User Manual

EtherNet/IP™ to Serial

Linking Device

Doc. Id. HMSI-27-354
Rev. 1.00

Connecting Devices™

HALMSTAD • CHICAGO • KARLSRUHE • TOKYO • BEIJING • MILANO • MULHOUSE • COVENTRY • PUNE • COPENHAGEN

HMS Industrial Networks
Mailing address: Box 4126, 300 04 Halmstad, Sweden
Visiting address: Stationsgatan 37, Halmstad, Sweden

E-mail: info@hms-networks.com
Web: www.anybus.com

Important User Information

This document contains a general introduction as well as a description of the technical features provided by the EtherNet/IP to Serial linking device, including the PC-based configuration software.

The reader of this document is expected to be familiar with PLC and software design, as well as communication systems in general. The reader is also expected to be familiar with the Microsoft® Windows® operating system.

Liability

Every care has been taken in the preparation of this manual. Please inform HMS Industrial Networks AB of any inaccuracies or omissions. The data and illustrations found in this document are not binding. We, HMS Industrial Networks AB, reserve the right to modify our products in line with our policy of continuous product development. The information in this document is subject to change without notice and should not be considered as a commitment by HMS Industrial Networks AB. HMS Industrial Networks AB assumes no responsibility for any errors that may appear in this document.

There are many applications of this product. Those responsible for the use of this device must ensure that all the necessary steps have been taken to verify that the applications meet all performance and safety requirements including any applicable laws, regulations, codes, and standards.

HMS Industrial Networks AB will under no circumstances assume liability or responsibility for any problems that may arise as a result from the use of undocumented features, timing, or functional side effects found outside the documented scope of this product. The effects caused by any direct or indirect use of such aspects of the product are undefined, and may include e.g. compatibility issues and stability issues.

The examples and illustrations in this document are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular implementation, HMS Industrial Networks AB cannot assume responsibility for actual use based on these examples and illustrations.

Intellectual Property Rights

HMS Industrial Networks AB has intellectual property rights relating to technology embodied in the product described in this document. These intellectual property rights may include patents and pending patent applications in the US and other countries.

Trademark Acknowledgements

Microsoft® and Windows® are registered trademarks of Microsoft, Inc. EtherNet/IP™ and ODVA™ are trademarks of ODVA, Inc. All other trademarks are the property of their respective holders.

Warning: This is a class A product. in a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

ESD Note: This product contains ESD (Electrostatic Discharge) sensitive parts that may be damaged if ESD control procedures are not followed. Static control precautions are required when handling the product. Failure to observe this may cause damage to the product.

EtherNet/IP to Serial Linking Device Linking Device User Manual
Copyright© HMS Industrial Networks AB
April 2016. Doc: HMSI-27-354

Table of Contents

Preface About This Document

Chapter 1 About the EtherNet/IP to Serial Linking Device

Chapter 2 Basic Operation

Chapter 3 Studio 5000 Implementation Example

Chapter 4 Navigating the Configuration Manager

Chapter 5 Basic Settings

Chapter 6 Nodes

Chapter 7 Transactions

Chapter 8 Frame Objects

Chapter 9 Commands

Chapter 10 DF1 Protocol Mode

Chapter 11 Process Tags Editor

Chapter 12 Subnetwork Monitor

Chapter 13 Node Monitor

Chapter 14 Data Logger

Chapter 15 Control and Status Registers

Appendix A File System

Appendix B FTP Server

Appendix C Advanced Network Configuration

Appendix D Web Server

Appendix E E-mail Client

Appendix F CIP Object Implementation

P. About This Document

For more information, documentation etc., please visit http://www.encompass.hms-networks.com/support.

P.1 Related Documents

Document name	Author
DF1 Protocol and Command Set - Reference Manual, 1770-6.5.16, October 1996	Allen-Bradley
Open Modbus/TCP Specification, Release 1.0	Schneider Electric
RFC 821	Network Working Group
RFC 1918	Network Working Group
ENIP Specifications	ODVA

P.2 Document History

Summary of Recent Changes

Change	Page(s)
-	-

Revision List

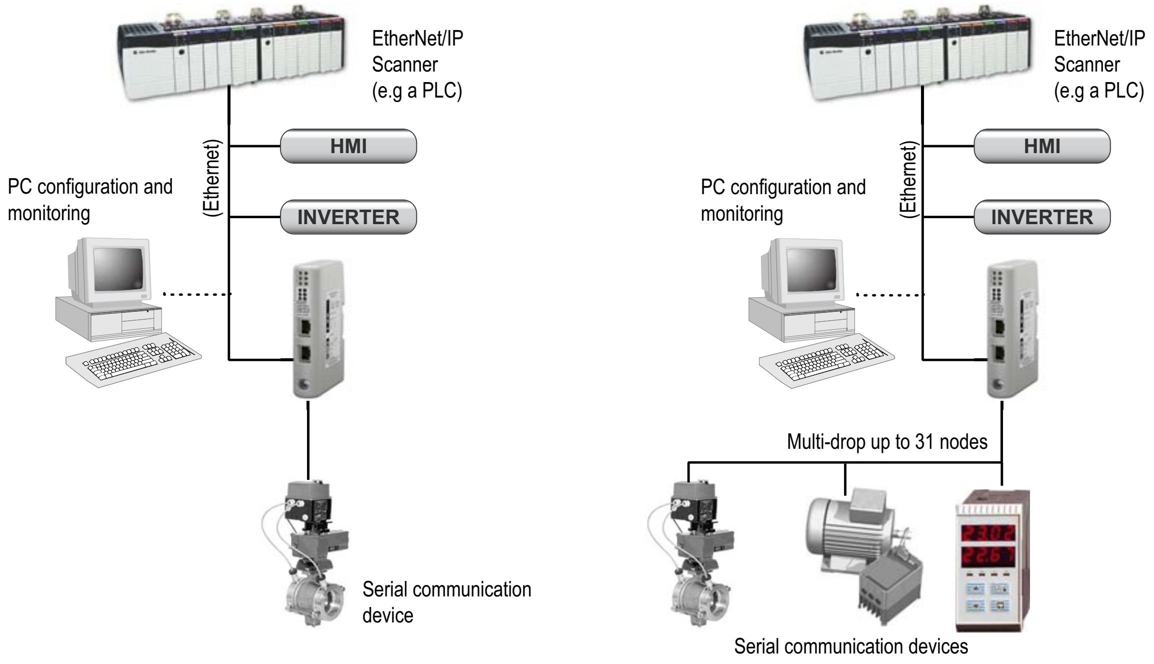
Revision	Date	Author	Chapter	Description
1.00	2016-04-01	KaD	All	First edition

P.3 Sales and Support

For general contact information and support, please refer to the contact and support pages at http://www.encompass.hms-networks.com/support/contact-support.

1. About the EtherNet/IP to Serial Linking Device

The EtherNet/IP to Serial linking device acts as a gateway between virtually any serial application protocol and an EtherNet/IP-based network. Integration of industrial devices is enabled with no loss of functionality, control and reliability, both when retro-fitting to existing equipment as well as when setting up new installations.



Single-Node Serial Subnetwork

Multi-Node Serial Subnetwork

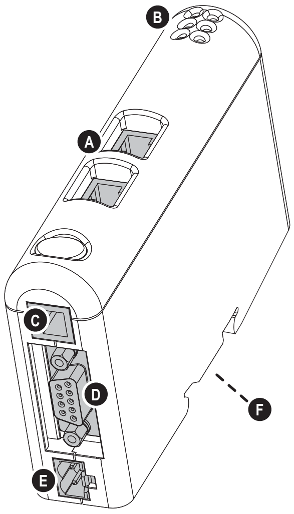
Subnetwork

The linking device can address up to 31 nodes, and supports the following physical standards:

- RS-232
- RS-422
- RS-485

1.1 External View

For wiring and pin assignments, see “Connector Pin Assignments” on page 101.



A: Ethernet Connectors

- “Ethernet Connector” on page 101

B: Status LEDs

- “Status LEDs” on page 12

C: PC Connector

Not used.

D: Subnetwork Connector

This connector is used to connect the device to the serial subnetwork.

- “Subnetwork Interface” on page 102

E: Power Connector

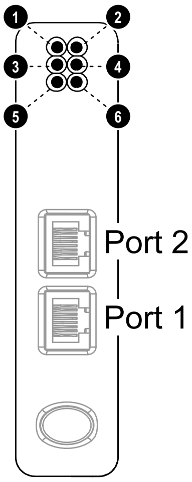
- “Power Connector” on page 101

F: DIN-rail Connector

The DIN-rail mechanism connects the device to PE (Protective Earth).

- “Hardware Installation” on page 13

1.2 Status LEDs



#	State	Status
1 - Module Status *(EtherNet/IP only)*	Off	No power
	Green	Controlled by a scanner in RUN state
	Green, flashing	Not configured, or scanner in IDLE state
	Red	Major fault (unrecoverable)
	Red, flashing	Minor fault (recoverable)
	Alternating Red/Green	Self-test
2 - Network Status	Off	No IP address (or no power)
	Green	Online, EtherNet/IP connection(s) established
	Green, flashing	Online, no EtherNet/IP connections established
	Red	Duplicate IP address detected, fatal error
	Red, flashing	One or more connections timed out
	Alternating Red/Green	Self-test
3 - Link/Activity 1	Off	No link (or no power)
	Green, flashing	Receiving/transmitting Ethernet packets (100 Mbit)
	Red, flashing	Receiving/transmitting Ethernet packets (10 Mbit)
4 - Link/Activity 2	Off	No link (or no power)
	Green, flashing	Receiving/transmitting Ethernet packets (100 Mbit)
	Red, flashing	Receiving/transmitting Ethernet packets (10 Mbit)
5 - Subnet Status[a]	Off	(no power)
	Green, flashing	Running correctly, but one or more transaction error(s) have occurred
	Green	Running
	Red	Transaction error/timeout or subnet stopped
6 - Device Status	Off	(no power)
	Alternating Red/Green	Invalid or missing configuration
	Green	Initializing
	Green, flashing	Running
	Red	Bootloader mode[b]
	Red, flashing	If the Device Status LED is flashing in a sequence starting with one or more red flashes, please note the sequence pattern and contact support.

[a] This LED shows green when all transactions have been active at least once. This includes any transactions using “change of state” or “change of state on trigger”. If a time-out occurs on a transaction, this LED will show red.

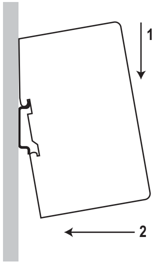
[b] The linking device is in bootloader mode. Firmware must be restored in order for it to work properly. Start up the configuration manager and connect to the linking device. Select Tools/Options/Module. Click Factory Restore to restore firmware. See “Tools” on page 33.

1.3 Hardware Installation

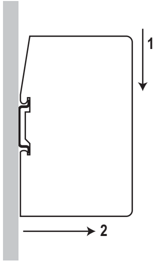
Perform the following steps to install the linking device:

1. Snap the device on to the DIN-rail.

   The DIN-rail mechanism works as follows:



To snap the device *on*, first press it downwards (1) to compress the spring in the DIN-rail mechanism, then push it against the DIN-rail as to make it snap on (2).



To snap the device *off*, push it downwards (1) and pull it out from the DIN-rail (2), as to make it snap off from the DIN-rail.

2. Connect the linking device to the EtherNet/IP network.
3. Connect the device to the serial subnetwork.
4. Connect the power cable and apply power.
5. For information about how to configure the linking device, see “Studio 5000 Implementation Example” on page 21.

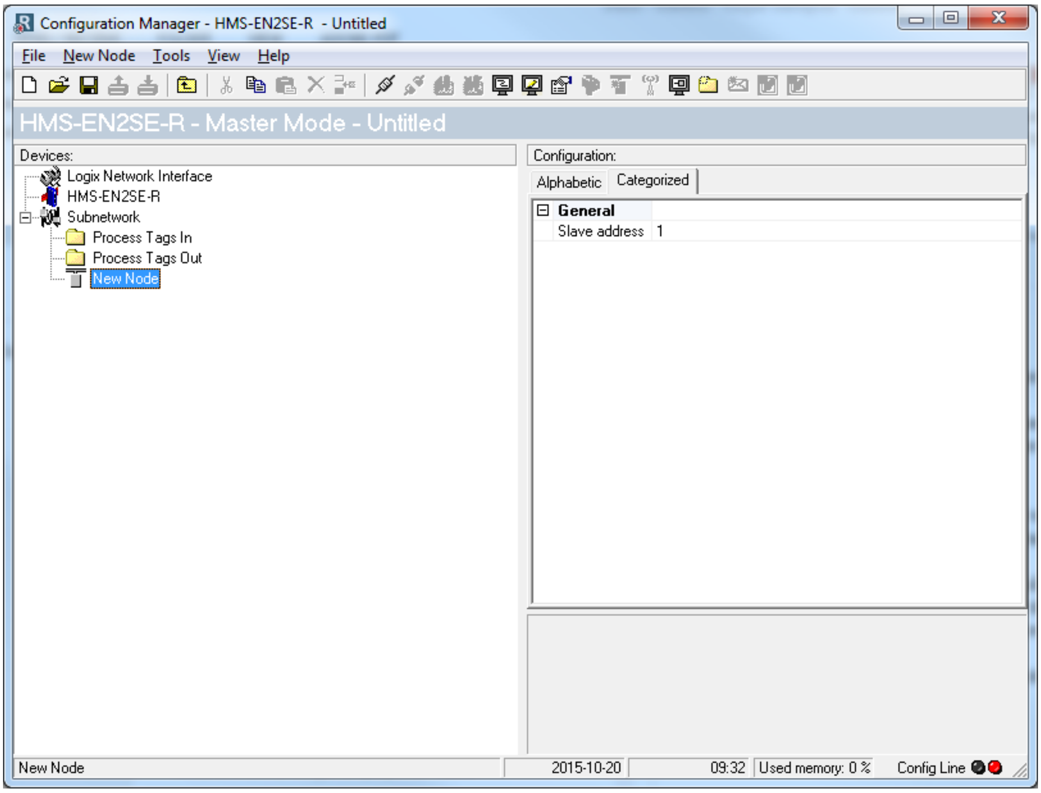
2. Basic Operation

2.1 General

The EtherNet/IP to Serial linking device is designed to exchange data between a serial subnetwork and a EtherNet/IP network. Unlike most other similar devices, the linking device has no fixed protocol for the subnetwork, and consequently can be configured to handle almost any form of serial communication.

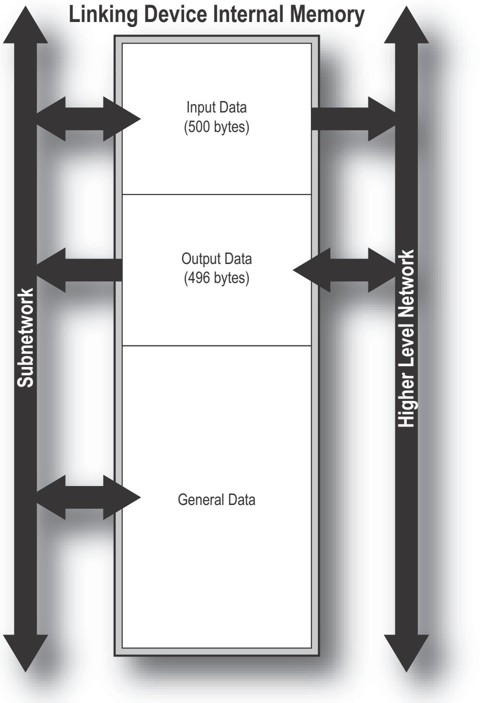
The linking device can issue serial telegrams cyclically, on change of state, or based on trigger events issued by the control system in the higher level network. It can also monitor certain aspects of the subnetwork communication and notify the higher level network when data has changed.

An essential part of the EtherNet/IP to Serial linking device package is the configuration manager software, an application used to supply the device with a description of the subnetwork protocol. The software is fully integrated into the Studio 5000 environment. No programming skills are required; instead, a visual protocol description system is used to specify the different parts of the serial communication.



Slave address 1

2.2 Data Exchange Model



Internally, data exchanged on the subnetwork and on the higher level network all resides in the same memory.

This means that in order to exchange data with the subnetwork, the higher level network simply reads and writes data to the different memory areas. The very same memory locations can then be exchanged on the subnetwork.

The internal memory buffer is divided into three areas, based on function:

- **Input Data (500 bytes)**

  This area can be read by the higher level network, the web server and the e-mail client.

  (Data representation on the higher level network is described later in this chapter).

- **Output Data (496 bytes)**

  This area can be read/written to by the higher level network, the web server and the e-mail client.

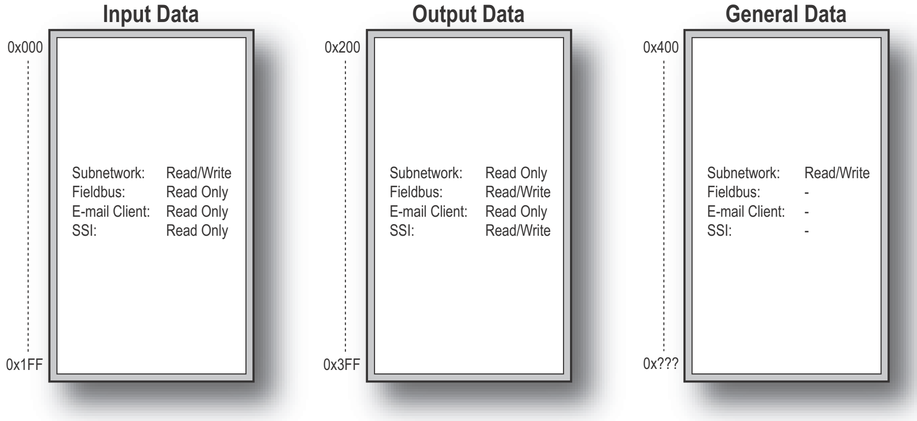
  (Data representation on the higher level network is described later in this chapter).

- **General Data (up to 1024 bytes)**

  This area cannot be accessed from the higher level network, but can be used for transfers between individual nodes on the subnetwork, or as a general "scratch pad" for data. The actual size of this area depends on the amount of data that is exchanged on the subnetwork. The gateway can handle up to 1024 bytes of general data.

2.2.1 Memory Map

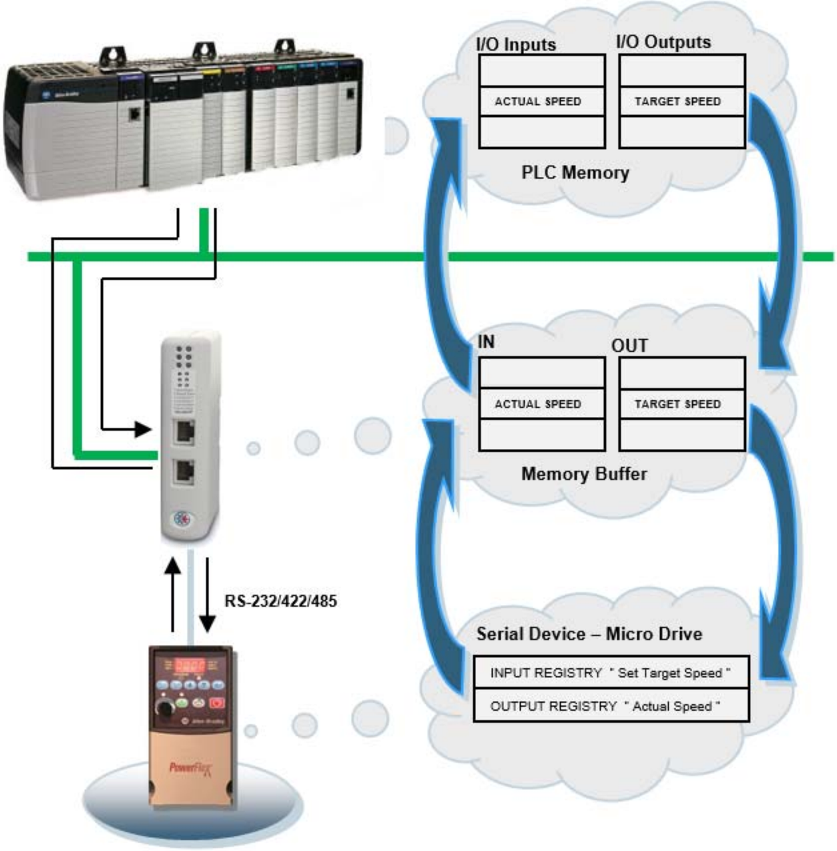
When building the subnetwork configuration using the configuration manager, the different areas described above are mapped to the memory locations (addresses) specified below.



