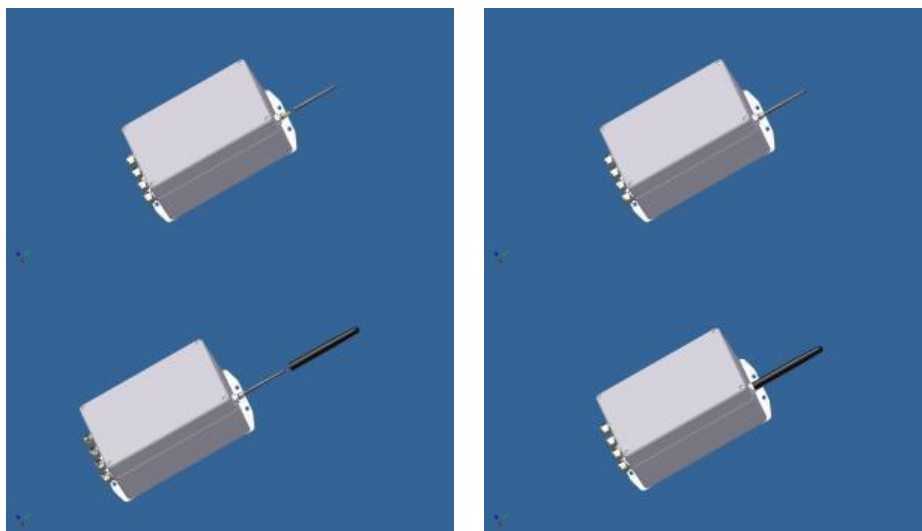


Figure 2. 58 – Mounting the antenna

Connections and Power Supply

DC Power Supply

The DF100 must be powered with a DC power supply. The following figure shows the electronic circuit board and the female⁴ connector, to which the power supply must be connected.



Figure 2.59 – Power supply connector

The nominal output voltage of the power supply must be 20-30 Vdc, knowing that the DF100 consumes 11W, max. We recommend using a DC power supply with rated output voltage of 24 Vdc. In this case, the power supply must be able to provide at least about 459 mA at 24 Vdc.

Before connecting the power supply to the DF100 and power it, make sure that the polarities (+ and -) and reference (///) indicated in the warning label, located on the mounting plate, and in the screen printing of electronic circuit board were respected.

NOTE

The procedures described below require opening the IP66 housing of DF100, which will expose parts of the product electronic circuitry. It is recommended therefore:

1. Disconnect the DF100 power connector before performing any of the procedures described below;
2. Disconnect any communication cables that are connected to the Ethernet and RS-485 connectors;
3. Respect labels and screen printings warnings of the product to avoid any electrical damage to electronic parts of DF100, such as those due to electrostatic discharges that can occur during handling the product;
4. Avoid performing the procedures described above in humid environments

Fuse – Location and Replacement

The DF100 electronic circuit is protected by a fuse of 5 x 20mm, 2A, 250V. It is inside a plastic housing, located nearby the female connector that receives the supply voltage input. See the following figure.

⁴ The male connector is supplied with the product.

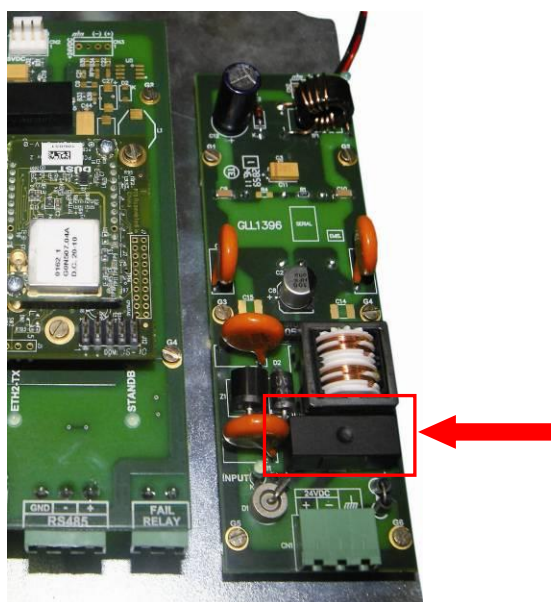


Figure 2. 60 – Fuse location

If necessary, follow the steps below to replace the fuse⁵:

1. Place a finger on the right side of the plastic housing cover and pull it slightly upwards. This will unlock the cover housing base;
2. Remove completely the housing cover⁶ to access the fuse to be replaced;
3. With the aid of, for example, a plastic rod, gently push the fuse from left to right, while the pulling lightly upward. This will release the fuse left end from the terminal socket and will allow that the removal operation can be finalized with the fingers;
4. With the empty fuse socket, initially place the ends of the new fuse on terminal socket and then press it down to make sure its perfect fit;
5. Take the housing cover and place it on its base . Press it down lightly until you hear a click.

Ethernet connection

The IP66 housing of DF100 has, on its bottom, cable glands specially designed to accommodate CAT5 Ethernet cables. However, it is not possible to pass an Ethernet cable (with RJ45 connectors) through cable glands. So follow these steps to prepare Ethernet cables, direct (straight through) or crossover for the DF100.

The following materials are required:

- CAT5 Ethernet Cable, shielded, with the required length;
- For each Ethernet cable, two RJ45 connectors, shielded;
- Crimping pliers for RJ45 connectors;
- Wrench;
- Phillips screwdriver.

Follow the steps below to prepare and connect the Ethernet cable to the DF100.

⁵ The DF100 is supplied with a spare fuse.

⁶ Do not dispose it because it will be used later on this same procedure to seal the plastic housing after installing new fuse.

Using the crimping plier, crimp a shielded RJ45 connector to one end of the Ethernet cable.



Using a Phillips screwdriver, unscrew the four screws located on the top, outside of the DF100 IP66 housing cover.



Open the DF100 IP66 housing.



Using a wrench, unscrew the cable gland protect cover, reserved for the Ethernet cable.



Remove the cable gland protect cover.



Pass the end of the Ethernet cable that was not crimped through the cable gland.



Put another RJ45 connector at the end of the Ethernet cable.



Again, using the crimping plier, crimp the connector at the end of the Ethernet cable that was passed through the cable gland.



Connect the cable to the ETH1 port of DF100.



Adjust the cable inside the DF100 housing and replace the protect cover on the cable gland.



Using a wrench, screw tightly the cable gland. Repeat this procedure if the ETH2 port is used for network redundancy (see notes).



NOTES

- It is recommended that the preparation of Ethernet cable be performed by a qualified electrician that follows the instructions contained in the Cable Specifications section in this manual.
- If the DF100 ETH2 port is not used, for HSE network redundancy, remove its respective cable gland (if installed) and replace it with a stainless steel plug (cap).

After connecting a CAT5 Ethernet cable, shielded, in the DF100 ETH1 port is necessary to connect it in the HSE control network hub/switch. If you prefer to connect to DF100 ETH1 port to the Ethernet network card of a personal computer or notebook, use a crossover Ethernet cable, shielded. Make sure that the length of used Ethernet cable does not exceed 100 m.

NOTE

For HSE network redundancy, consider the procedure described above and connect another Ethernet cable to the port ETH2 of DF100.

If the Modbus communication is not used, make sure to properly close the DF100 IP66 housing after the connection operation, described above.

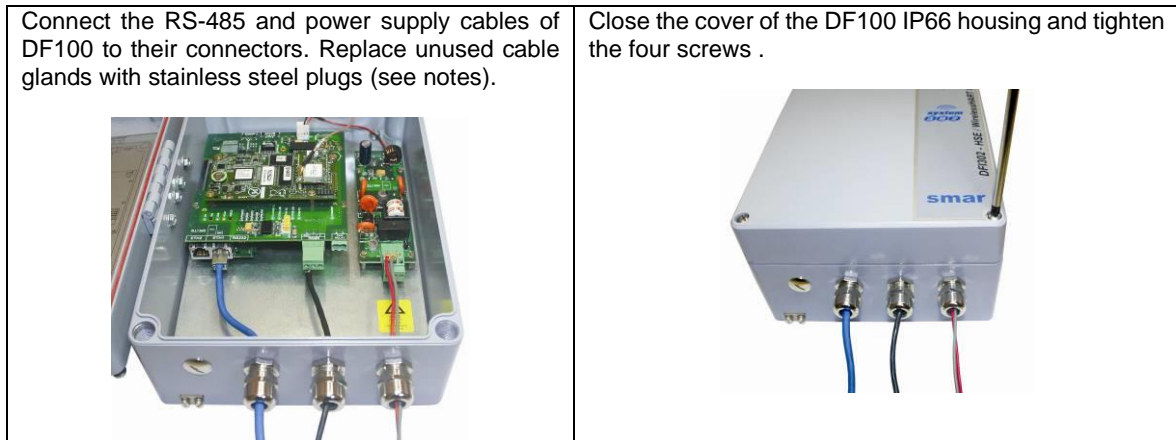
RS-485 Connection

The RS-485 connection requires 3 wires, for differential signals positive (+) and negative (-) and ground GND, respectively. It is recommended to use twisted pair cable, shielded, 2 x 1.5 mm² and terminating resistors at both ends of the bus⁷. Connect the shield to the GND ground pin. The DF100 operates at baud rate of 19200 bps, max. Thus, it is also recommended that the cable length does not exceed 1200 m.

⁷ The DF100 has switches to enable the termination resistor.

The RS-485 connector, which exists on the interface board is physically identical to the connector used in the input power supply board of the DF100. However, for its connection, the RS-485 needs to be rotated 180° relative to the power supply. In addition to the guidelines contained in the DF100 informational tag⁸ about the connector placement, this fact alerts the user when installing the connector to the RS-485 port and helps avoiding mistakes.

See below the finalizing of the connection procedure of RS-485 and power supply cables to the DF100.



NOTES

- If the Modbus communication is not used, remove its respective cable gland (if installed) and replace it with a stainless steel plug (cap).
- Make sure to properly close the DF100 IP66 housing after completing the connection operation, just described above

Configuring the DF100 to operate on a *WirelessHART* network

The DF100 is a DFI302 controller, and therefore has all the common characteristics of DFI302 controllers (eg, Reset, Factory Init, default IP address) are configured similarly. Thus, we describe below only the specific characteristics of the DF100.

Factory Init/Reset

The same options for Reset and operation modes that exist in other DFI302 controllers are also available for DF100. The Factory Init/Reset push-buttons in the DF100 are identified as **W1** and **W2**, respectively, and are placed at the top front of the interface board. Refer to Troubleshooting Section in this manual for details on operation modes.

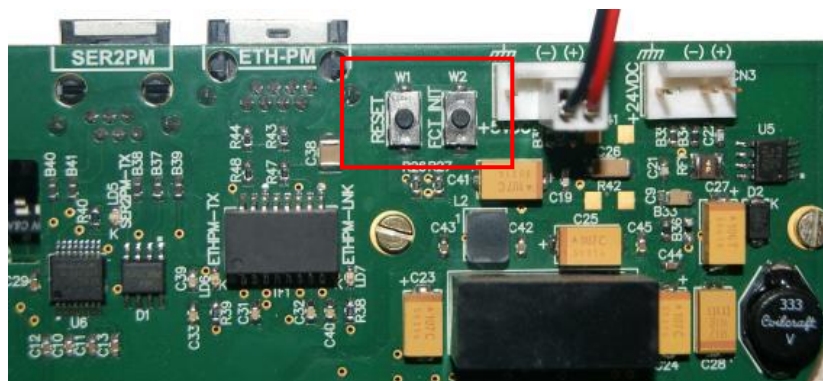


Figure 2. 61 – Detail of Factory Init and Reset push-buttons

⁸ The informational tag is located on the internal side of the IP66 metal housing cover.

NOTE

The informational tag contains the W1 and W2 BUTTONS table, which outlines how the push-buttons W1 and W2 may be used to apply an operation mode to DF100.

RS-485 Communication

The DF100 has one EIA-485 port and RS-485 switches (SW1, SW2, SW3 and SW4) that are available in the interface board, near to the RS-485 connector, to configure the *pull-up and pull-down* resistors and a terminating resistor to suit the communication bus. Refer to Technical specifications for the controllers section for further information about the RS-485 switches.

WirelessHART Network Identifier (Network Id)


The DF100 Web Server is the tool to be used for configuring the *WirelessHART* Network Id. Power the DF100 and launch a browser. In the address field (URL) of the browser, type the IP address⁹ of the current DF100 that will be configured. Then press the ENTER key. The DF100 home page will open.

From the home page, select the *WirelessHART* network configuration page, from **WirelessHART > Network.> Configuration**.


Figure 2. 62 – WirelessHART configuration page

Click the box associated to the *WirelessHART* Network Id, named **Access Point Network Id** and type a valid value, and then click **Apply Changes** button. Valid values for the Network Id are described in the Network Manager topic.

⁹ All DF100 is configured with default IP address equal to 192 168 164 100. If DF100 does not respond when is requested this address, we suggest the application of Mode 3 - Assign Automatic IP. To do so, use the push-buttons of Factory Init/Reset (W1 and W2).



Welcome to DF100 Webserver
TAG: DF100_WHG
Serial number: 3

 SMAR

- Information
- Setup
- Device Info
- WirelessHART
 - Network
 - Status
 - Configuration**
 - Motes
 - Security
 - Alarms
- NET I/O
- Manager

Network Configuration

Access Point Network Id:

32000

Maximum Number of Motes:

100

Network-wide minimum bandwidth:

100000

Maximum bandwidth limit for a single mote:

400

Maximum bandwidth limit for pipe:

480

Advanced Settings

Optimization:

☒ ON
☐ OFF

Access Point PA:

☒ ON
☐ OFF

Clear Channel Assessment:

☐ ON
☒ OFF

Bandwidth Profile:

☐ Manual
☒ P1
☐ P2

Upstream:

1024

Downstream:

256

Advertising:

128

Apply Changes

Clear Changes

Figure 2. 63 – Configuring the Network Identifier

NOTE

The standard value of the Network Identifier for the DF100 is 302 (0x012E, hexadecimal). User is recommended to replace this value by another that suits your needs.

ATTENTION

The value assigned to the *WirelessHART* network identifier must be kept confidential.

Join Key of the *WirelessHART* network

The DF100 WEB Server is also used for configuring the Join Key. With the DF100 powered and using a browser, launch the home page of the DF100¹⁰.

From this home page, select the *WirelessHART* network security page, from **WirelessHART > Network> Security**.

[illegible]

Figure 2. 64 – WirelessHART security page

A dialog box will appear for user authentication. Log in using *<smar, smar>* for the fields *<User name, password>* of the dialog box, and then, click **OK**.

¹⁰ To launch the DF100 home page, review the section on configuration of *WirelessHART* Network Identifier.

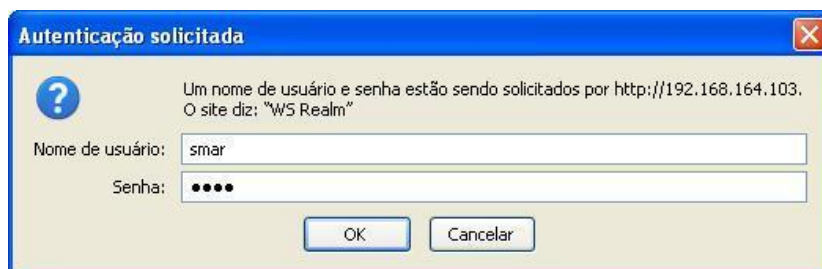


Figure 2. 65 – Dialog box for authentication

The page **Security Management** will open. Select the **Accept Common Join Key** and in the field **Common Join Key** type the desired key. And then, click **Apply Changes** button.



Figure 2. 66 – Configuring the Join Key

NOTE

The Standard value of the Join Key for the DF100 is 0000000000000000000000000302 (0x0000000000000000000000000302, hexadecimal). User is recommended to replace this value by another that suits your needs.

ATTENTION

The value assigned to the join key must be kept confidential.

Installing the DF100 in the field

As we have seen, unlike the other DFI302 controllers, the DF100 was designed to operate outdoors. This controller is designed with mounting tabs that allow its installation in walls or pipes.

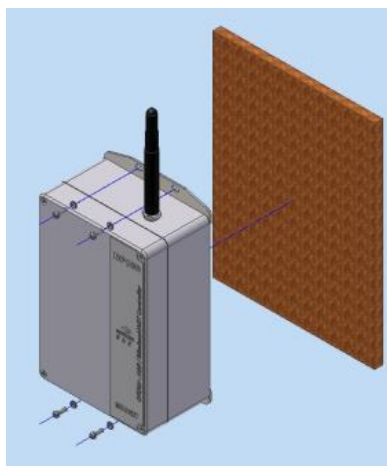


Figure 2. 67 – Wall mounting

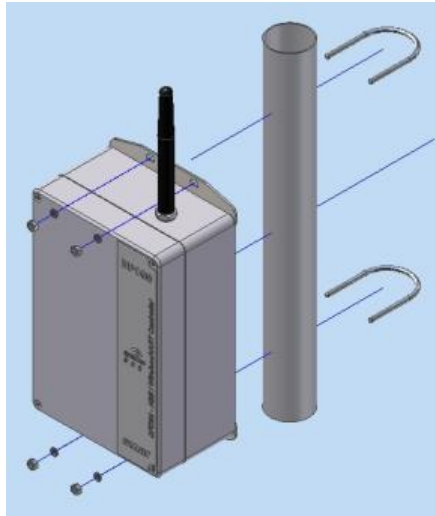


Figure 2.68 – Pipe mounting

It is recommended to install the DF100 in a central and strategic plant place, from which it both has a good line of sight of a minimum set of *WirelessHART* devices, and proximity of the control room where the host is. Once configured, the DF100, understood here as the *WirelessHART* Gateway, should be the first device to be energized (powered) in a planned *WirelessHART* network. In other words, the DF100 must always be energized before any other *WirelessHART* device (field device or adapter).

ATTENTION

The DF100 installation can only be done by qualified electricians.

Grounding

Use the grounding plate at the bottom, left, of the DF100 IP66 housing (see next figure) to properly ground it. Additional information on grounding can be found in the SYSTEM302 - Electrical Installation Guide.

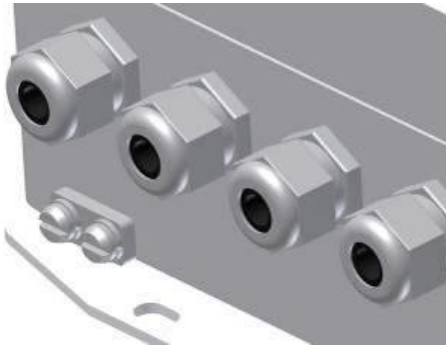


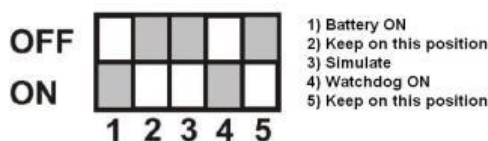
Figure 2.69 – Grounding plate

ATTENTION

The DF100 grounding can only be done by qualified electricians.

Important:

The DF100 has an internal battery that keeps the Real Time Clock (RTC) and its non volatile RAM (NVRAM) when there is lack of external supply. This battery can be either enabled or disabled, depending on the position of the switch 1, in the internal part of the DF100. To enable the battery, set the switch 1 to ON position as shown in the following figure:



In this configuration, when there is lack of energy, the RTC and the NVRAM will be supplied by the battery, allowing the retention of all configuration data. In case of equipment storage, it is recommended that the battery is turned off (switch 1 in OFF position). For further information about the battery, refer to the "Technical specification for controllers" section.

So, before installing the DF100 module, be sure the switch 1, which refers to the battery, is in the enabled position. If the battery is not enabled during the start-up, the DF100 controller starts in HOLD mode.

The Watchdog is a mechanism to detect if an important or high priority task stops in the controller. So, be sure the switch 4, which refers to the Watchdog, is in the ON position.

Knowing the topics of the above sections, a basic system using the DF100 controller can be composed of:

- Power Supply (24Vdc, 1A);
- DF100 – HSE/WirelessHART Controller;

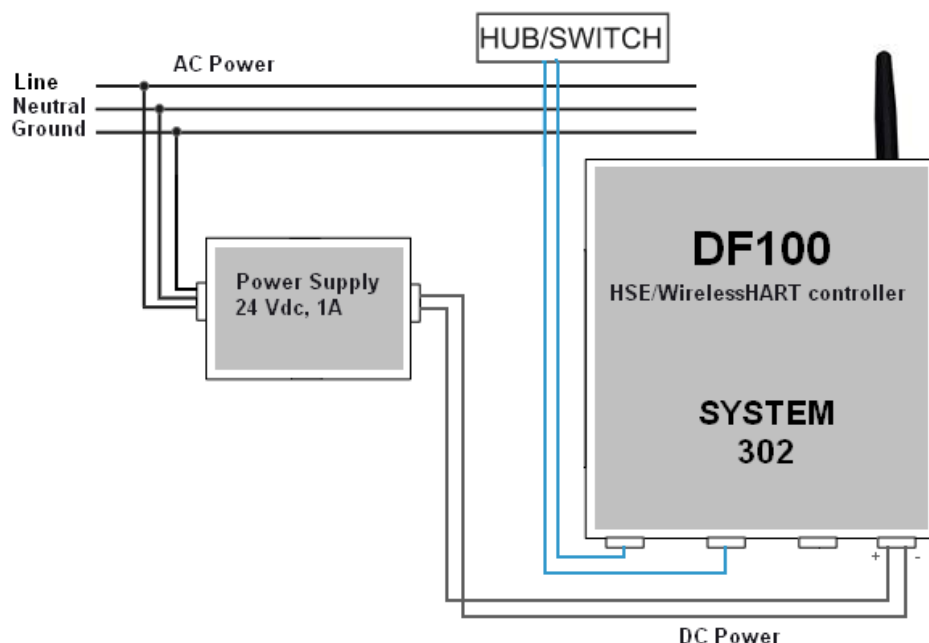


Figure 2. 70 – Basic system using the DF100 controller

The DF100 can be used outdoors, because it has IP66 rating. The wireless technology significantly reduces costs related to cabling, trays, and technical/engineering hours spent on maintaining the network.

The basic project topology is shown in the following figure. It has two control network segments interconnected to corporate networks. On the control network is added the DF100 controller. Finally, supervision and control stations complete the interconnection system.

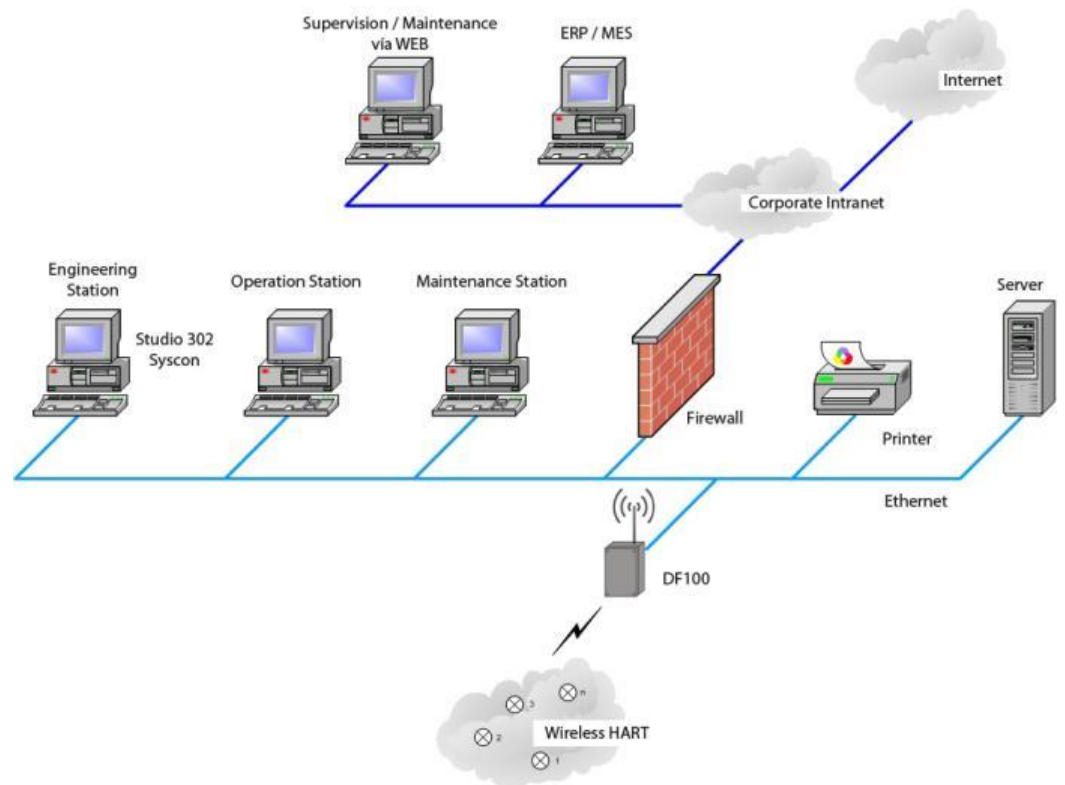


Figure 2. 71 – Wireless architecture for control network

Tools such as **Syscon** and **Studio302** are related to the architecture to integrate the controller and the wireless network to others industrial automation protocols. For further details refer to the Creating a Foundation fieldbus strategy using the DF100.

Dimensional Drawings of DF1A Rack and Modules

Dimensions in mm

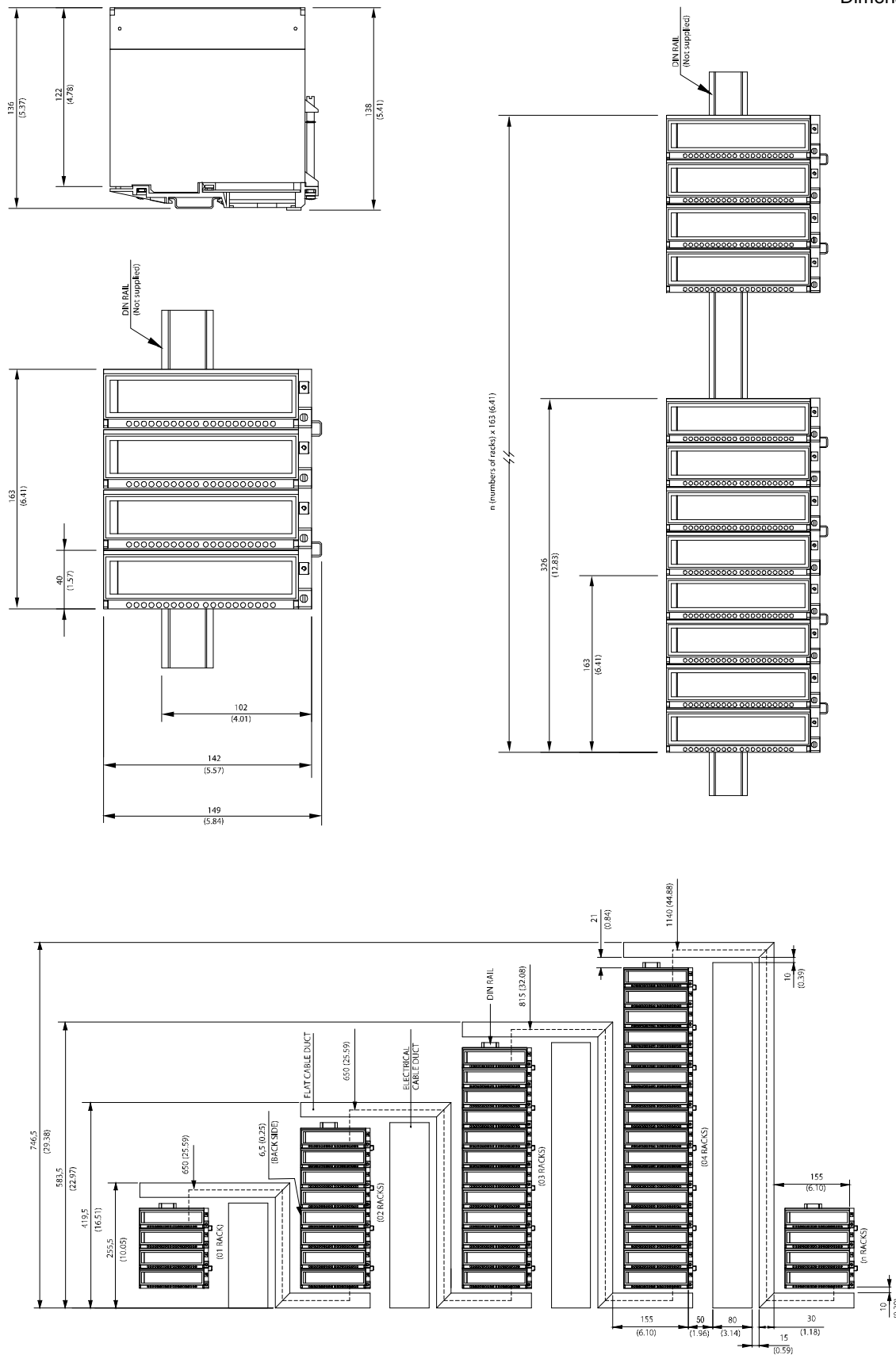
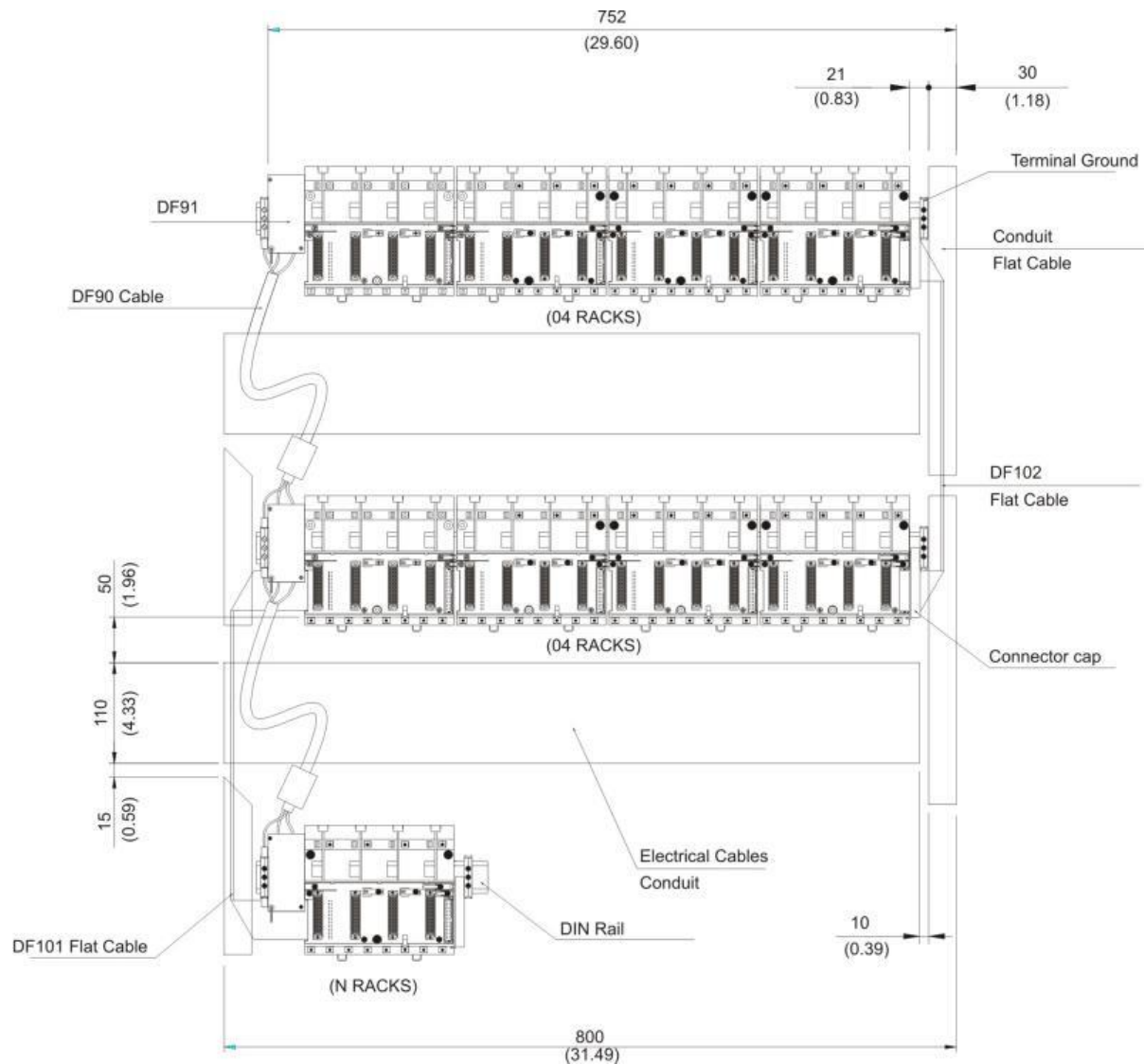
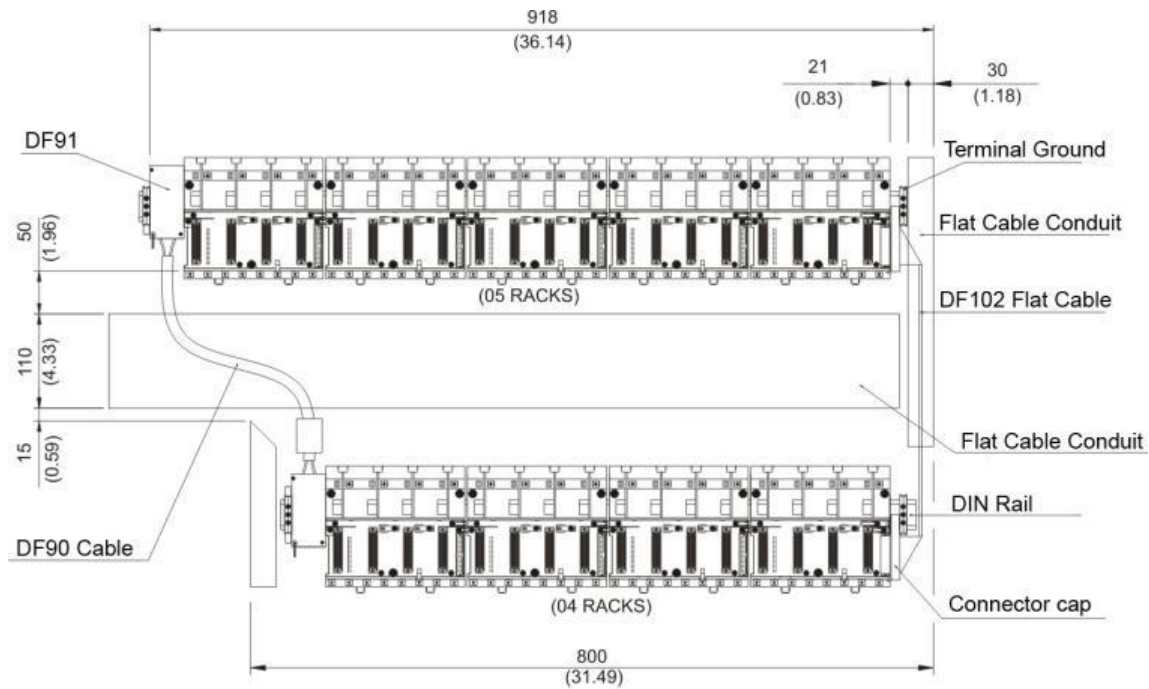


Figure 2.72 – Dimensional Drawings

Dimensional Drawings of DF93 and Modules

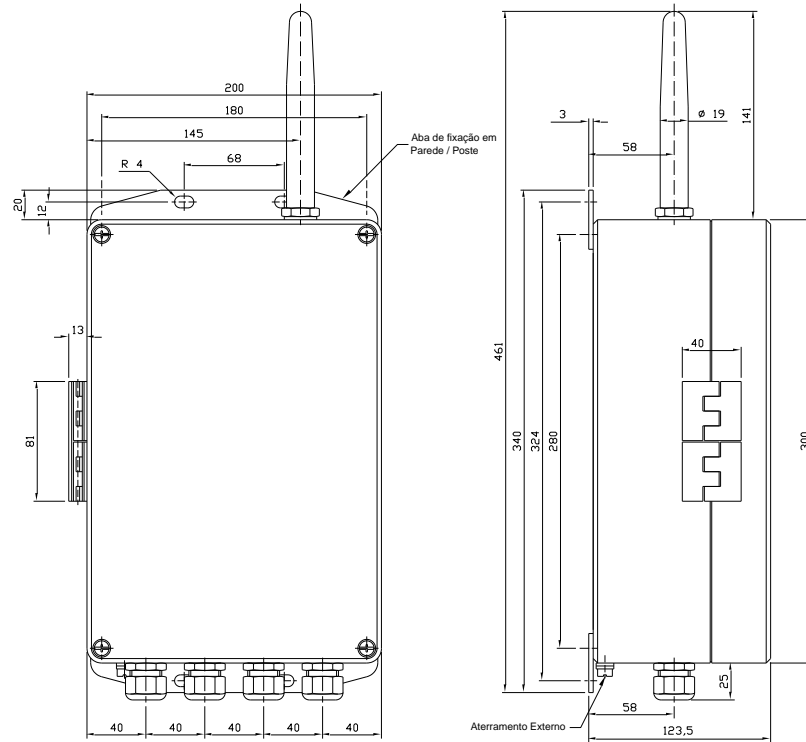
The following figures shows two possible combinations (dimensions in mm).





Dimensional drawing of DF100

Dimensions in mm



Installing the Studio302

Install the programs that compose the **SYSTEM302** using the installation DVD. For further details about installing the programs, refer to the **SYSTEM302** Installation Guide. The **Studio302** is the user-friendly, easy-to-use software tool that integrates all applications included in Smar's Enterprise Automation package.



Figure 2.73 – Installing Studio302

Getting License for DFI302 Servers

There are two ways to get the **DFI OLEServer** and **HSE OLEServer** license. One version is through Hard Lock protection (Hard Keys) and other through Software (Soft Key).

When using Hard Key, just connect it in the appropriated port in the computer (parallel or USB ports).

When using Soft Key, it is necessary to get a License Key through a SMAR contact. For this, use the application **LicenseView**, in the **Studio302** interface.

Click **Softkey**, and then, **Request License**.

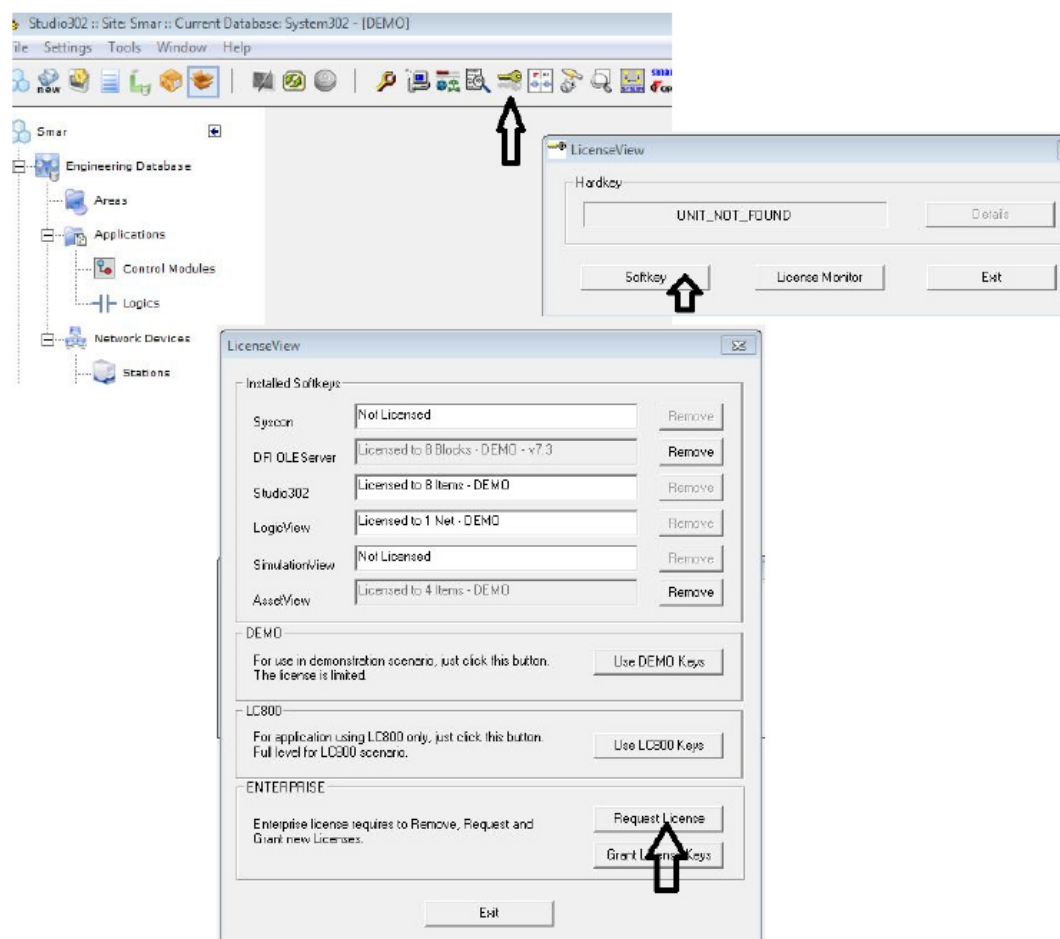


Figure 2.74 – Getting Licenses

Fill in the fields correctly, enter the **LN** (License Number) code that was received with the purchase of the product. See this code on the keyring that came with the installation media. A file (.zip) will be generated in the **\SmarOLEServers** folder for the user to send to Smar. Use the email (techsupport@smar.com.br) or access the Technical Support channel through the website (www.smar.com).

After sending the email to Smar, a key will be generated according to the corresponding **LN** level. This key will be returned via the email informed in the previous form. This generation is not automatic and, therefore, you must wait for business hours.

After receiving the email from Smar, launch the **LicenseView** application again, enter the key obtained in the **DFI OLE Server** field and click **Grant License Keys**. If everything is correct, a message will notify you that the license has been accepted.

NOTE

This license is valid for **DFI OLEServer** and **HSE OLEServer**.

At this moment, **Syscon**, **DFI OLEServer** and **HSE OLEServer** will be ready to be used.

Section 3

SETTING UP

DFI OLE Server and HSE OLE Server Settings

After success in Getting License procedures, the user should set some parameters related to the DFI OLE Server and HSE OLEServer. For further details about SMAR OLE Servers, refer to the specific section.

Connecting the DFI302 in the Subnet

DFI302 working environment is composed by a network (subnet) where IP addresses will be necessary for each connected instrument.

The automatic solution for attribution of these addresses is called DHCP (Dynamic Host Configuration Protocol) Server.

Using DHCP Server these IP addresses are generated automatically preventing any IP conflict between two distinct devices.

ATTENTION

To connect more than one **DFI302**, the following steps must be fully executed for each **DFI302**.

- 1- Plug the Ethernet cable (DF54) of the controller module to its respective subnet switch (or hub).

NOTE

For point-to-point connection (the module **DFI302** connected directly to the computer), uses the DF55 cross cable.

- 2- Turn on the controller module. Ensure that ETH10 and RUN indication LEDs are on.
- 3- Keep tight pressed the left push-button (Factory Init / Reset) and press the right push-button for three times. The FORCE LED will blink three times consecutively.

NOTE

If the user loses the number of times that the right push-button was pressed, just see the number of times that the FORCE LED is blinking at each second. It will turn to blink once after the fourth touch (this function is cyclic).

- 4- Release the left push-button and the system will execute the RESET, and subsequently will start the firmware with standard values for IP address and the subnet mask.

For Network WITH DHCP

- 5- If the network has a DHCP server (consult the network administrator), the controller is already connected to the subnet. Stop the steps here.

For Network WITHOUT DHCP

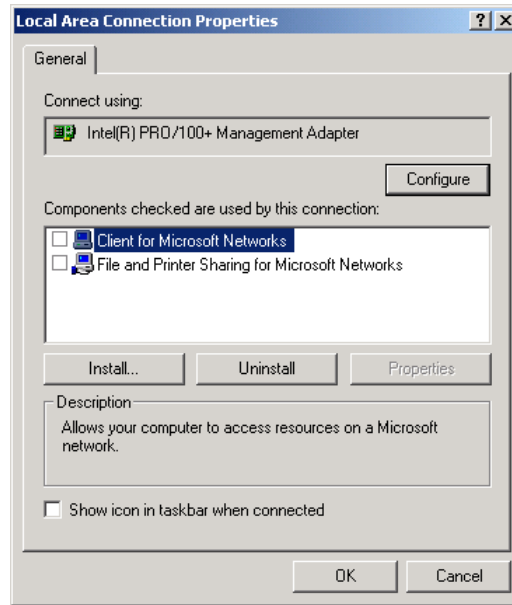
- 6- If the network does not have a DHCP server, the controller will have the default IP address 192.168.164.100 and it will need to follow the next steps.

The IP address of the user's computer needs to change for a while (network management knowledge is required). The following procedures are based on Windows 2000. Choose **Start → Settings → Control Panel**. Double-click **Network and Dial-Up Connections** option or similar.

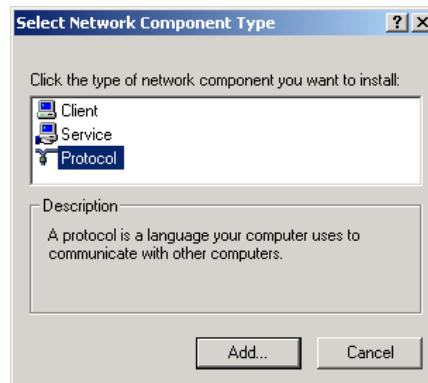
NOTE

Right-click **Local Area Connection**, choose **Properties** from the pop-up menu. Whether in the component list has TCP/IP protocol, the user should skip to step 9 or proceed the installation using the **Install** button.

7- Click **Install**.

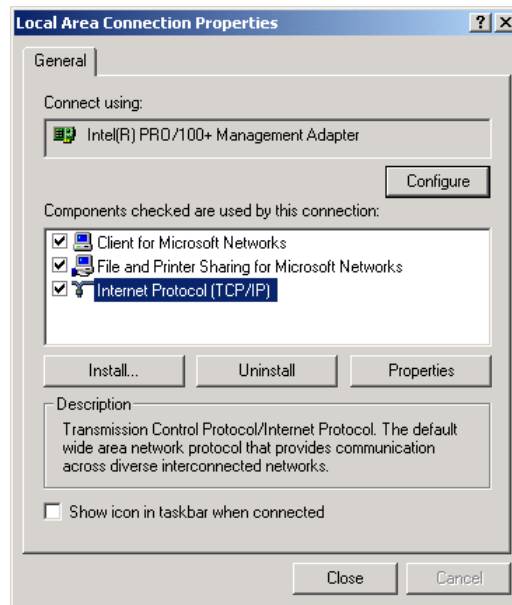


8- Select the network type **Protocol** and click **Add**.



9- Select **Internet Protocol** and click **Ok**.

10- Select **Internet Protocol (TCP/IP)** and click **Properties**.



- 11- Take a note of the original values of IP address and Subnet Mask of the computer to restore them when the operation ends.

NOTE

If the IP address is already something like: 192.168.164.XXX, skip to Step 14.

- 12- Change IP address and the Subnet Mask of the computer. It must select the same Subnet of DFI302 (164) and an IP address different of the DFI302 (100). The Network Administrator must provide the IP address.

NOTE

The values will be something like: IP Address 192.168.164.XXX and network mask (Subnet Mask) 255.255.255.0. Keep the default gateway value.

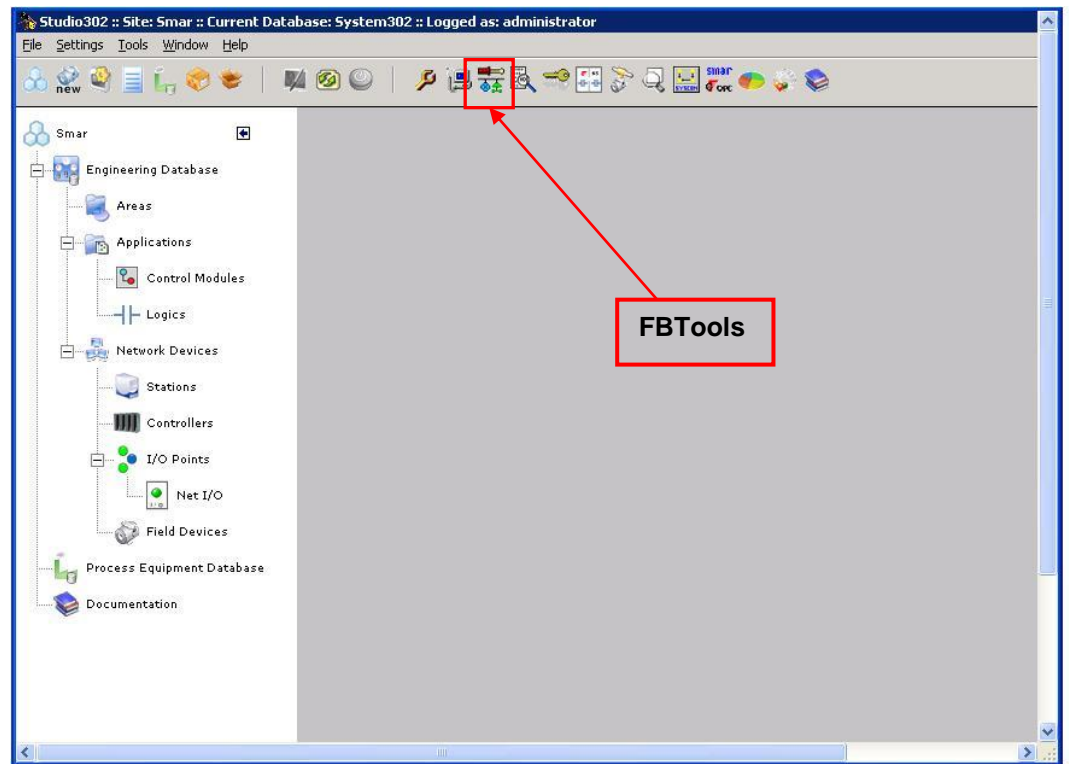
ATTENTION

Do not use the IP Address 192.168.164.100. This is already DFI302 default address.

- 13- Click **OK**.

- 14- Run the **FBTools** through **Studio302**. Click **Start menu** → **Programs** → **System302** →

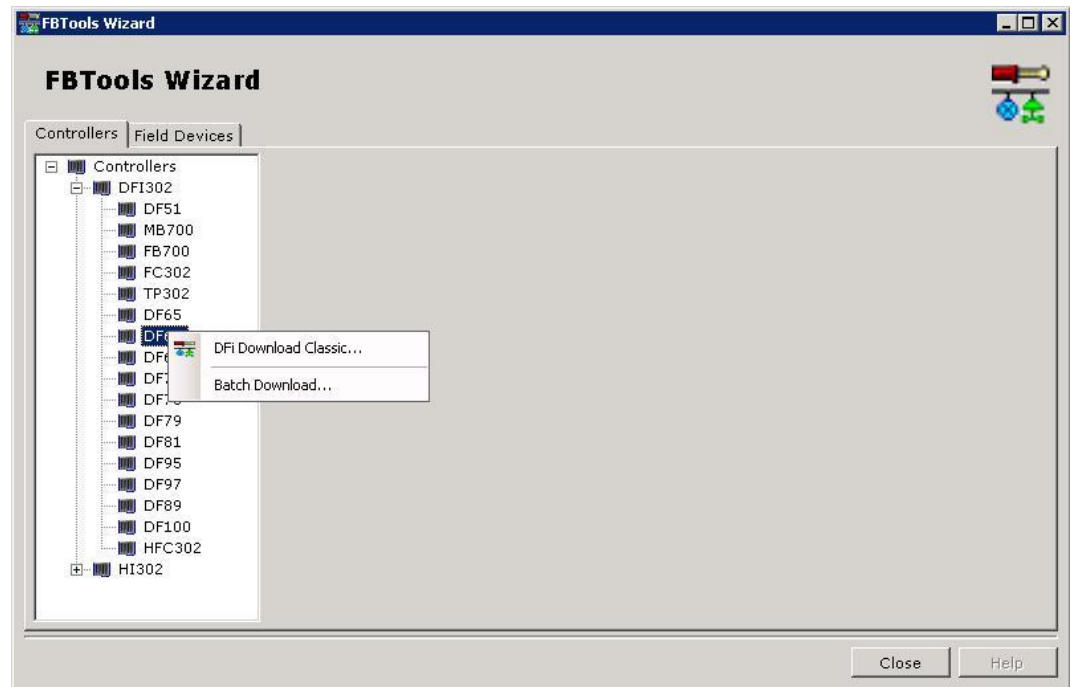
Studio302. Do a login in the system. In the **Studio302** interface click the  icon in the main toolbar. See the next figure.



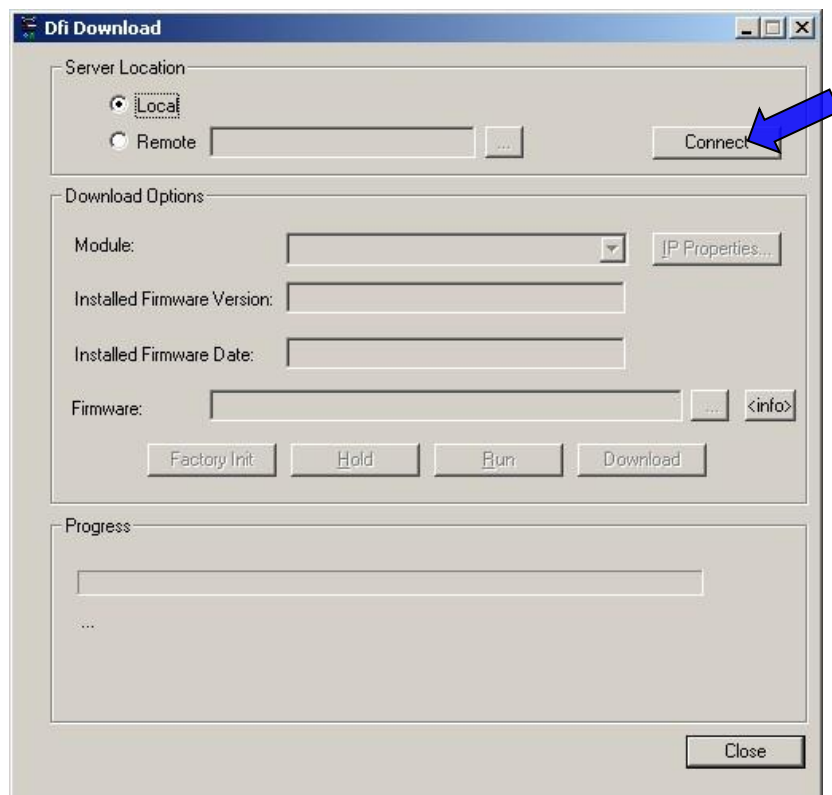
- 15- The following window will open. In the **Controllers** tab click the symbol and the **DFI302** and **HI302** options will appear. Click again the symbol in **DFI302** and select the controller module.



- 16- Right-clicking the controller the **Dfi Download Classic** and **Batch Download** options will appear. See the following figure.



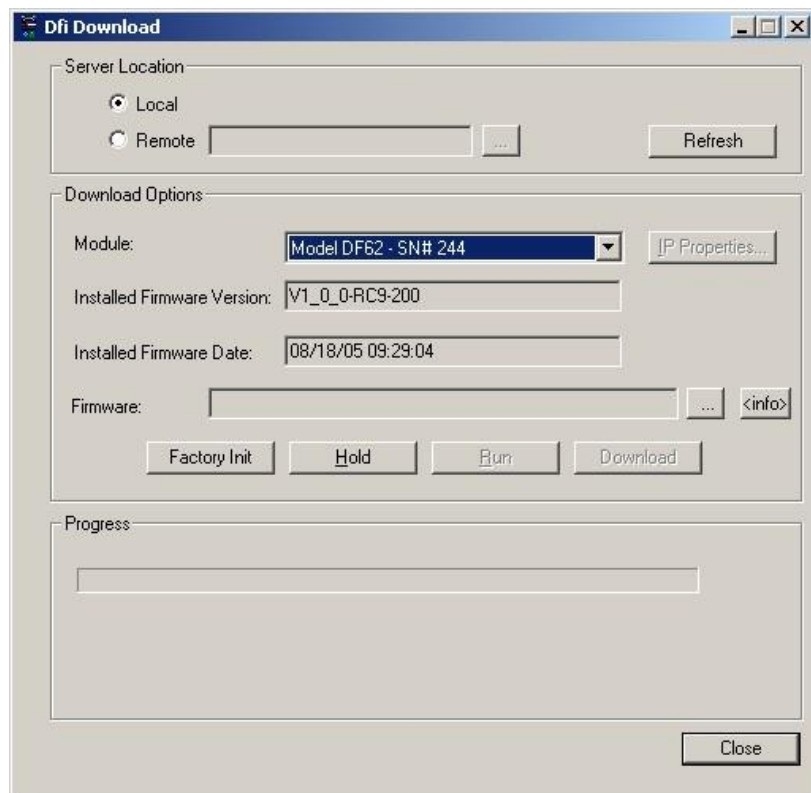
- 17- Select **Dfi Download Classic** and the following figure will appear. Select the DFI OLEServer path to be used (Local is the default path) and click **Connect**.



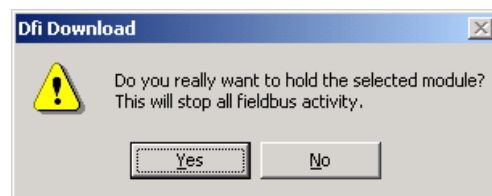
- 18- Select the controller in the **Module** box. Use the serial number as a reference that is in the external identification label.

ATTENTION

The non-observance of this step may imply in serious consequence.



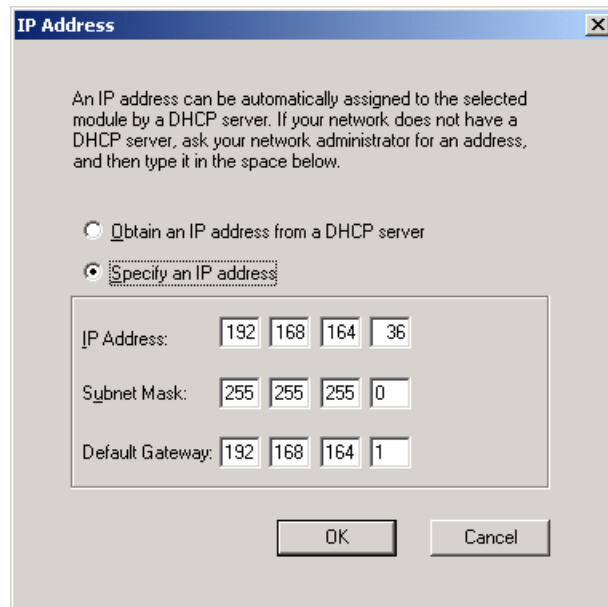
- 19- Click **Hold** to interrupt the firmware execution. When the user clicks the **Hold** button, the module will stop the firmware execution as well as all the activities in the fieldbus line. Confirm the operation by clicking **Yes**.



ATTENTION

This step will be necessary only if the **Hold** button is enabled; pointing out that the firmware is being fulfilled.

- 20- Check if the HOLD LED is ON. Click **IP Properties** to configure the IP address of the module. The IP Address dialog box will open.
- 21- The default option is to obtain the IP address from DHCP Server. Select the option **Specify an IP address** to change to another IP address.



- 22- Type the IP address, the subnet mask and the default gateway. The subnet mask should be the same of the user's computer original default address (Step 11). So, the computer settings can be restored later, and the network will show **DFI302** modules.

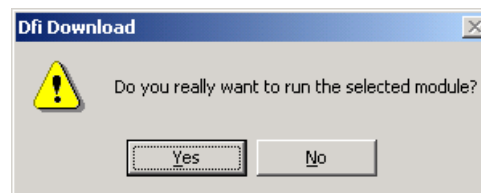
ATTENTION

Do not use the IP Address 192.168.164.100 (it is already being used by **DFI302**)

HINT

Write down the IP addresses that will be specified and the serial number of each controller module. It will help in the identification and diagnostics of possible failures.

- 23- Click **OK** to end up this operation. Go back to the **Internet Protocols (TCP/IP)** properties of the computer and restore the original values of the IP address and the subnet mask.
- 24- Click **Run** to execute the **DFI302** firmware again.
- 25- A dialog box will open to confirm the operation. Click **Yes** to continue.



- 26- The procedure to connect the **DFI302** to the subnet is complete. Repeat these steps above for the other modules.

NOTE

In case of there is more than one **DFI302** to be set up, perform the following command to clear **ARP table**, before setting up the next **DFI302**.

```
C:\>arp -d 192.168.164.100 < enter >
```

- 27- In the DOS prompt, type "C:\>arp -d 192.168.164.100 <enter>".

Updating the Firmware

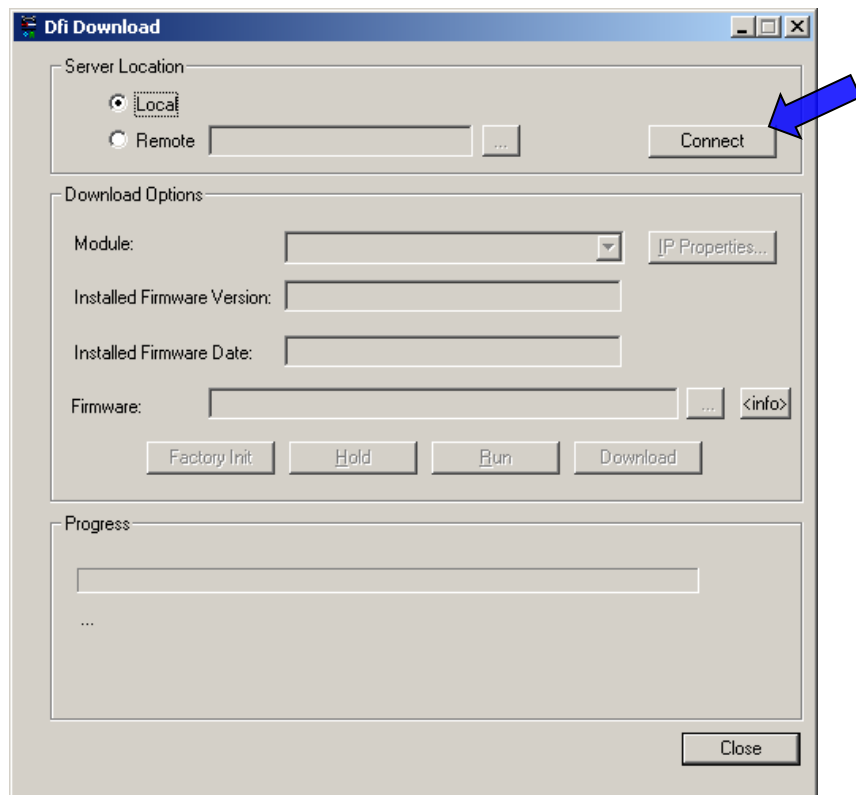
1. Make sure that the **DFI302** is ON and has been connected to the subnet, according to the procedures in "Connecting the DFI302 in the Subnet".
2. Run the **FBTools** as described in the step 14 of the previous topic.
3. Select the controller module and right-click it. Choose an option - **Dfi download Classic** or **Batch Download**.

The **Dfi Download Classic** option allows updating the firmware, changing the IPs of controllers and other devices.

The **Batch Download** option allows updating the firmware of up to 64 controllers simultaneously.

DFi Download Classic

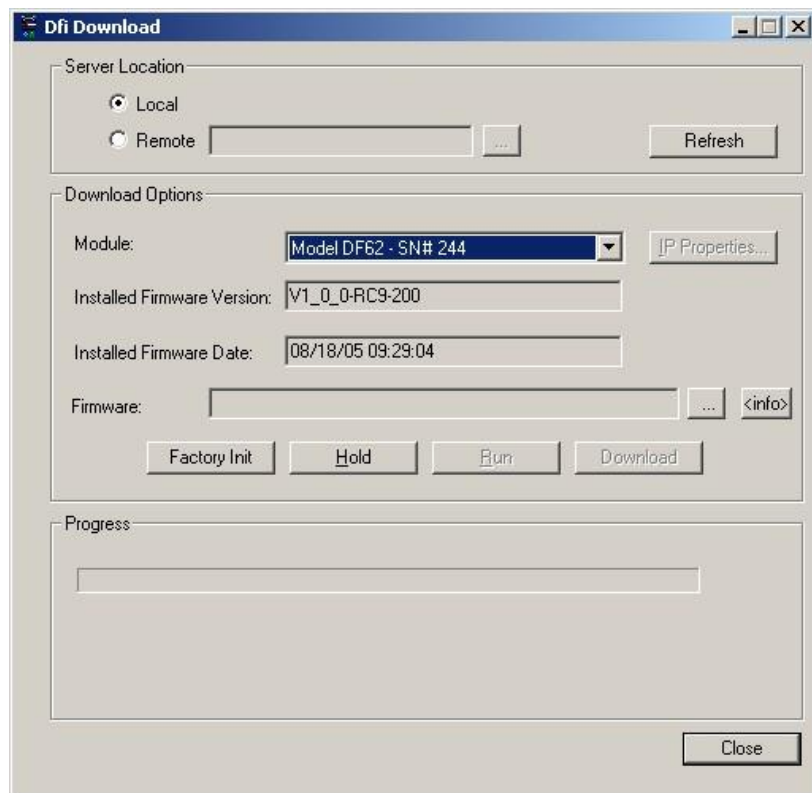
1. By selecting the **Dfi Download Classic** the **Dfi Download** window will open. Select the DFI OLEServer path to be used (Local is the default path), and click **Connect**.



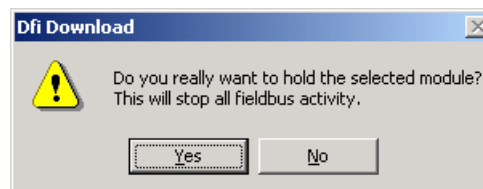
2. Select the controller module in the **Module** box. Use the serial number as a reference (see the external identification label).

ATTENTION

The non-observance of this step can imply in serious damages.




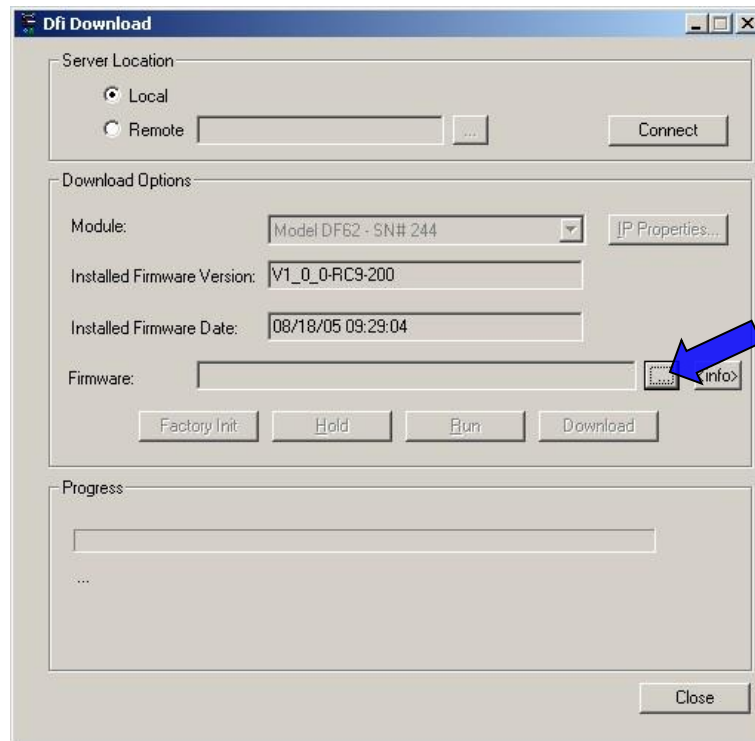
3. Click **Hold** to interrupt the firmware execution in the controller.
4. Afterwards all activities in the fieldbus network will be stopped. Confirm this operation by clicking **Yes**.



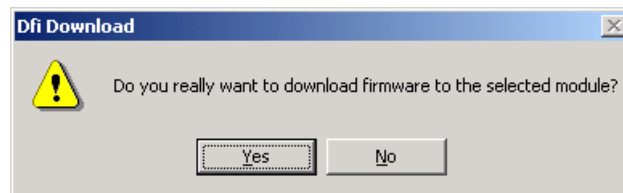
ATTENTION

For the steps below, it is necessary to have the **Hold** button enabled, indicating that the firmware is being executed.

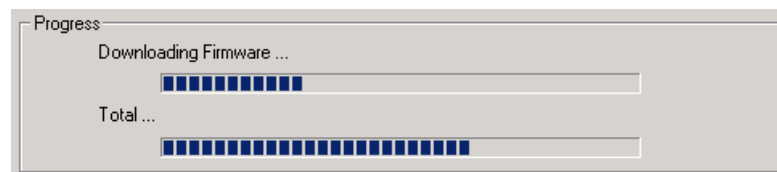
5. Check if the HOLD LED is ON.
6. Note that the **DFI Download** window shows the installed version and date of the current firmware loaded in the controller module.
7. Click the  button to select the firmware file to be downloaded (controller*.abs file).



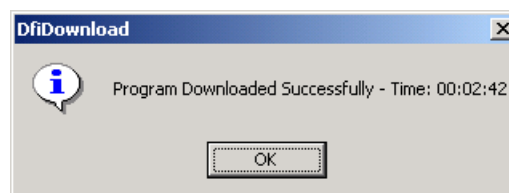
8. After selecting the firmware file, click **Connect**. The **Download** button will be enabled. Click it to start the firmware download.
9. A message box will come up requesting a confirmation. Click **Yes** to continue.



10. The progress bar at the bottom of the window will show the operation progress.



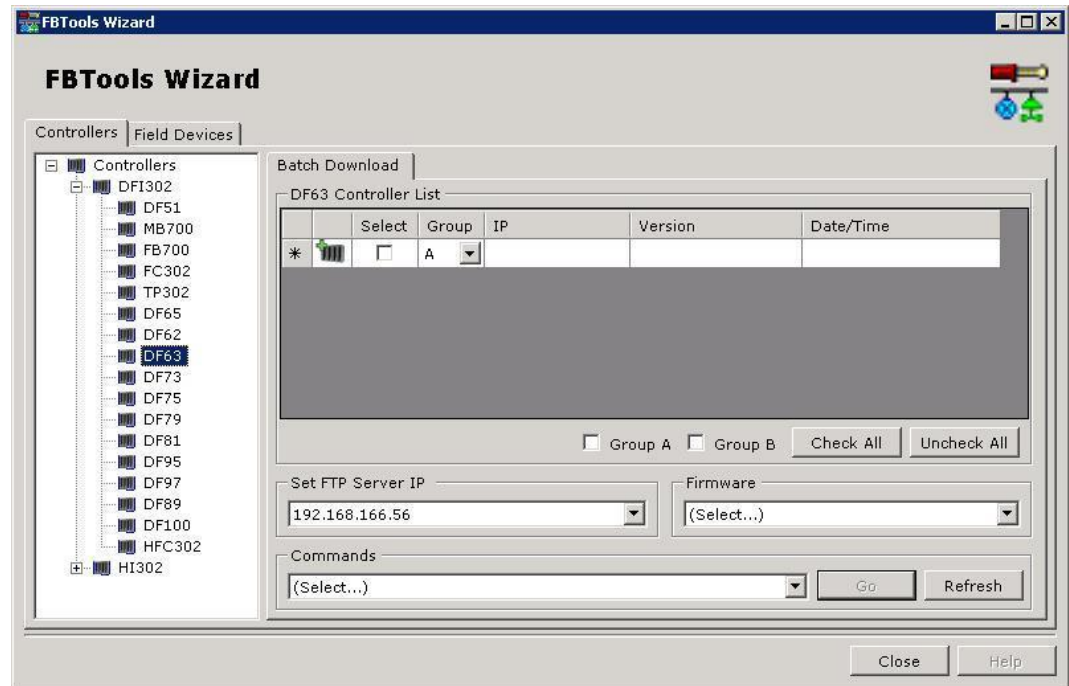
11. When the download is complete, a dialog box will appear confirming that the program was downloaded successfully. Click **OK** and wait a few minutes while the information is updated. The **DFI302** will be in "Run Mode". (Check if the RUN LED is ON.)



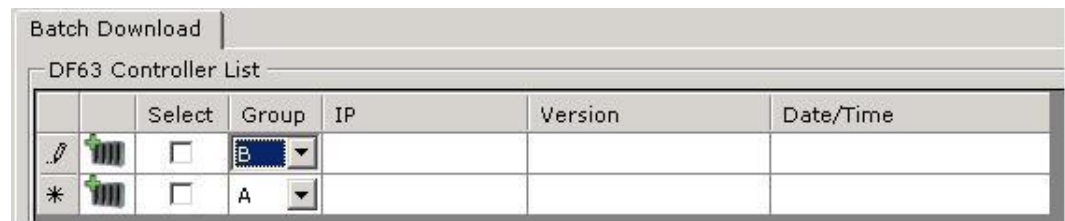
12. Click **Close** to close the **Dfi Download** window.

Batch Download

By selecting the **Batch Download** option, the following window will appear:



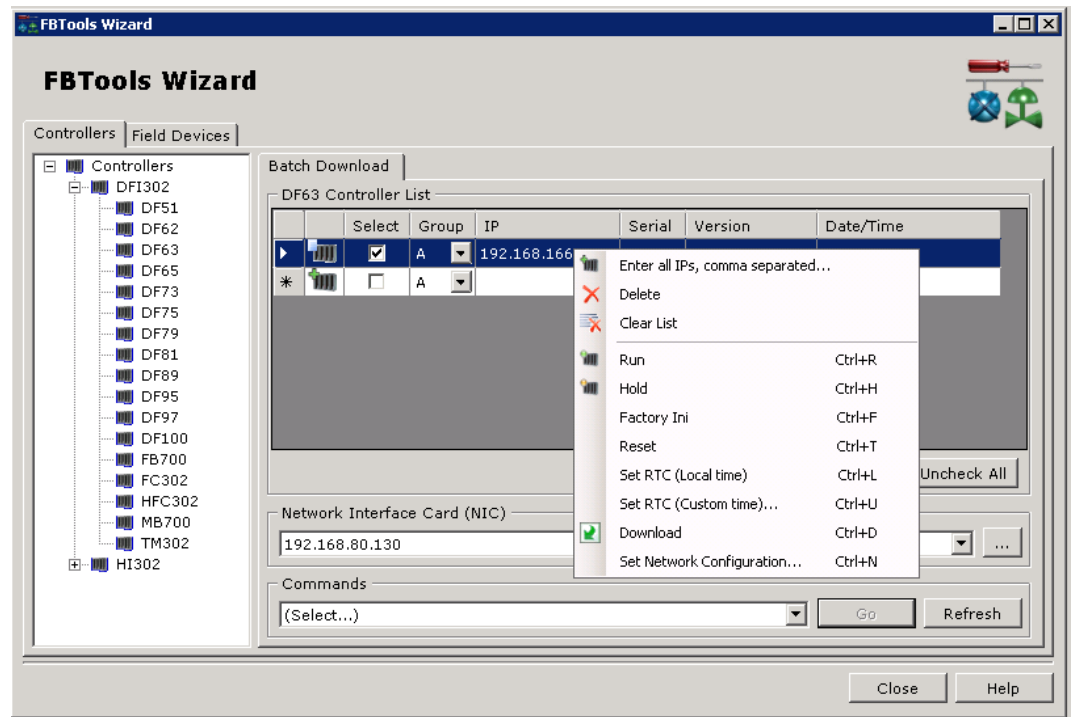
The controllers can be divided in two groups – A and B. The groups are used to classify the controllers. Typically, when redundancy is used, there is the option to change the firmware of all secondary controllers first, and then the primary ones. This procedure facilitates the hot swap maintenance of the plant without requiring stops. For this, the group A is used to classify all primary controllers, and group B the secondary controllers. See the following figure.



The symbols of the previous figure have the following meanings:

	Editing mode of a controllers list field
	Empty list item
	New controller can be inserted in this line
	Controller already registered on the list

Right-clicking the controllers list the following options will appear:



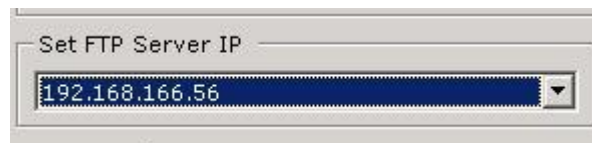
Through the **Enter all IPs, comma separated...** option the user can insert various IPs on the list simultaneously, separated by commas. After entering the IPs click **Insert** and make the association of groups A and B.

The **Delete** option deletes the selected IP and **Clear List** option clears the IPs list.

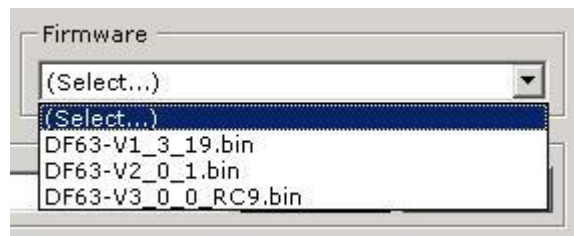
To select, or deselect, all controllers of groups A and B use the **Check All** or **Uncheck All** options, respectively. See the following figure:



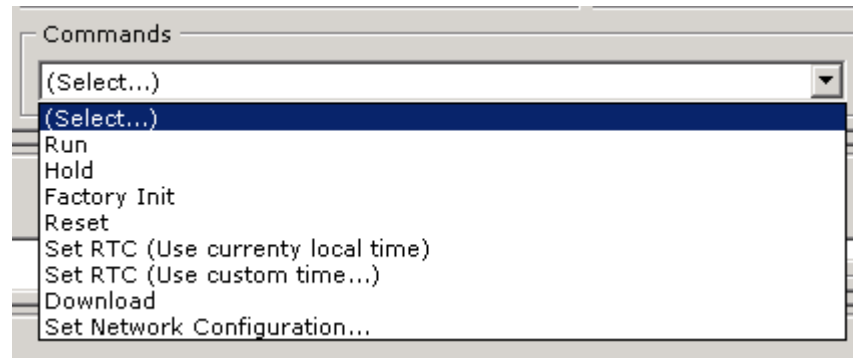
Up to 64 controllers can be updated simultaneously. The firmware file should have the **.bin** extension to be used by **Batch Download**. In the **FTP Server IP** field choose one of the options presented, because this IP chosen will be used by the controller to take the existent **.bin** file. For DF63 controller with serial number less than 7000, the firmware is distributed in two files **.bin** and **.txt**. If the serial number is greater than or equal to 7000, the firmware is distributed in two files with extensions **.7k** and **.txt**.



The available versions are in the **Firmware** field.

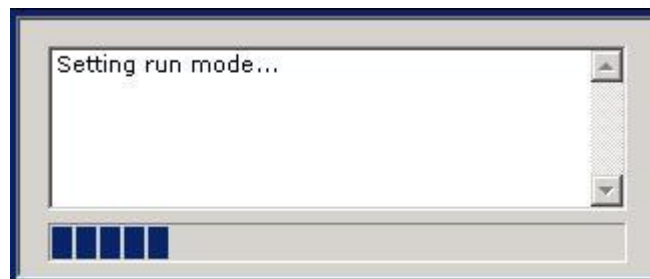


In the **Commands** field are the action options of the **Batch Download**. Select the controller, the command to be performed and click **Go**.

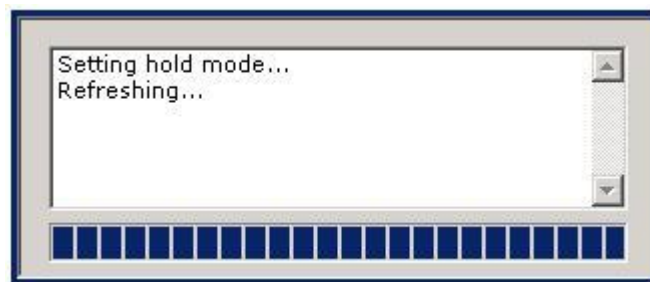


Here are the definitions of the above options:

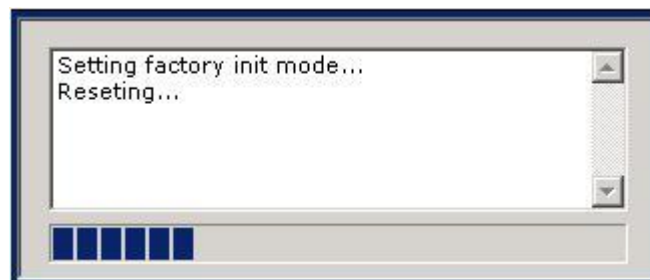
Run – Starts the firmware execution in the controller module. The following window will appear.



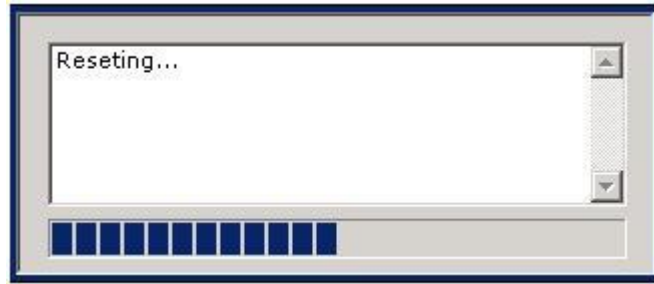
Hold - Interrupts the firmware execution in the controller. The following window will appear.



Factory Init – Erases the configurations of strategies and logics. It returns the controller to the same state it left the factory. The following window will appear:



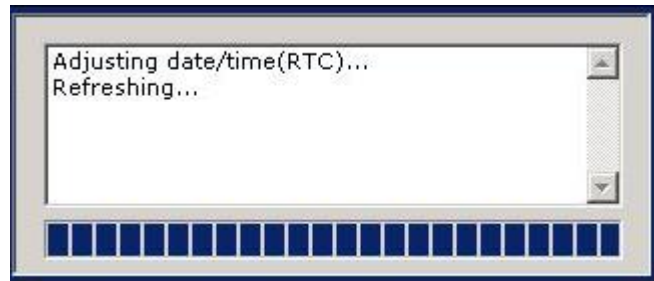
Reset – Restarts the controller, maintaining the configurations that were saved at the last download. Some dynamic parameters are erased, but not the static parameters. This is dependent on each function block. Refer to Function Blocks manual for further information. The following window will appear:



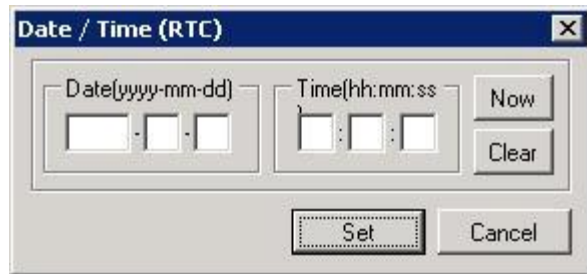
NOTE

In both cases, **Reset** and **Factory Init**, the firmware is kept. The controller's IP may be change only if is set the option to obtain it via DHCP Server. Otherwise the controller will keep the last IP assigned.

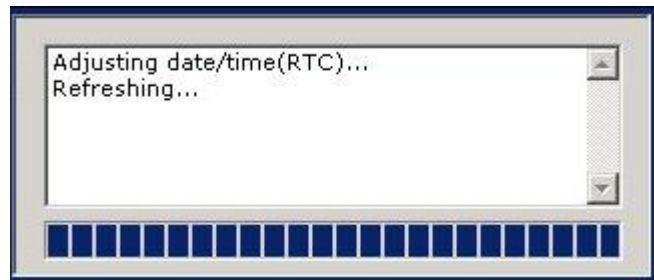
SetRTC (use currently local time) – Sends the *Localtime* to the controller. The following window will appear:



SetRTC (use custom time) – Sends the user-configured time to the controller in a properly window. See the following figure. Write the date and hour desired, and click **Set**. If you want to insert the current date and hour click **Now**. The **Clear** option clears the fields filled.



After filling the fields and click **Set** the following window will appear:

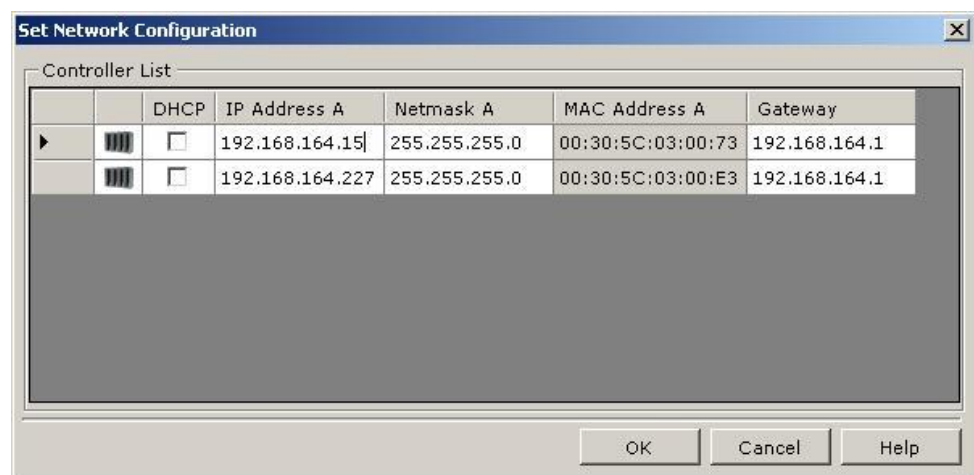


Download – Performs the firmware download. The following window will appear:

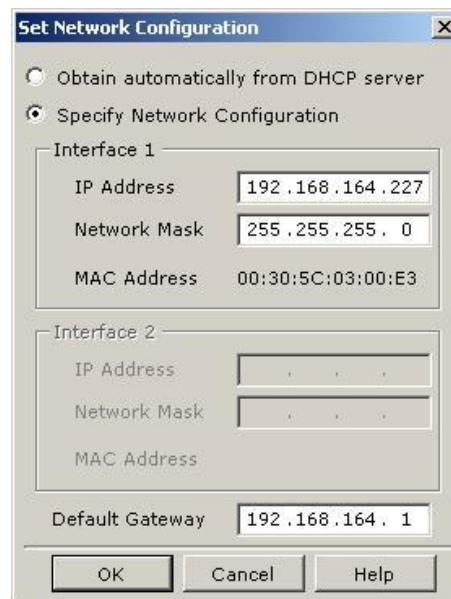


A progress bar will signal that the download is in progress. After this, confirm in the table that the information from the controller is corresponding to actions performed, for example, the firmware version.

Set Network configuration – This option allows controllers IP are changed in batch. The following figure will open:



If only one controller is selected the following figure will open:



For details about changing the controller's IP see the next topic.

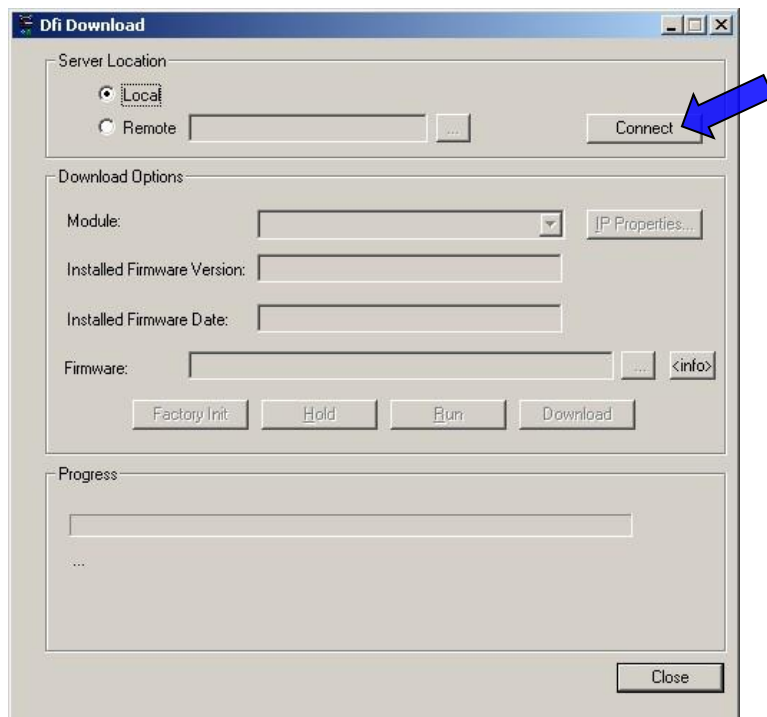
Changing IP address

Changing IP Controller

NOTE

To change the **DFI302** subnet, see the procedures in "Connecting the DFI302 in the Subnet". Follow these steps only to change the IP address.

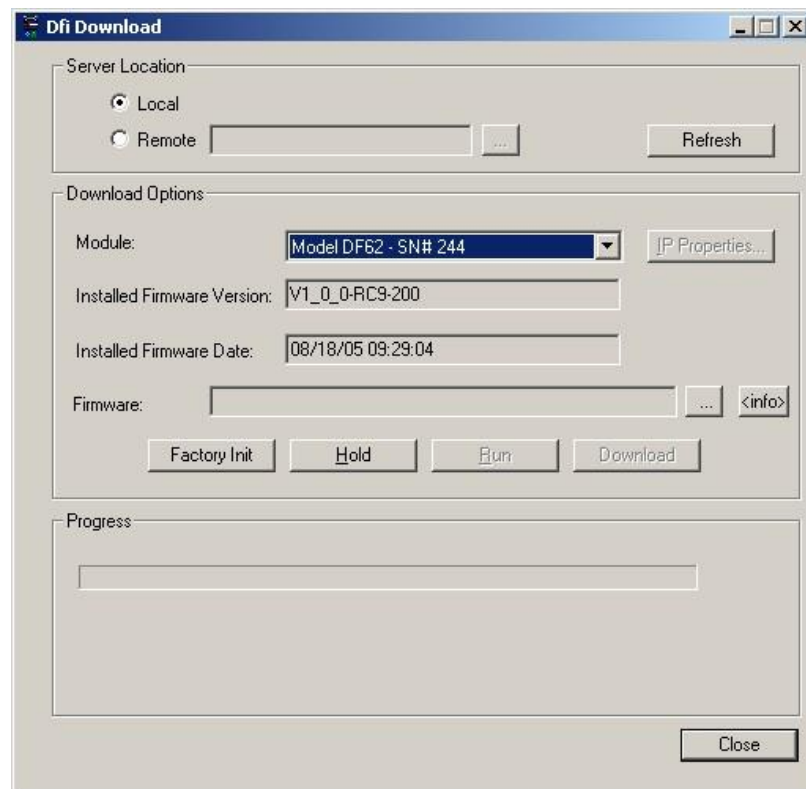
1. Make sure that the **DFI302** is ON and has been connected to the subnet, according to the procedures in "Connecting the DFI302 in the Subnet".
2. Run the **FBTools** as described in the previous topics.
3. Select the controller module, and click **Dfi Download Classic**. For the **Batch Download** option refer from step 4'.
4. The **DFi Download** dialog box will be open. Select the DFI OLEServer path to be used (Local is the default path) and click **Connect**.



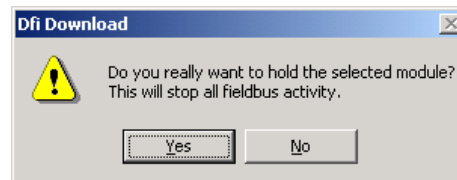
5. Select the controller module in the **Module** box. Use the serial number as a reference (see the external identification label).

ATTENTION

The non-observance of this step can imply in serious damages.



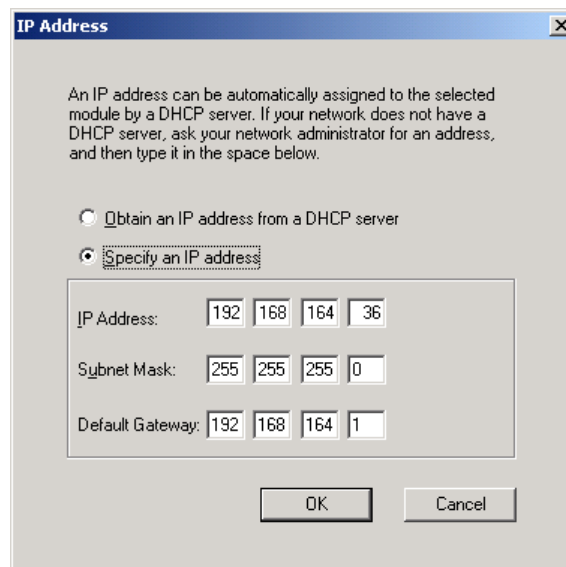
6. Click **Hold** to interrupt the firmware execution in the controller.
7. Afterwards all activities in the Fieldbus network will be stopped. Confirm this operation by clicking **Yes**.



ATTENTION

This step will be necessary only if the **Hold** button is enabled, indicating that the firmware is being fulfilled.

8. Check if the HOLD LED is ON.
9. Click **IP Properties** button at the **DFI Download** window. The **IP Address** window will open.
10. The default option is **Obtain the IP Address from a DHCP Server**. Click the **Specify an IP address** option to change to another IP address.



11. Type the IP address, the subnet mask and the default gateway (provided by the network administrator) to be associated to the **DFI302**.

ATTENTION

Do not use the IP Address 192.168.164.100 (it is already being used by **DFI302**). In addition, be sure that the chosen address is not in use.

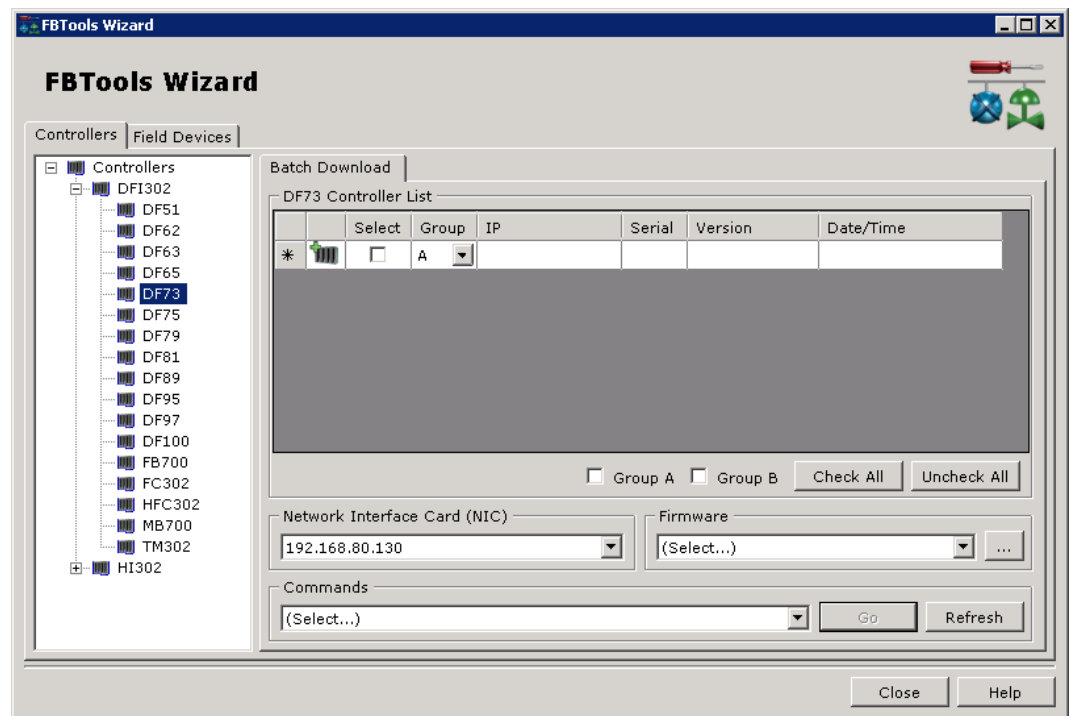
HINT

Write down the IP addresses that will be specified and the serial number of each controller module. It will help in the identification and diagnostics of possible failures.

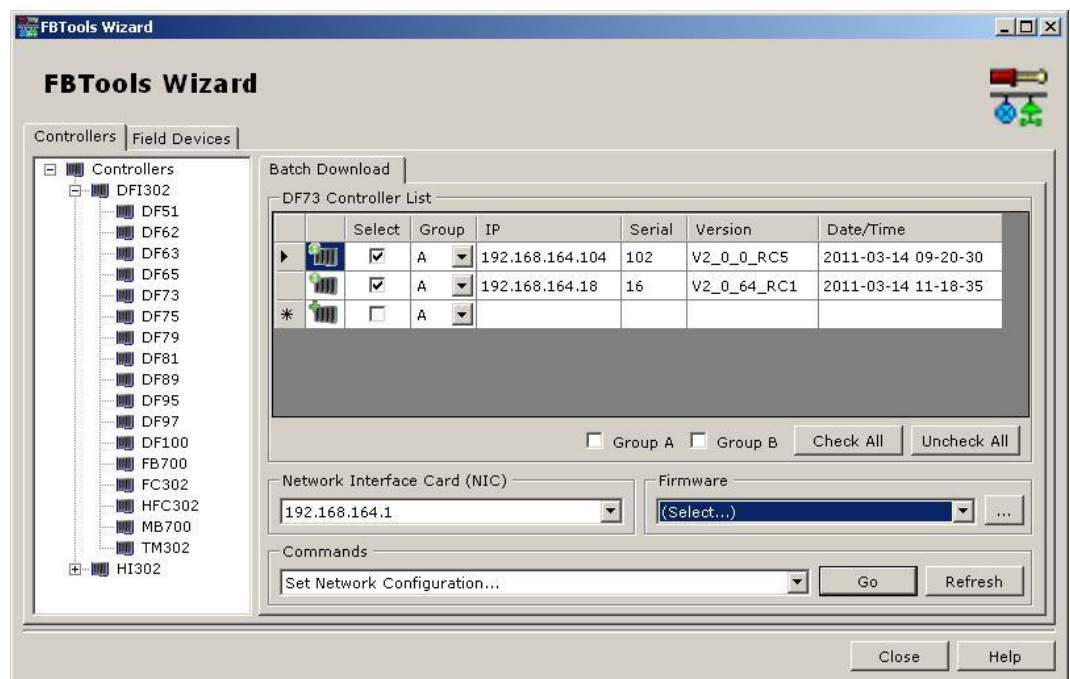
12. Click **OK** to conclude this operation.
13. After assigning a new IP address, the process will return to the **Dfi Download** window.
14. Click **Run** to execute the **DFI302** firmware again.
15. Click **Close** to close the **Dfi Download** window.

Batch Download

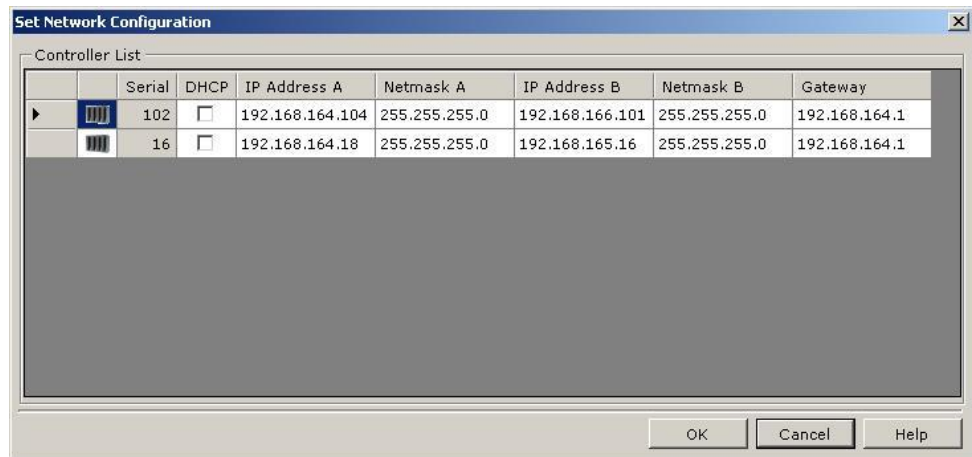
4'. The **Batch Download** window will open. With this interface multiple IPs can be changed simultaneously.



5'. Select the controller whose IP will be changed. See an example in the following window:



6'. In the **Commands** field select **Set Network Configuration** and click **Go**. The following window will open.



7'. Type the IP address, the subnet mask and the default gateway (provided by the network administrator) to be associated to the **DFI302**. Click **Ok** and the changes will be verified. For further information about error messages refer to the **FBTools** help.

ATTENTION

Do not use the IP Address 192.168.164.100 (it is already being used by **DFI302**). In addition, be sure that the chosen address is not in use.

HINT

Write down the IP addresses that will be specified and the serial number of each controller module. It will help in the identification and diagnostics of possible failures.

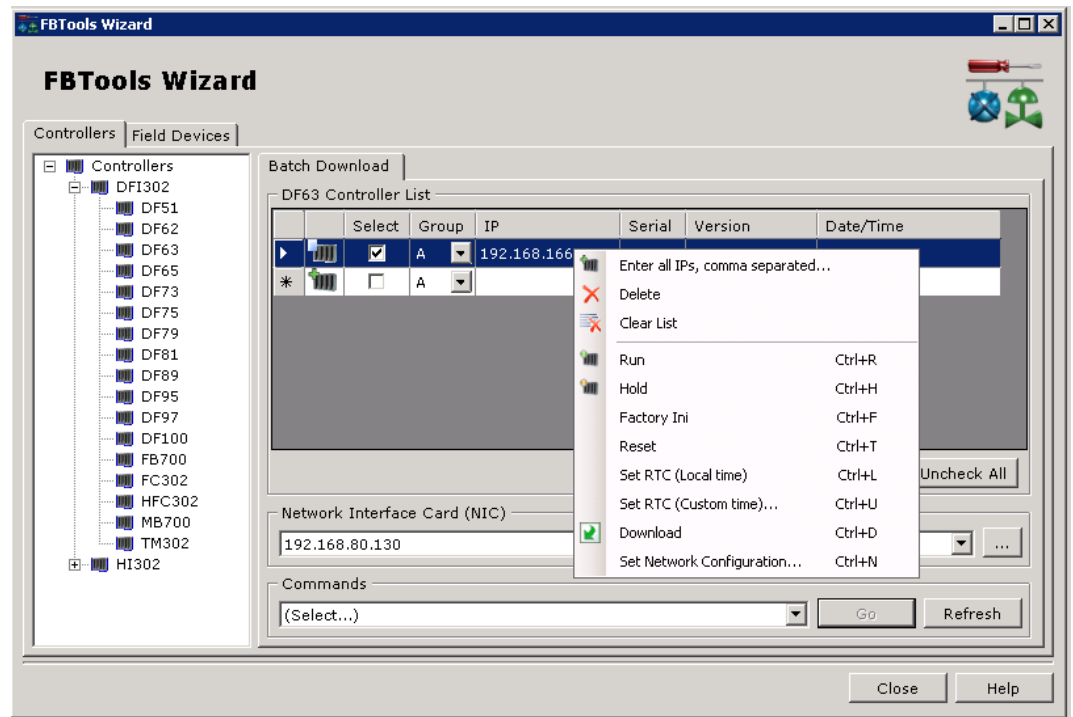
8'. After assigning a new IP address, the process will return to the **Batch Download** window.

9'. In the **Commands** field, select **Run** to execute the **DFI302** firmware again and click **Go**.

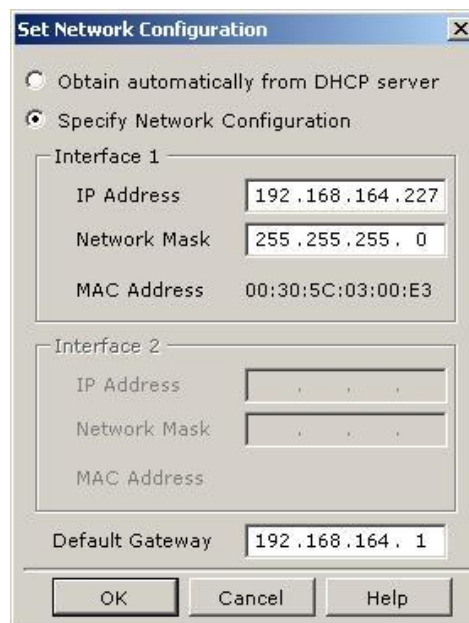
10'. Click **Close** to close the **Batch Download** window.

To change the IP address of only one controller the Batch Download option can be used. Follow the steps below:

1. Select the DFI302 controller;
2. Right-click it and the following window will appear:



3. Select **Set Network Configuration...** and the following window will appear:



4. The default option is **Obtain the IP Address from a DHCP Server**. Click the **Specify an IP address** option to change to another IP address.
5. Type the IP addresses, the subnet masks and the default gateways (provided by the network administrator) to be associated to the controller.

ATTENTION

Do not use the IP Address 192.168.164.100 (it is already being used by **DFI302**). In addition, be sure that the chosen address is not in use.

6. Click **Ok** to conclude this operation;

7. After assigning the new IP addresses, the process will return to the **Batch Download** window. In the **Commands** field select **Run** to execute the firmware again and click **Go**.
8. Click **Close** to close the **Batch Download** window.

Configuring the RTC

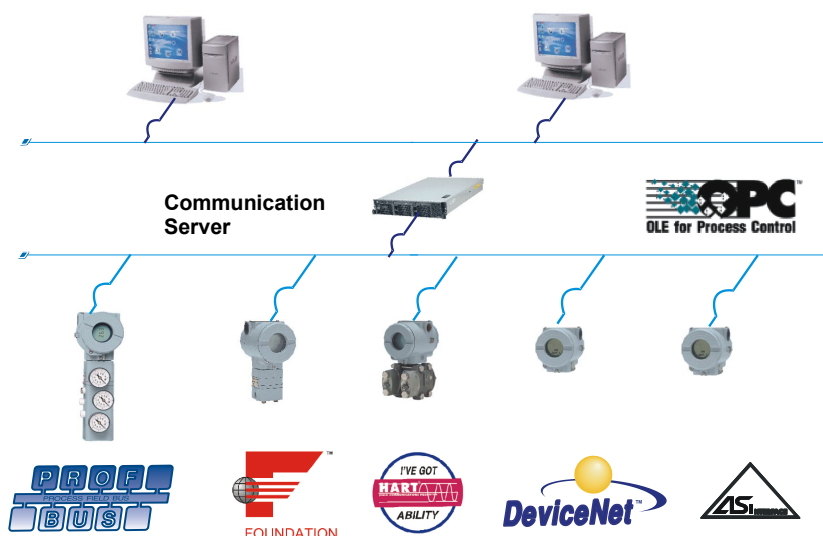
The Real Time Clock (RTC) of the DFI302 controllers, except DF51, can be configured manually, via **Batch Download** option of **FBTools**, and when available, maintained automatically synchronized via **SNTP**. For further information about **SNTP** refer to the **System302 ServerManager** appendix in the **Studio302** manual.

To configure the RTC of the DF51 controller refer to the topic "Enabling the synchronism by **SNTP** of the DF51" in this manual.

CONFIGURING THE OPC SERVERS

Introduction

Using all the benefits of OPC, formerly defined as OLE for Process Control, and at present defined as **Open Connectivity**, the user can have Fieldbus client applications for client/server based on systems at a higher level of programming, without having to deal with the details of specific Fieldbus protocol. The OPC servers for Smar controllers provide a consistent set of functions for supervision and configuration. This consistency minimizes code changes needed to make if the underlying protocol changes.



Client / server architecture via OPC

It is a distributed processing architecture, providing one single view of the system to users and applications, allowing use of services in a networked environment regardless of location, machine architecture or implementation environment.

OPC Compliant



Providing the server with an OPC interface allows any supervision client to access devices in a standard way. The OPC standard uniformizes information exchange among hardware manufacturers and supervisory (HMI - Human Machine Interface).

Smar OLE Servers are compliance-tested using OPC Foundation's Compliance Test Tool (CTT) and by attending Interoperability Workshops (IOP) annually.

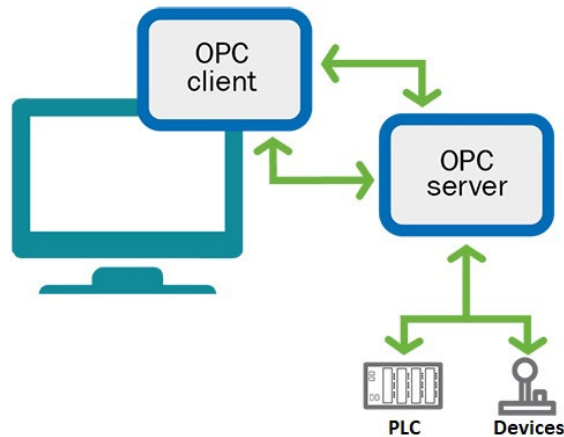
OLE for Fieldbus Configuration (OFC)

Going further on OPC benefits Smar has developed a set of functions for plant configuration which is named OFC (**OLE for Fieldbus Configuration**). This provides a way to both supervision and configuration clients work at the same time, remotely or not.

OPC – OLE for Process Control

OPC™ is a technology that allows business and supervision applications to access the plant floor data in a consistent manner. With wide industry acceptance and open architecture, OPC provides

many benefits like the OPC Server where hardware manufacturers only have to make one set of software components for customers to utilize in their applications. Another benefit is the OPC Client where software developers will not have to rewrite drivers because of feature changes or additions in a new hardware release. With OPC, system integration in a heterogeneous computing environment will become simple. Through COM/DCOM the environment shown below becomes possible.



Overview

OPC is based on Microsoft's OLE/COM technology. An OPC client can connect to OPC servers provided by one or more vendors.

Different vendors provide OPC servers. The code written by a vendor determines the devices and data that each server has access, the way how data items are named and the details about how the server physically accesses those data.

Within each server the client can define one or more OPC groups. The OPC groups provide a way for clients to organize the data which they are interested. For example, the group might represent items in a particular operator display or report. Data can be read and written. Connections based in exceptions can also be created between the client and the items in the group and can be enabled and disabled when is necessary. The 'freshness' (time resolution) of the data in the group can be specified. Within each group the client can define one or more OPC items.

The OPC items represent connections to data sources within the server. There is a value associated with each item, a quality mask and a time stamp. The quality mask is similar to that specified by Fieldbus. Note that the items are not the data sources - they are just connections to them. For example, the tags in a DCS system exist regardless of whether an OPC client is currently accessing them.

Local servers and remote servers

There are two ways of connecting OPC clients and OPC servers. The client can connect to a local server, running in the same machine. The other possibility is the connection with a remote server through the network, using the DCOM technology.

Minimum DCOM settings

1. Please certify that the hardware is installed according to its specific user manual.
2. Log with administrative rights on the local machine.
3. Certify that the TCP/IP and RPC protocols are installed in the computer.
4. Certify that the operating system is Windows 7, 10, Server 2008 or later.
5. Proceed with the installation using **SYSTEM302** setup.

There are five steps to configuring **DCOM**:

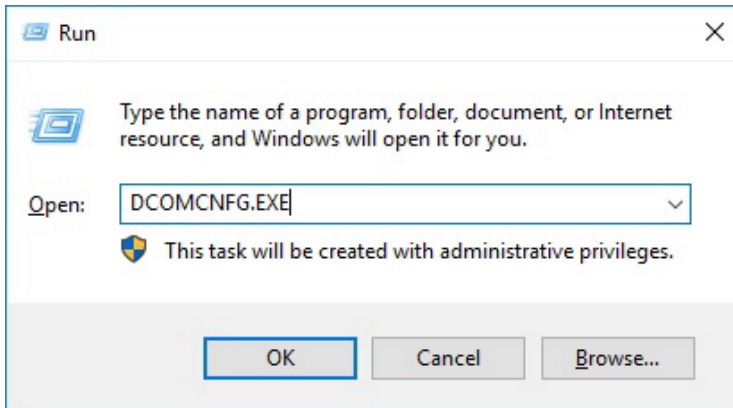
- Running **DCOMConfig.exe**
- Configuring the **General/Defaults** options

- Configuring **OPCENUM** parameters
- Configuring **OPC Server** parameters (HSE or DFI)
- Configuring Local Security Policies

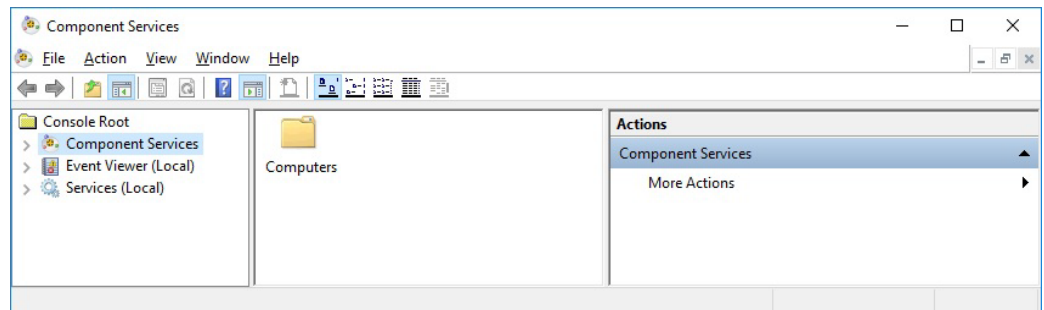
Running DCOMConfig.exe

To start **DCOM** configuration, type **DCOMCNFG.EXE** from the **Start** menu.

Use the **Start Menu** option or the **Windows** button option directly on the keyboard.

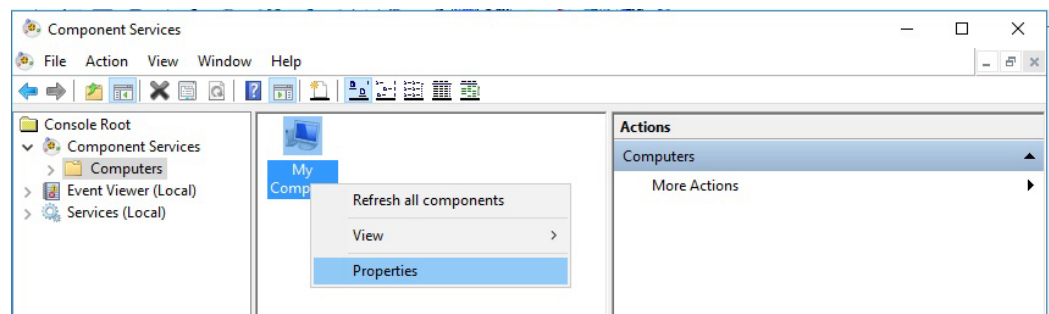


The **DCOM** configuration tool should take a few seconds to open and will be as shown in the following figure.

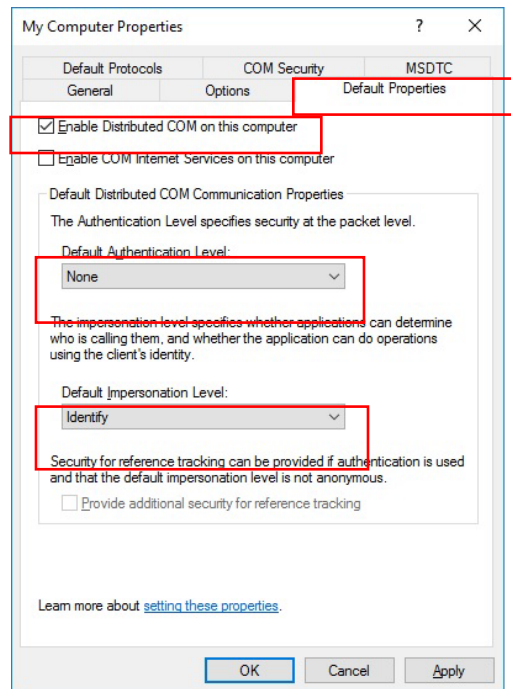


Configuring the General/Defaults options

Click **Computers** and then right-click **My Computer**. Select **Properties**.



In the window that will open choose **Default Properties**. See the following figure.



Configure the following options, as shown in the previous figure:

Check the **Enable Distributed COM on this computer**.

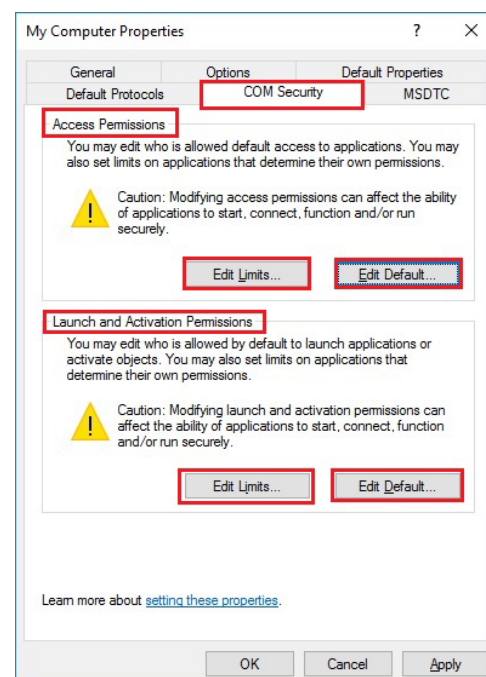
The Default Authentication Level must be set to **None**.

NOTE

This option may cause some conflict with certain Windows services, such as Windows Update. If you notice any bad behavior from any Windows component service, return to the Connect option.

The Default Impersonation Level option must be set to **Identify**.

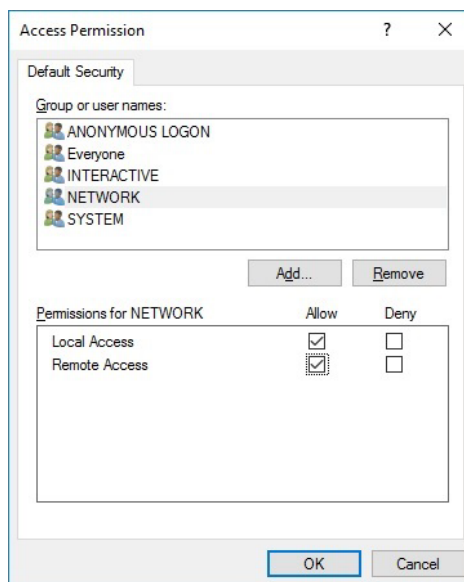
Next step: choose the **COM Security** tab.



The options that are marked with the red box must be changed.

Click **Edit Default** button in **Access Permissions** and verify that each of these accounts exists with the **Allow** access option checked.

- Anonymous logon
- Everyone
- Guest
- Network
- System
- Interactive



NOTE

If IIS (Internet Information Services) will be used as an OPC client, then the following accounts must also be added to the list of trusted:
(computer name) IIS_IUSRS;
IUSR

Do the same with the **Edit Default** option in **Launch and Activation Permissions** and assign **Allow Activation** rights to the same accounts as in the previous step.

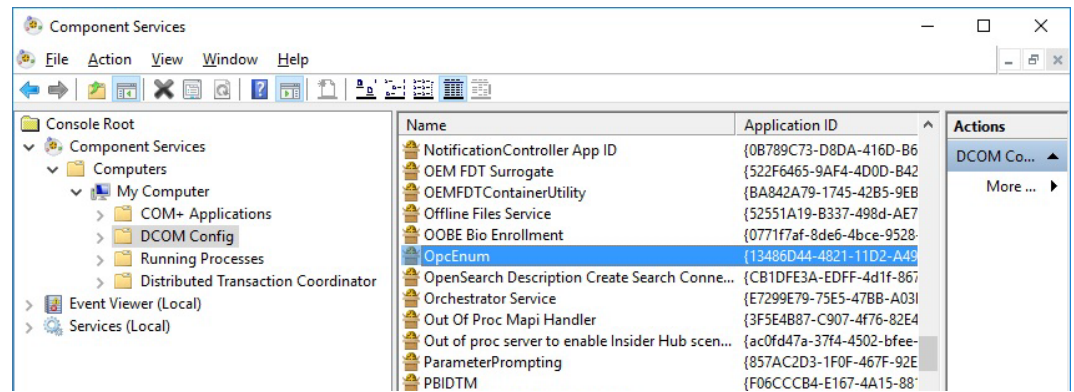
Make sure that the same settings have been adjusted for both **Edit Limits** options. These steps are done to create minimum security standards for **DCOM** configuration to ensure that a working connection can be established. After all settings are configured and the connection is established, security can be enhanced (parameters can be adjusted).

After these steps, click **Apply** and then **Ok** to close the current window.

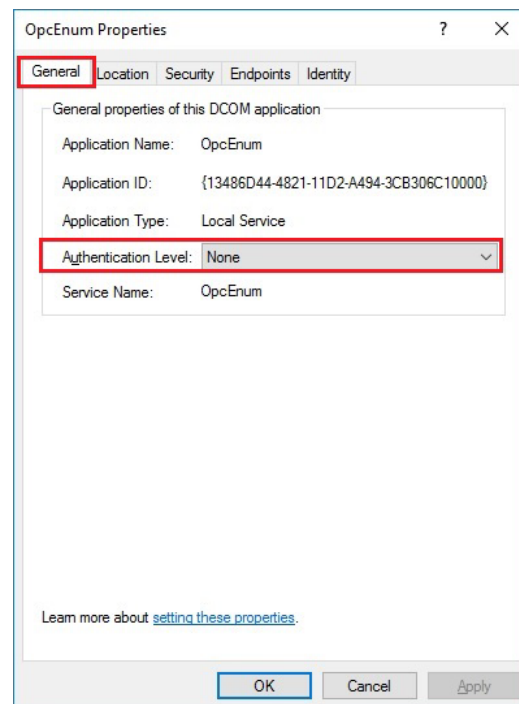
Configuring the OPCENUM parameters

Next step is to find **OpcEnum** in the list of **COM Component services**. Still in **DCOM Configuration tool**, expand the **My Computer** option and click **DCOM Config**.

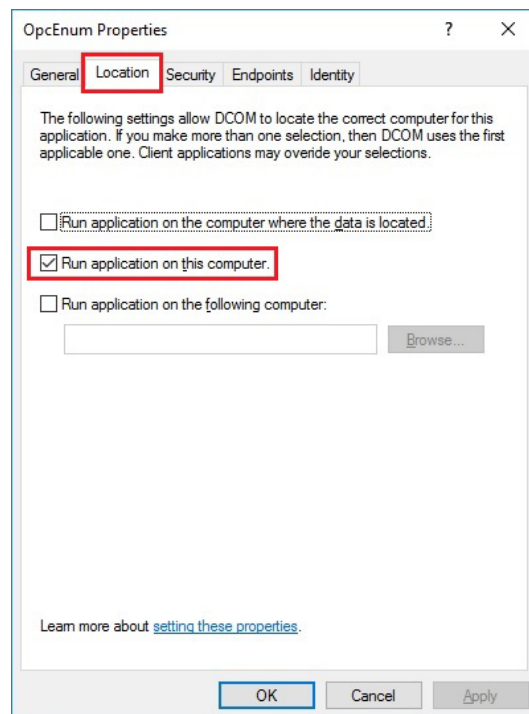
Locate **OpcEnum** in the list and right-click it and choose **Properties**.



The following window will open. On the **General** tab, select **None** for the **Authentication Level** option.

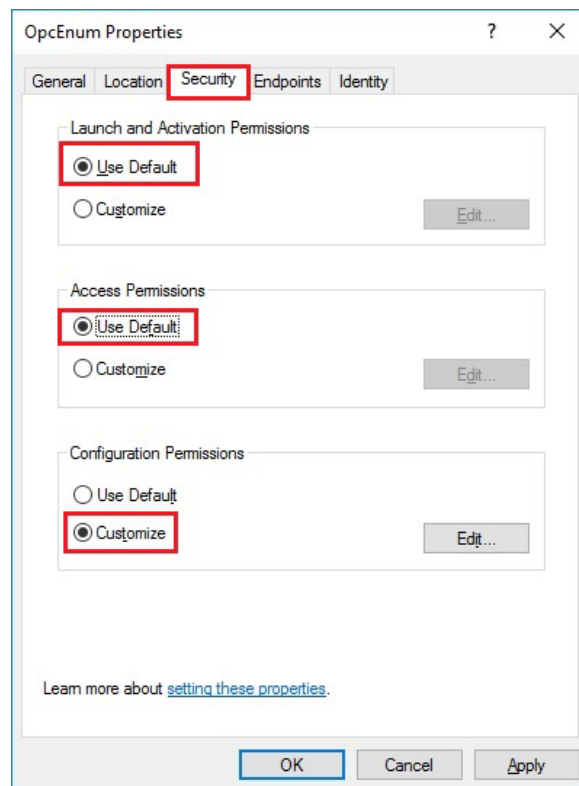


In the **Location** tab, make sure that it is in accordance with the following figure:



OPCEnum is a program that looks for the list of OPC Servers in your computer's Windows registry. **OPCEnum** needs to be able to run locally on the computer on which it is installed. Therefore, make sure that the option **Run application on this computer** is checked.

On the **Security** tab, make sure you have the following options as defined, as shown in the following figure:



There are three options to be defined in the **Security** tab:

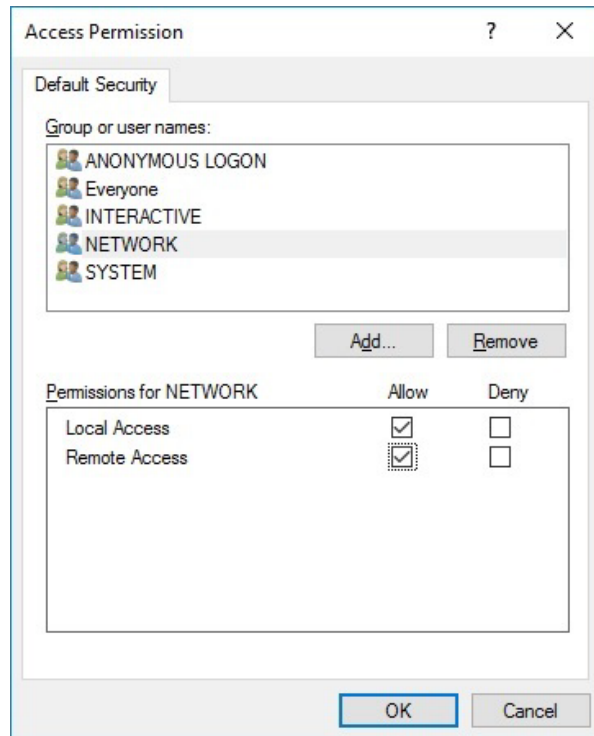
- Launch and Activation permission

Select the **Use Default** option

- Access permissions
- Select the **Use Default** option

- Configuration permissions
- Select the **Customize** option and then click **Edit**

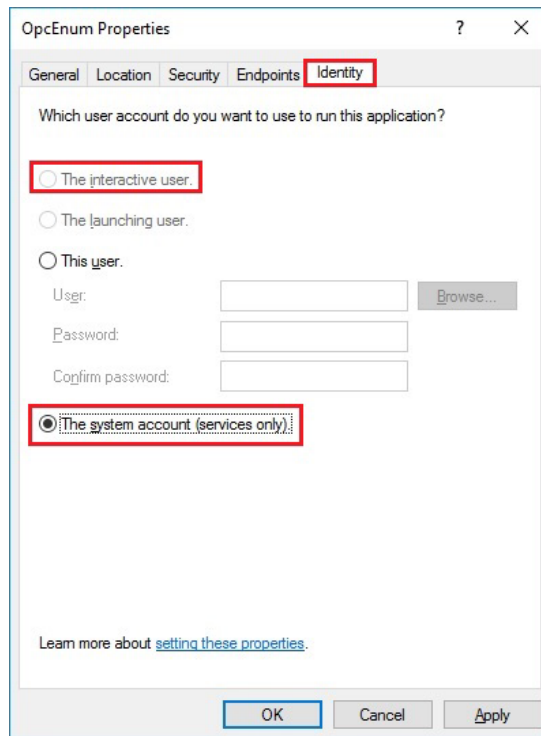
In the window that will open, be sure to specify which accounts may or may not have configuration permission, these are:



- Anonymous Logon
- Everyone
- Guest
- Interactive
- Network
- System

Make sure that all accounts above will receive **Full Control** rights.

If your system will use Internet Information Services (IIS) with an OPC client, be sure to add the following accounts additionally:
(computer name) IIS_IUSRS;
IUSR



In the **Identity** tab, use **The Interactive User** or **The system account**.

Click **OK** to save and close the **OpcEnum** options window.

Configuring OPC Server parameters (HSE or DFI)

In this configuration step, OPC Servers that were installed by **System302** and that will be used through remote access (**DCOM**) can be configured.

Locate the servers to be configured in the list that appears in the **DCOM Config** folder:

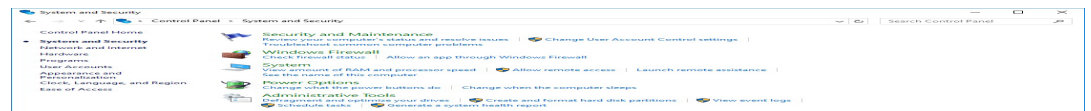
DFI OLEServer
HSE OLEServer
SNMP OPCServer

When you find the OPC Server to be configured in the list, right-click and open its **Properties** option. From there, follow the same steps and procedures that were shown in the **OPCENUM** configuration.

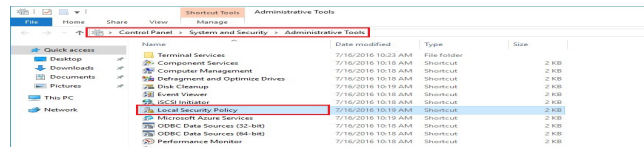
Configuring Local Security Policies

The following settings must be made on the two machines in the connection: **OPC Server** and **OPC Client** machines.

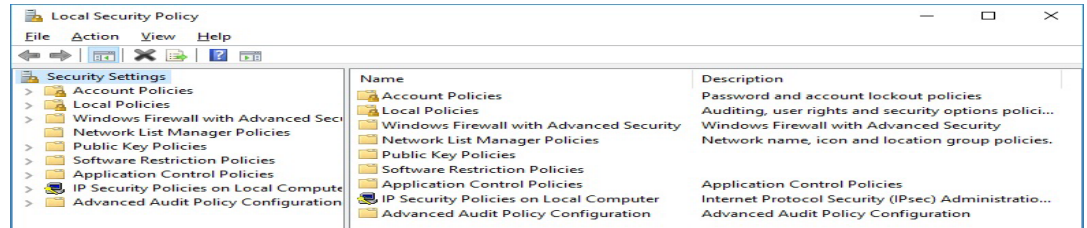
Locate the Local Security Policy: **Control Panel > Administrative Tools**



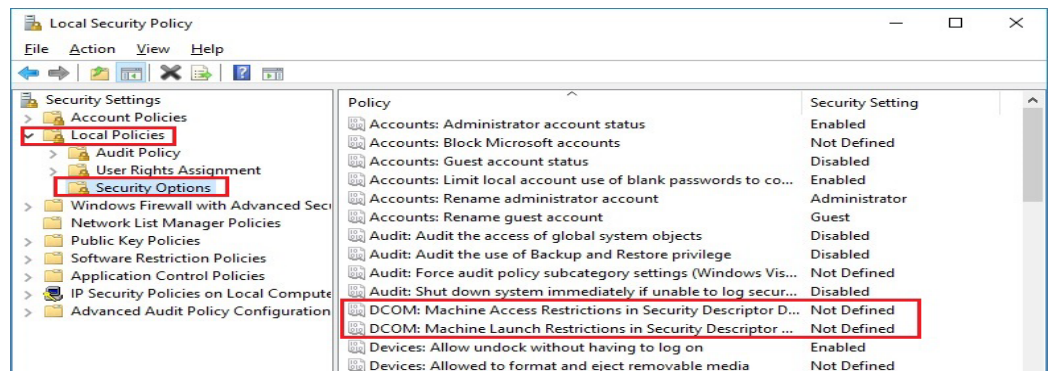
After choosing **Administrative Tools**, the following window will open. Select **Local Security Policy**.



Double-click **Local Security Policy**. The following window will open:



Click the **Local Policies** folder and open the **Security Options** folder.



DCOM policies

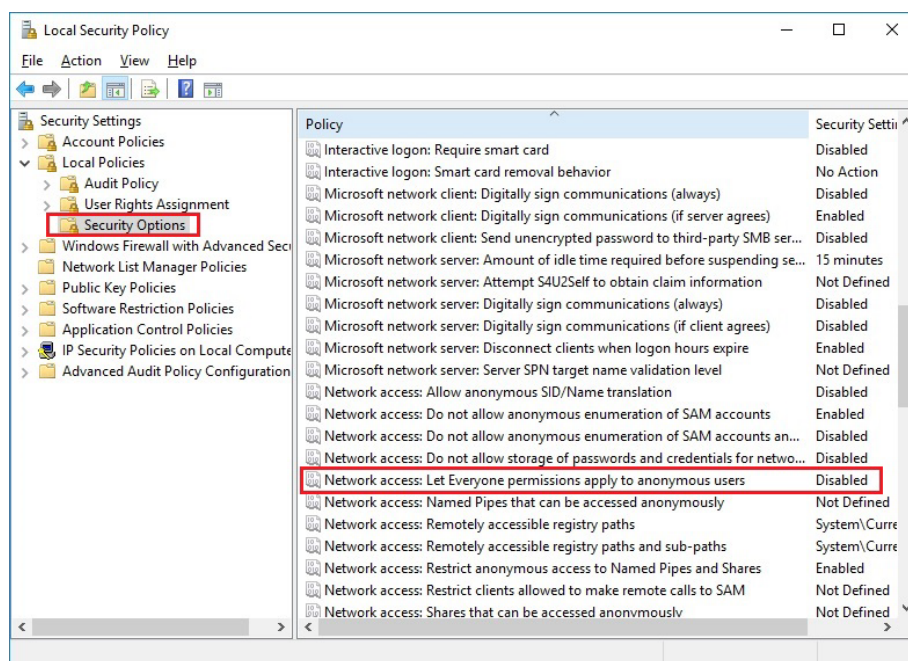
Look for the following options:

- DCOM: Machine Access Restrictions in the Security Descriptor Definition Language (SDDL) syntax
- DCOM: Machine Launch Restrictions in the Security Descriptor Definition Language (SDDL) syntax

These two options must be set to **NOT DEFINED**.

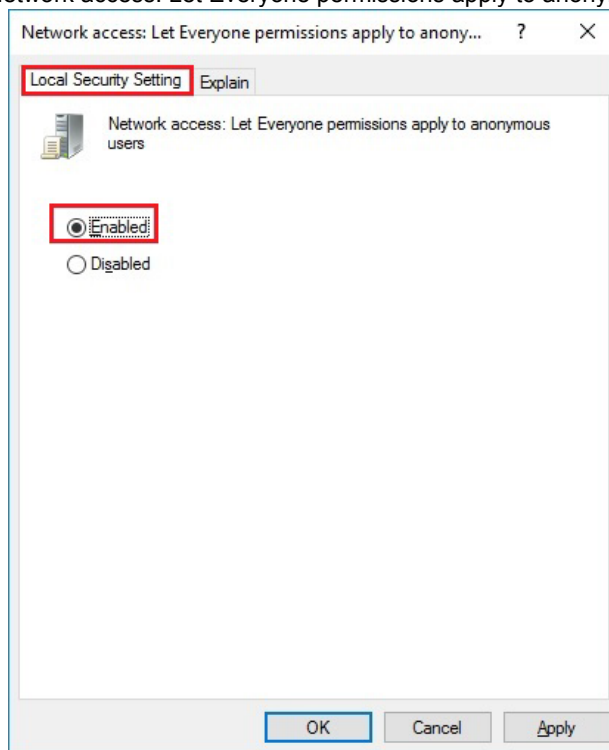
If any of these are defined, you will need to ask your IT professional or network administrator to obtain the necessary rights to access and modify these policies.

Network access, everyone permissions.



Everyone permissions also need to be applied to anonymous users. Locate the following option:

- Network access: Let Everyone permissions apply to anonymous users.



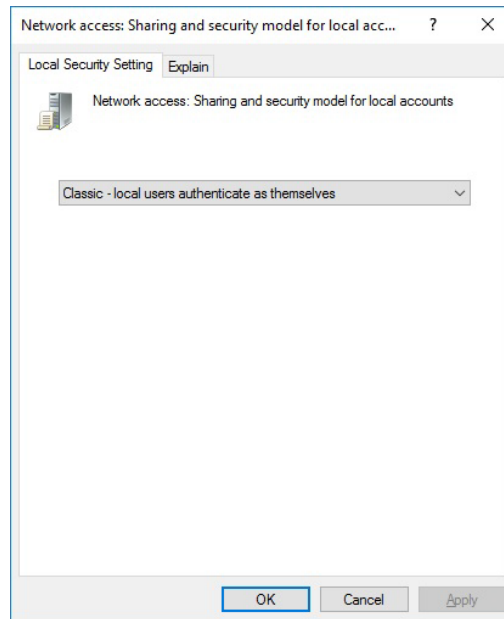
Click **OK** to save the settings and close the window.

- Network access: Sharing and Security model for local access

To configure the sharing and security model for local accounts, find the following option.

- Network access: Sharing the security model for local accounts.

The default value is **Classic**, which allows local users to authenticate as themselves.



Make sure the value is not set to **Guest Only**, which can prevent OPC connections.

Configuring Windows Firewall

1. The Windows Firewall can be accessed by the **Control Panel**.
 - 1.1. Select the **Exceptions** tab and all OPC clients and OPC Servers to be used in the machine. To add the servers, use the **Add Program** button. In the **Add Program** dialog, there is a listing of applications on the machine. If the application to be added is not listed, use the **Browse** button to find it. Smar OLE servers are located at **\Program Files\Smar\OLEServers** folder.
 - 1.2. Also add Microsoft Management Console (mmc.exe) and OPC Server Enumerator utility (OPCEnum.exe). They can be found at **\WINDOWS\system32** folder.
 - 1.3. Add TCP port 135 to the list of exceptions. This can be done clicking the **Add Port** button. In the **AddPort** dialog, fill the fields as follows:
 - Name: DCOM
 - Port number: 135
 - Choose the TCP button.

DFI OLE Server details

DFI OLE Server is an OPC server (*server-side*) used to carry out connection between an OPC client/supervisory (*HMI Human Machine Interface*) and **DFI302** connected in the network.

- Please certify that **DFI302** is correctly installed in the network.
- File: DfiSvr.exe
- ProgID: Smar.DFIOLServer.0
- Name: Smar OPC & Conf Server for DFI302

HSE OLE Server details

HSE OLE Server is an OPC server (*server-side*) used to carry out connection between an OPC client/supervisory (*HMI Human Machine Interface*) and any HSE device connected in the network.

HSE device definition: Any FOUNDATION fieldbus device type connected directly to HSE media. All HSE devices contain a FDA Agent, an HSE SMK, and an HSE NMA VFD. Examples include Linking Devices, I/O Gateways, and HSE field devices. **DFI302** is a Linking Device.

- Please certify that **DFI302** or any HSE Device is correctly installed in the network.
- File: HseSvr.exe
- ProgID: Smar.HSEOLEServer.0
- Name: Smar OPC & Conf Server for HSE

A&E OPC Server details

With the A&E OPC Server is possible that events and alarms of H1 fieldbus devices, correctly configured, on an HSE Fieldbus network generate alarms, and they can be seen with an OPC A&E client.

IMPORTANT

The Alarm and Events Server (A&E OPC Server) configuration requires prior knowledge of Fieldbus standard for function block configuration specially of those which are necessary to a fieldbus device generate events.

The Syscon is a configuration tool for A&E Server (for further details refer to Syscon manual). Follow the next steps to configure a fieldbus device to generate events, and to prepare the OPC A&E Server to receive those events.

- Configure the offline parameters of H1 device, with **Syscon**;
- Use the **Export Tags (Syscon's function)**, enabling the option that generates information for Alarms and Events (refer to Syscon manual) in order to generate the AlarmInfo.ini file.
- Download the configuration.
- Run the OPC A&E client.

- File: AESvr.exe
- ProgID: Smar.AEServer.0
- Name: Smar Alarm & Event Server

HDA OPC Server details

With the HDA OPC Server is possible that trends events generated from H1 fieldbus devices, correctly configured on an HSE Fieldbus network, can be stored and made available to an OPC HDA client.

The **Syscon** is a configuration tool for HDA Trend Server (for further details refer to **Syscon** manual). Follow the next steps to configure a fieldbus device to generate trend objects, and to prepare the HDA Server to receive those events.

- Select which devices and blocks will have their parameters monitored and stored by the HDA server (for further details refer to **Syscon** manual);
- Download the configuration;
- Use the **Export Tags** (**Syscon's** function) to generate the TrendInfo.ini file (it provides the information to the HDA server);
- Run the OPC HDA client.
- File: HDASvr.exe
- ProgID: Smar.HDAServer.0
- Name: Smar HDA OPC & Conf Server for System302

HSE Device Definition

Any device FOUNDATION fieldbus can be connected directly with an HSE media. All the HSE Devices have one FDA Agent, one HSE SMK and one HSE NMA VFD. For instance, Linking Devices, I/O Gateways, and HSE field devices. The **DFI302** is a Linking Device.

- Please certify that **DFI302** or any HSE Device is installed correctly under the network.
- File: HseSvr.exe
- ProgID: Smar.HSEOLEServer.0
- Name: Smar OPC & Conf Server for HSE

Information for firewall configuration

The OLE servers make extensive use of network resources to exchange data with **DFI302**. If the machines running the OLE servers are protected by a firewall, it is necessary to open the ports needed to make the network resources available. The following ports are used by the servers. The steps to be taken to open these ports are firewall-specific.

SE ports: 4987 (UDP) and 4988 (TCP)
HSE ports: 1089, 1090, 1091 and 3622 (all UDP)
SNTP port: 123 (TCP)
Modbus TCP port: 502 (TCP)
Telnet port: 23 (TCP)
HTTP port: 80, 8080 (all TCP)
DCOM port: 135 (TCP)
SNMP port: 161 (UDP)

Since **DFI302** also acts as a client, it is necessary to open ports higher than 1024.

SmarOleServer.ini Configuration

SmarOleServer.ini file, located under OleServers folder provides some SECTION and KEYS which are permitted enable and disable logs, set timeouts, configure network details, etc.

IMPORTANT

Not all keys and sections of the SmarOleServer.ini file are described here. The access by ServerManager defines which parameters usually have to be modified. The change in advanced parameters for which the user has no technical knowledge can lead to a system malfunction, or even make it inoperable.

Let's to describe these sections:

- In Log and LogForOPC Sections, it is possible to enable some log features and see the results in Events.log and EventOPC.log files respectively. Both files have its (.log#) file used for swap.

```
[Log]
GENERAL=0
DEBUG=0
MEMORY=0
INIT=1
DRIVER=0
TRANSFER=0
TRANSACTION=0
CONF=0
OPC=0
OPCDEBUG=0
IDSHELL=0
;=0      (Default) Log disabled
;=1      Enable log and see the results in Events.log and Events.log#
[LogForOPC]
GENERAL=0
;=0      (Default) Log disabled
;=1      Enable log and see the results in EventOPC.log and EventOPC.log#
```

- In NIC Adapter Section, in the case of more than one NIC adapter are installed in the machine, choose the desired NIC adapter to be connected with the local DFI OLE Server.

```
[NIC Adapter]
; If more than one NIC (Network Interface Card) are installed in the local machine
; it is necessary to inform the DFI OLE Server to use one (NIC) or two (NIC and NIC2) adapters.
; In the NIC key (next lines), set the IP which is configured in each NIC and remove ';'
;NIC=xxx.yyy.www.zz
```

- In DFI Time Settings Section, tune the better startup time which the DFI OLE Server takes to look for **DFI302** in the network. The default time use to be enough if not using routers.

```
[DFI Time Settings]
; Define a delay which the server will wait till complete DFI302 connection
;=13 (Default) 13 seconds before concluding server connection with DFI302s
NETWORK_STARTUP=13
```

- In Remote DFI Section, when using Routers in the Network topology, insert the other IPs located out of the local subnet. Do not forget to tune in DFI Time Settings Section, the better time necessary to DFI OLE Server.

```
[Remote DFI]
; Specify on this section IPs to be reached in remote networks.
; Remember to set Default Gateway under DFI settings using FBTools.
; Format: xxx.yyy.zzz.sss=1 enable IP polling.
;         xxx.yyy.zzz.sss=0 disable IP polling.
;192.168.164.100=0
```

- In the Supervision Section, it is possible to switch the Server (PCI or DFI) to emulation. This mode is only used for debug purposes.

[Supervision]

; This section is used for Supervision purposes.

; OPC_TIMEOUT is the maximum time which the Server waits for data refresh.

; EMULATION turns the Emulation on.

; EMULATION_RATE specifies the rate for refreshing of emulation.

OPC_TIMEOUT=30

;=30 (Default) 30 seconds

EMULATION=OFF

;=ON Activate Emulation Mode for Supervision

;=OFF (Default) Normal Mode

EMULATION_RATE=1000

;=1000 (Default) 1000 seconds, valid when EMULATION=ON

In the Configuration Section, it is possible to configure timeouts for each Syscon Configuration procedure. DO NOT CHANGE ANY VALUE IN THIS SECTION WITHOUT SMAR R&D RECOMMENDATION.

[Configuration]

;Default Timeout 10 seconds

Timeout.Default=30

Timeout.MULTILINKTOPOLOGYREQ=60

Topology Upload

The topology upload allows the OPC Server to build a list with the Fieldbus network devices and the corresponding function blocks. This feature is enabled setting the **TOPOLOGY_CACHE** parameter to **ON**, in the SmarOleServer.ini configuration file. The search for the function block when starting the supervision is optimized, because at the monitoring establishment the server has the block position that will be monitored, so it avoids multiple requests. The initial establishment of the supervision becomes a little slow. To balance this effect, it is possible to configure the quantity of supervision points will be requested for all devices, enabling them quickly, through the **BROADCAST_LIMIT** parameter. Through setting this parameter, the supervision of the essential points is done quickly, the search of remain points is optimized, decreasing the Ethernet network traffic.

The part of the SmarOleServer.ini file with the setting is showed below:

TOPOLOGY_CACHE=OFF

; TOPOLOGY_CACHE defines if the server will upload the network topology. The topology is

; used to optimize the supervision, avoiding broadcasts on GetIDReq.

;=ON Turn on this feature. It may take a little longer for the server to start receiving

; data.

;=OFF (Default) Turn off this feature. No topology information will be stored, and GetIDReq will

; be done using broadcast.

BROADCAST_LIMIT=100

; BROADCAST_LIMIT is used to speed up the supervision. The server will allow up to the limit

; value of items

; to be discovered by broadcast. This feature is useful when the server is instantiated and the user

; wants to supervise a set of items without waiting for the topology upload.

;=100 (Default)

System302 ServerManager Application

System302 ServerManager has been developed to handle almost all configurable features available in Smar OPC Servers like that described in SmarOleServer.ini configuration section. For further details about **System302 ServerManager**, refer to the **Studio302** manual.

Optimizing DF51 Access to Subnets

Once installed the original **SYSTEM302** package, defaults settings works fine for the most cases, when talking about DF51 connected to Ethernet; mainly when the typical system is composed by one computer, one Network Interface Adapter (NIC) and one subnet. If the topology is different, some steps are necessary to customize the access in optimal way, and sometimes to permit DF51 to work properly.

The SmarOleServer.ini file is used to configure many parameters related with subnet access, as seen previously.

Take a look in [NIC Adapter], [DFI Time Settings], [Remote DFI] and [Remote DFI2] sections, for more details.

[NIC Adapter]

If more than one NIC (Network Interface Card) are installed in the local machine it is necessary to inform the DFI OLE Server to use one (NIC) or two (NIC and NIC2) adapters. In the NIC key (next lines), set the IP which is configured in each NIC and remove ';':

```
NIC=192.168.164.20
NIC2=192.168.165.17
```

[DFI Time Settings]

The **NETWORK_STARTUP** parameter defines a delay which the server will wait till complete DFI connection = 13 (Default value) 13 seconds before concluding server connection with DFIs.

```
NETWORK_STARTUP=13
```

The **LOCAL_POLLING** parameter enables or disables local connection to DFIs. If this parameter is **ON** (Default), it will enable the connections to all available DFIs in the same subnet. If this parameter is **OFF**, it will disable the connections to all available DFIs in the same subnet. Only IPs listed in Remote DFI sections will be polled.

```
LOCAL_POLLING=ON
```

[Remote DFI]

Specify in this section IPs to be reached in remote networks by NIC with redundancy. Specify also in this section desired local IPs when **LOCAL_POLLING=OFF**. Remember to set Default Gateway under DFI settings using **FBTools**.

In the format xxx.yyy.zzz.sss, the IP polling can or not be enabled.

- 1 enables IP polling.
- 0 disables IP polling.

```
192.168.161.72=1
192.168.164.197=0
```

[Remote DFI2]

Specify in this section IPs to be reached in remote networks by NIC2 with redundancy. Specify also in this section desired local IPs when **LOCAL_POLLING=OFF**. Remember to set Default Gateway under DFI settings using **FBTools**.

In the format xxx.yyy.zzz.sss, the IP polling can or not be enabled.

- 1 enables IP polling.
- 0 disables IP polling.

```
192.168.164.71=1
192.168.161.19=1
```

Enabling the Synchronism by SNTP

The SNTP (*Simple Network Time Protocol*) client was incorporated as a new feature in the DFI302 demanded by the Market. Its purpose is to establish a synchronism among all equipments in the control area through a time server. So, the reports can be done using a consistent (unique) time reference. The SNTP protocol enabling in the controllers is done as follows:

Enabling the synchronism by SNTP in the DF51

It is implemented as a thread that sends SNTP unicast requests to specified servers. This thread may be started by the system if the user includes on the **Syscon** a DFI302 bridge (DF51), and in its transducer block, set a valid IP address in the **SNTP_PRI_TIME_SRV** parameter. This address will be the SNTP primary server, already set in a workstation by the Administrator.

If the user has an additional time server, the **SNTP_SEC_TIME_SRV** parameter can be used to configure the IP address of the secondary SNTP server.

The SNTP client sends and receives NTP IPv4 packages from the configured STNP/NTP servers. Such packages contain a timestamp indicating the current time. The package also contains information about where the server is in the hierarchy of time servers. The client will accept any NTP packages from servers using version 3 of the protocol.

If there are no packages from this primary server, after a time set on the **SNTP_REQ_TIMEOUT_ms** parameter, the client (DFI302), automatically, will change its time requests to another server which was set on the **SNTP_SEC_TIME_SRV** parameter, and it will be responsible for the client synchronism.

If there are no responses from all configured servers, the **NOT SYNCHRONIZED** message will be indicated in the **SNTP_PRI_TIME_SRV** and **SNTP_SEC_TIME_SRV** parameters. After establishing the communication with, at least, one server, the set IPs will reappear in these parameters.

Another parameter than can be set is the **SNTP_REQ_INTERVAL_ms** which determines the interval that the SNTP requests will be solicited to the NTP server.

The general algorithm is as follows: if the system clock has not been yet set via a NTP time updating, then the client will send out requests to SNTP server during the time intervals configured in the **SNTP_REQ_INTERVAL_ms** parameter.

After the first clock synchronization of the equipment, the SNTP client will continue sending requests according to the value set in **SNTP_REQ_INTERVAL_ms** parameter in order to keep the clock synchronized.

The system clock accuracy was defined in 1 second and is represented by **SNTP_TARGET_TIME_PRECISION_us** parameter. The SNTP client will update the system clock when the time difference with the received timestamp is greater than 0.5 seconds. The change is done as a step.

In the DFI302 transducer the parameter that will update the new time is **APPLICATION_TIME** parameter.

Setting SNTP Server on Windows Platform

In order to have the SNTP Client working, you will need a SNTP Server. This can be done in a computer running Windows™ Platform. Administrative knowledge will be necessary. Please, check the Network Administrator for further information.

TROUBLESHOOTING

Some configurator systems use a Windows™ API in order to obtain the data and hour of the system. When the Daylight-saving time is established, this API will be based on the USA time zone. So, it can have difference between the SNTP client and server hour.

To solve this problem, the following steps should be followed for Windows™ platform:

- Click Start→Settings→Control Panel;
- Double-click Date/Time;
- Select the time zone tab and choose USA&Canadian option as default;
- Disable the Automatically adjust clock for daylight saving changes option;
- Adjust the clock for the correct hour.

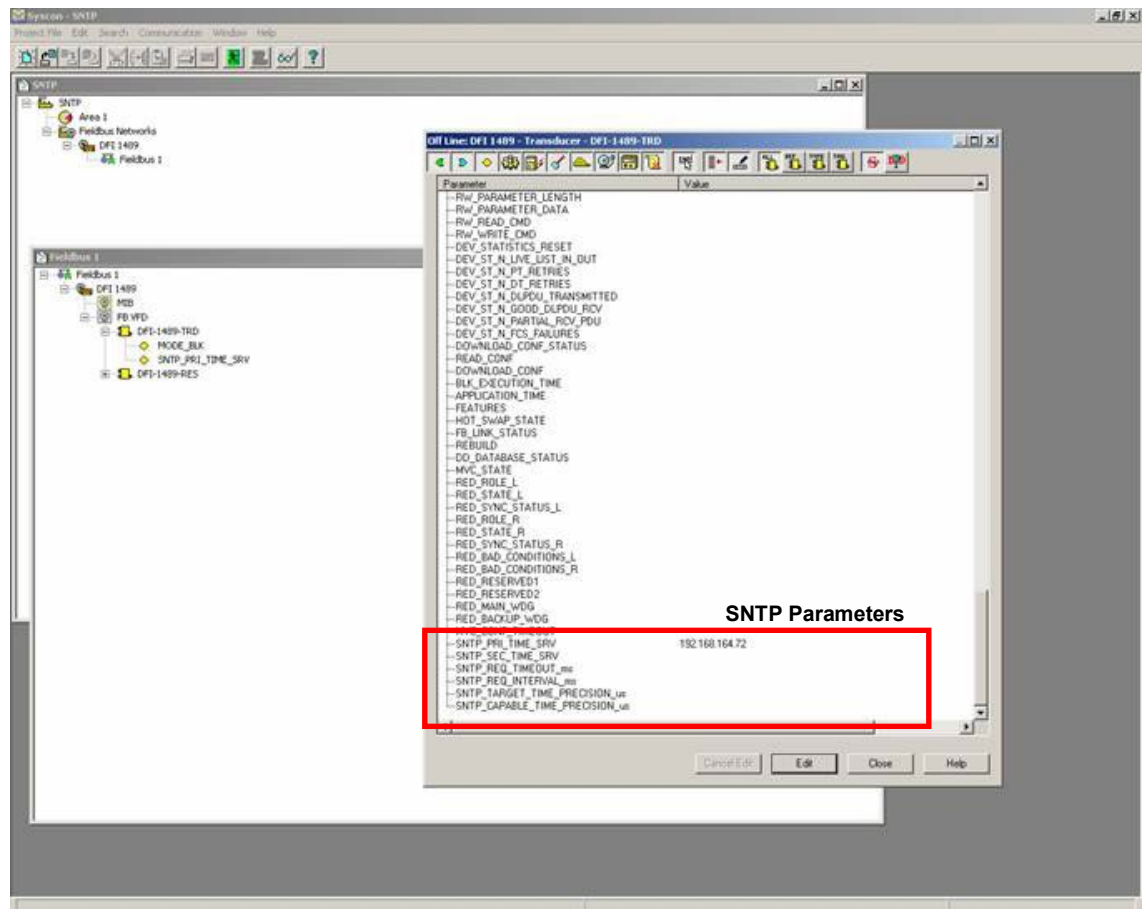
Note: This procedure must be done only when the local hour is different of that set in the configurator system.

Syscon Configuration

On the **Syscon** create a new strategy and add a DF51 bridge (firmware version 3.9.1 or later).

On this bridge inserts a Transducer Block and set the IP Address of the SNTP server on the **SNTP_PRI_TIME_SRV** parameter.

If the current server is running well, the **APPLICATION_TIME** will be updated and the SNTP client will work without any problem. Otherwise, the **APPLICATION_TIME** parameter will not be updated and the **NOT SYNCHRONIZED** message will be showed in the **SNTP_PRI_TIME_SRV** and/or **SNTP_SEC_TIME_SRV** parameter.



After the configuration download, it is possible in the Online mode to observe the update of the system clock in the **APPLICATION_TIME** parameter.

Device Revision and Capability Files

The SNTP protocol is available from the DD Rev 7. If the user does not have installed the files, contact Smar Technical support in order to get them, and if necessary, the firmware. Once getting the files (DD and Capability Files) copy them to the Device Support directory where the Smar applications are installed. For example: **C:\Program Files\Smar\Device Support\000302\0008**.

DFI302 OLEServer

In order to read this **APPLICATION_TIME** parameter on the supervisory, it is necessary to install the DFI302 OleServer version 3.8.1.3 or higher.

To receive data from Tags that are originally set do **BYTE ARRAY** as a formatted string, the **WORK_WITH_BSTR** key in the [Supervision] section of the SmarOleServer.ini file must be changed.

WORK_WITH_BSTR=0

*; WORK_WITH_BSTR enable/disable the feature of dealing with data type BSTR
; =0 (default) default array - NO BSTR
; =1 Default array and can deal with BSTR when requested
; =2 Default BSTR (visible string Foundation)*

Set this key to **WORK_WITH_BSTR=1** and in the OPC Client add the Tag using the requested data type.

Syscon will deal with this tag as usual, receiving as **BYTE ARRAY** and performing the change to see the time formatted as following:

Nov 23, 2004 09:23:44:980 + 2/32 of a millisecond.

The HMI requires the data type of the tag be in the string format. The DFI Server will perform the change, receiving **TIME_STAMP** as **BYTE ARRAY** (8 bytes) and passing it on as String.

Considerations about parameters and firmware

About Parameters

Some pre-defined limits and conditions were imposed for some parameters in order to have the plain functionality of the SNTP Server.

The default values are:

SNTP_REQ_TIMEOUT_ms = 4000 (4s);
SNTP_REQ_INTERVAL_ms = 10000 (10s);
SNTP_TARGET_TIME_PRECISION_us (Read/Write) = 1000000 (1s)
SNTP_CAPABLE_TIME_PRECISION_us (Read Only) = 1000000 (1s).

Limits values are imposed for: **SNTP_REQ_INTERVAL_ms** and **SNTP_REQ_TIMEOUT_ms**. The first will support a minimum value of 2000 (2s) and free maximum value. The second one (timeout) must to be set to 1000 (1s), i.e., **SNTP_MIN_REQUEST_TIME / 2** and a maximum value to the timeout parameters will be approximately 90% of the entered Request Interval value and never equals to or higher of it (otherwise it does not make any sense, the system will adjust the value automatically and will never work).

A special consideration should be done for **SNTP_PRI_TIME_SRV** and **SNTP_SEC_TIME_SRV** parameters. In these parameters, when the user sets the value 0 or the message **NOT SYNCHRONIZED** is being showed, the SNTP protocol will not work properly, and all tasks and connections will be stopped. On other hand, if the first parameter (**SNTP_PRI_TIME_SRV**) does not have a valid IP address or it is blank, but the second has one, the communication with the server will not be established and the message **NOT SYNCHRONIZED** will be indicated in both parameters. So, it is necessary just one valid SNTP/NTP server and if it is unique, it must be set on the **SNTP_SEC_TIME_SRV** parameter.

About System302 and SNTP component

The SNTP feature is incorporated on **SYSTEM302** Version 6.1.9 or later.

Enabling the synchronism by SNTP in the DF62, DF63, DF73, DF75, DF79, DF81, DF89, DF95, DF97 and DF100 controllers

To enable the synchronism by SNTP in the other DFI302 controllers refer to “How to synchronize the DFI302 Controllers with a common time” section in the **SYSTEM302 Handbook**.

Section 5

CONFIGURING STRATEGIES

Introduction

Syscon is the configuration tool for the **SYSTEM302** package to develop system strategies.

Strategy examples will be explained in the following sections. They were developed by using some DFI302 controllers. There is a section which shows how to add discrete control through ladder logic by using Flexible Function Block (FFB).

To create the strategy, it is necessary create the area before, or use an area template.

It is possible to create, or edit, an area from the **Studio302**. In the **Studio302** interface select **Areas**. A window will appear listing all the areas of database.

To create a new area from the **Studio302**, left-click inside the window **Areas**, then choose **New Area**.

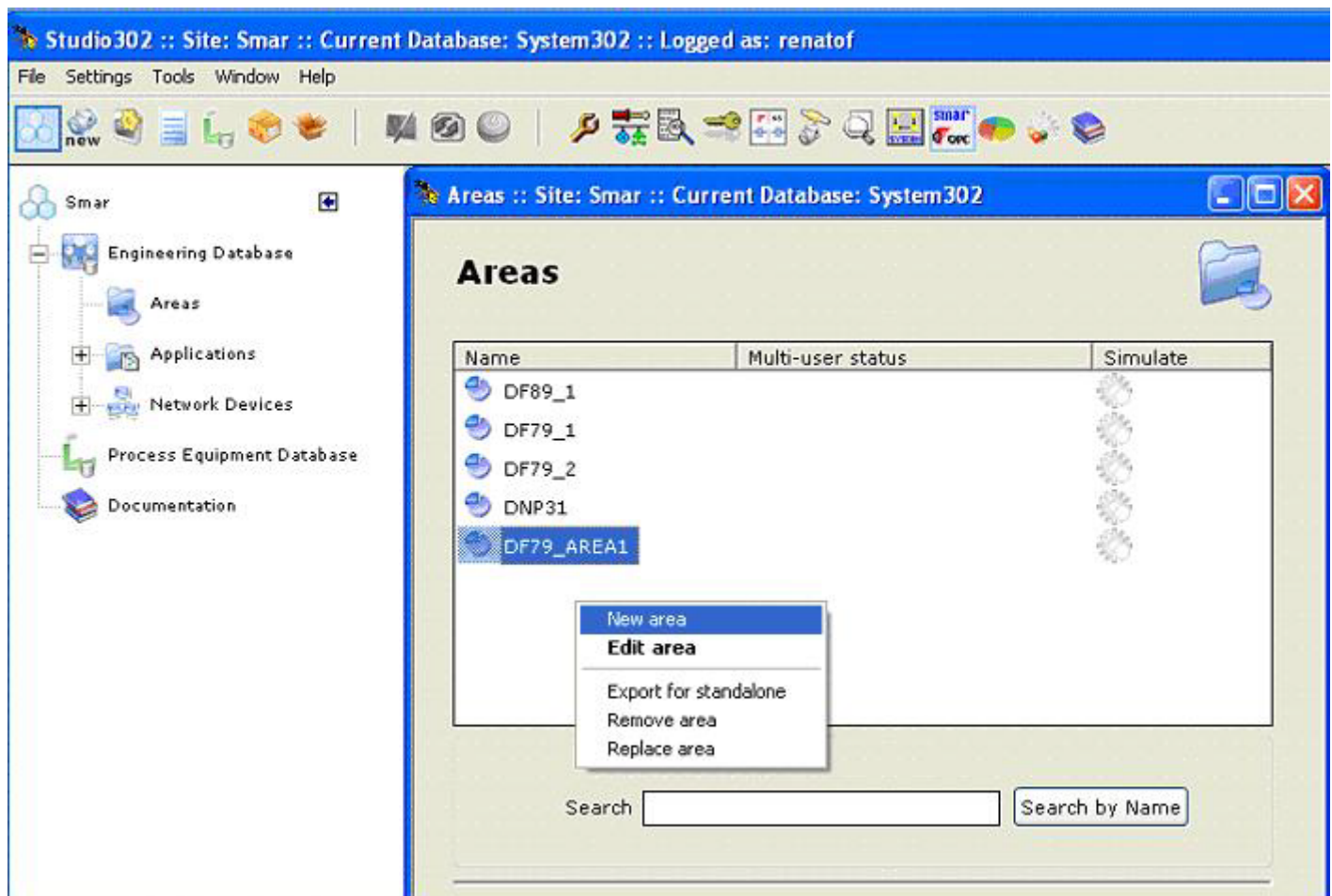




Figure 5. 1 – Creating an area from Studio302

Another way to create a new area is from **Syscon**. Click the icon  in the **Studio302** toolbar, and **Syscon** will be launched. Choose **File** → **New**, or through the **Syscon's** toolbar, choose **New** button . See the following figure.

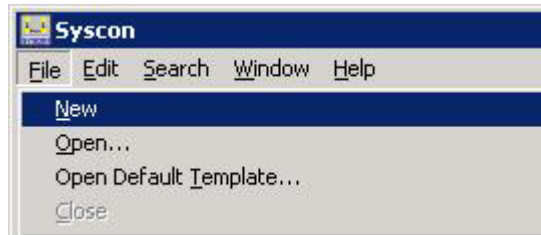


Figure 5. 2 – Creating new area from Syscon

The user will be able to choose the area type from the pop-up menu shown below.

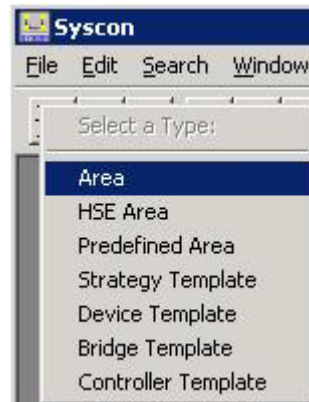


Figure 5. 3 – Choosing the area type

Area Types

The area type options will be explained in the following topics.

Area

Choosing **Area** option, a window to give a name to the area will open, as shown in the next figure:.

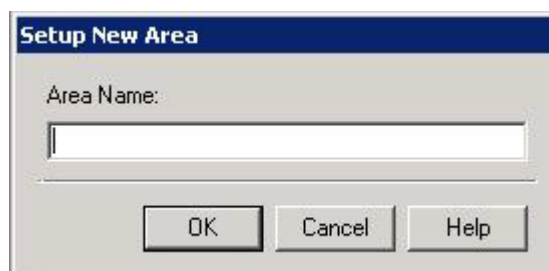


Figure 5. 4 – Naming the area

Type the area name, click **OK** and the following window will open:

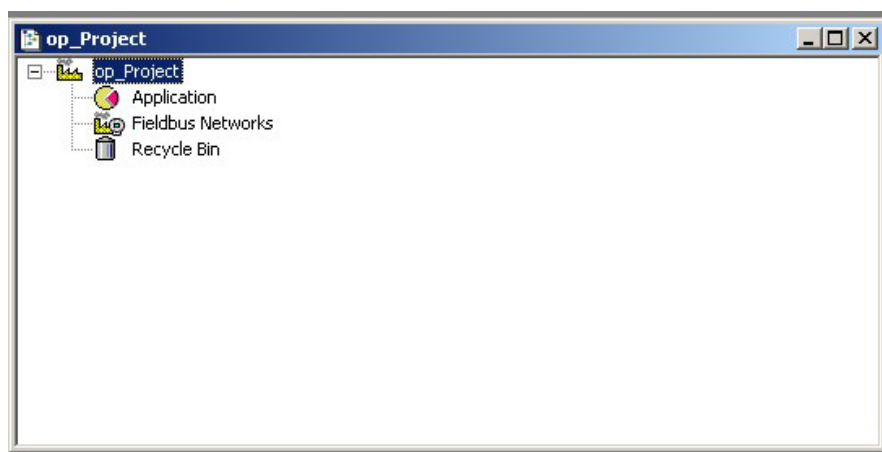


Figure 5. 5 – Area option selected

This window has the icons:

- **Application** – Logical Plant. To insert the control strategies into this part.
- **Fieldbus Networks** – Physical Plant. To insert devices and function blocks used in the project.

This area type may use the DF51 controller as bridge. For further details about this strategy type, refer to the section Creating a fieldbus strategy by using DF51.

HSE Area

When choosing **HSE Area** option, a window to give a name to the area will open. After naming the area, the following window will open:

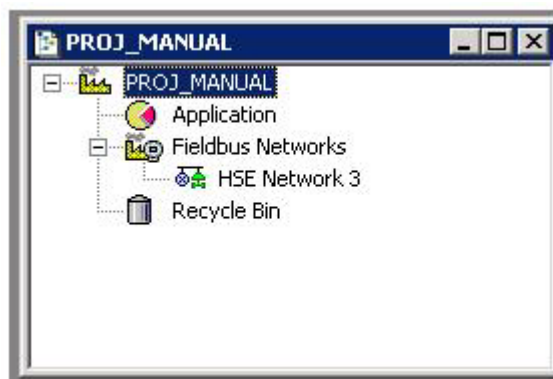



Figure 5. 6 – HSE Area option selected

This window has the icons:

- **Application** – Logical Plant. To insert the control strategies into this part.
- **Fieldbus Networks** – Physical Plant. To insert devices and function blocks used in the project.

If the **Fieldbus Networks** is not expanded, click the  sign next to its icon, and this way, the HSE network of this area will be shown. For this area type can be developed strategies which have, for example, DF62 as bridge, or DF73 as controller. For further details refer to the sections Creating a FOUNDATION fieldbus strategy by using the DF62/DF63 and Creating a Profibus configuration by using the DF73, DF95, or DF97.

Predefined Area

When the choice is **Predefined Area**, a window to select the area template is showed initially. The next figure shows this window with the template options:

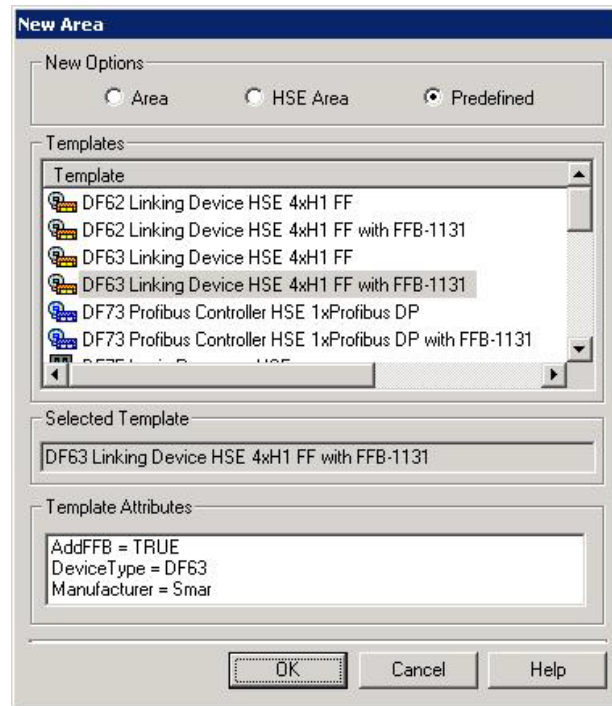


Figure 5.7 – Predefined Area option selected

When choosing one option, a window to give a name to the area will open. After naming the area, the following window will open:

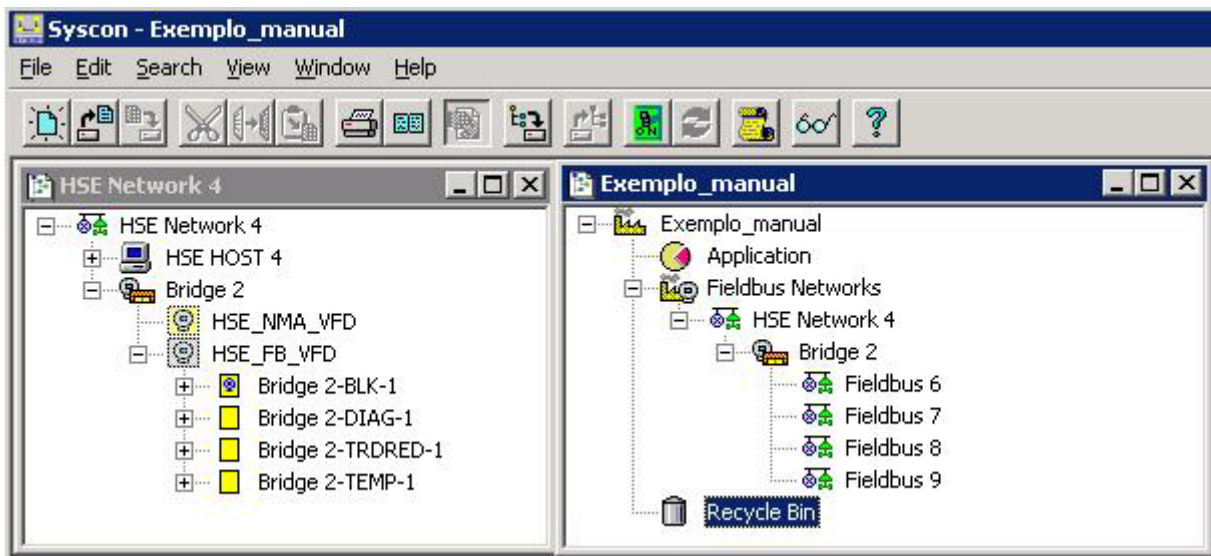


Figure 5.8 – Predefined Area option windows

To exemplify this area template, the DF62 was selected. For further details about this area type, refer to the **Syscon** manual.

Strategy Template

Choosing **Strategy Template** option, it shows the window to add strategies:

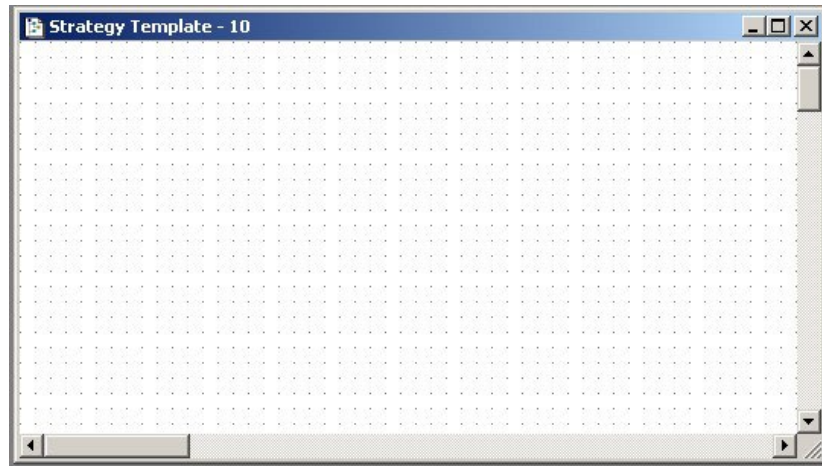


Figure 5.9 – Strategy Template option

Using this option, many projects can use this strategy.

Device Template

For the **Device Template** option, it shows the box to select the devices that will be templates for the project.

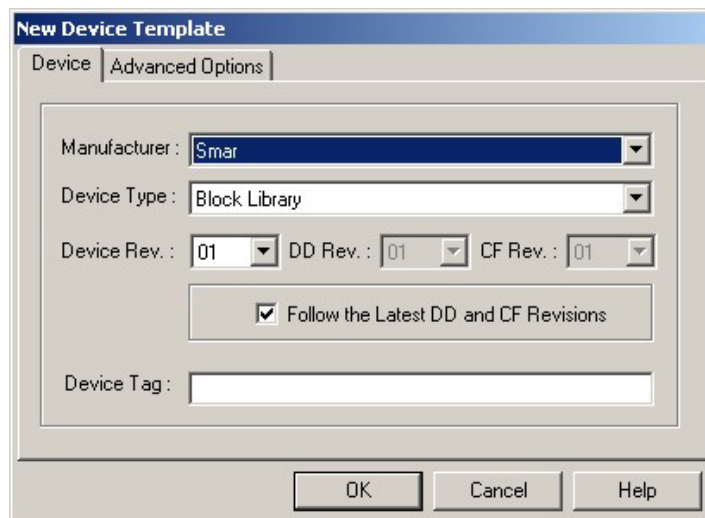



Figure 5.10 – Device Template option

Bridge Template

Choosing the **Bridge Template** option, it shows the box to select the device that will act as bridge for the area, as shown in the next figure:



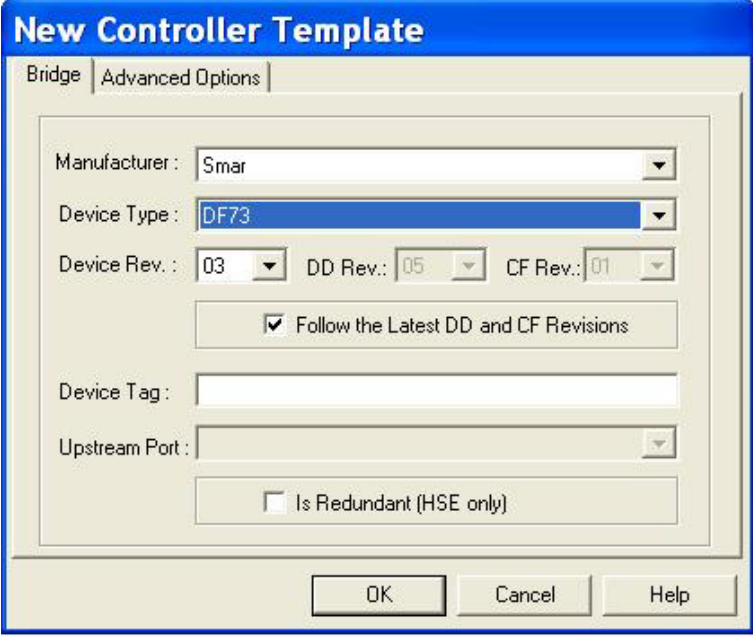
The 'New Bridge Template' dialog box features a blue title bar and two tabs: 'Bridge' and 'Advanced Options'. The 'Bridge' tab is active. It contains several input fields: 'Manufacturer' (dropdown menu with 'Smar' selected), 'Device Type' (dropdown menu with 'DF51' selected), 'Device Rev.' (dropdown menu with '07' selected), 'DD Rev.' (dropdown menu with '04' selected), and 'CF Rev.' (dropdown menu with '01' selected). Below these is a checkbox labeled 'Follow the Latest DD and CF Revisions' which is checked. There are also text input fields for 'Device Tag' and 'Upstream Port' (dropdown menu). At the bottom is a checkbox labeled 'Is Redundant (HSE only)' which is unchecked. At the very bottom are three buttons: 'OK', 'Cancel', and 'Help'.

Figure 5. 11 – Bridge Template option

The **Device Type** box displays the controllers, such as DF51, DF62, DF63, and other devices that are able to act as bridge.

Controller Template

Choosing the **Controller Template** option, it shows the box to select the device that will act as controller for the area, as shown in the next figure:



The 'New Controller Template' dialog box features a blue title bar and two tabs: 'Bridge' and 'Advanced Options'. The 'Bridge' tab is active. It contains several input fields: 'Manufacturer' (dropdown menu with 'Smar' selected), 'Device Type' (dropdown menu with 'DF73' selected), 'Device Rev.' (dropdown menu with '03' selected), 'DD Rev.' (dropdown menu with '05' selected), and 'CF Rev.' (dropdown menu with '01' selected). Below these is a checkbox labeled 'Follow the Latest DD and CF Revisions' which is checked. There are also text input fields for 'Device Tag' and 'Upstream Port' (dropdown menu). At the bottom is a checkbox labeled 'Is Redundant (HSE only)' which is unchecked. At the very bottom are three buttons: 'OK', 'Cancel', and 'Help'.

Figure 5. 10 –Controller Template option

The **Device Type** box displays the controllers, such as DF73, DF75, DF79, and others that act as controllers.

Section 6

ADDING FUNCTION BLOCKS

Introduction

The **DFI302** and fieldbus devices use function blocks to build strategies, such as PID, AI blocks, etc. This means that **Syscon** can be used to set up every part of the system - transmitters, positioners and controller - in a same language.

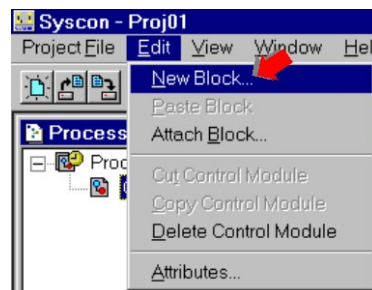
Using **Syscon** (for further details, see **Syscon** manual), follow the steps below to create a new block in the strategy.

Creating a New Block

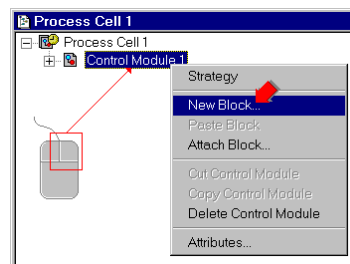
Create a new **Process Cell**, selecting the **Area** icon, and then **Edit** → **New Process Cell**.

Create a new **Control Module** opening the **Process Cell** window, and then selecting the **Process Cell** icon. Go to the **Edit** menu, and choose **New Control Module**.

And then, to create a new **Function Block**, select the **Control Module** icon on the **Process Cell** window, go to the **Edit** menu, and choose **New Block** (as shown in figure below).

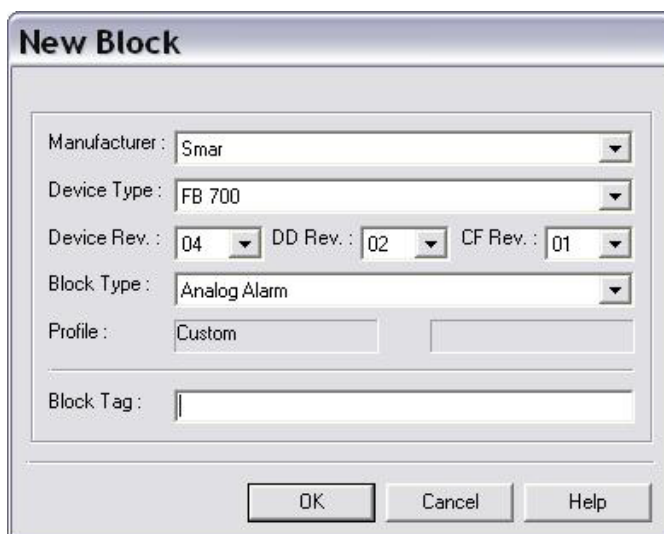


Another way to do this is right-clicking the **Control Module** icon to open its pop up menu. Click the **New Block** item. See the following figure:



The **New Block** dialog box will appear:

- Select a **Block Manufacturer** from the list.
- Select a **Device Type** provided by the selected manufacturer.
- Select the **Device Revision**.
- Select the **DD Revision**.
- Select the **CF Revision**.
- Select a **Block Type**.
- Type a related tag for the **Block**. Click **OK** to conclude the task.



New Block

Manufacturer : Smar

Device Type : FB 700

Device Rev. : 04 DD Rev. : 02 CF Rev. : 01

Block Type : Analog Alarm

Profile : Custom

Block Tag :

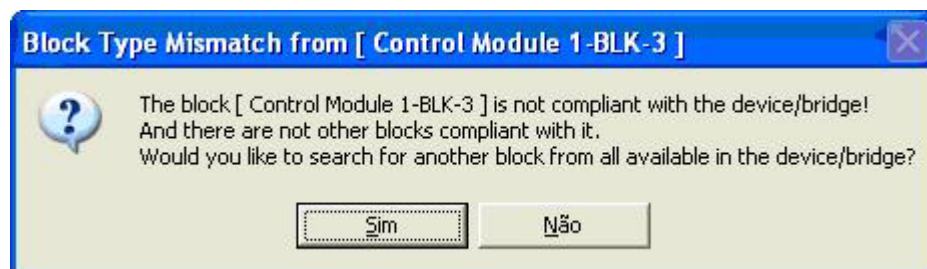
OK Cancel Help

If the user does not type a tag, the default tag **Block1** will be assigned.

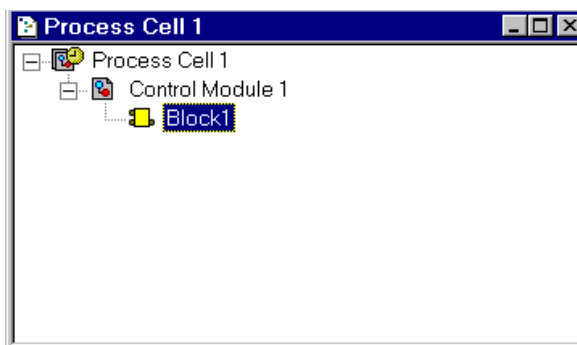
Syscon uses the latest revision of the **Device Revision** and the **DD Revision** as the default value for the new block. These values must be in according to the device used in the plant and its revision.

If a different Device Revision version is necessary, and the user already knows the corresponding value for the device, so the value can be changed in the window above.

Another way is to let the **Syscon** does the association automatically when the block is associated to the equipment. In this case, when a block is chosen in the strategy, and it is not supported by the equipment the **Syscon** shows the window below. It will allow the user to select another block which has equivalent functionalities of that does not exist. For further details refer to the **Syscon** manual.



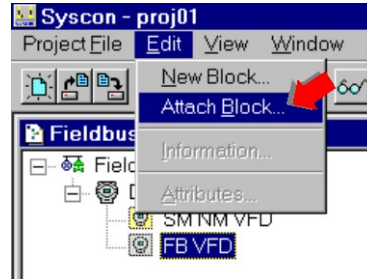
The **Process Cell** window will appear as shown in the figure below:



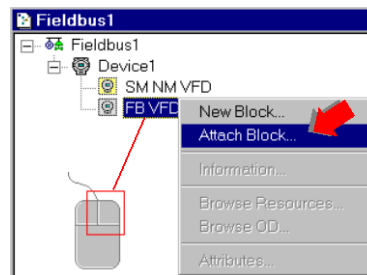
Attaching New Block

If one block was added to the **Logical Project**, this block can be attached to the device in the **Physical Project**:

For this on the **Fieldbus** window, select the **FB VFD** icon of the block that will be attached. Go to the **Edit** menu, and choose **Attach Block**. See the following figure:

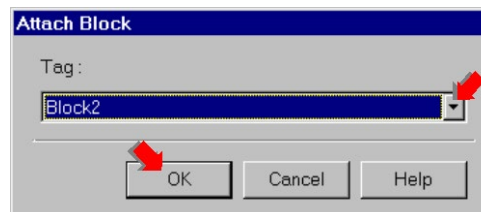


Or right-clicking the **FB VFD** a pop-up menu will appear. Choose **Attach Block**. See the following figure:



The **Attach Block** dialog box will appear.

Click the down arrow to select the block that will be attached. Click **OK** to add the block to the **Physical Project**:



Just click **Cancel** if you do not want to attach more blocks.

Section 7

ADDING LOGIC BY USING FLEXIBLE FUNCTION BLOCKS (FFB 1131)

Introduction

DFI302 has an advanced configuration resource by using Flexible Function Block (FFB 1131). Its purpose is to provide the connection between the ladder logic (usually used for discrete control strategies) and the continuous control strategies, configured through function blocks.

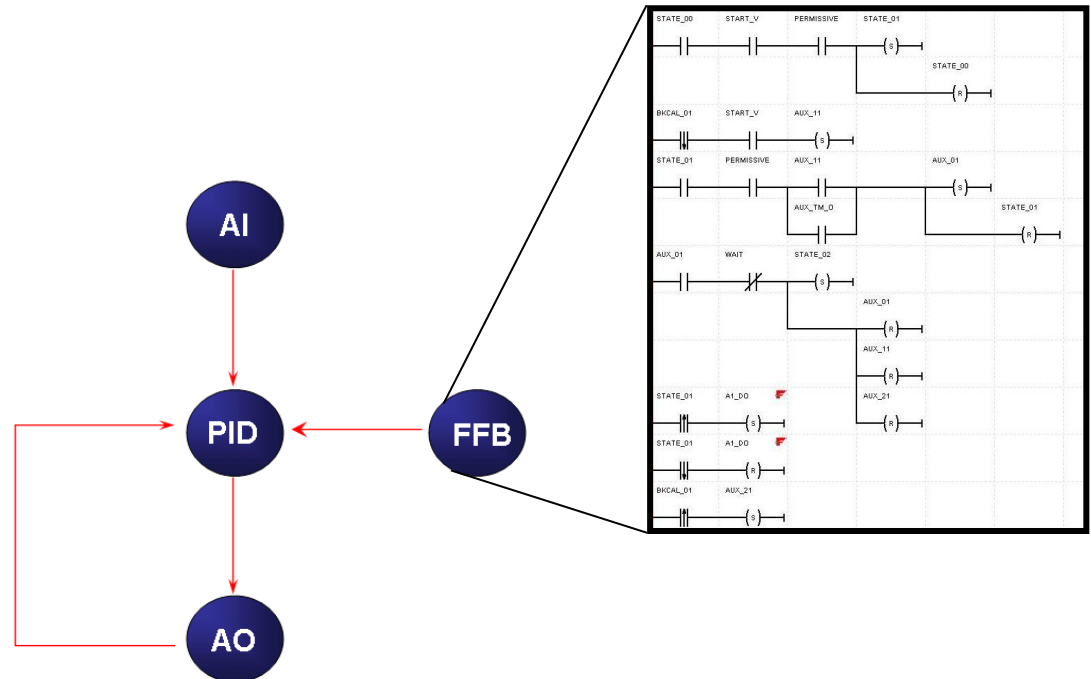


Figure 7.1 – Continuous and discrete control

NOTE

The DF62, DF63, DF73, DF75, DF79, DF81, DF89, DF95, and DF97 controllers support the FFB 1131. So, the example that will be showed can be applied to any of these controllers.

Area with FFB 1131

For the aimed application, there is an integration between 2 linking devices and also:

- 3 FOUNDATION™ fieldbus transmitters
- 10 FOUNDATION fieldbus valve positioners

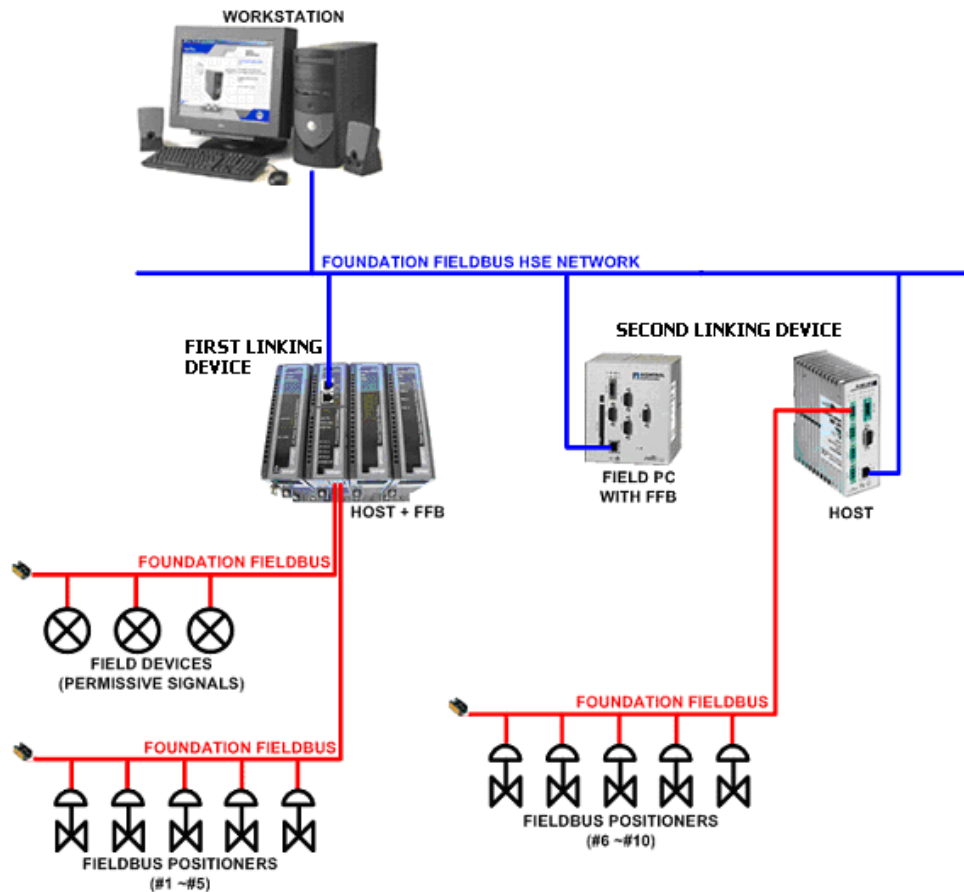


Figure 7.2 –Example of process


To create the control strategy using the FFB, it is necessary to open the **Syscon**. To create a new area on **Syscon**, choose **File → New**, or through the toolbar, choose **New** button . The dialog box shows the areas options. Select **Predefined Area** in the next figure:



Figure 7.3 –Area Options

The window with the options of areas templates will be showed next. The user should select one of the template types with FFB. The next figure shows the dialog box with one of the options selected.

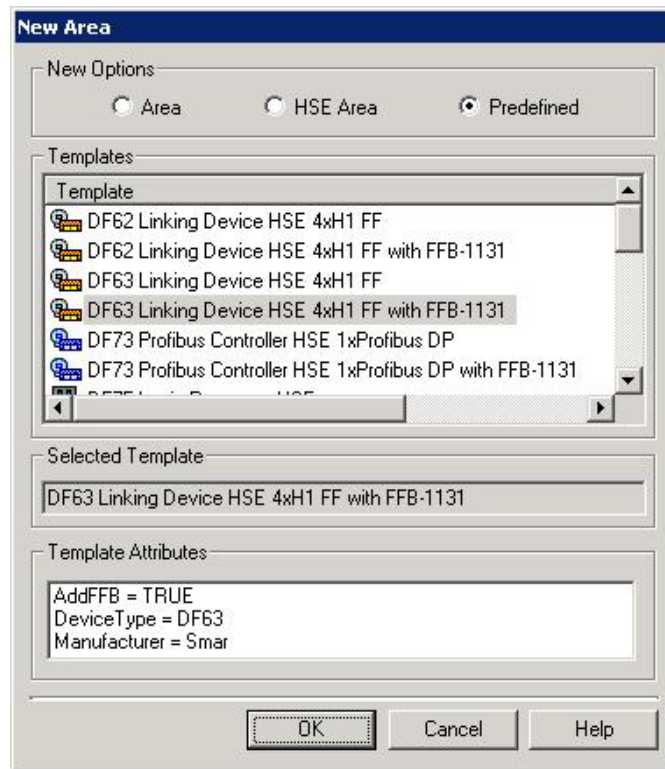


Figure 7.4 – Choosing the template using DF63

After choosing the area type, it opens a window to the user give a name to the new area. Type the name for the area in the **Area Name** box, and click **Ok**. Thus, two new windows will open in the **Syscon** automatically. See the following figure.