Input Data		Output Data		General Data	
Subnetwork:	Read/Write	Subnetwork:	Read Only	Subnetwork:	Read/Write
Fieldbus:	Read Only	Fieldbus:	Read/Write	Fieldbus:	-
E-mail Client:	Read Only	E-mail Client:	Read Only	E-mail Client:	-
SSI:	Read Only	SSI:	Read/Write	SSI:	-

2.2.2 Data Exchange Example

In the following example, a micro drive on the subnetwork exchanges information with a PLC on the higher level network, via the internal memory buffers in the linking device.



2.3 Subnetwork Protocol

2.3.1 Protocol Modes

The EtherNet/IP to Serial linking device features three distinct operating modes for subnetwork communication: 'Master Mode', 'DF1 Master Mode' and 'Generic Data Mode'. Note that the protocol mode only specifies the basic communication model, not the actual subnetwork protocol.

- **Master Mode**

  In this mode, the linking device acts as a master on the subnetwork, and the serial communication is query-response based. The nodes on the network are not permitted to issue messages unless first addressed by the linking device.

  For more information about this mode, see "Master Mode" on page 18.

- **DF1 Master Mode**

  In this mode, the linking device acts as a master on the subnetwork, using the DF1 protocol. The serial communication is query-response based. For more information about this mode, see "DF1 Protocol Mode" on page 58.

- **Generic Data Mode**

  In this mode, there is no master-slave relationship between the subnetwork nodes and the linking device; any node on the subnetwork, including the linking device, may spontaneously produce or consume messages.

  For more information about this mode, see "Generic Data Mode" on page 19.

2.3.2 Protocol Building Blocks

The following building blocks are used in the configuration manager to describe the subnetwork communication. How these blocks apply to the three protocol modes is described later in this document.

- **Node**

  A 'node' represents a single device on the subnetwork. Each node can be associated with a number of transactions, see below.

- **Transaction**

  A 'transaction' represents a complete serial telegram, and consists of a number of frame objects (see below). Each transaction is associated with a set of parameters controlling how and when to use it on the subnetwork.

- **Commands**

  A 'command' is simply a predefined transaction stored in a list in the configuration manager. This simplifies common operations by allowing transactions to be stored and reused.

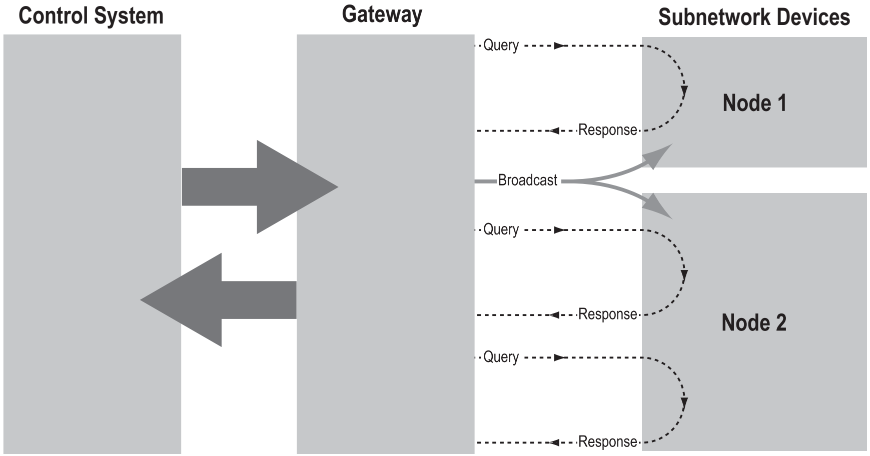
- **Frame Object**

  'Frame objects' are low level entities used to compose a transaction (see above). A frame object can represent a fixed value (a constant), a range of values (limit objects), a block of data or a calculated checksum.

2.3.3 Master Mode

In this mode, the communication is based on a query-response scheme; when the device issues a query on the subnetwork, the addressed node is expected to issue a response. Nodes are not permitted to issue responses/messages spontaneously, i.e. without first receiving a query.

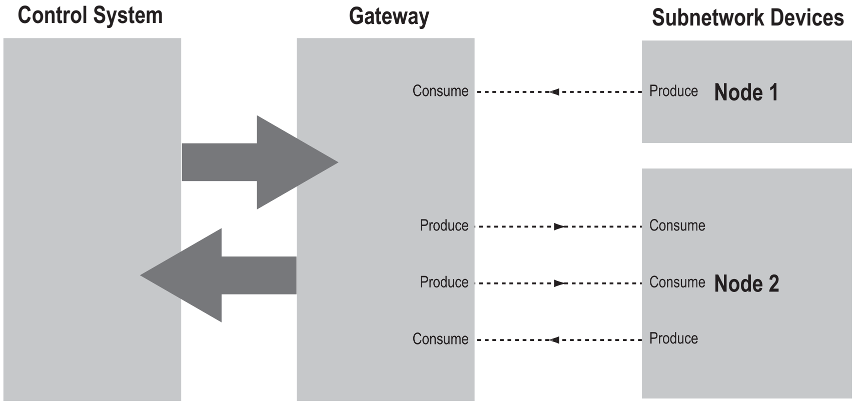
There is, however, one exception to this rule; the broadcaster. Most protocols offer some way of broadcasting messages to all nodes on the network, without expecting them to respond to the broadcasted message. This is also reflected in the device, which features a dedicated broadcaster node.



In Master Mode, the configuration manager comes preloaded with the most commonly used Modbus RTU commands, which can be conveniently reached by right-clicking on a node in the configuration manager and selecting 'Insert New Command'. Note, however, that this in no way prevents other protocols based on the same query-response message scheme from also being implemented.

2.3.4 Generic Data Mode

In this mode, there is no master-slave relationship between the nodes on the subnetwork and the device. Any node (including the linking device) may spontaneously produce or consume a message. Nodes are not obliged to respond to messages, nor do they need to wait for a query in order to send a message.

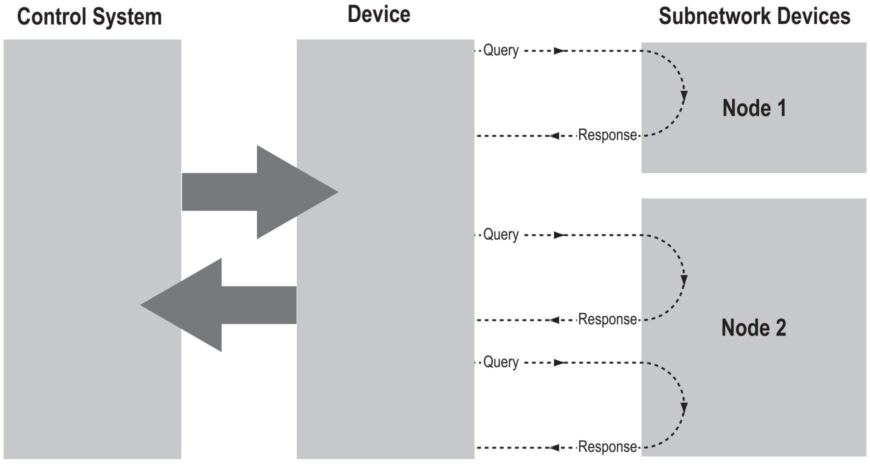


In the figure above, the linking device 'consumes' data 'produced' by a node on the subnetwork. This 'consumed' data can then be accessed from the higher level network. This also works the other way around; the data received from the higher level network is used to 'produce' a message on the subnetwork, for 'consumption' by a node.

2.3.5 DF1 Master Mode

In DF1 master mode, communication is based on "services". A "service" represents a set of commands and operations on the subnetwork, that is predefined in the linking device. Each service is associated with a set of parameters controlling how and when to use it on the subnetwork.

The communication is based on a query-response scheme, where the device issues a query on the subnetwork. The addressed node on the subnetwork is expected to issue a response to that query. Nodes are not permitted to issue responses spontaneously, i. e. without first receiving a query.



In DF1 Master Mode, the configuration manager comes preloaded with a number of services, that can be selected by the user. The actual DF1 commands, that perform the services during runtime, are predefined in the device. The configuration of the services is performed by right-clicking on a node in the configuration manager and selecting "Add Command".

For more information, please refer to "DF1 Protocol Mode" on page 58.

2.4 Linking Device IP Address Configuration

The linking device can retrieve the TCP/IP settings from a DHCP or BootP server.

The linking device also supports the HICP protocol used by the Anybus IPconfig tool. With this tool, it is possible to see and alter the TCP/IP settings for the linking device manually. The Anybus IPconfig tool can be downloaded on http://www.encompass.hms-networks.com/support.

2.5 EtherNet/IP

2.5.1 General

EtherNet/IP is based on the Control and Information Protocol (CIP), which is also the application layer for DeviceNet and ControlNet. The linking device acts as a Group 2 or 3 server on the EtherNet/IP network.

Input and output data is accessed using I/O connections or explicit messages towards the assembly object and the parameter input/output mapping objects.

See also...

- “CIP Object Implementation” on page 88
- “Logix Network Interface” on page 36

2.5.2 Data Types

The input and output data hold two types of data; I/O data and parameter data. I/O data is exchanged on change of value, and can be accessed using I/O connections towards the assembly object.

Parameter data can be accessed acyclically via the parameter input and output mapping objects. Note, however, that each instance attribute within these objects must be created manually using the configuration manager.

For more information see “Parameter Data Initialization (Explicit Data)” on page 121.

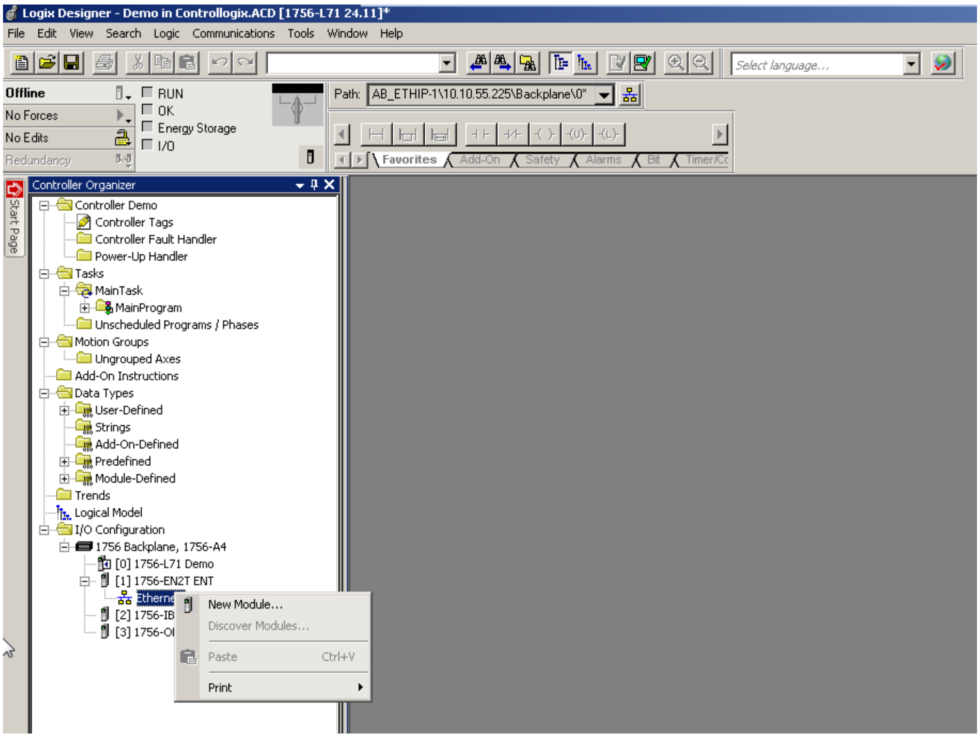
See also...

- “Assembly Object, Class 04h” on page 90
- “Parameter Data Input Mapping Object, Class B0h” on page 95
- “Parameter Data Output Mapping Object, Class B1h” on page 96
- “Logix Network Interface” on page 36

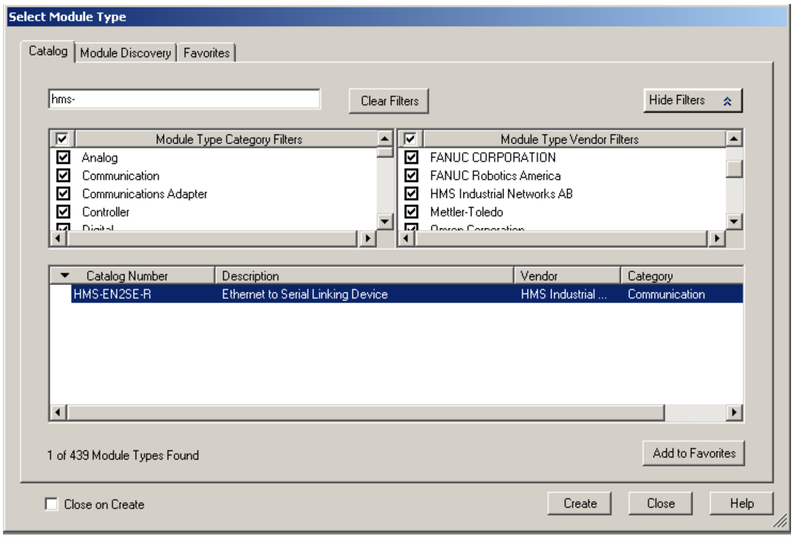
3. Studio 5000 Implementation Example

This section will guide you through all steps included in creating a basic configuration.

1. Start the Studio 5000 software. Expand the "I/O Configuration" folder in the tree view. Right-click "Ethernet" and select "New Module".



2. Select the HMS-EN2SE-R linking device and click "Create".



Search: hms-

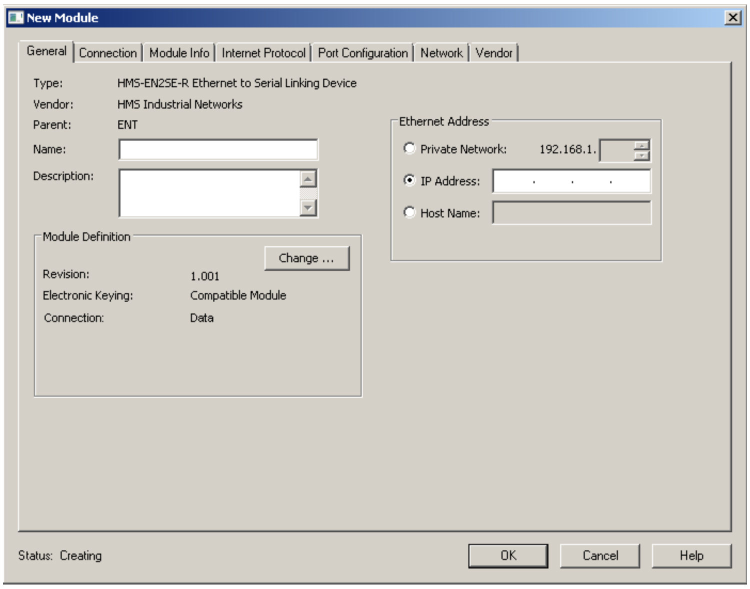
Module Type Category Filters	Module Type Vendor Filters
☑ Analog	☑ FANUC CORPORATION
☑ Communication	☑ FANUC Robotics America
☑ Communications Adapter	☑ HMS Industrial Networks AB
☑ Controller	☑ Mettler-Toledo
☑ Digital	☑ Omron Corporation

Catalog Number	Description	Vendor	Category
HMS-EN2SE-R	Ethernet to Serial Linking Device	HMS Industrial ...	Communication

1 of 439 Module Types Found

□ Close on Create

3. In the "New Module" window, assign a name to the module. The IP-address should be set via the BOOTP-DHCP server and entered in the IP-address field. Click "Change" in the "Module Definition" window.


New Module

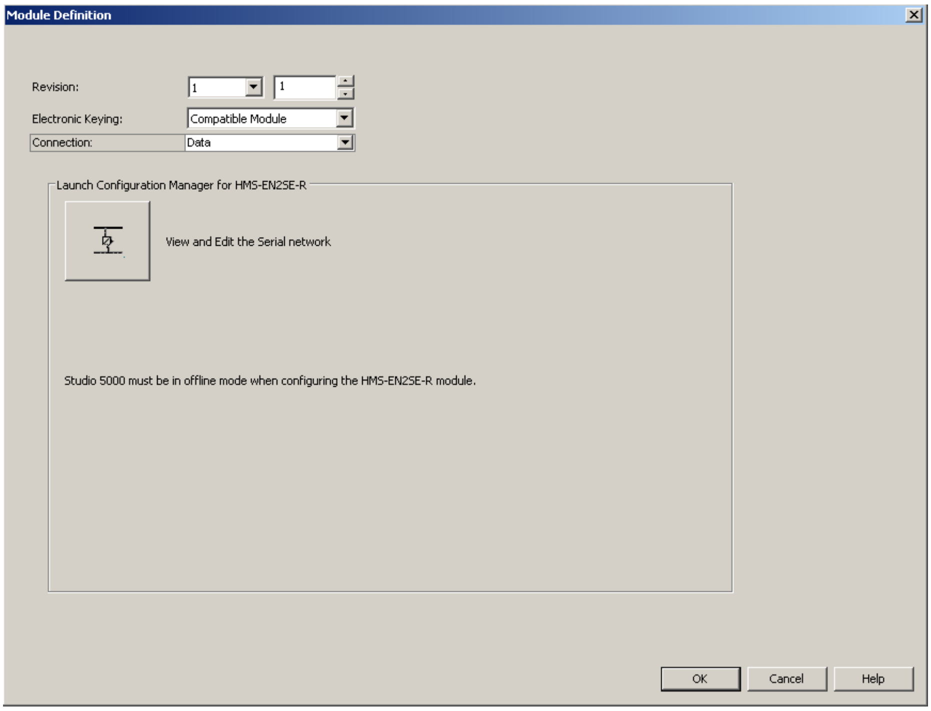
Type: HMS-EN2SE-R Ethernet to Serial Linking Device
Vendor: HMS Industrial Networks
Parent: ENT
Name:
Description:

Module Definition
Revision: 1.001
Electronic Keying: Compatible Module
Connection: Data

Ethernet Address
○ Private Network: 192.168.1.
● IP Address: . . .
○ Host Name:

Status: Creating

4. In the "Module Definition" window, launch the configuration manager for the HMS-EN2SE-R linking device.

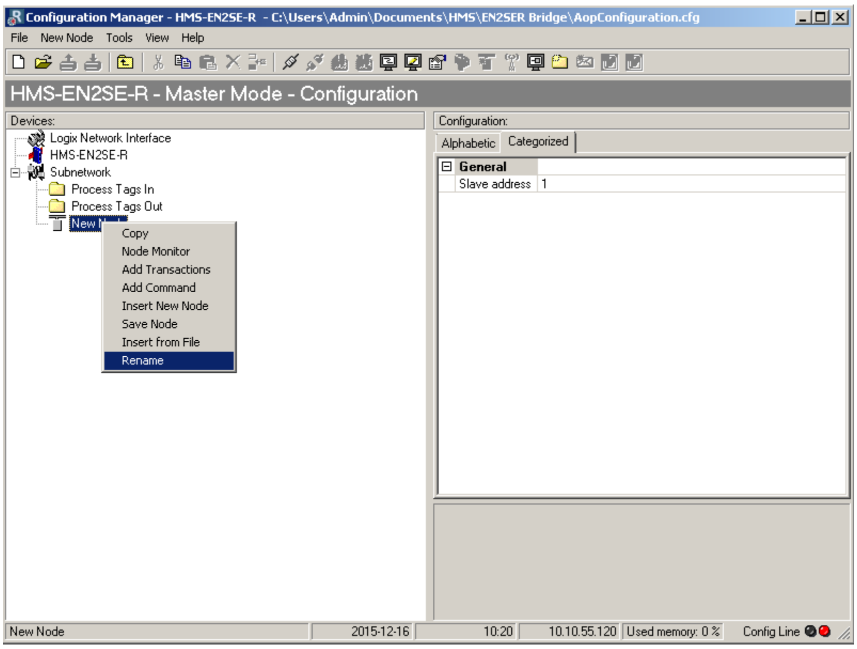

Module Definition

Revision: 1 1
Electronic Keying: Compatible Module
Connection: Data

Launch Configuration Manager for HMS-EN2SE-R
View and Edit the Serial network

Studio 5000 must be in offline mode when configuring the HMS-EN2SE-R module.

5. In the configuration manager, expand "Subnetwork". Right-click "New Node" and enter the name and the slave node address of the slave device.



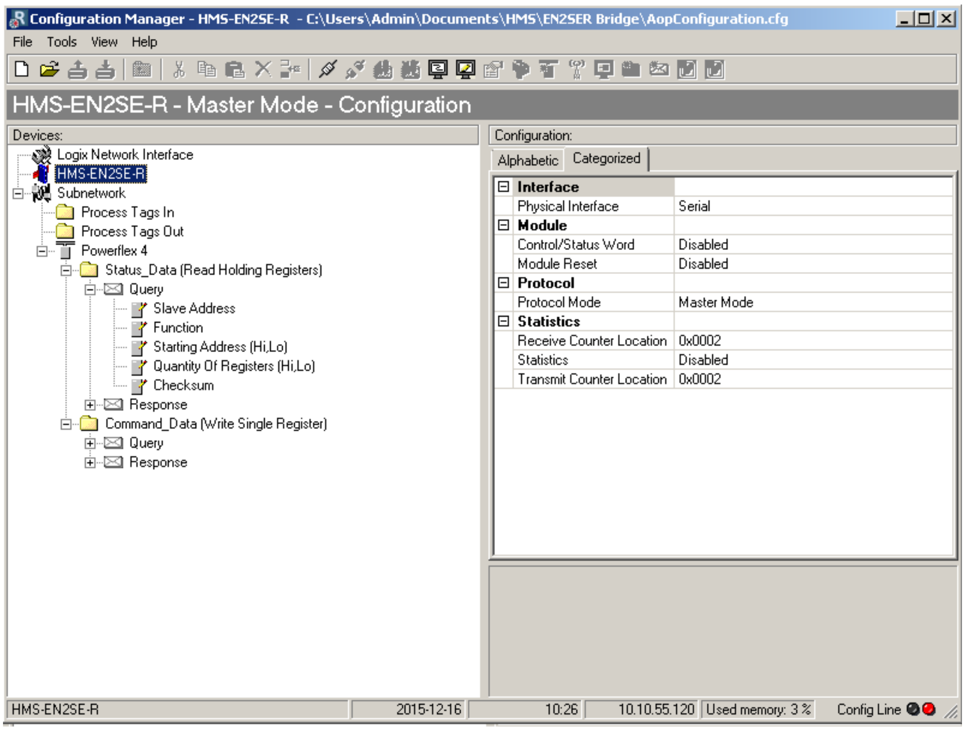
Configuration Manager - HMS-EN2SE-R - C:\Users\Admin\Documents\HMS\EN2SER Bridge\AopConfiguration.cfg
File New Node Tools View Help
HMS-EN2SE-R - Master Mode - Configuration
Devices:
Configuration:
Alphabetic Categorized
Copy
Node Monitor
Add Transactions
Add Command
Insert New Node
Save Node
Insert from File
Rename

```
Logix Network Interface
HMS-EN2SE-R
Subnetwork
    Process Tags In
    Process Tags Out
    New Node
```

General	
Slave address	1

New Node | 2015-12-16 | 10:20 | 10.10.55.120 | Used memory: 0 % | Config Line

6. To change general settings for the linking device, click "HMS-EN2SE-R".



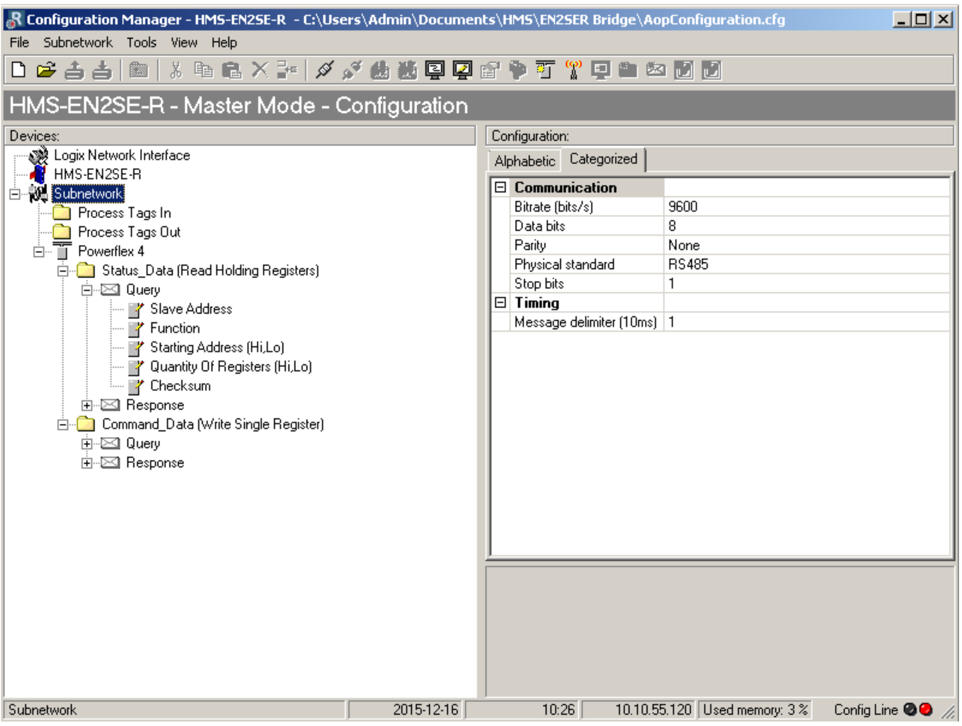
Configuration Manager - HMS-EN2SE-R - C:\Users\Admin\Documents\HMS\EN2SER Bridge\AopConfiguration.cfg
File Tools View Help
HMS-EN2SE-R - Master Mode - Configuration
Devices:
Configuration:
Alphabetic Categorized

```
Logix Network Interface
HMS-EN2SE-R
Subnetwork
    Process Tags In
    Process Tags Out
    Powerflex 4
        Status_Data (Read Holding Registers)
            Query
                Slave Address
                Function
                Starting Address (Hi,Lo)
                Quantity Of Registers (Hi,Lo)
                Checksum
            Response
        Command_Data (Write Single Register)
            Query
            Response
```

Interface	
Physical Interface	Serial
Module	
Control/Status Word	Disabled
Module Reset	Disabled
Protocol	
Protocol Mode	Master Mode
Statistics	
Receive Counter Location	0x0002
Statistics	Disabled
Transmit Counter Location	0x0002

HMS-EN2SE-R | 2015-12-16 | 10:26 | 10.10.55.120 | Used memory: 3 % | Config Line

7. To change communication and timing settings for the serial network, click "Subnetwork".



Configuration Manager - HMS-EN2SE-R - C:\Users\Admin\Documents\HMS\EN2SER Bridge\AopConfiguration.cfg
File Subnetwork Tools View Help
HMS-EN2SE-R - Master Mode - Configuration
Devices:
Logix Network Interface
HMS-EN2SE-R
Subnetwork
Process Tags In
Process Tags Out
Powerflex 4
Status_Data (Read Holding Registers)
Query
Slave Address
Function
Starting Address (Hi,Lo)
Quantity Of Registers (Hi,Lo)
Checksum
Response
Command_Data (Write Single Register)
Query
Response
Configuration:
Alphabetic
Categorized
Subnetwork
2015-12-16
10:26
10.10.55.120
Used memory: 3 %
Config Line

Communication	
Bitrate (bits/s)	9600
Data bits	8
Parity	None
Physical standard	RS485
Stop bits	1
Timing	
Message delimiter (10ms)	1

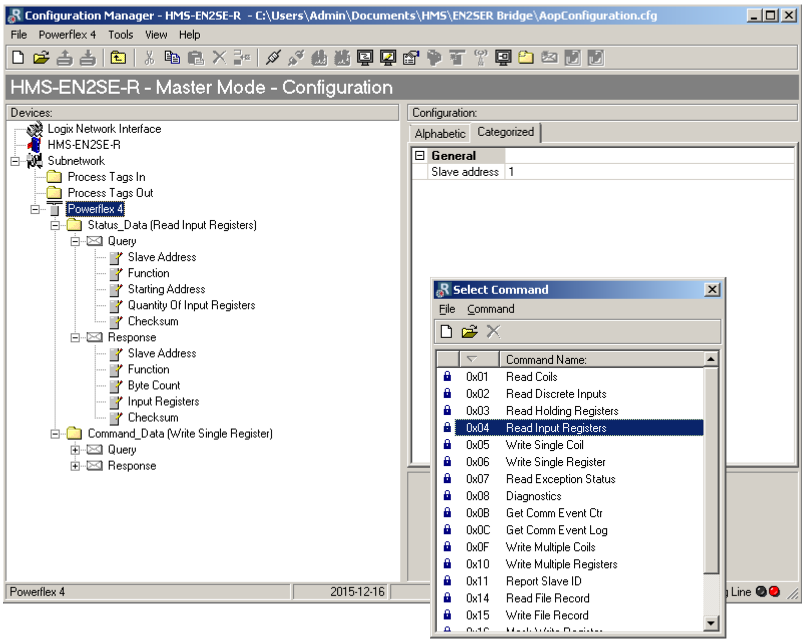
8. The remainder of the guide will assume the slave to be a Modbus client. Modbus commands can be added to the configuration by right-clicking the node and selecting "Add Command". Select the desired command in the "Select Command" window (See the Modbus Specification for complete information about Modbus commands). Rename the command and modify it according to the slave node's user manual. In this example, a "Read Input Registers" command "Status_Data" is added to the configuration.

- Set "Quantity of Input Registers" to 0x0001.

- Set "Byte Count" to 0x0002 (this value depends on the number of registers chosen in "Quantity of Input Registers").

- Set "Data Length" to 0x0002.

- Set "Data Location" to 0x0000. This points to where the resulting data will be located in the process data area.

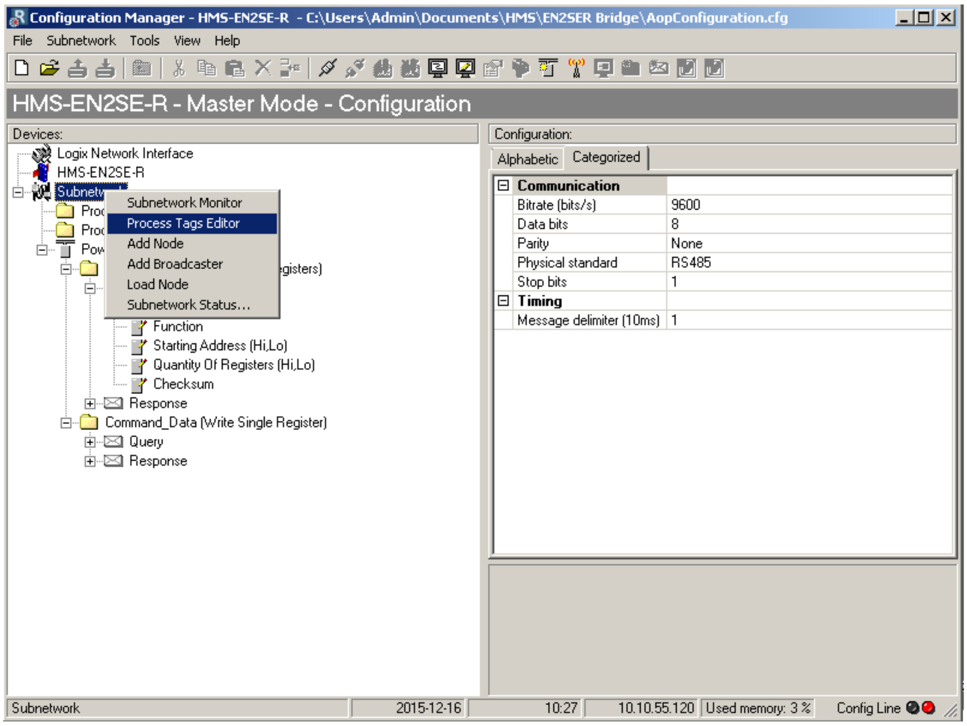

Configuration Manager - HMS-EN2SE-R - C:\Users\Admin\Documents\HMS\EN2SER Bridge\AopConfiguration.cfg
File
Powerflex 4
Tools
View
Help
HMS-EN2SE-R - Master Mode - Configuration
Devices:
Configuration:
Alphabetic
Categorized
Select Command
File
Command
Powerflex 4
2015-12-16
Line

Logix Network Interface
HMS-EN2SE-R
Subnetwork
Process Tags In
Process Tags Out
Powerflex 4
Status_Data (Read Input Registers)
Query
Slave Address
Function
Starting Address
Quantity Of Input Registers
Checksum
Response
Slave Address
Function
Byte Count
Input Registers
Checksum
Command_Data (Write Single Register)
Query
Response

General	
Slave address	1

		Command Name:
🔒	0x01	Read Coils
🔒	0x02	Read Discrete Inputs
🔒	0x03	Read Holding Registers
🔒	0x04	Read Input Registers
🔒	0x05	Write Single Coil
🔒	0x06	Write Single Register
🔒	0x07	Read Exception Status
🔒	0x08	Diagnostics
🔒	0x0B	Get Comm Event Ctr
🔒	0x0C	Get Comm Event Log
🔒	0x0F	Write Multiple Coils
🔒	0x10	Write Multiple Registers
🔒	0x11	Report Slave ID
🔒	0x14	Read File Record
🔒	0x15	Write File Record
🔒	0x16	Mask Write Register

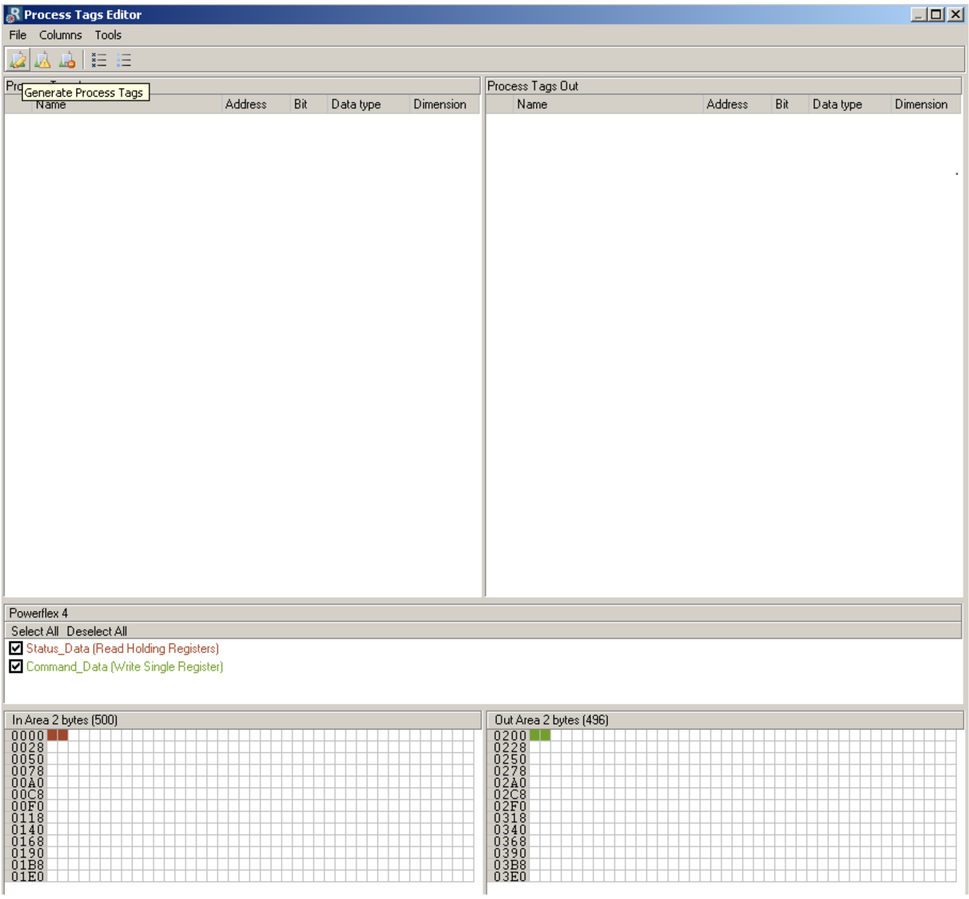
9. Open the "Process Tags Editor" by right-clicking "Subnetwork".


Configuration Manager - HMS-EN2SE-R - C:\Users\Admin\Documents\HMS\EN2SER Bridge\AopConfiguration.cfg
File Subnetwork Tools View Help
HMS-EN2SE-R - Master Mode - Configuration
Devices:
Logix Network Interface
HMS-EN2SE-R
Subnetwork
Subnetwork Monitor
Process Tags Editor
Add Node
Add Broadcaster
Load Node
Subnetwork Status...
Function
Starting Address (Hi,Lo)
Quantity Of Registers (Hi,Lo)
Checksum
Response
Command_Data (Write Single Register)
Query
Response
Configuration:
Alphabetic Categorized

Communication	
Bitrate (bits/s)	9600
Data bits	8
Parity	None
Physical standard	RS485
Stop bits	1
Timing	
Message delimiter (10ms)	1

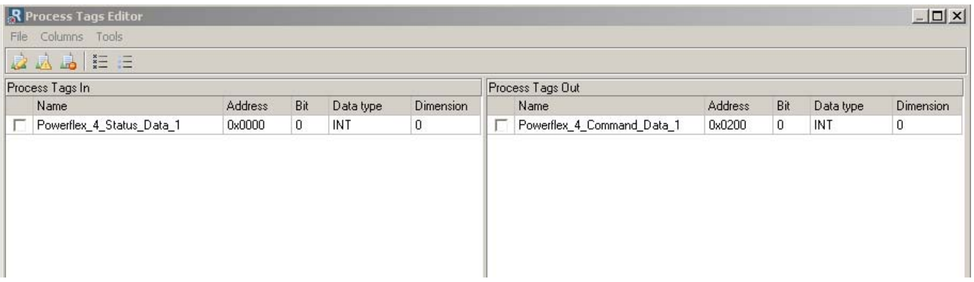
Subnetwork | 2015-12-16 | 10:27 | 10.10.55.120 | Used memory: 3 % | Config Line

10. The configuration made in the previous steps is visible in the in and out areas in the bottom section. To generate process tags for the configuration, click "Tools" and then "Generate Process Tags".


Process Tags Editor
File Columns Tools
Generate Process Tags
Name Address Bit Data type Dimension
Process Tags Out
Name Address Bit Data type Dimension
In Area 2 bytes (500)
0000 0028 0050 0078 00A0 00C8 00F0 0118 0140 0168 0190 01B8 01E0
Out Area 2 bytes (496)
0200 0228 0250 0278 02A0 02C8 02F0 0318 0340 0368 0390 03B8 03E0

Powerflex 4
Select All Deselect All
☑ Status_Data (Read Holding Registers)
☑ Command_Data (Write Single Register)

11. The generated process tags will be derived from the slave node's name and the Modbus command names. They can also be altered manually. When done, click "File" and "Exit".