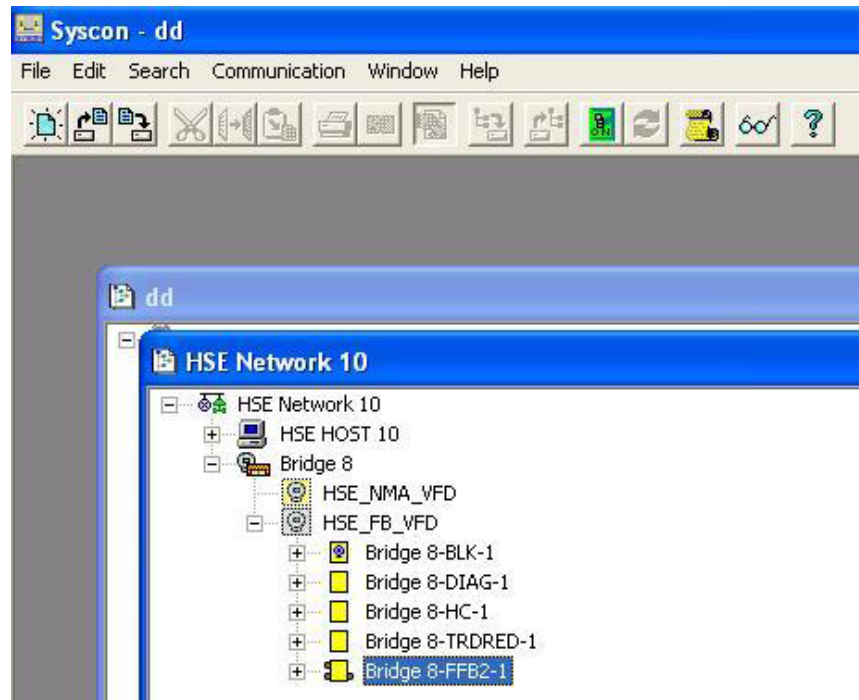


Figure 7.5 – Windows of Predefined Area

Arranging the Syscon windows

For better view of the area, choose **Window** option on the **Syscon** toolbar, and then **Tile**. The next figure shows the available windows, at this moment, for the area:

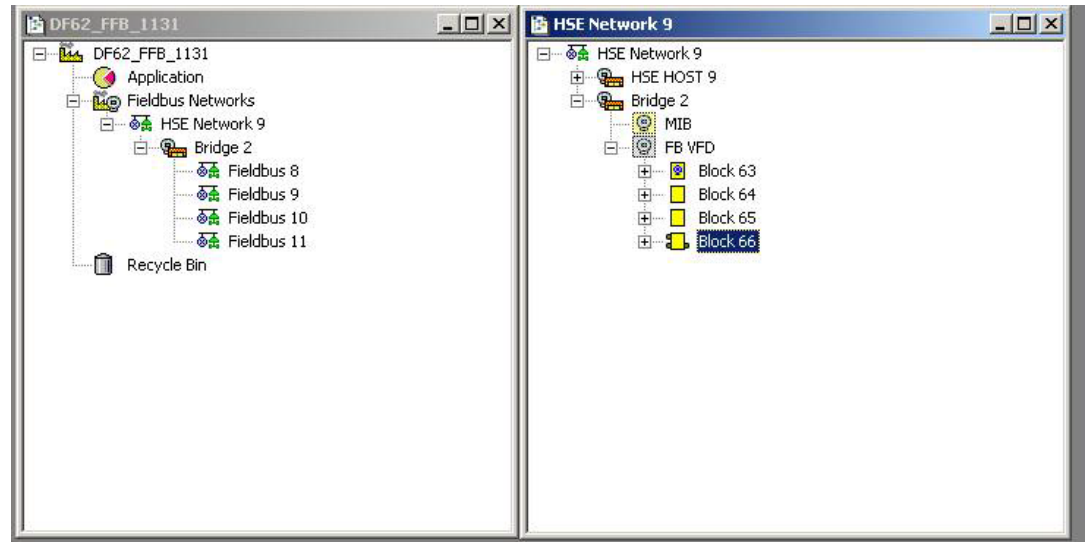



Figure 7.6 –Arranging the windows for the FFB area

Defining the FFB Parameters

To see the elements, such as blocks that were added to the area, click the **Details** button on the toolbar, . So, a description will appear just after the tag of each element.

In the figure below, click the **HSE Network 9*** window (this number is different if another area was created before. When a new HSE area is created, this number increases), and then, right-click the **Block 66**. The dialog box for the parameter definition will open:

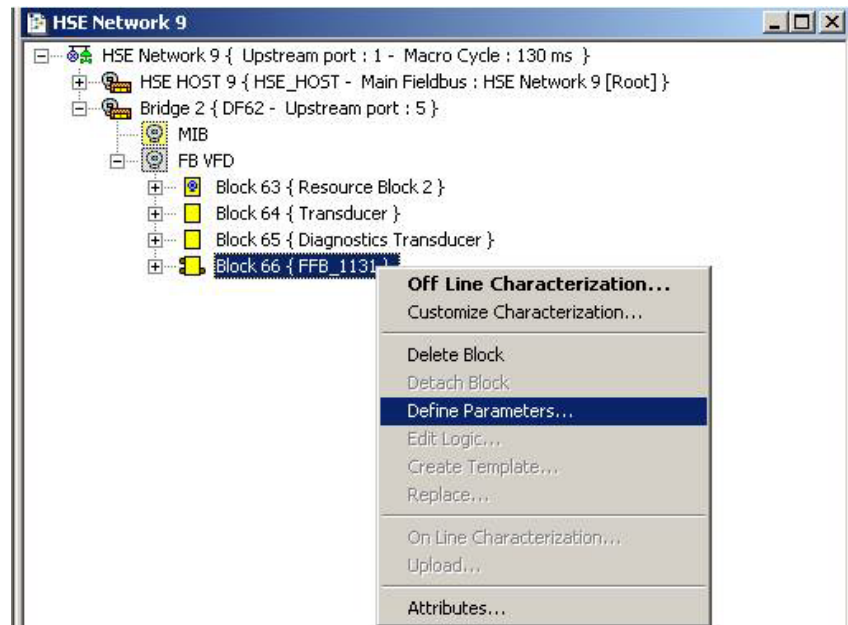


Figure 7.7 – Defining the FFB Parameters

To define which I/O types will be exchanged between the discrete and continuous logic, select **Define Parameters** option from the popup menu. The **FFB Parameters Definition** window will open as shown in the next figure:

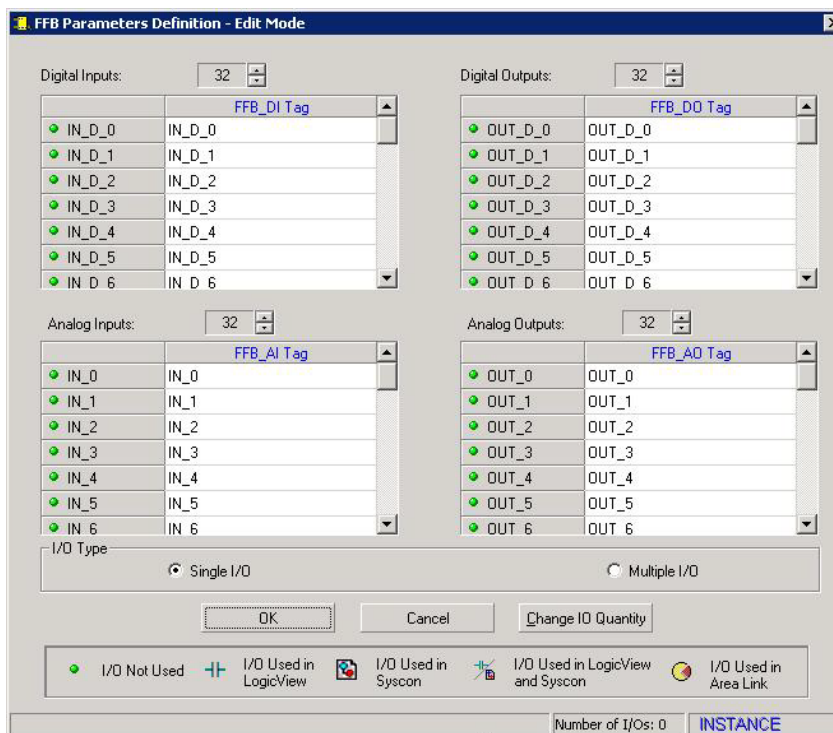


Figure 7.8 – Defining FFB Parameters

NOTE

From the 7.3 version of **SYSTEM302**, the FFB is automatically created, with the following number of parameters: 32 DO, 32 DI, 32 AO, 32 AI, 4 DO64, 4 DI64, 4 AI16, and 4 AO16, these last four types are created for the FFB2.

In the window above, the user can configure the number of analog and digital inputs and outputs: Analog Inputs, Analog Outputs, Digital Inputs, Digital Outputs, Analog Inputs16, Analog Outputs16, Digital Inputs64 or Digital Outputs64, respectively. After the user clicks **OK** the points DI, DO, AI, AO, DI64, DO64, AI16, and AO16 are generated. In **I/O Type** option are chosen how many and what parameters will be configured. In **Single I/O** option DI, DO, AI, and AO are configured. In **Multiple I/O** option DI64, DO64, AI16, and AO16 are configured. For further details about **FFB Parameters Definition** see the **Syscon's** manual.

To change the tags right-click the FFB block icon on **Syscon** (in the **Process Cell**, **Fieldbus** or **Strategy** window) and click **Edit User Parameter Tags**. The **User Parameter Tag** dialog box will open. For further information refer to **Syscon** manual.

If neither all I/O points are known at this moment, new I/O points may be defined later.

IMPORTANT

When a FFB block is used in a control strategy it is recommended to reserve extras parameters for future usage avoiding an impact of stopping the control during an incremental download. It will be necessary when a new strategy with new parameters were included. When new FFB parameters are added, as well as a change of parameter's name, the devices' DDs will be redefined, and this will demand a wider download, resulting in deleted links and deleted blocks, and the re-establishment of them. The utilization of extras parameters, which were previously defined, will not redefine new DDs and will demand only the establishment of new links which will use the reserved parameters.

However, from the 7.3 version of **SYSTEM302** when creating a new parameter, four other reserve parameters are automatically created for that same type.

Right-click the **FFB block** again. The dialog box to edit the logic will open. Now the user can select **Edit Logic** option to edit the internal logic of the function block.

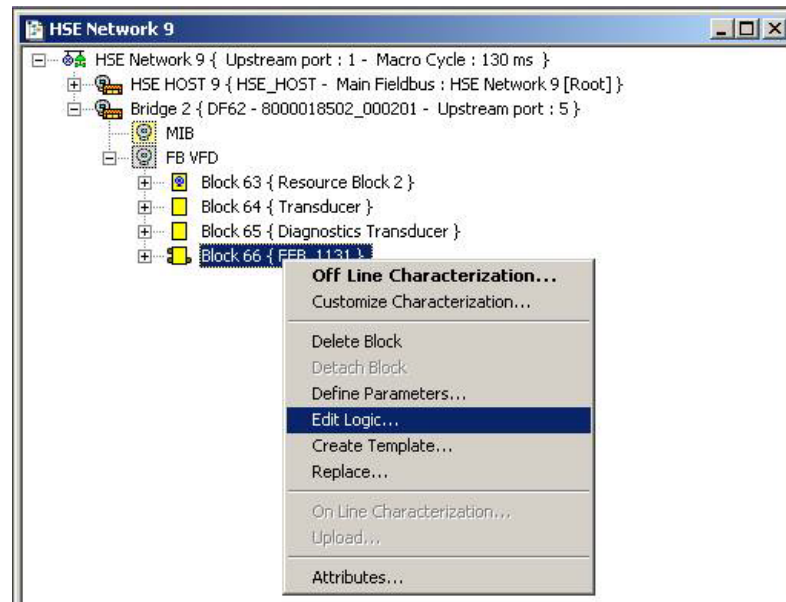


Figure 7.9 – Selecting the option to edit the logic

Just clicking this option, a new window will open. The configuration tool specialized in ladder logic will allow the user to build the discrete control. For further details about the ladder logic edition, refer to the **LogicView for FFB** manual.

ADDING I/O MODULES

Introduction

The DFI302 was specifically, and primarily, designed to operate with Fieldbus instruments. All common field device types are available in fieldbus versions. Therefore the amount of conventional I/O points required in a system is drastically reduced and will eventually be eliminated. However, since many applications require connection with devices that do not have Fieldbus communication, the DFI302 may also be fitted with conventional discrete and analogue I/O on an extended backplane. Each controller module can be fitted with an I/O-subsystem for up to 256 points or 1024 depending on the controller specification.

There are many types of modules available for the DFI302, designed to fit a broad range of applications in the automation and process control industry.

The following tables show the available I/O module types.

DISCRETE INPUTS		
MODEL	DESCRIPTION	I/O TYPE
DF11	2 Groups of 8 Digital Inputs 24 Vdc - Sink	16-discrete input
DF12	2 Groups of 8 Digital Inputs 48 Vdc - Sink	16- discrete input
DF13	2 Groups of 8 Digital Inputs 60 Vdc - Sink	16- discrete input
DF14	2 Groups of 8 Digital Inputs 125 Vdc - Sink	16- discrete input
DF15	2 Groups of 8 Digital Inputs 24 Vdc - Source	16- discrete input
DF16	2 Groups of 4 Digital Inputs 120 Vac	8- discrete input
DF17	2 Groups of 4 Digital Inputs 240 Vac	8- discrete input
DF18	2 Groups of 8 Digital Inputs 120 Vac	16- discrete input
DF19	2 Groups of 8 Digital Inputs 240 Vac	16- discrete input
DF20	1 Group of 8 Push-Button Switches	8- discrete input

DISCRETE OUTPUTS		
MODEL	DESCRIPTION	I/O TYPE
DF21	1 Group of 16 Digital Outputs 24 Vdc - Sink	16- discrete output
DF22	2 Groups of 8 Digital Outputs 24 Vdc - Source	16- discrete output
DF23	2 Groups of 4 Digital Outputs 120/240 Vac - Triac	8- discrete output
DF24	2 Groups of 8 Digital Outputs 120/240 Vac - Triac	16- discrete output
DF25	2 Groups of 4 NO Relay Outputs	8- discrete output
DF26	2 Groups of 4 NC Relay Outputs	8- discrete output
DF27	1 Group of 4 NO and 4 NC Relay Outputs	8- discrete output
DF28	2 Groups of 8 NO Relay Outputs without RC Protection	16- discrete output
DF29	2 Groups of 4 NO Relay Outputs without RC protection	8- discrete output
DF30	2 Groups of 4 NC Relay Outputs without RC protection	8- discrete output
DF31	1 Group of 4 NO and 4 NC Relay Outputs without RC protection	8- discrete output
DF69	2 Groups of 8 NO Relay Outputs	16-discrete output
DF71	2 Groups of 4 NO Relay Outputs without RC protection - Max 10 mA	8-discrete output
DF72	2 Groups of 4 NC Relay Outputs without RC protection - Max 10 mA	8-discrete output

COMBINED DISCRETE INPUTS AND OUTPUTS		
MODEL	DESCRIPTION	I/O TYPE
DF32	1 Group of 8 24 Vdc Inputs and 1 Group of 4 NO Relay	8- discrete input/4- discrete output
DF33	1 Group of 8 48 Vdc Inputs and 1 Group of 4 NO Relay	8- discrete input/4- discrete output
DF34	1 Group of 8 60 Vdc Inputs and 1 Group of 4 NO Relay	8- discrete input/4- discrete output
DF35	1 Group of 8 24 Vdc Inputs and 1 Group of 4 NC Relay	8- discrete input/4- discrete output
DF36	1 Group of 8 48 Vdc Inputs and 1 Group of 4 NC Relay	8- discrete input/4- discrete output
DF37	1 Group of 8 60 Vdc Inputs and 1 Group of 4 NC Relay	8- discrete input/4- discrete output
DF38	1 Group of 8 24 Vdc Inputs and 1 Group of 2 NO and 2 NC Relay	8- discrete input/4- discrete output
DF39	1 Group of 8 48 Vdc Inputs and 1 Group of 2 NO and 2 NC Relay	8- discrete input/4- discrete output
DF40	1 Group of 8 60 Vdc Inputs and 1 Group of 2 NO and 2 NC Relay	8- discrete input/4- discrete output

PULSE INPUTS		
MODEL	DESCRIPTION	I/O TYPE
DF41	2 Groups of 8 Low Frequency (0 - 100 Hz) 24 Vdc Pulse Inputs	16-pulse input
DF42	2 Groups of 8 High Frequency (0 - 10 KHz) 24 Vdc Pulse Inputs	16-pulse input
DF67	2 Groups of 8 High Frequency (0 - 10 KHz) AC Pulse Inputs	16-pulse input

ANALOG INPUTS		
MODEL	DESCRIPTION	I/O TYPE
DF44	1 Group of 8 Voltage/Current Analogue Inputs with Internal Shunt Resistors	8-analog input
DF57	1 Group of 8 Voltage/Current Differential Analogue Inputs with Internal Shunt Resistors	8-analog input
DF45	1 Group of 8 Low Signal Analogue Inputs for TC, RTD, mV and Ohm	8-temperature input

ANALOG OUTPUTS		
MODEL	DESCRIPTION	I/O TYPE
DF46	1 Group of 4 Voltage/Current Analogue Outputs	4-analog output

HART MODULES		
MODEL	DESCRIPTION	I/O TYPE
DF116	8 analog inputs with HART master interface (4-20 mA)	8 analog inputs
DF117	8 analog outputs with HART master interface (4-20 mA)	8 analog outputs

INTERFACES FOR I/O MODULES AND THEIR ACCESSORIES*	
CODE	DESCRIPTION
ITF - 005AC1	Interface for 16 points of 120 Vac input compatible with DF15.
ITF - 005AC2	Interface for 16 points of 240 Vac input compatible with DF15.
ITF - 001	Interface for 16 points of 24 Vdc input compatible with DF11.
ITF - 005DC	Interface for 16 points of 24 Vdc input compatible with DF15.
ITF - 101	Interface for 16 points digital output for relay with NA and NC contact compatible DF21.
ITF - 101FAC	Interface for 16 points digital output for relay with NA and NC contact with fuse for AC load compatible with DF21.
ITF - 101FDC	Interface for 16 points digital output for relay with NA and NC contact with fuse for DC load compatible with DF21.
ITF - 102	Interface for 16 points digital output for relay with NA and NC contact compatible DF22.
ITF - 102FAC	Interface for 16 points digital output for relay with NA and NC contact with fuse for AC load compatible with DF22.
ITF - 102FDC	Interface for 16 points digital output relay for relay with NA and NC contact with fuse for DC load compatible with DF22.
ITF - 120AC	Interface for 8 points digital output for AC load relay compatible with DF25.
ITF - 120DC	Interface for 8 points digital output for DC load relay compatible with DF25.
ITF - 123-7	Interface for 16 points digital output for relay with NA and NC contact compatible for AC load with DF69.
ITF - 1237FAC	Interface for 16 points digital output for relay with NA and NC contact with fuse for AC load compatible with DF28 or DF69.
ITF - 1237FDC	Interface for 16 points digital output for relay with NA and NC contact with fuse for DC load compatible with DF28 or DF69.
ITF - 304	Interface for 16-point AC pulse input compatible with DF67.
ITF - 401	Interface for 8-point analog input compatible with DF44 and DF57.
ITF - 402	Interface for 8-point analog input compatible with DF45.
ITF - 501	Interface for 8-point analog output without fuse compatible with DF46.
ITF - QDA-AC	Power distribution interface for 10 points 110/240 Vac @ 2A per point.
ITF - QDA-DC	Power distribution interface for 10 points 24 Vdc @ 2A per point.
ITF - C-10	Connection cable between DFI302 modules and ITF interfaces - 1.0 m.
ITF - C-15	Connection cable between DFI302 modules and ITF interfaces - 1.5 m.
ITF - C-20	Connection cable between DFI302 modules and ITF interfaces - 2.0 m.
ITF - C-25	Connection cable between DFI302 modules and ITF interfaces - 2.5 m.
ITF - C-30	Connection cable between DFI302 modules and ITF interfaces - 3.0m.
ITF - C-35	Connection cable between DFI302 modules and ITF interfaces - 3.5 m.
ITF - C-40	Connection cable between DFI302 modules and ITF interfaces - 4.0 m.
ITF - C-45	Connection cable between DFI302 modules and ITF interfaces - 4.5 m.
ITF - C-50	Connection cable between DFI302 modules and ITF interfaces - 5.0 m.

*For further information, please refer to the Panel Interfaces manual.

MODEL	DESCRIPTION
DF0	Blind module to fill empty slots
DF1A	Rack with 4 slots – support to shielded flat cable
DF2	Terminator for last the rack – right side
DF3, DF4A~DF7A	Flat cableS to connect 2 racks
DF9	Support for a single module
DF54	Twisted pair cable 100 Base-TX
DF55	Twisted pair cable 100 Base-TX – cross cable – length 2m
DF59	Cable RJ12 used to connect DF51 and DF58
DF68	Cable to connect redundant CPUs
DF76	Cable to connect coprocessors
DF78	Rack with 4 slots – It supports Hot Swap of CPUs and redundant I/O access
DF82	Synchronism cable to connect redundant controllers – length 500 mm
DF83	Synchronism cable to connect redundant controllers – length 1800 mm
DF84	IMB Soft Starter

DF90	IMB power cable
DF91	Lateral adapter
DF92	Rack with 4 slots for redundant CPUs, hot swap and diagnostic support
DF93	Rack with 4 slots, with diagnostic
DF96	Terminator for the last rack – left side
DF101	Flat cable to connect racks by left side – length 70 cm
DF102	Flat cable to connect racks by right side – length 65 cm
DF103	Flat cable to connect racks by right side – length 81 cm
DF104	Flat cable to connect racks by right side – length 98 cm
DF105	Flat cable to connect racks by right side – length 115 cm

Steps to configure I/O Modules

The first steps to configure DFI302 with I/O modules, need the knowledge on “How to Add a Function Block” using **Syscon** (the configuration tool). See the section “Adding Function Blocks”, for further information.

Add one Resource Block, one Hardware Configuration Transducer (HC) and one or more Temperature Transducers (when using temperature modules).

After the resource and these transducers blocks, the user can add the other blocks (AI, MAI, AO, MAO, DI, MDI, DO, MDO).

NOTES

1. From the 7.3 version of SYSTEM302 the function blocks (HCT, Temperature Transducers, AI, MAI, AO, MAO, DI, MDI, DO and MDO) can be used combined with FFB 1131 blocks. However, each output or temperature point must not be used simultaneously for function blocks and FFB 1131 blocks.
2. Using ladder logic (FFB 1131), for a better monitoring of the functional state of each used I/O module is recommended to use the **STATUS** block in the logic. Thus the system can be advised if some I/O module have a failure. So that is easier to find a damaged module. Insert and configure this block according to the **LogicView for FFB** manual.

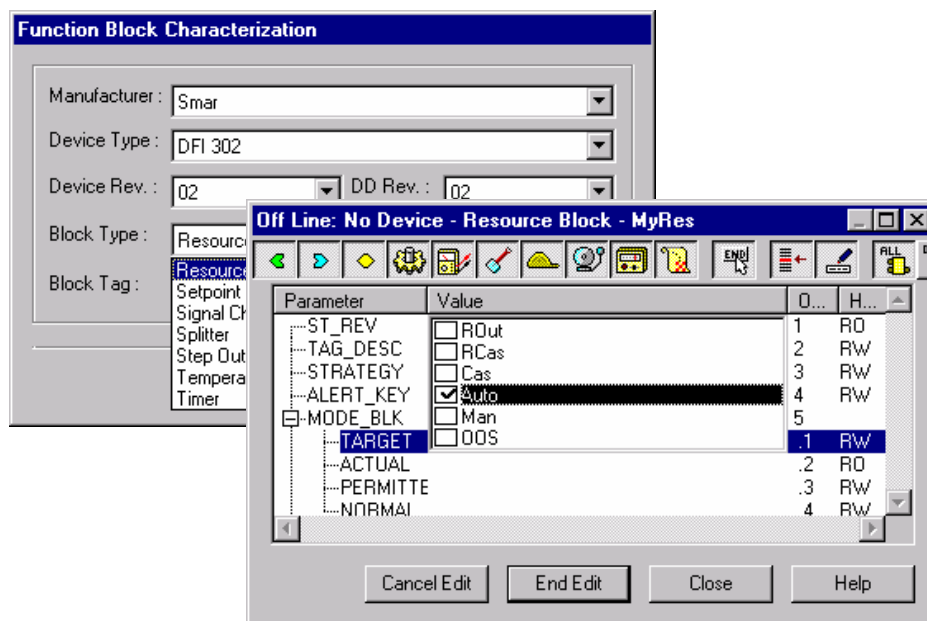
The order of the Resource, Transducers and block creation is very important because when **Syscon** does the configuration download, a lot of consistency checks will be done inside DFI302.

For instance, an AI block will not accept a channel configuration if the desired pointed hardware was not declared before in the Hardware Configuration Transducer.

A complete documentation about FOUNDATION fieldbus blocks and its parameters could be found under Function Blocks Manual inside the **SYSTEM302** documentation folder. The following steps are more related with details about DFI302, and the complete description about blocks will not be found here.

RES – Resource Block

This function block has already been instantiated in the device. So, it is necessary set the MODE_BLK.TARGET parameter to AUTO.



HCT – Hardware Configuration Transducer

This transducer configures the module type for each slot in the DFI302. The execution method of this transducer block will write to all output modules and it will read all the input modules.

If any I/O module has failed in this scan, it will be indicated in BLOCK_ERR as well in the MODULE_STATUS_x. It makes easy to find the module or even the sensor in failure. This function block has already been instantiated in the device, so set the MODE_BLK parameter to AUTO and fill IO_TYPE_Rx parameters with its respective module that has been used.

PARAMETER	VALID RANGE/ OPTIONS	DEFAULT VALUE	DESCRIPTION
ST_REV		0	
TAG_DESC		Spaces	
STRATEGY		0	
ALERT_KEY	1 to 255	0	
MODE_BLK		O/S	See Mode Parameter
BLOCK_ERR			
REMOTE_IO		Remote I/O Master	Reserved
IO_TYPE_R0		0	Select module type for the rack 0
IO_TYPE_R1		0	Select module type for the rack 1
IO_TYPE_R2		0	Select module type for the rack 2
IO_TYPE_R3		0	Select module type for the rack 3
IO_TYPE_R4		0	Select module type for the rack 4
IO_TYPE_R5		0	Select module type for the rack 5
IO_TYPE_R6		0	Select module type for the rack 6
IO_TYPE_R7		0	Select module type for the rack 7
IO_TYPE_R8		0	Select module type for the rack 8
IO_TYPE_R9		0	Select module type for the rack 9

PARAMETER	VALID RANGE/ OPTIONS	DEFAULT VALUE	DESCRIPTION
IO_TYPE_R10		0	Select module type for the rack 10
IO_TYPE_R11		0	Select module type for the rack 11
IO_TYPE_R12		0	Select module type for the rack 12
IO_TYPE_R13		0	Select module type for the rack 13
IO_TYPE_R14		0	Select module type for the rack 14
MODULE_STATUS_R0_3			Status of modules in rack 0-3.
MODULE_STATUS_R4_7			Status of modules in rack 4-7.
MODULE_STATUS_R8_11			Status of modules in rack 8-11.
MODULE_STATUS_R12_14			Status of modules in rack 12-14.
UPDATE_EVT			This alert is generated by any change to the static data.
BLOCK_ALM			The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

TEMP – Temperature Transducer

This is the transducer block for the module DF45, an eight low signal input module for RTD, TC, Ohm.

When using this module, the TEMP Transducer is necessary and must be added to **Syscon** Configuration, just before the Function Block, which will provide the interface with the I/O module. Therefore, create this block, set the MODE_BLK parameter to AUTO and fill parameters with range, sensor, etc, that will be used by the Temperature Module.

PARAMETER	VALID RANGE/ OPTIONS	DEFAULT VALUE	DESCRIPTION
ST_REV		0	
TAG_DESC		Spaces	
STRATEGY		0	
ALERT_KEY	1 to 255	0	
MODE_BLK		O/S	See Mode Parameter
BLOCK_ERR			
CHANNEL			The rack and slot number of the associated DF45 module coded as RRSXX.
TEMP_0			Temperature of point 0.
TEMP_1			Temperature of point 1.
TEMP_2			Temperature of point 2.
TEMP_3			Temperature of point 3.
TEMP_4			Temperature of point 4.
TEMP_5			Temperature of point 5.
TEMP_6			Temperature of point 6.
TEMP_7			Temperature of point 7.
VALUE_RANGE_0		0-100%	If it is connected to AI block, it is a copy of XD_SCALE. Otherwise the user can write in this scaling parameter.
SENSOR_CONNECTION_0	1 : differential 2 : 2-wire 3 : 3-wire	3	Connection of the sensor 0.
SENSOR_TYPE_0	See table below	Pt 100 IEC	Type of sensor 0.

PARAMETER	VALID RANGE/ OPTIONS	DEFAULT VALUE	DESCRIPTION
VALUE_RANGE_1		0-100%	If it is connected to AI block, it is a copy of XD_SCALE. Otherwise the user can write in this scaling parameter.
SENSOR_CONNECTION_1	1 : differential 2 : 2-wire 3 : 3-wire	3	Connection of the sensor 1.
SENSOR_TYPE_1	See table below	Pt 100 IEC	Type of sensor 1.
VALUE_RANGE_2		0-100%	If it is connected to AI block, it is a copy of XD_SCALE. Otherwise the user can write in this scaling parameter.
SENSOR_CONNECTION_2	1 : differential 2 : 2-wire 3 : 3-wire	3	Connection of the sensor 2.
SENSOR_TYPE_2	See table below	Pt 100 IEC	Type of sensor 2.
VALUE_RANGE_3		0-100%	If it is connected to AI block, it is a copy of XD_SCALE. Otherwise the user can write in this scaling parameter.
SENSOR_CONNECTION_3	1 : differential 2 : 2-wire 3 : 3-wire	3	Connection of the sensor 3.
SENSOR_TYPE_3	See table below	Pt 100 IEC	Type of sensor 3.
VALUE_RANGE_4		0-100%	If it is connected to AI block, it is a copy of XD_SCALE. Otherwise the user can write in this scaling parameter.
SENSOR_CONNECTION_4	1 : differential 2 : 2-wire 3 : 3-wire	3	Connection of the sensor 4.
SENSOR_TYPE_4	See table below	Pt 100 IEC	Type of sensor 4.
VALUE_RANGE_5		0-100%	If it is connected to AI block, it is a copy of XD_SCALE. Otherwise the user can write in this scaling parameter.
SENSOR_CONNECTION_5	1 : differential 2 : 2-wire 3 : 3-wire	3	Connection of the sensor 5.
SENSOR_TYPE_5	See table below	Pt 100 IEC	Type of sensor 5.
VALUE_RANGE_6		0-100%	If it is connected to AI block, it is a copy of XD_SCALE. Otherwise the user can write in this scaling parameter.
SENSOR_CONNECTION_6	1 : differential 2 : 2-wire 3 : 3-wire	3	Connection of the sensor 6.
SENSOR_TYPE_6	See table below	Pt 100 IEC	Type of sensor 6.
VALUE_RANGE_7		0-100%	If it is connected to AI block, it is a copy of XD_SCALE. Otherwise the user can write in this scaling parameter.
SENSOR_CONNECTION_7	1 : differential 2 : 2-wire 3 : 3-wire	3	Connection of the sensor 7.
SENSOR_TYPE_7	See table below	Pt 100 IEC	Type of sensor 7.
UPDATE_EVT			This alert is generated by any change to the static data.
BLOCK_ALM			The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

TBH – RIO HART Transducer Block

The **TBH** block (RIO HART Transducer Block) represents the HART device in the system. Through it the user can access any variable of the device.

This block contains parameters for the process to be used in the control strategy and ladder logic, identification parameters, Burst and diagnosis, as well as Bypass parameters (**HART_CMD**, **HART_RESP** and **HART_COM_STAT**) that are used by the configuration and asset management tools for transmission and reception of HART messages.

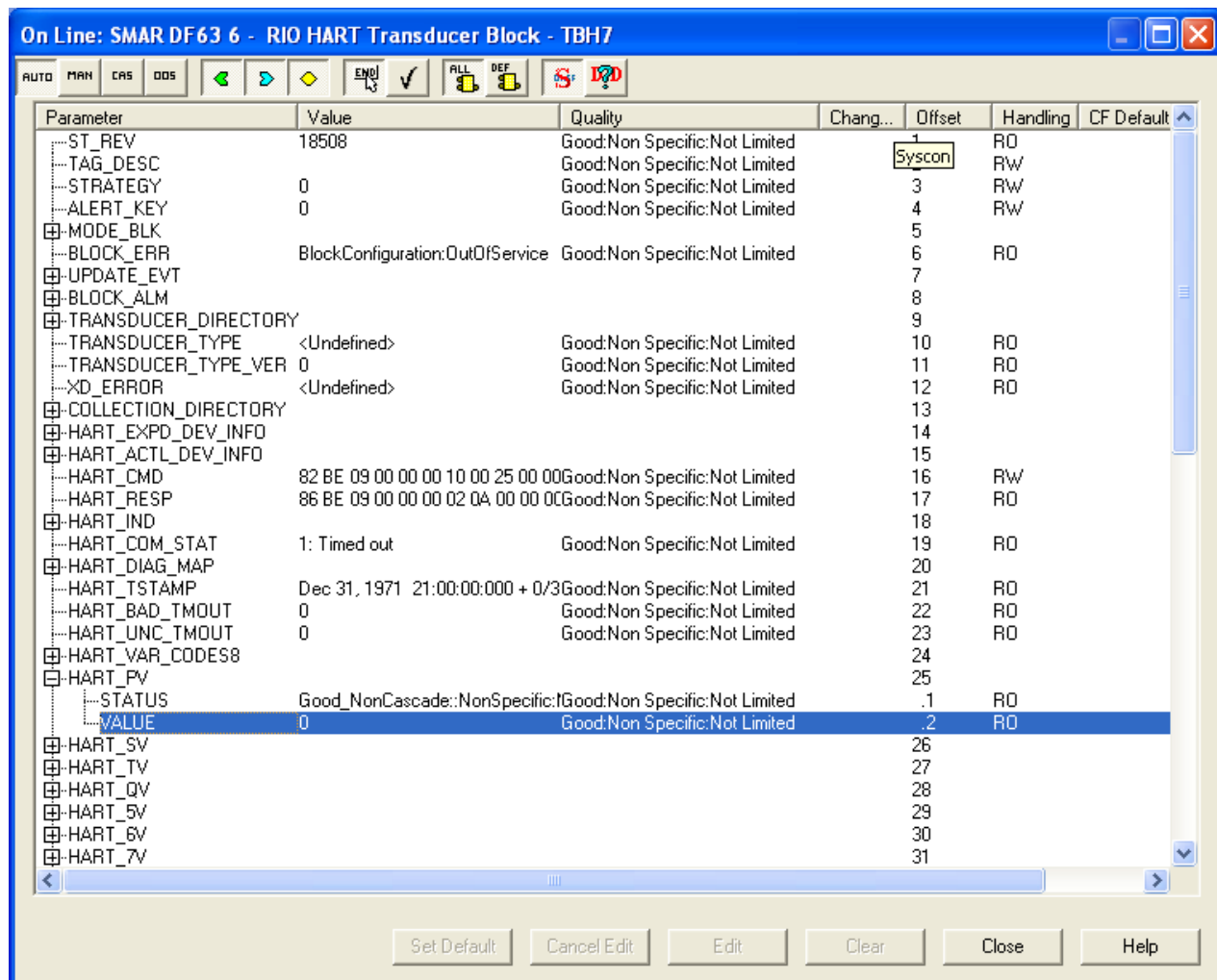
For each HART device installed must exist one corresponding **TBH** block on the system. The association of this block to the physical device must be done through the **RRSGP** parameter, following the rule **RRSGP** where **RR**: rack, **S**: slot, **G**: group (0 - 7) and **P**: point (9). Examples:

209 – Rack 0, Slot 2, group 0 and point 9

12319 – Rack 12, slot 3, group 1 and point 9

The group represents the HART device. The DF116/DF117 modules support up to 8 HART devices.

The point has to be configured with 9 because it represents the access to all available variables in the HART device: **HART_PV**, **HART_SV**, **HART_TV**, **HART_QV**, **HART_5V**, **HART_6V**, **HART_7V**, **HART_8V** and **PRIMARY_VALUE**.



TBH Block (RIO HART Transducer Block)

Configuration of the dynamic variables in the TBH block

The **TBH** block is flexible and allows the user to configure up to 8 digital variables to be dynamically read from the HART device. The configuration has to be done in the **HART_VAR_CODES8** parameter using indexes. The value and status of the variable for the index are shown on the corresponding parameter. See the following table.

HART_VAR_CODES8
[1] – HART_PV
[2] – HART_SV
[3] – HART_TV
[4] – HART_QV
[5] – HART_5V
[6] – HART_6V
[7] – HART_7V
[8] – HART_8V

Table – HART_VAR_CODES8[n]

The index configured in the **HART_VAR_CODES8[n]** parameter defines which variable has to be read from the device and also the HART command used by the DF116/DF117 module to read the variable. See the following table.

HART_VAR_CODES8[1..8] = 255	HART_VAR_CODES8[n] = (0 - 254)
<p>Read the variable by the HART #3 command.</p> <p>The #3 command is flexible and can return information with up to 4 process variables (PV, SV, TV, and QV). The number of variables returned by this command is determined by the device manufacturer according to their functionality.</p> <p>If the HART_VAR_CODE8[n] parameter is configured with the value 255, but there is not corresponding variable on #3 command, the default value should appear in the associated parameter.</p>	<p>HART command used to read the variable in question depends on the version of the HART device.</p> <p>HART 5: #33 command</p> <p>HART 6 and 7: #9 command</p> <p>Both commands return information from variables whose indexes (Device Variable Code) are defined on HART request message. These commands are flexible and accept up to 4 indexes, except the #9 command of HART 7 which accepts up to 8 indexes. For a list of indexes supported by the device, and the associated variables, is necessary to consult the manual or the manufacturer of the device.</p> <p>The list of variables of Smar's devices can be obtained in the "Indexes of Smar HART devices variables" topic on this document.</p> <p>If the index configured in the HART_VAR_CODES8[n] parameter does not exist in the device, the corresponding parameter and other parameters whose indexes are part of the same request message should appear with the default value.</p>

HART commands

NOTE

HART command is a data structure used by the HART protocol to group variables and features of the device.

Each command has an identification number, some commands, and therefore their IDs, are pre-defined by the HART specification. Other commands can be defined by the device manufacturer according to its functionality.

The composition/structure of HART and FOUNDATION fieldbus protocols differ in some points. Therefore, to ensure the integration of HART devices in FOUNDATION fieldbus systems in a transparent manner, some adjustments and conversions are required.

One of the necessary adjustments is related to the status of parameters associated to **HART_VAR_CODES8** (see table HART_VAR_CODES8[n]). In the HART protocol only the **#9**

command returns the variable status on its response.

Thus the parameters status whose variables are read by the # 3 or # 33 commands are obtained by interpreting the **DEVICE_STATUS** byte present in responses from all messages from the HART device. The following table shows how the **DEVICE_STATUS** bits are interpreted.

STATUS BIT	DEVICE_STATUS	HART STATUS
2	Loop Current Saturated	PoorAccuracy:NotLimited
3	Loop Current Fixed	ManualFixed:Constant
1	Non-Primary Variable Out of Limits	PoorAccuracy:NotLimited
0	Primary Variable Out of Limits	Bad:NotLimited
7	Device Malfunction	Bad:Constant
--	-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --	Good:Constant

The HART commands (**#3**, **#9** and **#33**) also return the measurement unit of the variable that can be seen in the **VAR_UNITS9[n]** parameter in the same position of the **HART_VAR_CODES8** parameter where the index variable was configured.

Conversion of HART Status to FOUNDATION fieldbus

The status from the HART device is converted to the corresponding FOUNDATION fieldbus status to fill the **TBH** block parameters that have status. See the following table:

HART STATUS	FOUNDATION FIELDBUS STATUS
Good:Constant	GoodNonCascade:GoodNCNonSpecific:Constant
Good:Low Limited	GoodNonCascade:GoodNCNonSpecific:LowLimited
Good:High Limited	GoodNonCascade:GoodNCNonSpecific:HighLimited
Good:Not Limited	GoodNonCascade:GoodNCNonSpecific:NotLimited
Poor Accuracy:Constant	Uncertain:UncertainNonSpecific:Constant
Poor Accuracy:Low Limited	Uncertain:UncertainNonSpecific:LowLimited
Poor Accuracy:High Limited	Uncertain:UncertainNonSpecific:HighLimited
Poor Accuracy: Not Limited	Uncertain:UncertainNonSpecific:NotLimited
Manual Fixed:Constant	GoodNonCascade:GoodNCNonSpecific:Constant
Manual Fixed:Low Limited	GoodNonCascade:GoodNCNonSpecific:LowLimited
Manual Fixed:High Limited	GoodNonCascade:GoodNCNonSpecific:HighLimited
Manual Fixed:Not Limited	GoodNonCascade:GoodNCNonSpecific:NotLimited
Bad:Constant	Bad:BadNonSpecific:Constant
Bad:Low Limited	Bad:BadNonSpecific:LowLimited
Bad:High Limited	Bad:BadNonSpecific:HighLimited
Bad : Not Limited	Bad:BadNonSpecific:NotLimited

Access to current analog signal (4-20 mA)

Each one of the 8 channels of DF116 and DF117 modules has analog circuit that allows the 4-20 mA current signal to be accessed in parallel to HART communication, without disturbing the communication signal. For it is essential that the physical installation of the module is correct.

The access to the input current (DF116) or to the output current (DF117) of the channel is done in the FOUNDATION fieldbus system through the **PRIMARY_VALUE** parameter of **TBH** block. In each channel must be installed only one HART device and its address must be 0. The multidrop mode is not allowed. The **PRIMARY_VALUE** parameter is associated to the current signal of the channel

where the device is installed.

The status of **PRIMARY_VALUE** parameter is in compliance with the Namur Standard adapted to the standard of FOUNDATION fieldbus status as in the following table.

CURRENT	FOUNDATION FIELDBUS STATUS
3.8 mA < current < 20.5 mA	GoodNonCascade:GoodNCNonSpecific:Constant
3.6 mA < current ≤ 3.8 mA	Uncertain:UncertainNonSpecific:LowLimited
20.5 mA ≤ current < 21.0 mA	Uncertain:UncertainNonSpecific:HighLimited
Current ≤ 3.6 mA	Bad:NonSpecific:Constant
Current ≥ 21.0 mA	Bad:NonSpecific:Constant

The unit of **PRIMARY_VALUE** parameter is available in the **VAR_UNITS9[9]** parameter of the **TBH** block.

The DF117 module has a safe mode configured by the **SAFE_BEHAVIOR** parameter of the **TBH** block, with the following values: 3.6 mA and 21 mA. These values represent the current value that will be controlled by the DF117 when, due to some fault in the controller (DF75), there is not data exchange between them.

Default value of HART parameters of TBH block

In situations where it could not read the required information from the HART device by the HART parameter of **TBH** block, this will appear with the default value. The most common conditions for this to happen are:

- HART device has not yet been identified (startup, wrong address, wrong installation);
- No HART device installed in the channel indicated by the **RRSGP** parameter of the block;
- Parameter does not exist on HART device. This depends on the HART protocol version of the device and is also mandatory implementation of the HART command that reads the parameter in question;
- Invalid index in **HART_VAR_CODES8**.

The table below shows the default value of some HART parameters of the block

PARAMETER	DEFAULT VALUE
HART_PV.VALUE	NAN (Not a number)
HART_PV.STATUS	Bad: Constant
HART_SV.VALUE	NAN (Not a number)
HART_SV.STATUS	Bad: Constant
HART_TV.VALUE	NAN (Not a number)
HART_TV.STATUS	Bad: Constant
HART_5V.VALUE	NAN (Not a number)
HART_5V.STATUS	Bad: Constant
HART_6V.VALUE	NAN (Not a number)
HART_6V.STATUS	Bad: Constant
HART_7V.VALUE	NAN (Not a number)
HART_7V.STATUS	Bad: Constant
HART_8V.VALUE	NAN (Not a number)
HART_8V.STATUS	Bad: Constant
VARUNITS9.[1]	0 (None Units)
VARUNITS9.[2]	0 (None Units)
VARUNITS9.[3]	0 (None Units)
VARUNITS9.[4]	0 (None Units)
VARUNITS9.[5]	0 (None Units)

VARUNITS9.[6]	0 (None Units)
VARUNITS9.[7]	0 (None Units)
VARUNITS9.[8]	0 (None Units)

Input/output parameters (link) of TBH block

The **TBH** block has 9 parameters that can be linked to other blocks to be used in the control strategy. The link is done through the **RRSGP** parameter according to the rule **RRSGP**.

The update time of HART parameters in the **TBH** block is indicated by **HART_TSTAMP** parameter.

Function Block Creation

The DFI302 and fieldbus devices use function blocks to build strategies, such as PID, AI blocks, etc. This means that **Syscon** can be used to set up every part of the system - transmitters, positioners and controller - in a same language. Once built the control strategy and chose the function blocks to be located in DFI302, set up the CHANNEL parameter for that function block, which does the interface with I/O modules.

CHANNEL Configuration

Using DFI302, the user can configure the number of I/O modules as well the I/O type (input or output, discrete, analog, pulse etc). The DFI302 is the unique device classified as a configurable I/O device. All I/O modules have the I/O points arranged as follow:

Rack	0 ~ 14
Slot	0 ~ 3
Group	0 ~ 1
Point	0 ~ 7

The value in the CHANNEL parameter is composed by these elements in the **RRSGP** form.

Rack (R): Each rack has four slots. The rack is numbered from 0 (first rack) till 14 (last rack). Therefore a single I/O point in the DFI302 can be identified by specifying the rack (R), slot (S), group (G) and point (P). The CHANNEL parameter in the multiple I/O blocks (MIO) must specify the whole group (8 points), the point must be 9, which mean the whole group.

NOTE

Each rack has a rotating switch to select address. The possible addresses are **0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F**. Note that the "F" address is not allowed when I/O is being accessed by HCT function block or DF65 co-processor. The address "F" is supported when the I/O access is done through FFB 1131, which may be configured using **LogicView for FFB** software. For further details about FFB1131 block refer to Adding Logic Using Flexible Function Blocks section.

Slot (S): One slot supports one I/O module, and it is numbered from 0 (first slot in the rack) till 3 (last slot in the rack).

Group (G): Ordinal number of group in the specified I/O module, it is numbered from 0 (first group) till number of groups minus 1.

Point (P): Ordinal number of I/O point in a group, it is numbered from 0 (first point) to 7(last point in the group), and 9 mean the whole group of points.

For example, a CHANNEL parameter equals to 1203, it means rack 1, slot 2, group 0 and point 3. If the CHANNEL parameter of a MAI block is 10119, it means rack 10, slot 1, group 1 and point 9 (whole group). Before setting the CHANNEL parameter, it is recommended to configure the hardware in the HCT block. Because the write check will verify if the I/O type configured in the HCT block is suitable for block type. Therefore setting the CHANNEL parameter of AI block to access an I/O type different of analog input will be rejected.



Module Specification Standard

The module specification is shown in a format similar to the example below. All of the module specifications explain functionality, field connection, and electrical characteristics, and shows a simplified schematic of the interface circuit for better understanding.

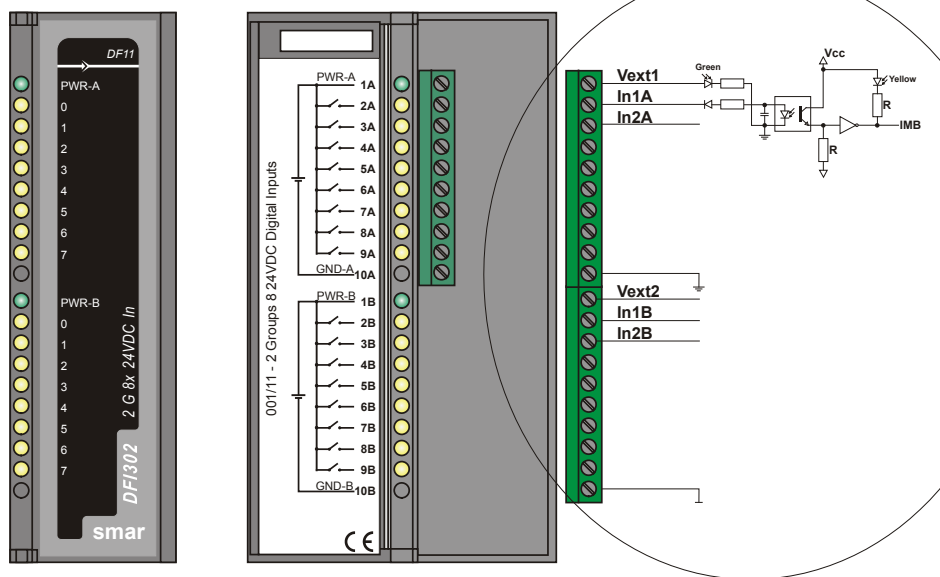
For further details about the complete line of I/O modules refer to Digital and Analog Input/Output Modules of DF1302 can be obtained downloading the file MESDAME.pdf. (folder K) in the Smar website (www.smar.com).

DF11/DF12/DF13/DF14 - DC Input Modules

DF11 (2 groups of 8 24 Vdc inputs isolated)
 DF12 (2 groups of 8 48 Vdc inputs isolated)
 DF13 (2 groups of 8 60 Vdc inputs isolated)
 DF14 (2 groups of 8 125 Vdc inputs isolated)

Description

This module detects the DC input voltage and converts it in a TRUE (ON) or FALSE (OFF) logic signal . It has 2 groups isolated optically.



Technical Specifications

ARCHITECTURE	
Number of Inputs	16
Number of Groups	2
Number of Points per Group	8

ADDING RACKS

DF1A – Rack with 4 slots

Description

A rack is basically a plastic support for the IMB circuit that carries the connectors where the modules are plugged in. These connectors that fit the modules are called slots.

New racks can be added according to the project requirements. Up to 16 Racks are allowed. Racks can be connected for Local I/O expansion using flat cables (DF3, DF4A ~ DF7A).

Remember that the distance between the first module and the last module of a DF1302 system, expanded by flat cables cannot exceed 22.97ft (7 meters).

NOTE

Each Rack has a rotating switch to select the address. The possible addresses are **0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F**. Note that the “F” address is not allowed when I/O is being accessed by HCT function block or DF65 co-processor. The address “F” is supported when the I/O access is done through FFB 1131, which can be configured using **LogicView for FFB** software. For further details about FFB1131 block refer to Adding Logic Using Flexible Function Blocks section.

There are restrictions related to the module location on the rack. The restrictions are as follows:

1. The first slot of rack 0 is always reserved for the power supply module.
2. The second slot of rack 0 is always reserved for the controller module.
3. All additional power Supplies need to be placed in the slot 0 of the desired Rack (jumper W1 in the rack must be cut before plugging the power supply).
4. The first rack must have a DF84 terminator when the controller (DF62, DF63, DF73, DF75, etc) executes local logic in discrete output cards.
5. The last rack must have a DF2 terminator installed.

Technical Specifications

DIMENSIONS AND WEIGHT	
Dimensions (W x H x D)	148.5 x 25 x 163 mm ; (5.85 x 0.98 x 6.42 in)
Weight	0.216 kg

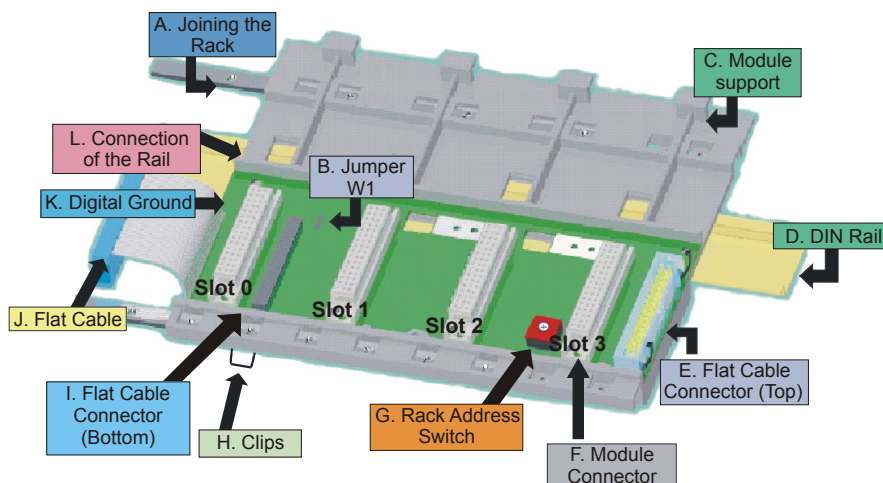


Figure 9. 1 – DF1A Rack

DF78 - Rack with 4 slots for Redundant CPUs

Description

The DF78 rack allows that two CPUs modules access the same I/O. This possibility is used when necessary redundancy and availability to the system. Up to 16 DF1A racks can be connected to DF78. Racks can be connected for Local I/O expansion using flat cables (DF3, DF4A ~ DF7A).

Remember that the distance between the first module and the last module of a DFI302 system, expanded by flat cables cannot exceed 22.97ft (7 meters).

There are restrictions related to the power supply and controllers position on the DF78 Rack. The restrictions are as follows:

- 1. The first and second slots of DF78 rack are always reserved for power supply modules.
- 2. The third and fourth slots on DF78 rack are always reserved for controller modules.

NOTE

Each Rack has a rotating switch to select the address. The possible addresses are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. Note that the “F” address is not allowed when I/O is being accessed by HCT function block or DF65 co-processor. The address “F” is supported when the I/O access is done through FFB 1131, which can be configured using **LogicView for FFB** software. For further details about FFB1131 block refer to Adding Logic Using Flexible Function Blocks section.

Technical Specifications

DIMENSIONS AND WEIGHT	
Dimensions (W x H x D)	148.5 x 25 x 163 mm ; (5.85 x 0.98 x 6.42 in)
Weight	0.216 kg

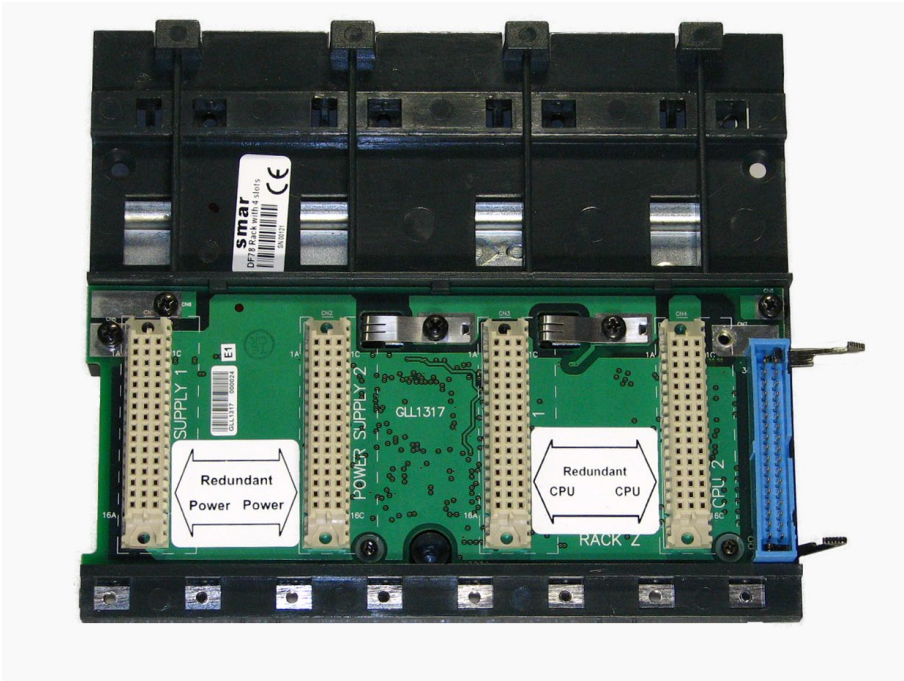


Figure 9. 2 – DF78 Rack

DF93 - Rack with 4 slots (with diagnostic)

Description

The DF93 rack is integral part of the new power system of DF1302. Its features provide low voltage drop through the IMB bus, so it is more efficient. Besides, the diagnostics resources of DF93 help in the problems detection minimizing the time stop and maintenance. The diagnostic can be obtained observing the diagnostics LEDs or through the status reading via controller.

The DF93 rack has Vcc and GND terminals at laterals (for power transmission). DF93's finishing avoids short circuits between the Vcc and GND connections at laterals.

As in the previous system, new racks can be added to the DF1302 system according to the application needs. Up to 16 racks are allowed. The racks can be connected among them (expanding the bus) using flat cables (DF101 to DF107), DF90 (IMB power cable), and DF91 (lateral adapter).

Remember that the distance between the first module and the last module of a DF1302 system, expanded by flat cables cannot exceed 22.97ft (7 meters).

NOTE

Each Rack has a rotating switch to select the address. The possible addresses are **0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F**. Note that the "F" address is not allowed when I/O is being accessed by HCT function block or DF65 co-processor. The address "F" is supported when the I/O access is done through FFB 1131, which can be configured using **LogicView for FFB** software. For further details about FFB1131 block refer to Adding Logic Using Flexible Function Blocks section.

There are restrictions related to the module location on the rack. The restrictions are as follows:

1. The first slot of rack 0 is always reserved for the power supply module.
2. The second slot of rack 0 is always reserved for the controller module.
3. All additional power supplies need to be placed in the slot 0 of the desired rack (jumper W1 in the rack must be cut and the DF90 cable from the previous racks must be disconnected before plugging the power supply).
4. The first rack must have a DF84 terminator when the controller (DF62, DF63, DF73, DF75, etc) executes local logic in discrete output cards.
5. The last rack must have a terminator installed - DF2 (right side) or DF96 (left side). For further details refer to Installing section.
6. Grounding terminals must be used.

Technical Specifications

DIMENSIONS AND WEIGHT	
Dimensions (W x H x D)	148.5 x 25 x 163 mm ; (5.85 x 0.98 x 6.42 in)
Weight	0.216 kg

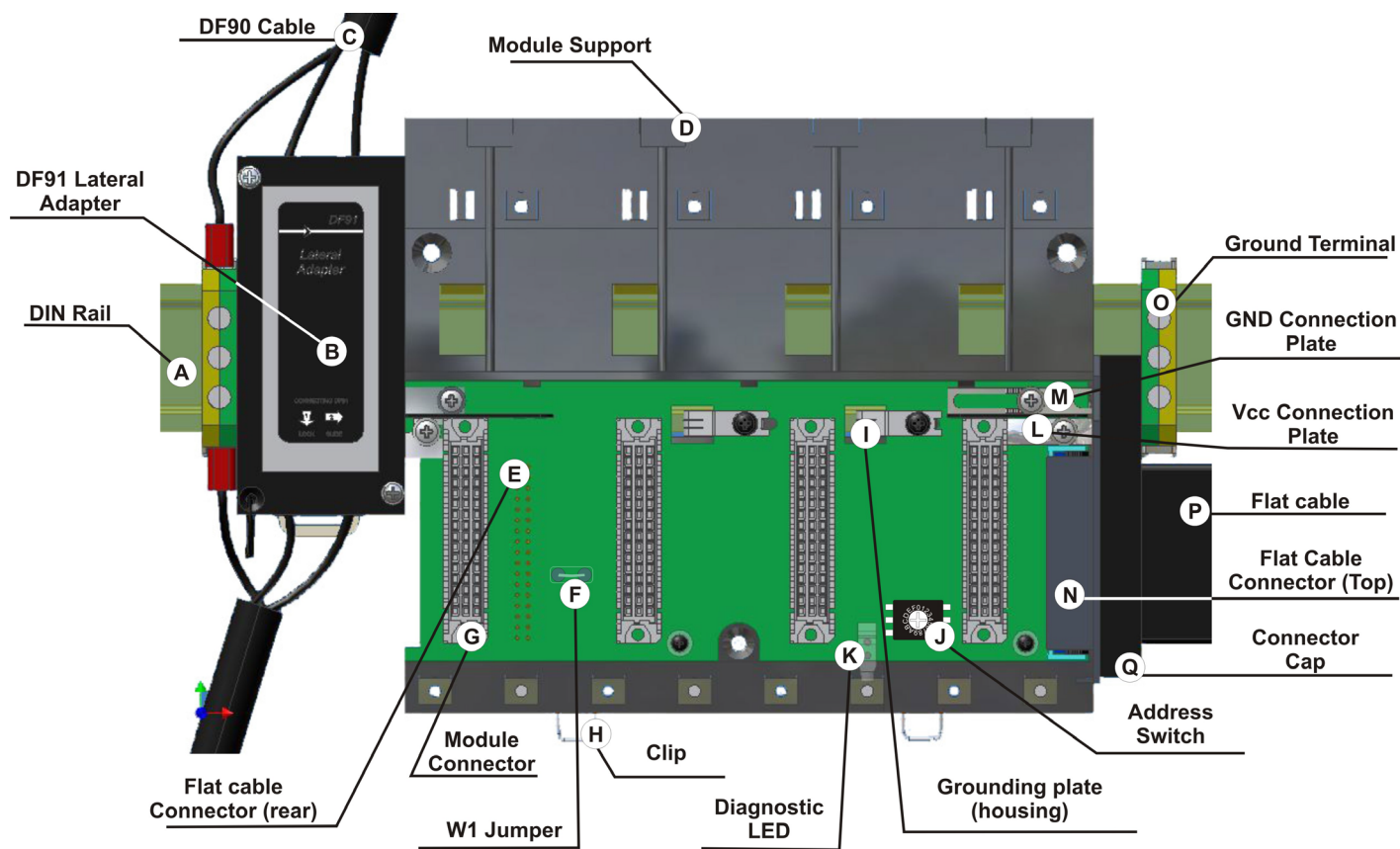


Figure 9. 3 – DF93 Rack

For compatibility with the EMC standards, if the power supply side connector on the left side of the rack is not connected, it should be capped with the left side protection according to the section Installing, Installing racks - DF92 and DF93 topic. This protection is provided along with the terminator DF2.

DF92 - Rack with 4 slots for redundant CPUs (with diagnostic support)

Description

The DF92 is the new rack for redundant controllers in the IMB. Its function is similar to the DF78, but DF92 is optimized to reduce voltage drop in the IMB, besides it has different pins to connect, in the future, power supplies with more than 3A.

The DF92 rack has Vcc and GND terminals at laterals (for power transmission). DF92's finishing avoids short circuits between the Vcc and GND connections at laterals.

Moreover, the DF92 supports power supplies diagnostics for those that have this feature. It helps in problems detection and giving the desired confidence in the availability offered by redundancy. The diagnostic can be obtained observing the diagnostics LEDs or through the status reading via controller.

The DF92 rack can be connected up to 16 DF93 racks. The racks can be connected among them (expanding the bus) using flat cables (DF101 to DF107), DF90 (IMB power cable) and DF91 (lateral adapter).

Remember that the distance between the first module and the last module of a DF1302 system, expanded by flat cables cannot exceed 22.97ft (7 meters).

There are restrictions related to the module location on the rack. The restrictions are as follows:

1. The first and second slots of DF92 rack are always reserved for power supply modules.
2. The third and fourth slots on DF92 rack are always reserved for controllers' modules.
3. Grounding terminals must be used.

NOTE

Each Rack has a rotating switch to select the address. The possible addresses are **0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F**. Note that the "F" address is not allowed when I/O is being accessed by HCT function block or DF65 co-processor. The address "F" is supported when the I/O access is done through FFB 1131, which can be configured using **LogicView for FFB** software. For further details about FFB1131 block refer to Adding Logic Using Flexible Function Blocks section.

Technical Specifications

DIMENSIONS AND WEIGHT	
Dimensions (W x H x D)	148.5 x 25 x 163 mm ; (5.85 x 0.98 x 6.42 in)
Weight	0.216 kg

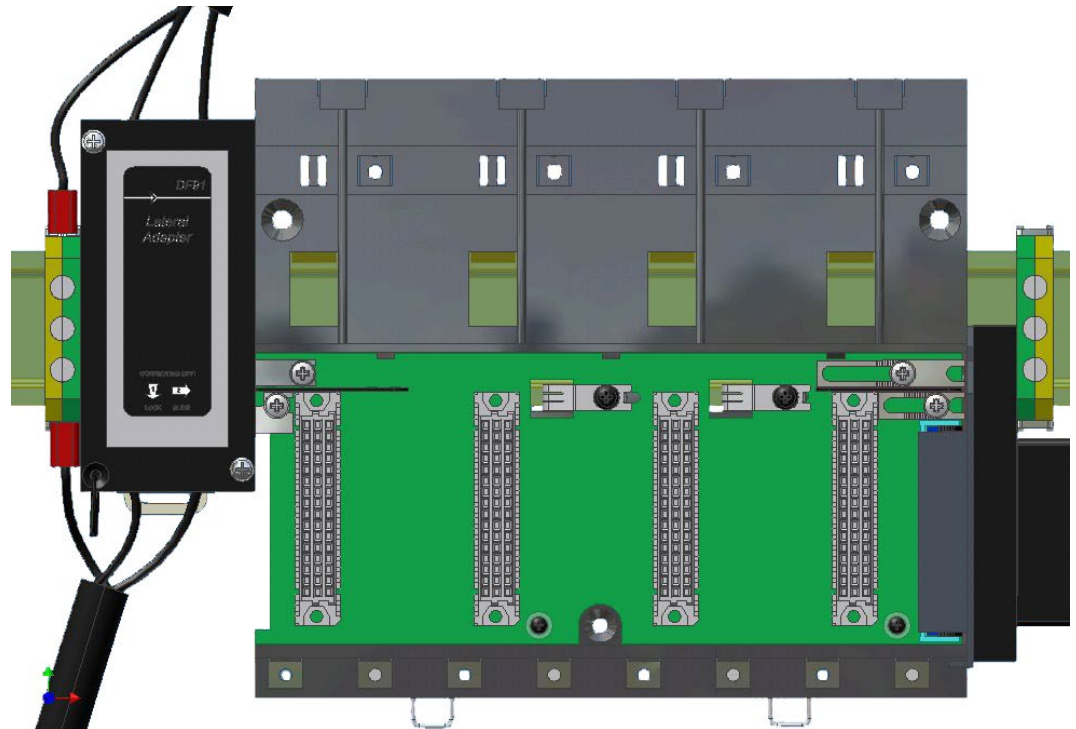


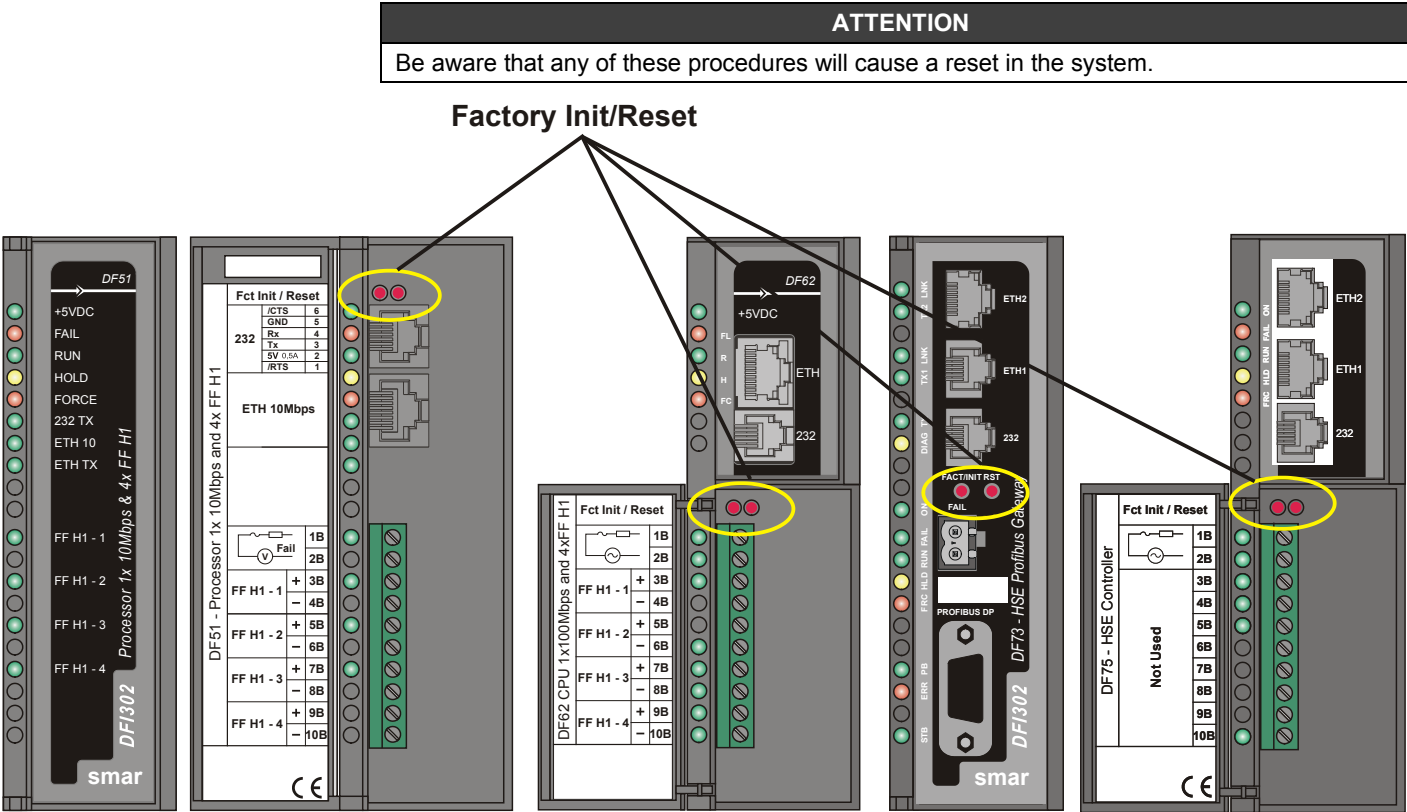
Figure 9. 4 – DF92 Rack

For compatibility with the EMC standards, if the power supply side connector on the left side of the rack is not connected, it should be capped with side protection according to the section Installing, Installing racks - DF92 and DF93 topic. This protection is provided along with the terminator DF2.

TROUBLESHOOTING

DFI302 has some initialization resources to solve certain kind of problems. Some resources will be described below.

Two small push buttons are available allowing the user to choose a specific controller action (see details in the following figure that shows two small buttons located in the controllers).



The next table summarizes the possible actions for the DF51 controller.

Procedure Name	Push Buttons procedure	Controller action
Reset	Press the right push-button.	Controller will reset, taking some seconds for correct system initialization. A new IP will be attributed automatically (when DHCP Server is available) or the last fixed configured IP will be kept, depending on the last setting via FBTools and/or Mode 3. Controller should go to Run Mode always, in case of a valid firmware available.
Mode 1 – Factory Init	Keep the left push-button pressed. Click the right push-button ensuring that the FORCE LED is blinking once a second. Release the left push-button and the system will execute the Reset, clearing previous configuration.	Controller will execute a factory initialization erasing the previous configurations downloaded via Syscon . A new IP will be attributed automatically (when DHCP Server is available) or the last fixed configured IP will be kept, depending on the last setting via FBTools and/or Mode 3. Controller should go to Run Mode always, in case of a valid firmware available.
Mode 2 – Hold	Keep pressed the left push-button and double-click the right push-button ensuring that the FORCE LED is blinking twice a second. Release the left push-button. The system will execute the Reset and be in Hold. Check if the HOLD and ETH LNK LED are ON.	Controller will fulfill a reset and will go to Hold mode. Verify that the HOLD and ETH10 LEDs remain ON. In this mode, you can use the FBTools Wizard to download the firmware or change IP address value and settings. If you want to go back to the execution mode (Run), press right push-button to reset.
Mode 3 – IP Automatic Assign	Keep pressed the left push-button and click three times the right push-button ensuring that the FORCE LED is blinking three times per second. Release the left push-button.	A new IP will be attributed automatically (when DHCP Server is available) or the default IP address (192.168.164.100) will be set. Controller should go to Run Mode always, in case of a valid firmware available.

The following table shows the actions for the DF62, DF63, DF73, DF75, DF79, DF81, DF89, DF95, and DF97 controllers:

Procedure Name	Push Buttons procedure	Controller action
Reset	Press the right Push-button.	Controller will reset, taking some seconds for correct system initialization. A new IP will be attributed automatically (when DHCP Server is available) or the last fixed configured IP will be kept, depending on the last setting via FBTools and/or Mode 3. Controller should go to Run or Hold mode depending on the last state before the Reset.
Mode 1 – Factory Init	Keep the left Push-button pressed. Click the right Push-button ensuring that the FORCE LED is blinking once a second. Release the left push-button and the system will execute the Reset, clearing previous configuration.	Controller will execute a factory initialization erasing the previous configurations downloaded via Syscon . A new IP will be attributed automatically (when DHCP Server is available) or the last fixed configured IP will be kept, depending on the last setting via FBTools and/or Mode 3. Controller should go to Run or Hold mode depending on the last state before the Reset.
Mode 2 – Hold	Keep pressed the left push-button and double-click the right push-button ensuring that the FORCE LED is blinking twice a second. Release the left push-button. System will execute the Reset and change the mode. The LED indicators will be HOLD or RUN depending on the mode.	Controller will execute a reset and will go to HOLD mode. Verify that the HOLD and ETH10 LED indicators remain ON. In this mode, you can use the FBTools Wizard to download the firmware or change IP address value and settings. Execute the mode 2 again If you want to go back to the execution mode (RUN), press right push-button to reset.
Mode 3 – IP Automatic Assign	Keep pressed the left push-button and click three times the right push-button ensuring that the FORCE LED is blinking three times per second. Release the left push-button.	A new IP will be attributed automatically (when DHCP Server is available) or the default IP address (192.168.164.100 for port 0 or 192.168.165.100 for port 1, for all controllers, except for DF62) will be set. Controller should go to Run or Hold mode depending on the last state before the Reset.

HINTS

- Once started any mode (Factory Init or Hold Mode) can be prevented. Remain the right push-button pressed, and release the left push-button first.
- If the user loses how many times have pressed the right push-button, just verify the number of times that the FORCE LED is blinking per second. After the fourth touch it will come to blink once a second (this function is cyclic).
- To press these push-buttons use some pointed instrument (for example a ballpoint pen).

NOTE

In case of there is more than one **DFI302** to be set up, fulfill the following command to clear **ARP table**, before setting up the next **DFI302**.

```
C:\>arp -d 192.168.164.100 < enter >
```

When to use the procedures of Factory Init/Reset

1. How to reset the DFI302 without disconnecting it?

Use the Reset procedure.

2. HOLD LED remains ON, how to proceed?

After turning on the DF51, if HOLD LED remains ON, the firmware may be invalid. The user should download it again. For further details refer to Setting Up section.

3. ETH10 LED does not turn on, how to proceed?

Check if the cable is connected correctly, or if it is not damaged. Check the specification of the cables:

DF54 – Standard Cable. To be used in network communication between DFI302 and Switch/Hub.

DF55 – Cross Cable. To be used in point-to-point communication between computer and DFI302.

4. The FORCE LED is blinking, how to proceed?

Make sure to turn ON the switch 1 in the controller module, knowing that the newest firmware versions are now blocking the boot procedure when the battery is turned off.

Make sure that the battery is under good conditions. Just go to a web browser, enter controller IP address and check under "Power off data retention" field, granting "Retention good (Vbatt >= 2.5V)". If the conclusion is to change the battery, send the equipment to repair. It is possible as a temporary procedure, to force the boot of the controller, just going to telnet and entering "unlockforce" command. But this is only recommended when missing spare controller.

Make sure that the power supply in the rack is working fine. Check the voltage between 16A and 16C pins and grant a minimal voltage of 4.95 Volts.

5. The FBTools does not show all the DFI302s that are in the subnet, how to proceed?

Probably there is an IP address conflict in this Subnet. Disconnect all the DFI302s from the Subnet and follow the procedures on "Connecting the DFI302 to its Subnet" for each module, ensuring that the addresses to be used are not associated with another device in the network.

6. The FBTools does not find the DF51.

- Make sure that the initial connection procedure was followed, the default IP address was assigned via Reset Mode 3, and the computer has the IP 192.168.164.101.
- The Ethernet cable used must be the DF54 when using Hub or Switch. For point-to-point connection (module DF51 linked directly to the computer) use cross cable (DF55).
- Check if the network adapter is OK, executing the **ping** command to its own IP, via DOS Prompt.
- Check if the Ethernet connection is OK, executing the **ping** command to the DF51.

7. **The DF51 was operating correctly, I turned it off then turned it on and no type of reset works anymore and the HOLD LED remains lit or blinking constantly.**

Some DF51 hardware versions (earlier than Revision 2 and Emission 1) do not have writing protections in the firmware and monitor areas. Occasionally, some wrong settings and software bugs would cause firmware and monitors crashes. In this case, the use of the Boot Flash is necessary.

8. **How I have to use a Boot Flash to reload the Boot Program.**

Send the device to our factory or contact the Smar technical support to get the Factory Procedure "How to load the DF51 Boot".

9. **The license is not accepted by the LicenseView program.**

Follow the procedure described below:

1. Try to register the DEMO license. In the **LicenseView** window there is a "Use a DEMO keys" button. If it works, the problem must be a mistake while typing the key.
2. If it still does not work, check the existence of SmarOlePath in System variables. Right-click on **My Computer** → **Properties** → **Advanced Tab** → **Environment Variables**. In case of the non-existence of SmarOlePath variable, execute "Interface Setup" from Smar folder to create it.

NOTE

Use only numbers and dashes "-". Do not use space or symbols such as: "@ # \$ % ^ & * () _ + < > , . / ? \ | { } [] ; : " .

3. Fulfill the server register again. In the SMAR shortcut folder ("**Program Files\Smar\OleServers**") run **Register.Bat** program.
4. If any previous option fails, the License file can be generated manually.
 - Use an "ASC" text editor (notepad, for instance) because the file cannot contain format characters. The file name and its contents are shown below:

File: Syscon.dat

SMAR-MaxBlocks-**55873-03243-22123-04737-10406**

File: OleServer.dat

#PCI OLE Server

SMAR-OPC_NBLOCKS8-**23105-23216-11827-2196**

File: DfiOleServer.dat

#DFI OLE Server

SMAR-DFIOPC_NBLOCKS8-**31297-58800-49898-13173**

The keys presented are for DEMO license, the user can use its own keys.

10. **I cannot switch the Modbus Blocks to "AUTO", even setting the Mode Block target to "AUTO", the actual mode block keeps on "O/S".**

In order to set the Modbus Blocks to "AUTO", it is necessary that the Mode Block of the DFI Resource Block has been set to "AUTO" and the LOCAL_MOD_MAP of each Modbus Block must be different from 255.

11. **I defined a value different from 255 to the LOCAL_MOD_MAP of a Modbus Block but it remains 255.**

Within the same type of Modbus Block (MBCM, MBCS, MBSS, MBSM), it is not possible to contain two blocks with the same LOCAL_MOD_MAP, whereas the value must be from 0 to 15.

12. **I tried to change the static value of a Modbus Block, but the value does not update.**

In order to update the static value of a Modbus Block, it is necessary to set the Block to "O/S", so that the static values can be changed.

13. **After changing a static value of a block, and set the Mode Block target to "AUTO", the actual one does not change into "AUTO".**

If any static parameter of a Modbus Block has been changed, the Block only will be set to "AUTO" after accomplishing the "On_Apply" in the Block MBCF.

Section 11

TECHNICAL SPECIFICATIONS FOR THE CONTROLLERS

DFI302 Specifications

ENVIRONMENT CONDITIONS	
Operating	0~60 °C, 20-90% RH non-condensing.
Storage	-20~80 °C, 20-90% RH non-condensing. Exception: DF51 module -20~25 °C, 20-90 % (To achieve ten years of battery life, without excessive battery discharge).

PROTECTION DEGREE	
IP20	2 – Protected against solid objects with diameter bigger than 12 mm. 0 – Without specific protection against water.

When possible, it is recommended to keep the DFI302 firmware version updated according to the following table. Consult updates in the download area for the DFI302 on Smar web site

Controller	SYSTEM302 7.0.X	SYSTEM302 7.1.X	SYSTEM302 7.2.X	SYSTEM302 7.3.2	SYSTEM302 7.3.4	SYSTEM302 7.3.5	SYSTEM302 8.0 / 8.1 / 8.2
DF51	V3.9.5	V3.9.5	V3.9.5	V3.9.5	V3.9.5	V3.9.5	V3.9.5
DF62	V1.3.27	V1.3.27	V2.1.3	V3.0.5	V4.2.0	V4.2.0	V4.3.0
DF63*	V1.3.27	V1.3.27	V2.1.4	V3.0.5	V4.2.0	V4.2.0	V4.4.16
DF73	V1.0.21	V2.0.73	V2.0.73	V3.0.4	V4.0.1	V4.2.0	V4.3.0
DF75	V1.3.26	V1.3.26	V2.1.3	V3.0.5	V4.2.0	V4.2.0	V4.4.14
DF79	-	V1.0.0	V1.1.1	V2.0.4	V3.0.2	V3.0.4	V3.0.4
DF81	-	-	V2.0.2	V2.0.2	V3.0.0	V3.0.0	V3.0.0
DF89	-	-	V1.0.0	V2.0.4	V3.0.2	V4.2.0	V4.2.5
DF95	-	-	V2.0.73	V3.0.4	V4.0.1	V4.2.0	V4.3.0
DF97	-	-	V2.0.73	V3.0.4	V4.0.1	V4.2.0	V4.3.0
DF100	-	-	-	-	V2.0.0	V2.0.0	V2.0.0

*For DF63-RTU refer to “Firmware version and Device Revision” of DF63 topic.

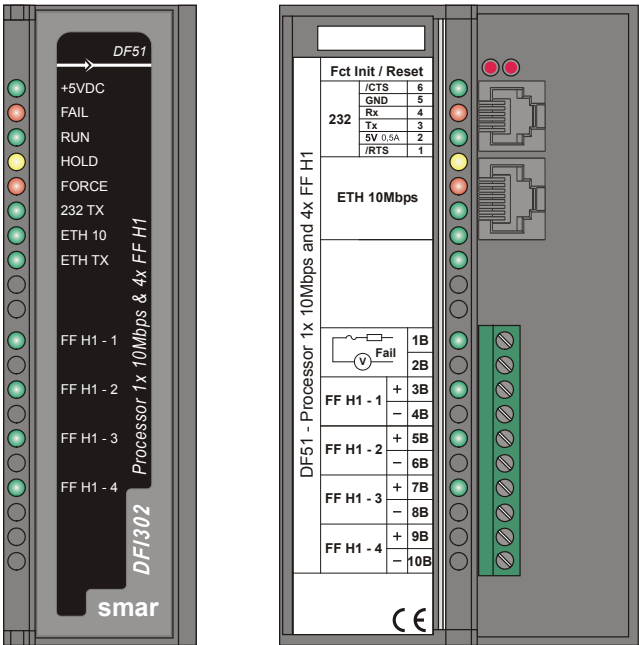
DF51 Specifications

Part Number

DF51 – Controller with 1 Ethernet 10 Mbps port and 4 H1 channels.

Description

The DF51 is the controller module that connects Fieldbus devices in the H1 bus, performing the LAS function (Link Active Schedule) in the network.



DF51 – Controller Module

Technical Specifications

DF51	
Type	32-bit RISC
Sustained Performance	50 MIPS
Memory for Code	2MB, 32-bit Flash Memory (Firmware upgrade available).
Memory for Data	2MB, 32-bit NVRAM (Data and configuration retention).
Fieldbus Interface	Numbers of Ports4, independent with DMA Physical Layer StandardISA-S50.02-1992 Baud Rate31.25 Kbps (H1) MAU TypePassive (no bus powered) Intrinsic SafetyNOT Compliant Isolation500 Vac (each channel)
Operation Voltage/Current	+5V ± 5% / 0.95A (typical).
Connector Ethernet	RJ-45.
Connector EIA-232	RJ-12.
FOUNDATION™ Function Blocks	100 (Maximum)

Battery

Type of battery	Lithium
Capacity	950 mAh
Devices maintained by the battery	RTC and NVRAM
Voltage	3.6 V (subject to revision when below 2.5 V)
Maximum life span	11 years (typical charge of 10 μ A)

Using the Fault Indication

The 1B and 2B terminals available in the DF51 may be used in a Fault Indication application. These terminals are only a NC Relay.

The NC Relay supports:

0.50 A @ 125 Vac
0.25 A @ 250 Vac
2.00 A @ 30 Vdc

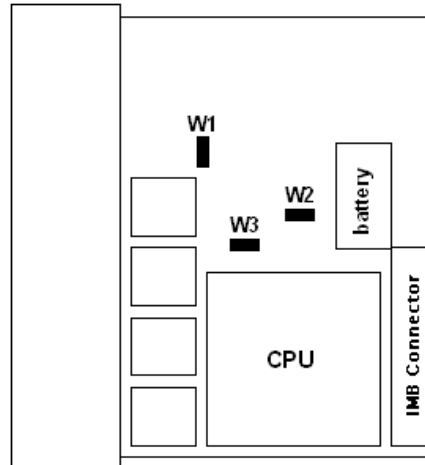
* Valid data for resistive loads.

Normally, DF51 forces this relay to stay open, but if any bad condition crashes the CPU, the hardware will close it. This status may be used in redundancy situation where the backup controller reads this contact and knows about the fault. Another choice is to use it to goes off an alarm.

NOTE

To comply with the EMC standard, the length of the cable connected to the fault relay must be less than 30 meters. The power supply for the load must not be from an external network.

Jumpers on Board

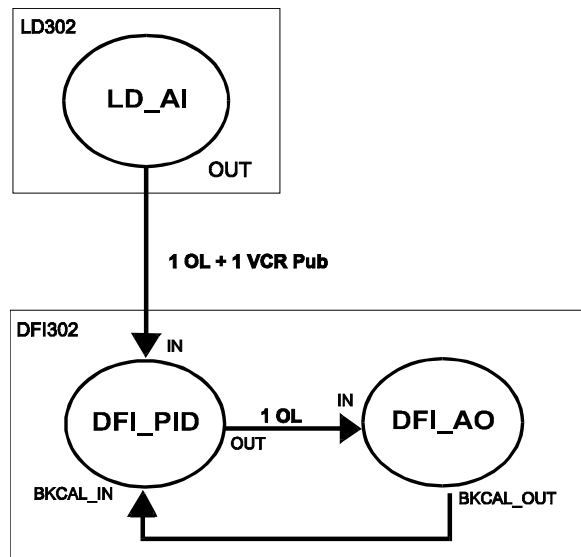


The W1 jumper (from DF51) must be enabled to make possible simulations through the Simulate (SIMULATE_D or SIMULATE_P) parameters of output and input function blocks.

Do not use W2 and W3 jumpers. They are only used to download firmware in the factory.

Fieldbus Limits

FOUNDATION fieldbus protocol uses the Publisher/Subscriber model for the communication among devices. When a link is set between two function blocks, the device with the function block that sends the data is considered a publisher, and the device that has the block receiving the data is considered a subscriber. See the following figure:



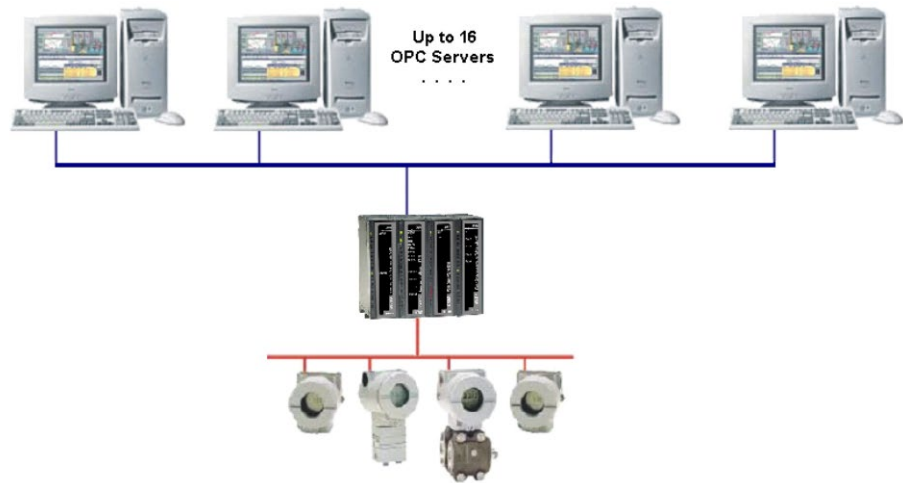
a) Internal links of the DF51 use only one Object Link (OL) and external links use 1OL + 1VCR Publisher (for blocks that send data) or 1OL + 1VCR Subscriber (for blocks that receive data). The DF51 can support up to 300 OLs (Object Links), 64 VCR publishers and 64 VCR subscribers.

b) The DFI302 controllers are responsible for executing the LAS table in the fieldbus networks where they are connected. The maximum entries limit that the DF51 is able to manage is equal to 70 between different field devices.

c) When using Smar field devices as Backup masters, pay attention to the maximum entries limit for LAS Table that is 50. So, it is necessary the planning for each fieldbus network in order to have until 50 links between different field devices when using Backup masters.

Supervision Limits

Each DF51 can supervise up to 400 tags simultaneously and support up to 16 OPC Servers connected. The typical and recommended topology uses 2 OPC Servers.



Modbus Limits

The DF51 can support up to 16 blocks for each type (MBCS, MBSM, MBCS and MBCM).

The DF51 has four types of Modbus function blocks that handle Modbus data: Modbus Control Master, Modbus Control Slave, Modbus Supervision Master and Modbus Supervision Slave. The first thing is to define if the DF51 will be Master or Slave.

The DF51 can have up to 16 blocks of one Modbus type. For example, if DF51 is master, it can have up to 16 blocks of each Master type. However, the user can have 16 Modbus Control Master and 16 Modbus Supervision Master Blocks. Each block has different numbers of inputs/outputs parameters. The numbers of inputs and outputs available are showed below:

Modbus Control Master (MBCM)

16 x 4 digital inputs = 64
 16 x 4 digital outputs = 64
 16 x 4 analog inputs = 64
 16 x 4 analog outputs = 64

Modbus Supervision Master (MBSM)

16 x 2 float value = 32
 16 x 2 percentage value = 32
 16 x 2 integer value = 32
 16 x 8 boolean value = 128

These scenarios show the DF51 controller as a Modbus Master device. Hence, the maximum number of Modbus blocks, inputs and outputs represents the Modbus limit.

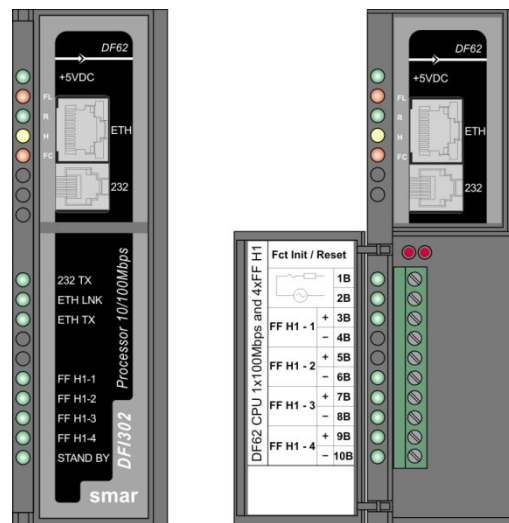
DF62 Specifications

Part Number

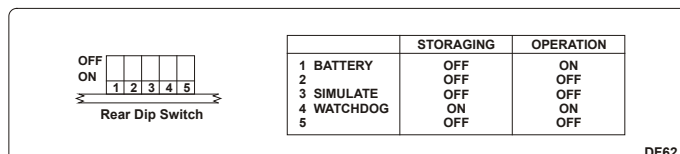
DF62 –HSE/ FOUNDATION fieldbus controller with 1 Ethernet port and 4 H1 channels

Description

DF62 module is the second generation of Smar HSE Linking Devices. With four H1 channels (FOUNDATION fieldbus), one 10/100 Mbps Ethernet port and capability of block execution. DF62 can work as a bridge H1-H1 or a linking device H1-HSE, allowing a wide communication between field devices and greater flexibility in the project of strategies of continuous control. Through the I/O cards, it is also possible to execute discrete control via relay diagram logic ("Ladder Diagram"), allowing a single and integrated system. The module DF62 also can act as Modbus gateway, allowing the interconnection of modules that are not fieldbus or HSE. DF62 also has redundant operation, giving higher security level for industrial process.



DF62 – Controller module



Characteristics and Controller Limits

- Four H1 channels (FOUNDATION fieldbus).
- It recommends the use of up to 32 field devices (8 devices per H1 channel). However, more devices can be used (up to 16 per H1 channel) under evaluation of performance according each application.
- Links characteristics:
 - 128 parameters can be linked externally via HSE and H1 (any proportion among HSE and H1 links totaling 128 links).
 - For external links among controller – H1 devices there is the limit of 16 Publisher links and 16 Subscribers links per each H1 port.
 - Regardless of the above limits there is the limit of 16 H1 bridge links (links among H1 ports of the same controller). This limit is shared by all ports, with no limit per port.
- Dynamic block instantiation.
- Maximum 100 function blocks.
- It supports Flexible Function Block with 242 parameters that can be linked by using interface between the discrete and analog control.
- LAS Function ("Link Active Scheduler");

- It supports up to 16 HART modules (DF116/DF117).

NOTE

The limit of HART modules per controller is 16, but you must remember that when mapping the points through fieldbus function blocks strategy, one transducer block is used for each HART device. In this case, the system may be limited to less than 16 HART modules depending on the maximum number of fieldbus function blocks supported by the controller.

Continuous Control with FOUNDATION™ Fieldbus

The DF62 controller acts like a bridge for the HSE main bus. It performs four functions:

- Message forwarding using Client/Server relationships
- Data republishing using Publisher/subscriber relationships
- Report forwarding using Report source/sink relationships
- Application clock time distribution

Discrete Control

DF62 module also has the capability of access I/O cards through the IMB (Inter-Module Bus), present in the backplane where the DF62 is mounted. Through the IMB, up to 16 racks DF1A or DF93 can be interconnected, each one having up to 4 cards. If there is a redundant controller is necessary the use of rack DF78 or DF92. If DF78 is used plus 16 racks DF1A can be added. If DF92 is used plus 16 racks DF93 can be used. Additional power supplies in other racks can be necessary depending on the quantity of cards.

DISCRETE CONTROL CHARACTERISTICS

I/O Points*	256 discrete or analog points (maximum)
Auxiliary Points	1024 points (maximum)
Ladder Function Blocks	300 blocks (maximum)
Analog Points Supervision	2400 points (maximum)
Configuration File	20 Kbytes (maximum)
Program Execution Cycle for 1000 boolean operations (without redundancy)	50 ms (minimum)** 90 ms (typical)***
Program Execution Cycle with redundancy	Increment of 10 ms (typical)**** up to 50ms (maximum) to execution cycle
Execution Average Time	5.8 ms/Kbytes of program (minimum) 10.5 ms/Kbytes of program (typical)

* The whole number of points includes inputs and outputs, analog or digitals. Maximum may change according I/O type used.

** 1131 Flexible Function Block adjusted to One (High Priority). Each 1000 boolean operations allocate 8.6 Kbytes.

*** Total execution time will change depending on the adjusted priority of 1131 FFB. The adjustment should be compatible with the quantity of function blocks and HSE links.

**** The whole execution time may change depending of the configuration file size.

Flexible Function Block Usage

The interconnection between the discrete control and the ladder logic can be done by using Flexible Function Block. For further details about the Flexible Function Block usage, refer to the section Adding logic by using Flexible Function Block and the **LogicView for FFB** manual.

Firmware version and Device Revision

Some Firmware Versions may change the Device Revision, so this is an important step to consider during the configuration procedure for the controller. The section “Adding Function Blocks” better describes the steps for this configuration. The current available versions are:

Firmware version 1.x: Device Revision = 2
 Firmware version 2.x: Device Revision = 3
 Firmware version 3.x: Device Revision = 4
 Firmware version 4.x: Device Revision = 5

Technical Specifications

Memory

TYPE	SIZE
Volatile Memory	8 Mbytes
Non-Volatile Memory	4 Mbytes
EEPROM	1 Kbytes
Flash to the program	4 Mbytes
Flash to monitor	2 Mbytes

Battery

Type of battery	Battery Panasonic BR-2/3AE2SP - Lithium
Capacity	1200 mAh
Devices maintained by the battery	RTC and NVRAM
Minimum life span	8 years (typical charge of 17 μ A)
Maximum life span	49 years (typical charge of 2.8 μ A)
Voltage	3 V (subject to revision when below 2.5 V)

Communication Ports and Channels

ETHERNET PORT	
Communication Rate	10/100 Mbps
Standard	IEEE 802.3u
Isolation	150 Vrms
Operation Mode	Full duplex
Connector	RJ45 with shield*

* Grounded to the rail used for fixing the rack in which the DF62 is installed.

H1 CHANNELS	
Number of H1 Channels	4
Communication Rate	31.25 kbps
Standard	EN 61158 EN 50170
Physical Layer	ISA-S50.02-1992
H1 Modem	FB3050P (3.3V)
MAU Type	Passive (bus not powered)
Isolation	500 Vac
Bus Current	40 mA

MODBUS PORT	
Communication Rate (Maximum)*	115200 bps
Standard	EIA-232
Connector**	RJ12 with shield
Maximum Current ***	0.5A @ 3.3V

*There is an increase in error rate as we increase the communication rate over 19200 bps. In many situations these errors can be acceptable, and they are not noticed by supervision.

** Grounded to the rail used for fixing the rack in which the DF62 is installed.

*** Internally protected by solid state fuse.

REDUNDANT PORT	
Maximum Communication Rate	115200 bps
Standard	EIA-232
Connector*	RJ12 with shield
Maximum Current **	0.5A @ 3.3V

* Grounded to the rail used for fixing the rack in which the DF62 is installed.

** Internally protected by solid state fuse.

FAILURE RELAY	
Output Type	Solid state relay, normally closed (NC), isolated
Maximum Voltage	30 Vdc
Maximum Current	200 mA
Overload Protection	Not available. It must be provided externally
Normal Operation	Opened contacts
Failure Condition	Closed contacts
Maximum cable length (maximum) connected to the relay	30 m

Observation: The power supply for the load must not be from an external network (outside the panel).

IMB BUS	
Voltage	5 Vdc
Bus	8 bits
Failure Signal	Yes
Hot Swap	Yes
Redundancy in the bus access	Yes, but only using the DF78 or DF92

Module Features

CONTROLLER	
CPU	Family ARM7TDMI
Bus	32 bits
Architecture	RISC
Performance	40 MIPS
CPU Cache	8 Kbytes
Clock	40 MHz
DMA	10 channels
Ethernet	MAC 10/100 integrated
Watchdog	Yes (200 ms of cycle)
Operation Voltage	3.3 V for I/O

MODULE	
Operation Voltage	5 V ($\pm 5\%$ of tolerance)
Typical Current	550 mA
Real Consumption	2,75 W
Environment Air Temperature	0 - 60° C (IEC 1131)
Storage Temperature	-20 - 80° C (IEC 1131)
Relative Air Humidity (Operation)	5% - 95% (non-condensing)
Cooling Mode	Air Convection
Dimensions (H x W x D) in mm	149 x 40 x 138 (without package)

Electrical Certification

DF62 follows the immunity test specification to equipment to industrial installation, as IEC61326:2002 standard.

ENCLOSE	
Electrostatic discharge (IEC61000-4-2)	4 kV/8 kV contact/air
EM field (IEC61000-4-3)	10 V/m
Rated power frequency magnet field (IEC61000-4-8)	30 A/m

AC POWER	
Voltage dip/short interruptions (IEC61000-4-11)	0,5 cycle, each polarity/100%
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

DC POWER	
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

I/O SIGNAL CONTROL	
Burst (IEC61000-4-4)	1 kV
Surge (IEC61000-4-5)	1 kV
Conducted RF (IEC61000-4-6)	3 V

I/O SIGNAL CONTROL CONNECTED TO POWER SUPPLY	
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

Emission Rate

ENCLOSE	
30 to 230 MHz (CISPR 16-1, CISPR 16-2)	40 dB (uV/m) quasi peak, measured at 10 m distance
239 to 1000 MHz (CISPR 16-1, CISPR 16-2)	40 dB (uV/m) quasi peak, measured at 10 m distance

AC MAINS	
0,15 to 0,5 MHz (CISPR 16-1, CISPR 16-2)	79 dB (uV) quasi peak 66 dB (uV) average
0,5 to 5 MHz (CISPR 16-1, CISPR 16-2)	73 dB (uV) quasi peak 60 dB (uV) average
5 to 30 MHz (CISPR 16-1, CISPR 16-2)	73 dB (uV) quasi peak 60 dB (uV) average

Indication LEDs

The LED names, colors, descriptions, and behaviors are showed in the table below.

LED	COLOR	DESCRIPTION	BEHAVIOR
+5V DC (ON)	Green	It indicates when the module is ON.	Solid green LED when power is on.
FAIL (FAIL)	Red	It indicates hardware failure.	Solid red LED when in fail.
RUN (RUN)	Green	It indicates when the controller is operating in normal mode.	Solid green LED when in operation.
HOLD (HLD)	Yellow	It indicates when the controller is in hold mode. In this mode the controller does not perform any application and does not interfere on the plant operation (access via I/O cards or via digital bus are disabled)	Solid yellow LED when the controller is in hold mode (HOLD).
FORCE (FRC)	Red	<p>1 – It indicates different modes of startup or maintenance requested by the operator via push-buttons (FACT INIT, HOLD and IP Address).</p> <p>2 – It indicates power failure when the operating voltage decays down the expected value of 4.8 V (low line).</p> <p>3 – Indicates a battery problem.</p>	<p>1 - Depending on the number of times that the right push-button is pressed, the FRC LED blinks at a given rate for a limited time, indicating the chosen mode (refer to Troubleshooting section for further information).</p> <p>2 – Permanently solid red. The module will reset if the voltage reaches 4.6V. (HLD and FAIL LEDs light up together temporarily).</p> <p>3 –FRC LED blinking and HLD LED solid red during the module start up – indicates low battery or the battery rear DIP switch is turned off (refer to Troubleshooting section for further information)</p>
232 TX	Green	It indicates activity in the RS-232 port.	Blinking green LED when the RS-232 port is transmitting data.
ETH LNK	Green	It indicates when the Ethernet link is active.	Solid green LED when the Ethernet link is established.
ETH TX	Green	It indicates communication activity in the Ethernet port.	Blinking green LED when the Ethernet port is transmitting data.
FF H1-1, FF H1-2, FF H1-3, FF H1-4	4 x Green	It indicates H1 channel activity.	Blinking green LED when the H1 channel is transmitting data.
STANDBY	Green	<p>If the HOLD LED is solid yellow and the STB LED is blinking, the firmware is being updating.</p> <p>If the HOLD LED is off, it indicates the controller role on redundancy as well as the synchronism status.</p>	There are several blinking patterns to indicate different synchronism status. Refer to redundancy section for further details.

DF63 Specifications

Part Number

DF63 –HSE/ FOUNDATION fieldbus Controller with 2 HSE Ethernet ports and 4 H1 channels

IMPORTANT

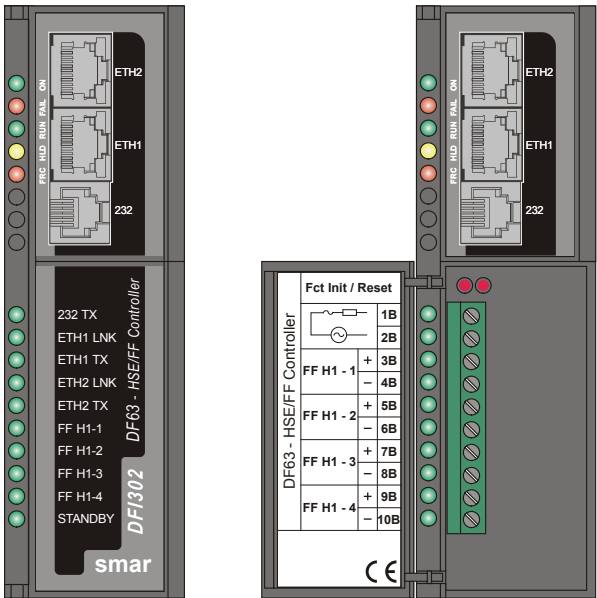
From serial number 7000 onwards there was a CPU upgrade.

DF63 modules (NETARM CPU) have serial numbers below 7000. Their specific characteristics will be indicated by (SN<7000).

DF63 modules (iMX6 CPU) have serial numbers above 7000. Their specific characteristics will be indicated by (SN>=7000).

Description

DF63 module is the second generation of Smar HSE Linking Devices. With four H1 channels (FOUNDATION™ fieldbus), two 10/100 Mbps Ethernet ports and capability of block execution. DF63 can work as a bridge H1-H1 or as linking device H1-HSE, allowing a wide communication between field devices and greater flexibility in the project of strategies of continuous control. It has two Ethernet ports for HSE network redundancy. Through the I/O cards, it is also possible to execute discrete control via relay diagram logic (“Ladder Diagram”), allowing a single and integrated system. The module DF63 also can act as Modbus gateway, allowing the interconnection of modules that are not fieldbus or HSE. DF63 also has redundant operation, giving higher security level for industrial process.



DF63 – Controller module

OFF
ON

1 2 3 4 5

Rear Dip Switch

	STORAGING	OPERATION
1 BATTERY	OFF	ON
2	OFF	OFF
3 SIMULATE	OFF	OFF
4 WATCHDOG	ON	ON
5	OFF	OFF

Characteristics and Controller Limits

- Four H1 channels (FOUNDATION™ fieldbus).
- It is recommended the use of up to 32 field devices (8 devices per H1 channel). However, more

devices can be used (up to 16 per H1 channel) under evaluation of performance according each application.

- Links characteristics:
 - 128 parameters can be linked externally via HSE and H1 (any proportion among HSE and H1 links totaling 128 links).
 - For external links among controller – H1 devices there is the limit of 16 Publisher links and 16 Subscribers links per each H1 port.
 - Regardless of the above limits there is the limit of 16 H1 bridge links (links among H1 ports of the same controller). This limit is shared by all ports, with no limit per port.
- Dynamic block instantiation.
- Maximum 100 function blocks.
- It supports Flexible Function Block with 242 parameters that can be linked by using interface between the discrete and analog control.
- LAS Function (“Link Active Scheduler”).
- It supports up to 16 HART modules (DF116/DF117).

NOTE

The limit of HART modules per controller is 16, but you must remember that when mapping the points through fieldbus function blocks strategy, one transducer block is used for each HART device. In this case, the system may be limited to less than 16 HART modules depending on the maximum number of fieldbus function blocks supported by the controller.

Continuous Control with FOUNDATION™ Fieldbus

The DF63 controller acts like a bridge for the HSE main bus. It performs four functions:

- Message forwarding using Client/Server relationships
- Data republishing using Publisher/subscriber relationships
- Report forwarding using Report source/sink relationships
- Application clock time distribution

Discrete Control

DF63 module also has the capability of access I/O cards through the IMB (Inter-Module Bus), present in the backplane where the DF63 is mounted. Through the IMB, up to 16 racks can be interconnected, each one having up to 4 cards. If there is a redundant controller is necessary the use of rack DF78 or DF92. If DF78 is used plus 16 racks DF1A can be added. If DF92 is used plus 16 racks DF93 can be used. Additional power supplies in other racks can be necessary depending on the load of the cards.

DISCRETE CONTROL CHARACTERISTICS

I/O Points*	256 discrete or analog points (maximum)
Auxiliary Points	1024 points (maximum)
Ladder Function Blocks	300 blocks (maximum)
Analog Points Supervision	2400 analog points (maximum)
Configuration File	20 Kbytes (maximum)
Program Execution Cycle for 1000 boolean operations (without redundancy)	50 ms (minimum)** 90 ms (typical)***
Program Execution Cycle with redundancy	Increment of 10ms (typical)**** up to 50ms (maximum) to execution cycle
Execution Average Time	5.8 ms/Kbytes of program (minimum) 10.5 ms/Kbytes of program (typical)

* The whole number of points includes inputs and outputs, analog or digitals. Maximum may change according I/O type used.

** 1131 Flexible Function Block adjusted to One (High Priority). Each 1000 boolean operations allocate 8,6 Kbytes.

*** Total execution time will change depending on the adjusted priority of 1131 FFB. The adjustment should be compatible with the quantity of function blocks and HSE links.

**** The whole execution time may change depending of the configuration file size.

Flexible Function Block Usage

The interconnection between the discrete control and the ladder logic can be done by using Flexible Function Block. For further details about the Flexible Function Block usage, refer to the section Adding logic by using Flexible Function Block and the LogicView for FFB manual.

Firmware version and Device Revision

Some Firmware Versions may change the Device Revision, so this is an important step to consider during the configuration procedure for the controller. The section “Adding Function Blocks” better describes the steps for this configuration. The current available versions are:

Firmware version 1.x: Device Revision = 1

Firmware version 2.x: Device Revision = 2

Firmware version 3.x: Device Revision = 3

Firmware version 4.x: Device Revision = 5

ATTENTION

The **DF63-RTU** controller is a dedicated version of the **DF63**, specially designed to operate in remote environments, where typically the communication channel requires special handling. See Appendix C for further details. Its firmware versions have some particularities:

Support for DNP3 and FFB1

SN < 7000: Firmware version: 2.2.6, Device Revision 4

SN ≥ 7000: Firmware version: 7.2.6, Device Revision 4

Support for DNP3 and FFB2

SN < 7000: Unavailable

SN ≥ 7000: Firmware version: 7.4.10, Device Revision 6

Technical Specifications

Memory

TYPE	SIZE	
Serial Number	SN<7000	SN≥7000
Volatile Memory	8 Mbytes	256 Mbytes
Non-Volatile Memory*	4 Mbytes	4 Gbytes Flash
EEPROM	1 Kbytes	256 Bytes EEPROM 32 Mbytes Serial Flash
Flash to the program	4 Mbytes	-
Flash to monitor	2 Mbytes	-

* It is kept by not rechargeable internal battery.

Battery

Type of battery	Battery Panasonic BR-2/3AE2SP - Lithium
Capacity	1200 mAh
Devices maintained	RTC and NVRAM (SN<7000) RTC (SN≥7000)
Minimum lifespan (estimated) ⁽¹⁾⁽²⁾⁽³⁾	10 years (typical) Variable with operating temperature. Important to disconnect the battery when the module is not installed.
Voltage	3 V (subject to revision when below 2.5 V)

(1) The battery only comes into operation when the DF63 is de-energized. In this condition, the lifespan of the battery in relation to its charge is estimated to be at least 10 years, for ambient temperatures up to 25°C.

(2) For this reason, it is important to keep the battery switch off when the module is not installed for operation (key 1 of the rear dip switch).

(3) The status of the battery lifespan can be monitored through the module's webserver in the “Information” page, “Power off data retention” field.

Capacitor (SN>=7000)

Type of capacitor ⁽¹⁾	GLL1493 (super capacitor)
Capacity	6 s hold (minimum)
Devices maintained	Processor, during shutdown.
Lifespan (estimated) ⁽²⁾⁽³⁾	10 years (typical) Variable with operating temperature.
Voltage	5.4 V (maximum)

(1) Consult technical assistance if replacement is necessary.

(2) The lifespan of the capacitor is a theoretical data, considering operation at fixed temperatures: 18 years, operating at 50°C, 11 years, operating at 55°C and 7 years, operating at 60°C.

(3) The status of the capacitor lifespan can be monitored through the module's webserver in the "Information" page, "Power off data retention" field.

Communication Ports and Channels

ETHERNET PORTS	
Communication Rate	10/100 Mbps
Standard	IEEE 802.3u
Isolation	SN < 7000: 150 Vrms SN >= 7000: 1500 Vrms
Operation Mode	Full duplex
Connector	RJ45 with shield*

* Grounded to the rail used for fixing the rack in which the DF63 is installed.

H1 CHANNELS	
Number of H1 Channels	4
Communication Rate	31.25 kbps
Standard	EN 61158 EN 50170
Physical Layer	ISA-S50.02-1992
H1 Modem	FB3050P (3.3V)
MAU Type	Passive (bus not powered)
Isolation	500 Vac
Bus Current	40mA

MODBUS PORT	
Communication Rate (Maximum)*	115200 bps
Standard	EIA-232
Connector**	RJ12 with shield
Maximum Current ***	0.5A @ 3.3V

*There is an increase in error rate as we increase the communication rate over 19200 bps. In many situations these errors can be acceptable, and they are not noticed by supervision.

** Grounded to the rail used for fixing the rack in which the DF63 is installed.

*** Internally protected by solid state fuse.

REDUNDANT PORT	
Maximum Communication Rate	115200 bps
Standard	EIA-232
Connector*	RJ12 with shield
Maximum Current **	0.5A @ 3.3V

* Grounded to the rail used for fixing the rack in which the DF63 is installed.

** Internally protected by solid state fuse.

FAILURE RELAY	
Output Type	Solid state relay, normally closed (NC), isolated
Maximum Voltage	30 Vdc
Maximum Current	200 mA
Overload Protection	Not available. It must be provided externally
Normal Operation	Opened contacts
Failure Condition	Closed contacts
Maximum cable length connected to the relay	30 m

Observation: The power supply for the load must not be from an external network (outside the panel).

IMB BUS	
Voltage	5 Vdc
Bus	8 bits
Failure Signal	Yes
Hot Swap	Yes
Redundancy in the bus access	Yes, but only using the DF78 or DF92 rack

Start-up time

TYPICAL START-UP TIME, FROM POWER-UP TO H1 START-UP	
SN < 7000	30 s
SN >= 7000	90 s

Module Features

CONTROLLER		
Serial Number	SN < 7000	SN >= 7000
CPU	Family ARM7TDMI	ARM Cortex™-A9, 800 MHz
Bus	32 bits	32 bits
Architecture	RISC	RISC
Performance	40 MIPS	2000 DMIPS
CPU Cache	8 Kbytes	L1 Instruction Cache: 32K L1 Data Cache: 32K L2 Cache: 512K
Clock	40 MHz	800 MHz
DMA	10 channels	-
Ethernet	MAC 10/100 integrated	MAC 10/100 Automated polarity detection and correction
Watchdog	Yes (200ms of cycle)	Yes (30 s of cycle)
Operation Voltage	3.3 V for I/O	3.3V

MODULE	
Operation Voltage	5V (\pm 5% of tolerance)
Typical Current	550 mA (SN<7000) 566 mA (SN>=7000)
Real Consumption	2.75 W (SN<7000) 2.83 W (SN>=7000)
Environment Air Temperature (*)	0 to 60° C (IEC 1131)
Storage Temperature	-20 to 80° C (IEC 1131)
Relative Air Humidity (Operation)	5% to 95% (non-condensing)
Cooling Mode	Air Convection
Dimensions (H x W x D) in mm	149 x 40 x 138 (without package)

*RTU Operation: In closed and unventilated panels, where temperature gradients may exist, make sure the air temperature immediately near the DF63-RTU is between -30°C and 70°C.

Electrical Certification*

DF63 follows the immunity test specification to equipment to industrial installation, as IEC61326:2002 standard.

ENCLOSE	
Electrostatic discharge (IEC61000-4-2)	4 kV/8 kV contact/air
EM field (IEC61000-4-3)	10 V/m
Rated power frequency magnet field (IEC61000-4-8)	30 A/m

AC POWER	
Voltage dip/short interruptions (IEC61000-4-11)	0,5 cycle, each polarity/100%
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

DC POWER	
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

I/O SIGNAL CONTROL	
Burst (IEC61000-4-4)	1 kV
Surge (IEC61000-4-5)	1 kV
Conducted RF (IEC61000-4-6)	3 V

I/O SIGNAL CONTROL CONNECTED TO POWER SUPPLY	
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

Emission Rate

ENCLOSE	
30 to 230 MHz (CISPR 16-1, CISPR 16-2)	40 dB (uV/m) quasi peak, measured at 10m distance
239 to 1000 MHz (CISPR 16-1, CISPR 16-2)	40 dB (uV/m) quasi peak, measured at 10m distance

AC MAINS	
0,15 to 0,5 MHz (CISPR 16-1, CISPR 16-2)	79 dB (uV) quasi peak 66 dB (uV) average
0,5 to 5 MHz (CISPR 16-1, CISPR 16-2)	73 dB (uV) quasi peak 60 dB (uV) average
5 to 30 MHz (CISPR 16-1, CISPR 16-2)	73 dB (uV) quasi peak 60 dB (uV) average

*SN >= 7000 - Tests pending

Firmware update with FBTools

DFi Download Classic	File extension: abs (SN < 7000)
Batch Download	File extension: bin and txt (SN < 7000) File extension: 7k and txt (SN >=7000)

Indication LEDs

The LED names, colors, descriptions, and behaviors are showed in the table below.

LED	COLOR	DESCRIPTION	BEHAVIOR
+5V DC (ON)	Green	It indicates when the module is ON.	Solid green LED when power is on.
FAIL (FAIL)	Red	It indicates hardware failure.	Solid red LED when in fail.
RUN (RUN)	Green	It indicates when the controller is operating in normal mode.	Solid green LED when in operation.
HOLD (HLD)	Yellow	It indicates when the controller is in hold mode. In this mode the controller does not perform any application and does not interfere on the plant operation (access via I/O cards or via digital bus are disabled)	Solid yellow LED when the controller is in hold mode (HOLD).
FORCE (FRC)	Red	<p>1 – It indicates different modes of startup or maintenance requested by the operator via push-buttons (FACT INIT, HOLD and IP Address).</p> <p>2 – It indicates power failure when the operating voltage decays down the expected value of 4.8 V (low line).</p> <p>3 – Indicates a battery problem.</p>	<p>1 - Depending on the number of times that the right push-button is pressed, the FRC LED blinks at a given rate for a limited time, indicating the chosen mode (refer to Troubleshooting section for further information).</p> <p>2 – Permanently solid red. The module will reset if the voltage reaches 4.6V. (HLD and FAIL LEDs light up together temporarily).</p> <p>3 –FRC LED blinking and HLD LED solid red during the module start up – indicates low battery or the battery rear DIP switch is turned off (refer to Troubleshooting section for further information)</p>
232 TX	Green	It indicates activity in the RS-232 port.	Blinking green LED when the RS-232 port is transmitting data.
ETH1 LNK	Green	It indicates when the Ethernet link is active (ETH1 port).	Solid green LED when the Ethernet link is established (ETH1 port).
ETH1 TX	Green	It indicates communication activity in the ETH1 port.	Blinking green LED when the ETH1 port is transmitting data.
ETH2 LNK	Green	It indicates when the Ethernet link is active (ETH2 port).	Solid green LED when the Ethernet link is established (ETH2 port).
ETH2 TX	Green	It indicates communication activity in the ETH2 port.	Blinking green LED when the ETH2 port is transmitting data.
FF H1-1, FF H1-2, FF H1-3, FF H1-4	4 x Green	It indicates H1 channel activity.	Blinking green LED when the H1 channel is transmitting data.
STANDBY	Green	<p>If the HOLD LED is solid yellow and the STB LED is blinking, the firmware is being updating.</p> <p>If the HOLD LED is off, it indicates the controller role on redundancy as well as the synchronism status.</p>	There are several blinking patterns to indicate different synchronism status. Refer to redundancy section for further details.

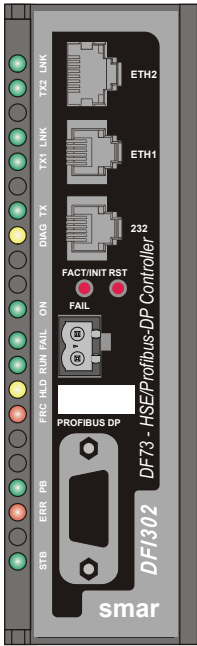
DF73 Specifications

Part Number

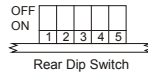

DF73 –HSE/Profibus DP Controller with 2 Ethernet ports e 1 Profibus DP channel

Description

DF73 module is the Smar solution for Profibus applications. Its main feature is working as Profibus DP-HSE gateway to provide power to the connectivity and flexibility to the system application. It allows wide communication between the Profibus DP and PA field devices. Through the HSE network and other DFI302 modules, it is possible the communication between field devices and other industrial protocols, providing greater flexibility to the control strategy projects. Through the I/O cards, it is also possible to execute discrete control via relay diagram logic (“Ladder Diagram”), allowing a single and integrated system. The module DF73 also can act as Modbus gateway (slave), allowing the interconnection of modules that are not fieldbus or HSE.



DF73 – Controller module



Rear Dip Switch

	STORAGING	OPERATION
1 BATTERY	OFF	ON
2	OFF	OFF
3 SIMULATE	OFF	OFF
4 WATCHDOG	ON	ON
5	OFF	OFF

DF73

Characteristics and Module Limits

- One Profibus DP channel supporting up to 12 Mbps;
- It supports up to 124 Profibus field devices;
- It supports up to 3584 bytes of input and 3584 bytes of output during the data interchange process;
- Limit of 64 external links by the HSE network;
- Up to16 server sessions and 16 client sessions;
- Maximum of 250 function blocks per DF73;
- One (1) Flexible Function Block (counted into the 250 allowed blocks), with 242 linked parameters to interface between the discrete and continuous control.
- It supports up to 16 HART modules (DF116/DF117).

NOTE

The limit of HART modules per controller is 16, but you must remember that when mapping the points through fieldbus function blocks strategy, one transducer block is used for each HART device. In this case, the system may be limited to less than 16 HART modules depending on the maximum number of fieldbus function blocks supported by the controller.

Continuous Control with Profibus

DF73 is a complete Profibus HSE controller with capacity to execute function blocks. Through the available System302, Studio302 and Syscon configuration tools is possible to configure the DF73 totally.

HSE Communication:

- Maximum of 512 link objects;
- Up to 50 requests for non-connected services can be pendent per connection;
- Supervision up to 2000 points per second;
- Configurable Views.

Discrete Control

DF73 module also has the capability of access I/O cards through the IMB (Inter-Module Bus), present in the backplane where the DF73 is mounted. Through the IMB, up to 16 racks DF1A or DF93 can be interconnected, each one having up to 4 cards. If there is a redundant controller is necessary the use of rack DF78 or DF92. If DF78 is used plus 16 racks DF1A can be added. If DF92 is used plus 16 racks DF93 can be used. Additional power supplies in other racks can be necessary depending on the load of the cards.

DISCRETE CONTROL CHARACTERISTICS

I/O Points*	1024 discrete points or 512 analog (maximum)
Ladder Function Blocks	2000 blocks (maximum)
Configuration File	120 Kbytes (maximum)**
Program Execution Cycle for 1000 boolean operations (without redundancy)	50 ms (minimum)*** 90 ms (typical)****
Program Execution Cycle with redundancy	Increment of 10ms (typical)***** up to 50 ms (maximum) to execution cycle
Execution Average Time	5.8 ms/Kbytes of program (minimum) 10.5 ms/Kbytes of program (typical)

* The whole number of points includes inputs and outputs, analog or digitals. Maximum may change according to I/O type used.

** 120 Kbytes are available in firmware version 2.x and later. Earlier versions limit is 60 Kbytes.

*** 1131 Flexible Function Block adjusted to One (High Priority). Each 1000 boolean operations allocate 8.6 Kbytes.

**** Total execution time will change depending on the adjusted priority of 1131 FFB. The adjustment should be compatible with the quantity of function blocks and HSE links.

***** The whole execution time may change depending of the configuration file size.

NUMBER OF I/O POINTS

Number of Profibus devices per network	125
Number of virtual points	4096
Number of Profibus discrete points	2048
Number of Profibus analog points	512

Firmware version and Device Revision

Some Firmware Versions may change the Device Revision, so this is an important step to consider during the configuration procedure for the controller. The section "Adding Function Blocks" better describes the steps for this configuration. The current available versions are:

Firmware version 1_0_X: (Not supported from SYSTEM302 V7.2.x)

Firmware version 2_0_X: Device Revision = 3

Firmware version 3_0_X: Device Revision = 5

Firmware version 4_0_X: Device Revision = 6

Technical Specifications

Memory

TYPE	SIZE
Volatile Memory	8 Mbytes
Non Volatile Memory *	4 Mbytes
EEPROM	1 Kbytes
Flash to the program	4 Mbytes
Flash to monitor	2 Mbytes
Flash to EC1 (Profibus)	4 Mbytes

* It is kept by not rechargeable internal battery.

Battery

Type of battery	Battery Panasonic BR-2/3AE2SP - Lithium
Capacity	1200 mAh
Devices maintained by the battery	RTC and NVRAM
Minimum life span	8 years (typical charge of 17 μ A)
Maximum life span	49 years (typical charge of 2.8 μ A)
Voltage	3 V (subject to revision when below 2.5 V)

Communication Ports and Channels

ETHERNET PORT	
Rate	10/100 Mbits/s
Standard	IEEE 802.3u
Isolation	150 Vrms
Operation mode	Full duplex
Connector	RJ45 with shield*

* Grounded to the rail used for fixing the rack in which the DF73 is installed

DP PROFIBUS CHANNEL	
Rate	From 9.6 Kbit/s to 12 Mbits/s
Standard	EN 50170 e EN 50254
Physical Layer	EIA RS-485
Profibus Modem	EC1 (Hilscher)
Connector	DB9

MODBUS PORT	
Maximum rate*	115200 bps
Standard	EIA-232
Connector**	RJ12 with shield
Maximum Current***	0,5A @ 3.3V

*There is an increase in error rate as we increase the communication rate over 19200 bps. In many situations these errors can be acceptable, and they are not noticed by supervision.

** Grounded to the rail used for fixing the rack in which the DF73 is installed

*** Internally protected by solid state fuse.

REDUNDANCY PORT	
Maximum rate	115200 bps
Standard	EIA-232
Connector*	RJ12 with shield
Maximum Current**	0.5A @ 3.3V

* Grounded to the rail used for fixing the rack in which the DF73 is installed

** Internally protected by solid state fuse.

FAILURE RELAY	
Output Type	Solid state relay, normally closed (NC), isolated
Maximum voltage	30 Vdc
Maximum Current	200 mA
Overload Protection	Does not have. It must be provided externally
Normal Operation	Opened contacts
Failure Condition	Closed contacts
Maximum cable length connected to the relay	30 m

Observation: The power supply for the load must not be from an external network (outside the panel).

IMB BUS	
Voltage	5 Vdc
Bus	8 bits
Failure indication	Yes
Hot swap	Yes

Module Characteristics

CONTROLLER	
CPU	Family ARM7TDMI
Bus	32 bits
Architecture	RISC
Performance	40 MIPS
CPU Cache	8 Kbytes
Clock	40 MHz
DMA	10 channels
Ethernet	MAC 10/100 integrated
Watchdog	Yes (200 ms of cycle)
Power Supply Voltage	3.3 V for I/O and 2.5V for core (552 mW)

CARD	
Power Supply Voltage	5 V ($\pm 5\%$ of tolerance)
Typical Current	620 mA
Real consumption	2.75 W
Environment Air Temperature	0 to 60° C according to the IEC 1131 standard
Storage Temperature	-20 to 80° C according to the IEC 1131 standard
Relative Humidity of the Operation Air	5% to 95% non condensing
Cooling Mode	Air convection
Weight	0.318 kg
Dimensions (H x W x D) in mm	149x40x138 (without enclose)

Electrical Certification

DF73 follows the immunity test specification to equipment to industrial installation, as IEC61326:2002 standard.

ENCLOSE	
Electrostatic discharge (IEC61000-4-2)	4 kV/8 kV contact/air
EM field (IEC61000-4-3)	10 V/m
Rated power frequency magnet field (IEC61000-4-8)	30 A/m

AC POWER	
Voltage dip/short interruptions (IEC61000-4-11)	0,5 cycle, each polarity/100%
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

DC POWER	
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

I/O SIGNAL/CONTROL	
Burst (IEC61000-4-4)	1 kV
Surge (IEC61000-4-5)	1 kV
Conducted RF (IEC61000-4-6)	3 V

I/O SIGNAL/CONTROL CONNECTED DIRECTLY TO POWER SUPPLY NETWORK	
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

Emission Rate

ENCLOSE	
30 to 230 MHz (CISPR 16-1, CISPR 16-2)	40 dB (uV/m) quasi peak, measured at 10m distance
239 to 1000 MHz (CISPR 16-1, CISPR 16-2)	40 dB (uV/m) quasi peak, measured at 10m distance

AC MAINS	
0,15 to 0,5 MHz (CISPR 16-1, CISPR 16-2)	79 dB (uV) quasi peak 66 dB (uV) average
0,5 to 5 MHz (CISPR 16-1, CISPR 16-2)	73 dB (uV) quasi peak 60 dB (uV) average
5 to 30 MHz (CISPR 16-1, CISPR 16-2)	73 dB (uV) quasi peak 60 dB (uV) average

Indication LEDs

The LED names, colors, descriptions, and behaviors are showed in the table below.

LED	Color	Description	Behavior when Primary	Function when Secondary	Function when in HOLD
ETH LNK 2 (LNK)	Green	It indicates when the Ethernet link is active (ETH2 port).	Solid green LED when the Ethernet link is established (ETH2 port).	Same function and behavior when Primary.	Same function and behavior when Primary.
ETH TX2 (TX2)	Green	It indicates communication activity in the ETH2 port.	Blinking green LED when the ETH2 port is transmitting data.	Same function and behavior when Primary.	Same function and behavior when Primary.
ETH LNK 1 (LNK)	Green	It indicates when the Ethernet link is active (ETH1 port).	Solid green LED when the Ethernet link is established (ETH1 port).	Same function and behavior when Primary.	Same function and behavior when Primary.
ETH TX1 (TX1)	Green	It indicates communication activity in the ETH1 port.	Blinking green LED when the ETH1 port is transmitting data.	Same function and behavior when Primary.	Same function and behavior when Primary.
232 TX (TX)	Green	It indicates activity in the RS-232 port.	Blinking green LED when the RS-232 port is transmitting data.	Same function and behavior when Primary.	Same function and behavior when Primary.
Diagnostic (DIAG)	Yellow	It indicates the operation mode in the RS-232.	OFF when the RS-232 port is connected to the NetArm processor. Solid yellow when the RS-232 port is connected to the EC1 to execute operations related to the Profibus.	Same function and behavior when Primary.	Must be ON. ON – Ready for connection to EC1. OFF – Not supported
+5V DC (ON)	Green	It indicates when the module is ON.	Solid green LED when the module is powered.	Same function and behavior when Primary.	Same function and behavior when Primary.
FAIL (FAIL)	Red	It indicates hardware failure.	Solid red LED when in fail	Same function and behavior when Primary.	Same function and behavior when Primary.
RUN (RUN)	Green	It indicates when the controller is operating in normal mode.	Solid green LED when in operation.	Same function and behavior when Primary.	Must be OFF. It is the inverse value of the HDL LED that indicates function in Monitor mode (maintenance mode of the device).
HOLD (HLD)	Yellow	It indicates when the controller is in hold mode. In this mode the controller does not perform any application and does not interfere on the plant operation (access via I/O cards or via digital bus are disabled)	Solid yellow LED when the controller is in hold mode (HOLD).	Same function and behavior when Primary.	Must be ON.
FORCE (FRC)	Red	1 – It indicates different modes of startup or maintenance requested by the operator via push-buttons (FACT INIT, HOLD and IP Address). 2 – It indicates power failure when the operating voltage decays down the expected value of 4.8 V (low line). 3 – Indicates a battery problem.	1 - Depending on the number of times that the right push-button is pressed, the FRC LED blinks at a given rate for a limited time, indicating the chosen mode (refer to Troubleshooting section for further information). 2 – Permanently solid red. The module will reset if the voltage reaches 4.6V. (HLD and FAIL LEDs light up together temporarily). 3 –FRC LED blinking and HLD LED solid red during the module start up – indicates low battery or the battery rear DIP switch is turned off (refer to Troubleshooting section for further information)	Same function and behavior when Primary.	Hardware failure or power failure indication ON – Failure of power supply or hardware OFF – Power Ok. BLINKING – Mode selection

LED	Color	Description	Behavior when Primary	Function when Secondary	Function when in HOLD
Profibus Network (PB)	Green	It indicates Profibus channel activity.	<p>ON – Profibus DP communication enabled</p> <p>OFF – Profibus DP communication disabled, communication (transmission) failure, short circuit on Profibus DP channel</p> <p>BLINKING – Multimaster mode operation</p>	Same function and behavior when Primary.	Must be OFF. Profibus DP communication is disabled in HOLD.
Profibus Error (ERR)	Red	It indicates it there is an error related to the Profibus network.	<p>ON - At least one slave device in this configuration is not communicating properly.</p> <p>OFF - All slave devices present in the configuration are communicating properly.</p> <p>BLINKING - Momentary failures in slaves or Primary recovering the Profibus DP channel.</p> <p>In the case of redundancy operation with no Profibus DP equipment in the network or Profibus DP cables disconnected the behavior is BLINKING.</p>	<p>ON – Failure on Profibus DP cable or on the Profibus DP configuration synchronism</p> <p>OFF – Synchronism OK and redundant master in standby</p> <p>BLINKING - Primary recovering the Profibus DP channel.</p> <p>This LED should always be OFF in the secondary. If it are ON some switchover failure may have occurred. The user must wait for recovery.</p>	<p>Must be OFF.</p> <p>Profibus DP communication is disabled in HOLD.</p>
STANDBY (STB)	Green	<p>If the HOLD LED is solid yellow and the STB LED is blinking, the firmware is being updating.</p> <p>If the HOLD LED is off, it indicates the controller role on redundancy as well as the synchronism status.</p>	There are several blinking patterns to indicate different synchronism status. Refer to redundancy section for further details.	Must be ON.	BLINKING – Updating the firmware (saving the firmware on flash memory)

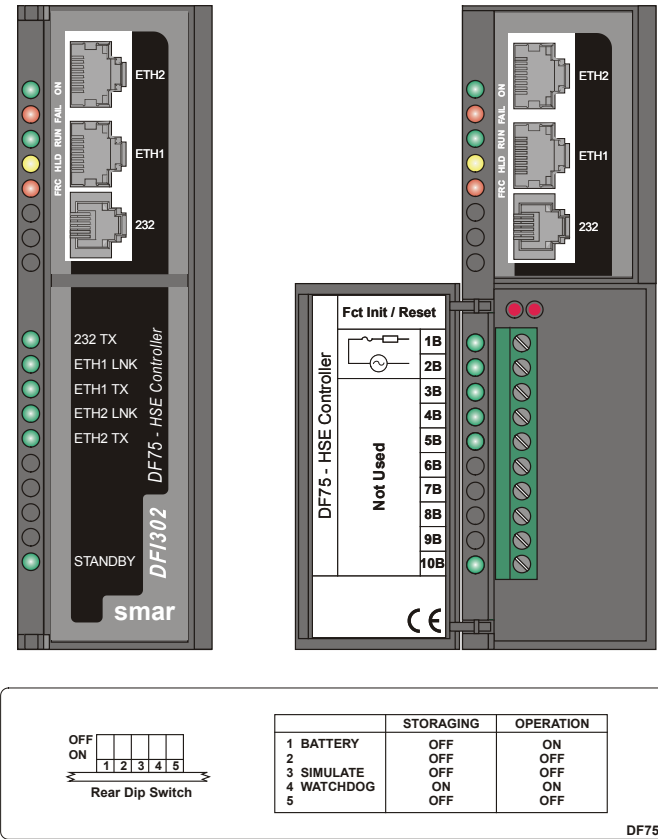
DF75 Specifications

Part Number

DF75 –HSE Controller

Description

DF75 module is the second generation of Smar Logic Controller including 2 Ethernet ports for HSE protocol and capability of FOUNDATION™ fieldbus block execution. DF75 is a HSE field device whose main purpose is the discrete control associated with continuous control through the use of FOUNDATION™ fieldbus blocks. Through the I/O cards, it is possible to execute the discrete control via relay diagram logic (Ladder Diagram). Besides that DF75 has two Ethernet ports that guarantee high availability of control and supervision. DF75 also have redundant operation, giving higher availability security level for industrial process.



Characteristics and Controller Limits

- 2 10/100 Mbps Ethernet ports;
- Support up to 100 FOUNDATION™ fieldbus function blocks;
- 128 parameters can be linked externally via HSE links;
- Support Flexible Function Block (FFB);
- Discrete control via relay diagram;
- Access to I/O modules;
- Webserver;
- Modbus Gateway;
- Redundant operation;
- Real Time Clock (RTC) and watchdog;
- Supervision for up to 2000 points per second.
- It supports up to 16 HART modules (DF116/DF117).

NOTE

The limit of HART modules per controller is 16, but you must remember that when mapping the points through fieldbus function blocks strategy, one transducer block is used for each HART device. In this case, the system may be limited to less than 16 HART modules depending on the maximum number of fieldbus function blocks supported by the controller.

Continuous Control with FOUNDATION™ Fieldbus

The DF75 module is a HSE device, with block execution capability. It has up to 100 blocks, including a Flexible Function Block to link FOUNDATION fieldbus control strategies with Ladder. Through configuration tools available in the **SYSTEM302**, such as the **Studio302** and **Syscon**, it is possible to configure the DF75 completely.

Discrete Control

DF75 module also has the capability of access I/O cards through the IMB (Inter-Module Bus), present in the backplane where the DF75 is mounted. Through the IMB, up to 16 racks can be interconnected, each one having up to 4 cards. If there is a redundant controller is necessary the use of rack DF78 or DF92. If DF78 is used plus 16 racks DF1A can be added. If DF92 is used plus 16 racks DF93 can be used. Additional power supplies in others racks can be necessary depending on the load of the cards.

DISCRETE CONTROL CHARACTERISTICS	
I/O Points*	1024 discrete points or 512 analog (maximum)
Auxiliary Points	4096 points (maximum)
Ladder Function Blocks	2000 blocks (maximum)**
Configuration File	120 Kbytes (maximum)**
Program Execution Cycle for 1000 boolean operations (without redundancy)	10 ms (minimum)*** 32 ms (typical)****
Program Execution Cycle with redundancy	Increment of 10ms (typical)***** up to 50 ms (maximum) to execution cycle
Execution Average Time	1.1 ms/Kbytes of program (minimum) 3.7 ms/Kbytes of program (typical)

* The whole number of points includes inputs and outputs, analog or digitals. Maximum may change according to I/O type used.

** 120 Kbytes and 2000 blocks are available in firmware version 2.x and later. Earlier versions limits are 60 Kbytes and 1200 blocks respectively.

*** 1131 Flexible Function Block adjusted to Zero (Very High Priority) and no other function blocks and HSE links are configured. Each 1000 boolean operations allocate 8.6 Kbytes.

**** Total execution time will change depending on the adjusted priority of 1131 FFB. The adjustment should be compatible with the quantity of function blocks and HSE links.

***** The whole execution time may change depending of the configuration file size.

Flexible Function Block Usage

The interconnection between the discrete control and the ladder logic can be done by using Flexible Function Block. For further details about the Flexible Function Block usage, refer to the section Adding logic by using Flexible Function Block and the LogicView for FFB manual.

Firmware version and Device Revision

Some Firmware Versions may change the Device Revision, so this is an important step to consider during the configuration procedure for the controller. The section "Adding Function Blocks" better describes the steps for this configuration. The current available versions are:

Firmware version 1.x: Device Revision = 1

Firmware version 2.x: Device Revision = 2

Firmware version 3.x: Device Revision = 3

Firmware version 4.x: Device Revision = 4

Technical Specifications

Memory

TYPE	SIZE
Volatile Memory	8 Mbytes
Non-Volatile Memory *	4 Mbytes
EEPROM	1 Kbytes
Flash to the program	4 Mbytes
Flash to monitor	2 Mbytes

* It is kept by not rechargeable internal battery.

Battery

Type of battery	Battery Panasonic BR-2/3AE2SP - Lithium
Capacity	1200 mAh
Devices maintained by the battery	RTC and NVRAM
Minimum life span	8 years (typical charge of 17 μ A)
Maximum life span	49 years (typical charge of 2.8 μ A)
Voltage	3 V (subject to revision when below 2.5 V)

Communication Ports and Channels

ETHERNET PORT	
Communication Rate	10/100 Mbps
Standard	IEEE 802.3u
Isolation	150 Vrms
Operation Mode	Full duplex
Connector	RJ45 with shield*

* Grounded to the rail used for fixing the rack in which the DF75 is installed.

MODBUS PORT	
Communication Rate (Maximum)*	115200 bps
Standard	EIA-232
Connector**	RJ12 with shield
Maximum Current ***	0.5A @ 3.3V

*There is an increase in error rate as we increase the communication rate over 19200 bps. In many situations these errors can be acceptable, and they are not noticed by supervision.

** Grounded to the rail used for fixing the rack in which the DF75 is installed.

*** Internally protected by solid state fuse.

REDUNDANT PORT	
Maximum Communication Rate	115200 bps*
Standard	EIA-232
Connector	RJ12 with shield**
Maximum Current ***	0.5A @ 3.3 V

* Rate for control information. Data traffic through Ethernet.

** Grounded to the rail used for fixing the rack in which the DF75 is installed.

*** Internally protected by solid state fuse.

FAILURE RELAY	
Output Type	Solid state relay, normally closed (NC), isolated
Maximum Voltage	30 Vdc
Maximum Current	200 mA
Overload Protection	Not available. It must be provided externally
Normal Operation	Opened contacts
Failure Condition	Closed contacts
Maximum cable length (maximum) connected to the relay	30 m

Note: The power supply for the load must not be from an external network (outside the panel).

IMB BUS	
Voltage	5 Vdc
Bus	8 bits
Failure Signal	Yes
Hot Swap	Yes
Redundancy in the bus access	Yes, but only using the DF78 or DF92 rack

Module Characteristics

CONTROLLER	
CPU	Family ARM7TDMI
Bus	32 bits
Architecture	RISC
Performance	40 MIPS
CPU Cache	8 Kbytes
Clock	40 MHz
DMA	10 channels
Ethernet	MAC 10/100 integrated
Watchdog	Yes (200 ms of cycle)
Operation Voltage	3.3 V for I/O

MODULE	
Operation Voltage	5 V ($\pm 5\%$ of tolerance)
Typical Current	550 mA
Real Consumption	2.75 W
Environment Air Temperature	0 - 60° C (IEC 1131)
Storage Temperature	-20 - 80° C (IEC 1131)
Relative Air Humidity (Operation)	5% - 95% (non-condensing)
Cooling Mode	Air Convection
Dimensions (H x W x D) in mm	149 x 40 x 138 (without package)

Electrical Certification

DF75 follows the immunity test specification to equipment to industrial installation, as IEC61326:2002 standard.

ENCLOSE	
Electrostatic discharge (IEC61000-4-2)	4 kV/8 kV contact/air
EM field (IEC61000-4-3)	10 V/m
Rated power frequency magnet field (IEC61000-4-8)	30 A/m

AC POWER	
Voltage dip/short interruptions (IEC61000-4-11)	0.5 cycle, each polarity/100%
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

DC POWER	
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

I/O SIGNAL CONTROL	
Burst (IEC61000-4-4)	1 kV
Surge (IEC61000-4-5)	1 kV
Conducted RF (IEC61000-4-6)	3 V

I/O SIGNAL CONTROL CONNECTED TO POWER SUPPLY	
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

Emission Rate

ENCLOSE	
30 to 230 MHz (CISPR 16-1, CISPR 16-2)	40 dB (uV/m) quasi peak, measured at 10m distance
239 to 1000 MHz (CISPR 16-1, CISPR 16-2)	40 dB (uV/m) quasi peak, measured at 10m distance

AC MAINS	
0.15 to 0.5 MHz (CISPR 16-1, CISPR 16-2)	79 dB (uV) quasi peak 66 dB (uV) average
0.5 to 5 MHz (CISPR 16-1, CISPR 16-2)	73 dB (uV) quasi peak 60 dB (uV) average
5 to 30 MHz (CISPR 16-1, CISPR 16-2)	73 dB (uV) quasi peak 60 dB (uV) average

Indication LEDs

The LED names, colors, descriptions, and behaviors are showed in the table below.

LED	COLOR	DESCRIPTION	BEHAVIOR
+5V DC (ON)	Green	It indicates when the module is ON.	Solid green LED when power is on.
FAIL (FAIL)	Red	It indicates hardware failure.	Solid red LED when in fail.
RUN (RUN)	Green	It indicates when the controller is operating in normal mode.	Solid green LED when in operation.
HOLD (HLD)	Yellow	It indicates when the controller is in hold mode. In this mode the controller does not perform any application and does not interfere on the plant operation (access via I/O cards or via digital bus are disabled)	Solid yellow LED when the controller is in hold mode (HOLD).
FORCE (FRC)	Red	<p>1 – It indicates different modes of startup or maintenance requested by the operator via push-buttons (FACT INIT, HOLD and IP Address).</p> <p>2 – It indicates power failure when the operating voltage decays down the expected value of 4.8 V (low line).</p> <p>3 – Indicates a battery problem.</p>	<p>1 - Depending on the number of times that the right push-button is pressed, the FRC LED blinks at a given rate for a limited time, indicating the chosen mode (refer to Troubleshooting section for further information).</p> <p>2 – Permanently solid red. The module will reset if the voltage reaches 4.6V. (HLD and FAIL LEDs light up together temporarily).</p> <p>3 –FRC LED blinking and HLD LED solid red during the module start up – indicates low battery or the battery rear DIP switch is turned off (refer to Troubleshooting section for further information)</p>
232 TX	Green	It indicates activity in the RS-232 port.	Blinking green LED when the RS-232 port is transmitting data.
ETH1 LNK	Green	It indicates when the Ethernet link is active (ETH1 port).	Solid green LED when the Ethernet link is established (ETH1 port).
ETH1 TX	Green	It indicates communication activity in the ETH1 port	Blinking green LED when the ETH1 port is transmitting data.
ETH2 LNK	Green	It indicates when the Ethernet link is active (ETH2 port).	Solid green LED when the Ethernet link is established (ETH2 port).
ETH2 TX	Green	It indicates communication activity in the ETH2 port .	Blinking green LED when the ETH2 port is transmitting data.
STANDBY	Green	<p>If the HOLD LED is solid yellow and the STB LED is blinking, the firmware is being updating.</p> <p>If the HOLD LED is off, it indicates the controller role on redundancy as well as the synchronism status.</p>	There are several blinking patterns to indicate different synchronism status. Refer to redundancy section for further details.

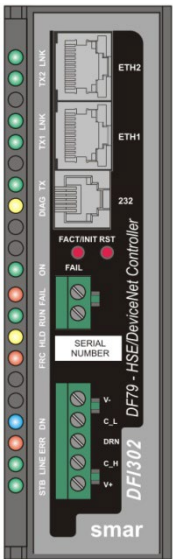
DF79 Specifications

Part Number

DF79 –HSE/DeviceNet Controller with 2 Ethernet ports e 1 DeviceNet channel

Description

DF79 module is the Smar solution for DeviceNet applications. Its main feature is working as DeviceNet-HSE gateway to provide power to the connectivity and flexibility to the system application. It allows wide communication between the DeviceNet devices. Through the HSE network and other DFI302 modules, it is possible the communication between field devices and other industrial protocols, providing greater flexibility to the control strategy projects. Through the I/O cards, it is also possible to execute discrete control via relay diagram logic ("Ladder Diagram"), allowing a single and integrated system. The module DF79 also can act as Modbus gateway, allowing the interconnection of modules that are not fieldbus or HSE.



DF79 – Controller module

OFF
ON

1	2	3	4	5
---	---	---	---	---

↔
Rear Dip Switch

	STORAGING	OPERATION
1 BATTERY	OFF	ON
2	OFF	OFF
3 SIMULATE	OFF	OFF
4 WATCHDOG	ON	ON
5	OFF	OFF

DF79

Characteristics and Module Limits

- One DeviceNet channel.
- Supports up to 63 nodes (addresses 0 to 63, is not recommended to use the address 63 to move devices in the network);
- Up to 2048 I/O discrete points and up to 512 I/O analog points in the DeviceNet network;
- Up to 1024 discrete points and 512 analog points with modules in the IMB (Conventional I/O);
- Limit of 64 external links by the HSE network
- Up to 16 server sessions and 16 client sessions
- Dynamic block instantiation
- One (1) Flexible Function Block with 242 linked parameters to interface between the discrete and continuous control
- It supports up to 16 HART modules (DF116/DF117).

NOTE

The limit of HART modules per controller is 16, but you must remember that when mapping the points through fieldbus function blocks strategy, one transducer block is used for each HART device. In this case, the system may be limited to less than 16 HART modules depending on the maximum number of fieldbus function blocks supported by the controller.

Continuous Control with DeviceNet

DF79 is a complete DeviceNet HSE controller with capacity to execute function blocks. Through the available **SYSTEM302**, **Studio302** and **Syscon** configuration tools is possible to configure the DF79 totally.

HSE Communication:

- Maximum of 512 link objects;
- Maximum of 250 function blocks;
- Up to 50 requests for non-connected services can be pendent per connection;
- Supervision up to 2000 points per second;
- Configurable Views.

Discrete Control

DF79 module also has the capability of access I/O cards through the IMB (Inter-Module Bus), present in the backplane where the DF79 is mounted. Through the IMB, up to 16 racks DF1A or DF93 can be interconnected, each one having up to 4 cards.

DISCRETE CONTROL CHARACTERISTICS

I/O Points*	Maximum of 2048 discrete points and 512 analog points in the DeviceNet network. Maximum of 1024 discrete points and 512 analog points with modules in the IMB.
Auxiliary Points	4096 points (maximum)
Ladder Function Blocks	1200 blocks (maximum)
Configuration File	120kbytes (maximum)**
Program Execution Cycle for 1000 boolean operations (without redundancy)	50 ms (minimum)*** 90 ms (typical)****
Execution Average Time	5.8 ms/Kbytes of program (minimum) 10.5 ms/Kbytes of program (typical)

* The whole number of points includes inputs and outputs, analog or digitals. Maximum may change according I/O type used.

** 120 Kbytes are available in firmware version 2.x and later. For earlier versions the limit is 60 Kbytes.

*** 1131 Flexible Function Block adjusted to One (High Priority). Each 1000 boolean operations allocate 8.6 Kbytes.

**** Total execution time will change depending on the adjusted priority of 1131 FFB. The adjustment should be compatible with the quantity of function blocks and HSE links.

Firmware version and Device Revision

Some Firmware Versions may change the Device Revision, so this is an important step to consider during the configuration procedure for the controller. The section "Adding Function Blocks" better describes the steps for this configuration. The current available versions are:

Firmware version 1.0: Device Revision = 1
 Firmware version 1.1: Device Revision = 2
 Firmware version 2.x: Device Revision = 3
 Firmware version 3.x: Device Revision = 4

Technical Specifications

Memory

TYPE	SIZE
Volatile Memory	8 Mbytes
Non Volatile Memory *	4 Mbytes
EEPROM	1 Kbytes
Flash to the program	4 Mbytes
Flash to monitor	2 Mbytes

Battery

Type of battery	Battery Panasonic BR-2/3AE2SP - Lithium
Capacity	1200 mAh
Devices maintained by the battery	RTC and NVRAM
Minimum life span	8 years (typical charge of 17 μ A)
Maximum life span	49 years (typical charge of 2.8 μ A)
Voltage	3 V (subject to revision when below 2.5 V)

Communication Ports and Channels

ETHERNET PORT	
Rate	10/100 Mbits/s
Standard	IEEE 802.3u
Isolation	150 Vrms
Operation mode	Full duplex
Connector	RJ45 with shield*

* Grounded to the rail used for fixing the rack in which the DF79 is installed

DEVICENET NETWORK	
Power Supply	24 Vdc \pm 20% Maximum consumption of module: 100 mA
Connector	Open style, from top to bottom: V-, CAN_L, DRN, CAN_H, V+ Isolation: 1000 Vrms
Communication Rates	125 kbps (trunk < 500 m) 250 kbps (trunk < 250 m) 500 kbps (trunk < 100 m)

MODBUS PORT	
Maximum rate*	115200 bps
Standard	EIA-232
Connector**	RJ12 with shield
Maximum Current***	0,5A @ 3.3V

*There is an increase in error rate as we increase the communication rate over 19200 bps. In many situations these errors can be acceptable and they are not noticed by supervision.

** Grounded to the rail used for fixing the rack in which the DF79 is installed

*** Internally protected by solid state fuse.

REDUNDANCY PORT (NOT OPERATIONAL - RESERVED)	
Maximum rate	115200 bps
Standard	EIA-232
Connector*	RJ12 with shield
Maximum Current**	0.5A @ 3.3V

* Grounded to the rail used for fixing the rack in which the DF79 is installed

** Internally protected by solid state fuse.

FAILURE RELAY	
Output Type	Solid state relay, normally closed (NC)
Maximum voltage	24 Vdc \pm 20%
Maximum Current	100 mA
Isolation	1000 Vrms
Overload Protection	Does not have. It must be provided externally
Normal Operation	Opened contacts
Failure Condition	Closed contacts
Maximum cable length connected to the relay	30 m

Observation: The power supply for the load must not be from an external network (outside the panel).

IMB BUS	
Voltage	5 Vdc
Bus	8 bits
Failure indication	Yes
Hot swap	Yes

Module Characteristics

CONTROLLER	
CPU	Family ARM7TDMI
Bus	32 bits
Architecture	RISC
Performance	40 MIPS
CPU Cache	8 Kbytes
Clock	40 MHz
DMA	10 channels
Ethernet	MAC 10/100 integrated
Watchdog	Yes (200 ms of cycle)
Power Supply Voltage	3.3 V for I/O and 2.5V for core (552 mW)

CARD	
Power Supply Voltage	5 V (\pm 10% of tolerance)
Typical Current	620 mA
Real consumption	2.75 W
Environment Air Temperature	0 to 60° C in compliance with the IEC 1131 standard
Storage Temperature	-20 to 80° C in compliance with the IEC 1131 standard
Relative Humidity of the Operation Air	5% to 95% non-condensing
Cooling Mode	Air convection
Weight	0.318 kg
Dimensions (H x W x D) in mm	149x40x138 (without enclose)

Electrical Certification

DF79 follows the immunity test specification to equipment to industrial installation, as IEC61326:2002 standard.

ENCLOSE	
Electrostatic discharge (IEC61000-4-2)	4 kV/8 kV contact/air
EM field (IEC61000-4-3)	10 V/m
Rated power frequency magnet field (IEC61000-4-8)	30 A/m

AC POWER	
Voltage dip/short interruptions (IEC61000-4-11)	0,5 cycle, each polarity/100%
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

DC POWER	
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

I/O SIGNAL/CONTROL	
Burst (IEC61000-4-4)	1 kV
Surge (IEC61000-4-5)	1 kV
Conducted RF (IEC61000-4-6)	3 V

I/O SIGNAL/CONTROL CONNECTED DIRECTLY TO POWER SUPPLY NETWORK	
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

Emission Rate

ENCLOSE	
30 to 230 MHz (CISPR 16-1, CISPR 16-2)	40 dB (uV/m) quasi peak, measured at 10m distance
239 to 1000 MHz (CISPR 16-1, CISPR 16-2)	40 dB (uV/m) quasi peak, measured at 10m distance

AC MAINS	
0,15 to 0,5 MHz (CISPR 16-1, CISPR 16-2)	79 dB (uV) quasi peak 66 dB (uV) average
0,5 to 5 MHz (CISPR 16-1, CISPR 16-2)	73 dB (uV) quasi peak 60 dB (uV) average
5 to 30 MHz (CISPR 16-1, CISPR 16-2)	73 dB (uV) quasi peak 60 dB (uV) average

Indication LEDs

The LED names, colors, descriptions, and behaviors are showed in the table below.

LED	COLOR	DESCRIPTION	BEHAVIOR
ETH LNK 2 (LNK)	Green	It indicates when the Ethernet link is active (ETH2 port).	Solid green LED when the Ethernet link is established (ETH2 port).
ETH TX2 (TX2)	Green	It indicates communication activity in the ETH2 port.	Blinking green LED when the ETH2 port is transmitting data.
ETH LNK 1 (LNK)	Green	It indicates when the Ethernet link is active (ETH1 port).	Solid green LED when the Ethernet link is established (ETH1 port).
ETH TX1 (TX1)	Green	It indicates communication activity in the ETH1 port.	Blinking green LED when the ETH1 port is transmitting data.
232 TX (TX)	Green	It indicates activity in the RS-232 port	Blinking green LED when the RS-232 port is transmitting data.
DIAG	Yellow	It indicates the operation mode in the RS-232 port.	On: RS232 port in local diagnostic mode. Off: RS232 port in normal operation mode (Modbus RTU) Blinking: NA
+5 VDC (ON)	Green	It indicates when the module is ON.	On: Module power supply is OK. Off: Without 5V of rack or hot swap circuit with problem. Blinking: NA
FAIL	Red	It indicates hardware failure.	Solid red LED when in fail
RUN	Green	It indicates when the controller is operating in normal mode.	Solid green LED when in operation.
HOLD (HLD)	Yellow	It indicates when the controller is in hold mode. In this mode the controller does not perform any application and does not interfere on the plant operation (access via I/O cards or via digital bus are disabled)	Solid yellow LED when the controller is in hold mode (HOLD).
FORCE (FRC)	Red	1 – It indicates different modes of startup or maintenance requested by the operator via push-buttons (FACT INIT, HOLD and IP Address). 2 – It indicates power failure when the operating voltage decays down the expected value of 4.8 V (low line). 3 – Indicates a battery problem.	1 - Depending on the number of times that the right push-button is pressed, the FRC LED blinks at a given rate for a limited time, indicating the chosen mode (refer to Troubleshooting section for further information). 2 – Permanently solid red. The module will reset if the voltage reaches 4.6V. (HLD and FAIL LEDs light up together temporarily). 3 –FRC LED blinking and HLD LED solid red during the module start up – indicates low battery or the battery rear DIP switch is turned off (refer to Troubleshooting section for further information)
DN	Blue	It indicates DeviceNet channel activity.	On: The DeviceNet communication was activated and all active devices in the configuration are working normally. Off: DeviceNet network was not configured. Blinking: Some active device in the configuration is missing or has a problem. Verify the device or deactivate it in the configuration (a download is necessary).
ERR	Red	It indicates it there is an error related to the DeviceNet network.	On: Error in the DeviceNet network. The 24V is missing, some node is missing or with error. Off: The DeviceNet network is configured and all active devices are working normally. Blinking: The DeviceNet communication was deactivated.
LINE	Green	It indicates that the bus is powered with 24 V.	On: Powered bus. Off: Bus is not powered Blinking: NA
STANDBY (STB)	Green	If the HOLD LED is solid yellow and the STB LED is blinking, the firmware is being updating. If the HOLD LED is off, it indicates the controller role on redundancy as well as the synchronism status.	There are several blinking patterns to indicate different synchronism status. Refer to redundancy section for further details.

*NA – Not applicable

DF81 Specifications

Part Number

DF81 – HSE/AS-i Controller with 2 Ethernet 100 Mbps ports and 2 AS-I channel.

Description

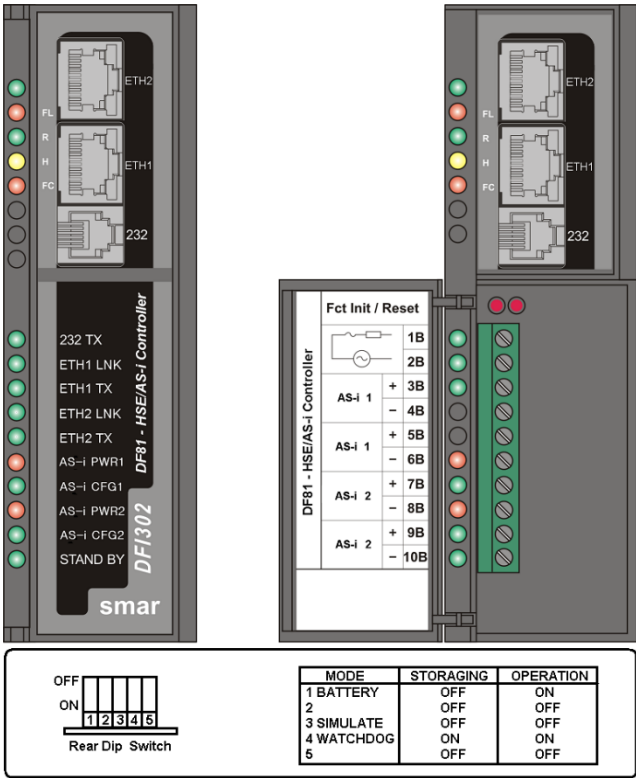
The HSE/AS-I controller follow the operation standard AS-i network and meets AS-i master requirements based on the version 2.1 (except analogical data treatment and of the flag Date Exchange of the masters).

DF81 meets the same quality principles and standards DFI302 family. It is composed by 2 AS-i bus (2 masters), 2 Ethernet 10/100 Mbps ports, 2 RS-232 serial ports (one for communication and other for synchronism) and ability for execution FOUNDATION™ fieldbus blocks.

Through the HSE network and other DFI302 modules, it is possible the communication between field devices and other industrial protocols, providing greater flexibility to the control strategy projects.

Through the I/O cards, it is also possible to execute discrete control via relay diagram logic ("Ladder Diagram"), allowing a single and integrated system.

The module DF81 also can act as Modbus gateway, allowing the interconnection of Modbus masters, network diagnostic by SNMP protocol (Simple Network Management Protocol), and AS-i network configuration by proprietary tool (**SMAR Network Configurator - SmarNetConf**).



DF81 – HSE/AS-i Controller

Characteristics and Module Limits

- 2 AS-i channel (2 masters) – version 2.1 (only discrete);
- 2 Ethernet 10/100 Mbps ports;
- Supports up to 124 slaves AS-i discrete (62 for channel – version 2.1);
- Access I/O modules;
- Up to 1024 discrete points and 512 analog points with modules in the IMB (conventional I/O);

- Limit of 128 external links by the HSE network (64 VCRs *publishers* and 64 VCRs *subscribers*);
- Dynamic block instantiation;
- FOUNDATION™ fieldbus function blocks;
- One (1) Flexible Function Block with 242 linked parameter to interface between the discrete and continuous control;
- One (1) Transducer Block for AS-i network diagnostic;
- Integrated Modbus protocol;
- Integrated Webserver;
- Simple Network Management Protocol (SNMP);
- It supports up to 16 HART modules (DF116/DF117).

NOTE

The limit of HART modules per controller is 16, but you must remember that when mapping the points through fieldbus function blocks strategy, one transducer block is used for each HART device. In this case, the system may be limited to less than 16 HART modules depending on the maximum number of fieldbus function blocks supported by the controller.

Continuous Control with AS-i

DF81 is a complete DeviceNet HSE controller with capacity to execute function blocks. Through the available **SYSTEM302**, **Studio302** and **Syscon** configuration tools is possible to configure the DF81 totally.

HSE Communication

- Maximum of 512 link objects;
- Maximum of 250 function blocks;
- Up to 50 requests for non-connected services can be pendent per connection;
- Supervision up to 2000 points per second;
- Configurable Views.

Discrete Control

DF81 module also has the capability of access up to 124 discrete AS-i slaves, being 62 per channel (version 2.1). According to the version 2.1 specification AS-i system, the scan time of the network devices is up to 10 ms per channel.

Besides discrete access of the AS-i devices, this module has ability to access I/O cards through IMB bus (Inter-Module Bus), present in the backplane where the DF81 is mounted.

Through IMB, up to 16 DF1A racks or DF93 can be interconnect, each one containing up to 4 cards. It is recommended, always, the use of bus terminator (BT302) for the last rack. In the AS-i bus it is not necessary to use terminators.

DISCRETE CONTROL CHARACTERISTICS

I/O Points*	Maximum 868 discrete points on the AS-i network (434 per channel). Maximum 1024 discrete points and 512 analog points with modules in the IMB.
Auxiliary Points	Maximum 4096 points
Function Blocks for Ladder	Maximum 1200 blocks
Configuration File	Maximum 120 Kbytes
Program Execution Cycle for 1000 boolean operations (without redundancy)	50 ms (minimum)** 90 ms (typical)***
Execution Average Time	5.8 ms/Kbytes of program (minimum) 10.5 ms/Kbytes of program (typical)

* The whole number of points includes inputs and outputs, analog or digitals. Maximum may change according to I/O type used.

** 1131 Flexible Function Block adjusted to One (High Priority). Each 1000 boolean operations allocate 8.6 Kbytes.

*** Total execution time will change depending on the adjusted priority of 1131 FFB. The adjustment should be compatible with the quantity of function blocks and HSE links.

Flexible Function Block Usage

The interconnection between the continuous control and the ladder logic can be done by using Flexible Function Block. For further details about the Flexible Function Block usage, refer to the section Adding logic by using Flexible Function Block and the **LogicView for FFB** manual.

Firmware version and Device Revision

Some Firmware Versions may change the Device Revision, so this is an important step to consider during the configuration procedure for the controller. The section “Adding Function Blocks” better describes the steps for this configuration. The current available versions are:

Firmware version 1.x: Device Revision = 01
Firmware version 2_0_2: Device Revision = 02

Technical Specifications

Memory

TYPE	SIZE
Volatile Memory	8 Mbytes
Non Volatile Memory	4 Mbytes
EEPROM	1 Kbytes
Flash to the program	4 Mbytes
Flash to monitor	2 Mbytes

Battery

CHARACTERISTICS	
Type of battery	Battery Panasonic BR-2/3AE2SP - Lithium
Capacity	1200 mAh
Devices maintained by the battery	RTC and NVRAM
Minimum life span	8 years (typical charge of 17 μ A)
Maximum life span	49 years (typical charge of 2.8 μ A)
Voltage	3 V (subject to revision when below 2.5 V)

Communication Ports and Channels

ETHERNET PORT	
Communication Rate	10/100 Mbps
Standard	IEEE 802.3u
Isolation	150 Vrms
Operation Mode	Full duplex
Connector	RJ45 with shield*

* Grounded to the rail used for fixing the rack in which the DF81 is installed.

AS-i CHANNEL	
Number of H1 Channels	2
Communication Rate	167 kbps
Standard	EN 50295 / IEC62026-2
Modem AS-i (Physical Layer)	A2SI (ZMD)
Maximum Voltage Supplied for Channel	24 Vdc, 8A
Maximum Consumption of Bus for Channel	60 mA
Level in Industrial Communication	Sensor-Actuator (bit oriented)
Procedure of Bus Access	Master-Slave through cyclic polling

MODBUS PORT	
Communication Rate (Maximum)*	115200 bps
Standard	EIA-232
Connector**	RJ12 with shield
Maximum Current ***	0.5A @ 3.3V

*There is an increase in error rate as we increase the communication rate over 19200 bps. In many situations these errors can be acceptable, and they are not noticed by supervision.

** Grounded to the rail used for fixing the rack in which the DF81 is installed.

*** Internally protected by solid state fuse.

REDUNDANT PORT (NON OPERATING – FUTURE USE)	
Maximum Communication Rate	115200 bps
Standard	EIA-232
Connector*	RJ12 with shield
Maximum Current **	0.5A @ 3.3V

* Grounded to the rail used for fixing the rack in which the DF81 is installed.

** Internally protected by solid state fuse.

FAILURE RELAY	
Output Type	Solid state relay, normally closed (NC), isolated
Maximum Voltage	30 Vdc
Maximum Current	200 mA
Overload Protection	Not available. It must be provided externally
Normal Operation	Opened contacts
Failure Condition	Closed contacts
Maximum cable length (maximum) connected to the relay	30 m

Note: The power supply for the load must not be from an external network (outside the panel).

IMB BUS	
Voltage	5 Vdc
Bus	8 bits
Access time for writing and reading	450 ns
Failure Signal	Yes
Hot Swap	Yes

Module Features

CONTROLLER	
CPU	Family ARM7TDMI
Bus	32 bits
Architecture	RISC
Performance	40 MIPS
CPU Cache	8 Kbytes
Clock	40 MHz
DMA	10 channels
Ethernet	MAC 10/100 integrated
Watchdog	Yes (200 ms of cycle)
Operation Voltage	3.3 V for I/O

I/O MODULES	
Operation Voltage	5 V (\pm 5% of tolerance)
Typical Current	550 mA
Real Consumption	2,75 W
Environment Air Temperature	0 - 60° C (IEC 1131)
Storage Temperature	-20 - 80° C (IEC 1131)
Relative Air Humidity (Operation)	5% - 95% (non-condensing)
Cooling Mode	Air Convection
Dimensions (H x W x D) in mm	149 x 40 x 138 (without package)

General Characteristics of the Network

TOPOLOGY	
Maximum Number of Participants - version 2.1	2 Masters; 62 Discrete Slaves for Master
Maximum Number of Participants - version 2.0	2 Masters; 31 Discrete Slaves for Master
Maximum distance between Master-Slave	100 m; 300 m with repeater
Maximum distance between Master-Slave	100 m; 500 m with repeater
Network structures allowed	Tree, star, linear

AS-i CABLES	
Number of conductors for data and power	2 (yellow cable)
Cable Type	Non-interlaced, without loop ground, without terminators (AS-i cables or 2x1,5 mm ² parallel)
Secondary Supply Cable	Black Flexible Cable

ENVIRONMENTAL CONDITIONS	
Ambient Temperature	0 to 60 °C
Ingress Protection	Standard up to IP 20
EMC (Emission)	Up to Class A

Electrical Certification

DF81 follows the immunity test specification to equipment to industrial installation, as IEC61326:2002 standard.

ENCLOSE	
Electrostatic discharge (IEC61000-42)	4 kV/8 kV contact/air
EM field	10 V/m
Rated power frequency magnet field (IEC61000-4-8)	30 A/m

AC POWER	
Voltage dip/short interruptions (IEC61000-4-11)	0,5 cycle, each polarity/100%
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

DC POWER	
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

I/O SIGNAL/CONTROL	
Burst (IEC61000-4-4)	1 kV
Surge (IEC61000-4-5)	1 kV
Conducted RF (IEC61000-4-6)	3 V

I/O SIGNAL/CONTROL CONNECTED DIRECTLY TO POWER SUPPLY NETWORK	
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

Emission Rate

ENCLOSE	
30 to 230 MHz (CISPR 16-1, CISPR 16-2)	40 dB (uV/m) quasi peak, measured at 10m distance
239 to 1000 MHz (CISPR 16-1, CISPR 16-2)	40 dB (uV/m) quasi peak, measured at 10m distance

AC MAINS	
0,15 to 0,5 MHz (CISPR 16-1, CISPR 16-2)	79 dB (uV) quasi peak 66 dB (uV) average
0,5 to 5 MHz (CISPR 16-1, CISPR 16-2)	73 dB (uV) quasi peak 60 dB (uV) average
5 to 30 MHz (CISPR 16-1, CISPR 16-2)	73 dB (uV) quasi peak 60 dB (uV) average

Indication LEDs

The LED names, colors, descriptions, and behaviors are showed in the table below.

LED	COLOR	DESCRIPTION	BEHAVIOR
+5 Vdc (ON)	Green	It indicates when the module is on.	Solid green LED when power is on.
FAIL (FL)	Red	It indicates hardware failure.	Solid red LED when in fail.
RUN (R)	Green	It indicates when the controller is operating in normal mode.	Solid green LED when in operation.
HOLD (H)	Yellow	It indicates when the controller is in hold mode. In this mode the controller does not perform any application and does not interfere on the plant operation (access via I/O cards or via digital bus are disabled)	Solid yellow LED when the controller is in hold mode (HOLD).
FORCE (FC)	Red	<p>1 – It indicates different modes of startup or maintenance requested by the operator via push-buttons (FACT INIT, HOLD and IP Address).</p> <p>2 – It indicates power failure when the operating voltage decays down the expected value of 4.8 V (low line).</p> <p>3 – Indicates a battery problem.</p>	<p>1 - Depending on the number of times that the right push-button is pressed, the FRC LED blinks at a given rate for a limited time, indicating the chosen mode (refer to Troubleshooting section for further information).</p> <p>2 – Permanently solid red. The module will reset if the voltage reaches 4.6V. (HLD and FAIL LEDs light up together temporarily).</p> <p>3 –FRC LED blinking and HLD LED solid red during the module start up – indicates low battery or the battery rear DIP switch is turned off (refer to Troubleshooting section for further information)</p>
232 TX (TX)	Green	It indicates activity in the RS-232 port.	Blinking green LED when the RS-232 port is transmitting data.
ETH1 LNK	Green	It indicates when the Ethernet link is active (ETH1 port).	Solid green LED when the Ethernet link is established (ETH1 port).
ETH1 TX	Green	It indicates communication activity in the ETH1 port	Blinking green LED when the ETH1 port is transmitting data.
ETH2 LNK	Green	It indicates when the Ethernet link is active (ETH2 port).	Solid green LED when the Ethernet link is established (ETH2 port).
ETH2 TX	Green	It indicates communication activity in the ETH2 port.	Blinking green LED when the ETH2 port is transmitting data.
STANDBY	Green	<p>If the HOLD LED is solid yellow and the STB LED is blinking, the firmware is being updating.</p> <p>If the HOLD LED is off, it indicates the controller role on redundancy as well as the synchronism status.</p>	There are several blinking patterns to indicate different synchronism status. Refer to redundancy section for further details.

LEDs connected to the AS-i bus:

LED	COLOR/STATE		DESCRIPTION	BEHAVIOR
AS-i PWR1 AS-i CFG1 AS-i PWR2 AS-i CFG2	Red	Green		
	On	Off	It indicates supply failure (<i>Power Fail</i>)	LED: red on and green off
	Blinking Quickly	Off	Channel powered (<i>Power On</i>) and masters not running (<i>Not ready</i>)	Blinking Quickly and pre-set intervals of red and green LED off
	Off	Blinking Quickly	Channel powered (<i>Power On</i>), there is not power failure, but the state master in Offline state	Red LED Blinking Quickly off and on pre-set intervals of green LED
	Off	Blinking Slowly	It indicates when the NORMAL_OPERATION flag is off OR its CONFIG_OK flag is not active OR the master is in configuration mode	Red LED off and green blinking slowly
	Off	On	It indicates normal operation, CONFIG_OK flag is active and the master is in protected mode	Red LED off and green on

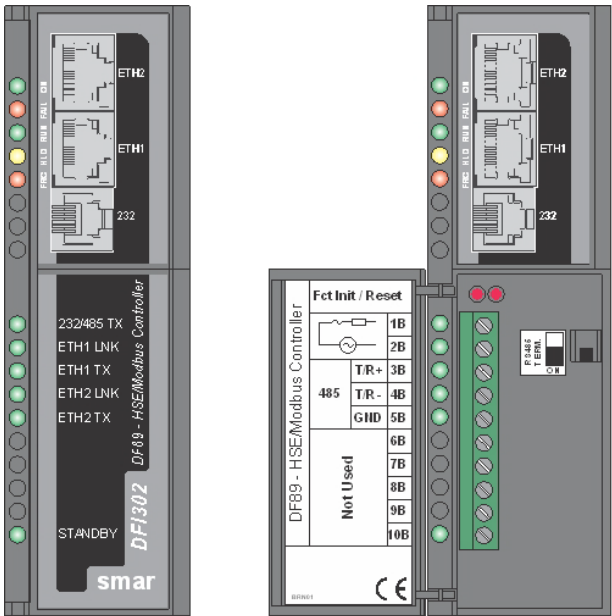
DF89 Specifications

Part Number

DF89 –HSE/Modbus Controller

Description

DF89 module is the Smar solution for Modbus applications in the **SYSTEM302**. Its main feature is working as Modbus-HSE controller to provide power to the connectivity and flexibility to the system application. Through the HSE network and other DFI302 modules, it is possible the communication between field devices and other industrial protocols, providing greater flexibility to the control strategy projects. Through the I/O cards, it is also possible to execute discrete control via relay diagram logic (“Ladder Diagram”), allowing a single and integrated system.



DF89 – HSE/Modbus Controller

OFF ON

1 2 3 4 5

Rear Dip Switch

	STORAGING	OPERATION
1 BATTERY	OFF	ON
2	OFF	OFF
3 SIMULATE	OFF	OFF
4 WATCHDOG	ON	ON
5	OFF	OFF

DF89

Characteristics and Controller Limits

- 2 10/100 Mbps Ethernet Ports;
- 1 RS-232/RS-485 Modbus RTU port (See NOTE DF89);
- Support up to 100 FOUNDATION™ fieldbus function blocks;
- Support Flexible Function Block (FFB);
- Discrete control via relay diagram;
- Access to I/O modules;
- Webserver;
- Modbus master and slave;
- Redundant operation;
- Real Time Clock (RTC) and watchdog;
- Supervision for up to 2000 points per second.
- It supports up to 16 HART modules (DF116/DF117).

NOTE

The limit of HART modules per controller is 16, but you must remember that when mapping the points through fieldbus function blocks strategy, one transducer block is used for each HART device. In this case, the system may be limited to less than 16 HART modules depending on the maximum number of fieldbus function blocks supported by the controller.

Continuous Control with FOUNDATION™ Fieldbus

The DF89 module is a HSE device, with block execution capability. It has up to 100 blocks, including a Flexible Function Block (FFB) to link FOUNDATION fieldbus control strategies with Ladder diagrams. Through configuration tools available in the **SYSTEM302** it is possible to configure the DF89 completely.

HSE Communication

- Maximum of 512 link objects;
- Limit of 128 linked parameters
- Dynamic block instantiation. Maximum of 100 function blocks;
- Support for Flexible Function Block, with 242 parameters which can be linked to interface between the discrete and continuous control.

Discrete Control

DF89 module also has the capability of access I/O cards through the IMB (Inter-Module Bus), present in the backplane where the DF89 is mounted. Through the IMB, up to 16 racks can be interconnected, each one having up to 4 cards. If there is a redundant controller is necessary, the use of rack DF78 or DF92. If DF78 is used plus 16 racks DF1A can be added. If DF92 is used plus 16 racks DF93 can be used. Additional power supplies in other racks can be necessary depending on the load of the cards.

DISCRETE CONTROL CHARACTERISTICS	
I/O Points*	512 discrete points or 256 analog (maximum)
Auxiliary Points	4096 points (maximum)
Ladder Function Blocks	1200 blocks (maximum)
Configuration File	60 Kbytes (maximum)
Program Execution Cycle for 1000 boolean operations (without redundancy)	50 ms (minimum)** 90 ms (typical)***
Program Execution Cycle with redundancy	Increment of 10ms (typical)**** up to 50 ms (maximum) to execution cycle
Execution Average Time	5.8 ms/Kbytes of program (minimum) 10.5 ms/Kbytes of program (typical)
Modbus	Modbus RTU Slave in the RS-232/RS-485 serial port and Modbus TCP Slave in the Ethernet port.
TCP/RTU Modbus Master (See NOTE DF89)	Up to 1024 discrete points and 512 analog. Up to 32 serial Modbus devices and 32 TCP Modbus. Up to 12 Modbus TCP connections.

* The whole number of points includes inputs and outputs, analog or digitals. Maximum quantity may change according I/O type used.

** Priority of 1131 Flexible Function Block adjusted to Zero (Very High Priority) and no other function blocks and HSE links are configured. Each 1000 boolean operations allocate 8.6 Kbytes.

*** Total execution time will change depending on the adjusted priority of 1131 FFB. The adjustment should be compatible with the quantity of function blocks and HSE links.

**** The whole execution time may change depending of the configuration file size.

NOTE DF89

- The characteristic of the DF89 as Modbus master described in “Creating a Modbus configuration using the DF89” section is only available from firmware version 4 (V4.xx) with Device Description 4.xx. Earlier versions of firmware and DD only support the features of slave Modbus described in “Adding Modbus” section.
- The DF89 has only RS-485 port from the serial number SN623. Modules with serial numbers less than SN623 support the version 4 of firmware and Modbus master and slave features described in “Creating a Modbus configuration using the DF89” section only for the RS-232 serial port.

Flexible Function Block Usage

The interconnection between the discrete control and the ladder logic can be done by using Flexible Function Block. For further details about the Flexible Function Block usage, refer to the section Adding logic by using Flexible Function Block and the LogicView for FFB manual.

Firmware version and Device Revision

Some firmware versions may change the Device Revision, so this is an important step to consider during the configuration procedure for the controller. The section “Adding Function Blocks” better describes the steps for this configuration. The current available versions are:

Firmware version 1.x: Device Revision = 1 (Only Modbus slave)

Firmware version 2.x: Device Revision = 2 (Only Modbus slave)

Firmware version 3.x: Device Revision = 3 (Only Modbus slave)

Firmware version 4.x: Device Revision = 4 (Modbus master and slave)

Technical Specifications

Memory

TYPE	SIZE
Volatile Memory	8 Mbytes
Non-Volatile Memory *	4 Mbytes
EEPROM	1 Kbytes
Flash to the program	4 Mbytes
Flash to monitor	2 Mbytes

* It is kept by not rechargeable internal battery.

Battery

Type of battery	Battery Panasonic BR-2/3AE2SP - Lithium
Capacity	1200 mAh
Devices maintained by the battery	RTC and NVRAM
Minimum life span	8 years (typical charge of 17 μ A)
Maximum life span	49 years (typical charge of 2.8 μ A)
Voltage	3 V (subject to revision when below 2.5 V)

Communication Ports and Channels

ETHERNET PORT	
Communication Rate	10/100 Mbps
Standard	IEEE 802.3u
Isolation	150 Vrms
Operation Mode	Full duplex
Connector	RJ45 with shield*

* Grounded to the rail used for fixing the rack in which the DF89 is installed.

MODBUS PORT (SEE NOTE DF89)	
Communication Rate (Maximum)*	115200 bps
Standard	RS-232 and RS-485 (one port with two physical media)
RS-232 Connector**	RJ12 with shield
RS-485 Connector	3-pin terminal block
Maximum Current ***	0.5A @ 3.3V

*There is an increase in error rate as we increase the communication rate over 19200 bps. In many situations these errors can be acceptable, and they are not noticed by supervision.

** Grounded to the rail used for fixing the rack in which the DF89 is installed.

*** Internally protected by solid state fuse.

REDUNDANT PORT	
Maximum Communication Rate	115200 bps*
Standard	RS-232
Connector	RJ12 with shield**
Maximum Current ***	0.5A @ 3.3 V

* Rate for control information. Data traffic through Ethernet.

** Grounded to the rail used for fixing the rack in which the DF89 is installed.

*** Internally protected by solid state fuse.

FAILURE RELAY	
Output Type	Solid state relay, normally closed (NC), isolated
Maximum Voltage	30 Vdc
Maximum Current	200 mA
Overload Protection	Not available. It must be provided externally
Normal Operation	Opened contacts
Failure Condition	Closed contacts
Maximum cable length (maximum) connected to the relay	30 m

The power supply for the load must not be from an external network (outside the panel).

IMB BUS	
Voltage	5 Vdc
Bus	8 bits
Failure Signal	Yes
Hot Swap	Yes
Redundancy in the bus access	Yes, but only using the DF78 or DF92 rack

Module Characteristics

CONTROLLER	
CPU	Family ARM7TDMI
Bus	32 bits
Architecture	RISC
Performance	40 MIPS
CPU Cache	8 Kbytes
Clock	40 MHz
DMA	10 channels
Ethernet	MAC 10/100 integrated
Watchdog	Yes (200 ms of cycle)
Operation Voltage	3.3 V for I/O

MODULE	
Operation Voltage	5 V ($\pm 5\%$ of tolerance)
Typical Current	550 mA
Real Consumption	2.75 W
Environment Air Temperature	0 – 60 °C (IEC 1131)
Storage Temperature	-20 – 80 °C (IEC 1131)
Relative Air Humidity (Operation)	5% - 95% (non-condensing)
Cooling Mode	Air Convection
Dimensions (H x W x D) in mm	149 x 40 x 138 (without package)

Electrical Certification

DF89 follows the immunity test specification to equipment to industrial installation, as IEC61326:2002 standard.

ENCLOSE	
Electrostatic discharge (IEC61000-4-2)	4 kV/8 kV contact/air
EM field (IEC61000-4-3)	10 V/m
Rated power frequency magnet field (IEC61000-4-8)	30 A/m

AC POWER	
Voltage dip/short interruptions (IEC61000-4-11)	0.5 cycle, each polarity/100%
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

DC POWER	
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

I/O SIGNAL CONTROL	
Burst (IEC61000-4-4)	1 kV
Surge (IEC61000-4-5)	1 kV
Conducted RF (IEC61000-4-6)	3 V

I/O SIGNAL CONTROL CONNECTED TO POWER SUPPLY	
Burst (IEC61000-4-4)	2 kV
Surge (IEC61000-4-5)	1 kV/2 kV
Conducted RF (IEC61000-4-6)	3 V

Emission Rate

ENCLOSE	
30 to 230 MHz (CISPR 16-1, CISPR 16-2)	40 dB ($\mu\text{V}/\text{m}$) quasi peak, measured at 10m distance
239 to 1000 MHz (CISPR 16-1, CISPR 16-2)	40 dB ($\mu\text{V}/\text{m}$) quasi peak, measured at 10m distance

AC MAINS	
0.15 to 0.5 MHz (CISPR 16-1, CISPR 16-2)	79 dB (uV) quasi peak 66 dB (uV) average
0.5 to 5 MHz (CISPR 16-1, CISPR 16-2)	73 dB (uV) quasi peak 60 dB (uV) average
5 to 30 MHz (CISPR 16-1, CISPR 16-2)	73 dB (uV) quasi peak 60 dB (uV) average

Indication LEDs

The LED names, colors, descriptions, and behaviors are showed in the table below.

LED	COLOR	DESCRIPTION	BEHAVIOR
+5V DC (ON)	Green	It indicates when the module is ON.	Solid green LED when power is on.
FAIL (FAIL)	Red	It indicates hardware failure.	Solid red LED when in fail.
RUN (RUN)	Green	It indicates when the controller is operating in normal mode.	Solid green LED when in operation.
HOLD (HLD)	Yellow	It indicates when the controller is in hold mode. In this mode the controller does not perform any application and does not interfere on the plant operation (access via I/O cards or via digital bus are disabled)	Solid yellow LED when the controller is in hold mode (HOLD).
FORCE (FRC)	Red	1 – It indicates different modes of startup or maintenance requested by the operator via push-buttons (FACT INIT, HOLD and IP Address). 2 – It indicates power failure when the operating voltage decays down the expected value of 4.8 V (low line). 3 – Indicates a battery problem.	1 - Depending on the number of times that the right push-button is pressed, the FRC LED blinks at a given rate for a limited time, indicating the chosen mode (refer to Troubleshooting section for further information). 2 – Permanently solid red. The module will reset if the voltage reaches 4.6V. (HLD and FAIL LEDs light up together temporarily). 3 –FRC LED blinking and HLD LED solid red during the module start up – indicates low battery or the battery rear DIP switch is turned off (refer to Troubleshooting section for further information)
232 TX	Green	It indicates activity in the RS-232 port.	Blinking green LED when the RS-232 port is transmitting data.
ETH1 LNK	Green	It indicates when the Ethernet link is active (ETH1 port).	Solid green LED when the Ethernet link is established (ETH1 port).
ETH1 TX	Green	It indicates communication activity in the ETH1 port	Blinking green LED when the ETH1 port is transmitting data.
ETH2 LNK	Green	It indicates when the Ethernet link is active (ETH2 port).	Solid green LED when the Ethernet link is established (ETH2 port).
ETH2 TX	Green	It indicates communication activity in the ETH2 port.	Blinking green LED when the ETH2 port is transmitting data.
STANDBY	Green	If the HOLD LED is solid yellow and the STB LED is blinking, the firmware is being updating. If the HOLD LED is off, it indicates the controller role on redundancy as well as the synchronism status.	There are several blinking patterns to indicate different synchronism status. Refer to redundancy section for further details.

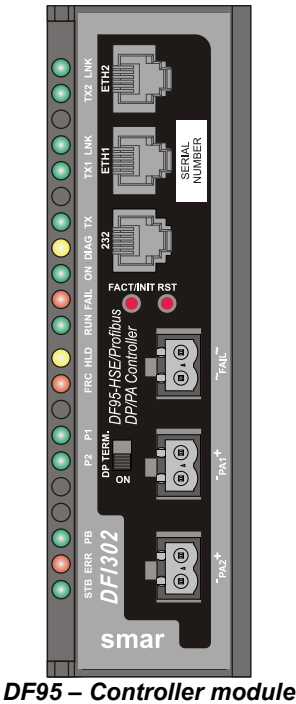
DF95 Specifications

Part Number

DF95 –HSE/Profibus DP Controller with 2 Ethernet 100 Mbps ports, 1 serial port, 2 Profibus PA ports and 1 Profibus DP channel

Description

DF95 module is the Smar solution for Profibus applications Its main feature is working as Profibus DP-HSE gateway to provide power to the connectivity and flexibility to the system application. It allows wide communication between the Profibus DP and PA field devices. Through the HSE network and other DFI302 modules, it is possible the communication between field devices and other industrial protocols, providing greater flexibility to the control strategy projects. Through the I/O cards, it is also possible to execute discrete control via relay diagram logic ("Ladder Diagram"), allowing a single and integrated system. The module DF95 also can act as Modbus gateway (slave), allowing the interconnection of modules that are not fieldbus or HSE.



DF95 – Controller module

OFF

ON

1

2

3

4

5

Rear Dip Switch

	STORAGING	OPERATION
1 BATTERY	OFF	ON
2	OFF	OFF
3 SIMULATE	OFF	OFF
4 WATCHDOG	ON	ON
5	OFF	OFF

DF95

Characteristics and Module Limits

- One Profibus DP channel supporting up to 12 Mbps;
- Two Profibus PA ports supporting up to 32 devices for channel;
- It supports up to 124 Profibus DP and PA field devices;
- It supports up to 3584 bytes of input and 3584 bytes of output during the data interchange process;
- Limit of 64 external links by the HSE network;
- Maximum of 250 function blocks per DF95;
- One (1) Flexible Function Block (counted into the 250 allowed blocks), with 242 linked parameters to interface between the discrete and continuous control.
- It supports up to 16 HART modules (DF116/DF117).

NOTE

The limit of HART modules per controller is 16, but you must remember that when mapping the points through fieldbus function blocks strategy, one transducer block is used for each HART device. In this case, the system may be limited to less than 16 HART modules depending on the maximum number of fieldbus function blocks supported by the controller.

Continuous Control with Profibus

DF95 is a complete Profibus HSE controller with capacity to execute function blocks. Through the available **SYSTEM302**, **Studio302** and **Syscon** configuration tools is possible to configure the DF95 totally.

HSE Communication:

- Maximum of 512 link objects;
- Supervision up to 2000 points per second;

Discrete Control

DF95 module also has the capability of access I/O cards through the IMB (Inter-Module Bus), present in the backplane where the DF95 is mounted. Through the IMB, up to 16 racks DF1A or DF93 can be interconnected, each one having up to 4 cards.

DISCRETE CONTROL CHARACTERISTICS

I/O Points*	1024 discrete points or 512 analog (maximum)
Ladder Function Blocks	2000 blocks (maximum)
Configuration File	120 Kbytes (maximum)
Program Execution Cycle for 1000 boolean operations (without redundancy)	50 ms (minimum)** 90 ms (typical)***
Program Execution Cycle with redundancy	Increment of 10ms (typical)**** up to 50 ms (maximum) to execution cycle
Execution Average Time	5.8 ms/Kbytes of program (minimum) 10.5 ms/Kbytes of program (typical)

* The whole number of points includes inputs and outputs, analog or digitals. Maximum may change according to I/O type used.

** 1131 Flexible Function Block adjusted to One (High Priority). Each 1000 boolean operations allocate 8.6 Kbytes.

*** Total execution time will change depending on the adjusted priority of 1131 FFB. The adjustment should be compatible with the quantity of function blocks and HSE links.