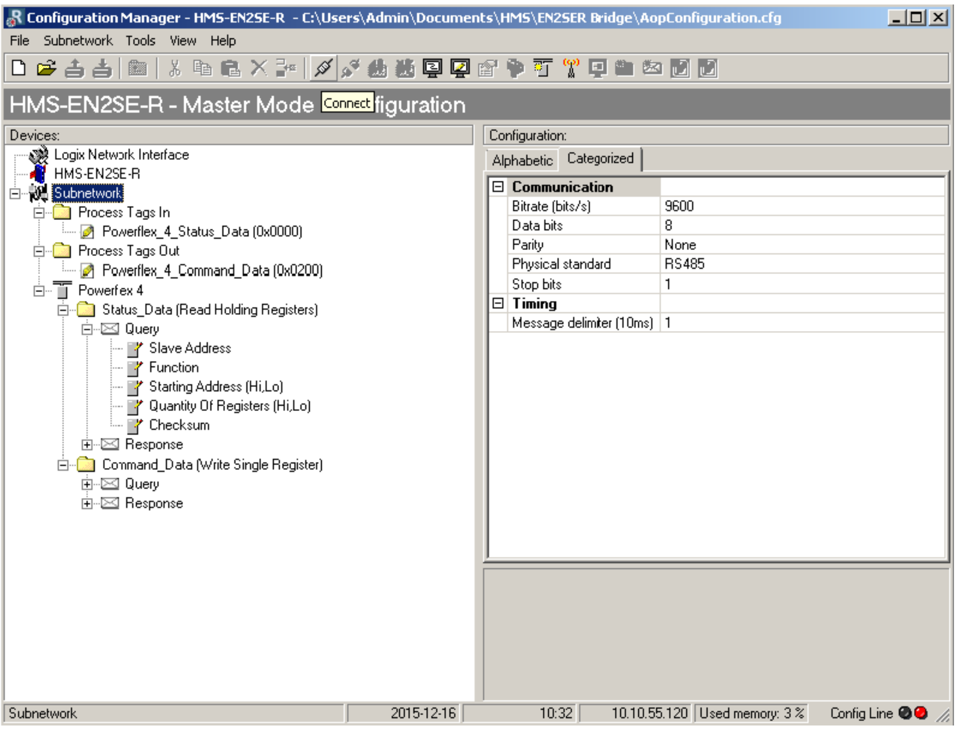
Process Tags In

	Name	Address	Bit	Data type	Dimension
□	Powerflex_4_Status_Data_1	0x0000	0	INT	0

Process Tags Out

	Name	Address	Bit	Data type	Dimension
□	Powerflex_4_Command_Data_1	0x0200	0	INT	0

12. Connect to the linking device by clicking the connect button in the toolbar.

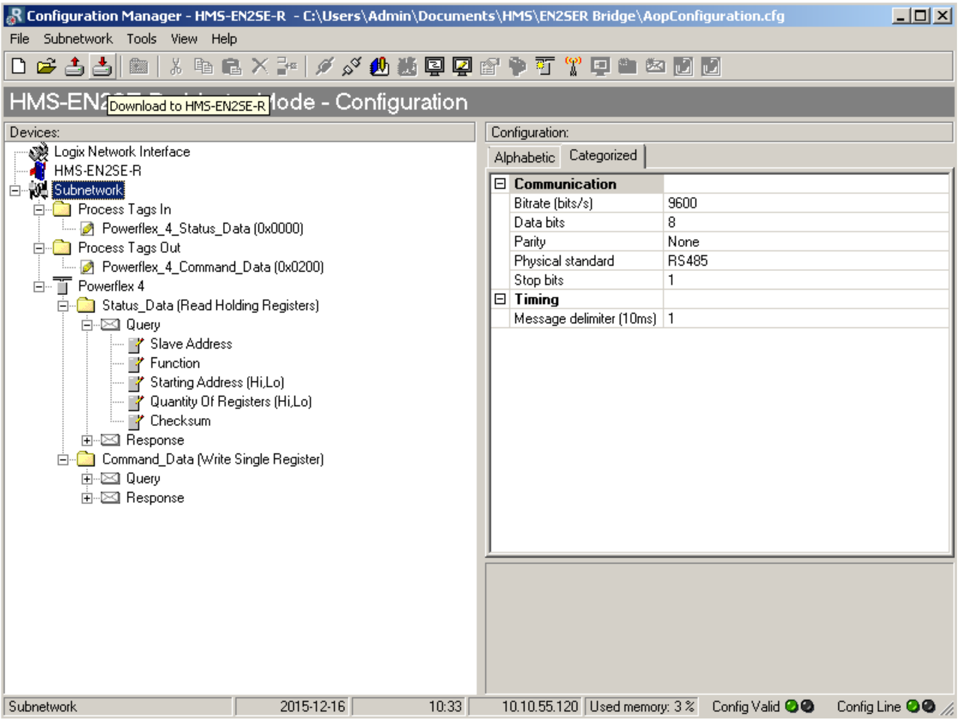

Configuration Manager - HMS-EN2SE-R - C:\Users\Admin\Documents\HMS\EN2SER Bridge\AopConfiguration.cfg
File Subnetwork Tools View Help
HMS-EN2SE-R - Master Mode Connect figuration
Connect
Devices:
Configuration:
Alphabetic Categorized

```
Logix Network Interface
HMS-EN2SE-R
Subnetwork
  Process Tags In
    Powerflex_4_Status_Data (0x0000)
  Process Tags Out
    Powerflex_4_Command_Data (0x0200)
  Powerflex 4
    Status_Data (Read Holding Registers)
      Query
        Slave Address
        Function
        Starting Address (Hi,Lo)
        Quantity Of Registers (Hi,Lo)
        Checksum
      Response
    Command_Data (Write Single Register)
      Query
      Response
```

Communication	
Bitrate (bits/s)	9600
Data bits	8
Parity	None
Physical standard	RS485
Stop bits	1
Timing	
Message delimiter (10ms)	1

Subnetwork | 2015-12-16 | 10:32 | 10.10.55.120 | Used memory: 3 % | Config Line

13. After a connection has been established, download the configuration to the linking device using the download button in the toolbar.

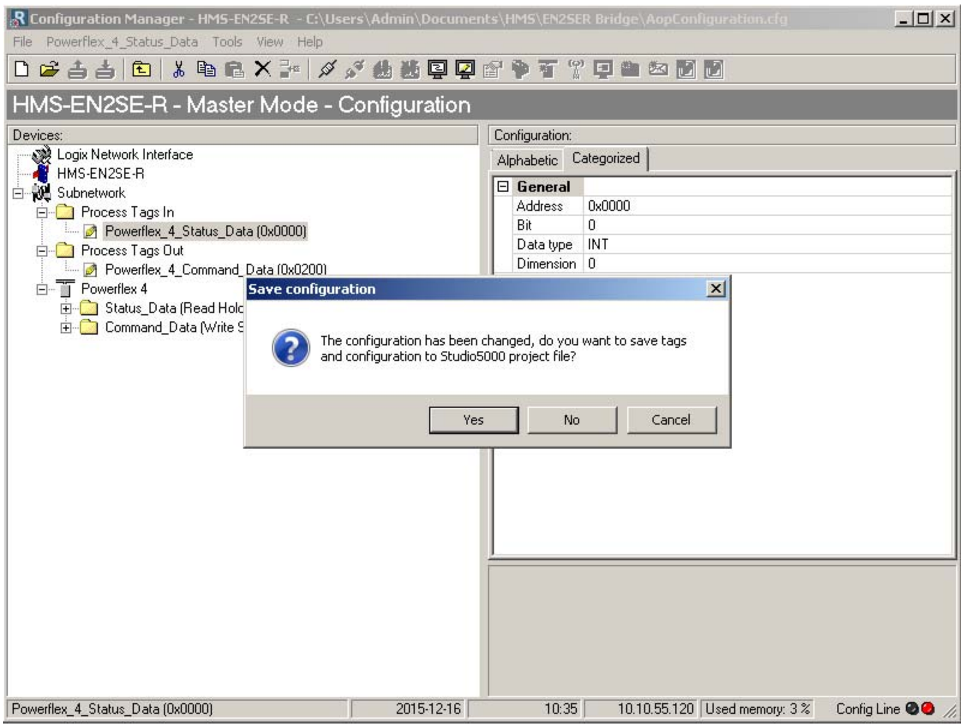

Configuration Manager - HMS-EN2SE-R - C:\Users\Admin\Documents\HMS\EN2SER Bridge\AopConfiguration.cfg
File Subnetwork Tools View Help
HMS-EN2
Download to HMS-EN2SE-R
lode - Configuration
Devices:
Configuration:
Alphabetic
Categorized

```
Logix Network Interface
HMS-EN2SE-R
Subnetwork
    Process Tags In
        Powerflex_4_Status_Data (0x0000)
    Process Tags Out
        Powerflex_4_Command_Data (0x0200)
    Powerflex 4
        Status_Data (Read Holding Registers)
            Query
                Slave Address
                Function
                Starting Address (Hi,Lo)
                Quantity Of Registers (Hi,Lo)
                Checksum
            Response
        Command_Data (Write Single Register)
            Query
            Response
```

Communication	
Bitrate (bits/s)	9600
Data bits	8
Parity	None
Physical standard	RS485
Stop bits	1
Timing	
Message delimiter (10ms)	1

```
Subnetwork | 2015-12-16 | 10:33 | 10.10.55.120 | Used memory: 3 % | Config Valid | Config Line
```

14. Exit the configuration manager and click "Yes" to apply all recent changes.


Configuration Manager - HMS-EN2SE-R - C:\Users\Admin\Documents\HMS\EN2SER Bridge\AopConfiguration.cfg
File Powerflex_4_Status_Data Tools View Help
HMS-EN2SE-R - Master Mode - Configuration
Devices:
Configuration:
Alphabetic
Categorized
Save configuration
Yes
No
Cancel

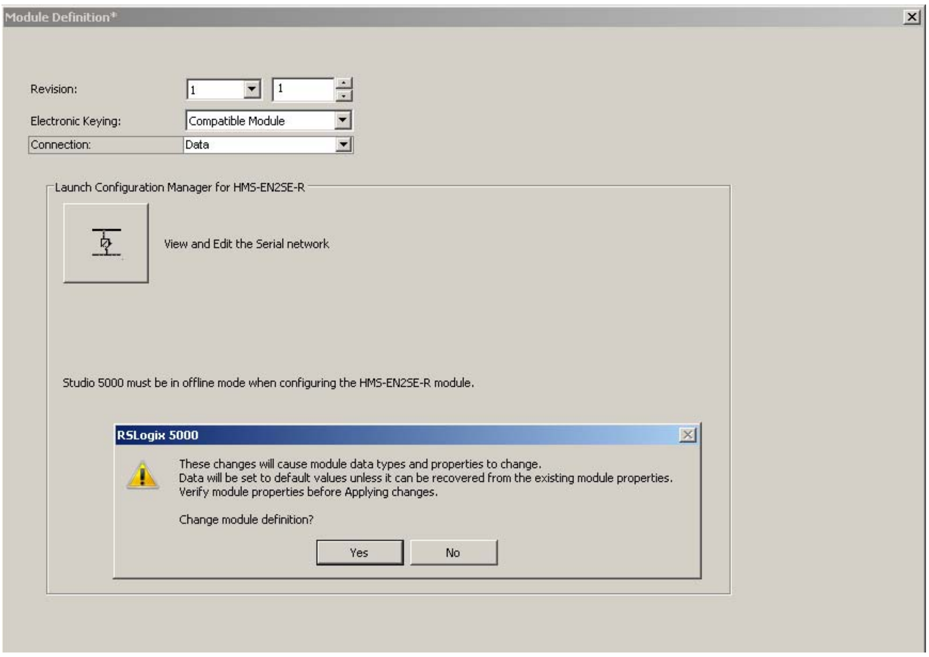
```
Logix Network Interface
HMS-EN2SE-R
Subnetwork
    Process Tags In
        Powerflex_4_Status_Data (0x0000)
    Process Tags Out
        Powerflex_4_Command_Data (0x0200)
    Powerflex 4
        Status_Data (Read Holc
        Command_Data (Write S
```

General	
Address	0x0000
Bit	0
Data type	INT
Dimension	0

The configuration has been changed, do you want to save tags and configuration to Studio5000 project file?

```
Powerflex_4_Status_Data (0x0000) | 2015-12-16 | 10:35 | 10.10.55.120 | Used memory: 3 % | Config Line
```

15. Click "Yes" to update the module definition in Studio 5000.



Module Definition*

Revision: 1 1

Electronic Keying: Compatible Module

Connection: Data

Launch Configuration Manager for HMS-EN2SE-R

View and Edit the Serial network

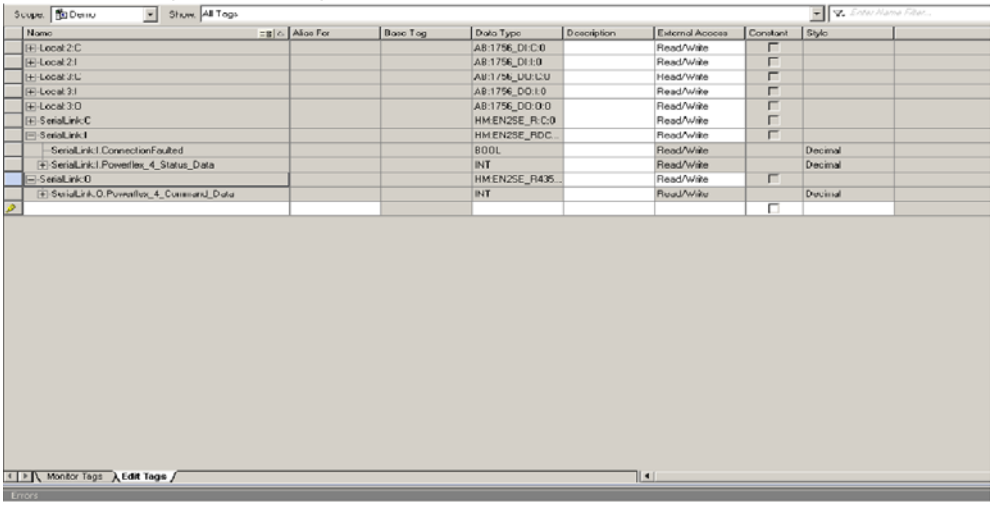
Studio 5000 must be in offline mode when configuring the HMS-EN2SE-R module.

RSLogix 5000

These changes will cause module data types and properties to change.
Data will be set to default values unless it can be recovered from the existing module properties.
Verify module properties before Applying changes.

Change module definition?

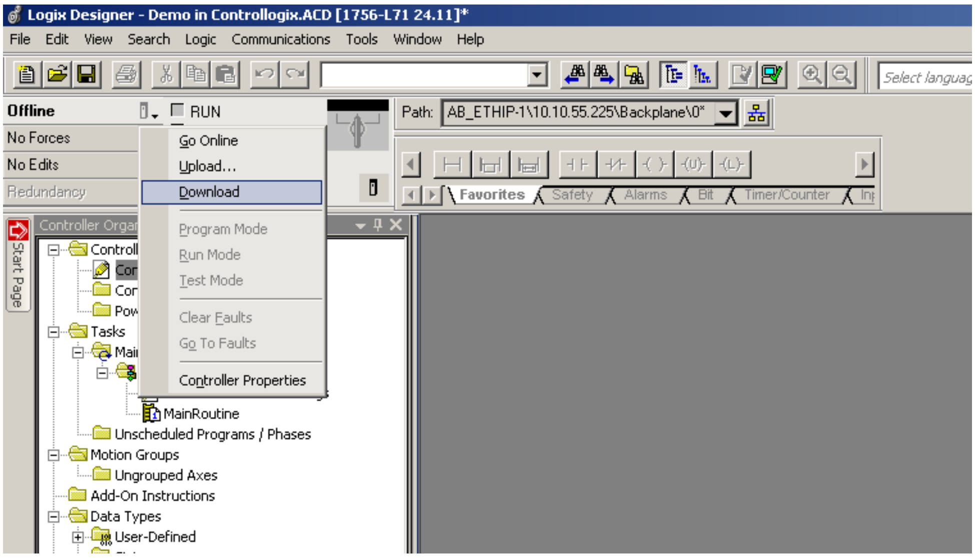
16. In the Controller Organizer, click "Controller Tags". Expand "SerialLink" to see the process tags created in the configuration manager.



Scope: Demo Show: All Tags

Name	Alias For	Base Tag	Data Type	Description	External Access	Constant	Style
⊞-Local:2:C			AB:1756_DI:C:0		Read/Write	□	
⊞-Local:2:I			AB:1756_DI:I:0		Read/Write	□	
⊞-Local:3:C			AB:1756_DO:C:0		Read/Write	□	
⊞-Local:3:I			AB:1756_DO:I:0		Read/Write	□	
⊞-Local:3:O			AB:1756_DO:O:0		Read/Write	□	
⊞-SerialLink:C			HM:EN2SE_R:C:0		Read/Write	□	
⊟-SerialLink:I			HM:EN2SE_RDC...		Read/Write	□	
SerialLink:I.ConnectionFaulted			BOOL		Read/Write		Decimal
⊞-SerialLink:I.Powerflex_4_Status_Data			INT		Read/Write		Decimal
⊟-SerialLink:O			HM:EN2SE_R435...		Read/Write	□	
⊞-SerialLink:O.Powerflex_4_Command_Data			INT		Read/Write		Decimal
						□	

17. Download the configuration to the Studio 5000 project by right-clicking the computer icon and then "Download".

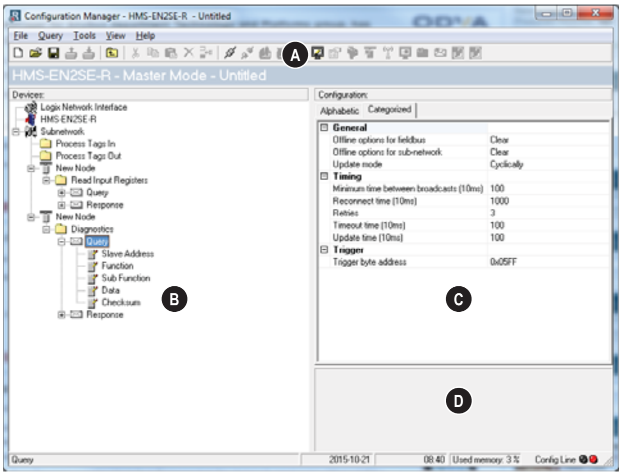


18. Run the demo in Studio 5000.

4. Navigating the Configuration Manager

4.1 Main Window

The main window in the EN2SE-R configuration manager can be divided into 4 sections as follows:



General	
Offline options for fieldbus	Clear
Offline options for sub-network	Clear
Update mode	Cyclically
Timing	
Minimum time between broadcasts (10ms)	100
Reconnect time (10ms)	1000
Retries	3
Timeout time (10ms)	100
Update time (10ms)	100
Trigger	
Trigger byte address	0x05FF

- **A: Drop-down Menus & Tool Bar**

  The second drop-down menu from the left will change depending on the current context. The Tool Bar provides quick access to the most frequently used functions.

- **B: Navigation Section**

  This section is the main tool for selecting and altering different levels of the subnetwork configuration.

  Entries preceded by a “+” holds further configuration parameters or “submenus”. To gain access to these parameters, the entry must be expanded by clicking “+”.

  There are three main levels in the navigation window, namely Logix Network Interface, HMS-EN2SE-R, and Subnetwork.

  Right-clicking on entries in this section brings out additional selections related to that particular entry.

- **C: Parameter Section**

  This section holds a list of parameters or options related to the currently selected entry in the Navigation Section.

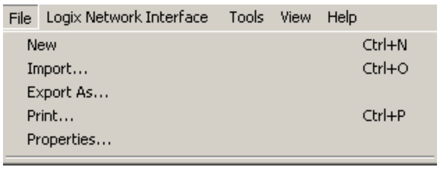
  The parameter value may be specified either using a selection box or manually, depending on the parameter itself. Values can be specified in decimal form (e.g. “42”), or in hexadecimal format (e.g. “0x2A”).

- **D: Information Section**

  This section holds information related to the currently selected parameter.

4.1.1 Drop-down Menus

File



- **New**

  Create a new configuration.

- **Import...**

  Import a previously created configuration.

- **Export As...**

  Export the current configuration under a new name.

- **Print...**

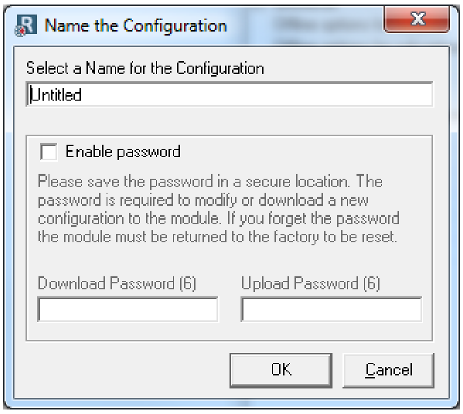
  Send details about the current configuration to a printer.

- **Properties...**

  Set the name and (optional) passwords for the configuration.

Item	Description
Select a Name for the Configuration	Enter a descriptive name for the new configuration
Enable Password	Enables password protection
Download Password(6)	Set passwords for downloading and uploading the configuration (max. 6 characters)
Upload Password(6)	

CAUTION: Always keep a copy of the password in a safe place. A lost password cannot be retrieved!



Select a Name for the Configuration

Untitled

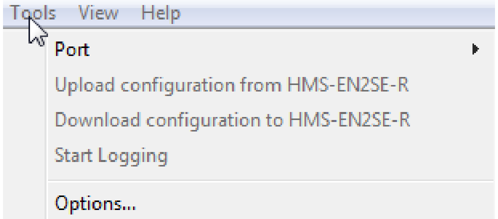
□ Enable password

Please save the password in a secure location. The password is required to modify or download a new configuration to the module. If you forget the password the module must be returned to the factory to be reset.

Download Password (6)

Upload Password (6)

Tools


Tools
View
Help
Port
Upload configuration from HMS-EN2SE-R
Download configuration to HMS-EN2SE-R
Start Logging
Options...

- **Port**

  Not used.

- **Upload configuration from HMS-EN2SE-R**

  Upload the configuration from the linking device to the configuration manager.

  Note: When uploading a configuration from the linking device the active tag structure in the configuration manager will be deleted.

- **Download configuration to HMS-EN2SE-R**

  Download the current configuration to the linking device.

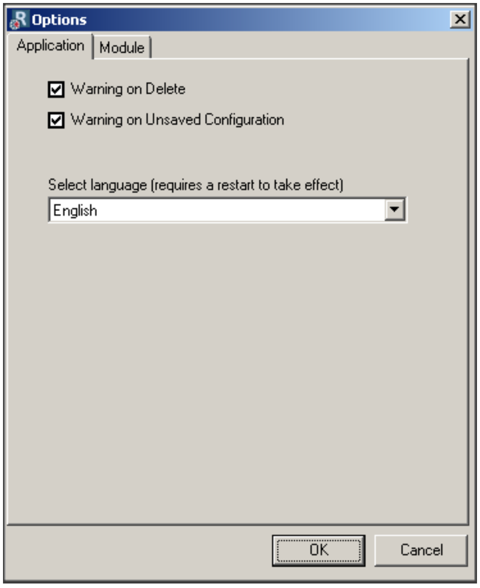
- **Start Logging**

  Start the Data Logger (see “Data Logger” on page 70).

  Note that when the Data Logger is active, this menu entry is changed to “Stop Logging”.