**** The whole execution time may change depending of the configuration file size.

NUMBER OF I/O POINTS

Number of Profibus devices per network	125
Number of virtual points	4096
Number of Profibus discrete points	2048
Number of Profibus analog points	512

Firmware version and Device Revision

Some Firmware Versions may change the Device Revision, so this is an important step to consider during the configuration procedure for the controller. The section "Adding Function Blocks" better describes the steps for this configuration. The current available versions are:

Firmware version 2_0_X: Device Revision = 1

Firmware version 3_0_X: Device Revision = 2

Firmware version 4_0_X: Device Revision = 3

Technical Specifications

Memory

TYPE	SIZE
Volatile Memory	8 Mbytes
Non-Volatile Memory *	4 Mbytes
EEPROM	1 Kbytes
Flash to the program	4 Mbytes
Flash to monitor	2 Mbytes
Flash to EC1 (Profibus)	4 Mbytes

Battery

Type of battery	Battery Panasonic BR-2/3AE2SP - Lithium
Capacity	1200 mAh
Devices maintained by the battery	RTC and NVRAM
Minimum life span	8 years (typical charge of 17 μ A)
Maximum life span	49 years (typical charge of 2.8 μ A)
Voltage	3 V (subject to revision when below 2.5 V)

Communication Ports and Channels

ETHERNET PORT	
Rate	10/100 Mbits/s
Standard	IEEE 802.3u
Isolation	150 Vrms
Operation mode	Full duplex
Connector	RJ45 with shield*

* Grounded to the rail used for fixing the rack in which the DF95 is installed

DP PROFIBUS CHANNEL	
Rate	From 9.6 Kbit/s to 12 Mbits/s
Standard	EN 50170 e EN 50254
Physical Layer	EIA RS-485
Profibus Modem	EC1 (Hilscher)
Connector	M12

PROFIBUS PA CHANNELS	
Number of H1 Channels	2
Communication Rate	31.25 kbps
Standard	EN 61158 EN 50170
Physical Layer	ISA-S50.02-1992
MAU Type	Passive (bus not powered)
Isolation	500 Vac

MODBUS PORT	
Maximum rate *	115200 bps
Standard	EIA-232
Connector**	RJ12 with shield
Maximum Current***	0,5A @ 3.3V

*There is an increase in error rate as we increase the communication rate over 19200 bps. In many situations these errors can be acceptable, and they are not noticed by supervision.

** Grounded to the rail used for fixing the rack in which the DF95 is installed

*** Internally protected by solid state fuse.

REDUNDANCY PORT	
Maximum rate	115200 bps
Standard	EIA-232
Connector*	RJ12 with shield
Maximum Current**	0.5A @ 3.3V

* Grounded to the rail used for fixing the rack in which the DF95 is installed

** Internally protected by solid state fuse.

FAILURE RELAY	
Output Type	Solid state relay, normally closed (NC), isolated
Maximum voltage	30 Vdc
Maximum Current	200 mA
Overload Protection	Does not have. It must be provided externally
Normal Operation	Opened contacts
Failure Condition	Closed contacts
Maximum cable length connected to the relay	30 m

Observation: The power supply for the load must not be from an external network (outside the panel).

IMB BUS	
Voltage	5 Vdc
Bus	8 bits
Failure indication	Yes
Hot swap	Yes

Module Characteristics

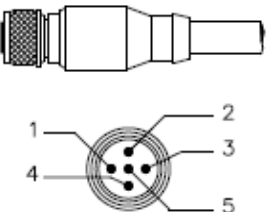
CONTROLLER – MAIN BOARD	
CPU	Family ARM7TDMI
Bus	32 bits
Architecture	RISC
Performance	40 MIPS
CPU Cache	8 Kbytes
Clock	40 MHz
DMA	10 channels
Ethernet	MAC 10/100 integrated
Watchdog	Yes (200 ms of cycle)
Power Supply Voltage	3.3 V for I/O and 2.5V for core (552 mW)

CONTROLLER – SECONDARY BOARD	
FPGA	Changes CycloneIII
Storage Memory	4KB
Processor	NiosII
Running Memory	1MB
Clock	85 MHz
Operating Voltage	3.3 V for I/O, 2.5V for PLL, 1.2V for core and 5V for communication channels.

CARD	
Power Supply Voltage	5 V (\pm 5% of tolerance)
Typical Current	750 mA
Real consumption	2.75 W
Environment Air Temperature	0 to 60° C according to the IEC 1131 standard
Storage Temperature	-20 to 80° C according to the IEC 1131 standard
Relative Humidity of the Operation Air	5% to 95% non-condensing
Cooling Mode	Air convection
Weight	0.318 kg
Dimensions (H x W x D) in mm	149x40x138 (without enclose)

M12 Connector

See in the following table the description of M12 Connector pins to connecting Profibus DP channel to DF95 controller.

M12 Connector Male	Pin	Signal
	1	*
	2	A (green)
	3	*
	4	B (red)
	5	*
	Screwed gland	SHD

*Not used

Indication LEDs

The LED names, colors, descriptions, and behaviors are showed in the table below.

LED	Color	Description	Behavior when Primary	Function when Secondary	Function when in HOLD
ETH LNK 2 (LNK)	Green	It indicates when the Ethernet link is active (ETH2 port).	Solid green LED when the Ethernet link is established (ETH2 port).	Same function and behavior when Primary.	Same function and behavior when Primary.
ETH TX2 (TX2)	Green	It indicates communication activity in the ETH2 port.	Blinking green LED when the ETH2 port is transmitting data.	Same function and behavior when Primary.	Same function and behavior when Primary.
ETH LNK 1 (LNK)	Green	It indicates when the Ethernet link is active (ETH1 port).	Solid green LED when the Ethernet link is established (ETH1 port).	Same function and behavior when Primary.	Same function and behavior when Primary.
ETH TX1 (TX1)	Green	It indicates communication activity in the ETH1 port.	Blinking green LED when the ETH1 port is transmitting data.	Same function and behavior when Primary.	Same function and behavior when Primary.
232 TX (TX)	Green	It indicates activity in the RS-232.port.	Blinking green LED when the RS-232 port is transmitting data.	Same function and behavior when Primary.	Same function and behavior when Primary.
Diagnostic (DIAG)	Yellow	It indicates the operation mode in the RS-232.	OFF when the RS-232 port is connected to the NetArm processor. Solid yellow when the RS-232 port is connected to the EC1 to execute operations related to the Profibus.	Same function and behavior when Primary.	Must be ON. ON – Ready for connection to EC1. OFF – Not supported
+5V DC (ON)	Green	It indicates when the module is ON.	Solid green LED when the module is powered.	Same function and behavior when Primary.	Same function and behavior when Primary.
FAIL (FAIL)	Red	It indicates hardware failure.	Solid red LED when in fail.	Same function and behavior when Primary.	Same function and behavior when Primary.
RUN (RUN)	Green	It indicates when the controller is operating in normal mode.	Solid green LED when in operation.	Same function and behavior when Primary.	Must be OFF. It is the inverse value of the HDL LED that indicates function in Monitor mode (maintenance mode of the device).
HOLD (HLD)	Yellow	It indicates when the controller is in hold mode. In this mode the controller does not perform any application and does not interfere on the plant operation (access via I/O cards or via digital bus are disabled)	Solid yellow LED when the controller is in hold mode (HOLD).	Same function and behavior when Primary.	Must be ON.
FORCE (FRC)	Red	1 – It indicates different modes of startup or maintenance requested by the operator via push-buttons (FACT INIT, HOLD and IP Address). 2 – It indicates power failure when the operating voltage decays down the expected value of 4.8 V (low line). 3 – Indicates a battery problem.	1 - Depending on the number of times that the right push-button is pressed, the FRC LED blinks at a given rate for a limited time, indicating the chosen mode (refer to Troubleshooting section for further information). 2 – Permanently solid red. The module will reset if the voltage reaches 4.6V. (HLD and FAIL LEDs light up together temporarily). 3 –FRC LED blinking and HLD LED solid red during the module start up – indicates low battery or the battery	Same function and behavior when Primary.	Hardware failure or power failure indication ON – Failure of power supply or hardware OFF – Power Ok. BLINKING – Mode selection

LED	Color	Description	Behavior when Primary	Function when Secondary	Function when in HOLD
			rear DIP switch is turned off (refer to Troubleshooting section for further information)		
Profibus Network (PB)	Green	It indicates Profibus channel activity.	ON – Profibus DP communication enabled OFF – Profibus DP communication disabled, communication (transmission) failure, short circuit on Profibus DP channel BLINKING – Multimaster mode operation	Same function and behavior when Primary.	Must be OFF. Profibus DP communication is disabled in HOLD.
Profibus Error (ERR)	Red	It indicates it there is an error related to the Profibus network.	ON - At least one slave device in this configuration is not communicating properly. OFF - All slave devices present in the configuration are communicating properly. BLINKING - Momentary failures in slaves or Primary recovering the Profibus DP channel. In the case of redundancy operation with no Profibus DP equipment in the network or Profibus DP cables disconnected the behavior is BLINKING.	ON – Failure on Profibus DP cable or on the Profibus DP configuration synchronism OFF – Synchronism OK and redundant master in standby BLINKING - Primary recovering the Profibus DP channel. This LED should always be OFF in the secondary. If it are ON some switchover failure may have occurred. The user must wait for recovery.	Must be OFF. Profibus DP communication is disabled in HOLD.
PA-1 PA-2	Green	It indicates Profibus channel activity.	ON - Indicates that there is at least one slave communicating in the PA channel. OFF - Indicates that there is not communication with any slave. It should always be ON in the case of slaves connected to the channel.	It must be OFF because on secondary the PA channel is OFF.	Must be OFF. Profibus PA communication is disabled in HOLD.
STANDBY (STB)	Green	If the HOLD LED is solid yellow and the STB LED is blinking, the firmware is being updating. If the HOLD LED is off, it indicates the controller role on redundancy as well as the synchronism status.	There are several blinking patterns to indicate different synchronism status. Refer to redundancy section for further details.	Must be ON.	BLINKING – Updating the firmware (saving the firmware on flash memory)

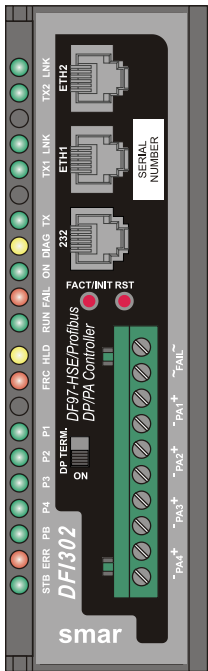
DF97 Specifications

Part Number

DF97 –HSE/Profibus DP Controller with 2 Ethernet 100 Mbps ports, 1 serial port, 4 Profibus PA ports and 1 Profibus DP channel

Description

DF97 module is the Smar solution for Profibus applications. Its main feature is working as Profibus DP-HSE gateway to provide power to the connectivity and flexibility to the system application. It allows wide communication between the Profibus DP and PA field devices. Through the HSE network and other DF1302 modules, it is possible the communication between field devices and other industrial protocols, providing greater flexibility to the control strategy projects. Through the I/O cards, it is also possible to execute discrete control via relay diagram logic (“Ladder Diagram”), allowing a single and integrated system. The module DF97 also can act as Modbus gateway (slave), allowing the interconnection of modules that are not fieldbus or HSE.



DF97 – Controller module

OFF

ON

1

2

3

4

5

Rear Dip Switch

	STORAGING	OPERATION
1 BATTERY	OFF	ON
2	OFF	OFF
3 SIMULATE	OFF	OFF
4 WATCHDOG	ON	ON
5	OFF	OFF

DF97

DF97

Characteristics and Module Limits

- One Profibus DP channel supporting up to 12 Mbps;
- Four Profibus PA ports supporting up to 32 devices for channel;
- It supports up to 124 Profibus DP and PA field devices;
- It supports up to 3584 bytes of input and 3584 bytes of output during the data interchange process;
- Limit of 64 external links by the HSE network;
- Maximum of 250 function blocks per DF95;
- One (1) Flexible Function Block (counted into the 250 allowed blocks), with 242 linked parameters to interface between the discrete and continuous control.
- It supports up to 16 HART modules (DF116/DF117).

NOTE

The limit of HART modules per controller is 16, but you must remember that when mapping the points through fieldbus function blocks strategy, one transducer block is used for each HART device. In this case, the system may be limited to less than 16 HART modules depending on the maximum number of fieldbus function blocks supported by the controller.

Continuous Control with Profibus

DF97 is a complete Profibus HSE controller with capacity to execute function blocks. Through the available **SYSTEM302**, **Studio302** and **Syscon** configuration tools is possible to configure the DF97 totally.

HSE Communication:

- Maximum of 512 link objects;
- Supervision up to 2000 points per second;

Discrete Control

DF97 module also has the capability of access I/O cards through the IMB (Inter-Module Bus), present in the backplane where the DF97 is mounted. Through the IMB, up to 16 racks DF1A or DF93 can be interconnected, each one having up to 4 cards.

DISCRETE CONTROL CHARACTERISTICS	
I/O Points*	1024 discrete points or 512 analog (maximum)
Ladder Function Blocks	2000 blocks (maximum)
Configuration File	120 Kbytes (maximum)
Program Execution Cycle for 1000 boolean operations (without redundancy)	50 ms (minimum)** 90 ms (typical)***
Program Execution Cycle with redundancy	Increment of 10ms (typical)**** up to 50 ms (maximum) to execution cycle
Execution Average Time	5.8 ms/Kbytes of program (minimum) 10.5 ms/Kbytes of program (typical)

* The whole number of points includes inputs and outputs, analog or digitals. Maximum may change according to I/O type used.

** 1131 Flexible Function Block adjusted to One (High Priority). Each 1000 boolean operations allocate 8.6 Kbytes.

*** Total execution time will change depending on the adjusted priority of 1131 FFB. The adjustment should be compatible with the quantity of function blocks and HSE links.

**** The whole execution time may change depending of the configuration file size.

NUMBER OF I/O POINTS	
Number of Profibus devices per network	125
Number of virtual points	4096
Number of Profibus discrete points	2048
Number of Profibus analog points	512

Firmware version and Device Revision

Some Firmware Versions may change the Device Revision, so this is an important step to consider during the configuration procedure for the controller. The section "Adding Function Blocks" better describes the steps for this configuration. The current available versions are:

Firmware version 2_0_X: Device Revision = 1

Firmware version 3_0_X: Device Revision = 2

Firmware version 4_0_X: Device Revision = 3

Technical Specifications

Memory

TYPE	SIZE
Volatile Memory	8 Mbytes
Non-Volatile Memory *	4 Mbytes
EEPROM	1 Kbytes
Flash to the program	4 Mbytes
Flash to monitor	2 Mbytes
Flash to EC1 (Profibus)	4 Mbytes

Battery

Type of battery	Battery Panasonic BR-2/3AE2SP - Lithium
Capacity	1200 mAh
Devices maintained by the battery	RTC and NVRAM
Minimum life span	8 years (typical charge of 17 μ A)
Maximum life span	49 years (typical charge of 2.8 μ A)
Voltage	3 V (subject to revision when below 2.5 V)

Communication Ports and Channels

ETHERNET PORT	
Rate	10/100 Mbits/s
Standard	IEEE 802.3u
Isolation	150 Vrms
Operation mode	Full duplex
Connector	RJ45 with shield*

* Grounded to the rail used for fixing the rack in which the DF97 is installed

DP PROFIBUS CHANNEL	
Rate	From 9.6 Kbit/s to 12 Mbits/s
Standard	EN 50170 e EN 50254
Physical Layer	EIA RS-485
Profibus Modem	EC1 (Hilscher)
Connector	M12

PROFIBUS PA CHANNELS	
Number of H1 Channels	2
Communication Rate	31.25 kbps
Standard	EN 61158 EN 50170
Physical Layer	ISA-S50.02-1992
MAU Type	Passive (bus not powered)
Isolation	500 Vac

MODBUS PORT	
Maximum rate*	115200 bps
Standard	EIA-232
Connector**	RJ12 with shield
Maximum Current***	0,5A @ 3.3V

*There is an increase in error rate as we increase the communication rate over 19200 bps. In many situations these errors can be acceptable, and they are not noticed by supervision.

** Grounded to the rail used for fixing the rack in which the DF97 is installed

*** Internally protected by solid state fuse.

REDUNDANCY PORT	
Maximum rate	115200 bps
Standard	EIA-232
Connector*	RJ12 with shield
Maximum Current**	0.5A @ 3.3V

* Grounded to the rail used for fixing the rack in which the DF97 is installed

** Internally protected by solid state fuse.

FAILURE RELAY	
Output Type	Solid state relay, normally closed (NC), isolated
Maximum voltage	30 Vdc
Maximum Current	200 mA
Overload Protection	Does not have. It must be provided externally
Normal Operation	Opened contacts
Failure Condition	Closed contacts
Maximum cable length connected to the relay	30 m

Observation: The power supply for the load must not be from an external network (outside the panel).

IMB BUS	
Voltage	5 Vdc
Bus	8 bits
Failure indication	Yes
Hot swap	Yes

Module Characteristics

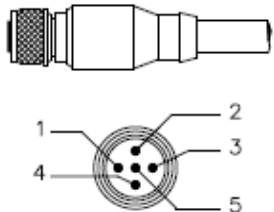
CONTROLLER – MAIN BOARD	
CPU	Family ARM7TDMI
Bus	32 bits
Architecture	RISC
Performance	40 MIPS
CPU Cache	8 Kbytes
Clock	40 MHz
DMA	10 channels
Ethernet	MAC 10/100 integrated
Watchdog	Yes (200 ms of cycle)
Power Supply Voltage	3.3 V for I/O and 2.5V for core (552 mW)

CONTROLLER – SECONDARY BOARD	
FPGA	Changes CycloneIII
Storage Memory	4KB
Processor	NiosII
Running Memory	1MB
Clock	85 MHz
Operating Voltage	3.3 V for I/O, 2.5V for PLL, 1.2V for core and 5V for communication channels.

CARD	
Power Supply Voltage	5 V ($\pm 5\%$ of tolerance)
Typical Current	620 mA
Real consumption	2.75 W
Environment Air Temperature	0 to 60° C according to the IEC 1131 standard
Storage Temperature	-20 to 80° C according to the IEC 1131 standard
Relative Humidity of the Operation Air	5% to 95% non-condensing
Cooling Mode	Air convection
Weight	0.318 kg
Dimensions (H x W x D) in mm	149x40x138 (without enclose)

M12 Connector

See in the following table the description of M12 Connector pins to connecting Profibus DP channel to DF97 controller.

M12 Connector Male	Pin	Signal
	1	*
	2	A (green)
	3	*
	4	B (red)
	5	*
	Screwed gland	SHD

*Not used

Indication LEDs

The LED names, colors, descriptions and behaviors are showed in the table below.

LED	Color	Description	Behavior when Primary	Function when Secondary	Function when in HOLD
ETH LNK 2 (LNK)	Green	It indicates when the Ethernet link is active (ETH2 port).	Solid green LED when the Ethernet link is established (ETH2 port).	Same function and behavior when Primary.	Same function and behavior when Primary.
ETH TX2 (TX2)	Green	It indicates communication activity in the ETH2 port.	Blinking green LED when the ETH2 port is transmitting data.	Same function and behavior when Primary.	Same function and behavior when Primary.
ETH LNK 1 (LNK)	Green	It indicates when the Ethernet link is active (ETH1 port).	Solid green LED when the Ethernet link is established (ETH1 port).	Same function and behavior when Primary.	Same function and behavior when Primary.
ETH TX1 (TX1)	Green	It indicates communication activity in the ETH1 port.	Blinking green LED when the ETH1 port is transmitting data.	Same function and behavior when Primary.	Same function and behavior when Primary.
232 TX (TX)	Green	It indicates activity in the RS-232 port.	Blinking green LED when the RS-232 port is transmitting data.	Same function and behavior when Primary.	Same function and behavior when Primary.
Diagnostic (DIAG)	Yellow	It indicates the operation mode in the RS-232.	OFF when the RS-232 port is connected to the NetArm processor. Solid yellow when the RS-232 port is connected to the EC1 to execute operations related to the Profibus.	Same function and behavior when Primary.	Must be ON. ON – Ready for connection to EC1. OFF – Not supported
+5V DC (ON)	Green	It indicates when the module is ON.	Solid green LED when the module is powered.	Same function and behavior when Primary.	Same function and behavior when Primary.
FAIL (FAIL)	Red	It indicates hardware failure.	Solid red LED when in fail.	Same function and behavior when Primary.	Same function and behavior when Primary.
RUN (RUN)	Green	It indicates when the controller is operating in normal mode.	Solid green LED when in operation.	Same function and behavior when Primary.	Must be OFF. It is the inverse value of the HDL LED that indicates function in Monitor mode (maintenance mode of the device).
HOLD (HLD)	Yellow	It indicates when the controller is in hold mode. In this mode the controller does not perform any application and does not interfere on the plant operation (access via I/O cards or via digital bus are disabled)	Solid yellow LED when the controller is in hold mode (HOLD).	Same function and behavior when Primary.	Must be ON.
FORCE (FRC)	Red	1 – It indicates different modes of startup or maintenance requested by the operator via push-buttons (FACT INIT, HOLD and IP Address). 2 – It indicates power failure when the operating voltage decays down the expected value of 4.8 V (low line). 3 – Indicates a battery problem.	1 - Depending on the number of times that the right push-button is pressed, the FRC LED blinks at a given rate for a limited time, indicating the chosen mode (refer to Troubleshooting section for further information). 2 – Permanently solid red. The module will reset if the voltage reaches 4.6V. (HLD and FAIL LEDs light up together temporarily). 3 –FRC LED blinking and HLD LED solid red during the module start up – indicates low	Same function and behavior when Primary.	Hardware failure or power failure indication ON – Failure of power supply or hardware OFF – Power Ok. BLINKING – Mode selection

LED	Color	Description	Behavior when Primary	Function when Secondary	Function when in HOLD
			battery or the battery rear DIP switch is turned off (refer to Troubleshooting section for further information)		
Profibus Network (PB)	Green	It indicates Profibus channel activity.	ON – Profibus DP communication enabled OFF – Profibus DP communication disabled, communication (transmission) failure, short circuit on Profibus DP channel BLINKING – Multimaster mode operation	Same function and behavior when Primary.	Must be OFF. Profibus DP communication is disabled in HOLD.
Profibus Error (ERR)	Red	It indicates it there is an error related to the Profibus network.	ON - At least one slave device in this configuration is not communicating properly. OFF - All slave devices present in the configuration are communicating properly. BLINKING - Momentary failures in slaves or Primary recovering the Profibus DP channel. In the case of redundancy operation with no Profibus DP equipment in the network or Profibus DP cables disconnected the behavior is BLINKING.	ON – Failure on Profibus DP cable or on the Profibus DP configuration synchronism OFF – Synchronism OK and redundant master in standby BLINKING - Primary recovering the Profibus DP channel. This LED should always be OFF in the secondary. If it are ON some switchover failure may have occurred. The user must wait for recovery.	Must be OFF. Profibus DP communication is disabled in HOLD.
PA-1 PA-2 PA-3 PA-4	Green	It indicates Profibus channel activity.	ON - Indicates that there is at least one slave communicating in the PA channel. OFF - Indicates that there is not communication with any slave. It should always be ON in the case of slaves connected to the channel.	It must be OFF because on secondary the PA channel is OFF.	Must be OFF. Profibus PA communication is disabled in HOLD.
STANDBY (STB)	Green	If the HOLD LED is solid yellow and the STB LED is blinking, the firmware is being updating. If the HOLD LED is off, it indicates the controller role on redundancy as well as the synchronism status.	There are several blinking patterns to indicate different synchronism status. Refer to redundancy section for further details.	Must be ON.	BLINKING – Updating the firmware (saving the firmware on flash memory)

DF100 Specifications

DF100 – HSE/*WirelessHART*™ Controller with 2 Ethernet 100 Mbps ports, 1 RS-485 port, and 1 *WirelessHART* channel

Description

The DF100 controller is a key element in the distributed architecture of the field control systems. Combines powerful communication features with access to field devices via *WirelessHART*™ protocol.

This controller has features entirely new when compared with the DFI302 modular line. The DF100 can be used in outdoors, because it has ingress protection IP66. In addition, it makes possible to work with the new specification HSE WIO of Fieldbus Foundation, and Modbus communication via RS-485 port.

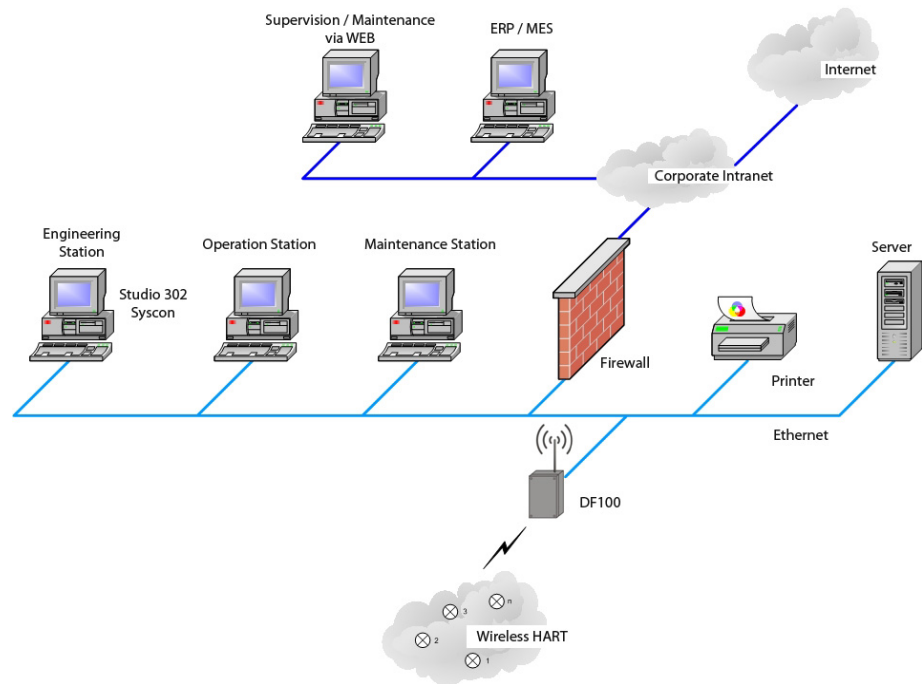


DF100 – WirelessHART/HSE controller

The adoption of wireless technology in the industry is a strategic decision to significantly reduce costs related to cabling, trays, and technical/engineering hours spent on maintaining the network.

A standardized wireless network infrastructure at this stage of the project meets the emerging standard *WirelessHART*™ and eliminates the costs and risks associated with the acquisition of proprietary wireless networks and instrumentation of a single supplier.

The project basic topology is shown in the following figure. It has two control network segments interconnected to corporate networks. On the control network is added the DF100 controller. Finally, supervision and control stations complete the interconnection system.



Wireless architecture for the control network

Tools such as **Syscon** and **Studio302** are related to the architecture to integrate the controller and the wireless network to others industrial automation protocols.

General characteristics and limits

The following characteristics are in all issues of DF100.

- 1 *WirelessHART*™ channel (HART 7 specification of HART Communication Foundation);
- 2 10/100 Mbps Ethernet ports;
- 1 RS-485 port (for Modbus communication)
- It integrates up to 100 *WirelessHART* devices¹;
- Modbus gateway;
- Integrated webserver;
- Real Time Clock (RTC) and watchdog;
- Support for HSE WIO of Fieldbus Foundation architecture;
- Ingress protection IP66 (supports outdoor);
- Operation temperature: -40 °C to 60 °C;
- Operation voltage: 20Vdc to 30Vdc, 11W maximum.

Continuous Control with FOUNDATION™ Fieldbus

The DF100 module is a HSE device, with block execution capability. Through configuration tools available in the **SYSTEM302**, such as the **Studio302** and **Syscon**, it is possible to configure the DF100 completely.

HSE Communication

- Maximum of 300 link objects;
- Limit of 128 linked parameters
- Dynamic block instantiation. Maximum of 170 function blocks;

HSE WIO Function Blocks and Transducers

The DF100 is in compliance with the new requirements of FOUNDATION fieldbus remote I/O based on

¹ The maximum number of *WirelessHART* devices may be less depending on the update rate desired.

High Speed Ethernet (HSE) and the integration of HART field devices through the *WirelessHART*.

These requirements allow that the industrial automation suppliers develop Wireless and Remote I/O (WIO) gateways running in a wireless HSE backhaul. The new HSE WIO function blocks and transducers that meet those requirements are listed below:

MNEMONIC	DESCRIPTION
TBHG	Transducer Block for HART Gateway
TBWH	Transducer Block for <i>WirelessHART</i>
WAI	WIO Analog Input
MAI16	Multiple Analog Input 16

Firmware version and Device Revision

Some firmware versions may change the Device Revision, so this is an important step to consider during the configuration procedure for the controller. The section "Adding Function Blocks" better describes the steps for this configuration. The current available version is:

Firmware version 2.x: Device Revision = 2

Technical Specifications

Memory

TYPE	SIZE
Volatile Memory	8 Mbytes
Non Volatile Memory *	4 Mbytes
EEPROM	1 Kbytes
Flash to the program	4 Mbytes
Flash to monitor	2 Mbytes

* It is maintained by the primary internal battery (not rechargeable)

Battery

Type of battery	Battery Panasonic BR-2/3AE2SP - Lithium
Capacity	1200 mAh
Devices maintained by the battery	RTC and NVRAM
Lifespan	More than 10 years*
Voltage	3 V (subject to revision when below 2.5 V)

* The battery switch (DIP Switch W1, position 1 on the CPU board) has to be turned off (position OFF) when the module is powered down and is getting out of operation for long periods of time, not being necessary to maintain the configuration data (data from volatile memory and RTC are lost but can be reloaded later).

If you need powering down the module for short periods of time without losing the configuration, the switch (W1, position 1) of the battery must not be turned off. In this case, the battery can keep data for more than 10 years at 25 °C, or at least 2.6 years in the worst temperature conditions.

Communication Ports and Channels

ETHERNET PORT	
Communication Rate	10/100 Mbps
Standard	IEEE 802.3u
Isolation	150 Vrms
Operation Mode	Full duplex
Connector	RJ45 with shield*

MODBUS PORT	
Communication Rate (Maximum)	19200 bps
Standard	EIA-485
Connector	Male, 3-pin, quick coupling type

WIRELESSHART PORT	
Operation Frequency	2.4 – 2.4835 GHz
Standard	802.15.4 (DSSS)
Number of channels	16 (IEEE 802.15.4)*

* You can make unavailable one or more channels as required by the regulatory agency of the country where it is installed.

FAILURE RELAY	
Output Type	Solid state relay, normally closed (NC), isolated
Maximum Voltage	30 Vdc
Maximum Current	200 mA
Overload Protection	Not available. It must be provided externally
Normal Operation	Opened contacts
Failure Condition	Closed contacts
Maximum cable length (maximum) connected to the relay	30 m

The power supply for the load must not be from an external network (outside the panel).

Module Characteristics

PROCESSOR	
CPU	Family ARM7TDMI
Bus	32 bits
Architecture	RISC
Performance	40 MIPS
CPU Cache	8 Kbytes
Clock	40 MHz
DMA	10 channels
Ethernet	MAC 10/100 integrated
Watchdog	Yes (1.6 s of cycle)
Operation Voltage	3.3 V for I/O

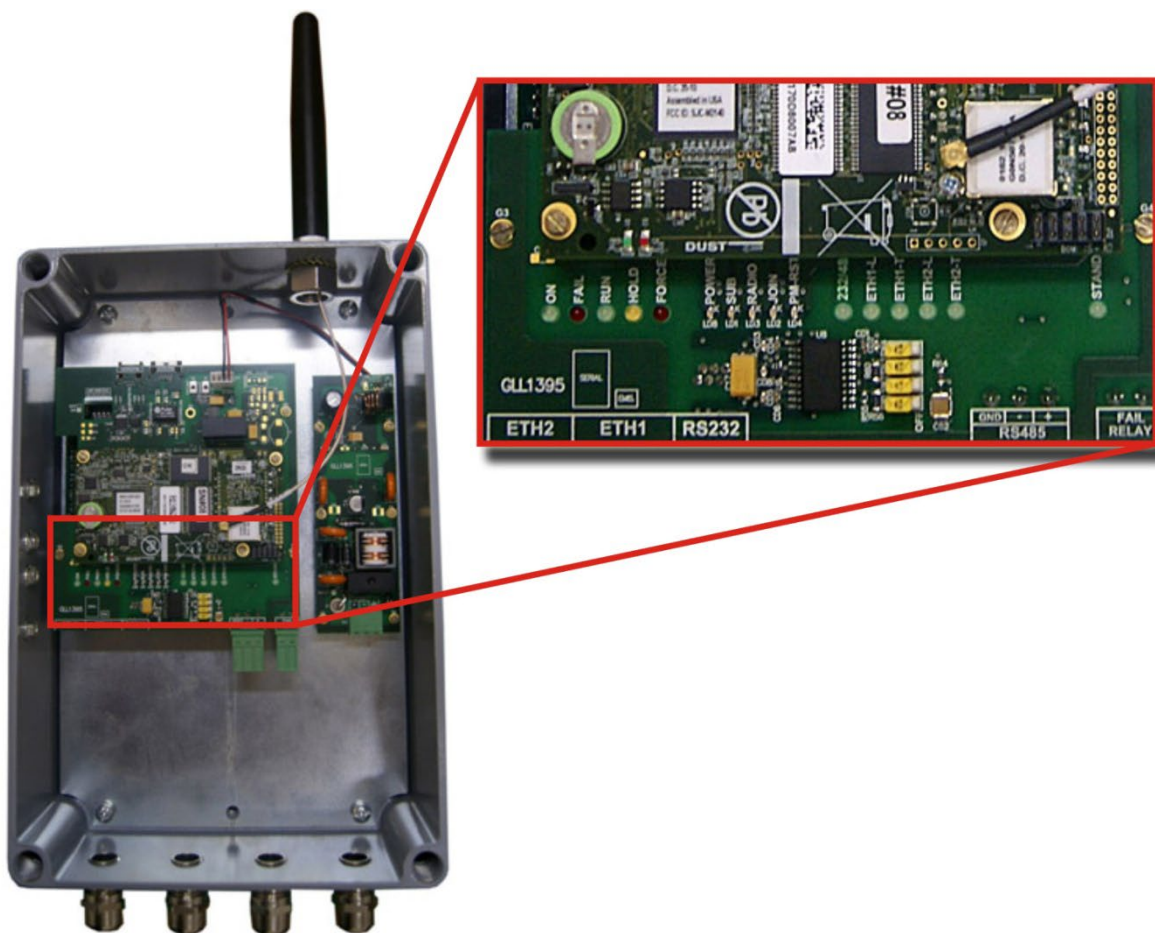
DF100 CONTROLLER	
Operation Voltage	20 Vdc to 30 Vdc
Typical Current	444 mA (average); 600 mA (peak) @ 18Vdc
Real Consumption	8 W (average), 11 W (peak)
Environment Air Temperature	-40 to 60 °C
Storage Temperature	-40 to 60 °C
Relative Air Humidity (Operation)	10% - 90% (non-condensing)
Cooling Mode	Air Convection
Dimensions (H x W x D) in mm	300 x 200 x 120 (without package)

Indication LEDs

The LED names, colors, descriptions, and behaviors are showed in the table below.

LED	COLOR	DESCRIPTION	BEHAVIOR
+5Vdc (ON)	Green	It indicates when the module is ON.	Solid green LED when power is on.
FAIL (FAIL)	Red	It indicates hardware failure.	Solid red LED when in fail.
RUN (RUN)	Green	It indicates when the controller is operating in normal mode.	Solid green LED in operation.
HOLD (HLD)	Yellow	It indicates when the controller is in hold mode. In this mode the controller does not perform any application and does not interfere on the plant operation (access via I/O cards or via digital bus are disabled)	Solid yellow LED when the controller is in hold mode (HOLD).
FORCE (FRC)	Red	<p>1 – It indicates different modes of startup or maintenance requested by the operator via push-buttons (FACT INIT, HOLD and IP Address).</p> <p>2 – It indicates power failure when the operating voltage decays down the expected value of 4.8 V (low line).</p> <p>3 – Indicates a battery problem.</p>	<p>1 - Depending on the number of times that the right push-button is pressed, the FRC LED blinks at a given rate for a limited time, indicating the chosen mode (refer to Troubleshooting section for further information).</p> <p>2 – Permanently solid red. The module will reset if the voltage reaches 4.6V. (HLD and FAIL LEDs light up together temporarily).</p> <p>3 –FRC LED blinking and HLD LED solid red during the module start up – indicates low battery or the battery rear DIP switch is turned off (refer to Troubleshooting section for further information)</p>
232/485 TX	Green	It indicates activity in the RS-485 port.	Blinking green LED when the RS-485 port is transmitting data.
ETH1 LNK	Green	It indicates when the Ethernet link is active (ETH1 port).	Solid green LED when the Ethernet link is established (ETH1 port).
ETH1 TX	Green	It indicates communication activity in the ETH1 port	Blinking green LED when the ETH1 port is transmitting data.
ETH2 LNK	Green	It indicates when the Ethernet link is active (ETH2 port).	Solid green LED when the Ethernet link is established (ETH2 port).
ETH2 TX	Green	It indicates communication activity in the ETH2 port.	Blinking green LED when the ETH2 port is transmitting data.
STANDBY	Green	It indicates if module is operating as Primary Master or if the firmware is being updating.	Solid green LED when the controller is active and it is the Primary.

The indication LEDs above mentioned can be viewed through the DF100 interface board, after opening the front cover. See the following figure.



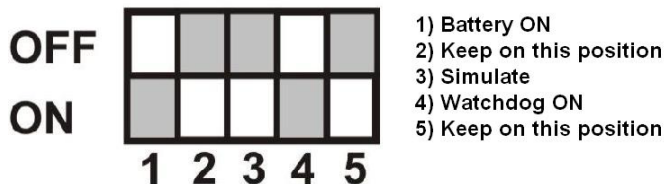
Detail of the indication LEDs in the DF100 interface board

LEDS related to *WirelessHART* Manager

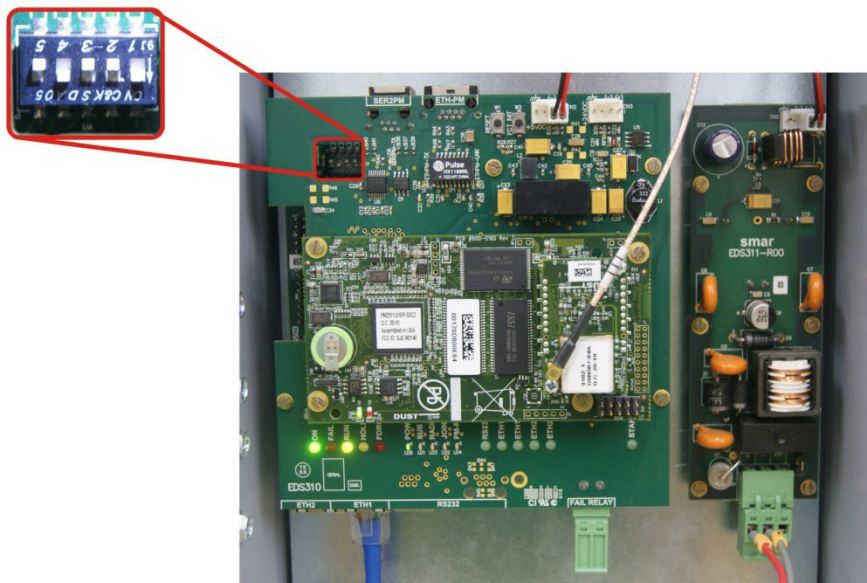
LED	COLOR	DESCRIPTION	BEHAVIOR
Power	Green	It indicates when the <i>WirelessHART</i> Manager is ON.	Solid green LED when the <i>WirelessHART</i> Manager is powered.
Subscription (SUB)	Yellow	It indicates that a client program is connected to the <i>WirelessHART</i> Manager and it is ready to receive notifications of data, alarms and events.	Solid yellow LED when a client receives a notification from the <i>WirelessHART</i> Manager.
Radio	Yellow	It indicates activity on the <i>WirelessHART</i> radio.	Blinking yellow LED when there is communication activity.
JOIN	Yellow	Indicates when the <i>WirelessHART</i> Manager has a <i>WirelessHART</i> field instrument in the process of addition (joining) to the network	Blinking yellow LED when an instrument is in joining process.
PM-RST	Red	Indicates that the DF100 controller restarted the <i>WirelessHART</i> Manager and it is out of service.	Solid red LED while the <i>WirelessHART</i> Manager is out of service.

Hardware configuration

DIP Switch W1: To the DF100 operation, turn on the switches 1 (Battery on) and 4 (Watchdog on), keeping them in ON as in the following figure.



The **DIP Switch W1** is installed on the CPU board; it can be accessed via DF100 interface board.



RS-485 Switches: the switches **SW1**, **SW2**, **SW3** and **SW4** are specific to Modbus communication operation via RS-485 port. The RS-485 switches are installed on the DF100 interface board, next to the RS-485 connector.

























The **SW1** switch can be used to enable the pull-up resistor. When enabled (ON position), this switch connects the pull-up resistor to the **TX+** line (pin + of the RS-485 connector) of the RS-485 bus. On the other hand, the **SW4** switch controls the pull-down resistor enabling. When it is enabled (ON position), connects the pull-down resistor to the line **RX-** (pin – of the RS-485 connector) of the RS-485 bus. We recommend the use of these resistors to prevent noisy signals is interpreted as having valid data on a bus without communication.



ATTENTION

Just a single set of pull-up and pull-down resistors can be enabled in the RS-485 bus.

The **SW2** and **SW3** switches can be used (simultaneously) to enable the terminator resistor of the DF100. A terminator resistor has the function to avoid signal reflections on the RS-485 bus. The RS-485 bus usually uses a terminator resistor at each end of the bus. It is recommended to enable the terminator for high rates of communication and use of long cables to cover long distances between devices connected to the RS-485 bus.

	SW1	SW2	SW3	SW4	
OFF					Resistors disabled
ON					
	SW1	SW2	SW3	SW4	
OFF					Pull-up and pull-down resistors enabled Terminator resistor disabled
ON					
	SW1	SW2	SW3	SW4	
OFF					Pull-up and pull-down resistors disabled Terminator resistor enabled
ON					

SW1) Pull-up resistor
SW2) Terminator
SW3) Terminator
SW4) Pull-down resistor

Refer to the Modbus Network section, item Bus Terminators and Stub lengths of the **SYSTEM302** Electric Installation Guide for further information.

CABLE SPECIFICATIONS

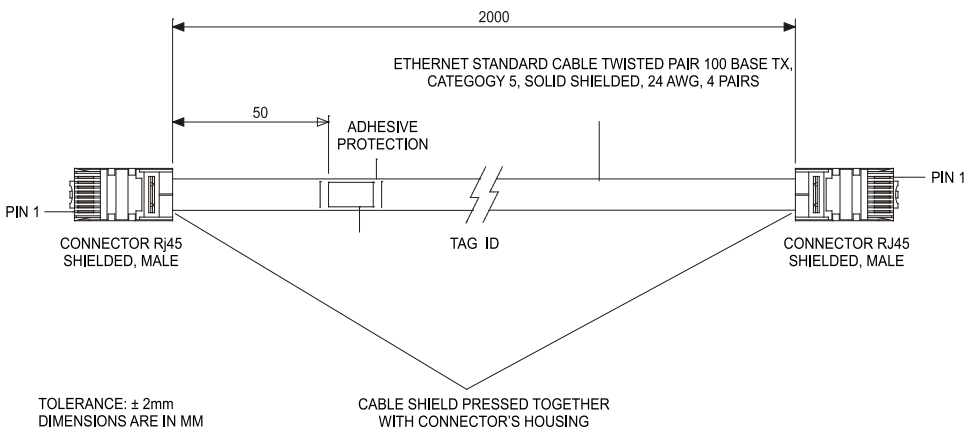
Ethernet Cable Specifications

To assembly a new Ethernet cable, the user should follow the specifications of the twisted cable pair, according to the part number DF54 or DF55.

DF54/DF55

DF54- Standard cable. To be used in network communication between controllers and Switch/HUB.

DF55- Cross cable. To be used in a point-to-point communication between PC and controller.



CABLE CONNECTION DIAGRAM

DF54			DF55 CROSS		
1	WHITE GREEN	1	1	WHITE GREEN	3
2	GREEN	2	2	GREEN	6
3	WHITE ORANGE	3	3	WHITE ORANGE	1
4	BLUE	4	4	BLUE	4
5	WHITE BLUE	5	5	WHITE BLUE	5
6	ORANGE	6	6	ORANGE	2
7	WHITE BROWN	7	7	WHITE BROWN	7
8	BROWN	8	8	BROWN	8

OBS: COLORS ARE MERELY A SUGGESTION.
IS IMPORTANT TO OBEY THE COLORS PAIRS AS THE CABLE CONNECTION DIAGRAM.

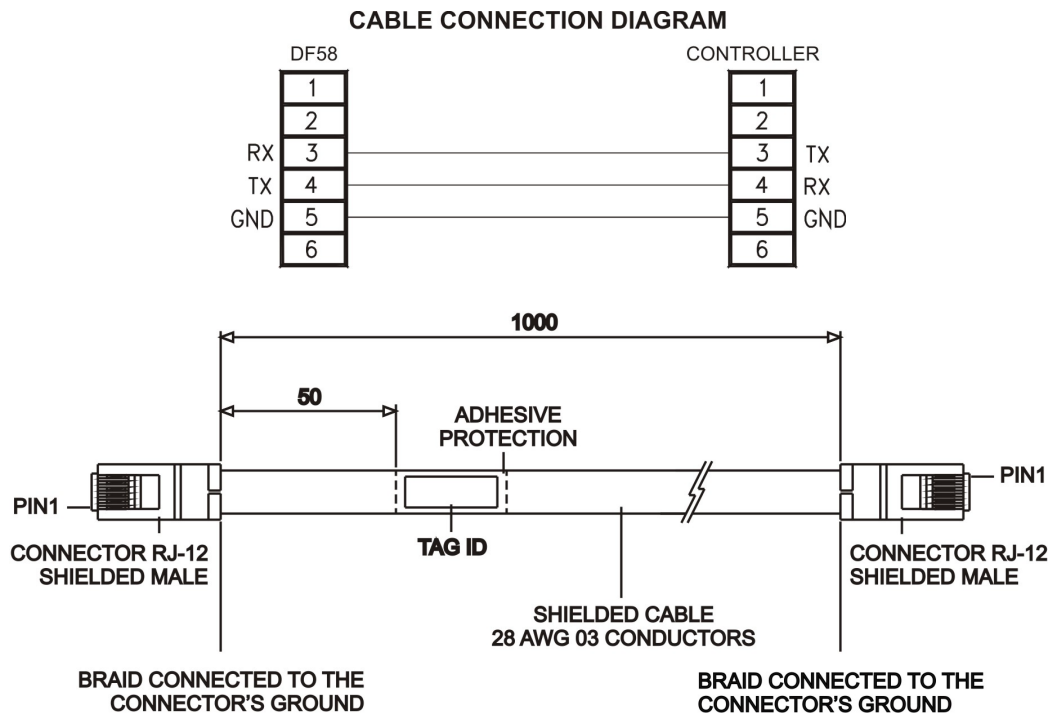
The DF54 cable has the following length options:

PRODUCT	CLASS	OPTION
DF54		TWISTED PAIR CABLE 100 BASE TX
1 – CABLE LENGTH	1	0.5 m
	2	2 m
	3	3 m
	4	5 m
	5	10 m

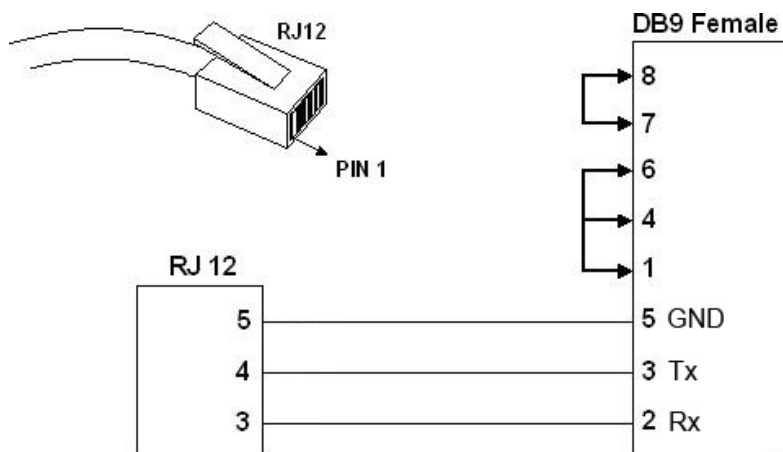
Serial Cable Specifications

DF59

To assembly a serial cable between **controller and DF58 (RS232/RS485 Interface)**, the user has to follow the specifications of the part number **DF59**.



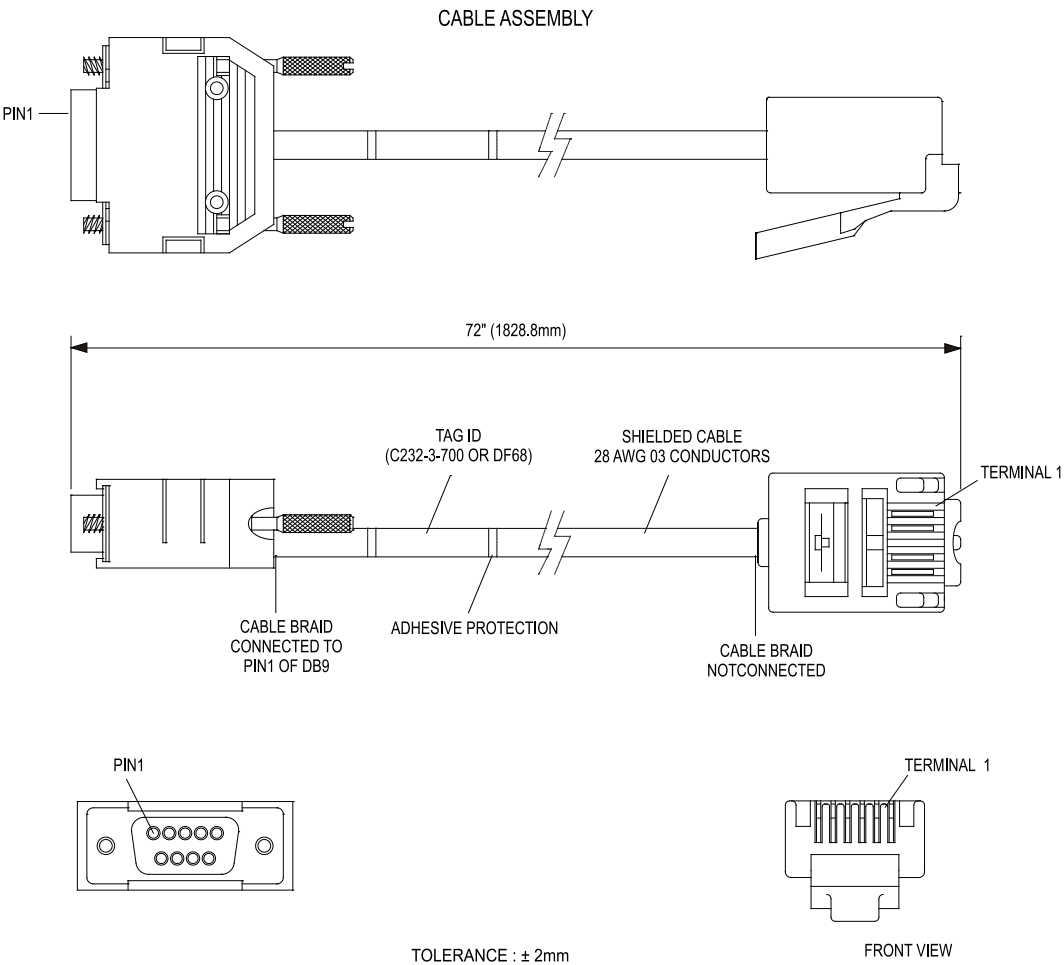
To assembly a serial cable between **controller and computer**, or between **DF58 (interface) and computer** see the following instructions that show a connection between RJ12 (used in the controller and DF58) and a female-DB9 connector (used in the computer):



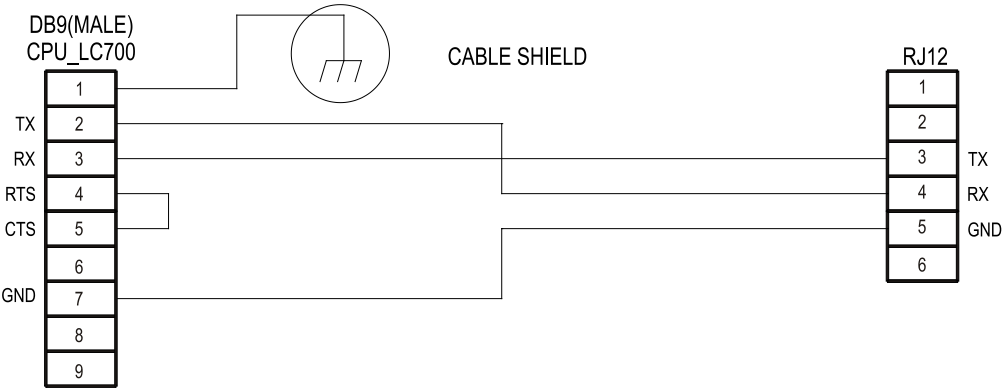
The jumpers under DB9 side are recommended but not necessary. It depends on the application running in the computer.

DF68

To assembly a serial cable between **DF51 controller** and **DF65 coprocessor**, the user should follow the specifications of the part number DF68.

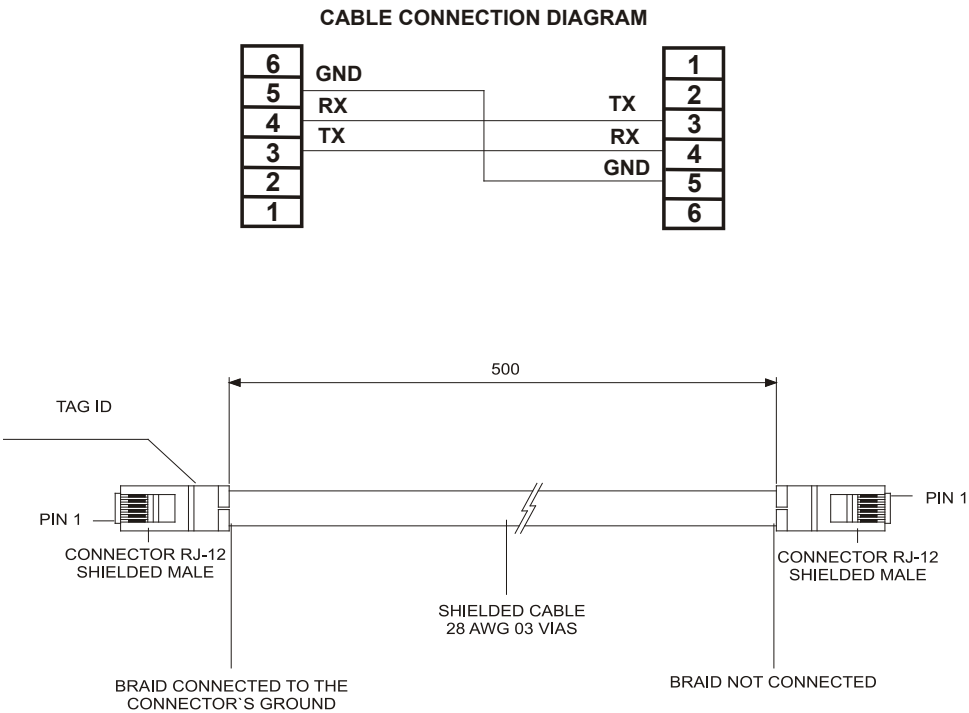


CABLE CONNECTION DIAGRAM



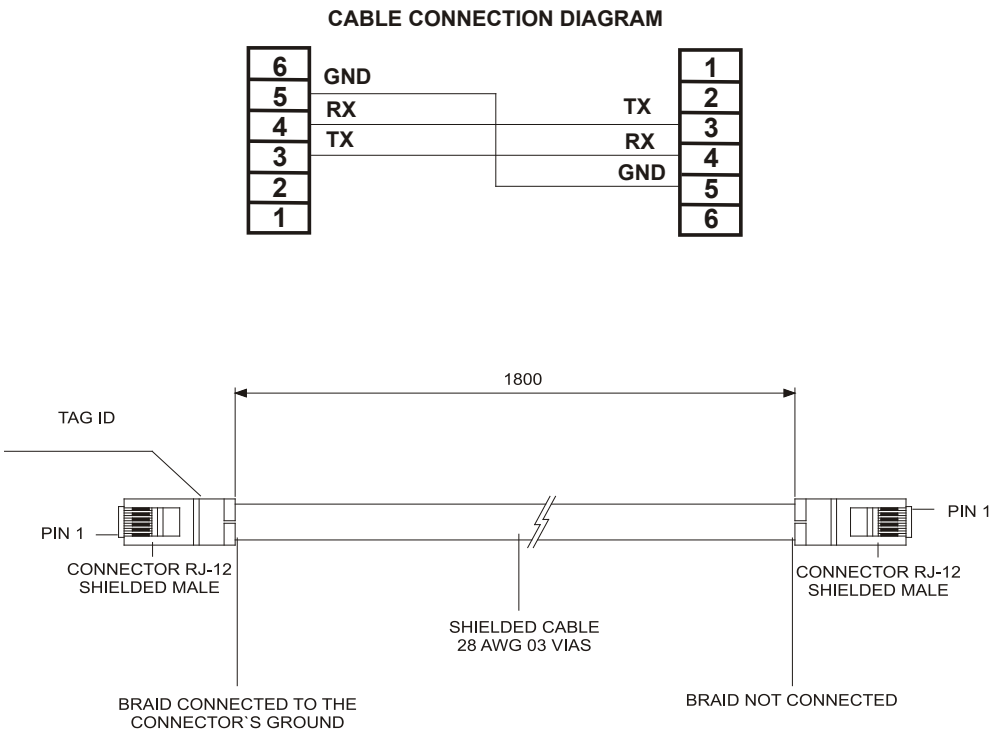
DF82

DF82 cable interconnects redundant controllers. The figure below shows the cable connection diagram.



DF83

DF83 cable interconnects redundant controllers. The figure below shows the cable connection diagram.



Cables for Racks Interconnection and Power Distribution

Depending on the rack model different types of cables are necessary to interconnect racks and for power distribution throughout the IMB bus. In the following table are the available cable types.

Code	Description
System based on DF1A and DF78	
DF3	DFI302 flat cable to connect two racks – length 6.5 cm
DF4A	DFI302 flat cable to connect two racks – length 65 cm
DF5A	DFI302 flat cable to connect two racks – length 81.5 cm
DF6A	DFI302 flat cable to connect two racks – length 98 cm
DF7A	DFI302 flat cable to connect two racks – length 110 cm
System based on DF92 and DF93	
DF90	IMB power cable
DF101	Shielded flat cable to connect racks by left side – length 70 cm
DF102	Shielded flat cable to connect racks by right side – length 65 cm
DF103	Shielded flat cable to connect racks by right side – length 81 cm
DF104	Shielded flat cable to connect racks by right side – length 98 cm
DF105	Shielded flat cable to connect racks by right side – length 115 cm

For further details about the correct cable installation, please, refer to Installing section.

Expansion flat cables for systems based on DF92 and DF93

These flat cables are used when the DFI302 is expanded in more than one row of racks (DF92 or DF93), i.e., in different DIN rail segments, one below the other. To ground the flat cables' shield, use ground terminals next to the connections among flat cables and racks.

- **DF101 - Flat cable to connect racks by left side**

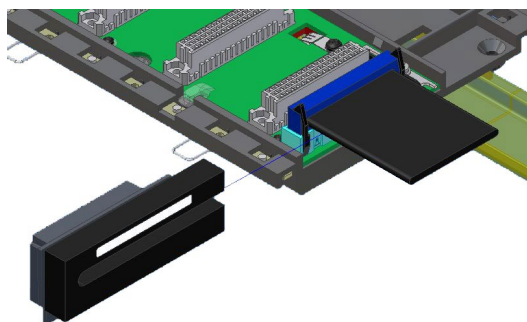
The DF101 is installed on the rear connectors of the left extremity rack of each row of racks, interconnecting the rows 2-3, 4-5 and 6-7 (if they exist). The available terminal next to each DF91 can be used for grounding. See the Installing section.

- **DF102, DF103, DF104 and DF105 - Flat cable to connect racks by right side**

They are installed on the upper connectors of the right extremity rack of each row of racks, interconnecting the rows 1-2, 3-4 and 5-6 (if they exist). See the Installing section.

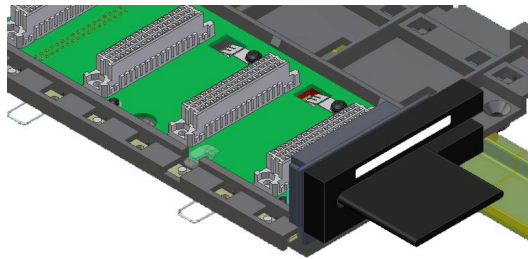
Flat cables protector (connector cap)

To meet the EMC requirements an ESD protector has to be installed on the flat cables connection, at right. In the following figure a flat cable protector is shown when it is being installed on the cable connector.



Installing the flat cables protector

The following figure shows the flat cable protector installed.



Flat cable protector installed

DF90 cable

The power expansion has to be used when the DFI302 is expanded in more than one row of racks, i.e., in different DIN rail segments, one below the other. The DF90 is the IMB power transmission cable. Its features provide low voltage drop and protection against electromagnetic interference.

The cable DF90 must be connected only through DF91. It cannot be directly installed in the racks, because it can damage the racks. For further details, see the Installing section.



IMB power cable (DF90)

Section 13

ADDING POWER SUPPLIES

Introduction

There are some recommendations when adding power supply modules to the system which should be considered.

First of all, an overview of the whole system is necessary at this time to better choose the modules (power supply, impedance etc). Each controller module needs at least one power supply for backplane. The addition of I/Os modules requires new calculations to the power supply.

NOTE

Using ladder logic (FFB 1131), for a better monitoring of the functional state of each used I/O module is recommended to use the **STATUS** block in the logic. Thus the system can be advised if some I/O module have a failure. So that is easier to find a damaged module. Insert and configure this block according to the **LogicView for FFB** manual.

The following table shows the available modules used as power supply, intrinsic safety barrier and fieldbus impedances.

MODEL	DESCRIPTION
DF50	Power Supply for Backplane 90-264 Vac
DF56	Power Supply for Backplane 20-30 Vdc
DF52	Power Supply for Fieldbus 90-264 Vac
DF60	Power Supply for Fieldbus 20-30 Vdc
DF53	Power Supply Impedance for Fieldbus (4 ports)
DF47-12	Intrinsic Safety Barrier for Fieldbus
DF47-17	
DF87	Power Supply for Backplane 20-30 Vdc, 5 A, redundant, with diagnostic
DF98	High Current Power Supply Impedance for Fieldbus

DF50 – Power Supply Module for Backplane (Redundant)

Description

This redundant power supply works independently or together with another redundant power supply module to ensure a constant supply of power to the application.

When two redundant power supplies are used, if one of them fails, the backup will automatically assume the operation. A relay is provided to indicate failure on each power supply giving the user a chance to replace the faulty one.

This module provides two voltage outputs:

- a) **5 Vdc @ 3 A:** distributed by Power Lines in the Inter-Module-Bus (IMB) throughout the racks to supply the module circuits;
- b) **24 Vdc @ 300 mA:** for external use through the terminals 1B and 2B.

The applied AC voltage, the 5 Vdc and the 24 Vdc are all isolated between them.

Installation and Configuration

For systems based on DF92 and DF93 rack, with DF90 and DF91

Redundant mode options

- **Splitting Power concept:** In this situation, two modules will supply power to a bus segment. If one of them was turned off or fails, the other power supply must be able to supply energy, alone, to the segment.
The **CH1** jumper (power supply) must be set in **R** position for both modules and **W1** jumper (power supply) must be opened for both modules.
- **Standby concept:** In this case, just one power supply provides energy to the system. If it was turned off or fails, the backup module will assume the operation. In both modules, the jumper **CH1** (power supply) must be set in the **R** position and **W1** jumper (power supply) must be placed only in the backup module.

Expansion of load capacity by adding power supplies or pairs of redundant power supplies

If the system consumption is greater than 3A, it can be subdivided in up to 8 groups sized for consumption of up to 3A each, and each group is individually powered by a power supply, or redundant pair of power supplies. More details on the Power supplies positioning topic.

Power supplies positions in the racks

On **DF92**, the pair of redundant power supplies must be installed in the first and second slots.

On **DF93** is recommended the placement of the redundant pair in the first and second slots, but it can be installed in any slots if necessary.

Configuration of “W1” and “CH1” jumpers

The **DF50 CH1** jumper always must be connected to the **R** position. The **W1** jumper (power supply) must be connected only in the **DF50** modules configured as “backup”, in the standby concept, as above mentioned in the redundant mode options.

For systems based on DF1A and DF78 racks

Non-redundant (single module): power consumption **limited** to 3A:

There is an addressing restriction related to the power supply location. The restriction is that the first rack (address 0) must always contain a power supply module at the first slot.
In the power supply module the **CH1** jumper must be set in **E** position.

Non-redundant (more than one module): power consumption **bigger** than 3A:

Additional modules are placed in the bus in parallel, but isolated one of the other. For systems based on **DF1A rack**, the power supplies modules must always be placed at the first rack's slot. The jumper **W1** (in the rack), where is the new power supply module, must be cut. The new power supply module will only supply power to the rack where it is sitting on and to the consecutive ones (never backwards).

In all power supplies modules, the **CH1** jumper must be set in **E** position.

Redundant mode

- **Splitting Power concept:** In this case of redundancy, the user may have two power supplies modules in parallel in first and third slots of rack **DF1A** or in the first and second slots of rack **DF78**. The **CH1** jumper (power supply) must be set in **R** position for both modules and **W1** jumper (power supply) must be opened for both modules. In this situation, the two modules will supply power to the bus.
- **Standby concept:** In this case, the main module must be placed in the first slot and the backup module in the third slot of rack **DF1A** or in the first and second slots of rack **DF78**. In both modules, the **CH1** jumper (power supply) must be set in the position **R** and **W1** jumper (power supply) must be placed only in the backup module.

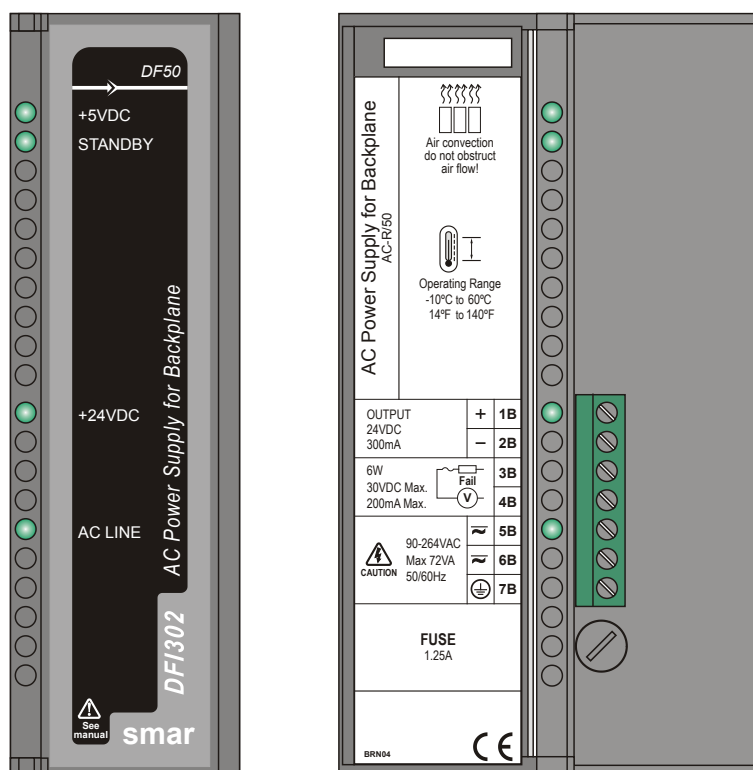


Figure 13.1 - AC Power Supply Module: DF50

Technical Specifications

INPUTS	
DC	127 to 135 Vdc
AC	90 to 264 VAC, 50/60 Hz (nominal), 47 to 63 Hz (range)
Inrush Current	< 36 A @ 220 Vac [$\Delta T < 740 \mu s$]
Time until Power Fail	6 ms @ 102 Vac (120 Vac – 15%) [Full Load]
Time until Shutdown	27 ms @ 102 Vac; > 200ms @ 220 Vac [Full Load]
Maximum consumption	72 VA
Indicator	AC LINE (Green LED)

OUTPUTS	
a) Output 1 (internal use)	5.2 Vdc +/- 2%
Current	3 A Maximum
Ripple:	100 mVpp Maximum
Indicator	+5 Vdc (Green LED)
Hold up Time	> 40 ms @ 120 Vac [Full Load]
b) Output 2 (external use)	24 Vdc +/- 10%
Current	300 mA Maximum
Ripple	200 mVpp Maximum
Short Circuit Current	700 mA
Indicator	+24Vdc (Green LED)

ISOLATION	
Input signal, internal outputs and the external output are isolated among them.	
Between the outputs and the ground	1000 Vrms
Between the input and output	2500 Vrms

FAILURE RELAY	
Type of Output	Solid state relay, normally closed (NC), isolated
Limits	6 W, 30 Vdc Max, 200 mA Max
Maximum Initial Contact Resistance	<13Ω
Overload Protection	Should be provided externally
Operation Time	5 ms maximum

TEMPERATURE	
Operation	-10 °C to 60 °C (14 °F to 140 °F)

DIMENSIONS AND WEIGHT	
Dimensions (W x H x D)	39.9 x 137.0 x 141.5 mm; (1.57 x 5.39 x 5.57 in)
Weight	0.450 kg

CABLES	
One wire	14 AWG (2 mm ²)
Two wires	20 AWG (0.5 mm ²)

NOTES	
<ol style="list-style-type: none"> 1) If the power consumption exceeds the power supplied, the DFI302 system may operate in an unpredictable manner that may causes damages to the equipment or risk of personal injury. Hence, the power consumption must be calculated correctly and install more power supplies modules, if it is necessary. 2) To increase the service life of your contacts and protect the modules from potential reverse voltage damage, connect externally a clamping diode in parallel with each inductive DC load or connect an RC snubber circuit in parallel with each inductive AC load. 3) The redundancy feature is only guaranteed for racks with GLL1270 Revision 2 or greater. For the models with their revisions less than the mentioned above, the technical support must be consulted in order to check the compatibility. 4) To meet the EMC standards requirements, the wires' length to the failure relay must be less than 30 meters. The power supply of activated load by the failure relay must not be from external network. 	

DF56 – Power Supply for Backplane (Redundant)

Description

This redundant power supply works independently or with another redundant power supply module to assure a constant power supply to the application. When two redundant power supply modules are used, both split the energy that is needed to supply the system. When one power supply fails, the other, automatically, will assume the operation. Each power supply has a relay to indicate failures allowing the user to replace damage modules.

This module has two voltage outputs:

a) **5 Vdc @ 3A** distributed by power lines in the Inter-Module-Bus (IMB) through racks to supply module circuits;

b) **24 Vdc @ 300 mA** for external use through terminals 1B and 2B.

The DC applied voltage and the 5Vdc and 24 Vdc are isolated.

Installation and Configuration

For systems based on DF92 and DF93 rack, with DF90 and DF91

Redundant mode

Splitting Power concept: In this situation, two modules will supply power to a bus segment. If one of them was turned off or fails, the other power supply must be able to supply energy, alone, to the segment.

Expansion of load capacity by adding power supplies or pairs of redundant power supplies

If the system consumption is greater than 3A, it can be subdivided in up to 8 groups sized for consumption of up to 3A each, and each group is individually powered by a power supply, or redundant pair of power supplies. More details on the Power supplies positioning topic.

Power supplies positions in the racks

On DF92, the pair of redundant power supplies must be installed in the first and second slots.

On DF93 is recommended the placement of the redundant pair in the first and second slots, but it can be installed in any slots if necessary.

Configuration of CH1 jumper

The DF56 **CH1** jumper always must be connected to the **R** position.

For systems based on DF1A and DF78 racks

Single Module: Less than 3 A are required.

There is an address restriction related to the location of the power supply. This restriction is the first rack (address 0) must have a power supply module in the first slot. The **CH1** jumper (power supply) must be set in the **E** position.

More Than One Module: More than 3 A are required.

For systems based on **DF1A rack** they must be placed in the first slot of the rack. Jumper **W1** on the rack that has the new power supply must be cut. Every new power supply will only supply energy to the rack in which it is located and, with the jumper cut off, it will not supply energy to the previous racks. All modules must have the **CH1** jumper (power supply) set in the **E** position.

Redundant Mode:

In redundant mode, the power supply modules must be placed in the first and third slots of rack **DF1A** or first and second slots of rack **DF78**. In both, the **CH1** jumper (power supply) must be set in the **R** position. In this condition, the power supply modules will split the power. This topology is called "split power mode".

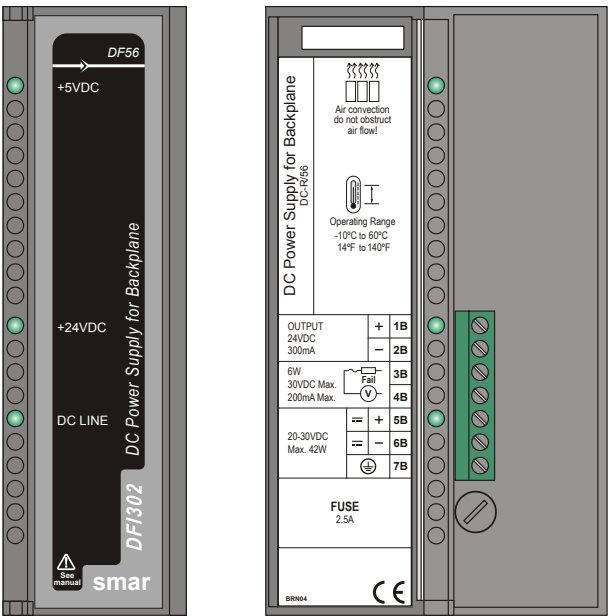


Figure 13.2 - DC Power Supply Module: DF56

Technical Specifications

INPUTS	
DC	20 to 30 Vdc
Inrush Current	< 20.6 A @ 30 Vdc [$\Delta T < 430 \mu s$]
Maximum Consumption	42 W
Indicator	DC LINE (Green LED)

OUTPUTS	
a) Output1 (internal use)	5.2 Vdc +/- 2%
Current	3 A Maximum
Ripple	100 mVpp Maximum
Indicator	+5 Vdc (Green LED)
Hold up Time	> 47 ms @ 24 Vdc [Full Load]
b) Output 2 (external use)	24 Vdc +/- 10%
Current	300 mA Maximum
Ripple	200 mVpp Maximum
Short Circuit Current	700 mA
Indicator	+24 Vdc (Green LED)

ISOLATION	
Input signal, internal outputs and the external output are isolated between them.	
Between outputs and ground	500 Vrms
Between input and output	1500 Vrms

FAILURE RELAY	
Type of Output	Solid state relay, normally closed (NC), isolated
Limits	6 W, 30 Vdc Max, 200 mA Max
Maximum Initial Contact Resistance	<13 Ω
Overload Protection	Should be provided externally.
Operation Time	5 ms maximum

TEMPERATURE	
Operation	-10 °C to 60 °C (14 °F to 140 °F)

DIMENSIONS AND WEIGHT	
Dimensions (W x H x D)	39.9 x 137.0 x 141.5 mm ; (1.57 x 5.39 x 5.57 in)
Weight	0.450 kg

CABLES	
One wire	14 AWG (2 mm ²)
Two wires	20 AWG (0.5 mm ²)

NOTES	
<p>1. If the power consumption exceeds the power supplied, the DFI302 system may operate in an unpredictable manner that may causes damages to the equipment or risk of personal injury. Therefore, the power consumption must be calculated correctly and a detailed analysis should be performed to define the installation of extra power supply modules.</p> <p>2. The hardware revisions which are GLL1279 Rev1 and previous revisions do not support redundancy feature.</p> <p>3. To meet the EMC standards requirements, the wires' length to the failure relay must be less than 30 meters. The power supply of activated load by the failure relay must not be from external network.</p>	

DF87 – Power Supply for Backplane (5 A, Redundant, with diagnostic)

Description

This redundant power supply works independently or with another redundant power supply module to assure a constant power supply to the backplane. When two redundant power supply modules are used, both split the energy that is needed to supply the system. When one power supply fails, the other, automatically, will assume the operation.

This module provides a 5 Vdc output voltage, isolated from the input, with capacity of 5 A.

The DF87 has advanced diagnostics, which are indicated by LEDs, and can be read by the DFI302 controller. It also has a relay that is activated (closed) to indicate failures.

The DF87 has three ranges to signal the diagnostics. The diagnostic signal **OK** means the DF87 is operating in correct range, ensuring that is far from the fault limits. If the DF87 is out of this range, before reaching the limits that stop its operation, warning diagnostics are flagged, allowing intervention before potential failures may occur. If the fault limits are reached, the DF87 stops the operation, disconnecting to the bus. Thus, the failure does not affect the performance of redundancy, the failure relay is activated (closed), and the possible causes of failures are indicated.

Installation and Configuration

Operation without Redundancy

Each DF87 powers one bus segment.

Redundant mode

Two modules will supply power to a bus segment. If one of them was turned off or fails, the other power supply must be able to supply energy, alone, to the segment.

For systems based on DF92 and DF93 rack, with DF90 and DF91

Expansion of load capacity by adding power supplies or pairs of redundant power supplies

If the system consumption is greater than 5A, it can be subdivided in up to 8 groups sized for consumption of up to 5A each, and each group is individually powered by a power supply, or redundant pair of power supplies. More details on the Power supplies positioning topic.

Power supplies positions in the slots

On DF92, the pair of redundant power supplies must be installed in the first and second slots.

On DF93 is recommended the placement of the redundant pair in the first and second slots, but it can be installed in any slots if necessary.

For systems based on DF1A

Expansion of load capacity by adding power supplies or pairs of redundant power supplies

With the DF1A is possible reach up to 3A per slot. If the system consumption is greater than 3A, it can be subdivided in up to 8 groups sized for consumption of up to 3A each, and each group is individually powered by a power supply, or redundant pair of power supplies. More details on the Power supplies positioning topic.

Power supplies positions in the slots

On DF1A, the pair of redundant power supplies must be installed in the first and third slots.

**ATTENTION**

- The power supply DF87 is not compatible with the DF78 rack. Use the DF92 rack if redundant controllers are needed together with the DF87.
- Even using the power supply DF87, the DF1A rack only supports 3A per slot.

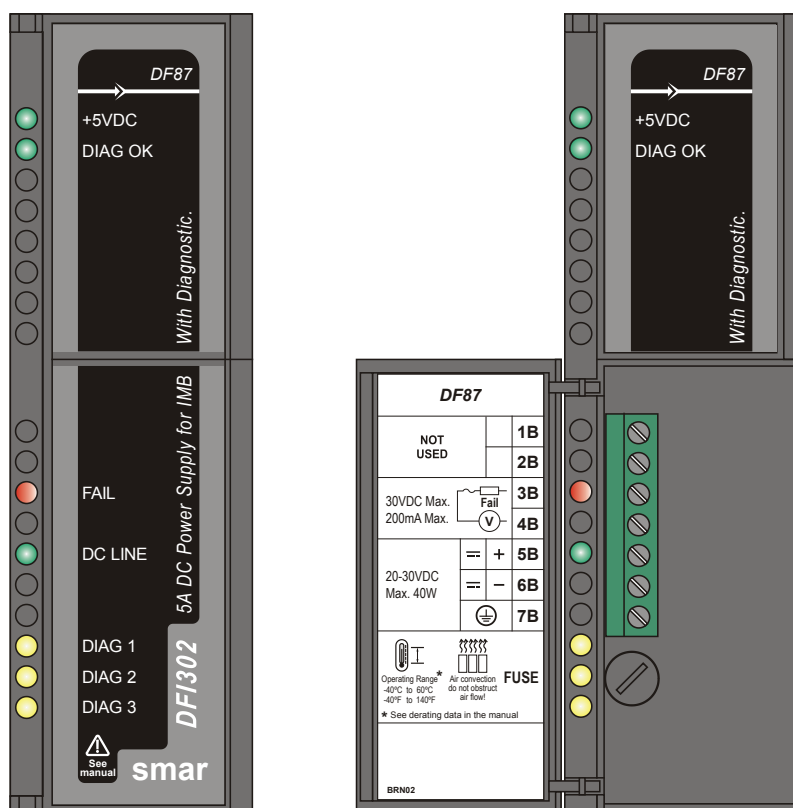


Figure 13.3 - DC Power Supply Module: DF87

Technical Specifications

INPUTS	
DC	20 to 30 Vdc 24 Vdc nominal
Maximum Consumption	40 W (@ 5A output)
Indicator	DC LINE (Green LED)
Fuse	8A 250V 5X20mm
OUTPUTS	
Output (internal use)	5.2 Vdc +/- 2%
Current	5 A Maximum (See derating curve in the figure 13.6)
Ripple	100 mVpp Maximum
Indicator	+5 Vdc (Green LED)
Hold up Time	> 4.7 ms @ 24 Vdc [Full Load]
ISOLATION	
Between outputs and ground	1500 Vdc
Between input and output	1500 Vdc

FAILURE RELAY	
Type of Output	Solid state relay, normally closed (NC), isolated
Limits	6 W, 30 Vdc Max, 200 mA Max
Maximum Initial Contact Resistance	<13 Ω
Overload Protection	Should be provided externally.
Operation Time	12 ms maximum

TEMPERATURE	
Operation	-40 °C to 60 °C (-40 °F to 140 °F) (See derating curve in the figure 13.6)

DIMENSIONS AND WEIGHT	
Dimensions (W x H x D)	39.9 x 137.0 x 141.5 mm ; (1.57 x 5.39 x 5.57 in)
Weight	0,453 Kg

CABLES	
One wire	14 AWG (2 mm ²)
Two wires	20 AWG (0.5 mm ²)

NOTE	
To meet the EMC standards requirements, IEC 61326, the wires' length to the failure relay must be less than 30 meters. The power supply of the load activated by the failure relay must not be from external network.	
If the power cables of the input are greater than 3 m, install the ferrite core "FAIR-RITE V0", attached to the product packing. To install it, involve with the ferrite core all cables that are connected to the 5B, 6B and 7B contacts of the front terminal block near to DF87.	

Diagnostics LEDs

The power supply DF87 has the following frontal LEDs, indicating the following situations shown in the figure below.

● +5VDC	Power converter on
● DIAG OK	Operation OK
● FAIL	Fail
● DC LINE	Input power on
● DIAG 1	Diagnostic code
● DIAG 2	Diagnostic code
● DIAG 3	Diagnostic code

Figure 13.4 - DF87 frontal LEDs

The following is a summary of situations and the status of the LEDs for warning diagnostics, allowing intervention before potential failures may occur in the DF87 power supply.

OK	Input Voltage Low	Input Voltage High	Output Current	Internal Temperature	Unrecognized	Out Protection Acting	Internal Problem (ripple, etc)
● DIAG OK	● DIAG OK	● DIAG OK	● DIAG OK	● DIAG OK	● DIAG OK	● DIAG OK	● DIAG OK
● DIAG 1	● DIAG 1	● DIAG 1	● DIAG 1	● DIAG 1	● DIAG 1	● DIAG 1	● DIAG 1
● DIAG 2	● DIAG 2	● DIAG 2	● DIAG 2	● DIAG 2	● DIAG 2	● DIAG 2	● DIAG 2
● DIAG 3	● DIAG 3	● DIAG 3	● DIAG 3	● DIAG 3	● DIAG 3	● DIAG 3	● DIAG 3

Figure 13.5 - Diagnostics LEDs

The FAIL LED indicates failure when is ON.

The following graph shows the behavior of the output current within the operation range of the DF87 in environments without artificial ventilation.

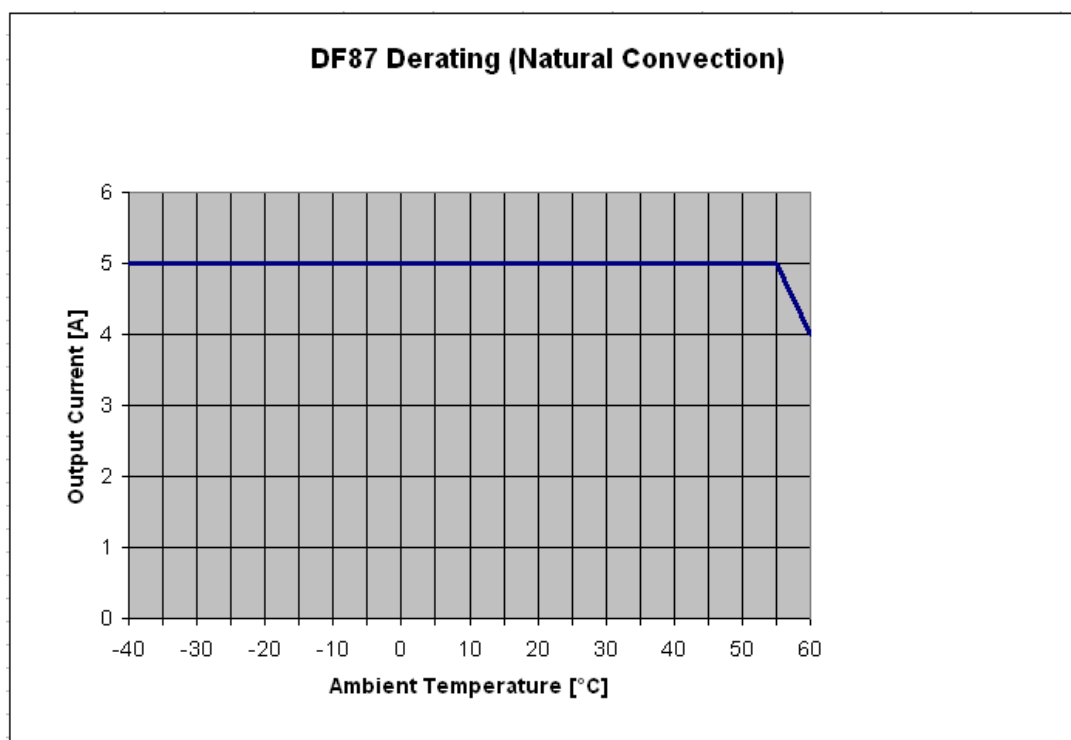


Figure 13.6 - Derating curve of the output current versus temperature, for environments without ventilation

Calculating the Power Consumption

Since the power available in the power supply is limited, it is important to calculate the power consumption of modules in use. The user can create a worksheet to summarize all supplied and required current from each module and associated equipment (such as operator interface).

Example of worksheet with the module's consumption, and some power supplies' specification.

DFI302 Power Budget										
Module	Description	Qty.	Consumption Unit Power (mA)		Total Power (mA)		Supply Unit Power (mA)		Total Power (mA)	
			@24 V	@5 V	@24 V	@5 V	@24 V	@5 V	@24 V	@5 V
DF51	Controller	1	0	950	0	950				
DF62	Controller		0	550	0	0				
DF63	Controller		0	550	0	0				
DF73	Controller		0	650	0	0				
DF75	Controller		0	550	0	0				
DF11	2*8 DI 24 Vdc		0	80	0	0				
DF12	2*8 DI 48 Vdc		0	80	0	0				
DF13	2*8 DI 60 Vdc		0	80	0	0				
DF14	2*8 DI 125 Vdc		0	80	0	0				
DF15	2*8 DI 24 Vdc (sink)		0	80	0	0				
DF16	2*4 DI 120 Vac		0	50	0	0				
DF17	2*4 DI 240 Vac		0	50	0	0				
DF18	2*8 DI 120 Vac		0	87	0	0				
DF19	2*8 DI 240 Vac	2	0	87	0	174				
DF20	8 switches		0	45	0	0				
DF44	8 AI		0	320	0	0				
DF57	8 AI		0	320	0	0				
DF45	8 Temperature inputs		0	55	0	0				
DF21	16 DO (transistor)		65	70	0	0				
DF22	2*8 DO (transistor)		65	70	0	0				
DF23	8 DO (triac)		0	70	0	0				
DF24	2*8 DO (triac)		0	115	0	0				
DF25	2*4 DO (relay)		134	20	0	0				
DF26	2*4 DO (relay)		134	20	0	0				
DF27	2*4 DO (relay)		134	20	0	0				
DF28	2*8 DO (relay)		180	30	0	0				
DF29	2*4 DO (relay)		134	20	0	0				
DF30	2*4 DO (relay)		134	20	0	0				
DF31	2*4 DO (relay)		134	20	0	0				
DF46	4 AO		180	20	0	0				
DF32	8 DI 24 Vdc, 4 DO (relay)		67	60	0	0				
DF33	8 DI 48 Vdc, 4 DO (relay)		67	60	0	0				
DF34	8 DI 60 Vdc, 4 DO (relay)		67	60	0	0				
DF35	8 DI 24 Vdc, 4 DO (relay)		67	60	0	0				
DF36	8 DI 48 Vdc, 4 DO (relay)		67	60	0	0				
DF37	8 DI 60 Vdc, 4 DO (relay)		67	60	0	0				
DF38	8 DI 24 Vdc, 4 DO (relay)		67	60	0	0				
DF39	8 DI 48 Vdc, 4 DO (relay)		67	60	0	0				
DF40	8 DI 60 Vdc, 4 DO (relay)		67	60	0	0				
DF53	4 Fieldbus Power Impedance	1	1500	0	1500	0				
TOTAL		4			1500	1074				
DF50		1					300	3000	300	3000

DFI302 Power Budget										
Module	Description	Qty.	Consumption Unit Power (mA)		Total Power (mA)		Supply Unit Power (mA)		Total Power (mA)	
			@24 V	@5 V	@24 V	@5 V	@24 V	@5 V	@24 V	@5 V
DF52		1					1500	0	1500	0
TOTAL		6							1800	3000

Power supplies and racks positioning

For systems based on DF92 and DF93 racks with DF90 and DF91

A power supply connected to a rack, in a system, provides current to the racks row that are horizontally interconnected to it by their terminals of lateral connections, and vertically through DF90 cables, thus forming a group of rows of racks that use the same power supply.

The system can have only one power supply (or pair of redundant power supplies) or it can be subdivided in several of these groups¹, each one powered by a power supply (or pair of redundant power supplies).

The recommended way to distribute the power is to divide the system in groups of horizontal rows of racks. In this scheme, each power supply must be positioned on the top left of the group of rows of racks that it powers. The rack where is the power supply must be the **W1** jumper cut and the DF90 cable must not be connected to the rows powered by other power supplies (top rows). See in the following figure an example of system powered by two power supplies, each one powers a part of rows represented in green and blue.

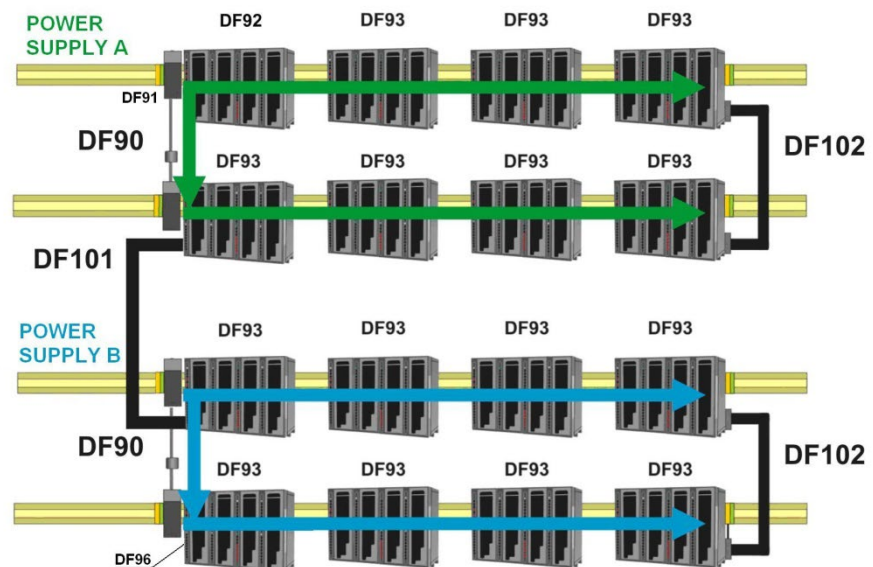


Figure 13.7 - System powered by two power supplies

Note that this system, for greater efficiency, is optimized for power distribution by groups of rows of racks. Thus, a power supply powers a whole number of rows it supports. However, in rare cases, with long rows or many modules with great consumption in the same row, there is the option to add power supplies in the middle of the row, dividing the power within this row. In this case, the power supply added powers only the modules positioned on the right in the same row, up to the end, or even where there is another power supply added. In the rack where the power supply was added, in this scheme, the **W1** jumper must be cut and left lateral connection terminal (+5 Vdc) must be disconnected (collapsed).

¹ Maximum 8 groups allowed when the DF50, DF56 or DF87 power supplies are used.

The system has diagnostic for voltage level distributed to racks. It also supports modules with great power consumption in any place on the bus. Nevertheless, is recommended to place those modules close to the power supplies, to avoid unnecessary power transmission.

Practical Rules for Projects

1. The system can have up to 17 racks (up to 16 DF93 racks and up to 1 DF92 rack).
2. A group of racks that consumes up to 5A of the same power supply must be physically placed within the limits specified in the table below, respecting the limit of 17 racks of previous item and the limits of power supply used. So that it is possible to power complete typical systems with a single power supply or redundant pair of power supplies.

N° of rows	Limited in:
1 row	Up to 12 racks in the row ²
2 rows	Up to 9 racks on each row
3 rows	Up to 6 racks on each row
4 rows	Up to 5 racks on each row
5 rows	Up to 4 racks on each row

Table 1 - Limits for systems with DF90, DF91, DF92 and DF93, with consumption of up 5A

3. Place the power supply always at the top left of the group that it powers.
4. Modules with consumption greater than 320 mA must be placed in the row where the power supply is, in the 4 racks closest of it.
The other modules can be in any position, but is recommended to place the modules with great consumption close to the power supplies, to avoid unnecessary power transmission.
5. The rack where is the power supply must be the **W1** jumper cut. The power supply of this rack must not be connected to racks powered by other power supplies (disconnect the DF90 cable or the Vcc connection plate which would connect it to another power supply).
6. On DF92, the pair of redundant power supplies must be installed in the first and second slots.
7. On DF93 is recommended the placement of the redundant pair in the first and second slots, but it can be installed in any slots if necessary.
8. In this system, **DF50** and **DF56** must be their **CH1** jumper always configured in **R**, even if they are not in redundant pairs.



ATTENTION

A mixture of these power supplies with the **CH1** configured in **R** and in **E** in any **DFI302** system, is not allowed!

Using third-party power supplies

The STEP OS 1AC 5DC 6.5 power supply from Phoenix Contact can be used to power a DFI302 system with DF90, DF91, DF92 and DF93, following the additional rules specified below:

1. Racks positioned according to the following limitations, for up 5A:

N° of rows	Limited in:
3 rows	4 racks in the row

2. Adjust the power supply output voltage in $5.19V \pm 0.8\%$ ³ (with current of $0.5A \pm 10\%$, at $25^{\circ}C \pm 10^{\circ}C$)

² If the system consumes up to 3A, a single row of up to 17 racks can be powered by a single power supply located on the first rack.

**ATTENTION**

Voltage above 6V can damage the system. Check the voltage of the third-party power supply before connecting it to the system.

3. The power supply output must be connected to DF91 of the first row, using one of the cable type recommended above, with a ferrite installed in this cable, near to DF91. This ferrite is the same used in the DF90 cable, (code 0431164181 from the manufacturer FAIR-RITE). The recommended cables are:

- The DF90;
- or up to 0.43 m of copper cable, parallel or twisted, 4 mm² gauge;
- or up to 0.64 m of copper cable, parallel or twisted, 6 mm² gauge;
- or up to 1.07 m of copper cable, parallel or twisted, 10 mm² gauge.

4. To use redundant power supplies, connect them to the same DF90.

For systems based on DF78 and DF1A Racks (legacy systems)

1. Observe the maximum current values from the power supply module specification. The limit for DF50 is 3 A, and for DF87 is 5 A.

2. After the connection with long cables (DF4A, DF5A, DF6A and/or DF7A) you have to put another power supply module in the first slot of the first rack.

3. Use up to 6 modules DF44/DF57 per power supply; always place consecutively the DF44/DF57 and close to the power supply. Because of the high current consumption of the modules DF44/DF57, a not desired voltage drop in the bus can occur if these modules are placed after other modules.

4. When is necessary to add interface modules, such as HI302, MB700, DF58, in the same bus which is used by output and input modules, is recommended that these modules are placed close to the power supply, because in the same way as described in the previous item, a not desired voltage drop in the bus can occur if these modules are placed after other modules.

5. Adding a new power supply module

- Determine the rack where the new power supply will be installed.
- Cut the jumper **W1** of the rack.
- Plug the new power supply at the first slot of the rack (slot 0).
- In this case, the **CH1** jumper of all modules **DF50** must be set in **E** position.

**ATTENTION**

- The power supply DF87 is not compatible with the DF78 rack. Use the DF92 rack if redundant controllers are needed together with the DF87.
- Even using the power supply DF87, the DF1A rack only supports 3A per slot.

³ Example of calibration procedure of the STEP OS 1AC 5DC 6.5.power supply from Phoenix Contact to use it with DF1302:
Adjust the power supply output between 5.17V to 5.21V, trying to get 5.19V (using voltmeter with a maximum of 0.4 % of total uncertainty – e.g.. multimeters Fluke 83III, 85 III, or 87 III, with annual calibration), at ambient temperature of 25°C ± 10°C.

Fault diagnostic in IMB

IMPORTANT

The diagnostic and security feature **IMB Fault State**, in this section, is available for the following controllers and their firmware versions:

DF62/DF63/DF75: from firmware version V4_0_5 and higher

DF73/DF95/DF97: from firmware version V4_1_0 and higher

DF89: from firmware version V4_2_0 and higher

The **IMB Fault State** feature allows detecting undervoltage conditions at the IMB bus when the DF93 rack, which supports diagnostic, is used.

In the **IMB Fault State** condition the access to I/O cards can be harmed. Thus, once detected this fault condition, the controller shall take the following actions:

- Total interruption of access to I/O cards;
- Signaling the condition in Force LED (FRC) of the frontal, blinking every 2s;
- Indication on **Resource** function block:
 - **BLOCK_ERR** parameter, with **DeviceFaultState** value;
 - **FAULT_STATE** parameter with **Active** value;
- Indication of bad (value 1) in the **Status** block of ladder logic (requires that there be some I/O card with recognized ID and being accessed in the same rack);
- Indication of racks statuses via **SNMP OPC Server** (pwr subgroup).

Conditions that may cause IMB Fault State:

- DF50 power supplies distributed over the IMB racks are undersized;
- Turning off some of the power supplies over the IMB racks;
- Turning off some of the connections between racks;
- Failures of hardware elements that may cause current leakage.

The IMB Fault State feature must be enabled on the **Resource** function block by setting the **FEATURE_SEL** parameter with **Fault State** bit. This configuration can be done via **Online Characterization** and does not require a new configuration download.

DF52 / DF60 – Power Supply for Fieldbus

Description

These modules were specially designed to supply the Fieldbus networks. The only difference between them is the input voltage:

DF52 (90 ~ 264 Vac)

DF60 (20 ~ 30 Vdc)

The **DF52** power supply is a non-intrinsically safe equipment with an universal AC input (90 to 264 Vac, 47 to 63 Hz or 127 to 135 Vdc), and a 24 Vdc output, isolated, with short circuit and overcurrent protection, ripple and fault indication, proper to power fieldbus elements.

The **DF60** power supply unit is a non-intrinsically safe equipment with a DC input (20 to 30 Vdc) and a 24 Vdc output, isolated, with short circuit and overcurrent protection, ripple and fault indication, proper to power fieldbus elements.

The interconnection of Fieldbus elements to the **DF52/DF60** is indicated in figure bellow. There is no overshoot when it is switched on or off. The **DF52/DF60** can power up to 4 fully loaded fieldbus networks.

NOTE

The length of the cables that interconnect the DF52/DF60 to the DF53/DF98 modules must not exceed 3 meters.

If any abnormal condition occurs in the output, such as overloading or short circuit, the **DF52/DF60** internal switches are automatically switched off to protect its circuit. When the outputs return to normal operation conditions, the circuit is automatically switched on.

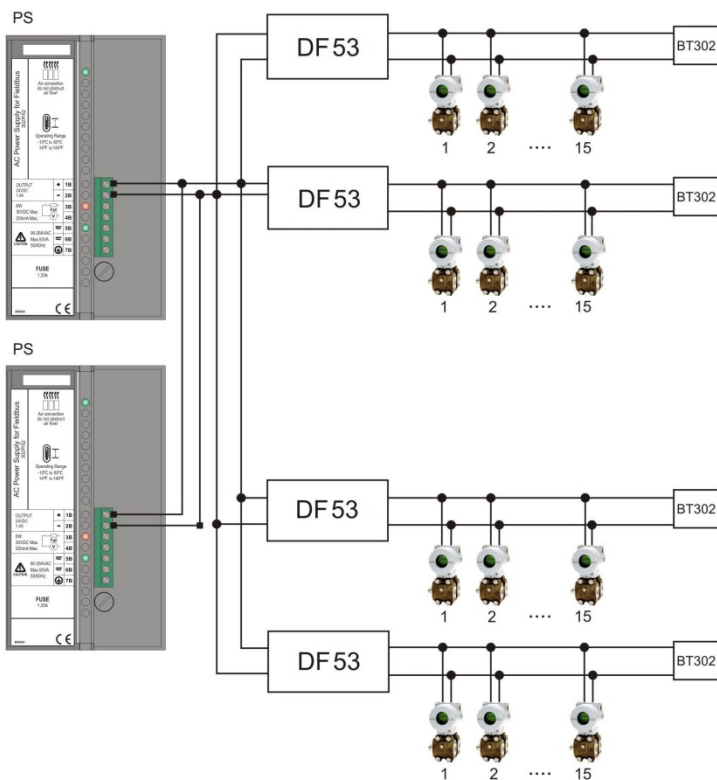


Figure 13.8 - System powered by DF52

The **DF52/DF60** modules allow redundancy without any component connected to their outputs.

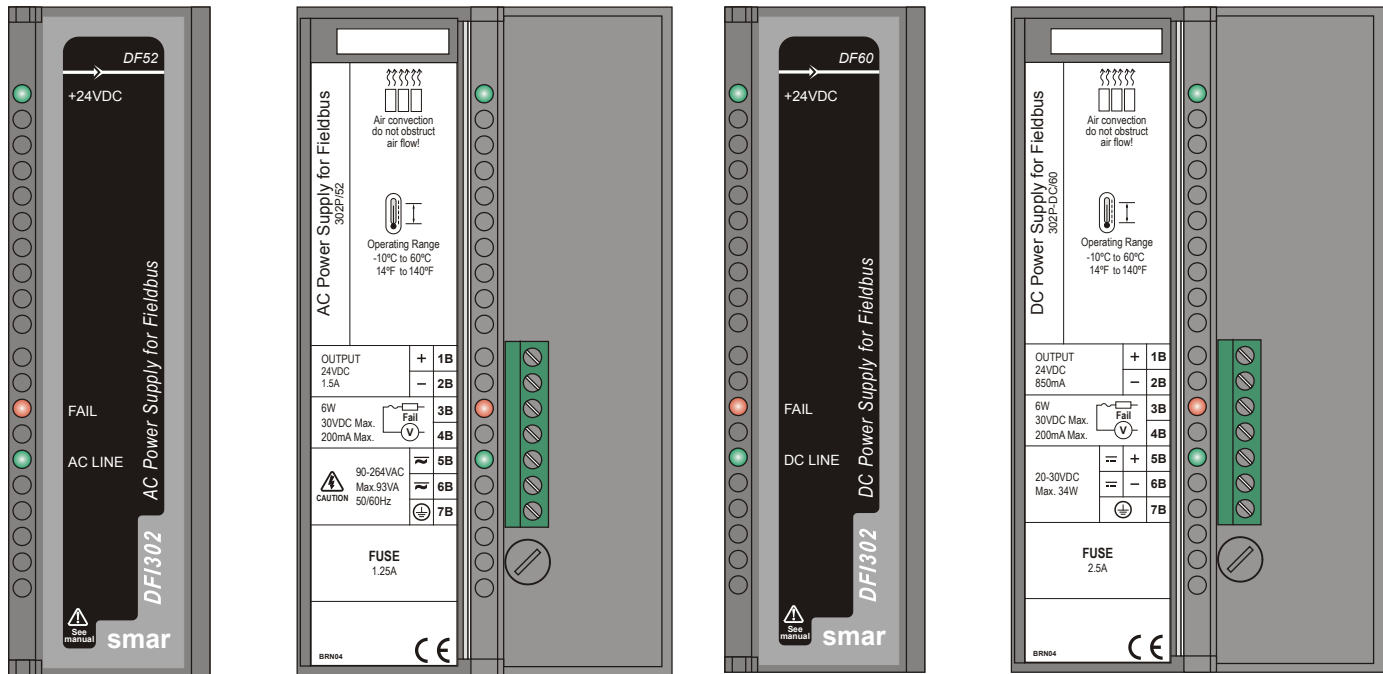


Figure 13.9 - Power Supply Module for Fieldbus: DF52/DF60

Technical Specifications

INPUTS DF52		
DC	127 to 135 Vdc	
AC	90 to 264 Vac, 50/60 Hz (nominal), 47 to 63 Hz (range)	
Maximum Inrush Current	< 30 A @ 220 Vac [ΔT < 640 μs]	
Maximum Consumption	93 VA	
Indicator	AC LINE (Green LED)	

INPUTS DF60		
DC	20 to 30 Vdc	
Maximum Inrush Current	< 24 A @ 30 Vdc [ΔT < 400 μs]	
Maximum Consumption	34 W	
Indicator	DC LINE (Green LED)	

OUTPUTS		
Output	24 Vdc ± 1%	
Current	DF52	DF60
	1.5 A (maximum)	850 mA (maximum)
Ripple	20 mVpp (maximum)	
Indicators	+24 Vdc (Green LED)	
	FAIL (Red LED)	

ISOLATION		
Input signal, internal outputs and the external output are isolated among them	DF52	DF60
Among outputs and ground	1000 Vrms	500 Vrms
Between input and output	2500 Vrms	1500 Vrms

FAILURE RELAY	
Type of Output	Solid state relay, normally closed (NC), isolated
Limits	6 W, 30 Vdc Max, 200 mA Max
Maximum Initial Contact Resistance	<13Ω
Overload Protection	Should be provided externally
Operation Time	5 ms maximum

DIMENSIONS AND WEIGHT	
Dimensions (W x H x D)	39.9 x 137.0 x 141.5 mm ; (1.57 x 5.39 x 5.57 in)
Weight	0.450 kg

TEMPERATURE	
Operation	-10 °C to 60 °C (14 °F to 140 °F)
Storage	-30 °C to 70 °C

NOTE	
To meet the EMC standards requirements, the wires' length to the failure relay must be less than 30 meters. The power supply of activated load by the failure relay must not be from external network.	

DF53/DF98 – Power Supply Impedance for Fieldbus

Description

These modules were specially designed to provide appropriate impedance for fieldbus networks.

DF53 (4 ports)

DF53-FC (4 ports)

DF98 (2 ports)

The purpose of this impedance is to implement an output circuit where the impedance is greater than $3\text{ K}\Omega$, and when assembling in parallel with two $100\ \Omega \pm 2\%$ terminators, it results in a $50\ \Omega$ line impedance approximately. This impedance can be implemented in a passive mode ($50\ \Omega$ resistance in series with a 100 mH inductance) or in an active mode, through an impedance control circuit.

The fieldbus power supply impedance is a non-isolated, active impedance control device, in compliance with IEC 61158-2 standard. This device provides an output impedance which, in parallel with the two bus terminators (a $100\ \Omega$ resistor in series with a $1\ \mu\text{F}$ capacitor) required by the standard, results in a pure resistive line impedance for a broad frequency range. The **DF53** and **DF98** cannot be used in intrinsic safety areas.

The figure shows the device block diagram. The **DF53/DF98** can be used in redundancy, connecting its output (+ and -) in parallel. In this case, use an external bus terminator (**BT302**) to allow maintenances or replacing the **DF53/DF98** in case of failure without interrupting the fieldbus communication.

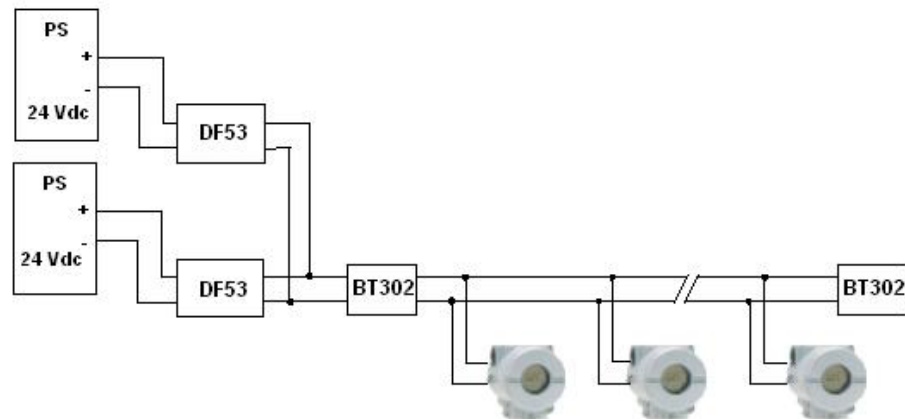


Figure 13.10 – System using the impedance DF53

The **DF53/DF98** modules have LEDs to indicate power supply and overcurrent. The input terminal block has two terminals (1A and 2A) that are connected to the external 24 Vdc . The power supply indication LED is green and it is energized while there is an external 24 Vdc power supply.

The overcurrent indication LED is red and it is energized only in case of an overcurrent caused by a short circuit in the plant or by an excessive number of devices connected. The following figure shows a **DF53/DF98** layout.

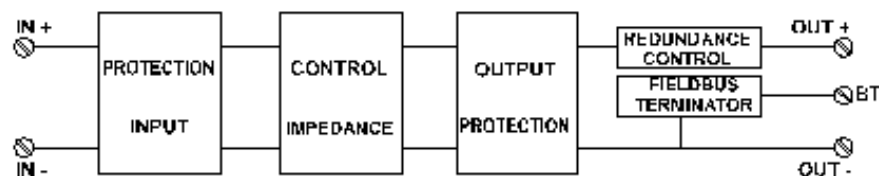


Figure 13.11 – DF53/DF98 block diagram

DF53: Eight terminals (3A to 10A) implementing four independent Fieldbus ports, four DIP switches for activating the bus termination, one green LED for power status, and four red LEDs indicating overcurrent.

DF53-FC: It has the same characteristics of DF53 and meets the requirements for hardware tests of OIML R117-1 (Flow Measurement System of Liquids).

DF98: Four terminals (3A/4A and 9A/10A) implementing two independent Fieldbus ports, two DIP switches for activating the bus termination, one green LED for power status, and two red LEDs indicating overcurrent.

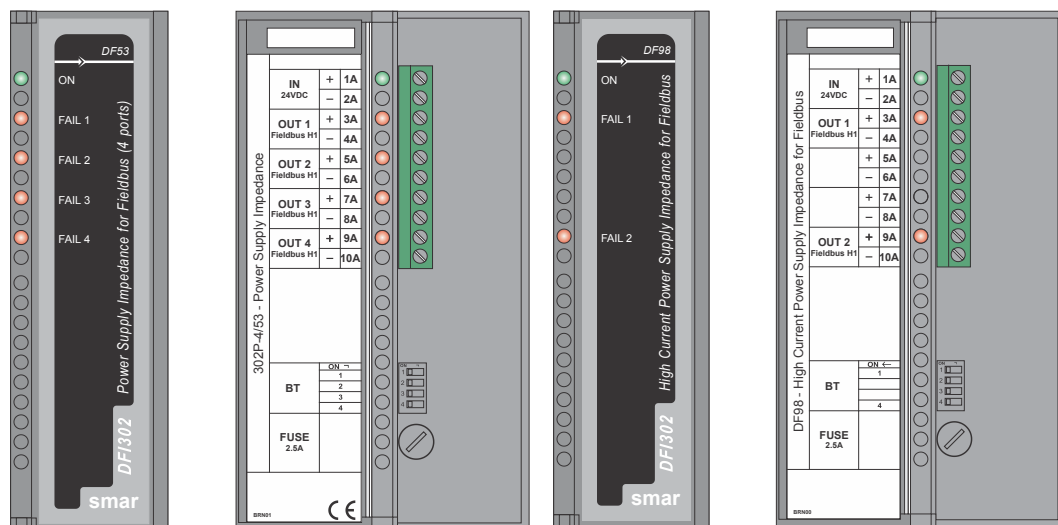


Figure 13.12 – Impedance for Fieldbus Module: DF53/DF98

Technical Specifications

INPUT	
DC	24 to 32 Vdc +/- 10%
OUTPUT	
Current	DF53: 340 mA per channel DF98: 500 mA per channel
INPUT FILTER	
Attenuation	10 dB in the input power ripple @60 Hz
CONSUMPTION	
Maximum power dissipated	DF53: 2,26 W per channel DF98: 3,43 W per channel
DIMENSIONS AND WEIGHT	
Dimensions (W x H x D)	39.9 x 137.0 x 141.5 mm; (1.57 x 5.39 x 5.57 in)
Weight (without package)	260 g
TEMPERATURE	
Operation	0 °C to 60 °C
Storage	-30 °C to 70 °C

SAFETY	
Output Overcurrent	DF53: 450 mA DF98: 600 mA
Input Fuse	2.5 A
Atmospheric Discharges	Input and output protected by transient suppressors
Intrinsic Safety	It cannot be applied directly

MAXIMUM LENGTH OF FIELDBUS WIRING		
DF53/DF98	No redundancy	1.900 m
	Redundant	1.900 m
DF53-FC	No redundancy	1.900 m
	Redundant	1.000 m

Installation

The **DF53/DF98** is a device specially designed for panel installation and it cannot be installed in unsheltered locations, as it cannot be exposed directly to the weather. The module can be connected to the panel directly on the DIN rail or using the auxiliary support provided with the module, fixed with screws. Refer to the "Installing" section for further details about installation and dimensional drawings of the module.

Maintenance and troubleshooting

The **DF53/DF98** is a robust device which basically requires no preventive maintenance. It is simply recommended to protect it from excessive dust accumulation and humid environments which might affect its output impedance.

The modules have LEDs which inform their operation status – one green LED which informs that the module is properly powered, and one red LED for each channel, that will be lit if any abnormal condition occurs in the field wiring.

These LEDs detect most of the problems which may occur in a Fieldbus installation. However, they might not detect other problems, such as:

- Excessive noise caused by the external power supply;
- Impedance lower than 20Ω in the communication line (note that such impedance may not be pure resistive and, therefore not detectable by the overcurrent circuit).

Such abnormal conditions may be easily detected by measurement instruments.

Because the **DF53/DF98** is a simple and compact device, it is recommended to replace faulty modules instead of electronic components during repair services.

DF47-12 and DF47-17 – Intrinsic Safety Barrier for Fieldbus

Description

The Intrinsically Safe (I.S.) technology incorporated in the DF47-12 and DF47-17 totally isolates the control network on the hazardous side of the barrier. The I.S. values of the power supply are designed for fieldbus devices, which are in compliance with the FISCO model.

The incorporation of a fieldbus repeater in compliance with IEC 61158-2 (31.25 kbits/s) essentially filters and boosts the incoming communication signal transmitting it to hazardous environment. The networks of the hazardous and safe sides of the DF47-12 and DF47-17 are completely independent from one another.

In addition the bus termination for the hazardous network is incorporated into the DF47-12 and DF47-17, which means that only a single far terminator is required.

NOTES

1. If the terminator of the DF53 module is not being used, it is necessary to install another external terminator in the safe area.
2. The model DF47 was discontinued due to the new FISCO requirements. The replacement by DF47-12 or DF47-17 models should be evaluated according to the current limits. The model DF47-17 supports up to 7 Smar's devices of the 302 series. If the replacement is using the DF47-12 model, it supports up to 5 Smar's devices of the 302 series.

- H1 Isolated Barrier and IS Power Supply in compliance with the FISCO Model.
- H1 Fieldbus Signal Repeater.
- In compliance with the IEC 61158-2, 31.25 kbits/s standard for fieldbus. (FOUNDATION fieldbus and PROFIBUS PA).
- IEC, FM & CENELEC Intrinsic Safety standards certified.
- In compliance with IEC 60079-27, FISCO and FNICO Power Supply.
- Dual Marking in compliance with IEC 60079-11 and IEC 60079-27.
- Bus terminator on hazardous area.

Installation

The selection and installation of the barrier should always be accomplished by competent technical personnel. Please contact Smar or our local representative if further information is required. According to the standards for hazardous areas the barrier DF47-12 or DF47-17 must be installed out of hazardous area. The inputs parameters for installation in hazardous area are in the Certificates for Hazardous Areas topic.

The barrier has to be installed on DF1A, DF93 or DF9 and fixed in a DIN rail. For further details see the section Installing.

Installation Principles

1. Ensure that there is an appropriate separation of intrinsically safe and non-intrinsically safe circuits (more than 50 mm or 1.97 inches), so the ignition energy from non-intrinsically safe circuit does not intrude into the intrinsically safe circuit.
2. Ensure that the limiting parameters of system design, for example total inductance and capacitance, upon which system approval is based are not exceeded.
3. Ensure that power system faults and ground potential differences do not generate system ignition.

Location

The barrier is normally installed in a dust-free and moisture-free enclosure located in the non-hazardous area. The enclosure should be as close as possible to the hazardous area to reduce cable runs and increased capacitance. If the barrier is installed in a hazardous area, it must be in a proper enclosure suited for the intended area. The only intrinsically safe terminals are at the barrier output.

Wiring

Intrinsically safe circuits may be wired in the same manner as conventional circuits installed for hazardous areas with two exceptions summarized as separation and identification. The intrinsically safe conductors must be separated from all other cables by placing them in separate conduits or by a separation of more than 50 mm or 1.97 inches of air. The raceways, cable trays, open wiring, and terminal boxes must be labeled "Intrinsically Safe Wiring" to prevent interference with other circuits.

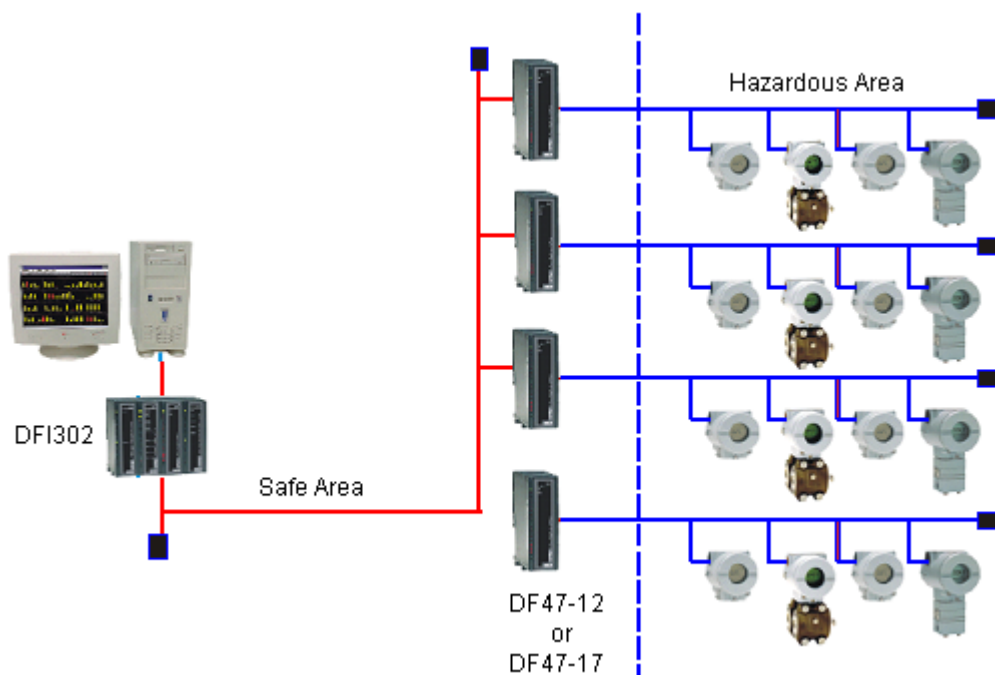


Figure 13.13 – DF47 installation

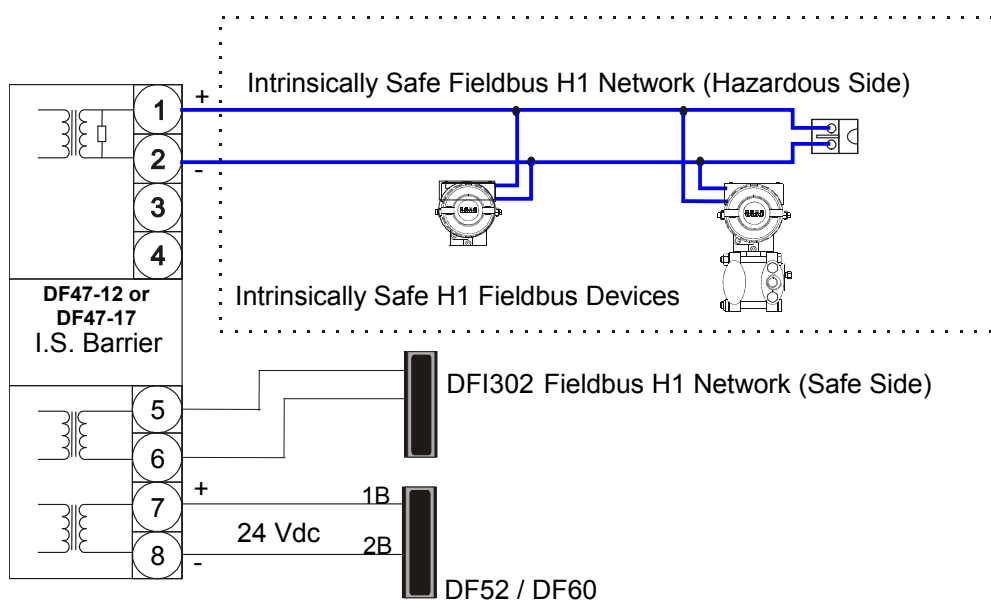


Figure 13.14 – DF47 installation

Technical Specifications

POWER	
Power Supply Input	Voltage: 24Vdc \pm 5%
	Current (max.): 350mA @ 24Vdc

HAZARDOUS AREA	
Power Supply Output	Maximum voltage available at the barrier terminals: $U_s = 13.8$ Vdc
	Maximum Current in typical operation (considering $U_s = 13.8$ Vdc) DF47-12: $I_s = 65$ mA DF47-17: $I_s = 90$ mA
	Current limiting resistor (typical) DF47-12: $R_i \geq 247.5 \Omega$ DF47-17: $R_i \geq 176.22 \Omega$
	Maximum output power DF47-12: $P_o = 1.2$ W DF47-17: $P_o = 1.72$ W
Safety Parameters (Hazardous Area)	Refer to the item "Hazardous locations approvals"
Internal Dissipation	3W maximum at 24Vdc input, nominal conditions (for non intrinsically safe circuits)
Cable Length, Number of Devices	Maximum cables lengths are determined by IS requirements, and depend on both the number of devices attached and the maximum acceptable voltage drop along the cable. Use FISCO cable.
Digital Signal Transmission	Compatible with 31.25 kbps – Fieldbus systems.
Fuse	In order to guarantee the product safe, the internal fuse change must be executed only by the manufacturer.
Terminals	Accommodate conductors up to 2.5 mm ² (22AWG).
Isolation	2500V galvanic isolation among input, output, and power supply terminals. Tested at 1500 Vrms minimum between hazardous and safe area terminals.

ENVIRONMENTAL CONDITIONS	
Ambient Temperature	0 to +60 °C (operation)
	-30 °C to 70 °C (storage)
Humidity	5% to 95% relative humidity

IMPORTANT	
By using active junction boxes you must consider their current consumption to calculate the total consumption of segment.	

Certification Information

Approved Manufacturing Locations

Smar Equipamentos Industriais Ltda – Sertãozinho, São Paulo, Brazil
Smar Research Corporation – Ronkonkoma, New York, USA

European Directive Information

This product complies with following European Directives:

EMC Directive (2004/108/EC) - Electromagnetic Compatibility

The equipment is in compliance with the directive and EMC test was performed according to IEC standards: IEC61326-1:2005 and IEC61326-2-3:2006. See table 2 from IEC61326-1:2005.

To comply with the EMC directive the installation must follow these special conditions:

- Use shielded, twisted-pair cable for powering the instrument and signal wiring.
- Keep the shield insulated at the instrument side, connecting the other one to the ground.

ATEX Directive (94/9/EC) - Electrical equipment and protective system intended for use in potential explosive atmospheres

The EC-Type Examination Certificate had been released by Nemko AS (CE0470) and/or DEKRA EXAM GmbH (CE0158), according to European Standards.

The certification body for Production Quality Assurance Notification (QAN) and IECEx Quality Assessment Report (QAR) is Nemko AS (CE0470).

Consult www.smar.com for the EC declarations of conformity for all applicable European directives and certificates.

Hazardous locations general information

Ex Standards:

IEC 60079-0 General Requirements

IEC 60079-11 Intrinsic Safety “i”

IEC 60079-27 Fieldbus intrinsically safe concept (FISCO)

Customer responsibility:

IEC 60079-10 Classification of Hazardous Areas

IEC 60079-14 Electrical installation design, selection and erection

IEC 60079-17 Electrical Installations, Inspections and Maintenance



WARNING

Explosions can result in death or serious injury, besides financial damage.

Installation of this instrument in an explosive environment must be in compliance with the national standards and according to the local environmental protection method. Before proceeding with the installation match the certificate parameters from the barrier, cable and device according to the environmental classification.

General Notes

Maintenance and Repair

The instrument modification or replaced parts supplied by any other supplier than authorized representative of Smar Equipamentos Industriais Ltda is prohibited and will void the Certification.

Marking Label

Once a device labeled with multiple approval types is installed, do not reinstall it using any other approval types. Scratch off or mark unused approval types on the approval label.

For Ex-i protection application

- Connect the instrument to a proper intrinsically safe barrier.
- Check the intrinsically safe parameters involving the barrier, equipment including the cable and connections.
- Associated apparatus ground bus shall be insulated from panels and mounting enclosures.
- When using shielded cable, isolate the not grounded cable end.
- Cable capacitance and inductance plus C_i and L_i must be smaller than C_o and L_o of the Associated Apparatus.

For FISCO System requirements (IEC 60079-27:2008)**FISCO Power Supplies**General

The power supply shall be resistive limited or have trapezoidal output characteristic. The maximum output voltage, U_o , shall be not greater than 17.5V nor less than 14V under the conditions specified in IEC60079-11 for the respective level of protection.

The maximum unprotected internal capacitance C_i and inductance L_i shall be not greater than 5nF and 10 μ H, respectively.

The output of the power supply may be connected to earth.

No specification of the internal capacitance C_i and L_i or the maximum external parameters L_o and C_o is required on the certificate or label.

The determination of power supply output parameters shall into account the possible opening, shorting and grounding of field wiring connected to the field terminals of the apparatus.

Additional requirements of "ia" and "ib" FISCO power supplies

The maximum output current I_o for any "ia" or "ib" FISCO power supply shall be determined in accordance with IEC60079-11 but shall not exceed 380 mA.

Table 1 – Assessment of maximum output current for use with "ia" and "ib" FISCO power supplies

U_o	Permissible current , for IIC (includes 1.5 safety factor)	Permissible current , for IIB (includes 1.5 safety factor)
14V	183 mA	380 mA
15V	133 mA	354 mA
16V	103 mA	288 mA
17V	81 mA	240 mA
17,5V	75 mA	213 mA
Note: The two largest current values for IIB are derived from 5.32W.		

Additional requirements of "ic" FISCO power supplies

The maximum output current I_o for an "ic" FISCO power supply shall be determined in accordance with IEC60079-11.

Table 2 - Assessment of maximum output current for use with "ic" FISCO power supplies

U_o	Permissible current , for IIC (includes 1.5 safety factor)	Permissible current , for IIB (includes 1.5 safety factor)
14V	274 mA	570 mA
15V	199 mA	531 mA
16V	154 mA	432 mA
17V	121 mA	360 mA
17,5V	112 mA	319 mA

GENERAL NOTES

- The intrinsically safe conductors must be of blue colored, based in the IEC standards.
- If one component of the intrinsically safe system is not FISCO comply, it is necessary to match all safety parameters among cable, device and barrier.
- Designated for connection to a Fieldbus system in compliance FISCO Model with parameters as follows:
 - Intrinsically safe apparatus interconnected to the power supply circuit (Fieldbus) shall be passive current sink (not supplying) and effective internal inductance/capacitance shall be within the following maximum values:

$$L_i \leq 10 \mu\text{H}$$

$$C_i \leq 5 \text{ nF}$$
 - With regard to cable-length parameters of Fieldbus interconnection-cable shall be within the following ranges:

$$\begin{aligned} \text{Resistance per unit length} \quad 15 \Omega/\text{km} \leq R' \leq 150 \Omega/\text{km} \\ \text{Inductance per unit length} \quad 0.4 \text{ mH}/\text{km} \leq L' \leq 1 \text{ mH}/\text{km} \\ \text{Capacitance per unit length (including shield)} \quad 80 \text{ nF}/\text{km} \leq C' \leq 200 \text{ nF}/\text{km} \end{aligned}$$

Where:

$$C' = C'_{\text{wire/wire}} + 0.5 \times C'_{\text{wire/shield}} \text{ when Fieldbus-circuit insulated.}$$

$$C' = C'_{\text{wire/wire}} + C'_{\text{wire/shield}} \text{ when shield is connected to the output of the Fieldbus power supply.}$$

Maximum length of each spur cable: 60m in IIC/IIB.
 - A Fieldbus-data-signal terminator, providing a capacitance less than or equal to $1.1 \mu\text{F}$ connected in series with a resistor greater than or equal to 100Ω , is integrated in the Barrier DF47-12 and DF47-17; similar terminator may be connected to the other end of the Fieldbus circuit.
 - When meeting the parameter mentioned above, maximum permissible Fieldbus-cable length including length of all spur cables for Group IIC is 1000 m.
 - When meeting the parameter mentioned above, maximum permissible Fieldbus-cable length including length of all spur cables for Group IIB and Group I is 5000 m.

* C_i : Input's capacitance, L_i : Input's inductance, C_o : Output's capacitance, L_o : Output's inductance

Hazardous locations approvals

FM Approvals (Factory Mutual)

DF47-12 FISCO Power Supply
Associated Intrinsic Safety (FM 3017363)

AIS Class I, Division 1, Groups A, B, C and D

AIS Class II, Division 1, Groups E, F and G

AIS Class III, Division 1

AIS Class I, Zone 0 [AEx ia], Group IIC

Special conditions for safe use:

Entity FISCO Trapezoidal Characteristic:

Terminals 1 and 2 Groups A/B IIC:

 $V_{oc} (U_o) = 15.0 \text{ V}$, $I_{sc} (I_o) = 140 \text{ mA}$, $I_{knee} (I_s) = 82 \text{ mA}$, $P_o = 1.2 \text{ W}$, $C_a (C_o) = 0.23 \text{ }\mu\text{F}$, $L_a (L_o) = 0.15 \text{ mH}$

Terminals 1 and 2 Groups C IIB

 $V_{oc} (U_o) = 15.0 \text{ V}$, $I_{sc} (I_o) = 140 \text{ mA}$, $I_{knee} (I_s) = 82 \text{ mA}$, $P_o = 1.2 \text{ W}$, $C_a (C_o) = 0.75 \text{ }\mu\text{F}$, $L_a (L_o) = 0.5 \text{ mH}$

 Integral Terminator: $R = 100 \text{ }\Omega$, $C = 1.0 \text{ }\mu\text{F}$, $C_i = 0$, $L_i = 0$

Note: The Fieldbus Isolated Barrier shall be installed in compliance with the enclosure, mounting, spacing and segregation requirements of the ultimate application, including a tool removable cover.

 Ambient Temperature: $-20^\circ\text{C} \leq T_a \leq +60^\circ\text{C}$
DF47-17 FISCO Power Supply
Associated Intrinsic Safety (FM 3017363)

AIS Class I, Division 1, Groups A, B, C and D

AIS Class II, Division 1, Groups E, F and G

AIS Class III, Division 1

AIS Class I, Zone 0 [AEx ia], Group IIC

Special conditions for safe use:

Entity FISCO Trapezoidal Characteristic:

Terminals 1 and 2 Groups A/B IIC:

 $V_{oc} (U_o) = 15.0 \text{ V}$, $I_{sc} (I_o) = 197 \text{ mA}$, $I_{knee} (I_s) = 115 \text{ mA}$, $P_o = 1.72 \text{ W}$, $C_a (C_o) = 0.21 \text{ }\mu\text{F}$, $L_a (L_o) = 0.15 \text{ mH}$

Terminals 1 and 2 Groups C IIB

 $V_{oc} (U_o) = 15.0 \text{ V}$, $I_{sc} (I_o) = 197 \text{ mA}$, $I_{knee} (I_s) = 115 \text{ mA}$, $P_o = 1.72 \text{ W}$, $C_a (C_o) = 0.7 \text{ }\mu\text{F}$, $L_a (L_o) = 0.5 \text{ mH}$

 Integral Terminator: $R = 100 \text{ }\Omega$, $C = 1.0 \text{ }\mu\text{F}$, $C_i = 0$, $L_i = 0$

Note: The Fieldbus Isolated Barrier shall be installed in compliance with the enclosure, mounting, spacing and segregation requirements of the ultimate application, including a tool removable cover.

 Ambient Temperature: $-20^\circ\text{C} \leq T_a \leq +60^\circ\text{C}$

EXAM (BBG Prüf - und Zertifizier GmbH)

Non Intrinsically safe circuits Parameters:

 Power Supply $U_n = 24 \text{ Vdc}$, $U_m = 250 \text{ Vac}$, $P_n = 3 \text{ W}$

 Fieldbus signal circuits $U_m = 250 \text{ Vac}$
DF47-12 FISCO Power Supply
Associated Intrinsic Safety (BVS 03ATEX E 411X)

Group II, Category (1) G, [Ex ia, EPL Ga], Groups IIB/ IIC FISCO Power Supply

Group I, Category (M2) [Ex ia, EPL Mb], Group I

Intrinsically safe fieldbus supply and signal circuit (FISCO-Model):

Safety parameters:

 $U_o = 15.0 \text{ Vdc}$, $I_o = 140.12 \text{ mA}$, $I_s = 80 \text{ mA}$, $P_o = 1200 \text{ mW}$, $R_i \geq 247.5 \text{ }\Omega$,

Characteristics trapezoidal

Special conditions for safe use

The Fieldbus-Isolated Barrier type DF47 -** shall be installed outside the hazardous area.

Wiring in the terminal box must satisfy the conditions of clause 6.3.11 and clause 7.6.e of EN60079-11:2007

Terminals or connectors for the intrinsically safe fieldbus supply and signal circuit circuits shall be arranged according to clause 6.21 or 6.2.2 of EN 60079-11:2007 respectively.

Local installation rules to determine Lo and Co are replaced by apparatus- and cable-parameters in clause 15.3.2.

For Group I application interconnection of fieldbus-apparatus to an intrinsically safe electrical system shall be assessed in a System Certificate, if required in local installation rules.

Ambient Temperature: $-20^{\circ}\text{C} \leq T_a \leq +60^{\circ}\text{C}$

The Essential Health and Safety Requirements are assured by compliance with:

EN 60079-0:2009 General Requirements

EN 60079-11:2007 Intrinsic Safety "i"

EN 60079-26:2007 Equipment with equipment protection level (EPL) Ga

EN 60079-27:2008 Fieldbus intrinsically safe concept (FISCO)

DF47-17 FISCO Power Supply

Associated Intrinsic Safety (BVS 03ATEX E 411X)

Group II, Category (1) G, [Ex ia, EPL Ga], Groups IIB/ IIC FISCO Power Supply

Group I, Category (M2) [Ex ia, EPL Mb] Group I

Intrinsically safe fieldbus supply and signal circuit (FISCO-Model):

Safety parameters:

$U_o = 15.0 \text{ Vdc}$, $I_o = 197 \text{ mA}$, $I_s = 115 \text{ mA}$, $P_o = 1720 \text{ mW}$, $R_i \geq 176.22 \Omega$,

Characteristics trapezoidal

Special conditions for safe use

The Fieldbus-Isolated Barrier type DF47 -** shall be installed outside the hazardous area.

Wiring in the terminal box must satisfy the conditions of clause 6.3.11 and clause 7.6.e of EN60079-11:2007

Terminals or connectors for the intrinsically safe fieldbus supply and signal circuit circuits shall be arranged according to clause 6.21 or 6.2.2 of EN 60079-11:2007 respectively.

Local installation rules to determine Lo and Co are replaced by apparatus- and cable-parameters in clause 15.3.2.

For Group I application interconnection of fieldbus-apparatus to an intrinsically safe electrical system shall be assessed in a System Certificate, if required in local installation rules.

Ambient Temperature: $-20^{\circ}\text{C} \leq T_a \leq +60^{\circ}\text{C}$

The Essential Health and Safety Requirements are assured by compliance with:

EN 60079-0:2009 General Requirements

EN 60079-11:2007 Intrinsic Safety "i"

EN 60079-26:2007 Equipment with equipment protection level (EPL) Ga

EN 60079-27:2008 Fieldbus intrinsically safe concept (FISCO)

CEPEL (Centro de Pesquisa de Energia Elétrica)

Non Intrinsically safe circuits parameters:

Power Supply $U_n = 24 \text{ Vdc}$, $P_n = 3 \text{ W}$

DF47-12 FISCO Power Supply

Associated Intrinsic Safety (CEPEL 06.1095 X)

[Ex ia, EPL Ga], Group IIB

Nominal values of the terminals Intrinsically safe (FISCO-Model):

$U_n = 14.0 \text{ V}$, $I_n = 75 \text{ mA}$, $P_n = 1200 \text{ mW}$

Safety parameters:

$U_m = 250 \text{ V}$, $U_o = 15 \text{ V}$, $I_o = 140.12 \text{ mA}$, $I_s = 80 \text{ mA}$, $P_o = 1200 \text{ mW}$, $R_i \geq 247.5 \Omega$

Ambient Temperature: -20 to 60°C

Special conditions for safe use

The certificate number with "X" indicates that:

- The equipment was projected to connect with a fieldbus system according to FISCO model, as IEC60079-27:2008. The Fieldbus intrinsic safety device connected to the fieldbus terminator must be current passive consumer (not generator) and must display $C_i \leq 5 \text{ nF}$ and $L_i \leq 10 \mu\text{H}$;
- Wiring in the terminal box must satisfy the conditions:

Resistance: $15 \Omega/\text{km} \leq R_c \leq 1500/\text{km}$

Capacitance (including loop): $45 \text{ nF}/\text{km} \leq C_c \leq 200 \text{ nF}/\text{km}$

Inductance: $0,4 \text{ mH}/\text{km} \leq L_c \leq 1\text{mH}/\text{km}$

- c) The maximum length allowed for Fieldbus cables is 1000 m for Group IIC and 5000m for Group IIB

The Essential Health and Safety Requirements are assured by compliance with:

ABNT NBR IEC 60079-0:2008 General Requirements

ABNT NBR IEC 60079-11:2009 Intrinsic Safety "i"

ABNT NBR IEC 60079-26:2008 Equipment with equipment protection level (EPL) Ga

IEC 60079-27:2008 Fieldbus intrinsically safe concept (FISCO)

DF47-17 FISCO Power Supply

Associated Intrinsic Safety (CEPEL 06.1095 X)

[Ex ia, EPL Ga], Group IIB

Nominal values of the terminals Intrinsically safe (FISCO-Model):

$U_n = 14.0 \text{ V}$, $I_n = 110 \text{ mA}$, $P_n = 1700 \text{ mW}$

Safety parameters:

$U_m = 250 \text{ V}$, $U_o = 15 \text{ V}$, $I_o = 197 \text{ mA}$, $I_s = 115 \text{ mA}$, $P_o = 1720 \text{ mW}$, $R_i \geq 176,22 \Omega$

Ambient Temperature: -20 to 60 °C

Special conditions for safe use

The certificate number with "X" indicates that:

- a) The equipment was projected to connect with a Fieldbus System according to FISCO model, as IEC60079-27:2008. The Fieldbus intrinsic safety device connected to the Fieldbus terminator must be current passive consumer (not generator) and must display $C_i \leq 5 \text{ nF}$ and $L_i \leq 10 \mu\text{H}$;
- b) Wiring in the terminal box must satisfy the conditions:
 - Resistance: $15 \Omega/\text{km} \leq R_c \leq 1500/\text{km}$
 - Capacitance (including loop): $45 \text{ nF}/\text{km} \leq C_c \leq 200 \text{ nF}/\text{km}$
 - Inductance: $0,4 \text{ mH}/\text{km} \leq L_c \leq 1\text{mH}/\text{km}$
- c) The maximum length allowed for Fieldbus cables is 1000 m for Group IIC and 5000m for Group IIB

The Essential Health and Safety Requirements are assured by compliance with:

ABNT NBR IEC 60079-0:2008 General Requirements

ABNT NBR IEC 60079-11:2009 Intrinsic Safety "i"

ABNT NBR IEC 60079-26:2008 Equipment with equipment protection level (EPL) Ga

IEC 60079-27:2008 Fieldbus intrinsically safe concept (FISCO)

Identification labels and control drawings

DF47-12 – Intrinsic Safety Barrier for Fieldbus

Labels

DF47-12 INTRINSIC SAFETY BARRIER FOR FIELDBUS

Safety Parameters:

FM

APPROVED

Intrinsically Safe Connections for,
CL I, DIV1, GP ABCDEFG and CL I, ZONE 0, GP IIC [AEx ia] IIC
"See Instalation drawing 102A0948 for FM FISCO parameters"

Ex

BVS 03 ATEX E 411 X
II (1)G [Ex ia Ga] IIB / IIC FISCO Power Supply
I (M2) [Ex ia Mb] I

CE

0470

Non Intrinsically Safe Fieldbus signal circuits.

Voltage Um AC 250 V - Max. Tamb. 60 °C

Intrinsically Safe Fieldbus supply - and signal circuit (FISCO).

Voltage (Uo) Voc DC 15.0 V

Short circuit current (Io) Isc 140 mA

Supply current at 15V (Is) Iknee 82 mA

Power Po 1.2 W

Current limiting resistor Ri ≥ 247.5 ohm

Characteristics trapezoidal

smar

DF47-12 Barreira de Segurança Intrinseca Fieldbus

FISCO Power Supply

[Ex ia Ga] IIB CEPEL 06.1095 X

Um = 250Vca Tamb: -20°C a 60°C

Ex

Valores Nominais [Ex ia Ga] IIB CEPEL 06.1095 X

Un = 14Vcc

Un = 75mA

PN = 1200mW

Uo = 15V

Io = 140,12mA

Po = 1200mW

Is = 80mA

Ri ≥ 247,5 Ω

Segurança

CEPEL

INMETRO

Tamb: -20°C a 60°C

Circuito não Intrinsecamente Seguro

Um = 250Vca Un = 24Vcc PN = 3W

CE

smar

13.32

NON-HAZARDOUS LOCATION				HAZARDOUS (CLASSIFIED) LOCATION CLASS I, II, III, GROUPS A, B, C, D, E, F, G.													
				REQUIREMENTS: 1- THE MAXIMUM NON-HAZARDOUS LOCATION VOLTAGE IS 250 VAC/DC. INSTALLATION MUST BE IN ACCORDANCE WITH THE NATIONAL ELECTRICAL CODE (NFPA 70, ARTICLE 504 OR 505) AND ANSI/ISA-RP12.06.01.													
<div><div>HAZARDOUS AREA</div><div>1 +</div><div>2 -</div><div>3</div><div>4</div></div>				<div>SIGNAL/POWER</div>				<div><div>Intrinsically safe fieldbus supply -and signal circuit (FISCO).</div><div><div>Voltage</div><div>Short circuit current</div><div>Supply current at 15V</div><div>Power</div><div>Current limiting resistor</div><div>Characteristics</div></div><div><div>(Uo) Voc</div><div>(Io) Isc</div><div>(Is) Iknee</div><div>Po</div><div>Ri ≥</div><div>trapezoidal</div></div></div>									
				<div><div>INTRINSIC SAFETY BARRIER</div><div><div>SAFE AREA</div><div>5 H1</div><div>6 H1</div><div>7 +</div><div>8 -</div></div></div>				<div><div>SIGNAL</div><div>POWER SUPPLY</div></div>									
APPROVAL CONTROLLED BY C.A.R.				DRAWN		CHECKED		PROJECT		APPROVAL							
2		MARCIAL 17/ 09/ 07	DÉLCIO 17/ 09/ 07	ALT-DE 0049/07	MOACIR 03/ 02/ 03		SINASTRE 03/ 02/ 03		BASÍLIO 03/ 02/ 03		DÉLCIO 03/ 02/ 03						
1		MOACIR 29/ 11/ 04	DÉLCIO 29/ 11/ 04	ALT-DE 0119/04	EQUIPMENT: DF47-12							NUMBER 102A0948		REV 02			
REV		BY		APPROVAL	DOC		CONTROL DRAWING							SCALE		SHEET 01/01	
<div><div>FM</div><div>APPROVED</div><div>smar</div></div>																	

DF47-17 – Intrinsic Safety Barrier for Fieldbus

Labels

DF47-17 INTRINSIC SAFETY BARRIER FOR FIELDBUS

Safety Parameters:

FM

APPROVED

Intrinsically Safe Connections for,
CL I, DIV1, GP ABCDEFG and CL I, ZONE 0, GP IIC [AEx ia] IIC
"See Instalation drawing 102A0949 for FM FISCO parameters"

Ex

BVS 03 ATEX E 411 X
II (1)G [Ex ia Ga] IIB / IIC FISCO Power Supply
I (M2) [Ex ia Mb] I

CE

0470

Non Intrinsically Safe Fieldbus signal circuits.

Voltage Um AC 250 V - Max. Tamb. 60 °C

Intrinsically Safe Fieldbus supply - and signal circuit (FISCO).

Voltage

(Uo) Voc DC 15.0 V

Short circuit current

(Io) Isc 197 mA

Supply current at 15V

(Is) Iknee 115 mA

Power

Po 1.72 W

Current limiting resistor

Ri ≥ 176.22 ohm

Characteristics

trapezoidal

smar

DF47-17 Barreira de Segurança Intrinseca Fieldbus

FISCO Power Supply

[Ex ia Ga] IIB CEPEL 06.1095 X

Um = 250Vca

Tamb: -20°C a 60°C

Ex

Valores Nominais

[Ex ia Ga] IIB CEPEL 06.1095 X

Un = 14Vcc

Un = 15V

IN = 110mA

Io = 197mA

PN = 1700mW

Pos = 1720mW

Is = 115mA

Ri ≥ 176,22 Ω

Segurança

INMETRO

CEPEL

Tamb: -20°C a 60°C

Circuito não Intrinsecamente Seguro

Um = 250Vca

UN = 24Vcc

PN = 3W

CE

smar

13.34

HAZARDOUS (CLASSIFIED) LOCATION
CLASS I, II, III, GROUPS A, B, C, D, E, F, G.

NON-HAZARDOUS LOCATION

REQUIREMENTS:
1- THE MAXIMUM NON-HAZARDOUS LOCATION VOLTAGE IS 250 V AC/DC.

INSTALLATION MUST BE IN ACCORDANCE WITH THE NATIONAL ELECTRICAL CODE (NFPA 70, ARTICLE 504 OR 505) AND ANSI/ISA-RP12.06.01.

1

+

2

-

3

4

HAZARDOUS AREA

SIGNAL/POWER

5

H1

6

H1

7

+

8

-

SAFE AREA

INTRINSIC SAFETY BARRIER

INTRINSICALLY SAFE FIELDBUS SUPPLY
-and signal circuit (FISCO):

Voltage

Short circuit current

Supply current at 15V

Power

Current limiting resistor

Characteristics

(Uo)Voc

(Io)sc

(Is)Iknee

Po

Ri

trapezoidal

15.0V DC

197mA

115mA

1.72W

≥ 176.22ohm

INTRINSICALLY SAFE OUTPUT FOR CLASS I, DIV. 1, GP A,B,C,D,E,F,G AND CLASS I, ZONE 0, [AEX ia II C];
CABLE PARAMETERS R'=15 TO 150ohms per Km, L'=0.4 to 1mH per km, C'= 80 to 200nF per km, INCLUDING SCREEN IF APPLICABLE
C'= C' CONDUCTOR / CONDUCTOR + 0.5 C' CONDUCTOR / SCREEN, IF BUS CIRCUIT IS POTENTIAL FREE (BALANCED).
C'= C' CONDUCTOR / CONDUCTOR + C' CONDUCTOR / SCREEN IF SCREEN IS CONNECTED WITH ONE TERMINAL OF THE BARRIER.

BARRIER INCLUDES TERMINATOR R=100ohms, C=1uF, Ci=0, Li=0

GP A,B/II C Ca (Co) = 0.21uF, La (Lo) = 0.15mH
GP C,D/II B,IIA Ca (Co) = 0.7uF, La (Lo) = 0.5mH

APPROVAL CONTROLLED BY C.A.R.

2

MARCIAL

17/ 09 / 07

DÉLCIO

17/ 09 / 07

ALT-DE

0049/07

1

MOACIR

29/ 11 / 04

DÉLCIO

29/ 11 / 04

ALT-DE

0119/04

REV

BY

APPROVAL

DÓC

DRAWN

MOACIR

03 / 02 / 03

CHECKED

SINASTRE

03 / 02 / 03

PROJECT

BASÍLIO

03 / 02 / 03

APPROVAL

DÉLCIO

03 / 02 / 03

EQUIPMENT:

DF47-17

CONTROL DRAWING

FM

APPROVED

smar

NUMBER

102A0949

SCALE

SHEET

01/01

REV

02

13.35

ADDING INTERFACES

Introduction

There are several interface modules available for DFI302 that provide a wide connectivity to different media used in the Automation and Process Control Industry.

In applications that connect Modbus RTU to DFI302, and require more than one Modbus device in the same Modbus network, it is necessary the use of RS-232/RS-485 module interface to provide a multipoint communication.

In the cases where only one Modbus device is used, and the distance between both devices are longer than 15 meters, the RS-232/RS-485 module interface is also necessary.

Originally, the controllers were designed to be connected to the Ethernet 10 Mbps port. In order to connect it in an Ethernet 100 Mbps Local Area Network, add the Ethernet Switch 10/100 Mbps module.

The following table shows the available Interface module types.

INTERFACES		
MODEL	DESCRIPTION	I/O TYPE
DF58	RS-232/RS-485 Interface	No I/O
DF61	Ethernet Switch 10/100 Mbps	No I/O

The specifications for each module are shown is the next pages.

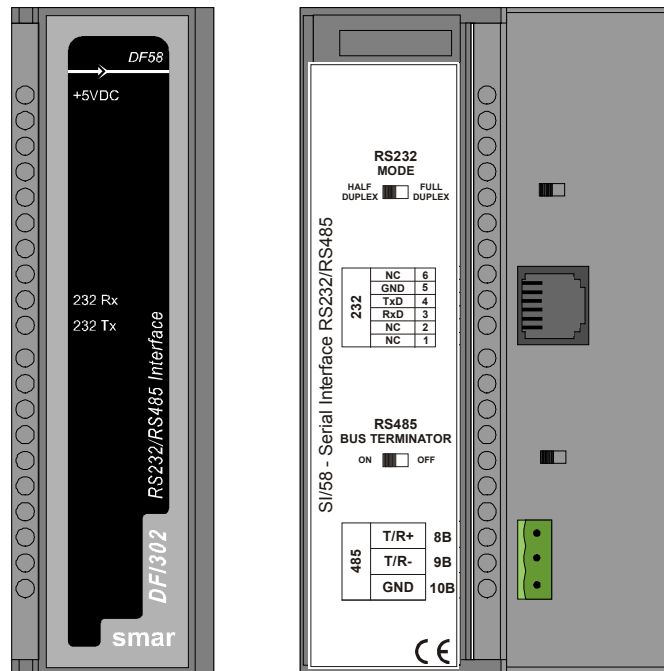
DF58 – RS-232/RS-485 Interface

Description

This module converts the electrical characteristics of the communication signal from the RS-232 to RS-485 specifications. Due to the different purposes of use between RS-232 and RS-485, where the first one is proper to peer-to-peer applications, this module was implemented to work automatically.

No control signal is necessary to control the RS-485 bus. Simply connect the transmission and reception lines on the both sides to have the interface operational.

The converter circuit isolates the signal to guarantee a safe connection between both systems. This module was designed to be use with DFI302/LC700 platforms and, no power supply was embedded in the board. It uses the +5 Vdc voltage from the rack to energize the circuit.



Interface Settings

There are two interfaces settings located on the front panel to adapt the interface to the applications: RS-232 Mode and RS-485 Bus Terminator.

RS-232 Mode: Half-Duplex/Full-Duplex

The RS-232 Mode setting adapts the use of RS-232/RS-485 Interface to the communication driver RS-232. Usually, these interfaces connect unidirectional buses with bidirectional buses. The unidirectional bus can contain Full-Duplex features caused by echoes of the transmitted message.

If the driver does not support the reception/transmission simultaneously, because the reception disables or discards the reflected message, select the **Half-Duplex** option. If the reflected message does not disturb the applications, select the **Full-Duplex** option.

RS-485 Bus Terminator: On/Off

The RS-485 is a multidrop bus. The transmitter driver is set to high impedance (Hi-Z) state when there is no message to be transmitted. However, the RS-485 bus requires a bus terminator in order to avoid noise problems during the idle time of the RS-485. A perfect line impedance match is necessary to activate only one terminator by bus. The other terminators must be deactivated.

Connectors

Two connectors are available on the front panel to interconnect two communication systems: RJ-12 type connector, used in RS-232 systems; and a terminal block type connector, used in RS-485 systems.

RJ12 Pin Assignment

PIN NUMBER	DESCRIPTION
1	Connected to pin 6
2	Not used
3	RxD: RS-232 input signal - reception
4	TxD: RS-232 output signal - transmission
5	GND: RS-232 grounding signal
6	Connected to pin 1

NOTE

Pins 1 and 6 are interconnected to allow the interconnection of the modem signals requested by communication drivers, such as Clear-To-Send (CTS) with Request-To-Send (RTS).

Block Terminal Pin Assignment

PIN NUMBER	DESCRIPTION
1	+: RS-485 Non-inverting signal
2	-: RS-485 Inverting Signal
3	GND: Reference for RS-485 Communication Signal.

NOTE

The pin GND sets the voltage reference for all RS-485 nodes. The RS-485 side of the RS-232/RS-485 Interface is isolated and left on floating state. To avoid undesirable high common mode voltage, it is recommended to set all RS-485 nodes to the same reference voltage by connecting all pins GND together and grounding at the same point.

Technical Specifications

GENERAL FEATURES	
Number of Communication Channel	1
Data Communication Interface	RS-232 / RS-485
Data rate	Up to 200 Kbps
RS-232 side	Enables RS-232 Half-Duplex or Full-Duplex mode
RS-485 side	Enables Embedded Bus Terminator activation
Isolation	1600 Vrms @ 1 minute, typical
Power Supply	Provided by the IMB bus, +5 Vdc @ 60 mA Typical

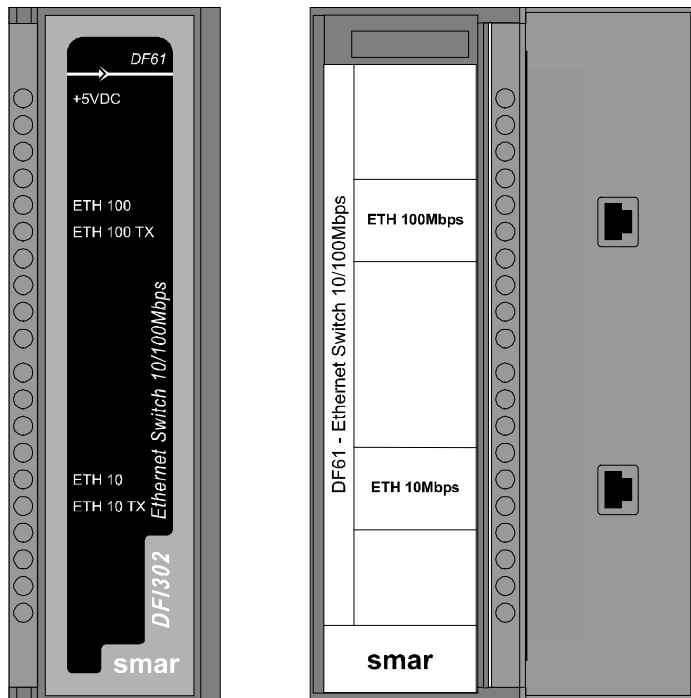
DF61 – Ethernet Switch 10/100 Mbps

This module connects the DF51 directly to an Ethernet 100 Mbps Local Area Network (LAN).

The only procedures to follow are:

- 1 - Fix the DF61 in a rack, and
- 2 - Using DF54 cable, connect DF61 to the 10 Mbps port of the DF51. Afterwards, the 100 Mbps port will be ready to be connected to the LAN.

Make sure that the Ethernet is perfectly communicating, check if the ETH10 and ETH100 LED indicators are ON (link connected), and ETH10TX and ETH100TX are blinking in the same rate of DF51.



ADDING MODBUS

Introduction

Several features of the DFI302 were designed to slowly migration of the existing plants to Fieldbus, protecting at the same time, as much as possible, the industry investments avoiding the requirements of the new devices. One of these features is the RS-232 Serial Communication Port, on the controller module that allows the connection to a wide range of legacy common devices in the existing plants using the Modbus Protocol.

The Modbus protocol has become one of the most popular factory standards. Existing projects have many devices and subsystems using this protocol.

The controller modules can act as master or slave device, and allows the communication through RS-232 or Ethernet ports, using in this last case, Modbus TCP/IP. The RS-232 Serial Port is integrated into the CPU module. When the RS-485 port is necessary, it is necessary to add an additional Interface Module (DF58).

NOTE DF89

The controller DF89 has different treatment and Modbus configurations according to the firmware version:

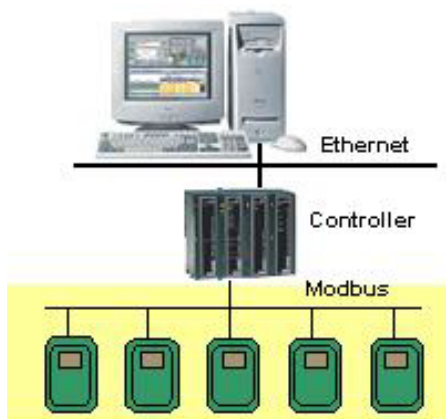
- From firmware version 4 and Device Description 4 the DF89 supports the master and slave Modbus features. The module configuration is different from the configuration of other DFI302 controllers. In this case refer to “Creating a Modbus configuration using the DF89” section.
- For firmware up to version 3 and Device Description 3 the DF89 only has the characteristic of Modbus slave. In this case, it has the same Modbus features of other controllers described in this section.

NOTE DF63-MBS

- The DF63-MBS controller is a DF63 with specific firmware (**DF63-MBS**), version V4.4.13 or higher.
- Firmware version 4.x: Device Revision = 5
- The MBCF (Modbus Configuration) block must be configured and there is no need for any other Modbus block to be instantiated. All internal points to the FFB (Flexible Function Block) will be mapped in Modbus registers automatically.

In master mode, the DFI302 can read and write data to the Modbus slaves.

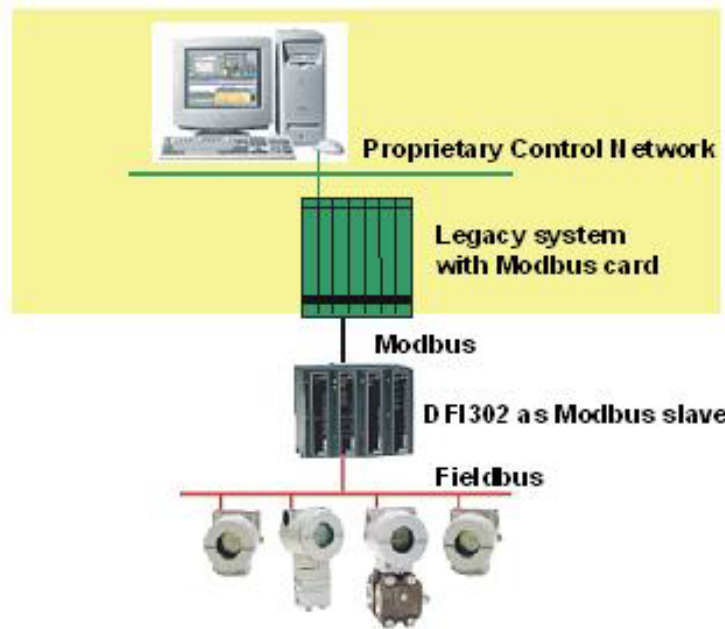
The DFI302 uses the standard Modbus commands for read and write data, making the slave data available for use in the control strategy or, for displaying and trending in the operator workstation. Operator can also operate the slave devices and execute other supervisory functions. The inputs are available as normal parameters that can be used in the control strategy or simply for monitoring, alarm and trend.



Most of the systems, such as DCS or PLC, have serial interface modules that support Modbus. These modules can be used to supervise Fieldbus instruments through the DFI302 controller as a gateway.

For the system connections, some extra interfaces can be necessary:

- For applications which the controller is connected through Modbus RTU protocol to the Modbus network with many Modbus devices, it is necessary the RS-232/RS-485 converter interface in order to provide the multipeer communication functionality;
- For the peer-to-peer applications where the Modbus device is used, but the distance with the controller is larger than 15 meters, it is also required a RS-232/RS-485 converter interface.



The DFI302 data can be read or written by the master, when the DFI302 operates in a slave mode. The data can be accessed in a binary format (functions 1, 2, 5 and 15) or through registers (functions 3, 4, 6 and 16).

Through DFI302, Fieldbus devices can be connected to an existing legacy control system allowing the existing system accesses some of the capabilities provided by Fieldbus devices. Traditional process variables and controller gains can be mapped from the Fieldbus to the database of existing system, but this system does not have the control level networking bandwidth and software capability, i.e., the fully benefits from the Fieldbus technology. However, this solution can be applied during the transition period to an open system.

It is easy to use the Modbus port. No DIP switches are required. The status LED on the front panel indicates active communication.

Steps to Configure Modbus

DFI302 uses **Syscon** to create all the required functionality, including Modbus. Refer to the chapter “**Adding Function Blocks**” for further information on adding function blocks to the **Syscon** configuration. Remember that Modbus function blocks are available in different DD revisions.

- 1) In order to include Modbus functionality inside DFI302, create first a MCBF block (Modbus Configuration Block).

NOTE

Remember that the Resource block must be created previously and set to **AUTO**.

The 'Block Attributes' dialog box shows the configuration for a Modbus Configuration block. The fields are as follows:

Manufacturer :	Smar		
Device Type :	DF62		
Device Rev. :	02	DD Rev. :	02
		CF Rev. :	01
Block Type :	Modbus Configuration		
Profile :	Custom		01
Block Tag :	MyMBCF		

Buttons: OK, Cancel, Help

- 2) Set parameters according to media, baud rate, addresses, etc.

The 'Off Line: Bridge 12 - Modbus Configuration - MCBF' window displays a list of parameters for configuration. The parameters are organized into a table with columns for Parameter, Value, Offset, and Han... (Handling).

Parameter	Value	Offset	Han...
ST_REV		1	RO
TAG_DESC		2	RW
STRATEGY		3	RW
ALERT_KEY		4	RW
MODE_BLK		5	
BLOCK_ERR		6	RO
MEDIA		7	RW
MASTER_SLAVE		8	RW
DEVICE_ADDRESS	Serial	9	RW
BAUD_RATE	TCP/IP	10	RW
STOP_BITS		11	RW
PARITY		12	RW
TIMEOUT		13	RW
NUMBER_RETRANSMISSIONS		14	RW
SLAVE_ADDRESSES		15	
RESTART_MODBUS		16	RW
TIME_TO_RESTART		17	RW
RTS_CTS		18	RW
ON_APPLY		19	RW
CHECK_COMM_STANDBY		20	RW

Buttons: Cancel Edit, End Edit, Clear, Close, Help

MBCF - Parameters Description

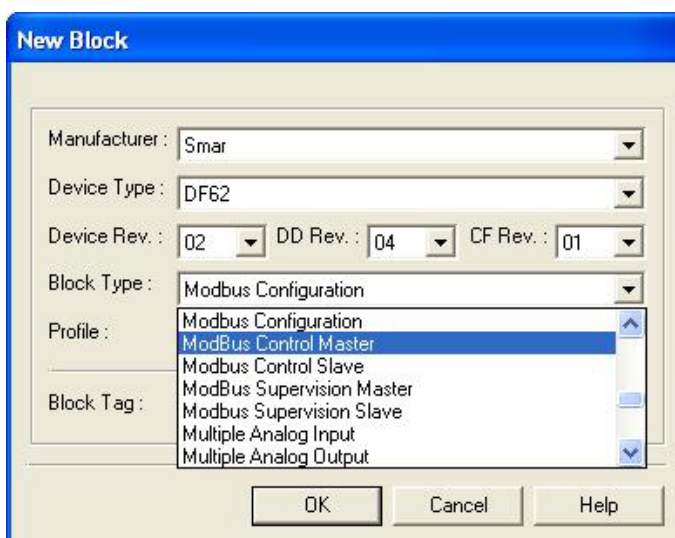
For further information about parameters description, please see Function Blocks Manual.

Idx	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store/ Mode	Description
1	ST_REV	Unsigned16		0	None	S/RO	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	MEDIA	Unsigned8	0:Serial, 1:TCP/IP	Serial	E	S	Define the type of Modbus channel.
8	MASTER_SLAVE	Unsigned8	0:Master, 1:Slave	Slave	E	S	Define if DFI is master or slave.
9	DEVICE_ADDRESS	Unsigned8	1-247	1	E	S	Define the DFI Modbus address (only for DFI slave).
10	BAUD_RATE	Unsigned8	0:110, 1:300, 2:600, 3:1200, 4:2400, 5:4800, 6:9600, 7:19200, 8:38400, 9:57600, 10:115200	19200	E	S	Define the baud rate (only for media serial).
11	STOP_BITS	Unsigned8	0:1, 1:2	1	E	S	Define the number of stop bits (only for media serial).
12	PARITY	Unsigned8	0:None, 1:Even, 2:Odd.	Even	E	S	Define the parity (only for media serial).
13	TIMEOUT	Unsigned16	200-65535	2000	ms	S	Time value to wait a response from a slave (for DFI master), or time to wait the OUTs are updated (for DFI slave).
14	NUMBER_RETRANS MISSIONS	Unsigned8	0-255	1		S	Number of retransmission if DFI doesn't receive response from slave.
15	SLAVE_ADDRESSES	DS-263				S	IP number and Modbus addresses of slaves (only for DFI master in TCP/IP media);
16	RESTART_MODBUS	Boolean		FALSE		S	Not used.
17	TIME_TO_RESTART	Unsigned16	100-65535 (Master) 0-65535 (Slave)	500	ms	S	When the device is working as master, it is the scan cycle of Modbus requests. When the device is working as slave, it is the delay for responses in Modbus TCP, aimed at restricting too short cycle by the master.
18	RTS_CTS	Boolean		FALSE		S	Enable or not handshaking.
19	ON_APPLY	Unsigned8	0:None, 1: Apply	None	E	S	Apply the changes made in the Modbus blocks.
20	CHECK_COMM_STANDBY	Unsigned8	0 ~ 255	0	NA	S / RW	It configures for the Standby if it will test the Modbus communication with the slave devices. 0: Disable test. 1 – 255: Enable test defining the time interval between each test (s).

Legend: E – Enumerated parameter; Na – Dimensionless parameter; RO – Read only; D – dynamic; N – non-volatile; S – Static
Gray Background Line: Default Parameters of Syscon

When using the RS-232, install the DF58 module (RS-232/RS-485 Interface Module) if it is necessary to communicate with more than one Modbus instrument, that is, in a multipoint network. Refer to chapter “**Adding Interfaces**”.

- Now, create the other blocks. The Modbus available blocks are: MBSS (Modbus Supervision Slave), MBSM (Modbus Supervision Master), MBCS (Modbus Control Slave) and MBCM (Modbus Control Master).



When using these blocks, set the **MODE_BLK.TARGET** to **AUTO**.

IMPORTANT

After downloading the whole configuration to DFI302, all the Modbus blocks will keep the element **MODE_BLK.ACTUAL** in **Out of Service**.

This protection allows the user to create all the necessary blocks, and adjust all the parameters, even in Online Characterization, and change the **MODE_BLK** to **AUTO** simultaneously, configuring the MBCF block and setting **ON_APPLY** parameter to **APPLY**.

Other important parameter that should be defined for all blocks is **LOCAL_MOD_MAP** (0 - 15). Sixteen blocks are accepted per each functionality. For MBSS and MBCS blocks, the **LOCAL_MOD_MAP** parameter is used as offset for calculation of the Modbus slave addresses. The 255 value (default) disable the block.



COMMUNICATION TIPS

1. For a better synchronization of the Modbus blocks, execute **ON_APPLY** in the master controller after the slave are already configured and running.
2. If the communication was not established on the first **ON_APPLY**, just do another **ON_APPLY** in the controller.

RESTRICTIONS

1. The MBSS block is only available at the DF51 controller.
2. The way that the Modbus works at DF73, DF75, DF79, DF81, DF89*, DF95, DF97, and DF63-MBS** controllers is different from other controllers available in the DFI302 system. The MBCF must be configured and is not necessary to instantiate any Modbus function blocks. All the internal values existing in FFB (flexible function blocks) are automatically mapped in Modbus registers. See details in this section under item "Using Modbus in DF73, DF75, DF79, DF81, DF89*, DF95, DF97, and DF63-MBS** controllers".
3. The roles of DF73, DF75, DF79, DF81, DF89*, DF95 and DF97at Modbus are only Slave.

*See "NOTE DF89"

**See "NOTE DF63-MBS"

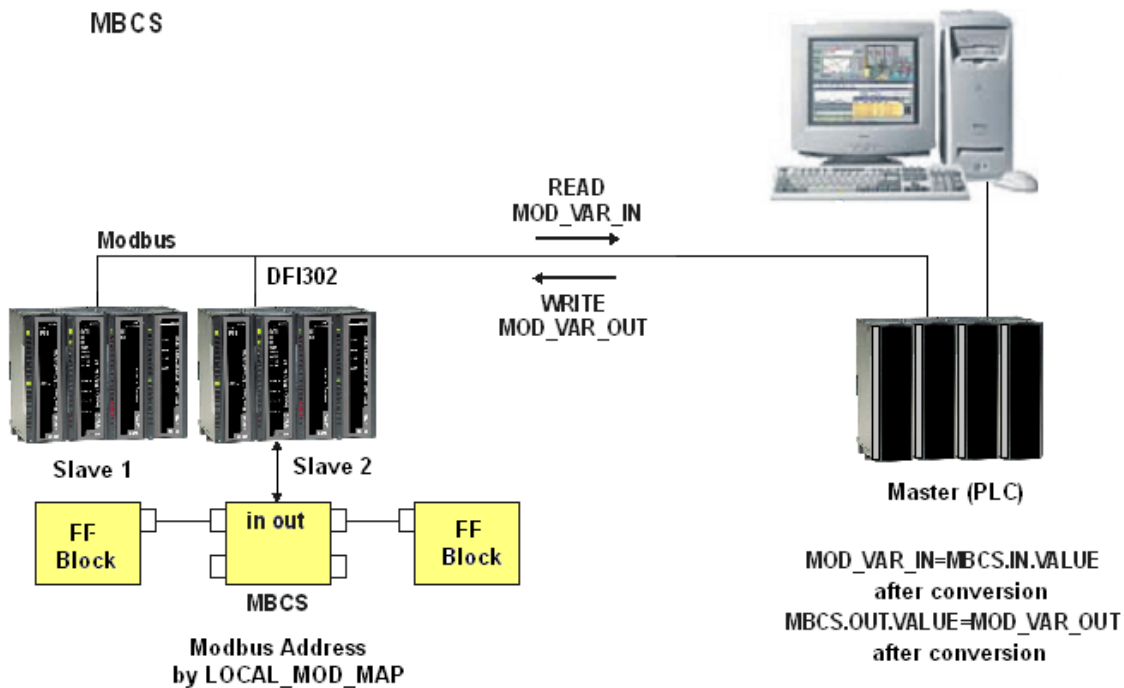
The following scenarios summarize some of the applications that use the DFI302 Modbus functionality.

MBCS – Modbus Control Slave

A master Modbus device needs to read and/or write some DFI302 Modbus registers, mapped directly to input and output variables at Fieldbus network.

Using **Syscon**, create one MBCF block and MBCS blocks (1 to 16 blocks are available). In the **Strategy** window, link these blocks with other FOUNDATION fieldbus blocks.

MBCS



When setting the parameters, define **LOCAL_MOD_MAP** (0 - 15) and the **IN** and **OUT** parameters will point to pre-defined Modbus addresses. See **Modbus Slave Addresses** section for further information.

Parameters Description

For further information about the parameters description, please refer to Function Blocks Manual.

Idx	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S/RO	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter.
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	LOCAL_MOD_MAP	Unsigned8	0 to 15	0		S / O/S	Define the Modbus addresses.
8	IN1	DS-65				N	Analog input 1
9	SCALE_CONV_IN1	DS-256				S / O/S	Information to generate constants A and B in equation $Y=A*X+B$.
10	IN2	DS-65				N	Analog input 2
11	SCALE_CONV_IN2	DS-256				S / O/S	Information to generate constants A and B in equation $Y=A*X+B$.
12	IN3	DS-65				N	Analog input 3
13	SCALE_CONV_IN3	DS-256				S / O/S	Information to generate constants A and B in equation $Y=A*X+B$.
14	IN4	DS-65				N	Analog input 4
15	SCALE_CONV_IN4	DS-256				S / O/S	Information to generate constants A and B in equation $Y=A*X+B$.
16	IN_D1	DS-66				N	Discrete input 1
17	IN_D2	DS-66				N	Discrete input 2
18	IN_D3	DS-66				N	Discrete input 3
19	IN_D4	DS-66				N	Discrete input 4
20	OUT1	DS-65				N / Man	Analog output 1
21	SCALE_CONV_OUT1	DS-257				S / O/S	Information to generate constants A and B in equation $Y=A*X+B$ plus output status.
22	OUT2	DS-65				N / Man	Analog output 2
23	SCALE_CONV_OUT2	DS-257				S / O/S	Information to generate constants A and B in equation $Y=A*X+B$ plus output status.
24	OUT3	DS-65				N / Man	Analog output 3
25	SCALE_CONV_OUT3	DS-257				S / O/S	Information to generate constants A and B in equation $Y=A*X+B$ plus output status.
26	OUT4	DS-65				N / Man	Analog output 4
27	SCALE_CONV_OUT4	DS-257				S / O/S	Information to generate constants A and B in equation $Y=A*X+B$ plus output status.
28	OUT_D1	DS-66				N / Man	Discrete output 1
29	STATUS_OUT_D1	Unsigned8				S / O/S	Status to OUT_D1 if master will not update.
30	OUT_D2	DS-66				N / Man	Discrete output 2
31	STATUS_OUT_D2	Unsigned8				S / O/S	Status to OUT_D2 if master will not update.
32	OUT_D3	DS-66				N / Man	Discrete output 3
33	STATUS_OUT_D3	Unsigned8				S / O/S	Status to OUT_D3 if master will not update.
34	OUT_D4	DS-66				N / Man	Discrete output 4
35	STATUS_OUT_D4	Unsigned8				S / O/S	Status to OUT_D4 if master will not update.
36	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
37	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, and connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: E – Enumerated parameter; Na – Dimensionless parameter; RO – Read only; D – dynamic; N – non-volatile; S – Static
Gray Background Line: Default Parameters of Syscon

NOTES

The **STATUS_OUT_Dx** and **STATUS_OUTPUT** components, used in **OUT** parameters, define the following rules to **OUTPUT STATUS** parameters:

- Once the user defines this component as “Set by master”, the output status will behave exactly as FOUNDATION fieldbus protocol works. In other words, the status will reflect the value, which the master is writing, but if after the **TIMEOUT** (defined in MBCF block) the status is not be updated, this status will be forced to **BAD COMMUNICATION**.
- Once the user defines this element with anything different from “Set by master”, this value will be reflected in output status, while communication is good. Otherwise, status goes to **BAD COMMUNICATION**.

Inputs and Outputs

This block has 4 digital inputs, 4 analog inputs, 4 digital outputs and 4 analog outputs that may be connected to others in Fieldbus or Modbus function blocks.

- IN1, IN2, IN3 and IN4 are analog inputs.
- IN_D1, IN_D2, IN_D3 and IN_D4 are digital inputs.
- OUT1, OUT2, OUT3 and OUT4 are analog outputs.
- OUT_D1, OUT_D2, OUT_D3 and OUT_D4 are digital outputs.

The digital inputs and outputs are **DS-66** data types, which have **Status** and **Value** (both Unsigned8) parameters. The analog inputs and outputs are **DS-65** data type, containing **Status** and **Value** parameters as well. The type of value parameters is float.

Scale Conversion Parameters

Each analog input or output has an extra parameter that should be set in order to use this block properly. This is done through the **SCALE_CONV_INn** and **SCALE_CONV_OUTn** parameters. These parameters are **DS-256** and **DS-257** represented by data structures:

The **DS-256** data structure has 5 elements to be configured:

- From EU 100 %
- From EU 0 %
- To EU 100 %
- To EU 0 %
- Data Type

The **DS-257** data structure has 6 elements to be configured:

- From EU 100 %
- From EU 0 %
- To EU 100 %
- To EU 0 %
- Data Type
- Output Status

Data Types

It is necessary to configure the data type, because Modbus variables have different formats.

This parameter only displays a number that refers to a specific format:

DATA TYPE NUMBER	DATA TYPE
1	Float
2	Unsigned 8
3	Unsigned 16
4	Unsigned 32
5	Integer8
6	Integer16
7	Integer32

DATA TYPE NUMBER	DATA TYPE
8	Swapped Float
9	Swapped Unsigned 8
10	Swapped Unsigned 16
11	Swapped Unsigned 32
12	Swapped Integer 8
13	Swapped Integer 16
14	Swapped Integer 32

The swapped data types were created in order to support the communication between Modbus devices and Profibus devices. Normally, it has the following options:

4 Bytes (2 Registers – Word)

Normal Data type: Inside Word – Motorola
Inside Register – Intel
Swapped Data type: Inside Word – Motorola
Inside Register – Motorola

2 Bytes

Swapped Data type: Status information is in the Most Significant Byte (MSB)

1 Byte

Swapped Data type: Value (MSB) and Status (LSB) are in the same register.

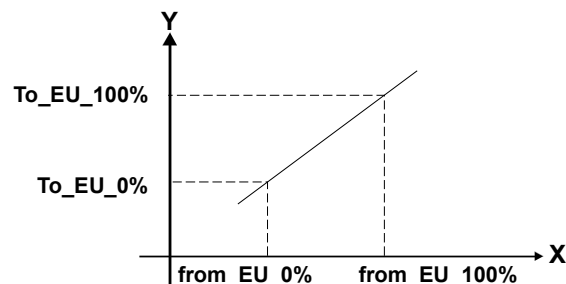
In case of the Swapped Integer 16 data type no change is done.

Procedures to convert FOUNDATION fieldbus parameter to Modbus variable:

Load **INn_VALUE**.
Calculate $Y = A * \text{INn_VALUE} + B$.
Convert Y to **DATA_TYPE_IN**, generating **MOD_VAR_IN**.
Store **MOD_VAR_IN**.

Procedures to convert Modbus variable to FOUNDATION fieldbus parameter:

Load **MOD_VAR_OUT**.
Convert **MOD_VAR_OUT** to float, generating Y
Calculate $\text{OUTn_VALUE} = (A * Y + B)$.
Store **OUTn_VALUE**.



$$A = (\text{To_EU_100\%} - \text{To_EU_0\%}) / (\text{From_EU_100\%} - \text{From_EU_0\%})$$

$$B = \text{To_EU_0\%} - A * \text{From_EU_0\%};$$

INn_VALUE, **OUTx_VALUE**: FOUNDATION fieldbus parameters

MOD_VAR_IN, **MOD_VAR_OUT**: Modbus variables

Y: auxiliary float variable.

Output Status

If Modbus master does not update the **OUTs** parameters in a time specified by user (parameter **TIMEOUT** in MBCF), a "BAD STATUS" will be generated. If **TIMEOUT** < Macrocycle, **TIMEOUT** = Macrocycle.

MBSS – Modbus Supervision Slave

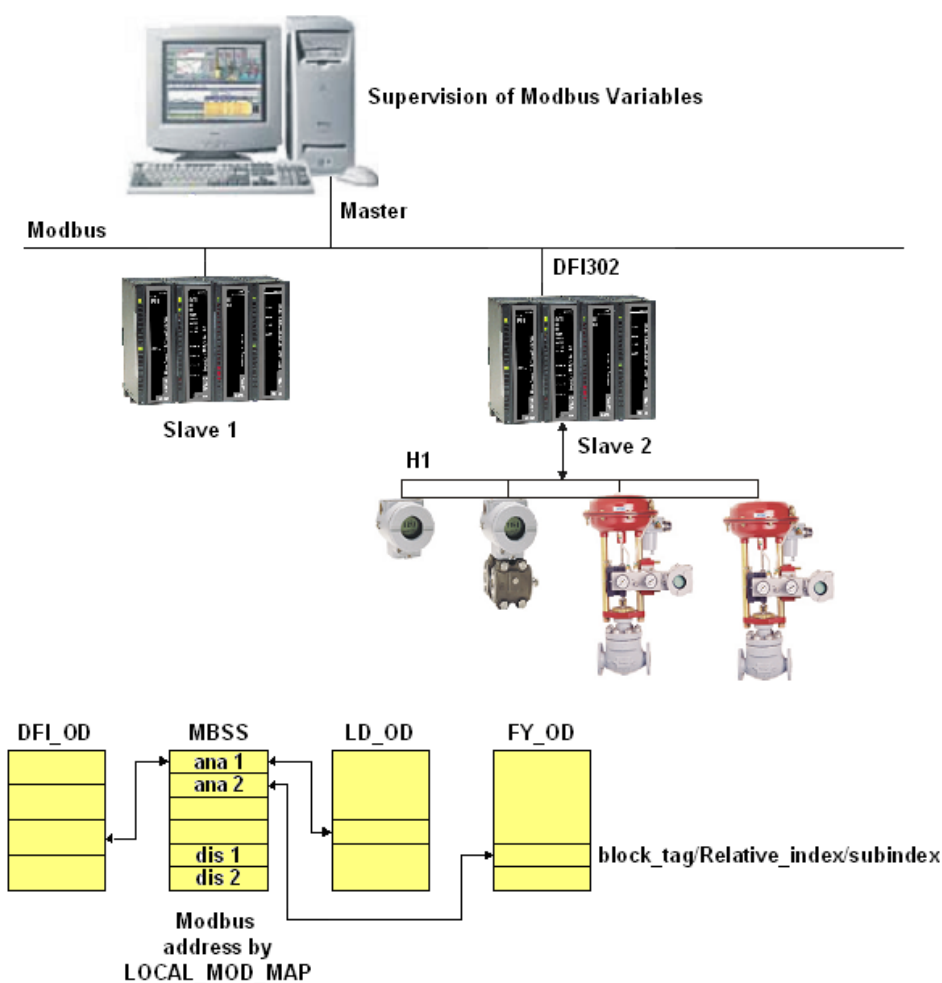
A master Modbus device needs to read and/or write some DFI302 Modbus registers, mapped directly to any variables at Fieldbus network. Using **Syscon**, create one MBCF and MBSS blocks (1 to 16 blocks are available). In the Characterization window, configure the blocks parameters setting Tag, Relative Index and Subindex of other parameters included in FOUNDATION fieldbus blocks.

ATTENTION

The MBSS block is available in the DF51 controller.

It is recommended the use of MBCS blocks instead of MBSS blocks. This provides a better performance to data obtainment and for transmission.

MBSS



When setting the parameters, define **LOCAL_MOD_MAP** (0 - 15) the **INs** and **OUTs** parameters will point to pre-defined Modbus addresses. Refer to **Modbus Slave Addresses** section for further information.

Parameters Description

For further information about parameters description, please refer to Function Blocks Manual.

Idx	Parameter	Data Type (length)	Valid Range/Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S/RO	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	LOCAL_MOD_MAP	Unsigned8	0 to 15	0		S / O/S	Define the ModBus addresses.
8	F_ID1	DS-262				S / O/S	Information to locate float parameter.
9	FVALUE1	Float		0		N	Value from requested float parameter.
10	F_ID2	DS-262				S / O/S	Information to locate float parameter.
11	FVALUE2	Float		0		N	Value from requested float parameter.
12	F_ID3	DS-262				S / O/S	Information to locate float parameter.
13	FVALUE3	Float		0		N	Value from requested float parameter.
14	F_ID4	DS-262				S / O/S	Information to locate float parameter.
15	FVALUE4	Float		0		N	Value from requested float parameter.
16	F_ID5	DS-262				S / O/S	Information to locate float parameter.
17	FVALUE5	Float		0		N	Value from requested float parameter.
18	F_ID6	DS-262				S / O/S	Information to locate float parameter.
19	FVALUE6	Float		0		N	Value from requested float parameter.
20	F_ID7	DS-262				S / O/S	Information to locate float parameter.
21	FVALUE7	Float		0		N	Value from requested float parameter.
22	F_ID8	DS-262				S / O/S	Information to locate float parameter.
23	FVALUE8	Float		0		N	Value from requested float parameter.
24	I_ID1	DS-262				S / O/S	Information to locate integer parameter.
25	IVALUE1	Integer32		0		N	Value from requested integer parameter.
26	I_ID2	DS-262				S / O/S	Information to locate integer parameter.
27	IVALUE2	Integer32		0		N	Value from requested integer parameter.
28	I_ID3	DS-262				S / O/S	Information to locate integer parameter.
29	IVALUE3	Integer32		0		N	Value from requested integer parameter.
30	I_ID4	DS-262				S / O/S	Information to locate integer parameter.
31	IVALUE4	Integer32		0		N	Value from requested integer parameter.
32	B_ID1	DS-262				S / O/S	Information to locate boolean parameter.
33	BVALUE1	Boolean		TRUE		N	Value from requested boolean parameter.
34	B_ID2	DS-262				S / O/S	Information to locate boolean parameter.
35	BVALUE2	Boolean		TRUE		N	Value from requested boolean parameter.
36	B_ID3	DS-262				S / O/S	Information to locate boolean parameter.
37	BVALUE3	Boolean		TRUE		N	Value from requested boolean parameter.
38	B_ID4	DS-262				S / O/S	Information to locate boolean parameter.
39	BVALUE4	Boolean		TRUE		N	Value from requested boolean parameter.
40	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
41	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, and connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.
42	BAD_STATUS	BitString (2)				D/RO	This parameter indicates if the status of correspondent variable is bad or not.

Legend: E – Enumerated parameter; Na – Dimensionless parameter; RO – Read only; D – dynamic; N – non-volatile; S – Static
Gray Background Line: Default Parameters of Syscon

NOTE

Every time a Modbus parameter changes, it is necessary to set the **ON_APPLY** parameter of the MBCF block to **APPLY**. Otherwise, these alterations will not be applied.

I_IDn, F_IDn, D_IDn parameters

The **I_IDn** are integer variables, **F_IDn** are float variables and **D_IDn** refers to boolean variables.

These parameters are **DS-262** data type. This data type has 3 elements:

- **Block Tag:** Indicates the Tag of the block that contains the variable that will be supervised. For example, if it is necessary to supervise the gain value of a PID block. Insert the Tag of the PID block that contains the gain parameter required to visualize the Modbus master device.
- **Relative Index:** Each parameter of a function block has this index. It is the first column of all function blocks parameters tables. Insert the relative index related to the desired parameter to be supervised. In the case above in order to monitor the gain parameter of the PID function block, the relative index is 23.
- **Subindex:** The Subindex is used for parameters that have a structure. In this case, it is necessary to indicate which element of the structure is being referred.

BVALUEx and IVALUEx parameters

The **BVALUEx** parameters can address FOUNDATION fieldbus block parameters of the following data types: Boolean, Integer8 and Unsigned8. Those data types are automatically converted to bit (0 or 1) and vice-versa for Modbus supervision and also converted to boolean parameter (**BVALUEx**).

The **IVALUEx** parameters can address FOUNDATION fieldbus block parameters of the following data types: Integer8, Integer16, Integer32, Unsigned8, Unsigned16 and Unsigned32.

Each analog parameter (**IVALUEx**) is mapped as two analog registers in Modbus, i.e., four bytes. Thus, when addressing a FOUNDATION fieldbus block parameter with one or two bytes, such parameter will be promoted to Unsigned32 or Integer32.

If Relative Index = 5 (**MODE_BLK**) e Subindex = 0, it is performed a writing in Subindex 1 and a reading in Subindex 2.

BAD_STATUS Parameter

This parameter indicates if the communication with master device is working properly. If the correspondent bit is in logic level 1 means that an error occurred during writing/reading in this respective parameter. The table below shows the values for this status parameter. If the communication with the specific parameter is good, there is no indication in **BAD_STATUS**. However, if the communication is bad, **BAD_STATUS** will indicate which parameter failed in the communication.

Relation between the bits in BAD_STATUS and Modbus addresses

BIT	VARIABLE
0	FVALUE1
1	FVALUE2
2	FVALUE3
3	FVALUE4
4	FVALUE5
5	FVALUE6
6	FVALUE7
7	FVALUE8
8	IVALUE1
9	IVALUE2
10	IVALUE3
11	IVALUE4
12	BVALUE1
13	BVALUE2
14	BVALUE3
15	BVALUE4

NOTE

Each bit corresponds to an OR function between the value and status, indicating if the communication with master is good or bad.

Data Type and Supported Structures by MBSS

The Modbus supervision blocks (MBSS) used in the controllers configured as slave, have some restrictions about data types and structures that they support when supervising the block parameters tags. In such case, the next table shows the data types and structures which can be monitored by the MBSS block.

DATA TYPE *
Boolean
Float
Unsigned 8
Unsigned 16
Unsigned 32
Integer8
Integer16
Integer32

STRUCTURE TYPES
DS-65
DS-66
DS-68
DS-69
DS-71
DS-72
DS-74
DS-159 (DC302)
DS-160 (DC302)

*For the DF51 controller the data types Swapped Float and Swapped Integer can be obtained by setting the **RTS_CTS** parameter to TRUE value in the MBSS block.

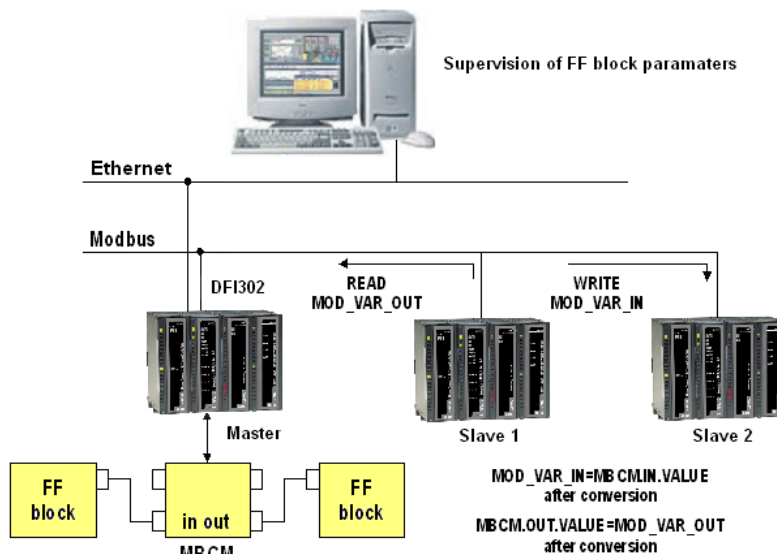
For further information about the blocks parameters and their data and structures types, as mentioned in the previous table, please see the Function blocks manual and in the **Data Type** field of each table presented in the manual.

MBCM – Modbus Control Master

A slave Modbus device needs to receive and/or overwrite some DFI302 Modbus registers, mapped directly to input and output variables at Fieldbus network.

Using **Syscon**, create one MBCM and MBCM blocks (1 to 16 blocks are available). In the **Strategy** window, link their parameters with other FOUNDATION fieldbus parameters. This application is very useful also with display view devices installed in factory floor.

MBCM



Set the **MASTER_SLAVE** parameter to master, in the MBCM block. Define **LOCAL_MOD_MAP** (0 - 15).

Parameters Description

For further information about parameters description, please refer to Function Blocks Manual.

Idx	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S/RO	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	LOCAL_MOD_MAP	Unsigned8	0 to 15	0		S / O/S	Define the Modbus addresses.
8	BAD_STATUS	Bitstring(2)		0	E	D / RO	Indicate if communication from slave is good or not (each bit corresponds to a Modbus variable).
9	IN1	DS-65				N	Analog input 1
10	SCALE_LOC_IN1	DS-259				S / M	Information to generate constants A and B in equation $Y=A*X+B$ plus the addresses in a slave device.
11	IN2	DS-65				N	Analog input 2
12	SCALE_LOC_IN2	DS-259				S / M	Information to generate constants A and B in equation $Y=A*X+B$ plus the addresses in a slave device.
13	IN3	DS-65				N	Analog input 3
14	SCALE_LOC_IN3	DS-259				S / M	Information to generate constants A and B in equation $Y=A*X+B$ plus the addresses in a slave device.
15	IN4	DS-65				N	Analog input 4
16	SCALE_LOC_IN4	DS-259				S / M	Information to generate constants A and B in equation $Y=A*X+B$ plus the addresses in a slave device.
17	IN_D1	DS-66				N	Discrete input 1
18	LOCATOR_IN_D1	DS-261				S / O/S	Addresses in a slave device.
19	IN_D2	DS-66				N	Discrete input 2
20	LOCATOR_IN_D2	DS-261				S / O/S	Addresses in a slave device.
21	IN_D3	DS-66				N	Discrete input 3
22	LOCATOR_IN_D3	DS-261				S / O/S	Addresses in a slave device.
23	IN_D4	DS-66				N	Discrete input 4
24	LOCATOR_IN_D4	DS-261				S / O/S	Addresses in a slave device.
25	OUT1	DS-65				N / Man	Analog output 1
26	SCALE_LOC_OUT1	DS-259				S / M	Information to generate constants A and B in equation $Y=A*X+B$ plus the addresses in a slave device.
27	OUT2	DS-65				N / Man	Analog output 2
28	SCALE_LOC_OUT2	DS-259				S / M	Information to generate constants A and B in equation $Y=A*X+B$ plus the addresses in a slave device.
29	OUT3	DS-65				N / Man	Analog output 3
30	SCALE_LOC_OUT3	DS-259				S / M	Information to generate constants A and B in equation $Y=A*X+B$ plus the addresses in a slave device.
31	OUT4	DS-65				N / Man	Analog output 4
32	SCALE_LOC_OUT4	DS-259				S / M	Information to generate constants A and B in equation $Y=A*X+B$ plus the addresses in a slave device.
33	OUT_D1	DS-66				N / Man	Discrete output 1
34	LOCATOR_OUT_D1	DS-261				S / O/S	Addresses in a slave device.
35	OUT2_D2	DS-66				N / Man	Discrete output 2
36	LOCATOR_OUT_D2	DS-261				S / O/S	Addresses in a slave device.
37	OUT_D3	DS-66				N / Man	Discrete output 3
38	LOCATOR_OUT_D3	DS-261				S / O/S	Addresses in a slave device.
39	OUT_D4	DS-66				N / Man	Discrete output 4
40	LOCATOR_OUT_D4	DS-261				S / O/S	Addresses in a slave device.

Idx	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
41	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
42	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, and connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: E – Enumerated parameter; Na – Dimensionless parameter; RO – Read only; D – Dynamic; N – Non-volatile; S – Static
Gray Background Line: Default Parameters of Syscon

NOTE

Every time a Modbus parameter is changed it is necessary to set the **ON_APPLY** parameter of the MBCF block to “**APPLY**”. Otherwise these alterations will not be effective.

NOTES

The **MODBUS_ADDRESS_OF_STATUS** components define the following rules to **OUTPUT STATUS** parameters:

- Once the user defines this component with a value different from ZERO, the output status will behave exactly as Fieldbus protocol works. The status will reflect the value which the master is reading but if after **TIMEOUT** (defined in MBCF block) the status is not updated, this status will be forced to **BAD COMMUNICATION**.
- Once the user defines this component with a value ZERO, the output status will be set automatically to **Good::NonCascade** and will also accept a characterization via **Syscon** (e.g. **Good::NonCascade**, etc). However, if after **TIMEOUT** (defined in MBCF block) the communication with the Modbus device is not running properly, the status will be forced to **BAD COMMUNICATION**.

LOCAL_MODE_MAP Parameter

All MBCM blocks added to the strategy must have different values for **LOCAL_MODE_MAP**. Otherwise the block will not work properly.

Inputs and Outputs

This block has 4 digital inputs and outputs and 4 analog inputs and outputs. These inputs and outputs may be connected to other FOUNDATION fieldbus function blocks to be connected to Modbus I/O modules or registers.

INn: Analog input. **DS-65** data type. **Value** and **Status**. In this parameter user will visualize the value of the parameter set for this input and its status.

IN_Dn: Digital input. **DS-66** data type. **Value** and **Status**. In this parameter user will visualize the value of the parameter set for this input and its status.

OUTn: Analog output. **DS-65** data type. **Value** and **Status**. In this parameter user will visualize the value of the parameter set for this output and its status.

OUT_Dn: Digital output. **DS-66** data type. **Value** and **Status**. In this parameter user will visualize the value of the parameter set for this output and its status.

SCALE_LOC_INn and SCALE_LOC_OUTn Parameters

These parameters are **DS-259** data type. They both convert the value to Engineering Units and address the variable in the Modbus network. The INn and OUTn inputs and outputs have **SCALE_LOC_INn** and **SCALE_LOC_OUTn** parameters associated. It is necessary to set these parameters, so the monitoring and data exchanges are properly made.

These parameters consist of the following elements:

- From Eu 100 %
- From Eu 0 %
- To Eu 100 %
- To Eu 0 %
- Data type

See in the following item how to set these elements.

Data Type: It is necessary to inform the data type of the variable. This parameter only displays a number that refers to a specific format:

DATA TYPE NUMBER	DATA TYPE
1	Float
2	Unsigned 8
3	Unsigned 16
4	Unsigned 32
5	Integer8
6	Integer16
7	Integer32
8	Swapped Float
9	Swapped Unsigned 8
10	Swapped Unsigned 16
11	Swapped Unsigned 32
12	Swapped Integer 8
13	Swapped Integer 16
14	Swapped Integer 32

The swapped data types were created in order to support the communication of Modbus devices with Profibus devices. Normally we have the following cases:

4 Bytes (2 Registers – Word)

Normal Data type: Inside Word – Motorola
Inside Register – Intel
Swapped Data type: Inside Word – Motorola
Inside Register – Motorola

2 Bytes

Swapped Data type: Status information is in the Most Significant Byte (MSB)

1 Byte

Swapped Data type: Value (MSB) and Status (LSB) are in the same register.

In case of the Swapped Integer 16 data type no change is done.

Slave Address: Inform the address of the slave required to reference to the IN input. For example, suppose a LC700 with device address equal to 3 and in this LC700 it is necessary to connect one of its inputs or outputs. Thus, slave address must be equal to 3.

Modbus Address of Value: Inform the Modbus address of the variable it will be referenced to the input or output. In the example of the previous element, suppose the Modbus address of the variable it will be referenced is 40032. So this element must receive this address.

Modbus Address of Status: In this parameter user informs the Modbus address where the status will be read or written. Each input and output has a correspondent status. The status interpretation follows the standard of the Fieldbus Foundation.

The treatment of inputs and outputs are described in the following table.

INPUT/OUTPUT	STATUS CONFIGURED (Modbus_Address_Of_Status ≠ 0)	STATUS NOT CONFIGURED (Modbus_Address_Of_Status = 0)
Inputs (IN_n , IN_Dn)	The block sends to the Modbus slave device the status corresponding of its input. (The status has the FOUNDATION fieldbus standard format)	No status information is sent to the slave device.
Outputs (OUT_n, OUT_Dn)	The block reads from the slave device the corresponding status. (The block interprets that the Modbus variable has the same format of the FOUNDATION fieldbus Status)	<ul style="list-style-type: none"> - The block updates the status to "Good Non Cascade" when the communication with the Modbus slave device is ok. - The block updates the status to "Bad No Communication with last value" when the communication with the Modbus slave device is not ok.

Float values use two Modbus registers, but it is necessary only to inform the first.

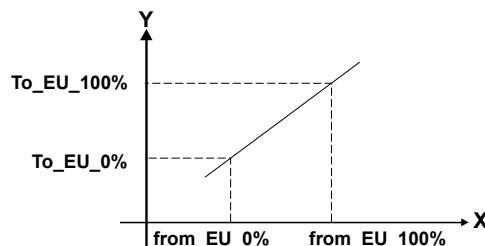
Procedure to convert FOUNDATION fieldbus parameter to Modbus variable:

Load **INx_VALUE**.
 Calculate $Y = (A * \text{INx_VALUE} + B)$.
 Convert Y to **DATA_TYPE_IN**, generating **MOD_VAR_IN**.
 Write **MOD_VAR_IN**.

Procedure to convert Modbus variable to FOUNDATION fieldbus parameter:

Read **MOD_VAR_OUT**.
 Convert **MOD_VAR_OUT** to float, generating Y
 Calculate $\text{OUTx_VALUE} = (A * Y + B)$.
 Store **OUTx_VALUE**.

$A = (\text{To_EU_100\%} - \text{To_EU_0\%}) / (\text{From_EU_100\%} - \text{From_EU_0\%})$
 $B = \text{To_EU_0\%} - A * \text{From_EU_0\%};$



IN_VALUE, OUT_VALUE: FOUNDATION fieldbus parameters
MOD_VAR_IN, MOD_VAR_OUT: Modbus variables
 Y: auxiliary float variable

Setting Inputs and Outputs of the MBCM block

To read a Modbus variable, connect it to an output of the MBCM function block. To write in a Modbus register, connect it to an input of the MBCM block.

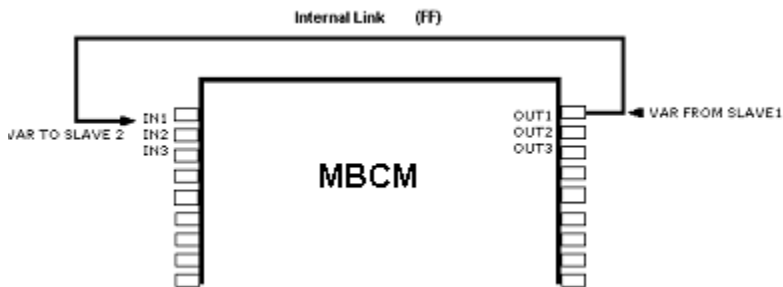
The Modbus protocol specifies the division of the address range to the variables.

- 00001 to 09999 → Digital Outputs.
- 10001 to 19999 → Digital Inputs.
- 30001 to 39999 → Analog Inputs.
- 40001 to 49999 → Analog Outputs.

Once the variables are mapped, defined and referenced in the MBCM block, it is possible to set the strategy.

The variables can be connected to other FOUNDATION fieldbus function blocks (connect the output or input of the block to blocks in the strategy), write in Modbus registers (connect the input of the MBCM block to a Modbus register).

Exchange data between two slaves (set the input of the MBCM block with the slave address and specific Modbus address where the value will be written and set the output of the MBCM block with the slave address and Modbus address of the variable where the value will be read). This last application is showed below:



BAD_STATUS Parameter

This parameter indicates if the communication between slaves was established properly. If the correspondent bit is in logic level 1 this means there was an error during writing/reading of the respective parameter. The table below indicates the values for these status values. If the communication with the specific parameter is good, there is no indication in **BAD_STATUS**, however, if the communication is bad, in the **BAD_STATUS** will be indicated which parameter fails in the communication.

Relation between the bits in BAD_STATUS and Modbus addresses

BIT	VARIABLE
0	IN1
1	IN2
2	IN3
3	IN4
4	IN_D1
5	IN_D2
6	IN_D3
7	IN_D4
8	OUT1
9	OUT2
10	OUT3
11	OUT4
12	OUT_D1
13	OUT_D2
14	OUT_D3
15	OUT_D4

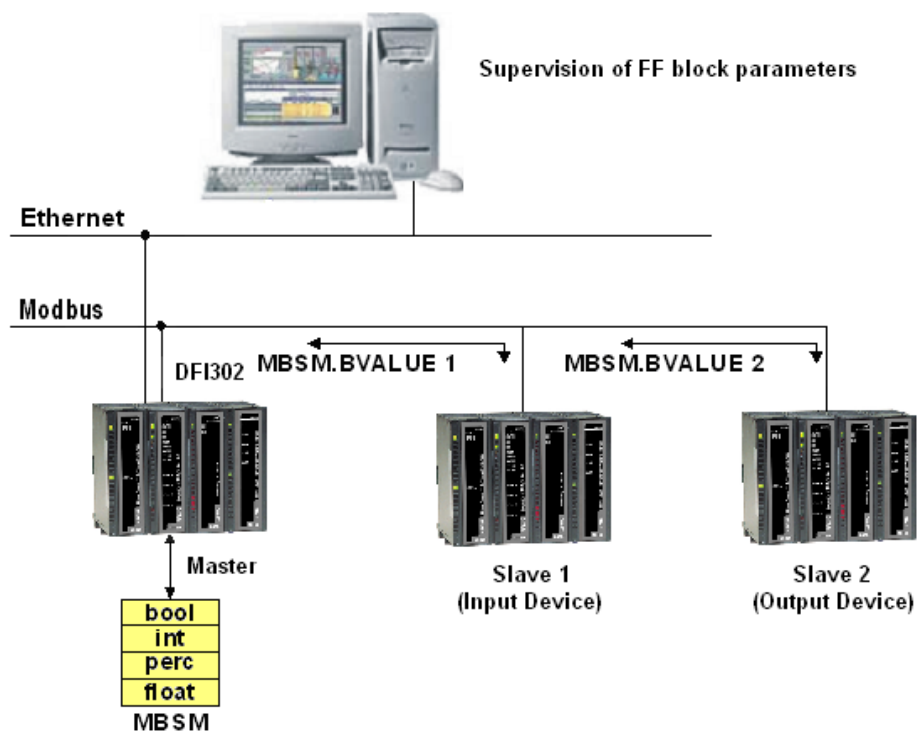
NOTE
Each bit corresponds to an OR between the value and status, indicating if communication with slave is good or bad. If it is only used the value, the status is considered zero. If it is only used the status, the value is considered zero.

MBSM – Modbus Supervision Master

A supervisory system connected to the processor, via **OPC Server**, needs to read and/or write some parameters, mapped directly in some Modbus registers.

Using **Syscon**, create one MBCF and MBSM blocks (1 to 16 blocks are available). In the Characterization window, configure these blocks setting parameters as slave address and parameter address.

MBSM



Configure the **MASTER_SLAVE** parameter to Master in the MBCF block. Define **LOCAL_MOD_MAP** (0 ~ 15).

Parameters Description

For further information about parameters description, please refer to Function Blocks Manual.

Idx	Parameter	Data Type (length)	Valid Range/ Options	Default Value	Units	Store / Mode	Description
1	ST_REV	Unsigned16		0	None	S/RO	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Mode Parameter
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	LOCAL_MOD_MAP	Unsigned8	0 to 15	0		S / O/S	Define the Modbus addresses.
8	BAD_STATUS	Bitstring(2)		0	E	D / RO	Indicate if communication from slave is good or not (each bit corresponds to a Modbus variable).
9	FLOCATOR1	DS-260				S / O/S	Information to locate float parameter
10	FVALUE1	Float		0		N	Value from requested address.
11	FLOCATOR2	DS-260				S / O/S	Information to locate float parameter
12	FVALUE2	Float		0		N	Value from requested address.
13	PLOCATOR1	DS-258				S / O/S	Information to locate percentage parameter
14	PVALUE1	Float		0		N	Value from requested address.
15	PLOCATOR2	DS-258				S / O/S	Information to locate percentage parameter
16	PVALUE2	Float		0		N	Value from requested address.
17	ILOCATOR1	DS-260				S / O/S	Information to locate integer parameter
18	ILENGTH1	Integer8	1,2,4	2		S / O/S	Data length.
19	IVALUE1	Interge32		0		N	Value from requested address.
20	ILOCATOR2	DS-260				S / O/S	Information to locate integer parameter
21	ILENGTH2	Integer8	1,2,4	2		S / O/S	Data length.
22	IVALEUE2	Interge32		0		N	Value from requested address.
23	BLOCATOR1	DS-260				S / O/S	Information to locate boolean parameter
24	BVALUE1	Boolean		TRUE		N	Value from requested addresses.
25	BLOCATOR2	DS-260				S / O/S	Information to locate boolean parameter
26	BVALUE2	Boolean		TRUE		N	Value from requested addresses.
27	BLOCATOR3	DS-260				S / O/S	Information to locate boolean parameter
28	BVALUE3	Boolean		TRUE		N	Value from requested addresses.
29	BLOCATOR4	DS-260				S / O/S	Information to locate boolean parameter
30	BVALUE4	Boolean		TRUE		N	Value from requested addresses.
31	BLOCATOR5	DS-260				S / O/S	Information to locate boolean parameter
32	BVALUE5	Boolean		TRUE		N	Value from requested addresses.
33	BLOCATOR6	DS-260				S / O/S	Information to locate boolean parameter
34	BVALUE6	Boolean		TRUE		N	Value from requested addresses.
35	BLOCATOR7	DS-260				S / O/S	Information to locate boolean parameter
36	BVALUE7	Boolean		TRUE		N	Value from requested addresses.
37	BLOCATOR8	DS-260				S / O/S	Information to locate boolean parameter
38	BVALUE8	Boolean		TRUE		N	Value from requested addresses.
39	UPDATE_EVT	DS-73			Na	D	This alert is generated by any change to the static data.
40	BLOCK_ALM	DS-72			Na	D	The block alarm is used for all configuration, hardware, and connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

Legend: E – Enumerated parameter; Na – Dimensionless parameter; RO – Read only; D – Dynamic; N – Non-volatile; S – Static
Gray Background Line: Default Parameters of Syscon

LOCAL_MODE_MAP Parameter

All MBSM blocks added to the strategy must have different values for **LOCAL_MODE_MAP**. Otherwise the block will not work properly.

FVALUEn, PVALUEn, IVALUEn and BVALUEn Parameters

The user may select these parameters according to the configuration requirements. If the variable to be supervised is float, it is necessary to use a **FVALUE** parameter. If it is a percentage, the **PVALUEn** will work. **IValue** refers to integer values and **BValue** refers to boolean values.

For each of these parameters are associated parameters to address them in the Modbus network, so that the MBSM block knows their location.

FLOCATORn Parameter

It refers to the **FVALUEn** parameter.

This parameter is of the data type **DS-260**, so it is required to set two elements for this parameter:

Slave Address: Insert the Address of the slave where it is located the variable required to monitor. For example, if in an application one LC700 was set with Device Address equal to 1. Thus, slave address must be equal to 1.

Modbus Address of Value: Type the Modbus address of the variable it will be monitored in the MBSM block. Suppose user needs to monitor the variable with Modbus address 40001 located in an I/O module of the slave with Device Address 1. Thus, **MODBUS_ADDRESS_OF_VALUE** must be equal to 40001.

The **FVALUEn** parameters will display the values of the variables set in **FLOCATORn**. Float values use two Modbus registers, but it is necessary only to inform the first.

Modbus Addresses

- 00001 to 09999 → Digital Outputs.
- 10001 to 19999 → Digital Inputs.
- 30001 to 39999 → Analog Inputs.
- 40001 to 49999 → Analog Outputs.

PLOCATORn Parameter

It refers to the **PVALUEn** parameter.

These parameters are **DS-258** data type. They both convert the value to Engineering Units and address the variable in the Modbus network. It is necessary to set these parameters so the monitoring is properly made.

Each of these parameters consists of the following elements:

- From Eu 100 %
- From Eu 0 %
- To Eu 100 %
- To Eu 0 %
- Data Type

See in the following item how to set these elements.

Data Type: It is necessary to inform the data type of the variable.

This parameter only displays a number that refers to a specific format.

DATA TYPE NUMBER	DATA TYPE MEANING
1	Float
2	Unsigned 8
3	Unsigned 16
4	Unsigned 32
5	Integer8
6	Integer16
7	Integer32
8	Swapped Float
9	Swapped Unsigned 8
10	Swapped Unsigned 16
11	Swapped Unsigned 32
12	Swapped Integer 8
13	Swapped Integer 16
14	Swapped Integer 32

The swapped data types were created in order to support the communication of Modbus devices with Profibus devices. Normally, there are the following cases:

4 Bytes (2 Registers – Word)

Normal Data type: Inside Word – Motorola
Inside Register – Intel
Swapped Data type: Inside Word – Motorola
Inside Register – Motorola

2 Bytes

Swapped Data type: Status information is in the Most Significant Byte (MSB)

1 Byte

Swapped Data type: Value (MSB) and Status (LSB) are in the same register.

In case of the Swapped Integer 16 data type no change is necessary.

Slave Address: Inform the address of the slave it is required to reference to the parameter **PVALUEn**. For example, suppose there is a LC700 with Device Address equal to 3 and in this LC700 it is required to monitor a specific variable. Thus, slave address must be equal to 3.

Modbus Address of Value: Inform the Modbus address of the variable it will be monitored. In the example of the previous element, suppose the Modbus address of the monitored variable is 40032. So this element must receive this address.

Procedure to convert FOUNDATION fieldbus parameter to Modbus variable:

Load **VALUEn**.

Calculate $Y = (A * \text{VALUEn} + B)$.

Convert Y to **DATA_TYPE_IN**, generating **MOD_VAR_IN**.

Write **MOD_VAR_IN**.

Procedure to convert Modbus variable to FOUNDATION fieldbus parameter:

Read **MOD_VAR_OUT**.

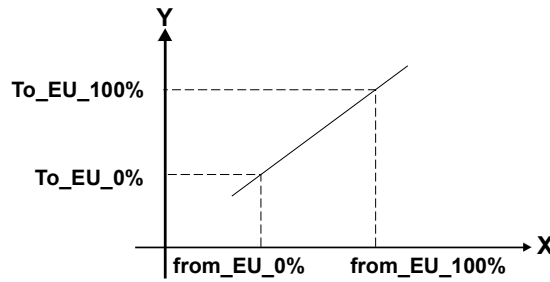
Convert **MOD_VAR_OUT** (from the data type) to float, generating Y

Calculate $\text{PVALUE} = (A * Y + B)$.

Store **OUTx_VALUE**.

$$A = (\text{To_EU_100\%} - \text{To_EU_0\%}) / (\text{From_EU_100\%} - \text{From_EU_0\%})$$

$$B = \text{To_EU_0\%} - A * \text{From_EU_0\%};$$



PVALUEn: FOUNDATION fieldbus parameters

MOD_VAR_IN, MOD_VAR_OUT: Modbus variables

Y: auxiliary float variable

ILOCATORn Parameter

It refers to the **IVALUEn** parameter:

Slave Address: Insert the address of the slave where it is located the variable to be monitored. For example, if in an application one LC700 was set with Device Address equal to 1. Thus, slave address must be equal to 1.

Modbus Address of Value: Type the Modbus address of the variable to be monitored in the MBSM block. Suppose it is necessary to monitor the variable with Modbus address 40001 located in an I/O module of the slave with Device Address 1. Thus, **MODBUS_ADDRESS_OF_VALUE** must be equal to 40001.

The **IVALUEn** parameters will display the values of the variables set in **ILOCATORn**.

BLOCATORn Parameter

It refers to the **BVALUEn** parameter.

This parameter is data type **DS-260**, so this parameter has two elements to be configured:

Slave Address: Insert the address of the slave where it is located the variable to be monitored. For example, if in an application one LC700 was set with Device Address equal to 1. Thus, slave address must be equal to 1.

Modbus Address of Value: Type the Modbus address of the variable to be monitored in the MBSM block. Suppose it is necessary to monitor the variable with Modbus address 40001 located in an I/O module of the Slave with Device Address 1. Thus, **MODBUS_ADDRESS_OF_VALUE** must be equal to 00001.

The **BVALUEn** parameters will display the values of the variables set in **BLOCATORn**.

BAD_STATUS Parameter

This parameter indicates if the communication between slaves was established properly. If the correspondent bit is in logic level 1 this means there was an error during writing/reading of the respective parameter. The table below shows the values for these status values.

Relation between the bits in BAD_STATUS and Modbus addresses

BIT	VARIABLE
0	B1
1	B2
2	B3
3	B4
4	B5
5	B6
6	B7
7	B8
8	I1
9	I2
10	P1
11	P2
12	F1
13	F2

Modbus Slave Addresses

MBCS		
PARAMETER	LOCAL_MOD_MAP = x OFFSET = 40 * x x = 0 ~ 15	e.g. LOCAL_MOD_MAP =1
IN1-Value	40001+ OFFSET 40002+ OFFSET	40041 40042
IN2-Value	40003+ OFFSET 40004+ OFFSET	40043 40044
IN3-Value	40005+ OFFSET 40006+ OFFSET	40045 40046
IN4-Value	40007+ OFFSET 40008+ OFFSET	40047 40048
OUT1-Value	40009+ OFFSET 40010+ OFFSET	40049 40050
OUT2-Value	40011+ OFFSET 40012+ OFFSET	40051 40052
OUT3-Value	40013+ OFFSET 40014+ OFFSET	40053 40054
OUT4-Value	40015+ OFFSET 40016+ OFFSET	40055 40056
IN1-Status	40017+ OFFSET	40057
IN2-Status	40018+ OFFSET	40058
IN3-Status	40019+ OFFSET	40059
IN4-Status	40020+ OFFSET	40060
OUT1-Status	40021+ OFFSET	40061
OUT2-Status	40022+ OFFSET	40062
OUT3-Status	40023+ OFFSET	40063
OUT4-Status	40024+ OFFSET	40064
IN_D1-Status	40025+ OFFSET	40065
IN_D2-Status	40026+ OFFSET	40066
IN_D3-Status	40027+ OFFSET	40067
IN_D4-Status	40028+ OFFSET	40068
OUT_D1-Status	40029+ OFFSET	40069
OUT_D2-Status	40030+ OFFSET	40070
OUT_D3-Status	40031+ OFFSET	40071
OUT_D4-Status	40032+ OFFSET	40072
IN_D1-Value	1+ OFFSET	41
IN_D2-Value	2+ OFFSET	42
IN_D2-Value	3+ OFFSET	43
IN_D2-Value	4+ OFFSET	44
OUT_D1-Value	5+ OFFSET	45
OUT_D2-Value	6+ OFFSET	46
OUT_D3-Value	7+ OFFSET	47
OUT_D4-Value	8+ OFFSET	48

MBSS		
PARAMETER	LOCAL_MOD_MAP = x OFFSET = 40 * x x = 0 ~ 15	e.g. LOCAL_MOD_MAP =1
F_ID1	42601+ OFFSET 42602+ OFFSET	42641 42642
F_ID2	42603+ OFFSET 42604+ OFFSET	42643 42644
F_ID3	42605+ OFFSET 42606+ OFFSET	42645 42646
F_ID4	42607+ OFFSET 42608+ OFFSET	42647 42648
F_ID5	42609+ OFFSET 42610+ OFFSET	42649 42650
F_ID6	42611+ OFFSET 42612+ OFFSET	42651 42652
F_ID7	42613+ OFFSET 42614+ OFFSET	42653 42654
F_ID8	42615+ OFFSET 42616+ OFFSET	42655 42656
I_ID1	42617+ OFFSET 42618+ OFFSET	42657 42658
I_ID2	42619+ OFFSET 42620+ OFFSET	42659 42660
I_ID3	42621+ OFFSET 42622+ OFFSET	42661 42662
I_ID4	42623+ OFFSET 42624+ OFFSET	42663 42664
B_ID1	2601+ OFFSET	2641
B_ID2	2602+ OFFSET	2642
B_ID3	2603+ OFFSET	2643
B_ID4	2604+ OFFSET	2644

NOTES

MBCS

The second column of the previous table shows the values that are attributed to the Inputs and Outputs of the MBCS block according to the value set for **LOCAL_MODE_MAP**. For example, if **LOCAL_MODE_MAP** is set equal to 1 it will result in the Modbus range of addresses showed in the third column. It must be clear that when this parameter is set, a whole range is selected, not a specific address.

INn and OUTn values use two Modbus registers (for example IN1, 40041 and 40042) because their data type is float. IN_Dn and OUT_Dn values use one Modbus register (for example IN_D1, 41). Status values also use only one register.

Once this Modbus range is defined, it is possible to set how the Modbus master will read them.

MBSS

Once values for **LOCAL_MODE_MAP** are set, Modbus Addresses are given to the variables to monitor. So, each integer, float or boolean variable will have a Modbus address associated with it.

For example, suppose **LOCAL_MODE_MAP** = 1 and a float value will be monitored. Setting the **F_ID1**:

F_ID1.Tag = Tag of the float parameter necessary to monitor.

F_ID1.Index= Index of the first column of the parameter necessary to monitor.

F_ID1.subindex = The Sub Index is used for parameters that have a structure. In this case it is necessary to indicate which element of the structure is being referred.

See the table **Modbus Slave Address** The Modbus addresses given to this parameter (remember, float values use two Modbus registers) are 42641 and 42642.

ATTENTION

The MBSS block is valid for DF51 controller.

It is recommended the use of MBCS block instead of MBSS block. This provides a better performance to data obtainment and for transmission.

Modbus Commands

When the DFI302 works as master, that is, reading the Modbus addresses, it uses the commands 1 (addresses from 1 to 9999), 2 (addresses from 10001 to 19999), 3 (addresses from 40001 to 49999) and 4 (addresses from 30001 to 39999).

When the DFI302 writes on the Modbus addresses, it uses, regarding the MBCM block, the commands 15 (addresses from 0 to 9999) and 16 (addresses from 40001 to 49999). Regarding the MBSM block, the DFI302 uses the commands 5 (addresses from 0 to 9999) and 6 (addresses from 40001 to 49999).

If the DFI302 works as slave, it responds to any of the commands above.

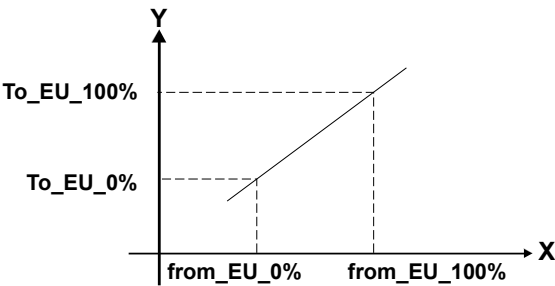
PERFORMANCE TIPS

For a better protocol performance, and an optimized response time among master and slaves communication, it is recommended that:

- a) The baud rate parameter to the serial communication should be configured to 38400 bps in both Modbus master and slave devices;
- b) The Timeout should be adjusted conveniently (as small as possible) for a shorter response time among the masters and slaves devices, and if any failures occur, the communication status is updated. For a problem related with “links” and “status” oscillations due to the configuration network is necessary to increase the timeout in the master device.
- c) The address mapping in the Modbus blocks has to be sequential. For example, when a master Modbus controller communicates with other slave Modbus device, in the MBCM and/or MBSM block you should configure the **SCALE_LOC_INX**, **SCALE_LOC_OUTX**, **FLOCATORX**, **PLOCATORX**, **ILOCATORX**, **BLOCATORX** parameters to sequential addresses, i.e., IN1: 40001, IN2: 40003, IN3: 40005 etc. In this way, the command is grouped in just one packet and is sent to the slave. This optimizes the traffic in the network.
- d) A better response time you will get as much as you succeed in the sequential addressing for the inputs parameters (IN_n and IN_Dn).
- e) In the outputs (OUT_n and OUT_Dn), each command can be group in just one packet if the addresses in the parameters has a gap less than 125 points to the analog and 2000 points to the discrete. For example, if in the OUT_n parameter the address is 45200 and in other parameter is 45570, there is a gap of 370 points (> 125), then you will have 2 commands in the bus. On the other hand, if it is possible to map the address in a range where its gap is less than 125, just one command will be sent in the line for the analog data. Similarly, this is applied to the discrete data if gaps less than 2000.
- f) Do not instantiate blocks that are not being used in the configuration or address whose slave devices that does not exist. This will reduce the CPU performance and the Timeout may occur prejudicing the communication of the others controllers.
- g) More tips can be found in the **Troubleshooting** section.

Scaling Conversion

This data structure consists of data used to generate constants A and B in equation $Y = A \cdot X + B$.



E	ELEMENT	DATA TYPE	SIZE
1	From EU 100%	Float	4
2	From EU 0%	Float	4
3	To EU 100%	Float	4
4	To EU 0%	Float	4
5	<div>Data Type (Use this parameter to convert Fieldbus to Modbus or Modbus to Fieldbus, where Modbus should be ...) Float = 1 Unsigned8 = 2 Unsigned16 = 3 Unsigned32 = 4 Integer8 = 5 Integer16 = 6 Integer32 = 7 Swapped Float = 8 Swapped Unsigned8 = 9 Swapped Unsigned16 = 10 Swapped Unsigned32 = 11 Swapped Integer8 = 12 Swapped Integer16 = 13 Swapped Integer32 = 14</div>	Unsigned8	1

Redundancy and Modbus

The redundancy in Modbus communication may be mapped to many possible scenarios. Some of them are supported by the controllers. The restrictions which should be considered are described below:

1. The Modbus TCP in the DF63 will be established in both Ethernet ports (ETH1 and ETH2). It is recommended to use both ports only in the scenarios where the DF63 is slave. When running as Modbus Master, the DF63 should not be configured to use ETH2 to avoid duplication in the bandwidth and it may damage the communication performance.
2. The both Ethernet ports (ETH1 and ETH2) DF73, DF75, DF79, DF81, DF89 (see note below), DF95 and DF97 controllers, which can be slaves only, are available to answer the Modbus requests.
3. Switch over conditions are related to bad conditions in the Primary controller (see description of these bad conditions in another section of this manual "Adding Redundancy to DFI302 HSE Controllers"). As slave Modbus, the controllers will not generate a switch over condition because of communication failures. It is always the master who should switch over in such conditions. When using TCP and all IPs are in the same subnet, the master should to decide when switch the request to other slave Ethernet connection.
4. As Modbus Master, DF62/DF63 (either TCP or Serial RTU) only switch over if the Primary does not have any answer from all slave modules.

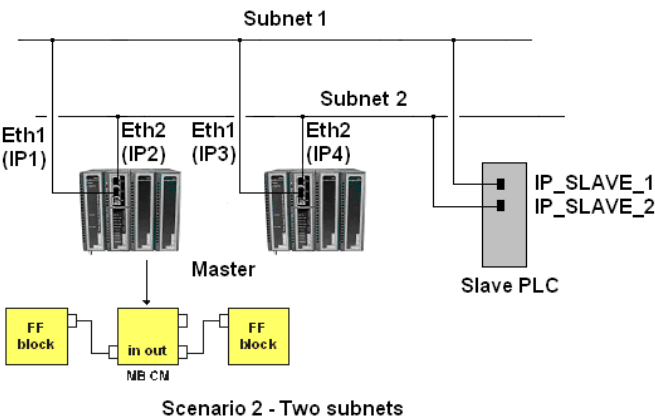
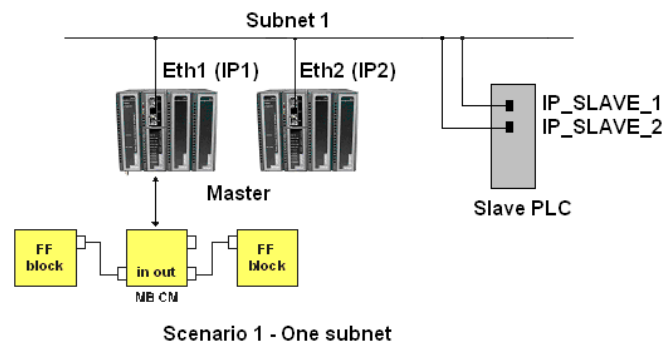
NOTE

This DF89 redundancy is valid for firmware versions up to version V4.xx and DD 04.xx. From this version, the DF89 has other configuration form that is shown in detail in the "Creating a Modbus configuration using the DF89."

Redundant DF63 as master and redundant PLC as slave, using Modbus TCP

There are two possible scenarios, in the first only one subnet is used and all IPs of equipment are on the same subnet.

In the second scenario, two subnets are used, and each device has an Ethernet port connected to one of them, see the following figure. The second scenario is the most recommended if the availability of the Modbus communication is an important factor, because in this case if there is some failure in one switch the another network/subnet will cover this failure.



Network topologies for DF63 redundant master and slave PLC with two Ethernet ports

The redundant PLC can either use two Ethernet cards or one card with two Ethernet ports.

Once the Primary DF63 executes the requests, it is assumed that any of the slave connections may receive and process the request. If one connection is not running properly the primary DF63 will use another available connection (connection switching).

If the Primary DF63 did not receive response to any of the connections and the Secondary DF63 has response to at least one of the connections, the DF63 pair will switch over (scenario Bad Condition/ controller switching).

IMPORTANT

For scenarios in which the two Ethernet ports of the slave PLC are used for connection to the Master via TCP, the **MBCF** block of the master has to be configured as follows:

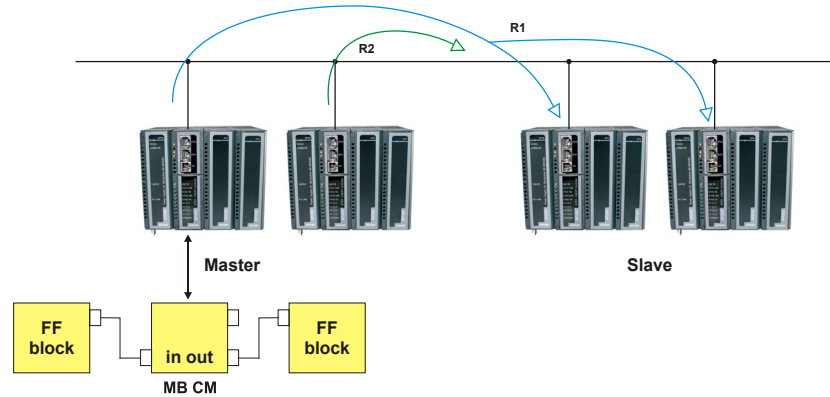
- The **SLAVE_ADDRESSES** parameter with the IPs of the slave’s ports (**IP_SLAVE_1** and **IP_SLAVE_2** parameters). No more than two IPs are supported by the same slave.
- In the **MODBUS_ADDRESS_SLAVE_1** and **MODBUS_ADDRESS_SLAVE_2** parameters must be configured the Modbus Address related to the slave, identical for both parameters because it is the same slave (see example in figure below).

NUMBER OF TRANSMISSIONS			14	RW
SLAVE_ADDRESSES			15	
IP_SLAVE_1	192.168.164.200		.1	RW
IP_SLAVE_2	192.168.165.200		.2	RW
IP_SLAVE_3			.3	RW
IP_SLAVE_4			.4	RW
IP_SLAVE_5			.5	RW
IP_SLAVE_6			.6	RW
MODBUS_ADDRESS_SLAVE_1	1		.7	RW
MODBUS_ADDRESS_SLAVE_2			.8	RW
MODBUS_ADDRESS_SLAVE_3			.9	RW
MODBUS_ADDRESS_SLAVE_4			.10	RW
MODBUS_ADDRESS_SLAVE_5			.11	RW
MODBUS_ADDRESS_SLAVE_6			.12	RW
RESTART MODBUS			16	RW

Partial view of the MBCF block - parameters related to the Modbus slave

Redundant DF63 as master and redundant PLC as slave, using Modbus RTU

In this scenario, the DF63 is redundant, using 232 serial port, and it is connected via multidrop to redundant PLC (both ports), using 232/485 converter. Once the active DF63 executes the requests (see R1 in the following picture), it is assumed that only one slave will process the request. In case of any "bad condition" the DF63 will switch over (see R2 in the following figure).



Redundant DF63 as slave and redundant PLC as Master, using Modbus TCP

In this scenario, the Primary and Slave DF63 will answer the reads requested by the Master PLC. The writes will be executed only by the Primary DF63. If for any reason the Secondary DF63 receives a write, so this request will be forwarded via redundancy path from Secondary DF63 to Primary DF63.

Redundant DF63 as slave and redundant PLC as Master, using Serial RTU

In this scenario, only the Primary DF63 will answer the reads and writes requested by the Master PLC.

Using Modbus in controllers DF73, DF75, DF79, DF81, DF89, DF95, DF97, and DF63-MBS

The use of Modbus in DF73, DF75, DF79, DF81, DF89 (see NOTE DF89), DF95, DF97, and DF63-MBS (see NOTE DF63-MBS) controllers is different from the way used by other controllers. This difference customizes the need for high performance in discrete and continuous Logic Ladder.

The main requirement for this module is simultaneously support Modbus through Serial port (RS-232) and TCP/IP. Therefore some rules must be followed in the MBCF block.

The instantiation and configuration of the MBCF block in the **Syscon** is mandatory to permit reading and writing operations via Modbus. Some block parameters are already configured automatically (see the following table in Parameters Description topic).

NOTE

The Modbus Supervision Slave (MBSS), Modbus Control Slave (MBCS), Modbus Supervision Master (MBSM), Modbus Control Master (MBCM) blocks and the bypass functionality (Modbus masters) they are not available for DF73, DF75, DF79, DF81, DF89, DF95, DF97, and DF63-MBS controllers.

Communication Channel

The DF73, DF75, DF79, DF81, DF89, DF95, DF97, and DF63-MBS controllers work simultaneously via RS-232 and TCP/IP (Default state). Therefore, requests from the Modbus master are answered to both on the RS-232 port and on the Ethernet ports of these controllers (Modbus slaves).

For this type of communication is recommended a maximum of 5 Modbus connections masters (4 via Ethernet port and 1 via Serial port) with the DF73, DF75, DF79, DF81, DF89, DF95, DF97, and

DF63-MBS (Modbus slaves). Is recommended that master's scan does not overload the slave with more than two requests per second.

Modbus Addresses

The DF73, DF75, DF79, DF81, DF89, DF95, DF97, and DF63-MBS Modbus addresses are automatically generated by **LogicView for FFB** for any Ladder input and output. There is the possibility to manually configure the addresses.

Further information about the **LogicView for FFB** Modbus addresses generation, please consult the section "Adding Logic using Functional Blocks (FFB 1131 - Flexible Function Blocks) " and the **LogicView for FFB** manual.

Parameters Description

For further details about the parameters description omitted in this section, please consult the Function Blocks Manual.

Idx	Parameter	Data type (Comp.)	Valid Strip / Options	Value Default	Units	Memory / Way	Description
1	ST_REV	Unsigned16		0	None	S/RO	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1a 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	See MODE_BLK parameter in the Function Blocks manual.
6	BLOCK_ERR	BitString(2)			E	D/RO	
7	MEDIA	Unsigned8	0:Serial, 1:TCP/IP	Serial	E	S	Parameter not used. Always Serial.
8	MASTER_SLAVE	Unsigned8	0:Master, 1:Slave	Slave	E	S	Parameter not used. Always Slave.
9	DEVICE_ADDRESS	Unsigned8	1-247	1	E	S	Defines the DFI modbus address (only for DFI slave).
10	BAUD_RATE	Unsigned8	0:110, 1:300, 2:600, 3:1200, 4:2400, 5:4800, 6:9600, 7:19200, 8:38400, 9:57600, 10:115200	19200	E	S	Defines the transmission rate (only for communication serial).
11	STOP_BITS	Unsigned8	0:1, 1:2	1	E	S	Defines the number of stop bits (only for media serial).
12	PARITY	Unsigned8	0: None, 1:Odd, 2:Even.	Odd	E	S	Defines the parity (only for communication serial).
13	TIMEOUT	Unsigned16	0-65535	1000	ms	S	Parameter not used.
14	NUMBER_RETRANSMISSIONS	Unsigned8	0-255	1		S	Parameter not used.
15	SLAVE_ADDRESSES	DS-263				S	Parameter not used.
16	RESTART_MODBUS	Boolean		False		S	Parameter not used.
17	TIME_TO_RESTART	Unsigned16	1-65535	0	ms	S	When the device is working as master, it is the time between the periodic scan of those commands. When the device is working as slave, it is the minimum time between each Modbus request and Modbus response. When the device is working as master, the default value is 1000 ms and when it is slave the default value is 0.
18	RTS_CTS	Boolean		False		S	Parameter not used.
19	ON_APPLY	Unsigned8	0: None, 1:Apply	None	E	S	Applies the changes done in the Modbus blocks.
20	CHECK_COMM_STANDBY	Unsigned8	0-255	0	Na	S/RW	Parameter not used.

Legend: E – Parameters List; Na – Parameter Dimensionless; RO – Read Only; D – Dynamic; N – Not Volatile; S - Static
Line with Completion of Gray Fund: Main parameters to be configured and Default of the Syscon.

Troubleshooting

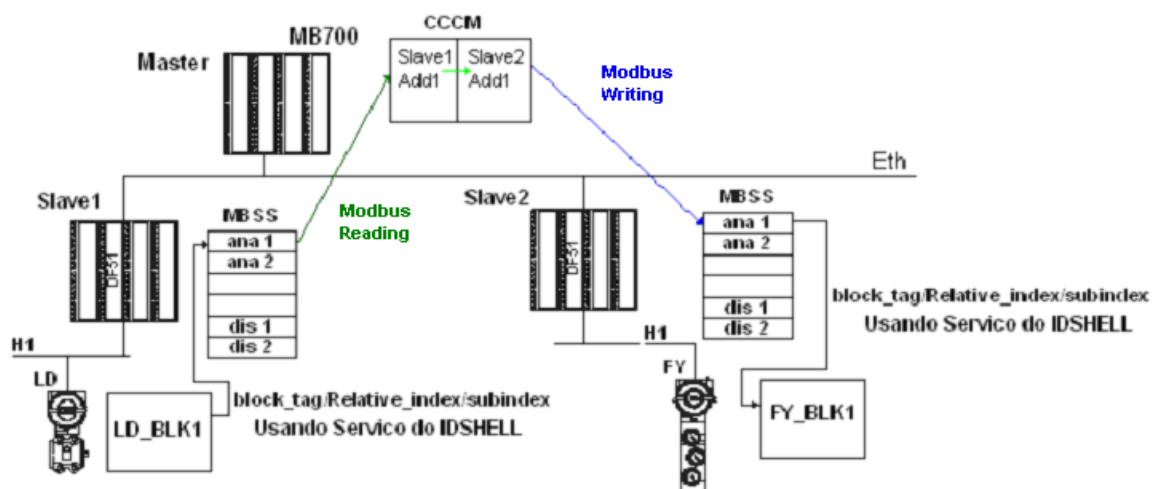
A) The communication or supervision with the bridge was lost due to the use of MBSS block

Problem: Modbus Supervision Slave block (MBSS) is used to control where master writes cyclically in the MBSS block. For this case, the writing/reading queue of the DF equipment line will be full. The possible causes are supervision stops or lose the communication between **Syscon** and bridge. This problem occurs when there are writings of MBSS in other block, if the MBSS is writing in the DF51 or in another transmitter.

Cause: The MBSS block aims supervision tasks (**IDSHELL**) and it is not its purpose receives cyclic writings. Another relevant point is that the writing has priority over the supervision. So, when Modbus master writes cyclically in the MBSS block and mainly in the TCP (where media is faster), and with frequent writings, the consequence will be the full supervision/writings queue in the **IDSHELL** block (DF51), so only Modbus tasks will be executed and it will not have time to execute other tasks.

Example for this scenario: MB700 working as master of DF51 (slave) in the TCP/IP media.

In the example below, Concentrate Control Master block (CCCM) of MB700 reads **LD_BLK1** data (which is in the LD302 transmitter) of the slave1 and writes data for the slave2 (writing in the **FY_BLK1** block of FY302).



Solution:

1) It recommends use the **MBCS** block when Modbus master writes cyclically, because this block does not use supervision tasks (**IDSHELL**) to send data for the Modbus. That means the data in the slave will be written during the block macro cycle time.

In the example of the previous figure, if the **MBSS** data direction is from slave2 to control block, then the **MBSS** block must be replaced by a **MBCS** block, which the **OUT_xx** of the **MBCS** will link to the control block.

2) If the **MBSS** block is necessary, the update time must be set long in the master device (time about some seconds). This time setting can be done in the **CONTROL_OFF_DUTY** parameter of the MB700 and in the **TIME_TO_RESTART** parameter of the DF51.

Observations: For cyclic writing in transmitters, the minimum time between the writings must be at least 2 seconds to not stop the supervision in the DF51.

B) Writing in static parameters of transmitters causes decrease of the EEPROM service life of the device

Problem: Using the MBSS block, master writes cyclically in static parameters of field devices. In this case, it can cause the decrease of the EEPROM service life because frequent writing.

Example for this scenario: MB700 working as master of the DF51 (slave) in the TCP/IP.

Using the example of the previous figure, the Concentrate Control Master block (CCCM) of MB700 writes cyclically in the **CT_VAL_1** parameter (which is a static parameter) of the Constant block (**FY_BLK1**) that is in the FY302 of slave2.

Solution:

One solution is to avoid writing in static parameters. An example is showed by using the Constant block. Instead of writing in the **CT_VAL_xx** parameter (which is a static parameter) with the block in **AUTO** mode, it is possible to write in the **OUT_xx** parameter (which is a dynamic parameter) with the block in **MAN** mode. In this case, when a reset occurs in the transmitter, the Constant block output will be zero (default) until the communication between the Modbus master and slave comes to normal situation.

C) Optimizing the communication by reducing the number of Modbus commands

Problem: When using Modbus writing commands is necessary to avoid gaps between the addresses. Gaps are intervals between two consecutive addresses. For example, there is writing for the addresses 2001 and 2005, and there is no writing for the addresses 2002, 2003 and 2004.

This problem is lesser for readings, because the reading supports gaps between the addresses and also it has a limit. The limit for discrete points is 1200 (it supports up to 1200 points in the same command. For example, the range address is from 2001 to 3201 and this range is in the same command). The analog points limit is 120.

Example for this scenario: MB700 working as master of a slave LC700 in the serial media.

Concentrate Control Master blocks are set in MB700 and there are cyclical writings (**IN_x** parameters) in the Modbus slave. In this configuration the Modbus points 1, 5, 6, 9, 11, 13, 15 are used. For this case are set six Modbus commands and only one can be used for that. Considering the baud rate equals to 9600 and the time equals to 100ms to execute each command, the writing of all variables will take 600 ms, and the writing in an optimized way will take only 100ms.

Solution:

If the DF51 is the slave device, it must be use variables in sequence of the same block **OUT_1**, **OUT_2**, **OUT_3**, etc.

Section 16

CREATING A FIELDBUS STRATEGY BY USING DF51

Introduction

This section describes the strategy configuration by using the DF51 controller as Bridge. The control loop is shown below.

PROJ_DF51

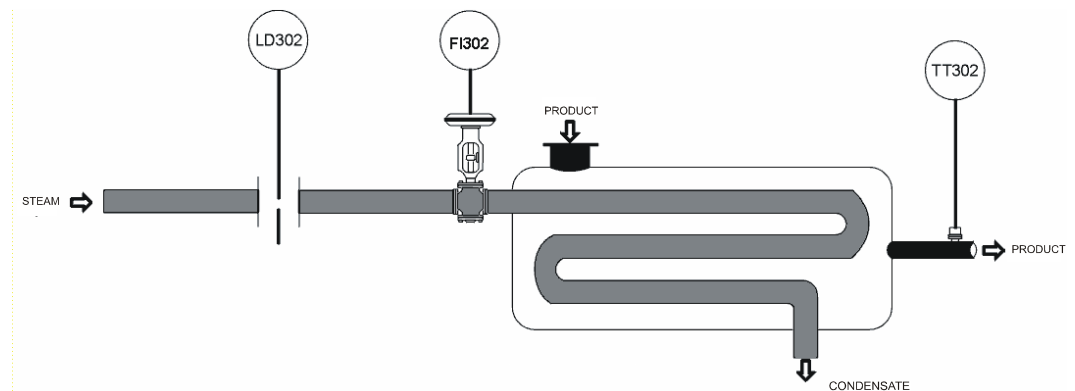


Figure 16. 1 – Example of temperature control process

The purpose of this process is to control the fluid output temperature (controlled variable) using steam flow rate (manipulated variable) to heat it. The fluid temperature will be sent to the master controller, where it will be compared to a temperature set point.

The master output would be the slave controller set point, which will control the steam flow rate to the heat exchanger.

Starting the Area

Step 1

It is possible to create, or edit, an area from the **Studio302**. In the **Studio302** interface select **Areas**. A window will appear listing all areas of database.

To create a new area from the **Studio302**, left-click inside the **Areas** window, then choose **New Area**.

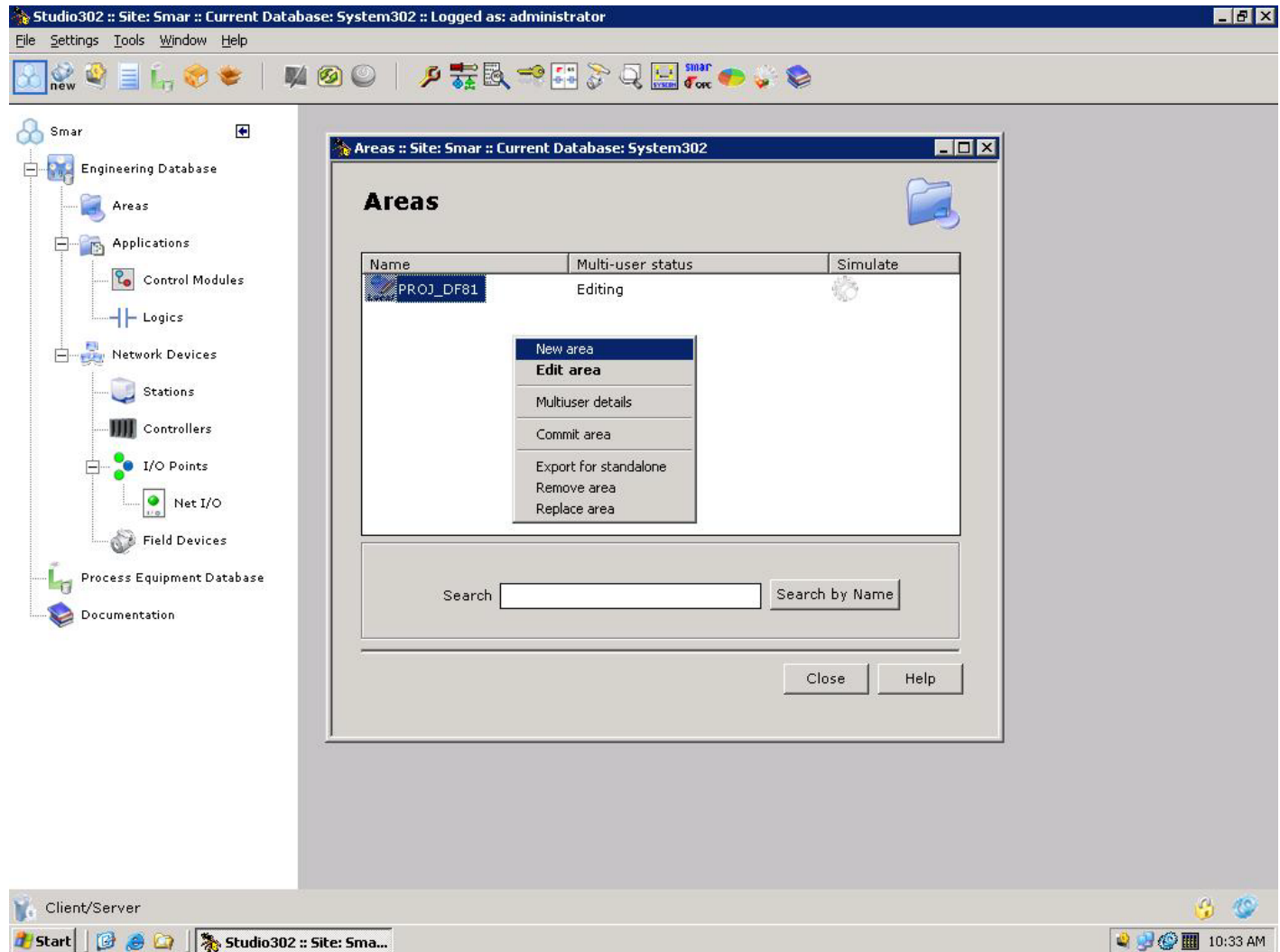



Figure 16. 2 – Creating a new area

Another way to create a new area is from **Syscon**. Click the icon  in the **Studio302** toolbar.

To create a new area on **Syscon**, choose **File → New**, or through the toolbar, choose **New** button . The dialog box shows the areas options. Select **Area** as shown in next figure:

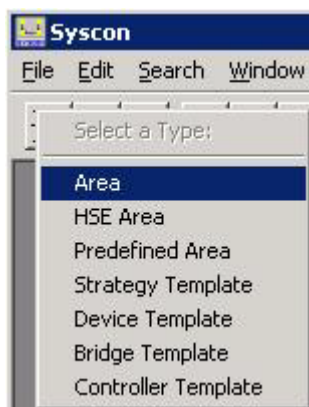


Figure 16. 3 – Options to create Syscon areas

After choosing the area type, it opens a window to the user give a name to the new area.

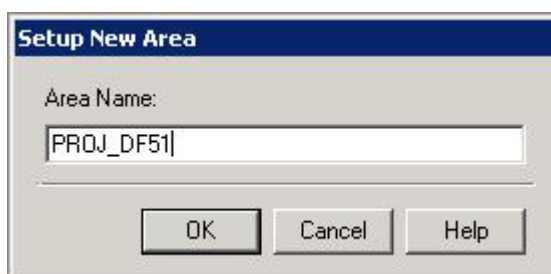


Figure 16. 4 – New area name

Type the name for the area in the **Area Name** box, and click **Ok**. For this example, it chooses **Proj_DF51** name.

A new window will appear. This window has:

- **Application** – Logical Plant. To insert control strategies into this part.
- **Fieldbus Networks** – Physical Plant. To add devices and function blocks to the area into this part.



Figure 16. 5 – Area divisions

Physical Plant Project

Step 2


In the main window, **PROJ_DF51**, right-click the **Fieldbus Networks** icon, , to select the Server. Choose **Communication Settings** option, or through the toolbar, choose **Communication→Settings**. The communication settings dialog box will open:



Figure 16.6 – Choosing the Server

Confirm if the **Smar.DFIOLEServer.0** option has already been selected. Otherwise, the user must select it, and then click **OK**.

Arranging Fieldbus windows

Step 3

Right-click the **Fieldbus Networks** icon, and then choose **New Fieldbus** option.

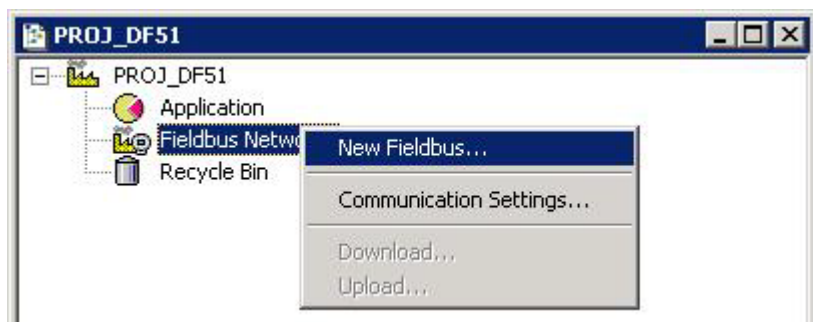


Figure 16.7 – Adding a Fieldbus channel

When selecting the **New Fieldbus** option, the dialog box to choose the channel type and name it with tags opens. If the user needs to name the channel with a specific tag, it must be written in this box. After that, click **OK**. Otherwise, the default tag will be attributed to the channel.



Figure 16.8 – Selecting the Fieldbus channel and naming it with a tag

In the **PROJ_DF51** window, the **CANAL_00** will be inserted into the **Fieldbus Networks**:

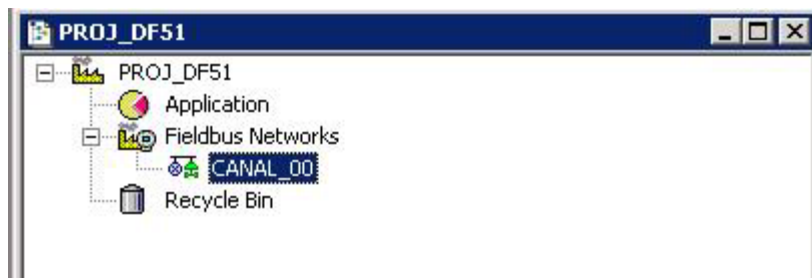


Figure 16.9 – Fieldbus channel inserted

Right-click the **CANAL_00** icon and choose **Expand**. A new window will appear.

To arrange the screen, click the main area window. So, choose **Window** menu on the **Syscon** toolbar, and then choose **Tile** option.

Adding Bridges

Step 4

Now the bridges that will be used for this area can be added.

First, the DF51 controller must be inserted. In the **PROJ_DF51** window, right-click the **CANAL_00** icon. Select **New → Bridge** as shown in the next figure:

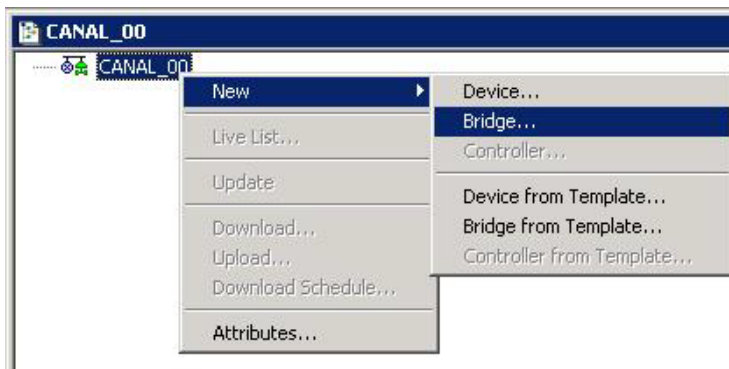


Figure 16.10 – Inserting the bridge to the area

After inserting the Bridge, it opens a window as shown below.

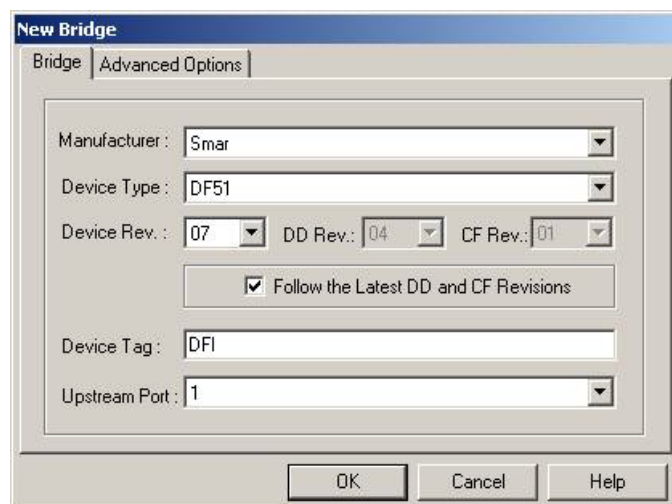
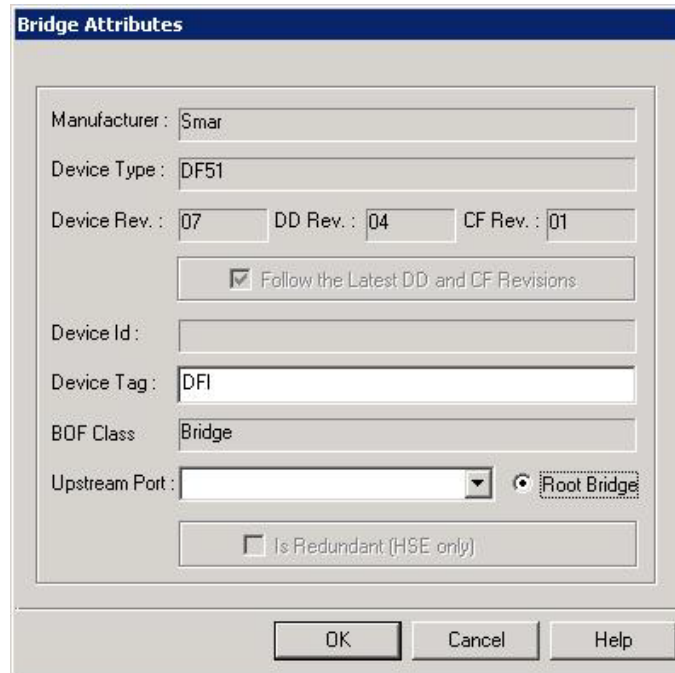


Figure 16.11 – Setting the Bridge

Select **DF51** in the **Device Type** box. In the **Device Tag** box, type **DFI** or another tag, and click **OK**.

IMPORTANT	
Not all characters are valid when naming the elements, so pay attention:	
The valid characters are:	
A-Z a-z 0-9 # { } [] () + -	
The invalid characters are:	
~ ` ! @ # \$ % ^ & * = : ; , . < > ? / ' " \	

In the **CANAL_00** window, right-click the **DFI** icon, , Select **Attributes** option to see the bridge's attributes and, if necessary, change its tag. Click **OK**. See the next figure:



The **Bridge Attributes** dialog box contains the following fields and controls:

- Manufacturer:** Smar
- Device Type:** DF51
- Device Rev.:** 07
- DD Rev.:** 04
- CF Rev.:** 01
- ☒ Follow the Latest DD and CF Revisions
- Device Id:** (empty)
- Device Tag:** DFI
- BOF Class:** Bridge
- Upstream Port:** (dropdown menu)
- ☒ Root Bridge
- ☐ Is Redundant (HSE only)

Buttons at the bottom: **OK**, **Cancel**, **Help**.

Figure 16. 12 – Bridge attributes

Adding Fieldbus Devices

Step 5

After adding the bridge for the area, it is possible to insert the field devices. First, return to the **PROJ_DF51** window and right-click the **CANAL_00** channel. Select **New→Device**

The dialog box for choosing the device and naming it with a tag will be shown. The user can select **Smar** in the **Manufacturer** box, select **TT302** in the **Device Type** box, and then name this device with the tag **TIC001**, or with another tag. After finishing, click **OK**:

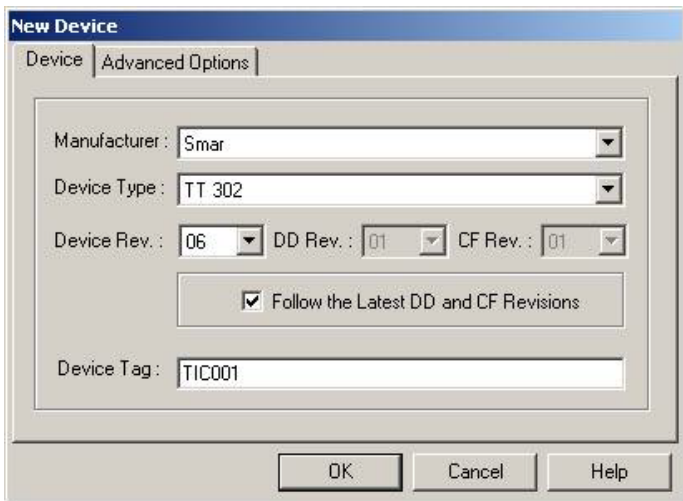


Figure 16. 13 – Setting the Fieldbus Device

Repeat this procedure to add a transmitter (LD302) and a converter (FI302) in the flow control.

After adding the devices to the area, the **CANAL_00** channel will be as shown in the following figure:

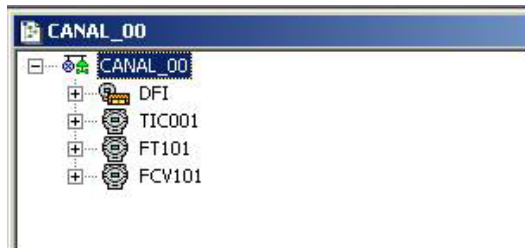



Figure 16. 14 – Device added into the Fieldbus channel

Adding Function Blocks

Step 6

Now the user can add Function Blocks. To add a new Function Block (FB), click the  sign, and right-click the **Virtual Field Device (FB VFD)** icon. Select **New Block** item.

The FB VFD is responsible for the data management.

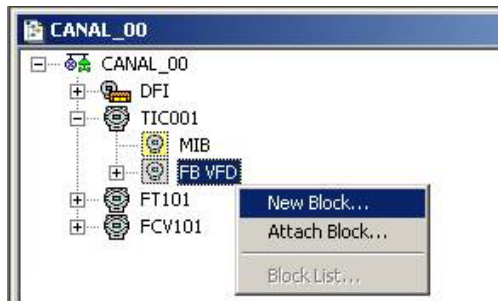


Figure 16. 15 – Selecting new blocks

The **Function Block Characterization** dialog box will appear. The **Block Type** option shows the Smar's FB.

Select the block in the **Block Type** box and name it in the **Block Tag** box. The next figure shows adding the Analog Input function block.

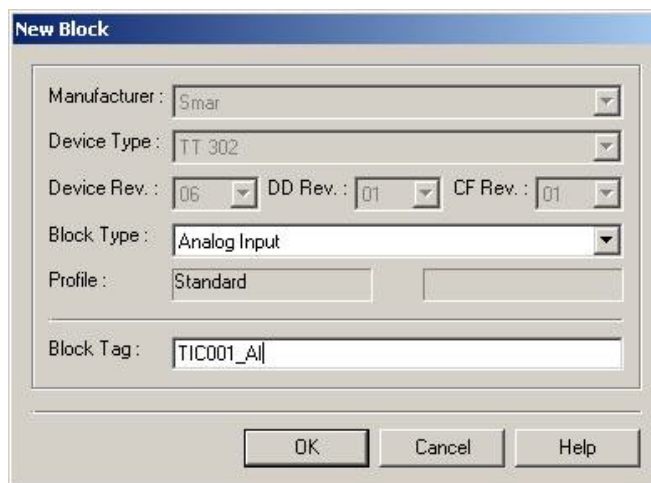


Figure 16.16 – Adding function blocks to the device

For this example, it is necessary to add AI, PID and AO function blocks to build a cascade control configuration.

NOTE

From the **Syscon** version 6.00, it is not necessary to configure the Transducer (TRD), Resource Block (RES), Diagnostics (DIAG) and Display (DSP) blocks, because they are preinstantiated in the devices.

The channel configuration with all function blocks and devices is showed below. For better identification of the Transducer, Resource and Diagnostics function blocks name them with specific tags.

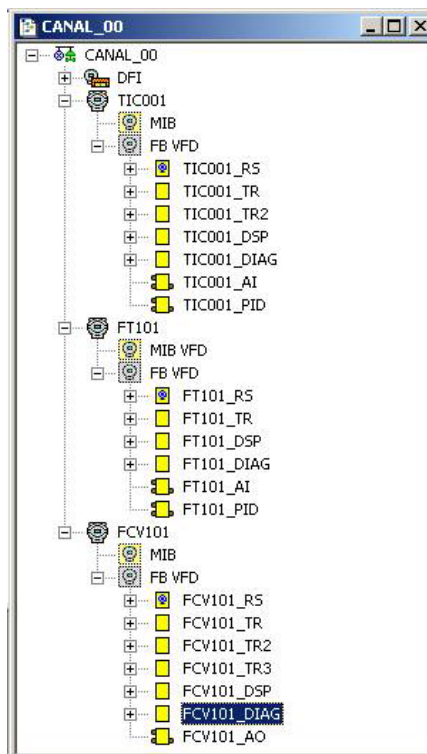


Figure 16.17 – Fieldbus channel composition

Now the strategy on the Application (Logical Plant) can be developed. First it is necessary to establish a new process cell.

Creating New Process Cells

Step 7

The **Logical Plant** can be divided in several process cells, according to the plant.

To create a new process cell, right-click the **Application** icon and select **New Process Cell** item.

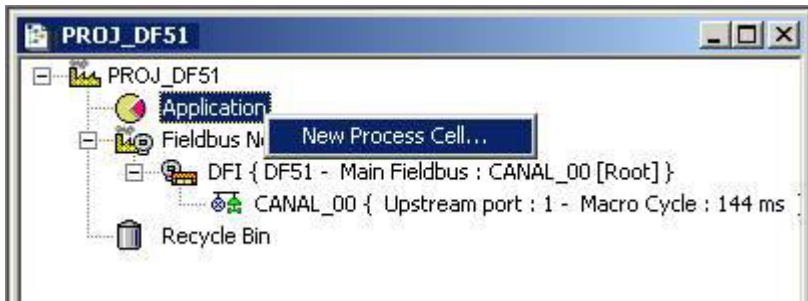


Figure 16. 18 – Adding a Process Cell

The dialog box to attribute the tag to the **Process Cell** will open:

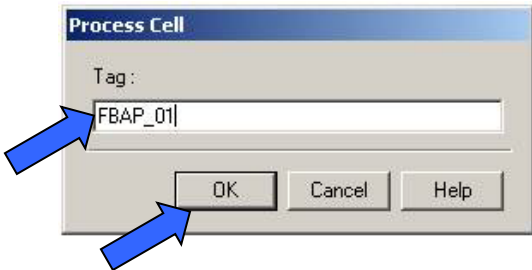


Figure 16. 19 – Attributing tag to the Process Cell

If the user needs name the Process Cell with a specific tag, can enter it in the **Tag** box, and click **OK**. To create more process cells, the procedure above has to be repeated.

After inserting the Process Cell, the **PROJ_DF51** window will be according to the following figure:

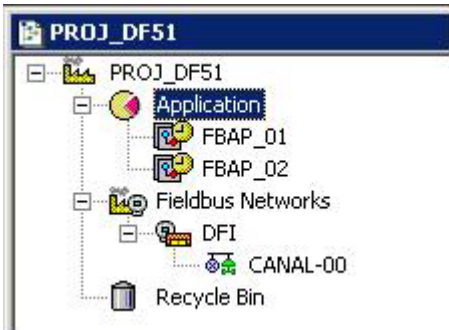


Figure 16. 20 – Area window after inserting the Process Cell

NOTE

The user must remember that **Application** is a virtual division. It only divides a large plant. For example: if the plant has two networks, they can be FB (**FBApplications**) in the **Syscon**. One **Application** can have several **FB Applications**, but a **FB Application** can not be in more than one **Application**.

Creating a Control Module (FBApplication)

Step 8

Now the user can create a Function Block Application, Control Module, in the **Application** section.

Right-click the **FBAP_01** icon and select **Expand** item.



Figure 16.21 – Creating a FB Application

To arrange the screen, click the area window. So, choose the **Window** menu on the **Syscon** toolbar and then choose **Tile** option.

Return to the **FBAP_01** window. Right-click the **FBAP_01** item and choose **New Control Module**. See the next figure.

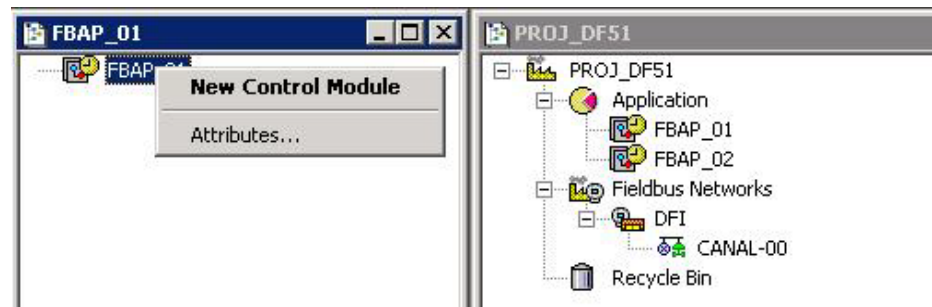


Figure 16.22 – Creating the new Control Module

The **Control Module** dialog box will appear. Name it with a tag related to the Process Cell. Click **OK** to conclude this task.

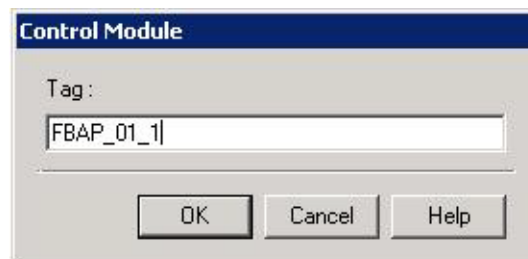


Figure 16.23 – Attributing tag to the Control Module

IMPORTANT

Remember that not all characters are valid when naming the elements with tags.

The valid characters are:

A-Z a-z 0-9 # { } [] () + -

The invalid characters are:

~ ` ! @ # \$ % ^ & * = | : ; , . < > ? / ' " \

Inserting Blocks in the Control Module

Step 9

Now the user can add function blocks for the devices in the Logical Plant. Right-click the **FBAP_01_1** item and select **Attach Block** option, as shown in the next figure.



Figure 16. 24 – Attaching blocks to the FBAP

The **Attach Block** dialog box will open:

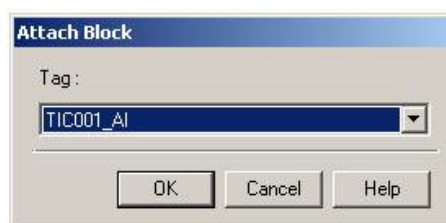


Figure 16. 25 – Inserting blocks to the FB Application

The available function blocks for the Control Module are showed in the **Attach Block** box. For the aimed strategy, the function blocks that must be inserted will appear in the box. So, select them one by one, and click **OK**.

When the Attach Block process ends, the Control Module will be as shown in the next figure:

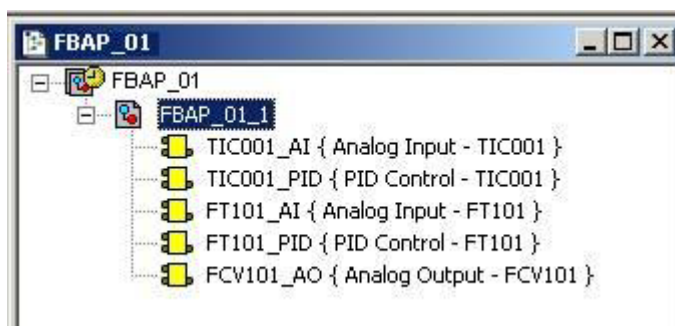


Figure 16. 26 – Blocks added to the FB Application

Another way to attach the blocks to the Control Module is left-clicking the element and drop it to the window.

Configuring the Control Strategy

Step 10

Now the user is ready to develop the control strategy.

First, right-click the FBAP icon and choose **Strategy**. The **Strategy** window will appear as shown in the following figure.

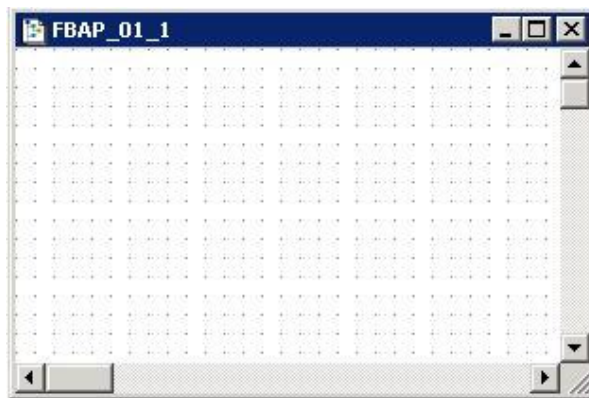


Figure 16.27 – Strategy window

At this moment there are 3 or 4 windows opened in the **Syscon**. Minimize the **CANAL_00** window. To arrange these windows, click the **FBApplication** window, and then the **Proj_DF51** window. On the toolbar, choose **Window → Tile**.


If the user does not have a monitor upper than 17", it is recommended to minimize the strategy window. Thus the whole area can be visualized.

The strategy window offers several tools for drawing. Refer to the **Syscon**'s Help for further details.

Adding Blocks to the Strategy window

Step 11

Now the function blocks can be added to the **FBAP_01_1** window.

In order, click the first block,  TIC001_AI1, and drop it into the strategy window. A function block will be created automatically.

The following figure shows the function block added to the strategy window:

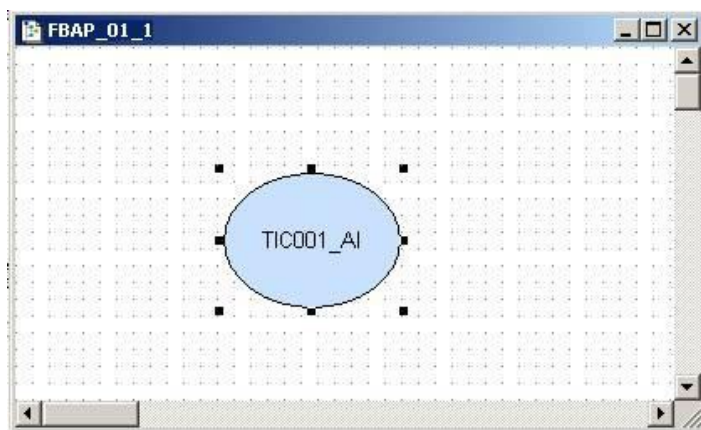



Figure 16.28 – Block inserted into the strategy window

The drag-and-drop procedure must be repeated for the other blocks such as TIC001_PID, FT101_AI, FT101_PID e FCV101_AO.

Linking the Blocks

Step 12

There is a specific tool to link the blocks, the **Link** button, , on the **Strategy** toolbar.

Click this button on the toolbar, and then in the TIC001_AI function block. The dialog box for linking the input and output parameters will appear. Select **OUT**, and then click the **OK** button as shown in the following figure.

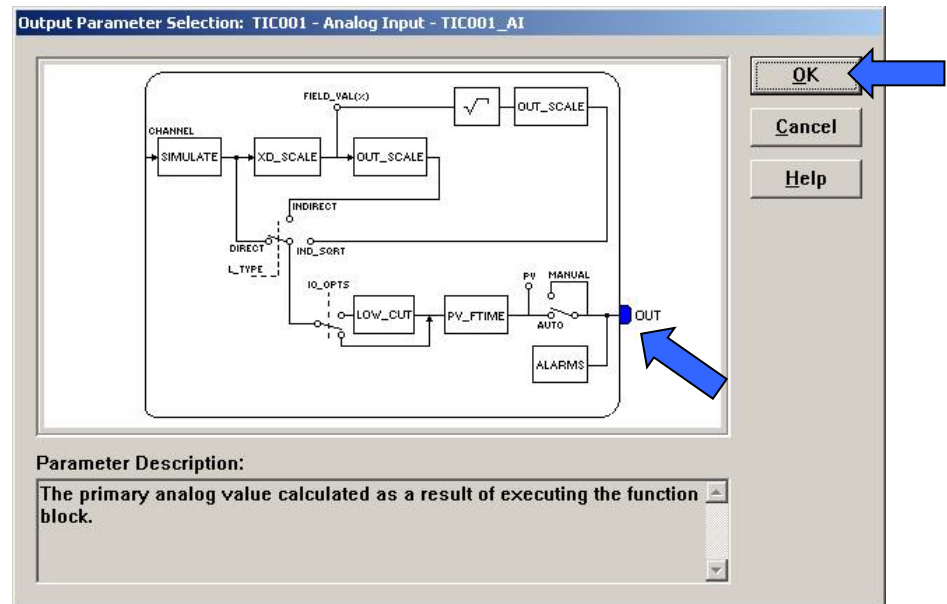


Figure 16. 29 – Linking the function blocks

Move the mouse cursor up to the block that will be linked.

The user also does the fast link procedure just right-clicking the function block.

The links necessary for this strategy are:

Direct Links:

- OUT(TIC001_AI) → IN(TIC001_PID)
- OUT(TIC001_PID) → INFT101_PID)
- OUT(FT101_PID) → CAS_IN(FCV101_AO)
- OUT(FT101_AI) → CAS_IN(FT101_PID)

Back Links:

- BKCAL_OUT(PID_LD302) → CAS_IN(PID_TT302)
- BKCAL_OUT(AO_FI302) → BKCAL_IN(PID_LD302)

After linking the parameters specified above, the strategy window will be as shown in the following figure.

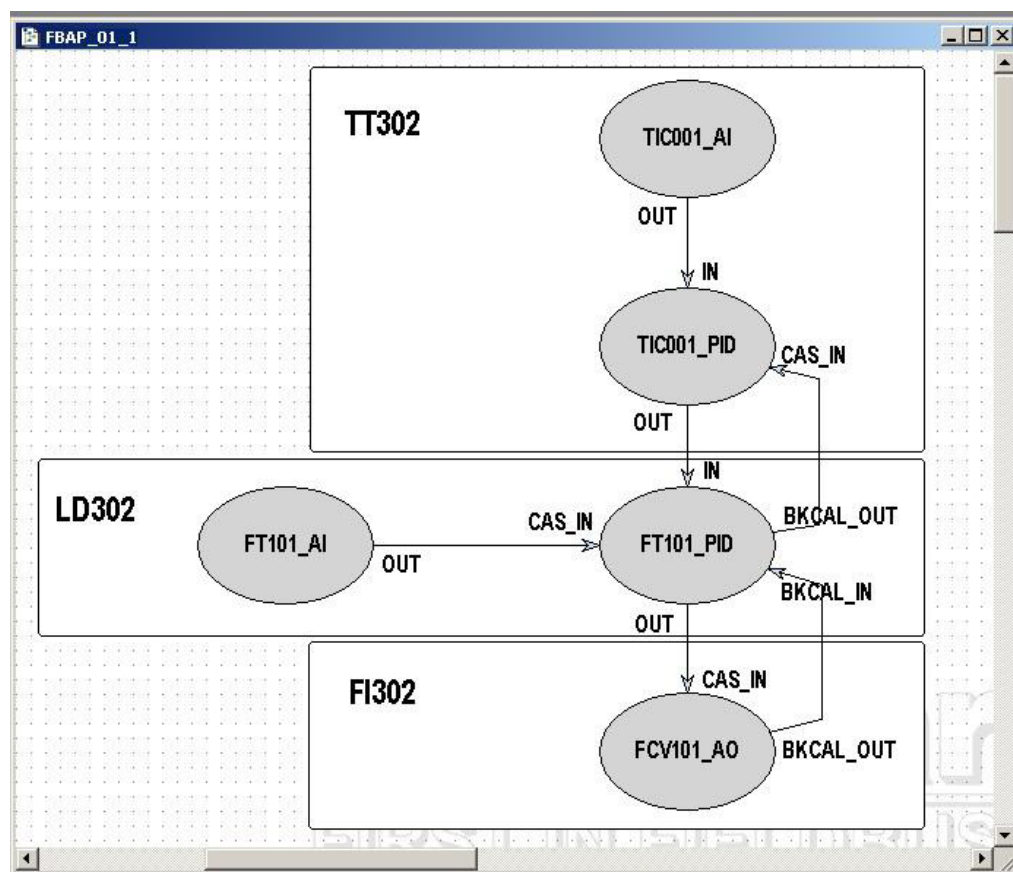


Figure 16. 30 – Links among function blocks

Function Block Characterization

Step 13

The function blocks must be set according to the application for them. So, it is necessary to do the block characterization.

The online and offline modes are possible for the block characterization. In the offline mode, the parameters are set before starting the communication between the devices. The online characterization is executed directly in the devices when the plant is already communicating.

To change the function block parameters, consider the following topics:

1. In the Strategy window

Select the block to characterize. Right-click it, and select the **Off Line Characterization** option, or double-click it. The following figure shows the block that is being done the offline characterization:

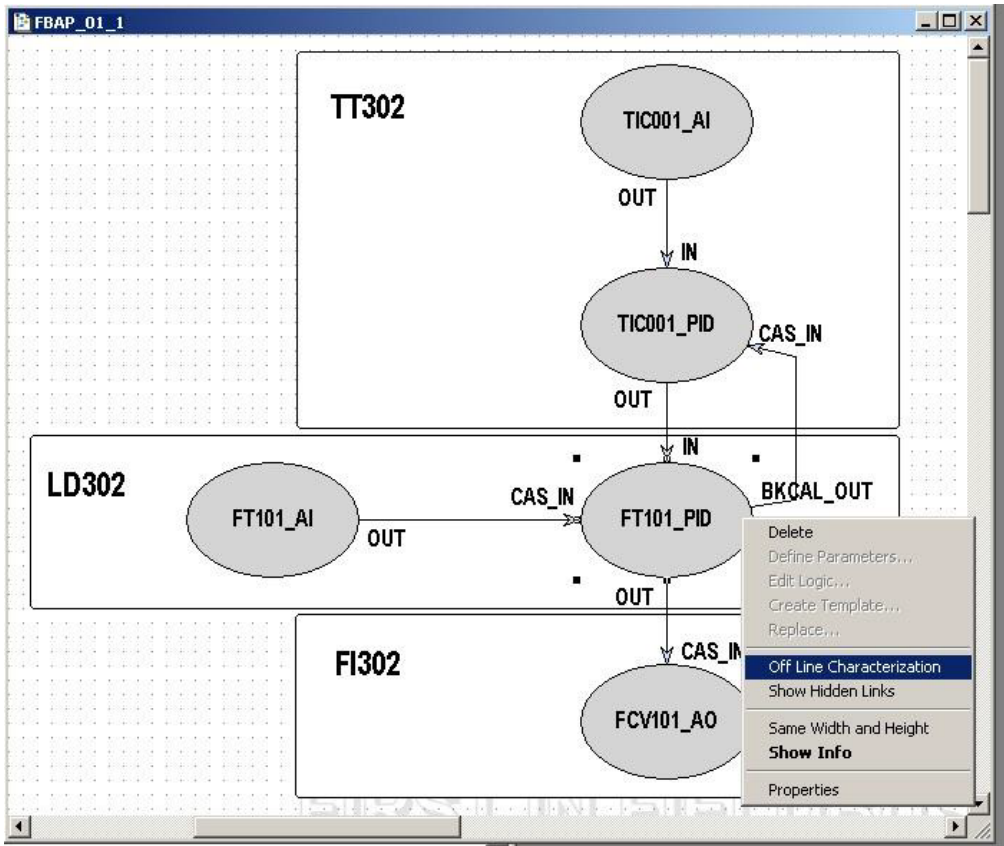


Figure 16. 31 – Offline characterization in the Strategy window

2. In the CANAL_00 window

Another way to do the offline characterization is right-clicking the function block, and then selecting the **Off Line Characterization** option, as shown in the next figure:

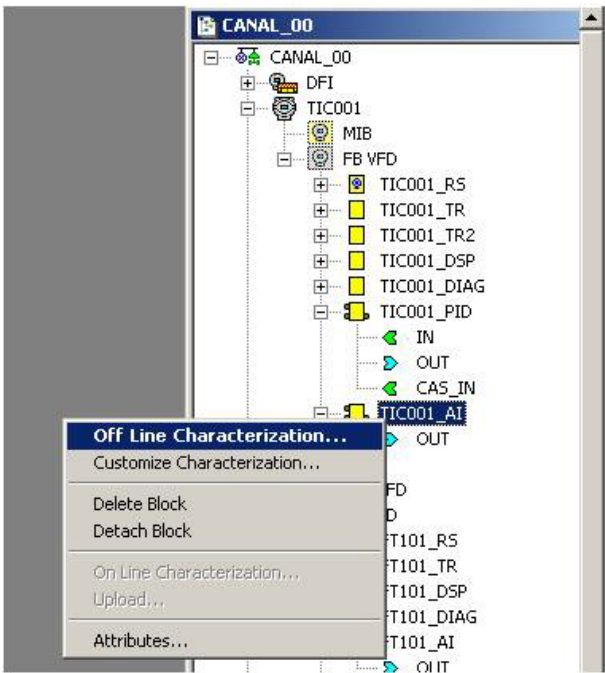


Figure 16. 32 – Offline characterization in the Fieldbus channel window

For both situations, the **Off Line Characterization** dialog box will appear:

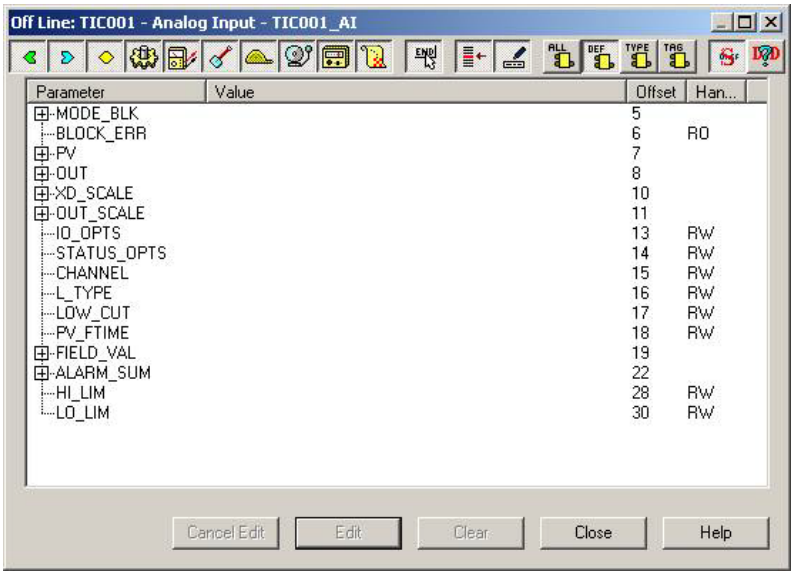


Figure 16. 33 – Function Block Characterization dialog box

Double-click at the right side of the parameter to change it. Another option is click once, and then in the **Edit** button to start editing the parameter value. At the ending, click the **End Edit** button.

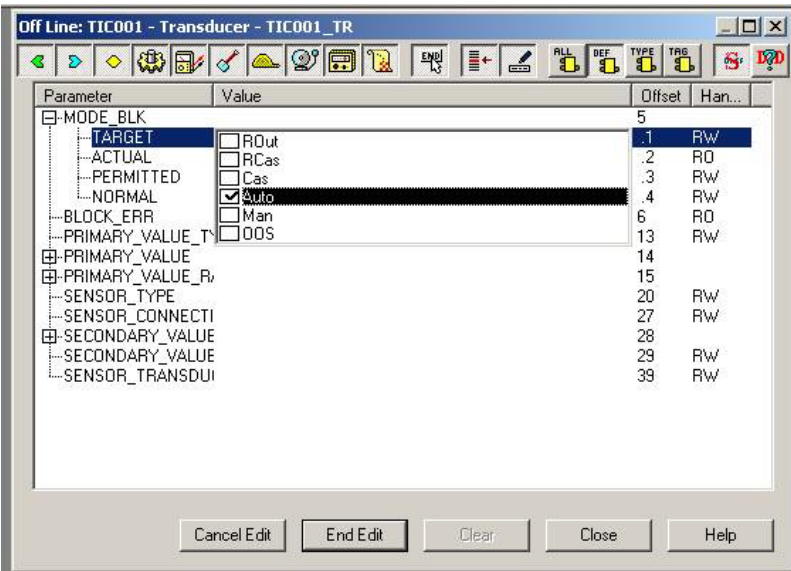


Figure 16. 34 – Editing the parameter in the Function block Characterization box


The list below shows the parameters that must be set for this area:

DEVICE	TAG	BLOCK	PARAMETER
LD302	FT101	TR	MODE_BLK.Target = AUTO
		RS	MODE_BLK.Target = AUTO
		DSP	MODE_BLK.Target = AUTO
			BLOCK_TAG_PARAM_1= FT101_AI
			INDEX_RELATIVE_1 = 8
			MNEMONIC_1 = VAZAO
			ACCESS_1 = MONITORING
			ALPHA_NUM_1 = MNEMONIC
			DISPLAY_REFRESH = UPDATE DISPLAY
			MODE_BLK.Target = AUTO
		AI	XD_SCALE.EU_100 = 100
			XD_SCALE.EU_0 = 0
			XD_SCALE.UNITS_INDEX = inH2O(4°C)
			OUT_SCALE.EU_100 = 100
			OUT_SCALE.EU_0 = 0
			OUT_SCALE.UNITS_INDEX = %
			CHANNEL = 1
			L_TYPE = INDIRECT
		PID_1	MODE_BLK.Target = AUTO
			PV-SCALE.EU_100 = 100
			PV-SCALE.EU_0 = 0
			PV-SCALE.UNITS_INDEX = %
			OUT_SCALE.EU_100 = 100
			OUT_SCALE.EU_0 = 0
			OUT_SCALE.UNITS_INDEX = %
			GAIN = 0.5
			RESET = 1
			RATE = 0

DEVICE	TAG	BLOCK	PARAMETER
TT302	TIC001	TR	MODE_BLK.Target = AUTO
			SENSOR_TYPE = PT100IEC
			SENSOR_CONNECTION = THREE WIRES
			SENSOR_TRANSDUCER_NUMBER = 1
		RS	MODE_BLK.Target = AUTO
		DSP	MODE_BLK.Target = AUTO
			BLOCK_TAG_PARAM_1 = TT100_AI
			INDEX_RELATIVE_1 = 8
			MNEMONIC_1 = TEMP
			ACCESS_1 = MONITORING
			ALPHA_NUM_1 = MNEMONIC
			DISPLAY_REFRESH = UPDATE DISPLAY
			MODE_BLK.Target = AUTO
		AI	XD_SCALE.EU_100 = 500
			XD_SCALE.EU_0 = 0
			XD_SCALE.UNITS_INDEX = °C
			OUT_SCALE.EU_100 = 100
			OUT_SCALE.EU_0 = 0
			OUT_SCALE.UNITS_INDEX = %
			CHANNEL = 1
			L_TYPE = INDIRECT
		PID	MODE_BLK.Target = AUTO
			PV_SCALE.EU_100 = 100
			PV_SCALE.EU_0 = 0
			PV_SCALE.UNITS_INDEX = %
			SP = 50
			GAIN = 0.5
			RESET = 1
			RATE = 0

DEVICE	TAG	BLOCK	PARAMETER
FI302	FCV101	TR	MODE_BLK.Target = AUTO
			TERMINAL_NUMBER = 1
		RS	MODE_BLK.Target = AUTO
			MODE_BLK.Target = AUTO
		DSP	BLOCK_TAG_PARAM_1 = FCV102_AO
			INDEX_RELATIVE_1 = 9
			MNEMONIC_1 = VALVULA
			ACCESS_1 = MONITORING
			ALPHA_NUM_1 = MNEMONIC
			DISPLAY_REFRESH = UPDATE DISPLAY
		AO	MODE_BLK.Target = AUTO
			PV_SCALE.EU_100 = 100
			PV_SCALE.EU_0 = 0
			PV_SCALE.UNITS_INDEX = %
			XD_SCALE.EU_100 = 20
			XD_SCALE.EU_0 = 4
			XD_SCALE.UNITS_INDEX = mA

After the parameter setting, the user can start the equipment communication. It is necessary commissioning the devices in order to attribute the tags, IDs and device addresses properly. If this procedure is not executed, the **Syscon** will detect the noncommissioned device and the download for this device will be aborted. Finishing the equipment commissioning, the download process can start. The download process can be executed, for example, returning to the **Proj_DF51** window,

right-clicking the **Fieldbus Networks** icon, , and selecting the **Download** option. For further details about the available download types, refer to the **Syscon** manual.

Optimizing the Supervision

There are some important steps to be done in the DF51 configuration that can improve the supervision time. Before starting these procedures, a brief description of the System302 architecture is presented to facilitate the understanding of the effects of changes in each parameter.

Take a look in the next figure. The user will be able to follow the data flow from source (field device) to the destination (HMI). Starting in the field device, the data is collected by DFI302 during the **Background** time included in the Fieldbus macrocycle. When using **MVC** (Multiple Variable Container), these data are grouped in an optimized way. The **Supervision Time** controls the rate that a MVC is read from the field device. Each **Update Time**, DFI302 sends the data to DFI OLE Server which updates its database. All the OPC Groups will be updated according to the **OPC Update Rate**.

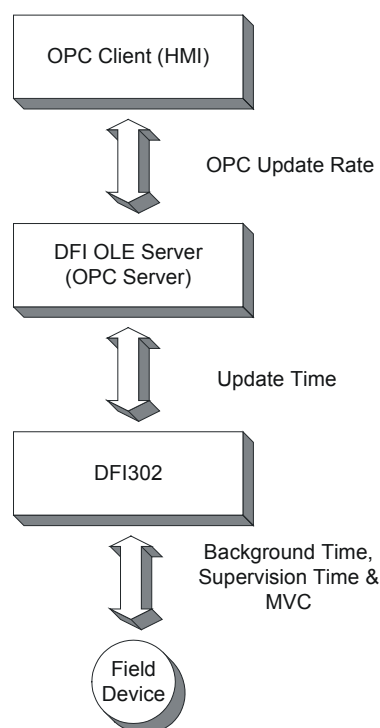


Figure 16. 35 – SYSTEM302 Architecture

The next steps must be configured in order to get a better and optimized time for each system.

Background time

One of the first parameters to be set is Background time (or Background traffic). The **Syscon** calculates the macrocycle according to the number of links in the configuration and allows the user to add a Background time. A minimum value is automatically set by **Syscon**, and must be calculate to have an ideal background to each Fieldbus Network.

The Background time is calculated based on the formula used for the Fieldbus macrocycle. The macrocycle is composed of Operational and Background traffics.

The ideal macrocycle for non-Redundant Systems is:

$$\text{Ideal macrocycle non-Redundant} = ((30 \times \text{NDEV}) + (30 \times \text{NEL})) \times 1.2$$

The ideal macrocycle for Redundant Systems is:

$$\text{Ideal macrocycle Reduntant} = ((60 \times \text{NDEV}) + (30 \times \text{NEL})) \times 1.2$$

Where, NDEV is the Number of Field Devices in the Fieldbus Network
NEL is the Number of External Links (between Field Devices)

After you know the ideal macrocycle, go to Fieldbus Attributes on **Syscon** and adjust the Background Traffic to the new and acceptable value.

IMPORTANT

Once adjusted in every Fieldbus Channel, run a complete configuration download.

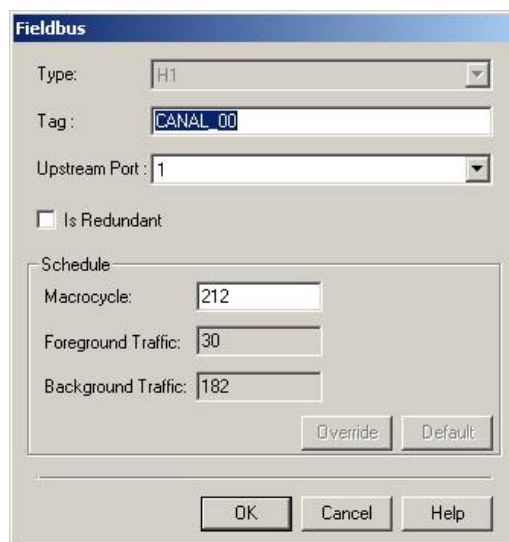
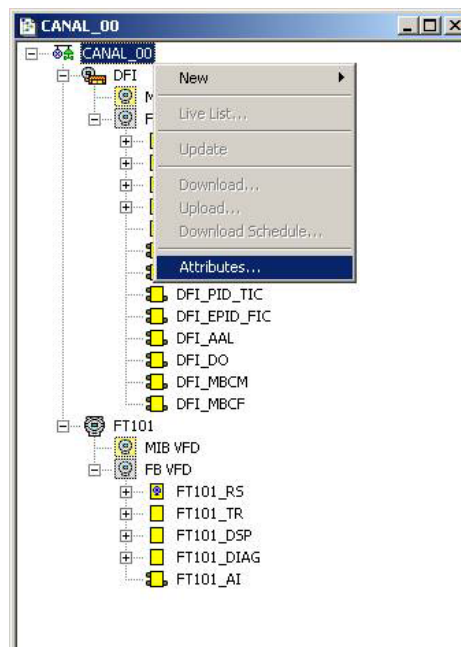


Figure 16. 36 – Adjusting the macrocycle

MVC (Multiple Variable Containers)

Multiple variable containers are a data container that will have all device data. If this parameter is disabled, the data are sent through block views. Each block has four views that give a lot of overhead to the communication.

The MVCs come to optimize this communication sending only one big packet per device instead of four small ones per block. Just set the **MVC_ENABLE** parameter inside the **DF51 Transducer Block** to enable this feature. All changes done on this parameter will take effect only after new Supervision startup.

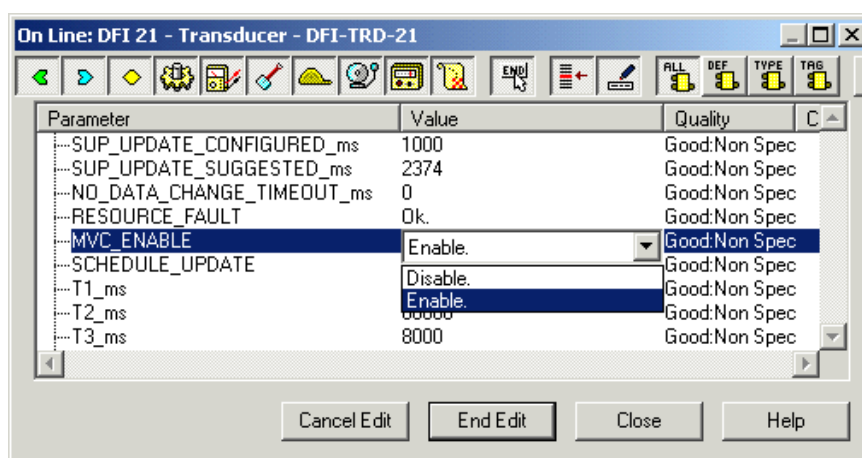


Figure 16. 37 – Setting the MVC_ENABLE parameter

Supervision Time

The Supervision Time is the time required for the DF51 to collect all field device's data and submit to the supervision workstation. Remember that these data are sent through the Background time of the macrocycle.

During the Supervision time, the internal database is refreshed. Therefore, this procedure only makes sense in a system that is already up and running, together with all HMI (Human Machine Interface) software.

The DF51 transducer block has three other parameters that are also used to optimize the supervision in System302.

- Parameter 1: **SUP_UPDATE_CONFIGURED_ms**
- Parameter 2: **SUP_UPDATE_SUGGESTED_ms**

These two parameters define the time that the bridge has to poll the supervision data from the devices. Start setting up the **SUP_UPDATE_CONFIGURED_ms** as 2 times the ideal macrocycle. After 10 minutes approximately, the parameter **SUP_UPDATE_SUGGESTED_ms** will indicate an optimal time and a change may be done again.

- Parameter 3: **NO_DATA_CHANGE_TIMEOUT_ms**

On data change is a mechanism to optimize the data transference between the bridge and the HMI software. With this mechanism the bridge will only send data that has changed.

The HMI has a timeout for the data, i.e., if it does not receive a communication point after a certain period it will indicate lack of communication. Here is the point where **NO_DATA_CHANGE_TIMEOUT_ms** comes in. This parameter defines the timeout to the bridge. If a certain value does not change over that period, it will be sent to the HMI anyway, avoiding the expiration of the HMI timeout.

NOTES

- Good values for the **NO_DATA_CHANGE_TIMEOUT_ms** parameter are between 2500 and 6000, depending on the loaded configuration.
- All the changes done on this parameter takes effect after new Supervision startup.

Update Time

The **UPDATE_TIME** is used by DF51 to refresh the DFI OLE Server database.

Normally only the dynamic data are refreshed. Static data are refreshed each **NO_DATA_CHANGE_TIMEOUT**.

Using **Syscon**, open the Online Characterization for DF51 Transducer Block, and adjust the parameters **UPDATE_TIME** and **NO_DATA_CHANGE_TIMEOUT** to the values. Have in mind that adjusting **UPDATE_TIME** to 200 ms, the DF51 will refresh the data more frequently than the default value (1000 ms) and it will load a little bit more the Ethernet traffic.

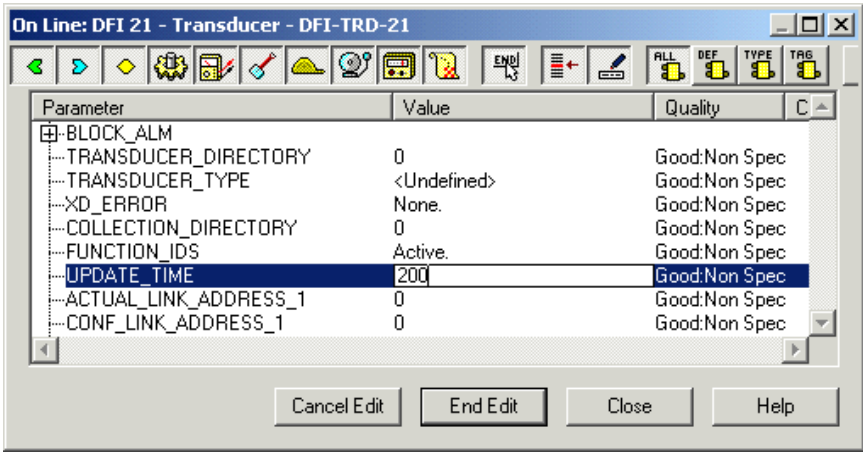


Figure 16. 38 – Setting the UPDATE_TIME parameter

OPC Update Rate

The client (HMI) can specify an “update rate” for each group. This determines the time between when the exception limit is checked. In other words, if the group is set to 1 second, but the data is changing each 500 ms, the client will be advised each 1 second. The update rate is a request from the client and the server will respond with an update rate that is as close as possible to that requested.

Each client has specific ways to configure this rate. Consult the manual for the HMI and do it as necessary.

ADDING REDUNDANCY TO THE DF51 CONTROLLER

Introduction

The main key for fault tolerance and in order to achieve the great system availability is to have redundant devices. DF51 controllers are able to work in a Hot Standby* redundancy mode, which offers redundancy for all its functionalities and databases.

Also there is an option to work with the legacy mode LAS (Link Active Scheduler) redundancy. This chapter will present the characteristics of each mode and the procedures that the user should follow to configure the system in a redundant way. Here follows an overview of the modes.

Hot Standby Redundancy

With the Hot Standby mode full redundancy is achieved, heavily improving the plant availability and safety. This mode offers redundancy for all the DF51 functionalities and databases:

- Gateway: 1 Ethernet port ↔ 4 H1 ports;
- Link Active Scheduler (LAS);
- Controller (running Function blocks);
- Modbus Gateway.

NOTE

A **Link Active Scheduler (LAS)**, equipment is the entity in the FOUNDATION™ fieldbus H1 network responsible for coordinating the communication, i.e. it basically dictates when each device is allowed to publish/subscribe data to/from the FOUNDATION fieldbus H1 network.

That is, the same redundancy capability achieved with the legacy mode “LAS redundancy”, with the Hot Standby mode is achieved too.

This mode suits specially the cases where the DFI302 has function blocks in its configuration. Function blocks on DFI302 can be interesting mainly in two cases:

- Integration to legacy systems through Modbus protocol (using Modbus function blocks);
- Advanced function blocks or strategies (DFI302 execute function blocks with much better performance than the field devices).

The procedures for configuration and maintenance are as simple as for a non-redundant system, saving time to get the system running. Only one configuration download is necessary to configure the redundant pair at the first plant start-up. And in the case of replacement of a failed module none download of configuration or user intervention is necessary. The new module inserted is automatically commissioned, receiving the whole configuration from the module in operation.

Since the system supports the modules placed physically separated (even far from each other), common fault causes are avoided. In other words, with the processor modules placed at different backplanes, or even at different rooms, failures in one of the backplanes or in one of the rooms will not affect the whole system.

* Hot Standby: “Redundancy strategy where the Standby module is powered and synchronized with the Active module, standing ready to assume if necessary”.

NOTES

- The 4TH FOUNDATION fieldbus H1 channel is used as the synchronization path between the modules. Thus, this channel will not be used as a usual FOUNDATION fieldbus H1 channel and should not have devices connected.
- DF51 in Hot Standby mode uses the flat address "0x05" when it publishes. Because third-party devices do not support the flat address, they are not able to establish links with the DF51 in Hot Standby redundancy.
- Hot Standby redundancy is available only for **SYSTEM302** Version 6.1.7 and above.

Link Active Scheduler (LAS) Redundancy

This is a legacy mode of redundancy suitable only for the case where DF51 does not have function blocks in its configuration. That is, in such case the function blocks are on the field devices. This is a completely distributed control philosophy where DF51 perform two main functions:

- Gateway: 1 Ethernet port ↔ 4 H1 ports;
- Link Active Scheduler.

For this scenario, with the LAS redundancy the control, operation, and supervision redundancies are also guaranteed.

Redundant System Architecture

In order to have a true redundant system, not just all the devices must be redundant but also the entire system topology must be thought as redundant. The more elements with redundancy ability the system have, better reliability and availability can be achieved. A typical and simple redundant topology based on DF51 can be seen in figure 17.1.

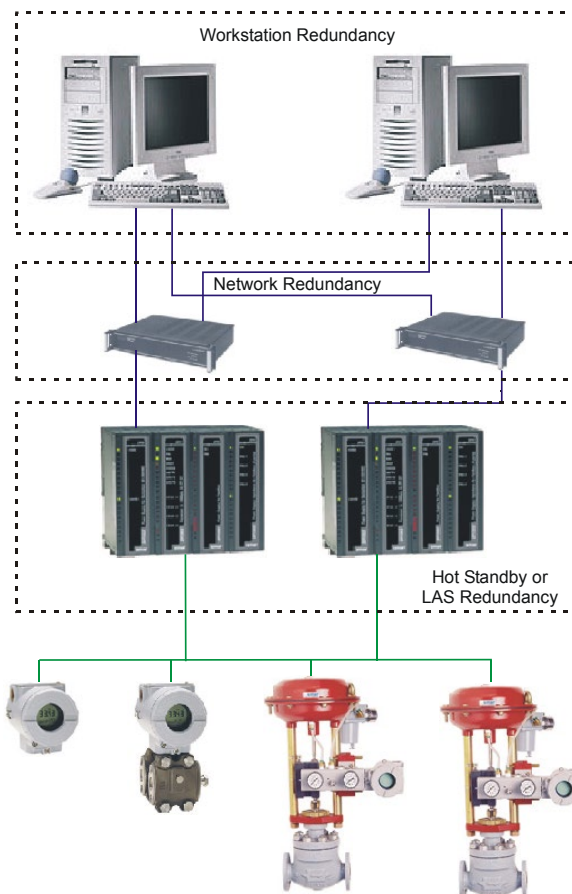


Figure 17.1 – Redundant System Architecture

System Pre-requirements

The requirements listed here apply to both redundancy modes.

The version of firmware for redundant systems has the termination "R". It indicates a firmware suitable for redundant applications. With the redundant firmware, the module initializes by default in Hot Standby mode, in a safety state called "Sync_Idle". The user as will be seen forth can change the redundancy mode later, if necessary.

The **Syscon** configuration should be created as it is usually done for a non-redundant system (in case of questions, please refer to section 3 of this manual). The unique difference (now that redundancy is involved) is that it is necessary to add a transducer function block to the bridge. This transducer will be used then to initialize the redundancy.

In the **Syscon** configuration, the tag for the transducer block can be any, preferentially a meaningful tag concerned to the DF51 tag or to the plant. Be careful to not use tags already in use in the same plant. Further information on **Syscon** operation, can be found in its own manual.

For any of the redundancy modes it is necessary first of all to configure the network redundancy. Refer to the Configuring Servers section for further details.

Configuring Hot Standby Redundancy

In order to enable the Hot Standby redundancy and monitor its status, some parameters available in the DF51 transducer block should be used.

Most redundancy parameters have a suffix. The suffix "L" means Local, or that the parameter brings information of the module that is being monitored directly through the DFI OLE Server. The suffix "R" means Remote, or that the parameter brings information that the Local module knows about the other module through the synchronization path.

Here is presented a functional description of these parameters in order to understand how the Hot Standby redundancy works. For further information on these parameters see also the transducer block description table (Function Blocks manual).

TAG_DESCRIPTION

This parameter shows the DF51 serial number to allow the identification of controller with problem.

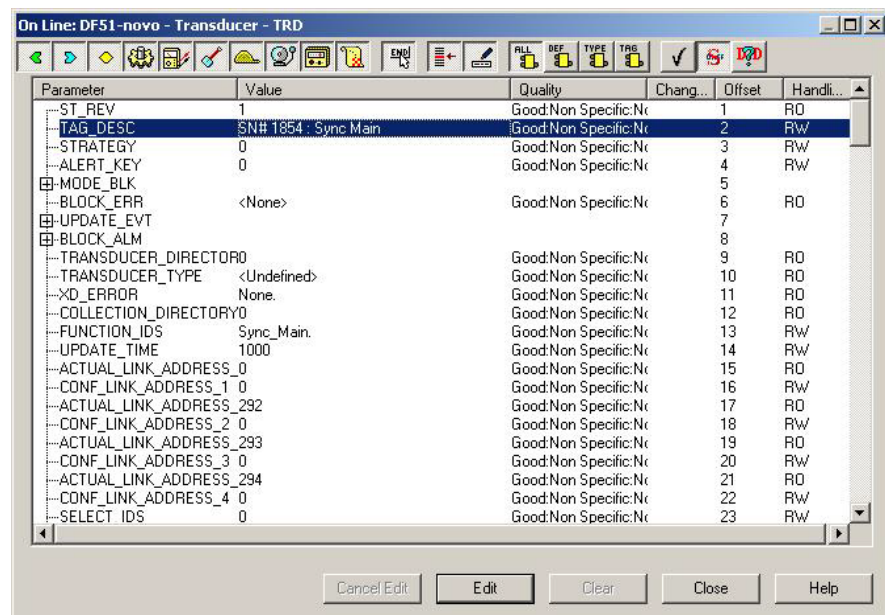


Figure 17.2 – TAG_DESCRIPTION parameter shows the DF51 Serial Number

FUNCTION_IDS

This is the unique parameter to be configured. The user must assign one module to be the **Main** setting **Sync_Main**. After that, through the synchronism path the other one automatically will be set as **Backup**. This designates physically who should be, in other words, the Preferential and the Redundant processor module respectively. This way, **Main** and **Backup** can be understood simply as labels.

RED_ROLE_L / RED_ROLE_R

It reflects the configuration made at **FUNCTION_IDS**, identifying the **Role** of the module, **Sync_Main** or **Sync_Backup**.

RED_STATE_L / RED_STATE_R

Active - runs all the tasks and generates all the data.

Standby – does not run the tasks, but just receives all the data generated by the other one and stands ready to assume, if necessary.

Not Ready – redundancy is not available.

The different failures that may occur in such system, lead it to a switch over, when the **Standby** becomes **Active** and vice-versa in a bumpless way. The possible reasons for a switch over, divided in two types, are as follows:

General Failures

When the whole processor module fails, this comprises:

- Hardware failure
- Power off
- Removal of the processor module from the backplane.

Bad Condition Failures

When one of the processor module interfaces fails:

- Modbus communication failure (hardware or cable; in case of operating as master).
- H1 channel failure (hardware or cable).

The system is capable of checking which one has the best conditions, electing it as the **Active**.

It is certain the recovery of one failure at a time. That is, once a fail has occurred, a second fail will be recovered by redundancy just if the first fail has been fixed. While the failure is not fixed, the system has the redundancy not fully available (in case of **Bad Condition Failures**) or even not available (in case of **General Failures**).

For the case of **General Failures**, as soon as the failed module recovers a healthy state or is replaced, the modules automatically become a redundant pair again. That is, the system automatically recognizes a new inserted module.

RED_SYNC_STATUS_L / RED_SYNC_STATUS_R

This parameter reflects all the possible status of the synchronism between the modules.

SYNC STATUS	DESCRIPTION
Stand Alone	Just one module is operating. If the system has been synchronized at least once, and this value appears, it indicates that the other module had a General Failure.
Synchronizing	The modules are checking their configuration with each other in order to reach the Synchronized status. It can take up to 9 minutes at maximum (while the system waits for the module in "Not Ready" state get its live lists completed).
Updating Remote	Just after the download of the configuration, the module transfers the whole configuration to the other one through the synchronism path.
Maintenance	The module is being configured by the other module through the synchronism path or by the Syscon . If it appears for both "L" and "R" parameters it indicates that none of the modules have been configured.
Synchronized	The modules are in perfect synchronism. The Active continuously updates the Standby databases.

SYNC STATUS	DESCRIPTION
Warning: Role Conflict	If a spare module is connected in the panel with the same Role of that one is already running, this warning is shown. The procedure to fix this conflict is to perform a Factory Init in the spare module.
Warning: Sync Cable Fail	If a failure occurs in the synchronism cable, this warning is shown. The system will not have the redundancy until the synchronism cable is fixed.
Warning: Updating Remote Fail	If a failure occurs in the transfer of configuration from the Active to the Standby, this warning is shown. The procedure is to perform a Factory Init in the module that is not Active and wait until the transfer is completed successfully.

RED_BAD_CONDITIONS_L / RED_BAD_CONDITIONS_R

It can present one or more value (concatenated) as follows:

BIT	BAD CONDITION	DESCRIPTION
0	Modbus	When working as master and if no Modbus slave device answers, it means that Modbus communication is in bad conditions. It can be caused by failures on the communication path or even a failure on the slave.
1	H1-1	Indicates failure on an H1 channel, specifying each channel had the failure.
2	H1-2	
3	H1-3	
4	LiveList	Indicates that the some Live List was not completed.

The desirable and most probable value is **<none>** for both modules (L and R), which assures good conditions for both, and therefore, redundancy fully available. This parameter can have two functions as follows:

A **Bad Condition failure** for the Active module leads the system to a switch over. In this case, this parameter acts as record of the reason of the last switch over.

When a **Bad Condition failure** occurs for the standby module this parameter shows this condition as an alarm. Thus, warning the operator that the standby presents a specific problem, it allows proactive maintenance in order to have redundancy fully available.

RED_MAIN_WDG / RED_BACKUP_WDG

These are watchdogs that indicate the communication status between the HMI and the processor modules. While their values are incrementing within 2 seconds the respective network connections (Main and Backup) are working fine.

As a simple rule, the redundancy is fully available, ONLY if the modules are **Synchronized** and have **<none>** in **BAD_CONDITIONS** parameters (L and R).

The following operations can be performed without process interruption: replacing a module with failure, fixing the system when the H1 cable breaks, firmware update, and adding redundancy to a system in operation.

NOTE
<p>The most new DF51 modules have a LED labeled as "Standby" at the front to indicate the redundancy state of the module.</p> <p>When the LED is "on", it means that the module is in standby. When the LED is "off", the module may be either in Active or Not Ready. If one of the modules is in Standby, the other is surely in Active.</p>


In the next paragraph there are some steps for the Hot Standby Redundancy configuration and maintenance. It is recommended that the steps are all read and understood before are executed.


First time configuration procedure

This is the procedure to configure the system with Hot Standby Redundancy for the first time, at the plant start-up.

1 - With the H1 connector disconnected, execute a Factory Init in both modules in order to grant the default state.

2 - Connect both modules together, through the H1 channels (1 to 4).

3 - Open the desired configuration in the **Syscon** and put it in **On-line mode**. Right-click the **bridge** icon and in the option **Attributes** choose one of the modules listed in the field **Device Id**. The chosen module will be that one to be configured as **Main**. In the main window, right-click the icon  and then select **Export Tags** option on the popup menu.

4 - Even in the bridge icon, right-click the field **FB VFD**, and then choose **Block List**. A new window will open showing all the blocks pre-instantiated in the module. In this window, right-click the **transducer**  performing an **Assign Tag** with the tag that is predicted in the configuration. Close the **Block List** window.

5 - Right-click the **transducer** icon in the **bridge** and choose **On Line Characterization** (see note about operation modification of DF51 in the **SYSTEM302 V7.x** or higher). Set the parameter **FUNCTION_IDS** as **Sync_Main**. Through the synchronism path, the other module automatically will be initialized as **Backup**. After that, both the parameters **RED_SYNC_STATUS** (L and R) will indicate **Maintenance**, which means that neither of the modules was configured yet.

6 - If necessary, perform **Assign Tag** for all the field devices. Wait until the Live Lists of all the channels are complete. So, configure the system through the Active module executing all necessary downloads exactly the same way for a non-redundant DFI302 system.

7 – As soon as downloads are completed successfully, the transducer will show the following phases:

- The Active will transfer the whole configuration to the other module (**RED_SYNC_STATUS_L** as **Updating Remote** and **RED_SYNC_STATUS_R** as **Maintenance**).
- After the configuration is successfully transferred, the modules can take some time to synchronize (**RED_SYNC_STATUS** parameters (L and R) as **Synchronizing**). This is the time necessary to the modules to check the configuration with each other.
- Finally, the modules will synchronize (**RED_SYNC_STATUS** parameters (L and R) as **Synchronized** and **RED_STATE_R** as **Standby**). Once the system is on these conditions, the Active will be constantly updating the Standby.

NOTE

About operation modification of DF51 in the System302 V7.x or higher

Any change of name and number of blocks in DF51 will only be recognized by **SYSTEM302 v7.x** after new startup of OPC Server. This is due to change in the default configuration of **TOPOLOGY_CACHE = ON** parameter in the configuration file **SmarOleServer.ini**.

Firmware version V3.9.5 or higher eliminates the need to restart the OPC server.

Changing the configuration

Just follow the steps 6 and 7 of the section "First time configuration procedure".

Replacing a module with failure

1 - With the H1 connector disconnected, insert the new module in the backplane.

2 - Update the firmware in the new module, if necessary. Perform a Factory Init in the new module in order to grant the default state.

3 - Connect the H1 connector to the new module.

4 - The new module will be automatically recognized by the Active and both will stay in Synchronizing for some time. As soon the system get the **Synchronized** status and **<none>** in the **BAD CONDITIONS** parameters, the redundancy will be fully available and failure simulations can be performed.

Fixing the system when the H1 cable breaks

If a fail occurs in a segment of H1 line such that it affects only one module, the redundancy will cover this fail. But, if the H1 cable is reconnected at once, the noise introduced in the line will cause communication problems for some time. In order to avoid that problem, follow the procedure below.

1 - Put the module affected by H1 cable fail in **Hold**.

2 - Fix up the cable connection.

3 - Perform a Reset in the affected module in order it returns operating. The module will be automatically recognized by the Active and both will stay in Synchronizing for some time. As soon the system get the **Synchronized** status and **<none>** in the parameters **BAD CONDITIONS**, the redundancy will be fully available and failure simulations can be performed.

Firmware update without process interruption

This procedure describes how to update the firmware of both the modules without process interruption.

1 – Be sure the system is in the **Synchronized** status and it has **<none>** in the parameters **BAD CONDITIONS**. So, use **FBTools** to update the firmware of the Active module. At this moment, the other module will take over.

2 - After the firmware update was finished, the modules will start to synchronize with each other, with the Active transferring all the configuration to the other one. Wait for the system get the **Synchronized** status and it has **<none>** in the **BAD_CONDITIONS** parameters.

3 - Use **FBTools** to update the firmware of the Active module. At this moment, the other module will take over.

4 - After the firmware update was finished, the modules will start to synchronize with each other, with the Active transferring all the configuration to the other one. As soon the system get the **Synchronized** status and has **<none>** in the parameters **BAD_CONDITIONS**, the redundancy is fully available again and failure simulations can be performed.

Adding redundancy to a system in operation

If a non redundant system is intended to be redundant in the future, at the plant start-up, the following conditions must be obeyed:

1 - The 4th H1 port should be reserved as synchronization path. That is, this port should not have devices connected to it.

2 - Predict H1 channels cabling considering that a Backup module will be added in the future (the H1 channels of the Main module should be connected in parallel with the respective H1 channels of the Backup module).

3 - Predict that the LAN architecture can be expanded, in order to attend what is described in the Redundant System Architecture.

4 - The single module should use a redundant firmware (a version terminated in R). The **FUNCTION_IDS** parameter should be set as **Sync_Main**. This way the module will work in **Stand Alone** state and will be ready to recognize a new pair inserted at any time.

Obeying these conditions, redundancy can be added at a later time without process interruption. The procedure to add redundancy to the system is just follow the same steps described in the section “Replacing a module with failure”.

Configuring LAS Redundancy

In the next paragraph there are some steps for the configuration and maintenance of this legacy mode. It is recommended that the steps are all read and understood before are executed.

First time configuration procedure

This is the procedure to configure the system with LAS Redundancy for the first time, at the plant startup.

Active Module

1 - With the H1 connector disconnected, execute a Factory Init in both modules to grant the default state.

2 – Connect the H1 connector to the Active module. Keep the Backup module with the H1 connector disconnected for a while.

3 - Open the desired configuration in the **Syscon** and put it in **On-line** mode. Right-click the **bridge** icon and with the option **Attributes** choose the module to be configured as **Active** in the field Device Id.

4 - Even in the bridge icon, right-click the field **FB VFD**, and then click **Block List**. A new window will be opened showing all the blocks preinstantiated in the module. Then, in this window, right-click the **transducer** performing an **Assign Tag** with the tag that is predicted for the Active in the configuration. Close the **Block List** window. In the main menu go to **Export** and click **Tags**.

5 - Right-click the **transducer** icon in the bridge and choose **On Line Characterization** (see note about operation modification of DF51 in the **SYSTEM302** V7.x or higher). Set the **FUNCTION_IDS** parameter as **Active**.

6 - Even in the transducer, set the **SYSTEM_OPERATION** parameter as **Redundant**.

7 - If necessary, perform Assign Tag for all the field devices. Wait until the Live Lists of all the channels are complete. So, configure the system through the Active module executing all necessary downloads exactly the same way for a non-redundant DFI302 system.

NOTE

About operation modification of DF51 in the **SYSTEM302** V7.x or higher

Any change of name and number of blocks in DF51 will only be recognized by **SYSTEM302** v7.x after new startup of OPC Server. This is due to change in the default configuration of **TOPOLOGY_CACHE = ON** parameter in the configuration file **SmarOleServer.ini**. Firmware version V3.9.5 or higher eliminates the need to restart the OPC server.

Backup module

IMPORTANT – before connecting the H1 connector to the Backup module, follow the steps below:

1 - Right-click the **bridge** icon and with the option **Attributes** choose the module to be configured as Backup in the field Device Id.

2 - In the configuration change temporarily the tag of the transducer (Backup must have it different from that one used for Active). In the main menu go to **Export** and click **Tags**.

3 - Even in the bridge icon, right-click the field **FB VFD** and then click **Block List**. A new window will be opened showing all the blocks pre-instantiated in the module. Then, in this window, right-click the **transducer** performing an **Assign Tag** with the tag that is predicted for the Backup in the configuration. Close the **Block List** window.

4 - Right-click the **transducer** icon in the bridge and choose **On Line Characterization**. Set the **FUNCTION_IDS** parameter as **Passive**.

5 – And then, connect the H1 connector to the new module, and after that set the **FUNCTION_IDS** parameter as Backup.

6 - Even in the transducer, set the **SYSTEM_OPERATION** parameter as **Redundant**. Wait until the Live Lists of all the channels are complete.

7 - For each one of the channels used in the configuration right-click the **Fieldbus** icon and choose the option **Download Schedule**.

NOTE

The **SCHEDULE_UPDATE** parameter in the transducer should not be used anymore. Instead of it use the option **Download Schedule** as described in the step above.

Replacing an Active module with failure

If the Active module fails, the Backup module takes over as LAS (Link Active Scheduler).

This is the procedure for the case the Active module must be replaced:

1 - With the H1 connector disconnected, insert the new module in the backplane.

2 - Update the firmware in the new module, if necessary. Perform a **Factory Init** in the new module in order to grant the default state.

IMPORTANT- before connecting the H1 connector to the new module, the user must follow the steps below:

3 - Open the desired configuration in the **Syscon** and put it in On-line mode. Right-click the **bridge** icon and with the option **Attributes** choose the module to be configured as **Active** in the field **Device Id**.

4 - Even in the bridge icon, right-click the field **FB VFD** and then click **Block List**. A new window will be opened showing all the blocks pre-instantiated in the module. Then, in this window, right-click the **transducer** performing an **Assign Tag** with the tag that is predicted for the Active in the configuration. Close the **Block List** window. In the main menu go to **Export** and click **Tags**.

5 - Right-click the **transducer** icon in the bridge and choose **On Line Characterization**. Configure the **FUNCTION_IDS** parameter as **Passive**.

6 - And then, connect the H1 connector to the new module, and after that set the **FUNCTION_IDS** parameter as **Active Not Link Master**.

7 - Even in the transducer, set the **SYSTEM_OPERATION** parameter as **Redundant**. Wait until the Live Lists of all channels are completed.

8 - For each one of the channels used in the configuration right-click the **Fieldbus** icon and choose the option **Download Schedule**.

9 - Change the **FUNCTION_IDS** parameter from Active Not Link Master to **Active**.

Replacing a Backup module with failure

If the Backup module fails the Active module remains as LAS (Link Active Scheduler).

The procedure for the case the Backup module must be replaced is the following:

1 - With the H1 connector disconnected, insert the new module in the backplane.

2 - Update the firmware in the new module, if necessary. Perform a Factory Init in the new module in order to grant the default state.

IMPORTANT - before connecting the H1 connector, the user must follow the steps below:

3 - Right-click the **bridge** icon and with the option **Attributes**, choose the module to be configured as **Backup** in the **Device Id** option.

4 - In the configuration, change, for a while, the transducer tag (the Backup module must have a different tag in comparison with the Active). In the **Syscon** main menu, go to **Export** and click **Tags**.

5 – Even in the **bridge** icon, right-click **FB VFD**, and then click **Block List**. A new window will open showing the pre-instantiated blocks in the module. So, in this window, right-click in the **transducer** to do an **Assign Tag** with the tag that is the Backup in the configuration. Close the **Block List** window.

6 – Right-click in the **bridge** icon of the transducer and choose **On Line Characterization**. Configure the **FUNCTION_IDS** parameter as **Passive**.

7 – And then, connect the H1 connector to the new module, and after that set the **FUNCTION_IDS** parameter as **Backup**.

8 – Even in the transducer, configure the **SYSTEM_OPERATION** parameter as **Redundant**. Wait until the Live Lists of all channels are completed.

9 – For each one of the channels used for the configuration, right-click in the **Fieldbus** icon and choose the **Download Schedule** option.

Placing the system into operation after a general power failure

There is also a procedure to place the modules into operation after both have been turned off. If you turn them on at the same time, there will be many collisions on the H1 network because both modules (Active and Backup) will try to become the LAS at the same time. It will cause a delay for the perfect communication to be established. In order to avoid this problem, turn on first the Active module and wait until it is on line. Then, turn the Backup module on.

Fixing the system when the H1 cable breaks

If a fail occurs in a segment of H1 line such that it affects only one module, the redundancy will cover this fail. But, if the H1 cable is reconnected at once, the noise introduced in the line will cause communication problems for some time.

In order to avoid that problem, follow the procedure below.

1 - Put the module affected by H1 cable fail in Hold.

2 - Fix up the cable connection.

3 - Perform a Reset in the affected module in order it returns operating. The redundancy will be fully available and failure simulations can be performed.

Firmware update without process interruption

This procedure describes how to update the firmware of both the modules without process interruption.

1 - Use **FBTools** to update the firmware of the Active module. At this moment, the other module will take over.

2 - After the firmware update had finished successfully, follow the steps 4 to 9 of “Replacing an Active module with failure”.

3 – Wait around one minute in order the Active module become the LAS again (the Active is always the preferential in this mode of redundancy).

4 - Use **FBTools** to update the firmware of the Backup module.

5 – After the firmware update had finished successfully, follow the steps 1 to 6 of the section “Configuring the system for the first time- Backup Module”.

ADDING LOGIC CONFIGURATION USING CO-PROCESSORS MODULES

NOTE

This section refers to the configuration between the DF51 controller and the DF65 co-processor. Therefore, this feature (adding logic) is also supported by DF62, DF63, DF73, DF75, DF79, DF81, DF89, DF95, and DF97 controllers but using the Flexible Function Block (FFB). See the Adding logic by using flexible function blocks section.

Introduction

As described previously, the DFI302 allows the users use several function blocks that can access input and output modules. However, for some applications, the logic configuration using function blocks is not suitable.

Using DF65 (co-processor module), users can implement a logic configuration through ladder language, and interact with all modules of the DFI302 system.

The figure below shows the system overview:

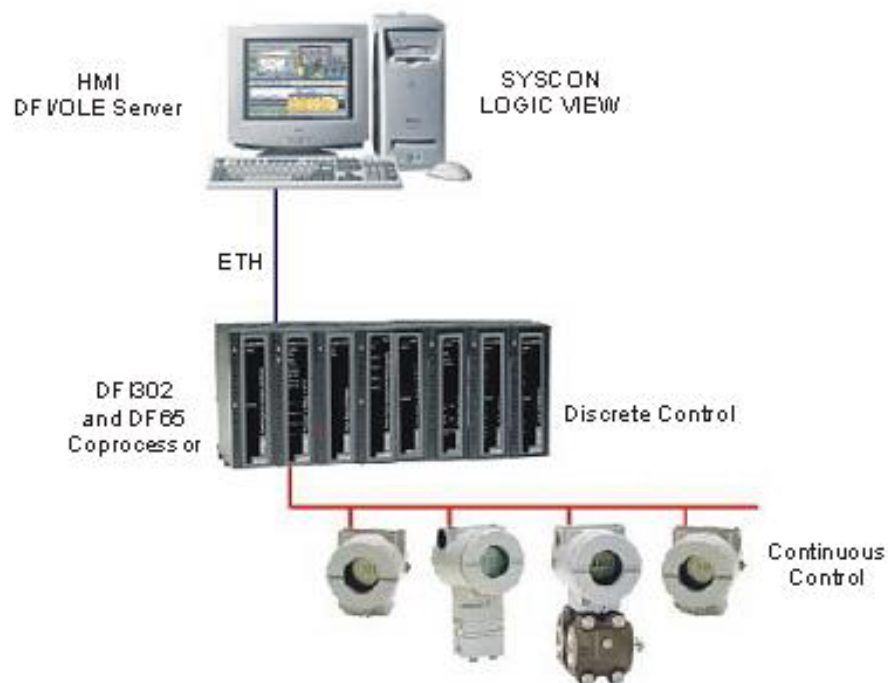


Figure 18. 1 - DFI302 System and Co-processor

DF65 Configuration

The DF65 co-processor uses the **LogicView** software for its configuration. Remember that the DF51 (Processor) must be configured as Master, and the DF65 (co-processor) must be configured as Slave. The physical connection between both is done via DF68 when using a RS-232 port. Another option could be using the DF58 module to allow a RS-485 connection.

In order to set the configuration parameters of the DF65, it is necessary to put the communication switch in the default position, if the user does not know how the DF65 was set or it is the first time you are testing this communication.

Serial Communication Settings

The DF65 has a group of 4 switches located between the communication ports. Using a small screwdriver and make sure that the lower switch is pointing to the left.

The switch in this position sets the DF65 on the Modbus default communication parameters. This means that the Device ID, also called Device Address, is number 1, the baud rate is 9600 bps and the parity bit is even. These parameters can be changed later using the **LogicView** software, but they will only take effect when the Communication Switch is back to the Non-Default position.

Physical Layer and Timeout

In order to visualize the DF51 in the **LogicView** software it is necessary to configure some parameters in advance.

Using **FBTools**, check the IP address of the DF51 to be configured in the **LogicView**. Thus, every configuration performed will be sent to DF65, this means that the DF51 will act as bypass.

Remember that the DF65 must have the same baud rate as DF51 (9600 bps default).

Choose **Comm. Settings** on the **Tools** menu. Select the **Interface** tab, and then **Ethernet (Modbus/TCP)**. Enter the IP address of the DF51 that the **LogicView** will establish communication. See the figure below:

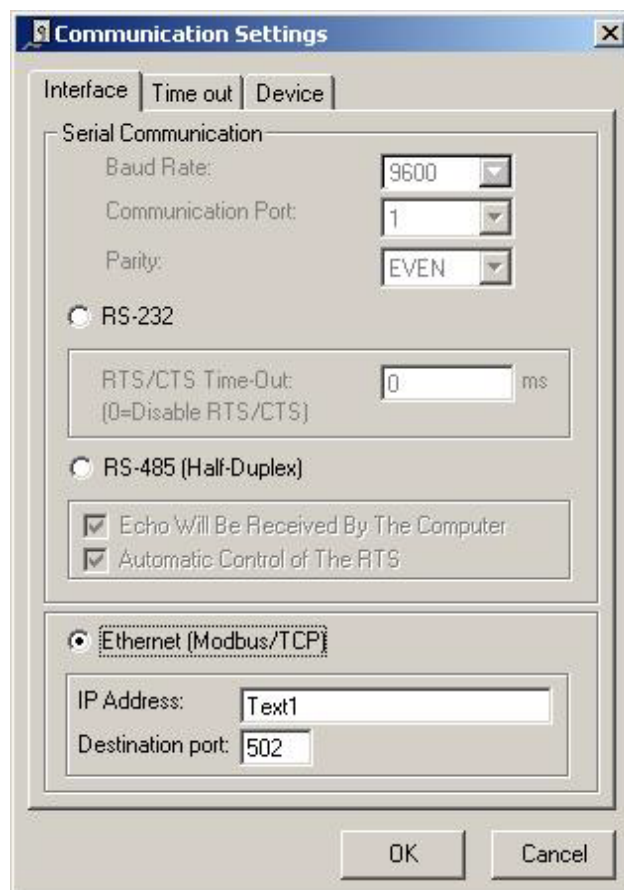


Figure 18. 2 – Setting the IP address of DF51

Next, select the **Time out** tab. Then set the time out and the number of retries for the computer in case of bad communication.

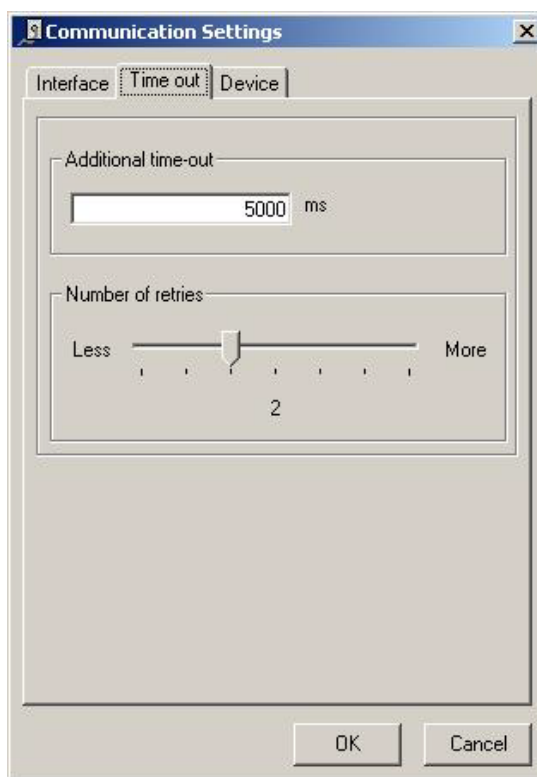



Figure 18.3 –Setting the Time out parameter

Now the user can create the network ladder configuration and send it to the DF65. In case of plant startup see the **LogicView** manual for further details.

Changing DF65 Communication Settings

Open the **DF65 Online** box using the menu **Tools → Online**, or click .

The **LogicView** will connect to the DF65 as soon as the online mode is established. If the **LogicView** cannot detect any DF65 co-processor, it will wait in “time out” state and the **DF65 Online** dialog box will keep opened. This gives the user a chance to modify some parameters to correctly set the communication. If the **LogicView** finds that the DF65 matches the communication parameters already configured, it will add information to the *Device Version*, *Device Release*, *Device Configuration Name* and to the present *Status*.

Remember that the DF65 has a communication switch indicating that the default communication parameters are active. In this case, the address is number 1, the baud rate is 9600 bps and parity is even.

To see the DF51 parameters, select **Default** option under **Communication Parameter**. In this condition it is not possible to make changes in the Serial Port frame. Refer to **LogicView** manual for further details.

Logic Configuration Download

Make sure that all steps described above were performed correctly:

- ✓ Physical connection (cables).
- ✓ Location of DF51 in the subnet via **FBTools**.
- ✓ Correct configuration of the serial communication between DF65 and DF51 (DF65 DIP switches, baud rate, parity, serial communication channel, etc).

- ✓ Correct configuration of the communication between **LogicView** and DF65, that is., through Ethernet using the DF51 as a bridge to bypass Modbus data.

In **LogicView**, create a new Logic Ladder configuration or load an existing one. Send the configuration to the DF65.

Configuring DF51 Modbus Blocks

To establish the communication between the DF51 and the co-processor, it is necessary to add Modbus blocks to control the communication, the supervision and the data exchange between the DF65 and DF51.

To add these Modbus blocks in the **Syscon**, the user must work with two DD versions. To add Modbus blocks the user must select Dev Rev = 02 and DD Rev = 01 and insert them into the **Process Cell**. In order to do this, the user should right-click the **FB VFB** of the DFI302, which is added to **Fieldbus Networks**. Then select "**Attach Block**", or use the "**drag-and-drop**" option to move the blocks.

In the **Syscon**'s Logic Plant (whose tag is Area 1, for example), click the **Area 1** and choose **New Process Cell**. Select the necessary Modbus blocks for the configuration.

For further information about how to insert Modbus blocks, the user should consult the Adding Modbus section. The user must include a Resource block and a MBCF block (Modbus Configuration Block) before starting the configuration of the MBSM supervision block and the MBCM control block.

Supervising DF65 Co-processor data using MBSM Block

After configuring the MBSM block, it is necessary to select the Modbus addresses of the input and output variables to be monitored.

In the **LogicView** program, choose **Modbus Address** on the toolbar, then read and write the Modbus addresses.

In the **Syscon** Logic Plant, create a MBSM block and configure all the necessary parameters by inserting the Modbus addresses of the variables. Thus, the user will be able to monitor the Modbus variables in the **Syscon**.

Data Exchange between DF65 co-processor and DF51 using MBCM block

Add a MBCM block to the Logic Plant. Select the Modbus addresses of the variables to be controlled and monitored.

The MBCM block can be configured to read Modbus variables. Then write them on the DF51. It can also be configured to read Fieldbus variables, and then should be written on the DF65.

Another possibility is to connect two DF65 peer-to-peer communications. See in the figure below:

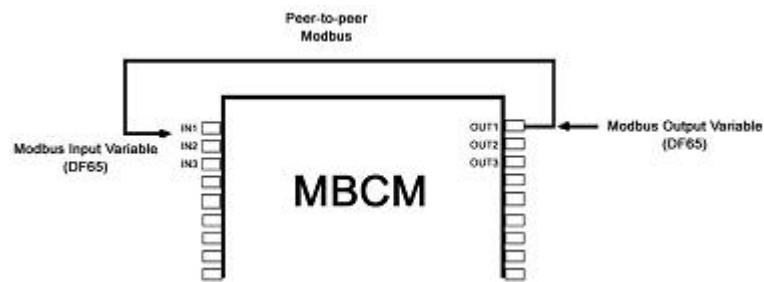


Figure 18. 4 –Setting the MBCM block

This figure shows how to configure MBCM parameter blocks.

Modbus Input variables - transmitters, discrete sensor data, etc, mapped to the Fieldbus world through a MBCM block. The user inserts the Modbus variable address on the configuration parameters of the MBCM block.

Modbus Output variables - data mapped to Modbus environment as, for example, an alarm signal, an interlock, etc, can be sent to the Logic Co-processor System through the block MBCM. The user inserts the Modbus variable address on the configuration parameters of the MBCM block.

Peer-to-Peer - it is possible to read a variable of a module connected to the DF65 and write its value in other module through the MBCM block. The following example describes a simple application of these function features. In order to make the explanation easier, discrete output and input modules are used, although it is possible to use the same procedure to analog variables.

Example of communication between DF51 and DF65 with Ladder Logic

In the example above, there are two modules: a DF20 digital ON/OFF switch input module, and a digital relay output module. Two configurations will be done to implement the communication, supervision and data exchange between DF65 and DF51.

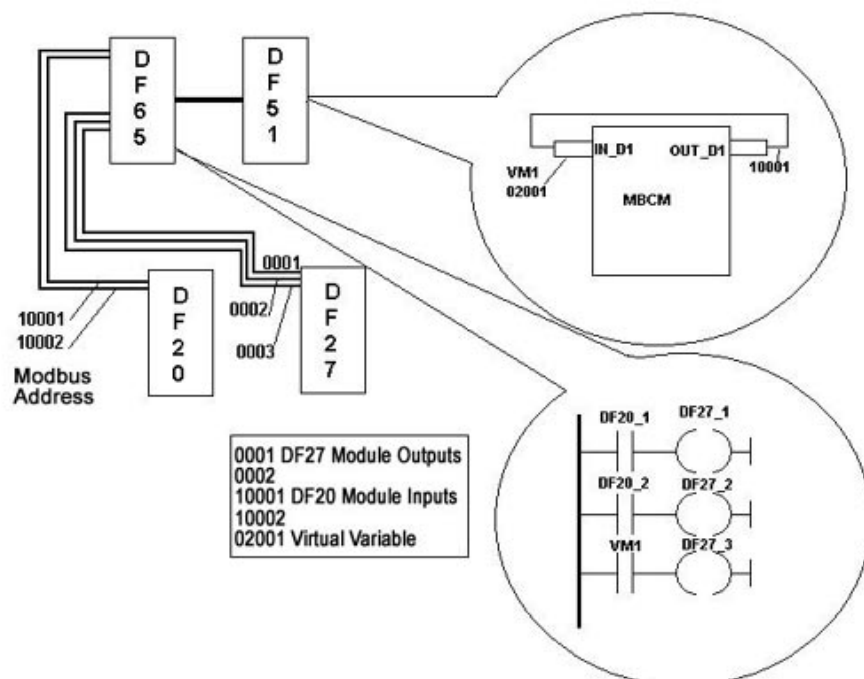


Figure 18. 5 – Communication between DF51 and DF65 with ladder logic

In **LogicView**, begin a new configuration. Add the DF20 and DF27 modules, and a virtual module. Then insert the following simple ladder logic.

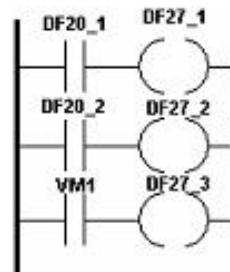


Figure 18. 6 –Ladder logic example

Switches 1 and 2 on module DF20 are connected to the contacts, and the outputs of these contacts are connected to two coils connected to the module DF27 outputs. Similarly, a virtual variable was assigned to a third contact. The Modbus addresses of these variables are:

- DF20_1 → 10001
- DF20_2 → 10002
- DF27_1 → 1
- DF27_2 → 2
- DF27_2 → 3
- VM1 → 02001

In the **Syscon**, create a new configuration. Add Resource, MBCF, MBSM and MBCM blocks. Remember that a Modbus input variable must be inserted in an output parameter of the MBCM block. Thus, the Modbus 10001 address in **LOCATOR_OUT_D1.Modbus_ADDRESS_OF_VALUE**. After, make a copy of the Modbus input variable of the DF20_1. The **LOCATOR_OUT_D1.Modbus_ADDRESS_OF_VALUE** parameter must be 02001. So, the input value of the MBCM block will be written in the address 02001, and in this case is a virtual variable associated to the contact.

To finish the configuration, open the control strategy (Logic Part) in the **Syscon** and connect the IN_D1 input to the OUT_D1 output.

This example uses only discrete variables and modules, however analog input and output variables and modules can also be used, as well as Fieldbus modules connected to Modbus modules and variables. For example, an alarm block output can be associated with the output of a module connected to a DF65. A PID block output can be associated with the output of an analog output module connected to the DF65. Thus, it is possible to split the plant control, so that, while the DF65 performs the discrete control, the DF51 performs the process control.

How to configure the communication and data exchange between the DF65 and the DF51

In LogicView

- ✓ On the **Tools** menu choose **Comm. Settings**, then select **Ethernet Modbus(TCP/IP)** and insert the DF51 IP address in which the DF65 will communicate.
- ✓ Test the communication between **LogicView** and DF65, which is done via Ethernet and the serial connection. In case of failure, check using the **FBTools** if the IP of the DFI address is correct. See if the DF65 communication switches are correct. The fourth switch must be pointing to the left. Check if the cables are connected correctly.
- ✓ In the **LogicView**, create a new configuration or open an existing one. Download the configuration to the DF65.

In Syscon

- ✓ Open the **Syscon**. On **Project File** menu choose **New**, and then choose **Project**. The **Syscon** will open a window to save the configuration. Save it.
- ✓ Right-click the **Area 1** icon and choose **New Process Cell**. Assign a tag to this process cell. Right-click the **Process Cell** and choose **Expand**. On the new window, right-click the **New Control Module** and assign a tag.
- ✓ Right-click the **Control Module** and choose **New Block**. Select Resource and MBCF blocks, and configure them as described in this manual. Add the MBSM and MBCM blocks according to the project requirements. Right-click **Fieldbus Networks** and choose **New Fieldbus**.
- ✓ Right-click **New Fieldbus** and choose **Expand**. Click the **Fieldbus** and choose **New Bridge**. Then, select **Smar** and **DFI302**, making sure that the DD will support the Modbus blocks. Then, right-click **FB VFD** and choose **Attach Block** on the popup menu. Attach all blocks created before and, if necessary, create other Modbus function blocks.
- ✓ Choose **Strategy** by right-clicking **Control Module**. Drag the blocks to the new window that need to have two inputs set. Remember that the Resource, MBCF and MBSM blocks do not need to be included in this strategy.
- ✓ In the main project window, right-click the project icon and choose **Export Tags**. Click **OK**.

- ✓ Right-click the **Fieldbus Network** icon and choose **Comm. Settings**. Check if the server ID is Smar.DFIOLESERVER.0.
- ✓ Click the DFI located in the main project window. Make sure that the Device ID is correct.
- ✓ Download the configuration.
- ✓ In the MBCF block, choose **On Line Characterization** and change the **ON_ APPLY** parameter to **Apply**.
- ✓ The user will be able to monitor the system through the **LogicView** and **Syscon**, simultaneously.

DF65 – Co-processor Module

- DF65** - Co-processor module with 28 Kbytes of non-volatile memory for user configuration and a microcontroller of 15 MHz with real time clock and master of remote I/O system.
- DF65R** - Co-processor module with 23 Kbytes of non-volatile memory for user configuration and a microcontroller of 15 MHz with real time clock and Redundant master of remote I/O system.
- DF65E** - Co-processor module with 52 Kbytes of non-volatile memory for user configuration and a microcontroller of 15 MHz with real time clock and master of remote I/O system.
- DF65ER** - Co-processor module with 44 Kbytes of non-volatile memory for user configuration and a microcontroller of 15 MHz with real time clock and Redundant master of remote I/O system.

	DF65	DF65R	DF65E	DF65ER
Configuration Memory Size	28K	23K	52K	44K
Discrete Points (physical + virtual)	1024	1024	2000	2000
Analog Points	1024	1024	1024	1024

Description

The DF65 is the logical co-processor module for the DFI302 system. It is the module that runs the programmed configuration and interacts with all the other modules of the DF65 System.

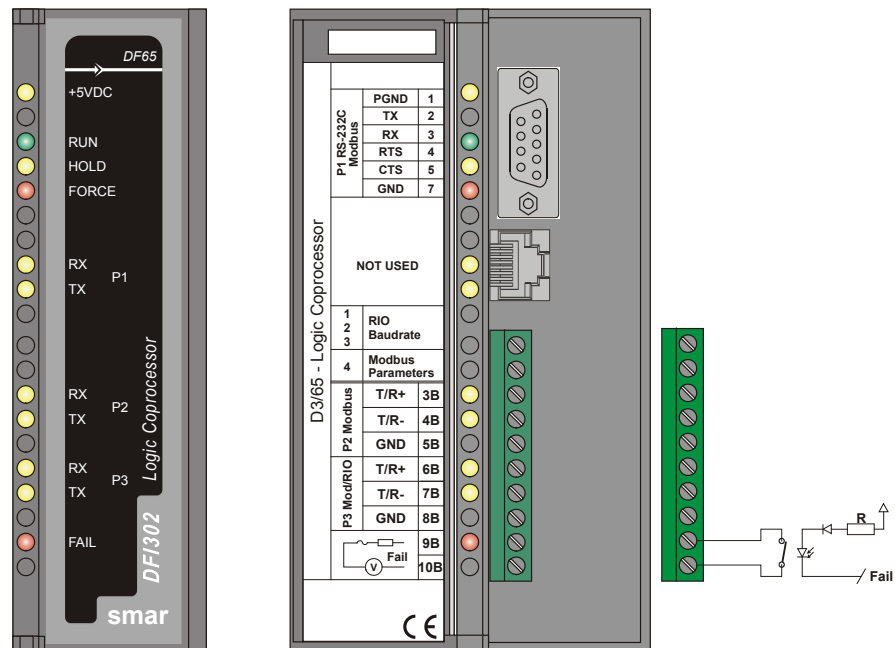


Figure 18. 7 –Co-processor Module

It must be plugged into the second slot of the rack addressed with 0 (zero). This rack number is adjusted through a rotating switch located in the electronic circuit of the rack. The first slot of rack 0 must always be reserved for the power supply module.

The firmware update is done using SMAR's software **DF65Tools**.

NOTE

The DF65 can read all I/O modules that support Module ID since the option "Use Module I/O with ID" in the **LogicView** is enabled. Thus, in systems having modules that do not have these features or systems that combine modules with these features it is necessary to disable flag in the **LogicView**.

In case the connection of the DB9 port is permanent, the DB9-EXT cable must be used which allows the user to close the module front panel.

Technical Specifications

MEMORY

Type	Electrical Erasable Programmable Read Only Memory (EEPROM)
Available Size	DF65 – 28Kbytes, DF65R – 23Kbytes DF65E – 52Kbytes, DF65ER – 44Kbytes

CONFIGURATION/OPTIMIZATION

Software Package	LogicView Version 6.5 or higher
Operating System	Windows NT, Windows 2000 and Windows XP

COMMUNICATION PORTS

Quantity	3
Types	1-EIA-232-C (P1) 2-EIA-485 (multidrop, P2 and P3)
Connectors	Female DB9 to EIA-232-C (P1) Terminal blocks for EIA-485, DF66
Label	See Modules and Accessories
Baud Rate/Address	P1: 9600 bps P2 9600-115200 bps P3 (Modbus): 9600 bps ~115200 bps P3 (RIO): 57600 bps ~230400 bps
Protocol	Modbus RTU (Slave)
Slave Address	2 a 127, Set by User(1 is default) 2 to 121, set by user (1 is the default address)
Maximum number of DF65 per network	31

INTERNAL POWER

Provided by the IMB Bus	5 Vdc @ 320 mA
Total Maximum Dissipation	1.6 W
Power Indicator	Green LED +5 Vdc

FAILURE RELAY

Output Type	Solid state relay, normally close (NC)
Limits	6 W, 30 Vdc Max, 200 mA Max.
Maximum Initial Contact Resistance	<13Ω
Status Indication	Red LED- Fail
Logic Indicator	LED ON (relay closed)
Over Load Protection:	Should be provided externally
Operation Time	5 ms maximum

OTHER LEDS		
RUN		Green LED- indicates program running
HOLD		Yellow LED- indicates program in hold mode
FORCE		Red LED- indicates all inputs and/or outputs are in the force mode.
Rx Tx	P1	Displays Modbus Communication (EIA-232) RX (Yellow) TX(Green)
Rx Tx	P2	Displays Modbus Communication (EIA-485) RX (Yellow) TX(Green)
Rx Tx	P3	Displays Modbus Communication (EIA-485) RX (Yellow) TX(Green)
FAIL		Red LED- Indication of failure

DIMENSIONS AND WEIGHT	
Dimensions (W x H x D)	39.9 x 137.0 x 141.5 mm (1.57 x 5.39 x 5.57 in)
Weight	0.286 kg

CABLES	
One Wire	14 AWG (2 mm ²)
Two Wires	20 AWG (0.5 mm ²)

NOTES
To increase service of life of the relays and to protect the modules from reverse voltage damages, externally connect a clamping diode in parallel with each inductive DC load or externally connect an RC snubber circuit in parallel with each inductive AC load.

Communication Channels

The DF65 has three communication channels that supply the user with 3 independent channels (ports) identified as P1 (EIA-232), P2 (EIA-485) and P3 (EIA-485).

The three channels may be used at the same time with the following features:

- P1 (EIA-232) is for programming and monitoring, point-to-point short connections.
- P2 (EIA-485) is for programming and monitoring, point-to-point or multidrop in long industrial connections.
- P3 can mimic P2 or act as a Master channel for a DF66 connection DF66 interface module (DF66). A rotary switch on the co-processor module card can set the P3 channel behavior.
- Any of the ports can be connected to the Smar ENET-700 or ENET-710 (Ethernet-Serial) converter module.

Due to communication requirements, the DF65 has a Modbus ID that is unique and a different baud rates for P1 and P2/P3. The Modbus baud rates are set through the software **LogicView**.

Restrictions:

- Only one channel can be used at the same time to monitor the network through the **LogicView**. Other channels will be locked after the first one begins.
- All channels are referred with the same address.
- Channels P2 and P3 shares the same baud rate. P1 has a dedicated one.

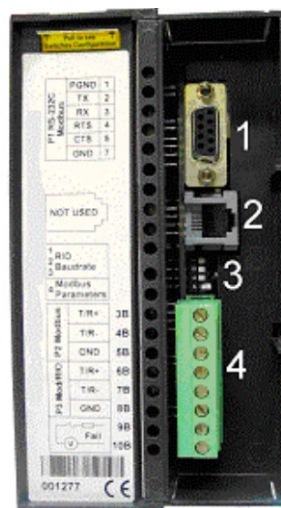


Figure 18.8 – DF65 Communication Channels

In the previous figure is shown:

- 1- DF65 P1 port. It is an EIA-232 channel;
- 2- SSIO channel used by the redundant DF65. See "DF65 Redundant Module" for further information.
- 3- DIP switches used to change the co-processor communication parameters.
- 4- Ports P2 and P3 of the DF65. These are EIA-485 channels.

There are three serial communication ports on the co-processor: one EIA-232 port (P1) and two EIA-485 ports (P2 and P3). The user should set to any of these ports: baud rate, parity and other specific parameters.

Port P1

Baud rate (9600 bps) for DF65.
Baud rate (9600 bps) for DF65R.
Parity (Odd or Even)

RTS/CTS Timeout

CTS: Signal that indicates the device is ready for transmission.

RTS: Signal to request data transmission

The PC requests a connection with the co-processor. The DF65 receives this request and processes it. Next, the DF65 sends the RTS signal and it waits for the CTS signal during the time interval set on the RTS/CTS Timeout parameter.

OFF Duty

Available time for communication when the co-processor is not running a ladder diagram. The biggest value set for Off Duty, the biggest time available for communication.

Time Delay

The PC sends a frame to the DF65, i.e., makes a communication request. The DF65 waits the amount of time set on Time Delay to process the frame request and send the answer.

NOTES

For **better performance** of the system, it is recommended:

- Off duty must be set as 20% of the Ladder execution cycle.
- **Time Delay** depends on the Workstation's processor. If the processor is higher than a Pentium MMX 233 MHz, set the Time Delay to 5 ms. Otherwise the Time Delay must be set with the default value.
- When the DIP switch 4 is in the default position or if the value set for Off duty is equal to zero, then the real value set for Off duty will be 20% of the cycle (modules scan plus Ladder execution).

Device Communication Baud Rate and Device Address

The co-processor module has a key where the user may select default communication parameters (DCP, Switch 4 ON) or programmed communication parameters (PCP, Switch 4 OFF). Default parameters are address 1 and baud rate 9600 bits/s.

In the PCP position the user can select a new address and/or baud rate, using **LogicView**. The new adjustment of parameters will be accepted only after the key being moved to the PCP position. In the PCP position the user is also able to change the communication parameters. In this case they will be accepted immediately after they have been sent.

The default values apply to three channels: P1, P2 and P3. The user can set the DIP switches using a key as shown in the next the figure:

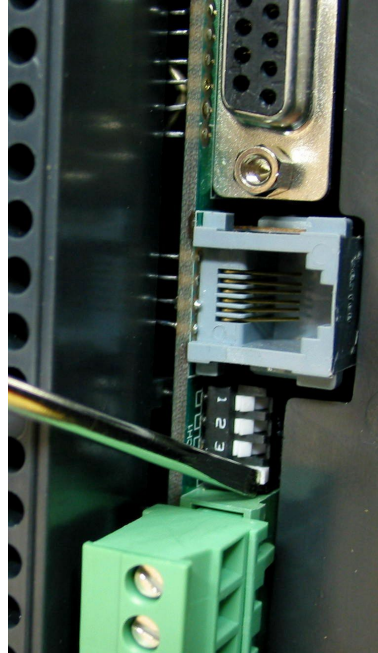


Figure 18. 9 – Setting the Communication Parameters of the DF65

Operation Modes

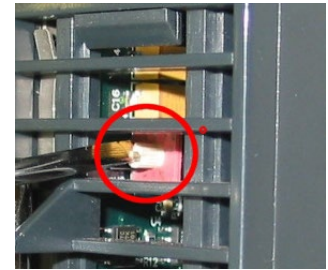
The DF65 may act as:

- A co-processor with three Modbus/RTU channels
- Master co-processor in a system with DF66, where 2 channels work as Modbus/RTU and the remaining one (P3) as remote master channel.

In order to change the operation mode of the DF65 the user needs to change the Rotary Switch position. This rotary key is located underneath the DF65 module. See the next figure to locate it.



Figure 18.10 – Locating the DF65 Rotary Switch



Detail: The Rotary Switch

DF65 with Three Modbus RTU Channels

When the DF65 is used as a regular co-processor, it means that no DF66 module is being used. Also, either P3 as P1 and P2 may act as Modbus/RTU channels. Note that none of them can act as a Modbus master.

P1, P2 and P3 are independent Modbus/RTU slave channels. They can be used at the same time. P1 is recommended for point-to-point connections whereas P2 and P3 may be used in two different networks (master and slave) to increase accessibility of the co-processor through two computers (masters) or in a redundant architecture where the user can alternate between the channels in a failure situation.

Settings:

- The rotary key must be in the position 8;
- One Modbus ID set for P1, P2 and P3 (set on **LogicView**);
- The P1, P2 and P3 baud rates are set on **LogicView**.

DF65 Master in a DF66 System

The DF65 can be also a master in a DF66 system. This means that the DF65 acts as the main processing unit acquiring data from the DF66 modules in its own IMB and also from any DF66 connected to it.

In this case P3 is completely dedicated to DF66 communication data. P1 and P2 are slave Modbus/RTU independent channels. P1 is used for point-to-point applications whereas P2 may be used in a multidrop network.

Settings:

- The rotary Switch must be in the position 0;
- One ID Modbus for P1 and P2 set on **LogicView**;
- P1 and P2 baud rates are set on **LogicView**;
- P3, the remote channel, has its own dedicated baud rate set through the frontal DIP switch.

Factory Init

The user can carry out a procedure to make the DF65 to assume its factory configuration. This procedure is called factory init.

To do it:

1. Remove the DF65 from the power supply;
2. Put the Rotary Switch on the position 7;
3. Put the DIP Switch 4 in the default position;
4. Connect the DF65 to the power supply and wait until the HOLD LED starts to blink;
5. Remove the DF65 from the power supply;
6. Set the rotary switch.

NOTE

After the factory init procedure the user must set the rotary switch when a new download is done.

DF66 – Remote I/O Communication Interface

DF66 – 28 kbytes of non-volatile configuration memory and a microcontroller of 15 MHz with DF66 (Slave)

DF66E – 52 kbytes of non-volatile configuration memory and a microcontroller of 15 MHz with DF66 (Slave)

Description

DF66 modules are located close to the field devices and allow a flexible architecture to the system. DF66 units use the same I/O modules designed for the regular DF65 system combined with the DF66 module.

The DF66 module must be used together with DF65 and DF65R.

The DF66E module must be used together with DF65E and DF65ER.

NOTE

The DF65 firmware update is done by the SMAR **DF65Tools** software.

Adding a DF66 unit

To add a DF66 system, click the **Go to I/O module page**, and then click the **Add/Delete** DF66 system. As shown in the following figure.

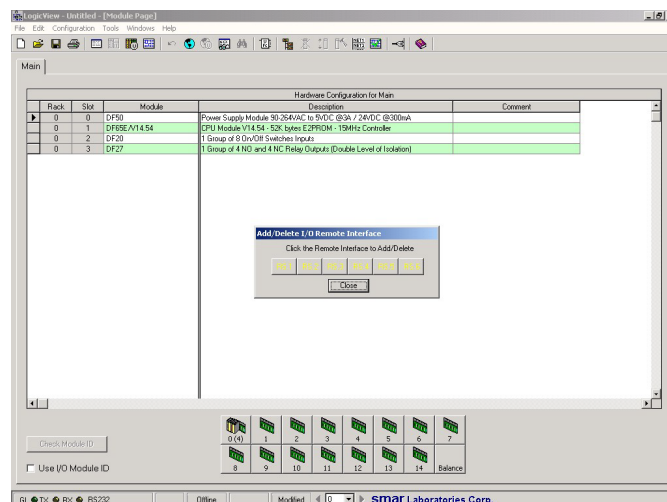


Figure 18. 11 – Adding a remote system

Choose a DF66 system, and click the corresponding button.

ATTENTION

The address and baud rate of the DF66 interface module must be set. Interface and Power Supply modules for DF66 are automatically set.

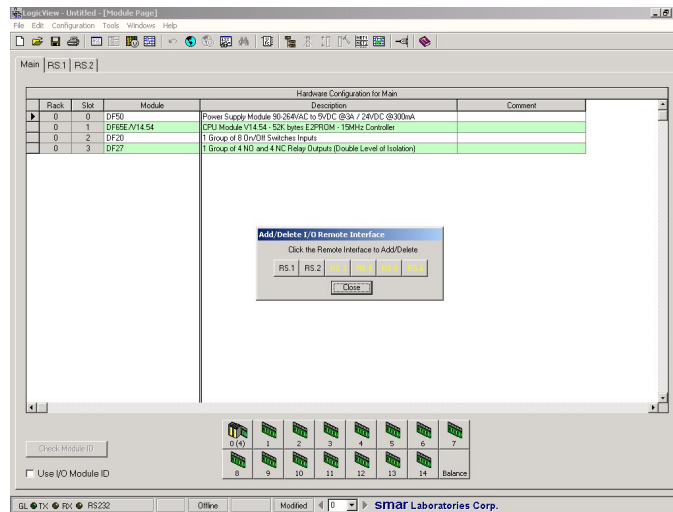


Figure 18. 12 – Adding or removing a remote system

Remote I/O Architecture

The DF66 System is basically composed of a Master Unit and up to 6 Slave Units. Master Unit and Slave Units are connected through a multidrop cable that can reach a total length of 4000 ft. The proper cable length and baud rate depend on the level of noise in the application environment.

The available Rack/Slots will limit the total number of modules per system and the maximum number of Discrete and Analog Points that the DF65 will be able to work. Each DF66 needs, at least, one power supply. A DF65/ DF66 system structure is shown as follows:

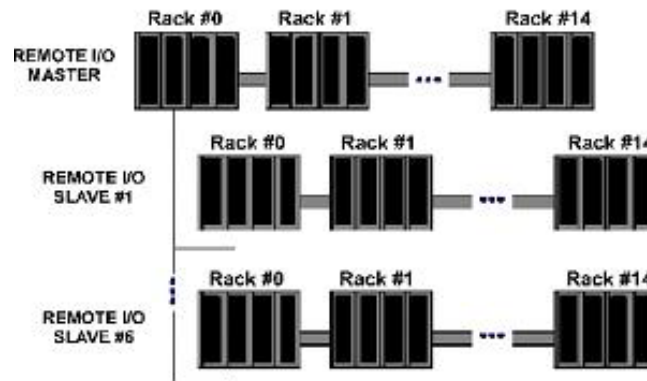


Figure 18. 13 – DF66 Architecture

Baud rate and Address Setting

Baud Rate Settings

Each DF66 Interface Module (Master or Slave) has a DIP switch to set the baud rate (communication speed). The DIP switch is located in the frontal panel of the module and can be accessed with a small screwdriver.

Make sure to disconnect the module while setting the switch.

Also note that both the Master and Slave I/O interface must be configured with the same baud rate.

Setting Address of the DF66 Interface Module

There is also a dedicated rotary switch at the bottom of the slave module to set the slave device address. Each remote unit connected to the master unit must have a unique address. The available addresses are: 1, 2, 3, 4, 5 and 6.

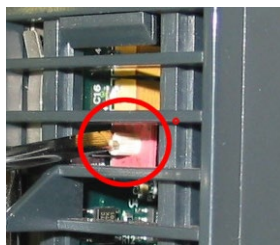


Figure 18. 14 – Setting Address of the DF66 Interface Module

DF65R/DF65ER – Redundant Co-processor Module

Introduction

The redundancy with the DF65 is based on a hot-standby mechanism. Therefore, just one co-processor runs the ladder logic as well as the communication with HMI at a specific moment. The active co-processor scans the DF66 module while the passive co-processor monitors the status of the active co-processor through another port.

However, there is communication between the co-processors in order to keep the synchronization of the configuration and also provide fresh dynamic data to the passive co-processor. For example: the Modbus variables.

The algorithm responsible for choosing the active co-processor tries to minimize the number of control switches; therefore, if a co-processor runs as active, it remains in this state, unless something happens to put this co-processor in bad conditions.

Terminology and Initial Descriptions

Main and Backup Co-processors

The redundancy status depends on a lot of items as: configuration of the co-processor, communication status through SSIO, communications with the DF66 modules DF66. This status defines whether the co-processor will run the ladder logic as well as scanning the RIOs.

The co-processor can assume two roles: active and passive. The active co-processor runs the ladder logic and scans the Modbus variables. The passive co-processor periodically checks the active co-processor in order to check if it is necessary to take control.

The co-processor's configuration is set through the rotary switch in the co-processor module:

- **Rotary Switch in the position 0 (Main):** When both co-processors are powered-up at the same time and, the algorithm checks the rotary switch position (of both co-processors) to choose which co-processor is active and which one is the passive.

- **Rotary Switch in the position 9 (Backup):** The co-processor set as backup transfers the control to the other co-processor its current status is both passive or both active. This configuration is used only when the co-processors are simultaneously powered and when both are passive or active at the same time before the last power-down.

Configuration Transfer

During the power up procedure, the configuration of the active co-processor is passed to the passive co-processor.

This transfer is always done from active to passive and never in the contrary direction and it uses the Synchronous Serial I/O (SSIO) Port of the Co-processor. This synchronous serial communication channel is used exclusively to transfer the configuration.

During this process two new terms need to be defined:

- **Sender:** This is the co-processor responsible for the communication. The sender co-processor handles the configuration transfer between the co-processors.
- **Addresser:** This is the co-processor that receives the configuration.

Configuration synchronism between the Co-processors

This is accomplished in two main phases:

Phase 1- Configuration Transfer

When a sequential power up procedure takes place, the first co-processor powered-up becomes the active co-processor and it transfers the configuration to the other one.

Phase 2- Updating Transfer

After the Configuration Transfer Phase, it is necessary to transfer only the dynamic variables and the settings that can be done without configuration download.

ATTENTION

This refers to Modbus variables, configuration of switches in the co-processor module and communication settings.

Rules to Select the Sender and Addresser

When the co-processor detects the presence of other co-processor through the SSIO, they exchange some status information to decide which will be the sender of configuration.

Three main cases can occur:

#	RULE	SCENARIO
0	If configuration name, date and time in both co-processors are the same and Good-Config, the rules to select the active co-processor will be applied, and the active co-processor will be also the Sender.	Power cycle of configured system
1	If configurations mismatch and a sequential power up happened and the first co-processor has a Good-Config, then the first co-processor to power up will be the sender.	Replace of co-processor module
2	All other cases (configuration mismatch and simultaneous power up OR one or both with bad-config): co-processors do not enter transfer status.	First power up of system after a firmware download or abnormal cases.

The SSIO port Diagram

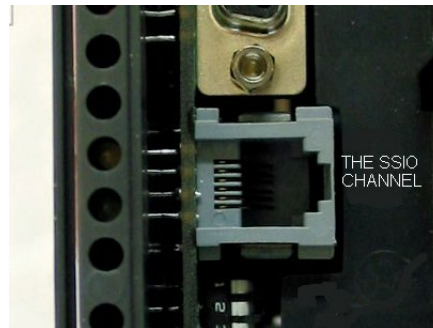


Figure 18. 15 – The SSIO channel

In order to transfer the configuration from the active co-processor to the passive it is necessary to use a special cable to connect the two co-processors. The cable part number is DF76.

This port cable is a 4 wires-cable that implements a full-duplex channel whose baud rate is 1.875 Mbps.

Architecture

Consider the figure below:

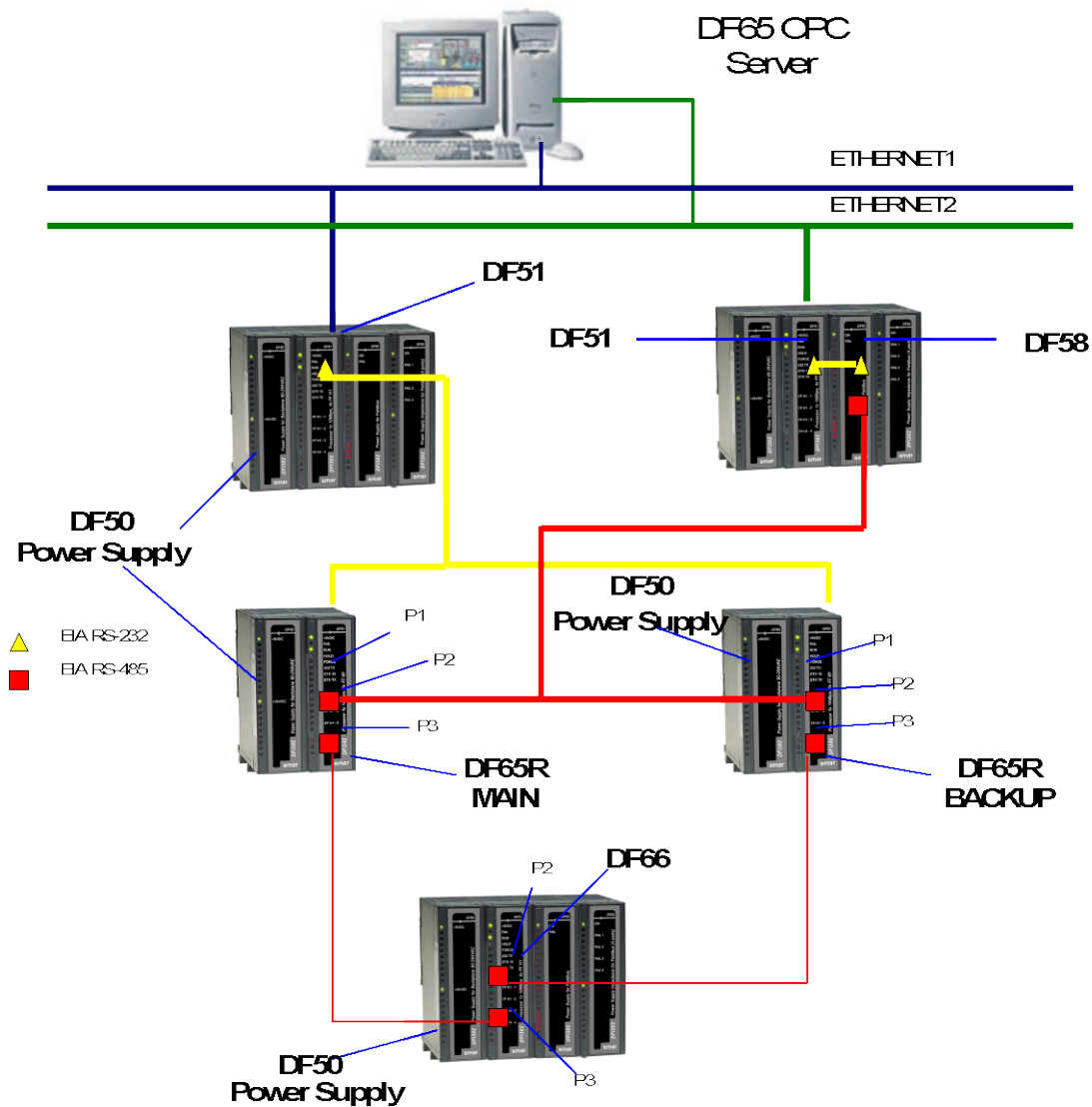


Figure 18. 16 – Redundant Architecture Example

The above architecture shows an illustrative example of the hardware used in the DF65R redundancy.

- Two Ethernet paths;
- Two DF50 modules allow complete redundancy;
- Two co-processors with main and backup roles;
- A DF66 module scans the Modbus variables;
- The DF76 cable connects the two co-processors.

NOTE

The I/O modules are only included in the racks where the DF66 module is present.

Power up Procedures

Breakers

- One breaker to the main co-processor.
- One breaker to the backup co-processor.
- One breaker to the DF66 modules.

Power up sequence

Power up sequence refers to the order that each co-processor is turned on. There is a specific breaker configuration.

1. First, the breaker of the power supplies of the DF66 modules is turned on.
2. Next, each one of the two breakers of each co-processor is turned on.
3. Before the co-processors start searching for another co-processor the state of both becomes passive.
4. After powering up a co-processor and if it does not find an active co-processor, then, during a two seconds interval it will search for another co-processor.
5. If it does not find another active co-processor it becomes the active co-processor.

I - The algorithm checks which co-processor was active in the last power down. The active co-processor in the last power down becomes the current active co-processor. The co-processor passive in the last power down becomes the current passive.

II - If both co-processors were passive or active simultaneously in the last power down, the algorithm checks the rotary keys to define which co-processor will be active. If the rotary key is in the main position (0), then the co-processor is considered active. If the rotary is in the backup position, then the algorithm considers this co-processor as passive.

For further explanation, please check the flowchart in the next page.

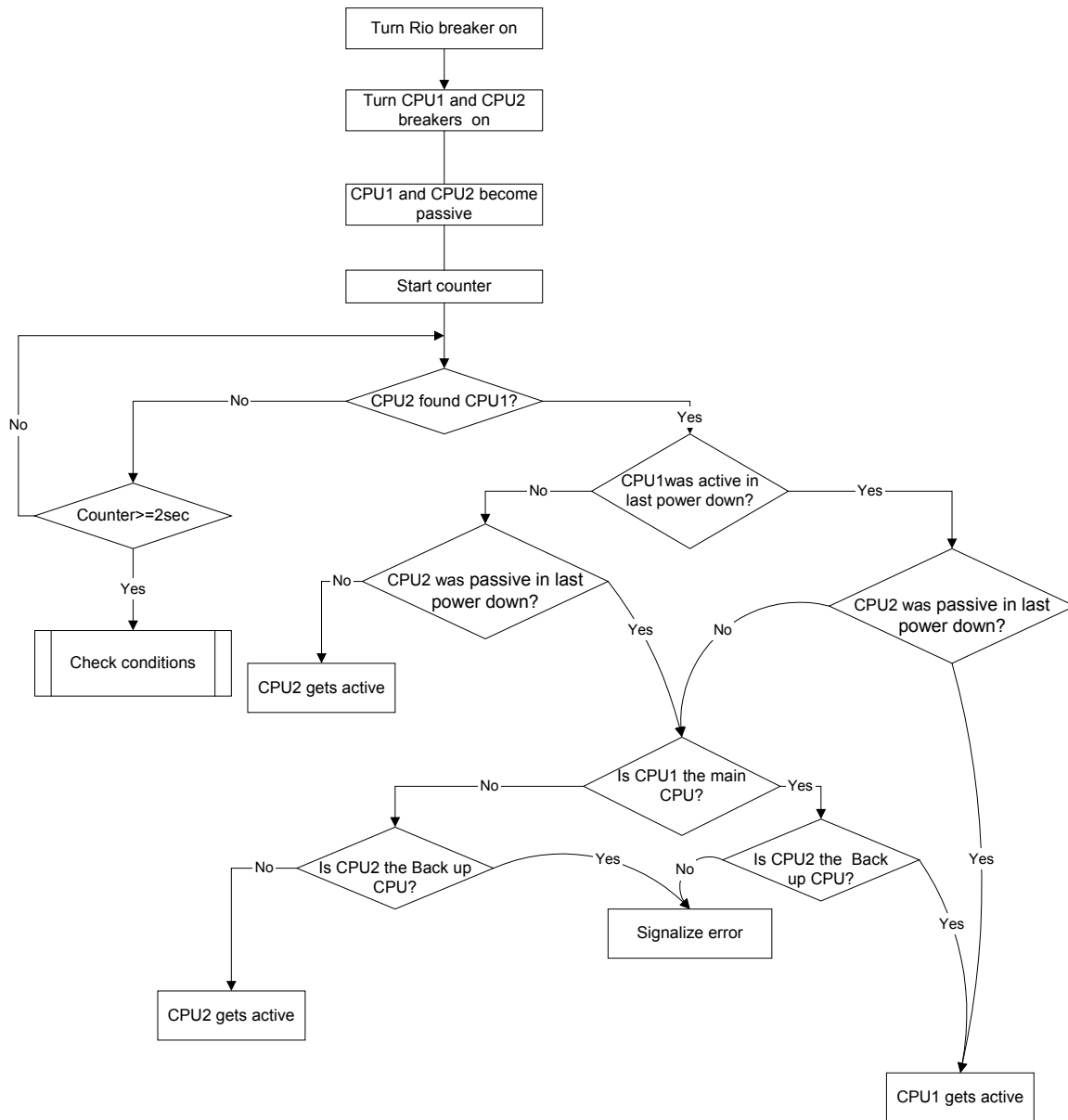


Figure 18. 17 – Flow chart representing the algorithm that decides which co-processor will be active

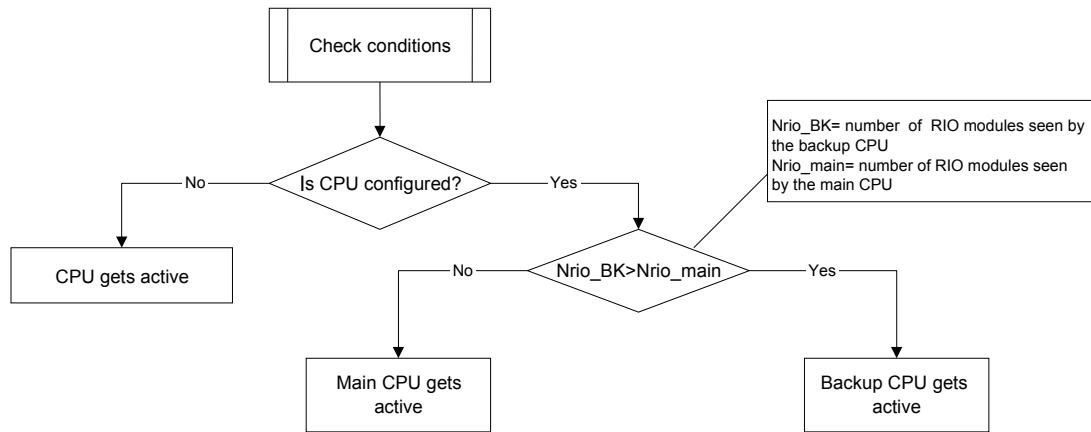


Figure 18. 18 – Flow chart representing the check conditions procedure

Communication with Remote Input and Output (RIO) Modules

The DF66 modules are scanned only if the name of the configuration and date are the same in the main co-processor and DF66. The passive co-processor sends polling commands to check periodically whether it is necessary to assume the control.

As seen in the system architecture figure, the DF66 modules are connected to the system through two different and redundant channels.

There are two ports to be considered:

DF66 Active Port: Port being scanned by the active co-processor, i.e., through this port the Modbus variables are read and written.

DF66 Passive Port: In this port the passive co-processor checks the conditions of the active and passive ports.

Switch Over Procedure

During the normal functioning of the system, there's a switching procedure. It is based on the number of the DF66 that the active co-processor can communicate with (**NRIO active**) and the number of DF66s that the passive co-processor can communicate with (**NRIO passive**).

If NRIO active is shorter than NRIO passive, then there is a switching over procedure that makes the current active co-processor the passive co-processor and the old passive co-processor the current active co-processor. This procedure ensures that the system reads the biggest amount of I/O Modbus variables.

LEDS for Status Indication

- A solid RUN LED (green, ON or OFF) indicates if the co-processor is in active state while a blink RUN LED indicates that the co-processor is in passive state.
- A solid HOLD LED (yellow, ON or OFF) indicates if the co-processor has been configured properly, while a blink HOLD LED indicates that it was not.
- A solid FORCE LED (red, ON or OFF) indicates if the co-processor is in FORCE-IN, FORCE-OUT or SAFE-OUT mode, or not.
- FORCE LED indicator on passive co-processor is blinking. It indicates that the hardware settings are not correct (rotary switch, BR for DF66 or firmware version).

Section 19

CREATING A FOUNDATION FIELDBUS STRATEGY BY USING DF62/DF63

Introduction

This section describes the strategy configuration by using the DF62 controller as bridge. The control loop is shown below:

PROJ_DF62

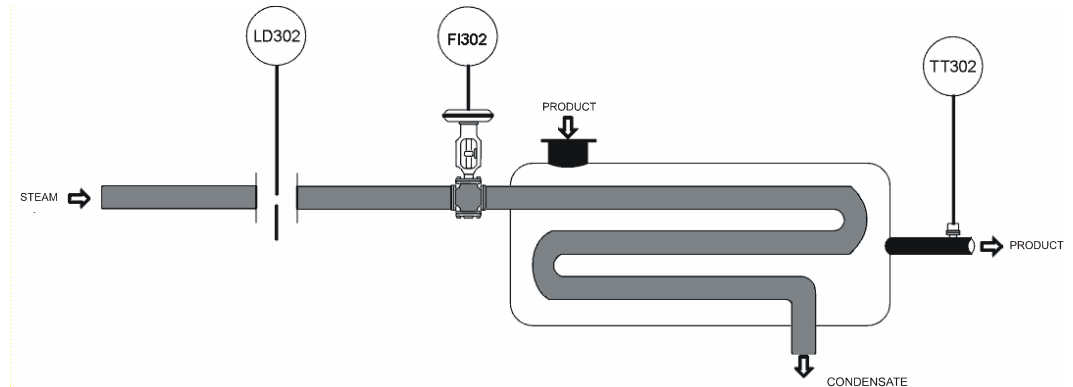


Figure 19. 1 - Example of temperature control process

The purpose of the process is to control the fluid output temperature (controlled variable) using steam flow rate (manipulated variable) to heat it. The fluid temperature will be sent to the master controller, where it will be compared to a temperature set point.

The master output would be the slave controller set point, which will control the steam flow rate to the heat exchanger.

NOTE

This strategy can also be done by using the DF63 controller.

Starting the Area

Step 1

It is possible to create, or edit, an area from the **Studio302**. In the **Studio302** interface select **Areas**. A window will appear listing all areas of database.

To create a new area from the **Studio302**, left-click inside the **Areas** window, then choose **New Area**.

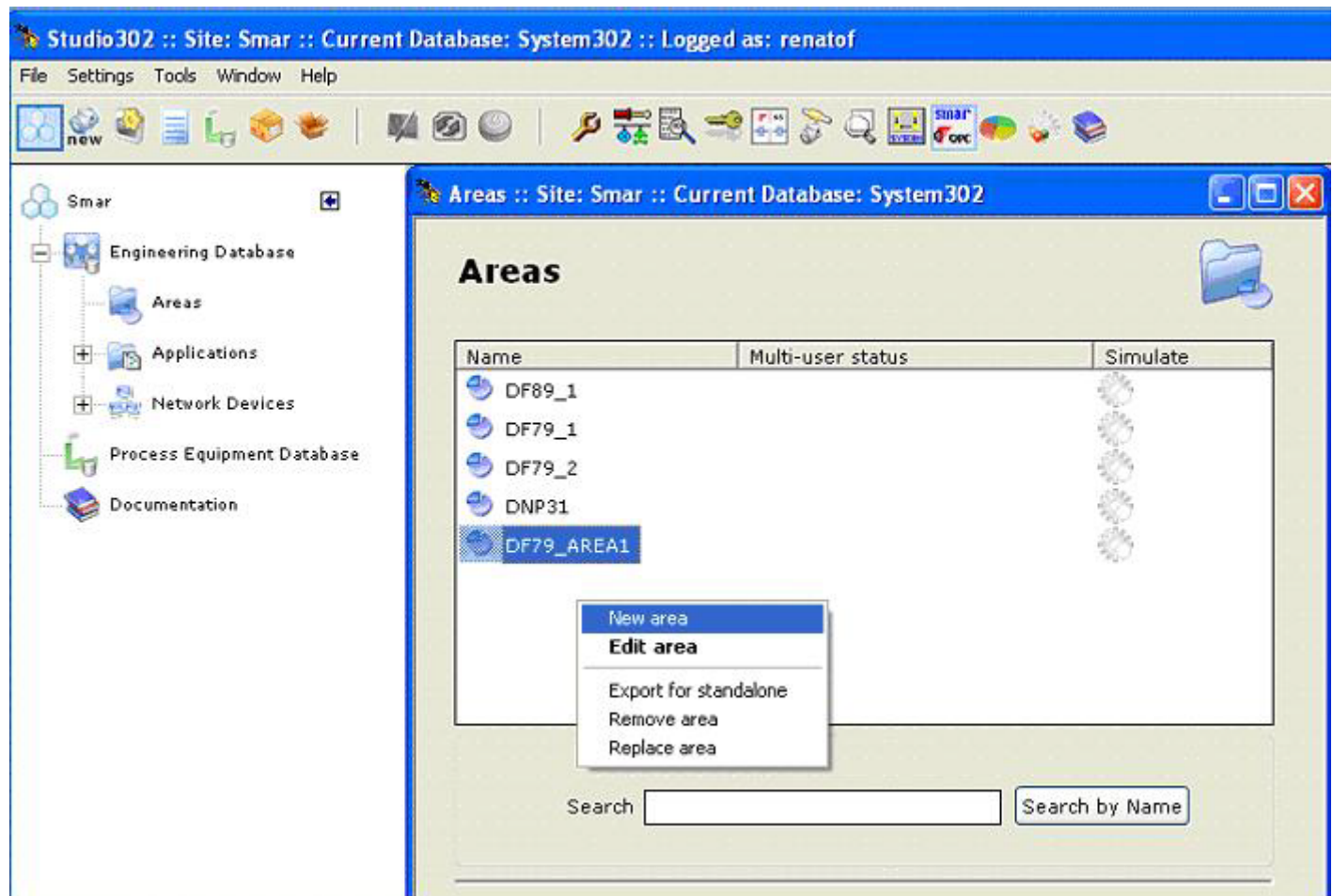



Figure 19. 2 – Creating a new area

Another way to create a new area is from **Syscon**. Click the icon  in the **Studio302** toolbar.