- **Options**

  This will open the following window:



☑ Warning on Delete

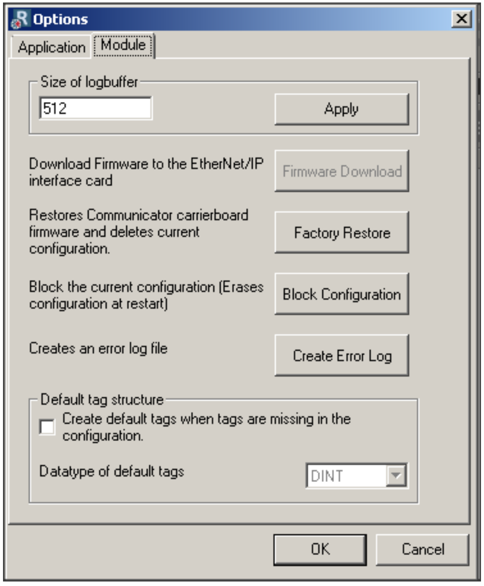
☑ Warning on Unsaved Configuration

Select language (requires a restart to take effect)

English

Item	Description
Warning on Delete	A confirmation dialog is displayed each time something is deleted.
Warning on Unsaved Configuration	A confirmation dialog is displayed when closing the configuration manager with unsaved data.
Select language	Selects which language to use. The new setting will be active the next time the application is launched.

Selecting the “Module” tab will reveal additional properties:



Size of logbuffer: 512

Download Firmware to the EtherNet/IP interface card

Restores Communicator carrierboard firmware and deletes current configuration.

Block the current configuration (Erases configuration at restart)

Creates an error log file

Default tag structure

□ Create default tags when tags are missing in the configuration.

Datatype of default tags: DINT

Item	Description
Size of logbuffer	By default, the Data Logger can log up to 512 entries in each direction. If necessary, it is possible to specify a different number of entries (valid settings range from 1...512). Click “Apply” to validate the new settings. See also “Data Logger” on page 70.
Firmware Download	Download firmware to the embedded network interface. Warning: Use with caution.
Factory Restore	Restores the device firmware to the original state (does not affect the embedded network interface).
Block Configuration	When selected, the downloaded configuration will not be executed by the device. Warning: Use with caution.
Create Error log	Creates an error log file
Default Tag Structure	Checking this box will automatically generate tags that to match the configuration, according to the selected data type.

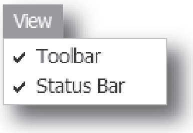
View

- **Toolbar**

  Enables/disables the toolbar icons at the top of the main window.

- **Status Bar**

  Enables/disables the status bar at the bottom of the main window.



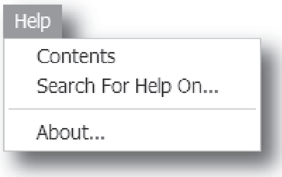
Help

- **Contents/Search For Help On...**

  Opens a built-in browser window with a link to the support website.

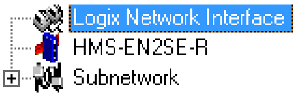
- **About...**

  Displays general information about the device and the current version of the configuration manager.



5. Basic Settings

5.1 Logix Network Interface

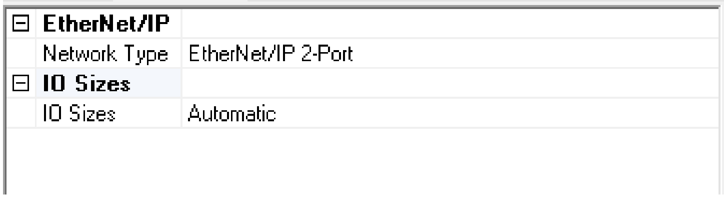


Logix Network Interface
HMS-EN2SE-R
Subnetwork

(Select 'Logix Network Interface' in the navigation section to gain access to the parameters described in this section).

General

During start-up the Logix network interface of the linking device is initialized to fit the configuration created in the EN2SE-R configuration manager.



⊟ **EtherNet/IP**	
Network Type	EtherNet/IP 2-Port
⊟ **IO Sizes**	
IO Sizes	Automatic

To be able to participate on the network, the following settings must be correctly made:

Network Type

This parameter is set to "EtherNet/IP 2-Port".

I/O Sizes

Specifies how data from the internal memory buffer will be exchanged over EtherNet/IP. This can either be handled automatically based on the subnetwork configuration, or specified manually.

See also "Linking Device IP Address Configuration" on page 20.

Value	Description
Automatic	All data will be represented as I/O Data on EtherNet/IP.
User defined	Additional parameter properties appear; "IO Size In" and "IO Size Out". The specified amount, starting at address 0x0000 of the respective memory buffers, will be reserved for and represented as I/O Data. The remainder will be reserved for Parameter Data.

5.2 EN2SE-R Parameters

Interface

Only serial communication is currently supported.

Control/Status Word

See "Control and Status Registers" on page 72.

Value	Description
Enabled	Enable the Control and Status Registers. The "Data Valid"-bit in the Control Register must be set to start the subnetwork communication.
Enabled but no startup lock	This setting is similar to "Enabled", except that the control system is not required to set the "Data Valid"-bit to start the subnetwork communication.
Disabled	This setting completely disables the Control and Status Registers.

Module Reset

This parameter specifies how the device will behave in the event of a fatal error.

Value	Description
Enabled	The device will be restarted, and no error will be indicated to the user.
Disabled	The device will halt and indicate an error.

Protocol Mode

This parameter specifies which protocol mode to use for the subnetwork. See "Protocol Modes" on page 17.

Value	Description
Generic Data Mode	This mode is primarily intended for Produce & Consume-based protocols, where there are no Master-Slave relationship between the gateway and the nodes on the subnetwork.
Master Mode	This mode is intended for "Query & Response"-based protocols, where a single Master exchanges data with a number of Slaves.
DF1	This mode is intended for the DF1 protocol. The linking device can only be configured as a Master with half-duplex communication. Note: This is the only mode available if you intend to configure the device for DF1.

Statistics

The Transmit- and Receive Counters indicate how many transactions that have successfully been exchanged on the subnetwork. This feature is primarily intended for debugging purposes.

- **Receive Counter Location**

  Specifies the location of the Receive Counter in the internal memory buffer.

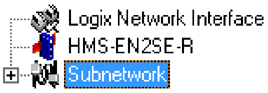
- **Statistics**

  Enables/disables the Receive and Transmit Counters.

- **Transmit Counter Location**

  Specifies the location of the Transmit Counter in the internal memory buffer.

5.3 Subnetwork Parameters



Communication

These parameters specify the actual communication settings used for the subnetwork.

Parameter	Description	Master Mode and Generic Mode
Bitrate (bits/s)	Selects the bit rate	1200 2400 4800 9600 19200 35700 38400 57600
Data bits	Selects the number of data bits	7, 8
Parity	Selects the parity mode	None, Odd, Even
Physical standard	Selects the physical interface type	RS232, RS422, RS485
Stop bits	Number of stop bits	1, 2

Start- and End Character

Note: These parameters are only available in Generic Data Mode.

Start and end characters are used to indicate the beginning and end of a serial message. For example, a message may be initiated with <ESC> and terminated with <LF>. In this case, the Start character would be 0x1B (ASCII code for <ESC>) and the End character 0x0A (ASCII code for <LF>)

Parameter	Description	Valid settings
End character value	End character for the message, ASCII	0x00–0xFF
Use End character	Determines if the End character shall be used or not	Enable / Disable
Start character value	Start character for the message, ASCII	0x00–0xFF
Use Start character	Determines if the Start character shall be used or not	Enable / Disable

Timing (Message Delimiter)

The parameters in this category differs slightly between the different protocol modes.

- **Master Mode**

  The Message Delimiter specifies the time that separates two messages in steps of 10 ms. If set to 0 (zero), the gateway will use the standard Modbus delimiter of 3.5 characters (the actual number of ms will be calculated automatically based on the currently used communication settings).

- **Generic Data Mode**

  The Message Delimiter specifies the time that separates two messages in steps of 10 µs.

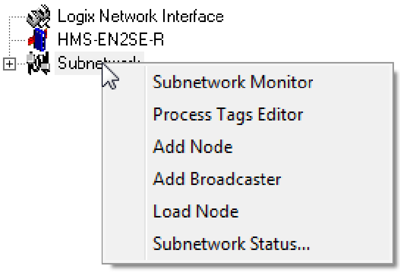
6. Nodes

6.1 General

In the configuration manager, a node represents a single device on the network. Although the device does not feature a scan list in the traditional sense, all nodes and their transactions will be processed in the order they were defined.

The maximum number of nodes that can be created in configuration manager is 31.

6.2 Adding & Managing Nodes

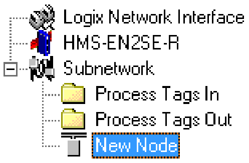


Function	Description
Subnetwork Monitor	Launch the subnet monitor (see “Subnetwork Monitor” on page 66)
Process Tags Editor	Launch the process tags editor
Add Node	Add a node to the configuration
Add Broadcaster[a]	Add a broadcaster node to the configuration
Load Node	Add a previously saved node
Subnetwork Status...	View diagnostic information about the subnetwork

a. This function is only available in Master Mode.

6.3 Node Parameters

6.3.1 Master Mode and Generic Data Mode



To gain access to the parameters described in this section, select a node in the Navigation Section.

Parameter	Description
Slave Address	The value entered here may be used to set the node address in certain commands. For more information, see “The Command Editor” on page 55.

7. Transactions

7.1 General

As mentioned previously, transactions are representations of the actual serial telegrams exchanged on the serial subnetwork. Although the device does not feature a scan list in the traditional sense, all nodes and their transactions will be processed in the order they were defined in the configuration manager.

Transactions are handled slightly differently in the three protocol modes:

- **Master Mode**

  For regular nodes, transactions always come in pairs; a query and a response. The query is issued by the device, while responses are issued by the slaves on the subnetwork. The Broadcaster can only send transactions.

- **Generic Data Mode**

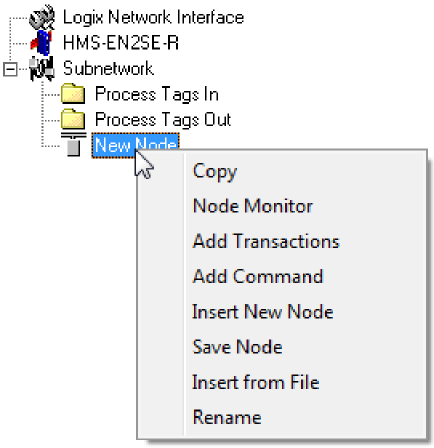
  Transactions can be added as desired for both directions. Transactions sent to the subnetwork are called “Transaction Produce”, and transactions issued by other nodes are called “Transaction Consume”.

- **DF1 Master Mode**

  Please refer to “DF1 Protocol Mode” on page 58.

Theoretically, the linking device supports up to 150 transactions. The actual number may however be less depending on the memory requirements of the defined transactions.

7.2 Adding & Managing Transactions

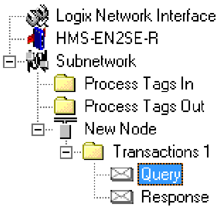


Function	Description
Copy	Copy a node to the clipboard
Delete[a]	Delete a node
Node Monitor	Launch the node monitor (see “Node Monitor” on page 67)
Add Transaction(s)[b]	On regular nodes, this adds a Query and a Response. The two transactions will be grouped in order to increase readability. On the Broadcaster, a single transaction will be added.
Add Transaction Consume[c]	Add a “Consume”-transaction
Add transaction Produce[c]	Add a “Produce”-transaction
Add Command	Add predefined transactions to the node
Insert New Node	Insert a new node above the currently selected one
Save Node	Save the selected node
Insert from File	Insert a previously saved node above the currently selected node
Rename	To increase readability, each node can be given a unique name using this function

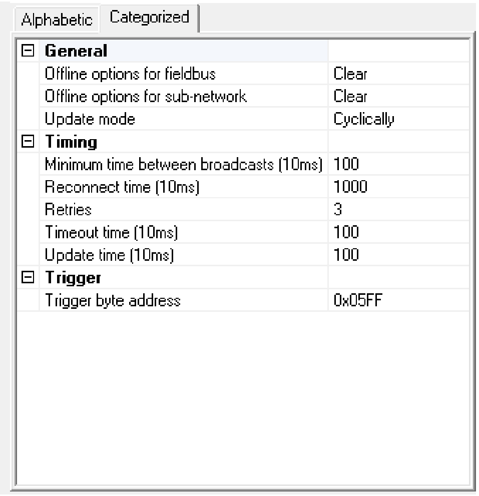
[a] Only available if more than one node exists
[b] Only available in Master Mode
[c] Only available in Generic Data Mode

7.3 Transaction Parameters (Master Mode)

7.3.1 Parameters (Query & Broadcast)



- Logix Network Interface
- HMS-EN2SE-R
- Subnetwork
  - Process Tags In
  - Process Tags Out
  - New Node
    - Transactions 1
      - Query
      - Response



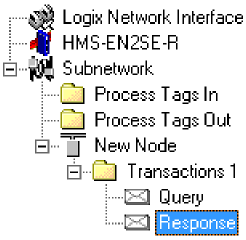
Alphabetic | Categorized

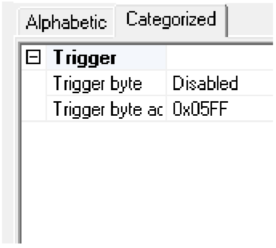
General	
Offline options for fieldbus	Clear
Offline options for sub-network	Clear
Update mode	Cyclically
Timing	
Minimum time between broadcasts (10ms)	100
Reconnect time (10ms)	1000
Retries	3
Timeout time (10ms)	100
Update time (10ms)	100
Trigger	
Trigger byte address	0x05FF

Parameter	Description
Minimum time between broadcasts (10 ms)	This parameter specifies how long the device shall wait after transmitting a broadcast transaction before processing the next entry in the scan list. The value should be set high enough to allow the slave devices time to finish the handling of the broadcast. The entered value is multiplied by 10. An entered value of 5 will result in 50 ms. Note: This setting is only relevant for the Broadcaster node.
Offline options for the network	This parameter specifies the action to take for this transaction if the higher level network goes offline. This affects the data that is sent to the subnetwork. • Clear - The data destined for the slave devices is cleared (set to zero) • Freeze - The data destined for the slave device is frozen • NoScanning -The updating of the subnetwork is stopped
Offline options for the subnetwork	This parameter specifies the action to take for this transaction if the subnetwork goes offline. This affects the data that is reported to the control system. • Clear - Data is cleared (0) on the higher level network if the subnetwork goes offline • Freeze - Data is frozen on the higher level network if the subnetwork goes offline
Reconnect time (10 ms)	This parameter specifies how long the device shall wait before attempting to reconnect a disconnected node. A node will be disconnected in case the maximum number of retries (below) has been reached. The entered value is multiplied by 10. An entered value of 5 will result in 50 ms. Note: This setting is not relevant for the Broadcaster node.
Retries	This parameter specifies how many times a timeout may occur in sequence before the node is disconnected.
Timeout time (10 ms)	This parameter specifies how long the device will wait for a response from a node. If this time is exceeded, the device will retransmit the Query until the maximum number of retries (see above) has been reached. The entered value is multiplied by 10. An entered value of 5 will result in 50 ms.
Trigger byte address	This parameter specifies the location of the trigger byte in internal memory (only relevant when “Update mode” is set to “Change of state on trigger”). Valid settings range from 0x200 to 0x3FF and 0x400 to 0xFFF

Parameter	Description
Update mode	This parameter is used to specify when the transaction shall be sent to the slave: • Cyclically The transaction is issued cyclically at the interval specified in the “Update time” parameter. • On data change The data area is polled for changes at the time interval defined by Update time. A transaction is issued when a change in data is detected. • Single shot The Query is issued once at start up. • Change of state on trigger The Query is issued when the trigger byte value has changed. This feature enables the control system to notify the linking device when to issue a particular Query. To use this feature correctly, the control system must first update the data area associated with the Query/transaction, then increase the trigger byte by one. The location of the trigger byte is specified by the “Trigger byte address” parameter. The trigger byte is checked at the interval specified in the “Update time” parameter.
Update time (10 ms)	This parameter specifies how often the transaction will be issued in steps of 10 ms (relevant only when “Update mode” is set to “Cyclically”, “On data change” or “Change of state on trigger”). The entered value is multiplied by 10. An entered value of 5 will result in 50 ms.

7.3.2 Parameters (Response)





Trigger

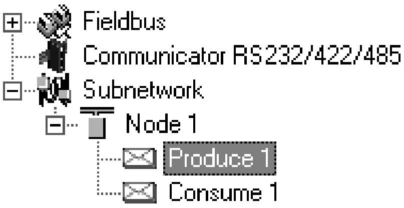
Trigger byte: Disabled

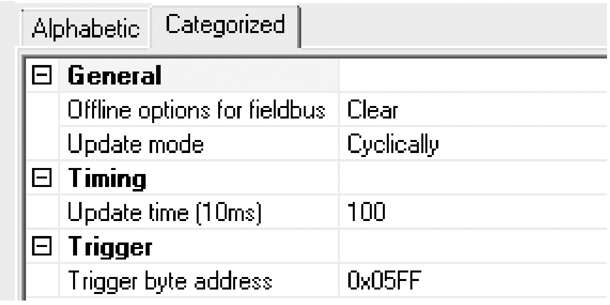
Trigger byte ac: 0x05FF

Parameter	Description
Trigger byte	This parameter is used to enable/disable the trigger functionality for the response. If enabled, the device will increase the trigger byte by one when the gateway receives new data from the subnetwork. This can be used to notify the control system of the updated data. The location of the trigger byte is specified by the “Trigger byte address” parameter below.
Trigger byte address	This parameter specifies the location of the trigger byte in the internal memory buffer. Valid settings range from 0x000 to 0x1FF and 0x400 to 0xFFF

7.4 Transaction Parameters (Generic Data Mode)

7.4.1 Produce Transactions



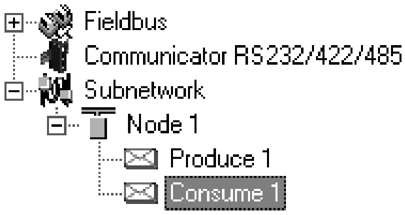


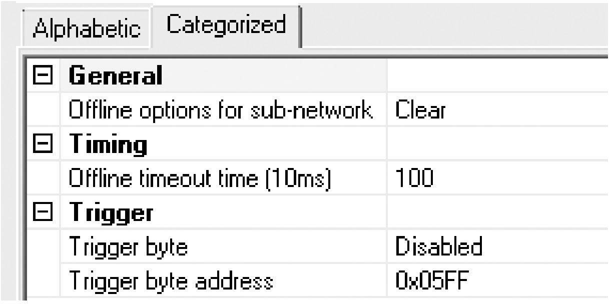
⊟ **General**	
Offline options for fieldbus	Clear
Update mode	Cyclically
⊟ **Timing**	
Update time (10ms)	100
⊟ **Trigger**	
Trigger byte address	0x05FF

Parameter	Description
Offline options for fieldbus	This parameter specifies the action to take for this transaction if the higher level network goes offline. This affects the data that is sent to the subnetwork. • Clear Data is cleared (0) on the subnetwork if the higher level network goes offline • Freeze Data is frozen on the subnetwork if the higher level network goes offline • NoScanning Stop subnet scanning for this transaction if the higher level network goes offline
Update mode	The update mode for the transaction: • Cyclically The transaction is sent cyclically at the interval specified in “Update Time”. • On data change The data area is polled for changes at the time interval defined by Update time. A transaction is issued when a change in data is detected. • Single shot The transaction is sent once at startup. • Change of state on trigger The transaction is sent when the trigger byte has changed. This feature enables the control system to notify the device when to issue a particular transaction. To use this feature correctly, the control system must first update the data area associated with the transaction, then increase the trigger byte by one. The location of the trigger byte is specified by the “Trigger byte address” parameter. The trigger byte is checked at the interval specified in the “Update time” parameter.
Update time (10 ms)	This parameter specifies how often the transaction will be issued in steps of 10ms (relevant only when “Update mode” is set to “Cyclically”, “On data change” or “Change of state on trigger”). The entered value is multiplied by 10. An entered value of 5 will result in 50 ms.