To create a new area on **Syscon**, choose **File → New**, or through the toolbar, choose **New** button . The dialog box shows the areas options. Select **HSE Area** as shown in next figure:

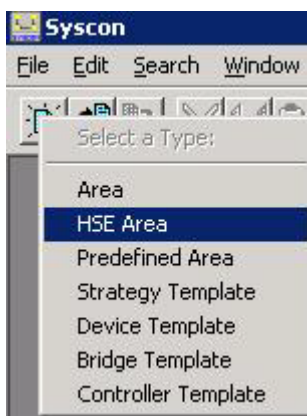


Figure 19. 3 – Options to create Syscon areas

After choosing the area type, it opens a window to the user give a name to the new area.



Figure 19. 4 – New area name

Type the name for the area in the **Area Name** box, and click **Ok**. For this example, it chooses **Proj_DF62** name.

A new window will appear. This window has:


- **Application** – Logical Plant. To insert control strategies into this part.
- **Fieldbus Networks** – Physical Plant. To add devices and function blocks to the area into this part.



Figure 19. 5 – Area divisions

Physical Plant Project

Step 2

In the main window, **Proj_DF62**, right-click the **Fieldbus Networks** icon, , to choose the Server. Choose **Communication Settings** option, or through the toolbar, choose **Communication→Settings**. The communication settings dialog box will open.

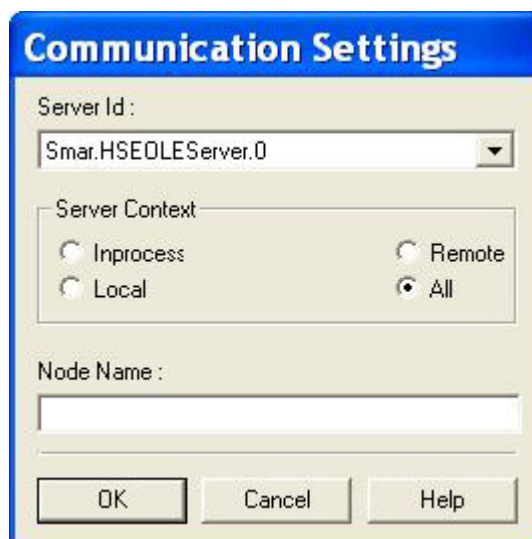


Figure 19.6 – Choosing the Server

Confirm if the **Smar.HSEOLEServer.0** option has already been selected. Otherwise, the user must select it, and then click **OK**.

Arranging the Fieldbus windows

Step 3


After selecting the Server for the area, click the  sign placed at left of the **New Fieldbus**. The HSE network will appear with a tag, for example, **HSE Network 1***. Right-click this item and choose **Expand** option. The Figure below shows the HSE network:



Figure 19.7 – Creating a HSE network

To arrange the screen, click the area window. So, choose **Window** menu on the **Syscon** toolbar, and then **Tile** option.

Adding the Bridge

Step 4

Right-clicking the **HSE Network 8**, it opens a dialog box to add new devices. Choosing **New** option, it is possible to select devices such as **Bridges**, **Controllers** and **Devices** for the area. For the aimed control, choose **Bridge** option. Confirm this choice observing the following figure.

* This number changes if another area was created before. When a new HSE area is created, this number increases.

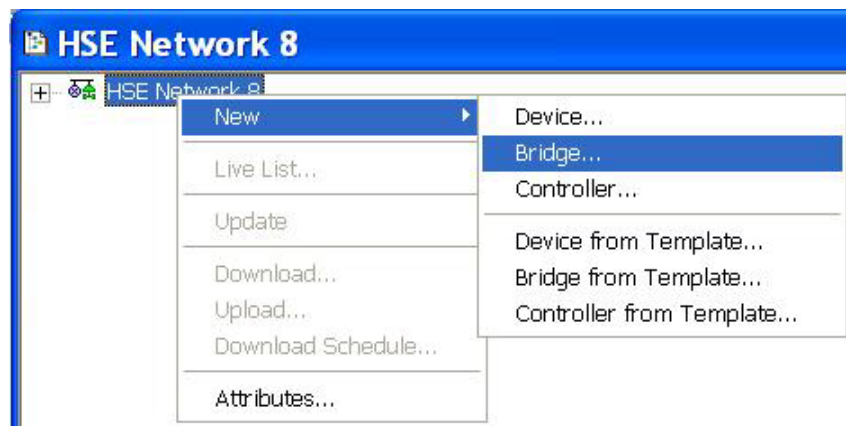


Figure 19. 8 – Choosing the Bridge

After selecting that option, it opens a window as shown in the next figure.

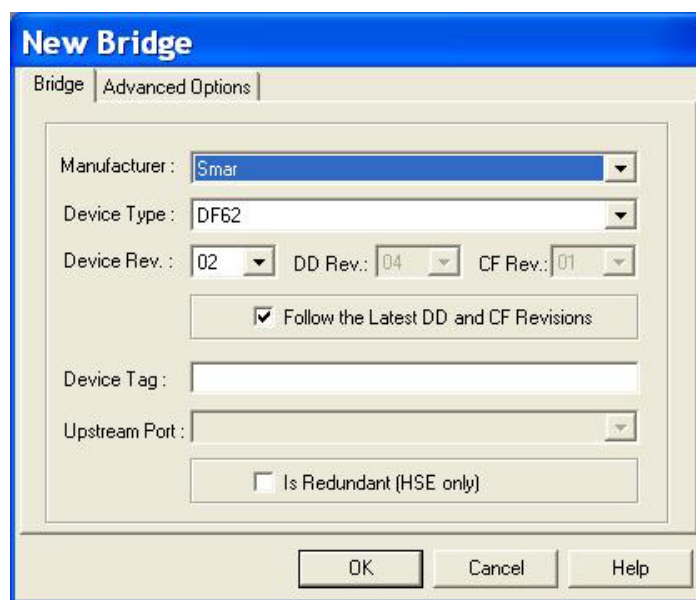


Figure 19. 9 – Setting the Bridge

Select the **DF62** device in the **Device Type** box. In the **Device Tag** box, enter **DF62** or another tag, and click **OK**.

IMPORTANT

Not all characters are valid when naming the elements, so pay attention;

The valid characters are:

A-Z a-z 0-9 # { } [] () + -

The invalid characters are:

~ ` ! @ # \$ % ^ & * = | : ; , . < > ? / ' " \

Return to the **HSE Network 8** window. Right-click the **Bridge** icon, **DF62**, and select **Attributes** item, as shown in the next figure:



Figure 19. 10 – Setting the bridge attributes

Thus, the bridge attributes can be seen and, if necessary, changes its tag. Click **Ok**. See the next figure.

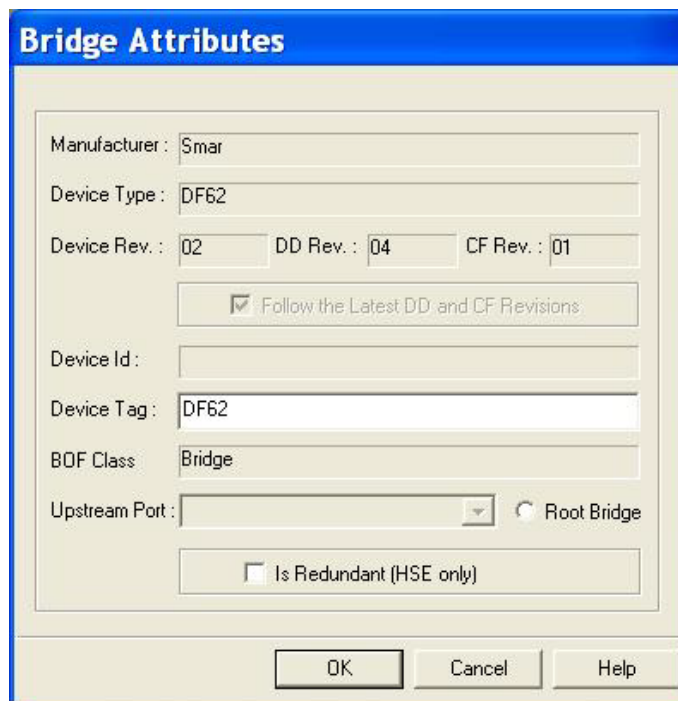


Figure 19. 11 – Bridge attributes

Adding Fieldbus Devices

Step 5

After adding the bridge, it is possible to insert the field devices. First, return to the **Proj_DF62** window and right-click the **DF62** bridge. To configure the DF62 channels, select **New Fieldbus** option. The following figure shows the Fieldbus channel creation:



Figure 19. 12 – Creating Fieldbus channels

When choosing **New Fieldbus** option, the dialog box to choose the channel type and name it with tags opens:

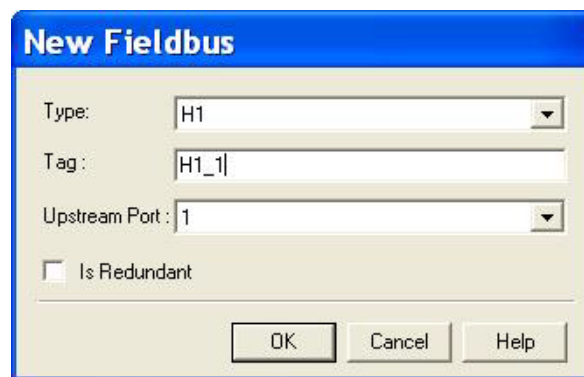


Figure 19. 13 – Attributing tags to the Fieldbus channels

Choose the fieldbus type and name it with a tag. If the user does not name it with a specific tag, the default tag will be *Fieldbus 1* (This number changes if another channel was created previously). To rename the fieldbus channel, right-click it, and then choose **Expand** item. When the Fieldbus window opens, right-click it and choose **Attributes** option. So, the user can rename the fieldbus channel.

Return to the **Proj_DF62** window. Right-click the fieldbus channel created and choose **Expand** option on the popup menu. The figure below shows the fieldbus channel:



Figure 19. 14 – Creating the Fieldbus channel

In the **H1_1** window, right-click **H1_1** to choose **New→Device**. The next figure shows adding devices on the network:

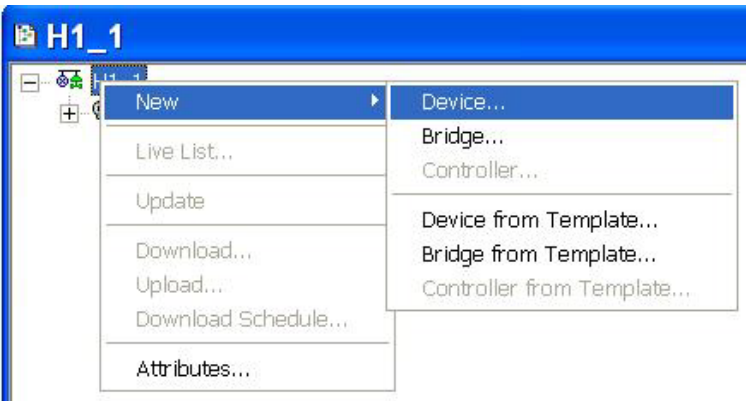


Figure 19. 15 – Adding devices in the Fieldbus channel

The dialog box to select and name the devices opens. Choose **SMAR** in the **Manufacturer** box, and then select **TT302** in the **Device Type** box. Name the device with a tag, and click **OK**:

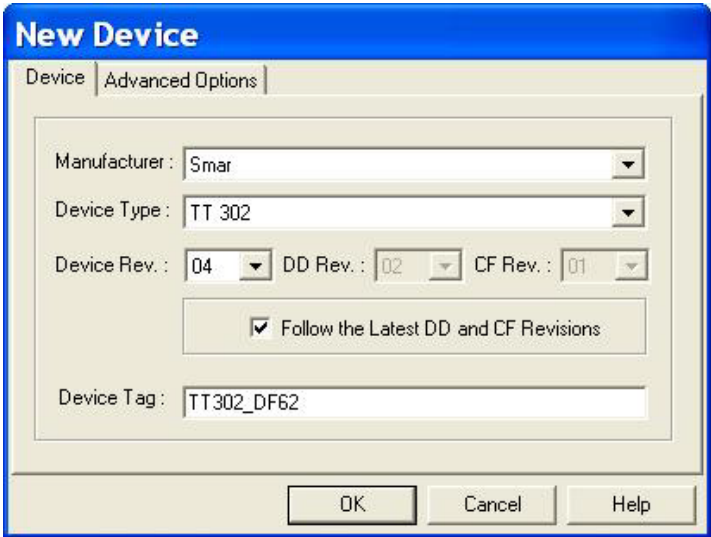


Figure 19. 16 – Setting the Fieldbus devices


Repeat this procedure to add a transmitter (LD302) and a converter (FI302) in the flow control.
After adding the devices to the area, the **H1_1** channel will be according to the following figure:



Figure 19. 17 – Devices added into the Fieldbus channel

Adding Function Blocks

Step 6

Now the user can add Function Blocks. To add a new Function Block (FB), click the  sign, and right-click the **Virtual Field Device** (FB VFD) icon. Choose **New Block** option.

The FB VFD is responsible for the data management.

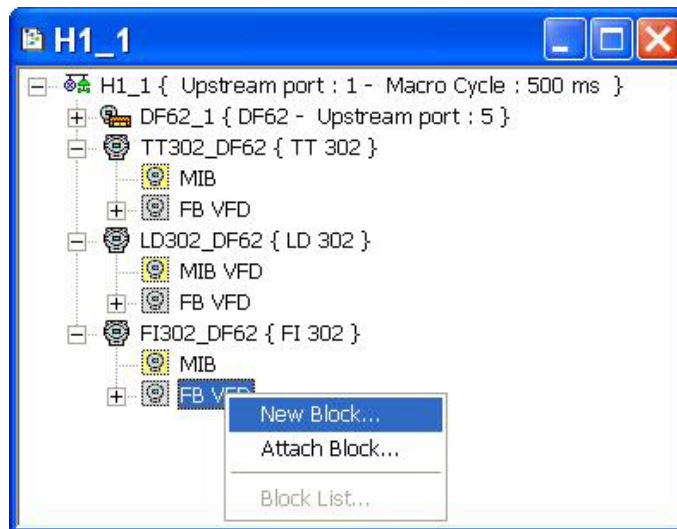


Figure 19. 18 – Choosing New Blocks

The **Function Block Characterization** dialog box will appear. The **Block Type** option shows the SMAR's FB.

Select the block in the **Block Type** box, and name it in the **Block Tag** box. The following figure shows adding the Analog Input function block.

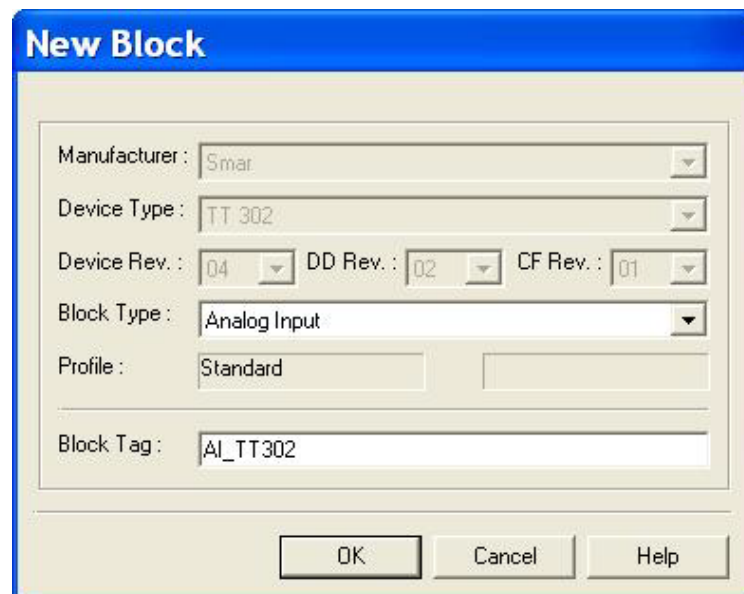


Figure 19. 19 – Adding function blocks

For this example, it is necessary to add AI, PID, and AO function blocks to build a cascade control configuration.

NOTE

From the **Syscon** version 6.00, it is not necessary to configure the Transducer (TRD), Resource Block (RES), Diagnostics (DIAG) and Display (DSP) blocks, because they are preinstantiated in the devices.

The channel configuration with all function blocks and devices is presented in the next figure. For better identification of the Transducer, Resource and Diagnostics function blocks name them with specific tags.

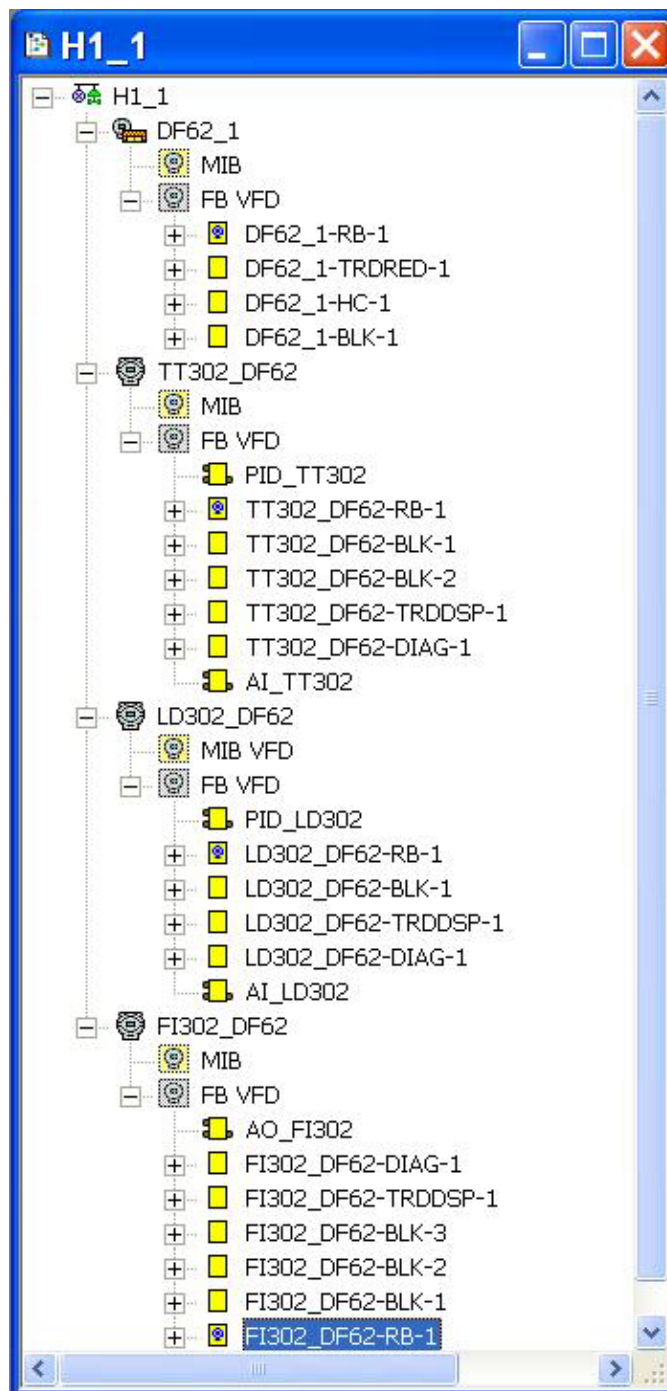


Figure 19. 20 – Fieldbus channel composition

Now the strategy area Application (Logical Plant) can be developed. First it is necessary to establish a new process cell.

Creating New Process Cells

Step 7

The **Logical Plant** can be divided in several process cells, according to the plant.

To create a new Process Cell, right-click the **Application** icon, and select **New Process Cell** item.

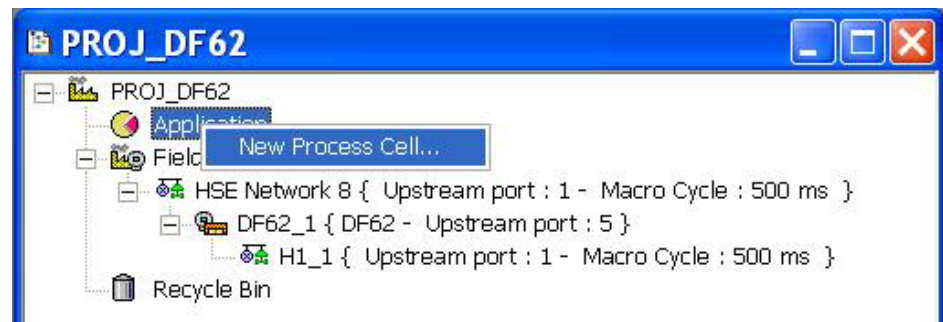


Figure 19.21 – Inserting the Process Cell

The dialog box to name the **Process Cell** will open:

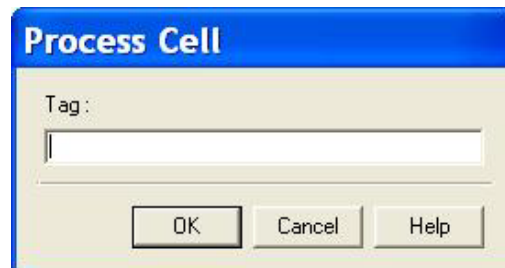


Figure 19.22 –Attributing tag to the Process Cell

If the user needs name the Process Cell with a specific tag. Enter it in the **Tag** box, and click **OK**. To create more process cells, the procedure above can be repeated.

After inserting the Process Cell, the **Proj_DF62** window will be according to the next figure:

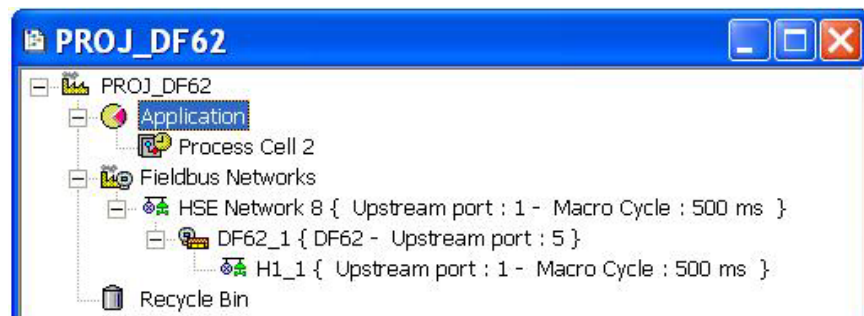


Figure 19.23 – Area window after inserting the Process Cell

NOTE

The user must remember that **Application** is a virtual division. It only divides a large plant. For example: if the plant has two networks, they can be **Process Cells** in the **Syscon**. One **Application** can have several **Process Cells**, but a **Process Cell** cannot be in more than one **Application**.

Creating a Control Module

Step 8

Now the user can create a Control Module in the Application section.

Right-click the **Process Cell2** icon, and choose **Expand** item.

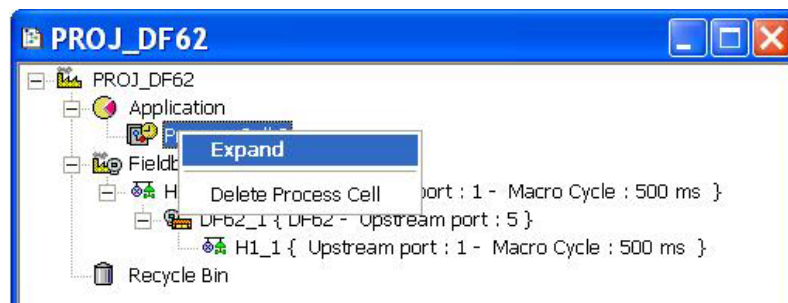


Figure 19. 24 – Creating the Control Module

To arrange the screen, click the area window. So, choose **Window** menu on the **Syscon** toolbar, and then **Tile** option.

As following, return to the **Process Cell2** window. Right-click the **Process Cell2** item, and choose **New Control Module**. The figure below shows creating the New Control Module.

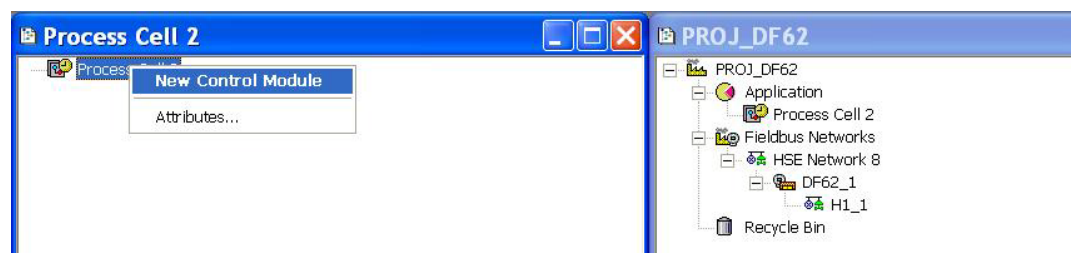


Figure 19. 25 – Creating the New Control Module

The **New Control Module** dialog box will appear. Name it with the tag related to the application. To continue, click **OK**.

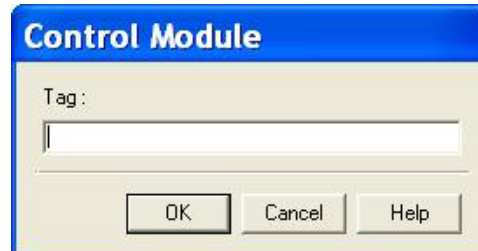


Figure 19. 26 – Attributing tag to the Control Module

IMPORTANT

Remember that not all characters are valid when naming the elements with tags.

Inserting Blocks in the Control Module

Step 9

Now the user can add function blocks for the devices in the **Logical Plant**. Right-click the **Control Module 2** item, and choose **Attach Block** option, as shown in the next figure:

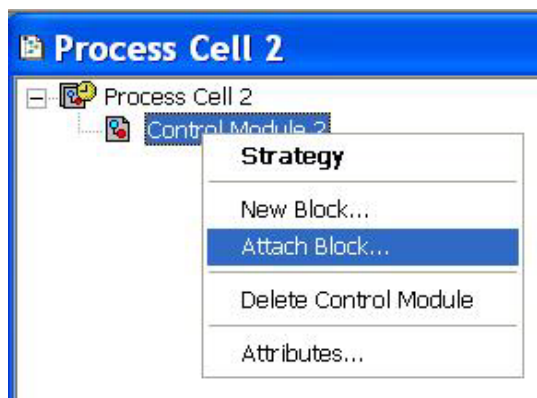


Figure 19. 27 – Adding new functions blocks to the FBAP

The **Attach Block** dialog box will open as shown below:



Figure 19. 28 – Attaching blocks to the Control Module

The available function blocks for the application are showed in the **Attach Block** box. For the aimed control, the function blocks that must be inserted will appear in the box. So, select them one by one, and click the **OK** button.

When the Attach Block process ends, the application will be as shown in the following figure:

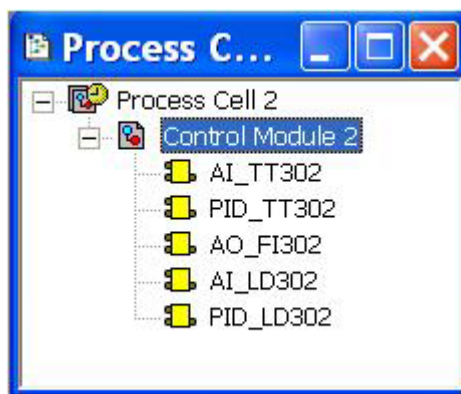


Figure 19. 29 – Blocks added to the Control Module

Another way to attach the blocks is left-clicking the element and drop it to the window.

A new tag can be done to the **Control Module** right-clicking it, and selecting **Attributes**. For the aimed example will be attributed **Estrategia_ProjDF62**.

Configuring the Control Strategy

Step 10

Now the user is ready to develop the control strategy.

First, right-click the **Estrategia_ProjDF62** icon, and select **Strategy**. The **Strategy** window will appear as shown in the following figure.

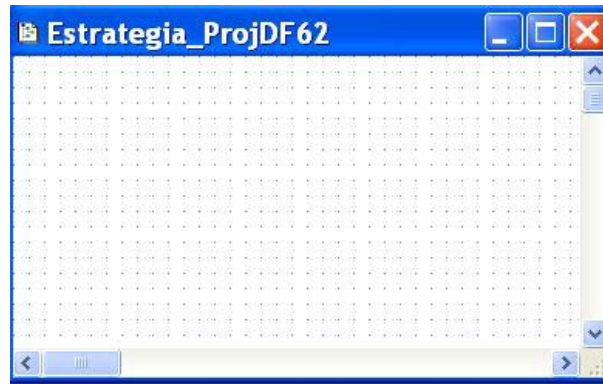


Figure 19. 30 – Strategy window

At this moment there are 3 or 4 windows opened in the **Syscon**. Minimize the **H1_1** window. Click the **Proj_DF62** window. On the toolbar, choose **Window → Tile**.


If the user does not have a monitor upper than 17", it is recommended to minimize the strategy window. Thus, it is possible to see the whole area.

The strategy window offers several tools for drawing. Refer to the **Syscon** manual for further details.

Adding Blocks to the Strategy window

Step 11

Now the function blocks can be added to the **Estrategia_Proj_DF62** window.

In order, click the first block,  **AI_TT302**, and drop it into the strategy window. A function block will be created automatically.

The next figure shows the function block added to the strategy window:

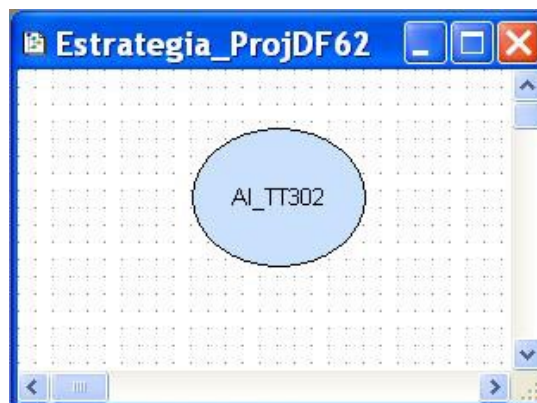


Figure 19. 31 – Block inserted into the strategy window

Repeat the drag-and-drop procedure for the other blocks such as **AI_LD302**, **PID_LD302**, **PID_TT302**, and **AO_FI302**.

Linking the Blocks

Step 12

There is a specific tool to link the blocks, the **Link** button, , on the **Strategy** toolbar.

Click this button on the toolbar, and then in the TT302_AI function block. The dialog box for linking the input and output parameters will appear. Select **OUT**, and then click the **OK** button as shown in the following figure.

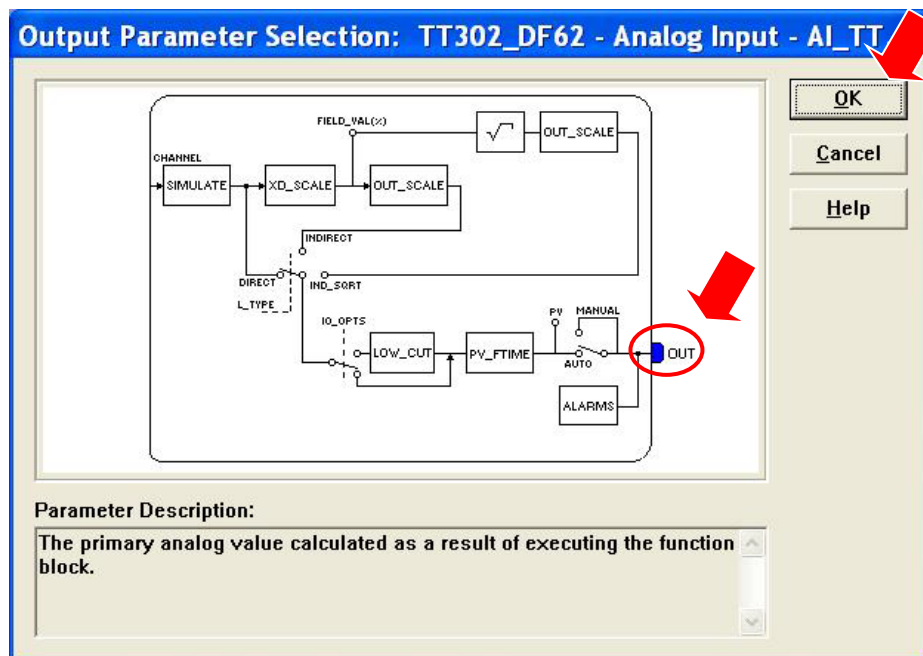


Figure 19.32 – Linking the Function Blocks

Move the mouse cursor up to the block that will be linked.

The user also does the fast link procedure just right-clicking the function block. The links necessary for this strategy are:

Direct Links:

- OUT(AI_TT302) → IN(PID_TT302)
- OUT(PID_TT302) → IN(PID_LD302)
- OUT(PID_LD302) → CAS_IN(AO_FI302)
- OUT(AI_LD302) → CAS_IN(PID_LD302)

Back Links:

- BKCAL_OUT(PID_LD302) → CAS_IN(PID_TT302)
- BKCAL_OUT(AO_FI302) → BKCAL_IN(PID_LD302)

After linking the parameters specified above, the strategy window will be as shown in the following figure:

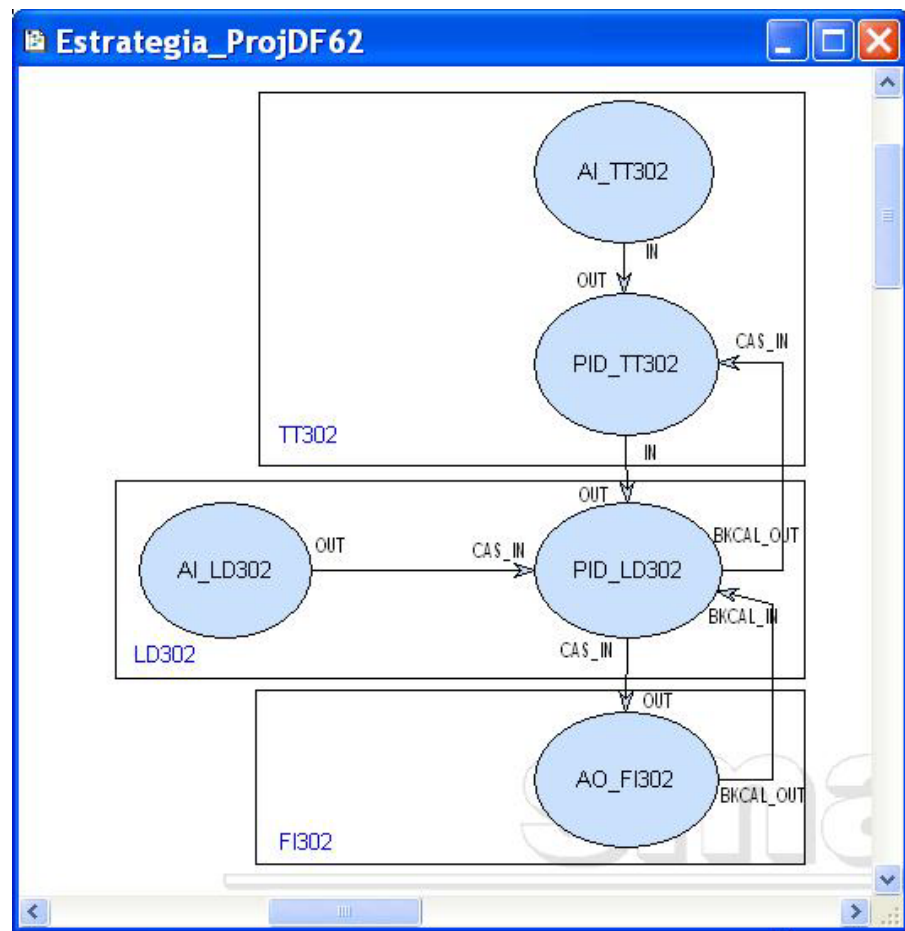


Figure 19. 33 – Links between the Function Blocks

Function Block Characterization

Step 13

The function blocks that are in the strategy must be set according to the application for them. So, it is necessary to do the block characterization.

The online and offline modes are possible for the block characterization. In the offline mode, the parameters are set before starting the communication between the devices. The online characterization is executed directly in the devices when the plant is already communicating.

To change the Function Block parameters, consider the following topics:

1. In the Strategy window

Select the block to characterize. Right-click it, and select the **Off Line Characterization** option, or double-click it. The following figure shows the block that is being done the offline characterization:

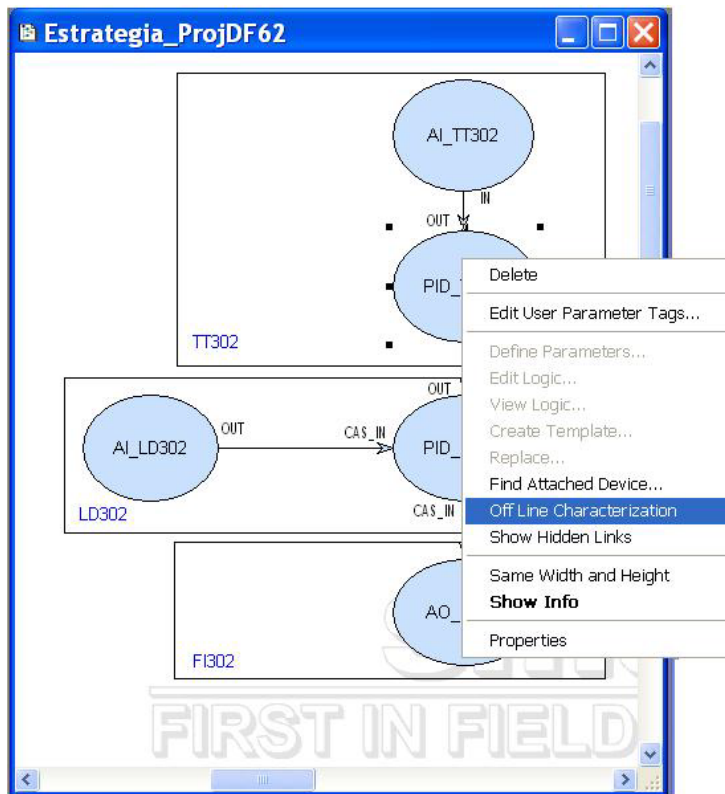


Figure 19. 34 – Offline characterization in the Strategy window

2. In the H1_1 window

Another way to do the offline characterization is right-clicking the function block, and then selecting the **Off Line Characterization** option, as shown in the following figure:

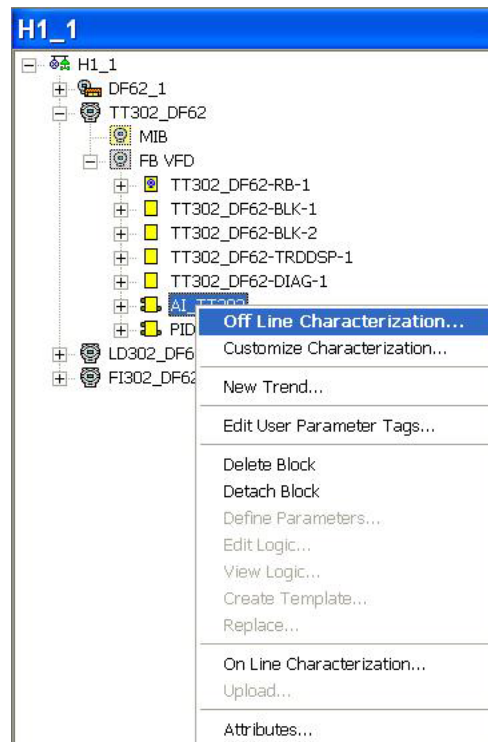


Figure 19. 35 – Offline characterization in the Fieldbus channel window

For both situations, the **Off Line Characterization** dialog box will appear:

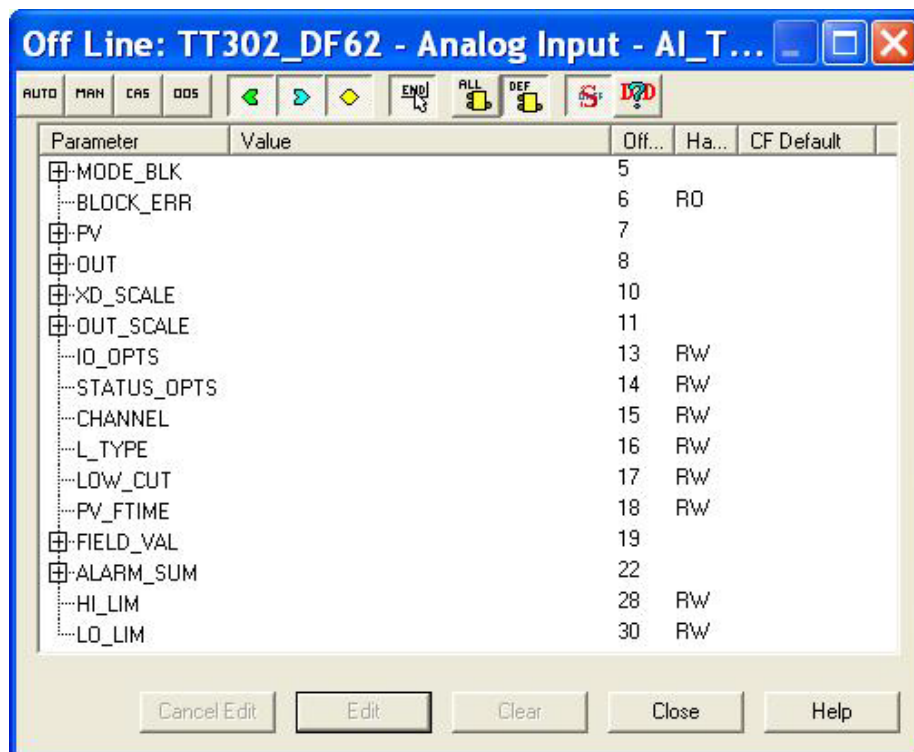


Figure 19.36 – Function Block Characterization dialog box

Double-click at the right side of the parameter to change it. Another option is click it once, and then in **Edit** to start editing the parameter value. At the ending, click **End Edit**.

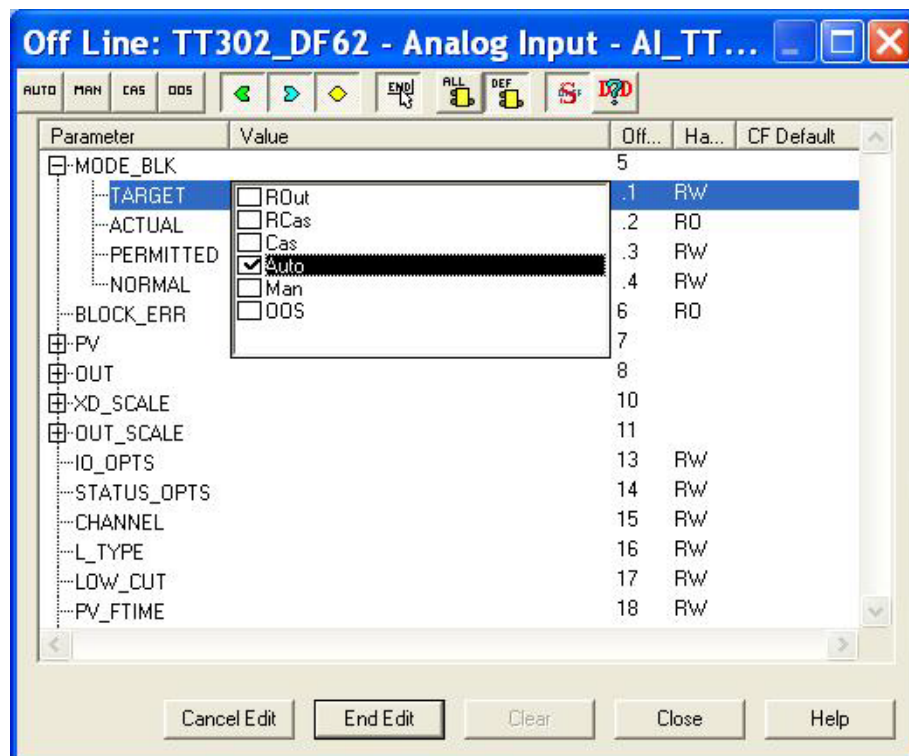


Figure 19.37 – Editing the parameter in the Function Block Characterization box


The list below shows the parameters that must be set for this area:

DEVICE	TAG	BLOCK	PARAMETER
LD302	LD302_DF62	TR	MODE_BLK.Target = AUTO
		RS	MODE_BLK.Target = AUTO
		DSP	MODE_BLK.Target = AUTO
			BLOCK_TAG_PARAM_1= FT101_AI
			INDEX_RELATIVE_1 = 8
			MNEMONIC_1 = VAZAO
			ACCESS_1 = MONITORING
			ALPHA_NUM_1 = MNEMONIC
			DISPLAY_REFRESH = UPDATE DISPLAY
		AI_LD302	MODE_BLK.Target = AUTO
			XD_SCALE.EU_100 = 100
			XD_SCALE.EU_0 = 0
			XD_SCALE.UNITS_INDEX = inH2O(4°C)
			OUT_SCALE.EU_100 = 100
			OUT_SCALE.EU_0 = 0
			OUT_SCALE.UNITS_INDEX = %
		PID_LD302	CHANNEL = 1
			L_TYPE = INDIRECT
			MODE_BLK.Target = AUTO
			PV-SCALE.EU_100 = 100
			PV-SCALE.EU_0 = 0
			PV-SCALE.UNITS_INDEX = %
			OUT_SCALE.EU_100 = 100
			OUT_SCALE.EU_0 = 0
			OUT_SCALE.UNITS_INDEX = %
			GAIN = 0.5
			RESET = 1
			RATE = 0

DEVICE	TAG	BLOCK	PARAMETER
TT302	TT302_DF62	TR	MODE_BLK.Target = AUTO
			SENSOR_TYPE = PT100IEC
			SENSOR_CONNECTION = THREE WIRES
			SENSOR_TRANSDUCER_NUMBER = 1
		RS	MODE_BLK.Target = AUTO
		DSP	MODE_BLK.Target = AUTO
			BLOCK_TAG_PARAM_1 = TT100_AI
			INDEX_RELATIVE_1 = 8
			MNEMONIC_1 = TEMP
			ACCESS_1 = MONITORING
			ALPHA_NUM_1 = MNEMONIC
			DISPLAY_REFRESH = UPDATE DISPLAY
		AI_TT302	MODE_BLK.Target = AUTO
			XD_SCALE.EU_100 = 500
			XD_SCALE.EU_0 = 0
			XD_SCALE.UNITS_INDEX = °C
			OUT_SCALE.EU_100 = 100
			OUT_SCALE.EU_0 = 0
			OUT_SCALE.UNITS_INDEX = %
		PID_TT302	CHANNEL = 1
			L_TYPE = INDIRECT
			MODE_BLK.Target = AUTO
			PV_SCALE.EU_100 = 100
			PV_SCALE.EU_0 = 0
			PV_SCALE.UNITS_INDEX = %
			SP = 50
			GAIN = 0.5
			RESET = 1
			RATE = 0

DEVICE	TAG	BLOCK	PARAMETER
FI302	FI302_DF62	TR	MODE_BLK.Target = AUTO
			TERMINAL_NUMBER = 1
		RS	MODE_BLK.Target = AUTO
		DSP	MODE_BLK.Target = AUTO
			BLOCK_TAG_PARAM_1 = FCV102_AO
			INDEX_RELATIVE_1 = 9
			MNEMONIC_1 = VALVULA
			ACCESS_1 = MONITORING
			ALPHA_NUM_1 = MNEMONIC
			DISPLAY_REFRESH = UPDATE DISPLAY
		AO_FI302	MODE_BLK.Target = AUTO
			PV_SCALE.EU_100 = 100
			PV_SCALE.EU_0 = 0
			PV_SCALE.UNITS_INDEX = %
			XD_SCALE.EU_100 = 20
			XD_SCALE.EU_0 = 4
			XD_SCALE.UNITS_INDEX = mA

After the parameter setting, the user can start the device communication. It is necessary commissioning the devices to attribute the tags, IDs, and device addresses properly. If this procedure is not executed, the **Syscon** will detect the not commissioned device and the download for this device will be aborted. Finishing the device commissioning, the download process can start. The download process can be executed, for example, returning to the **Proj_DF62** window, right-

clicking the **Fieldbus Networks** icon, , and selecting the **Download** option. For further details about the available download types, refer to the **Syscon** manual.

Macrocycle for the H1 Channel

The macrocycle for each channel depends on the configuration, but can be estimated by using data such as quantity of links, field devices and quantity of blocks. The following examples show this calculation.

Data for configuration 1, with high traffic:

- 50 object links in the H1 network H1 (30 ms per link)
- 8 field devices per channel
- 4 function blocks per device
- Supervision of one dynamic view per block

Background time

For the “background traffic”, time related to the supervision and asynchronous messages, is necessary at least 960 ms ($8 \times 4 = 32$ views, 30 ms for each one). For the “foreground traffic”, time related to links and control, is necessary at least 1500 ms (50 links and 30 ms for each one). So, the minimum macrocycle is 2460 ms. Therefore, a security extra time is added, about 20%, and the macrocycle will be equal to 3 s. The quantity of dynamic views per second is 10 (32 views/ 3s) and the publish/subscribe time is 3 s (equals to the macrocycle).

Data for configuration 2, with low traffic:

- 4 object links in the H1 network (30 ms per link)
- 8 field devices per channel
- 2 function blocks per device
- Supervision of one dynamic view per block

Calculating the limits:

- Background traffic: 480 ms ($8 \times 2 = 16$ views, 30 ms for each one)
- Foreground traffic: 120 ms (4 links and 30 ms for each one)
- Macrocycle: 720ms (600 ms + 20% security extra time)
- Quantity of dynamic views per second: 22 (16 views/ 0.720 s)
- Publish/subscribe time: 720 ms (macrocycle)

So, the macrocycle can be estimated:

- $\text{Macrocycle} = (\text{Background traffic} + \text{Foreground traffic}) \times 1.2$
- $\text{Background traffic} = \text{Quantity of views} \times 30 \text{ ms}$
- $\text{Foreground traffic} = \text{Quantity of links} \times 30 \text{ ms}$

The time equals to 30 ms is an average value estimated for the calculations above, and it is used for the most of situations.

CREATING A PROFIBUS CONFIGURATION BY USING DF73, DF95 OR DF97

Introduction

This section will show a strategy configuration using the DF73 controller. The architecture of the Profibus control loop can be seen in the following figure. It is possible to implement the same strategy using the DF95 or DF97 controllers.

The purpose of the process is to control the product output temperature using steam flow rate to heat it. The product temperature will be sent to the master controller, where it will be compared to the temperature set point.

The condensed steam is recovered by an intermediate tank, and the tank's level is kept through the control of a liquid pump which returns the condensed steam to the process.

PROJ_DF73

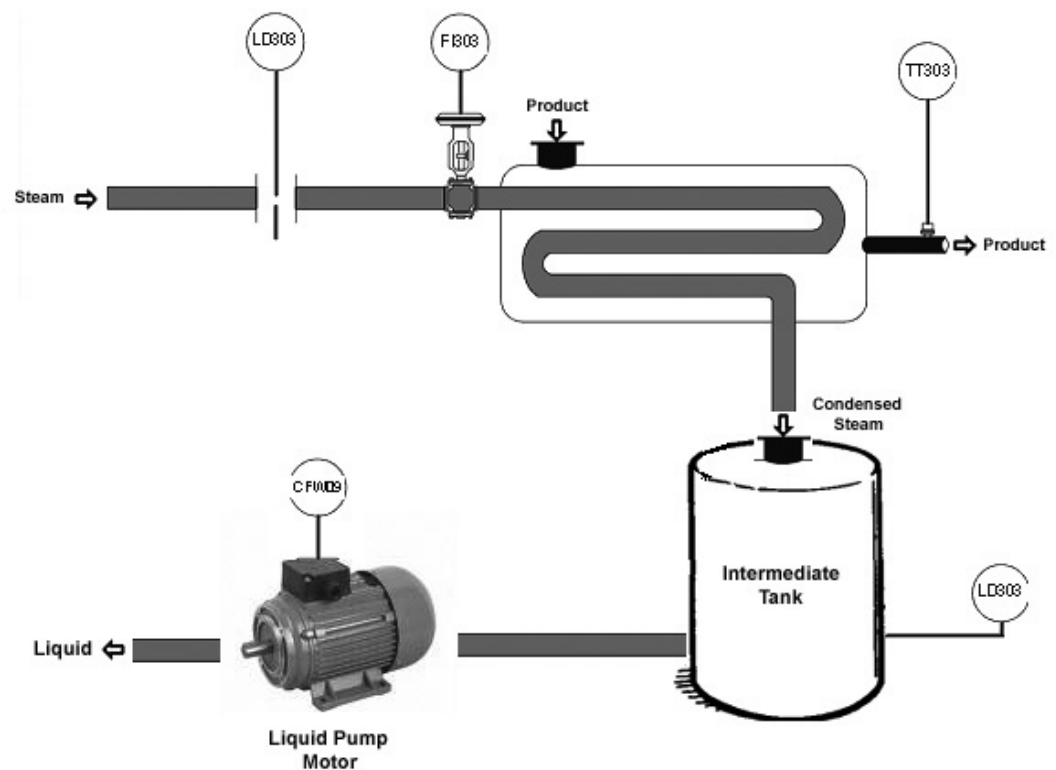


Figure 20. 1 – Example of Process

For the example above the following Profibus devices were used: an AC Drive (CFW09) to control the pump's motor, a LD303 for the steam flow rate measurement, a FI303 to control the steam flow rate, a TT303 for the product temperature measurement, and a LD303 for the tank level measurement.

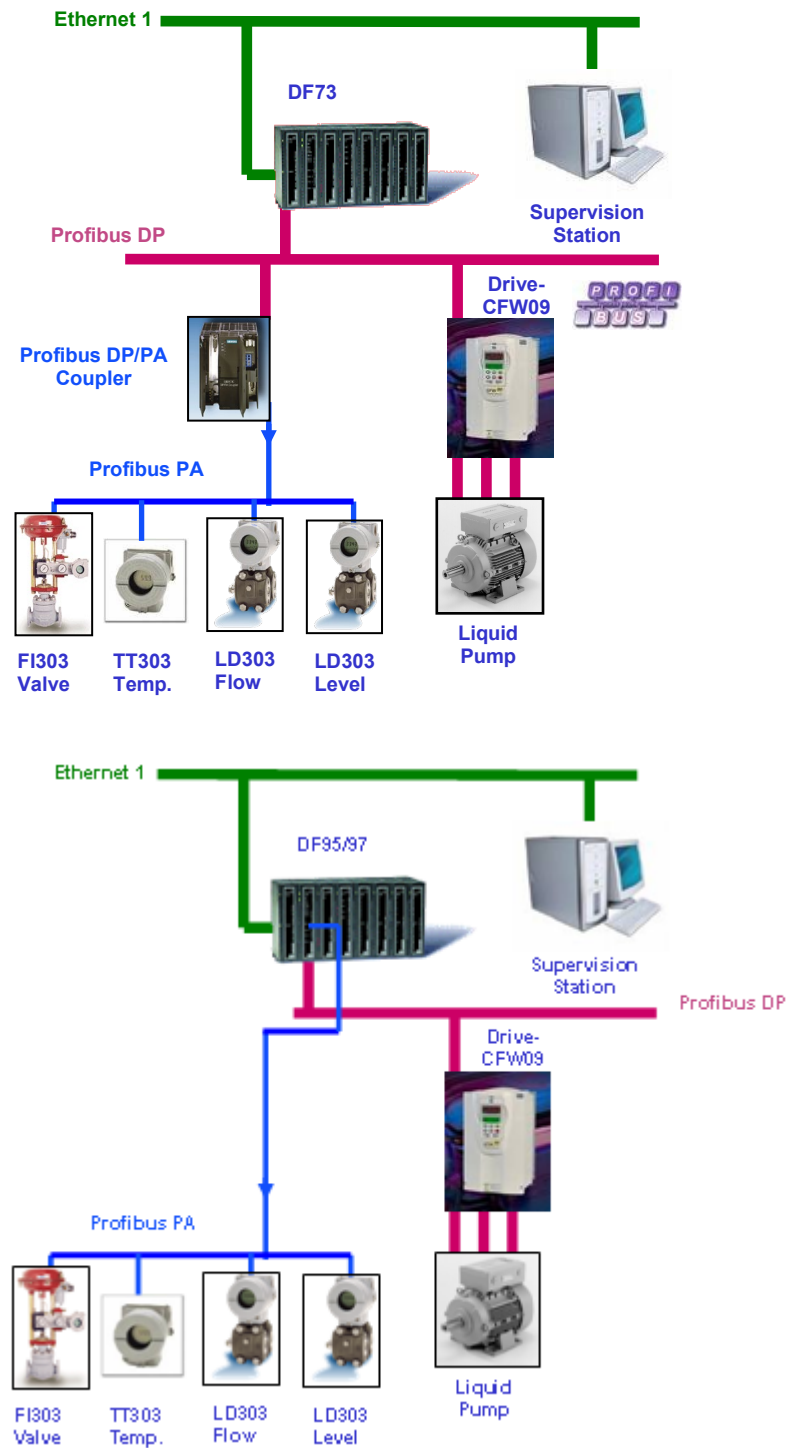


Figure 20. 2 – Profibus DP and PA control network architecture

The addresses, tags and modules which were attributed to the devices are as follows:

Device	Device TAG	Address	Modules
FI303	FI303_Vazao	3	eSP+RB+RCASIN+RCASOUT+POS_D+CB EMPTY_MODULE EMPTY_MODULE
TT303	TT303_Temp	4	Analog Input (short) EMPTY_MODULE
LD303	LD303_Vazao	5	Analog Input (short) Total_Settot
LD303	LD303_Nivel	6	Analog Input (short) EMPTY_MODULE
CFW09	Bomba_Liquido	7	6wIn/6wOut

In the next topics will show, step by step, how to configure the DF73 controller for the aimed example. At the **SYSTEM302** is possible to do control logic of two ways: ladder or function blocks. The steps 1 to 7 are necessary to configure the Profibus network. The step 8 shows how to do the configuration using ladder logic. From step 8' hereafter is shown how to do the logic using function blocks. Also is possible to mix the two configurations, i.e., part in ladder and part in function blocks.

IMPORTANT

Before starting the Profibus configuration in the **SYSTEM302** is necessary to obtain information about each slave device configuration (module address, baud rate, supported types of cyclic communication and mapping structure). Also is necessary the electronic identification file of the device GSD. This information is obtained with the respective Profibus device manufacturer.

Starting the Area

Step 1

It is possible to create, or edit, an area from the **Studio302**. In the **Studio302** interface select **Areas**. A window will appear listing all the areas of database.

To create a new area from the **Studio302**, left-click inside the **Areas** window, then choose **New Area**.

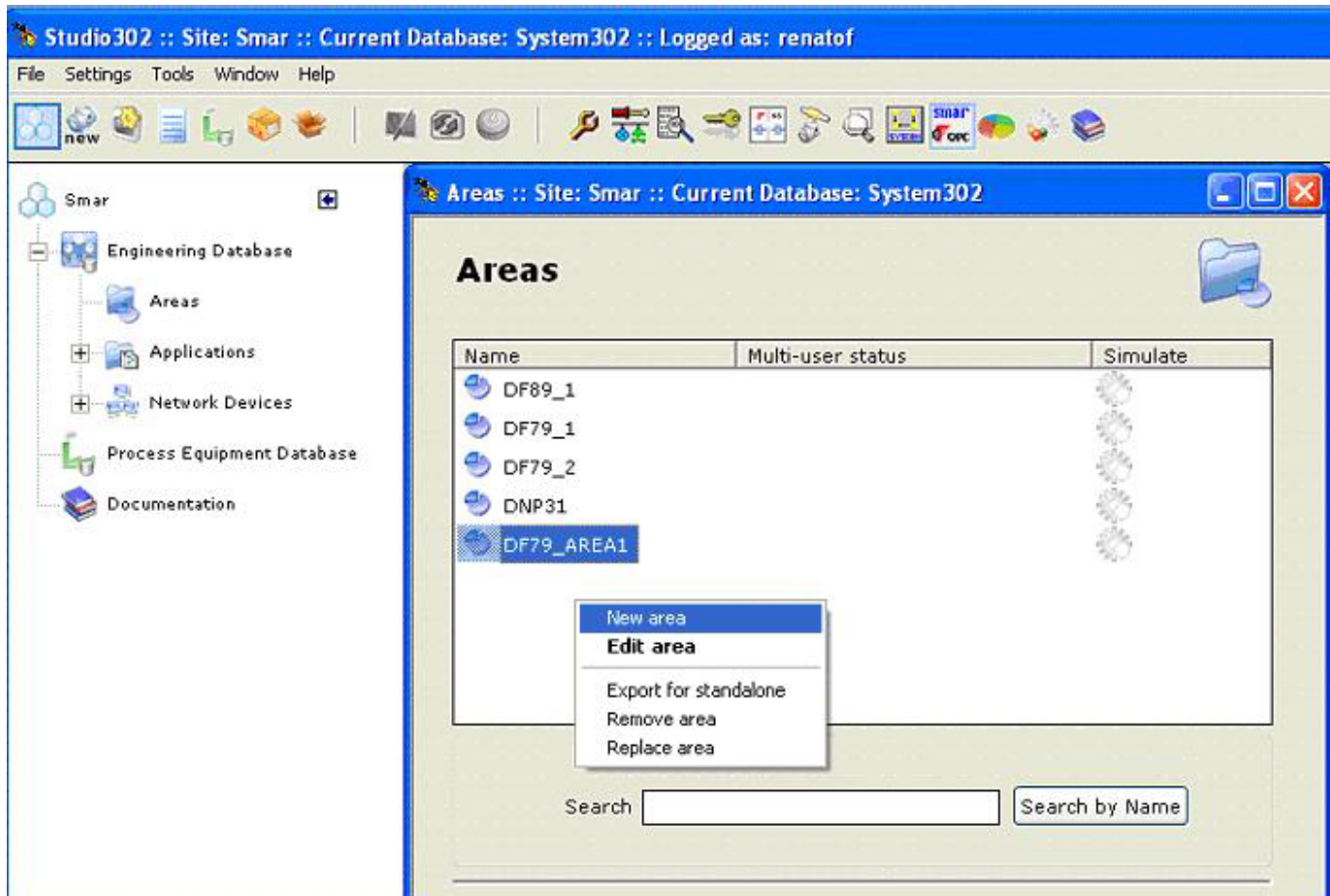



Figure 20. 3 - Creating a new area

Another way to create a new area is from **Syscon**. Click the icon  in the **Studio302** toolbar.

To create a new area on **Syscon**, choose **File → New**, or through the toolbar, choose **New** button . The dialog box shows the areas options. Select **HSE Area** as shown in the next figure:

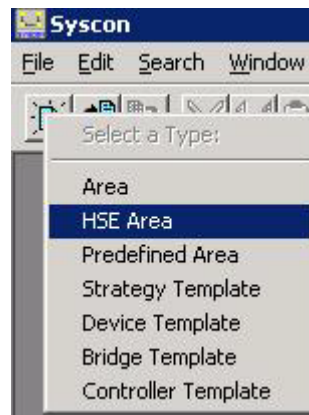


Figure 20. 4 - Options to create Syscon areas

After choosing the area type, it opens a window to the user give a name to the new area.

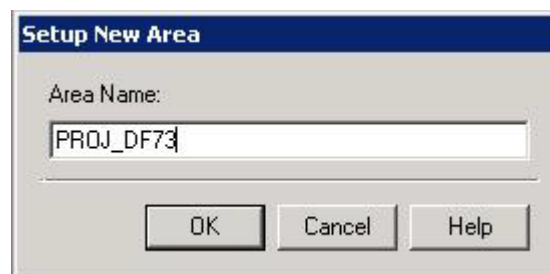


Figure 20. 5 – New area name

Type the name for the area in the **Area Name** box, and click **Ok**. For this example, it chooses **PROJ_DF73** name.

A new window will appear. It shows:

- **Application** – Logical Plant. To insert the control strategies into this part.
- **Fieldbus Networks** – Physical Plant. To insert devices and function blocks used in the area.

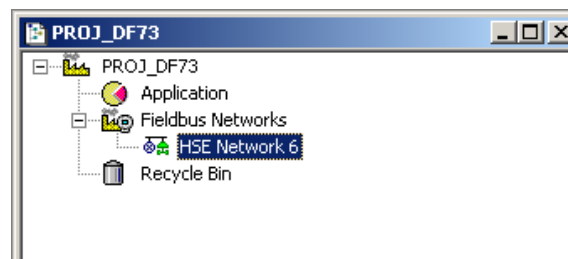


Figure 20. 6 - Area divisions

Physical Plant Project

Step 2


In the main window, **PROJ_DF73**, right-click the **Fieldbus Networks** icon, , to select the Server. Choose **Communication Settings** option, or through the toolbar, choose **Communication→Settings**. The communication settings dialog box will open.




Figure 20.7 – Choosing the Server

Confirm if the **Smar.HSEOLEServer.0** option has already been selected. Otherwise, the user must select it, and then click **OK**.

Arranging the Fieldbus windows

Step 3

After selecting the Server for the area, click the  sign placed at left of the **Fieldbus Networks**. The HSE network will appear with a tag, for example, *HSE Network 1**. Right-click this item and choose the **Expand** option. The following figure shows the HSE network:

NOTE

* This number changes if another area was created before. When a new HSE area is created, this number increases.



Figure 20.8 – Creating the HSE network

To arrange the screen, click the area window. So, choose **Window** menu on the **Syscon** toolbar, and then choose **Tile** option.

Inserting the Controller

Step 4

Right-clicking the **HSE Network 1** icon, a dialog box to add new devices will appear. By clicking **New**, it is possible to select devices such as Bridges, Controllers and Devices for the area. For the aimed control, it selects the **Controller** option. Confirm this choice as shown in next figure.

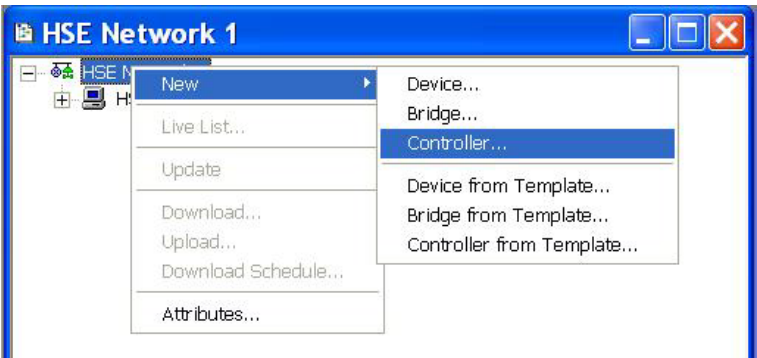


Figure 20. 9 – Choosing the controller for the area

After adding the new controller, it opens a window as shown in following figure.

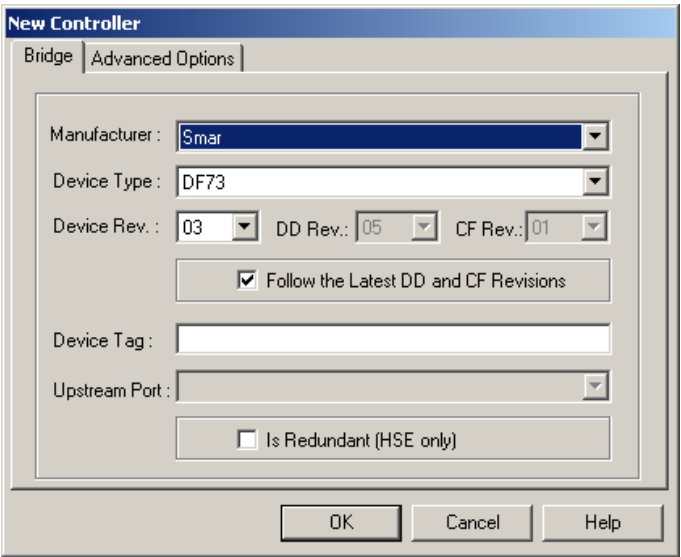


Figure 20. 10 – Setting the controller

Select **DF73** in the **Device Type** box. In the **Device Tag** box, type **DF73** or another tag, and click **OK** to conclude this task. As previously mentioned can be used DF95 or DF97 controllers in this area. The following steps are similar to those controllers.

IMPORTANT

Not all characters are valid when naming the elements.
The valid characters are:
A-Z a-z 0-9 { } [] () + -
The invalid characters are:
~ ` ! @ # \$ % ^ & * = | : ; , . < > ? / ' " \

TIP

Is possible to create an initial HSE configuration in an easier way using templates. In this case a configuration exists with some common steps created previously. For example, the steps 1 to 4 can be replaced by a creation of template through main menu **File →New→Predefined Area** choosing **DF73 Profibus Controller HSE 1x Profibus DP** or **DF73 Profibus Controller HSE 1x Profibus DP with FFB-1131**.

Adding Profibus Devices

Step 5

After inserting the controller for the area, it is possible to insert Profibus devices. Before, return to the **PROJ_DF73** window and right-click the controller, **DF73**. Then choose **New Network** option to configure the DF73 channels. See the Profibus bus creation in following figure.

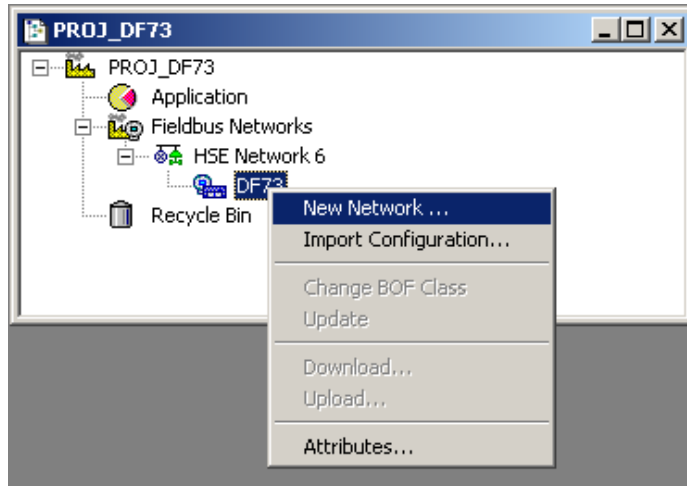


Figure 20. 11 – Creating the Profibus bus

When selecting **New Network**, the **Network Configurator** tool opens. The **Network Configurator** is the configuration tool for the Profibus network. Through it, the user has the structure of the Profibus network (devices, baud rate, and connection type). For further information about this tool, refer to its Help.




Figure 20. 12 – Network Configurator window

The controller DF73 has already been inserted to the configuration previously, so it is the Master device in the Profibus network. The Profibus devices that will be inserted to the network are named slaves.

IMPORTANT

The address 1 in the Profibus network is the default address of the DF73. It is not recommended to use the address 125 for moving the devices in the Profibus network. The address 126 can be used for acyclic parameterization of the field devices.

To insert slave devices, click the button  (**Insert Device**) placed on the toolbar, or through the menu **Insert→Device....** Place the mouse cursor just below where the Master device is placed and left-click. See the next figure.

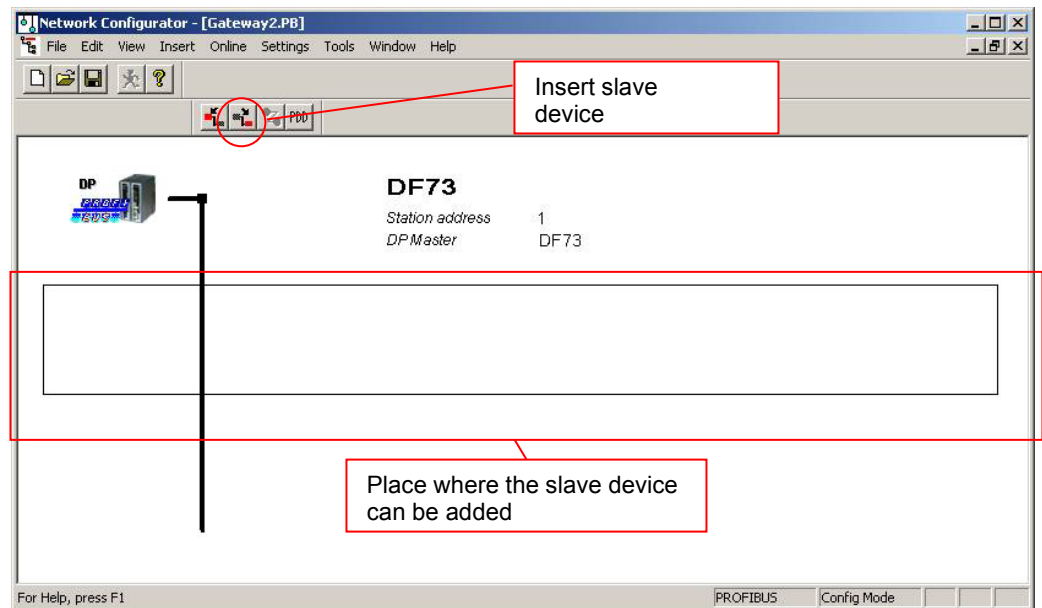


Figure 20. 13 – Placing the slave device in the Profibus network

The window for inserting slave devices will open, as shown below:

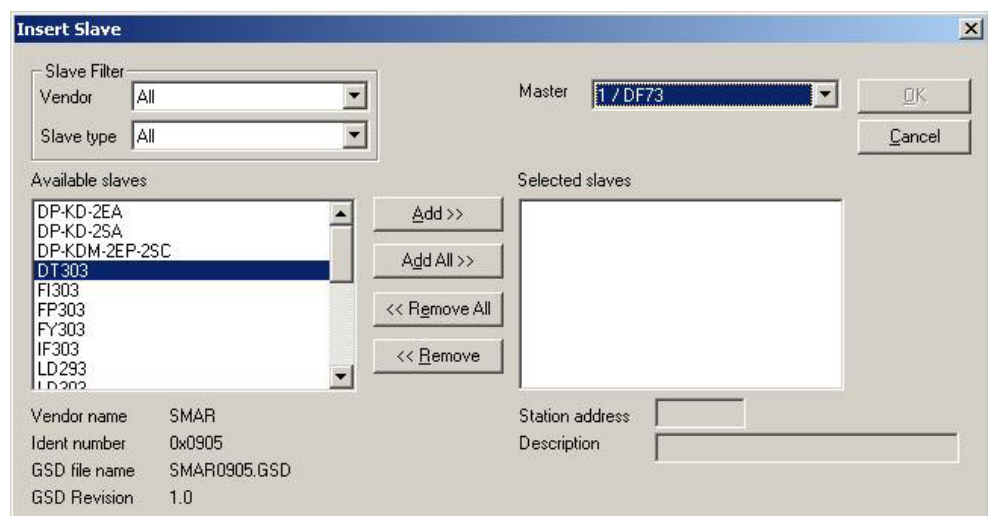


Figure 20. 14 – Inserting slave device

The user can select the slave device in this window. First the user has to obtain the configuration information of the Profibus slave device (refer to device's manual). Follow these steps to insert slave devices:

- A. Select the device in the **Available slaves** box;
- B. Click **Add** button;
- C. Type the address for the device in the **Station Address** field (this address has to correspond to the physical configuration of the device in the Profibus network. Further details about how to configure the device's address refer to device's manual);
- D. Type the tag for the device in the **Description** field. If the user does not write any tag for the device, the default tag will be **Device1**.
- E. Click **OK**.

NOTE

The address as well as the tags of the devices must be unique in the Profibus network and also in the **Studio302** Database (Workspace). For further information refer to **Studio302** manual.

When these operations finish the Profibus network will be as the following figure.

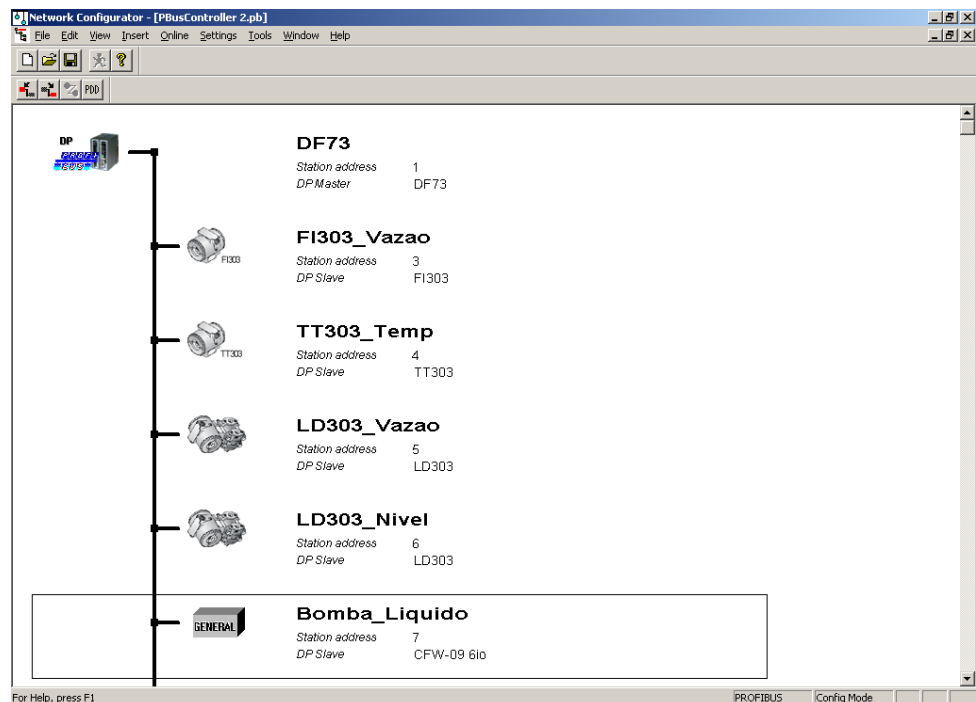


Figure 20. 15 – Profibus network created

Inserting Slave Devices that are no present in the “Available Slaves” list

The GSD files have the description of each Profibus device. They are provided by devices manufacturers. These files define the devices specific features in the Profibus network and their objects. The set of these files forms the devices database. The **SYSTEM302** already has a GSD database of the more common manufacturers, but is possible to add new devices in the **SYSTEM302** database.

If there is a device which will be in the Profibus network, but is not available in the devices list, the user has to contact the device manufacturer to take the GSD file, and also to take the *.bmp file (with the device icon) if it exists.

The GSD and BMP files obtained from the manufacturers should be copied respectively to the folders below and follow the path of the installation:

```
\SMAR\ NetWork Configurator\Fieldbus\Profibus\GSD
\SMAR\ NetWork Configurator\Fieldbus\Profibus\BMP
```

Setting Profibus Devices

Step 6

After creating the Profibus network, the devices have to be configured. For example, right-click the device corresponding to the flow meter. Choose the **Slave Configuration** option to configure the slave device.

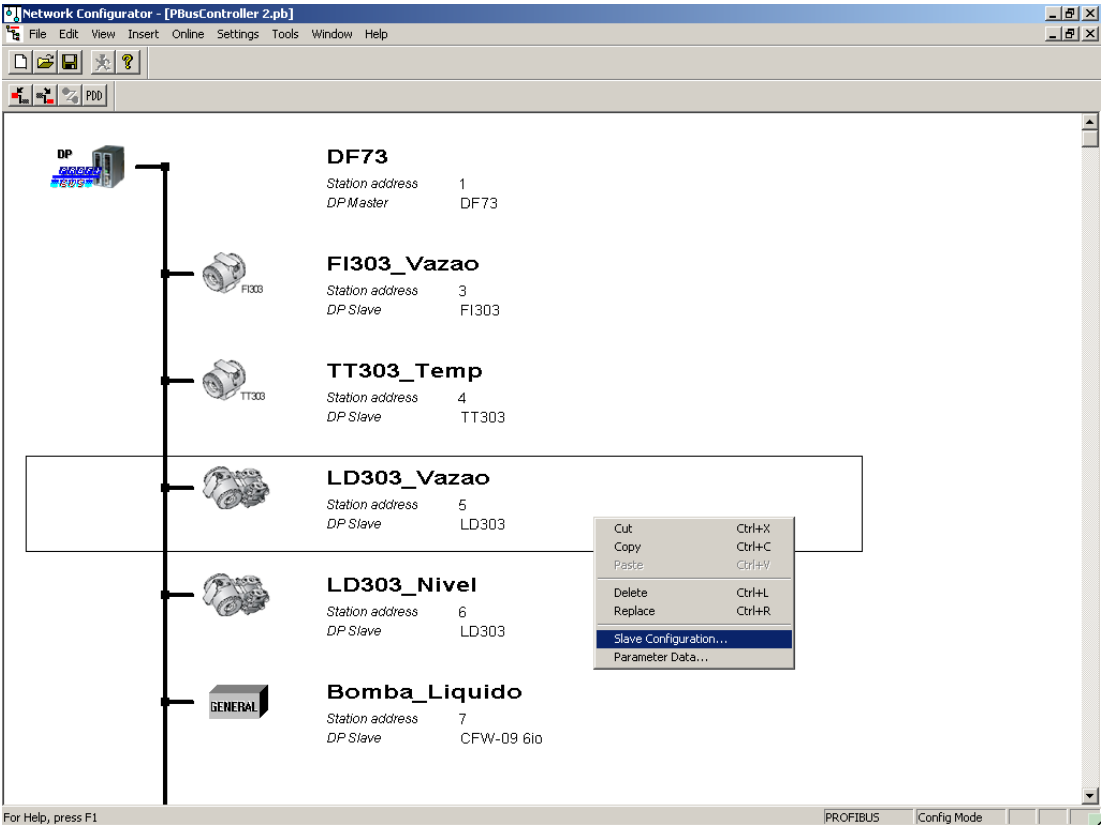


Figure 20. 16 – Setting the slave device

The configuration window also will open double-clicking the device.

The window for device configuration will open. See the figure below.

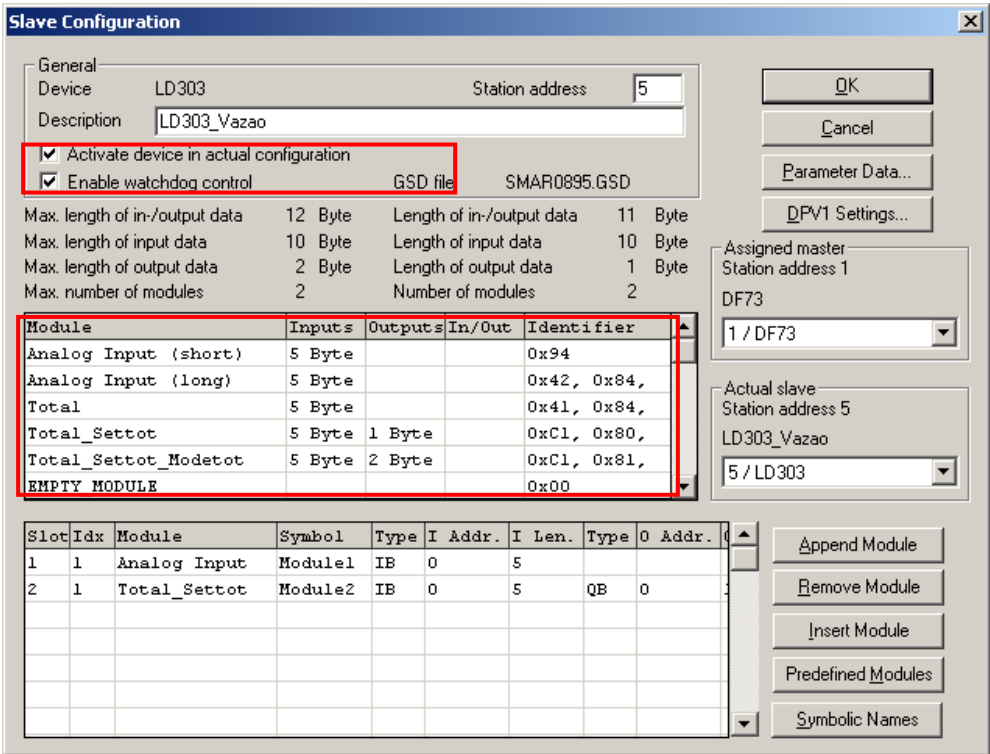


Figure 20. 17 – Setting the slave device

IMPORTANT

It is recommended that the *Enable watchdog control* option always is selected; therefore in the case of communication interruption the slaves, which were configured with this option, will attribute a safety value in them outputs after the configured watchdog time.

The data configuration is done by choosing the modules. This configuration can be fixed (if the device has a single module), or dependent of the modules selection and its quantity (if the device has more than one module).

The configurations for **PROJ_DF73** will be as follows:

The **FI303** supports up to 3 outputs channels (see FI303 manual). When inserting the **FI303_Vazao** is possible to see that it supports up to 3 modules. Since one valve will be controlled only one module will be used. For the others the **EMPTY_MODULE** will be chosen indicating that these modules will not be used. The manufacturer provides in the device's GSD file the available modules, the quantity and size of each module.

Slave Configuration

General
 Device: FI303 Station address: 3
 Description: FI303_Vazao
☒ Activate device in actual configuration
☒ Enable watchdog control GSD file: SMART999.GSD

Max. length of in-/output data	75 Byte	Length of in-/output data	25 Byte
Max. length of input data	45 Byte	Length of input data	15 Byte
Max. length of output data	30 Byte	Length of output data	10 Byte
Max. number of modules	3	Number of modules	3

Module	Inputs	Outputs	In/Out	Identifier
EMPTY_MODULE				0x00
eSP		5 Byte		0x82, 0x84,
SP		5 Byte		0xA4
eSP + RB + POS_D	7 Byte	5 Byte		0xC6, 0x84,
eSP + CHECKBACK	3 Byte	5 Byte		0xC3, 0x84,
eSP + RB + POS_D + CB	10	5 Byte		0xC7, 0x84,

Assigned master
 Station address 1
 DF73
 1 / DF73

Actual slave
 Station address 3
 FI303_Vazao
 3 / FI303

Slot	Idx	Module	Symbol	Type	I Addr.	I
1	1	eSP+RB+RCASIN+RCASOUT+POS_D+CB	Module1	IB	0	15
2	1	EMPTY_MODULE	Module2			
3	1	EMPTY_MODULE	Module3			

Buttons: OK, Cancel, Parameter Data..., DPV1 Settings..., Append Module, Remove Module, Insert Module, Predefined Modules, Symbolic Names

Figure 20. 18 – Setting the FI303_Vazao

For the TT303 - **TT303_Temp** – was chosen only one module, but it has capacity for two temperature inputs (but only one measurement is done). The **EMPTY_MODULE** was added indicating that the second channel will not be used.

Slave Configuration

General

Device: TT303 Station address: 4

Description: TT303_Temp

☒ Activate device in actual configuration

☒ Enable watchdog control GSD file: SMAR089A.GSD

Max. length of in-/output data: 10 Byte Length of in-/output data: 5 Byte

Max. length of input data: 10 Byte Length of input data: 5 Byte

Max. length of output data: 0 Byte Length of output data: 0 Byte

Max. number of modules: 2 Number of modules: 2

Module	Inputs	Outputs	In/Out	Identifier
Analog Input (short)	5 Byte			0x94
Analog Input (long)	5 Byte			0x42, 0x84,
EMPTY_MODULE				0x00

Slot	Idx	Module	Symbol	Type	I Addr.	I Len.	Type
1	1	Analog Input (short)	Module1	IB	15	5	
2	1	EMPTY_MODULE	Module2				

Assigned master: Station address 1, DF73, 1 / DF73

Actual slave: Station address 4, TT303_Temp, 4 / TT303

Buttons: OK, Cancel, Parameter Data..., DPV1 Settings..., Append Module, Remove Module, Insert Module, Predefined Modules, Symbolic Names

Figure 20. 19 – Setting the TT303_Temp

For the **LD303_Vazao** two modules were chosen. The first will read the flow rate, and the second will do the totalization. This feature is from the transmitter and it will avoid that this calculation is done by the controller.

Slave Configuration

General

Device: LD303 Station address: 5

Description: LD303_Vazao

☒ Activate device in actual configuration

☒ Enable watchdog control GSD file: SMAR0895.GSD

Max. length of in-/output data: 12 Byte Length of in-/output data: 11 Byte

Max. length of input data: 10 Byte Length of input data: 10 Byte

Max. length of output data: 2 Byte Length of output data: 1 Byte

Max. number of modules: 2 Number of modules: 2

Module	Inputs	Outputs	In/Out	Identifier
Analog Input (short)	5 Byte			0x94
Analog Input (long)	5 Byte			0x42, 0x84,
Total	5 Byte			0x41, 0x84,
Total_Settot	5 Byte	1 Byte		0xC1, 0x80,
Total_Settot_Modetot	5 Byte	2 Byte		0xC1, 0x81,
EMPTY_MODULE				0x00

Slot	Idx	Module	Symbol	Type	I Addr.	I Len.	Type
1	1	Analog Input (short)	Module1	IB	20	5	
2	1	Total_Settot	Module2	IB	25	5	QB

Assigned master: Station address 1, DF73, 1 / DF73

Actual slave: Station address 5, LD303_Vazao, 5 / LD303

Buttons: OK, Cancel, Parameter Data..., DPV1 Settings..., Append Module, Remove Module, Insert Module, Predefined Modules, Symbolic Names

Figure 20. 20 – Setting the LD303_Vazao

The configuration for the **LD303_Nivel** is the same one for **TT303_Temp**, because the added module will indicate the tank level, it is not necessary other measurements. After this, add the **EMPTY_MODULE**.

[illegible]

Figure 20. 21 – Setting the LD303_Nivel

The **Bomba_Liquido** device supports only one module type, thus add one module (see CFW09 manual), which will correspond to the 16 inverter status bits, 16 control bits, an integer used for set point, an integer used for speed reference, and other integers which vary according to the device's model and device's version.

Slave Configuration

General

Device: CFW-09 Gio Station address: 7

Description: Bomba_Liquido

☒ Activate device in actual configuration

☒ Enable watchdog control GSD file: CFW096IO.GSD

Max. length of in-/output data: 416 Byte Length of in-/output data: 24 Byte

Max. length of input data: 244 Byte Length of input data: 12 Byte

Max. length of output data: 244 Byte Length of output data: 12 Byte

Max. number of modules: 24 Number of modules: 1

Module	Inputs	Outputs	In/Out	Identifier
6wIn/6wOut	6 Word	6 Word		0x55, 0x65

Slot	Idx	Module	Symbol	Type	I Addr.	I Len.	Type	O Addr.	O L
1	1	6wIn/6wOut	Module1	IW	0	6			
1	2	6wIn/6wOut	Module1				QW	0	6

Assigned master: Station address 1 DF73 1 / DF73

Actual slave: Station address 7 Bomba_Liquido 7 / CFW-09 Gio

Buttons: OK, Cancel, Parameter Data..., DPV1 Settings..., Append Module, Remove Module, Insert Module, Predefined Modules, Symbolic Names

Figure 20. 22 – Setting the Bomba_Liquido

When inserting devices finishes, the mapping must be done. It consists of mapping the values, bytes and data to be reading/written from the devices to the master memory. The procedure will be described in the following topic.

After selecting the inputs and outputs for the Profibus devices, set the baud rate for the Profibus network. Click the master device and choose **Settings→Bus Parameter** on the toolbar. The dialog box will open:

Bus Parameter

Baud rate: 500 kBits/s

Optimize: By User


Buttons: OK, Cancel, Edit...

Figure 20. 23 – Selecting the baud rate for the Profibus network

Select the baud rate in the **Baud Rate** list, in the **Optimize** field select **By User**. Click **Edit...** if is necessary to configure the bus parameters. If is not necessary to configure the bus parameters select **Standard** and click **Ok**. The communication parameters will be described in the Profibus network time parameters topic, for further details, please consult it.

NOTE

The baud rate and communication parameters are dependent of the slaves in the network. The device's GSD file has the supported baud rates. The configuration tool will warning if some slave does not support the selected baud rate.

 Save the configuration before exit from **Network Configurator**.

Step 7

When the previous steps were finish, the Profibus network configuration can be saved, and then, exit from **Network Configurator**. The **Mapping Tool** launches automatically. With this tool is possible to map the points of Profibus network in available points which will be used in ladder logic or in function blocks logic. This tool configures all characteristics of Profibus points (user tags, data types, scales, etc). See the following figure.

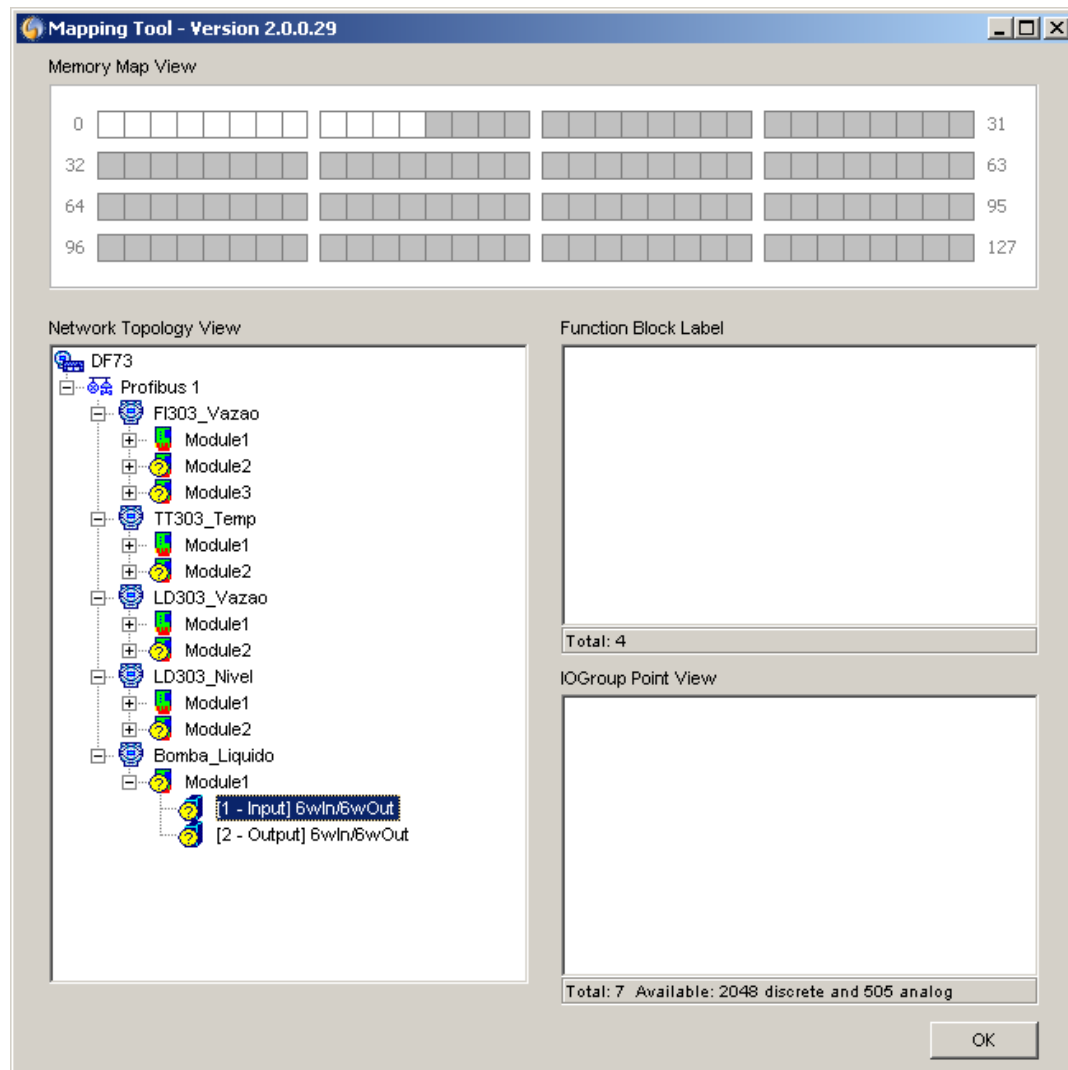



Figure 20. 24 – Mapping Tool window

In the last figure can be seen the **Network Topology View** window with the devices which were inserted in the Profibus network with their modules. In the **Function Block Label** window are the points which will be seen at **Syscon**, and the **IOGroup Point View** window are the points which will be seen at **LogicView for FFB**. The purpose is to configure each network “point” (or byte) in their respective data types.

The items with the symbol  have to be configured. The user has to refer to the Profibus DP devices manuals, such as CFW09 manual, to know each byte function of the Profibus module. The manual has the bytes' meaning (if they are used as analog or digital).

According to the drive manual the inputs and outputs bytes have the following attributions:

Digital Inputs			Digital Outputs		
Byte 0	Bit 0	ON Status 0=Stopped / 1=Rotating	Byte 0	Bit 0	Mask for Rotate/Stop 1= Enabled / 0=Disable Rotate/Stop
Byte 0	Bit 1	General Enable 0=Enabled / 1=Disabled.	Byte 0	Bit 1	Mask for General Enable 1= Enabled / 0=Disable General Enable
Byte 0	Bit 4	Local and Remote Status 0=Local / 1=Remote	Byte 0	Bit 4	Mask for Select Local and Remote 1= Enabled / 0=Disable Local/Remote
Byte 0	Bit 5	Power fault 0=Without power fault / 1=With	Byte 0	Bit 7	Mask for Reset 1= Enabled / 0=Disable Reset
Byte 0	Bit 7	Active Error 0=No / 1=Yes	Byte 1	Bit 0	Rotate/Stop Command 1=Rotate / 0=Stop
			Byte 1	Bit 1	General Enable Command 1= Enabled / 0= General Disable
			Byte 1	Bit 4	Local and Remote Command 0=Local / 1=Remote
			Byte 1	Bit 7	Reset Command 0=No Reset / 1=Reset

Analog Inputs		Analog Outputs	
Byte 1	Error Code (0-255)		
Bytes 2 and 3	Motor Speed (0-8192)	Bytes 2 and 3	Speed Reference (0-8192)

Click the item which will be configured (for example **Bomba_Liquido->Module1->[1 – Input] 6wIn/6wOut**) and in the window upper side will appear the available bytes. See the next figure.

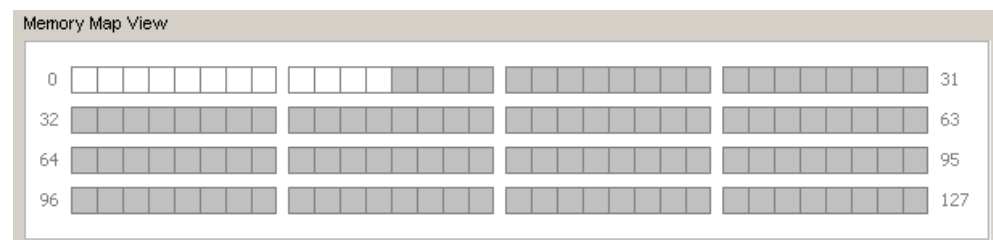


Figure 20. 25 – Available bytes

Click the byte, and the following window will appear.

The figure shows a 'Map Wizard' dialog box. It has a title bar with a close button. The main area contains the text 'Select fields values to map.' Below this, there are three fields: 'Data type' with a dropdown menu showing 'Bit', 'Block type' with a dropdown menu showing 'None', and 'Group tag' with an empty text box. At the bottom, there are three buttons: '< Back', 'Next >', and 'Cancel'.

Figure 20. 26 – Configuring the data type

Choose the variable data type of the slave device (**Data type**). The available types will depend on the chosen device. To **Bomba_Liquido** the type is **Bit**.



Figure 20. 27 – Configuring the input discrete points

The following steps have instructions about how to use the points, which were mapped in the **Network Configurator**, in the control logic. As above mentioned there are two ways to map inputs and outputs – ladder logic or function blocks.

Mapping Profibus IO Points to be Used in Ladder

Step 8

First the points will be mapped to be used in ladder, thus the **Block Type** (None) option will not be configured. The user can give a tag to group in **Group tag** option. If **Bit** option is selected the following window will appear.



Figure 20. 28 – Configuring the bits

Select the bits which are necessary to the application, click **Finish** and the configured points will appear in the **Mapping Tool** window.

When analog data type was chosen the scale parameters for this point has to be configured. The speed and its reference for the drive have to be configured as **unsigned16** and the scale has to be 0-8192 for **sensor0-sensor100**. The error code has to be mapped as **unsigned8** and without scale (**PV0-PV100** equals to **sensor0-sensor100** within 0-255)

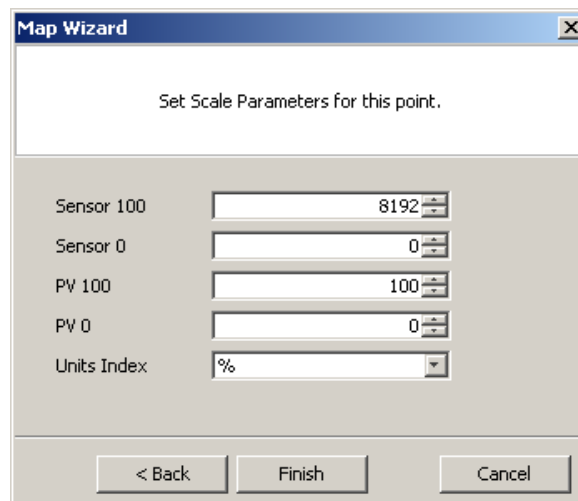


Figure 20.29 – Setting the scale parameters

Right-clicking this point and choosing **Delete** it can be deleted, or some attributes can be defined for better identification through the **Attributes** option.

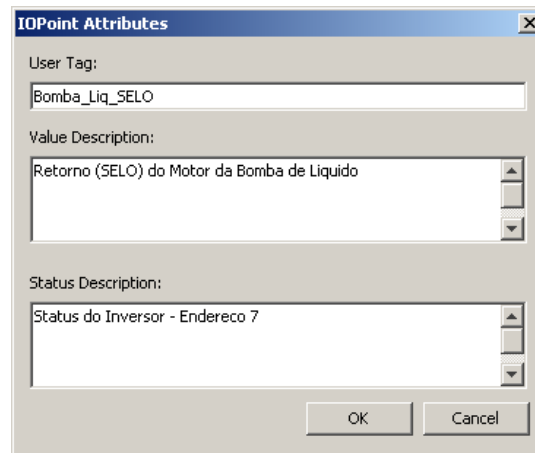


Figure 20.30 – Changing the point attributes

Fill the necessary fields, and click **OK**.

After mapping and configuring all points, click **OK** to exit from **Mapping Tool**.

After finish the mapping, a FFB block has to be inserted in the configuration using the **Syscon**. This block is necessary to edit the ladder logic. For further details refer to Adding Function Blocks section or **Syscon** manual. Right-click FFB block, which was added, and then click **Define Parameters**.

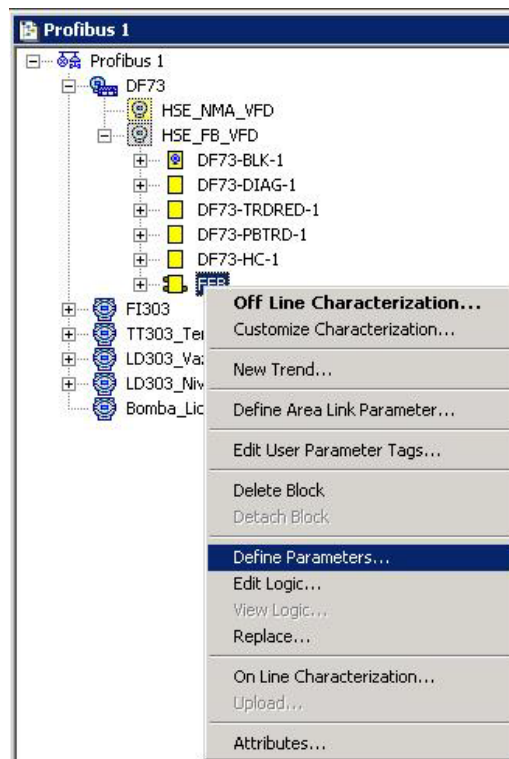


Figure 20. 31 – Defining FFB parameters (1)

The following window will appear.

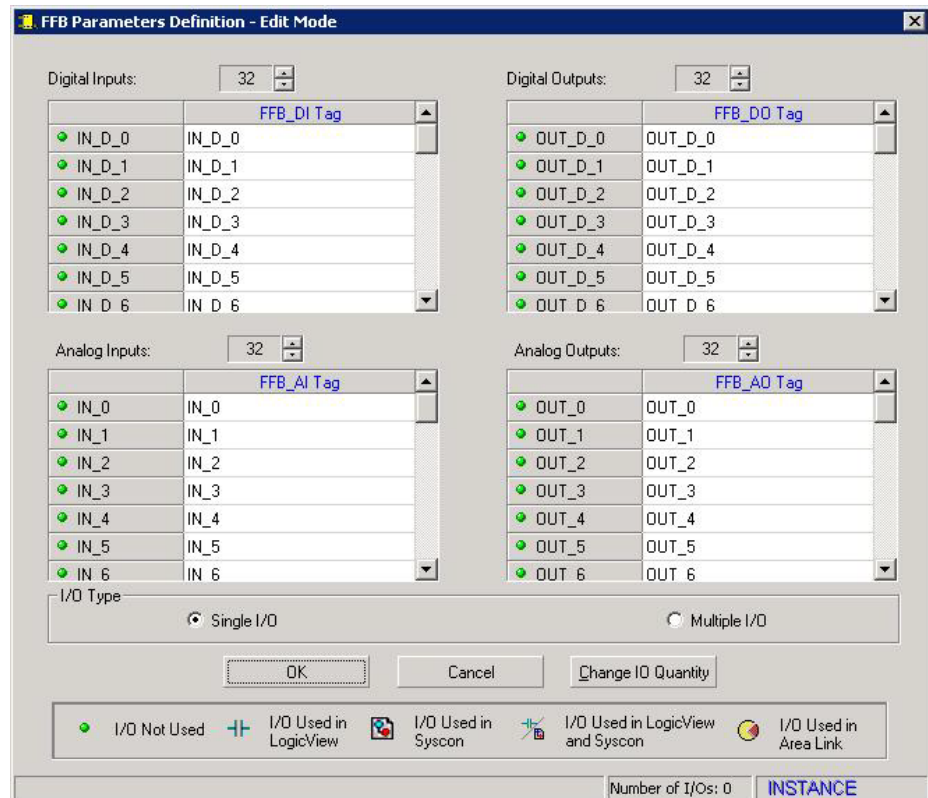


Figure 20. 32 – Defining FFB parameters (2)

If necessary define the inputs and outputs of FFB. Otherwise just click **OK**.



Return to **Syscon**, save the configuration, and do an **Export tags**.

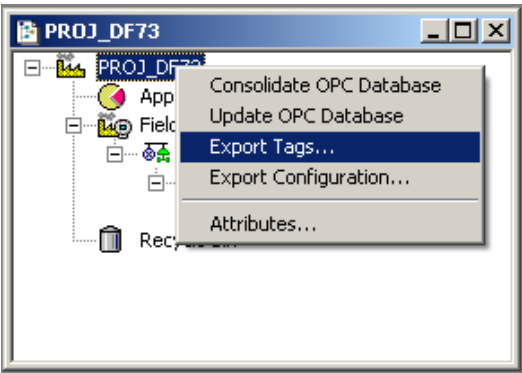


Figure 20. 33 – Export tags

After finish this operation, right-click the FFB block, and then select **Edit Logic**. The **LogicView for FFB** will launch, and the ladder logic can be edited.

For the first time the logic is edited the **Refresh Data** command has to be executed thus the IO points configured in **Mapping Tool** will be updated in the ladder. Right-click **Network I/O** in the **Hierarchy** window of the **LogicView for FFB**. See the next figure.

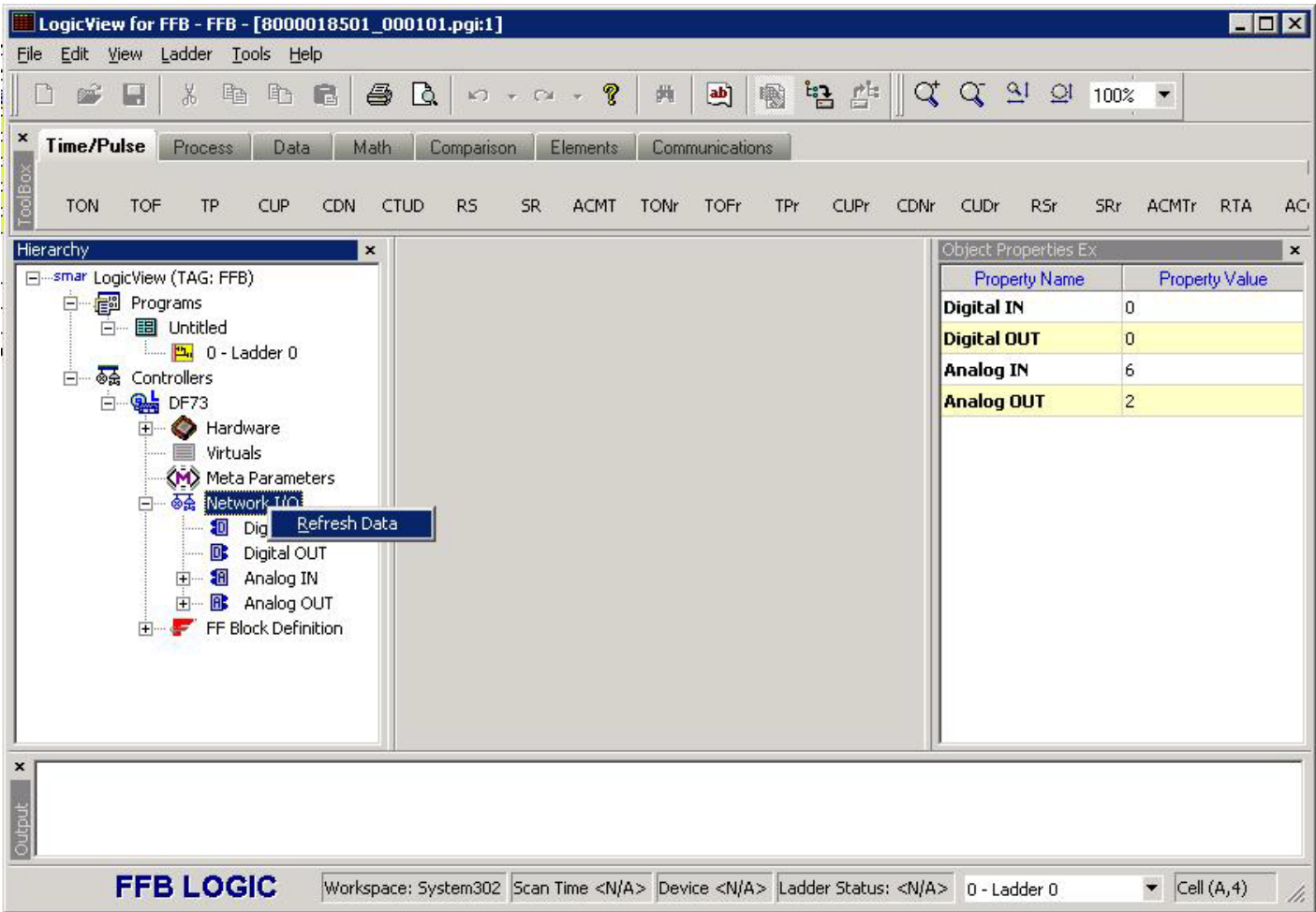


Figure 20. 34 – Refresh data

The NetIO points can be seen in the **Hierarchy** window. See the next figure.

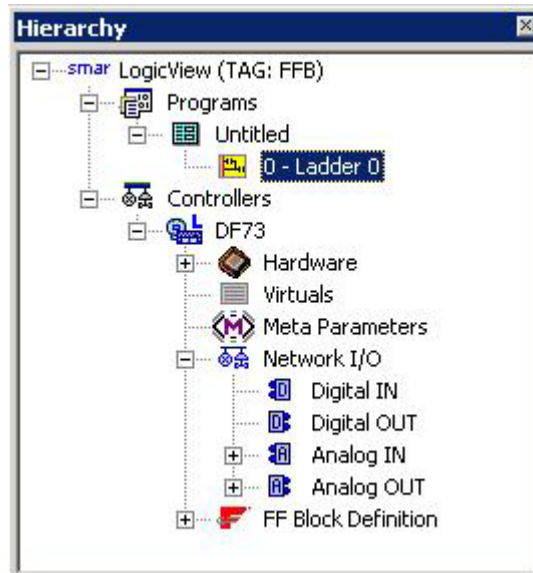



Figure 20. 35 – Network I/O at LogicView for FFB

After that procedure the IO point of the Profibus network are already available to be used in the logic. The user has a complete function library which can be used in the logic. For further details see the **LogicView for FFB** manual.

See in the following figure the EPID usage in the **LogicView for FFB** to implement the liquid pump control. All Profibus IO points appear with the symbol  in the ladder logic.

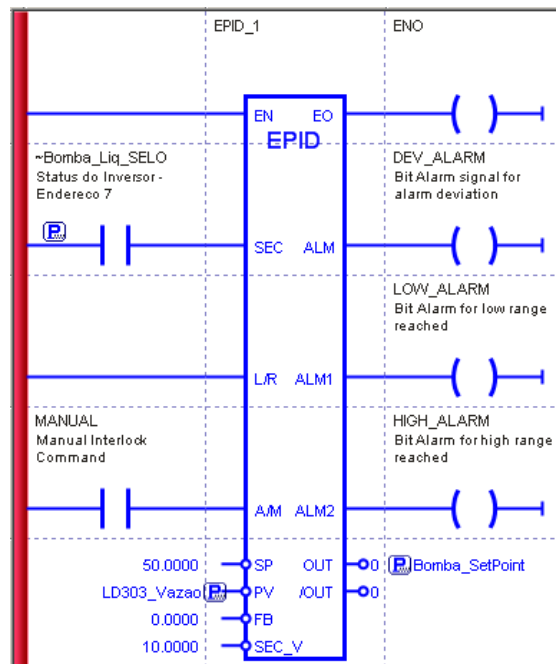



Figure 20. 36 – Editing logic

After ladder configuration the user can download the configuration to the device. Another option is to configure the function blocks at **Syscon**. If you also want to work with function blocks is necessary to follow the steps 9 to 12 to edit them. In this topic only the ladder was used to configure the ladder. Before downloading the configuration you have to save it in the **LogicView for FFB**. Exit from it, and return to **Syscon**.

To start a communication with the devices, first, is necessary to commission the controller. In this way the tags, IDs and addresses of each device will be assigned correctly. If this procedure was not done, the **Syscon** will detect the uncommissioned device and its download will be canceled. When the devices commissioning finishes the download process can start. The download process can be

done, for example, returning to **Proj_DF73** window, right-clicking **Fieldbus Networks** icon, , and then, select **Download** option. For further details about commissioning and possible download types refer to **Syscon** manual.

Mapping Profibus IO Points to be Used in Function Blocks

In the same way of the configuration done through ladder, the configuration can be done with function blocks. In this topic the same Profibus IO points of the step 8 will be mapped, but now the function blocks will be used. As above mentioned, for the network creation and devices insertion the steps are common up to step 7.

Step 8'

In the **Mapping Tool** window, to map the IO points in blocks, is necessary to configure the **Block Type** field as you can see in the following figure. When **None** is chosen, as in the step 8, the mapping will be done in ladder.

To map the **Bomba_SetPoint** output, for example, choose the **Unsigned 15** data type. And then, in **Block type** choose **Multiple Digital Output**. Give a tag to the block in **Block Tag**.

NOTE

Every point of Profibus network is already configured automatically to work with ladder. To map in function blocks the user has to choose this option configuring the **Block Type** field of the **Map Wizard** window of **Mapping Tool** with the respective IO block.

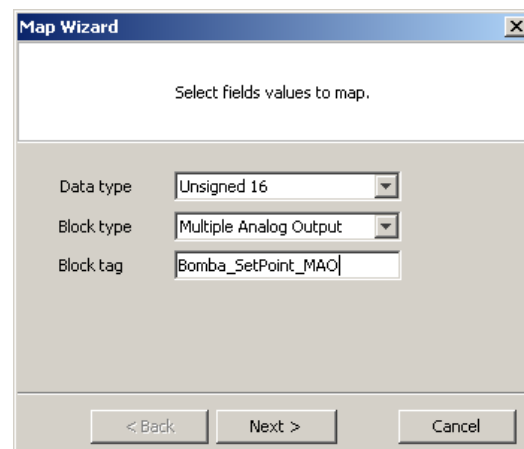


Figure 20. 37 – Configuring the data type

Click **Next**, and a window for bits configuration will open, select bit 1. Click **Finish**, and the configured point will appear in the **Mapping Tool** window.

NOTE

The **Mapping Tool** (MT) always tries to minimize the number of created blocks in the configuration. So, for a same slave device, if was not chosen any block tag the MT will allocate the point in an already existent block (the allocation availability will be verified, that is, the block has to be the same type and the block has to have empty points).

If the user informs an existent block tag, the MT will use this block to allocate the point. If the user informs a block tag that does not exist, the MT will create a new block.

Right-clicking this point and choosing **Delete** it can be deleted, or some attributes can be defined for better identification through the **Attributes** option.

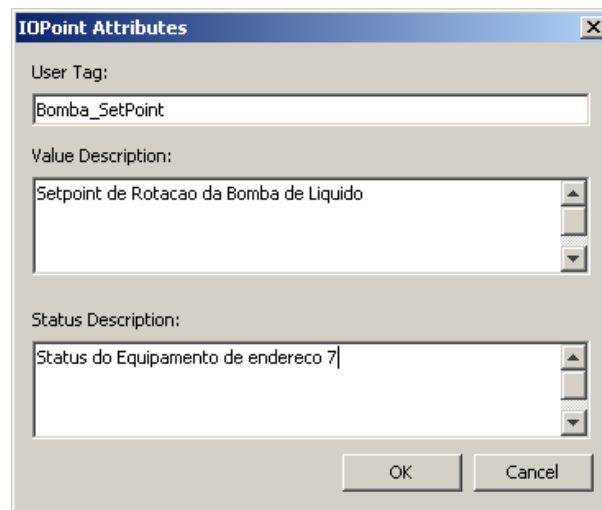


Figure 20.38 – Changing the point attributes

Fill the necessary fields, and click **OK**.

After mapping and configuring all points, the **Mapping Tool** will be as in the following figure.

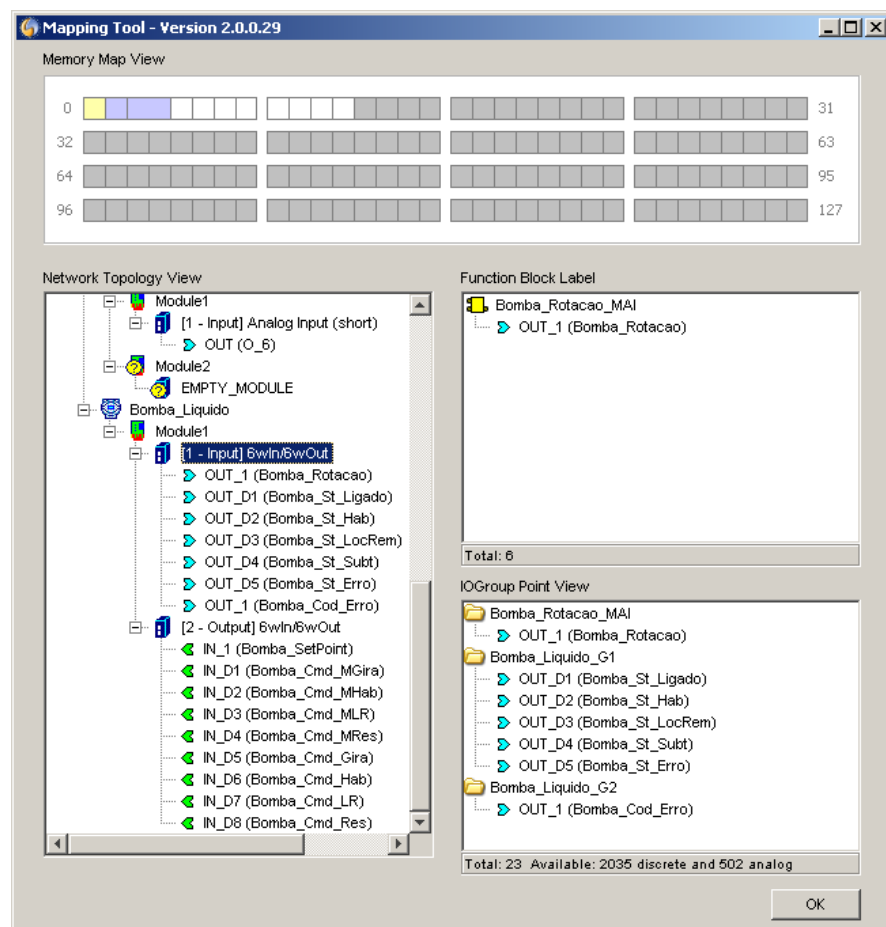


Figure 20.39 – Mapped points in the Mapping tool

Click **Ok** to exit from **Mapping Tool**. Return to **Syscon**. The mapped blocks are already inserted in the area.

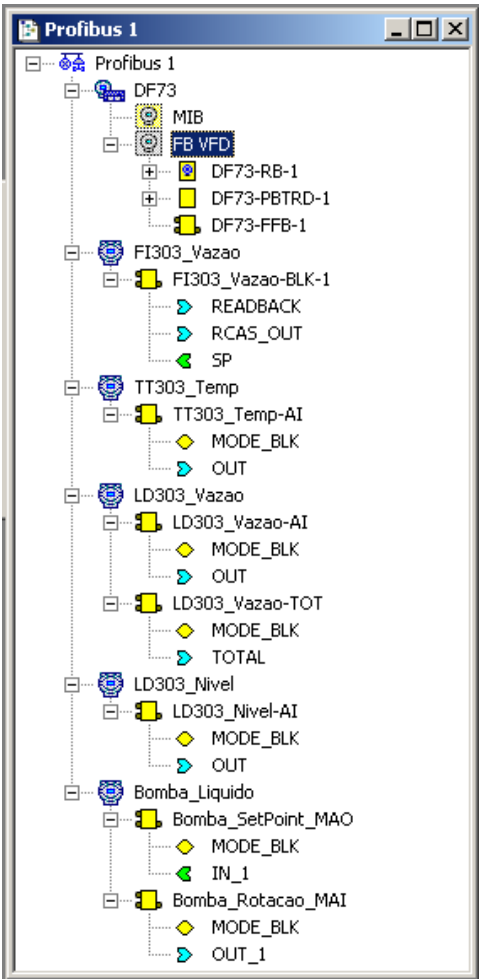


Figure 20. 40 – Mapped function blocks in the Syscon

The device's attributes can be verified right-clicking it. The following window will appear, and the information cannot be modified.

The screenshot shows a dialog box titled "Device Attributes". It contains three input fields:

- Vendor ID :** HMS Fieldbus Systems AB
- Product Code :** CFW-09 Gio
- Address:** 7

 At the bottom right of the dialog is an "OK" button.

Figure 20. 41 – Verifying the device's attributes

The block's attributes can be verified right-clicking it. The following window will appear, and the information cannot be modified.

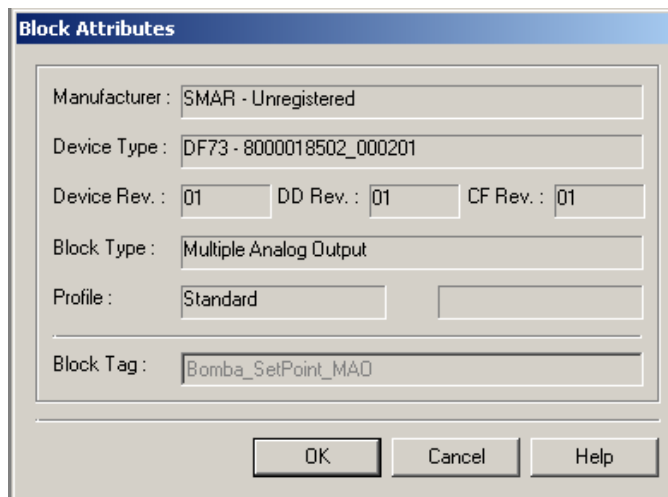


Figure 20.42 – Verifying the block's attributes

Adding Other Function Blocks

Step 9

Now the user can add Function Blocks that will be used in the strategy.

To insert a new FB (Function Block), click the sign , right-click the **FB VFD** (Virtual Field Device) icon and choose **New Block**. The FB VFD is responsible for the data management.

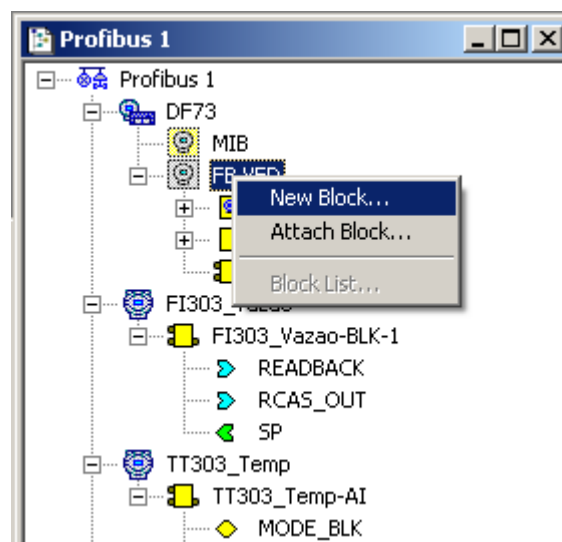


Figure 20.43 – Inserting new function blocks

The **Function Block Characterization** dialog box will appear. The **Block Type** option has the available function blocks.

Select the block in the **Block Type** field and type its tag in the **Block Tag** field:

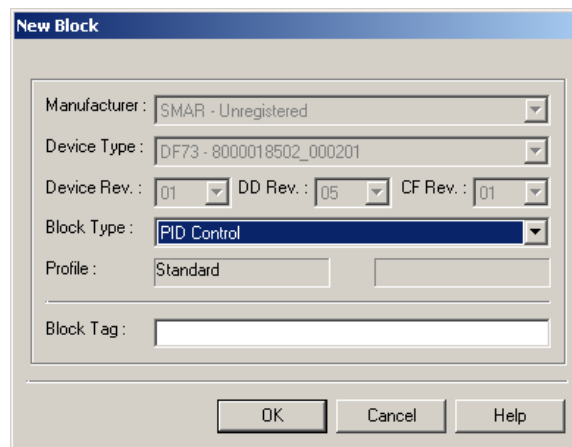


Figure 20.44 – Selecting the function blocks that will be added to the devices

For this example, the PID block should be added to the DF73. The Profibus network can be seen in the next figure.

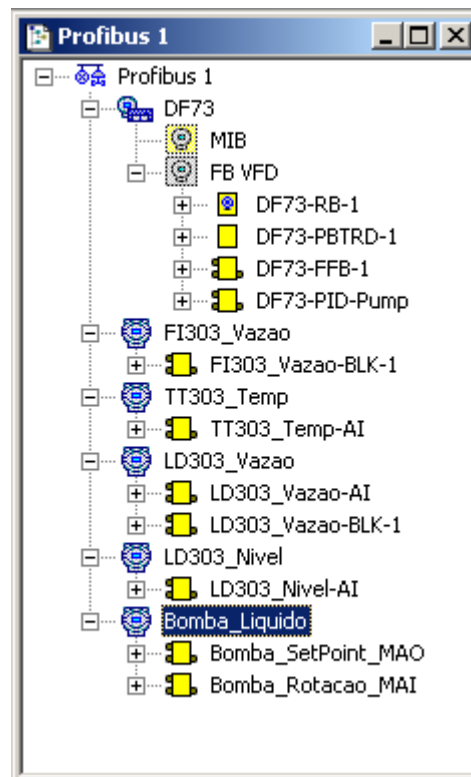


Figure 20.45 – Profibus network with devices and function blocks inserted

Creating New Process Cells

Step 10

The user can create the strategy for the **Application (Logical Plant)**. First, it is necessary to create a new process cell.

The **Logical Plant** can be divided in several areas according to the area.

To create a new process cell, right-click the **Application** icon and choose **New Process Cell** item.

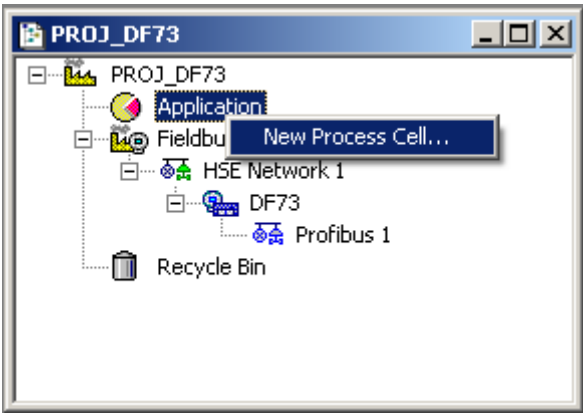


Figure 20. 46 – Inserting the Process Cell

The dialog box to name the **Process Cell** will open:

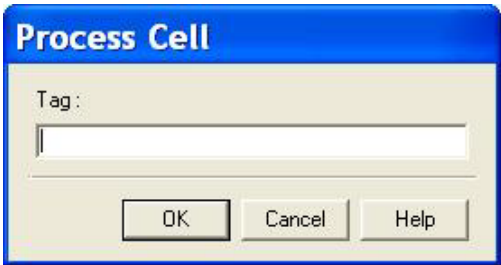


Figure 20. 47 – Naming the Process Cell

If the user needs to give a name for the Process Cell, enter the name in the tag box, and click **OK**. The default tag is **Process Cell 1**. This number will increase when new process cells are created. To create others, the user has to repeat the procedure.

After inserting a Process Cell, the **Proj_DF73** window will be according to the following figure.

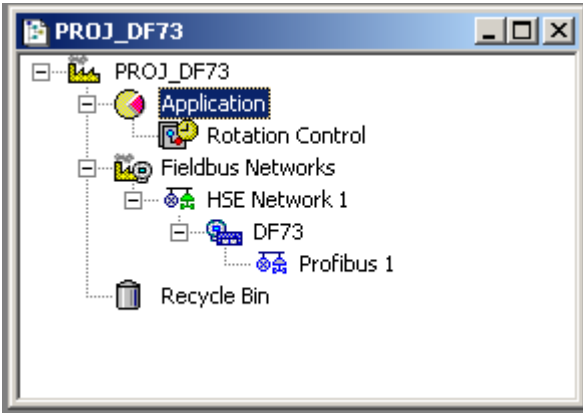


Figure 20. 48 – Area window with the Process Cell added

NOTE

The user must remember the **Application** is just a virtual division, and its purpose is to divide a large project.
For example: if the plant has 2 networks, they can be called **Process Cells** in the Syscon. One **Application** can have several **Process Cells**, but the **Process Cells** can not be in more than one **Application**.

Creating a Control Module

Step 11

Right-click the **Process Cell 1** icon, and choose **Expand** item.

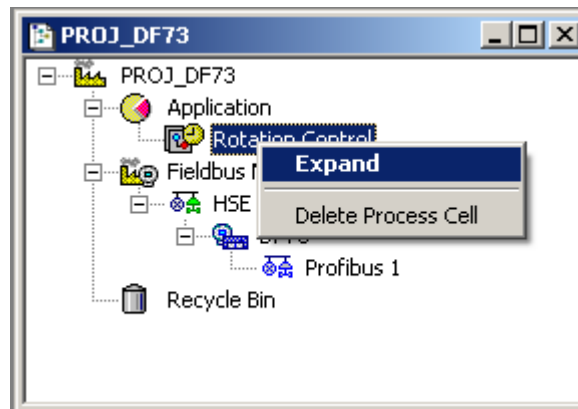


Figure 20. 49 –Creating a control module

To arrange the screen, click the **Process Cell 1** window. So, choose **Window** menu on the **Syscon** toolbar, and then **Tile** option.

As following, return to the **Process Cell 1** window. Right-click the **Process Cell 1** item, and choose **New Control Module**. The next figure shows creating the New Control Module.

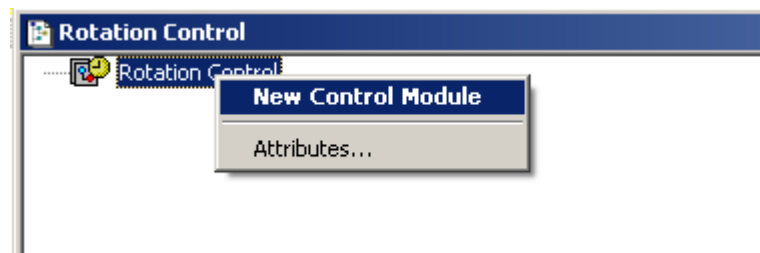


Figure 20. 50 – Creating a New Control Module

The **New Control Module** dialog box will appear. Name it with the tag related to the application. To conclude this task click **OK**.

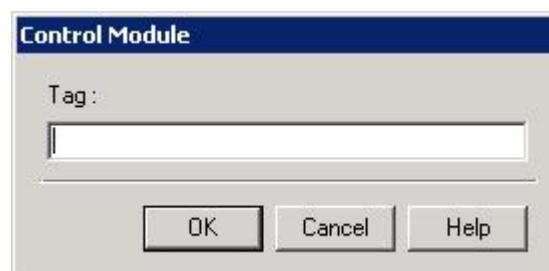


Figure 20. 51 – Naming the Control Module

IMPORTANT

Remember that not all characters are valid when naming the elements with tags.

Inserting Blocks to the Control Module

Step 12

The user can insert (attaching) the blocks for the corresponding devices in the **Logical Plant**. Right-click the **Control Module 1** icon, and then select **Attach Block** option, as shown in the following figure.

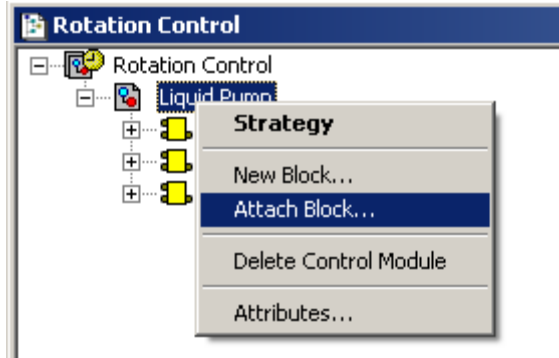


Figure 20.52 – Attaching blocks to the Control Module

The **Attach Block** dialog box will open as shown in the following figure.

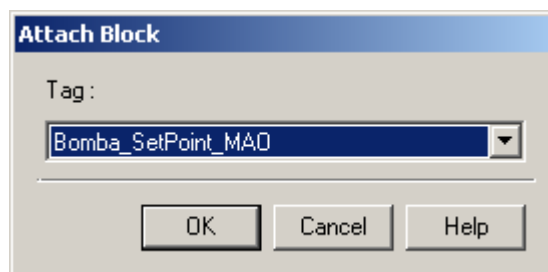


Figure 20.53 – Inserting blocks to the Control Module

The available function blocks for the application are in the **Attach Block** box. For the aimed control, the function blocks that must be inserted will appear in this box. So, select them, one by one, and click **OK**.

For the example of valve opening control, at the end of Attach Block process, the **Control Module** will be as in the next figure.

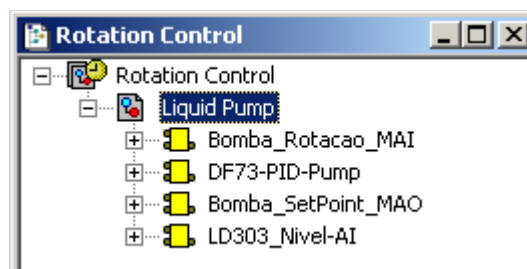


Figure 20.54 – Blocks inserted in the Control Module

Another way to attach the blocks is left-clicking the element and dropping it to the window.

Configuring the Control Strategy

Step 13

Now the user is ready to develop the control strategy.

First, right-click the **Control Module 1** icon, and then select **Strategy** item. The **Strategy** window will appear as shown in the following figure.

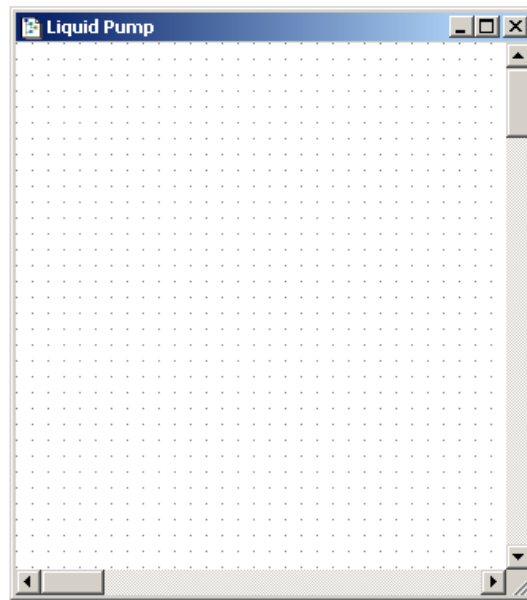


Figure 20. 55 – Strategy window

The Control Module attributes can be changed by right-clicking the **Control Module 1** icon, and then select **Attributes** option. Do the necessary changes, and click **OK**. For further information about Control Module refer to the **Syscon** manual.

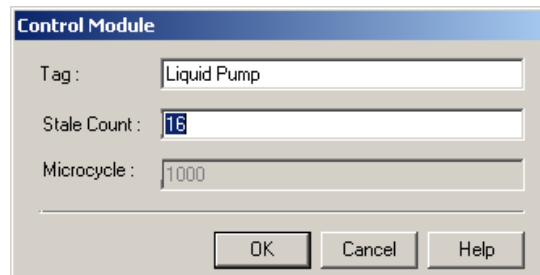


Figure 20. 56 – Changing the control module attributes

At this moment there are 3 or 4 windows opened in the **Syscon**. To arrange the screen, click the title of the **Proj_DF73** window. On the toolbar, choose **Window → Tile**.


If the user does not have a monitor upper than 17", it is recommended to minimize the strategy window. Thus, it is possible to see the whole area.

The strategy window offers several tools for drawing. Refer to the **Syscon** manual for further details.

Adding Blocks in the Strategy Window

Step 14

Now the function blocks can be added to the **Liquid Pump** window.

In order, click the first block,  Bomba_Rotacao_MAI, and drop it into the strategy window. A function block will be created automatically.

The next figure shows the function block added to the strategy window:

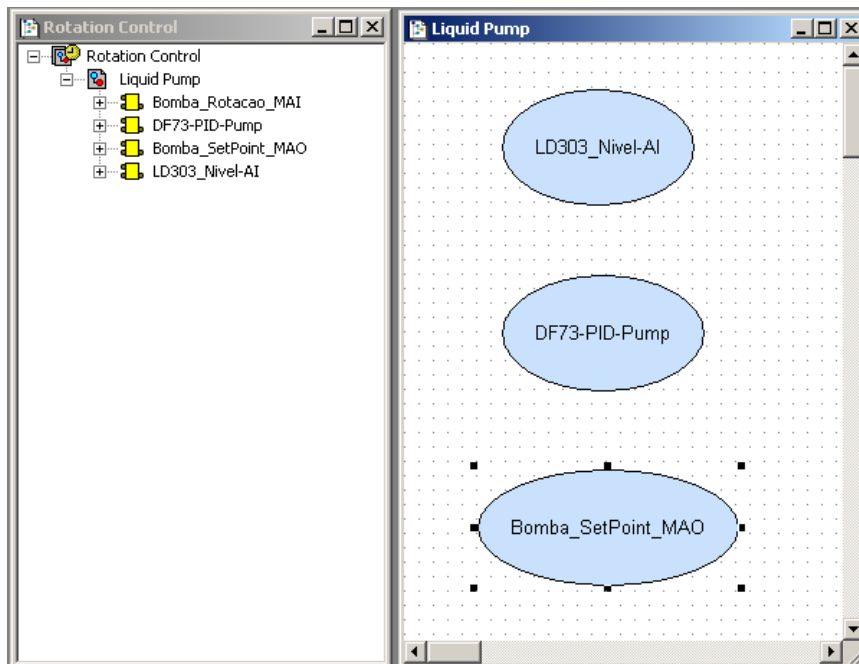



Figure 20. 57 – Block inserted in the strategy window

The drag-and-drop procedure must be repeated for the other blocks.

Linking the Blocks

Step 15

There is a specific tool to link the blocks, the **Link** button, , on the **Strategy** toolbar. For the aimed control link the **OUT** output of the LD303_NIVEL-AI (where the level measurement was mapped) to the **IN** input of the PID block.

Click this button on the toolbar, and then in the LD303_NIVEL-AI function block. The dialog box for linking the input and output parameters will appear. Select **OUT** block output, and then click **OK** as shown in the following figure.

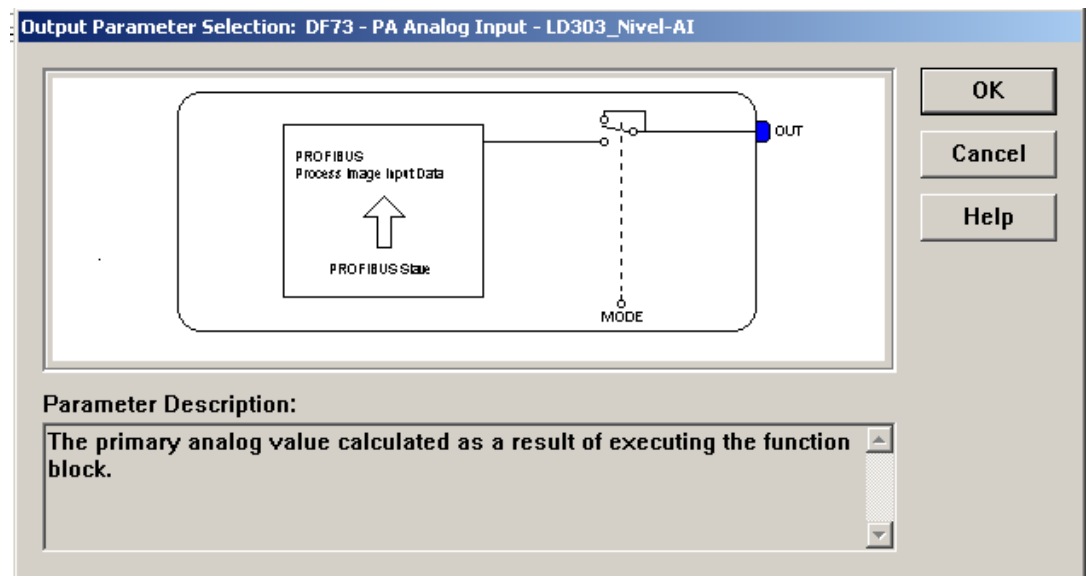


Figure 20.58 – Linking the function blocks

The user also does the fast link procedure just right-clicking the function block, and does the necessary links.

After linking the specified parameters above, the strategy window will be as shown in the following figure.

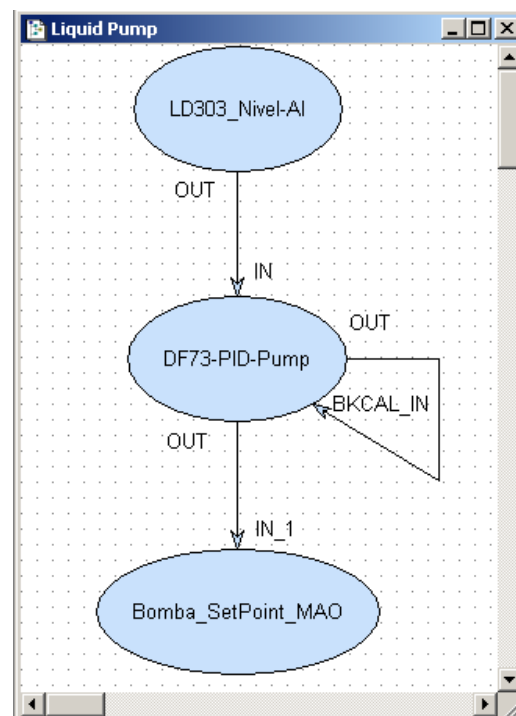


Figure 20.59 – Linking the Blocks

Block Characterization

Step 16

The function blocks that are in the area must be set according to the application for them. So, it is necessary to do the block characterization.

The online and offline modes are possible for the block characterization. In the offline mode, the parameters are set before starting the communication among the devices. The online characterization is executed directly in the devices when the plant is already communicating, and the download to the devices was done.

To change the function block parameters, consider the following topics:

1. In the Strategy window

Select the block to characterize. Right-click it and select **Off Line Characterization**, or double-click it. The next figure shows the block offline characterization.

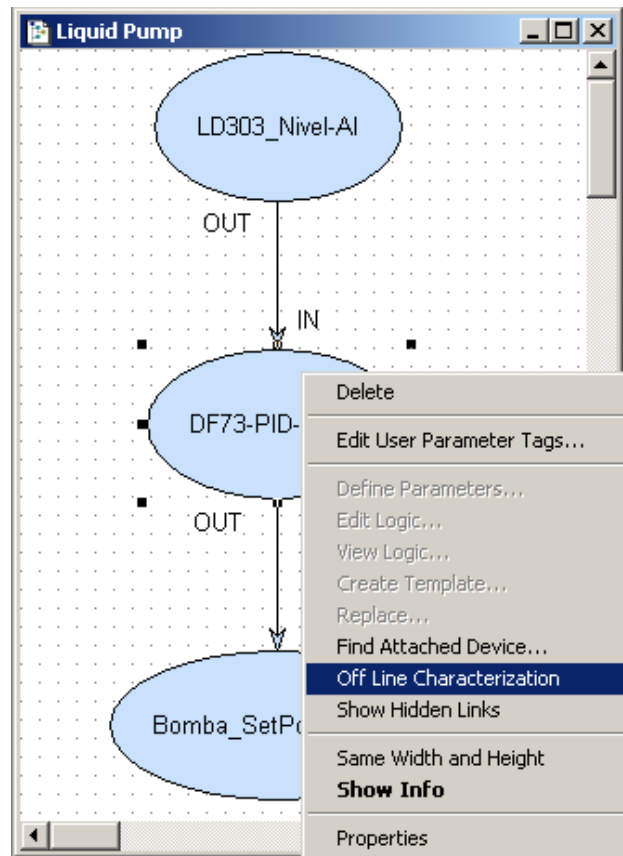


Figure 20. 60 - Offline characterization in the Strategy window

2. In the Profibus1 window

Another way to do the offline characterization is right-clicking the function block, and then selecting the **Off Line Characterization** option, as shown in the following figure.

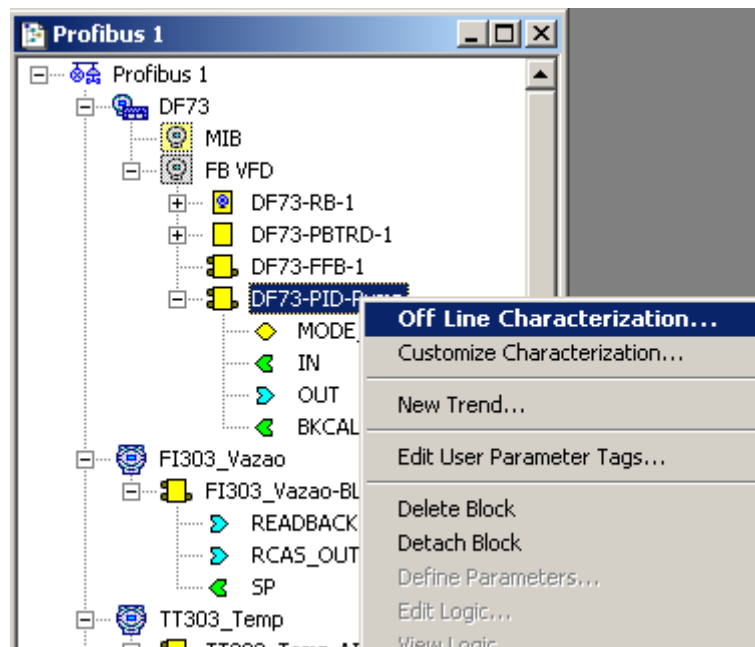


Figure 20.61 - Offline characterization in the Profibus1 window

For both situations, the **Block Characterization** dialog box will appear:

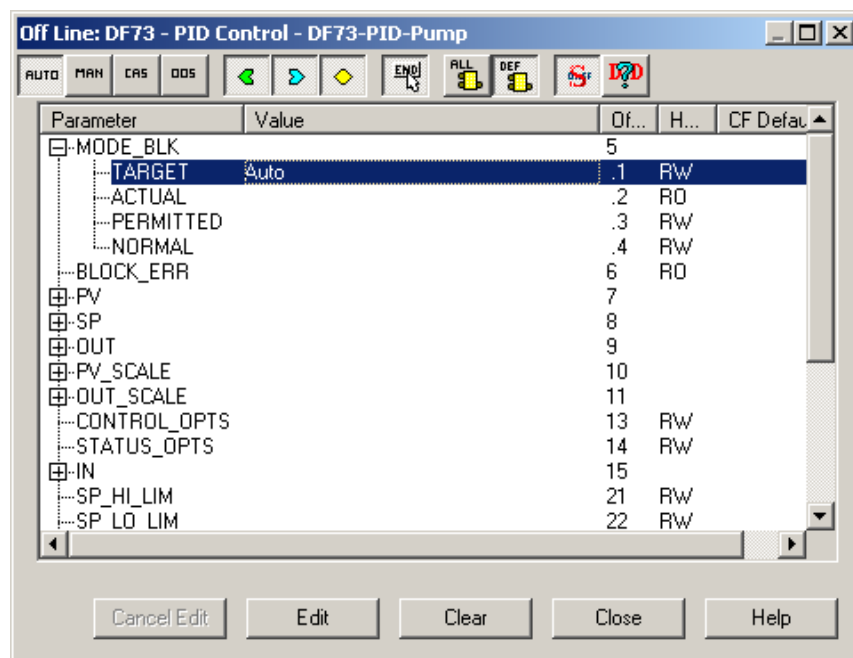


Figure 20.62 - Function block characterization dialog box

Double-click at the right side of the parameter to change it. Another option is click once and then click **Edit** to start editing the parameter value. At the ending, click **End Edit**.

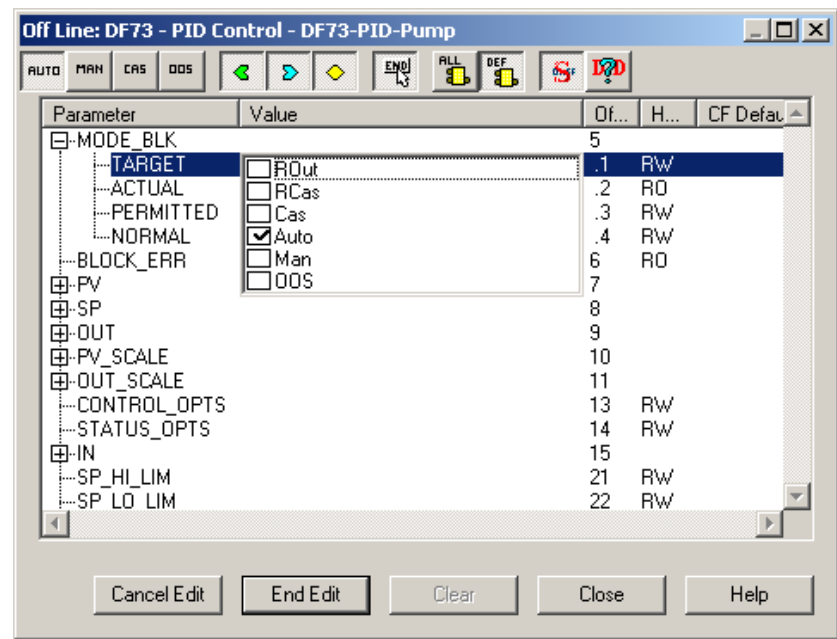


Figure 20. 63 - Editing the parameter in the Function Block Characterization box

An important parameter which has to be configured in all function blocks is **MODE_BLK**. It determines the block operation – Out of Service (OOS), Manual (MAN) or Automatic (AUTO). All blocks for the aimed control have to be with **MODE_BLK.TARGET** parameter in Auto.

Each block has a specific configuration parameter. For further information refer to Function Blocks manual.

NOTE

For the IO Profibus blocks is not necessary to configure the **CHANNEL** parameter. This parameter only will be used when IO blocks are used to map local IO points (IMB).

The flow control for heating the product can be done as follows:

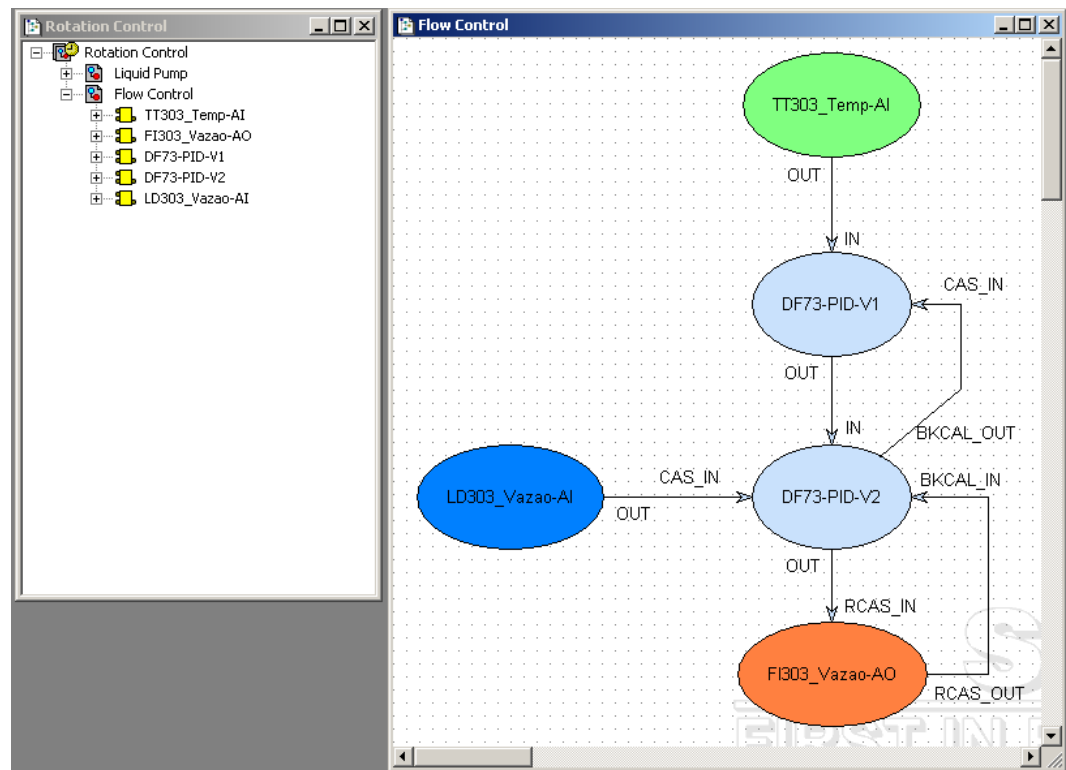


Figure 20.64 - Function blocks links

Commissioning and configuration download to the controller


After the parameters setting, the user can start the device communication. It is necessary commissioning the devices to attribute the tags, IDs and device addresses properly. If this procedure is not executed, the **Syscon** will detect the uncommissioned device and the download for this device will be canceled. Finishing the device commissioning, the download process can start.

The download process (**Syscon**) can be done in two main ways:

- Download of the plant – in this case will be downloaded all configurations of all controllers.
- Individual download in the controller – in this case will be downloaded all configurations of a single controller.

The controller configuration can be performed in several ways depending of the alteration. But the download performed by **Syscon** is the most recommended, because it controls the changes of all tools.

Syscon has two download types: total and incremental. The main difference is that the “total” cleans initially the configuration before downloading the new one while the “incremental” only downloads that has changed.

The download process can be executed, for example, returning to the **Proj_DF73** window, right-clicking the **Fieldbus Networks** icon, , and then selecting the **Download** option

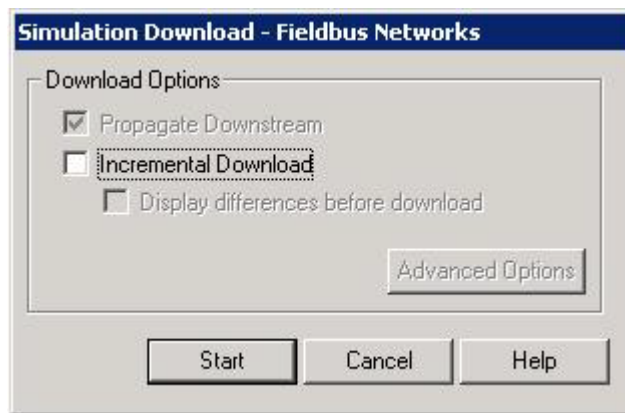


Figure 20.65 - Download options

Selecting the incremental download, and then **Advanced Options** the user can select several items to be downloaded. See the figure below.

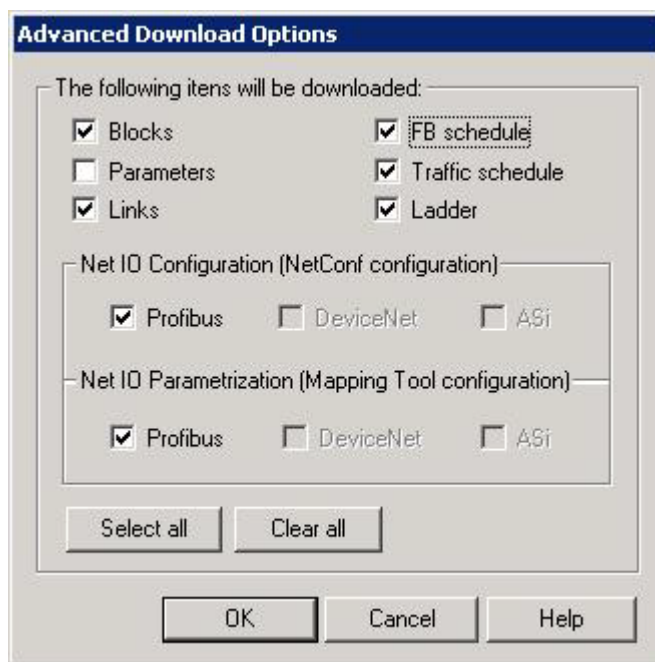


Figure 20.66 - Advanced download options

The **NetIO Configuration** option refers to the network topology configuration edited by the **Network Configurator**, and if it is selected the incremental download will be done. This operation must be done carefully because it stops the Profibus network.

The **NetIO Parameterization** option refers to the points parameterization configuration of network devices (points' tags, scales, data types) and when it is selected the incremental download will be done.

The next topic is maintenance guide which informs the recommended download type depending on the operation performed.

For further details about the available download types and commissioning, refer to the **Syscon** manual.

SYSTEM302 maintenance procedure

In this section is presented a summary of download procedures that are recommended in cases of configuration maintenance using **SYSTEM302**.

SYSTEM302 has several download types and here is recommended a general type to meet the majority of cases. However other ways can be used. For further information refer to the **Syscon** manual.

In the **Recommended Download** column is the type of chosen download. At **Syscon** it can be incremental on the plant or in the controller. Refer to the **Syscon** manual for further information.

MAINTENANCE EVENT	SOFTWARE	RECOMMENDED DOWNLOAD
Do a new SYSTEM302 configuration using Profibus controller and ladder logic.	Syscon, Network Configurator, Mapping Tool, LogicView for FFB	In Syscon do the plant incremental download*
Changing only the ladder logic.	LogicView for FFB	In LogicView for FFB do the download of the logic.
Add FFB parameters and make HSE links or only add parameters to the FFB block that already has HSE link.	Syscon	In Syscon do the plant incremental download*
Add or remove a FFB link to internal blocks within the same controller.	Syscon	In Syscon do the controller incremental download*
Add a new device in the network or change its configuration (address, cyclic mapping, tag)	Network Configurator	In Syscon do the controller incremental download*
Change device configuration (address, cyclic mapping, tag)	Network Configurator	In Syscon do the controller incremental download*
Change the interface parameters of the field device (scale, little/big endian, accessed bytes)	Mapping Tool	In Syscon do the controller incremental download*
Change media parameters (baud rate, communication configuration).	Network Configurator	In Network Configurator do the download of Profibus network*
Change configuration parameters of the field device hardware (acyclic configuration) or enable/disable the slave device.	Network Configurator	In Network Configurator do the download of Profibus network*

Table 01 – Recommended download procedures

Note: * These operations will stop the Profibus channel.

Profibus Network Time Parameters

The **Network Configurator** already has all time parameters of the Profibus network in TBit form, avoiding modifications when the baud rate value is changed. It is important to remember that the baud rate value is related to the slave which supports lower baud rate and the maximum distance of the Profibus DP network.

Bus Parameter

Baud rate: 500 kBits/s

Optimize: By User

Buttons: OK, Cancel, Edit...

Edit Bus Parameter

Baud rate: 500 kBits/s

Buttons: OK, Cancel

Slot Time	200	tBit	Target Rotation Time	3908	tBit
Min. Station Delay of Responders	11	tBit	Target Rotation Time	7.8160	ms
Max. Station Delay of Responders	100	tBit	GAP Actualization Factor	10	
Quiet Time	0	tBit	Max Retry Limit	1	
Setup Time	1	tBit	Highest Station Address	7	

Tid1	37	tBit	Poll Timeout	10	ms
Tid2	100	tBit	Data Control Time	1200	ms
Auto Clear			Min Slave Interval	25.000	ms
<input checked="" type="radio"/> Auto clear mode OFF <input type="radio"/> Auto clear mode ON			Watchdog control	200	ms

Figure 20. 67 - Editing the bus parameters

Each time parameter will be described separately:

Slot Time	300	tBit
Min. Station Delay of Responders	11	tBit
Max. Station Delay of Responders	150	tBit
Quiet Time	0	tBit
Setup Time	1	tBit

Figure 20. 68 - Time parameters

- **Slot Time (T_{SL})** – It is the maximum time that the master waits for a slave answer after it has done a request to the slave. If the master does not receive an answer after this time it will retry. This time is exactly the period between the last sent bit and the first bit received. It is also used in the token pass.
- **Min. Station Delay of Responders (\min_T_{SDR})** – It is the smaller time period between the last bit reception and the first bit sent of a station. There is a default period in the Profibus standard.
- **Max. Station Delay of Responders (\max_T_{SDR})** – It is the greater time period between the last bit reception and the first bit sent of a station. It is informed in the GSD file of each device.
- **Quiet Time (TQ)** – If a repeater is used or the media changes inducing a delay in the line, a time period is applied because the line has to be in silence. A consistent value needs to be **TQ < \min_T_{SDR}** .

- **Setup Time (TSET)** – It is the minimum reaction time period between the answer reception and the sending of new request by the master. It is informed in the GSD file of each device.

Target Rotation Time	4046	tBit
Target Rotation Time	2.6973	ms
GAP Actualization Factor	10	
Max Retry Limit	1	
Highest Station Address	25	

Figure 20. 69 - Time parameters

- **Target Rotation Time (T_{TR})** – It is the designed time for a token cycle (between 2 token receptions) including high and low priority operations (cyclic and acyclic), errors and GAPL maintenance. The details will be shown later.
- **GAP Actualization Factor (GAP)** – It is used to define the **TGUD** (GAP Update Time). **TGUD = G.TTR** , **0 < G < 100**. The GAP is the addresses list's name between a detected station and other. At each GAP Update Time, the master verifies the status of a device from the GAP List (**GAPL**). This procedure is part of the network maintenance.
- **Max. Retry Limit (Retry)** – It is the maximum number of retries which has to be performed for data reading from a station.
- **Highest Station Address (HSA)** – It is the highest station address which was configured. Address within the HSA and 127 are not verified and they are eliminated from the GAP List.

Poll Timeout	10	ms
Data Control Time	1200	ms
Min Slave Interval	0.600	ms
Watchdog control	200	ms

Figure 20. 70 - Time parameters

- **Poll Timeout** – It is important to master to master connection. This is the maximum time that a master has to answer a request from another master.
- **Data Control Time** – At each interval defined by Data Control Time, the master has to inform the slaves about its operation state. As a rule, the **Data Control Time** has to be greater or equals to **6.Tw**. The master can assume 4 states (OFFLINE, STOP, CLEAR, OPERATE).
- **Min Slave Interval** – The master has to assure that the time between two consecutive readings, in the same slave in two cycles, is not lesser than the slave can support. This parameter is specified in the slave's GSD file.
- **Watchdog control (T_{WD})** – In case of master failure, it is the time the slave waits before setting its outputs in **fail-safe**. As a rule **Twd > TTR**.

Tid1	37	tBit
Tid2	150	tBit
Auto Clear		
<input checked="" type="radio"/> Auto clear mode OFF <input type="radio"/> Auto clear mode ON		

Figure 20. 71 - Time parameters

Depend on the message type we have:

- **Idle Time (TID)** is formed by 2 elements TID1 and TID2.
- **TID1** – Maximum time calculated after the master/ slave has received a confirmation, answer or token (only master), and before sending the next request/pass token.
- As some frames do not have confirmation (SDN – Send Data with no Acknowledge) – a broadcast for example, TID2 is calculated in a different way.
- **TID2** – Maximum time calculated after the master/ slave has sent a message which does not have confirmation, and the next request/pass token.
- **TID1** = $\max \{ TQ + 2.TSET + 2.Tbit + TSYN, \min_TSDR \}$.
- **TID2** = $\max \{ TQ + 2.TSET + 2.Tbit + TSYN, \max_TSDR \}$.
- **Safety margin: TSM** = $2.Tbit + 2.TSET + TQ$
- **TSYN – Sync Time** – Minimum time interval that a station has to maintain in the idle state (idle = 1) in the media before starting a new request.
- **TSYN** is defined as 33 TBit. (**TSYN=33TBit**)
- **Auto Clear** – If a request is not answered by a device within the allowed retries number and the **Auto Clear** option is active, the master removes the device from the **data_exchange** condition and clears its outputs (scan memory).

Default Values of the Profibus Standard

Baud rate in kbit/s	187.5	500	1500	3000	6000	12000
Cable Distance (m)	1000	400	200	100	100	100
Default-DP-Master						
TSL (tBit)	100	200	300	1500	2000	2000
Min TSDR (tBit)	11	11	11	11	11	11
Max TSDR (tBit)	60	100	150	45	55	75
TSET (tBit)	1	1	1	4	8	16
TQUI (tBit)	0	0	0	3	6	9
G	10	10	10	10	10	10
HAS	126	126	126	126	126	126
Max_retry_limit	1	1	1	2	3	4

The strikethrough values, for example ~~2000~~, are recommendations from the manufacturers; they are not from the Profibus DP specification.

Recommended Values

The values below attend the conditions for network good working for a great quantity of configurations:

Baud rate in kbit/s	500
Cable Distance (m)	400
Default-DP-Master	
TSL (tBit)	200
Min TSDR (tBit)	11
Max TSDR (tBit)	100
TSET (tBit)	1
TQUI (tBit)	0
GAP	10
HAS (High Station Address)	126
Max_retry_limit	4
Twd (ms)	5000
TTR	50000

These values are default to the Profibus configurations in the **SYSTEM302**.

The *Watchdog Time* (**Twd**) can be decreased, in case only Profibus DP devices are used, it is recommended $\text{Min}(2,5 \times \text{TTR}, 1000\text{ms})$.

If the configuration is being performed for the DF95 and DF97 to 12Mbps using the value of the **Slot Time** (**TSL**) should not be less than 6000 tbit.

Required values when using third party devices

When there is a DP/PA coupler in the Profibus network, many times, the network parameters are conditioned to values specified on their manuals. In the next table are the more common devices which require special configurations.

Parameter	Reference from Manual or Standard	Siemens SIMATIC DP/PA Coupler	DP/PA Coupler Pepper+Fuchs Model: KFD2-BR-1.PA.93
Baud Rate	BaudRate	45,45Kbps	93,75Kbps
Slot Time	TSL	640	4095
Min TSDR	MinTSDR	12	22
Max TSDR	MaxTSDR	400	1000
Setup Time	TSET	95	150
Quiet Time	TQUI	0	0
GAP Factor	GAP	10	10
High Station Address	HSA	126	126
Max retry Limit	Max Retry	3	3
Watchdog Time	TWD	5000	5000
Target Rotation Time	TTR	50000	50000

Network Diagnostics

There are many ways to identify communication failures in the Profibus network. Some options are: **Network Configurator**, Transducer block, **Syscon's** Live List or module's LEDs. Some ways will be explained on this topic.

Network diagnostic using the Network Configurator

To do a network diagnostic through **Network Configurator** first is necessary to connect the **Network Configurator** to the controller. This can be done through serial communication (linking a serial cable of the computer to the controller RS-232 port) or through the Ethernet network as shown in the following figure.

First inform the master which will be used. Select the master device of the configuration, and then in the toolbar select **Settings→Device Assignment**. On the popup choose **CIF TCP/IP Driver** option, and click **OK**. A window will open to insert the master IP address.

Device Assignment ODM TCP/IP Driver

Driver Description
Driver: ODMTcplp V2.032

Add IP Address
IP Address: 10 . 0 . 0 . 14 Add

Board Selection

	IP Address	Type	Serial Number	MAC Address	Address Switch
<input checked="" type="checkbox"/>	10.0.0.14	SMAR	0	00-30-5C-08-00-0E	0
<input type="checkbox"/>					
<input type="checkbox"/>					
<input type="checkbox"/>					
<input type="checkbox"/>					
<input type="checkbox"/>					
<input type="checkbox"/>					
<input type="checkbox"/>					

Filtered Device(s)

	IP Address	Type	Serial Number	MAC Address	Address Switch

Select device

OK Cancel NetIdent Rescan Set IP Address

Figure 20. 72 – Doing the master Device Assignment

To do a network diagnostic just click the master device, and then **Online→Debug Mode**.

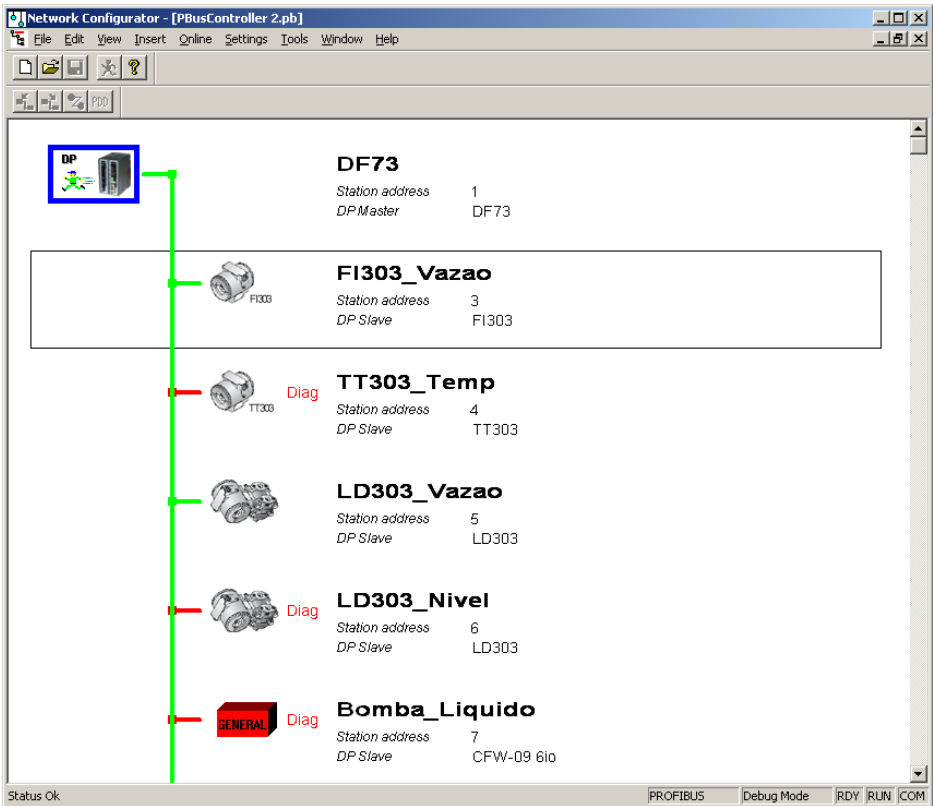


Figure 20. 73 – Doing the network diagnostic

The red spur indicates a device with some configuration failure. By clicking the device a description of the failure appear.

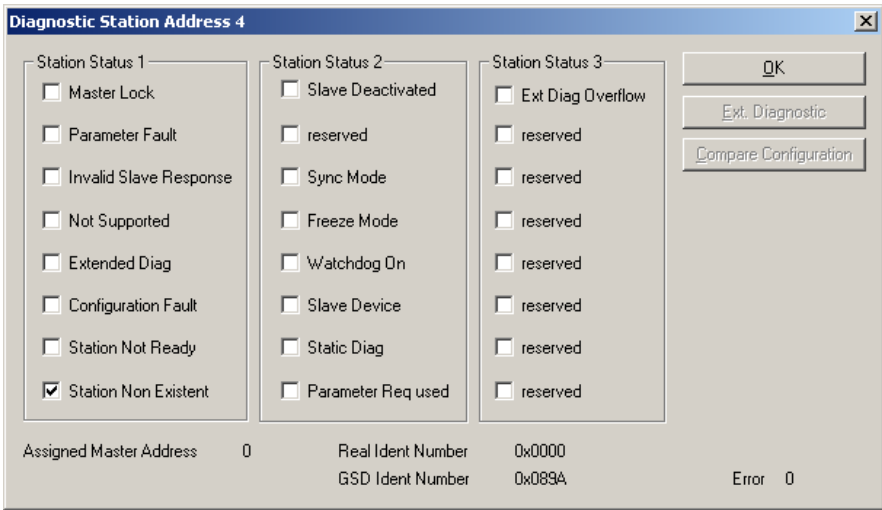


Figure 20. 74 – Device failure description

The example above shows an error due to some physical or address problem the master does not receive answers from the configured node.

IMPORTANT

The Ident Number is the device's model and it is used as a key allowing the configuration. If the device's Ident Number (physical) does not match with that is in the GSD file (configuration), the parameterization will not be concluded and the configuration will be canceled.

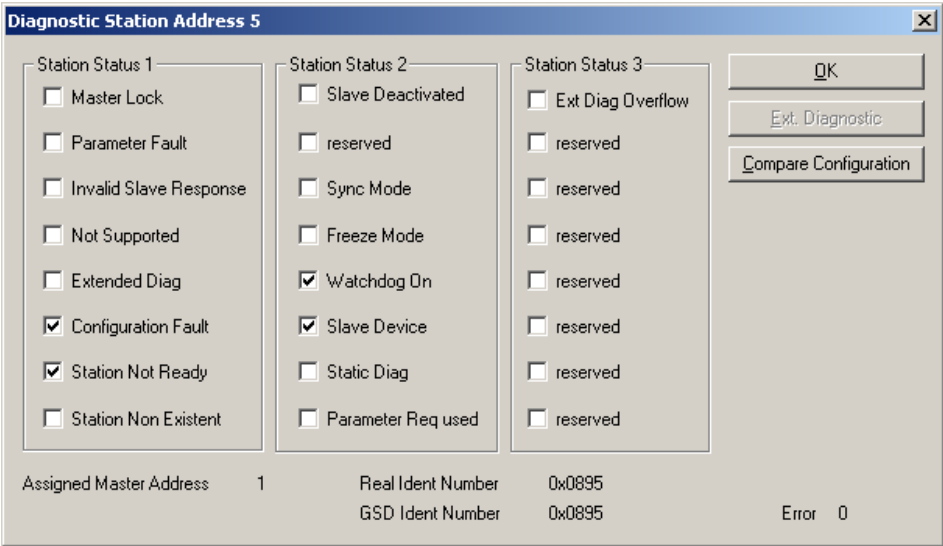


Figure 20. 75 – Device failure description

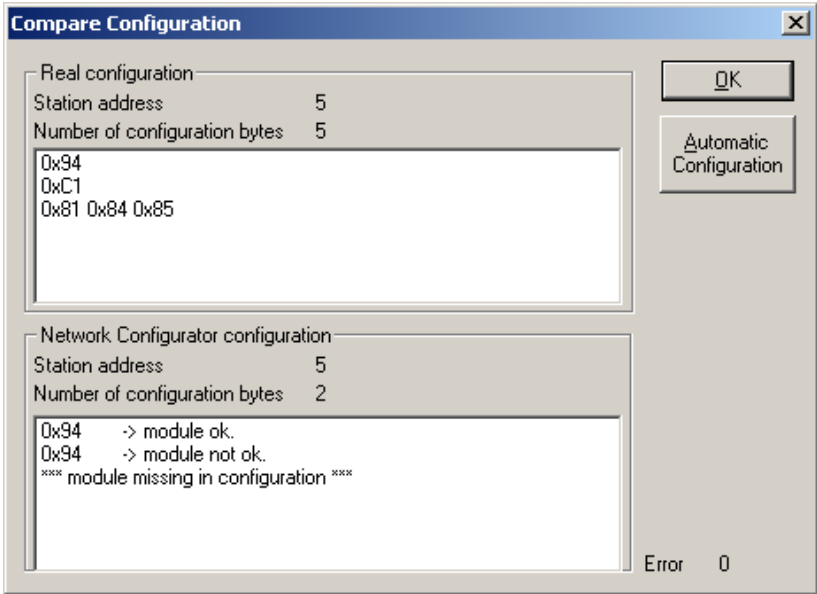


Figure 20. 76 – Device failure description

The example above shows a configuration error of Profibus device, because the LD303 was configured with two AI modules, and it would have to be configured with AI and Totalizer or AI and EMPTY_MODULE, if the flow rate is not totalized.

IMPORTANT

If an address is repeated in the network, i.e., at least two nodes were configured with the same address, a conflict will happen and the master will consider the first answer as valid at each request. The work will be unstable for both addresses. In this case the master will cancel continuously the communication with both devices and it will notify with an error diagnostic.

Extended Diagnostic in the Network Configurator

The DF73 as Profibus DP master has resources to show the communication status, disconnections status, communication errors and number of retries.

The access to the extended diagnostics is done by clicking the master and accessing the **Online** menu.

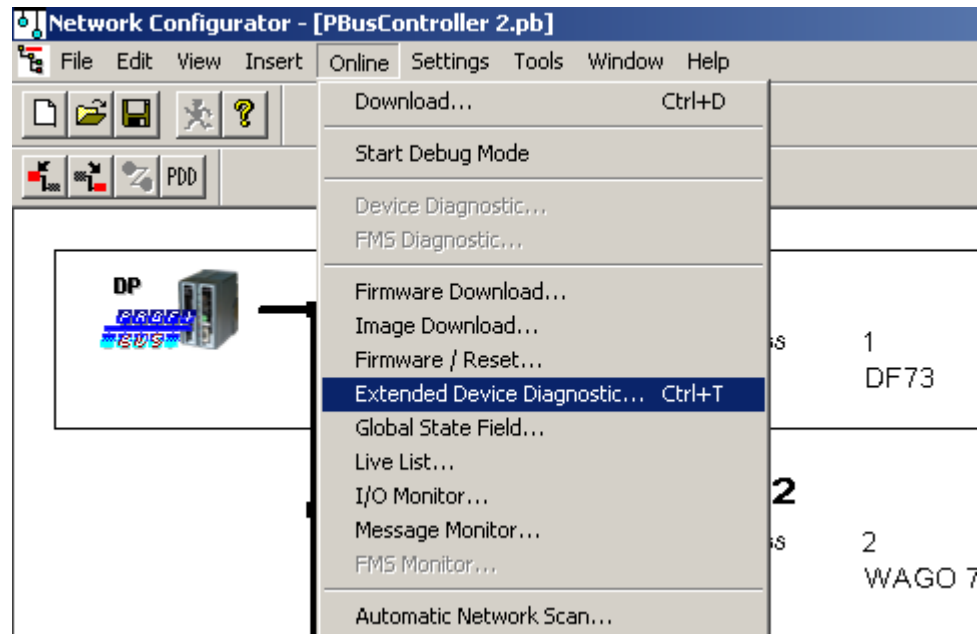


Figure 20. 77 – Extended diagnostic

The next window will appear:

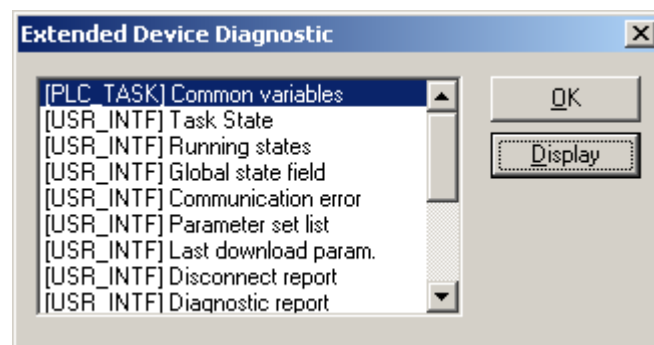


Figure 20. 78 – Extended diagnostic options

Communication error

Through the **Communication error** item is possible to verify the error status of each slave. This error status is not increased (it does not record the errors); it only shows the error status at this moment.

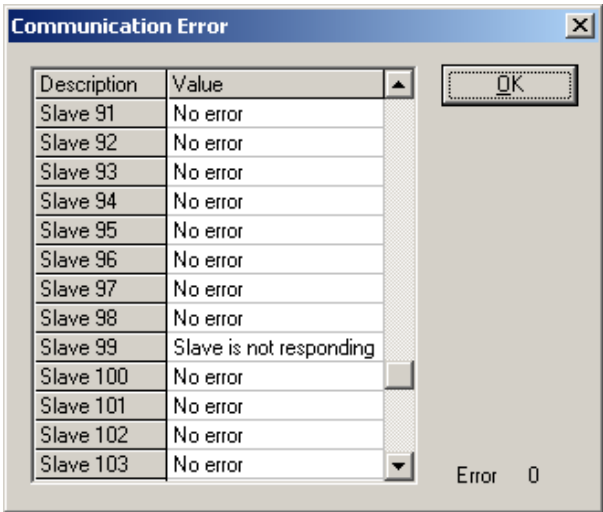


Figure 20. 79 – Status of slaves’ communication errors

Disconnect Report

It shows the number of communication disconnections due to communication errors. The counter increases only when the master is exchanging data with the slave and after finishing all retries (see **Max Retry Limit**).

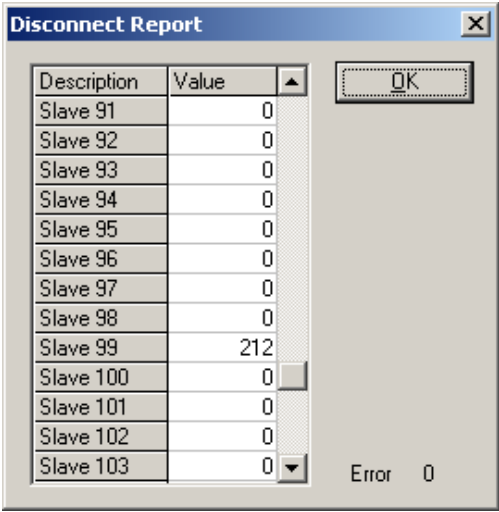


Figure 20. 80 – Disconnect report

Diagnostic Report

It shows the number of new diagnostics which the slaves make available. At each data exchange the bits, if they are configured by the slaves, are verified when inform that there is available a new diagnostic information, and the master has to do a new diagnostic request.

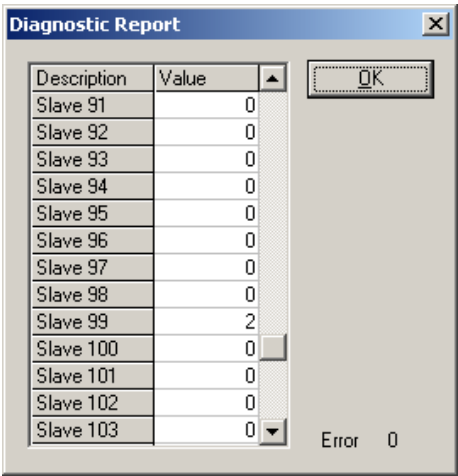


Figure 20. 81 – Diagnostic report

Retry for Slave

It shows the number of communication tries which were done by specific slave (see **Max Retry Limit**) when it does not answered.

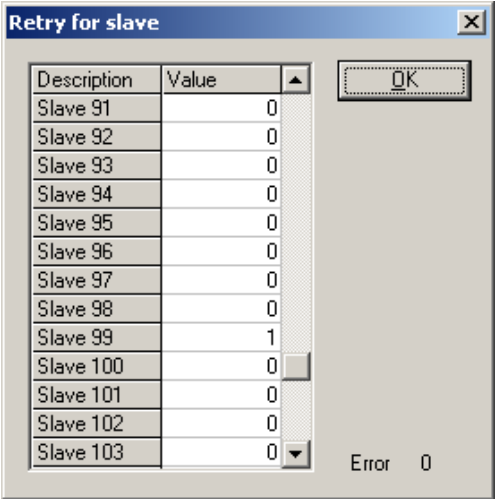


Figure 20. 82 – Retry for slave

CREATING A DEVICENET CONFIGURATION BY USING DF79

Introduction

This section will show a strategy configuration using the DF79 controller. The next figures show an example of process used to control the conveyor movement for filling up bottles with some fluid. The conveyor moves until the sensor detects the bottle. At this moment the valve (ON/OFF) opens and fills up the bottle. The architecture of the DeviceNet control loop can be seen in the following figure.

PROJ_DF79

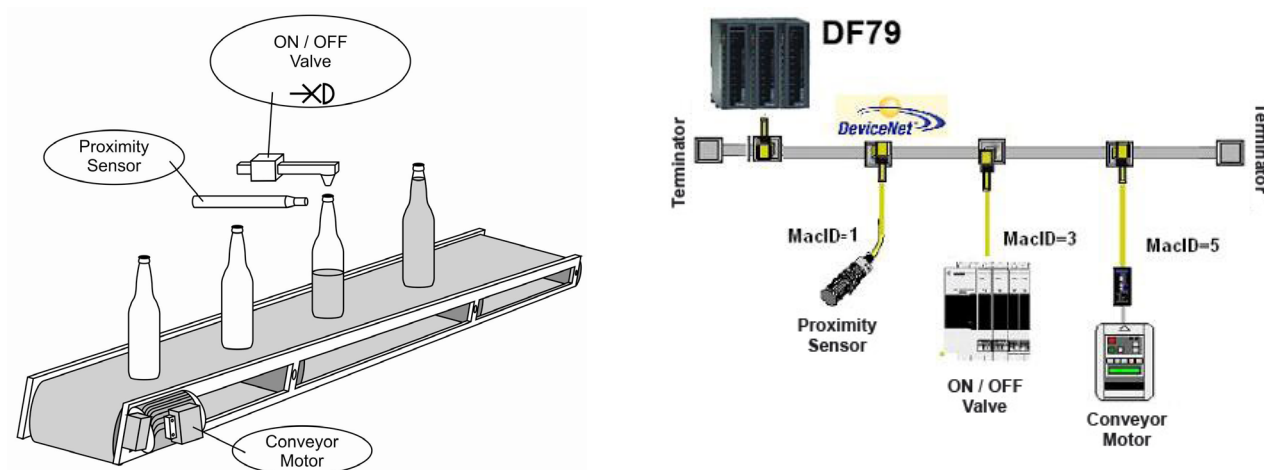


Figure 21. 1 – Example of Process

For the example above the following DeviceNet devices were used: an AC Drive to control the conveyor motor, a sensor to detect the bottle and a discrete I/O card to turn on and turn off the valve.

In the next topics will show, step by step, how to configure the DF79 controller for the aimed example. At the **SYSTEM302** is possible to do control logic of two ways: ladder or function blocks. The steps 1 to 7 are necessary to configure the DeviceNet network. The step 8 shows how to do the configuration using ladder logic. From step 8' hereafter is shown how to do the logic using function blocks. Also is possible to mix the two configurations, i.e., part in ladder and part in function blocks.

IMPORTANT

Before starting the DeviceNet configuration in the **SYSTEM302** is necessary to obtain information about each slave device configuration (module address, baud rate, supported types of cyclic communication and mapping structure). Also is necessary the electronic identification file of the device EDS. This information is obtained with the respective DeviceNet device manufacturer.

Starting the Area

Step 1

It is possible to create, or edit, an area from the **Studio302**. In the **Studio302** interface select **Areas**. A window will appear listing all the areas of database.

To create a new area from the **Studio302**, left-click inside the **Areas** window, then choose **New Area**.

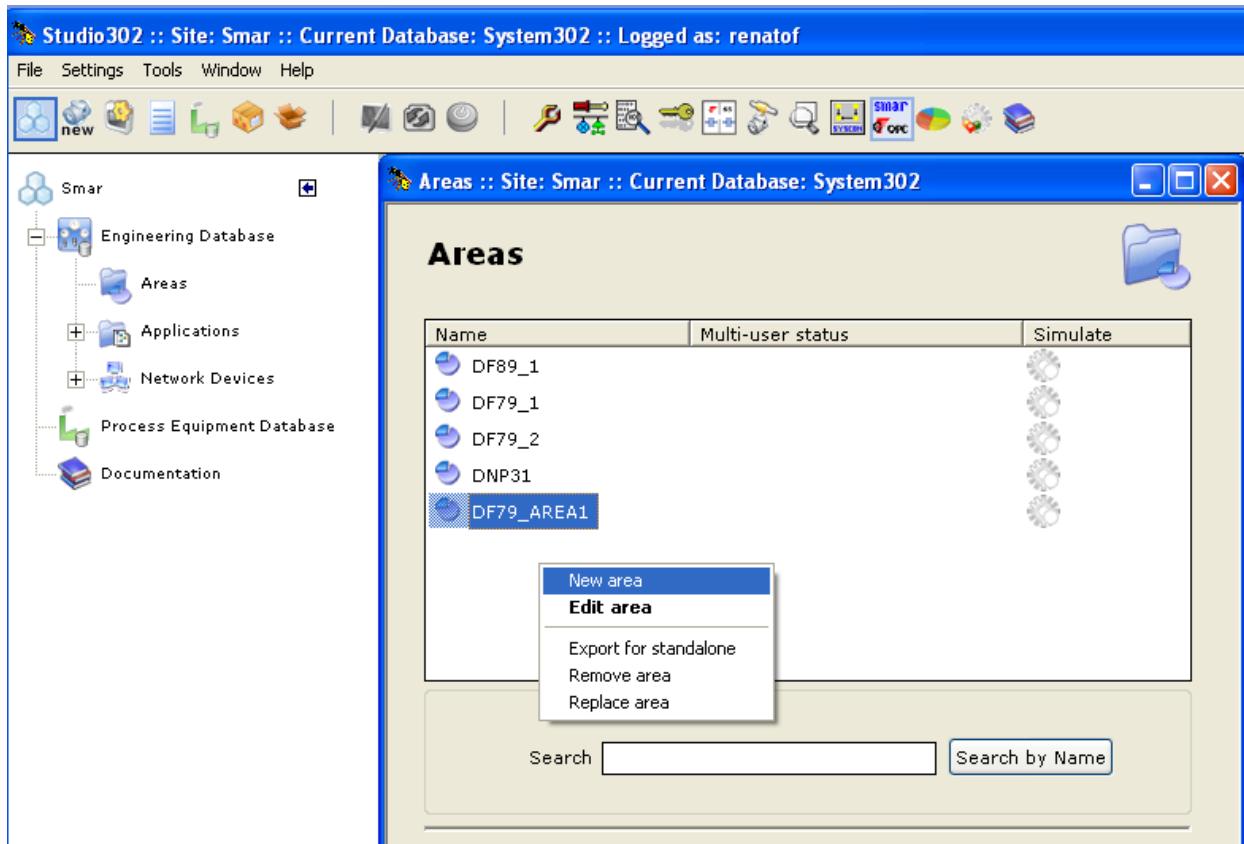


Figure 21. 2 – Creating a new area

Another way to create a new area is from **Syscon**. Click the icon  in the **Studio302** toolbar.

To create a new area on **Syscon**, choose **File → New**, or through the toolbar, choose **New** button .

The dialog box shows the areas options. Select **HSE Area** as shown in next figure:



Figure 21. 3 - Options to create Syscon areas

After choosing the area type, it opens a window to the user give a name to the new area.

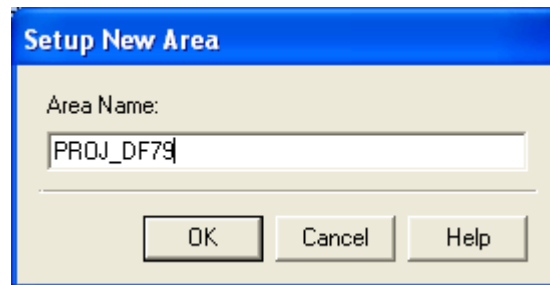


Figure 21. 4 – New area name

Type the name for the area in the **Area Name** box, and click **Ok**. For this example, it chooses **PROJ_DF79** name.

A new window will appear. It shows:


- **Application** – Logical Plant. To insert the control strategies into this part.
- **Fieldbus Networks** – Physical Plant. To insert devices and function blocks used in the area.



Figure 21. 5 - Area divisions

Physical Plant Project

Step 2

In the main window, **PROJ_DF79**, right-click the **Fieldbus Networks** icon, , to select the Server. Choose **Communication Settings** option, or through the toolbar, choose **Communication→Settings**. The communication settings dialog box will open.

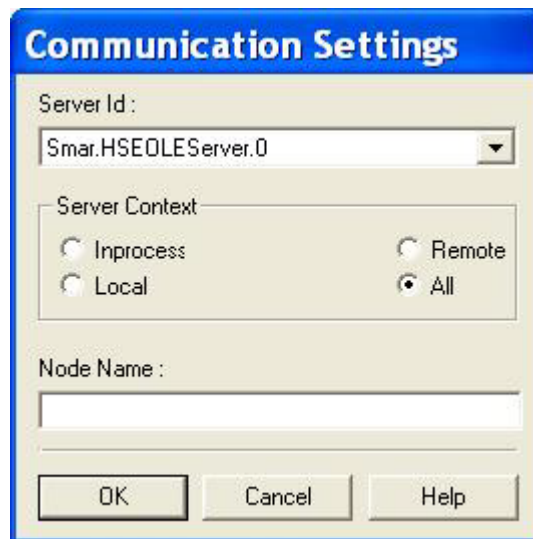


Figure 21.6 – Choosing the Server

Confirm if the **Smar.HSEOLEServer.0** option has already been selected. Otherwise, the user must select it, and then click **OK**.

Arranging the Fieldbus windows

Step 3


After selecting the Server for the area, click the  sign placed at left of the **Fieldbus Networks**. The HSE network will appear with a tag, for example, *HSE Network 1**. Right-click this item and choose the **Expand** option. The following figure shows the HSE network:



Figure 21.7 – Creating the HSE network

To arrange the screen, click the area window. So, choose **Window** menu on the **Syscon** toolbar, and then choose **Tile** option.

* This number changes if another area was created before. When a new HSE area is created, this number increases.

Inserting the Controller

Step 4

Right-clicking the **HSE Network 1** icon, a dialog box to add new devices will appear. By clicking **New**, it is possible to select devices such as Bridges, Controllers and Devices for the area. For the aimed control, it selects the **Controller** option. Confirm this choice as shown in next figure.

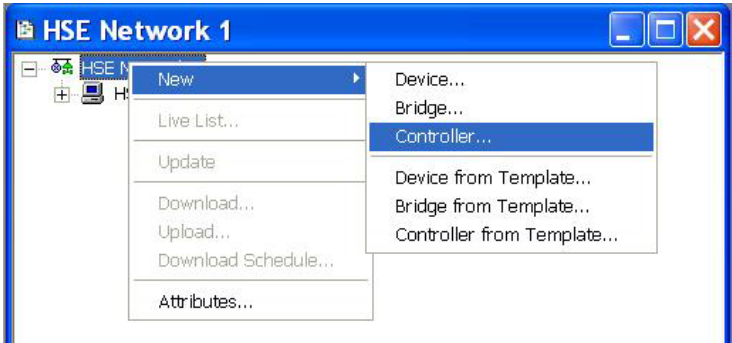


Figure 21. 8 – Choosing the controller

After adding the new controller, it opens a window as shown in following figure.

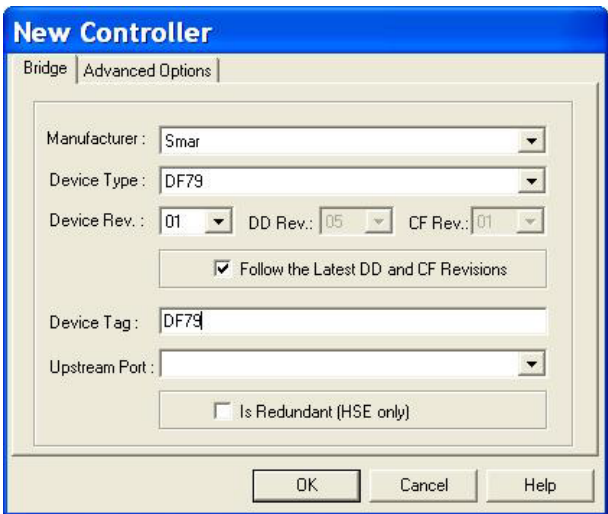


Figure 21. 9 – Setting the controller

Select **DF79** in the **Device Type** box. In the **Device Tag** box, type **DF79** or another tag, and click **OK** to conclude this task.

IMPORTANT

Not all characters are valid when naming the elements.

The valid characters are:

A-Z a-z 0-9 # { } [] () + -

The invalid characters are:

~ ` ! @ # \$ % ^ & * = | : ; , . < > ? / ' " \

TIP

Is possible to create an initial HSE configuration in an easier way using templates. In this case a configuration exists with some common steps created previously. For example, the steps 1 to 4 can be replaced by a creation of template through main menu **File → New → Predefined Area** choosing **DF79 DeviceNet HSE** or **DF79 DeviceNet HSE with FFB-1131**.

Adding DeviceNet Devices

IMPORTANT

Before launch the **Network Configurator** the user must save the configuration on **Syscon**, and then go to the next step.

Step 5

After inserting the controller, it is possible to insert DeviceNet devices. Before, return to the **PROJ_DF79** window and right-click the controller, **DF79**. Then choose **New Network** option to configure the DF79 channels. See the DeviceNet bus creation in following figure.

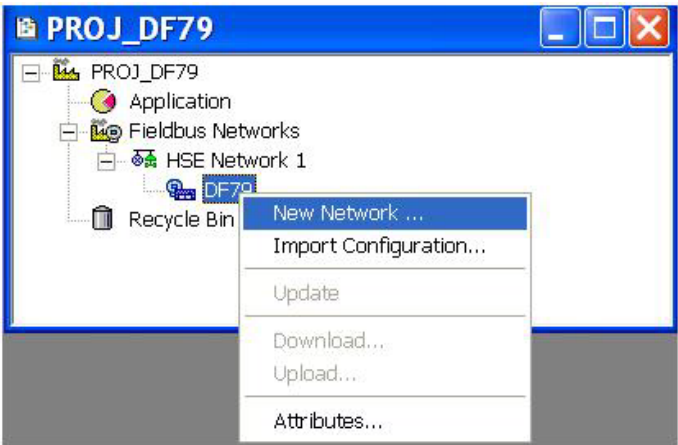


Figure 21. 10 – Creating the DeviceNet bus

When selecting **New Network**, the **Network Configurator** tool opens. The **Network Configurator** is the configuration tool for the DeviceNet network. Through it, the user has the structure of the DeviceNet network (devices, baud rate, and connection type). For further information about this tool, refer to its Help.

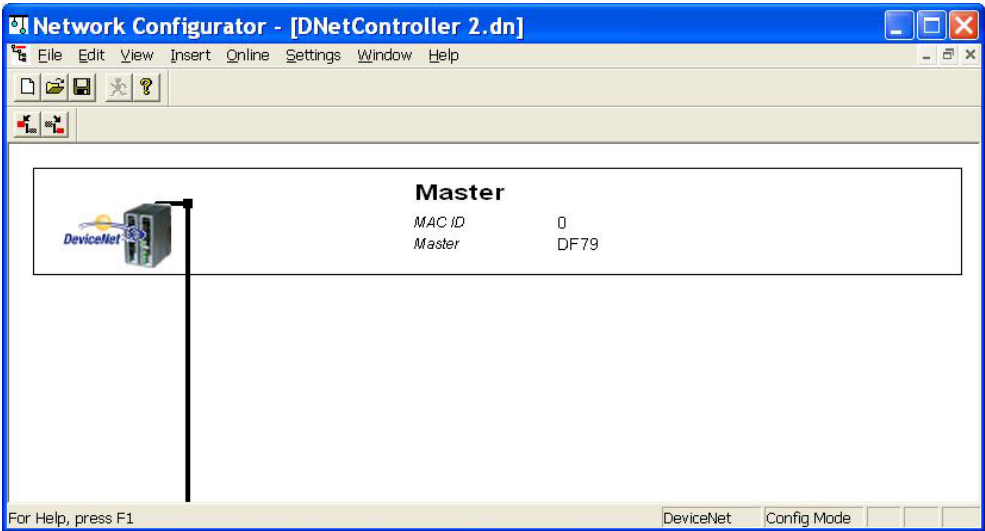



Figure 21. 11 – Network Configurator window

The controller DF79 has already been inserted to the configuration previously, so it is the Master device in the DeviceNet network. The DeviceNet devices that will be inserted to the network are named slaves.

IMPORTANT

The address 0 in the DeviceNet network is the default address of the DF79 (MAC ID). It is recommended does not use the address 63 for moving the devices in the DeviceNet network.

To insert slave devices, click the button  (**Insert Device**) placed on the toolbar, or through the menu **Insert→Device....** Place the mouse cursor just below where the Master device is placed and left-click. See the next figure.

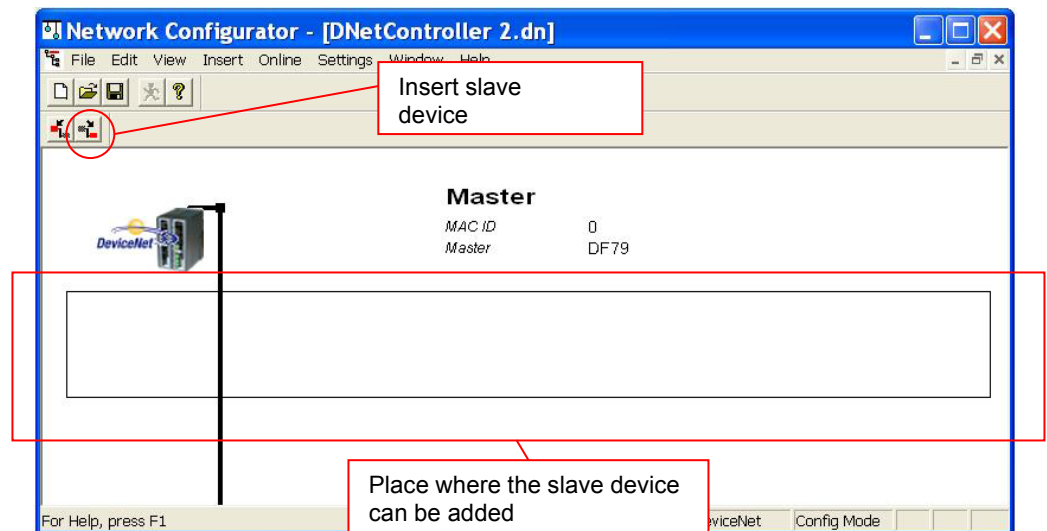


Figure 21. 12 – Placing the slave device in the DeviceNet network

The window for inserting slave devices will open, as shown below:

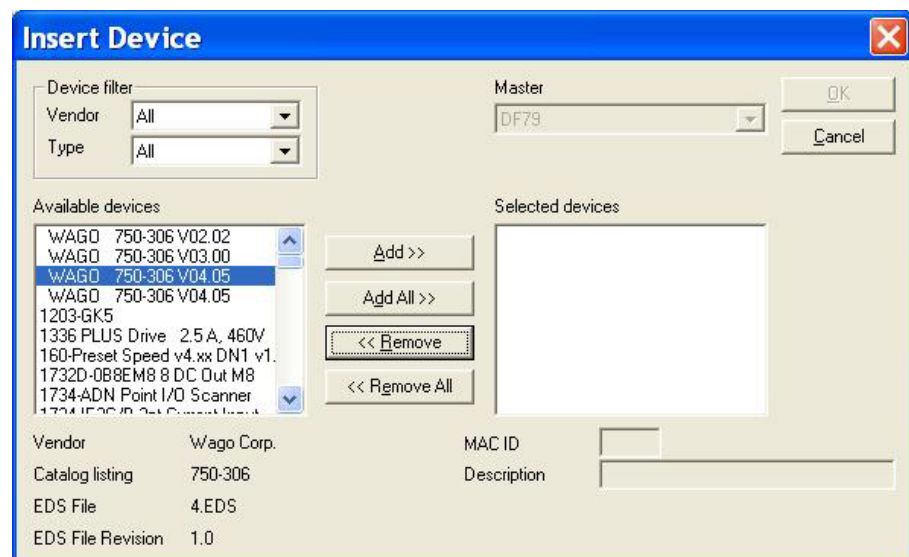


Figure 21. 13 – Inserting slave device

The user can select the slave device in this window. For the aimed control, the first device is the proximity sensor. First the user has to obtain the configuration information of the DeviceNet slave device (refer to device's manual). Follow these steps to insert Slave devices:

- A. Select the device in the **Available slaves** box;
- B. Click **Add** button;
- C. Type the address for the device in the **MAC ID** field (this address has to correspond to the physical configuration of the device in the DeviceNet network. Further details about how to configure the device's address refer to device's manual);
- D. Type the tag for the device in the **Description** field. If the user does not write any tag for the device, the default tag will be **Device1**.
- E. Click **OK**.

NOTE

The address as well as the tags of the devices must be unique in the DeviceNet network and also in the **Studio302** Database (Workspace). For further information refer to **Studio302** manual.

The user also has to insert the devices corresponding to the ON/OFF valve and the conveyor motor. When these operations finish the DeviceNet network will be as the following figure.

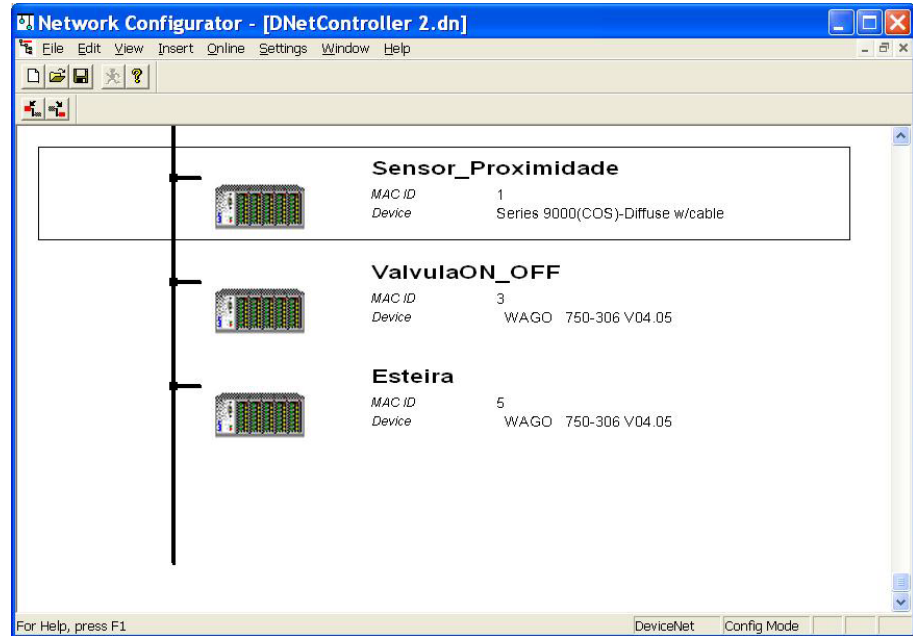


Figure 21. 14 – DeviceNet network created

Inserting Slave Devices that are no present in the “Available Slaves” list

The EDS files have the description of each DeviceNet device. They are provided by devices' manufacturers. These files define the devices' specific features in the DeviceNet network and their objects. The set of these files forms the devices' database. The **SYSTEM302** already has an EDS database of the more common manufacturers, but is possible to add new devices in the **SYSTEM302** database.

If there is a device which will be in the DeviceNet network, but is not available in the devices' list, the user has to contact the device's manufacturer to take the EDS file, and also to take the *.ico file (with the device icon) if it exist. The new files must be placed in the following paths:

- EDS files: ...\\Smar\\Network Configurator\\Fieldbus\\DevNet\\EDS
- Drawings or bitmaps (*.ICO): ...\\Smar\\Network Configurator\\Fieldbus\\DevNet\\BMP

NOTE

The module drawing (icon) only will be associated to the EDS file if it has the following line (that some manufactures does not include) inside the [Device] section:
Icon = "filename.ico". For example:

```
[Device]
:
    Icon = "thumbnail4.ico";
```

To check if the drawing is associated to the EDS, create in the **Network Configurator** an EDS of the product and check if a correct icon is associated. Otherwise a **Network Configurator** default drawing will appear.

Setting DeviceNet Devices

Step 6

After creating the DeviceNet network, the devices have to be configured. For example, right-click the device corresponding to the ValvulaON_OFF. Choose the **Device Configuration** option to configure the slave device.

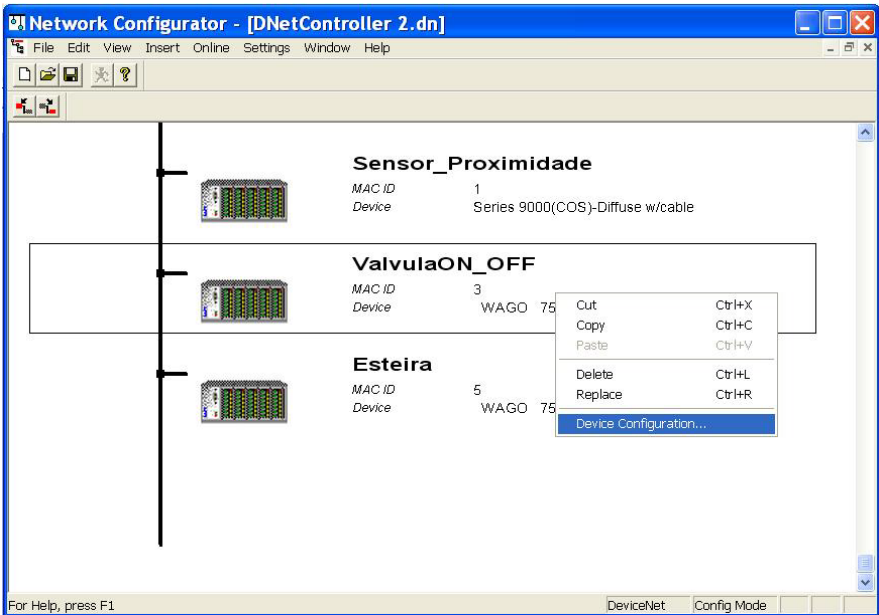


Figure 21. 15 – Setting the slave device

The configuration window also will open double-clicking the device.

The window for device configuration will open. See the figure below.

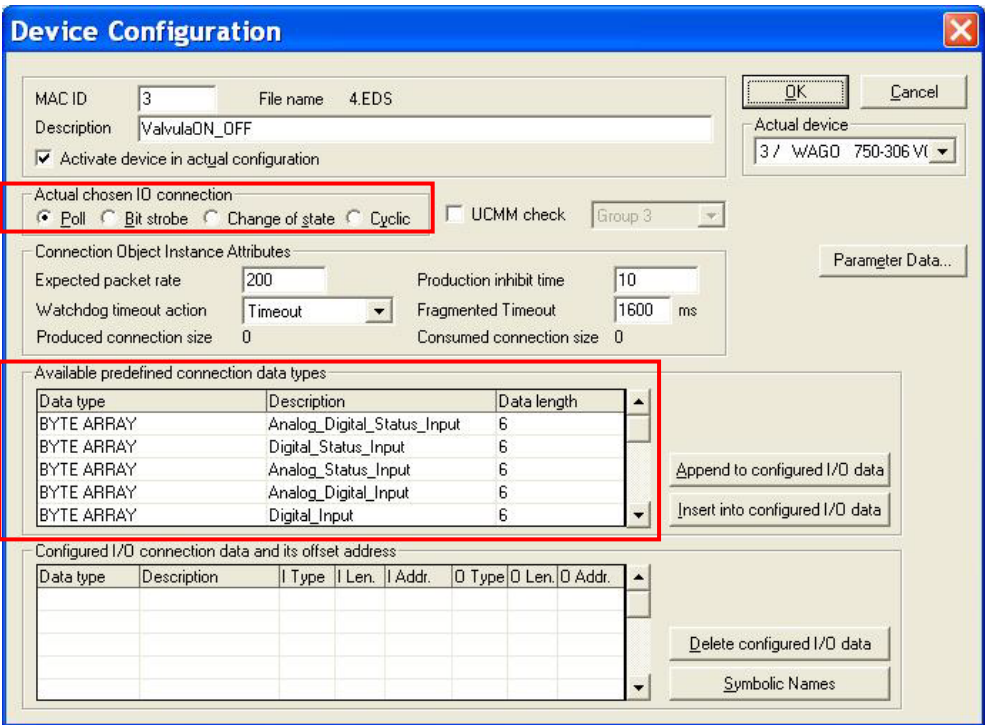


Figure 21. 16 – Setting the slave device

Are necessary to configure the IO connection type (IO Connection), and also the data order

(depending of slaves' cards configuration). The DeviceNet supports 4 types of IO connections: Poll, Bit Strobe, Change of State and Cyclic. The devices can support different configurations. In the **DeviceNet Advanced Topics** section each one of the items above will be detailed. Refer to the DeviceNet slave devices' manuals for further information.

The control parameters configuration (I/O message data exchanging) is done by choosing the data types and the size in the **Available predefined connection data types** box. This configuration can be fixed (if the device has a fixed number of IO points), or dependent of cards' arrangement (if the slave has a variable number of IO points).

The configurations for **PROJ_DF79** will be as follows:

The **Sensor_Proximidade** device supports IO communication of the types Polling, Bit Strobe and Change of State. This last will be chosen for the aimed example. Usually the manufacturer provides in the device's EDS only the supported types, in this case can be any of the available communications' types. The device's communication data is 1 bit of data, because it has only one digital input (**Sensor1** option was selected in the **Configured I/O Connection data and offset address** box). See the next figure.

Device Configuration

MAC ID: 1 File name: 00010006001E0100.EDS
 Description: Sensor_Proximidade
☒ Activate device in actual configuration

Actual device: 1 / Series 9000(COS)-Dif

Actual chosen IO connection:
☐ Poll ☐ Bit strobe ☒ Change of state ☐ Cyclic ☐ UCMM check Group 3

Connection Object Instance Attributes:
 Expected packet rate: 200 Production inhibit time: 10
 Watchdog timeout action: Timeout Fragmented Timeout: 1600 ms
 Produced connection size: 1 Consumed connection size: 0

Available predefined connection data types:

Data type	Description	Data length
BYTE ARRAY	Sensor	1

Append to configured I/O data
 Insert into configured I/O data

Configured I/O connection data and its offset address:

Data type	Description	I Type	I Len.	I Addr.	O Type	O Len.	O Addr.
BYTE ARRAY	Sensor1	IB	1	0			

Delete configured I/O data
 Symbolic Names

Figure 21. 17 – Setting the sensor

For the **ValvulaON_OFF** was chosen a device with a variable number of IO, and the following cards were added: 4 discrete inputs and 4 discrete outputs. In this case the slave device supports the four types of IO communication. For the aimed example was chosen **Polling**. The IO cards configuration will be: Digital_Input_11 with 2 bytes, because the device has 4 digital inputs (1 byte) in addition to their status (1 byte), and Digital_Output_11 with 1 byte, because the device has 4 discrete outputs (1 byte). See the following figure.

Device Configuration

MAC ID: 3 File name: 4.EDS

Description: ValvulaON_OFF

☒ Activate device in actual configuration

Actual device: 3 / WAGO 750-306 VI

Actual chosen IO connection: ☒ Poll ☐ Bit strobe ☐ Change of state ☐ Cyclic ☐ UCMM check Group 3

Connection Object Instance Attributes

Expected packet rate: 200 Production inhibit time: 10

Watchdog timeout action: Timeout Fragmented Timeout: 1600 ms

Produced connection size: 2 Consumed connection size: 1

Available predefined connection data types

Data type	Description	Data length
BYTE ARRAY	Analog_Digital_Status_Input	6
BYTE ARRAY	Digital_Status_Input	6
BYTE ARRAY	Analog_Status_Input	6
BYTE ARRAY	Analog_Digital_Input	6
BYTE ARRAY	Digital_Input	6

Append to configured I/O data

Insert into configured I/O data

Configured I/O connection data and its offset address

Data type	Description	I Type	I Len	I Addr	O Type	O Len	O Addr
BYTE ARRAY	Digital_Input_11	IB	2	1			
BYTE ARRAY	Digital_Output_11				QB	1	0

Delete configured I/O data

Symbolic Names

Figure 21. 18 – Setting the valve

For the conveyor (**Esteira**) was chosen a device with a variable number of IO points, and the following cards were added: 4 discrete inputs, 1 analog input and 4 discrete outputs. In this case the slave device supports the four types of IO communication. For the aimed example was chosen **Polling**. The IO cards configuration will be: Digital_Input_11 with 2 bytes, because the device has 4 digital inputs (1 byte) in addition to their status (1 byte), Analog_Input_11 with 2 bytes, because the analog variable has 12 bit-resolution accepting integer values from 0 to 4095, i.e., 2 bytes, and Digital_Output_11 with 1 byte, because the device has 4 discrete outputs (1 byte). See the following figure.

Device Configuration

MAC ID: 5 File name: 4.EDS

Description: Esteira

☒ Activate device in actual configuration

Actual device: 5 / WAGO 750-306 VI

Actual chosen IO connection: ☒ Poll ☐ Bit strobe ☐ Change of state ☐ Cyclic ☐ UCMM check Group 3

Connection Object Instance Attributes

Expected packet rate: 200 Production inhibit time: 10

Watchdog timeout action: Timeout Fragmented Timeout: 1600 ms

Produced connection size: 4 Consumed connection size: 1

Available predefined connection data types

Data type	Description	Data length
BYTE ARRAY	Analog_Digital_Status_Input	6
BYTE ARRAY	Digital_Status_Input	6
BYTE ARRAY	Analog_Status_Input	6
BYTE ARRAY	Analog_Digital_Input	6
BYTE ARRAY	Digital_Input	6

Append to configured I/O data

Insert into configured I/O data

Configured I/O connection data and its offset address

Data type	Description	I Type	I Len	I Addr	O Type	O Len	O Addr
BYTE ARRAY	Digital_Input_11	IB	2	3			
BYTE ARRAY	Analog_Input_11	IB	2	5			
BYTE ARRAY	Digital_Output_11				QB	1	1

Delete configured I/O data

Symbolic Names

Figure 21. 19 – Setting the conveyor

When inserting devices finishes, the mapping must be done. It consists of mapping the values, bytes and data to be reading/written from the devices to the master memory. The procedure will be described in the following topic.

After selecting the inputs and outputs for the DeviceNet devices, set the baud rate for the DeviceNet network. Click the Master device and choose **Settings→Bus Parameter** on the toolbar. The dialog box will open:



Figure 21. 20 – Selecting the baud rate for the DeviceNet network

Select the baud rate in the **Baud Rate** list and click **OK**.

NOTE

The baud rate of the DeviceNet network has to be configured according to the DeviceNet device which has the lowest baud rate, and all DeviceNet devices must be configured with the same baud rate.



Save the configuration before exit from **Network Configurator**

Step 7

When the previous steps were finish, the DeviceNet network configuration can be saved, and then, exit from **Network Configurator**. The Mapping Tool launches automatically. With this tool is possible to map the points of DeviceNet network in available points which will be used in ladder logic or in function block logic. This tool configures all characteristics of DeviceNet points (user tags, data types, scales, etc). See the following figure.

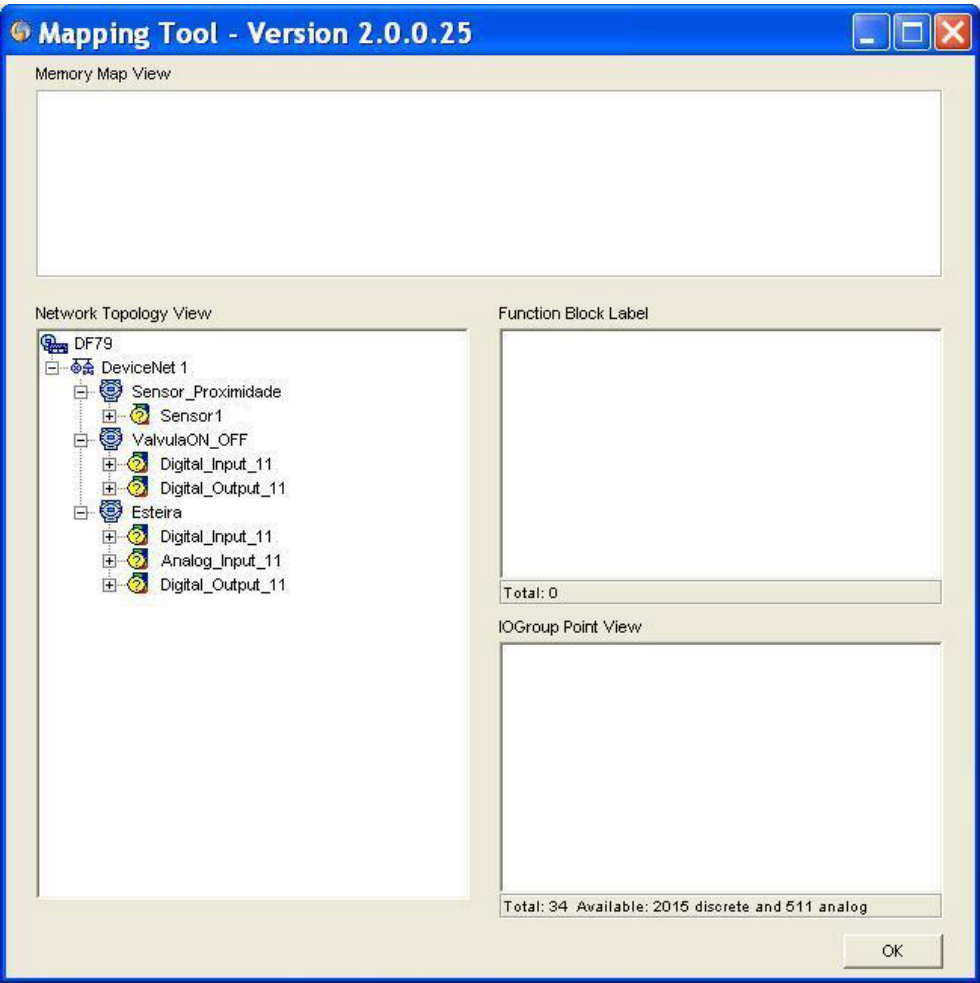



Figure 21. 21 – Mapping Tool window

In the last figure can be seen the **Network Topology View** window with the devices which were inserted in the DeviceNet network with their modules. In the **Function Block Label** window are the points which will be seen at **Syscon**, and the **IOGroup Point View** window are the points which will be seen at **LogicView for FFB**. The purpose is to configure each network “point” (or byte) in their respective data types.

The items with the symbol  have to be configured. Click the item which will be configured (for example, **Sensor1**), and in the window upper side will appear the available bytes. To the sensor was configured 1 byte for data, and only the first bit has the sensor's value. See the next figure.

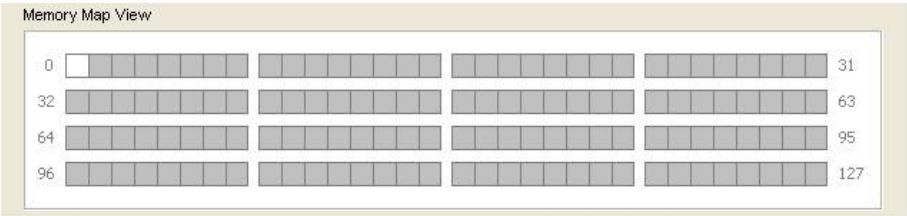


Figure 21. 22 – Available bytes

Click the byte, and the following window will appear.



Figure 21. 23 – Configuring the data type

Choose the variable data type of the slave device (**Data type**). The available types will depend on the chosen device. To **Sensor1** the type is **Bit**. In this example was chosen bit, because only one data bit will be configured.

The following steps have instructions about how to use the points, which were created in the **Network Configurator** and mapped in the **Mapping Tool**, in the control logic. As above mentioned there are two ways to use inputs and outputs points – ladder logic or function blocks.

Mapping DeviceNet IO Points to be Used in Ladder

Step 8

First the points will be mapped to be used in ladder, thus the **Block Type** (None) option will not be configured. The user can give a tag to group in **Group tag** option. If **Bit** option is selected the following window will appear.



Figure 21. 24 – Configuring the bits

Select the bits which are necessary to the application, click **Finish** and the configured points will appear in the **Mapping Tool** window. It is important the user correctly select the bit of the desired information. For example, for the **Sensor_Proximidade** that has only one data bit, the bit selected is the bit 0.

When analog data type was chosen the scale parameters for this point has to be configured. For example, for the **Esteira** that has one analog input of 12 bits the sensor scale must be from 0 to 4095.

Map Wizard

Set Scale Parameters for this point.

Sensor 100	100
Sensor 0	0
PV 100	100
PV 0	0
Units Index:	%

< Back Finish Cancel

Figure 21. 25 – Setting the scale parameters

Right-clicking this point and choosing **Delete** it can be deleted, or some attributes can be defined for better identification through the **Attributes** option.

IOPoint Attributes

User Tag:
Esteira_DI1

Value Description:
Retorno da esteira

Status Description:

OK Cancel

Figure 21. 26 – Changing the point attributes

NOTE

For the analog points is important to know the bytes order (big endian or little endian). The greater number of DeviceNet devices has the little endian standard. To view or change the bytes order in the **Mapping Tool**, right-click the device and select **bit** or **little**.

Fill the necessary fields, and click **OK**.

After mapping and configuring all points, click **OK** to exit from **Mapping Tool**.

After finish the mapping, a FFB block has to be inserted in the configuration using the **Syscon**. This block is necessary to edit the ladder logic. For further details refer to Adding Function Blocks section or **Syscon** manual. Right-click FFB block, which was added, and then click **Define Parameters**.

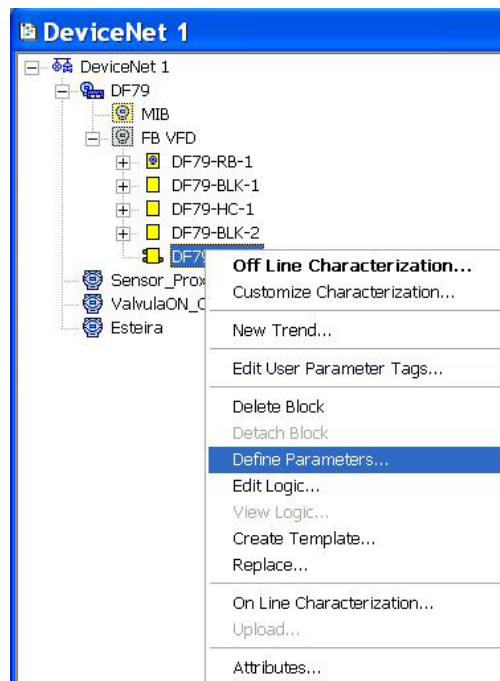


Figure 21. 27 – Defining FFB parameters (1)

The following window will appear.

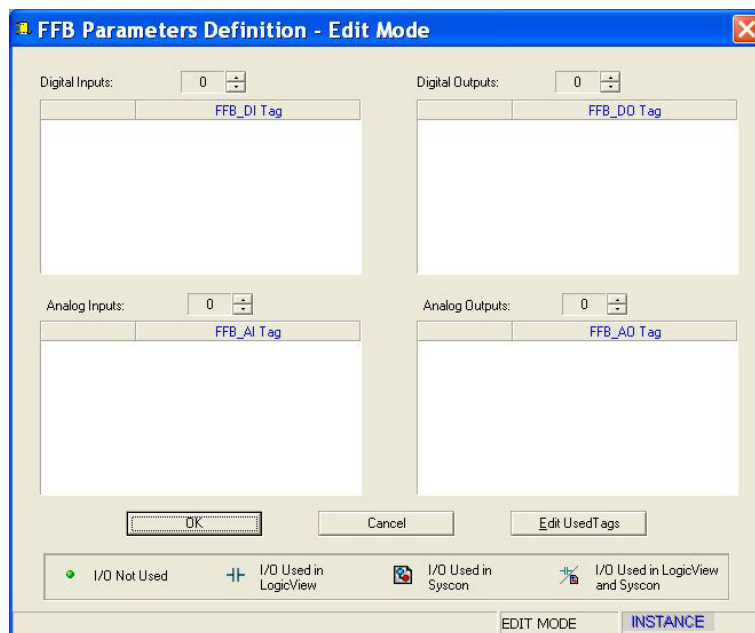


Figure 21. 28 – Defining FFB parameters (2)

If necessary define the inputs and outputs of FFB. Otherwise just click **OK**.



Return to **Syscon**, save the configuration, and do an **Export tags**.



Figure 21. 29 – Export tags

After finish this operation, right-click the FFB block, and then select **Edit Logic**. The **LogicView for FFB** will launch, and the ladder logic can be edited.

For the first time the logic is edited the **Refresh Data** command has to be executed thus the IO points configured in **Mapping Tool** will be updated in the ladder. Right-click **Network I/O** in the **Hierarchy** window of the **LogicView for FFB**. See the next figure.

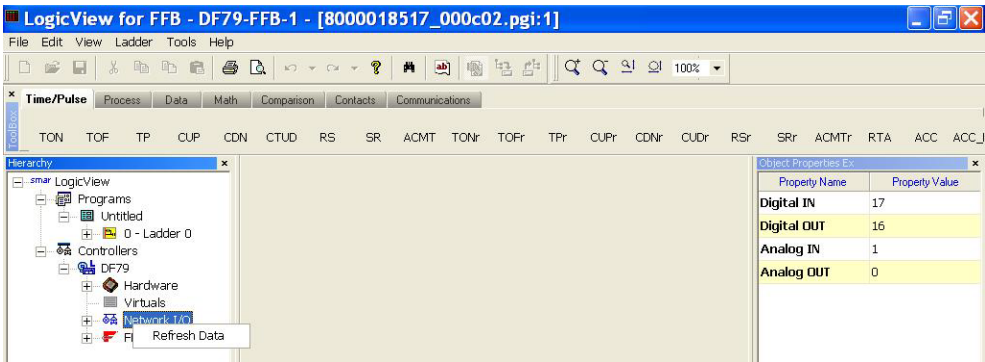


Figure 21. 30 – Refresh data

The NetIO points can be seen in the **Hierarchy** window. See the next figure.

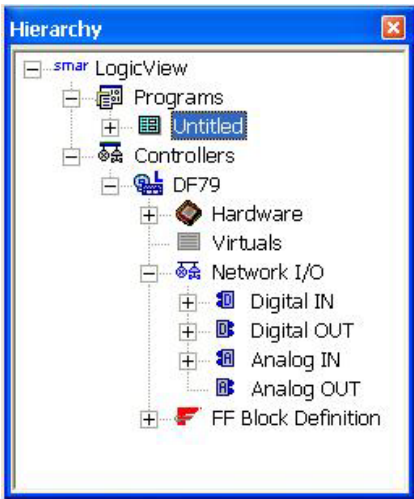



Figure 21. 31 – Network I/O at LogicView for FFB

After that procedure the IO point of the DeviceNet network are already available to be used in the logic. The user has a complete function library which can be used in the logic. For further details see the **LogicView for FFB** manual.

See in the following figure the proximity sensor's point linked to TON block (timer). All DeviceNet IO points appear with the symbol  in the ladder logic.

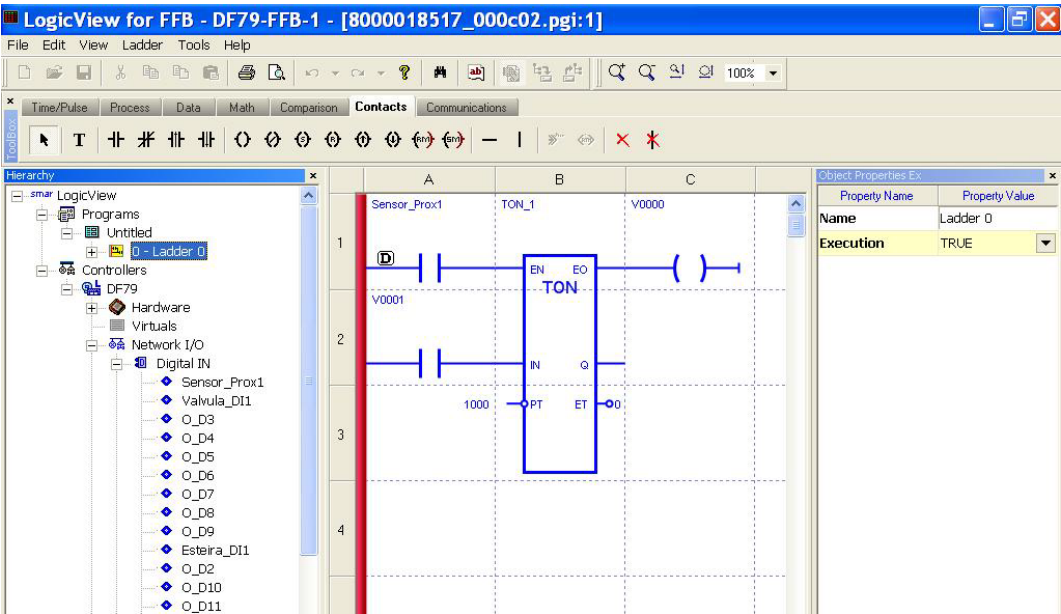



Figure 21. 32 – Editing logic

After ladder configuration the user can download the configuration to the device. Another option is to configure the function blocks at **Syscon**. If you also want to work with function blocks is necessary to follow the steps 9 to 12 to edit them. In this topic only the ladder was used to configure the ladder. Before downloading the configuration you have to save it in the **LogicView for FFB**. Exit from it, and return to **Syscon**.

To start a communication with the devices, first, is necessary to commission the controller. In this way the tags, IDs and addresses of each device will be assigned correctly. If this procedure was not done, the **Syscon** will detect the uncommissioned device and its download will be canceled. When the devices' commissioning finishes the download process can start. The download process can be done, for example, returning to **Proj_DF79** window, right-clicking **Fieldbus Networks** icon, , and then, select **Download** option. For further details about commissioning and possible download types refer to **Syscon** manual.

Mapping DeviceNet IO Points to be Used in Function Blocks

In the same way of the configuration done through ladder, the configuration can be done with function blocks. In this topic the same DeviceNet IO points of the step 8 will be mapped, but now the function blocks will be used. As above mentioned, for the network creation and devices' insertion the steps are common up to step 7.

Step 8'

In the Mapping Tool window, to map the IO points in blocks, is necessary to configure the **Block Type** field as you can see in the following figure. When **None** is chosen, as in the step 8, the mapping will be done in ladder.

To map the **Sensor_Proximidade** input, for example, choose the **Bit** data type. And then, in **Block type** choose **Multiple Digital Input**. Give a tag to the block in **Block Tag**.

NOTE

Every point of DeviceNet network is already configured automatically to work with ladder. To map in function blocks the user has to choose this option configuring the **Block Type** field of the **Map Wizard** window of **Mapping Tool** with the respective IO block.

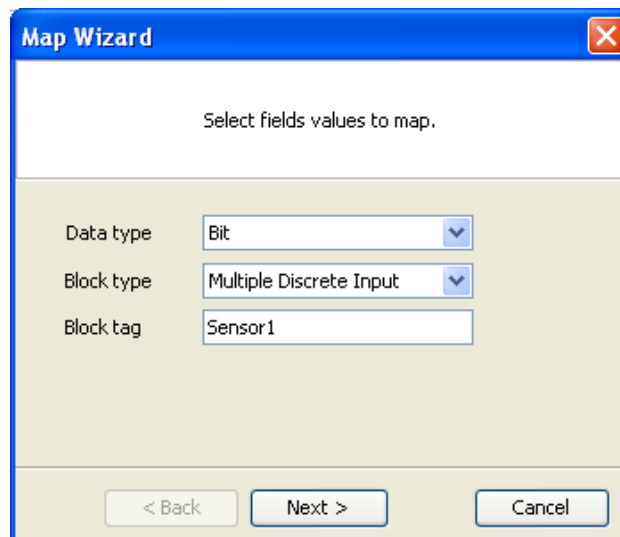


Figure 21.33 – Configuring the data type

Click **Next**, and a window as the figure 21.24 will open, select bit 1. Click **Finish**, and the configured point will appear in the **Mapping Tool** window.

NOTE

The **Mapping Tool** (MT) always tries to minimize the number of created blocks in the configuration. So, for a same slave device, if was not chosen any block tag the MT will allocate the point in an already existent block (the allocation availability will be verified, that is, the block has to be the same type and the block has to have empty points).

If the user informs an existent block tag, the MT will use this block to allocate the point. If the user informs a block tag that does not exist, the MT will create a new block.

Right-clicking this point and choosing **Delete** it can be deleted, or some attributes can be defined for better identification through the **Attributes** option.

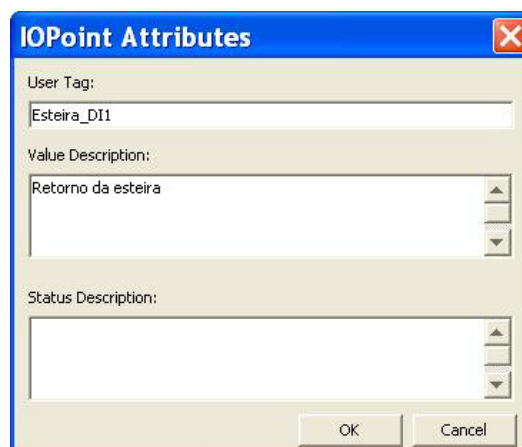


Figure 21.34 – Changing the point attributes

Fill the necessary fields, and click **OK**.

After mapping and configuring all points, the **Mapping Tool** will be as in the following figure.

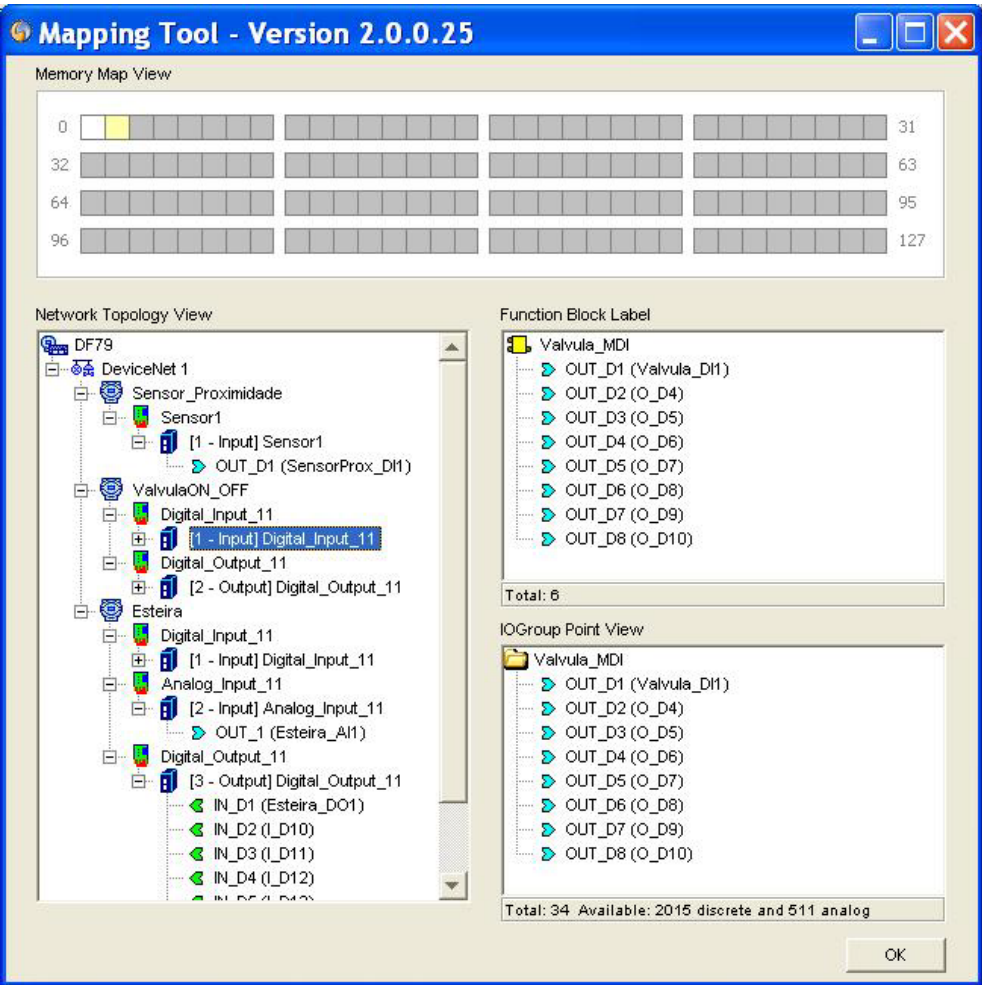


Figure 21. 35 – Mapped points in the Mapping tool

Click **Ok** to exit from **Mapping Tool**. Return to **Syscon**. The mapped blocks are already inserted in the area.

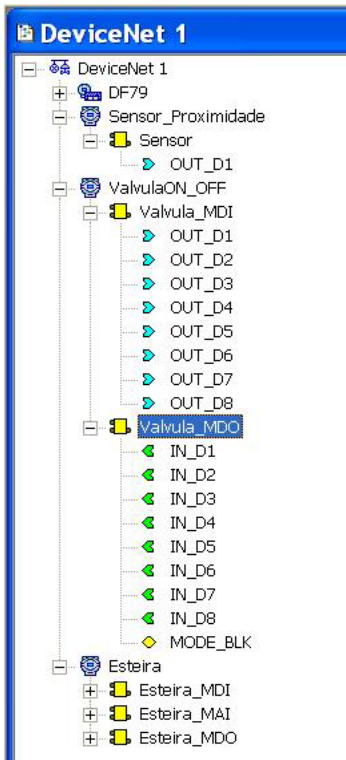


Figure 21. 36 – Mapped function blocks in the Syscon

The device's attributes can be verified right-clicking it. The following window will appear, and the information cannot be modified.

The 'Device Attributes' window is a standard Windows-style dialog box with a blue title bar. It contains three text input fields:

- 'Vendor ID' containing the text 'Wago Corp.'
- 'Product Code' containing the text 'WAGO 750-306 V04.05'
- 'Address' containing the text '3'

 At the bottom right corner, there is an 'OK' button.

Figure 21. 37 – Verifying the device's attributes

The block's attributes can be verified right-clicking it. The following window will appear, and the information cannot be modified.

The 'Block Attributes' window is a standard Windows-style dialog box with a blue title bar. It contains several text input fields:

- 'Manufacturer' containing 'SMAR - Unregistered'
- 'Device Type' containing 'DF79 - 8000018517_000c02'
- 'Device Rev.' containing '02'
- 'DD Rev.' containing '01'
- 'CF Rev.' containing '01'
- 'Block Type' containing 'Multiple Discrete Input'
- 'Profile' containing 'Standard'
- 'Block Tag' containing 'Valvula_MDI'

 At the bottom, there are three buttons: 'OK', 'Cancel', and 'Help'.

Figure 21. 38 – Verifying the block's attributes

Adding Other Function Blocks

Step 9

Now the user can add Function Blocks that will be used in the strategy.

To insert a new FB (Function Block), click the sign , right-click the **FB VFD** (Virtual Field Device) icon and choose **New Block**. The FB VFD is responsible for the data management.



Figure 21. 39 – Inserting new function blocks to the device

The **Function Block Characterization** dialog box will appear. The **Block Type** option has the available function blocks.

Select the block in the **Block Type** box and type its tag in the **Block Tag** box:

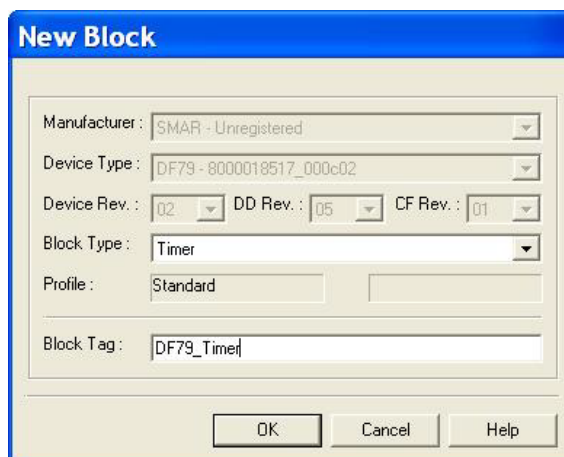


Figure 21. 40 – Selecting the function blocks that will be added to the device

For this example, the Timer and Constant blocks should be added to the DF79. The DeviceNet network can be seen in the next figure.

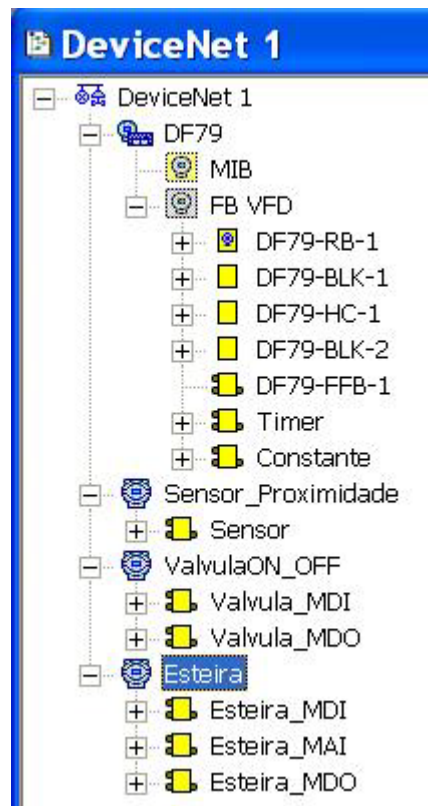


Figure 21. 41 – DeviceNet network with devices and function blocks inserted

Creating New Areas

Step 10

The user can create the strategy for the **Application (Logical Plant)**. First, it is necessary to create a new area.

The **Logical Plant** can be divided in several areas according to the area.

To create a new Process Cell, right-click the **Application** icon and choose **New Process Cell** item.

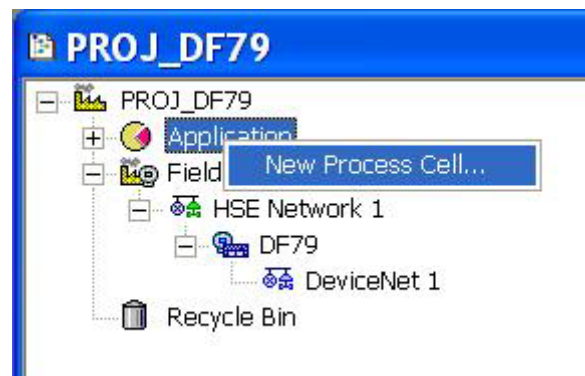


Figure 21. 42 – Inserting the Process Cell

The dialog box to name the **Process Cell** will open:

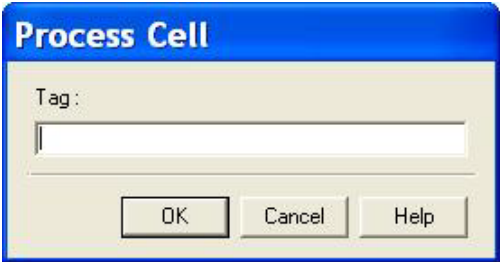


Figure 21. 43 – Naming the Process Cell

If the user needs to give a name for the Process Cell, enter the name in the tag box and click **OK**. The default tag is **Process Cell 1**. This number will increase when new areas are created. To create more areas, the user can repeat the last step.

After inserting a Process Cell, the **Proj_DF79** window will be according to the following figure.



Figure 21. 44 – Area window with the Process Cell added

NOTE
The user must remember the Application is just a virtual division, and its purpose is to divide a large project. For example: if the plant has 2 networks, they can be called Process Cells in the Syscon . One Application can have several Process Cells , but the Process Cells can not be in more than one Application .

Creating a Control Module

Step 11

Right-click the **Process Cell 1** icon and choose **Expand** item.



Figure 21. 45 –Creating a control module

To arrange the screen, click the **Process Cell 1** window. So, choose **Window** menu on the **Syscon** toolbar and then **Tile** option.

As following, return to the **Process Cell 1** window. Right-click the **Process Cell 1** item and choose **New Control Module**. The next figure shows creating the New Control Module.



Figure 21. 46 – Creating a New Control Module

The **New Control Module** dialog box will appear. Name it with the tag related to the Application Area. To conclude this task click **OK**.

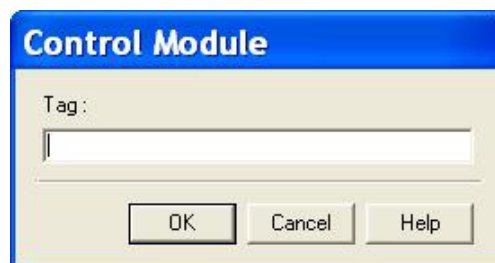


Figure 21. 47 – Naming the Control Module

IMPORTANT

Remember that not all characters are valid when naming the elements with tags.

Inserting Blocks to the Control Module

Step 12

The user can insert (attaching) the blocks for the corresponding devices in the **Logical Plant**. Right-click the **Control Module 1** icon, and then select **Attach Block** option, as shown in the following figure.

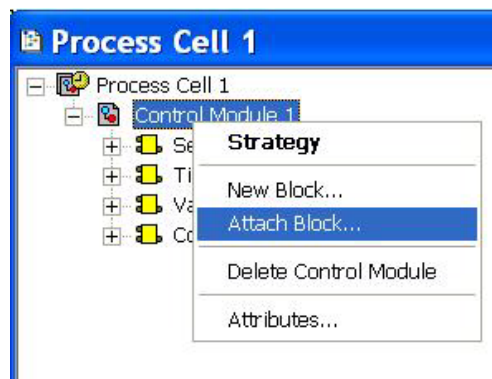


Figure 21. 48 – Attaching blocks to the Control Module

The **Attach Block** dialog box will open as shown in the following figure.

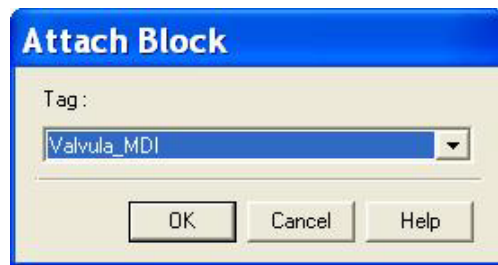


Figure 21.49 – Inserting blocks to the Control Module

The available function blocks for the FB Application are in the **Attach Block** box. For the aimed control, the function blocks that must be inserted will appear in this box. So, select them, one by one, and click the **OK** button.

For the example of valve opening control, at the end of Attach Block process, the **Control Module** will be as in the next figure.

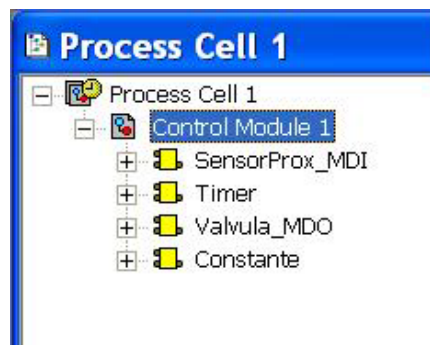


Figure 21.50 – Blocks inserted in the Control Module

Another way to attach the blocks to the FB Application is left-clicking the element and dropping it to the window.

Configuring the Control Strategy

Step 13

Now the user is ready to develop the control strategy.

First, right-click the **Control Module 1** icon, and then select **Strategy** item. The Strategy window will appear as shown in the following figure.

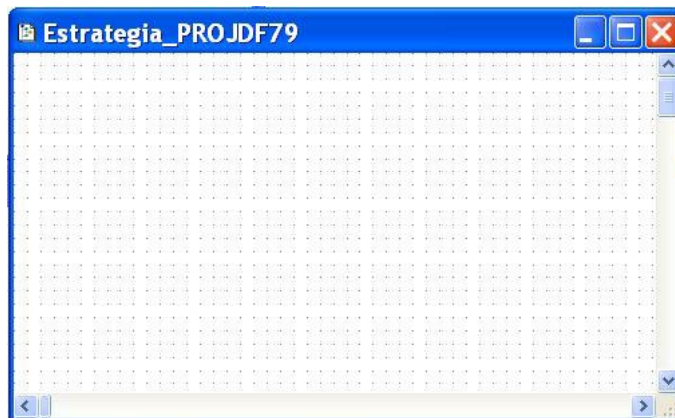


Figure 21.51 – Strategy window

The Control Module attributes can be changed by right-clicking the **Control Module 1** icon, and then select **Attributes** option. Do the necessary changes, and click **OK**. For further information about Control Module see **Syscon** manual.

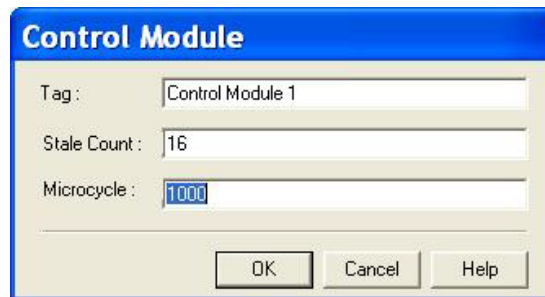


Figure 21.52 – Changing the control module attributes

At this moment there are 3 or 4 windows opened in the **Syscon**. To arrange the screen, click the title of the **Proj_DF79** window. On the toolbar, choose **Window → Tile**.


If the user does not have a monitor upper than 17", it is recommended to minimize the strategy window. Thus, it is possible to see the whole area.

The strategy window offers several tools for drawing. Refer to the **Syscon** manual for further details.

Adding Blocks in the Strategy Window

Step 14

Now the function blocks can be added to the **Estrategia_PROJDF79** window.

In order, click the first block,  **SensorProx_MDI**, and drop it into the strategy window. A function block will be created automatically.

The next figure shows the function block added to the strategy window:

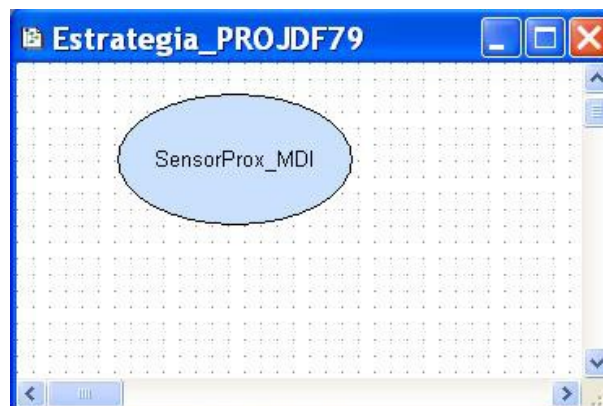


Figure 21.53 – Block inserted in the strategy window

The drag-and-drop procedure must be repeated for the other blocks.

Linking the Blocks

Step 15

There is a specific tool to link the blocks, the **Link** button,  on the **Strategy** toolbar. For the aimed control link the **OUT_D1** output of the SensorProx (where the proximity sensor was mapped) to the **IN_D1** input of the TIMER block.

Click this button on the toolbar, and then in the SensorProx_MDI function block. The dialog box for linking the input and output parameters will appear. Select **OUT_D1** block output, and then click **OK** as shown in the following figure.

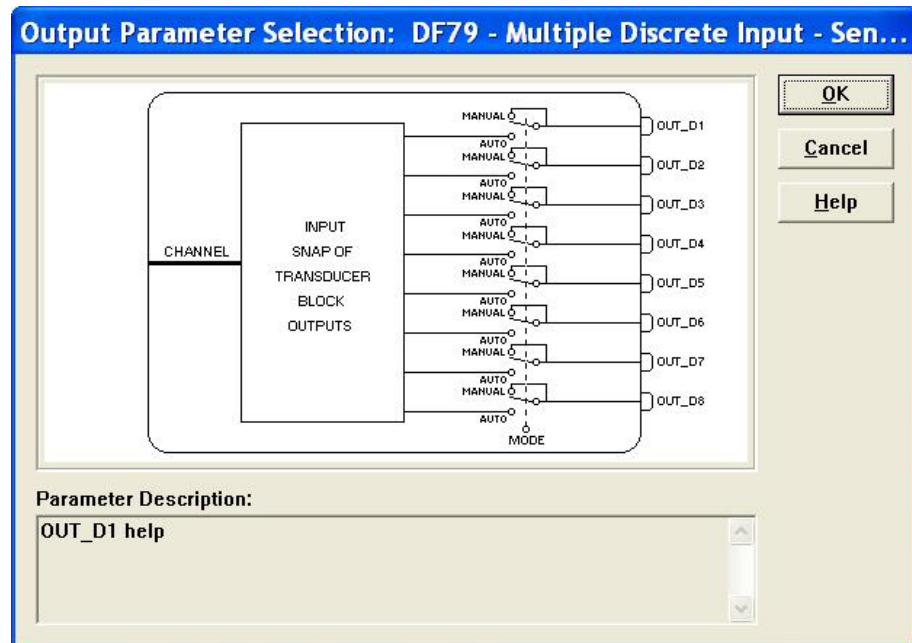


Figure 21. 54 – Linking the function blocks

The user also does the fast link procedure just right-clicking the function block, and does the necessary links.

After linking the specified parameters above, the strategy window will be as shown in the following figure.

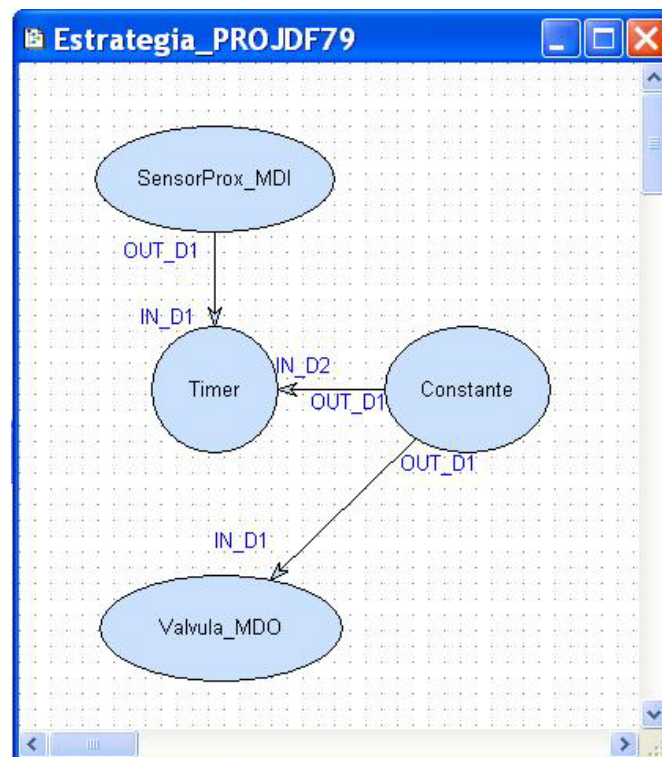


Figure 21. 55 – Linking the Blocks

Block Characterization

Step 16

The function blocks that are in the area must be set according to the application for them. So, it is necessary to do the block characterization.

The online and offline modes are possible for the block characterization. In the offline mode, the parameters are set before starting the communication among the devices. The online characterization is executed directly in the devices when the plant is already communicating, and the download to the devices was done.

To change the function block parameters, consider the following topics:

1. In the Strategy window

Select the block to characterize. Right-click it, and select **Off Line Characterization** or double-left-click it. The next figure shows the block that is being done the offline characterization.

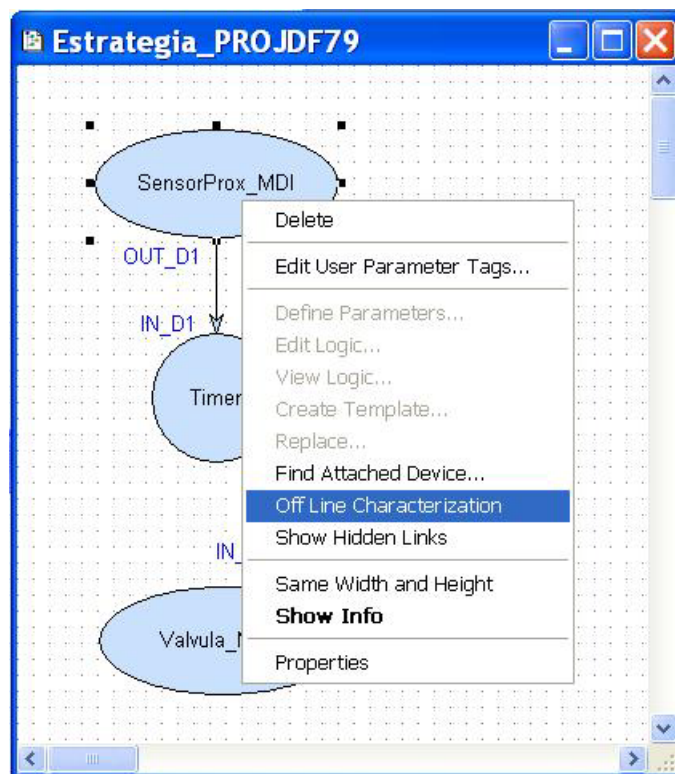


Figure 21. 56 - Offline characterization in the Strategy window

2. In the DeviceNet1 window

Another way to do the offline characterization is right-clicking the function block, and then selecting the **Off Line Characterization** option, as shown in the following figure.

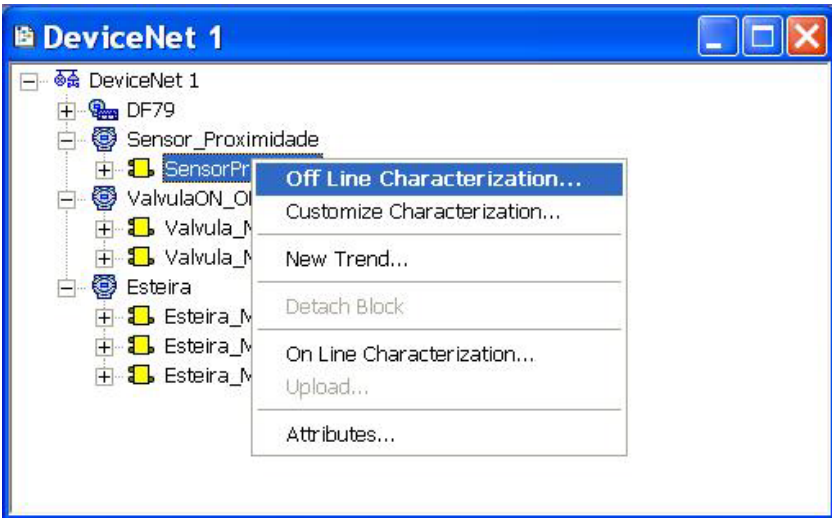


Figure 21. 57 - Offline characterization in the DeviceNet1 window

For both situations, the **Block Characterization** dialog box will appear:

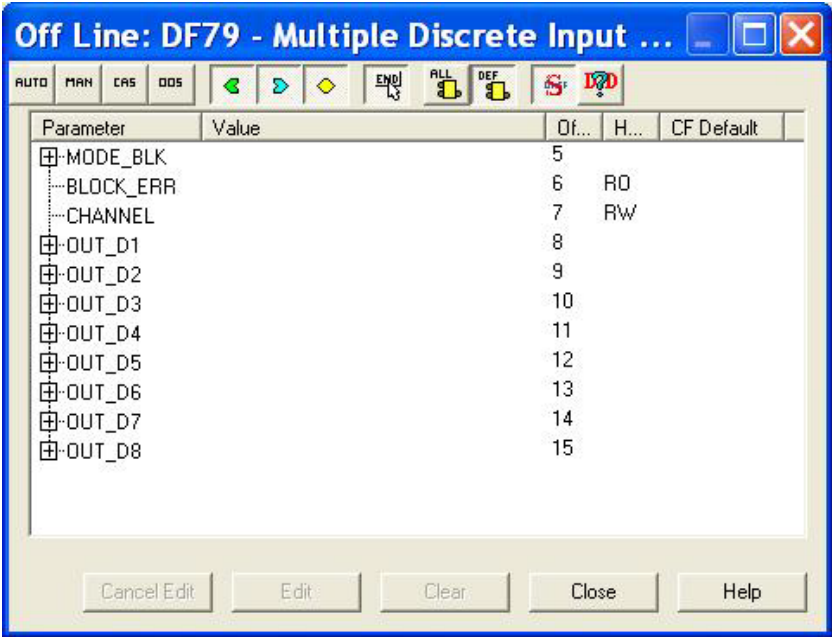


Figure 21. 58 - Function block characterization dialog box

Double-click at the right side of the parameter to change it. Another option is click once and then in the **Edit** button to start editing the parameter value. At the ending, click **End Edit** button.

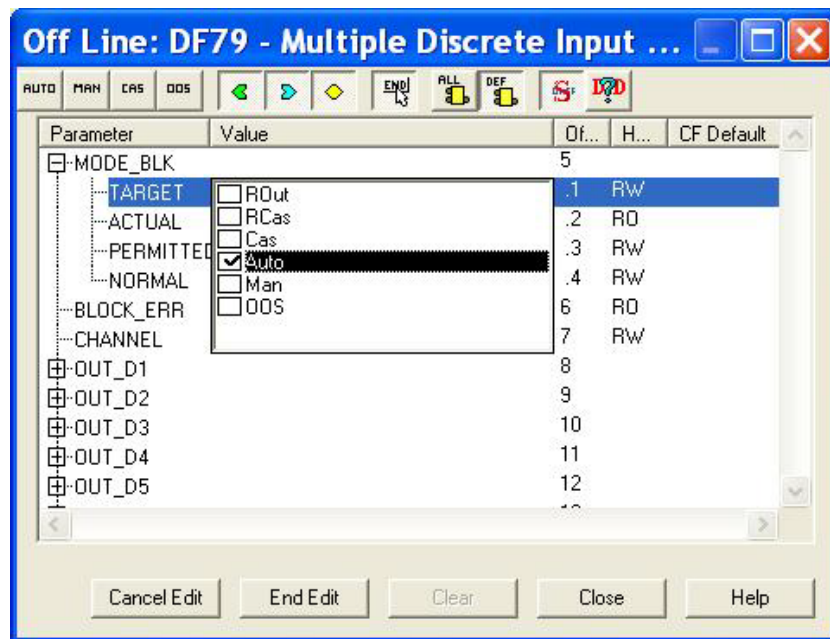


Figure 21. 59 - Editing the parameter in the Function Block Characterization box

An important parameter which has to be configured in all function blocks is **MODE_BLK**. It determines the block operation – Out of Service (OOS), Manual (MAN) or Automatic (AUTO). All blocks for the aimed control have to be with **MODE_BLK.TARGET** parameter in Auto.

Each block has a specific configuration parameter. For further information about each block refer to Function Blocks manual.

NOTE

For the IO DeviceNet blocks is not necessary to configure the **CHANNEL** parameter. This parameter only will be used when IO blocks are used to map local IO points (IMB).

Commissioning and configuration download to the controller


After the parameter setting, the user can start the device communication. It is necessary commissioning the devices to attribute the tags, IDs and device addresses properly. If this procedure is not executed, the **Syscon** will detect the uncommissioned device and the download for this device will be canceled. Finishing the device commissioning, the download process can start.

The download process (**Syscon**) can be done in two main ways:

- Download of the plant – in this case will be downloaded all configurations of all controllers.
- Individual download in the controller – in this case will be downloaded all configurations of a single controller.

The controller configuration can be performed in several ways depending of the alteration. But the download performed by **Syscon** is the most recommended, because it controls the changes of all tools.

Syscon has two download types: total and incremental. The main difference is that the “total” cleans initially the configuration before downloading the new one while the “incremental” only downloads that has changed.

The download process can be executed, for example, returning to the **Proj_DF79** window, right-clicking the **Fieldbus Networks** icon, , and then selecting the **Download** option

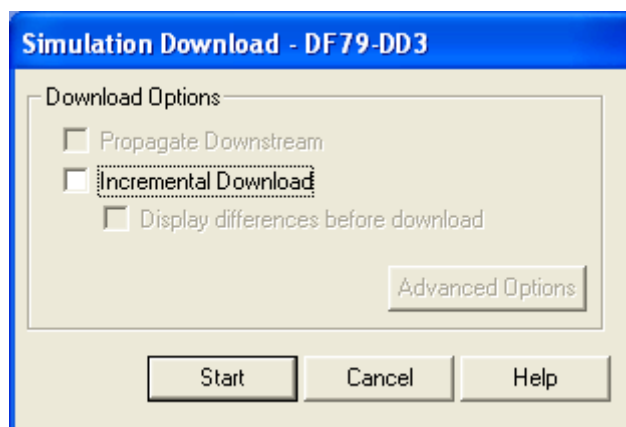


Figure 21.60 – Download options

Selecting the incremental download, and then **Advanced Options** the user can select several items to be downloaded. See the figure below.

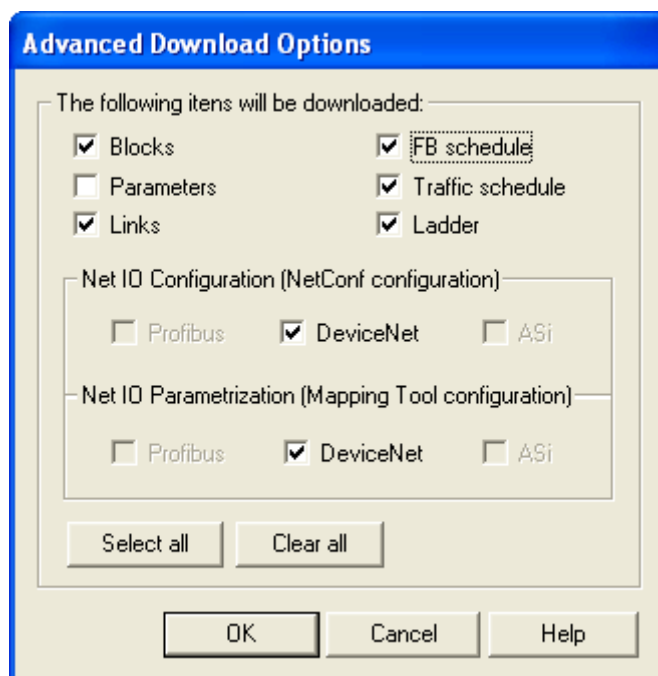


Figure 21.61 – Advanced download options

The **NetIO Configuration** option refers to the network topology configuration edited by the **Network Configurator**, and if it is selected the incremental download will be done. This operation must be done carefully because it stops the DeviceNet network.

The **NetIO Parameterization** option refers to the points parameterization configuration of network devices (points' tags, scales, data types) and when it is selected the incremental download will be done.

The next topic is maintenance guide which informs the recommended download type depending on the operation performed.

For further details about the available download types and commissioning, refer to the **Syscon** manual.

SYSTEM302 maintenance procedure

In this section is presented a summary of download procedures that are recommended in cases of configuration maintenance using **SYSTEM302**.

SYSTEM302 has several download types and here is recommended a general type to meet the majority of cases. However other ways can be used. For further information refer to **Syscon's** manual.

In the column **Recommended Download** is the type of chosen download. At **Syscon** it can be incremental on the plant or in the controller. Refer to **Syscon's** manual for further information.

MAINTENANCE EVENT	SOFTWARE	RECOMMENDED DOWNLOAD
Do a new SYSTEM302 configuration using DeviceNet controller and ladder logic.	Syscon, Network Configurator, Mapping Tool, LogicView for FFB	In Syscon do the plant incremental download*
Changing only the ladder logic.	LogicView for FFB	In LogicView for FFB do the download of the logic.
Add FFB parameters and make HSE links or only add parameters to the FFB block that already has HSE link.	Syscon	In Syscon do the plant incremental download*
Add or remove a FFB link to internal blocks within the same controller.	Syscon	In Syscon do the controller incremental download*
Add a new device in the network or change its configuration (address, cyclic mapping, tag)	Network Configurator	In Syscon do the controller incremental download*
Change device configuration (address, cyclic mapping, tag)	Network Configurator	In Syscon do the controller incremental download*
Change the interface parameters of the field device (scale, little/big endian, accessed bytes)	Mapping Tool	In Syscon do the controller incremental download*
Change media parameters (baud rate, communication configuration).	Network Configurator	In Network Configurator do the download of DeviceNet network*
Change configuration parameters of the field device hardware (acyclic configuration) or enable/disable the slave device.	Network Configurator	In Network Configurator do the download of DeviceNet network*

Table 01 – Recommended download procedures

Note: * These operations will stop the DeviceNet channel.

DeviceNet Advanced Topics

The DeviceNet protocol has two basic message types - IO messages or IO connections, and explicit messages. Each one is appropriate to the particular type of data, as described below:

IO Messages: type of synchronous telegram dedicated to move priority data between a producer and one or more consumers. Divided according to the data exchange method that are:

Polled	Communication method in which the master sends a command message to a specific slave (peer to peer) and the slave's response is directly to the master.
Change of state	Communication method in which data exchange between master and slave only occurs when there are changes in the monitored/controlled values up to certain time limit (heartbreaker). When this limit is reached the transmission and reception will occur even if no changes were done.
Bit-Strobed	Communication method in which the master generates a multicast request in the network bus and all the slaves with bit-strobed communication response one after another.
Cyclic	Communication method in which the slave updates its data in the network master at pre-defined time intervals.

Explicit message: type of general use telegram and non-priority. It is mainly used in asynchronous tasks such as device's parameterization and device's configuration.

UCMM (Unconnected Message Manager): it is responsible to process the requests and responses of the unconnected explicit messages. These characteristics are supported by devices with peer to peer communication capability.

Devices which support UCMM must support group 3 messages. The devices which do not support UCMM support messages of group 2 Unconnected. Group 3 or group 2 Unconnected are acyclic message types to establish initially the connection between master and slaves or for acyclic information exchange among them.

The DeviceNet protocol requires a connection to a device must be previously established in order to exchange information with that device. To establish a connection each DeviceNet product may send an UCMM message group 3 or Group 2 Unconnected. After establishing communication, the connection is then used to exchange information from one node to another, or to connect additional I/O. Once the I/O connections have been established, I/O data can be transferred between the devices in the network.

ATTENTION

To establish a communication with the device the master has to know if it supports UCMM or not. By default the DF79 assume that every slave device does not support UCMM. Information about support or not to UCMM are from the device's manufacturer or can be discovered through self-detected device shown later in this section.

The words "produced" and "consumed" are used below. They have as reference the slave device. The "produced" word indicates that the slave produced reading data to the master (DI, AI), that are produced by slave, and the consumed word indicates writing data deriving from master (DO, AO) that are consumed by slave.

I/O Messages Configuration in the Network Configurator

The configuration of I/O messages is done in the **Network Configurator** as in the step 6 previously shown. The figure below shows **IO connection** information.

Figure 21. 62 – Editing window of IO connection parameters

The information from **Actual chosen IO Connection** is dependent from the configured slave device. This information is obtained from manufacturer EDS file. Generally all devices support Polled type communication (Poll). It is possible to configure more than one communication type to the same slave since it supports it, for example, polled and bit-strobed.

The **UCMM check** (Group 1, 2, 3) option has to be selected when the slave supports UCMM messages.

The **Expected packet rate (EPR)** means:

- When **IO Connection** is **Cyclic** the EPR controls the time that the data will be produced.
- When **IO Connection** is **Change of state** the EPR indicates the watchdog time of the connection (heartbreaker).

The **Production inhibit time (PIT)** is the minimum delay time, in milliseconds, to produce new data. In this case the device suppresses new data production up to the PIT to expire.

The **Watchdog Timeout Action** field defines the watchdog function when the device's watchdog timer expires. In this case it may have the following actions:

- **Transition to Timeout:** The connection assumes the timeout status and remains in this state until it to be reset or deleted.
- **Auto Delete:** The connection is automatically deleted when there is inactivity or a watchdog timer occurs.
- **Auto Reset:** The connection is automatically reset when inactivity or watchdog timeout occurs.

The **Fragmented Timeout** field is the *timeout*, in milliseconds, when there are fragmented messages. Any messages larger than 8 bytes will be fragmented. For example, a device with more than 8 bytes produced or consumed creates a fragmented message.

Network Configurator online communication with the controller

With the **Network Configurator** is possible to communicate directly with the DF79 controller. It is important to perform some functions such as configuration download to the DeviceNet slave, to check some network diagnostics, viewing and changing slave parameters, baud rate changes, and others.

This communication can done via serial port (connecting a serial cable to the computer and to the controller's RS-232 port) or through Ethernet network. To inform which serial port will be used, first select the configuration master device, and then in the **Network Configurator** toolbar select **Settings→Device Assignment**. The following figure will appear.

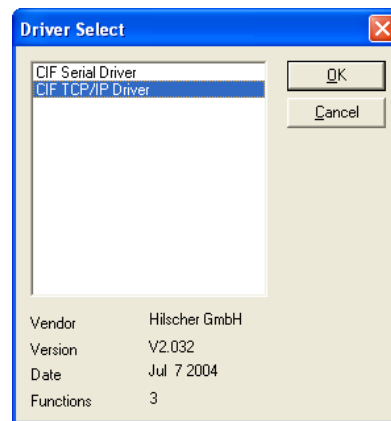


Figure 21. 63 – Choosing the communication driver

Select the **CIF TCP/IP Driver** option if the TCP/IP port will be used for communication; it is the typical case, and click **OK**. The following figure will appear. Then, enter the master's IP, and click **Add**.

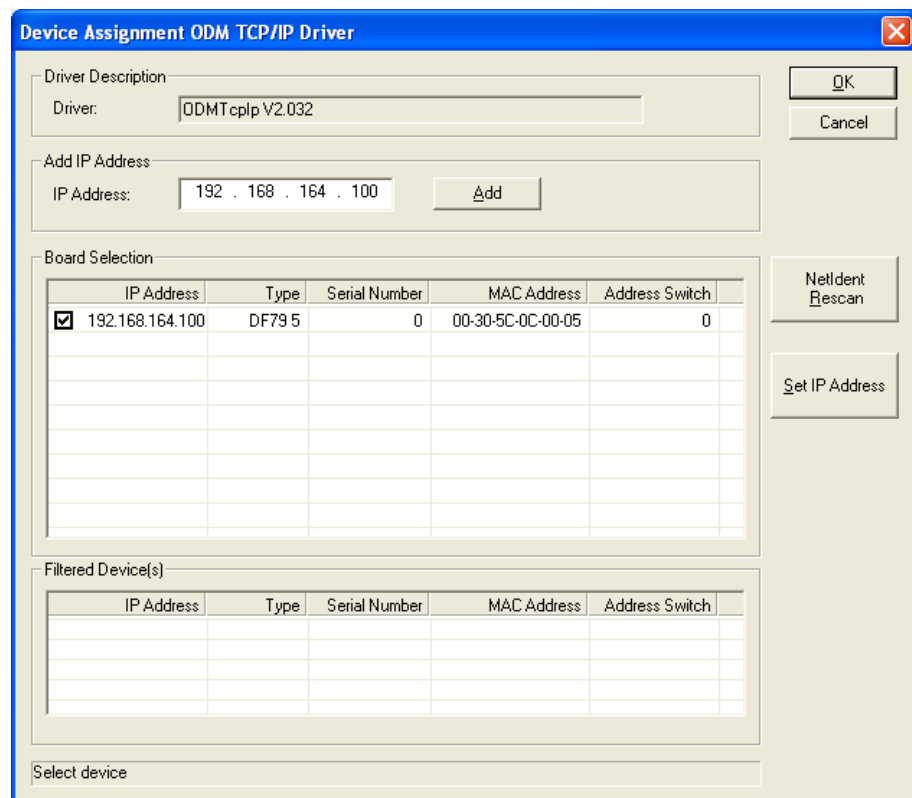


Figure 21. 64 – Device Assignment of the master device

In the **Board Selection** frame a line for a controller on the network should appear. Select this line, and click **OK**.

After these steps the **Network Configurator** already communicates online with the controller. In the following sections are detailed the main operations of online communication with the controller.

ATTENTION

After to establish the online communication the user can download the configuration by the **Network Configurator**. However this download is incomplete unlike that done by **Syscon**. The download by **Network Configurator** only deals with the communication with slaves, and does not deal of the control logic.

Therefore, every time that some equipment is included, the amount of IO message parameters or the devices' tags are changed the user has to do a download by **Syscon**.

Explicit message configuration

To configure the devices which have explicit message configuration is necessary to click **Parameter data** button in the device configuration window.

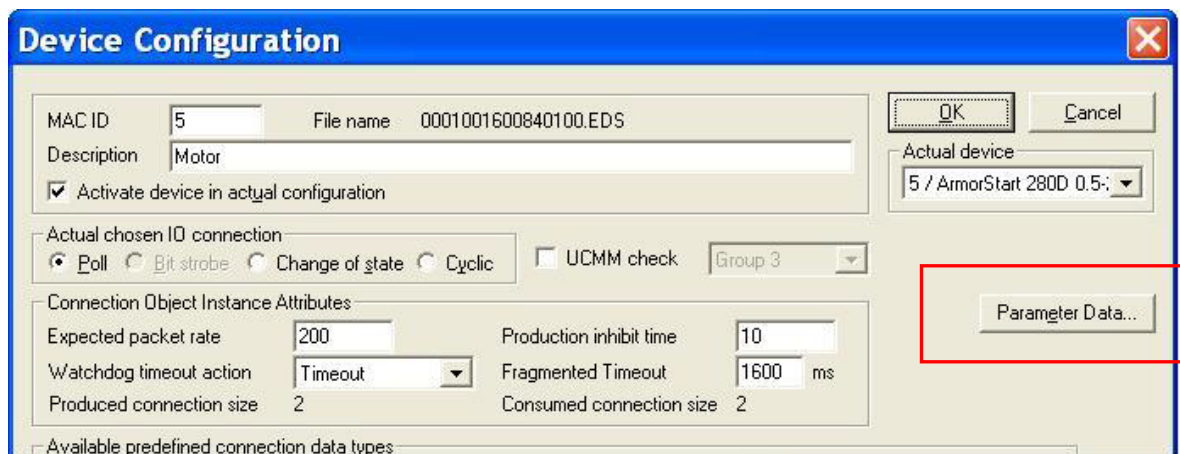


Figure 21. 65 – Parameter Data Option

All the slave's parameters will appear. See an example in the following figure. The parameter data window is divided in two parts: the supported parameters are in the upper part (in this case, the *ReadOnly* and *ReadWrite* are showed) and in the bottom part are the parameters configured by user.

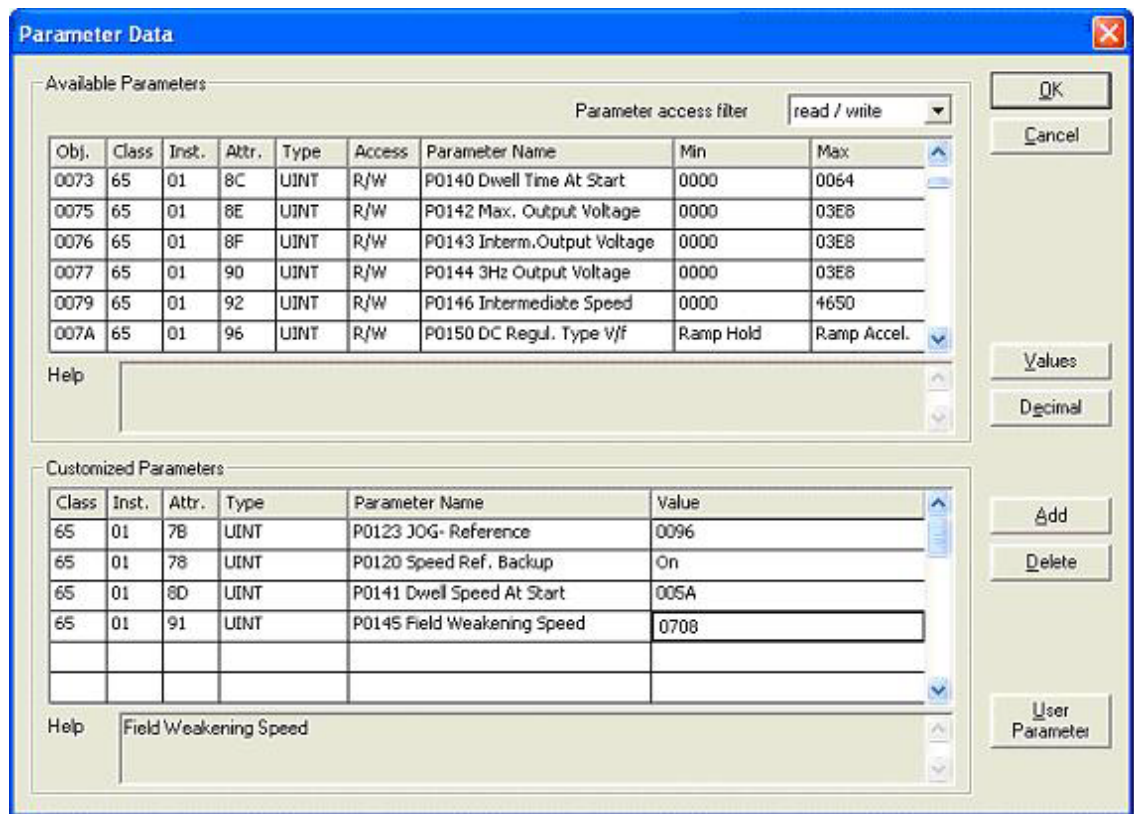


Figure 21.66 – Editing window of the explicit message parameters

To configure a parameter the user has to select it in the **Available Parameters** section. Double-clicking the parameter, automatically it enters in the user-configured parameter list in **Customized Parameters**. In this section the user can change its value and save it to be downloaded with the configuration. The parameters which have "R/W" (ReadWrite) in the column **Access** can be configured by user.

ATTENTION

The change of configuration parameters is done offline and in order to change the new values in the slave devices is required to download (which can be used both by the **Network Configurator** or by **Syscon**). But all download operation stops the DeviceNet channel.

Online Reading/Writing of slave's parameters

The change of slave's parameters may be done online through explicit messages. For example, is necessary to change the value of **P0145 Field Weakening Speed** parameter of the previous figure without stopping the slave communication.

The online change is done through the **Get device attribute / Set device attribute** command.

To change or read a parameter, first, the online communication between the **Network configurator** and controller must be established, as mentioned before in the **Network Configurator online communication with the controller** topic. Then, in the **Network configurator** main window click **Online** menu, and select **Get/Set device attribute**. See the following figure.

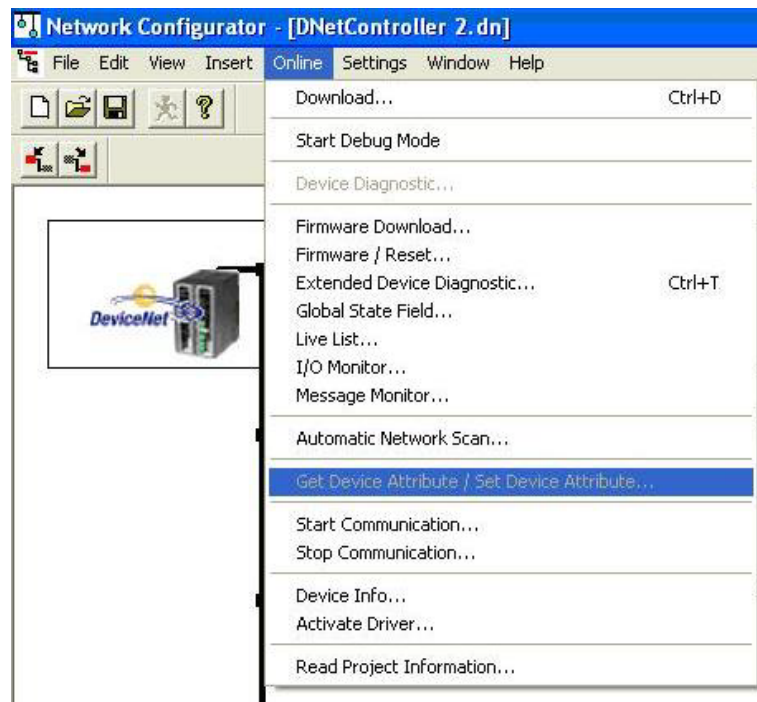


Figure 21. 67 - Online menu – Get/Set Device Attribute option

And then, inform the class, instance and attribute of the parameter. This information is in the **Parameter data** window in decimal values. Consider the example in Figure 21. 66, to change the **P0145 Field Weakening Speed** parameter is necessary to set class = 101 (0x65), instance = 1 (0x01) and attribute = 145 (0x91). Finally choose to read (**Get**) or to write (**Write**). The values always appear in hexadecimal. The next topic has an example of **get/set parameter**.

ATTENTION

Changing parameters online does not save them in the configuration. If a different value is set offline (as mentioned before) in the next download the parameters that were configured online may not have persistence.

Changing the address via software

The address changing of some DeviceNet devices must to be done only via software. In these cases the user must first communicate with the slave on its current address and in the **Get device attribute / Set device attribute** window to inform class = 03, attribute = 01, instance = 01. This address is standard and common to all devices.

For example, to change the address of PSH5 from 52 to 53, first communicate online with the device (at the address 52). Then in the **Get attribute/set attribute** window just configure class = 03, attribute = 01, instance = 01, and value = 35h (53 in decimal). See an example in the following figure:

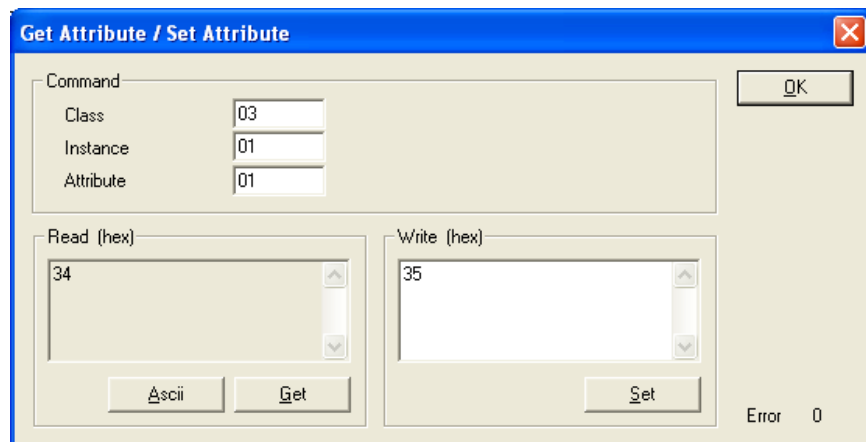


Figure 21. 68 – Get attribute/set attribute window

ATTENTION

The user can also change the slave address through the **Live list** option from the **Online** menu of **Network Configurator**. On DF79, this procedure may stop the DeviceNet channel, and the controller has to be restarted.

For further details refer to the **Troubleshooting** section of **SYSTEM302 Handbook**.

Enabling and disabling a device in the configuration

Another interesting feature of the **Network Configurator** is the ability to enable or disable a slave device in the configuration without affecting any configuration mapping or logic. This operation is done in the device configuration window (Figure 21.16) selecting or not the **activate device in actual configuration** check box.

When the device is disabled an **X** appears in the main window of **Network configurator** indicating that the master is not scanning it. In this state despite the controller does not change the logic configuration and the device's mapping, it does not perform any operation with this device (cyclic and acyclic communication).

The controller will recognize this changing only after the configuration to be downloaded. This download can be done via Network Configuration or via **Syscon**. Remember that this operation stops the DeviceNet network.

To enable again the device proceed in the same way described above. In this case, the **X** disappears from the **Network Configurator** main window and the controller returns to update the device's data in the logic.

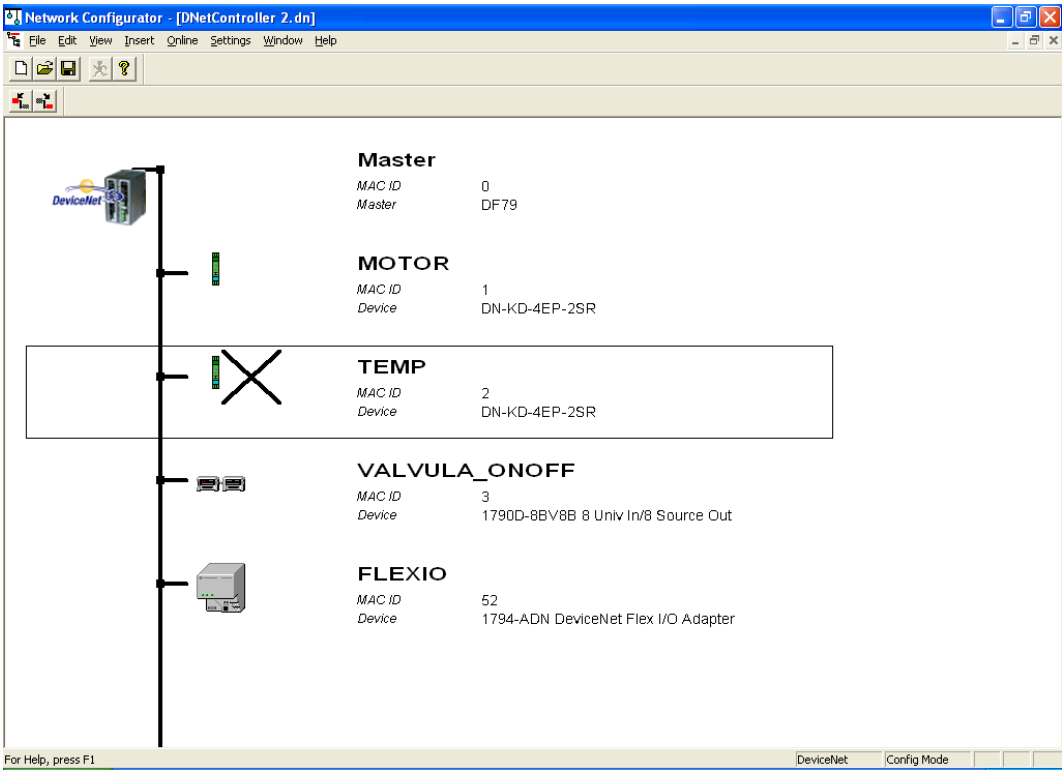


Figure 21. 69 – Example of disabled device (device with MAC-ID=2)

Network Diagnostics

There are many ways to identify communication failures in the DeviceNet network. Some options are: Network Configurator, Transducer block, or module’s LEDs. These options will be explained in this topic.

Network diagnostic using the Network Configurator

To do a network diagnostic through **Network Configurator** first is necessary to its connection to the controller (refer to **Network Configurator online communication with the controller** topic). Thus, some diagnostics are possible, such as Livelist, Start Debug Mode, IO Monitor, and others. They will be explained following.

Livelist

One of the first diagnostics that can be done is the **Livelist** of network devices. The DeviceNet **Livelist** is accessed through menu **Online → Livelist**. This Livelist shows the addresses of all active devices on the network, configured or not.

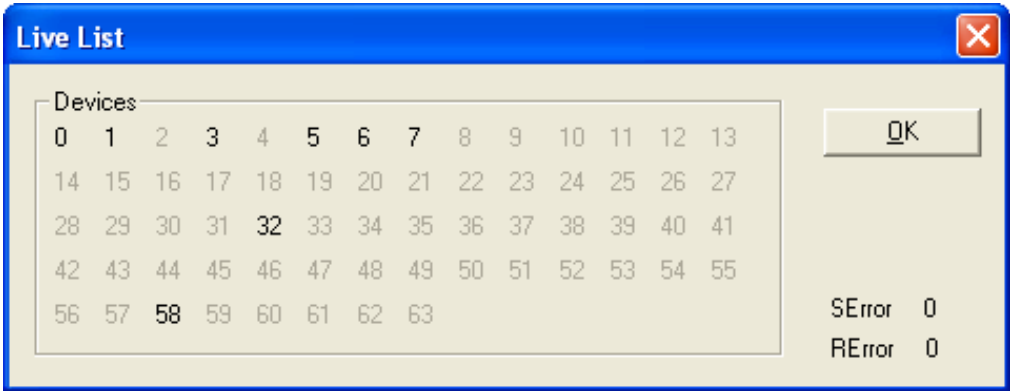


Figure 21. 70 – LiveList of the devices on the network

ATTENTION
<p>Sometimes after trying to access a device that is not configured is necessary to download the configuration through the Network Configurator to restart the communication of DeviceNet network or with the controller. However, from the moment the configuration is valid the download operation is unnecessary. This is necessary when, for example, there is a network physically configured, but the Livelist shows nothing of the device.</p> <p>Remember that every download operation stops the DeviceNet channel.</p>

Global Diagnostic

Another available diagnostic in the **Network Configurator** is the **Global State Field**. This diagnostic provides an overview of the configured devices. It is used when there are many devices on the network, because it concentrates on a single window the diagnostic of the network master and the devices’ status. To access this diagnostic click **Online→Global State Field**.

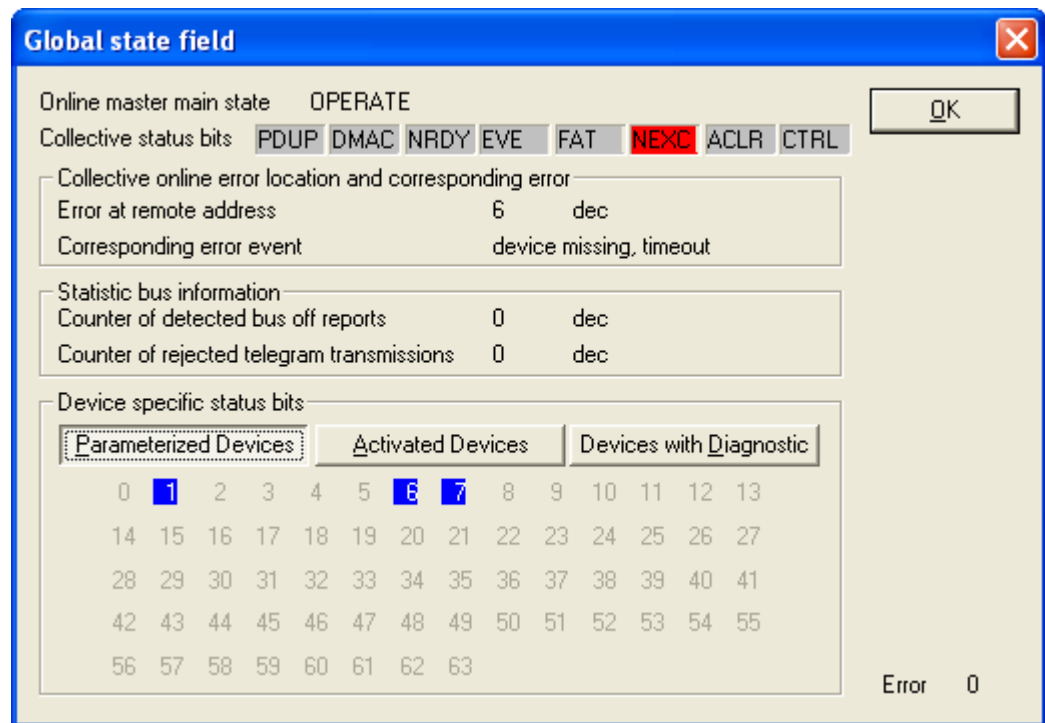


Figure 21. 71 – Global diagnostic of the network's events

The figure above has an example of global diagnostic. Each parameter will be explained below:

Online master main state: indicates the current status of the DeviceNet network master. Possible status:

OFFLINE (00h), STOP (40h), CLEAR (80h), and OPERATE (C0h).

Collective status bits: these bits indicate an overview of the network global events, deriving from the master and the devices.

BIT	MESSAGE	DESCRIPTION
0	PDUP	DUPLICATE-MAC-ID is performed – If this bit is activated the controller is checking the duplicate MAC_ID.
1	DMAC	DUPLICATE-MAC-ID detected - Indicates that the controller found on the network equipment with the same MAC_ID.
2	NRDY	HOST-NOT-READY-NOTIFICATION- If this bit is activated the controller is not available for communication.
3	EVE	EVENT-ERROR The controller detected a short circuit in the bus. The number of detected events is in the bus_error_Cnt variable.
4	FAT	FATAL-ERROR Due to information overload on the bus, the communication cannot continue, there is no available bandwidth.
5	NEXC	NON-EXCHANGE-ERROR Up this moment one or more devices do not perform the cyclic data exchange.
6	ACLR	AUTO-CLEAR-ERROR Controller stopped the communication with all devices and performed the auto-clear.
7	CTRL	CONTROL-ERROR Parameterization error.

Table 02 – Description of GLOBAL_ERROR parameter

NOTE

The bits shown on table 02 refer to **GLOBAL_ERROR** parameter of the transducer block described following. However, the bits of **Collective status bit** are inverted compared to the **GLOBAL_ERROR**, i.e., for the **Collective status bit** the flag **CTRL** corresponds to the bit 0 and **PDUP** corresponds to the bit 7.

Collective online error location and corresponding error: in this frame is specified in which network device the problem was detected, through the *Error at remote address* parameter, and in the *Corresponding error event* there is a brief description of it. If more than one device has an available diagnostic, on this frame the diagnostic that will appear is the one of the device with lesser MAC-ID.

Statistic bus information: in this frame appears the diagnostic of the physical network. Through the *Counter of detected bus off reports* parameter the low-quality transmission on the bus is counted (bus off) and the *Counter of rejected telegram transmissions* indicates the number of canceled transmissions and number of restarts of the CAN chip.

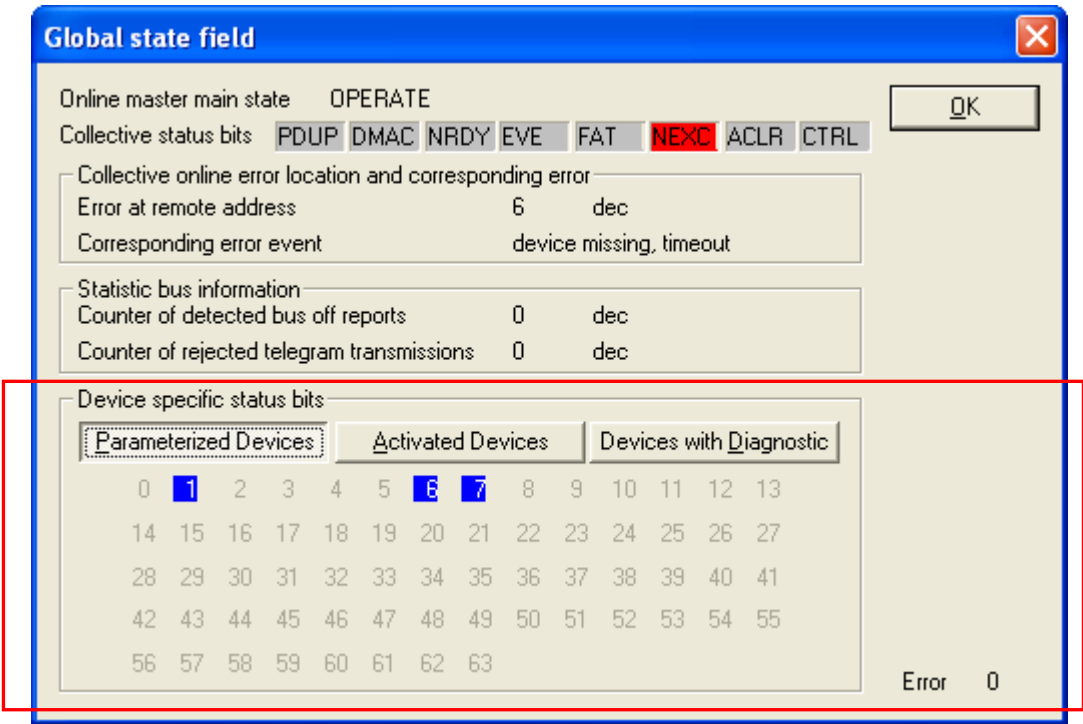


Figure 21. 72 - View of Parameterized Devices status bits

Device specific status bits: this frame presents a summary of the current status of all network devices.

- **Parameterized devices:** on this tab all included devices on the network and present on the last download are shown, even if they do not exist physically or have any kind of problem.
- **Activated devices:** on this tab all devices that reached the stage of data exchanging are shown. This indicates that master and slaves are exchanging cyclic I/O messages between them. Note that in the figure 21.72, compared to the previous figure, the device in the MAC-ID 6 is not present, because a **NEXC** event was reported to its address (*Error at remote address*)

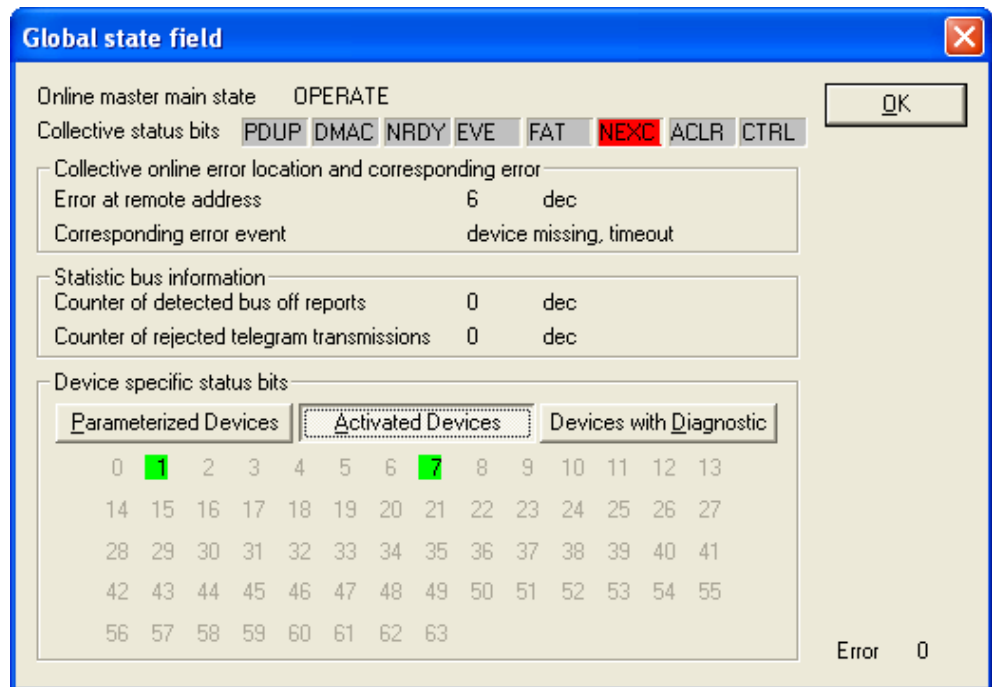


Figure 21.73 – Activated Devices status bits

- **Devices with Diagnostic:** in the following figure is presented a list of devices with some diagnostic. Note that the devices with MAC-ID 6 and 7 are on the list, but only the diagnostic of the lower MAC-ID is on **Collective online error location and corresponding error** frame.

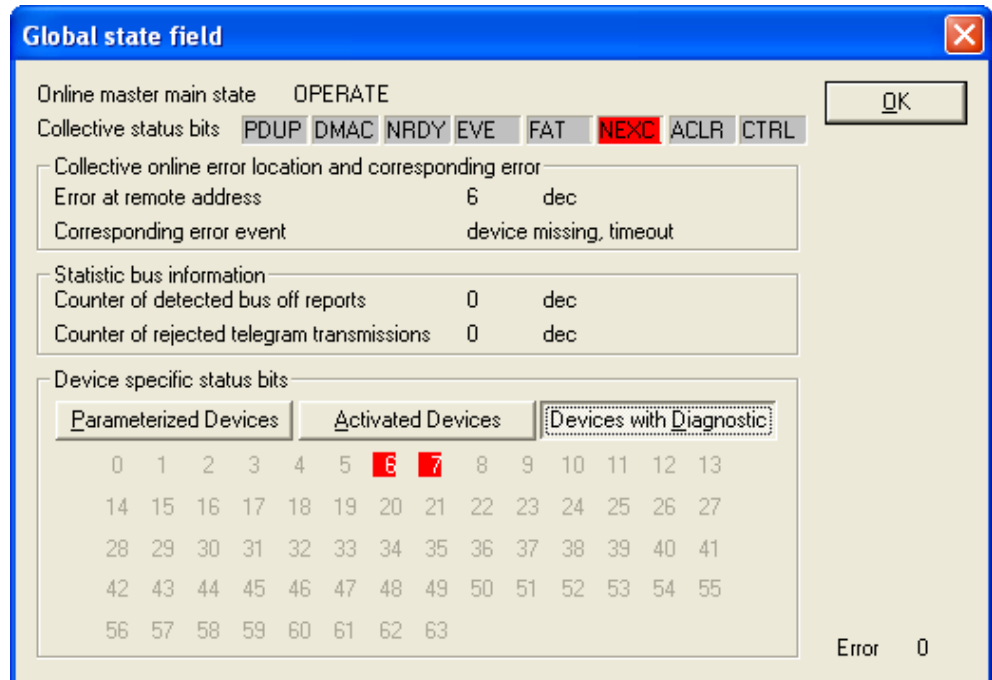


Figure 21.74 – Devices with Diagnostic status bits

By clicking the missing devices, a detailed diagnostic of the error can be obtained, similar to that presented in the Start Debug Mode, which will be shown in the next item. Notice in the following figure that after clicking the diagnosis for the MAC-ID 7, its corresponding bit in the list devices with diagnostic ceases to be red, i.e., the diagnosis was read, and only return to be checked if there is one more instance to that address.

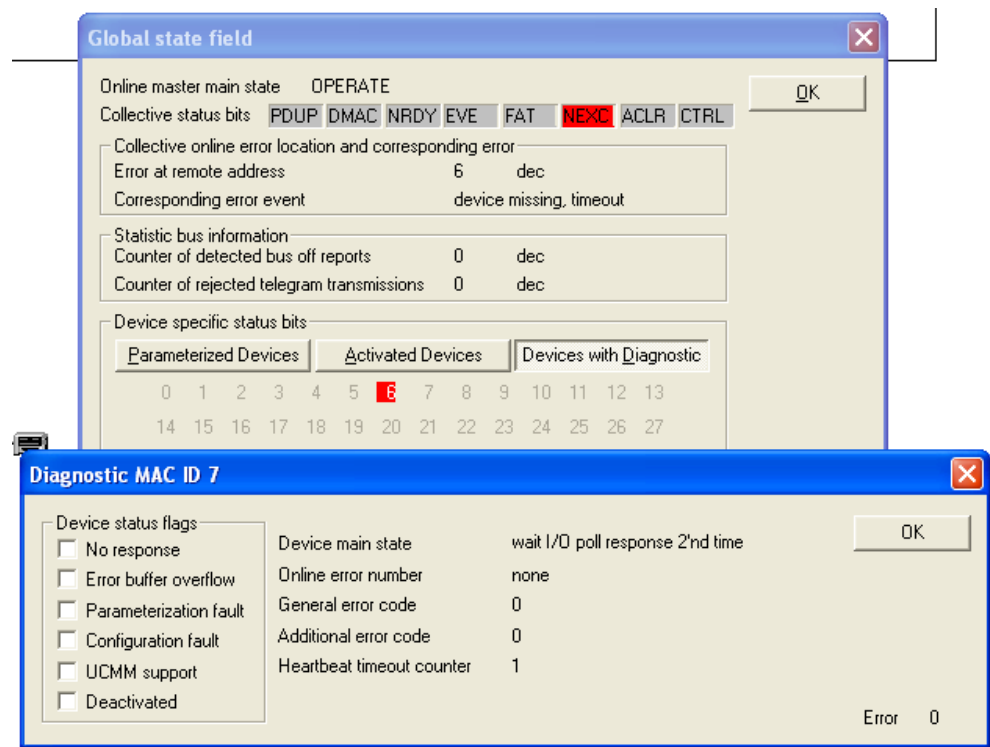


Figure 21. 75 – Detail of the error from the Devices with Diagnostic status bits list

Start Debug Mode

Another available diagnostic in the **Network Configurator**, for the network device status, is the **Start Debug Mode**. To access this diagnosis, select the master device, and click **Online→Debug Mode**. See the following figure.

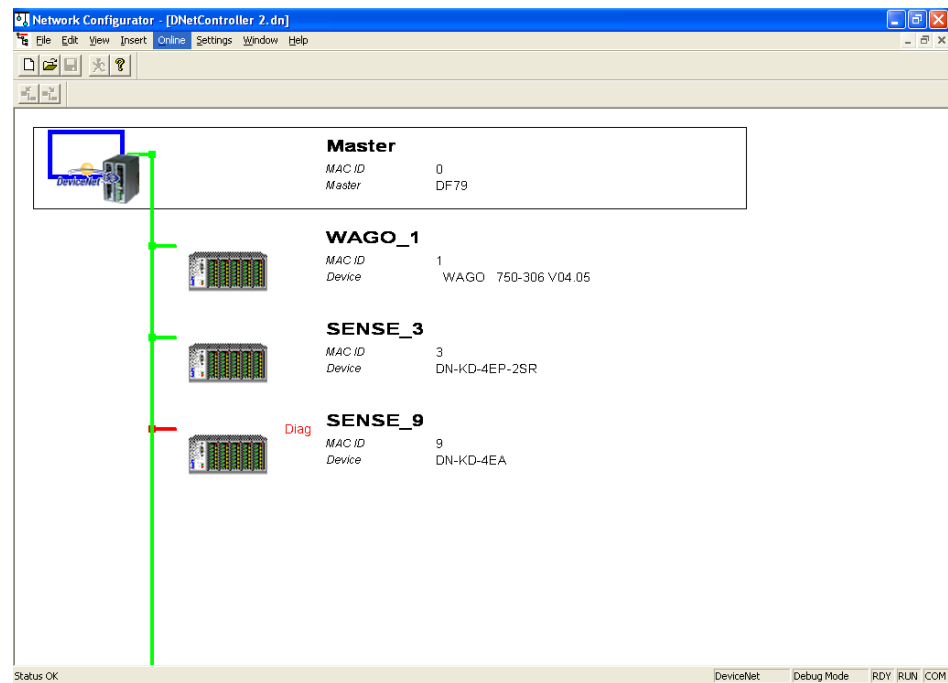


Figure 21. 76 – Doing the network diagnostic

In the previous figure there is a red-marked device with some available diagnosis. By clicking that device a window with diagnostic information will appear. See the next figure.

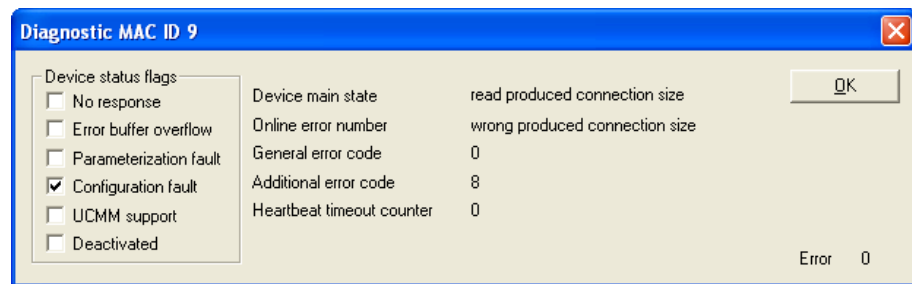


Figure 21. 77 – Description of device's failure

The window is divided into several fields detailing the diagnosis:

- Device status flag
- Device main state
- Online error number
- General error code
- Additional error code
- Heartbeat timeout counter

Each one of this field can help solving the problem in question. They will be detailed following.

Device status flag shows the most general device's diagnosis. In the table below there are its possible values.

BIT	FLAG	DESCRIPTION	POSSIBLE SOLUTION
0	<i>No Response</i>	Device does not respond to requests for communications.	Check installations, cables and baud rate
1	<i>Error Buffer overflow</i>	Internal buffer of device is not consumed.	Network with low time scan, device cannot send data produced.
2	<i>Parameterization Fault</i>	Equipment was denied access to at least one attribute configured for writing.	<i>Host</i> trying to write in internal attribute in inadequate time. For example, an attempt to write in speed set point of a frequency inverter during operation.
3	<i>Configuration Fault</i>	Configuration failure – difference between the configuration and the produced/consumed data by the device.	The quantity of produced/consumed configured is different from that available in the device.
4	<i>UCMM Support</i>	If this flag is selected the UCMM communication is supported.	Device can establish direct connections to other devices which support UCMM.
7	<i>Deactivate</i>	Device is deactivated in the current configuration.	Device was not commissioned and the configuration was not downloaded.

Table 03 – Description of Device status flags frame

Remember that “*produced*” indicates data which are produced by slaves and “*consumed*” indicates data which are consumed by slaves.

Device main state shows the device current status on the network, as follows.

VALUE	MESSAGE
0	State machine enter
1	Device inactive, not handled
2	Own MAC ID, state waiting for all incoming duplicate MAC-ID requests
3	initialize internal predefined master slaves structures
4	allocated predefined master slave connection set request
5	wait for predefined master slave allocation connection response
6	release predefined master slave connection set request
7	wait for predefined master slave release connection response
8	initialize internal I/O configured structures
9	allocate configured I/O connection request
10	wait for I/O allocation response
11	release I/O connection request
12	wait for I/O connection release response
13	read consumed connection size
14	wait for read consumed connection size response
15	compare consumed connection size with internal configured one
16	read produced connection size
17	wait for read produced connection size response
18	compare produced connection size with internal configured one
19	configure the I/O connection structures and register it
20	set expected packet rate
21	wait for set expected packet rate response
22	I/O poll request 1'st time
23	wait for I/O poll response
24	I/O poll request 2'nd time
25	wait for I/O poll response
26	I/O poll request 3'rd time
27	wait for I/O poll response
28	heart beat timeout to the device
30	open unconnected explicit connection request 1'st time
31	wait for unconnected explicit connection response
32	open unconnected explicit connection request 2'nd time
33	wait for unconnected explicit connection response
34	close unconnected connection request
35	wait for close unconnected connection response
36	release all established connections request
37	wait for connection release response
38	open user unconnected explicit connection request
39	wait for user explicit connection response
40	user predefined master slave allocate connection request
41	wait for user allocation response
42	user close unconnected connection request

43	wit for user close unconnected response
44	get or set user defined attribute request
45	wait for user defined get or set attribute response
46	send or wait fragmented get or set attribute

Table 04 – Description of device main state field

Online error number indicates the current error related to the device, listed following.

VALUE	MESSAGE	ERROR SOURCE	DESCRIPTION/ POSSIBLE SOLUTION
0	no error		
1	device guarding failed, after device was operational	Slave device	check if device is still running
30	device access timeout	Slave device	device do not respond, check the baud rate and its MAC-ID
32	device rejects access with unknown error code	Slave device	use single device diagnostic to get reject code
35	device response in allocation phase with connection error	Slave device	use single device diagnostic to get additional reject code
36	produced connection (process data input length in the view of the DEVICE) is different to the configured one	Slave device / Configuration	use single device diagnostic to get real produced connection size
37	consumed connection (process data output length in the view of the DEVICE) size is different to the configured one	Slave device / Configuration	use single device diagnostic to get real consumed connection size
38	device service response telegram unknown and not handled	Slave device / Controller	use single device diagnostic to get real consumed connection size
39	connection already in request	Slave device	connection will be automatically released
40	number of CAN-message data bytes in read produced or consumed connection size response unequal 4	Slave device	device don't have operability with DEVICE and norm description
41	predefined master slave connection already exists	Slave device / Controller	connection will be automatically released
43	sequence error in device polling response	Slave device	two first segments in multiplexed transfer were received
44	fragment error in device polling response	Slave device	fragmentation counter while multiplexed transfer differs from the awaited one
45	sequence error in device polling response	Slave device	middle or last segment was received before the first segment
46	length in bit strobe device response unequal produced connection size	Slave device	
47	sequence error in device COS or cyclic response	Slave device	two first segments in multiplexed transfer were received
48	fragment error in device COS or cyclic response	Slave device	fragmentation counter while multiplexed transfer differs from the awaited one
49	sequence error in device COS or cyclic response	Slave device	middle or last segment was received before the first segment
50	length in COS or cyclic device response unequal produced connection size	Slave device	
51	UCMM group not supported	Slave device	change the UCMM group
52	Device Keying failed: Vendor ID mismatch	Slave device / Configuration	check configured Vendor ID against devices Vendor ID

53	Device Keying failed: Device Type mismatch	Slave device / Configuration	check configured Device Type against devices Device Type
54	Device Keying failed: Product Code mismatch	Slave device / Configuration	check configured Product Code against devices Product Code
55	Device Keying failed: Revision mismatch	Slave device / Configuration	check configured Revision against devices Revision
59	double device address configured in actual configuration	Configuration	each device at DeviceNet must have its own MAC-ID
60	whole size indicator of one device data set is corrupt	Configuration	download error in the current database, contact technical support
61	size of the additional table for predefined master slave connections is corrupt	Configuration	download error in the current database, contact technical support
62	size of predefined master slave I/O configuration table is corrupt	Configuration	download error in the current database, contact technical support
63	predefined master slave I/O configuration does not correspond to the additional table	Configuration	number of I/O modules and the number of configured offset addresses are different
64	size indicator of parameter data table corrupt	Configuration	value of size indicator too small
65	number of inputs declared in the additional table does not correspond to the number in the I/O configuration table	Configuration	each entry in the I/O configuration must have only one entry in the additional table
66	number of outputs declared in the additional table does not correspond to the number in the I/O configuration table	Configuration	each entry in the I/O configuration must have only one entry in the additional table
67	unknown data type in I/O configuration detected	Configuration	support BOOLEAN, BYTE, WORD, DWORD and STRING only
68	data type of a defined I/O module in a connection does not correspond with the defined data size	Configuration	following type and size are valid BOOLEAN = 1 byte UINT8 = 1 byte UINT16 = 2byte UINT32 = 4byte
69	configured output address of one module oversteps the possible address range of 3584 bytes	Configuration	the process data image is limited to 3584 bytes
70	configured input address of one module oversteps the possible address range of 3584 bytes	Configuration	the process data image is limited to 3584 bytes
71	one predefined connection type is unknown	Configuration	support of cyclic, polled, change of state, bit strobed only
72	multiple connections defined in parallel	Configuration	supporting only one type of connection to one device
73	the configured Exp_Packet_Rate value of one connection is less than the Prod_Inhibit_Time value	Configuration	expected packet rate must be larger than the production inhibit time

Table 05 – Description of Online error number field

General error code provides further details about the error indicated by **Online error number**, such as type and error source. These values are defined in the DeviceNet specification and inserted into the list so that the error is detected. See an example for the error 35:

ONLINE_ERROR	DESCRIPTION	GENERAL_ERROR_CODES
35	Device rejects requested command with an error message.	Error code containment of the response 2 = Resources unavailable 8 = Service not supported 9 = Invalid attribute value 11 = Already in request mode 12 = Object state conflict 14 = Attribute not settable 15 = Privilege violation 16 = Device state conflict 17 = Reply data too large 19 = Not enough data 20 = Attribute not supported 21 = Too much data 22 = Object does not exist

Additional error code provides additional information about the error. This code is freely inserted by manufacturer and is only available when **General error code** is different from zero.

Heartbeat timeout counter counts the number of times the slave was not scanned on the network. This variable can indicate message transmission quality for this slave; since the **Expected packet rate** configuration parameter indicates the time that a data pack has to be detected. Double-click the device and the **Device Configuration** window will appear, thus is possible configure this parameter.

In the next topic are some more common errors in the DeviceNet device commissioning.

Common errors of commissioning

Device does not exist

In the error that appears in the figure above the **Device Status Flag** is indicating *No Response*. This error should indicate that the device is out of the network or it is not responding.

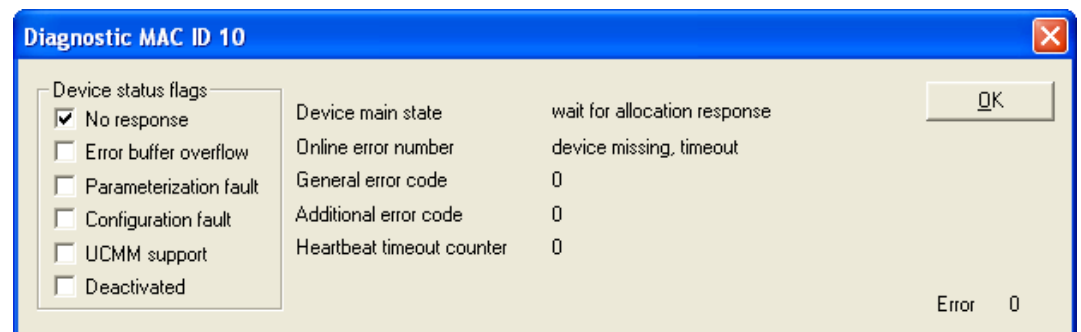


Figure 21. 78 –Error diagnostic window of a nonexistent device

If the device is on the network and is not responding, the following should be considered:

- Physically the device may not connected correctly (network LED is red);
- Device supports UCMM communication and this option is not selected in the configuration (refer to I/O Messages Configuration in the Network Configurator)
- The network is incorrectly configured (small time for EPR and PIT). In this case the master may not be scanning this device due to small time configured for PIT.

Failure of device configuration

The error of figure 21.77 means: The **Device Status Flag** is indicating **Configuration fault** because there is a problem in the device configuration. The **Device main state** and **Online error number** fields indicate that the device is “producing” a different number of bytes from the expected.

In the example above the number of I/O bytes configured is wrong. The correct number is 8 bytes – module with 4 analog inputs. But only 2 bytes were configured (as if it were using only the first input).

Failure in the online communication between the Network Configurator and the controller

The **Network Configurator** has two fields for errors that indicate the communication between the **Network Configurator** and the controller: **SError** and **RError**. These two fields are common for all online communication of the **Network Configurator** with the DF79 and are called **Online Data Manager**. The following figure shows an example of these errors.

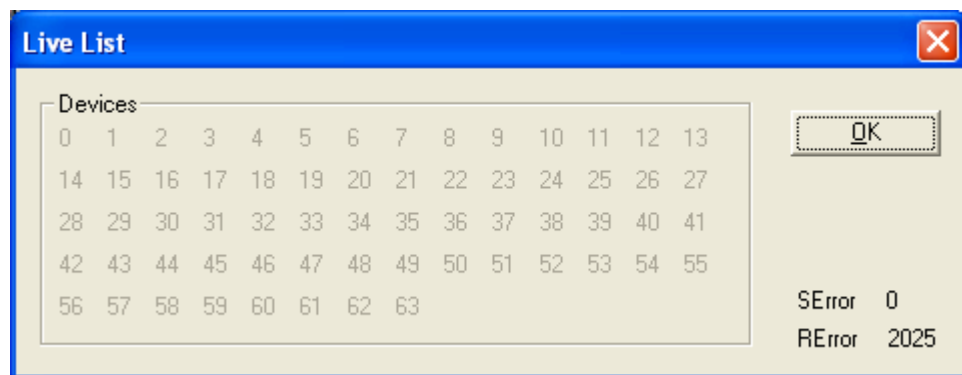


Figure 21. 79 –Example of SError and RError

The **SError** and **RError** fields can show a communication without errors when they are equal to zero. There most common error codes are listed below. Further information and other error codes are in the help of the **Network Configurator**.

VALUE	DESCRIPTION	POSSIBLE PROBLEM SOLUTION
0	Communication OK.	
2025	Message timeout	Exit from software and launch again.
8031	Send error	Exit from software and launch again.

Table 06 – Description of SError and RError codes

The auto detected devices

The **Network Configurator** can automatically recognize the device configuration (**Online** → **Automatic network scan** option). This is very useful when there is unknown device and is necessary to configure it for the first time. Below is an example of auto detection.

Address	Supported Functions	Device Name	Poll Size	Poll Size	BitStr. Size	BitStr. Size	Cyc/COS. Size	Cyc/COS. Size	Chosen Config.
			Produced	Consumed	Produced	Consumed	Produced	Consumed	
MAC ID 0	Not found								
MAC ID 1	Not found								
MAC ID 2	Cyc., COS, Bit	WAGO 750-306	6	5	1	8	2	5	Change of
MAC ID 3	Poll, Expl. Msg		1	0	0	0	0	0	Polling
MAC ID 4	Not found								
MAC ID 5	Not found								
MAC ID 6	Not found								
MAC ID 7	Not found								
MAC ID 8	Not found								
MAC ID 9	Not found								
MAC ID 10	Not found								
MAC ID 11	Not found								
MAC ID 12	Not found								
MAC ID 13	Not found								
MAC ID 14	Not found								
MAC ID 15	Poll, Expl. Msg		8	2	0	0	0	0	Polling
MAC ID 16	Not found								
MAC ID 17	Bit Str., Poll, Expl.		1	1	1	8	0	0	Polling

SErrors: 0
RErrors: 0

Figure 21. 80 – Example of autodetection

In the example above the device 2 supports **Cyclic**, **COS (Change of state)** and **BitStrobe** communication types, but the suggestion was “COS”. If the chosen option was COS the input bytes would be 2 (for DF79 would be *produced* for the slave) and the output bytes would be 5 (*consumed* for the slave).

Likewise for the device 15 which supports only **Polling** and **Explicit message** (acyclic data) communication types, the suggestion was **Polling**. It has 8 input bytes (produced) and 2 output bytes (consumed).

To effect the proposed configuration just click **Automatic Configuration** (but remember that it will erase the previous configuration). In the **Chosen config** column is suggested that cyclic communication is configured, but this choice can be changed.

NOTE

In the **Device name** column, sometimes, devices from some manufacturers are not recognized (that for some devices that do not have all information inside them). For example, The devices' names with MAC ID 15 and 17 were not recognized. When the device supports UCMM this information appear in the **Supported functions** column of the previous figure.

ATTENTION

- After the auto detection the user can accept or reject the new configuration. If the new configuration is accepted all previous configurations will be deleted and a generic EDS file will be used. It has only cyclic communication characteristics (I/O connection). If were necessary use acyclic parameters (explicit message) the user has to choose the correct EDS.
- The **Automatic configuration** only identifies the number of bytes; it does not specify the data types of the slaves. Thus a discrete device and an analog device (with Word, Dword or Byte) always will be their configuration mapped on “bytes” type.
- After perform an auto detection is necessary to restart the communication of all network devices. So, this function will stop the DeviceNet channel.

Devices diagnostic through their LEDs

Another diagnostic way is through the devices and DeviceNet master LEDs. These LEDs have standard network functions that will be common to any device. The table below shows names, colors, descriptions and behavior of the frontal LEDs of the DF79 controller.

LED	COLOR	DESCRIPTION	BEHAVIOR
DN	Blue	It indicates DeviceNet channel activity.	On: The DeviceNet communication was activated and all active devices in the configuration are working normally. Off: DeviceNet network was not configured, or inversion of the power cables or the data cables. Blinking: Some active device in the configuration is missing or has a problem. Verify the device or deactivate it in the configuration (a download is necessary).
ERR	Red	It indicates it there is an error related to the DeviceNet network.	On: Error in the DeviceNet network (short circuit, noise, terminator is missing). The 24V is missing, some node is missing or with error. Off: The DeviceNet network is configured and all active devices are working normally. Blinking: The DeviceNet communication was deactivated.
LINE	Green	It indicates that the bus is powered with 24V.	On: Powered bus. Off: Bus is not powered Blinking: NA*

Table 07 – Description of the DeviceNet network LEDs for the DF79 controller

*NA – Not applicable

The table below shows the description of the DeviceNet network LEDs in the slave devices.

LED	COLOR	DIAGNOSTIC	ACTION
Net	Green	On: The communication on the bus is ok.	On: No action
		Off: Bus is not powered.	Off: Power the bus with 24V.
		Blinking: Master is not scanning the slave.	Blinking: Check if there is a master on the line, if yes repeats the download. <ul style="list-style-type: none"> In DF79 if the point is mapped on the Network Configurator and was not mapped on Mapping Tool the controller will not configure any I/O message to that device. Check if the network time scan is small.
	Red	On: There is a physical failure on the line (a short circuit was detected). Address conflict.	On: Check DeviceNet network problems: <ul style="list-style-type: none"> Power off the device or the network (<i>reset</i> the device). Check if the network impedance is 60 ohms (2 resistors of 120 ohms at each end). Check if the device is the unique at that address (after changing the address you must reset the slave).
		Off: Physical communication is ok.	Off: No action.

Table 08 – Description of DeviceNet network LEDs for the slaves

Diagnostic through Transducer block

Location of Global Diagnostic in the Communication Transducer

The communication transducer is a block of DF79 and serves only to diagnose the operation conditions of the DeviceNet network and its devices, providing online the operating status of connection, configuration and communication.

The global diagnostic is on the **DN_COMM_TRD** block, from the offset 9, below the **DN_BAUD_RATE** parameter up to slaves' list with I/O connection in operation, **DN_IO_LIST**.

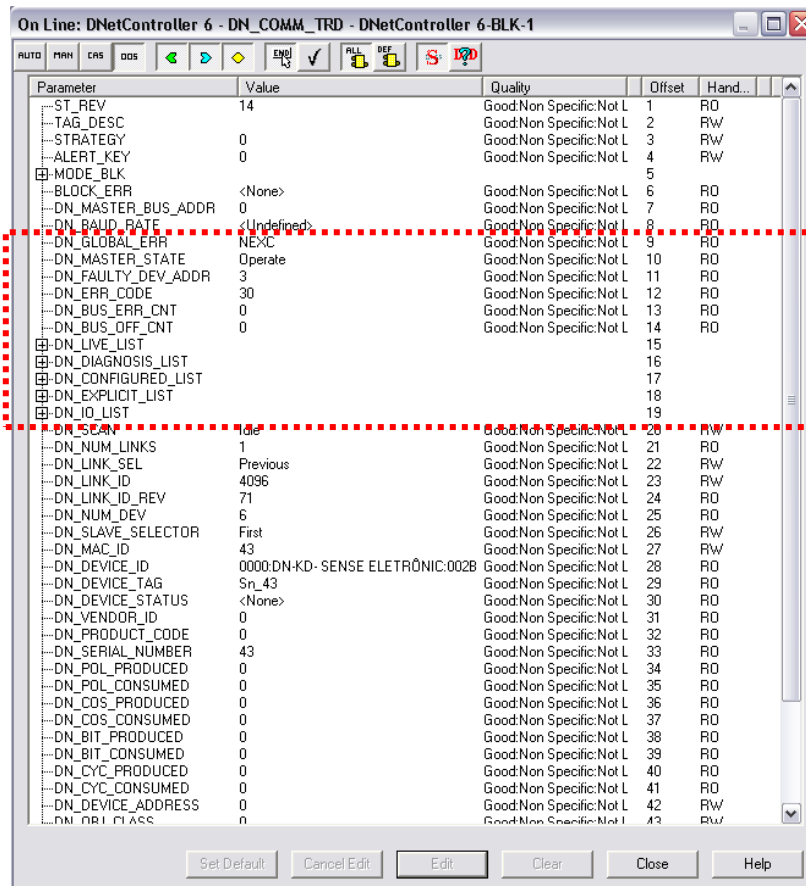


Figure 21. 81 – Global diagnostic parameters in the DN_COMM_TRD block

In the following table is listed the link between the information on the global diagnostic area of the **Network Configurator** and its respective presentation at the communication transducer block **DN_COMM_TRD**.

NETWORK CONFIGURATOR GLOBAL STATE FIELD	TRANSDUCER BLOCK DN_COMM_TRD
Collective status bits	DN_GLOBAL_ERR
Online master state	DN_MASTER_STATE
Error at remote address	DN_FAULTY_DEV_ADDR
Corresponding error event	DN_ERR_CODE
Counter of rejected telegrams transmissions	DN_BUS_ERR_CNT
Counter of detected bus off report	DN_BUS_OFF_CNT
Parameterized Devices	DN_CONFIGURED_LIST
Activated Devices ¹	DN_EXPLICIT_LIST
	DN_IO_LIST
Devices with diagnostic	DN_DIAGNOSIS_LIST

Table 09 – Mapping of global diagnostic information in the DN_COMM_TRD block

¹ In DN_COMM_TRD is possible to check which devices are with *explicit messages* (used to configure the device) activated. The Network Configurator only watches if there are data exchange, serving for this communication by I/O scan or configuration messages.

The maps of bits for status information are listed, consisting of slave address and its respective status. It should be emphasized that the addressing starts in one (1) even to 64, so the master is on address 1 and all the slaves have their address shifted by one.

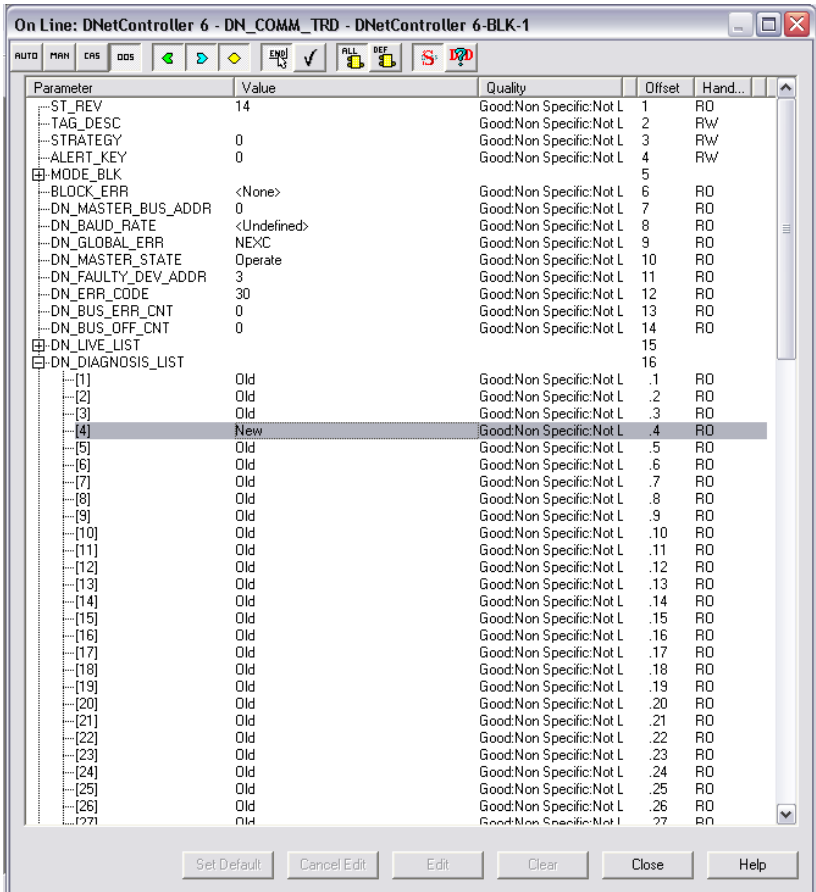


Figure 21. 82 – Example of the expanded status list

In the table below are the names used to indicate the status on each status list in order to adapt to the nature of the information returned.

STATUS LIST	TRUE	FALSE
DN LIVE LIST ²	Online	Offline
DN CONFIGURED_LIST	Configured	Pending
DN EXPLICIT_LIST	Connected	No Connection
DN IO_LIST	Connected	No Connection
DN DIAGNOSIS_LIST	New	Old

Table 10 – Names used in the status list

² The first position in the *Livelist* list shows the name “Master”, indicating that the controller is on this position by default.

Location of Individual Diagnostic in the Communication Transducer

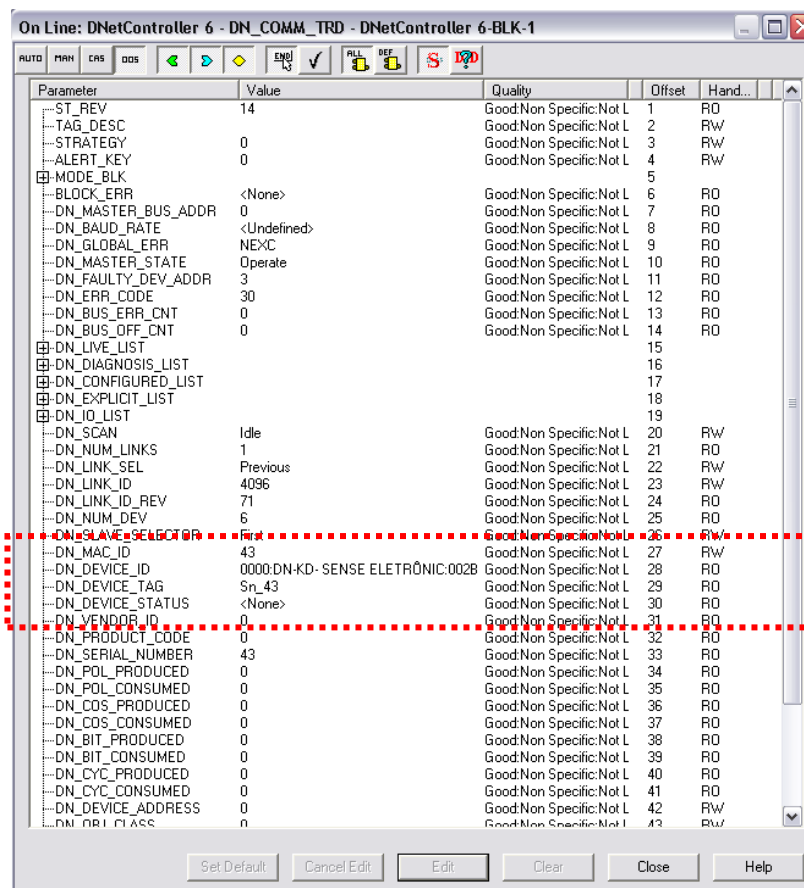


Figure 21. 83 – Detail of the browsing methods on the Transducer DN_COMM_TRD

In order to facilitate the acquisition of diagnostic information, the **DN_COMM_TRD** block has the **DN_SLAVE_SELECTOR** parameter, with the following browsing commands among the slaves:

OPTION	ACTION
<i>First</i>	Goes to the first slave of the <i>Live list</i>
<i>Next</i>	Shift to the next slave of the <i>Live list</i>
<i>Previous</i>	Return to the previous slave of the <i>Live list</i>
<i>Last</i>	Goes to the last slave of the <i>Live list</i>

Table 11 – Browsing commands

Or the user can choose the slave directly, through the **DN_MAC_ID** parameter. The information about Device ID, Device Tag and Status are obtained through the **DN_DEVICE_ID**, **DN_DEVICE_TAG** and **DN_DEVICE_STATUS** parameters, respectively.

The **DN_DEVICE_STATUS** item returns the status byte presented in the previous figure. The possible messages that can be recovered on this field are listed below.

MESSAGE	SIGNIFICATION
<None>	Slave operates normally
NoResp	Slave does not respond
Prm_Fault	Slave does not allow writing at least one attribute
Cfg_Fault	Number of bytes consumed or produced does not match to the value configured
Deact	Slave deactivated in the current configuration

Table 12 – Status byte recover in the transducer DN_COMM_TRD

DeviceNet controller specific blocks

DeviceNet Communication Transducer

This block provides the following characteristics:

- Live list of the slaves devices;
- Slaves devices diagnostic;
- Online configuration of slaves' configuration parameters.

See in the following table the parameters description.

Idx	Parameter	Data Type	Valid Range	Initial Value	Unit	Storage/ Mode	Views	Description
1	ST_REV	Uns6	0 ... 65535	0	None	S/RO	1, 2, 3, 4	
2	TAG_DESC	VisStr(32)		Space	Na	S/---	---	
3	STRATEGY	Uns16	0 ... 65535	0	None	S/---	4	
4	ALERT_KEY	Uns8	0 to 255	0	None	S/---	4	
5	MODE_BLK	DS-69	Auto;OOS	OOS	Na	S/---	1, 3	
6	BLOCK_ERR	Bitstr(2)		0, 0	E	D/RO	1, 3	
7	MASTER_BUS_A DDR	Uns8	0...63	0	None	S/RO	2	DeviceNet master bus address.
8	BAUD_RATE	Uns8	3: 125 kbps 2: 250 kbps 1: 500 kbps	0	None	S/RO	2	DeviceNet bus baud rate.
9	GLOBAL_ERR	Bitstr(2)	See Table 2	0		D/RO	1, 3	See Table 2
10	MASTER_STATE	Uns8	0x00: Offline 0x40: Stop 0x80: Clear 0xC0: Operate	0		D/RO	3	This parameter represents the main state of the master.
11	FAULTY_DEV_AD DR	Uns8	0...63	0		D/RO	1, 3	Faulty device address.
12	ERR_CODE	Uns8		0		D/RO	1, 3	Error code of the faulty device. See user manual.
13	BUS_ERR_CNT	Uns16		0		D/RO	1, 3	Number of detected low transmission qualities.
14	BUS_OFF_CNT	Uns16		0		D/RO	1, 3	Number of canceled transmissions and CAN-chip reinitializations.
15	LIVE_LIST	Uns8[64]	0: Online 1: Master 255: Offline	0		D/RO	1, 3	List of connected devices on the network.
16	DIAGNOSIS_ LIST	Uns8[64]	0: Old 1: New	0		D/RO		List of devices with new diagnostic information available.
17	CONFIGURED_ LIST	Uns8[64]	0: Pending 1: Configured	0		D/RO		List of configured devices.
18	EXPLICIT_LIST	Uns8[64]	0: No Connect 1: Connected	0		D/RO		List of devices with established explicit connection.
19	IO_LIST	Uns8[64]	0: No connect 1: Connected	0		D/RO		List of devices with established I/O connection.
20	DN_SCAN	Uns8	0: Idle 1: Scan 2: Scanning 3: Error	0		D/RW		Performs automatic scan of the DeviceNet network.
21	NUM_LINKS	Uns8		1		S/RO	2	It defines the supported number of buses.
22	LINK_SEL	Uns8	0: First 1: Next 2: Previous 3: Last	0: First		D/RW		Select which bus will have the displayed information.
23	LINK_ID	Uns16		0		S/RW	2	It identifies the bus currently selected.
24	LINK_ID_REV	Uns32		0		S/RO	2	It contains the live list revision of the current bus.

Idx	Parameter	Data Type	Valid Range	Initial Value	Unit	Storage/ Mode	Views	Description
25	NUM_DEV	Uns16		0		D/RO	1, 3	It identifies the number of devices on the current bus.
26	SLAVE_SELECTOR	Uns8	0: First 1: Previous 2: Next 3: Last	0		D/RW		Slave address selector to read scan information.
27	MAC_ID	Uns8	1...63 and 255	255		D/RW	1, 3	Slave address selector to read scan information.
28	DEVICE_ID	VisStr(32)		Spaces		D/RO	2	It identifies the device ID of the selected device in the Slave Selector/MAC_ID.
29	DEVICE_TAG	VisStr(32)		Spaces		S/RO	2	It identifies the device Tag of the selected device in the Slave Selector/MAC_ID.
30	DEVICE_STATUS	BitsStr(2)		0		D/RO	1, 3	See Table 3
31	VENDOR_ID	Uns16	0	0		D/RO	1, 3	Vendor ID code according to ODVA.
32	PRODUCT_CODE	Uns16	0	0		D/RO	1, 3	Manufacturer specific device type code.
33	SERIAL_NUMBER	Uns32	0			D/RO	1, 3	Manufacturer specific device serial number.
34	POL_PRODUCED	Uns8	0	0		D/RO		Produced size in polling connection.
35	POL_CONSUMED	Uns8	0	0		D/RO		Consumed size in polling connection.
36	COS_PRODUCED	Uns8	0	0		D/RO		Produced size in Change of State connection.
37	COS_CONSUMED	Uns8	0	0		D/RO		Consumed size in Change Of State connection.
38	BIT_PRODUCED	Uns8	0	0		D/RO		Produced size in bit strobe connection.
39	BIT_CONSUMED	Uns8	0	0		D/RO		Consumed size in bit strobe connection.
40	CYC_PRODUCED	Uns8	0	0		D/RO		Produced size in cyclic connection.
41	CYC_CONSUMED	Uns8	0	0		D/RO		Consumed size in cyclic connection.
42	DEVICE_ADDRESS	Uns8	0...63	0		D/RW		Slave address to establish an explicit connection.
43	OBJ_CLASS	Uns16	-	0		D/RW		Object class to establish an explicit connection.
44	OBJ_INSTANCE	Uns16	-	0		D/RW		Object instance to establish an explicit connection.
45	INST_ATTRIBUTE	Uns16	-	0		D/RW		Attribute number to establish an explicit connection.
46	ATTR_VALUE	OctSt(32)	-	0		D/RW		Attribute value read/written from/to the device.
47	GET_SET_ATTRIBUTE	Uns8	0: Idle 1: Get 2: Set 3: Error	0		D/RW		Control and status of the Get/Set operation.
48	DNM_CFG_STATUS	Uns8	0: Invalid 1: Updating 2: Using	0		S/RO		Inform the user if the DNM has been configured.
49	IO_MAP_CFG_STATUS	Uns8	0: Invalid 1: Updating 2: Using	0		S/RO		Inform the user if the I/O mapping has been done.

Legend: E – Enumerated parameter; Na – Dimensionless parameter; RO – Read only; D – dynamic; N – non-volatile; S – Static
 Gray Background Line: Default Parameters of Syscon

Table 13 – Parameters description of DeviceNet Communication Transducer block

CREATING AN AS-I CONFIGURATION BY USING DF81

Introduction

This section will show a strategy configuration using the DF81 controller. The next figures show an example of process used to control the conveyor movement for filling up bottles with some fluid. The conveyor moves until the sensor detects the bottle. At this moment the valve (ON/OFF) opens and fills up the bottle. The architecture of the AS-i control loop can be seen in the following figure.

PROJ_DF81

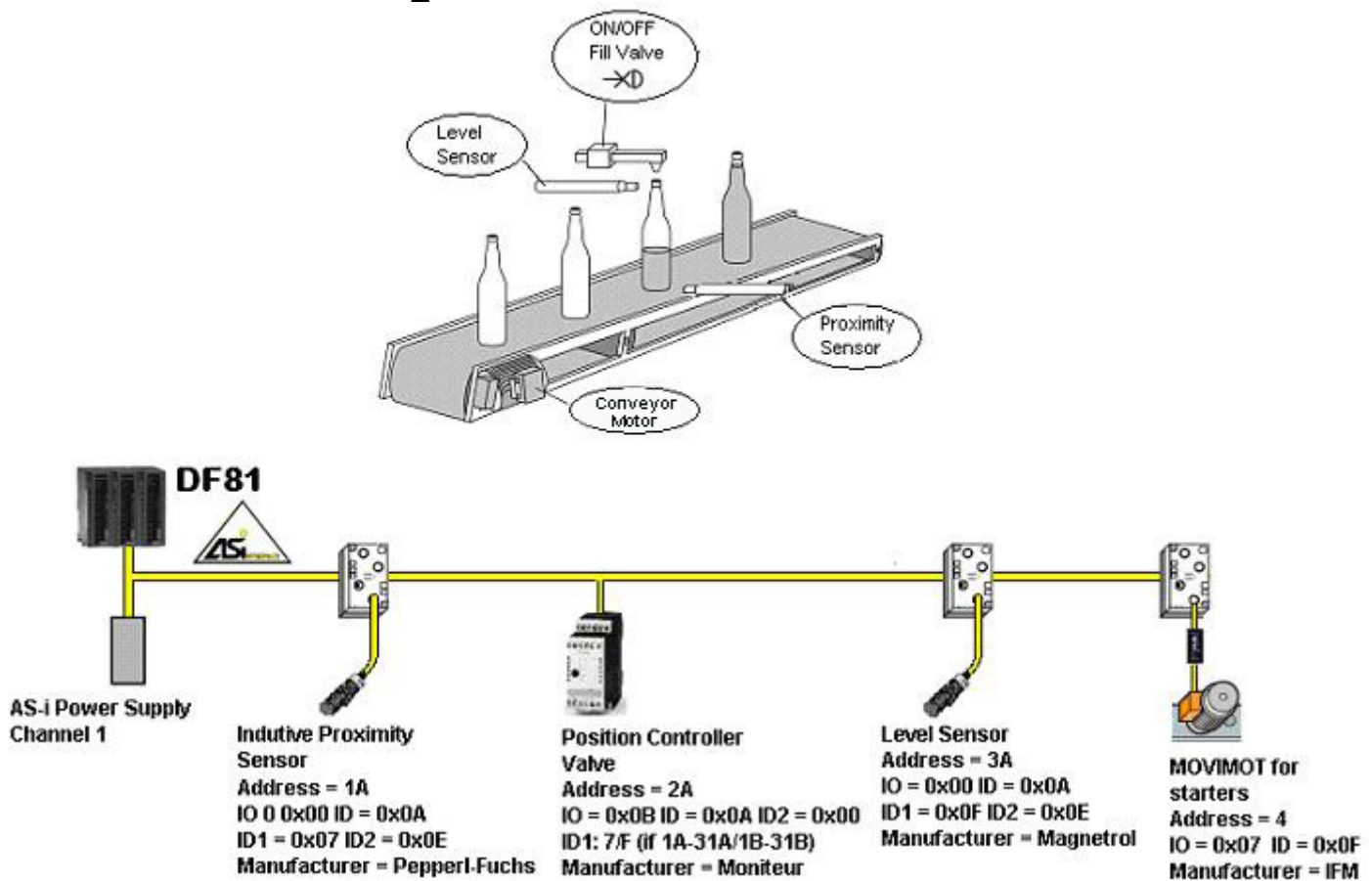


Figure 22. 1 – Example of Process with AS-i Network

For the example above the following devices were used: a power supply for the controller, a power supply for AS-i bus, a DF81 controller, a sensor to detect the bottle, a position controller to turn on and turn off the valve, a level sensor and an IFM Electronic Movimot module for a conveyor motor control.

Often a single application requires more modules of sensors. Inductive and capacitive proximity sensors offer the best solutions, as there are in various formats, have integrated LED display, are easy to assemble and safe in operation.

Besides the known modules input/output with IDC technology (connection by "vampires") and yellow flat cable, the AS-i network has increased the number of slaves integrated in actuators. The compact module Movimot is directly mounted on the drive frequency set on the engine. Two rotational speeds and two directions of rotation can be controlled.

In the next topics will show, step by step, how to configure the DF81 controller for the aimed example. For the DF81 can only make control logic via ladder. The AS-i points are not mapping MDI, MDO blocks etc. Step 8 shows how to setup using Ladder.

Starting the Area

Step 1

It is possible to create, or edit, an area from the **Studio302**. In the **Studio302** interface select **Areas**. A window will appear listing all the areas of database.

To create a new area from the **Studio302**, left-click inside the **Areas** window, then choose **New Area**.

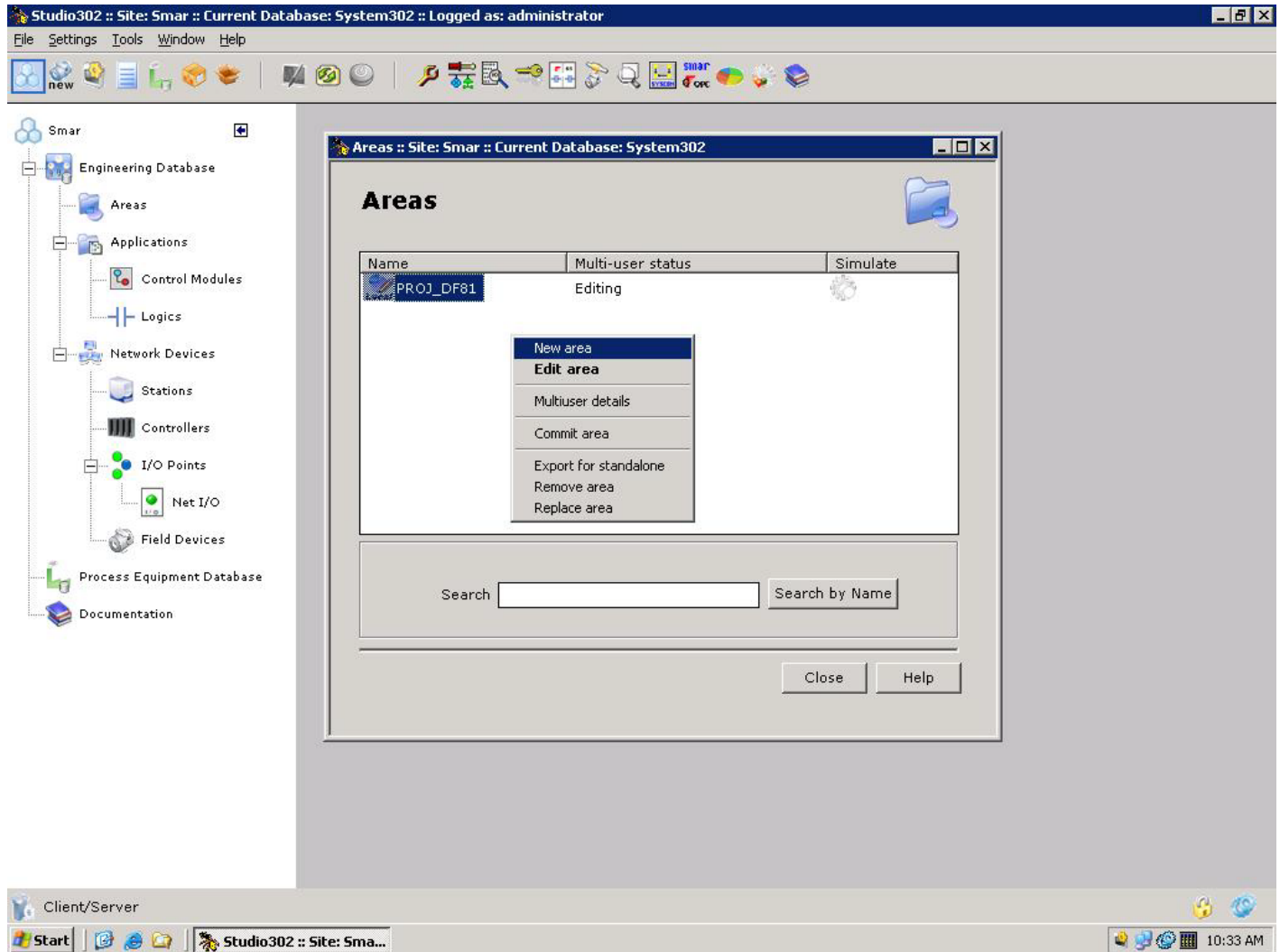



Figure 22.2 - Creating a new area

Another way to create a new area is from **Syscon**. Click the icon  in the **Studio302** toolbar.

To create a new area on **Syscon**, choose **File → New**, or through the toolbar, choose **New** button . The dialog box shows the areas options. Select **HSE Area** as shown in next figure:

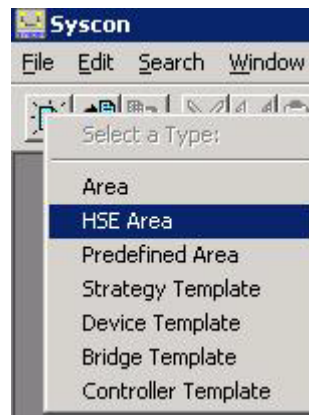


Figure 22.3- Options to create Syscon areas

After choosing the area type, it opens a window to the user give a name to the new area.

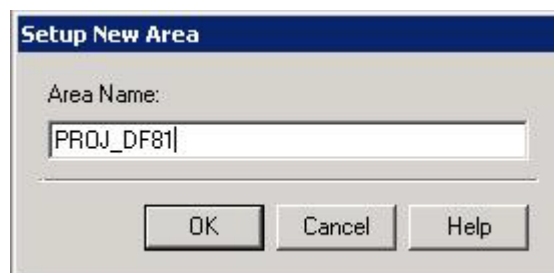


Figure 22.4- New area name

Type the name for the area in the **Area Name** box, and click **Ok**. For this example, it chooses **PROJ_DF81** name.

A new window will appear. It shows:

- **Application** – Logical Plant. To insert the control strategies into this part.
- **Fieldbus Networks** – Physical Plant. To insert devices and function blocks used in the area.

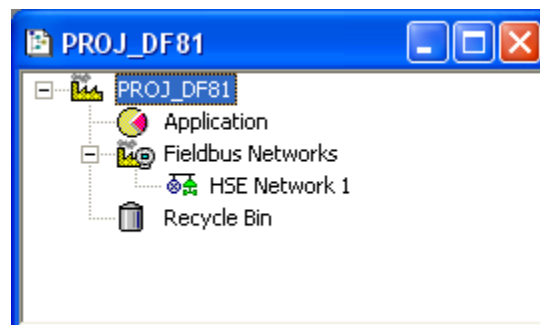


Figure 22.5 – Area Division

Physical Plant Project

Step 2


In the main window, **PROJ_DF81**, right-click the **Fieldbus Networks** icon, , to select the Server. Choose **Communication Settings** option, or through the toolbar, choose **Communication→Settings**. The communication settings dialog box will open.




Figure 22.6 – Choosing the Server

Confirm if the **Smar.HSEOLEServer.0** option has already been selected. Otherwise, the user must select it, and then click **OK**.

Arranging the Fieldbus windows

Step 3

After selecting the Server for the area, click the  sign placed at left of the **Fieldbus Networks**. The HSE network will appear with a tag, for example, *HSE Network 1**. Right-click this item and choose the **Expand** option. The following figure shows the HSE network:

NOTE

* This number changes if another area was created before. When a new HSE area is created, this number increases.



Figure 22.7 – Creating the HSE network

To arrange the screen, click the area window. So, choose **Window** menu on the **Syscon** toolbar, and then choose **Tile** option.

Inserting the Controller

Step 4

Right-clicking the **HSE Network 1** icon, a dialog box to add new devices will appear. By clicking **New**, it is possible to select devices such as Bridges, Controllers and Devices for the area. For the aimed control, it selects the **Controller** option. Confirm this choice as shown in the next figure.

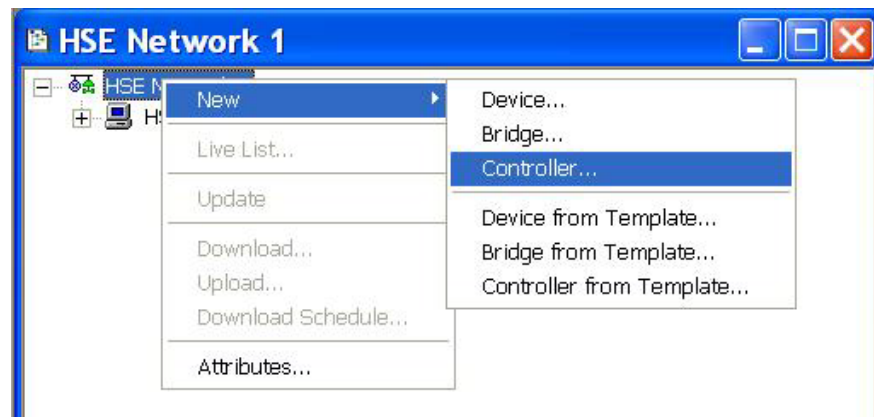


Figure 22.8 – Choosing the controller for the area

After adding the new controller, it opens a window as shown in following figure.

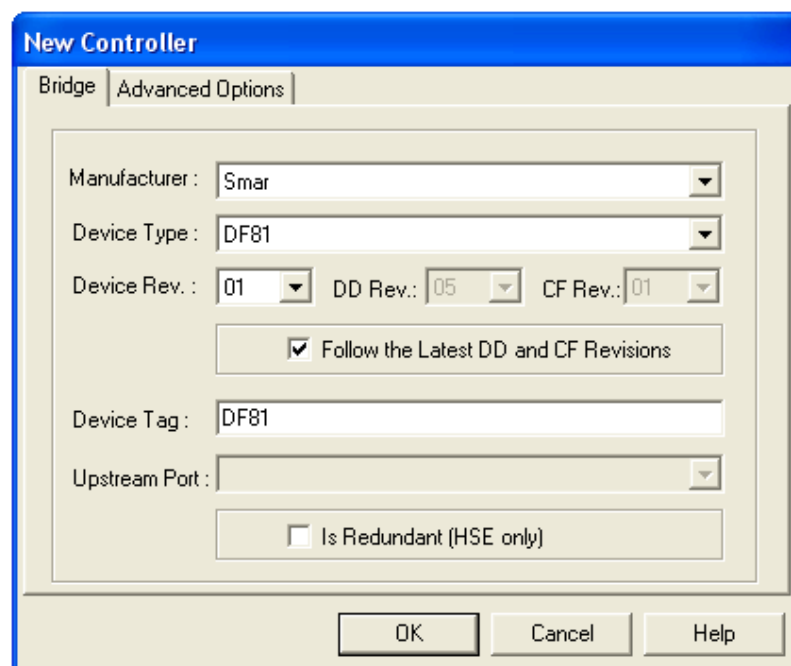


Figure 22.9 – Setting the controller

Select **DF81** in the **Device Type** box. In the **Device Tag** box, type **DF81** or another tag, and click **OK** to conclude this task.

IMPORTANT

Not all characters are valid when naming the elements.

The valid characters are:

A-Z a-z 0-9 { } [] () + -

The invalid characters are:

~ ` ! @ # \$ % ^ & * = | : ; , . < > ? / ' " \

TIP

Is possible to create an initial HSE configuration in an easier way using templates. In this case a configuration exists with some common steps created previously. For example, the steps 1 to 4 can be replaced by a creation of template through main menu **File → New → Predefined Area** choosing **DF81 AS-i Controller HSE 2xAsi HSE** or **DF81 AS-i Controller HSE 2xAsi with FFB-1131**.

Adding AS-i Devices

IMPORTANT

After to open **Network Configuration Tool** (environment for setting a specific network) the user must save the configuration in **Syscon**, to commission the DF81 controller and after to go to the next step.

Step 5

After inserting the controller for the area, it is possible to insert AS-i devices. Before, return to the **PROJ_DF81** window and right-click the controller, **DF81**. Then choose **New Network** option to configure the DF81 channel. DF81 controller has 2 channels that can be configured, but only one at a time. See the AS-i bus creation in the following figure.

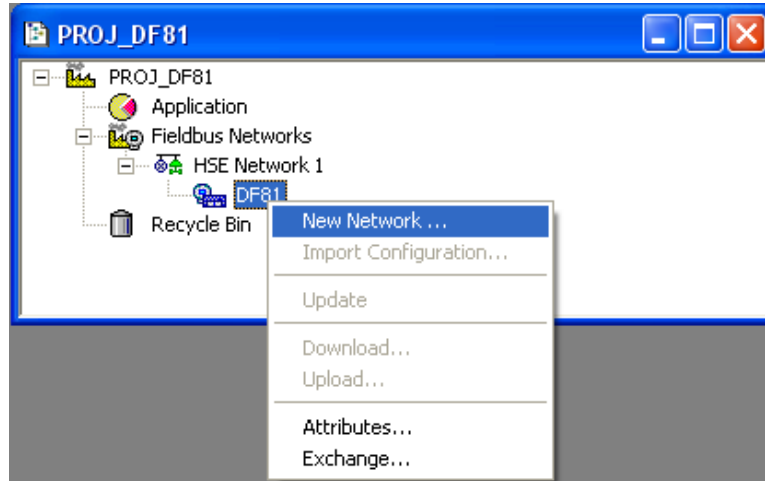


Figure 22.10 – Creating the AS-i bus

When selecting **New Network**, the window to select the channel opens. In this case, you must define a tag for the channel, as well as choose which channels want to create through the **Port** box (1 or 2).

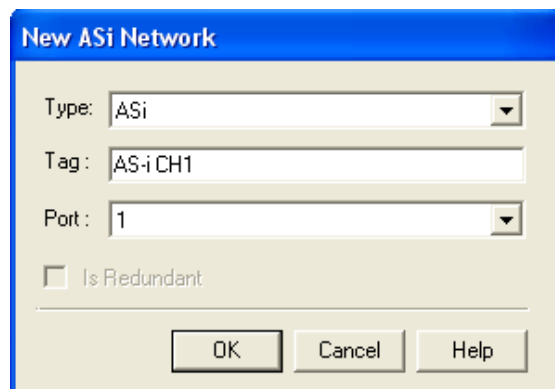


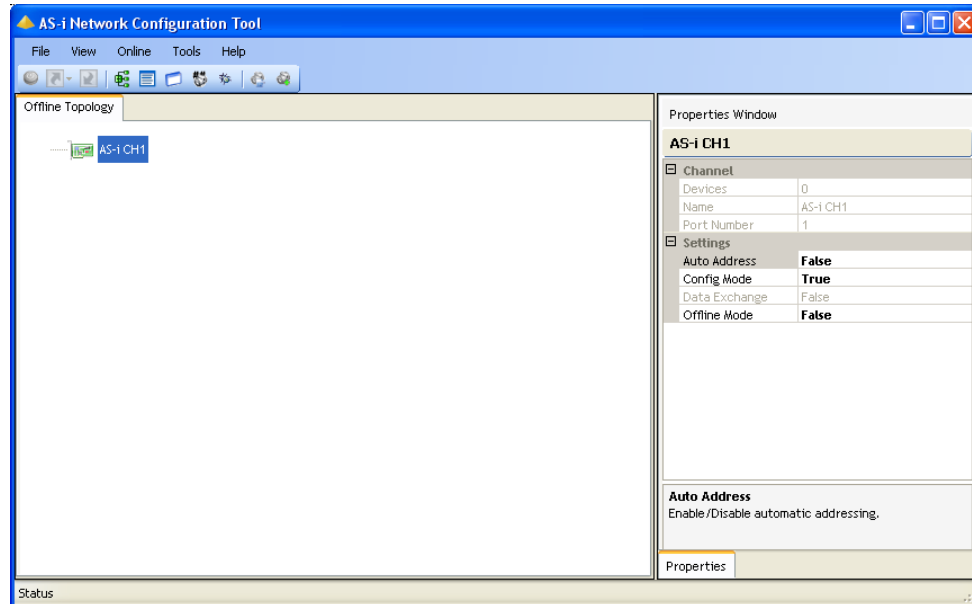
Figure 22.11 – Selecting the channel to be created

By clicking **OK**, the **Network Configuration tool** opens. It is the configuration tool for the AS-i network. Through it, the user has the structure of the AS-i bus (configuration of the DF81 master flags, devices, settings code IO, ID, ID1 e ID2). For further information about this tool, refer to its help.

As the controller DF81 has already been inserted to the configuration previously, and the configured channel was set to Channel 1 and did not have equipment in its configuration, it will appear empty in the network AS-i with only the instantiated channel. This channel is a master in the AS-i network which must be unique. Since the AS-i devices would be inserted are called devices slaves.

IMPORTANT

- 1) The AS-i network supports only one master per bus and its addresses are defined at the time of the channel creation (1 or 2). These addresses are used for **SYSTEM302** architecture and not for AS-i network itself.
- 2) Is allowed to issue only one channel at a time in the **Network Configuration Tool**.
- 3) Is it right to have the bus only a slave with address zero. It must obtain a valid address, so that after it is inserted by a slave factory default comes with address zero.

**Figure 22.12 – Network Configuration Window**

These slaves may be present in one or more lists described below.

List of Detected Devices (LDS): Each slave corresponds to a bit of that list, which is activated when the slave was detected correctly.

List of Activated Devices (LAS): In this list the corresponding bit to the slave is activated when it was activated properly.

List of Projected Devices (LPS): This list is in non-volatile memory and represents the slaves that were expected to be connected to the AS-i network when it is turned on.

List of Peripheral Fault (LPF): In this list the corresponding bit in the slave is activated when a high signal is detected in FID pin of the slave.

In the execution control phase of AS-i network the masters states are reported to the host through some flags described below.

Flag	Meaning
ASI_ECF_NORMAL_OPERATION	It indicates that the master is cyclically transiting between phases of normal operation.
ASI_ECF_CONFIG_OK	This flag is activated when setting nominal and real detected are in according. This is a simple way to obtain information about the configuration.
ASI_ECF_PERIPHERAL_OK	It indicates that List of Periphery Fault is empty.
ASI_ECF_ZERO_ADDR_DEV_DETECTED	It indicates the presence of a slave with address "0" - which is not allowed in normal operation.
ASI_ECF_AUTO_ADDRESS_ASSIGNED	Allows the master to assign a new address for a slave.
ASI_ECF_OFFLINE_READY	Activated when the offline phase is complete.
ASI_ECF_AUTO_ADDRESS_AVAILABLE	It indicates that there are conditions to occur the automatic addressing*.
ASI_ECF_CONFIG_MODE_ACTIVE	It indicates if the master is in the Configuration Mode (True) or Protected (False).
ASI_ECF_ASI_POWER_FAILURE	It indicates bus voltage below the lower limit.
ASI_MHF_OFFLINE	When activated by the User, the master goes to the offline phase.

ASI_MHF_AUTO_ADDRESS_ENABLE	It indicates that automatic addressing is activated.
ASI_MHF_DATA_EXCHANGE	Enables the exchange of data between master and slave.

Thus, it is necessary that the user "designs" the AS-i network that is or will be present in the field. This projection can be given in two ways: the user can manually enter each slave (it is not necessary that it is physically in the field). Another way is to perform the upload of the detected slaves and then to send this information to the AS-i master, i.e. to perform the download procedure. These procedures are described as follows.

Inserting Devices in the Topology

To insert the slave device, click the  (**Insert Devices**) in the toolbox, or through **Tools→Insert Devices** menu. Place the mouse cursor over the channel and click **Insert Devices**.

The window for insertion of the slave device will open, as shown in the following figure.

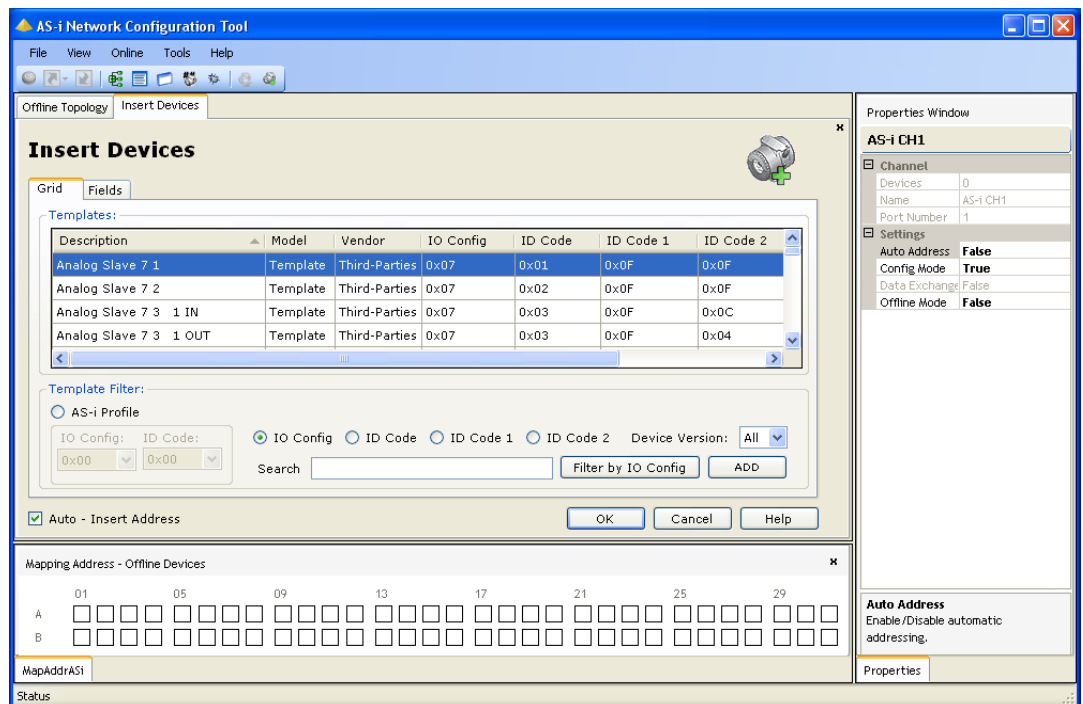


Figure 22.13 – Inserting the Slave Devices

In this window, should be selected slave device that will be inserted. For the aimed example, the first chosen equipment is the proximity sensor. First, the information of AS-i configuration from slave device (IO, ID, ID1, ID2 codes) must be obtained. Follow the next steps to insert the slave device:

- A. Select the device in the option box **Templates**. Use filters to find the desired device;
- B. Click **Add**;
- C. In the **Address** field, assign an address for this device (this address must match the physical configuration of device in the AS-i network);
- D. In the **Device Tag** field, give a tag for this device. If the user does not assign a tag to the equipment, the tag will be the default - Device 1,
- E. Click **OK**.

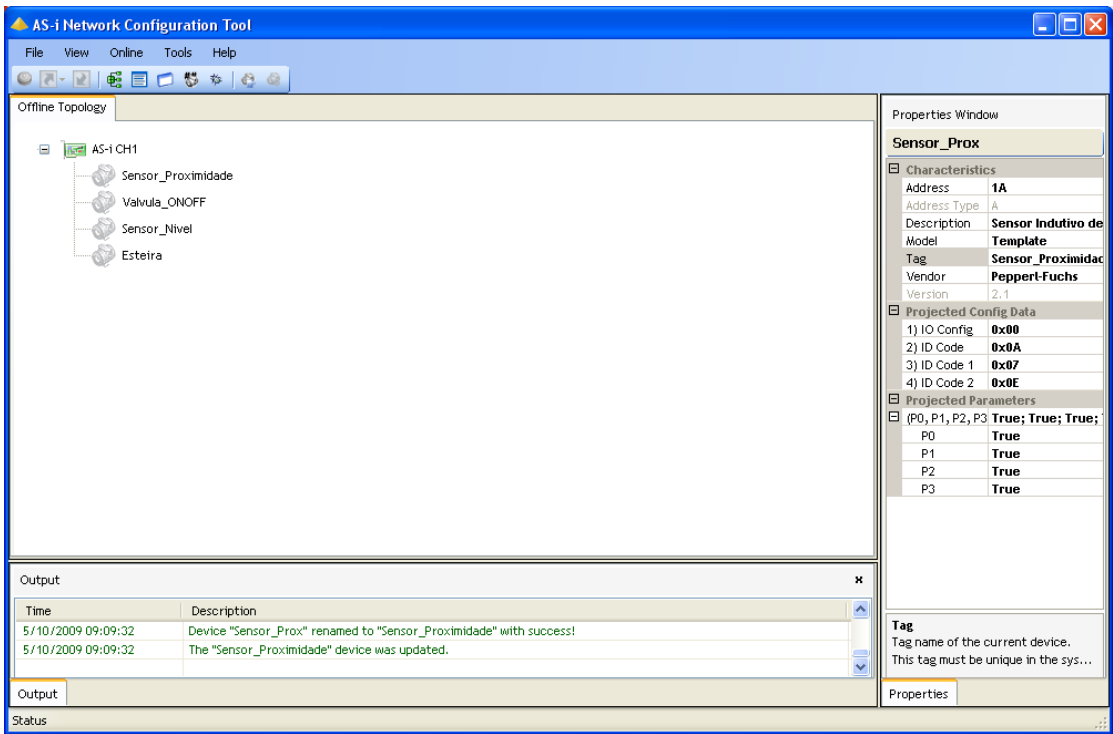


Figure 22.14 – AS-i Network Created

NOTE

The addresses assigned as well as the device tags should be unique on the AS-i network and also in the **Studio302** Database (Workspace). For further details refer to **Studio302** manual.

Enter also the equipment related to the ON/OFF valve, level sensor and conveyor motor. Finally, the AS-i network will be the figure 22.14.

Since the equipment properties are equal to the equipment that is or will be installed, the next step is to perform the Download of the configuration designed. However, it is necessary that the master is in configuration mode, as shown in the next figure. If the flag is off (**False**) just change this condition and perform a **Download** only of the channel (AS-i CH1) via **Online** menu, with the **Download-> AS-i Channel** option.

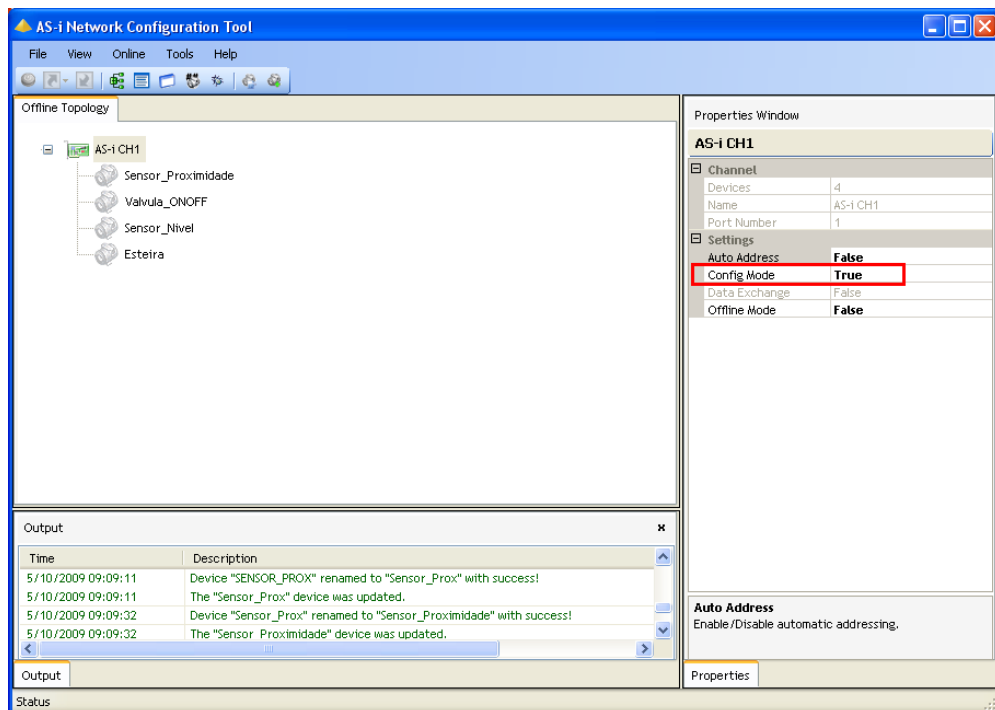



Figure 22.15 – Enable Flag Config Mode

Performing the Upload of Detected Device

The second way of designing an AS-i network is to perform the upload of detected devices, i.e., physically installed in the field, because in this way all parameters already pre-configured by the manufacturer are automatically loaded into Offline Topology. The following are the steps to perform the **Upload** procedure of the detected devices.