Parameter	Description
Trigger byte address	This parameter specifies location of the trigger byte in the internal memory buffer. If “Update mode” is set to “Change of state on trigger”, the memory location specified by this parameter is monitored by the device. Whenever the trigger byte is updated, the linking device will produce the transaction on the subnetwork. This way, the control system can instruct the linking device to produce a specific transaction on the subnetwork by updating the corresponding trigger byte. The trigger byte should be incremented by one for each activation. Please note that the trigger byte address must be unique to each transaction. It can not be shared by two or more transactions. Note: This parameter has no effect unless the “Update mode” parameter is set to “Change of state on trigger”. Valid settings range from 0x200 to 0x3FF and 0x400 to 0xFFF

7.4.2 Consume Transactions



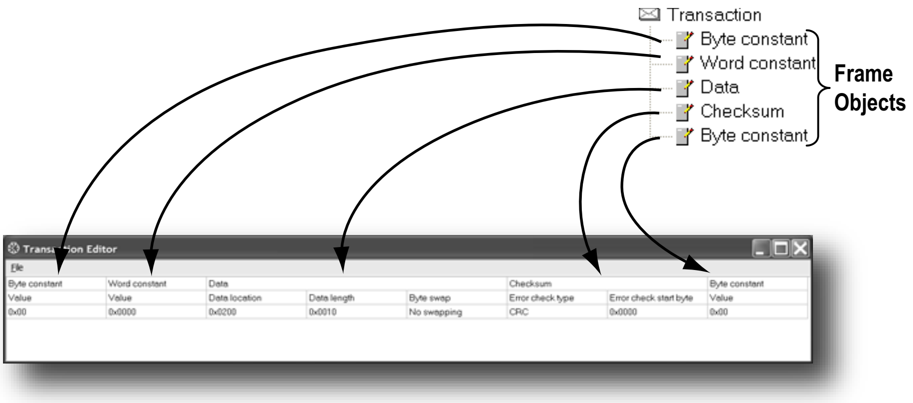


⊟ **General**	
Offline options for sub-network	Clear
⊟ **Timing**	
Offline timeout time (10ms)	100
⊟ **Trigger**	
Trigger byte	Disabled
Trigger byte address	0x05FF

Parameter	Description
Offline options for subnetwork	This parameter specifies the action to take for this transaction if the subnetwork goes offline. This affects the data that is sent to the higher level network. • Clear Data is cleared (0) on the higher level network if the subnetwork goes offline • Freeze Data is frozen on the higher level network if the subnetwork goes offline
Offline timeout time (10 ms)	This parameter specifies the maximum allowed time between two incoming messages in steps of 10ms. If this time is exceeded, the subnetwork is considered to be offline. A value of 0 disables this feature, i.e. the subnetwork can never go offline. The entered value is multiplied by 10. An entered value of 5 will result in 50 ms.
Trigger byte	• Enable Enables the trigger byte. The location of the trigger byte must be specified in “Trigger byte address”. The trigger byte value will be increased each time a valid transaction has been consumed by the device. The trigger byte will also be increased if the offline option is set to “Clear” and the offline timeout time value is reached. This feature enables the control system to be notified each time new data has been consumed on the subnetwork. • Disable Disables the trigger byte functionality.
Trigger byte address	This parameter specifies the location of the trigger byte in the internal memory buffer. Valid settings range from 0x000 to 0x1FF and 0x400 to 0xFFF. Please note that the trigger byte address must be unique to each transaction. It can not be shared by two or more transactions.

7.5 Transaction Editor

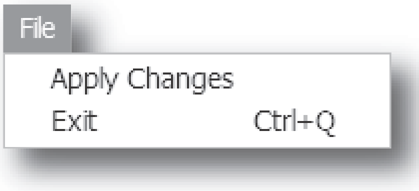
The Transaction Editor can be used to edit the individual frame objects of a transaction. The same settings are also available in the parameter section of the main window, however the Transaction Editor presents the frame objects in a more visual manner.



Byte constant	Word constant	Data			Checksum		Byte constant
Value	Value	Data location	Data length	Byte swap	Error check type	Error check start byte	Value
0x00	0x0000	0x0200	0x0010	No swapping	CRC	0x0000	0x00

To edit the value of a parameter, click on it and enter a new value using the keyboard. When editing transactions which are based on predefined commands, certain parts of the transaction may not be editable.

The File menu features the following entries:



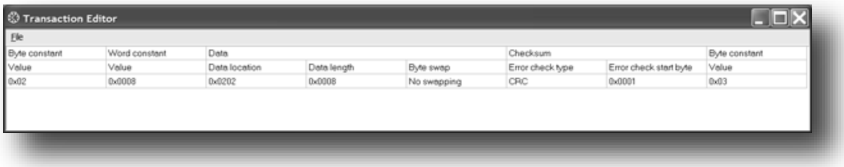
- **Apply Changes**

  This will save any changes and exit to the main window.

- **Exit**

  Exit without saving.

Example:



Byte constant	Word constant	Data			Checksum		Byte constant
Value	Value	Data location	Data length	Byte swap	Error check type	Error check start byte	Value
0x02	0x0008	0x0202	0x0008	No swapping	CRC	0x0001	0x03

The transaction created in this example are built up as follows:

The first byte holds the STX (0x02) followed by two bytes specifying the length of the data field (in this case 8). The next 8 bytes are data and since this is a “query”-transaction, the data is to be fetched from the Output Area which starts at address location 0x202. No swapping will be performed on the data. This is followed by a two-byte checksum. The checksum calculation starts with the second byte in the transaction.

The transaction ends with a byte constant, the ETX (0x03).

8. Frame Objects

8.1 General

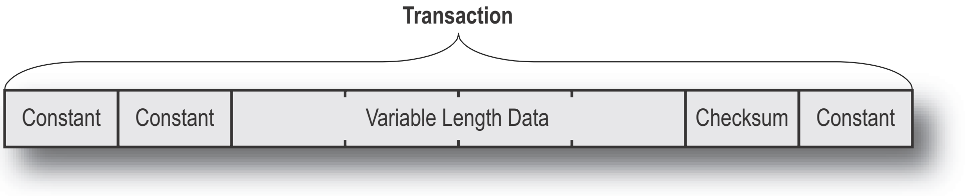
Each transaction consists of Frame Objects which makes up the serial telegram frame. Each Frame Object specifies how the linking device shall interpret or generate a particular part of the telegram.

There are 5 types of frame objects, which are described in detail later in this chapter:

- Constant Objects
- Limit Objects
- Data Objects
- Variable Data Objects
- Checksum Objects

Example:

The following Transaction consists of several frame objects; three constants, a data object, and a checksum object.



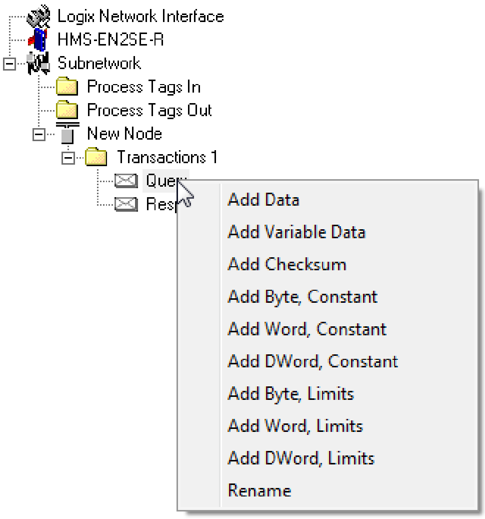
8.2 Adding and Editing Frame Objects

To add a frame object to a Transaction, right-click on the Transaction in the Navigation Section and select one of the entries in the menu that appears.

The entry called “Edit Transaction” will launch the Transaction Editor, which is used to edit transactions and frame objects in a more visual manner. For more information, see “Transaction Editor” on page 46.

To edit parameters associated with a particular frame object, select the frame object in the Navigation Section. The settings for that frame object will be displayed in the Parameter Section.

It is also possible to edit the frame objects in a transaction in a more visual manner using the Transaction Editor, see “Transaction Editor” on page 46.



8.3 Constant Objects (Byte, Word, Dword)

Constant Objects have a fixed value and come in three sizes:

- **Byte**
  8 bits
- **Word**
  16 bits
- **Dword**
  32 bits

Constants are handled differently depending on the direction of the transaction:

- **Produce/Query Transactions**
  The linking device will send the value as it is without processing it.
- **Consume/Response Transactions**
  The linking device will check if the received byte/word/dword matches the specified value. If not, the message will be discarded.

To set the value of the object, select it in the Navigation Section and enter the desired value in the Parameter section.

Parameter	Description
Value	Constant value

8.4 Limit Objects (Byte, Word, Dword)

Limit Objects have a fixed range and come in three sizes:

- **Byte**
  8 bits
- **Word**
  16 bits
- **Dword**
  32 bits

Limit Objects are handled differently depending on the direction of the transaction:

- **Produce/Query Transactions**
  This object shall not be used for such transactions (value will be undefined).
- **Consume/Response Transactions**
  The linking device will check if the received byte/word/dword fits inside the specified boundaries. If not, the message will be discarded.

There are 3 types of interval objects:

- **Byte**
  8 bit interval
- **Word**
  16 bit interval
- **Dword**
  32 bit interval

To set the range of the object, select it in the Navigation Section and enter the desired range in the Parameter section as follows:

Parameter	Description
Maximum Value	This is the largest allowed value for the range. Range:0x00 to 0xFFh(byte) 0x0000 to 0xFFFFh(word) 0x00000000 to 0xFFFFFFFFh(dword) **Note:** The value must be larger than the Minimum Value.
Minimum Value	This is the smallest allowed value for the range. Range:0x00 to 0xFEh(byte) 0x0000 to 0xFFFEh(word) 0x00000000 to 0xFFFFFFFEh(dword) **Note:** The value must be less than the Maximum Value.

8.5 Data Object

Data Objects are used to represent raw data as follows:

- **Produce/Query Transactions**

  The specified data block is forwarded from the higher level network to the subnetwork.

- **Consume/Response Transactions**

  The specified data block is forwarded from the subnetwork to the higher level network.

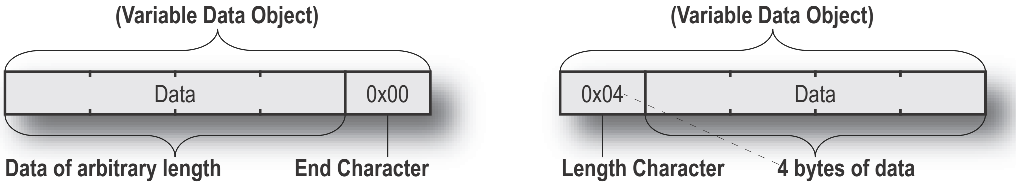
To specify the properties of the object, select it in the Navigation Section and enter the desired settings in the Parameter section as follows:

Parameter	Description
Byte Swapping	• No Swapping No swapping is performed on the data • Swap 2 bytes A, B, C, D becomes B, A, D, C • Swap 4 bytes A, B, C, D becomes D, C, B, A
Data Length	The length of the data block, in bytes. In case of a Response or Consume transaction, incoming messages where the data size differs from the value specified here will be discarded. Maximum data length allowed for one frame is 300 bytes.
Data Location	The location of the data block in the internal memory buffer.

8.6 Variable Data Object

Note: Only one Variable Data Object is permitted for each transaction.

This object is similar to the Data Object, except that it has no predefined length. Instead, an End or Length-character specifies the size of the data block as follows:



- **Produce/Query Transactions**

  The specified data block will be forwarded from the higher level network to the subnetwork. The control system must supply an End or Length character in order for the device to know the size of the data block.

  The End- or Length-character itself may either be forwarded to the subnetwork or discarded.

- **Consume/Response Transactions**

  The specified data block is forwarded from the subnetwork to the higher level network. The End- or Length-character will be generated by the device automatically (if applicable).

  The End- or Length-character itself may either be forwarded to the higher level network or discarded.

To specify the properties of the object, select it in the Navigation Section enter the desired settings in the Parameter section as follows:

Parameter	Description
Byte Swapping	• No Swapping No swapping will be performed on the data • Swap 2 bytes A, B, C, D becomes B, A, D, C • Swap 4 bytes A, B, C, D becomes D, C, B, A
Fill unused bytes	• Enabled[a] Fill unused data with the value specified in “Filler byte”. • Disabled Don’t fill
Filler byte	Filler byte value. Only used if “Fill unused bytes” has been enabled.
Data Location	The offset in the internal memory buffer where the data shall be read from / written to
Object Delimiter (Produce/Query)	• Length Character Length character visible in internal memory buffer but *not* sent out on the subnetwork • Length Character Visible Length character visible in internal memory buffer *and* sent out on the subnetwork • End Character End character visible in internal memory buffer but *not* sent out on the subnetwork • End Character Visible End character visible in the internal memory buffer *and* sent out on the subnetwork • No Character No end- or length-character generated in the internal memory buffer
Object Delimiter (Consume/Response)	• Length Character Length character visible in internal memory buffer but *not* received from the subnetwork • Length Character Visible Length character visible in internal memory buffer *and* received from the subnetwork • End Character End character visible in internal memory buffer but *not* received from the subnetwork • End Character Visible End character visible in the internal memory buffer *and* received from the subnetwork • No Character No end or length characters included in the received string or generated in the internal memory buffer
End Character Value	End Character value[b]
Maximum Data Length	The maximum allowed length (in bytes) of the variable data object. If the actual length of the data exceeds this value, the message will be discarded. The value must not exceed 256 bytes, which is the maximum data length allowed for one frame.

[a] Only relevant for Consume/Response transactions

[b] Only used if “Object Delimiter” is set to “End Character” or “End Character Visible”

8.7 Checksum Object

Most serial protocols features some way of verifying that the data has not been corrupted during transfer. The Checksum Object calculates and includes a checksum in a transaction.

Parameter	Description
Error Check Start byte	Specifies the byte offset in the transaction to start checksum calculations on.[a]
Error Check Type	This parameter specifies which type of algorithm to use: • CRC (2 bytes) CRC-16 with 0xA001 polynome (Modbus RTU standard) • LRC (1 byte) All bytes are added together as unsigned 8-bit values. The two's complement of the result will be used as a checksum. (Modbus ASCII standard with Error Check Start Byte = 0x01 and Representation = ASCII) • XOR (1 byte) All bytes are logically XOR:ed together. The resulting byte will be used as a checksum. • ADD (1 byte) All bytes are added together as unsigned 16-bit values. The lowest 8 bits in the result will be used as a checksum.
Error check type combined with	The binary value can be converted to its one's or two's complement. This conversion is carried out before ASCII formatting (see next parameter). • None The checksum binary value is transmitted without conversion. • One's complement The checksum value will be converted to its one's complement (inverse code). Example: 00001100 will be transmitted as 11110011 • Two's complement The checksum value will be converted to its two's complement (complement code). Example: 00001100 will be transmitted as 11110100
Representation	• Binary The checksum is transmitted in binary format. • ASCII All characters in the checksum are converted to ASCII values.

[a] In Generic Data Mode the Start character (if used) will not be included in the checksum calculation.

9. Commands

This information is only valid for the Master and Generic Data modes. For DF1 master mode, please refer to “Services” on page 60.

9.1 General

As mentioned previously, commands are actually predefined transactions that can be stored and reused. Just like regular transactions, commands consist of frame objects and are representations of the actual serial telegrams exchanged on the serial subnetwork.

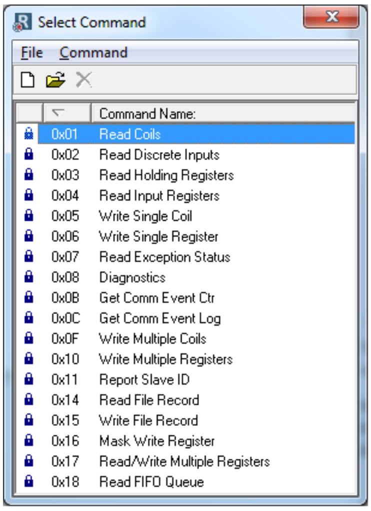
Adding a command to a node actually results in (a) transaction(s) being added according to the directions specified in the command. The frame objects in such a transaction may retrieve their values not only from parameters in the parameter section, but also from other sources such as the “SlaveAddress”-parameter (see “Node Parameters” on page 39). In such case, the parameters in the parameter section will be greyed out and cannot be edited directly.

In Master Mode, the configuration manager comes preloaded with commands for most common Modbus RTU functions. Additional commands can easily be added using the Command Editor (see “The Command Editor” on page 55). For DF1 Master Mode, see “Services” on page 60. In Generic Data Mode, no predefined commands exist, but custom ones may be implemented as desired.

9.2 Adding & Managing Commands

To add a command to a node, right-click on the node in the Navigation Section and select “Add Command”.

A list of commands will appear:



		Command Name:
🔒	0x01	Read Coils
🔒	0x02	Read Discrete Inputs
🔒	0x03	Read Holding Registers
🔒	0x04	Read Input Registers
🔒	0x05	Write Single Coil
🔒	0x06	Write Single Register
🔒	0x07	Read Exception Status
🔒	0x08	Diagnostics
🔒	0x0B	Get Comm Event Ctr
🔒	0x0C	Get Comm Event Log
🔒	0x0F	Write Multiple Coils
🔒	0x10	Write Multiple Registers
🔒	0x11	Report Slave ID
🔒	0x14	Read File Record
🔒	0x15	Write File Record
🔒	0x16	Mask Write Register
🔒	0x17	Read/Write Multiple Registers
🔒	0x18	Read FIFO Queue

Select the desired command in the list, and select “Add Command” in the “Command”-menu. The specified command will be added to the node.

Just like other transactions, the frame objects of added command may be edited in the Navigation/Parameter Section or using the Transaction Editor. Note however that certain frame objects may be locked for editing.

9.2.1 Drop-down Menu

File

This menu features the following entries:

- **Select**

  Add the currently selected Command to the node.

- **Exit**

  Exit without adding a command to the node.

Command

This menu is used to manage the commands in the list:

- **Add Command**

  Add a custom command to the list, and open the new command in the Command Editor.

  See also "The Command Editor" on page 55.

- **Edit Command**

  Edit the currently selected command using the Command Editor.

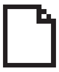
  See also "The Command Editor" on page 55.

- **Delete Command**

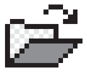
  Delete the currently selected command from the list. Note that some commands are fixed and cannot be deleted.

9.2.2 Toolbar Icons

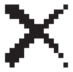
The toolbar features icons for the Add, Edit and Delete Command functions.



Add Command



Edit Command



Delete Command

9.3 The Command Editor

9.3.1 General

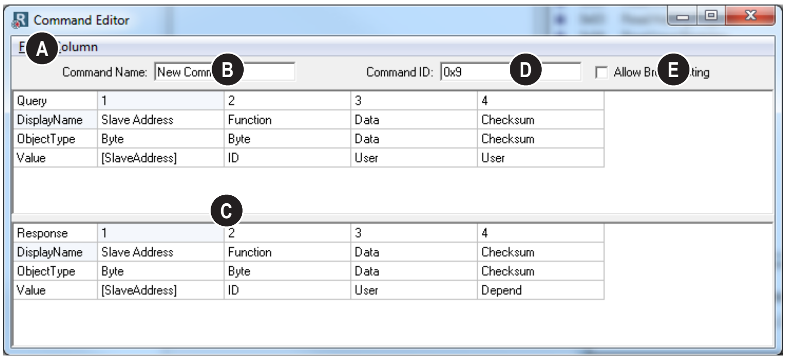
The Command Editor is used to define new commands and edit existing ones. This makes it possible to build a library of commands, which can be stored and reused at a later stage.

Note that the Command Editor is somewhat protocol-dependent in the sense that certain frame objects may not be deleted or altered.

The examples in this section use Master Mode. The procedures involved are similar in Generic Data Mode, but without the limitations imposed by the Modbus RTU protocol.

9.3.2 Basic Navigation

Open the Command Editor by selecting “Edit Command” or “Add Command” from the “Command”-menu.



Command Name: New Comn

Command ID: 0x9

□ Allow Br ting

Query	1	2	3	4
DisplayName	Slave Address	Function	Data	Checksum
ObjectType	Byte	Byte	Data	Checksum
Value	[SlaveAddress]	ID	User	User

Response	1	2	3	4
DisplayName	Slave Address	Function	Data	Checksum
ObjectType	Byte	Byte	Data	Checksum
Value	[SlaveAddress]	ID	User	Depend

A: Drop-down Menu

See “Drop-down Menu” on page 56.

B: Name of Command

Actual name of the command, in text form.

C: Command Transactions

This section holds the actual transactions associated with the command. This can either be a query-response pair, or a single transaction, depending on the protocol mode etc.

D: Command ID

This can be used as desired when building the command, e.g. to specify the function code.

E: Other Settings

Setting	Description
Allow Broadcasting	Specifies if it is allowed to broadcast the command (only relevant in Master Mode)
Produce	The command is producing data (Generic Data Mode only)
Consume	The command is consuming data (Generic Data Mode only)

9.3.3 Drop-down Menu

File

This menu features the following entries:

- **Apply Changes**
  Save changes and exit to the main window.

- **Exit**
  Exit without saving.

Column

The functions in this menu alters the structure of the command.

- **Append Column**
  Add another column to the command.

- **Insert Column**
  Insert a column at the selected position.

- **Delete Column**
  Delete the column at the selected position.