- A. Click the  button (*Toggle Start/Stop Communication*) to start the communication;
- B. Select the channel (**AS-i CH1**) right-click it and select **Upload Channel→Detected Devices**. This procedure also can be done through **Online** menu and **Upload→Detected Devices** option.
- C. At this moment will appear a new window (see below) comparing the configuration of Offline topology with the configuration on the field;
- D. Click **Confirm Upload**;

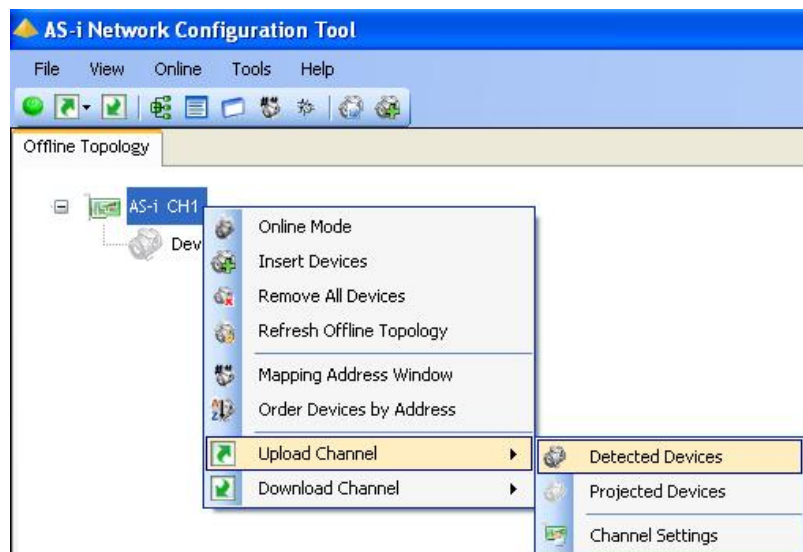


Figure 22.16 – Upload Detected Devices

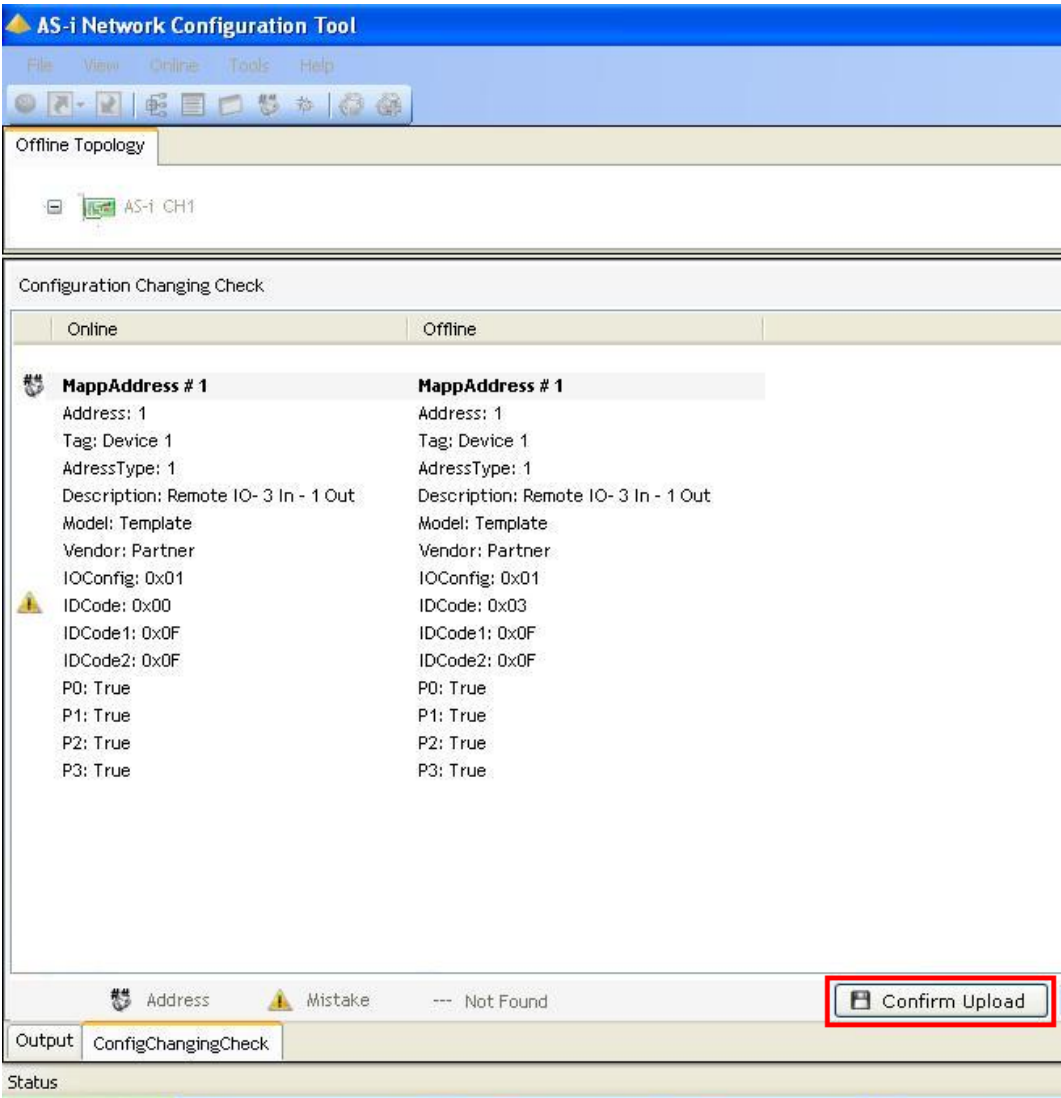




Figure 22.17 –Comparing between Devices Offline Topology and the Detected

Finally it needs to **Download** the configuration to put the devices in the AS-i master memory. To do this, just select the button  (**Propagate DownStream**). There are other **Download** options: selecting the channel or via **Online** menu. For further information about the differences between these options, please consult the tool help.

Inserting New AS-i Devices in Standard List

SYSTEM302 already has a database of more common AS-i devices in the market. If there is any device that requires a specific parameter and is not on the available device list you can use the **Device Profile Editor** to create new equipment **Template**.

To create a new device, click the  button (**Device Profile Editor**) in the toolbar, or through **Tools→Device Profile Editor** menu. The window to create the slave device will open, as shown in the next figure.

Select the **Customize** tab and enter the specific parameterization of slave device. Click **Save** to add this device to the standard list. Click **Close** to close the **Device Profile Editor** and the device will be already available for use in the list of **Insert Devices** window.

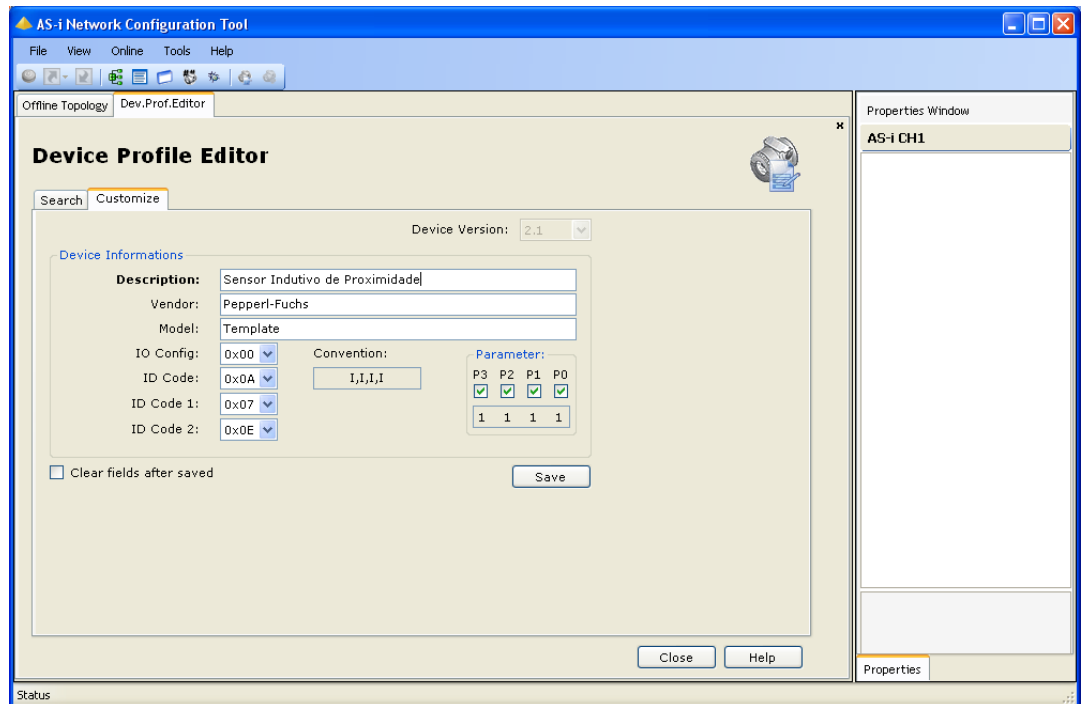


Figure 22.18 – Creation of New Slave Devices

Configuring the AS-i Devices

Step 6

After creating the AS-i network, the devices have to be configured. For example, click the device corresponding to the ONOFF_Valve. The right side of the Topology Offline window will present the device properties.

Basically, it has the information necessary for the **SYSTEM302** architecture (**Tag**, **Description**, **Model** and **Vendor**) and the AS-i network configurations (**Address**, **IO Config**, **ID Code**, **ID Code 1** and **ID Code 2** and **Projected Parameters**). All this information is available in device manual and must be configured correctly.

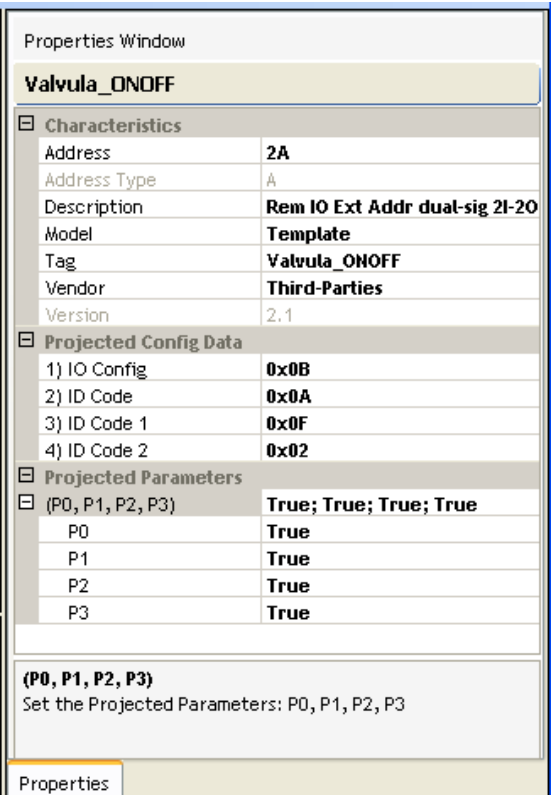


Figure 22.19 – Setting the Slave Device

NOTE

In the AS-i network configuration tool there is not **Save** option to save the configuration. All configurations created or changed is automatically stored in the **Studio302** Database (Workspace).

Step 7

When the previous steps were finish, the AS-i network configuration can be saved, and then, exit from **Network Configuration tool**. The **Mapping Tool** launches automatically. With this tool is possible to map the points of AS-i network in available points which will be used in ladder logic. This tool configures all characteristics of AS-i points (user tags, data types, scales, etc). See the following figure.

NOTE

In the case of AS-i network its points are not mapped in function blocks, because of the particular characteristics of AS-i network that is linked to the discrete points used in the Ladder.

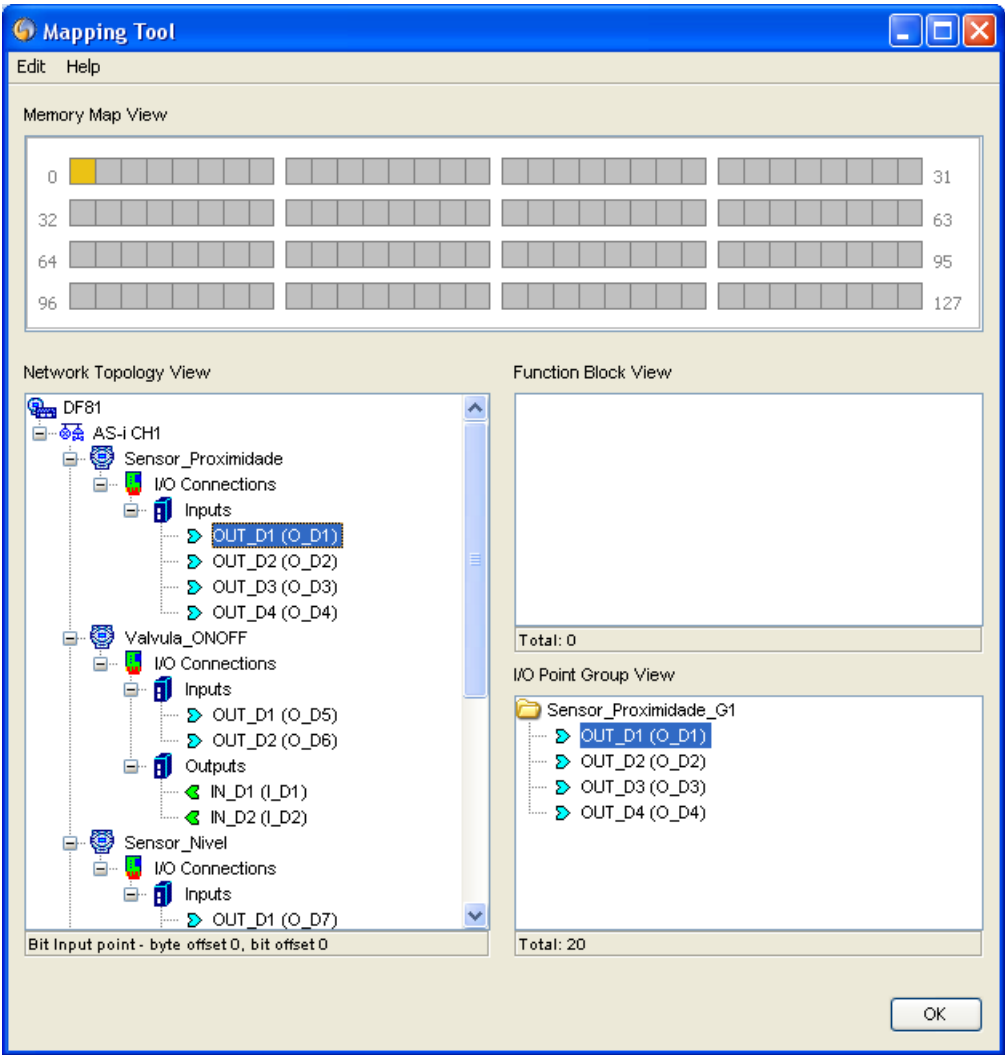


Figure 22. 20 –Mapping Tool Window

In the previous figure the devices which were inserted in the AS-i network with their modules can be seen in the **Network Topology View** window. In the **Function Block Label** window are the points which will be seen at **Syscon**, and the **IOModule Point View** window are the points which will be seen at **LogicView for FFB**. The purpose is to configure each network “point” (or byte) in their respective data types.

NOTE
All points of AS-i network are automatically configured and have only the **Data type** bit type.

The next steps show how to use the points mapped in the **Network Configuration Tool** in the control logic. As mentioned earlier in the AS-i network there is only one way to map the inputs and outputs – ladder logic.

Mapping AS-i I/O points to be used in Ladder

Step 8

The points are automatically mapped in the AS-i network and will be available for use in the ladder. See the next figure that the bits have been automatically selected by the tool. By default all inputs and outputs of the equipment will be mapped.



Figure 22. 21 – Configuring the bits

Click **Cancel** to go to the initial screen or **OK** to exit from **Mapping Tool**.

After the operation is carried out successfully, insert a FFB block in the configuration using **Syscon**. This block is required to edit any ladder. For further details refer to the section Adding Function Blocks or **Syscon** manual. Right-click the FFB block, and then select **Define Parameters**.

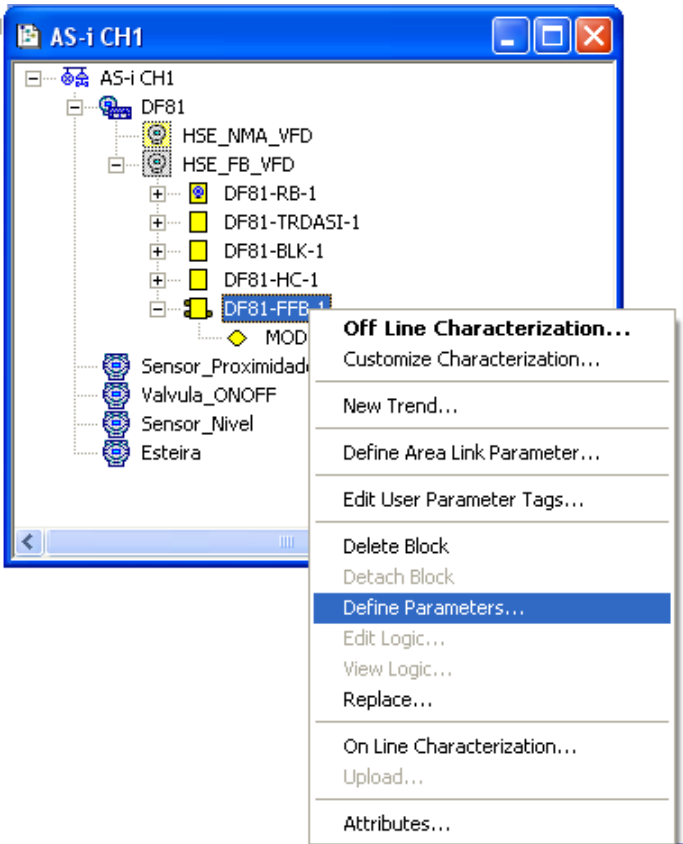


Figure 22. 22 – Defining FFB parameters (1)

The follow window will open:

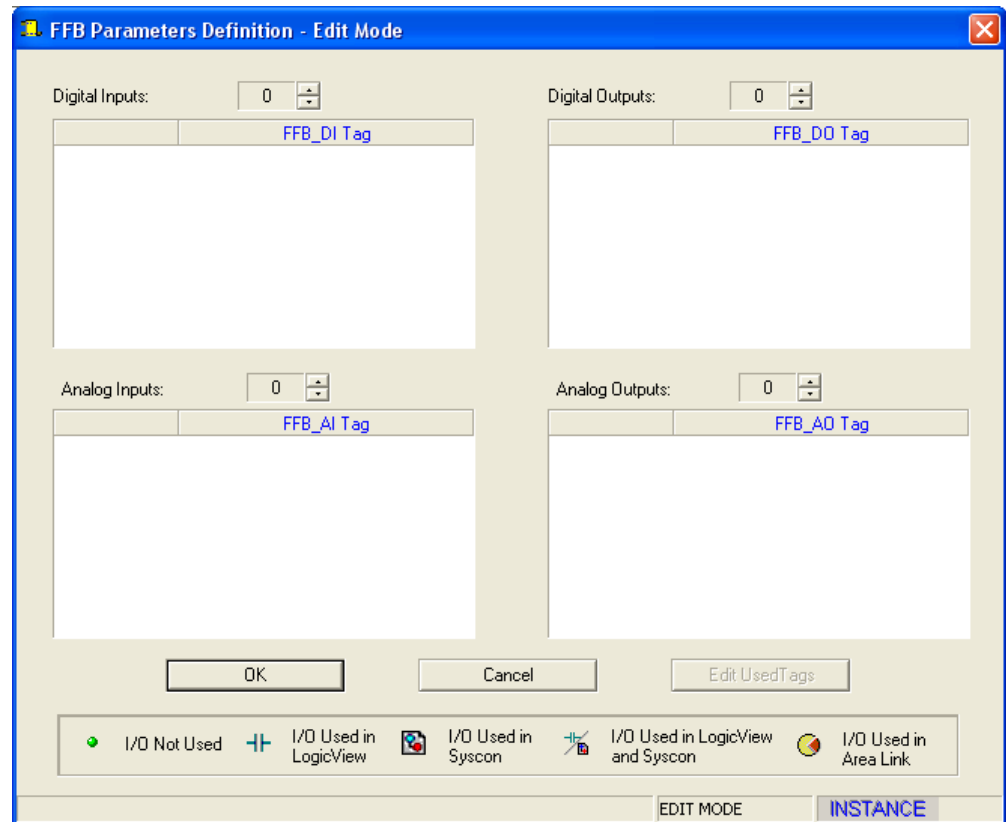


Figure 22. 23 – Defining FFB parameters (2)

If necessary define the inputs and outputs of FFB. Otherwise just click **OK**.



Return to **Syscon**, save the configuration, and do an **Export tags**.

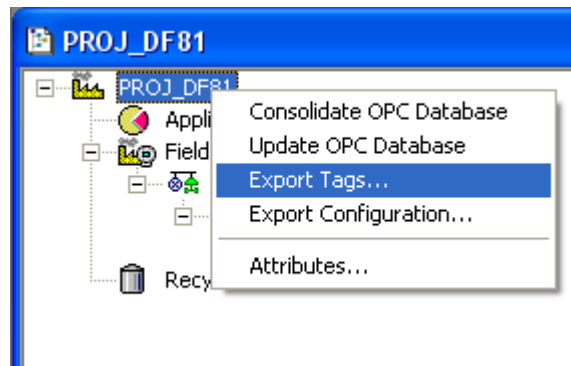


Figure 22. 24 – Export tags

After finish this operation, right-click the FFB block, and then select **Edit Logic**. The **LogicView for FFB** will be launched, and the ladder logic can be edited.

For the first time the logic is edited the **Refresh Data** command has to be executed thus the IO points configured in **Mapping Tool** will be updated in the ladder. Right-click **Network I/O** in the **Hierarchy** window of the **LogicView for FFB**. See the next figure.

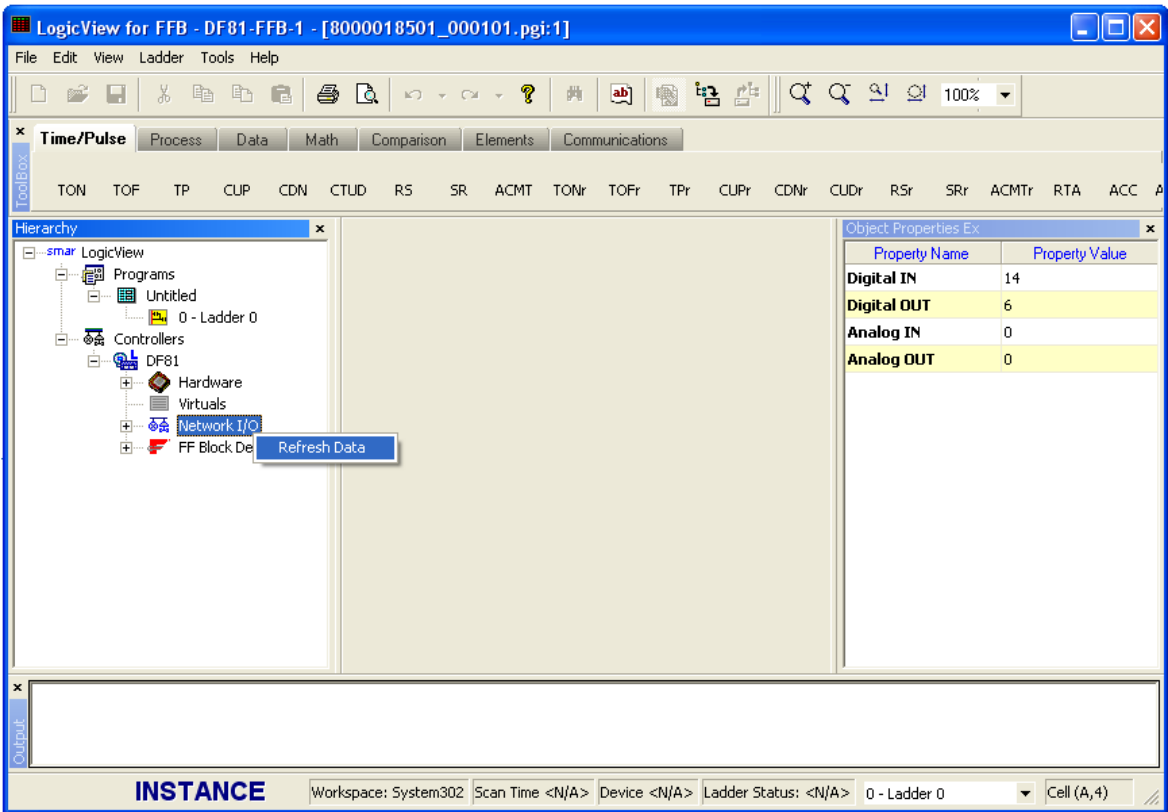


Figure 22. 25 – Refresh data

The NetIO points can be seen in the **Hierarchy** window. See the next figure.

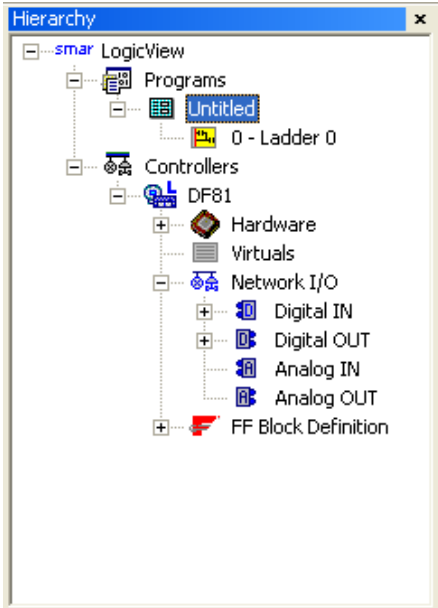



Figure 22. 26 – Network I/O at LogicView for FFB

After that procedure the IO point of the AS-i network are already available to be used in the logic. The user has a complete function library which can be used in the logic. For further details see the **LogicView for FFB** manual.

In the following figure is shown the logic with respect to such proposed. The point of the proximity sensor connected in a TON block (Timer). All the AS-i IO points appear with the symbol  in the ladder logic.

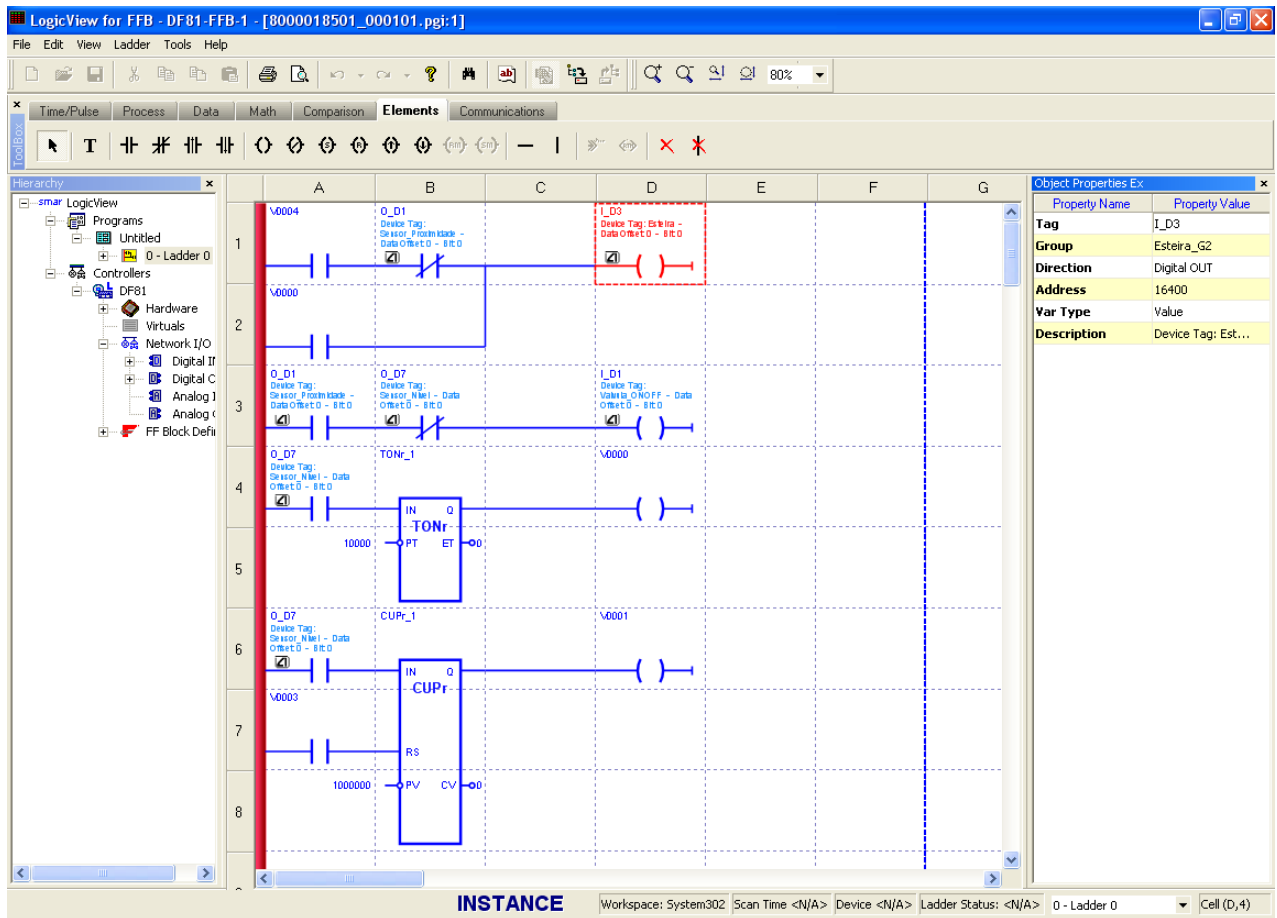



Figure 22. 27 – Editing Logic

After ladder configuration the user can download the configuration to the device. Another option is to configure the function blocks at **Syscon**. In this topic only the ladder was used to configure the ladder. Before downloading the configuration you have to save it in the **LogicView for FFB**. Exit from it, and return to **Syscon**.


To start a communication with the devices, first, is necessary to commission the controller. In this way the tags, IDs and addresses of each device will be assigned correctly. If this procedure was not done, the **Syscon** will detect the uncommissioned device and its download will be canceled. When the devices' commissioning finishes the download process can start. The download process can be

done, for example, returning to **Proj_DF81** window, right-clicking **Fieldbus Networks** icon, , and then, select **Download** option. For further details about commissioning and possible download types refer to **Syscon** manual.

Network Diagnostics

There are many ways to identify communication failures in the AS-i network. Some options are: Network Configurator, Transducer block, **Syscon** Live List or module's LEDs. Some ways will be explained in this topic.

Network diagnostic using the Network Configurator

To do a network diagnostic through **Network Configuration Tool** first is necessary to connect it to the controller. This can be done by  button (**Toggle Start/Stop Communication**). After, right-click the channel, and select **Online Mode** option as shown in the following figure.

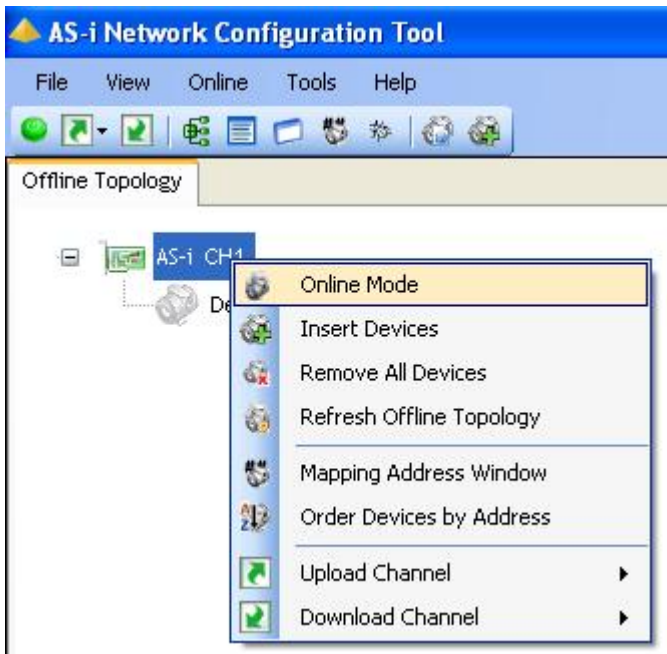


Figure 22. 28 – Selecting the Visualization in the Online Mode

After this procedure is performed the **Online Mode** window will open (see figure bellow). This view will bring the most of the necessary information on the installed AS-i network.

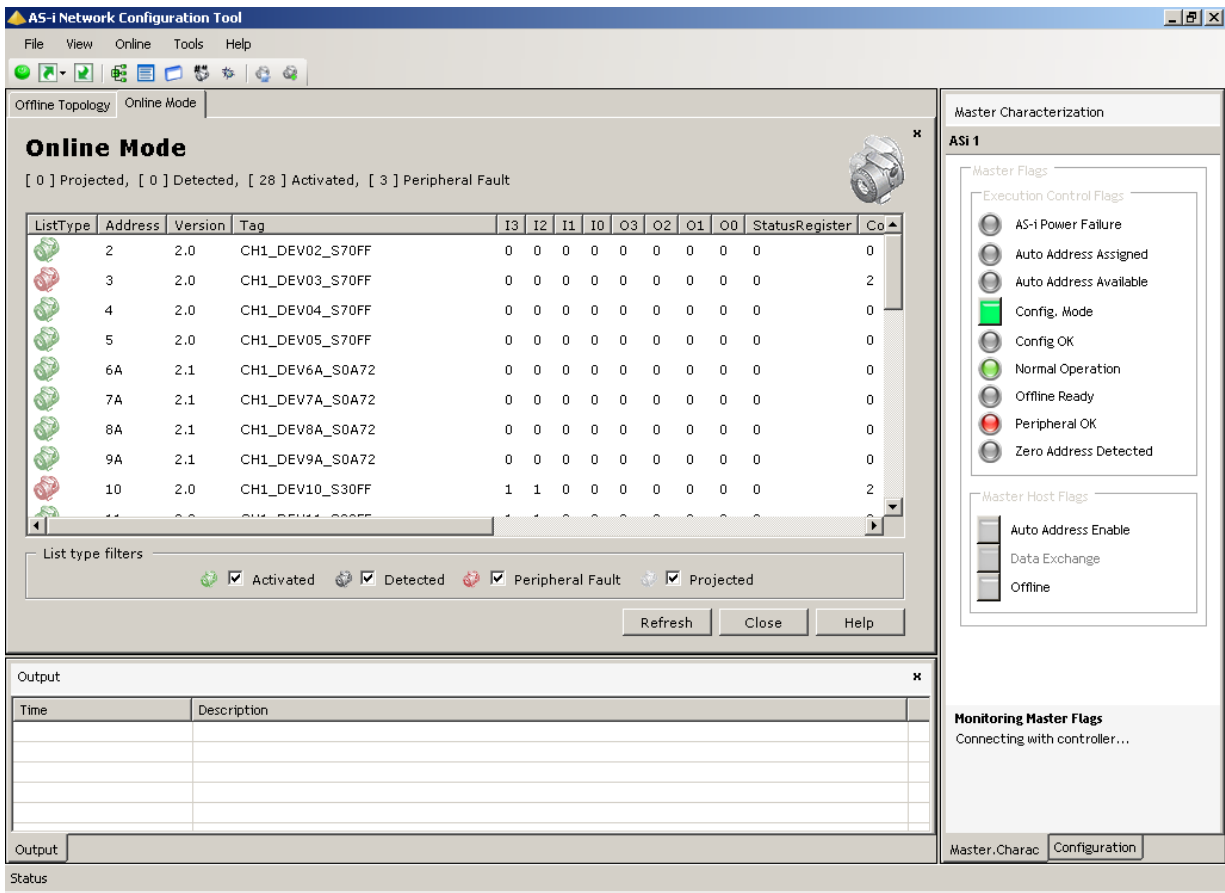


Figure 22. 29 –AS-i Network Diagnostic

The previous figure shows problems with equipment 3 and 10 that went to the LPF and on the other side; there was a red sign in **Peripheral OK** flag of the AS-i Master indicating that there are problems with devices on the network. In addition, there are parameters as **StatusRegister** and **CommErrors**.

StatusRegister indicates the information recorded in nonvolatile memory of the slave (for further information see "Specific blocks of AS-i controller").

CommErrors represents the number of communication failures that occurred between the master and a specific slave. For this case there is an increase of 3, it is the maximum number of attempted communications of the master with the slave in each communication cycle.

Network diagnostic using the Communication Transducer Block

The figure below represents the AS-i Communication Transducer Block. In this block have been mapped the main parameters of the AS-i network that can also be seen in **Network Configuration Tool**. The difference is that the writings in certain parameters are limited to the user. The treatment is done by the configuration tool of the network.

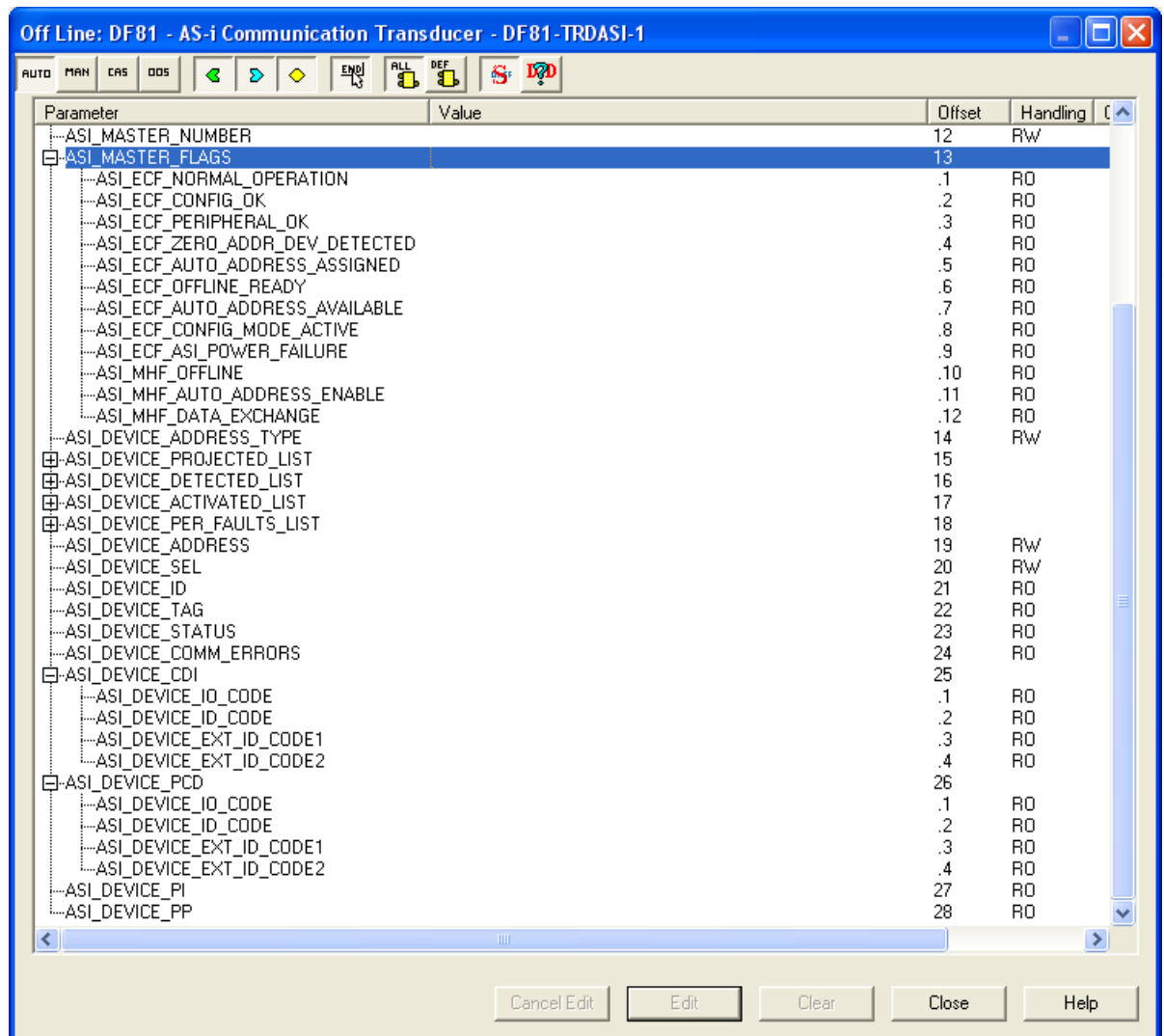


Figure 22. 30 – Transducer Block AS-i Communications

Network diagnostic using the LEDs' Controller

The following table has the indication of the AS-i bus status via the LEDs' controller.

LED	COLOR/STATE		DESCRIPTION	BEHAVIOR
AS-i PWR1 AS-i CFG1 AS-i PWR2 AS-i CFG2	Red	Green		
	On	Off	It indicates supply failure (<i>Power Fail</i>) on the channel.	LED: red on and green off
	Blinking Quickly	Off	Channel powered (<i>Power On</i>) and master is not running (<i>Not ready</i>)	Blinking quickly and pre-set intervals of the red LED and green LED off
	Off	Blinking Quickly	Channel powered (<i>Power On</i>), there is not power failure, but the master is Offline.	Red LED off and green LED blinking quickly on pre-set intervals.
	Off	Blinking Slowly	It indicates when the NORMAL_OPERATION flag is off OR its CONFIG_OK flag is not active OR the master is in configuration mode	Red LED off and green LED blinking slowly
	Off	On	It indicates normal operation, CONFIG_OK flag is active and the master is on protected mode.	Red LED off and green on

Table 22.1 – Description of the state of LEDs Controller

Specific Blocks of AS-i Controller

AS-i Communication Transducer

This block provides the following characteristics:

- Live list of the slaves devices;
- Devices list (LPS, LDS, LAS, LPF);
- Slaves devices diagnostic;
- Online configuration of slaves' configuration parameters.

See in the following table the parameters description.

Idx	Parameter	Data Type	Valid Range	Initial Value	Unit	Storage/ Mode	Views
1	ST_REV	Unsigned16		0	Na	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	Na	S	
4	ALERT_KEY	Unsigned8	1 - 255	0	Na	S	
5	MODE_BLK	DS-69		O/S	Na	S	See Parameter Mode.
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	ASI_NUM_LINKS	Unsigned8	1 - 2	2	Na	S / RO	This parameter defines the number of buses/channels supported for the DF81
8	ASI_LINK_SEL	Unsigned8	0: First 1: Next 2: Previous 3: Last 255: Host	0	E	S/RW	Select which buses will have the information available in items 11 to 17
9	ASI_LINK_ID	Unsigned16		4096	Na	D / RW	This parameter identifies the bus DF81 currently selected in the Chooser Links
10	ASI_LINK_ID_REV	Unsigned8		0	Na	D / RO	Contains the revision of the live list of current DF81 bus
11	ASI_NUM_DEV	Unsigned16	1-63	1	Na	D / RO	Identifies the number of equipment in the AS-i bus current DF81. That's up to 62 slaves, more 1 that identifies the master
12	ASI_MASTER_NUMBER	Unsigned8	1 - 2	1	Na	D / RW	This parameter selects the master/channel to be monitored
13	ASI_MASTER_FLAGS	DS-293		O/S	E	D / RO	Structure that contains execution flags and control of AS-i master

Idx	Parameter	Data Type	Valid Range	Initial Value	Unit	Storage/ Mode	Views
14	ASI_DEVICE_ADDRESS_TYPE	Unsigned8	0: Not applicable. 1: Original version: from 1 to 31. 2: Extended version: from 1A to 31A. 3: Extended version: from 1B to 31B. 4: Version not supported.	Not applicable	E	D/RW	Identifies the type of equipment to address the list of AS-i master
15	ASI_DEVICE_PROJECTED_LIST	Unsigned8(32)	0: Not Projected 1: Projected 2: Used by an old version device	0	Na	D / RO	Device list designed for AS-i network
16	ASI_DEVICE_DETECTED_LIST	Unsigned8(32)	0: Not Detected 1: Detected 2: Used by an old version device	0	Na	D / RO	Device list detected in the AS-i network
17	ASI_DEVICE_ACTIVATED_LIST	Unsigned8(32)	0: Not Activated 1: Activated 2: Used by an old version device	0	Na	D / RO	Device list activated in the AS-i network
18	ASI_DEVICE_PER_FAULTS_LIST	Unsigned8(32)	0: Peripheral_OK 1: Peripheral Fault 2: Used by an old version device	0	Na	D / RO	Device list failed in the AS-i.
19	ASI_DEVICE_ADDRESS	Unsigned8	1-63	63	Na	D / RW	It identifies the address of the device. Address 63 is used to identify the network master and the address 0 is not valid
20	ASI_DEVICE_SEL	Unsigned16	0:First 1:Next 2:Previous 3:Last 255: Host	0	E	S/RW	Select which device in the current DF81 bus will have the information shown in items 19 to 29
21	ASI_DEVICE_ID	VisibleString(32)		Spaces	Na	D / RO	It identifies the ID of the device selected in the Chooser Equipment
22	ASI_DEVICE_TAG	VisibleString(32)		Spaces	Na	D / RO	Identifies the tag of the equipment of the device selected in the Chooser Equipment. This tag comes from the network configuration tool.
23	ASI_DEVICE_STATUS	Unsigned8	0: Good. No status register is set. 1: Flag S0: Address Volatile Fault. 2: Flag S1: Peripheral Error. 3: Flag S2: Reserved. 4: Flag S3: Read Error Non-Volatile Memory. 5: Bad. Could not evaluate status register flags. 6: Not Applicable.	6	E	D/RO	It identifies the communication status of the AS-i master with the equipment selected in the Chooser Equipment
24	ASI_COMM_ERRORS	Unsigned8	0-255	0	Na	D/RO	It identifies the communication errors between AS-i master with the equipment selected in the Chooser Equipment
25	ASI_DEVICE_CDI	DS-297		O/S	Na	S/RO	It identifies the current configuration of the equipment selected in the Chooser Equipment
26	ASI_DEVICE_PCD	DS-297		O/S	Na	S/RO	It identifies the configuration designed the equipment selected in the Chooser Equipment
27	ASI_DEVICE_PI	Unsigned8	0: 0000 1: 0001 ... 15: 1111 16: Not applicable.	0	E	S/RO	It identifies the current parameters of the equipment selected in the Chooser Equipment
28	ASI_DEVICE_PP	Unsigned8	0: 0000 1: 0001 ... 15: 1111 16: Not applicable.	0	E	S/RO	It identifies the parameters of the designed equipment selected in the Chooser Equipment

Legend: E – Parameter List; Na – Dimensionless parameter; RO – Read Only; D – Dynamic; N – nonvolatile; S - Static
Fill line with Gray Background: Default Parameters of Syscon

Table 22.2 – Description of the parameters of AS-i Communication Transducer block

The AS-i transducer block has two special structures: DS-293 and DS-297. Its parameters are presented below.

Description of the DS-293 Structure (AS-i Master Flags)

Flag	Meaning
ASI_ECF_NORMAL_OPERATION	It indicates that the master is cyclically transiting between phases of normal operation
ASI_ECF_CONFIG_OK	This flag is enabled when nominal configuration matches with the real configuration detected. This is a simple way to obtain information about the configuration
ASI_ECF_PERIPHERAL_OK	It indicates that List of Periphery Fault is empty
ASI_ECF_ZERO_ADDR_DEV_DETECTED	It indicates the presence of a slave with address "0" - is not permitted in normal operation
ASI_ECF_AUTO_ADDRESS_ASSIGNED	Allows the master to assign a new address for a slave
ASI_ECF_OFFLINE_READY	Activated when the offline phase is complete
ASI_ECF_AUTO_ADDRESS_AVAILABLE	It indicates that there are conditions to occur the automatic addressing*
ASI_ECF_CONFIG_MODE_ACTIVE	Indicates the master is in the "Configuration" mode (True) or "Protected" (False)
ASI_ECF_ASI_POWER_FAILURE	It indicates bus voltage below the lower limit
ASI_MHF_OFFLINE	When activated by the user, the master takes to the offline phase
ASI_MHF_AUTO_ADDRESS_ENABLE	Indicates that the automatic addressing is enabled
ASI_MHF_DATA_EXCHANGE	Enables the data exchange between master and slave

*As a matter of terminology, confusion may occur between this flag and **Auto_Address_Enable**. Note that the first is user-defined, allowing the self address if the conditions are met, which is indicated by the flag **Auto_Address_Available**.

It is important to note that four of these flags are enabled by user – host – and affect the behavior of the master: last two on the list, **Auto_Address_Available** flag and the **Config Mode** flag. All others cannot be changed by user and are controlled by the master.

Description of the DS-297 Structure (AS-i Slaves Configuration)

Parameter	Meaning
ASI_DEVICE_IO_CODE	Through this request the master receives a response to configuration input / output (IO configuration) of a slave. Along with the ID code of the slave (ID code) configuration of IO uniquely identifies a type of slave
ASI_DEVICE_ID_CODE	The ID code of the slaves in compliance with version 2.1 of the specification are in addition to the unique ID code, read by the master by request "Read ID Code", two other codes - "Extended ID Code 1" and "Extended ID Code 2". Together they serve to identify different slaves. Slaves complying with the new specification, for example, have ID code "A" in hexadecimal, as an ID code "B" indicates a slave "safety at work". All slaves with the same ID code "A" also have the other two ID codes
ASI_DEVICE_EXT_ID_CODE1	It is used to read the code extended 1 slave, when it exists. This code, unlike the original, can be modified by the user.
ASI_DEVICE_EXT_ID_CODE2	This code extends the possibilities for configuring the slaves, and as the unique ID code cannot be modified by the user, is defined in a nonvolatile way by the manufacturer.

CREATING A MODBUS CONFIGURATION USING THE DF89

Introduction

This section will present how to configure a strategy using the DF89 as controller. The DF89 controller can operate in several Modbus scenarios. Therefore different application approaches will be shown throughout the topics of this section.

NOTE

- The characteristics of the DF89 described in this section are only available from the firmware version 4 (V4.xx) with Device Description 4.xx.
- The Modbus configuration characteristics of firmware versions prior to 4 are the same as described in section “Adding Modbus” of this manual.
- The DF89 only has RS-485 port from the serial number SN623.
- Modules with serial number less than SN623 support firmware version 4 and features described in this section only for RS-232 serial port.

To illustrate the configuration, consider as example a process has two transmitters, one slave, one IHM and one supervision station (HMI). The network architecture is shown on the figure below. The scenarios used as examples are:

- 1) The DF89 stands for the RTU Modbus master of the slave PLC (PLC_ID1) and the slave transmitter (TRM_ID2) on the RS-485 serial port.
- 2) The DF89 stands for the TCP Modbus master of the slave transmitter (TRM_ID3) on the Ethernet.
- 3) The DF89 is the TCP Modbus slave for the IHM (IHM Master).
- 4) The supervision station (HMI Master) can read the DF89 data (concentrator) or the data directly from the slaves on the DF89 serial port (Bypass).

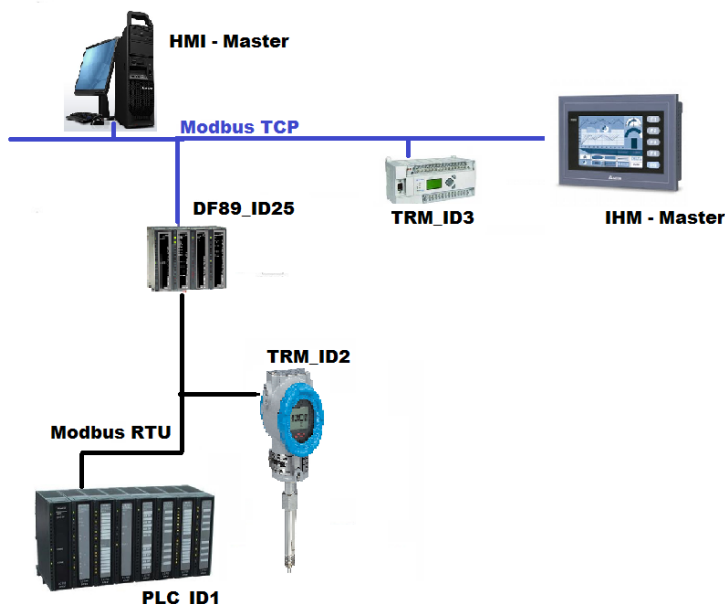


Figure 23. 1 – Example of process using several Modbus scenarios with the DF89

The configuration for each one of these scenarios is explained step by step on the next topics. At the beginning, the steps 1 to 4 are the same for all scenarios. Beyond step 5 there will be differences according to the scenario.

SYSTEM302 enables to execute the control logic by two ways: ladder or function blocks. Steps 5 to 8 are necessary to configure the Modbus network. Step 9 shows how to map the Modbus network points in available points for use in Ladder or Function Block logic. Step 10 shows how to make the configuration using ladder logic and the step 10' using function blocks. It also possible to combine both configurations, i.e., part in ladder and part in function blocks.

IMPORTANT

Before configuring the Modbus network it is necessary to previously seek information on each slave device configuration such as communication rate, data type, point address and functions supported. This information is supplied by the respective Modbus device manufacturer

Creating a Basic Area

This topic will show how to create a basic DF89 area to be used on any of the above scenarios.

Step 1

To create a new area (on the SYSTEM302 language), right-click inside the **Studio302 Areas** window and choose **New Area** as on the next figure.

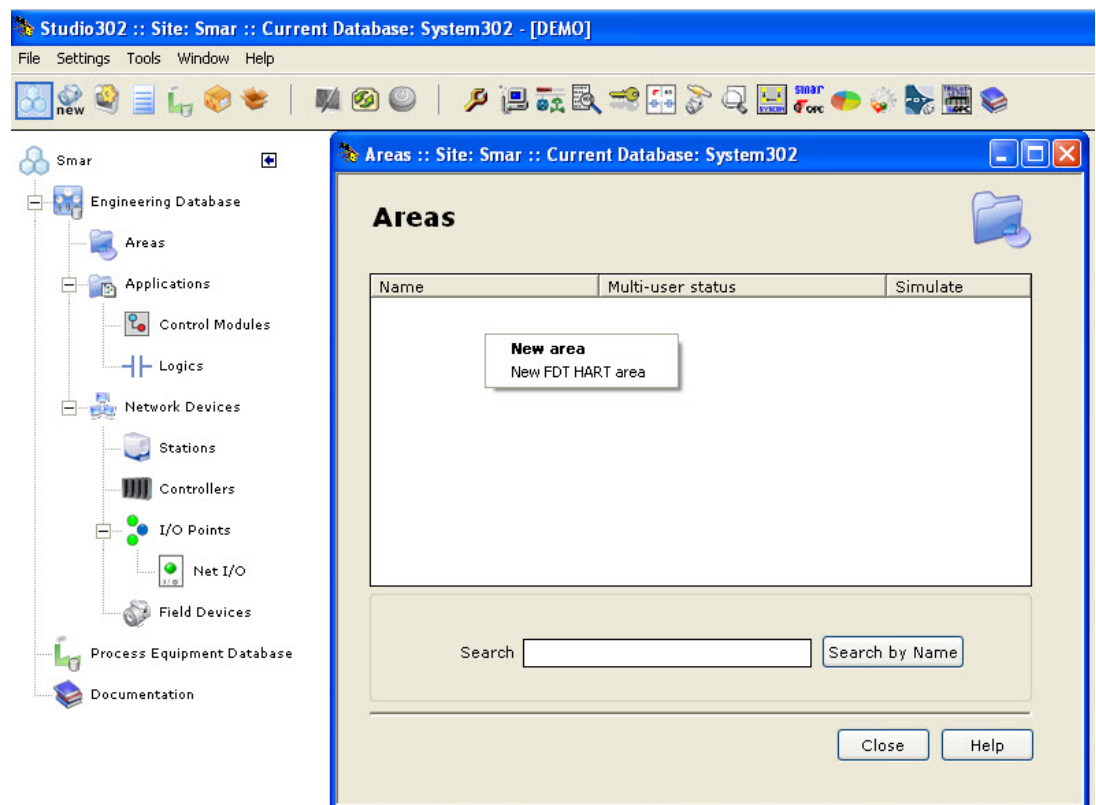


Figure 23. 2 - Options to create a new area on the Studio302

Next, a window will be shown for the user to choose the type of configuration. As the DF89 controller is HSE device, you may choose **HSE Area** or **Predefined** by using a template.

The next window will show the types of existing templates. For every controller there are options with the FFB1131 block or without it. Here, a DF89 template will be used with the FFB1131.

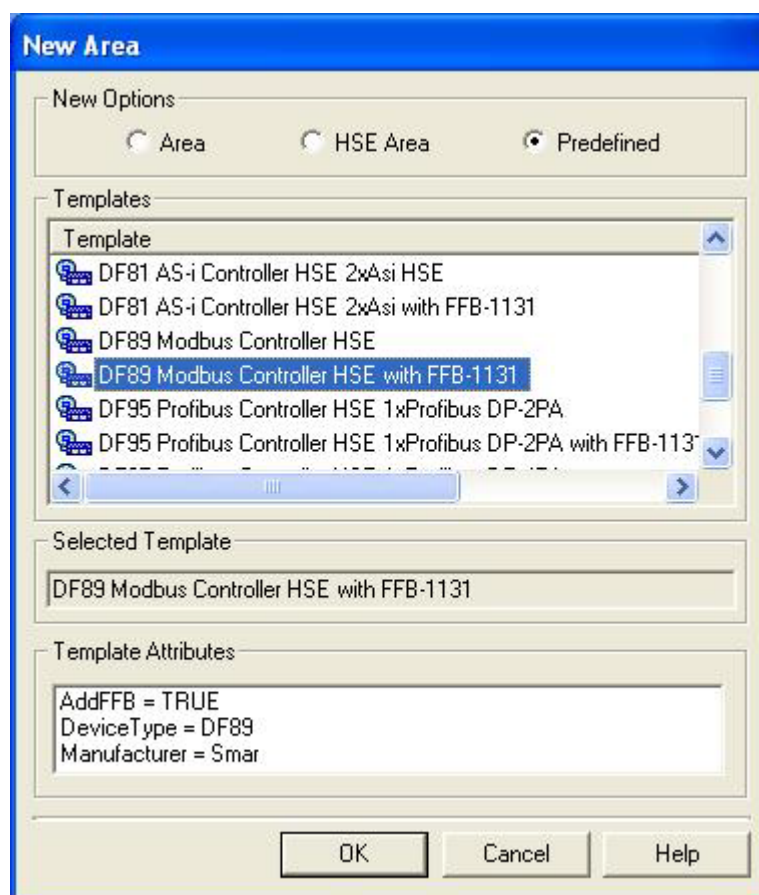


Figure 23. 3 - Pre-defined configuration window

Choose the template and click **OK**.

Step 2

The user must enter the area name as shown on the figure below.

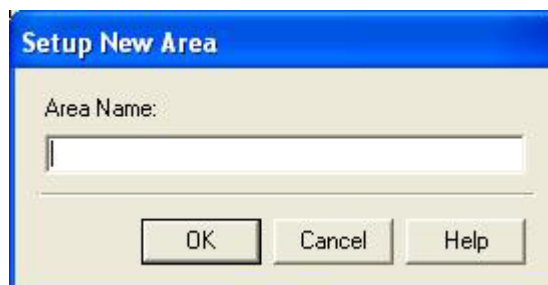


Figure 23. 4 – New area name

Type the name **PROJ_DF89** in the **Area Name** field and click **OK**. Thus, the new area will be created with the DF89 controller. See the following figure.

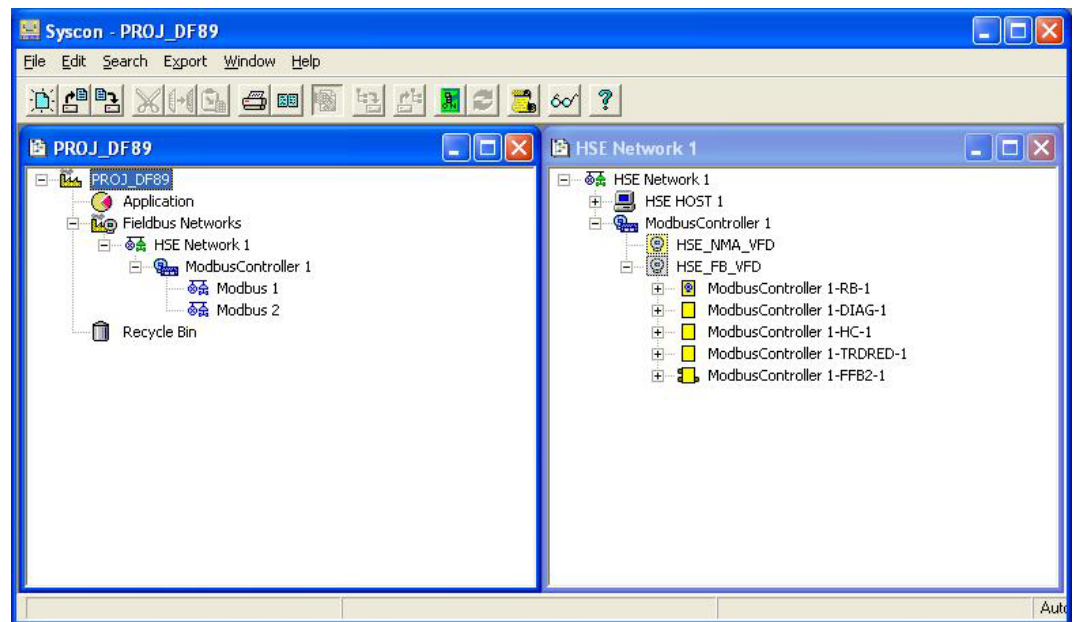


Figure 23.5 – Syscon with the area and HSE channel windows

The figure above shows the **Syscon** window, with the HSE fieldbus network on the left (**HSE Network 1**); below it, there is only one DF89 controller with default tag (**ModbusController 1**). Below the DF89 controller, two channels were created supported by this device: the **Modbus 1**, which corresponds to the serial channel, and the **Modbus 2**, which corresponds to the TCP channel.

The second window on the right represents the expansion (or view) of the HSE channel, which has only the **ModbusController 1**. The other HSE controllers can be added by simply clicking the channel and ask for adding new devices.

Step 3

These default tags can be altered simply by right-clicking the corresponding tag on the main window and choose **Attributes**. The figure below shows the DF89 tag altered to **Area1_DF89**.




Figure 23.6 – Controller attribute window

To alter the channel tag, on the main window, click the channel (for example, Modbus 1) and command window expansion. A new window will open. At the window edge, right-click the channel tag and choose **Attributes**. The figure below shows the configurations for the controller and the channels with the new tags. The tags were changed as follows:

HSE – from **HSE Network 1** to **Area 1 Canal HSE**

DF89 Modbus Serial – from **Modbus 1** to **DF89 Modbus RTU**

DF89 Modbus TCP – from **Modbus 2** to **DF89 Modbus TCP**

To ease the visualization, the **Details** button  on the toolbar shows the type of object between brackets {}: channel, controller, and block. The figure below shows the new tags with the respective types.

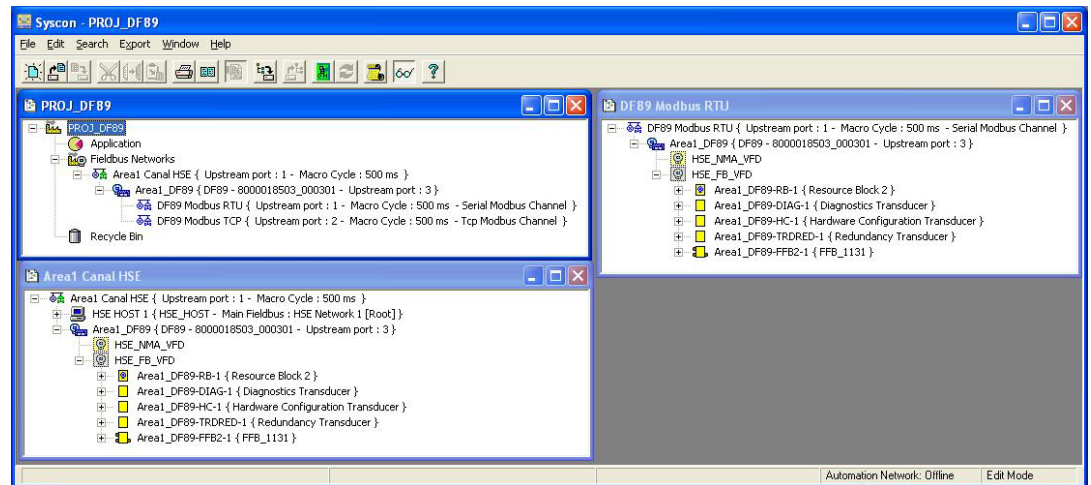


Figure 23. 7 – Syscon with the area windows (above left), for the HSE channel (below left) and for the serial channel (above right).

IMPORTANT

Not all characters are valid when naming the elements.

The valid characters are:

A-Z a-z 0-9 # { } [] () + -

The invalid characters are:

~ ` ! @ # \$ % ^ & * = | : ; , . < > ? / ' " \

ATTENTION

The easier method to create a configuration was used here, based on template. In this case, the configuration is created with all of the necessary channels. If the user does not want a specific channel, it can be excluded.

There is another way of creating the configuration, when the controller and the channel are created manually. In this case, in the step 1 in the figure 23.3, the type of area to be chosen must be the **HSE Area** option, and only an empty HSE configuration will be created. Step 4 described below shows how to create a configurator starting from HSE window. For more details refer to the **Syscon** manual.

The option to create from a template will only be available for new areas.

Step 4

This step will only serve to add a new controller or when the configuration is created only with the HSE Host. In the example of this chapter is not even necessary step.

To add the DF89 controller, at the HSE channel window, right-click the **Area1 Canal HSE**, choose **New Controller** and then **D89**, as shown on the next figure.

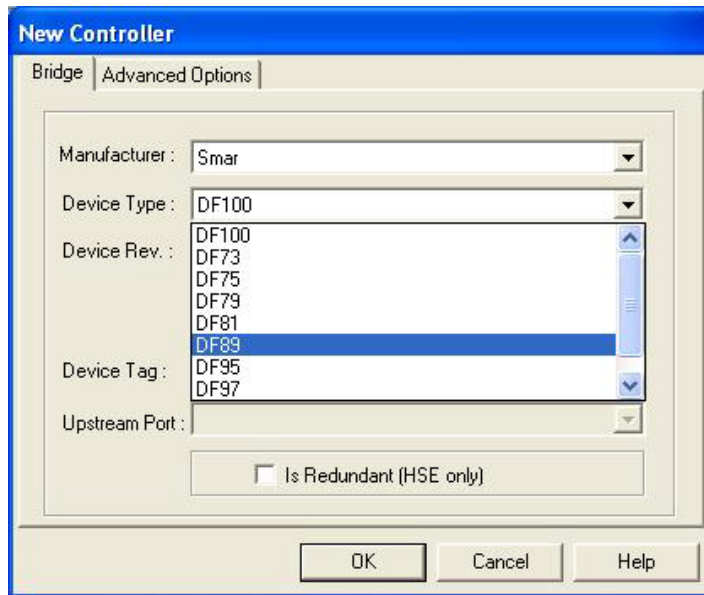


Figure 23. 8 – Creating the DF89 Controller manually

The new controller will be added. Then, still on the area window right-click and choose **New Network**. The window shown on the figure below will appear. Fill out the channel type and tag. Click **OK** and **SmarNetCont** will open automatically.

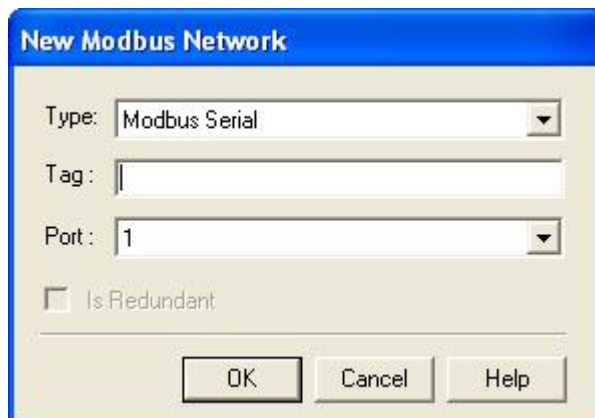


Figure 23. 9 – Creating the Modbus channel

DF89 configuration as Modbus Serial Master

This section will detail every step to make the following configuration:

- Add devices on the DF89 serial channel
- Add Modbus points on the devices
- Map the Modbus points for the **SYSTEM302** logic

For the example of the process on figure 23.1 are used one PLC slave (PLC_ID1) and one transmitter (TRM-ID2) connected to the RS-485 of the DF89 using Modbus RTU.

IMPORTANT

On the Modbus serial scenario the DF89 has two physical means, RS-232 and RS-485, although only the DF89 has a serial port. In other words, it will work only with the configuration (baud rate, stop, parity and timeout) for both means. The DF89 supports up to 32 devices on the serial channel.

Step 5

Modbus network configuration

SmarNetConf or **Network Configurator** is the Modbus network configuration tool. On it, the user will define the network parameters, all slaves that compose them and all parameters necessary for Modbus communication. The **SmarNetConf** has only the vision of the respective channel (Serial or TCP), meaning that on the serial channel vision only the device corresponding to the chosen channel will be shown.

Whenever the user needs to configure the DF89 channel as **Modbus Master** he should use **SmarNetConf**. This will be possible on the following cases:

- The DF89 is master of Modbus RTU slaves on the serial channel (RS-232/RS-485);
- The DF89 is bypass of devices on the serial port;
- The DF89 is master of Modbus RTU slaves on the TCP channel (ETH1);

More details will be provided on the respective scenario section. For other information on the **SmarNetConf** consult the tool help.

To launch the **SmarNetConf** first expand the channel window, which in this example is **DF89 Modbus RTU** serial channel. Select and right-click it. Choose the **Modify Configuration** option. See the next figure.

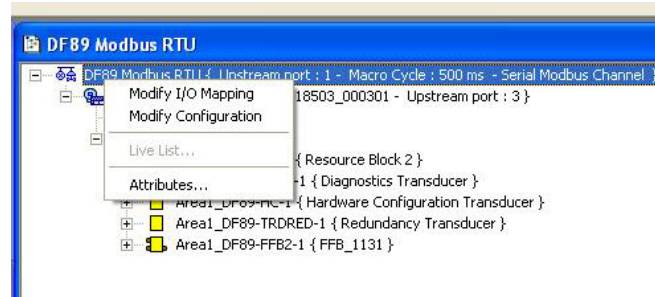


Figure 23. 10 – Launching SmarNetConf to configure a Modbus Serial device

The figure below shows the main **SmarNetConf** serial channel window.

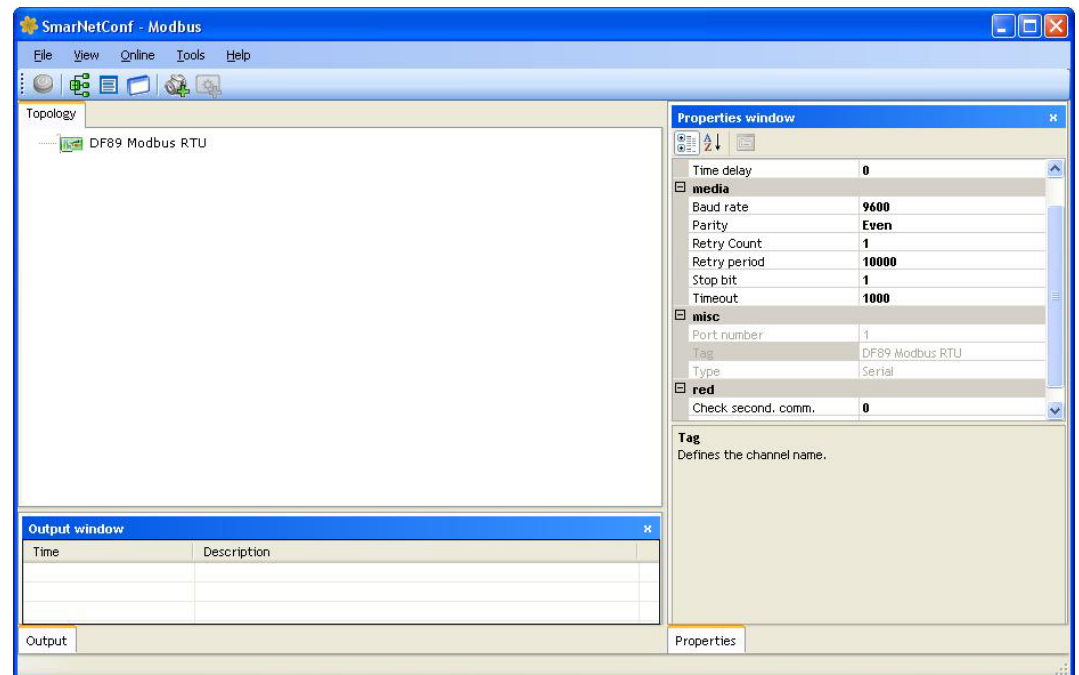


Figure 23. 11 – Main window of the SmarNetConf Modbus configurator

The configurator has three main windows as shown on the previous figure: the **Topology** window, at left, which shows a tree structure for the network device at a given channel, the **Properties** window, at right, which shows all the properties of the item selected on the topology window and, finally, the **Output** window (for events and error logs) at the bottom part of the window.

The properties channel window must have the channel parameters filled out: Baud rate, stop bit, parity, timeout, etc. Each one of these parameters is detailed on the **SmarNetConf** parameterization section.

NOTE

All devices must have the same media parameterization (*Baud rate, Stop Bit and parity*). The DF89 always considers the data bits parameter equals to 8.

Step 6

Configuring Modbus devices

After creating the Modbus network, the device should be configured. To insert a slave device, click on the **Insert Device** button located on the toolbar, or through the **Insert→Device** menu. The next figure shows the window to insert the slave devices, where the user can add one of several devices of the same type.

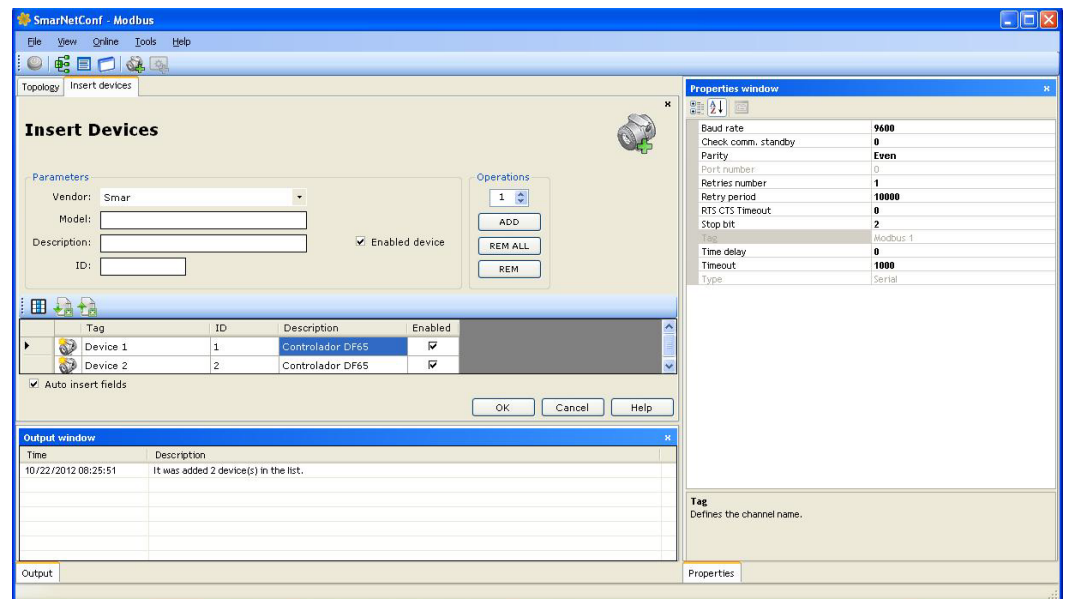


Figure 23. 12 – Adding device on the Modbus network

In this window, select the slave device to be inserted. For the present example, two serial channels will be included. First, get the information for configuring the slave device (see its manual). Follow the steps below to insert the slave device:

- A. Fill out the device **Model**, **Description** and **ID** fields;
- B. Select the quantity of device of a given type;
- C. Click **Add**. A table will be shown with amount of selected device;
- D. On the **ID** field, attribute an address for the device;
- E. On the **Tag** field, the device tag is attributed. In case the user does not attribute a tag to the device, a default tag will be attributed;
- F. Click **OK**. The added devices will be included on the main window.

NOTE

The attributed device addresses and tags must be the only ones in the Modbus network and also in the **Sudio302** (Workspace) database. For more details refer to the **Studio302** manual.

Both devices on the proposed example (PLC_ID1 e TRM_ID2), on the serial channel, will be shown on **SmarNetconf**. See the next figure.

Step 7

How to edit the devices properties

After adding the device, its properties can be changed. The main ones are tag, ID and if the device is or is not enabled. Other properties can be changed in this window. An explanation for each property is shown on the **SmarNetConf** parameters section.

For example, to edit the PLC_ID1 properties, right-click the corresponding Modbus device. At the open pop up, choose the **Edit Device Properties** option to configure the slave device parameters.

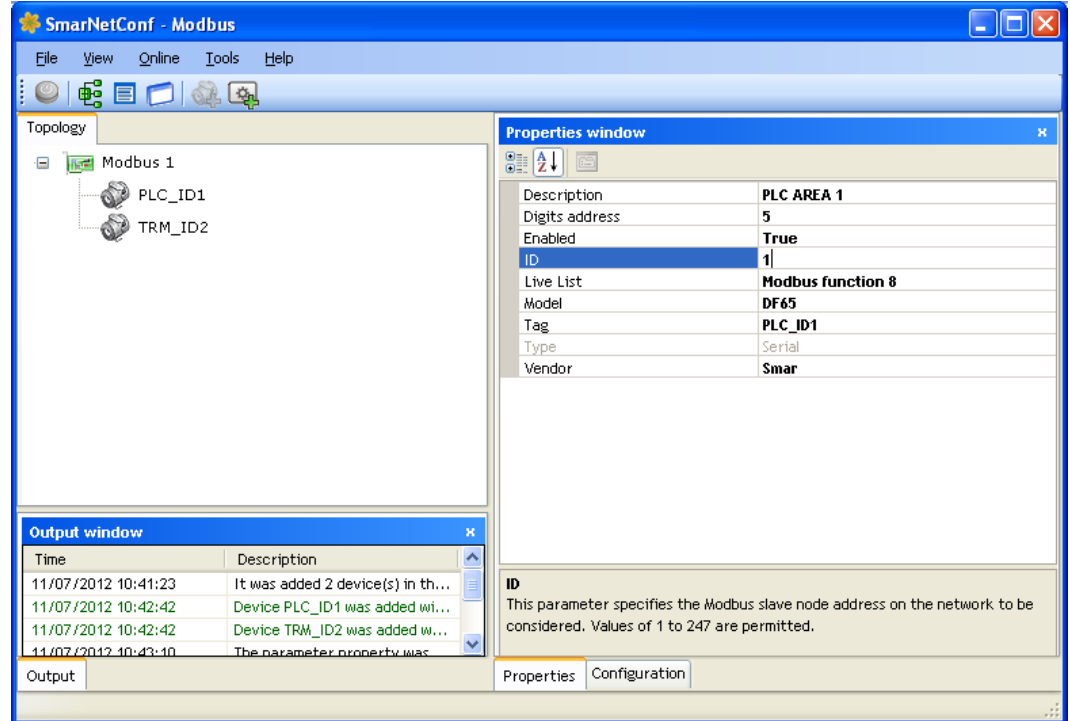


Figure 23. 13 – Editing the device properties

Step 8

Adding commands to the device

The slave variables to be supervised or written are mapped by commands, which can be individual or multiple ones. For example, if the analog points should be supervised on addresses 40001 and 40010. This supervision can be performed through the two individual commands (cmd1 for 40001 reading and cmd2 for 40010 reading) or a multiple command (cmd1 –reading beyond 40001 reading with 10 registers). This last form is most common and most optimized.

The command configuration window is accessed right-clicking the corresponding Modbus device, for example, the PLC_ID1. On the open pop up, choose the **Insert Commands** option to configure the device Modbus commands. The following figure shows the example of a command addition window.

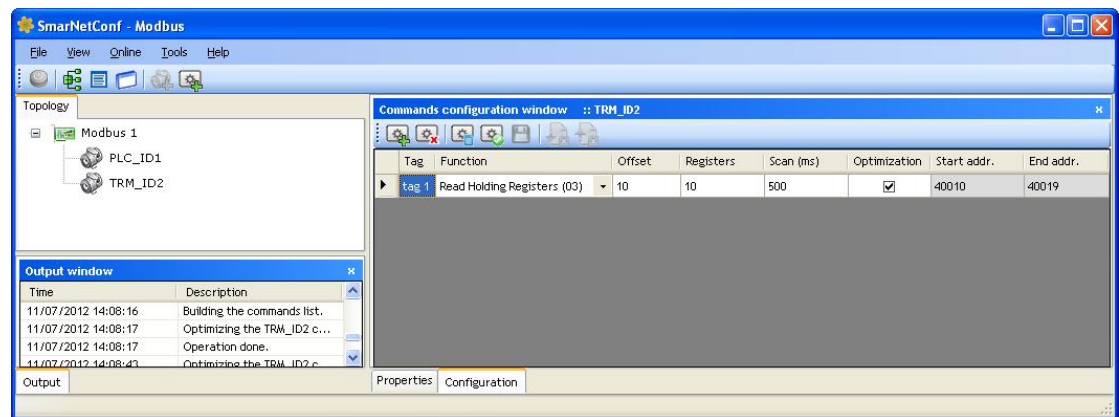


Figure 23. 14 – Configuring the Modbus device commands

The commands configuration window on the above figure shows a toolbar in which it is possible to add, delete, check if the configuration is correct, save the configuration, import and export to a xls format spreadsheet. More information about these features refers to the **SmarNetConf** help.

For each new command, it is created a table line up to the limit of 30 commands per device. On the Modbus protocol the addressing takes into account the functions, the offset and number of registers. Hence, for each command, the user should pay attention to the mapping of these three parameters. The detailed command description is shown on the next **SmarNetConf** parameter section.

The DF89, when working as master, admits the Modbus functions shown on the table below. At the “data type” column RW means that it supports reading and writing and RO supports only reading.

Modbus Function	Description	Data Type
1	Read Coils (0x)	Discrete Input (RW)
2	Read Discrete Inputs (1x)	Discrete Input (RO)
3	Read Holding registers (4x)	Analog Input (RW)
4	Read Input registers (3x)	Analog Input (RO)
5	Write Single Coil (0x)	Discrete Output
6	Write Single Register (4x)	Analog Output
15	Write Multiple Coils (0x)	Discrete Output
16	Write Multiple Registers (4x)	Analog Output

Table 23. 1 – Modbus Functions Supported

After including the command check the configuration and save it. This saving indicates that every device configuration command made on the SYSTEM302 database will be effective. In order for this to happen there can be no inconsistencies on the commands mounted (like the inclusion of repeated points on the commands or a command bigger than the limit. This is done by the **Check**



and **Save**



buttons in the **SmarNetConf** toolbar.

The **Scan** and **Optimization** fields are advanced parameters that can remain as **Default**. The **Scan** field indicates the minimum time desired between the scan on the respective command. The **Optimization** field indicates that the Modbus command is to be optimized or not. The latter, if the user chooses not to optimize it, will be activated the same way it was configured.

Finally, the **Start Address** and **End Address** fields are ways to visualize the Modbus points. The **Start Address** represents the beginning of the Modbus range addressed, while the **End Address** represents the end of that command. This type of visualization can be of 5 digits (under the Modbus Modcon specification) or 6 digits (under a Modbus 2.0 spec). The device **Digits Address** property indicates if the visualization will be with 5 or 6 digits.

For example, on figure 23.13 a visualization of 5 digits was used. The first command was mapped as follows: analog reading “03” function, offset equal to 10 and number of registers equal to 10. This represents the addressing of the Modbus 40010 to 40019 registers. Should the chosen visualization be that of 6 digits, so the corresponding Modbus **StartAddress** addresses would be 400010 and the **EndAddress** would be 400019.

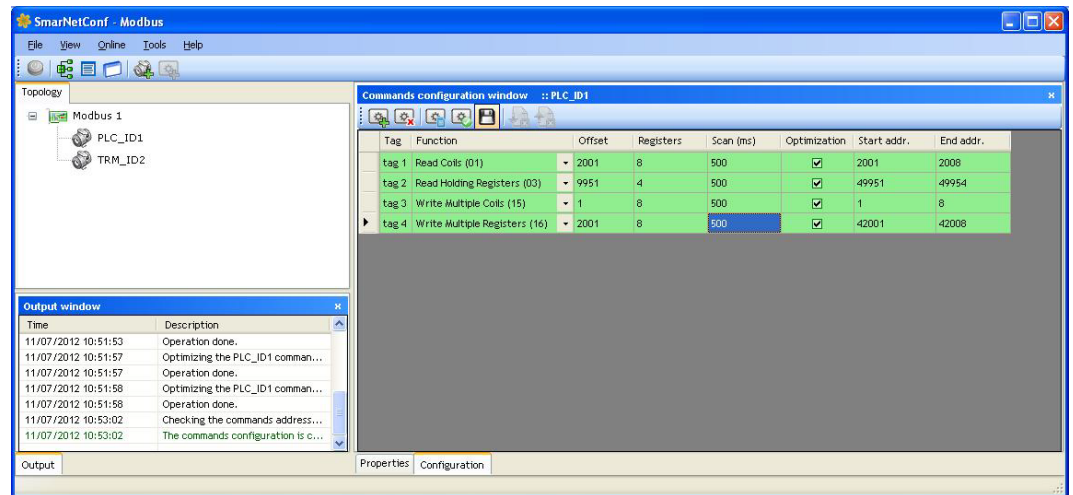


Figure 23. 15 – Configuring the PLC_ID1 commands

When the previous steps are finished, the Modbus channel configuration can be saved and the **Smar Network Configurator** can be closed. Automatically the **Mapping Tool** launches.



Do not forget to save the commands configuration before exiting the **Smar Network Configurator**. This is the only point to be saved manually. The changes of the properties on the channel or the device are saved automatically.

Step 09

Mapping the Modbus points for the SYSTEM302 logic

The **Mapping tool** is responsible for mapping the Modbus network points on points available for use in Ladder Logic or Function Blocks logic. This tool also configures all the features of the Modbus network points (user tags, data types, scales, etc.). Observe the next figure:

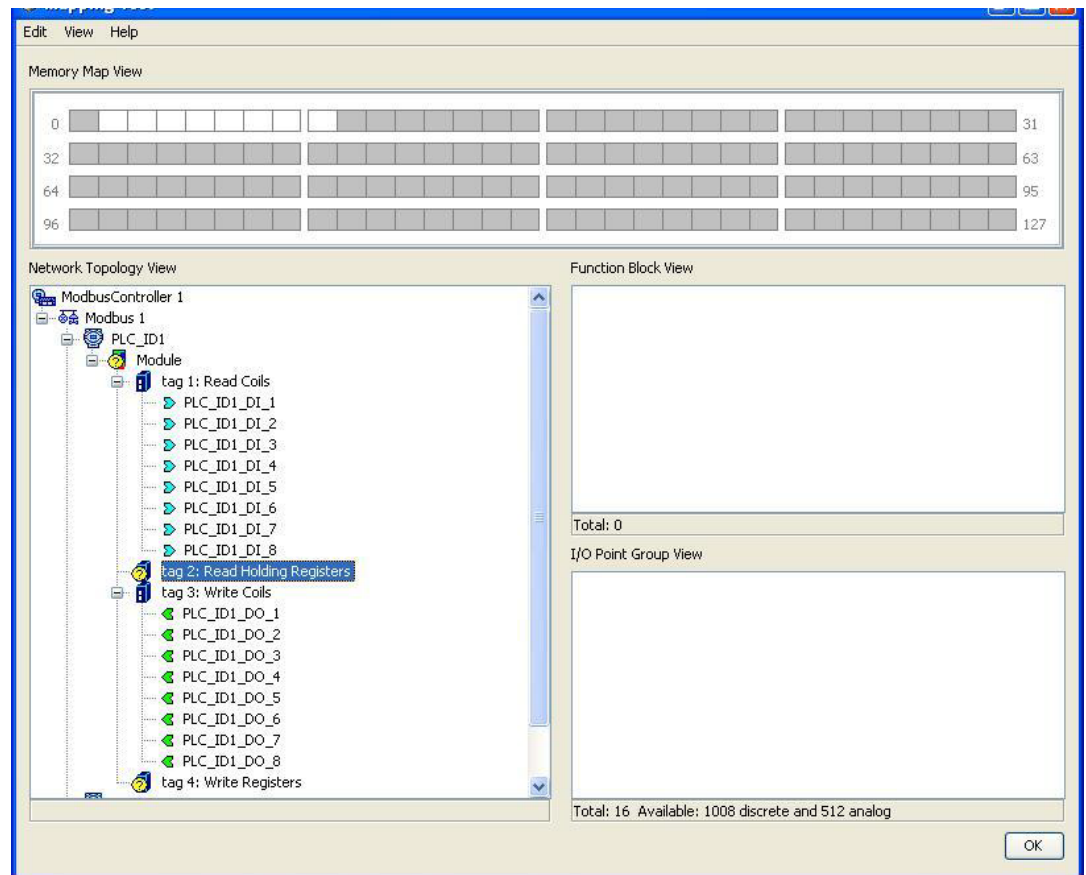



Figure 23. 16 – Mapping Tool window

As seen on the previous figure, the device was inserted on the Modbus network with their respective commands. The **Function Block Label** shows the points that will be visualized on the **Syscon** and on the **IO Point Group View** window are shown the points to be seen on the **LogicView for FFB**. The purpose here is to configure each network point (or byte) at their respective data type.

A **Mapping Tool** maps automatically by default all registers created on the **SmarNetConf** with discrete functions (Modbus 1, 2, 5, 15 functions) as Boolean type. The user can only alter the point tags. The analog points that appear with the  symbol must be configured.

To configure the point, click the command to be configured; on the next figure, for example, the command will be **Tag2**. Remember that on the **SmarNetConf**, **Tag2** was mapped on the Modbus from 49941 to 49954 register range.

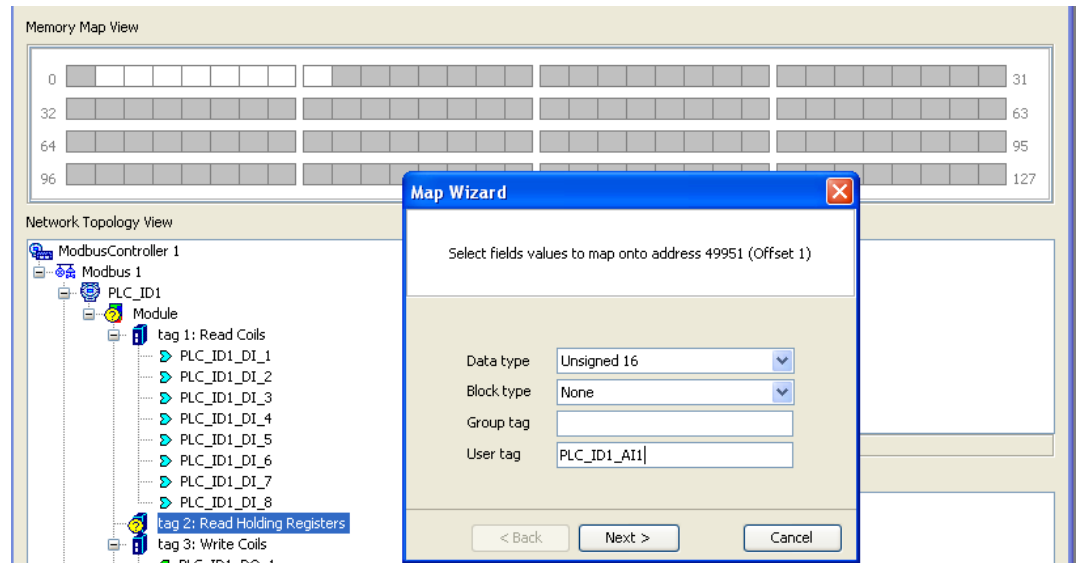


Figure 23. 17 – Mapping the analog points on Mapping Tool

On the window upper part, the available bytes will be shown in white. Click the byte and a window to configure the point will appear. For our example on **Tag2** the first available byte was clicked. According to the mapping window it corresponds to the 49951 address (Byte Offset 1).

Choose the data type (**Data type**) of the slave device variable. The available types depend on the chosen device and must be obtained in the slave device manual. For our example of the data type of the point 49951 is **Unsigned16**. For analog points can also be set for each scale point. More information about Data type will be shown in the Advanced Topics section. The figure below shows the configuration of the point scale.

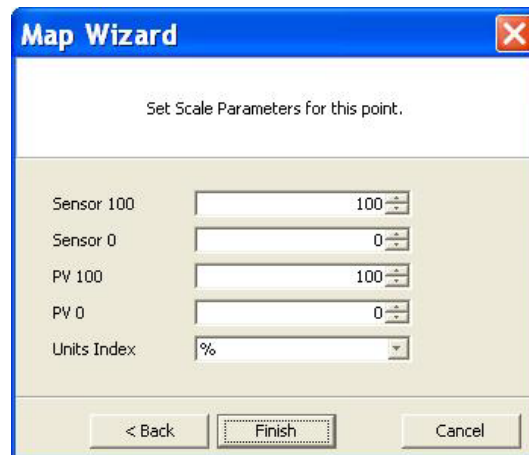


Figure 23. 18 – Configuring the scale parameters

By right-clicking the point you can remove it with the **Delete** option, or define some attributes that identify it better, with the **Attributes** option. The **Mapping tool** automatically creates the description with the Modbus address. This is useful to identify the point through the Modbus address in the ladder logic. Fill out the necessary fields and click **OK**.

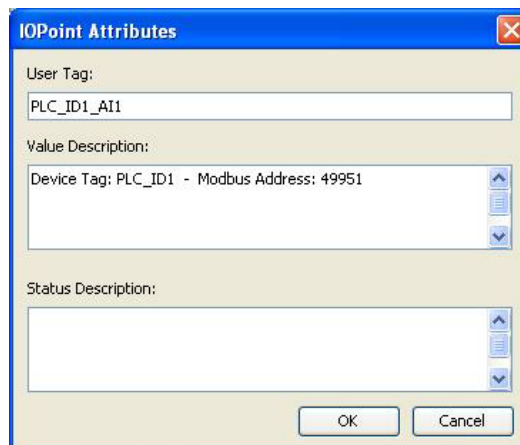


Figure 23. 19 – Altering the attributes of the mapped points

It is still possible to choose between the data on the direct or swapped format for analog types with 4 bytes (float, uns32 and signed32). This device information tells if it uses data with the swapped format or not. To configure the swapped format, right-click the device on **Mapping tool** and choose **Swapped**, as shown on the figure below. The Advance Topics section explains in detail the type of swapped data.

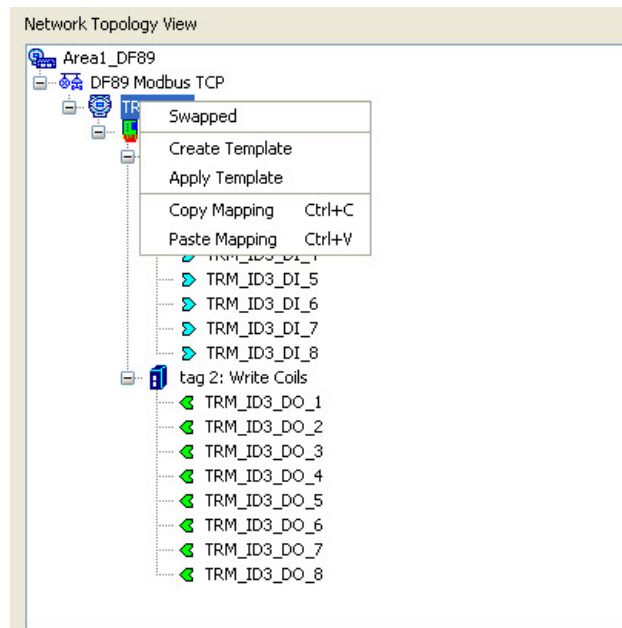


Figure 23. 20 – Example of configuration

After mapping and configuring all points, click **OK** to close the **Mapping Tool**.



Save the configuration after returning to the **Syscon**.

DF89 Configuration as Modbus TCP Master

This section will detail all steps to make the following configuration:

- a. Add devices on the DF89 TCP channel
- b. Add Modbus points on the slaves
- c. Map all Modbus points for the **SYSTEM302**

As an example of the figure 23.1 process, the Modbus TCP Master configuration uses the PLC_ID3 slave transmitter.

To configure the Modbus TCP channel, the procedure is the same used on the serial channel from step 5, where the Modbus TCP also is the **SmarNetConf**. To that effect, choose the Modbus TCP as channel type on the figure 23.7 window and expand the TCP channel. From the area window expand the TCP channel (DF89 Modbus TCP). Select the channel on the window top and right-click it. Choose the **Modify Configuration** option.

The figure below shows the TCP channel main **SmarNetConf** window.

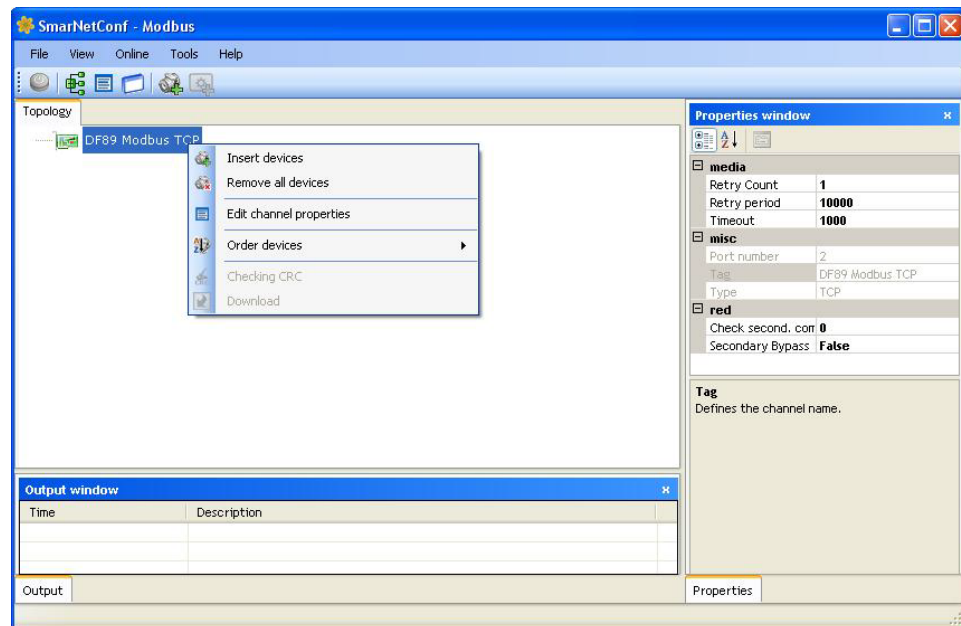


Figure 23. 21 – SmarNetConf Modbus Configurator main window

The CHECK SECOND. COMM. parameter has similar functionality of CHECK_COMM_STANDBY parameter of MBCF block that is used by other DF1302 controllers. So the secondary DF89 (standby) will perform a communication test with the slave devices in the period configured in this parameter.

Configuring the Modbus devices

To insert a slave device, click the **Insert Device** button located on the toolbar, or through the **Insert→Device** menu, or still right-clicking the TCP channel as shown on the next figure. The window shown on the next figure is the window for slave device insertion where the user can add one or several devices of the same type.

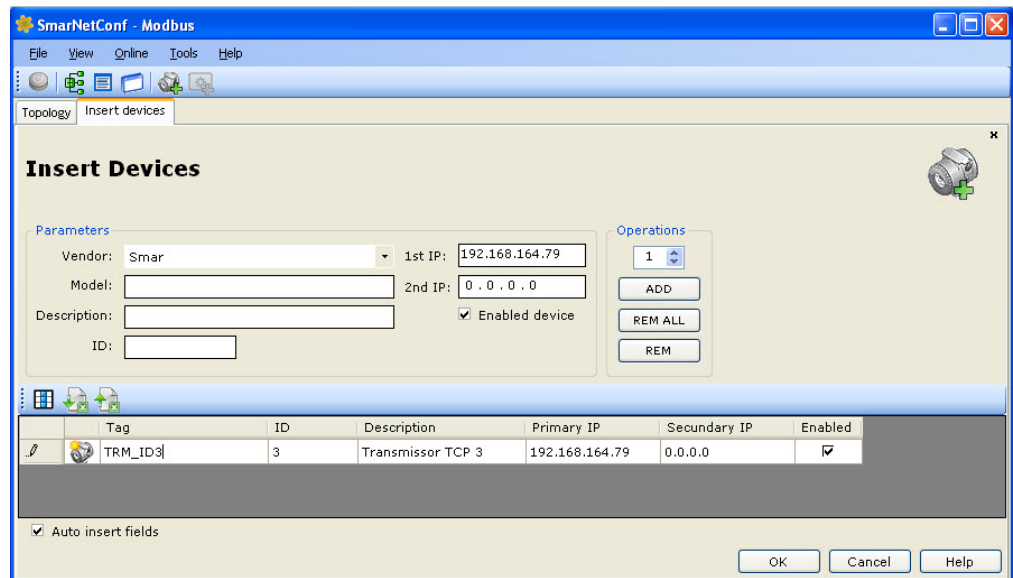


Figure 23. 22 – Adding device on the Modbus network

Select on this window the slave device to be inserted. One device only will be included on the proposed example. First, get the slave device Modbus configuration on its manual. Follow the steps below to insert the slave device:

- A. Fill out the **Model**, **Description** and **ID** of the device;
- B. Fill out the device IP address on the **1st IP** field. If the device is redundant, the **2nd IP** will be chosen as redundant path. For more details see the Redundancy section;
- C. Select the device quantity of a determined type. In this case it will be only one.
- D. Click **ADD**. A table will be shown with the quantity of selected device.
- E. On the **ID** field, attribute an address for this device.
- F. On the **Tag** field is attributed the device tag. If the user does not attribute a tag, a default tag is attributed.
- G. Click **OK**. The added device will be included on main window.

NOTE

The addresses attributed, as well as the device tags must be the only ones on the Modbus network and also on the **Studio302** Workspace database. Refer to the **Studio302** manual.

Editing the devices properties and commands

After adding the device, its properties can be changed. To alter the device properties, add and delete commands, follow the same procedures described for the serial channel. The device properties and commands are described in detail on the SmarNetConf Parameters section.

For the TCP channel example there is a TRM_ID3 device. The following figure shows the **SmarNetConf** screen with the **TRM_ID3** device of the proposed example.

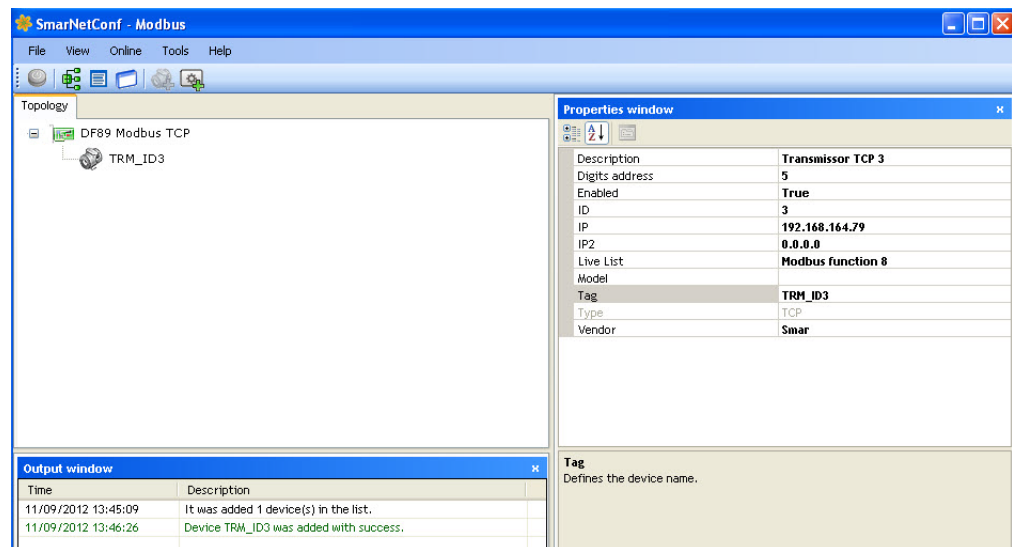


Figure 23. 23 – Example of the TCP channel properties

Making logic with the Modbus I/O Points

This section will detail all steps to make the following configurations:

- Make a logic in blocks or ladder with the slave Modbus points
- Download the configuration.

Until now the Modbus TCP and Serial slaves were created and their points were mapped. The next steps will show how to use the points mapped on the **Network Configurator** for the control logic. As mentioned previously, there are two ways to map inputs and outputs on ladder logic or function blocks. Each tool has its specific block library. It is possible to blend both logics through the FFB 1131 block.

Initially the steps will be shown to use only the Ladder logic. After that, it will be explained how to mount the logic using function blocks. Which tool to use will be user's choice.

Mapping the Modbus points to be used on Ladder

Step 10

As the points will be mapped to be used on ladder, by returning to the figure 23.17, it is not necessary to use the **Block Type** option (None). A tag can be attributed to the group on the **Group Tag** option or leave the default tag (**Group Tag** is important for use in blocks). Finally the **User Tag** to be shown on the ladder logic will be chosen. In our example the 49951 point will be **PLC_ID1_AI1** as seen on figure 23.25.

Sometimes, especially when the **LogicView for FFB** is already open, the points mapped on the **Mapping Tool** do not appear on **LogicView for FFB**. If so, execute the **Refresh Data** command to update the I/O points configured on the **Mapping Tool** on the ladder logic by right-clicking **Network I/O** at the **Hierarchy** window of the **LogicView for FFB**. See the next figure.

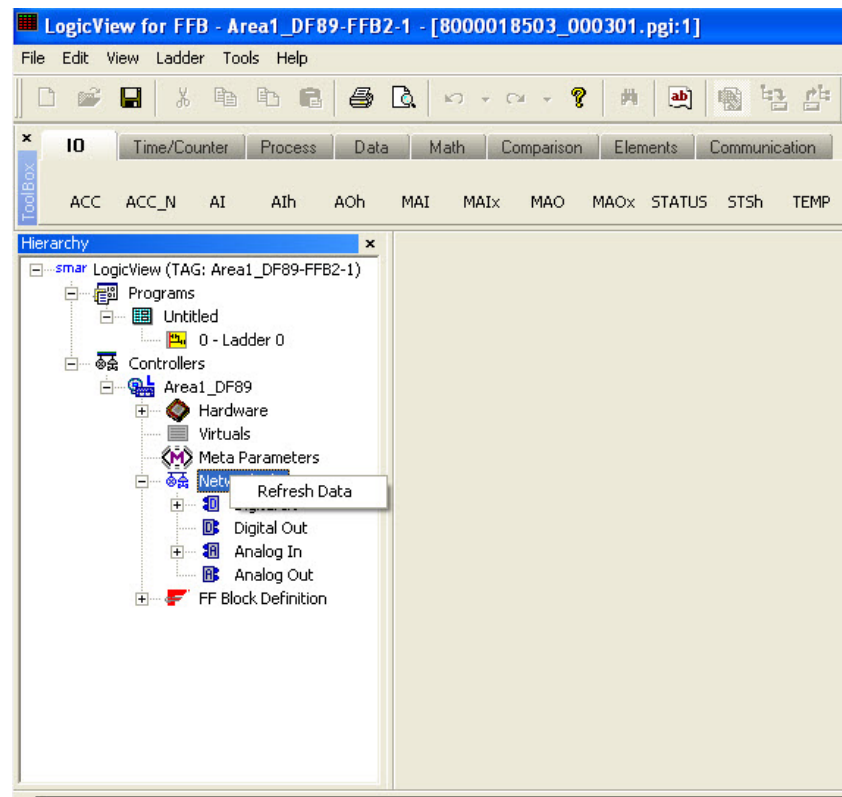


Figure 23. 24 – Executing the Refresh Data

The NetIO points can be seen on the **LogicView for FFB Hierarchy** window. On the **Object Properties** window is generated initially a description of the Modbus address variable. On the figure below, **Hierarchy** is on the left side window with all the Modbus points mapped on the **Mapping tool**. On the **LogicView for FFB** all points of both channels are shown. On the right side of the figure is shown the **Object properties** window of the **PLC_ID1_AI_1** selected point that was configured on the Modbus 49951 address.

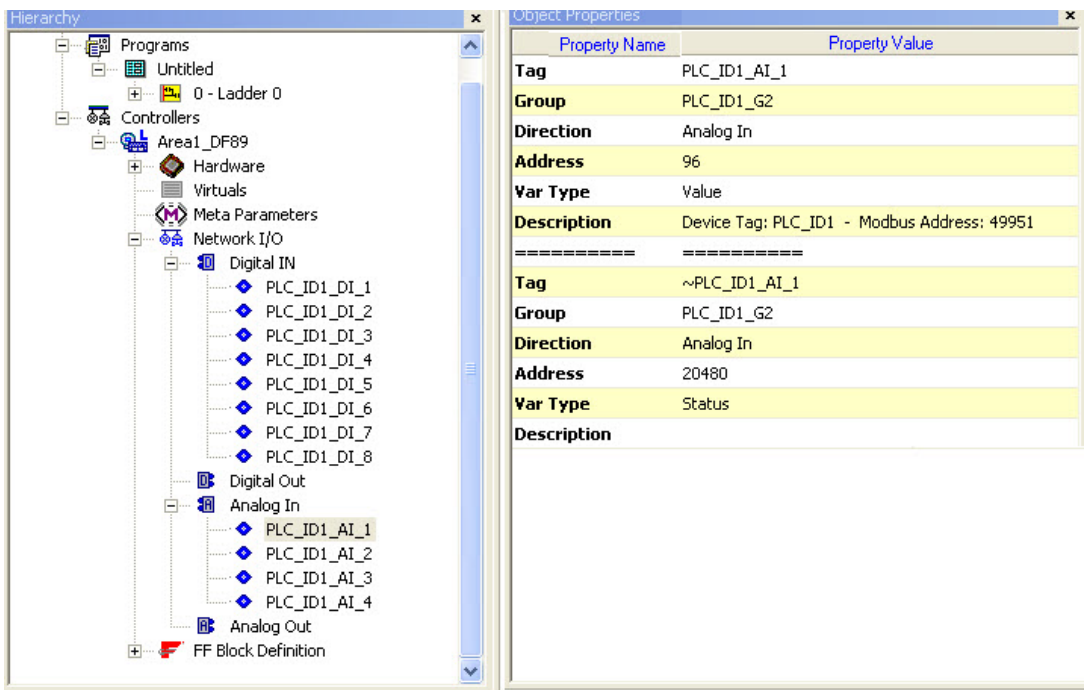


Figure 23. 25 – Network I/O on LogicView for FFB

After this procedure, the Modbus network I/O points are available to be used on the ladder logic,

counting on a complete function library to be utilized by the user. For more details see the **LogicView for FFB** manual.

The figure below shows two Modbus points used on ladder logic. The discrete **PLC_ID1_DI_1** point and the analog **PLC_ID1_AI_1** point. All of the Modbus I/O points appear as the **(MB)** symbol on ladder logic.

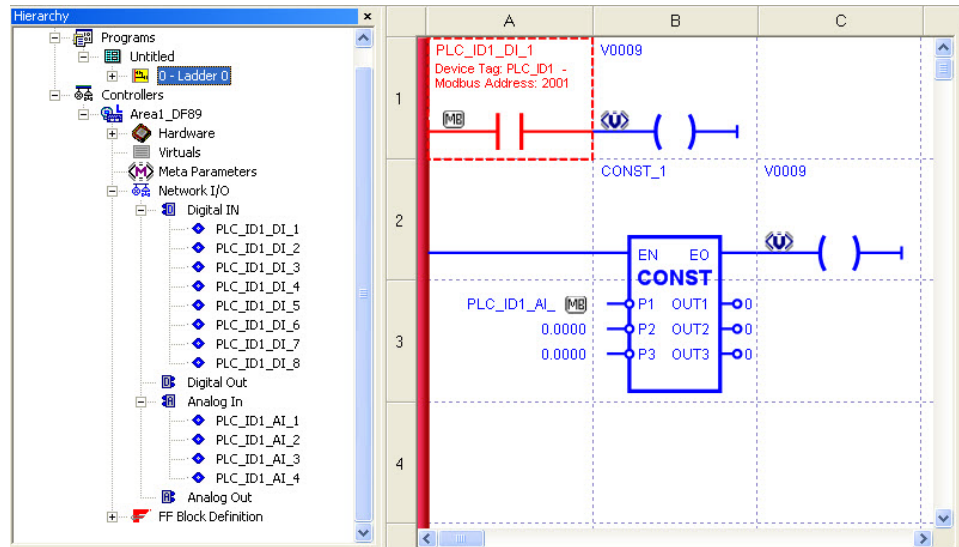


Figure 23. 26 – Editing the logic

The configuration of the serial channel using ladder logic only ends here. Save the **LogicView for FFB** and return to the **Syscon**.

After configuring the ladder, the user can download the configuration on the device, or instead choose to configure function blocks on the **Syscon**. Should the user wants to work also with function blocks, it is necessary to follow the steps 9 to 12 for the function blocks edition.

To follow the same example, it is still necessary to configure also the TCP channel. To configure the TCP channel without using function blocks go to the **Configuring TCP Channel** step. To download only the configuration up to this point, go to the **Configuration Download** step.

Step 11

Mapping the Modbus points to be used in Function Blocks

Similarly to the configuration was done through the ladder, the configuration can be done using function blocks. This section will show the mapping of the same Modbus I/O point of step 10, but now using function blocks.

As stated previously, for the creation and insertion of network devices the steps are common up to step 8. That is, we must follow the complete procedure for the area creation of **Syscon** and set the controller, as done in steps 1-4 and the devices should also be inserted in the same way as done for the ladder.

To map the I/O points in function blocks in the **Mapping Tool** window of as shown in the figure 23:17, it is necessary to fill the **BlockType** field. When is chosen **None**, as shown in step 10, indicates that the point only is mapped in the ladder.

The user can only choose to map the I/O points in function blocks of type multiple inputs (analog or discrete - MAI and MDI) or multiple outputs (analog or discrete - MDO and MAO).

At this point the user is ready to develop the control strategy. In this case the user may use other function blocks to make the desired control strategy. To interconnect the blocks the user needs to make links between them. To open a control strategy in **Syscon** is necessary to create a control module, and then, the strategies. For details on these procedures, it is recommended to check the **Syscon** manual.

An important parameter that must be configured in all function blocks is the MODE_BLK parameter that determines the Out of Service (OOS), Manual (MAN) or Automatic (AUTO) block operations. All blocks in this area must have the MODE_BLK.TARGET in AUTO. Each block has a specific parameter configuration. For more information about each block see the Function Blocks manual.

NOTE

The Modbus I/O blocks do not need to configure the CHANNEL parameter. This parameter will only work when using the I/O blocks to map the local I/O (IMB).

Commissioning and configuration download to the controller

After configuring the parameters, the user can begin communicating with the devices. It is necessary to perform the device commissioning so that each device tags, IDs and addresses be attributed adequately. If this procedure is not observed, the **Syscon** will detect the non-commissioned device and the download will be aborted. The commissioning consists in combining a network DF89 controller and the configurator used on the configuration. The same configuration tag will be attributed to the controller.

To commission, simply enable the online communication with the network equipment through the ON button on the **Syscon** toolbar shown below.



Figure 23. 27 – Enabling Online mode

Then, click the controller or the channel and choose **Commission**. More information on how to commission refer to the **Syscon** manual.

After finishing the devices commissioning, the download process may begin. Download can be done in several **Syscon** points and depends on the purpose: on all controllers, on only one controller, etc. However, the **Syscon** download is recommended for being the most complete, as it controls the alterations performed on all tools.

The **Syscon** has two download procedures: conventional mode (non-incremental) and the incremental download.

- On conventional or non-incremental download the entire configuration is downloaded on the device.
- Incremental download includes a phase that compares the configuration stored on the plant physical device with the **Syscon** configuration, stored on the **Studio302** database. Only the differences are downloaded to avoid that unnecessary information are sent to the device. On this option, the download scope can be limited by the user.

The figure below shows the device download with the option for performing the incremental download or not (which, in this case would be the total download).

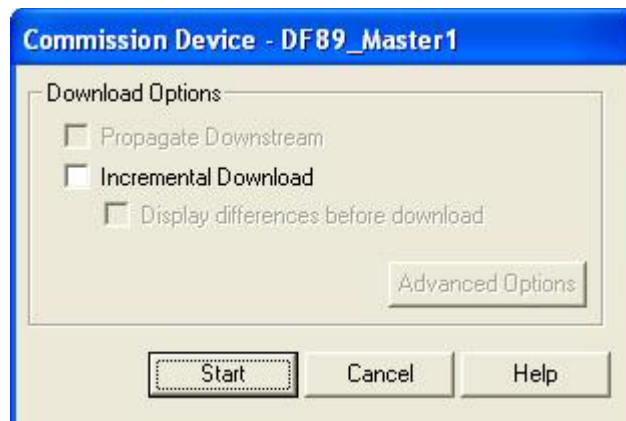


Figure 23. 28 – Performing the download

The download operation can affect a network of devices (when selecting a Fieldbus network, an HSE network or a Linking Device channel) or a particular device, possibly a HSE device (Bridge, Linking Device, Controller), or a H1 device. In the latter case, the operation is also called **partial download**.

The download operation will be available only when the **Syscon** is on online mode, communicating with the plant and with the device already commissioned.

NOTE

If the user has finished the commissioning, the **Syscon** suggests a download. Sometimes, after the commissioning is performed, it is necessary to click on the **Refresh** button at the right side of the online button on the Windows toolbar.

If the user chooses the incremental download, this option must be selected in the previous window. A window will appear with the incremental download configuration options. If there are some of the selected options, it means that **Syscon** will compare initially that option with the controller and will only download if there are differences. Before downloading, the **Syscon** still will show the user a window with the differences. Only when the user verifies the differences the new configuration will be really downloaded.

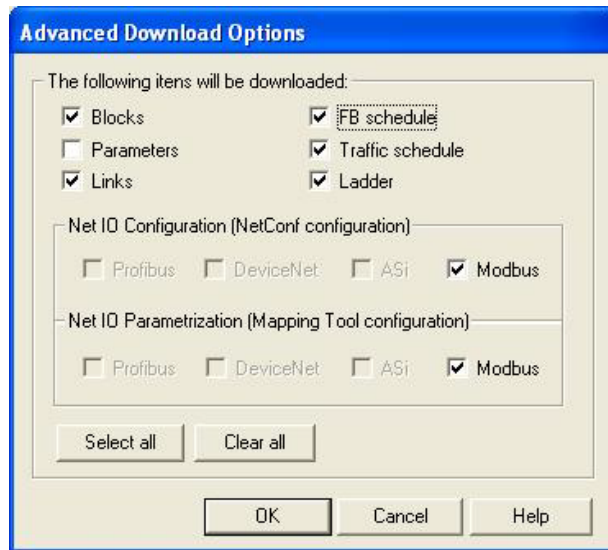


Figure 23. 29 – Selecting download options

On the present example, only **PROJ DF89** is to be downloaded. So, right-click the controller and select the **Download** option. On the download window, select **Start**.

After performing the download, the user can check the logic parameters supervision on the device. To supervise the parameters, both of ladder and blocks, the user must perform **Export Tags**. This procedure indicates that the database will be updated with the desired supervision information. **Export tags** is done on the area window by right-clicking its name.

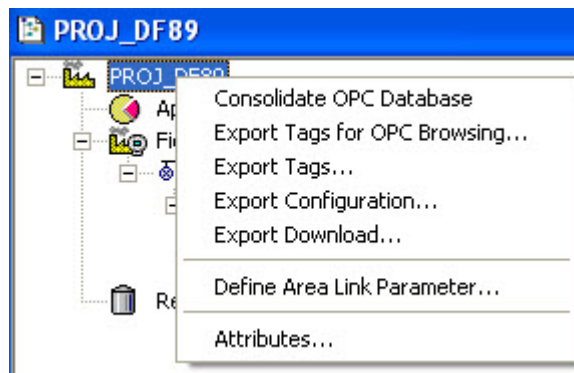


Figure 23. 30 – Performing the Export Tags

After this operation the user can supervise the configuration parameters. New ladder logic or block alterations may be applied. The **Export Tags** procedure must be repeated only if new ladder logic or block alterations are included.

DF89 Configuration as Modbus Slave

DF89 can work as Modbus Serial slave or TCP. In this case, any Modbus Master equipment can read the points on the DF89 controller, be them conventional I/O, Modbus I/O points of slaves scanned by the DF89 or other DF89 internal points. On the figure 23.1 example, the DF89 is the Modbus slave of an IHM Modbus TCP.

To configure the DF89 as Modbus slave, a block must be first created on **Syscon - Modbus Slave Configuration**. To that effect, right-click the **HSE_FB_VFB** of DF89 controller, as shown on the next figure.

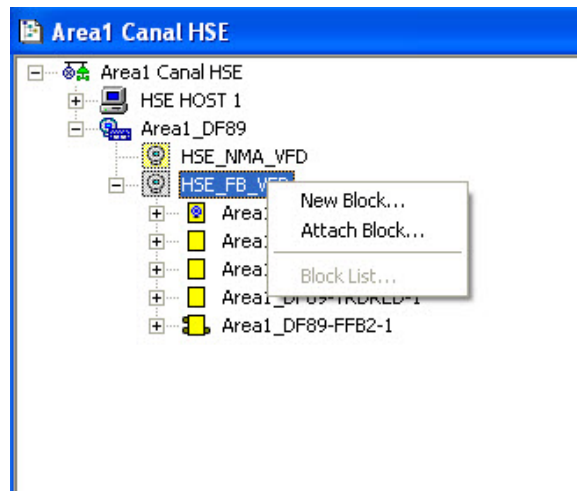


Figure 23. 31 – Including a new block on the controller

Then, a **Modbus Slave Configuration** block must be created. This block represents the entire DF89 configuration as Modbus slave (Serial or TCP).

After creating the block, the user may configure the parameters. As Modbus TCP slave the main parameters are **MODE_BLK** and **DEVICE_ADDRESS**. On figure 23.1, the DF89 has the Modbus address 25.

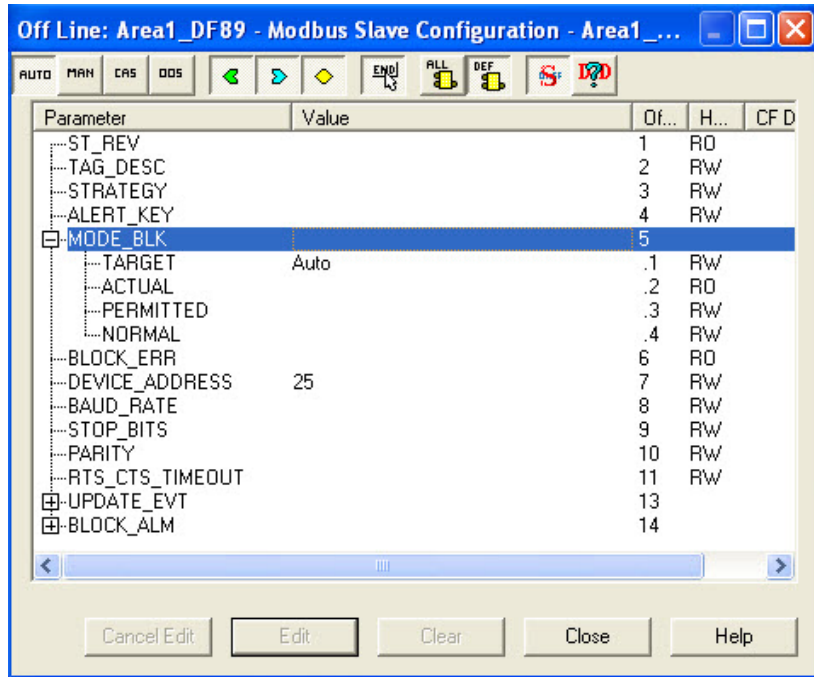


Figure 23. 32 – Modbus Slave configuration parameters

The Modbus Serial slave configuration requires the following procedure:

- 1) No serial channel can be created on the **Syscon**, because the DF89 has only one serial channel. The serial channel as master has priority over the slave function.
- 2) **MODE_BLK** and **DEVICE_ADDRESS** parameters must be configured on the **MBSCF** block.
- 3) Serial media parameters must be configured on the **MBSCF** block: Baud rate, Stop Bits and Parity.
- 4) The serial can be checked if it is enabled as slave or master on the **Modbus Transducer** block **SERIAL_CHANNEL**.

Only the points mapped on ladder logic can be mapped as Modbus slaves. Each Modbus point on ladder has a respective Modbus address. To get information about the point, click it and check the **Object Properties** window, as shown on the next figure.

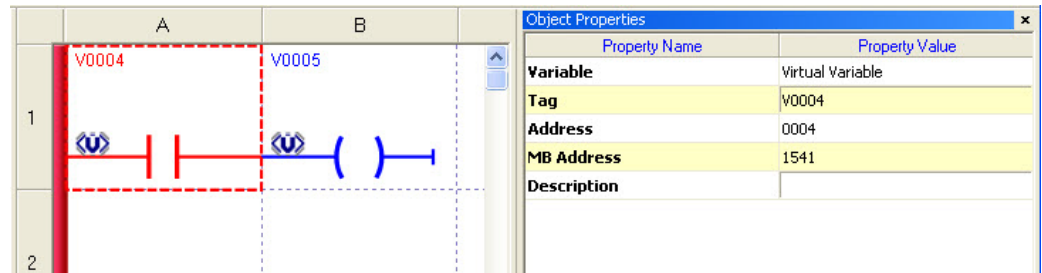


Figure 23. 33 – Example of how to get the Ladder point Modbus address

On the example of the figure above, the V0004 contact has **1541** as Modbus MB address.

NOTES

- 1) All analog points are floating type.
- 2) Only Modbus multiple writing functions are admitted, namely, functions 15 and 16.
- 3) The DF89 is already produced as Modbus TCP and Serial slave on address 247 with baud rate 9600/1/Even.

DF89 Configuration as Bypass Modbus only

Another DF89 feature is to work as **Bypass Modbus**, i.e. Modbus TCP master equipment can access a slave Modbus serial equipment situated below the DF89.

The DF89 configuration as bypass is done by creating a serial channel. If the user wishes to work only as bypass, simply create a serial channel on the **SmarNetConf** and configure the serial media.

On figure 23.1 example, the serial channel with devices has already been created, so it is already supporting the bypass automatically. Take as example the IDM Master TCP accessing any point of the **TRM_ID2** or **PCL_ID1** slaves simply by attributing the **ID** of the respective device (**ID 1** or **ID 2**) and the DF89 IP.

NOTES

- To work as bypass, the Modbus slave address cannot be the same as the **MBSCF** block **DEVICE_ADDRESS**.
- The media configuration is done only on **SmarNetConf**. On the bypass case, the same download can be used on the modification of any media parameter.
- The bypass is supported by the two DF89 Ethernet ports: ETH1 and ETH2.
- On the redundancy case, it is also possible to access the secondary port as bypass, simply by configuring the parameter of the **BYPASS_SECONDARY** device on **SmarNetConf**.

Advanced Modbus topics

There are two main Modbus specifications: an old one, created by Modcon company – the predecessor of Modbus, which follows the Modcon **PI-MBUS-300** 1996 reference book. The current Modbus version follows the Modbus Organization specification – Application Protocol Specification – since 2002. Both specs are totally compatible, with only one difference of visualization and size on the addressing band.

The Modbus register is a 2-byte number (Word) that indicates a given parameter on the slave device. For example, address 1 indicates an I/O point, while address 2 indicates another I/O point. This Modbus Modcon addressing has specific ranges depending on the function shown below, taking into consideration that the Modbus protocol function indicates if it is desired working with register reading or writing. The functions supported and the description for each function are on table 23.3.

- Functions 1, 5, 15 – addressing range Modbus Modcon 0xxxx (0001 – 10000)
- Functions 2 – addressing range Modbus Modcon 1xxxx (10001 – 20000)
- Functions 3, 6, 16 – addressing range Modbus Modcon 4xxxx (40001 – 50000)
- Function 4 – addressing range Modbus Modcon 3xxxx (30001 – 40000)

For the current version, also called *extended Modbus*, this range no longer exists, since the frame format already ignore it. This kept the addressing range from being restricted to the 10000 range registers.

A favorite format to use these ranges as reference is the quantity of bytes on the Modbus address, i.e., if it uses 5 or 6 digits. The table below compares old and new addressing as address examples.

EXAMPLE OF USER INPUT		VIEW 5 DIGITS		VIEW 6 DIGITS		MODBUS IN THE BUS	
Function	Start Address	Start Address	Start Address	Start Address	Function	Start Address	Start Address
1	2000	2000	2000	2000	1	1999	1999
3	10	40010	400010	400010	3	9	9
2	2500	12500	102500	102500	2	2499	2499
4	1533	31533	301533	301533	4	1532	1532
3	15200	Cannot be seen	115200	115200	3	15199	15199

Table 23. 2 – Example of Modbus address on the old protocol (5 digits) or the current protocol (6 digits) and how the protocol stays internally (Modbus online)

The table above shows examples of Modbus addressing using both specifications: Modcon (with 5 digits) and the current one (with 6 digits).

The user input columns show an example of a given Modbus function and register.

The Modbus column on the bus shows a would-be Modbus protocol. Usually, the user does not see it, because it is inside the protocol for communication between devices. In this case, the initial bus

address is always a relative address, i.e., they generate the same command on the bus. In this case, the initial bus address is always a relative address discounted from the function base and it is unique for both representations.

Table 23.2 shows that both protocol views do not have any difference for communication, i.e., they generate the same bus command. The user then chooses which way he wants to work with, the old or the current one. The only difference will be when the device will have a very large address range (over 10000 registers) and so only the 6-digit format will be feasible.

NOTE

The DF89 supports the Modbus TCP and the Modbus RTU serial protocols. For both protocols the standard is the same under the old or current specs, and they are totally supported by the controller. The DF89 default is the 5-digit visualization, which is the most used.

There are other non-standard Modbus protocol extensions, like Enron and Daniel, which are not supported by the DF89.

Types of data supported

On the Modbus, the points are mapped in registers. These registers differ, depending on the type of variable. For discrete types (functions 1, 2, 5, or 15), each register occupies a 2-byte word. Besides this information it is necessary to know which type of data corresponds to these registers. The information on the data types supported by the DF89 Modbus are the following:

Modbus Function	Data types Supported	Size in Bytes	Nr of Modbus Registers Necessary
1, 2, 5 or 15	BIT	1/8	1
3, 4, 6 or 16	UNSIGNED 8	1	1
3, 4, 6 or 16	UNSIGNED 16	2	1
3, 4, 6 or 16	UNSIGNED 32	4	2
3, 4, 6 or 16	FLOAT_SINGLE	4	2
3, 4, 6 or 16	SIGNED 8	1	1
3, 4, 6 or 16	SIGNED 16	2	1
3, 4, 6 or 16	SIGNED 32	4	2

Table 23. 3 – Data types supported

Besides these data types, the analog data of 4 bytes size can be swapped or not. In this case, the type of swapped data has a little-endian data inversion relative to the big-endian non-swapped data type. This information is selected in the device inside the **Mapping Tool** and it is valid for the whole device.

For example, considering the number of 4 bytes 12345678 allocated on the Modbus 40001 and 40002 addresses:

Datatype	Register 40001	Register 40002
Unsigned32	5678	1234
Swapped Unsigned32	1234	5678

Supervision and Control using DF89

The Modbus variables in the slave device can be mapped in DF89 by user in the following usage scenarios: supervision or control. As supervision, the variables are read cyclically via Modbus from the slave device and stored in the memory of DF89 and can be written on demand. As control, the variables are read cyclically (and do not allow writing) or written cyclically (and are never read).

Characteristic	Usage scenario: Supervision	Usage Scenario: Control
Modbus command configuration on SmarNetConf	Only Modbus commands for reading: - Read Coil (command 1) - Read Holding Register (command 3)	Modbus command to read or write: - Read Discrete Input (command 2) - Read Input Register (command 4) - Read Coil (command 1) - Read Holding Register (command 3) - Write Single Coil (command 5) - Write Single Register (command 6) - Write Multiple Coil (command 15) - Write Multiple Register (command 16)
Type of Modbus variable in the slave device and type of possible Modbus command.	Coil (address 0x.xxx or 0xx.xxx) and Holding Register (address 4x.xxx or 4xx.xxx) : Modbus command for reading	Discrete Input (address 1x.xxx or 1xx.xxx) and Input Register (address 3x.xxx or 3xx.xxx): Modbus command for reading Coil (address 0x.xxx or 0xx.xxx) and Holding Register (address 4x.xxx or 4xx.xxx): Modbus command for reading or written
Modbus variable on DF89 (cache)	Discrete Input (address 1x.xxx or 1xx.xxx) or Input Register (address 3x.xxx or 3xx.xxx) with float data type.	If Modbus command for reading: Discrete Input (address 1x.xxx or 1xx.xxx) or Input Register (address 3x.xxx or 3xx.xxx) with float data type. If Modbus command for written: Coil (address 0x.xxx or 0xx.xxx) or Holding Register (address 4x.xxx or 4xx.xxx)
Frequency of sending Modbus command for reading	Cyclic according to SmarNetConf configuration.	Cyclic according to SmarNetConf configuration.
Frequency of sending Modbus command for written	User demand. When writing occurs via Smar.DfiOleServer.	Cyclic according to SmarNetConf configuration.
Application	DF89 works as concentrator for an OPC Server/Client (it scans the slave devices and stores in the DF89 memory for host access). Despite the Modbus variable on DF89 is Discrete Input or Input Register, the written via Smar.DfiOleServer will be converted to written Modbus command to the slave device.	DF89 transfers cyclically a variable: - from a Modbus device to another one - from a Modbus device to a FF device - from a FF device to a Modbus device

Supervision

For example, in the HMI of example of figure 23.1 it is desired to map the point 40020 , which was a PID set point, and a contact 2001 that would be an on/off of a pump. The point will be created in **SmarNetConf** and **Mapping tool** as shown in steps 5 to 9 and would be as follows.

Modbus Function (SmarNetConf)	Start Address (SmarNetConf)	Point Tag (Mapping Tool)	Description
Read Coils (01)	2001	PLC_ID1_DI_1	On/Off Pump
Read Holding Registers (03)	40020	PLC_ID1_AI_1	SP of PID

These points were used in the ladder in a particular control strategy, and could also be mapped in supervisory via Smar.DfiOleServer. In supervisory, the value of points **PLC_ID1_DI_1** and **PLC_ID1_AI_1** can be changed by the user where the change will be mapped into a Modbus command of writing (function 5,6,15 or 16) and will be sent to the slave. The value of the next reading will already be the value really affected on the slave. If the point continues with the previous value means that the writing was not recognized by the slave.

On ladder it is not possible to alter the set point value, as the online ladder alteration does not admit writing on input parameters like this.

Peer-to-Peer of I/O point

A point mapped as peer-to-peer must be read on some slave device and written on other slave device.

For example, if the user wants to read the **PLC_ID1** set point value and write the **TRM_ID2** set point. In this case, the **PLC_ID1** set point must be mapped with any reading function (n.1, 2, 3 or 4), and the **TRM_ID2** mapped with any writing function (n. 5, 6, 15 or 16). So, both ladder or function block logic can be used so that the point mapped as **PLC_ID1.PID1_SP** be written on **TRM_ID2.PID2_SP**.

Scan time between commands

The DF89 has the additional characteristic of choosing the scan time by command. By default, this parameter is 500 ms and must suit most applications. This value can be adjusted from 0 to 65500 ms (65.5 s).

The Modbus scan time allows the user setting different times for each command as multiples of 100 ms. For variable requiring a smaller periodicity, as analog variables, the Modbus time can be slower, for example, in relation to discrete points.

Scan time is also useful to prioritize the bypass. It refers only to the commands determined for the DF89 to execute as a Modbus master and does not influence the Modbus bypass at all. So, if using a 1000 ms scan time and there is only one command, the DF89 will execute the scan command at each second. The rest of the time it can serve the bypass. That is, this makes bypass supervision much faster, as the bypass range is bigger in this case.

This is time is the time the user. However, as the scan depends on other commands to be executed, sometimes this time cannot be effected if the scan is much bigger than necessary. For example, imagine if the user has 5 commands to execute a device data (5 commands, including the optimization). And suppose that the user set the time of 100 ms for all commands. If the time for each command is 100 ms, then the time to do all the scan would be 500 ms. As the user configured 200 ms (desired time) the time had never been able to be done because the scan occurs at least every 500ms. If the user changes the time to 1000ms then will exist a gap between the scan of 500ms.

Optimization of commands

The DF89 has an additional command optimization feature. It consists on reducing the number of commands sent online and optimizes the Modbus communication. Use only the non-selected option when the Modbus slave really does not support the optimized commands.

Enabling and disabling a device in the configuration

Another interesting **Network Configurator** feature is the possibility of enabling and disabling a slave device on the configuration without affecting the device properties window where the user executes the device scan.

When the device is disabled, although the logic configuration and device mapping are not altered, the controller does not perform any other operation with this device, not sending any command to the slave.

The figure below shows a sample configuration with disabled devices. When the device is turned off it appears with a red (x) in the device drawing. In the figure below only the devices 10, 18 and 30 are enabled. This option is very useful when there are devices for maintenance.

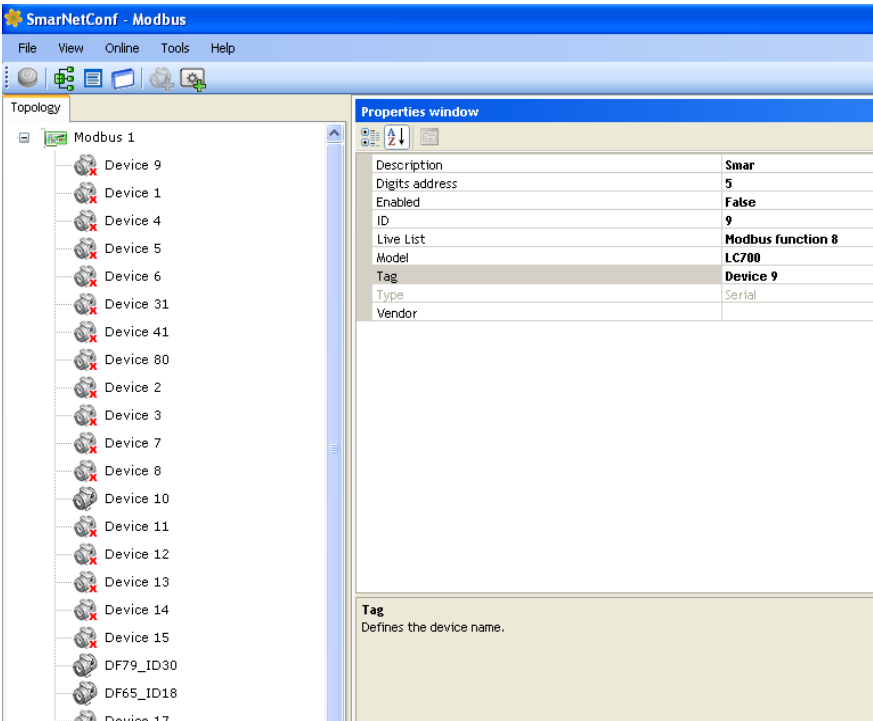


Figure 23. 34 – Example of configuration with enabled and disabled devices

This enabling change needs a download to the controller to recognize it. This download operation can be performed from the **Smar Network Configuration** or from **Syscon**.

To enable again the device, return to device’s properties window and execute a new download.

Network Configurator online communication with the controller

Direct communication with the DF89 is possible through the **SmarNetConf**. It can facilitate some functions like slave channel parameter download, among others. This communication is carried out through the Ethernet network.

To enable Modbus communication through **SmarNetConf**, first enable communication with the OPC server.

SYSTEM302 Maintenance Procedure

This section presents a summary of download procedures recommended for cases of maintenance of a **SYSTEM302** configuration.

As the **SYSTEM302** has several download modes, there must be a general way that meets most cases. However, there are other procedures for the same alteration (consult the **Syscon** manual).

On the **Download Recommended** column the reference is the **Syscon** incremental download on the plant or on the controller. These operations are explained in detail in the **Syscon** manual.

MAINTENANCE EVENT	ALTERATION TOOL	RECOMMENDED DOWNLOAD
Make a new SYSTEM302 configuration with the Modbus controller and ladder logic. FFB block has no HSE links to other controllers.	Syscon, Network Configurator, Mapping Tool, LogicView for FFB	On Syscon make an incremental plant download with all selected options or a download on the controller (*).
Include a new device or a new command on the Modbus network	Network Configurator, Mapping Tool	On Syscon make an incremental plant download with all selected options.
Alter the device configuration on the Modbus network (device address, Modbus points addressing)	Network Configurator	Make an incremental plant download on the controller on Syscon(*).
Change specific parameters on the Modbus channel (<i>baud rate, timeout</i> or other parameter).	Network Configurator	Download the Modbus network on the Network Configurator (*)
Enable or disable Modbus device keeping the entire Modbus configuration	Network Configurator	Download the Modbus network on the Network Configurator (*)
Add parameters on the FFB block and make HSE links or only add FFB parameters to a FFB block that already has HSE link.	Syscon	On Syscon make an incremental plant download (*).
Include or remove FFB link for internal blocks inside the same controller.	Syscon	Make an incremental download on the controller without the enabled NetIO options (*).
Change of mapping parameters I/O points (scale, point tags, alteration of accessed or swapped bytes).	Mapping Tool	On Syscon make an incremental download on the controller with the NetIO Parameterization enabled option (*).
Change slave device configuration parameters (tag, model or number of visualization digits) or enable/disable the slave device.	Network Configurator	Download the Modbus network on the Network Configurator (*).
Alteration only of ladder logic.	LogicView for FFB	Download the logic on the LogicView for FFB.

Table 23.4 – Recommended download procedures

(*) These operations will stop the Modbus network channel

Smar Network Configurator Parameters

Serial Channel Parameters

Category	Parameter	Valid Range	Default	Unit	Description
Advanced					
	RTS CTS Timeout	0-65535	0	ms	Specifies the maximum waiting time for the CTS become active after sending RTS on the serial ports.
	Time delay	0-65535	0	ms	Waiting time between Rx reception and the next Tx transmission.
Media					
	Baud rate	0:110, 1:300, 2:600, 3:1200, 4:2400, 5:4800, 6:9600, 7:19200, 8:38400, 9:57600, 10:115200	9600	E	Define the baud rate of serial ports.
	Parity	0:Even, 1:Odd, 2:None.	Even	E	Defines the serial ports parity
	Retry Count	0-5	1		Number of retransmissions in case the master does not receive a response from the slave.
	Retry period	0-65536	10000		Time spent on a new live list check for the devices that failed on the scan.
	Stop bit	0:1, 1:2	1	E	Defines the number of stop bits for the serial ports.
	Timeout	0-65536	1000	ms	Waiting time for a slave response after sending a command to the serial ports.
Misc					
	Port number				1
	Tag				Channel tag
	Type				Serial
Redundancy					
	CHECK SECOND COMM	0 – Disable test 1 – 65535 Interval between each test	0	ms	Parameter configured for the secondary redundant port if user wants the communication test between slave devices.
	SECONDARY BYPASS	True/False	False		Parameter configured for the secondary redundant port if wanting to use the path as bypass for a TCP master to access the serial slave below the DF89.

TCP Channel Parameters

Category	Parameter	Valid Range	Default	Unit	Description
Media					
	RETRY COUNT	0-5	1		Number of retransmissions if the master does not receive a response from the slave.
	RETRY PERIOD	0-65536	0	ms	Time of retransmission if the master does not get a response from the slave.
	TIMEOUT	0-65536	1000	ms	Waiting time for a slave response after sending a command for the TCP ports.
Misc					
	PORT NUMBER			(read only)	2
	TAG			(read only)	Channel Tag
	TYPE			(read only)	TCP
Redundancy					
	CHECK SECOND COMM	0 – Disable test 1 – 65535 Interval between each test	0	ms	Parameter configured for the secondary DF89. If the user wants to perform the periodic communication test with the slaves to check the communication.
	Secondary Bypass	0 – False 1 – True	False		

Serial Device Parameters

Category	Parameter	Valid Range	Default	Unit	Description
Advanced					
	Digits address	5 or 6	5		Indicates if the "Modbus Address" will show the number with 5 or 6 digits, indicating if it is the old Modbus Modicon standard or a new one.
	Live List	1 – Use Modbus function 8. 2 – Use the First command	1		This parameter indicates which command will be used by the master and check if the device is alive. By default the loop back command (function 8) is used, though the device may not support this command. In this case, the user may choose option 2 and the first configured command will be sent.
Main					
	Enabled	0 – False 1 – True	1		This Field defines if, and on what conditions, the command must be executed. If the parameter is configured as 0, the command is disabled and won't be executed in the normal polling sequence. If the parameter is configured as 1, the command will be executed on each command list scan.
	Slave address	1-247			This parameter specifies the node address, on the network, of the Modbus slave that will be considered. Values 1 to 247 are allowed.
Misc					
	Description				
	Model				
	Tag				Device tag
	Type				Serial
	Vendor				

TCP Device Parameters

Category	Parameter	Valid Range	Default	Unit	Description
Advanced					
	Digits address	5 or 6	5		Indicates if the "Modbus Address" will show the number with 5 or 6 digits, indicating if it is the old Modbus Modicon standard or a new one.
	Live List	1 – Use Modbus function 8. 2 – Use the First command	1		This parameter indicates which command will be used by the master to check if the equipment is alive (livelist). By default the loop back command (function 8) is used, though the equipment may not support this command. In this case, the user may choose option 2 and the first configured command will be sent.
Main					
	Enabled	0 – False 1 – True	1		This field defines if the command must be executed and under which conditions. If the parameter is configured as 0, the command is disabled and will not be executed on the normal polling sequence. If the parameter is configured as 1, the command is executed at each command list scan.
	Slave address	1-247			This parameter specifies the node address, on the network, of the slave device that will be considered. Values 1 to 247 are allowed.
	IP		Spaces		Device main IP address
	IP2		Spaces		Redundant device IP address. See redundancy cases.
Misc					
	Description				
	Model				
	Tag				Device tag
	Type				TCP
	Vendor				

Parameters of device commands (serial or TCP)

Parameter	Valid Range	Default	Unit	Description
TAG				Device Tag
FUNCTION	The options are in according to Modbus supported functions table	1		This parameter specifies the Modbus function that will be executed by the command. These function codes are defined by the Modbus protocol. The next table defines the purpose of each action supported by the module.
OFFSET	0 to 65535	0		This parameter specifies the initial Modbus register or the digital point address to be considered by Modbus slave equipment command. Refer to each slave device documentation on the network to attribute their registers and digital point address.
REGISTERS	0 to 65535	0		This parameter specifies the number of registers and digital points to be associated to the command. Functions 5 and 6 ignore this field and their only apply to a single data point. For functions 1, 2 and 15, configure the number of digital registers to be associated to the command. For functions 3, 4 and 16 configure the number of analog registers to be associated to the command. Note: the register that may be associated to a point depends on the data type. See more about association of register and point in the item data type supported described in this section.
SCAN	0 to 65500 (multiple of 100)	500	ms	This parameter specifies the minimum interval to execute continuous commands periodically. This parameter must be multiple of 100. So, if the scan time is 500 the command will be executed at each 500 ms minimum.
OPTIMIZATION	0 – False 1 – True	1		When the parameter is true (checked on), indicates that it will be used to optimize the Modbus command where the algorithm tries to group multiple internal points of the same type and the same device in the same command.
START ADDR				This Field is a composition of function and Relative Address, which will be the Modbus address in the user language. It will depend also of the Nr_digits_addr parameter. See the Modbus Address mounting.

Modbus Functions Supported

The table below lists all Modbus functions supported by DF89 and their description. This function code follows the current Modbus specification Modbus Organization – Application Protocol Specification.

Modbus Function	MODBUS_FUNCTION Parameter	Class	Description
1	Read Coils (0x)	RW	Reading of Input/Output Discrete Points.
2	Read Discrete Inputs (1x)	RO	Reading of Input Discrete Points
3	Read Holding registers (4x)	RW	Reading of Input/Output Analog Points
4	Read Input registers (3x)	RO	Reading of Input Analog Points
5	Write Single Coil (0x)	W	Writing of Discrete Point (only 1 register)
6	Write Single Register (4x)	W	Writing of Analog Point (only 1 register)
15	Write Multiple Coils (0x)	W	Writing of Discrete Points (1 or more registers)
16	Write Multiple Registers (4x)	W	Writing of Analog Points (1 or more registers)

RW - Read and Write, RO - Read Only, W - Write Only

Besides these functions is also supported the function of checking the device function (8 - check). This function is used when starting the device to know if is alive or not (LiveList). In the LiveList parameter of the device the user may choose to use this function or the first function defined in the command table.

Specific blocks of DF89 controller

Modbus Communication Transducer

This block provides the following functionalities:

- Slave device live list;
- Slave device diagnostics;
- Parameter online configuration of slave configuration.

See in the table below the parameters description.

Idx	Parameter	Data type	Valid Range	Default	Memory	Mode	Unit	Description
1	ST_REV	Uns16			S	RO		
2	TAG_DESC	Octet			S	RW		
3	STRATEGY	Uns16			S	RW		
4	ALERT_KEY	UNS8			S	RW		
5	MODE_BLK	DS_69			S	RW		
6	BLOCK_ERR	Bitstr			D	RO		
7	SERIAL_CHANNEL	Uns8	0 : Slave 1 : Master	0	S	RO		Shows if the serial port is being used as Modbus master or slave.
8	BAUD_RATE	uns8	3:1200, 4:2400, 5:4800, 6:9600, 7:19200, 8:38400, 9:57600, 10:115200	9600	S	RO		Shows the current baud rate used on master or slave on the serial port.
9	STOP_BITS	uns8	0:1; 1:2	1	S	RO		Shows the current stop bit used on master or slave serial port.
10	PARITY	uns8	0:Even 1:Odd 2:None	Even	S	RO		Shows the current parity used on master or slave serial port.
11	SERIAL_TIMEOUT	Uns16	Positive	1000	S	RO	ms	Waiting time for the slave response using the serial port.
12	SERIAL_NR_RETRIES	Uns8	Positive	2	S	RO		Number of attempts of re-sending the failing serial port command.
13	SERIAL_RETRY_PERIOD	Uns16	Positive	5000	S	RO	ms	Time spent to re-send the command to the slave that does not respond.
14	RTS_CTS_TIMEOUT	uns16	Positive	0	S	RO	ms	Specifies the maximum waiting time for the CTS to be activated after the RTS is configured on the serial ports.
15	TIME_DELAY	uns16	Positive		S	RO	ms	Waiting time between Rx reception and the next Tx transmission.
16	SERIAL_CHK_COM_SEC	uns16	Positive		S	RO	sec	Indicates to redundancy if the information that the check will be made on the secondary path was configured.
17	BYPASS_SECONDARY	uns8	0=False; 1=True		S	RO		Indicates to redundancy if the user can use or not the secondary path as bypass
18	SERIAL_ENABLE_DEVS	Uns8			D	RO		List of devices enabled on the serial network. The number represents the slave device address.
19	SERIAL_LIVE_LIST	Uns8			D	RO		List of devices connected to the serial network. The number represents the slave device address.

20	SERIAL_ERR_CNT	Uns32		0	D	RO		Serial errors number counter. Use the RESET_COUNTERS to restart the counter.
21	SERIAL_SCAN_TIME	Uns32		0	D	RO		Time spent by the serial port to scan the Modbus variables.
22	TCP_TIMEOUT	uns16	Positive		S	RO		Waiting time for a slave response after sending a command on the TCP port.
23	TCP_NR_RETRIES	uns8	Positive		S	RO		Number of retransmissions if no the slave response is received.
24	TCP_RETRY_PERIOD	uns16	Positive		S	RO		Time to re-send the command to slave that is not responding.
25	TCP_CHK_COM_SEC	uns16	Positive		S	RO	s	Indicates to redundancy if the check on the secondary path will be configured
26	TCP_ENABLE_DEVS	Uns8			D	RO		List of device enabled on the TCP network. The number represents the slave device address
27	TCP_LIVE_LIST	Uns8			D	RO		List of device connected to the TCP network.
28	TCP_ERR_CNT	Uns32		0	D	RO		TCP error number counter. Use RESET_COUNTERS to restart this counter.
29	TCP_SCAN_TIME	Uns32		0	D	RO		Time spent by the TCP port to scan the Modbus variables.
30	RESET_COUNTERS	Uns8			D	RW		Restart the TCP and Serial error counters.
31	NUM_LINKS	Uns8		1	S	RO		Defines the number of buses supported.
32	LINK_SEL	Uns8	0: First ; 1: Next ; 2: Previous ; 3: Last	First	D	RW		Selects which bus will show its information. Internal LiveList use.
33	LINK_ID	uns16		0	D	RW		Identifies the currently selected bus. Internal LiveList use.
34	LINK_ID_REV	Uns32		0	S	RO		Current bus LiveList revision. Internal use.
35	NUM_DEV	Uns16		0	D	RO		Number of devices on current bus for use on the LiveList.
36	SLAVE_SELECTOR	Uns8	0: First ; 1: Next ; 2: Previous ; 3: Last	First	D	RW		Slave address selector to read LiveList information. Internal use.
37	SLAVE_ADDR	uns8	1 to 247	0	D	RW		Slave parameter address for LiveList. Internal use.
38	UPDATE_EVT	DS-73			D			
39	BLOCK_ALM	DS-72			D			

Modbus Slave Configuration

See in the table below the parameters description.

Idx	Parameter	Data type	Valid Range	Default	Memory	Mode	Unit	Description
1	ST_REV	UNS16		0	S	RO		
2	TAG_DESC	OCT(32)		Spaces	S	RW		
3	STRATEGY	UNS16		0	S	RW		
4	ALERT_KEY	UNS8	1 to 255	0	S	RW		
5	MODE_BLK	DS_69	OOS , AUTO	OOS	S	RW		
6	BLOCK_ERR	Bitstr(2)			D	RO		
7	DEVICE_ADDRESS	uns8	1 to 247	1	S	RW		Defines the DFI302 Modbus address when working as Modbus slave.
8	BAUD_RATE	uns8	0:110, 1:300, 2:600, 3:1200, 4:2400, 5:4800, 6:9600, 7:19200, 8:38400, 9:57600, 10:115200	9600	S	RW		Defines the DFI302 serial port baud rate when working as Modbus slave. Defines the DFI302 Modbus address when working as Modbus slave.
9	STOP_BITS	uns8	0:1, 1:2	1	S	RW		Defines the DFI302 serial port stop bit when working as Modbus slave.
10	PARITY	uns8	0:Even, 1:Odd, 2:None	Even	S	RW		Defines the DFI302 serial port parity when working as Modbus slave.
11	RTS_CTS_TIMEOUT	uns16		0	S	RW	ms	When the value is other than zero, it indicates that the RTX_CTS function is enabled and the value is the connection timeout. Used when the DFI302 works as Modbus slave.
12	ENABLE_CHK_COM_SEC	Uns8		0-False ; 1-True	S	RW		When the DFI302 works as Modbus slave, this flag enables the redundancy, checks the communication on the secondary.
13	UPDATE_EVT	DS-73			D			
14	BLOCK_ALM	DS-72			D			

Redundancy and Modbus

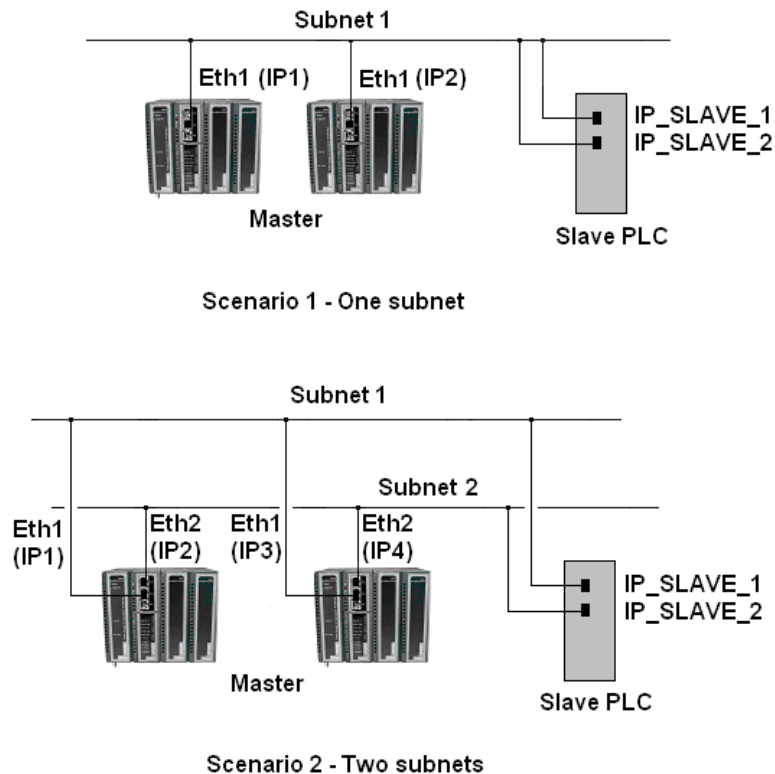
The redundancy in Modbus communication may be mapped to many possible scenarios. Some of them are supported by the controllers. The restrictions which should be considered are described below:

1. The Modbus TCP in the DF89 will be established in both Ethernet ports (ETH1 and ETH2). It is recommended to use both ports only in the scenarios where the DF89 is slave. When running as Modbus Master, the DF89 should not be configured to use ETH2 to avoid duplication in the bandwidth and it may damage the communication performance.
2. The both Ethernet ports (ETH1 and ETH2) are available to answer the Modbus requests when the DF89 is slave.
3. Switch over conditions are related to bad conditions in the Primary controller (see description of these bad conditions in another section of this manual "Adding Redundancy to DFI302 HSE Controllers"). As slave Modbus, the controllers will not generate a switch over condition because of communication failures. It is always the master who should switch over in such conditions. When using TCP and all IPs are in the same subnet, the master should decide when switch the request to other slave Ethernet connection.
4. As Modbus Master, DF89 (either TCP or Serial RTU) only switch over if the Primary does not have any answer from all slave modules.

Redundant DF89 as master and PLC, with two Ethernet ports, as slave, using Modbus TCP

There are two possible scenarios, in the first only one subnet is used and all IPs of equipment are on the same subnet.

In the second scenario, two subnets are used, and each device has an Ethernet port connected to one of them, see the following figure. The second scenario is the most recommended if the availability of the Modbus communication is an important factor, because in this case if there is some failure in one switch the another network/subnet will cover this failure.



**Figure 23.35 – Example of configuration with enabled and disabled devices
Network topologies for DF89 redundant master and slave PLC with two Ethernet ports**

The redundant PLC can either use two Ethernet cards or one card with two Ethernet ports.

Once the Primary DF89 executes the requests, it is assumed that any of the slave connections may receive and process the request. If one connection is not running properly, the primary DF89 will use another available connection (connection switching).

If the Primary DF89 did not receive response to any of the connections and the Secondary DF89 has response to at least one of the connections, the DF89 pair will switch over (scenario Bad Condition/ controller switching).

Redundant DF89 as master and redundant PLC as slave, using Modbus RTU

In this scenario, the DF89 is redundant, using RS-485 serial port, and it is connected via multidrop to redundant PLC (both ports). Once the active DF89 executes the requests (see R1 in the following figure), it is assumed that only one slave will process the request. In case of any "bad condition" the DF89 will switch over (see R2 in the following figure).

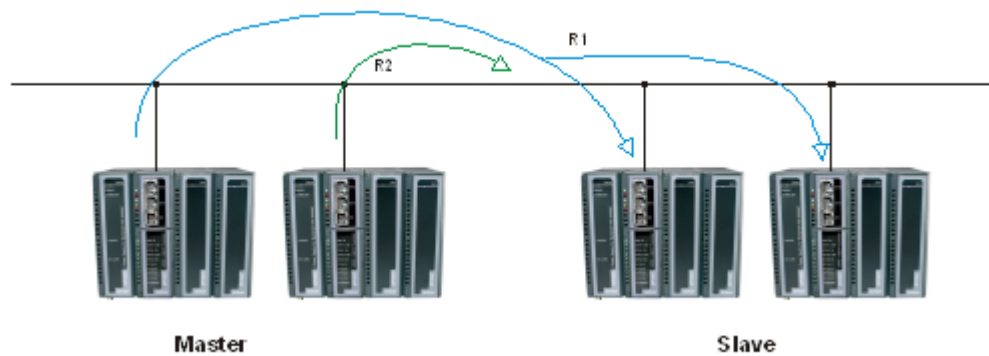


Figure 23. 36 – DF89 redundant master and PLC redundant slave, using Modbus RTU

Redundant DF89 as slave and redundant PLC as Master, using Modbus TCP

In this scenario, the Primary and the Secondary DF89 answer the reads requested by the Master PLC. The writes will be executed only by the Primary DF89. If for any reason the Secondary DF89 receives writing, so this request will be forwarded via redundancy path from Secondary DF89 to Primary DF89.

Redundant DF89 as slave and redundant PLC as Master, using Modbus RTU

In this scenario, only the Primary DF89 will answer the readings and writings requested by the Master PLC.

CREATING A FOUNDATION FIELDBUS STRATEGY BY USING THE DF100

Introduction

This section describes the fieldbus strategy configuration by using the DF100 controller. The figure below shows a ball mill (rotary) whose product temperature must be monitored.

PROJ_DF100

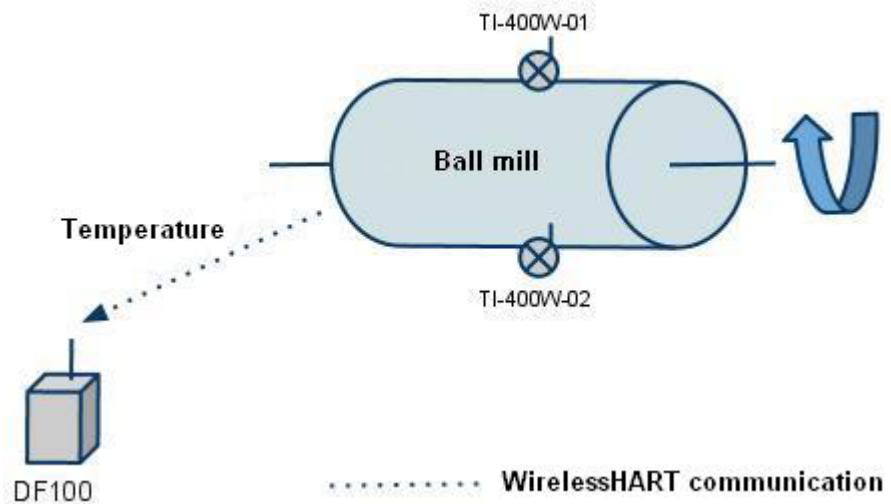


Figure 24. 1 - Example of temperature monitoring in a ball mill

The purpose is to monitor the product temperature at the center of the ball mill. The product temperature is measured by two wireless temperature transmitters installed in the hull of the mill, in its central part, in diametrically opposite positions. Measured temperatures will be transmitted from each of the temperature transmitters to the DF100 (*WirelessHART* Gateway) via *WirelessHART* communication. In the DF100, positioned near the ball mill and connected to the HSE network control, there will be special transducer blocks that will map the temperatures received from the transmitters. In addition to the transducers, the DF100 has function blocks that are used to calculate the average temperature of the mill from the two measured temperatures.

Starting an Area

Step 1

It is possible to create, or edit, an area from the **Studio302**. In the **Studio302** interface select **Areas**. A window will appear listing all areas of database.

To create a new area from the **Studio302**, right-click inside the **Areas** window, then choose **New Area**.

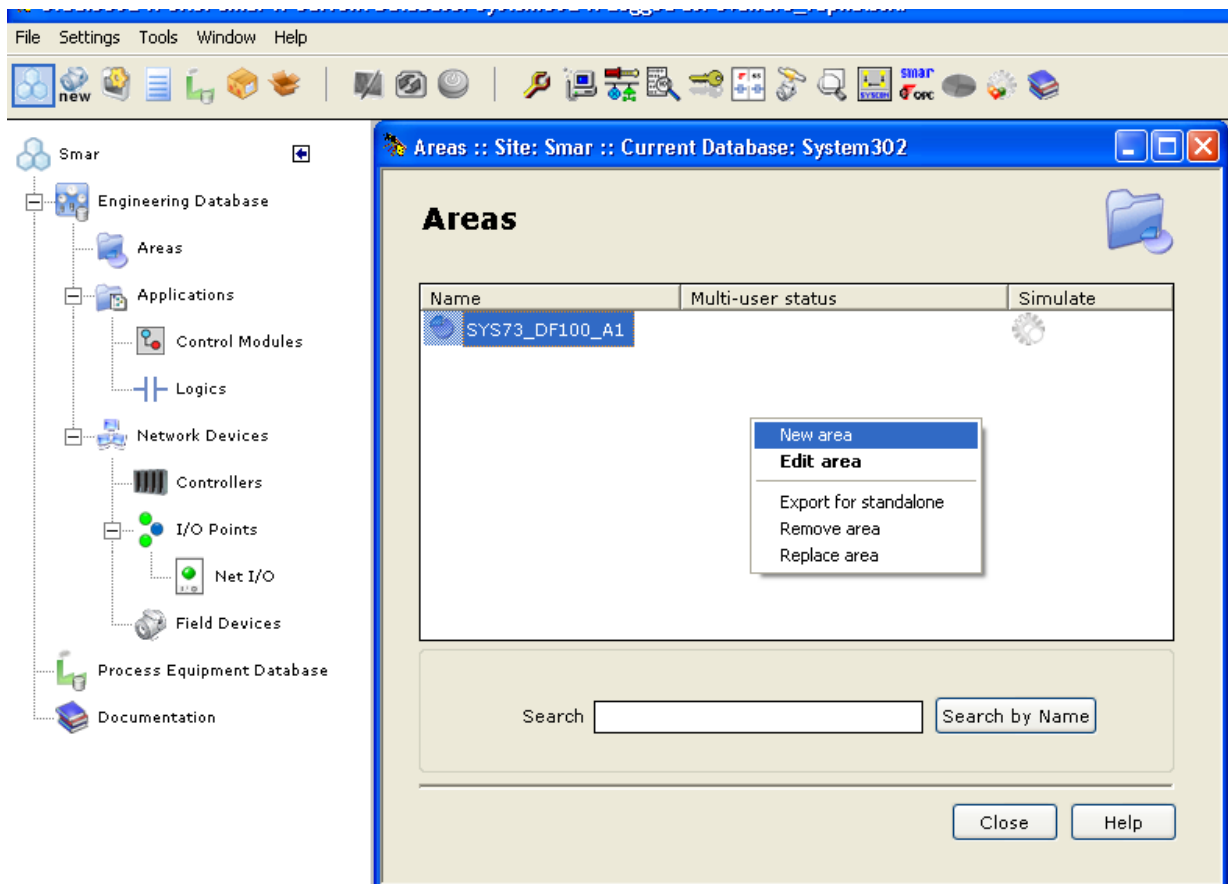



Figure 24. 2 - Creating a new area

Another way to create a new area is from **Syscon**. Click the icon  in the **Studio302** toolbar.

To create a new area on **Syscon**, choose **File → New**, or through the toolbar, choose **New** button . The dialog box shows the areas options. Select **HSE Area** as shown in next figure:

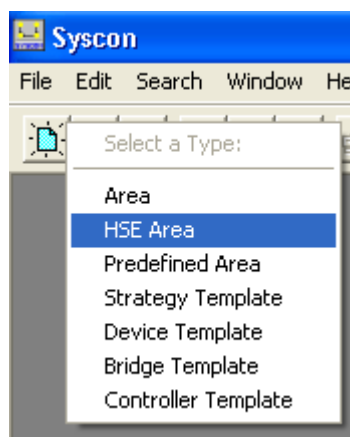


Figure 24. 3 - Options to create Syscon areas

After choosing the area type, it opens a window to the user give a name to the new area.

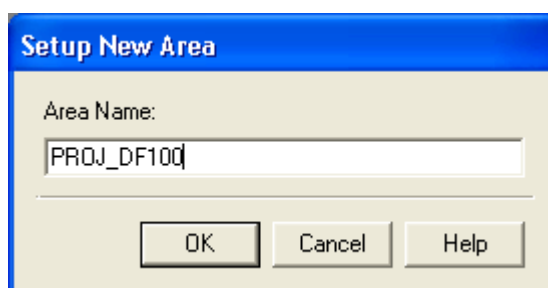


Figure 24. 4 - New area name

Type the name for the area in the **Area Name** box, and click **Ok**. For this example, it chooses **PROJ_DF100** name.

A new window will appear. This window has:

- **Application** – Logical Plant. To insert control and/or monitoring strategies into this part.
- **Fieldbus Networks** – Physical Plant. To add devices and blocks (resources, transducers and function blocks) to the area into this part.

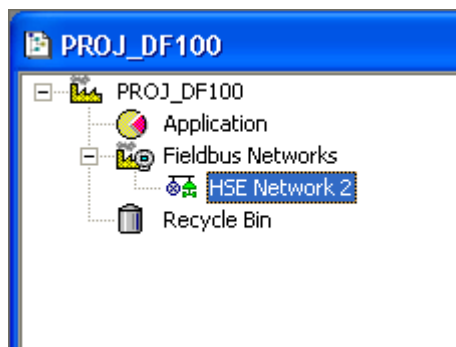



Figure 24. 5 - Area divisions

Physical Plant Project

Step 2

In the main window, **PROJ_DF100**, right-click the **Fieldbus Networks** icon, , and choose **Communication Settings** option, or through the toolbar, choose **Communication→Settings**. The communication settings dialog box will open.

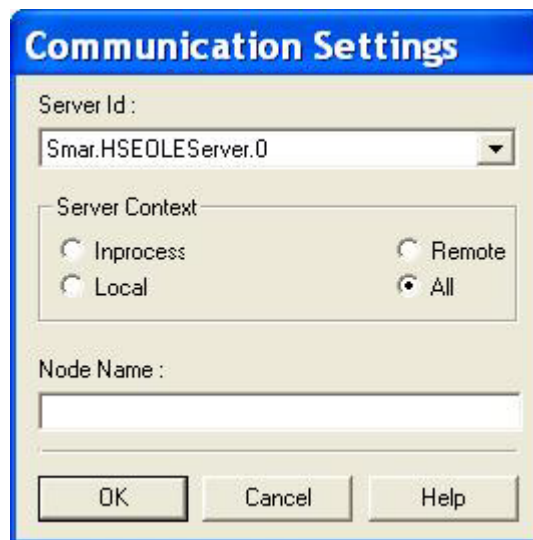



Figure 24. 6 - Choosing the Server

Confirm if the **Smar.HSEOLEServer.0** option has already been selected. Otherwise, the user must select it, and then click **OK**.

Arranging the Fieldbus windows

Step 3

After selecting the Server for the area, click the  sign placed at left of the **New Fieldbus**. The HSE network will appear with a tag, for example, **HSE Network 1***. Right-click this item and choose **Expand** option. The figure below shows the HSE network:

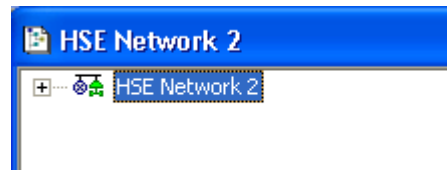


Figure 24. 7 - Creating an HSE network

To arrange the screen, click the area window. So, choose **Window** menu on the **Syscon** toolbar, and then **Tile** option.

Adding the Controller

Step 4

Right-clicking the **HSE Network 2**, it opens a dialog box to add new devices. Choosing **New** option, it is possible to select devices such as **Bridges**, **Controllers** and **Devices** for the area. For the aimed strategy, choose **Controller** option. Confirm this choice observing the following figure.

* This number changes if another area was created before. When a new HSE area is created, this number increases.

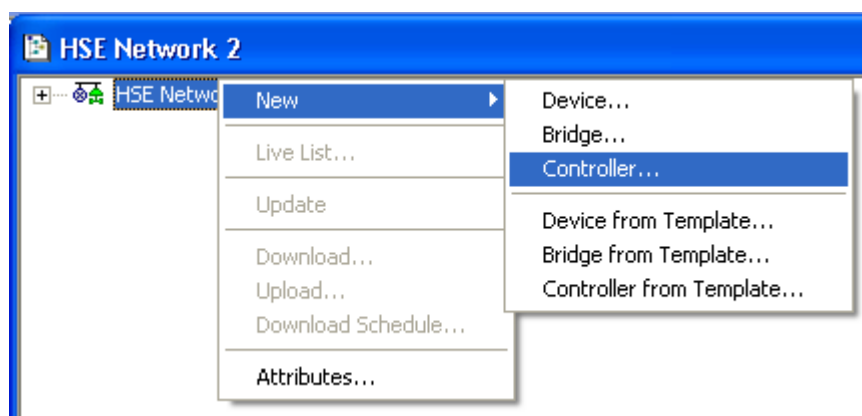


Figure 24. 8 - Choosing the Controller

After selecting that option, it opens a window as shown in the next figure.

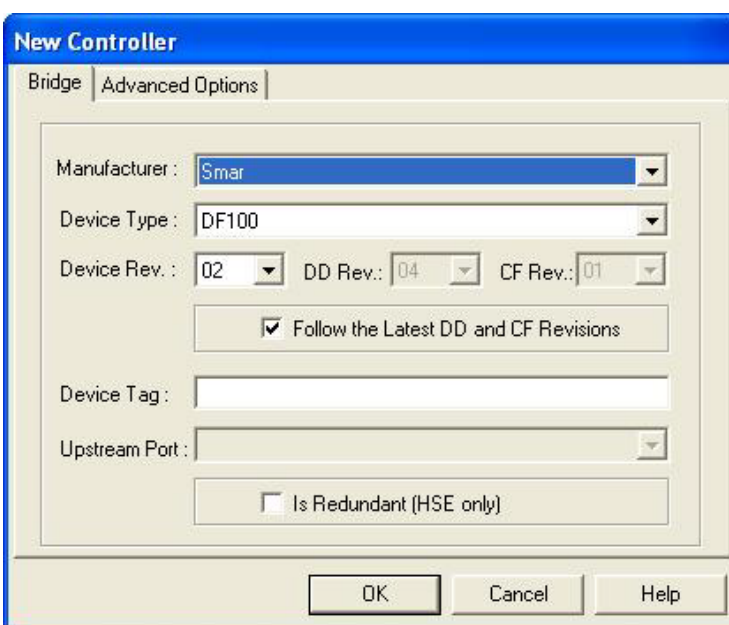


Figure 24. 9 - Setting the Controller

Select the **DF100** device in the **Device Type** box. In the **Device Tag** box, enter **DF100** or another tag, and click **OK**.

IMPORTANT
Not all characters are valid when naming the elements, so pay attention; The valid characters are: A-Z a-z 0-9 # { } [] () + -
The invalid characters are: ~ ` ! @ # \$ % ^ & * = : ; , . < > ? / ' " \

At this moment, the following blocks will be created in the configuration:

- Resource block;
- Diagnostic transducer block;
- Transducer block for HART Gateway (Transducer Block for HART Gateway - TBHG);
- One transducer block for the first HART or *WirelessHART* field device (Transducer Block for *WirelessHART* - TBWH).

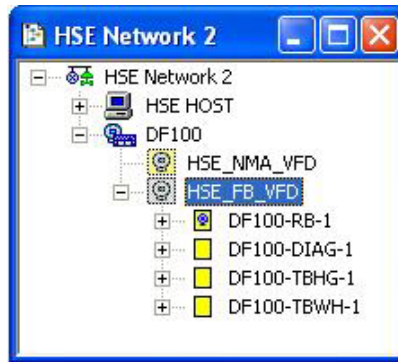


Figure 24. 10 – Blocks created after adding the controller

Quite simply, we understand that the DF100 (*WirelessHART* Gateway) and the *WirelessHART* and HART¹ field devices will be mapped on SYSTEM302 by the transducer blocks. This is possible due to the HSE WIO technology, embedded in the DF100.

The DF100 will be mapped by the transducer block (single and mandatory) called Transducer Block for HART Gateway (TBHG). Furthermore, each HART and *WirelessHART* field device will be mapped by transducer blocks called Transducer Blocks for *WirelessHART* (TBWH). There will be as many TBWH blocks as there are HART and *WirelessHART* field devices needed in the configuration.

Adding HSE WIO Transducer and Function Blocks

Step 5

The first HSE WIO transducer block, **TBHG**, is automatically created when the DF100 controller was added to the Physical Plant. Make sure that its **MODE_BLK** parameter is configured to Auto (Automatic) before downloading the configuration on the DF100. This block, as monitored in Online mode, shows the list of HART and/or *WirelessHART* field devices currently connected to the DF100. Thus, we can understand that the **TBHG** provides to the system and to the user the Live List of field devices connected to the DF100 via *WirelessHART* network. See Library B Function Blocks Instruction Manual for additional information on the **TBHG** block.

The next step is to associate the **TBWH** block to the first wireless temperature transmitter of this example. This **TBWH** was also automatically created (with the **DF100-TBWH-1** tag) when the DF100 controller was added. The transmitter Long Tag has to be attributed (suppose **TI-400W-01**²) to the **Block Tag** of **TBWH** block. The following figures show the association between **TBWH** and the transmitter.

¹ The DF100 integrates HART, 4 to 20 mA conventional field devices, via *WirelessHART* adapters.

² In online mode, the Long Tags of transmitters connected to DF100 can be found in the **LIVE_LIST_TAG_x** parameters of TBHG block.

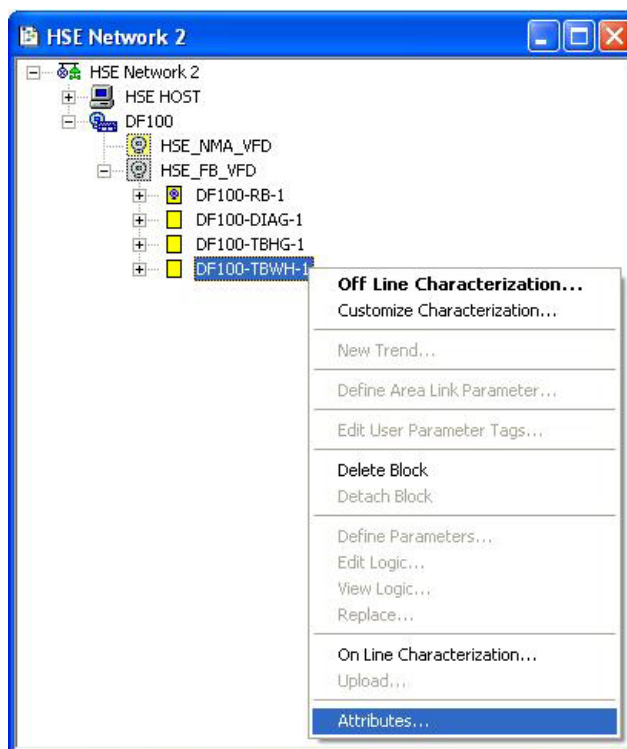


Figure 24. 11 – Changing the attributes of TBWH block

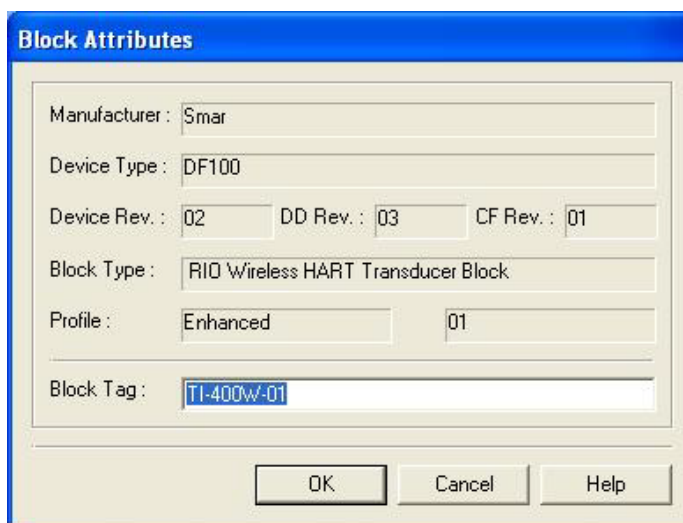



Figure 24. 12 – Changing the attributes of TBWH block

Depending on the strategy, you can add as many **TBWH** blocks as there are HART and *Wireless*HART field devices supported by the DF100 (see section Technical Specifications). For the purposed example, just add another **TBWH** block, which will map the second temperature transmitter. Its Long Tag will be **TI-400W-02**.

To add a new Function Block (FB), click the  sign, near to the DF100 controller, and right-click the **Virtual Field Device** (HSE_FB_VFD) icon. Choose **New Block** option.

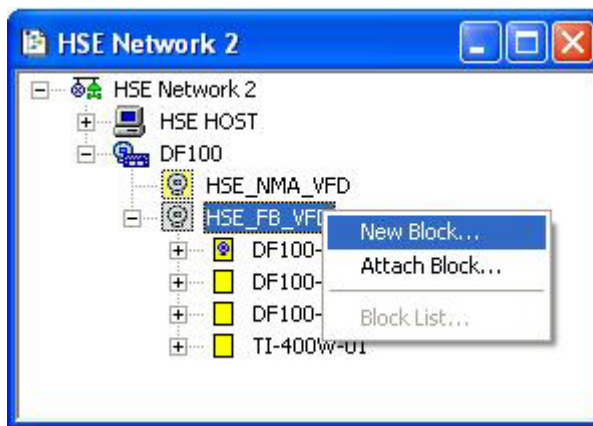


Figure 24. 13 – Adding New Blocks

The **New Block** dialog box will appear. In the **Block Type** option the blocks designed to the controller can be selected.

Select the block in the **Block Type** box, and name it in the **Block Tag** box. Click **OK**. The following figure shows adding the second **TBWH** block that will map the transmitter whose Long Tag is **TI-400W-02**

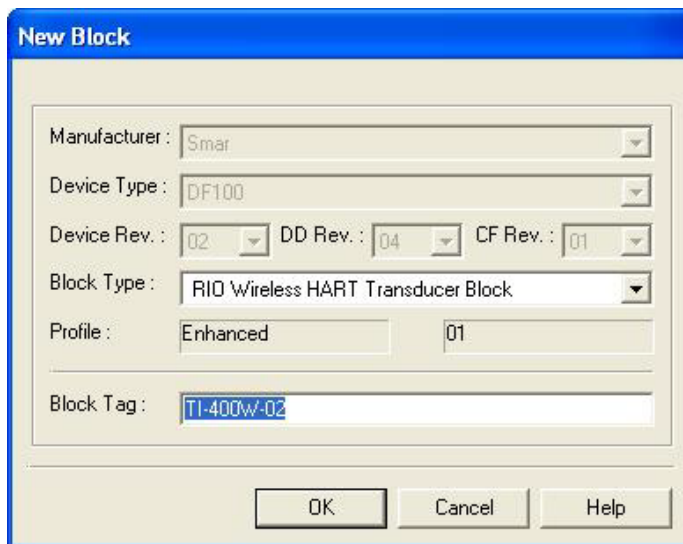


Figure 24. 14 – Adding TBWH block

As a general rule, for each **TBWH** block added, it is recommended:

- Assign the Long Tag of the HART or *WirelessHART* field device to the block tag attribute of **TBWH**.
- Configure the **MODE_BLK** parameter to Auto (Automatic) before downloading the configuration on the DF100.

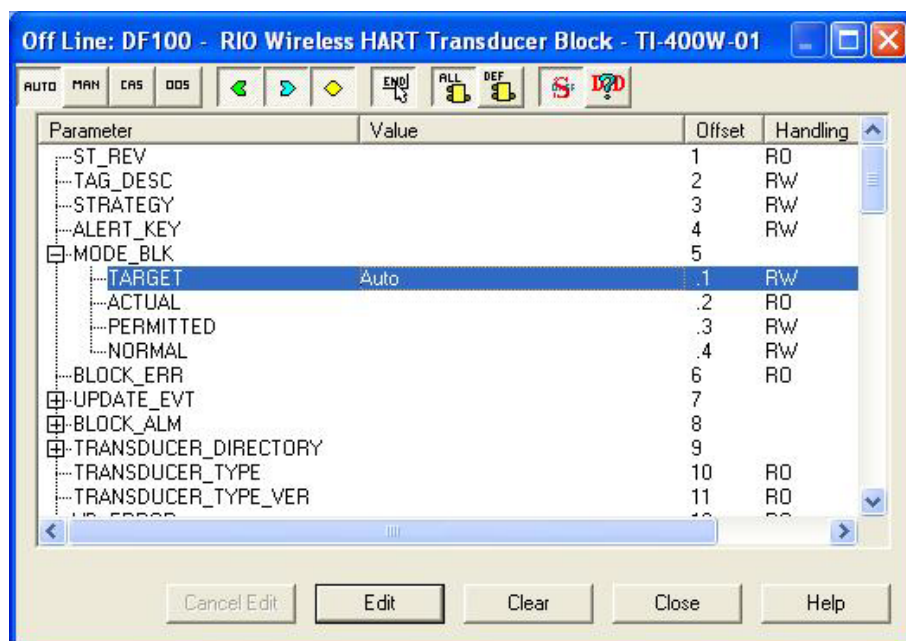


Figure 24. 15 – Configuring the MODE_BLK parameter

- Configure the **HART_EXPD_DEV_INFO** parameter with the attributes of the field device that will be mapped by **TBWH**. Refer to Library B Function Blocks Instruction Manual as well as the HART or *WirelessHART* field device manual³ to configure correctly this parameter.

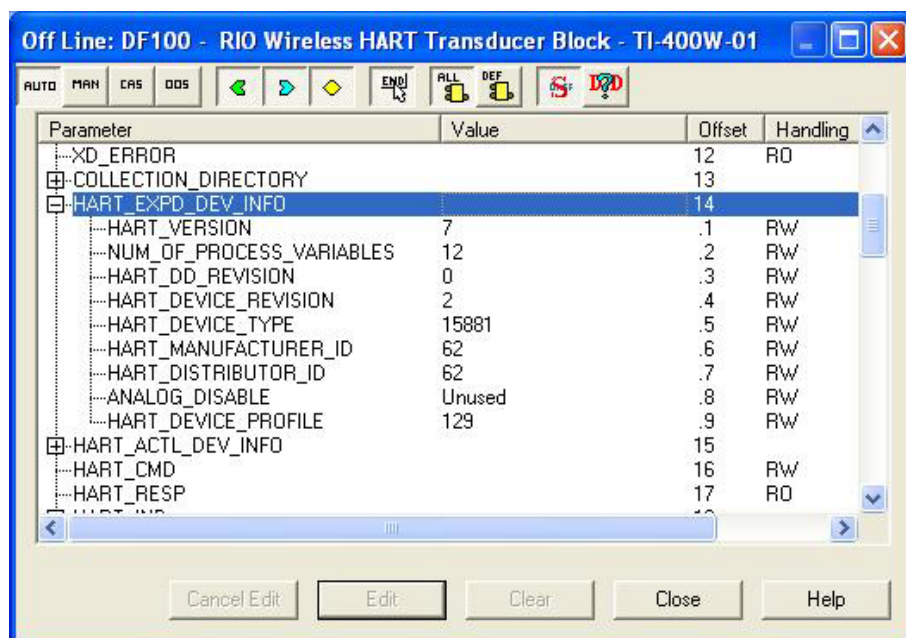


Figure 24. 16 – Configuring the HART_EXPD_DEV_INFO parameter

³ In online mode, if the desired field device is connected to DF100 and commissioned, it is recommended to configure the attributes of **HART_EXPD_DEV_INFO** parameter with the values of the respective attributes of **HART_ACTL_DEV_INFO** parameter.

NOTE - HART_EXPD_DEV INFO

The above parameterized attributes are specific to the considered field device - in this example the TT400 Smar WirelessHART was used. Therefore, important to note that the values assigned to the attributes of **HART_EXPD_DEV_INFO** parameter depend on the type of device (pressure, temperature, density, etc.) and its firmware version.

The table below summarizes the attributes values of the **HART_EXPD_DEV_INFO** parameter for some Smar WirelessHART field devices:

Attribute	LD400WH Firmware 3.0.1	DT400WH Firmware 1.0.0	TT400WH Firmware 2.0.1	TT481WH Firmware 1.0.0
HART_VERSION	7	7	7	7
NUM_OF_PROCESS_VARIABLES	11	4	12	18
HART_DD_REVISION	1	1	1	1
HART_DEVICE_REVISION	3	1	2	1
HART_DEVICE_TYPE	15882	15888	15881	15890
HART_MANUFACTURER_ID	62	62	62	62
HART_DISTRIBUTOR_ID	62	62	62	62
ANALOG_DISABLE	Unused	Unused	Unused	Unused
HART_DEVICE_PROFILE	129	129	129	129

- Knowing that the second temperature transmitter used in this example, whose tag is **TI- 400W-02**, it is also a TT400 Smar WirelessHART with identical firmware to the **TI-400W-01** transmitter, configure the **HART_EXPD_DEV_INFO** parameter exactly as was done for the TBWH of **TI- 01-400W**.
- Configure the parameters **HART_VAR_CODES8** and **VAR_NAMES9**.
- The codes for **HART_VAR_CODES8** can be found in the “Note - HART_VAR_CODES8” field.
- The names used in each **VAR_NAMES9** will uniquely identify (in the scope of the DF100 that contains TBWH) the variables **HART_PV**, **HART_SV**, **HART_TV**, **HART_QV**, **HART_5V**, **HART_6V**, **HART_7V**, **HART_8V** AND **ANALOG_VALUE**, respectively. This procedure is necessary, as the input HSE WIO function blocks will use these names to address the values of the **HART** variables that must be configured in the parameter. The figure below shows the parameterization of **VAR_NAME** that will identify the **HART_PV** variable of the TI-400W-01 transmitter. It is important to note that only one **VAR_NAME** has been parameterized and that, depending on the strategy adopted, not all **VAR_NAME** will need to be parameterized. See the Library B Function Block Instruction Manual for additional information on the **TBWH** transducer block, its parameters and required parameterizations.

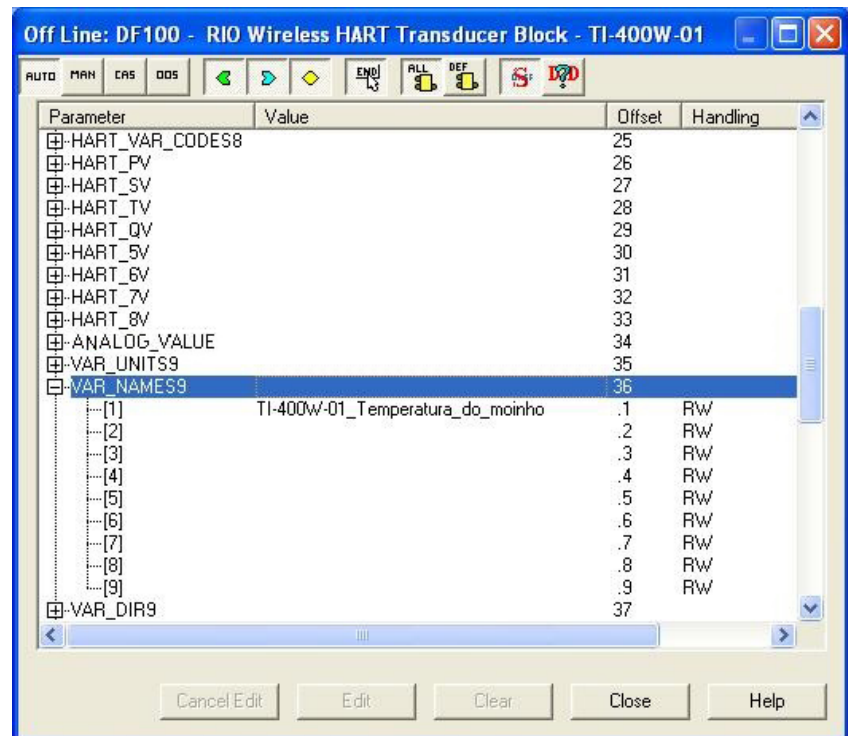


Figure 24. 17 – Configuring the VAR_NAMES9 parameter of TBWH tagged TI-400W-01

Similarly for the TBWH block tagged as TI-400W-02, the VAR_NAMES9 parameter will be configured with the TI-400W-02_Temperatura_do_moinho value.

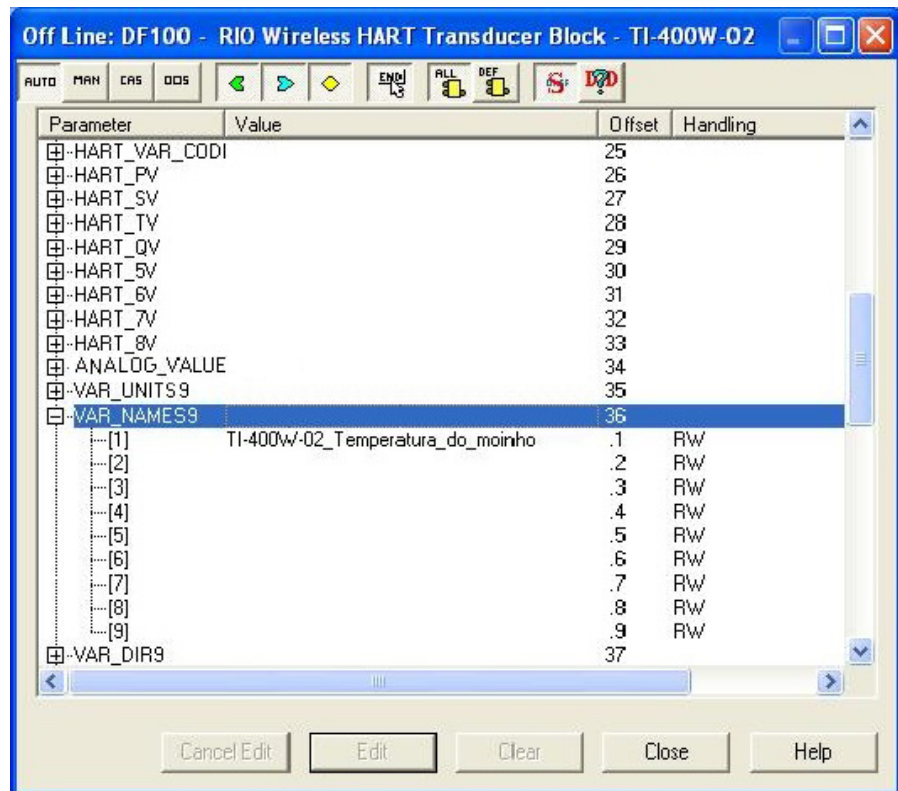


Figure 24. 18 – Configuring the VAR_NAMES9 parameter of TBWH tagged TI-400W-02

NOTE - HART_VAR_CODES

HART_VAR_CODES8 parameters for some Smar WirelessHart equipment. For other equipment not documented here, please consult the manufacturer.

LD400WH

Variable		Description	Units
Number	Mnemonic		
0	OUT	Output Variable related to analog output	Milliamperes (fixed)
1	OUT%	Ranged Output Variable related to analog output	Percentage (fixed)
2	PRES	Primary Variable related to measured pressure	PV Units (configurable)
3	PV%	Ranged Process Variable	Percentage (fixed)
4	PV	Process Variable in engineering units	PV Units or User Units (configurable)
5	TEMP	Secondary Variable related to sensor temperature	Degrees Celsius (fixed)
6	TOT	Totalization	Total units
7	SP%	Ranged Setpoint	Percentage (fixed)
8	SP	Setpoint in engineering units	PV Units or User Units (configurable)
9	ERR%	Deviation (PV% - SP%)	Percentage
10	BAT	Battery Voltage	Volts (fixed)

TT400WH

Variable Number	Mnemonic	Description	Units
0	OUT	Output Variable related to analog output	Milliampere (fixed)
1	PV%	Process Variable in percentage units	Percent (fixed)
2	PV	Process Variable in engineering units	PV Unit
3	TEMP	Environment temperature	PV Unit
4	BACK	Backup Sensor Value	PV Unit
5	-	Not Used	-
6	-	Not Used	-
7	ADTMP	Environment Temperature ADC value	None
8	AD23	Channel 23 ADC value	None
9	AD34	Channel 34 ADC value	None
10	AD24	Channel 24 ADC value	None
11	BAT	Battery Voltage	Volt (fixed)

DT400WH

Variable		Description	Units
Number	Mnemonic		
0	PV	Primary Variable related to measured density	PV Units (configurable)
1	PV%	Ranged Process Variable	Percentage (fixed)
2	SV	Secondary Variable related to sensor temperature	SV Units (configurable)
3	BAT	Battery Voltage	Volts (fixed)

TT481WH

Variable Number	Mnemonic	Description	Units
0	SNS1	Sensor 1 Process Variable	Sensor PV Unit
1	SNS2	Sensor 2 Process Variable	Sensor PV Unit
2	SNS3	Sensor 3 Process Variable	Sensor PV Unit
3	SNS4	Sensor 4 Process Variable	Sensor PV Unit
4	SNS5	Sensor 5 Process Variable	Sensor PV Unit
5	SNS6	Sensor 6 Process Variable	Sensor PV Unit
6	SNS7	Sensor 7 Process Variable	Sensor PV Unit
7	SNS8	Sensor 8 Process Variable	Sensor PV Unit
8	ENV	Environment temperature	°C (fixed)
9	BAT	Battery Voltage	Volt (fixed)
10	PERC1	Sensor 1 Process Variable Percentage	%
11	PERC2	Sensor 2 Process Variable Percentage	%
12	PERC3	Sensor 3 Process Variable Percentage	%
13	PERC4	Sensor 4 Process Variable Percentage	%
14	PERC5	Sensor 5 Process Variable Percentage	%
15	PERC6	Sensor 6 Process Variable Percentage	%
16	PERC7	Sensor 7 Process Variable Percentage	%
17	PERC8	Sensor 8 Process Variable Percentage	%

- Configure the **LOCAL_MOD_MAP** parameter with the value that the **TBWH** block should have to be used with Modbus. The first **TBWH** has to be configured with **LOCAL_MOD_MAP** equals to 0, the second one with 1 and so on up to 99. Refer to Modbus Protocol Support section for further information about Modbus mapping and addressing.

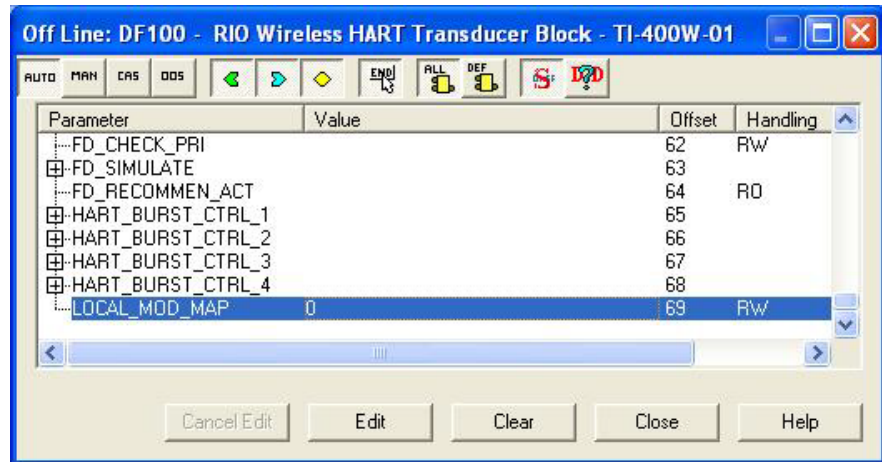


Figure 24. 19 – Configuring the LOCAL_MOD_MAP parameter

From now on, we will add to the configuration HSE WIO input function blocks. In the DF100 the **WIO Analog Input (WAI)** and **Multiple Analog Input 16 (MAI16)** blocks are available. These blocks are variations of the HSE WIO for conventional **Analog Input (AI)** and **Multiple Analog Input (MAI)** blocks, respectively. Despite having similar algorithms, the HSE WIO input function blocks have, besides data types specifically designed for the HSE WIO technology, parameters named **CHANNEL_TAG**. A HSE WIO input function block (such as **WAI** or **MAI16**) uses the **CHANNEL_TAG** parameter instead of **CHANNEL** parameter to correctly address a HART variable, from a **TBWH** transducer block. Knowing that, besides the already known parameterization steps for the conventional input function blocks, it is necessary to parameterize the **CHANNEL_TAG** parameters with HART variable names. In other words, the names parameterized for the **CHANNEL_TAG** can be any names among those parameters parameterized in the **VAR_NAMES9** parameters of the **TBWH** transducer blocks. Refer to the Function Blocks Manual for additional information on the HSE WIO function blocks, its parameters and parameterization.

For the configuration of this example, add two **WAI** blocks. The addition of the **WAI** blocks is similar to addition of **TBWH** block just described above. Each **WAI** block will address a wireless temperature transmitter (and its data) via **TBWH** transducer block. For each **WAI** of the purposed example, the **CHANNEL_TAG** is configured with the **VAR_NAME** assigned to the **HART_PV** variable (see **TBWH** block parameterization above) of the addressed wireless temperature transmitter. The two following figures demonstrate the **CHANNEL_TAG** configuration of the **WAI** blocks that will address the temperature transmitters **TI-400W-01** and **TI-400W-02**, respectively.

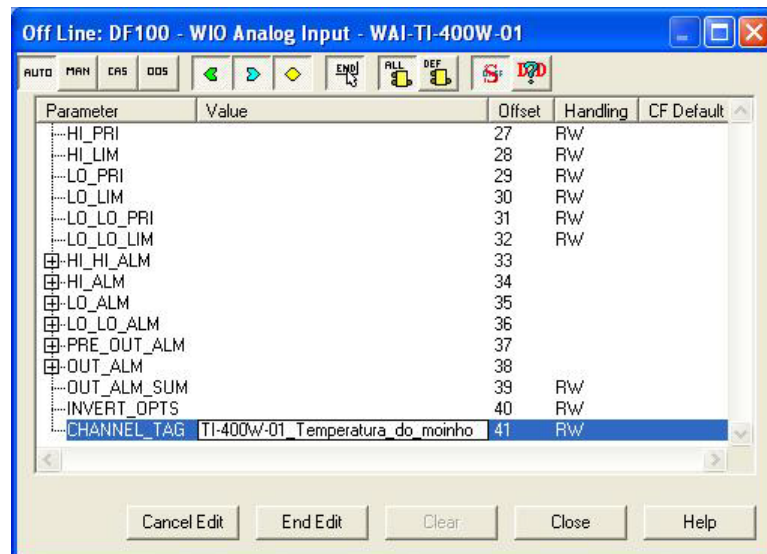


Figure 24. 20 – Configuring the **CHANNEL_TAG** that addresses the **HART_PV** of **TI-400W-01**

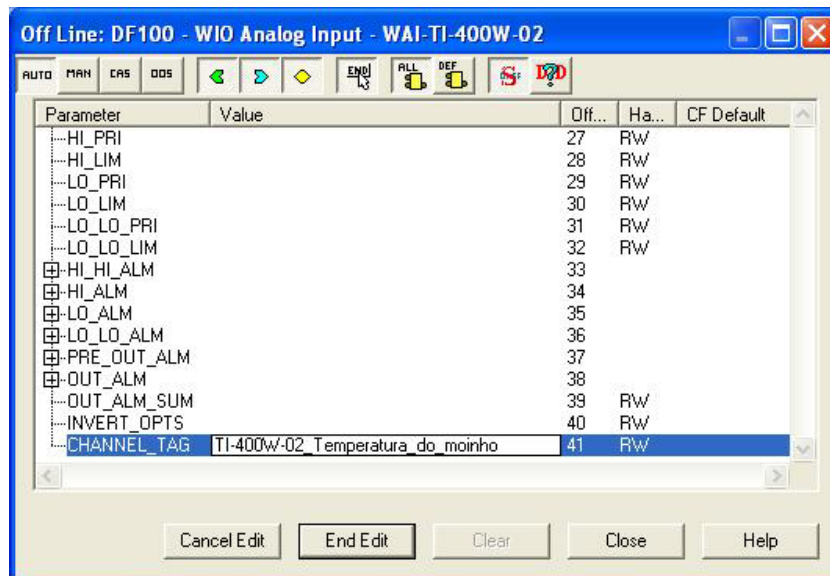


Figure 24. 21 – Configuring the CHANNEL_TAG that addresses the HART_PV of TI-400W-02

With the addition of the **WAI** blocks we have finished adding the HSE WIO transducers and function blocks. See the figure below for having a general idea of HSE WIO blocks configured so far.

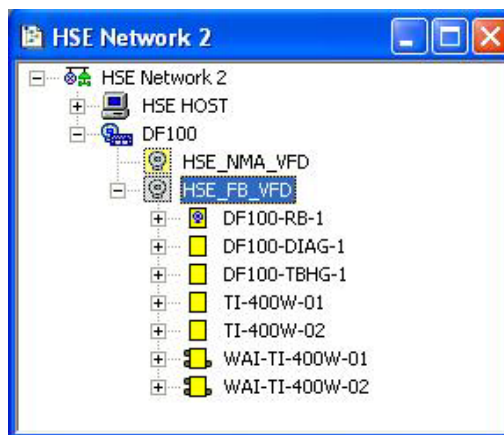


Figure 24. 22 – HSE WIO transducers and function blocks

Adding Conventional Function Blocks

Step 6

The user can also add others conventional function blocks, i.e., non-specific HSE WIO ones. For that, in the **New Block** dialog box, select the desired function block in the **Block Type** box, and then, in the **Block Tag** box, give a tag to the block. Click **OK**. The next figure shows the addition of the **ARTH** function block (AR Block).

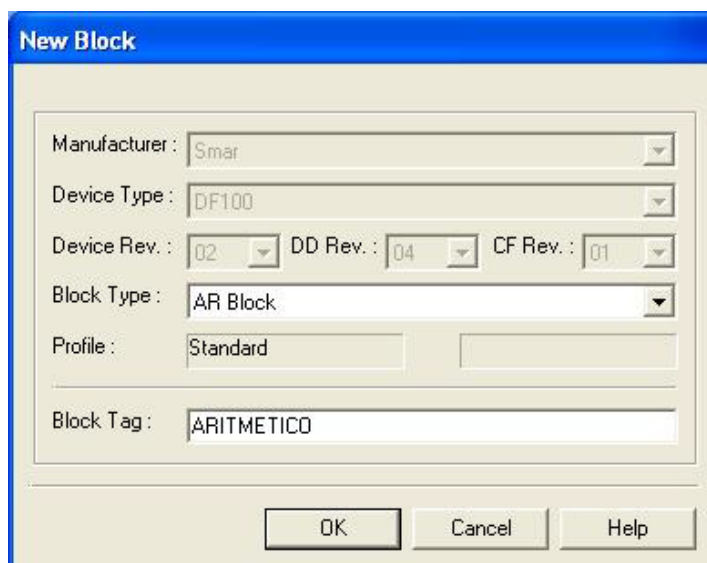


Figure 24. 23 –Adding a conventional function block

For this example, the **ARTH** block complements the strategy and it will be used in the DF100 to calculate the average temperature in the ball mill.

NOTE

Only transducer and function blocks were added in the DF100 (*WirelessHART* Gateway). The HART and *WirelessHART* field devices not support function blocks. They are only mapped in the DF100 through **TBWH** transducer blocks.

The configuration with all the function and transducers blocks, conventional or not, is presented in the following figure.

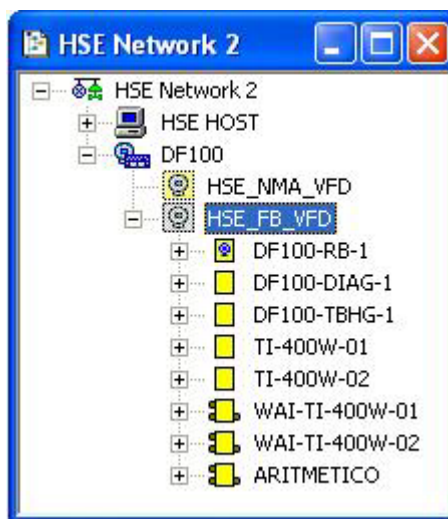


Figure 24. 24 – Complete wireless configuration

Now, the strategy area can be developed (**Application, Logic Plant**). First is necessary to create a new Process Cell.

Creating New Process Cells

Step 7

The **Logical Plant** can be divided in several process cells, according to the plant.

To create a new process cell, right-click the **Application** icon, and select **New Process Cell** item.

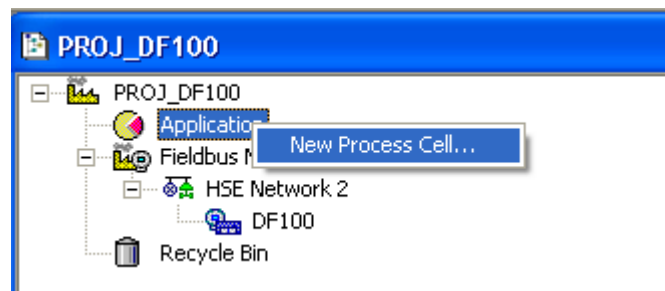


Figure 24. 25 – Inserting a Process Cell

The dialog box to name the **Process Cell** will open:

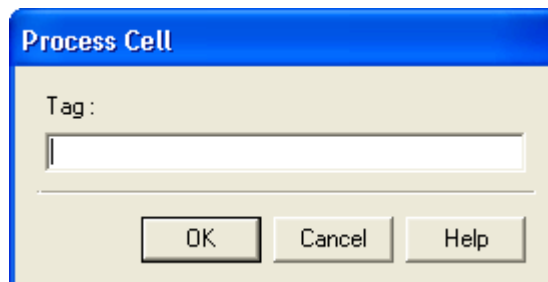


Figure 24. 26 – Attributing tag to the Process Cell

If the user needs name the Process Cell with a specific tag. Enter it in the **Tag** box, and click **OK**. To create more process cells, the procedure above can be repeated.

After inserting the Process Cell, the **PROJ_DF100** window will be according to the next figure:

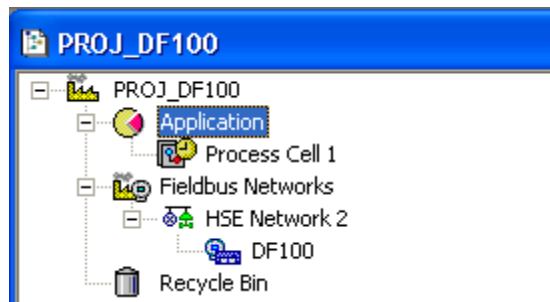


Figure 24. 27 – Area window after inserting the Process Cell

NOTE

The user must remember that **Application** is a virtual division. It only divides a large plant. For example: if the plant has two networks, they can be **Process Cells** in the **Syscon**. One **Application** can have several **Process Cells**, but a **Process Cell** cannot be in more than one **Application**.

Creating a Control Module

Step 8

Now the user can create a Control Module in the Application section.

Right-click the **Process Cell1** icon, and choose **Expand** item.

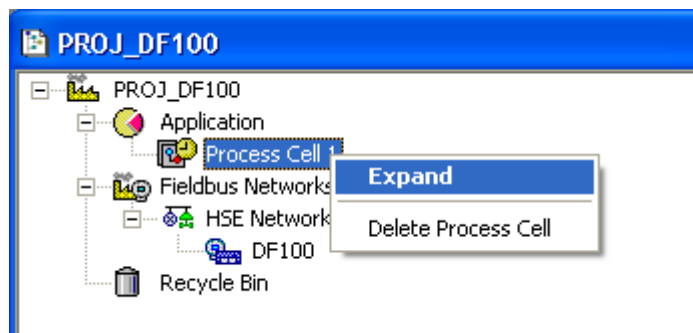


Figure 24.28 – Creating the Control Module

To arrange the screen, click the **Process Cell 1** window. So, choose **Window** menu on the **Syscon** toolbar, and then **Tile** option.

As following, return to the **Process Cell1** window. Right-click the **Process Cell1** item, and choose **New Control Module**. The figure below shows creating the New Control Module.

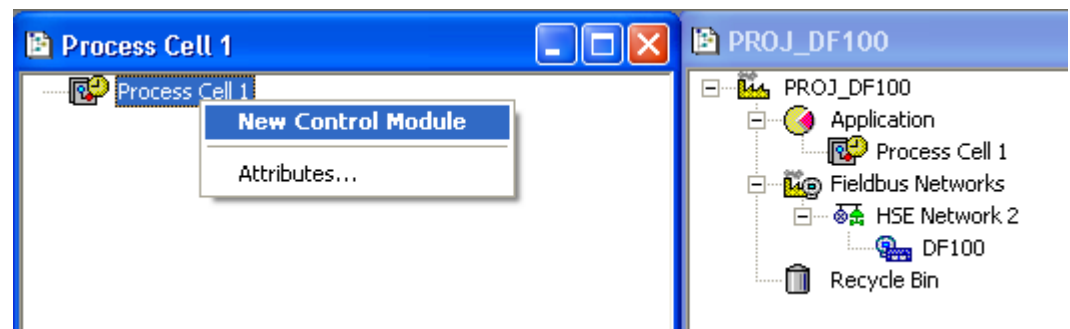


Figure 24.29 – Creating the New Control Module

The **New Control Module** dialog box will appear. Name it with the tag related to the application. To continue, click **OK**.

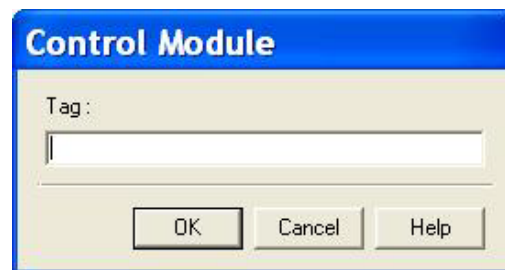


Figure 24.30 – Attributing tag to the Control Module

IMPORTANT

Remember that not all characters are valid when naming the elements with tags.

Inserting Blocks in the Control Module

Step 9

Now the user can add function blocks which will participate in the strategy of the mill temperature monitoring in the **Logical Plant**. Right-click the **Control Module 2** item, and choose **Attach Block** option, as shown in the next figure:

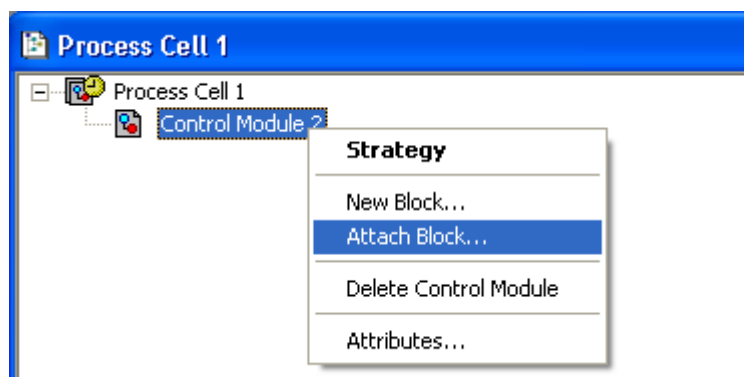


Figure 24. 31 –Adding new functions blocks to the FBAP

The **Attach Block** dialog box will open as shown below:

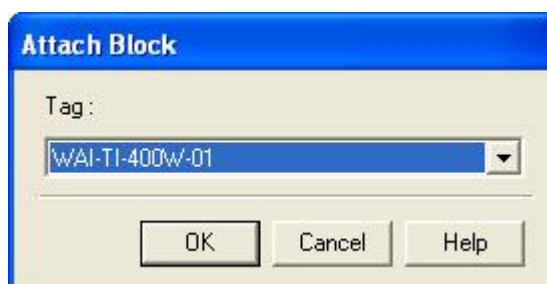


Figure 24. 32 –Attaching blocks to the Control Module

The available function blocks for the application are showed in the **Attach Block** box. For the aimed strategy, the function blocks that must be inserted will appear in the box. So, select them one by one, and click the **OK** button.

When the **Attach Block** process ends, the application will be as shown in the following figure:

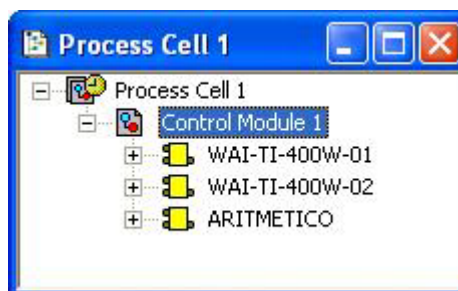


Figure 24. 33 –Blocks added to the Control Module

Another way to attach the blocks is left-clicking the element and drop it to the window.

A new tag can be done to the **Control Module** right-clicking it, and selecting **Attributes**. For the aimed example will be attributed **Temperatura Moinho de Bolas**

Configuring the Control Strategy

Step 10

Now the user is ready to develop the control strategy.

First, right-click the **Temperatura Moinho de Bolas** icon, and select **Strategy**. The **Strategy** window will appear as shown in the following figure.

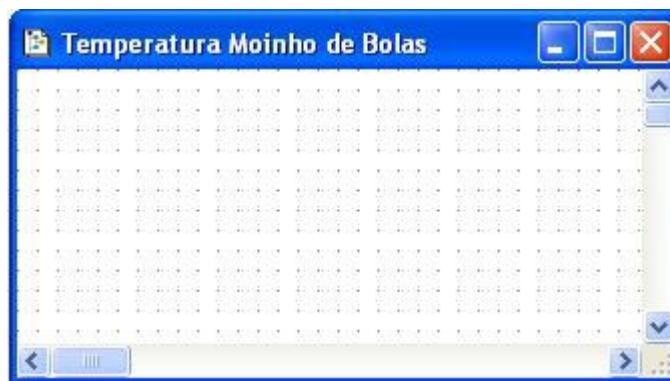


Figure 24. 34 –Strategy window

It is recommended to minimize the strategy window. Thus, it is possible to see the whole area.

The strategy window offers several tools for drawing. Refer to the **Syscon** manual for further details.

Adding Blocks to the Strategy window

Step 11

Now the function blocks can be added to the **Temperatura Moinho de Bolas** window.

In order to get this, click the first block, **WAI-TI-400W-01**, and drop it into the strategy window. A function block will be automatically created.

The next figure shows the function block added to the strategy window:

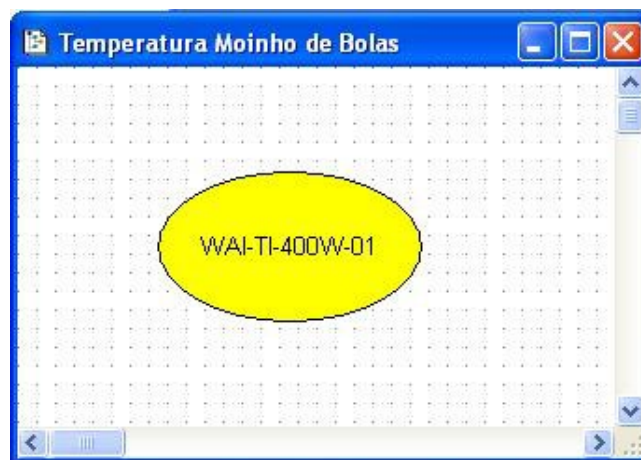


Figure 24. 35 –Block inserted into the strategy window

Repeat the drag-and-drop procedure for the other blocks **WAI-TI-400W-02** and **ARITMETICO**.

Linking the Blocks

Step 12

There is a specific tool to link the blocks, the **Link** button, , on the **Strategy** toolbar.

Click this button on the toolbar, and then in the **WAI-TI-400W-01** function block. The dialog box for linking the input and output parameters will appear. Select **OUT**, and then click the **OK** button as shown in the following figure.

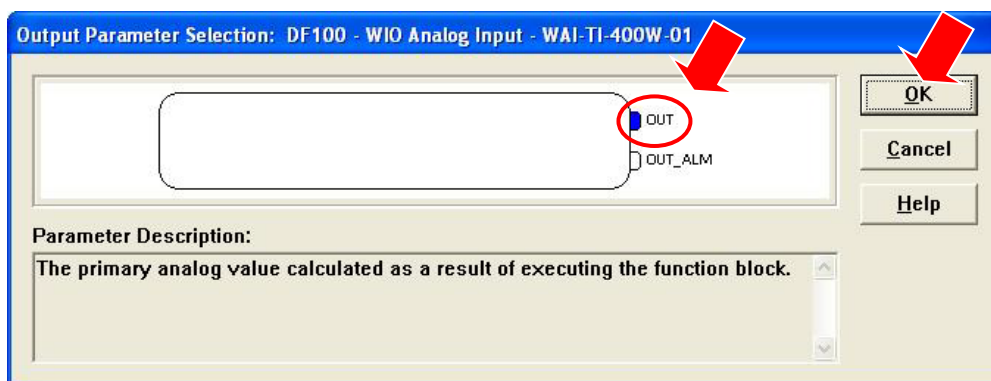


Figure 24. 36 –Linking the Function Blocks

Move the mouse cursor up to the block that will be linked.

The user also does the fast link procedure just right-clicking the function block. The links necessary for this strategy are:

Direct Links:

- OUT(WAI-TI-400W-01) → IN_1(ARITMETICO)
- OUT(WAI-TI-400W-02) → IN_2(ARITMETICO)

After linking the parameters specified above, the strategy window will be as shown in the following figure:

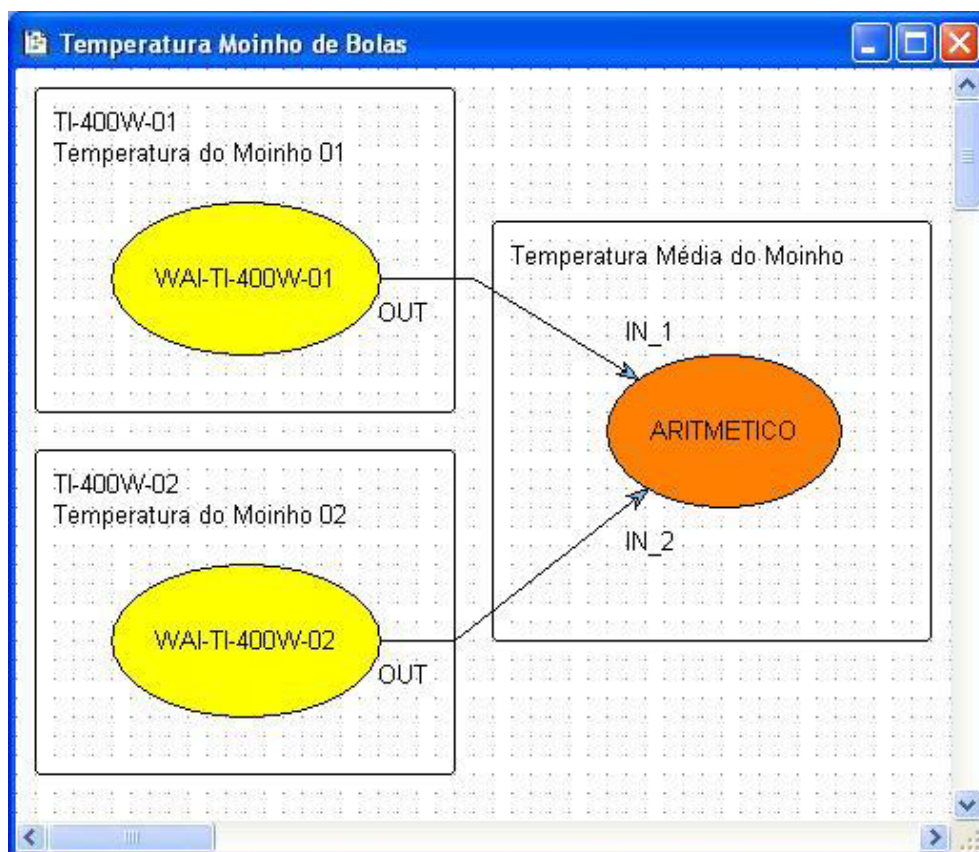


Figure 24. 37 –Links between the Function Blocks

Function Block Characterization

Step 13

The function blocks that are in the strategy must be set according to the application for them. So, it is necessary to do the block characterization.

The online and offline modes are possible for the block characterization. In the offline mode, the parameters are set before starting the communication between the devices. The online characterization is executed directly in the devices when the plant is already communicating.

To change the function block parameters, consider the following steps:

Select the block to characterize. Right-click it, and select the **Off Line Characterization** option, or double-click it. The following figure shows the block that is being done the offline characterization:

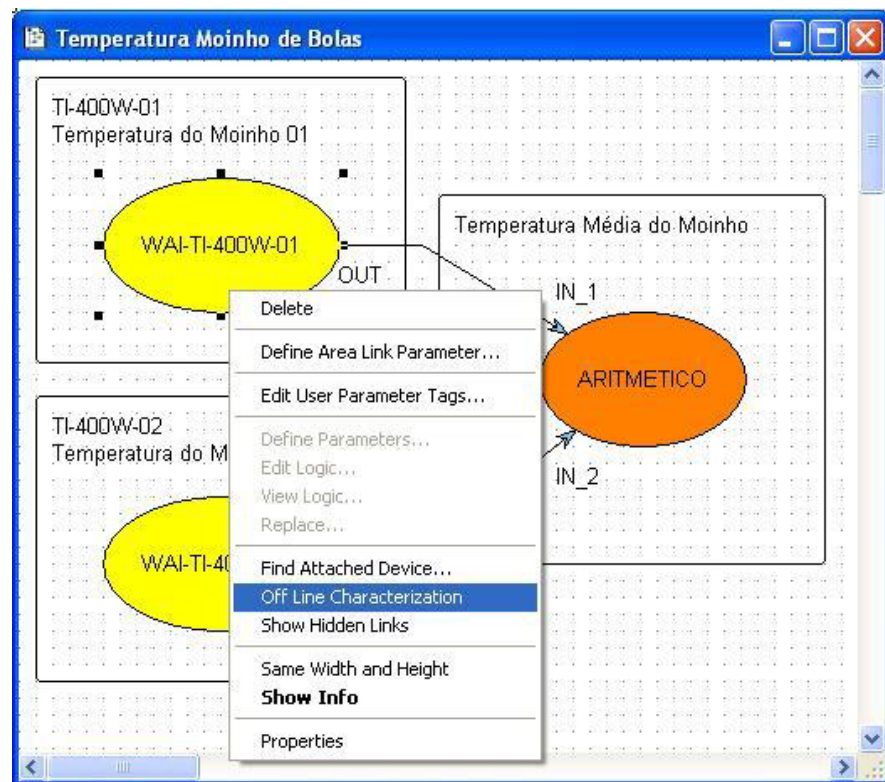


Figure 24. 38 –Offline characterization in the Strategy window

The **Off Line Characterization** dialog box will appear:

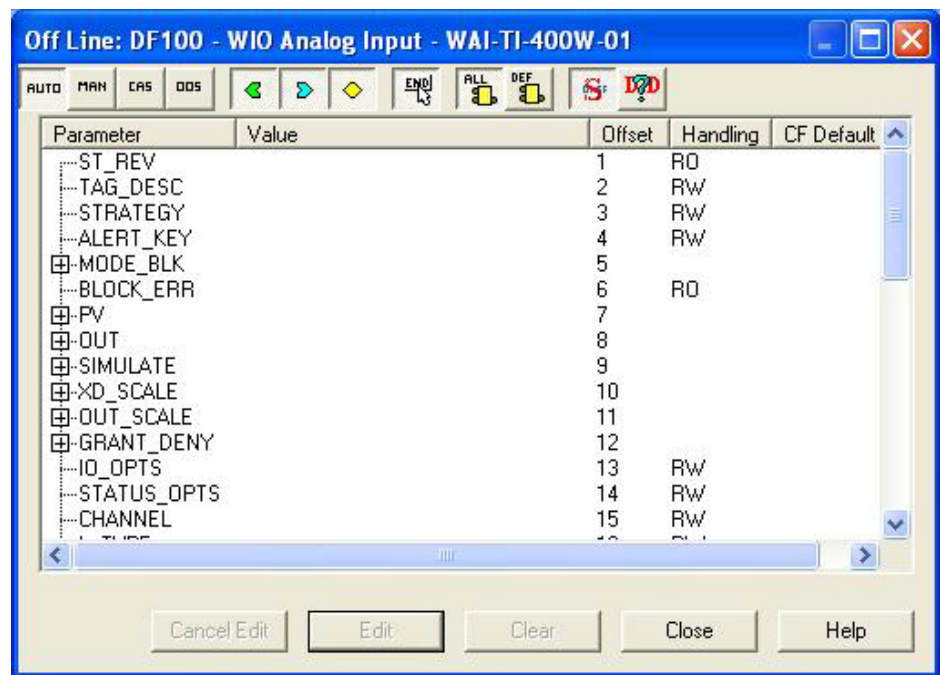


Figure 24. 39 –Offline characterization in the function block

Double-click at the right side of the parameter to change it. Another option is click it once, and then in **Edit** to start editing the parameter value. At the ending, click **End Edit**.

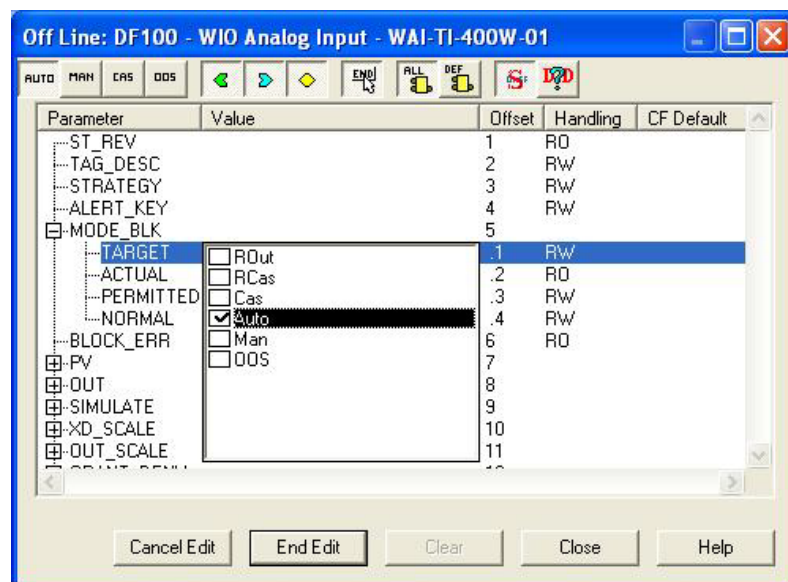



Figure 24. 40 –Editing the parameter in the Function Block Characterization box

The following list shows the parameters that must be set for this area:

DEVICE	TAG	BLOCK	PARAMETER
DF100	DF100	DF100-RB-1	MODE_BLK.Target = AUTO
		DF100-DIAG-1	MODE_BLK.Target = AUTO
		DF100-TBHG-1	MODE_BLK.Target = AUTO
		TI-400W-01	MODE_BLK.Target = AUTO
			HART_EXPD_DEV_INFO. HART_VERSION = 7 NUM_OF_PROCESS_VARIABLES = 12 HART_DD_REVISION = 0 HART_DEVICE_REVISION = 2 HART_DEVICE_TYPE = 15881 MANUFACTURER_ID = 62 DISTRIBUTOR_ID = 62 ANALOG_DISABLE = Unused DEVICE_PROFILE = 129
			VAR_NAMES9[1] = TI-400W-01_Temperatura_do_moinho
			LOCAL_MOD_MAP = 0
		TI-400W-02	MODE_BLK.Target = AUTO
			HART_EXPD_DEV_INFO. HART_VERSION = 7 NUM_OF_PROCESS_VARIABLES = 12 HART_DD_REVISION = 0 HART_DEVICE_REVISION = 2 HART_DEVICE_TYPE = 15881 MANUFACTURER_ID = 62 DISTRIBUTOR_ID = 62 ANALOG_DISABLE = Unused DEVICE_PROFILE = 129
			VAR_NAMES9[1] = TI-400W-02_Temperatura_do_moinho
			LOCAL_MOD_MAP = 1
		WAI-TI-400W-01	MODE_BLK.Target = AUTO
			CHANNEL_TAG = TI-400W-01_Temperatura_do_moinho
		WAI-TI-400W-02	MODE_BLK.Target = AUTO
			CHANNEL_TAG = TI-400W-02_Temperatura_do_moinho
		ARITMETICO	MODE_BLK.Target = AUTO
			ARITH_TYPE = Average

After the parameter setting, the user can start the device communication. It is necessary commissioning the DF100. If this procedure is not executed, the **Syscon** will detect the not commissioned controller and the download for this device will be aborted. Finishing the commissioning, the download process can start. The download process can be executed, for

example, returning to the **PROJ_DF100** window, right-clicking the **Fieldbus Networks** icon, , and selecting the **Download** option. For further details about the available download types, refer to the **Syscon** manual.

DF100 conventional function blocks

- Resource (RS2)
- Diagnostic transducer block (DIAG)

Control and calculation function blocks

- PID Control (PID)
- Enhanced PID control (EPID)
- Advanced PID control (APID)
- Arithmetic (ARTH)
- Splitter (SPLT)
- Enhanced signal characterizer (ECHAR)
- Integrator (INTG)
- Enhanced analog alarm (EAALM)
- Input selector (ISEL)
- Setpoint ramp generator (SPG)

- Enhanced setpoint ramp generator (ESPG)
- Timer and logic (TIME)
- Lead lag (LLAG)
- Output signal selector and dynamic limiter (OSDL)
- Constant and contained RW (CTRW)
- Flip-flop and edge trigger (FFET)
- Analog data transfer (ADT)
- Discrete data transfer (DDT)

DF100 specific blocks

Listed below are the blocks defined specifically for DF100. For additional information about them see Library B Function Blocks Instruction Manual.

- Transducer Block for HART Gateway (TBHG)
- Transducer Block for WirelessHART (TBWH)
- ROM Analog Input (ROMAI)
- 16 Multiple Analog Input (MAI16)

Modbus Protocol Support

Supported Characteristics

- RS-485 port: physical level EIA-485, Modbus RTU, characteristics configured in the **MBCF** block
 - BAUD_RATE: up to 19.2 Kbps
 - STOP_BITS: 1 or 2
 - PARITY: Even, odd or none
 - MASTER_SLAVE: Master or Slave
- ETH1 and ETH2 ports: physical level Ethernet TCP/IP, Modbus TCP/IP
 - Modbus TCP/IP : multimaster protocol
 - ETH1: DF100 can perform simultaneously the role of master and slave, and it is not necessary configuration
 - ETH2: DF100 performs only slave role
- Bypass: bridge functionality, conversion of the physical level. When the RS-485 port is configured as Master, Modbus commands sent via TCP / IP to the IP Address of a given controller, but whose ID is different from of this DF100, the command is retransmitted on the RS-485 port and the response that comes to this port will be retransmitted to the Ethernet port.
- The DF100 address on the Modbus network is configured in the **MBCF** block, **DEVICE_ADDRESS** parameter. This device identifier (ID) is the same of the three ports (RS-485, ETH1 and ETH2).
- It is possible to configure the registers swap through **MBCF** block, **RTS_CTS** parameter, and it is applicable to all ports (RS-485, ETH1 and ETH2).
 Exemplifying the functionality of the registers swap, the order of bytes in the Modbus message is the following if the swap was not configured (**RTS_CTS** = False)
 101.325 = 0x42 (MSB) 0xCA 0xA6 0x66 (LSB)
 Register 402.601 = 0xA6 0x66
 Register 402.602 = 0x42 0xCA
 In the response message in which the two registers were requested we have:
 0xA6 0x66 0x42 0xCA

 If configured to perform swap of registers (**RTS_CTS** = True) we have:
 Register 402.601 = 0x42 0xCA
 Register 402.602 = 0xA6 0x66
 In the response message in which the two registers were requested we have:
 0x42 0xCA 0xA6 0x66
- Types of standard commands supported as master and slave:

FUNCTION CODE	DESCRIPTION
03	Reading of a Holding Registers range (reading and writing variables)
04	Reading of an Input Registers range (reading variables)
06	Writing in a single Holding Register
16	Writing in a Multiple Registers range

- Native mapping: Variables mapped to Modbus regardless of configuration. Field devices variables are mapped to Modbus variables in Input Register (read only) or Holding Register (reading and writing). See the item "Native Mapping" for details.
- Configured mapping: By configuring the MBCS Modbus block, you could map some block parameters of the DF100 (aiming at an addresses sequence, and so use the reading or writing commands in registers range) as Holding Register.

Native mapping

Mapped variables

Several **TBWH** block parameters of the DF100 are mapped to Modbus. In other words, as the **TBWH** maps field devices, a variable set of each field device is mapped to Modbus. For each instantiated **TBWH** block whose **LOCAL_MOD_MAP** parameter is properly parameterized⁴, the following parameters / variables are available to Modbus:

⁴ The valid values interval for the **LOCAL_MOD_MAP** parameter is defined in the Parameters description of the **TBWH** block.

- **TBWH** block tag
- **FD_SIMULATE**;
- **LIVE_LIST_ST**;
- **HART_PV**;
- **HART_SV**;
- **HART_TV**;
- **HART_QV**;
- **HART_5V**;
- **HART_6V**;
- **HART_7V**;
- **HART_8V**;
- **PRIMARY_VALUE**
- **VAR_UNITS9**;
- **HART_VAR_CODES8**.

NOTES

1. All bits of Field Diagnostics are mapped. See definition of Field Diagnostics for DF100.
2. For **HART_PV** to **HART_8V** and **PRIMARY_VALUE** variables, Status and Value are mapped.

Discrete Inputs Variables

The Modbus address for any discrete input variable can be calculated by the following equation.

$$EndereçoModbusDI = 104.001 + LOCAL_MOD_MAP * 50 + Offset$$

Where:

- **EndereçoModbusDI**: Modbus address of the discrete input variable.
- **LOCAL_MOD_MAP**: value of the same name parameter that was configured by user in the **TBWH** block.
- **Offset**: displacement of the variable with respect to the base address

The following table presents the DF100 discrete input variables that are mapped to Modbus, as well as its respective offsets, lengths (in bits), and Modbus addresses.

NOTE

The Modbus addresses were calculated for few **LOCAL_MOD_MAP** values. If you need the Modbus address of any other discrete input variable whose **LOCAL_MOD_MAP** is not on the table, use the above equation to calculate it.

Discrete Input Variable	Number of Bits	Offset	LOCAL_MOD_MAP											
			0	1	2	3	4	5	10	20	30	40	50	99
FD_SIMULATE	32	0	104001	104051	104101	104151	104201	104251	104501	105001	105501	106001	106501	108951
LIVE_LIST_ST	1	32	104033	104083	104133	104183	104233	104283	104533	105033	105533	106033	106533	108983

Input Register Variables

The Modbus address for any input register variable can be calculated by the following equation.

$$\text{EndereçoModbusIR} = 300.001 + \text{LOCAL_MOD_MAP} * 80 + \text{Offset}$$

Where:

- **EndereçoModbusIR:** Modbus address of the input register variable.
- **LOCAL_MOD_MAP:** value of the same name parameter that was configured by user in the TBWH block.
- **Offset:** displacement of the variable with respect to the base address

The following table presents the DF100 input register variables that are mapped to Modbus, as well as its respective offsets, lengths (in bits), and Modbus addresses.

NOTE

The Modbus addresses were calculated for few **LOCAL_MOD_MAP** values. If you need the Modbus address of any other input register variable whose **LOCAL_MOD_MAP** is not on the table, use the above equation to calculate it.

Input Register Variable	Number of Registers	Offset	LOCAL_MOD_MAP											
			0	1	2	3	4	5	10	20	30	40	50	99
HART_PV.Status	1	0	300001	300081	300161	300241	300321	300401	300801	301601	302401	303201	304001	307921
HART_PV.Value	2	1	300002	300082	300162	300242	300322	300402	300802	301602	302402	303202	304002	307922
HART_SV.Status	1	3	300004	300084	300164	300244	300324	300404	300804	301604	302404	303204	304004	307924
HART_SV.Value	2	4	300005	300085	300165	300245	300325	300405	300805	301605	302405	303205	304005	307925
HART_TV.Status	1	6	300007	300087	300167	300247	300327	300407	300807	301607	302407	303207	304007	307927
HART_TV.Value	2	7	300008	300088	300168	300248	300328	300408	300808	301608	302408	303208	304008	307928
HART_QV.Status	1	9	300010	300090	300170	300250	300330	300410	300810	301610	302410	303210	304010	307930
HART_QV.Value	2	10	300011	300091	300171	300251	300331	300411	300811	301611	302411	303211	304011	307931
HART_5V.Status	1	12	300013	300093	300173	300253	300333	300413	300813	301613	302413	303213	304013	307933
HART_5V.Value	2	13	300014	300094	300174	300254	300334	300414	300814	301614	302414	303214	304014	307934
HART_6V.Status	1	15	300016	300096	300176	300256	300336	300416	300816	301616	302416	303216	304016	307936
HART_6V.Value	2	16	300017	300097	300177	300257	300337	300417	300817	301617	302417	303217	304017	307937
HART_7V.Status	1	18	300019	300099	300179	300259	300339	300419	300819	301619	302419	303219	304019	307939
HART_7V.Value	2	19	300020	300100	300180	300260	300340	300420	300820	301620	302420	303220	304020	307940
HART_8V.Status	1	21	300022	300102	300182	300262	300342	300422	300822	301622	302422	303222	304022	307942
HART_8V.Value	2	22	300023	300103	300183	300263	300343	300423	300823	301623	302423	303223	304023	307943
PRIMARY_VALUE.Status	1	24	300025	300105	300185	300265	300345	300425	300825	301625	302425	303225	304025	307945
PRIMARY_VALUE.Value	2	25	300026	300106	300186	300266	300346	300426	300826	301626	302426	303226	304026	307946
BLOCK_TAG	16	27	300028	300108	300188	300268	300348	300428	300828	301628	302428	303228	304028	307948
VAR_UNITS9	9	43	300044	300124	300204	300284	300364	300444	300844	301644	302444	303244	304044	307964

Holding Register Variables

The Modbus address for any holding register variable can be calculated by the following equation.

$$\text{EndereçoModbusHR} = 404.001 + \text{LOCAL_MOD_MAP} * 20 + \text{Offset}$$

Where:

- **EndereçoModbusHR**: Modbus address of the holding register variable.
- **LOCAL_MOD_MAP**: value of the same name parameter that was configured by user in the **TBWH** block.
- **Offset**: displacement of the variable with respect to the **TBWH** base address

The following table presents the DF100 holding register variables that are mapped to Modbus, as well as its respective offsets, lengths (in bits) and Modbus addresses.

NOTE

The Modbus addresses were calculated for few **LOCAL_MOD_MAP** values. If you need the Modbus address of any other holding register variable whose **LOCAL_MOD_MAP** is not on the table, use the above equation to calculate it.

Holding Register Variable	Number of Registers	Offset	LOCAL_MOD_MAP											
			0	1	2	3	4	5	10	20	30	40	50	99
HART_VAR_CODE S8	8	0	404001	404021	404041	404061	404081	404101	404201	404401	404601	404801	405001	405981

General notes about Native Addressing

- If the requested register in Modbus commands involves (exclusively or not) reserve address, the DF100 will respond with a zero value for such register.
- The Modbus of DF100 is based on version known as **Combined Scenario**. Thus, the DF100 allows enabling / disabling the swap of variable registers of float type through the **RTS_CTS** parameter of Modbus **MBCF** block.
- The Modbus addresses, of any available Modbus point at DF100, may also be consulted through the information page of the DF100 Web Server - see next figure. After launching it, if necessary, use the arrows of decreasing or increasing advancement to determine the desired value for the **LOCAL_MOD_MAP**. Once you determine the value of **LOCAL_MOD_MAP**, Modbus addresses will be automatically updated.

Welcome to DF100 Webserver TAG: DF100_WHG Serial number: 8

SMAR
Information
Setup
Device Info
WirelessHART
Network
Manager
Modbus
Addresses

LOCAL_MOD_MAP (Informative)

<< < 0 > >>

Input Register

Variable	Num. of Regs.	Offset	Modbus Address
HART_PV.Status	1	0	300001
HART_PV.Value	2	1	300002
HART_SV.Status	1	3	300004
HART_SV.Value	2	4	300005
HART_TV.Status	1	6	300007
HART_TV.Value	2	7	300008
HART_QV.Status	1	9	300010
HART_QV.Value	2	10	300011
HART_5V.Status	1	12	300013
HART_5V.Value	2	13	300014
HART_6V.Status	1	15	300016
HART_6V.Value	2	16	300017
HART_7V.Status	1	18	300019
HART_7V.Value	2	19	300020
HART_8V.Status	1	21	300022
HART_8V.Value	2	22	300023
PRIMARY_VALUE.Status	1	24	300025
PRIMARY_VALUE.Value	2	25	300026
BLOCK_TAG	16	27	300028
VAR_UNITS9	9	43	300044

Discrete Input

Variable	Num. of Bits	Offset	Modbus Address
FD_SIMULATE	32	0	104001
LIVE_LIST_ST	1	32	104033

Holding Register

Variable	Num. of Regs.	Offset	Modbus Address
HART_VAR_CODES8	8	0	404001

Figure 24. 41 –Web page with Modbus addresses for LOCAL_MOD_MAP equals to zero

Modbus Combined Scenario

The DF100 supports the **Modbus Combined Scenario**, in which it is able to perform the role as master and slave simultaneously via Modbus TCP / IP regardless of any configuration.

The selection between master and slave would only apply to the RS-485 port, which could be communicating simultaneously to the Modbus TCP / IP on ETH1 and ETH2 ports.

ADDING REDUNDANCY TO THE DFI302 HSE CONTROLLERS

Introduction

To meet the requirements for fault tolerance, system availability and safety of the industrial process, the following DFI302 HSE controllers support redundancy: DF62, DF63, DF73, DF75, DF89, DF95 and DF97.

The Hot Standby redundancy strategy is used when the Primary controller executes all tasks, and the Secondary controller is the one that, continuously synchronized with the Primary, keeps ready to assume the process in case the Primary controller fails. This event, where the Secondary assumes the process by changing its function to Primary, is also called switch over, and occurs in a bumpless and autonomous way.

In those controllers the redundancy is Device D-3 type, in compliance with the specification “High Speed Ethernet (HSE) Redundancy Specification FF-593” of Fieldbus Foundation. By this capacity (Device D-3), during the entire operation time, the controller pair is seen as a single device by the configurator. Thus, actions as commissioning, decommissioning, configuration download, and parameterizations affect both controllers (Primary and Secondary).

The different failures types, such as failures in the communication ports, are indicated even if occur in the Secondary, this allows proactive maintenance and thus ensure the redundancy availability.

This new generation of Hot Standby redundancy of the DFI302 HSE controllers brings better diagnostic and failure detection capacities, autonomy during startup and transparency for the configuration tools.

IMPORTANT

The characteristics described in this section are valid for DF62, DF63, DF73, DF75, DF89, DF95, and DF97 controllers, except the information related to the FOUNDATION™ fieldbus H1 channels which are only applied to the DF62 and DF63 controllers.

It is assumed that the user is familiar with **Studio302** and **Syscon**. For further information, refer to the respective manuals.

Hot Standby Redundancy

With the Hot Standby redundancy full redundancy is achieved, heavily improving the fault tolerance, plant availability and safety. All the controller functionalities and databases have redundancy:

1. Device redundancy (Hardware redundancy);
2. Network redundancy (or LAN redundancy, for controllers with two Ethernet ports – DF63, DF73, DF75, DF89, DF95, and DF97);
3. Ethernet Gateway ↔ 4 FOUNDATION fieldbus H1 ports;
4. Link Active Scheduler (LAS) in the FOUNDATION fieldbus H1 channels;
5. Controller (running function blocks, including FFB/ Ladder logic);
6. Supervision;
7. Modbus Gateway ↔ 4 FOUNDATION fieldbus H1 ports;
8. Synchronism channel redundancy.

The procedures for configuration and maintenance are as simple as for a non-redundant system, saving time to get the system running. Only one configuration download is necessary to configure the redundant pair. And in case of replacement of a failed controller none configuration download or user intervention is necessary. The new module inserted is automatically recognized, receiving the whole configuration from the controller in operation.

Preparing a Redundant System

In order to have a true redundant system, not just all the devices must be redundant but also the entire system architecture must be designed as redundant. The more elements with redundancy ability the system have, better reliability and availability can be achieved. A typical redundant architecture based on DFI302 controllers can be seen in the next figure.

Ethernet network architectures

In the following topics are presented the Ethernet network architectures required for the controllers with two Ethernet ports (DF63, DF73, DF75, DF89, DF95, and DF97), and also for the DF62 which have one Ethernet port and requires a different network architecture.

In both cases the IP address of the controllers and network cards must follow the Class C (subnet mask 255.255.255.0), which means that all IP addresses on the same subnet must have the first three bytes equal.

For reference purpose, in each pair of controllers, we designate one of them as **A** and another one as **B**. It may be associated with the controllers' position in the panel. For example, the controllers on the left side are "A", and the controllers on the right side are "B". This designation (A and B) is an important reference for some of the procedures in this section, as well as for documentation of the controllers IP list. This is a static identification, different from the controllers function (Primary / Secondary) which is a dynamic state of the pair and can take any sequence depending on the failures that may cause a switch over.

IMPORTANT

It is highly recommended that the automation Ethernet networks are physically separated from other networks such as the corporate network, of common use. This is necessary to guarantee the safety and good operation of the automation Ethernet network.

Network architecture for controllers with two Ethernet ports

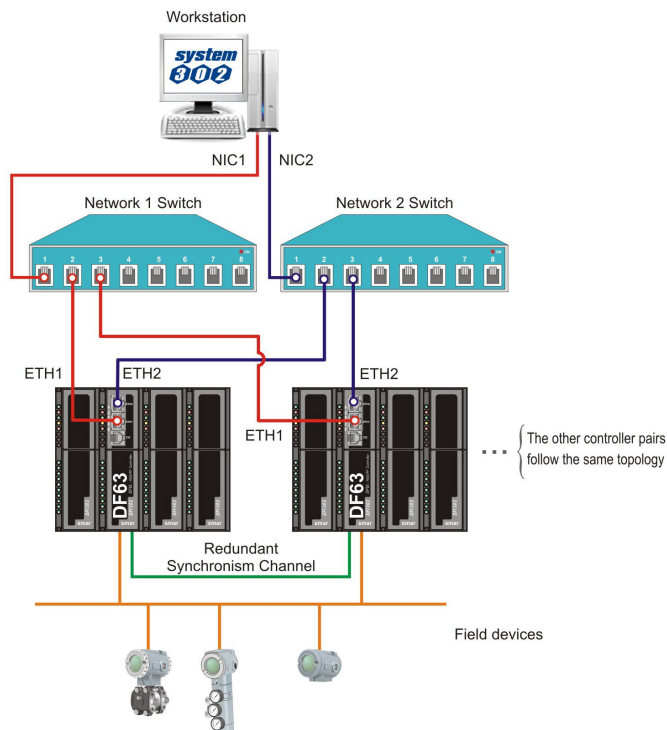


Figure 25. 1 – Network architecture for controllers with two Ethernet ports

In the Figure 25.1 are presented the network requirements, detailed as follows:

- For controllers with two Ethernet ports, the networks' nodes have to be equal (the last byte of IP addresses must be equals). For the ETH1 port must be used one subnet, and for the ETH2 port must be used another subnet.
Example: the first DF63 port (ETH1) = 192.168.164.34, the second DF63 port (ETH2) = 192.168.165.34.
This way, there will be two subnets: 192.168.164.X (red network) and 192.168.165.X (blue network), the first serves all ETH1 ports, and the second serves all ETH2 ports of all the controllers. These two subnets must be designed to be physically separated, using different network elements.
- The workstations must have two network cards (NIC1 and NIC2) and each one must have the IP configured in one of the subnets as explained previously. Example: NIC1 = 192.168.164.250 and NIC2 = 192.168.165.250.

This way, for controllers with two Ethernet ports, using the network architecture described the network redundancy is obtained. Thus, a failure that affects one of the network segments will be covered by the network redundancy, adopting the other network path which was not affected by the failure.

Network architecture for the DF62 controller

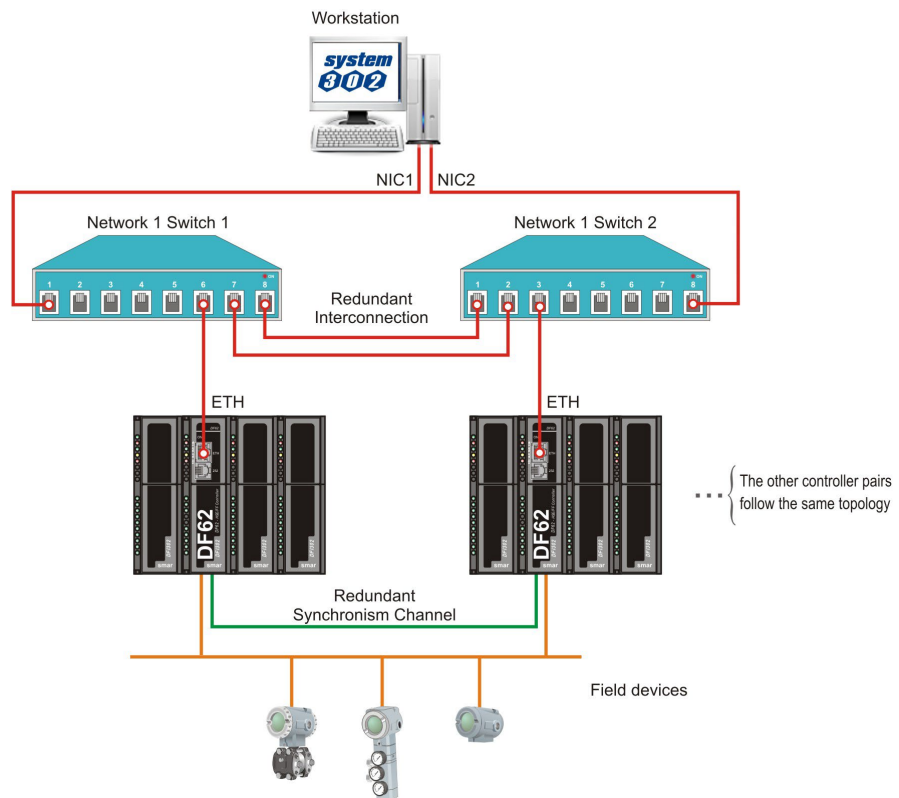


Figure 25. 2 – Network architecture for the DF62 controller

In the Figure 25.2 are presented the network requirements when using the DF62, detailed as follows:

- Just one subnet must be used (in the figure above the red network represents only one subnet). Example: all IP addresses in the range 192.168.164.X (only the last byte changes). But two switches and two network cards must be used (NIC1 and NIC2), because this way we still have redundancy on the network segments between workstations and switches.

- The network cards may have any IP addresses, but necessarily in the same subnet.
E.g.: NIC1 = 192.168.164.250 and NIC2 = 192.168.164.251.
- The controllers A must be connected to the *switch* 1 and the controllers B must be connected to the *switch* 2 (see Figure 25.2). That is, for each controller pair, one of the DF62 will be connected to a switch and the other DF62 of the same pair will be connected to the other switch.

IMPORTANT

It is mandatory that the **switches are connected by two paths**. This is necessary to guarantee that a single failure in the interconnection cable between the switches does not affect the HSE control links which may exist between different pairs of controllers. Necessarily the switches must support RSTP (Rapid Spanning Tree Protocol) and have this feature enabled. Preferably the switches should have ports *auto sensing* type to avoid preoccupations with types of cable and ports that will be used for these interconnections.

To increase the project safety, the interconnections should be performed with the analysis of a network administrator or IT professional. Using switches with no RSTP support may cause an interruption on the Ethernet network.

This way, for a system with pairs of DF62 controllers, using the previous network architecture, the network redundancy is achieved between the workstations and switches. A failure that occurs on a segment between the workstations and switches will be covered by the network redundancy adopting the other network path which was not affected by the failure.

For a failure that occurs in a segment between switches and controllers, and affects the current Primary controller, it will be covered by the controllers' redundancy through the switch over of them.

In the tables 25.1 and 25.2 is presented an example of the IPs' configuration for the two architectures described in this section, both for the workstations and the controllers involved.

		Controllers with one Ethernet port	Controllers with two Ethernet ports
Station 1	NIC1	192.168.164.250	192.168.164.250
	NIC2	192.168.164.251	192.168.165.250
Station 2	NIC1	192.168.164.252	192.168.164.251
	NIC2	192.168.164.253	192.168.165.251

Table 25.1 – Example of the IPs' addresses list for the workstations, considering the two types of network architecture

		Controllers with one Ethernet port	Controllers with two Ethernet ports	
Pairs	Controllers	ETH port IPs	ETH1 port IPs	ETH2 port IPs
PAIR 1	DF A	192.168.164.10	192.168.164.10	192.168.165.10
	DF B	192.168.164.11	192.168.164.11	192.168.165.11
PAIR 2	DF A	192.168.164.12	192.168.164.12	192.168.165.12
	DF B	192.168.164.13	192.168.164.13	192.168.165.13
PAIR 3	DF A	192.168.164.14	192.168.164.14	192.168.165.14
	DF B	192.168.164.15	192.168.164.15	192.168.165.15

Table 25.2 – Example of the IPs' addresses list for the controllers, considering the two types of network architecture

Configuring the System302 ServerManager and Syscon


At the **Studio302** toolbar, click the button  and the **System302 ServerManager** dialog box will open.



Figure 25. 3 – System302 ServerManager

Click the **Network** option and the next window will open.

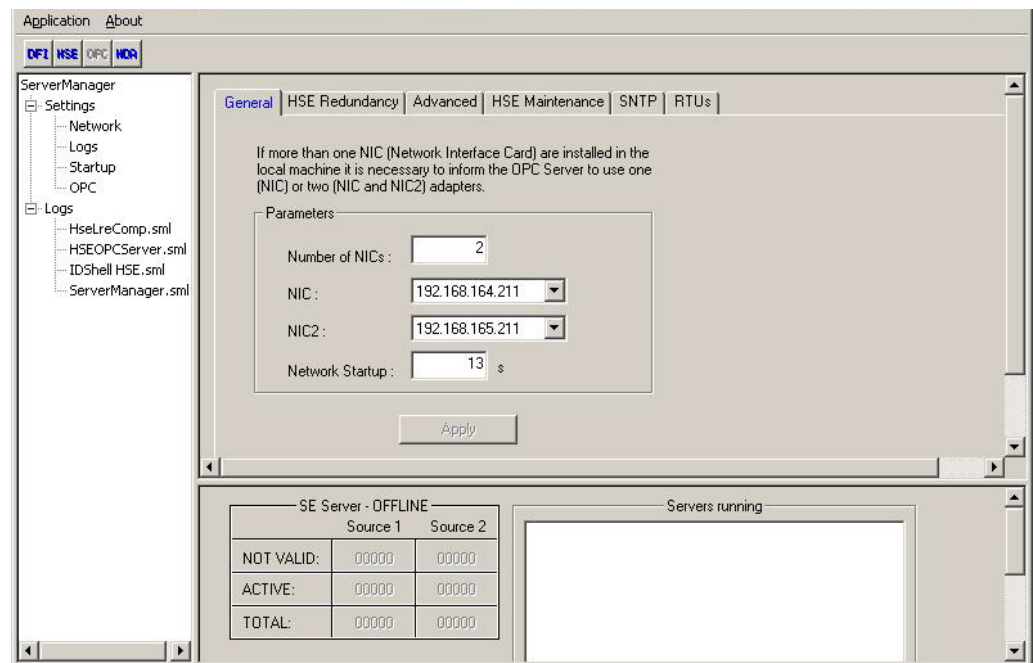


Figure 25. 4 – System302 Serve Manager: General tab

At the **General** tab, type the number of NICs (network cards) used in the machine, in this case is 2 (redundant system). Select the IP address of the NICs used by the **System302 ServerManager**.

At the **HSE Redundancy** tab, configure the fields as in the following figure.

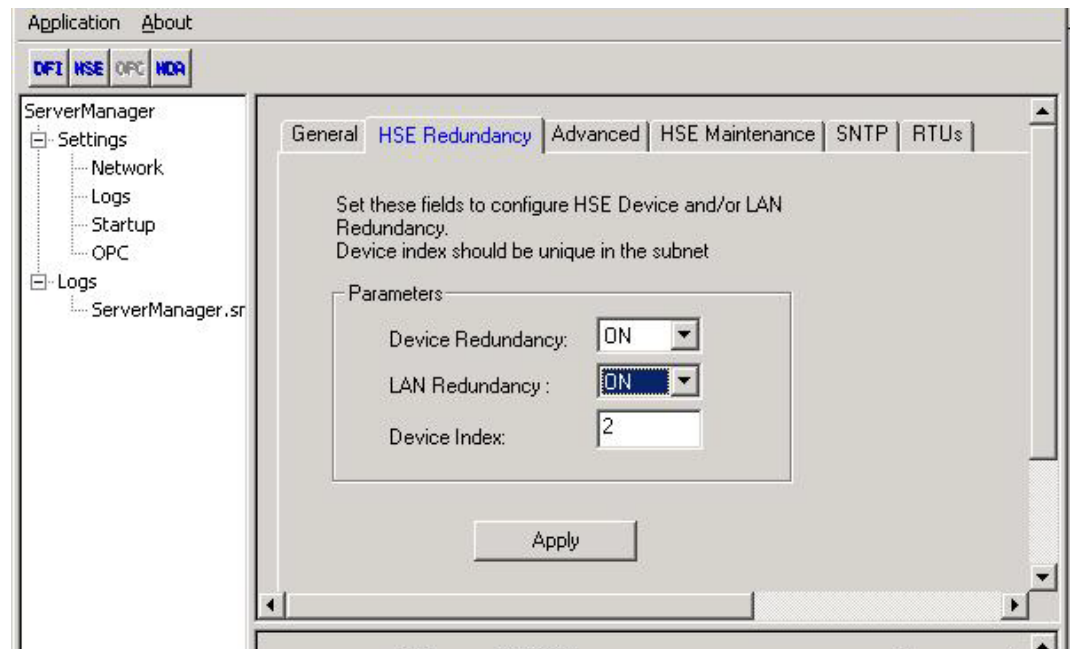


Figure 25. 5 – System302 ServerManager: HSE Redundancy tab

Select **ON** for **Device Redundancy**. If there are two network cards the network will be redundant. In this way select **ON** for **LAN Redundancy** (even using the DF62 controller).

At the **Device Index** text box, type a value between **1** and **9** for each machine, and every machine must have a unique number. In the HSE network, the **Device Index** represents the network address for each device for routing purposes, because of this it needs to be unique.

At the **Syscon**, the only care to be taken in the control strategy configuration related to redundancy is:

- Right-click each controller which will be redundant and choose **Attributes** option;
- Configure the item "**Is Redundant (HSE Only)**" as enabled.

Synchronism channel

A RS232 serial port is dedicated to synchronism between the Primary and Secondary controllers using the DF82 (0.5 m) or DF83 (1.8 m) cables. See the next figures.

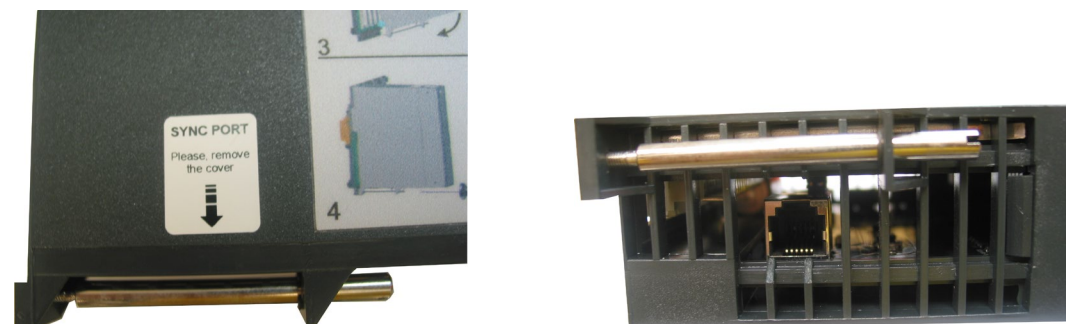


Figure 25. 6 - Label to identify the serial synchronism port (Left) and the serial synchronism port in the bottom part of the module (Right)

Thus, the distance between controllers is limited to 1.8 m. Therefore they have to be installed preferentially in the same panel, but with different power supplies and independent no-breaks.

DIFFERENTIAL

The DFI302 HSE controllers have redundancy of synchronism channel as a differential, with up to three possible paths: between ETH1 ports, between ETH2 ports and through serial port. Thus is ensured a greater availability of the own device's redundancy.

IMPORTANT

- The synchronism between the controllers is done through the serial port mainly during the startup. After controllers' startup, the synchronism is done through the Ethernet ports, ensuring a greater transfer rate for synchronism. If there is a communication failure in an Ethernet port, the synchronism is reestablished by another one. If there is a communication failure in both Ethernet ports, the synchronism will be done through the serial synchronism port.
- Exception: If a ladder application is used the synchronism has to be done through the Ethernet ports because demands a greater transfer rate. If there is a communication failure in both Ethernet ports, the controller pair will lost the synchronism of the ladder application, thus, the redundancy is not fully available, because the Secondary will not be updated to assume if the Primary fails. It is important that failures are repaired in order to have the redundancy available again.
- It is mandatory that the synchronism serial cable (DF82/DF83) remain connected all the time. This peer-to-peer connection is what determines the formation of a redundant controller pair during the plant startup, and also during restart up, after scheduled stops.

FOUNDATION fieldbus H1 channels

FOUNDATION fieldbus H1 redundant segments: For each FOUNDATION fieldbus H1 channel, from a common point in the panel, a segment can be ramified up to the Primary controller, and another one up to the Secondary controller offering failure tolerance in these segments.

Accessing the I/O bus

To allow the access to the I/O modules in a redundant way, a proper hardware topology must be used. Thus, it is necessary to use the DF78 or DF92 rack, where in the first two slots (Power Supply 1 and Power Supply 2) the power supplies DF50 (AC/DC) or DF56 (DC/DC) must be inserted, thus the power supply redundancy is provided. The controllers must be installed side by side in the CPU 1 and CPU 2 slots. The next two figures demonstrate the use of DF78 rack.

DF78 and DF92 racks allow direct access to the I/O modules in a safe and transparent way when redundant controllers are used. Also is possible hot swap (insertion/extraction) of the controllers for maintenance purposes.

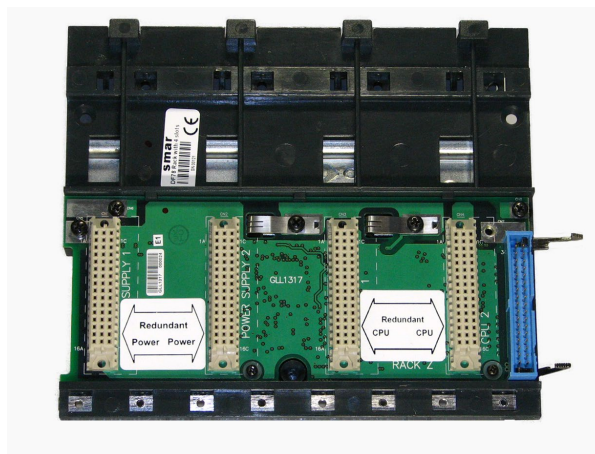


Figure 25. 7 –DF78 Rack



Figure 25. 8 – Example of modules arrangement in the DF78 rack (DF50-DF50-CPU1-CPU2)

Hot Standby Redundancy Working

Starting up the redundancy

The controller which starts up first becomes the Primary. In case of both controllers which form a pair are started up at the same time, both will assume the same role that they were operating previously (nonvolatile information).

In the absence of nonvolatile information (startup immediately after firmware updating or Factory Init mode) and in case of both controllers are started up at the same time, the controller which has a bigger serial number will be elected as Primary, and its partner will be the Secondary.

IMPORTANT

The controllers have conditions to define their role (Primary or Secondary) autonomously during the startup and no user action is necessary.

Besides the information above mentioned, during the entire operating time of the controller pair and in conditions free of failures, we have the following:

- There is not physical difference between Primary and Secondary controllers;
- There is not preference for a controller rather than another, or for a rack position rather than another, to determine which controller will be the Primary.

Operational Transparency

In **Syscon** and **LogicView for FFB** the redundancy is seen in a transparent manner to the user, i.e., the redundant pair is seen as a single device. This concept is known as “operational transparency of redundancy”.

In practice, the configurator will always be connected to the controller that is the current Primary. Thus, all download or configuration actions performed in the configurator will be destined to the current Primary. The synchronism implemented in the controllers' firmware is responsible for updating constantly the Secondary. If there is a switch over, the new Primary is automatically recognized by the configurator, so that at no time the user has to take extra care due to redundancy in the actions of the operation or maintenance of the plant.

Switch over conditions

The different failures that may occur in such system, lead it to a switch over, when the Secondary becomes Primary, and vice-versa in a bumpless way. The possible reasons for a switch over, divided in two types, are as follows:

General Failures

When the whole controller fails, this comprises:

- Hardware failure;
- Power off;
- Removal of the controller from the rack.

Bad Condition Failures

When one of the Primary controller's ports fails:

- Failure in all Ethernet cables directly connected to the Primary;
- Failure in an H1 channel (hardware or cables) of the Primary;
- Failure in the Modbus communication (hardware or cables; in case of operating as master).

The system is capable of checking which controller has the best conditions, electing it as the Primary. It is assured the recovery of one failure at a time. That is, once a failure has occurred, a second failure will be recovered by redundancy only if the first failure has been fixed. While the failure is not fixed, the system has the redundancy not fully available (in case of **Bad Condition Failures**) or even not available (in case of **General Failures**).

For the case of **General Failures**, as soon as the failed controller recovers a healthy state or is replaced, the controllers automatically become a redundant pair again. That is, the system automatically recognizes a new inserted controller. After a failure, the maximum switch over time may alter depending on the specific failure, but typically takes 2 seconds or less.

In order to monitoring the redundancy state, some parameters available in the Redundancy Transducer function block (TRDRED) should be used. See the following table. For further details see the Function Blocks manual.

Parameter	Valid Range/ Options	Description
RED_PRIMARY_SN	0 ~ 65535	Indicates the Serial Number of the Primary controller.
RED_SECONDARY_SN	0 ~ 65535	Indicates the Serial Number of the Secondary controller.
RED_SYNC_STATUS	0: Not defined 1: Stand Alone 2: Synchronizing 3: Updating Secondary 4: Synchronized 5: WARNING: Role Conflict 6: WARNING: Sync Cable Fail 7: WARNING: Updating Secondary Fail	Indicates the Synchronism status of the controller pair. 0: Default value just after start up. 1: Stand alone operation (no redundancy). 2: Checking configuration for synchronization. 3: Primary transferring configuration to the Secondary. 4: The Primary continuously updates the Secondary dynamic databases. 5: It was not possible to solve autonomously the Role. (Primary / Secondary). 6: Fail on all the synchronism channels (redundancy unavailable). 7: Primary fails before synchronism is completed (redundancy unavailable).
RED_PRIMARY_BAD_CONDITIONS	Bit 0: Modbus 1: H1-1 2: H1-2 3: H1-3 4: H1-4	Bad conditions for the Primary / Secondary controllers.
RED_SECONDARY_BAD_CONDITIONS	5: LiveList 6: ETH1 7: Reserved 8: ETH2 9: Serial Sync Cable 10: Unable to Sync	

Table 25.3 – Description of main parameters of Redundancy Transducer function block

Bit	Variable	Description
0	Modbus	When working as master and if no Modbus slave device answers, it means that Modbus communication is in bad conditions. Failures on the communication path or even a failure on the slave can cause it.
1	H1-1	Failure on an H1 channel, specifying each channel had the failure.
2	H1-2	
3	H1-3	
4	H1-4	
5	LiveList	H1 Live List was not completed on Secondary controller.
6	ETH1	Synchronism failure on ETH1 port.
7	Reserved	-
8	ETH2	Synchronism failure on ETH2 port.
9	Serial Sync Cable	Failure on serial synchronism cable.
10	Unable to Sync	Firmware versions with incompatible synchronism.

Table 25.4 – Description of RED_PRIMARY_BAD_CONDITIONS and RED_SECONDARY_BAD_CONDITIONS parameters bits

IMPORTANT
To know how to proceed when warnings appear in RED_SYNC_STATUS parameter and the indications of BAD_CONDITIONS parameters refer to Troubleshooting topic.

Standby LED Behavior

The available blinking patterns for the Standby LED are summarized below. The next figure shows a graphic representation for these patterns.

- a) PRIMARY IN STAND ALONE: Standby LED turned off all the time. It indicates that there is no partner connected.
- b) SECONDARY SYNCHRONIZED: Standby LED turned on all the time. It indicates that the Secondary controller is totally synchronized with the Primary controller and the redundancy is available.
- c) PRIMARY WITH PARTNER: Each three seconds, Primary's Standby LED blinks briefly. It indicates that the Primary controller has a partner.
- d) SECONDARY SYNCHRONIZING: Standby LED blinking slowly turned off about one second and turned on about one second. It indicates that the configuration synchronism is on progress.
- e) ROLE CONFLICT: Standby LED blinking fast. It indicates the controller did not define its role during the startup. The Primary will pause for two seconds the blinking each ten times. The Secondary will blink permanently.
- f) PRIMARY - CABLE FAILURE: Standby LED blinking twice in the Primary, quickly at each two seconds. It indicates failure in the synchronism serial cable.
- g) SECONDARY – CABLE FAILURE: Standby LED blinking four times in the Secondary, quickly, at each two seconds. It indicates failure in the synchronism serial cable.
- h) PRIMARY FAILS DURING SECONDARY UPDATING: Standby LED blinking three times in the Secondary, quickly, at each two seconds. It indicates that there has been a general failure on Primary before reaching the **Synchronized** status.

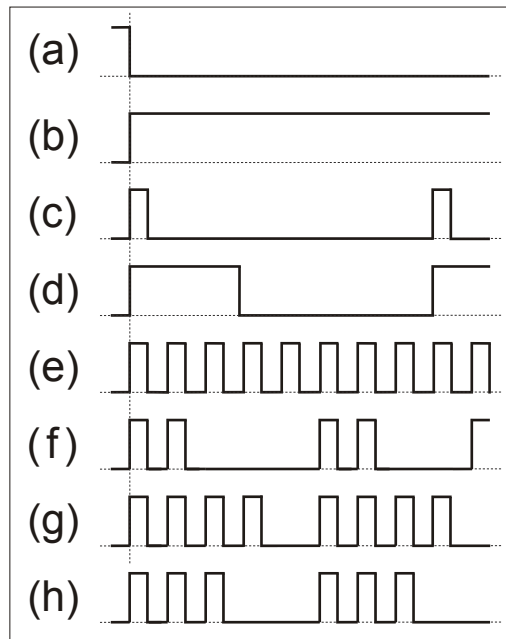


Figure 25.9 – LED Standby behavior

Procedures for Hot Standby Redundancy

The next steps are for configuration and maintenance when using the Hot Standby redundancy. It is recommended that the steps be all read and understood before they are executed.

IMPORTANT

Before executing any of the following procedures, make sure you have followed the guidelines of the topic Preparing a redundant system.

At this section, the following expressions and their respective definitions are used:

- Hold Mode: stops the firmware execution in the controller as well as all tasks in the process.
- Run Mode: executes the firmware again.
- Factory Init Mode: restores the factory configuration, erasing the user's configuration.

For further information about those expressions and how to update the firmware refer to sections Setting Up or Troubleshooting in this manual.




Configuring for the first time a redundant system

This is the procedure to configure the system with Hot Standby redundancy for the first time, at the plant startup.

1 – With the rack not powered, connect the synchronism serial cable to each controller.

2 – Connect both controllers through the H1 channels (1 to 4), in case of DF62/DF63. Connect the Ethernet cables to the corresponding controllers' ports.

3 – Power the rack where the controllers are installed. The controllers will decide autonomously which one will be Primary and Secondary. Wait until the Standby LED of one controller turns on permanently, what means the roles have been defined and the controller pair is synchronized.

4 - At the **Studio302** click **Areas** , choose the desired configuration. It will be open at **Syscon**, and then click **On-line Mode** option . Execute the commissioning of controllers and field devices. Download the configuration right-clicking **Fieldbus Networks** . For further information refer to **Syscon** manual, specially the section Creating a FOUNDATION fieldbus configuration.

5 – The controller pair will synchronize the configuration (Standby LED will blink). When the controller pair is synchronized (Standby LED turned on permanently in Secondary), the Primary will constantly update the Secondary with the process dynamic variables.

As soon the controller pair get the **Synchronized** status and **<none>** in the **BAD_CONDITIONS** parameters, the redundancy will be fully available and failure simulations can be performed.

Changing the configuration

Download the new configuration to the device commissioned in the **Syscon**. The controller pair will re-synchronize automatically.

Replacing a controller with failure

To ensure a high safety process when replacing a controller, make sure of following the steps below:

1 – With the new controller disconnected from the rack, turn off the battery switch, at least, for 30 seconds. Set the **BATTERY** position to **OFF**, wait for 30 seconds and return to the **ON** position.

2 – You must connect the synchronism cable (DF82/DF83) just before inserting the new controller. This will avoid role conflict between controllers.

3 - If possible, connect all cables: besides the synchronism cable, H1 channels (1 to 4) in case of DF62/DF63 and the Ethernet ports.

4 – Insert the new controller in the rack.

5 – In case all cables have been connected before the insertion of new controller, the synchronism will automatically start (the Standby LED should blink in the new controller). When the system is synchronized (Standby LED turned on permanently), the Primary controller will constantly update the Secondary with the process dynamic variables.

6 – In case only the synchronism cable have been connected before the insertion, a hot insertion of the H1 cables (in DF62/DF63) may be necessary. In this case, set the new controller to **Hold** (none executing), connect the H1 cables, and also the Ethernet cables. Then, return the controller to **Run** mode (execution).

7 - As soon the system get the **Synchronized** status and **<none>** in the **BAD_CONDITIONS** parameters, the redundancy will be fully available and failure simulations can be performed.

8 – For any situation different from **Synchronized** status, refer to section Standby LED Behavior to make a better diagnosis.

Adding redundant controllers in a non-redundant system

Even a non-redundant controller supports redundant operation, working as Primary in Stand Alone state.

Thus, a non-redundant system, in operation, may have redundant controllers added later without process interruption. Only is necessary that the non-redundant system has foreseen some cares according to the section Preparing a Redundant System. The procedure is the same of the previous section (Replacing a controller with failure).

Firmware update without process interruption

It is possible updating the controllers' firmware to new versions, which have new features or improvements, without process interruption.

For reference purpose in the following procedure, we designate one of the controllers as **A** and another as **B**. Suppose the controller A is the one which was Primary at the beginning of the procedure execution. That is, the references A and B are static, and the user can even associate the controller A as the one in the rack's left side and controller B as the one in the rack's right side.

Follow the steps below:

1. Be sure the system is in the **Synchronized** status and it has **<none>** in the **BAD_CONDITIONS** parameters. So, using **FBTools** update the firmware of the controller A (current Primary). At this moment, the other controller (B) will take over becoming the current Primary.
2. After finishing the firmware's update of controller A, the controller pair will synchronize with the current Primary (B) transferring the entire configuration to the other one (A). Wait for the system get the **Synchronized** status and it has **<none>** in the **BAD_CONDITIONS** parameters.
3. Using **FBTools** update the firmware of the current Primary controller (B). At this moment, the other controller (A) will take over becoming the current Primary.
4. After finishing the firmware's update, the controller pair will synchronize with the current Primary (A) transferring the entire configuration to the other one (B). As soon the system get the **Synchronized** status and it has **<none>** in the **BAD_CONDITIONS** parameters, the redundancy is fully available again and failure simulations can be performed.

After finishing this procedure both controllers will have updated firmware and the original configuration will be preserved without need of process interruption.

Troubleshooting

Role Conflict

This exceptional situation occurs when some procedure was not followed. It is signaled by the **RED_SYNC_STATUS** parameter (value **5: WARNING: Role Conflict**) and by the Standby LED (see Standby LED behavior topic).

There is a chance to occur role conflict only when a controller has already had a partner operating in redundancy when one of the controllers is exchanged without doing a Factory Init in the new controller inserted. In such situation the redundancy does not define the function of the new controller because of security reasons. The user is responsible to decide which controller has the right configuration.

Solution: the user has to do a Factory Init in the controller which will be the Secondary (the entire configuration of this controller will be erased and it will receive the configuration from another controller).

Correction of synchronism cables failure

Once there is a failure in some of the synchronism media (Serial, ETH1, ETH2) it is signaled by the **BAD_CONDITIONS** parameters, respectively with: **Serial Sync Cable**, **ETH1** and **ETH2** (see Table 25.4). Even the synchronism channel is redundant (with up to three paths); it is recommended that as soon as a failure is signaled in some of the paths, it has to be fixed.

Indications of bad condition on ETH1 and ETH2 will be signaled on both the Primary controller and the Secondary controller. This indication advises that the controllers are unable to perform synchronization for a given path (ETH1/ETH2). The individual diagnosis for each of the Ethernet ports can be accomplished through **ethLinkStatus** parameters available via SNMP.

Typical example of failure scenario

The indication of zero in **ethLinkStatus** for both ports indicates that the Ethernet cables are connected (Media presence). This condition of **LinkStatus** is also signaled via ETH1 LNK and ETH2 LNK LEDs in the controllers' frontal. If in this condition the indication of bad condition for both ports (ETH1 and ETH2) appears, it may be an indication that in some of the controllers the cables were inversely connected (ETH1 cable on ETH2 port, and ETH2 cable on ETH1 port).

Solution:

- Check if the connectors are properly fitted;
- Check the synchronism cables with failure indication as well as the network elements in case of a failure in the Ethernet ports.
- Check if the ports are connected to their networks (ETH1 port to network 1, ETH2 port to the network 2).

Primary fails before completing the synchronism.

This exceptional situation occurs when some procedure was not followed. It is signaled by the **RED_SYNC_STATUS** parameter (value 7: **WARNING: Updating Secondary Fail**) and by the Standby LED (see Standby LED behavior topic).

There is a chance to occur this failure only when a redundant pair is not yet with **Synchronized** in the **RED_SYNC_STATUS** parameter and then the current Primary is turned off. In such situation, when the redundancy is not available yet, the Secondary does not have conditions to take over the plant in a safe way. In this situation the Secondary keeps the same function and indicates this state as a safety condition.

Solution:

- In case the user knows that the Primary, which was turned off recently, has the complete configuration, set the Secondary to Hold, and then, turn on the Primary. Some seconds after that, remove the Secondary from Hold. The controller pair will synchronize, and only after they get **Synchronized** status and <none> in the **BAD CONDITIONS** parameters, failure simulations can be performed.
- In case the user does not trust the Primary configuration, follow the same procedure above mentioned, however download the configuration again.

Correction of an H1 cable failure

Cables failures in H1 segments which affect only one controller (Primary or Secondary) are signaled in the **BAD CONDITIONS** parameters allowing an immediate maintenance.

In case the failure occurs in an H1 cable segment which affects only the Primary controller, the redundancy will cover this failure, performing a switch over. In case the failure affects only the Secondary, it will not affect the process but even so it will be signaled by the **RED_SECONDARY_BAD_CONDITIONS** parameter allowing proactive maintenance.

During the maintenance, in case the H1 cable is reconnected at once, the noise inserted in the line will cause communication problems for some time, what is undesirable. To avoid this, the next procedure has to be followed.

- 1 – Set the controller, which was affected by the H1 cable failure, to Hold mode.
- 2 – Fix the H1 cable connection.
- 3 – Remove the controller from Hold mode. The controller will be automatically recognized by the Primary controller. As soon as the controller pair has **Synchronized** status and <none> in the **BAD_CONDITIONS** parameters, the redundancy will be fully available again.

Correction of bad conditions – Modbus

Check:

- If there are any cables failures in the paths related to the Modbus communication topology.
- The parameterization of the Modbus function blocks.
- If the converters/devices used in the Modbus communication topology are working properly.
- If the Modbus slave device is correctly configured and working.

Correction of bad conditions – Live List

Check:

- If the H1 cables have some connection problem or noise;
- Problems with line terminators (BT302): bad contact, missing or excess of BT302;
- Poor grounding;
- Water in the junction boxes or inside the devices;
- Transmitters with poor isolation;
- Transmitter digital card with some problem;

In case a deeper investigation is necessary, it is recommended to use the **FBView** software which is integral part of **SYSTEM302**. The **FBView** manual, in the Signal Quality and Live List topics, brings the necessary procedures.

Correction of bad conditions – Synchronism incompatibility

When executing the procedure Firmware update without process interruption in general will occur momentarily a situation where a controller has a firmware version and the other controller has another one. The following momentary situations may happen:

a) Secondary has a newer firmware version than the Primary (Upgrade): the synchronism is compatible and the controller pair synchronizes normally. That is, this scenario is perfectly supported. An exception may happen if the firmware versions are not compatible where the controller pair does not synchronize indicating this situation as “**Unable to Sync**” in the **BAD_CONDITIONS** parameters.

DF62/DF63/DF73/DF75: firmware version 1.x is not compatible with 2.x.

b) Secondary with an older firmware version than the Primary (Downgrade): the synchronism is not compatible and the controller pair does not synchronize indicating this situation as “**Unable to Sync**” in the **BAD_CONDITIONS** parameters. That is, this scenario is not supported in the redundancy context.

Solution for case b:

This scenario (downgrade) must be avoided. Once a plant is operating with a firmware version in the controllers, and for some reason the user desires to operate the plant with a previous firmware version the option is: with the plant stopped, change the firmware of all controllers (Primaries and Secondaries), and then follow the procedure of the Configuring for the first time a redundant system topic.

ADDING REDUNDANCY WITH REDUNDANT I/O MODULES

Introduction

To meet the requirements for fault tolerance, system availability and safety in the industrial process, the DFI302 controllers work with a Hot Standby redundancy strategy, where all the levels, including conventional Input and Outputs signals, may be configured and installed in a redundant manner.

In this strategy, the Primary and the Secondary controllers are connected to a set of redundant I/O scanners, which are dedicated to read and write the redundant I/O modules. The complete path from sensor until operation station is totally redundant. In case of one fault, an event will alarm the user, and the availability will be granted in a bumpless way.

IMPORTANT

The characteristics described in this section are supported by the DF62, DF63, DF73, DF75, DF89, DF95, and DF97 controllers at this time. Ask to check availability to other DFI302 controllers.

R-Series Ordering Codes

The following components are necessary to built R-Series I/O Redundancy in DFI302.

RACKS AND ACCESSORIES	
DF106	Master Rack - 6 slots for I/O redundancy
DF110 -1	Slave Rack - 10 slots for I/O redundancy - Terminal blocks
DF110 -2	Slave Rack - 10 slots for I/O redundancy – Interface cabling
DF109	Thin stub cable (0,40m)
DF119	Thick cable (1,0m) for DF106-DF109 or DF106-DF110
SCANNERS	
DF107	Master Scanner for I/O Redundancy
DF108	Slave Scanner for I/O Redundancy
I/O MODULES	
DF111	1 Group of 16 Redundant Digital Inputs 24 Vdc - Source
DF112	1 Group of 16 Redundant Digital Outputs 24 Vdc - Sink
DF113	1 Group of 8 Redundant Current Analog Inputs
DF114	1 Group of 8 Redundant Current Analog Outputs

The following components may complement R-Series IO Redundancy in DFI302.

CODE	DESCRIPTION
DF87	Power Supply for Backplane 20-30VDC (5A, Advanced Diagnostic)
DF0-R	Box Used In Empty Slots
ITF-CR-10 ITF-CR-15 ITF-CR-20 ITF-CR-25 ITF-CR-30 ITF-CR-35 ITF-CR-40 ITF-CR-45 ITF-CR-50	Interface cabling (1 m to 5 m)

ITF-DIG	Passive Interface Panel for 16 Digital Input and/or Output Module - DC Obs. The active components must be external connected
ITF-AN-IOR	Interface Panel for 8 Analog Input and/or Output Module Obs. Exclusive for R-Series

R-Series IO Redundant System Overview

In order to have a true Conventional I/O redundant system, all the parts and paths must be redundant. The hardware topology for Input and Output Redundant segments based on DFI302 controllers can be seen in the following figure. The system supports up to 16 pairs of R-Series I/O modules. This means 128 analog or 256 discrete I/O values, or a mix of them.

The R-series, redundant I/O module system, consists of 1 master rack and up to 4 slave racks; they are addressed as 0, 1, 2 and 3. It is possible for each slave rack add up to 4 pairs of I/O modules, addressed as pair 0, 1, 2 and 3. See the following figure.

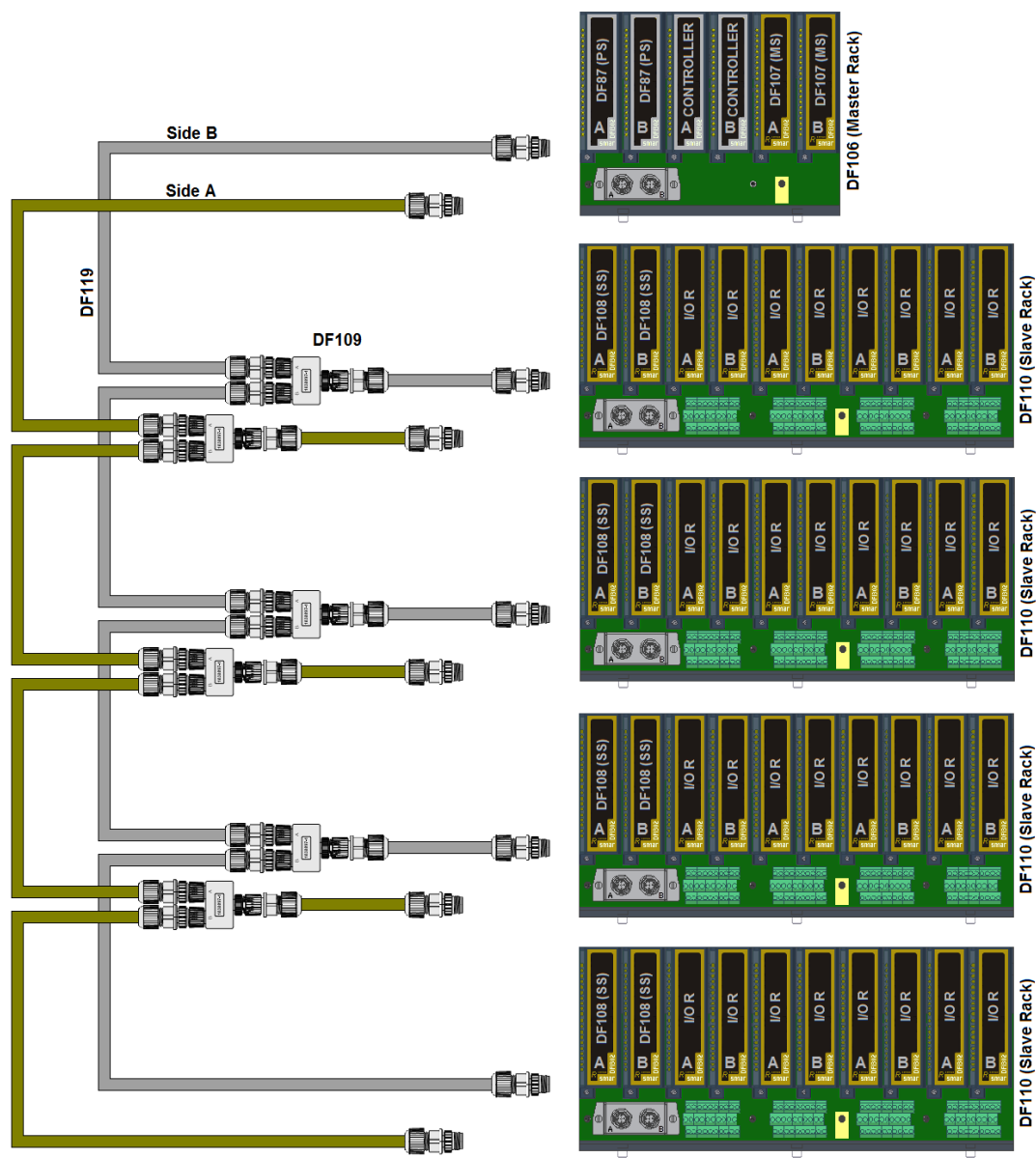


Figure 26. 1 - Conventional I/O Redundant Overview

IMPORTANT

For a better understanding, for each pair of modules - power supply, controller, scanner or I/O module, consider the left module (front view) as module A and the right module as module B. See the figure 26.1.

The controller A only has access to scanners A. The controller B only has access to scanners B.

The set controller A, master scanner A, DF109/DF119 cables and slave scanners A is conventionally called "Side A".

The set controller B, master scanner B, DF109/DF119 cables and slave scanners B is conventionally called "Side B".

The side whose controller is the current primary will be the Primary side.

The slave scanners (A and B) have access to all I/O modules (A and B) of the same rack, making the redundancy of I/O modules independent from redundancy of controllers.

See the Switchover Scenarios section for a complete understanding of redundancy in the "R-Series" system.

This architecture supports until one failure in any element of the system, which means that any hardware failure is covered by a second hardware working in a hot standby way.

NOTE

It is recommended that each identified fault has corrective action to make the system redundancy available again to cover new faults.

All I/O modules have the self-diagnostic feature to all points.

Scanners continually measure the health of each I/O modules to update the main controllers. The main controllers may use the status of the I/O modules in the control logic as safety interlock and provide the same rich information to HMI Stations. For further information about SNMP diagnostic items refer to Appendix B on this manual.

The diagnostic status for the whole system is available, as OPC and Simple Network Management Protocol (SNMP) parameters, available to HMI stations through its respective servers.

When maintenance is needed, the system permits hot swap of the modules, including power supplies, controllers, scanners and I/O modules.

The racks were built to avoid any kind of maintenance. No active component is mounted in this rack.

For further information about technical characteristics of R-Series modules, racks, cables and scanners refer to the Digital and Analog Input/Output Modules of DFI302 manual.

Switchover scenarios

The switchover of controllers/scanners (side) is independent from switchover of I/O modules, i.e., one do not interfere with the other.

Switchover of I/O modules

It occurs when there is a failure in the hardware of module A or a failure in the 24V external power supply of that module, when the module B becomes the primary. Once the failure in the module A is repaired, it becomes the primary again. That is, different from the behavior of the controller redundancy, for the redundant I/O modules, in the absence of faults in the module A, it will always be the preferred primary.

Each pair of redundant I/O modules checks the conditions from each other, working autonomously, regardless of the controller scan and ensures the switch in less than 100 microseconds. In case of failure of the primary I/O module, the secondary module takes over ensuring that the digital field devices remain powered, and the process remains undisturbed.

Switchover of controller/scanner

The controller switchover will occur when there is a failure in any part of the Primary side:

- Controller failure;
- Master scanner failure;
- Failure of DF109/DF119 cables;
- Slave scanner failure.

In this case when there is controller switchover, the side of the new primary controller will become the primary side. Once the faulty element is repaired, the current primary side will remain as primary, because for controllers redundancy there is no preferred primary.

Adding the R-Series I/O modules to a redundant system

The information of this section are only complementary, it does not intend to explain how the function blocks are instantiated, neither how the controllers and devices are configured. For further information on configuring control strategies refer to the user manual of **Syscon**. On developing logics for discrete control and configuration of redundant I/O modules refer to the **LogicView for FFB** manual.

Starting the Area

It is possible to create, or edit, an area from the **Studio302**. In the **Studio302** interface select **Areas**. A window will appear listing all areas of database.

To create a new area from the **Studio302**, left-click inside the **Areas** window, then choose **New Area**.

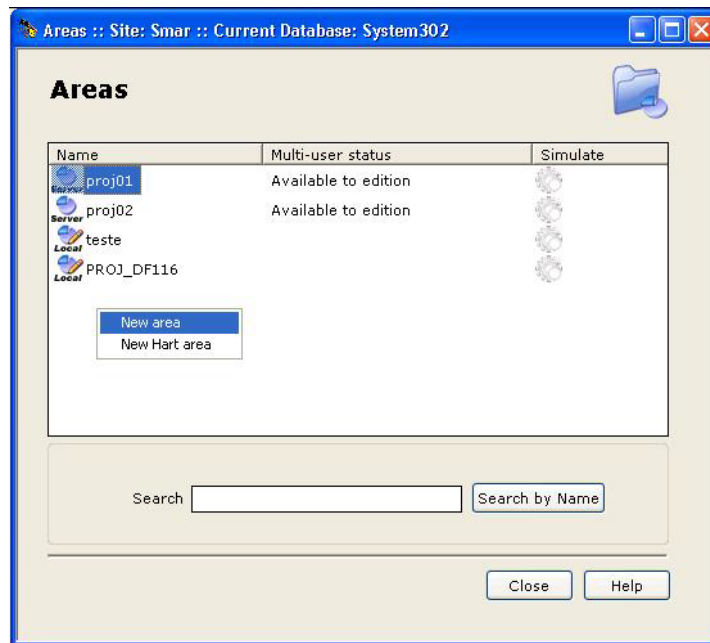


Figure 26. 2 - Creating a new area on Studio302

Another way to create a new area is from **Syscon**. Click the icon  in the **Studio302** toolbar.

To create a new area on **Syscon**, choose **File → New**, or through the toolbar, choose **New** button . The dialog box shows the **Area** options. Select **Predefined Area** as shown in next figure:

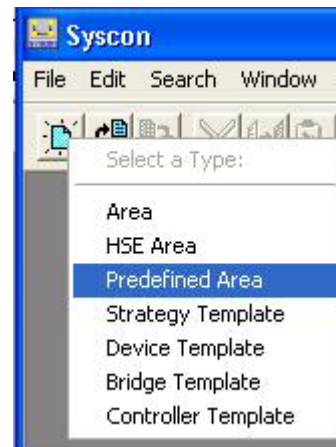


Figure 26. 3 - Options to create Syscon areas

After choosing the area name, a window will appear and the user has to choose the template type that will be used. In this case, the DF75 controller with FFB function block was chosen.

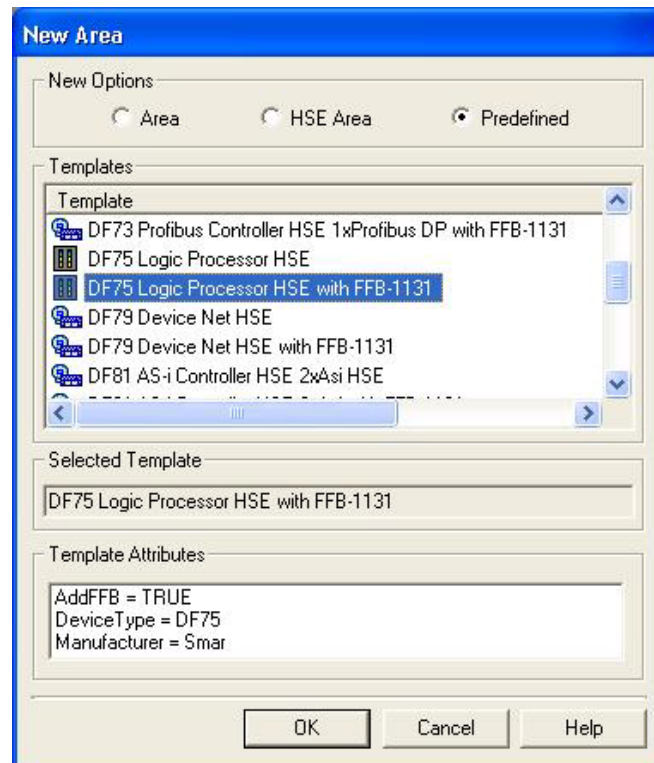


Figure 26. 4 – Template options on Syscon

Type the name for the area in the **Area Name** box, and click **Ok**. For this example, it chooses **PROJ_RSERIES**.

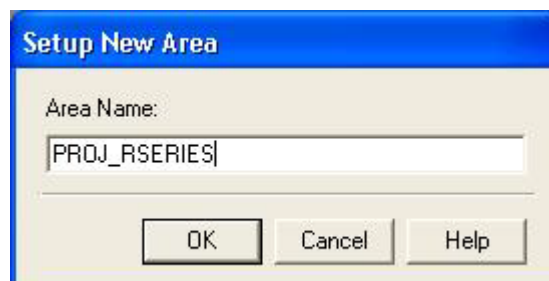


Figure 26. 5 – New area name

A new window will appear. This window has:

- **Application** – Logical Plant. To insert control strategies into this part.
- **Fieldbus Networks** – Physical Plant. To add devices and function blocks to the area into this part.

The area shall be as follows:

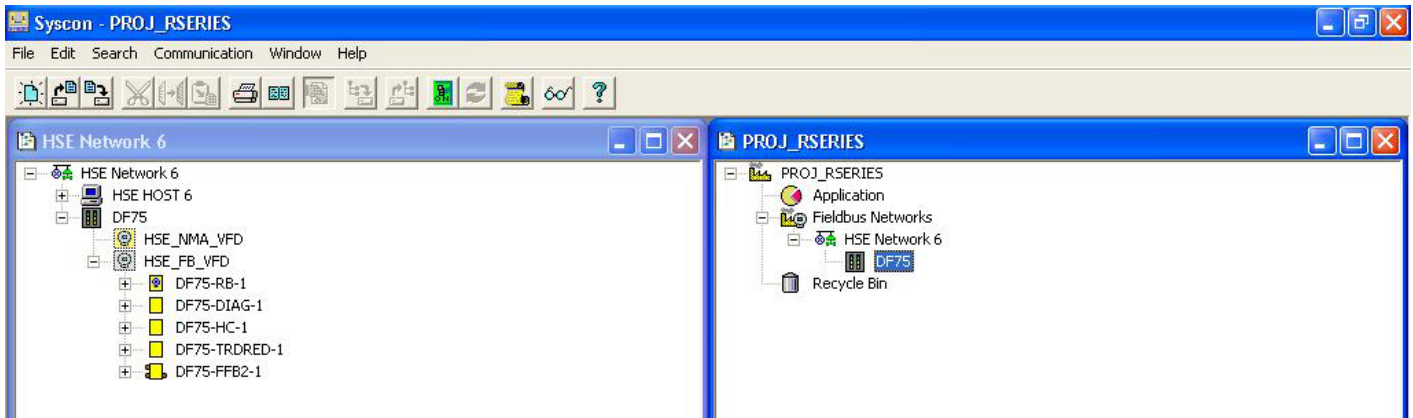


Figure 26.6 – PROJ_RSERIES area created on Syscon

IMPORTANT

The R-Series modules cannot be configured through the **HCT** function block on **Syscon**. The configuration of modules parameters must be done through the **LogicView for FFB**. For further information refer to its manual.

Creating a logic from the FFB function block

Right-click the **DF75-FFB2-1** block and choose **Edit Logic** option.

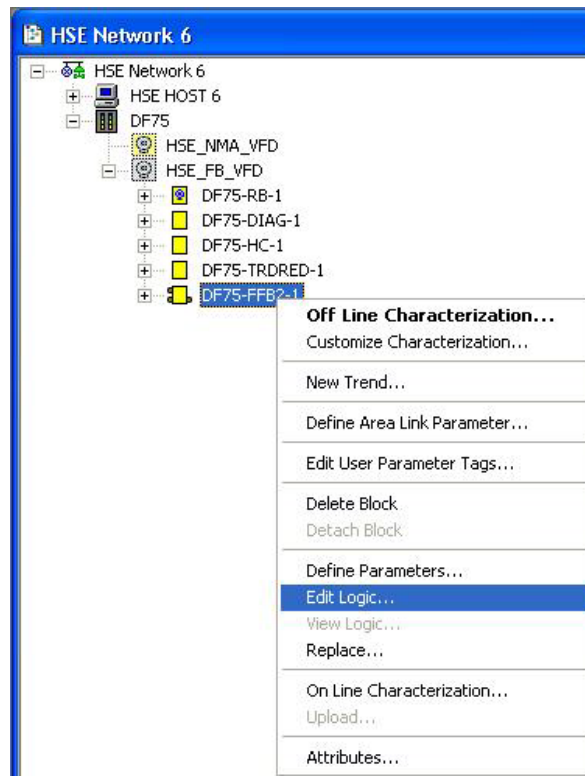


Figure 26.7 – Edit Logic option

The LogicView for FFB shall be as follows:

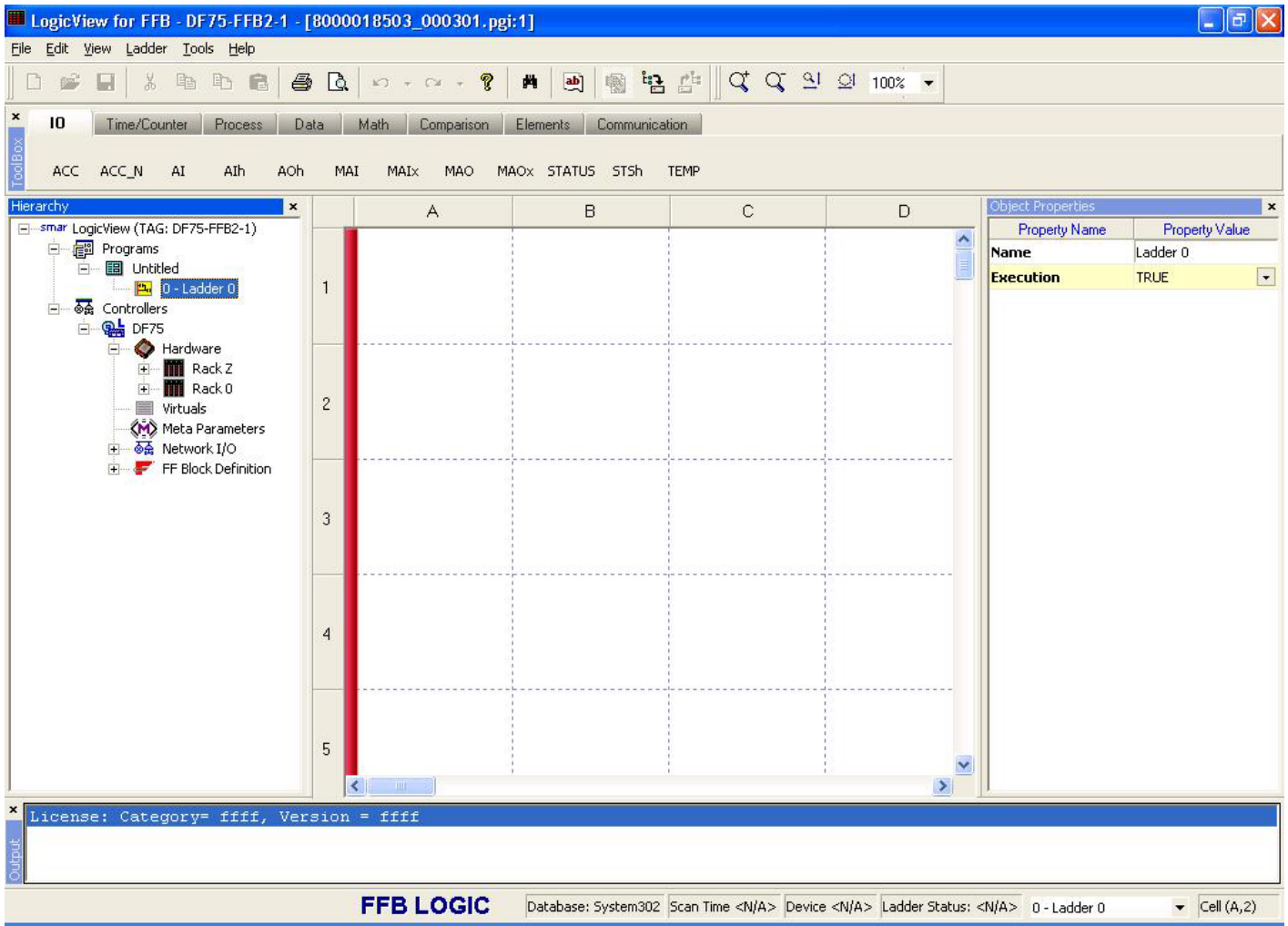


Figure 26. 8 – Editing the configuration on LogicView for FFB

Configuring the hardware on LogicView for FFB

In the Hierarchy window right-click **Hardware** and choose **Hardware Configuration** option.

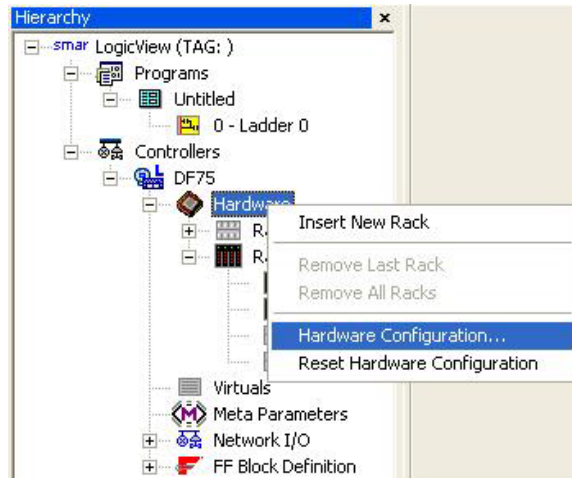


Figure 26. 9 – Configuring the hardware on LogicView for FFB

The following window will appear:

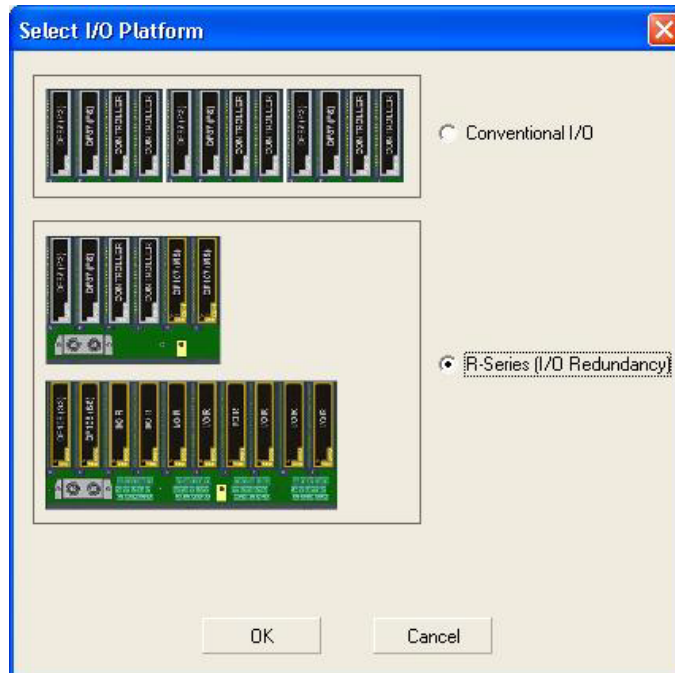


Figure 26. 10 – Selecting the I/O platform

Choose the **R-Series (I/O Redundancy)** option and click **Ok**. The following window will appear to choose the modules that form the system.

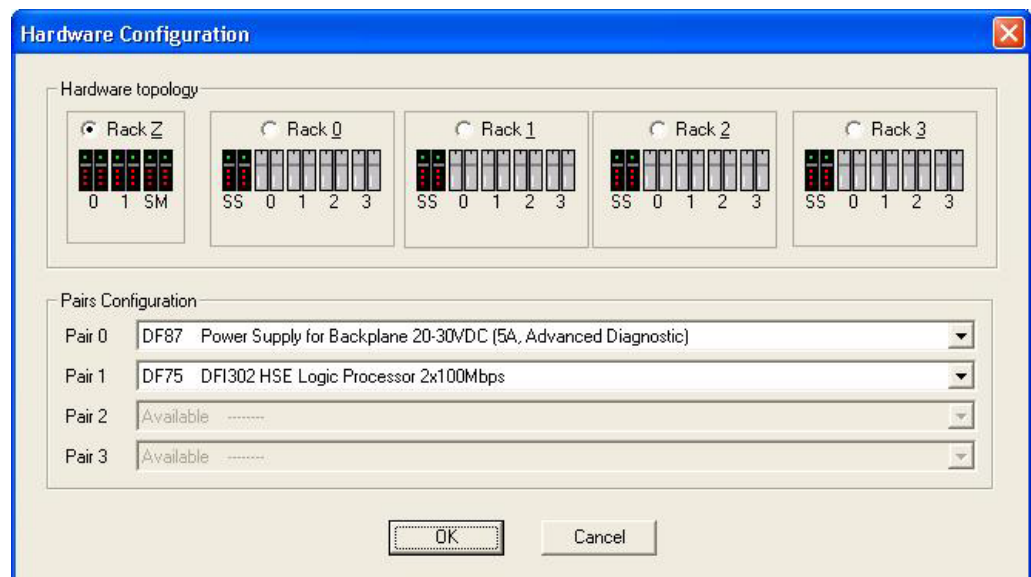


Figure 26. 11 – Configuring the hardware

The controller and the power supply are preconfigured. To choose the redundant I/O modules click the desired rack and the pairs can be defined. See the next figure:

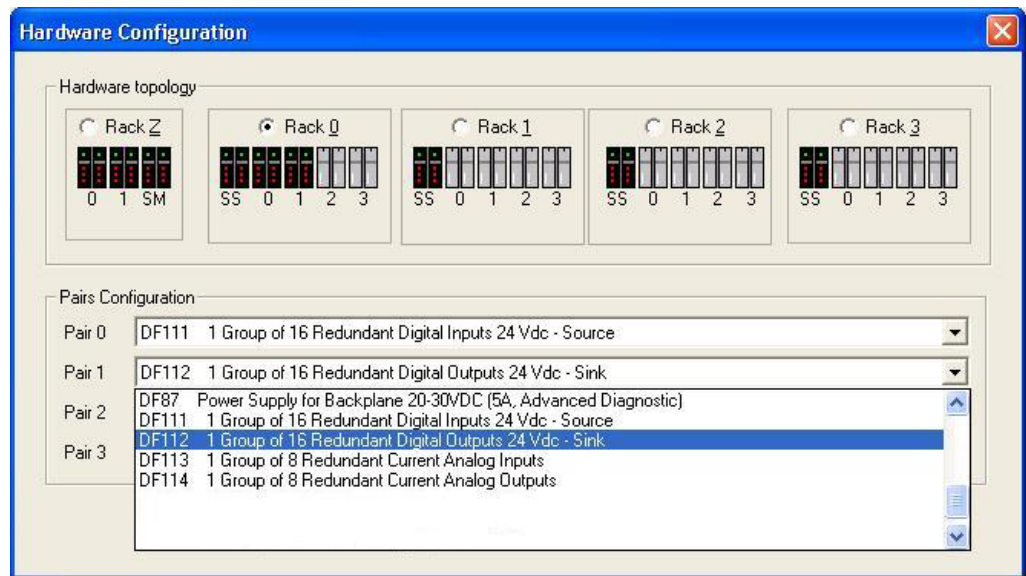


Figure 26. 12 – Choosing the redundant pairs

After choosing the redundant pairs click **OK**. The next step is to configure the logic in the ladder drawing area of **LogicView for FFB**.

Diagnostics via SNMP

The diagnostic information of the redundant I/O system is available through the Smar SNMP OPC Server for DFI302. For further information, please, refer to Appendix B on this manual.

Appendix A

smar	SRF – Service Request Form	
	DFI302 – Fieldbus Universal Bridge	Proposal N°:
COMPANY INFORMATION		
Company: _____		
Unit: _____		
Invoice: _____		
COMMERCIAL CONTACT		
Full Name: _____		
Phone: _____ Fax: _____		
E-mail: _____		
TECHNICAL CONTACT		
Full Name: _____		
Phone: _____ Extension: _____		
E-mail: _____		
EQUIPMENT DATA		
Model: _____		
Serial Number: _____		
PROCESS DATA		
Process Type (Ex. boiler control): _____		
Operation Time: _____		
Failure Date: _____		
FAILURE DESCRIPTION		
(Please, describe the failure. Can the error be reproduced? Is it repetitive?)		

OBSERVATIONS		

USER INFORMATION		
Company: _____		
Contact: _____		
Section: _____		
Title: _____ Signature: _____		
Phone: _____ Extension: _____		
E-mail: _____ Date: ____/____/____		
For warranty or non-warranty repair, please contact your representative. Further information about address and contacts can be found on www.smar.com/contactus.asp		

SNMP DIAGNOSTICS

Introduction

The SNMP (Simple Network Management Protocol) is a protocol that is part of the TCP/IP protocols and is available in most computers and in all manageable Ethernet network equipment.

The HSE DFI302 controllers of Smar (DF62, DF63, DF73, DF75, DF79, DF81, DF95, DF97 and DF100) bring embedded the SNMP protocol as an alternative for diagnosis.

The main benefits of using SNMP for diagnostic are:

- Lower cost to the controller CPU performance compared to the supervision of function blocks;
- In redundant controllers, possibility to separately access Primary and Secondary, providing what the Fieldbus Foundation normalization for redundancy calls "Diagnostic Visibility";
- Possibility of integration of controllers' SNMP items directly to information systems that support SNMP managers. In this way, is possible to treat the SNMP items of controllers and network equipment, e.g. routers, in an integrated way.

DIFFERENTIAL

Providing individual access to Primary and Secondary controllers, the SNMP enables proactive maintenance, as even physical faults or bad conditions on Secondary Controller can be identified or alarmed. Thus, it is easier to maintain the availability of the own redundancy of the controller pair.

Working principle

SNMP is based on MIBs (Management Information Base) that are manageable databases. These MIBs are implemented in the equipment and overall map their important items in a standardized way (read/write).

The SNMP protocol defines agents and managers. In **SYSTEM302**, each DFI302 controller module is an agent and the **SNMP OPC Server** is the manager. Thus, the **SNMP OPC Server** performs the reading/writing interface of the SNMP protocol for the OPC standard, widely used as an interface between the applications of supervision/maintenance/asset management and equipment.

In DFI302 controllers, SNMP can be used for the following purposes:

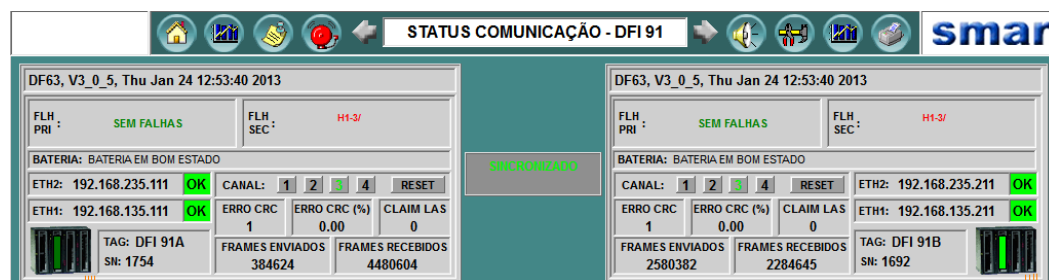
- Diagnostic of failures and bad conditions;
- Alarms and events monitoring.

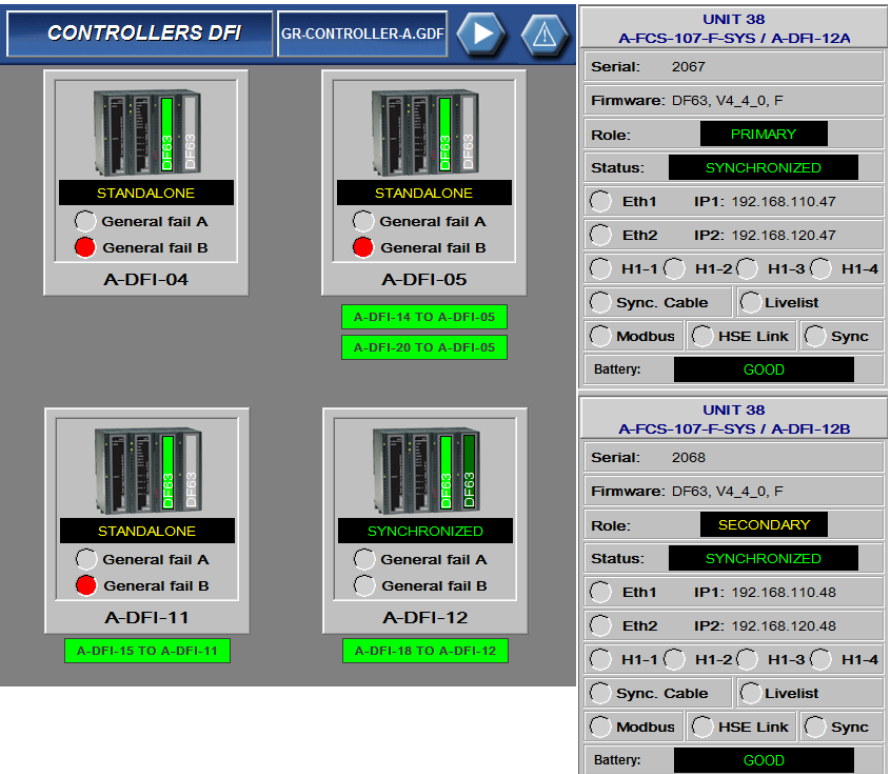
The steps for **SNMP OPC Server** configuration are in the user manual of **Studio302**, **Appendix A**, section **Configuring the SNMP OPC Server**.

After the **SNMP OPC Server** configuration, by any OPC client, such as **ProcessView** and **TagView**, it is possible to perform a search on the items available for that particular type of controller. With this you can choose the items you want to use the OPC client.

For the diagnostic information in this section, the OPC Server component that must be active is the **SNMP OPC Server for Smar DFI302** (smar.snmpopcserver.0).

See the following examples of supervisory screens (HMI) using SNMP items:





Integration and Network Security

For better security automation network, is a requirement that it is physically separated from the corporate LAN and the Internet.

If it is necessary to integrate a third-party SNMP network manager to external network elements, for example send alarms from the SNMP to pagers or SMS, you may need to have the automation network integrated to the corporate network. Therefore, it is recommended to use specialized firewalls to guarantee the automation network security. In this case it is recommended that the project be monitored by **Smar**.

Using network management tools via SNMP it is also possible access the diagnostic items of controllers directly by this protocol without using **SNMP OPC Server** in this case. It is thus possible that the management of network equipment, for example switches, and DFI302 controllers are integrated in the same environment.

Overview of diagnostic features for DFI302 controllers

Below are presented which diagnostic groups are available for each controller.

Controller	DeviceInfo	Ethernet	Fieldbus	ired	pwr	redundancy	serial	profibus	devicenet	asibus	whbus
DF62	X	X	X	X	X	X	X				
DF63	X	X	X	X	X	X	X				
DF75	X	X		X	X	X	X				
DF73	X	X				X	X	X			
DF95	X	X				X	X	X			
DF97	X	X				X	X	X			
DF79	X	X					X		X		
DF81	X	X					X			X	
DF89	X	X				X	X				
DF100	X	X					X				X

X: Feature available

Diagnostic groups common to all controllers

The following tables show the items available in DFI302 controllers. After these tables additional information for some items will be presented. For all tables the abbreviations have the following meaning: RO - Read-only and RW - Read/Write.

NOTE

For a better understanding of the items listed in the following tables, refer to the corresponding section prior to the feature. E.g.: section for controller redundancy.

General information of the controllers

ITEM	DESCRIPTION	VALID RANGE	SNMP SYNTAX	ACCESS
deviceInfo.deviceDescription	Device type, date and version of firmware.		DisplayString (SIZE(0..255))	RO
deviceInfo.serialNumber	Device serial number		INTEGER (0..65535)	RO
deviceInfo.upTime	Current up time in seconds		Counter32	RO
deviceInfo.numRestarts	Number of restarts		INTEGER (0..65535)	RO
deviceInfo.batteryStatus	Battery status	0: noBattery 1: maintenance 2: good	INTEGER	RO
deviceInfo.cpuLoad	CPU load in percentage		INTEGER(0..100)	RO

Table 1 – Diagnostic items available by SNMP in DFI302 controllers

Diagnostic/status of Ethernet ports

For all controllers $n = 2$ (two Ethernet ports), except for the DF62 controller wherein $n = 1$ (Ethernet port).

ITEM	DESCRIPTION	VALID RANGE	SNMP SYNTAX	ACCESS
ethernet.defaultGateway	Default gateway IP address		IpAddress	RO
ethernet.ethPort[n].ipAddress[n]	Current IP address for port [n]		IpAddress	RO
ethernet.ethPort[n].netmask[n]	Current netmask for port [n]		IpAddress	RO
ethernet.ethPort[n].ethResetStats[n]	This value controls when port [n] statistics should be reset. To reset the statistics, write a value greater than zero. The statistics will be reset and the item value will return to zero.		INTEGER (0..65535)	RW
ethernet.ethPort[n].numPacketsSent[n]	Number of packets sent by port [n]		Counter32	RO
ethernet.ethPort[n].numPacketsReceived[n]	Number of packets received by port [n]		Counter32	RO
ethernet.ethPort[n].numMulticastSent[n]	Number of multicast packets sent by port [n]		Counter32	RO
ethernet.ethPort[n].numMulticastReceived[n]	Number of multicast packets received by port [n]		Counter32	RO
ethernet.ethPort[n].numNoStackBuffers1	Number of times that a packet is		Counter32	RO

ITEM	DESCRIPTION	VALID RANGE	SNMP SYNTAX	ACCESS
	received by port [n] but the driver does not have a buffer for copying it			
ethernet.ethPort[n].numCRCErrors[n]	Number of packets received by port [n] with CRC error		Counter32	RO
ethernet.ethPort[n].numCollisions[n]	Number of collisions at port [n]		Counter32	RO
ethernet.ethPort[n].ethLinkStatus[n]	Link status for Ethernet port [n]	0: Unavailable 1: Link established	INTEGER (0..65535)	RO

Table 2 – Diagnostic items available by SNMP for Ethernet ports

Diagnostic/status of serial port

ITEM	DESCRIPTION	SNMP SYNTAX	ACCESS
serial.serResetStats	This values controls when serial statistics should be reset. To reset the statistics, write a value greater than zero. The statistics will be reset and the item value will return to zero.	INTEGER (0..65535)	RW
serial.numSerBytesSent	Number of bytes sent by serial port.	Counter32	RO
serial.numSerBytesReceived	Number of bytes received by serial port.	Counter32	RO
serial.numSerFCSErrors	Number of serial frames received with FCS error.	Counter32	RO

Table 3 –Diagnostic items available by SNMP for serial port

Specific diagnostic groups

For all tables the abbreviations have the following meaning: RO - Read-only and RW - Read/Write.

Diagnostic/status of the controller redundancy (DF62, DF63, DF73, DF75, DF89, DF95 and DF97)

ITEM	DESCRIPTION	VALID RANGE	SNMP SYNTAX	ACCESS
redundancy.role	Current role in redundancy	0: undefined 1: primary 2: secondary	INTEGER	RO
redundancy.status	Current redundancy synchronism status	0: undefined 1: standAlone 2: synchronizing 3: updatingSecondary 4: synchronized 5: roleConflict 6: SyncCableFail 7: updatingSecFail 8: reserved1 9: reserved2 10:reserved3 11: reserved4	INTEGER	RO
redundancy.primaryIP	IP address of the primary device		IpAddress	RO
redundancy.secondaryIP	IP address of the secondary device		IpAddress	RO
redundancy.primarySN	Serial number of the primary device		INTEGER (0..65535)	RO
redundancy.secondarySN	Serial number of the secondary device		INTEGER (0..65535)	RO
redundancy.primaryBadCond	Bad condition indications for primary device. Bit-string type. See beside the bits meaning.	0: Modbus 1: H1-1 (only DF62/DF63) 2: H1-2 (only DF62/DF63) 3: H1-3 (only DF62/DF63) 4: H1-4 (only DF62/DF63) 5: LiveList (only DF62/DF63) 6: Eth1 7: Reserved 8: Eth2 (not applicable to DF62) 9: Serial Sync Cable 10: Unable to Sync	INTEGER (0..10)	RO
redundancy.secondaryBadCond	Bad condition indications for secondary device. Bit-string	0: Modbus 1: H1-1 (only DF62/DF63)	INTEGER (0..10)	RO

ITEM	DESCRIPTION	VALID RANGE	SNMP SYNTAX	ACCESS
	type. See beside the bits meaning.	2: H1-2 (only DF62/DF63) 3: H1-3 (only DF62/DF63) 4: H1-4 (only DF62/DF63) 5: LiveList (only DF62/DF63) 6: Eth1 7: Reserved 8: Eth2 (not applicable to DF62) 9: Serial Sync Cable 10: Unable to Sync		
redundancy.numSwitchOvers	Number of switchovers performed		Counter32	RO
redundancy.numStandAlones	Number of StandAlone occurrences		Counter32	RO
redundancy.redResetStats	This value controls when redundancy statistics should be reset. To reset the statistics, write a value greater than zero. The statistics will be reset and the item value will return to zero.		INTEGER (0..65535)	RW
redundancy.statsSince	Date and time redundancy statistics were reset.		DisplayString (SIZE(0..255))	RO
redundancy.localBadCond	Bad condition indications for specified device. Bit-string type. See beside the bits meaning.	0: Modbus 1: H1-1 (only DF62/DF63) 2: H1-2 (only DF62/DF63) 3: H1-3 (only DF62/DF63) 4: H1-4 (only DF62/DF63) 5: LiveList (only DF62/DF63) 6: Eth1 7: Reserved 8: Eth2 (not applicable to DF62) 9: Serial Sync Cable 10: Unable to Sync	INTEGER (0..10)	RO

Table 4 – Diagnostic items available by SNMP for redundant controllers

Diagnostic/status of the Foundation H1 ports (DF62 and DF63)

For the following items of H1 Fieldbus ports, DF62 and DF63 have n ranging from 1 to 4 (4 H1 ports).

ITEM	DESCRIPTION	SNMP SYNTAX	ACCESS
fieldbus.applicationTimer	Current time of application in 1/32 ms since Jan, 1 st 1972	INTEGER64 (0..18446744073709551615)	
fieldbus.fbPort[n].portID[n]	Fieldbus port [n] identification	INTEGER (0..65535)	RO
fieldbus.fbPort[n].fbResetStats[n]	This value controls when port [n] statistics should be reset. To reset the statistics, write a value greater than zero. The statistics will be reset and the item value will return to zero.	INTEGER (0..65535)	RW
fieldbus.fbPort[n].numPDUSent[n]	Number of fieldbus frames sent by port [n]	Counter32	RO
fieldbus.fbPort[n].numPDUReceived[n]	Number of fieldbus frames received by port[n]	Counter32	RO
fieldbus.fbPort[n].numFCSerrors[n]	Number of fieldbus frames received with FCS error by port [n]	Counter32	RO
fieldbus.fbPort[n].numClaimLAS[n]	Number of claim LAS at port [n]	Counter32	RO

Table 5 –Diagnostic items available by SNMP for H1 ports

Diagnostic of redundant I/O modules system (DF62, DF63 and DF75)

The redundant I/O module system, **R-series**, consists of 1 master rack and up to 4 slave racks, addressed as 0, 1, 2 and 3, it being possible for each slave rack add up to 4 pairs of I/O modules addressed as 0, 1, 2 and 3. Refer to the section **Adding Redundancy with Redundant I/O Modules** for further information, such as conditions for switching of controller or I/O modules.

For a better understanding, for each pair of modules (power supply, controller, scanner and I/O module), consider the left module (front view) as module A, and the right module as module B.

ITEM	DESCRIPTION	VALID RANGE	SNMP SYNTAX	ACCESS
iored.msStatus	Detailed status for master scanner module. See note 1.	See table 8	INTEGER	RO
iored.ss0Status	Detailed status for slave scanner module with address 0. See note 1.	See table 9	INTEGER	RO
iored.ss1Status	Detailed status for slave scanner module with address 1. See note 1.			RO
iored.ss2Status	Detailed status for slave scanner module with address 2. See note 1.			RO
iored.ss3Status	Detailed status for slave scanner module with address 3. See note 1.			RO
iored.module00Astatus	Detailed status for I/O module 00A. See note 2. 00A → 0: Rack 0, 0: Pair 0, A: left module.	See table 10	INTEGER	RO
...				
Same rule up to the last I/O module				
iored.module33Bstatus	Detailed status for I/O module 33B. See note 2. 33B → 3: Rack 3, 3: Pair 3, B: right module.			RO
iored.pair00.point000	Detailed status for point 0 – pair 00. See note 3. 000 → 0: Rack 0, 0: Pair 0, 0: Point 0	See tables 11 and 12	INTEGER	RO
...				
iored.pair00.point0015	Detailed status for point 15 – pair 00. See note 3. 0015 → 0: Rack 0, 0: Pair 0, 15: Point 15			RO
...				
Same rule up to the last pair				
iored.pair33.point330	Detailed status for point 0 – pair 33. See note 3. 330 → 3: Rack 3, 3: Pair 3, 0: Point 0			RO
...				
iored.pair33.point3315	Detailed status for point 15 – pair 33. See note 3. 3315 → 3: Rack 3, 3: Pair 3, 15: Point 15			RO

Table 6 – Diagnostic items available by SNMP for redundant I/O modules

NOTES
1 - When this value is provided by module A, only the controller A will have access. The same is valid for module B and controller B.
2 - Unlike the scanners status, for I/O modules, any controller (A or B) can get the status of the modules (A and B) simultaneously.
3 - For modules containing only 8 points, should be considered only the first eight point status.

Details of status bits for I/O redundant modules

All status items of the I/O redundant system are based on self-diagnosis embedded in all modules. If you need to evaluate details of the modules status, it is possible to interpret each bit for the diagnosis of specific fault conditions.

For tables Master Scanner Status, Slave Scanner Status and I/O Module Status, the bit 7 is the most significant (MSB). This bit is a consolidation of the status of all lower bits, reflecting the overall status of the module. Being this bit the most significant; it is possible to have the overall status module just checking the ranges as shown below. Thus, it is not necessary to use bit masks.

VALUE	STATUS	DESCRIPTION
0 to 127	BAD	Module with failure.
128 to 254	WARNING	Module without failure, but with some warning*.
255	GOOD	Module without failure and without warning*.

Table 7 – Range of Status Values

NOTE

* An alert (warning) does not interfere immediately in the module's functionality, but in the occurrence of other faults the redundant system can be compromised. It is recommended that each warning is identified, reading each bit of the item, and that corrective action is taken.

<i>iored.msStatus - Detailed status for Master Scanner module</i>				
BIT	Identification	Description	Value	
			1	0
0	Current in PWR_0	Overcurrent indication in the first + 5V power input from the power supply A (master rack)	Current OK	Current higher than normal
1	Current in PWR_1	Overcurrent indication in the second + 5V power input from the power supply B (master rack)	Current OK	Current higher than normal
2	Voltage in PWR_0	Voltage level indication in the first + 5V power input from the power supply A (master rack)	Normal voltage	Low voltage
3	Voltage in PWR_1	Voltage level indication in the second + 5V power input from the power supply B (master rack)	Normal voltage	Low voltage
4	Partner MS	Indication of redundant partner status	Partner master scanner OK	Partner master scanner with failure
5	Controller MS	Indication of controller status	Controller OK	Controller with failure
6	Reserved	Reserved	Always 1	The value never goes to 0
7	Module Status	Bit that consolidates the module status	Module Ok	Module with failure

Table 8 – Master Scanner Status

<i>iored.ss[n]Status - Detailed status for Slave Scanner module</i>				
BIT	Identification	Description	Value	
			1	0
0	Current in PWR_0	Overcurrent indication in the first + 5V power input from the power supply A (master rack)	Current OK	Current higher than normal
1	Current in PWR_1	Overcurrent indication in the second + 5V power input from the power supply B (master rack)	Current OK	Current higher than normal
2	Voltage in PWR_0	Voltage level indication in the first + 5V power input from the power supply A (master rack)	Normal voltage	Low voltage
3	Voltage in PWR_1	Voltage level indication in the second + 5V power input from the power supply B (master rack)	Normal voltage	Low voltage
4	Partner SS	Indication of redundant partner status	Partner slave scanner OK	Partner slave scanner with failure
5	Reserved	Reserved	Always 1	The value never goes to 0
6	Reserved	Reserved	Always 1	The value never goes to 0
7	Module Status	Bit that consolidates the module status	Module Ok	Module with failure

Table 9 – Slave Scanner Status

<i>iored.moduleXXYstatus - Detailed status for I/O module</i>				
BIT	Identification	Description	Value	
			1	0
0	HS_PWR_0_5V	Voltage level indication in the first + 5V power input from the power supply A (master rack)	Normal voltage	Low voltage
1	HS_PWR_1_5V	Voltage level indication in the first + 5V power input from the power supply B (master rack)	Normal voltage	Low voltage
2	LS_INT_VCC	Voltage level indication of the I/O module internal power supply provided by the external power supply (VEXT).	Normal voltage	Low/High voltage
3	LS_EXT_VCC	Voltage level indication of the external power supply (VEXT).	Normal voltage	Low/High voltage
4	Reserved	Reserved	Always 1	The value never goes to 0
5	Reserved	Reserved	Always 1	The value never goes to 0
6	IO_STATUS	Bit that consolidates the status of the I/O module points (AND logic of all module points).	Points with failure	Points with failure
7	Module Status	Bit that consolidates the module status	Module Ok	Module with failure

Table 10 – I/O module status

<i>iored.pairXX.pointYY - Detailed status of point for digital I/O module</i>		
Status	Description	Value
GOOD	Point Ok	255
BAD	Error in self-diagnosis. Point with failure.	3
	Unavailable I/O module	0

Table 11 – Point status (Digital I/O module)

<i>iored.pairXX.pointYY - Detailed status of point for analog I/O module</i>		
Status	Description	Value
GOOD	Point Ok	255
BAD	Error in self-diagnosis. It was not possible to assess the point.	3
	Error in self-diagnosis. Reading the point value is below the allowed tolerance.	2
	Error in self-diagnosis. Reading the point value is above the allowed tolerance.	1
	Unavailable I/O module	0

Table 12 - Point status (Analog I/O module)

Diagnostic/status of the DF87 power supply and DF93 racks (DF62, DF63 and DF75)

ITEM	DESCRIPTION	VALID RANGE	SNMP SYNTAX	ACCESS
pwr.rackZ	Voltage status of rack Z (rack 16)	0: Voltage Ok 1: Low voltage 2: Unavailable status	INTEGER	RO
pwr.rack0	Voltage status of rack 0			RO
...				RO
pwr.rack15	Voltage status of rack 15			RO
pwr.pwrsupply[n].rackPS[n]	Power Supply Location [n] - rack	0 to 3 17 – power supply does not configured	INTEGER	RO
pwr.pwrsupply[n].slotPS[n]	Power Supply Location [n] - slot	0 to 3 4 – power supply does not configured	INTEGER	RO
pwr.pwrsupply[n].statusPS[n]	Overall status of the power supply	<128: Failure (bad) 128 to 254: Ok (good), with some alert (warning) 255: OK, without failure or alerts (warnings)	INTEGER	RO
pwr.pwrsupply[n].vinPS[n]	Input voltage status	255 – Ok (good)	INTEGER	RO
pwr.pwrsupply[n].voutPS[n]	Output voltage status	130 – Warning Hi		RO
pwr.pwrsupply[n].ioutPS[n]	Input current status	129 – Warning Low		RO
pwr.pwrsupply[n].ripplePS[n]	Ripple voltage status	124 – Fail Low		RO
pwr.pwrsupply[n].tempPS[n]	Status of the power supply temperature	120 – Fail Hi 0 – Power supply is unavailable or it is not powered		RO

Table 13 – Diagnostic items available by SNMP for DF87 power supplies and DF93 racks

IMPORTANT
<p>For diagnostic information of DF87 power supply there are some restrictions:</p> <ul style="list-style-type: none"> - Only two power supplies can be diagnosed ($n = 1$ or $n = 2$). To $3 \leq n \leq 6$ - future use. - For DF92 and DF106 racks the default positions are: Rack16Slot0 and Rack16Slot1 - For DF93 racks, the default position of the power supply is Rack0Slot0. - For DF1A racks, the default position of the power supply is Rack0Slot0. <p>If there are more than two DF87 power supplies in the system, the diagnostic is only possible through the DIAG1, DIAG2 and DIAG3 LEDs in the module front. Refer to the section Adding power supplies for further information.</p>

SNMP items - Additional Information

Some SNMP items require a better understanding as to the valid range values and possible user actions. For these items here additional information is presented.

Battery status

deviceInfo.batteryStatus

- 0: noBattery: DIP switch number 1, on the back of the module, in OFF position (check) or discharged battery (required replacement as soon as possible);
- 1: maintenance: Battery end of life (required replacement as soon as possible);
- 2: good: battery in good condition.

Redundancy

Most redundancy items follow the same principles of the redundancy transducer block (**TRDRED**) parameters. For details, refer to the section **Adding Redundancy to DFI302 HSE Controllers** of this manual.

The items started in "num" are counters and represent statistical information. There are also items to perform the restart or reset of the counters listed at the moment that the user wants. Below are detailed some counters not available in **TRDRED** block.

redundancy.numSwitchOvers

Indicates the number of times the controller pair exchanged its function (Primary ⇔ Secondary) due to some failure in any of the interfaces (H1, Ethernet, Modbus or R-Series Scanners/Cable).

redundancy.numStandAlones

Indicates the number of times the controller was in StandAlone state (without redundancy). It can also be understood as the number of times that the controller has taken the primary function due to overall failure of the partner controller (shutdown or general hardware failure).

Fieldbus

fieldbus.fbPort[n].numFCSErrors[n]

Indicates the number of FCS errors (Frame Check Sequence) at the reception of port [n].

If this item is increasing in a short period of time, refer to the **FBView** manual, section 3.4.1 **Signal Quality**, to assess the possible problems on the bus.

fieldbus.fbPort[n].numClaimLAS[n]

Indicates the number of frames "Claim LAS" type on the bus of port [n]. It is expected that these commands occur only at equipment startup and when occur failure/shutdown of the current master LAS. The intermittent increase of this item the system regime condition can also indicate master LAS hardware failure or installation problems on the bus. For details refer to **FBView** manual, section 3.4.3 **Link Master - LAS**.

DF63-RTU

Introduction

The **DF63-RTU** controller is a dedicated version of the **DF63**, specially designed to operate in remote environments, where typically the communication channel requires special handling. This **DF63-RTU** controller incorporates the DNP3 protocol and does not support the controller redundancy feature.

The RTUs are connected to the MTU (Master Terminal Unit, located in the control center) via two main communication protocols. Through DNP3 protocol and using PPP router, it makes possible to reduce of bandwidth usage during normal operation of the whole system. The DNP3 protocol guarantees data storage during communication failures between RTUs and MTU, including a proper timestamp for all transmitted data. Through the HSE protocol is possible to diagnose and configure the system remotely from the MTU.

A typical use of **DF63-RTU** is wireless network via Tetra radio, using IP/PPP protocol via router, such as **DF125**.

The field instruments will be connected via FOUNDATION fieldbus H1 channel to the **DF63-RTU** controller, which will then be able to analyze and transfer all required data to the MTU for supervision. The configuration and maintenance should be done by **SYSTEM302**.

Configuring RTUs

The **System302 ServerManager** configures the parameters for the plant network communication. It is executed from the **Studio302** toolbar.

Click the button **System302 ServerManager** on the **Studio302** toolbar and the following dialog box will open:



Figure C.1 – System302 ServerManager

The option **Network** allows configuring the network interface cards used by the **System302 ServerManager**, and also redundancy, advanced parameters for HSE network, specific parameters for SNTP and parameters for the case of a system composed by RTUs. For further details about **System302 ServerManager**, please, refer to **Studio302** manual.

Choose **Network** and then select the **RTUs** tab. The following window will appear. Select **Enable RTU Mode** option to enable the RTU system parameters.

Figure C.2 – System302 ServerManager – RTUs tab

Enable RTU Mode:

Use this checkbox to enable or disable RTU mode. For RTU mode, it is necessary to have a router with NAT capabilities (Network Address Translation) between the workstations and the RTU. It is assumed that the RTU is connected to this router using the first network interface (ETH1). The second network interface (ETH2) will only be used for engineering/maintenance through a network cable between the workstation and RTU. After connecting this network cable all HSE communication through ETH1 will stop. When the cross cable is disconnected, the HSE communication via ETH1 will be enabled again.

HSE Persistency:

- **Read only (locked)** – Blocks writing in database of HSE devices (persistency);
- **Writable (unlocked)** – Allows writing in database of HSE devices (persistency);

Workstation Role:

- **Engineering/Maintenance (ETH2)** – In this role the second network interface (ETH2) is used for engineering and maintenance operations through a network cable between the workstations and RTU.
- **Operation (ETH1)** – In this role the first network interface (ETH1) is used for operation. It is necessary the NAT function of the router between the workstation and RTU.

Network Settings:

- **RTU Ethernet 1** – IP Address used in RTU for its first network port.
- **RTU Ethernet 2** – IP Address used in RTU for its second network port.
- **Router IP** – IP Address used on router with NAT function for its WAN network port. Access the DF125 webserver, when using this router, and see this information in **Diagnostics→Network Status, MODEM/WAN Status** section.

IMPORTANT

There is a parameter directly related to OPC operations by **LogicView for FFB** that is changed from the options defined in this **RTUs** tab.

OPC Timeout - is changed by default from the choices made on this tab, namely:

- 1500 milliseconds if RTU mode is disabled or in Engineering / Maintenance mode.
- 15000 milliseconds or value customized by **LogicView for FFB** when in operation mode.

From this chosen scenario, in case any change is made, please use **Apply** button so that new values are applied. Click **Apply All** to confirm the alterations and exit the **System302 Server Manager**. The changes will take effect after restarting the **System302 Server Manager**.

DNP3 Blocks

DNP3CF – Configuration Block

Overview

This block allows configuration of several communication parameters of the DNP3 protocol.

Description

In this block several communication parameters between **DFI302** and DNP3 Master can be set. Ethernet and serial (EIA-232) medias are currently supported.

The mode block controls the data update from Fieldbus to DNP3. The mode in automatic indicates that the block is in normal operation exchanging data from fieldbus to DNP3. The mode in out-of-service stops the data exchange.

DNP3 Address

Using the parameter **DNP_SLAVE_ID**, the user must attribute an unique DNP3 address to the **DFI302**, avoiding conflict with any other device in the DNP3 network which it is connected. The default value for this parameter is 4.

DNP Communication Media

DFI302 communication may use Ethernet (TCP or UDP) and serial (EIA-232) as media. The behavior is controlled by **DNP_MEDIA** parameter. Each media type has its corresponding additional setup parameters, described below.

Serial Media

Rate of Transference

It is possible to select the baud rate of data in the serial port through the **BAUD_RATE** parameter. It allows the selection among the following baud rates:

- 0:100 bps
- 1:300 bps
- 2:600 bps
- 3:1200 bps
- 4:2400 bps
- 5:4800 bps
- 6:9600 bps (default)
- 7:19200 bps
- 8:38400 bps
- 9:57600 bps
- 10:115200 bps

Parity

PARITY parameter defines the type of parity to the serial port.

- 0: No parity
- 1: Even Parity
- 2: Odd parity

Serial or Ethernet Medium

Timeout and Number of Retransmissions

Timeout parameter (**TIME_OUT**) is the time waited for answer from a slave after a message has been sent to the serial port or to the Ethernet. The default value is 1000 ms. This parameter is related to the parameter **NUMBER_RETRANSMISSIONS**.

NUMBER_RETRANSMISSIONS parameter is the number of times the **DFI302** will retry to establish communication with the slave device after not getting a reply. The time waited for this answer is set by the **TIME_OUT** parameter. User may select a value in the range 0 to 255 to this parameter. The default value is 1.

Diagnosis using DNP3

The **DFI302** allows supervising many diagnoses of controllers and Fieldbus devices through DNP3 protocol. The table below has the DNP3 addresses (Group, Variation, Point) for each diagnosis.

By default, the diagnostic parameters are disabled. The **DIAG_CLASS** parameter allows enabling and configuring the class of all diagnostic parameters. All parameters of the table below follow the same class specified in **DIAG_CLASS**.

If the event class is chosen (Class 1, 2 or 3), the diagnostic updating follows the rule of events generation described in "Event Rule" column of the table below. For some parameters any value change automatically generates an event. Some parameters the event is generated for a timer adjusted by **DIAG_TIME_EVENT** parameter.

The status flag of DNP3 points when the points are enabled always will be "Online" (Bit 1 is on) for all points of diagnostic.

DNP3 Group	DNP3 Variation	DNP3 Point	Data Type	Client Access	Parameter	Event Rule	Description
30	1	384	Uns32	RO	serialNumber	Any changes	Device serial number.
30	1	385	Uns32	RO	upTime	DIAG_TIME_EVENT	Current up time in seconds.
30	1	386	Uns32	RO	numRestarts	Any changes	Number of restarts.
30	1	387	Uns32	RO	batteryStatus	Any changes	Battery status. Valid range: 0 – no battery 1 – maintenance 2 – Good
30	1	388	Uns32	RO	cpuLoad	Any changes	CPU load in percentage.
30	1	389	Uns32	RO	redundancy. Role	Any changes	Current role in redundancy. Valid Range: Undefined (0) Primary (1) Secondary (2)
30	1	390	Uns32	RO	redundancy. Status	Any changes	Current redundancy synchronism status. Valid range: undefined (0) standalone (1) synchronizing (2) updatingSecondary (3) synchronized (4) roleConflict (5) syncCableFail (6) updatingSecFail (7)
30	1	391	Uns32	RO	fieldbus.numP DUSent[1]	DIAG_TIME_EVENT	Number of fieldbus frames sent at port [n].
30	1	392	Uns32	RO	fieldbus.numP DUReceived[1]	DIAG_TIME_EVENT	Number of fieldbus frames received at port[n]
30	1	393	Uns32	RO	fieldbus.numF CSErrors[1]	DIAG_TIME_EVENT	Number of fieldbus frames received with FCS error at port [n].
30	1	394	Uns32	RO	fieldbus.numCl aimLAS[1]	DIAG_TIME_EVENT	Number of claim LAS at port [n].
30	1	395	Uns32	RO	fieldbus.numP DUSent[2]	DIAG_TIME_EVENT	Number of fieldbus frames sent at port [n].
30	1	396	Uns32	RO	fieldbus.numP DUReceived[2]	DIAG_TIME_EVENT	Number of fieldbus frames received at port[n]
30	1	397	Uns32	RO	fieldbus.numF CSErrors[2]	DIAG_TIME_EVENT	Number of fieldbus frames received with FCS error at port [n].

DNP3 Group	DNP3 Variation	DNP3 Point	Data Type	Client Access	Parameter	Event Rule	Description
30	1	398	Uns32	RO	fieldbus.numClaimLAS[2]	DIAG_TIME_EVENT	Number of claim LAS at port [n].
30	1	399	Uns32	RO	fieldbus.numPDU Sent[3]	DIAG_TIME_EVENT	Number of fieldbus frames sent at port [n].
30	1	400	Uns32	RO	fieldbus.numPDU Received[3]	DIAG_TIME_EVENT	Number of fieldbus frames received at port [n].
30	1	401	Uns32	RO	fieldbus.numFCS Errors[3]	DIAG_TIME_EVENT	Number of fieldbus frames received with FCS error at port [n].
30	1	402	Uns32	RO	fieldbus.numClaimLAS[3]	DIAG_TIME_EVENT	Number of claim LAS at port [n].
30	1	403	Uns32	RO	fieldbus.numPDU Sent[4]	DIAG_TIME_EVENT	Number of fieldbus frames sent at port [n].
30	1	404	Uns32	RO	fieldbus.numPDU Received[4]	DIAG_TIME_EVENT	Number of fieldbus frames received at port [n].
30	1	405	Uns32	RO	fieldbus.numFCS Errors[4]	DIAG_TIME_EVENT	Number of fieldbus frames received with FCS error at port [n].
30	1	406	Uns32	RO	fieldbus.numClaimLAS[4]	DIAG_TIME_EVENT	Number of claim LAS at port [n].

Table 1 – DNP3 Addresses (Group, Variation, Point) for diagnosis of controllers and fieldbus devices

Optionally it is possible to reset some internal counters or then the events buffer through DNP3 writing commands. In the table below all these parameters are listed. All parameters have the “Disable” class and are automatically zero after receiving a value 1.

DNP3 Group	DNP3 Variation	DNP3 Point	Data type	Client Access	Parameter	Description
40	1	384	Uns32	RW	fieldbus.fbResetStats[1]	This value controls when port [n] statistics should be reset. To reset the statistics, write a value greater than zero. The statistics will be reset, and the item value will return to zero.
40	1	385	Uns32	RW	fieldbus.fbResetStats[2]	This value controls when port [n] statistics should be reset. To reset the statistics, write a value greater than zero. The statistics will be reset, and the item value will return to zero.
40	1	386	Uns32	RW	fieldbus.fbResetStats[3]	This value controls when port [n] statistics should be reset. To reset the statistics, write a value greater than zero. The statistics will be reset, and the item value will return to zero.
40	1	387	Uns32	RW	fieldbus.fbResetStats[4]	This value controls when port [n] statistics should be reset. To reset the statistics, write a value greater than zero. The statistics will be reset, and the item value will return to zero.
40	1	388	Uns32	RW	Clear Buffer	Clear all Discrete and Analog Events from the buffer.

Table 2 – Parameters for reset of internal counters and events buffer

BLOCK_ERR

The **BLOCK_ERR** of the **DNP3CF** block will reflect the following causes:

- Out of Service: When the block is in O/S mode.

Supported Modes

O/S and Auto.

Parameters

Idx	Parameter Name	Data Type	Valid Range	Default Value	Store	Units	Description
1	ST_REV	UNS16		0	S		
2	TAG_DESC	OCT(32)		Spaces	S		
3	STRATEGY	UNS16		0	S		
4	ALERT_KEY	UNS8	1 to 255	0	S		
5	MODE_BLK	DS_69	OOS, AUTO	OOS	S		
6	BLOCK_ERR	Bitstr(2)			D		
7	DNP_MEDIA	UNS8	0:Serial, 1:TCP 2:UDP	1	S		Define the media type of DNP Channel.
8	DNP_SLAVE_ID	uns16	0 to 65534	4	S		This is the DNP address for the module.
9	DNP_MASTER_ID	uns16	0 to 65534	3	S		This is the DNP Master address.
10	BAUD_RATE	uns8	0:110, 1:300, 2:600, 3:1200, 4:2400, 5:4800, 6:9600, 7:19200, 8:38400, 9:57600, 10:115200	9600	S		Define the baud rate (only for serial media).
11	STOP_BITS	uns8	0:1, 1:2	1	S		Define the number of stop bits (only for serial media).
12	PARITY	uns8	0:None, 1:Even, 2:Odd.	Even	S		Define the parity (only for serial media).
13	USE_MODEM	uns8	0-No;1-Yes	No	S		This parameter is used to define if a dial-up modem is used on the DNP slave port. If the value is set to No, no modem is used. If the parameter is set to Yes, a modem is used.
14	IDLE_TIMEOUT	uns16		300	S	sec	Defines the number of seconds to wait before closing the DNP communications channel due to inactivity.
15	LINK_STATUS_PERIOD	uns16		0	s	msec	Define how often to send link status requests if no DNP3 frames have been received. Only use a value greater than 0 if IDLE_TIMEOUT is 0.
16	FIRST_CHARACTER_DELAY	uns16		0	S	msec	Minimum time, in milliseconds, after receiving a character before we will attempt to transmit a character on this channel. This is generally useful when using a modem or some other communication device that requires a minimum time between receive and transmit.

Idx	Parameter Name	Data Type	Valid Range	Default Value	Store	Units	Description
17	FRAME_TIMEOUT	uns32		15000	S	msec	Maximum amount of time to wait for a complete frame after receiving valid frame sync characters
18	LINK_CONFIRM_MODE	uns8	0:Never, 1:Sometimes, 2:Always	Never	S	E	IED can request acknowledgment from master station when sending data.
19	LINK_CONFIRM_TOUT	uns32		2000	S	msec	Maximum amount of time to wait for a link level confirm if requested. Even if confirm Mode is set to NEVER this will still be used for Link Test frame and Request Link Status if they are sent.
20	LINK_MAX_RETRY	uns8		3	S	na	Maximum number of link layer retries if link layer confirms times out.
21	APP_CONFIRM_TOUT	uns32		10000		msec	Specifies how long the slave DNP device will wait for an application layer confirmation from the master.
22	MULTI_FRAG_CONFIRM	uns8	0-No;1-Yes	Yes	S	E	Specify whether application layer confirmations will be requested for non-final fragments of a multi fragment response. Application layer confirmations are always requested for responses that contain events.
23	SEL_OPERATE_TOUT	uns32		5000	S	msec	Specifies the maximum amount of time that a select will remain valid before the corresponding operate is received. If an operate request is received after this period has elapsed since the previous select the select will not be valid and the operate request will fail.
24	UNSOL_ALLOWED	uns8	0-No;1-Yes	No	S	E	Set if the slave unit will send unsolicited response messages. If set to No, the slave will not send unsolicited responses. If set to Yes, the slave will send unsolicited responses. This feature requires collision avoidance on a multi-drop network.
25	UNSOL_CLASS1_MAX_EVENTS	uns32		5	S	na	If unsolicited responses are enabled, specifies the maximum number of events in the corresponding class to be allowed before an unsolicited response will be generated.
26	UNSOL_CLASS1_MAX_DELAY	uns32		5000	S	msec	If unsolicited responses are enabled, specifies the maximum amount of time in milliseconds after an event in the class 1 is received before an unsolicited response will be generated.
27	UNSOL_CLASS2_MAX_EVENTS	uns32		5	S	na	If unsolicited responses are enabled, specifies the maximum number of events in the corresponding class to be allowed before an unsolicited response will be generated.

Idx	Parameter Name	Data Type	Valid Range	Default Value	Store	Units	Description
28	UNSOL_CLASS2_MAX_DELAY	uns32		5000	S	msec	If unsolicited responses are enabled, specifies the maximum amount of time in milliseconds after an event in the class 2 is received before an unsolicited response will be generated.
29	UNSOL_CLASS3_MAX_EVENTS	uns32		5	S	na	If unsolicited responses are enabled, specifies the maximum number of events in the corresponding class to be allowed before an unsolicited response will be generated.
30	UNSOL_CLASS3_MAX_DELAY	uns32		5000	S	msec	If unsolicited responses are enabled, specifies the maximum amount of time in milliseconds after an event in the class 3 is received before an unsolicited response will be generated.
31	UNSOL_RETRY_DELAY	uns32		5000	S	msec	Specifies the time to delay after an unsolicited confirm timeout before retrying the unsolicited response.
32	UNSOL_MAX_RETRY	uns16		3	S	na	Specify the maximum number of unsolicited retries before changing to the 'offline' retry period described below. This parameter allows you to specify up to 65535 retries. If you want an infinite number of retries set. UNSOL_OFFLINE_RETRY_DELAY to the same value as UNSOL_RETRY_DELAY in which case this parameter becomes a don't care.
33	UNSOL_OFFLINE_RETRY_DELAY	uns32		30000	S	msec	Specifies the time, in milliseconds, to delay after an unsolicited timeout before retrying the unsolicited response after UNSOL_MAX_RETRY have been attempted.
34	BINARY_CHANGE_EVENT_SCAN	uns32	0-65535	0	S	msec	Scan period of the Binary points.
35	ANALOG_CHANGE_EVENT_SCAN	uns32	0-65535	0	S	msec	Scan period of the Analog points.
36	DNP_TIME_SYNCHRONIZATION	uns8	0-No;1-Yes	No	S	na	Request of time synchronism.
37	CLOCK_VALID_PERIOD	uns32		1800000	S	msec	Specifies how long the local clock will remain valid after receiving a time synchronization.
38	MAX_FRAME_SIZE	uns32	28-292	292	S	na	Maximum link layer frame size.
39	MAX_FRAGMENT_SIZE	uns32	249-2048	2048	S	na	Maximum application layer fragment size. The DNP3 specification strongly recommends the maximum fragment size to be 2048.
40	MAX_DISC_BUFFER	uns32	0 - 45000	10	S	na	Maximum number of discrete records to be buffered.
41	MAX_ANA_BUFFER	uns32	0 - 45000	10	S	na	Maximum number of analog records to be buffered.
42	CLEAR_BUFFERS	uns8	0-No;1-Yes	No	D	na	Clears all buffered events.

Idx	Parameter Name	Data Type	Valid Range	Default Value	Store	Units	Description
43	DIAG_CLASS	uns8	0 - Class 0 1 - Class 1 2 - Class 2 3 - Class 3 4 - Disable	Disable	S	na	Determine the procedure of the diagnostic parameters: If the DIAG_CLASS is "disabled", no diagnosis will be generated in the DNP3. Otherwise, all the parameters are enabled and follow the same class chosen.
44	DIAG_TIME_EVENT	uns16		60	S	sec	Time to generate events of dynamic diagnostic parameters to dnp3.
45	UPDATE_EVT	DS-73			D		
46	BLOCK_ALM	DS-72			D		

Table 3 – Parameters of DNPCF block

DNPCS – DNP Control Slave

Description

DNPCS block generates a communication strategy between a DNP3 master and a FOUNDATION fieldbus slave. In this case, the slave is represented by Smar's linking device **DFI302** that has slave behavior for the DNP3 network. It allows DNP3 variables to be associated with fieldbus variables and data between these two "worlds" to be exchanged through **DFI302**.

Inputs and Outputs

This block has 8 digital inputs, 8 analog inputs, 8 digital outputs and 8 analog outputs that may be connected to other fieldbus function blocks or to the DNP3 world.

- IN1 up to IN8 are analog inputs.
- IN_D1 up to IN_D8 are digital inputs.
- OUT1 up to OUT8 are analog outputs.
- OUT_D1 up to OUT_D8 are digital outputs.

Digital outputs and Digital inputs are of the DS-66 data type. Thus, they contain both a Status and a value (both Unsigned 8). The analog outputs and inputs are of the DS-65 data type, containing status and value as well. Type of values is Float.

The block operation mode defines the mode to operate the inputs and outputs. In Out-of-Service (OOS) the block does not accomplish the scan of inputs and outputs and does not update the DNP3 buffer. In Automatic mode (AUTO) the block updates the inputs in DNP3 buffer and updates the buffer with the outputs. In Manual mode (MAN) only the outputs are disconnected from DNP3 and the user may to operate these manually.

Binary Output Command – Control Relay Output Block

The discrete output points support the timer function used for operations as Control Relay Outputs Block (for complete information, consult the DNP3 Technical Standard, Volume 6, Part 2.). In the table below, the necessary DNP3 fields are described to be configured by master.

DNP3 Field	Valid Range	Description
Operation Type	0: NULL 1: PULSE_ON 2: PULSE_OFF 3: LATCH_ON 4: LATCH_OFF	This field is used to specify the control operation: If only the LATCH_ON or LATCH_OFF actions are necessary, then the following fields should be set: Count=1, OnTime=0, OffTime=0. For PULSE_ON and PULSE OFF action the operation follows as the Table 2.
Trip Close Code [TCC]	0: NULL	Always Null.
OnTime	Positive	This is the duration, expressed as the number of milliseconds, that the output drive remains active.
OffTime	Positive	This is the duration, expressed as the number of milliseconds that the output drive remains non-active.
Clear	0 – False 1 – True	When the clear bit is set, the device removes pending control commands for that index and stop any control operation that is in progress for that index. The indexed point shall go to the state that it would have if the command were allowed to complete normally.
Count		This is the number of times the outstation shall execute the operation. Counts greater than 1 generate a series of pulses or repeated operations for the point. Both <i>On-time</i> and <i>Off-time</i> values are obeyed.

Table 4 – Operation Fields as Control Relay Output Block (only discrete outputs)

The algorithm is based on Table 2, where there is a priority of descending order in the interruption.

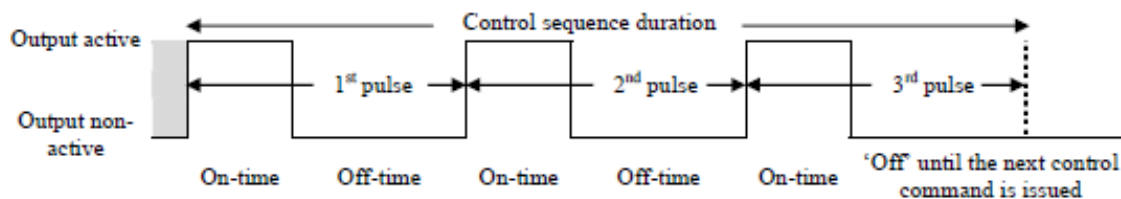
TCC	Clear	Op Type	Count	OnTime	OffTime	Description
0	1	NULL	X	X	X	OUT_Dx.value = 0
0	1	PULSE_ON or LATCH_ON or LATCH_OFF	>0	>0	>0	Forced OUT_Dx.value=0 for a macrocycle, after to start the pulses generation.
0	0	PULSE_ON or LATCH_ON or LATCH_OFF	>0	=0	=0	OUT_Dx.value = 1
0	0	PULSE_ON or LATCH_ON or LATCH_OFF	>0	>0	>0	Generate the quantity of requested pulses. If the output point is in 1, the first pulse will start immediately, where the handling is the same indifferently of the output point value before the command.
Other conditions						Ignored the command.

Where X is any value.

Table 5 – Operation Algorithm of Control Relay Output Block

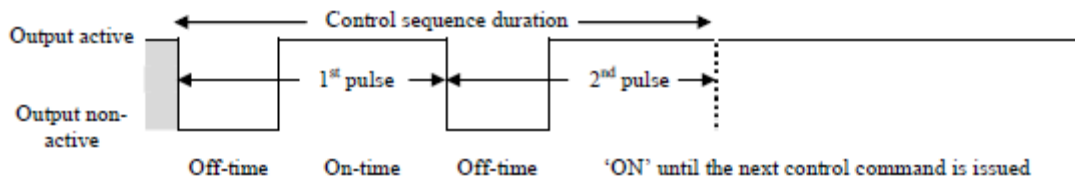
Examples

1) PULSE ON Features



PULSE ON with Count = 3

2) PULSE OFF Features



PULSE OFF with Count = 2 (Not interoperable)

Scale Conversion

The scale conversion for DNP3 protocol has two purposes:

- Conversion of an analog value internal of the block for a DNP3 value express in engineering units.
- Conversion of an analog value of the DNP3 for an internal value of the block in engineering units.

The scale parameters are composed by the following fields:

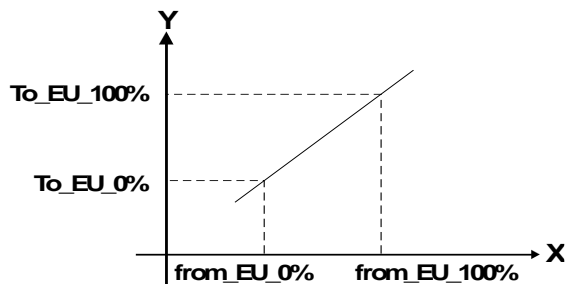
- **FROM_EU_100%** - defines the higher input unit value (actual data unit);
- **FROM_EU_0%** - defines the lower input unit value (actual data unit);
- **TO_EU_100%** - defines the higher output unit value (desired data unit);
- **TO_EU_0%** - defines the lower output unit value (desired data unit).

Observation

The conversion procedure is $Y=Ax+B$ shown in the figure below:

$$A = (TO_EU_100\% - TO_EU_0\%) / (FROM_EU_100\% - FROM_EU_0\%)$$

$$B = TO_EU_0\% - A * FROM_EU_0\%$$



The X axis represents the input variable (From_EU) and the Y axis represents the output variable (To_EU).

To read the variables from Fieldbus, IN_x represents the input and the DNP3 variable represents the output. To write variables in Fieldbus, the DNP3 variable represents the input and OUT_x represents the output.

Addressing DNP3 Points

Each input and output point of the DNP3 block receive a DNP3 address. The addressing is through the LOCAL_MAP parameter that configures an addresses range according to the instance of block. The user must configure a value of LOCAL_MAP of 0 to 15, where for each DNP3 block must exist a different LOCAL_MAP. The LOCAL_MAP value of 255 disables the DNP3 communication of the respective block. The addresses range for each data type is defined as:

$$\text{DNP3 Point} = \text{Start_Addr} + \text{LOCAL_MAP} * 8 + (x-1)$$

Where:

“Start_Addr” is determined according to the Table below and the parameter type.

“LOCAL_MAP” is a block parameter that can change of 0 to 15. The default is 255 (disabled block points).

“x” corresponds to the point index varying of 1 to 8. For example, OUT_D5 corresponds to x equal to 5.

DNP3 Parameter	Start_Addr	DNP3 Group	DNP3 Variation	Data Type	Client Access
IN_Dx-Value	0	1	1	Bool	RO
IN_x-Value	0	30	5	Float	RO
IN_x-Status	128	30	2	Int16	RO
IN_Dx-Status	136	30	2	Int16	RO
OUT_Dx-Value	0	10 or 12	1	Bool	RW
OUT_x-Value	0	40	3	Float	RW
OUT_x-Status	128	40	2	Int16	RW
OUT_Dx-Status	136	40	2	Int16	RW

Table 6 – Start_Addr Values

The DNP3 addressing is composed also of the group that represents the operation type: reading, writing, discrete or analog and the change of point that represents the DNP3 data type.

For example, consider a **DNP3** block configuration with **LOCAL_MAP** equal to 3 and it wants to read the value of IN_1 and IN_5 points using DNP3. Then the point addresses are:

For reading of DNPCS3.IN_1-Value point has:

DNP3.Group = 30

DNP3.Variatin = 5

$\text{DNP3.Point} = 0 + 3 * 8 + (1-1) = 24$

To read the DNP3 point is necessary the "30.5.24" addressing.

For writing of DNPCS3.OUT_5-Value point has:

DNP3.Group = 40

DNP3.Variatin = 3

$\text{DNP3.Point} = 0 + 3 * 8 + (1-1) = 28$

To read the DNP3 point is necessary the "40.3.28" addressing.

The table below has the corresponding DNP addresses (Group, Variation, Point) for each point of **DNPCS** block for the first four and the last **LOCAL_MAP**. For the other addresses, it must be calculated according to the equation shown above.

DNPCS	LOCAL_MAP		0	1	2	3	15
IO PARAM	DNP Group	DNP Variation	DNP Point					
IN_D1-Value	1	1	0	8	16	24		120
IN_D2-Value			1	9	17	25		121
IN_D3-Value			2	10	18	26		122
IN_D4-Value			3	11	19	27		123
IN_D5-Value			4	12	20	28		124
IN_D6-Value			5	13	21	29		125
IN_D7-Value			6	14	22	30		126
IN_D8-Value			7	15	23	31		127
IN_1-Value	30	5	0	8	16	24		120
IN_2-Value			1	9	17	25		121
IN_3-Value			2	10	18	26		122
IN_4-Value			3	11	19	27		123
IN_5-Value			4	12	20	28		124
IN_6-Value			5	13	21	29		125
IN_7-Value			6	14	22	30		126
IN_8-Value			7	15	23	31		127
IN_1-Status	30	2	128	144	160	176		368
IN_2-Status			129	145	161	177		369
IN_3-Status			130	146	162	178		370
IN_4-Status			131	147	163	179		371
IN_5-Status			132	148	164	180		372
IN_6-Status			133	149	165	181		373
IN_7-Status			134	150	166	182		374
IN_8-Status			135	151	167	183		375
IN_D1-Status	30	2	136	152	168	184		376
IN_D2-Status			137	153	169	185		377
IN_D3-Status			138	154	170	186		378
IN_D4-Status			139	155	171	187		379
IN_D5-Status			140	156	172	188		380
IN_D6-Status			141	157	173	189		381
IN_D7-Status			142	158	174	190		382
IN_D8-Status			143	159	175	191		383
OUT_D1-Value	10 or 12	1	0	8	16	24		120
OUT_D2-Value			1	9	17	25		121
OUT_D3-Value			2	10	18	26		122

DNPCS	LOCAL_MAP		0	1	2	3	15
IO PARAM	DNP Group	DNP Variation	DNP Point					
OUT_D4-Value			3	11	19	27		123
OUT_D5-Value			4	12	20	28		124
OUT_D6-Value			5	13	21	29		125
OUT_D7-Value			6	14	22	30		126
OUT_D8-Value			7	15	23	31		127
OUT_1-Value	40	3	0	8	16	24		120
OUT_2-Value			1	9	17	25		121
OUT_3-Value			2	10	18	26		122
OUT_4-Value			3	11	19	27		123
OUT_5-Value			4	12	20	28		124
OUT_6-Value			5	13	21	29		125
OUT_7-Value			6	14	22	30		126
OUT_8-Value			7	15	23	31		127
OUT_1-Status	40	2	128	144	160	176		368
OUT_2-Status			129	145	161	177		369
OUT_3-Status			130	146	162	178		370
OUT_4-Status			131	147	163	179		371
OUT_5-Status			132	148	164	180		372
OUT_6-Status			133	149	165	181		373
OUT_7-Status			134	150	166	182		374
OUT_8-Status			135	151	167	183		375
OUT_D1-Status	40	2	136	152	168	184		376
OUT_D2-Status			137	153	169	185		377
OUT_D3-Status			138	154	170	186		378
OUT_D4-Status			139	155	171	187		379
OUT_D5-Status			140	156	172	188		380
OUT_D6-Status			141	157	173	189		381
OUT_D7-Status			142	158	174	190		382
OUT_D8-Status			143	159	175	191		383

Table 7 – Examples of corresponding DNP addresses for each point in the DNPCS block

DNP3 Status

The status of a certain input and output point of block can be accessed through the DNP3 protocol of two modes:

- Through the “Flag” attribute of DNP3 point;
- Through the corresponding FF status value in the DNP3 IN_x-Status, IN_Dx-Status, OUT_x-Status, OUT_Dx-Status points.

Each input and output point has a “Flag” DNP3 that indicates the point status. The “Flag” indicates that the point is Good (Value 1) or Bad (Value 0) according to Foundation FF status of the input and output point. For example, considers a FF control logic with the DNPCS0.IN_1 block (**LOCAL_MAP** equals to zero) as shown the Figure 1.

When the status of IN_1 input is “Good” then “Flag” DNP3 of the IN_1 point will be 1 indicating that the point is ok. When IN_1. Status is “Bad” due to link failure between the control block and **DNPCS** or propagation of a failure in another equipment, then “Flag” DNP3 of the point will be 0.

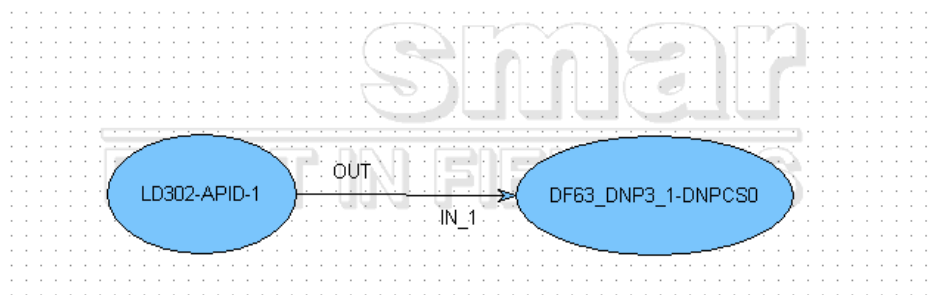


Figure C.3 – Example of Control Logic using DNPCS Block

Another mode to report FF status through the DNP3 protocol is using the FF status through DNP3 status points. In this case an analog variable must be mapped in the master. This variable represents the corresponding FF status. Each input and output point of the block has a corresponding zzzz_Status point (where zzzz can be IN_x, IN_Dx, OUT_x, OUT_Dx according to the Table above).

For example, considers the IN_1 parameter from the example above. To access the FF status of IN_1 point through DNP3 it is necessary to read the “IN_1-Status” point (as **LOCAL_MAP** equals to zero then the DNP3 address of point is 128). When the IN_1 input is “Good Non Cascade Non Specific” then the DNP3 value of this point will be 128. When the IN_1 input presents a “Bad No Communication with last usable value” status the DNP3 value of the point will be 20. For more information about the FF status consult the Chapter of Function Blocks Instruction Manual - Composition of Status item.

For block outputs, the status handling is: If **STATUS_OUTx** parameter is equal to:

- Bad OOS – then the output status will be always Good Non Cascade.
- Set by master - the master set the status and if it is not updated in timeout, it will be bad.
- Another status – the status will be the respective, but it is not updated in timeout it will be bad.

The timeout is determined by **DNCPF.ConnectionTimeout** parameter. The internal timer is zero in the following conditions:

- All change in **DNPCS** block of OOS to AUTO.
- All DNP3 writing in any output parameter of the respective **DNPCS** block.

BLOCK_ERR

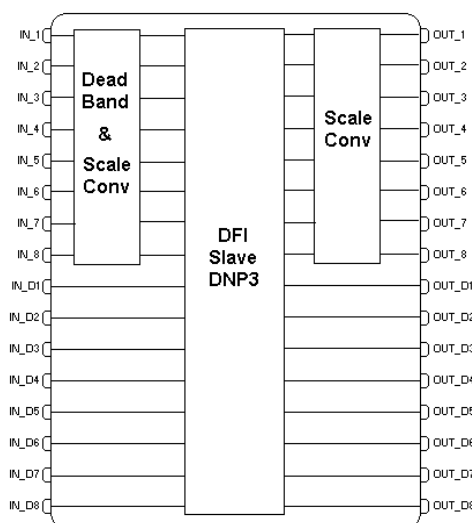
The **BLOCK_ERR** of the **DNPCS** block will reflect the following causes:

- Out of Service: When the block is in O/S mode.

Supported Modes

O/S, Man and Auto.

Schematic



Parameters

Idx	Parameter Name	Data Type	Valid Range	Default Value	Units	Store/ Mode	Description
1	ST_REV	uns16				S	
2	TAG_DESC	Octet				S	
3	STRATEGY	uns16				S	
4	ALERT_KEY	uns8				S	
5	MODE_BLK	DS_64	OOS, MAN, AUTO			S	See Mode Parameter.
6	BLOCK_ERR	Bitstring				D	
7	IN_1	DS_65				D	Analog input 1.
8	IN_2	DS_65				D	Analog input 2.
9	IN_3	DS_65				D	Analog input 3.
10	IN_4	DS_65				D	Analog input 4.
11	IN_5	DS_65				D	Analog input 5.
12	IN_6	DS_65				D	Analog input 6.
13	IN_7	DS_65				D	Analog input 7.
14	IN_8	DS_65				D	Analog input 8.
15	IN_D1	DS_66				D	Digital input 1.
16	IN_D2	DS_66				D	Digital input 2.
17	IN_D3	DS_66				D	Digital input 3.
18	IN_D4	DS_66				D	Digital input 4.
19	IN_D5	DS_66				D	Digital input 5.
20	IN_D6	DS_66				D	Digital input 6.
21	IN_D7	DS_66				D	Digital input 7.
22	IN_D8	DS_66				D	Digital input 8.
23	OUT_1	DS_65				D	Analog output 1.
24	OUT_2	DS_65				D	Analog output 2.
25	OUT_3	DS_65				D	Analog output 3.
26	OUT_4	DS_65				D	Analog output 4.
27	OUT_5	DS_65				D	Analog output 5.
28	OUT_6	DS_65				D	Analog output 6.
29	OUT_7	DS_65				D	Analog output 7.
30	OUT_8	DS_65				D	Analog output 8.
31	OUT_D1	DS_66				D	Digital output 1.
32	OUT_D2	DS_66				D	Digital output 2.
33	OUT_D3	DS_66				D	Digital output 3.
34	OUT_D4	DS_66				D	Digital output 4.
35	OUT_D5	DS_66				D	Digital output 5.
36	OUT_D6	DS_66				D	Digital output 6.
37	OUT_D7	DS_66				D	Digital output 7.
38	OUT_D8	DS_66				D	Digital output 8.
39	LOCAL_MAP	uns8	0-15, 255	255		S	Define the Start addresses of the DNP areas.
40	CLASS_IN	uns8	0 - Class 0 1 - Class 1 2 - Class 2 3 - Class 3 4 - Disable	Class 0		S	Define the class for input points.

Idx	Parameter Name	Data Type	Valid Range	Default Value	Units	Store/ Mode	Description
41	CLASS_OUT	uns8	0 - Class 0 1 - Class 1 2 - Class 2 3 - Class 3 4 - Disable	Disable		S	Define the class for output points.
42	DEAD_BAND_1	float		0.0	IN_1	S	Dead band for the IN_1 input where the variation into this range would not cause writing in the block parameter.
43	DEAD_BAND_2	float		0.0	IN_2	S	Dead band for the IN_2 input where the variation into this range would not cause writing in the block parameter.
44	DEAD_BAND_3	float		0.0	IN_3	S	Dead band for the IN_3 input where the variation into this range would not cause writing in the block parameter.
45	DEAD_BAND_4	float		0.0	IN_4	S	Dead band for the IN_4 input where the variation into this range would not cause writing in the block parameter.
46	DEAD_BAND_5	float		0.0	IN_5	S	Dead band for the IN_5 input where the variation into this range would not cause writing in the block parameter.
47	DEAD_BAND_6	float		0.0	IN_6	S	Dead band for the IN_6 input where the variation into this range would not cause writing in the block parameter.
48	DEAD_BAND_7	float		0.0	IN_7	S	Dead band for the IN_7 input where the variation into this range would not cause writing in the block parameter.
49	DEAD_BAND_8	float		0.0	IN_8	S	Dead band for the IN_8 input where the variation into this range would not cause writing in the block parameter.
50	SCALE_IN_1	DS_299		From: 0-100% To: 0-100%		S	Scale conversion from the IN_1 input to DNP value.
51	SCALE_IN_2	DS_299		From: 0-100% To: 0-100%		S	Scale conversion from the IN_2 input to DNP value.
52	SCALE_IN_3	DS_299		From: 0-100% To: 0-100%		S	Scale conversion from the IN_3 input to DNP value.
53	SCALE_IN_4	DS_299		From: 0-100% To: 0-100%		S	Scale conversion from the IN_4 input to DNP value.
54	SCALE_IN_5	DS_299		From: 0-100% To: 0-100%		S	Scale conversion from the IN_5 input to DNP value.
55	SCALE_IN_6	DS_299		From: 0-100% To: 0-100%		S	Scale conversion from the IN_6 input to DNP value.
56	SCALE_IN_7	DS_299		From: 0-100% To: 0-100%		S	Scale conversion from the IN_7 input to DNP value.
57	SCALE_IN_8	DS_299		From: 0-100% To: 0-100%		S	Scale conversion from the IN_8 input to DNP value.
58	SCALE_OUT_1	DS_299		From: 0-100% To: 0-100%		S	Scale conversion from the DNP to OUT_1 value.
59	SCALE_OUT_2	DS_299		From: 0-100% To: 0-100%		S	Scale conversion from the DNP to OUT_2 value.
60	SCALE_OUT_3	DS_299		From: 0-100% To: 0-100%		S	Scale conversion from the DNP to OUT_3 value.

Idx	Parameter Name	Data Type	Valid Range	Default Value	Units	Store/ Mode	Description
61	SCALE_OUT_4	DS_299		From: 0-100% To: 0-100%		S	Scale conversion from the DNP to OUT_4 value.
62	SCALE_OUT_5	DS_299		From: 0-100% To: 0-100%		S	Scale conversion from the DNP to OUT_5 value.
63	SCALE_OUT_6	DS_299		From: 0-100% To: 0-100%		S	Scale conversion from the DNP to OUT_6 value.
64	SCALE_OUT_7	DS_299		From: 0-100% To: 0-100%		S	Scale conversion from the DNP to OUT_7 value.
65	SCALE_OUT_8	DS_299		From: 0-100% To: 0-100%		S	Scale conversion from the DNP to OUT_8 value.
66	STATUS_OUT_1	uns8		Bad OOS		S	Define the status of the OUT_1 parameter. If chose "Set by Master", the status will be provided by the Master DNP (needs to be in the FF Status write in the DNP Address of the status point). If STATUS_OUT_1 is any other status, this will be good status.
67	STATUS_OUT_2	uns8		Bad OOS		S	Define the status of the OUT_2 parameter. If chose "Set by Master", the status will be provided by the Master DNP (needs to be in the FF Status write in the DNP Address of the status point). If STATUS_OUT_2 is any other status, this will be good status.
68	STATUS_OUT_3	uns8		Bad OOS		S	Define the status of the OUT_3 parameter. If chose "Set by Master", the status will be provided by the Master DNP (needs to be in the FF Status write in the DNP Address of the status point). If STATUS_OUT_3 is any other status, this will be good status.
69	STATUS_OUT_4	uns8		Bad OOS		S	Define the status of the OUT_4 parameter. If chose "Set by Master", the status will be provided by the Master DNP (needs to be in the FF Status write in the DNP Address of the status point). If STATUS_OUT_4 is any other status, this will be good status.
70	STATUS_OUT_5	uns8		Bad OOS		S	Define the status of the OUT_5 parameter. If chose "Set by Master", the status will be provided by the Master DNP (needs to be in the FF Status write in the DNP Address of the status point). If STATUS_OUT_5 is any other status, this will be good status.
71	STATUS_OUT_6	uns8		Bad OOS		S	Define the status of the OUT_6 parameter. If chose "Set by Master", the status will be provided by the Master DNP (needs to be in the FF Status write in the DNP Address of the status point). If STATUS_OUT_6 is any other status, this will be good status.

Idx	Parameter Name	Data Type	Valid Range	Default Value	Units	Store/ Mode	Description
72	STATUS_OUT_7	uns8		Bad OOS		S	Define the status of the OUT_7 parameter. If chose "Set by Master", the status will be provided by the Master DNP (needs to be in the FF Status write in the DNP Address of the status point). If STATUS_OUT_7 is any other status, this will be good status.
73	STATUS_OUT_8	uns8		Bad OOS		S	Define the status of the OUT_8 parameter. If chose "Set by Master", the status will be provided by the Master DNP (needs to be in the FF Status write in the DNP Address of the status point). If STATUS_OUT_8 is any other status, this will be good status.
74	STATUS_OUT_D1	uns8		Bad OOS		S	Define the status of the OUT_D1 parameter. If chose "Set by Master", the status will be provided by the Master DNP (needs to be in the FF Status write in the DNP Address of the status point). If STATUS_OUT_D1 is any other status, this will be good status.
75	STATUS_OUT_D2	uns8		Bad OOS		S	Define the status of the OUT_D2 parameter. If chose "Set by Master", the status will be provided by the Master DNP (needs to be in the FF Status write in the DNP Address of the status point). If STATUS_OUT_D2 is any other status, this will be good status.
76	STATUS_OUT_D3	uns8		Bad OOS		S	Define the status of the OUT_D3 parameter. If chose "Set by Master", the status will be provided by the Master DNP (needs to be in the FF Status write in the DNP Address of the status point). If STATUS_OUT_D3 is any other status, this will be good status.
77	STATUS_OUT_D4	uns8		Bad OOS		S	Define the status of the OUT_D4 parameter. If chose "Set by Master", the status will be provided by the Master DNP (needs to be in the FF Status write in the DNP Address of the status point). If STATUS_OUT_D4 is any other status, this will be good status.
78	STATUS_OUT_D5	uns8		Bad OOS		S	Define the status of the OUT_D5 parameter. If chose "Set by Master", the status will be provided by the Master DNP (needs to be in the FF Status write in the DNP Address of the status point). If STATUS_OUT_D5 is any other status, this will be good status.

Idx	Parameter Name	Data Type	Valid Range	Default Value	Units	Store/ Mode	Description
79	STATUS_OUT_D6	uns8		Bad OOS		S	Define the status of the OUT_D6 parameter. If chose "Set by Master", the status will be provided by the Master DNP (needs to be in the FF Status write in the DNP Address of the status point). If STATUS_OUT_D6 is any other status, this will be good status.
80	STATUS_OUT_D7	uns8		Bad OOS		S	Define the status of the OUT_D7 parameter. If chose "Set by Master", the status will be provided by the Master DNP (needs to be in the FF Status write in the DNP Address of the status point). If STATUS_OUT_D7 is any other status, this will be good status.
81	STATUS_OUT_D8	uns8		Bad OOS		S	Define the status of the OUT_D8 parameter. If chose "Set by Master", the status will be provided by the Master DNP (needs to be in the FF Status write in the DNP Address of the status point). If STATUS_OUT_D8 is any other status, this will be good status.
82	UPDATE_EVT	DS-73				D	
83	BLOCK_ALM	DS-72				D	

Table 8 – Parameter of DNPCS block

DS-299 – Scale Convert			
E	Element	Data Type	Length
1	From EU 0%	Float	4
2	From EU 100%	Float	4
3	To EU 0%	Float	4
4	To EU 100%	Float	4