9.3.4 Editing a Command

As mentioned previously, the transaction section in the Command Editor represents the actual transactions associated with the command. Each column represents a frame object within the transaction.

Each column features four rows with the following parameters:

- **Query/Response/Produce/Consume**
  The upper right cell indicates the direction of the transaction.

- **DisplayName**
  Each column can be named so that the different parts of the command appears in a more user friendly manner when editing its settings in the Transaction Editor or in the Parameter Section of the Main Window.

- **ObjectType**
  This row specifies the type of frame object that shall be used for the column.

- **Value**
  This row specifies where the frame object shall retrieve its value/settings.

Value	Description
Depend	This setting is only relevant for Responses in Master Mode. The value will be retrieved from the corresponding part of the “Query”-transaction.
Id	Value will be retrieved from the “Command ID”-setting (see “Basic Navigation” on page 55).
User	Settings associated with the object can be edited by the user.
[SlaveAddress]	Value will be retrieved from the “SlaveAddress”-parameter (see “Node Parameters” on page 39).
(other settings)	Other settings are no longer supported.

9.3.5 Example: Specifying a Modbus-RTU Command in Master Mode

In the following example, a Modbus-RTU command is created in Master Mode. In Modbus-RTU, a transaction always feature the following parts:

- Slave Address (1 byte)
- Function Code (1 bytes)
- A data field
- CRC (CRC-16)

Furthermore, each command always consists of a query and a response.

- **Example Query**

Query	1	2	3	4
DisplayName	Slave Address	Function	Data	Checksum
Object Type	Byte Object	Byte Object	Data Object	Checksum Object
Value	[SlaveAddress]	ID	User	User
	The value of this byte constant will be set using the "SlaveAddress" parameter (see "Node Parameters" on page 39).	*The value of this byte constant will be set using the "Command ID"-field.*	*The size and location of the data associated with this object is determined by the user.*	*The checksum type etc can be selected by the user. By default, this is set to match the Modbus-RTU standard.*

- **Example Response**

Response	1	2	3	4
DisplayName	Slave Address	Function	Data	Checksum
Object Type	Byte Object	Byte Object	Data Object	Checksum Object
Value	[SlaveAddress]	ID	User	Depend
	This value is linked to the "SlaveAddress" parameter in the parameter window.	*The value of this byte constant will be set using the "Command ID"-field.*	*The size and location of the data associated with this object is determined by the user.*	*This object will retrieve its settings from the corresponding object in the Query.*

By default, the Modbus-RTU-specific frame objects are already in place, and a data object is inserted between the function code and the CRC. These objects cannot be moved or deleted, however it is possible to add additional objects between the function code and the CRC as desired.

Name the new command by entering its name in the "Command Name" field, and enter a suitable function code in the "Command ID"-field. If the command is allowed to be broadcast, check the "Allow Broadcasting" check box.

10. DF1 Protocol Mode

This mode makes the linking device act as a DF1 protocol master on the subnetwork.

10.1 Communicator Parameters

Interface

Currently, only serial communication is supported.

Control/Status Word

(See "Control and Status Registers" on page 72).

Value	Description
Enabled	Enable the Control and Status Registers. The "Data Valid"-bit in the Control Register must be set to start the subnetwork communication.
Enabled but no startup lock	This setting is similar to "Enabled", except that the control system is not required to set the "Data Valid"-bit to start the subnetwork communication.
Disabled	This setting completely disables the Control and Status Registers.

Module Reset

This parameter specifies how the device will behave in the event of a fatal error.

Value	Description
Enabled	The device will be restarted, and no error will be indicated to the user.
Disabled	The device will halt and indicate an error.

Protocol Mode

This parameter specifies which protocol mode to use for the subnetwork.

Value	Description
DF1-Master	This mode is intended for the DF1 protocol. The linking device can only be configured as a Master with half-duplex communication. Note: This is the only mode available if you intend to configure the device for DF1.

See also "Protocol Modes" on page 17.

Statistics

The Transmit- and Receive Counters indicate how many transactions that have successfully been exchanged on the subnetwork. This feature is primarily intended for debugging purposes.

- **Receive Counter Location**

  Specifies the location of the Receive Counter in the internal memory buffer.

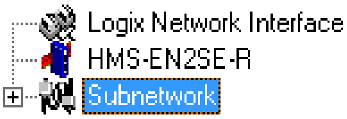
- **Statistics**

  Enables/disables the Receive and Transmit Counters.

- **Transmit Counter Location**

  Specifies the location of the Transmit Counter in the internal memory buffer.

10.2 Subnetwork Parameters



Communication

These parameters specify the actual communication settings used for the subnetwork.

Parameter	Description	Valid Settings
Baudrate (bits/s)	Selects the bit rate	2400 4800 9600 19200 38400 (Default)
Data bits	Selects the number of data bits	8
Parity	Selects the parity mode	None, Odd, Even
Physical standard	Selects the physical interface type	RS232, RS422, RS485
Stop bits	Number of stop bits	1

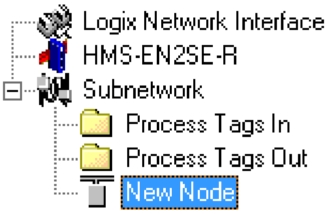
DF1 Settings

Parameter	Description	Default
Master Node Address	Node address of the master, valid values: 0–254	1
Poll time, active slaves (10 ms)	Determines how often the slave shall be polled in steps of 10 ms	100 ms[a]
Poll time, inactive slaves (10 ms)	Determines how often the slave shall be polled in steps of 10 ms	1000 ms[b]

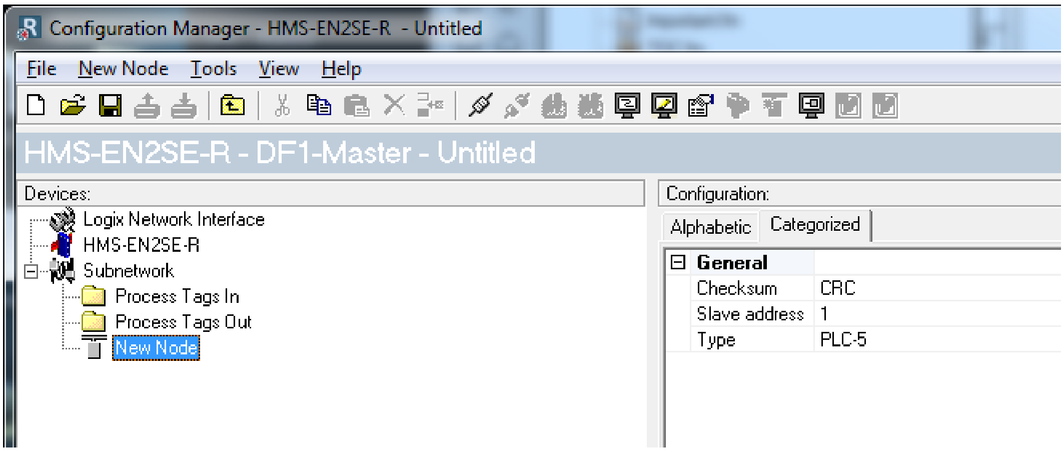
[a] The default value is given as 10 in the parameter window. Each change of 10 ms either increases or decreases this value by 1, i.e. 9 represents a poll time of 90 ms and 11 represents a poll time of 110 ms.

[b] The default value is given as 100 in the parameter window. Each change of 10 ms either increases or decreases this value by 1, i.e. 99 represents a poll time of 990 ms and 101 represents a poll time of 1010 ms.

10.3 Node Parameters



To gain access to the parameters described in this section, select a node in the navigation section. For more information about nodes, see “Nodes” on page 39.



Checksum CRC
Slave address 1
Type PLC-5

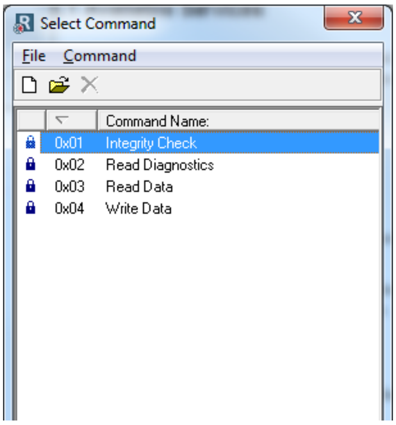
Parameter	Description	Valid Settings
Checksum	Selects the type of checksum on the network.	BCC CRC (default)
Slave Address	The value entered here sets the node address.	0-254
Type	The PLC type of the slave	PLC-5 SLC500 MicroLogix

10.4 Services

Services are commands that can be stored and reused. The user configures each slave with services that can be issued from the master. A total of 50 services are allowed.

The linking device supports a selection of DF1 commands. When the device is going to execute a service, it automatically chooses the appropriate DF1 command(s) that are used to perform the service on the selected DF1 node type.

10.4.1 Available Services



Select Command

	Command Name:
0x01	Integrity Check
0x02	Read Diagnostics
0x03	Read Data
0x04	Write Data

Right click on the node, and choose Add Command. A pop-up window will show the four different services that are available:

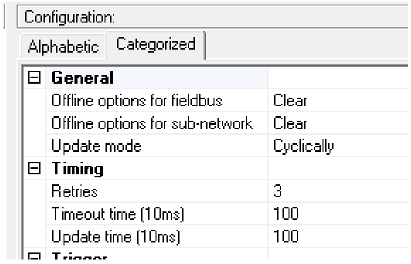
- Integrity check
- Read diagnostics
- Read data
- Write data

A maximum of 50 services in total (for all nodes) can be selected.

The predefined services can be configured to suit the application. Select a service to show the parameters.

General Configuration Parameters

These parameters are common to all services, but the settings are individual to each instance of a service.



Configuration:

General	
Offline options for fieldbus	Clear
Offline options for sub-network	Clear
Update mode	Cyclically
Timing	
Retries	3
Timeout time (10ms)	100
Update time (10ms)	100
Trigger	

General:

Parameter	Description	Valid settings
Offline options for fieldbus	The action to take for this service if the fieldbus goes offline. This option affects the data that is sent out to the subnetwork.	Clear Freeze Noscanning
Offline options for subnetwork	The action to take for this service if the subnetwork goes offline. This option affects the data that is reported to the fieldbus master.	Clear Freeze
Update mode	The update mode for this service	Cyclically On data change Single shot Change of state on trigger

Timing:

Parameter	Description	Default
Retries	The number of times to resend this service before the node is disconnected	3
Timeout time (10 ms)	The time to wait before resending this service (in steps of 10 ms)[a]	1000 ms
Update time (10 ms)	The minimum time between two services of this kind (in steps of 10 ms)[a]	1000 ms

[a] The default value is given as 100 in the parameter window. Each change of 10 ms either increases or decreases this value by 1, i.e. 99 represents a poll time of 990 ms and 101 represents a poll time of 1010 ms.

Trigger:

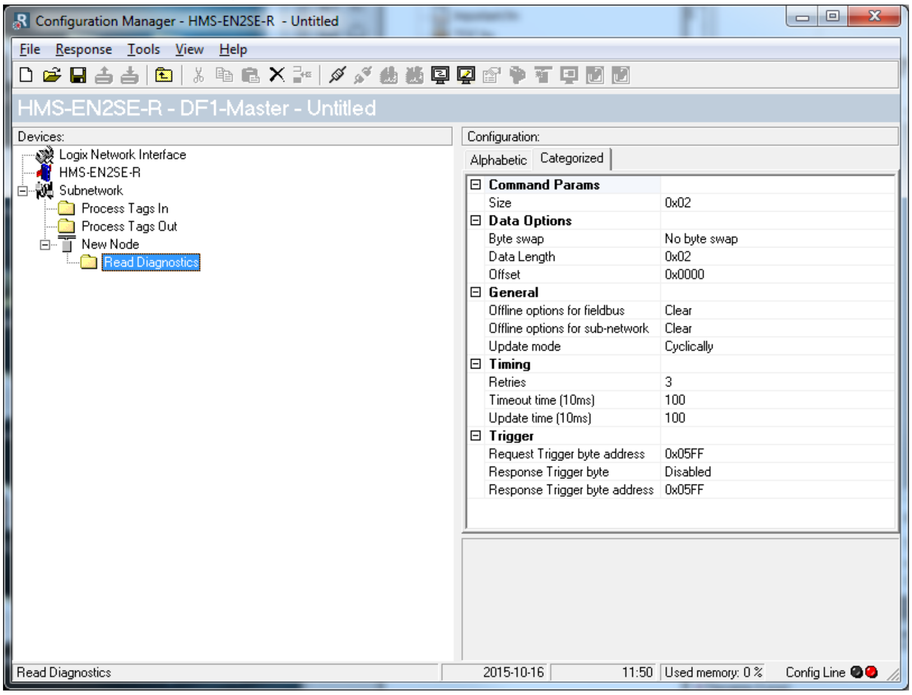
Parameter	Description	Default
Request Trigger byte address	The memory location of the trigger byte this service uses for updates on trigger byte changes	0x05FF
Response Trigger byte	Enables/disables the trigger byte	Disabled
Response Trigger byte address	The memory location of the trigger byte this service uses for updates on trigger byte changes Valid settings range from 0x200 to 0x3FF and 0x400 to 0xFFF	0x05FF

10.5 Integrity Check

This service checks that a node is up and running correctly. A telegram is sent to the node. The node mirrors and returns the telegram. No configuration is needed, apart from the general parameters, common to all services.

10.6 Read Diagnostics

This service reads diagnostic information from the device.



Command Params	
Size	0x02
Data Options	
Byte swap	No byte swap
Data Length	0x02
Offset	0x0000
General	
Offline options for fieldbus	Clear
Offline options for sub-network	Clear
Update mode	Cyclically
Timing	
Retries	3
Timeout time (10ms)	100
Update time (10ms)	100
Trigger	
Request Trigger byte address	0x05FF
Response Trigger byte	Disabled
Response Trigger byte address	0x05FF

Command parameters

The command parameter Size decides the amount of data that can be read. The size is given in bytes which means that it always has to be an even number as only whole elements can be read from the slave. One bit/integer element is 2 bytes and one float element is 4 bytes. The range of the size differs, depending on node type:

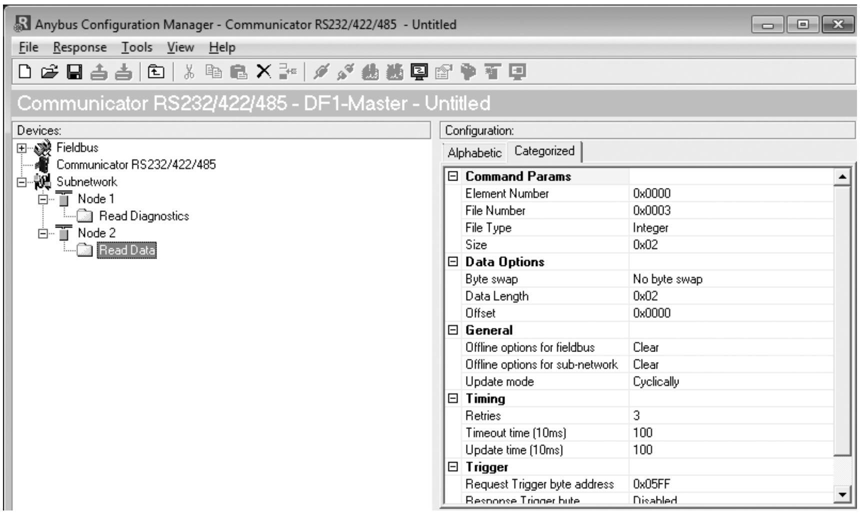
	PLC-5	SLC500	MicroLogix
Size range (in bytes)	1–26	1–28	1–26

Data options

Parameter	Description	Valid settings
Byte swap	Determines if the data shall be swapped	No byte swap Swap words Swap double words
Data length	The number of bytes, read from the DF1 network, to write to the area determined by the Offset parameter	≤ Size
Offset	The offset in the internal memory buffer in the device, where the data shall be read.	

10.7 Read Data

This service is used to read data from the nodes in the subnetwork.


Anybus Configuration Manager - Communicator RS232/422/485 - Untitled
File Response Tools View Help
Communicator RS232/422/485 - DF1-Master - Untitled
Devices:
Fieldbus
Communicator RS232/422/485
Subnetwork
Node 1
Read Diagnostics
Node 2
Read Data
Configuration:
Alphabetic
Categorized

Command Params	
Element Number	0x0000
File Number	0x0003
File Type	Integer
Size	0x02
Data Options	
Byte swap	No byte swap
Data Length	0x02
Offset	0x0000
General	
Offline options for fieldbus	Clear
Offline options for sub-network	Clear
Update mode	Cyclically
Timing	
Retries	3
Timeout time (10ms)	100
Update time (10ms)	100
Trigger	
Request Trigger byte address	0x05FF
Response Trigger byte	Disabled

Command Parameters

Parameter	Description	Valid settings
Element Number	The element number of the data file to be accessed within the slave.	PLC-5: 0–999 SLC500: 0–255 MicroLogix: 0–255
File number	The file number of the data file to be accessed.	PLC-5: 3, 7, 8, 10–999 SLC500: 3, 7, 8, 10–255 MicroLogix: 3, 7, 8, 10–255
File type	The file type of the data to be accessed.	Integer Bit Float
Size	The number of bytes to read from the slave. One bit/integer element is 2 bytes and one float element is 4 bytes. The parameter must have an even value as only whole elements can be read from the slave.	PLC-5: 2–240 SLC500: 2–236 MicroLogix: 2–242

Data Options

Parameter	Description	Valid settings
Byte swap	Determines if the data shall be swapped.	No byte swap Swap words Swap double words
Data length	The number of bytes, read from the DF1 network, to write to the area determined by the Offset parameter	≤ Size
Offset	The offset in the internal memory buffer in the module, where the data shall be read. See “Memory Map” on page 15. Note: If the control and status registers are enabled (default), first available data location will be: Input area 0x002, Output area 0x202.	-

10.8 Write Data

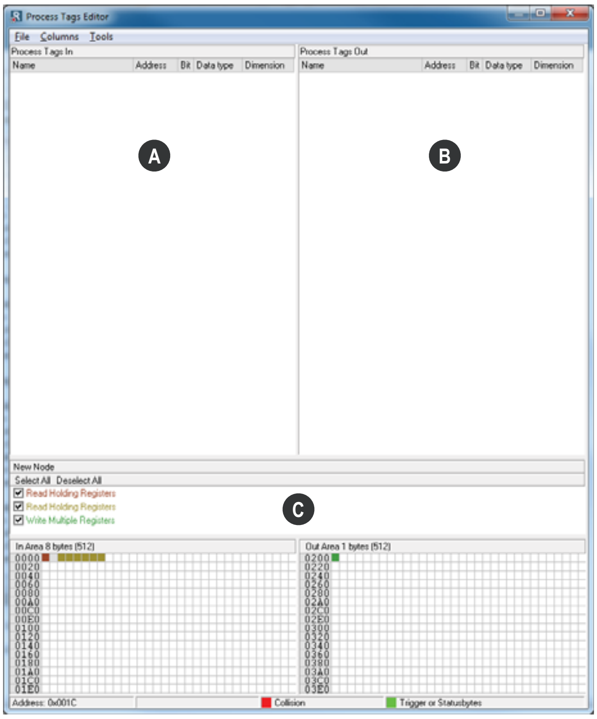
This service is used to write data to the nodes in the subnetwork. The parameters to be configured are the same as for the service Read Data. The only difference is that data is read from the internal memory buffer in the linking device and written to the subnetwork bus, instead of being written to the internal memory buffer.

11. Process Tags Editor

11.1 General

The process tags editor makes it possible to generate, validate and alter process tags based on the configuration. These can later be saved and used in Studio 5000. It is also possible to make the process tags first, and then use them as a guideline when creating the configuration.

11.2 Navigating the Process Tags Editor


Process Tags Editor
File Columns Tools
Process Tags In
Name Address Bit Data type Dimension
Process Tags Out
Name Address Bit Data type Dimension
A
B
New Node
Select All Deselect All
☑ Read Holding Registers
☑ Read Holding Registers
☑ Write Multiple Registers
C
In Area 8 bytes (512)
0000 0020 0040 0060 0080 00A0 00C0 00E0 0100 0120 0140 0160 0180 01A0 01C0 01E0
Out Area 1 bytes (512)
0200 0220 0240 0260 0280 02A0 02C0 02E0 0300 0320 0340 0360 0380 03A0 03C0 03E0
Address: 0x001C
🟥 Collision
🟩 Trigger or Statusbytes

A: Process Tags In

All process tags for input data are found here.

B: Process Tags Out

All process tags for output data are found here.

C: Process Data Overview

This area presents a graphical representation of the input and output process data areas. All configured data will be visible here.

11.2.1 Drop-down Menu

File

There is only one entry in this menu:

- **Exit**
  This will close the process tags editor.

Columns

This menu specifies the number of columns in the process data overview section.

- **Free**
  The number of columns depends on the width of the window.
- **Multiple**
  The number of columns will be fixed to a multiple of 8.

Tools

- **Generate Process Tags**
  This option will automatically generate process tags based on the configuration.
- **Validate Process Tags**
  This option will validate all process tags available in the configuration. A process tag marked with red color means any or many of the following:
  - The designated memory address of the process tag is outside the process data area boundaries.
  - The memory address of the process tag is not dividable by four, for data types DINT or REAL or a tag whose dimension is above zero.
  - The memory address of the process tag is not dividable by two for data type INT.
  - The memory addresses of two process tags overlap in the process data area.
  - The designated memory address of the process tag is greater than the corresponding designated memory address in the configuration.

12. Subnetwork Monitor

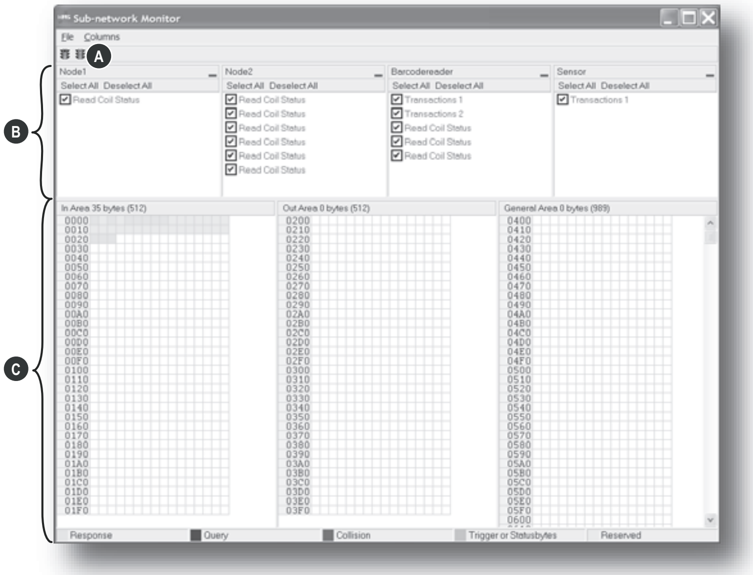
12.1 General

The subnetwork monitor is intended to simplify configuration and troubleshooting of the subnetwork. Its main function is to display the data allocated for subnetwork communication and detect if any area has been allocated twice (i.e if a collision has occurred).

All configured nodes, and their transactions, are listed in the middle of the screen (B). Selecting and deselecting single transactions makes it possible to view any combination of allocated data.

Note: The subnetwork monitor has a negative influence on the overall performance of the linking device. Therefore the monitor functionality should be used with care.

12.2 Operation


Sub-network Monitor
File
Columns
A
B
C
Node1
Node2
Barcodereader
Sensor
Select All Deselect All
In Area 35 bytes (512)
Out Area 0 bytes (512)
General Area 0 bytes (989)
0000 0010 0020 0030 0040 0050 0060 0070 0080 0090 00A0 00B0 00C0 00D0 00E0 00F0 0100 0110 0120 0130 0140 0150 0160 0170 0180 0190 01A0 01B0 01C0 01D0 01E0 01F0
0200 0210 0220 0230 0240 0250 0260 0270 0280 0290 02A0 02B0 02C0 02D0 02E0 02F0 0300 0310 0320 0330 0340 0350 0360 0370 0380 0390 03A0 03B0 03C0 03D0 03E0 03F0
0400 0410 0420 0430 0440 0450 0460 0470 0480 0490 04A0 04B0 04C0 04D0 04E0 04F0 0500 0510 0520 0530 0540 0550 0560 0570 0580 0590 05A0 05B0 05C0 05D0 05E0 05F0 0600
Response
Query
Collision
Trigger or Statusbytes
Reserved

Node1
☑ Read Coil Status

Node2
☑ Read Coil Status
☑ Read Coil Status
☑ Read Coil Status
☑ Read Coil Status
☑ Read Coil Status
☑ Read Coil Status

Barcodereader
☑ Transactions 1
☑ Transactions 2
☑ Read Coil Status
☑ Read Coil Status
☑ Read Coil Status

Sensor
☑ Transactions 1

A: Start Network & Stop Network Icons

These icons controls the subnetwork activity. To stop all activity, click on the red light. To start the subnetwork again, click on the green light.

B: Nodes / Transactions

To view data blocks associated with a transaction, select the transaction in the list. The corresponding data will then appear in the Monitor Section (C).

C: Monitor Section

This section visualizes how data is allocated in the Input, Output and General Data areas.

Color	Meaning
White	Not allocated
Yellow	Data allocated by a Response or Consume transaction
Blue	Data allocated by a Query or Produce transaction
Red	Collision; area has been allocated more than once
Grey	Reserved (illustrates memory consumption, area can be allocated if necessary)
Green	Data allocated by Trigger byte, Transmit/Receive Counter, or Control/Status Registers

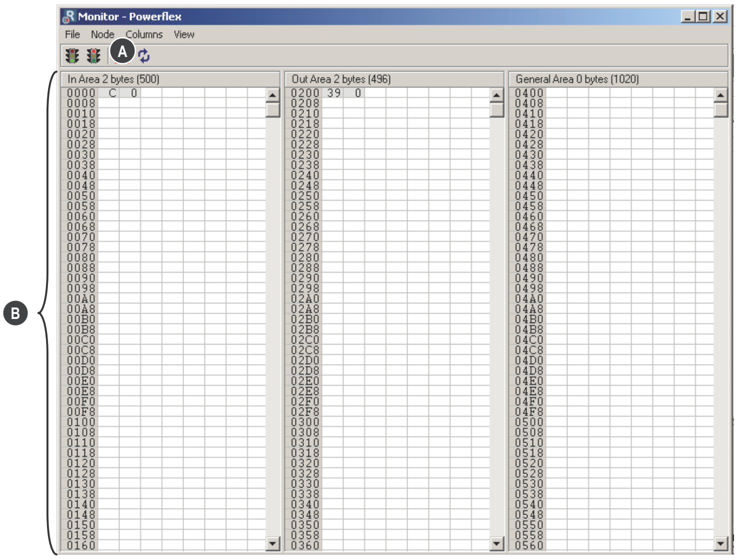
13. Node Monitor

13.1 General

The node monitor can provide valuable information when setting up the communication with the sub-network, by allowing individual commands to be issued manually, and monitoring the response (if applicable). It also provides an overview of the memory used by a particular node.

Note: The node monitor has a negative influence on the overall performance of the linking device, i.e. it should be used only when necessary.

13.2 Navigating the Node Monitor



```
In Area 2 bytes (500)   Out Area 2 bytes (496)   General Area 0 bytes (1020)
0000  C  0              0200  39  0              0400
0008                    0208                     0408
0010                    0210                     0410
0018                    0218                     0418
0020                    0220                     0420
0028                    0228                     0428
0030                    0230                     0430
0038                    0238                     0438
0040                    0240                     0440
0048                    0248                     0448
0050                    0250                     0450
0058                    0258                     0458
0060                    0260                     0460
0068                    0268                     0468
0070                    0270                     0470
0078                    0278                     0478
0080                    0280                     0480
0088                    0288                     0488
0090                    0290                     0490
0098                    0298                     0498
00A0                    02A0                     04A0
00A8                    02A8                     04A8
00B0                    02B0                     04B0
00B8                    02B8                     04B8
00C0                    02C0                     04C0
00C8                    02C8                     04C8
00D0                    02D0                     04D0
00D8                    02D8                     04D8
00E0                    02E0                     04E0
00E8                    02E8                     04E8
00F0                    02F0                     04F0
00F8                    02F8                     04F8
0100                    0300                     0500
0108                    0308                     0508
0110                    0310                     0510
0118                    0318                     0518
0120                    0320                     0520
0128                    0328                     0528
0130                    0330                     0530
0138                    0338                     0538
0140                    0340                     0540
0148                    0348                     0548
0150                    0350                     0550
0158                    0358                     0558
0160                    0360                     0560
```

A: Drop-down Menu & Toolbar Icons

See “Drop-down Menu” on page 68 and “Toolbar Icons” on page 69.

B: Monitor Section

This section displays the data associated with the node. Areas in dark grey are reserved for the Status & Control Registers, and areas displayed in light grey represent the data that is used by the node.

The data displayed in this section will be refreshed based on the refresh-icons in the toolbar. For more information, see “Toolbar Icons” on page 69.

13.2.1 Drop-down Menu

File

There is only one entry in this menu:

- **Exit**
  This will close the Node Monitor. Note however that if the node has been disabled using "Stop Node" (see below), it will not resume data exchange until enabled again using "Start node".

Node

This menu controls the data exchange for the node. This feature can help isolate problems associated with a particular node.

- **Start Node**
  Enable the transactions associated with the node.

- **Stop Node**
  Disable the transactions associated with the node.

Columns

This menu specifies the number of columns in the Monitor Section.

- **Free**
  The number of columns depends on the width of the window.

- **8 Multiple**
  The number of columns will be fixed to 8.

View

This menu specifies the data representation in the Monitor Section.

- **Hex**
  Display the data in hexadecimal format.

- **Decimal**
  Display the data in decimal format.

13.2.2 Toolbar Icons

The toolbar features icons for the most commonly used functions.

- **Start Node & Stop Node**

  These icons corresponds to the functions in the “Node” menu.

  See also “Node” on page 68.

  Start Stop

- **Resume Refresh & Stop Refresh**

  The data displayed in the Monitor Section will normally be refreshed automatically (cyclically).

  Click on “Stop” to stop automatic data refresh. Data will now only be refreshed if you click “Refresh” (see below).

  Press “Resume” to resume automatic refreshing of data.

  Stop Resume

- **Refresh**

  Refreshes the data displayed in the Monitor Section.

  Refresh

14. Data Logger

14.1 General

This feature allows the subnetwork traffic to be logged into a buffer for examination. This may provide valuable information when debugging the lowest levels of the subnetwork communication.

Note that the logger function is part of the device itself and is separate from the configuration manager. This means that logging can be performed even if the linking device is physically disconnected from the PC running configuration manager.

14.2 Operation

Start & Stop Logging

- **Start logging**
  Select “Start Logging” in the “Tools”-menu. The configuration manager will then prompt for the desired mode of operation, see below.

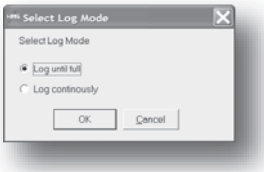
- **Stop logging**
  Select “Stop Logging” in the “Tools”-menu. This will open the log window, see below.

Modes of Operation

Select the desired mode of operation and click “OK” to start logging data.

- **Log until full**
  Data will be logged until the log buffer is full.

- **Log continuously**
  Data will be logged continuously until logging is stopped by clicking “Stop Logging”. The log buffer will contain the most recent data.



Select Log Mode

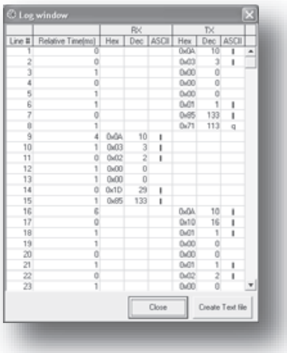
● Log until full
○ Log continously

Log Window

The logged data is displayed in hexadecimal, decimal and ASCII format for both directions. The time between the log entries is displayed in a separate column.

The data may optionally be saved in ASCII text format by clicking “Create Text file”.

Click “Close” to exit.



		RX			TX		
Line #	Relative Time(ms)	Hex	Dec	ASCII	Hex	Dec	ASCII
1	0				0x0A	10	▮
2	0				0x03	3	▮
3	1				0x00	0	
4	0				0x00	0	
5	1				0x00	0	
6	1				0x01	1	▮
7	0				0x85	133	▮
8	1				0x71	113	q
9	4	0x0A	10	▮			
10	1	0x03	3	▮			
11	0	0x02	2	▮			
12	1	0x00	0				
13	1	0x00	0				
14	0	0x1D	29	▮			
15	1	0x85	133	▮			
16	6				0x0A	10	▮
17	0				0x10	16	▮
18	1				0x01	1	▮
19	1				0x00	0	
20	0				0x00	0	
21	1				0x01	1	▮
22	0				0x02	2	▮
23	1				0x00	0	

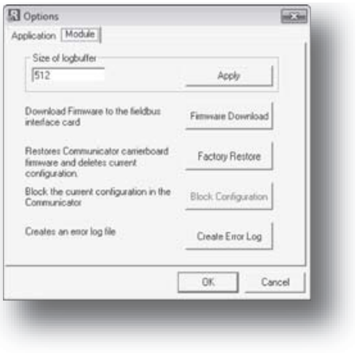
14.3 Configuration

By default, the log buffer can hold 512 bytes of data in each direction. To specify a different size for the buffer, select “Options” in the “Tools”-menu.

A window with various settings will appear. Select the “Module” tab, and enter the desired number of buffer entries under “Size of logbuffer” (valid settings range from 1–512).

Click “Apply” to validate the new settings.

Click “OK” to exit.



Size of logbuffer: 512

Download Firmware to the fieldbus interface card

Restores Communicator carrierboard firmware and deletes current configuration.

Block the current configuration in the Communicator

Creates an error log file

15. Control and Status Registers

15.1 General

The Control and Status Registers are disabled by default, but can be enabled using the configuration manager (see "Control/Status Word" on page 37). These registers form an interface for exchanging status information between the subnetwork and the network control system.

The main purpose of these registers is to...

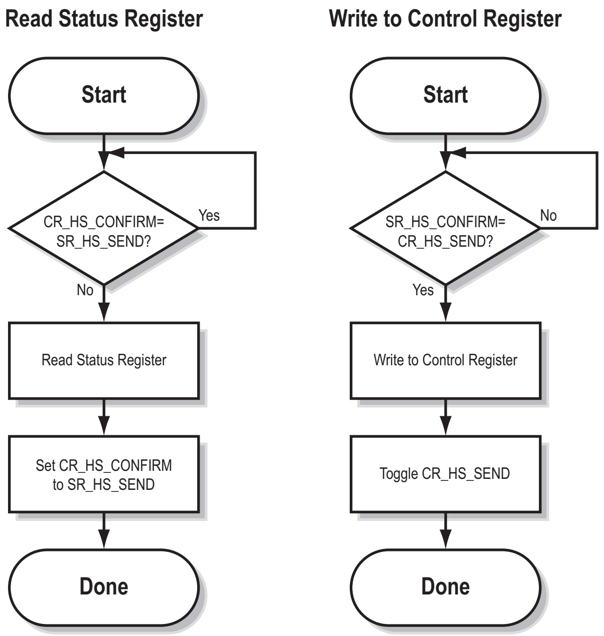
- Report subnetwork related problems to the network control system
- Ensure that only valid data is exchanged in both directions
- Enable the network control system to start/stop data exchange with selected nodes on the subnetwork

If enabled, these registers occupy the first two bytes in the input and output data areas (0x000–0x001 and 0x200–0x201 respectively), which means they can be accessed from the network just like any other data in these areas.

Note: Internally, these registers are stored in Motorola-format (i.e. MSB first). If the higher level network uses a different byte order, the upper and lower bytes will appear swapped.

15.1.1 Handshaking Procedure

A special handshaking procedure, which is illustrated in the two flowcharts below, must be followed when accessing these registers to ensure that both parts receive proper information.


Read Status Register
Write to Control Register
Start
Start
CR_HS_CONFIRM= SR_HS_SEND?
Yes
No
SR_HS_CONFIRM= CR_HS_SEND?
No
Yes
Read Status Register
Write to Control Register
Set CR_HS_CONFIRM to SR_HS_SEND
Toggle CR_HS_SEND
Done
Done

15.1.2 Data Consistency

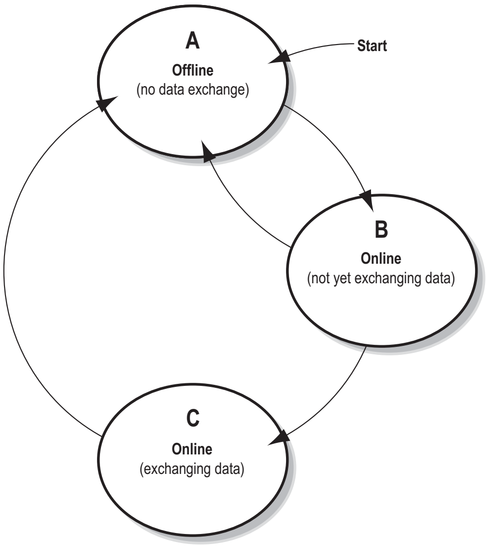
The “Data Valid”-bits in the Control and Status Registers are used to ensure data consistency during start-up and network offline/online transitions.

If the “Control/Status Word”-parameter in the configuration manager is set to “Enabled”, the device will wait for the network control system to set the “Data Valid”-bit in the Control Register before it starts exchanging data on the subnetwork.

If the same parameter is set to “Disabled” or “Enabled but no startup lock”, communication will start as soon as the network goes online.

State Machine

The network participation can be described using a state machine as described below.



A: Offline (No data exchange)

1. Clear the “Data Valid”-bit in the Control Register.
2. Write initial data to the Output Area according to the subnetwork configuration.
3. Wait until the network control system and the device are online on the network, and shift to state B.

B: Online (Not yet exchanging data)

4. Wait until the “Data Valid”-bit in the Status Register is cleared by the device.
5. Set the “Data Valid”-bit in the Control Register.
6. When the “Data Valid”-bit in the Status Register is set by the device, shift to state C.
7. If the device goes offline on the network, shift to state A.

C: Online (Exchanging data)

Exchanging valid data in both directions.
If the device goes offline on the network, shift to state A.

Note: The linking device cannot spontaneously clear the “Data Valid”-bit in the Status Register.

Latency

The “Data Valid”-bit in the Status Register may in some cases be delayed. This latency can be caused by a missing node or a bad connection to a node with a long timeout value assigned to it.

Therefore, the network control system should not wait for this bit to be set before communicating with the subnetwork devices; it should be considered as an aid for the network control system to know when all data has been updated.

15.2 Status Register Contents (Device to Control System)

15.2.1 General Information

The Status Register is (if enabled) located at 0x000–0x001 and constitutes a bit-field as follows:

bit(s)	Name	Description
15	Send (SR_HS_SEND)	These bits control the handshaking towards the network control system. See also... - “Handshaking Procedure” on page 72 - “Control Register Contents (Control System to Device)” on page 76
14	Confirm (SR_HS_CONFIRM)	
13	Data Valid (Master Mode and DF1 Master Mode Only)	This bit is set when all transactions have been executed successfully at least once. Once set, it will not change. 1:Data Valid 0:Data not Valid Note: This bit is not used in Generic Data Mode.
12... 8	Status Code	This field holds the last status report from the device. See also... - “Status Codes in Master Mode and DF1 Master Mode” on page 74 - “Status Code in Generic Data Mode” on page 75
7... 0	Data	

Note: Internally, this is treated as a Motorola-format word (i.e. MSB first). If the higher level network uses a different byte order, the upper and lower bytes will appear swapped.

15.2.2 Status Codes in Master Mode and DF1 Master Mode

(This table is valid only in Master Mode and DF1 Master Mode).

Code	Condition	Type	Data	Description
0x00	Retransmission Counter Updated	Warning	Counter	The number of retransmissions on the subnetwork has increased. If this problem persists, this may eventually trigger a Single- or Multiple Node(s) Missing condition.
0x01	Single Node Missing	Error	Slave address	A single node is missing.
0x02	Multiple Nodes Missing	Error	Number of nodes	Multiple nodes are missing.
0x03	Buffer Overrun	Warning	Slave address	A node returned more data than expected.
0x04	Other Error	Error	Slave address	Undefined error
0x1F	No Error	Warning	-	No errors

Note: Conditions of type “Error” will eventually be followed by a “No Error” condition when the cause has been resolved. Conditions of type “Warning” are however considered informational and may not necessarily be followed by a “No Error” condition later on.

15.2.3 Status Code in Generic Data Mode

(This table is valid only in Generic Data Mode).

Code	Condition	Type	Data	Description
0x00	Invalid Transaction Counter Updated	Error	Counter	The number of invalid transactions (i.e. received transactions which does not match any of the consume-transactions defined in the subnetwork configuration) has increased.
0x01	Frame Error	Warning	-	End character is enabled, but a message delimiter timeout occurs prior to receiving it.
0x02	Offline Timeout Counter Updated	Error	Counter	The of number of timed out consume-transactions has increased. See also... - "Consume Transactions" on page 45 (Offline timeout time)
0x03	Buffer Overrun	Warning	-	A node returned more data than expected - or - the device was unable to finish processing a message prior to receiving a new one.
0x04	Other Error	Error	-	Undefined error
0x1F	No Error	Warning	-	No errors

Note: Conditions of type "Error" will eventually be followed by a "No Error" condition when the cause no longer is detected. Conditions of type "Warning" are however considered informational and may not necessarily be followed by a "No Error" condition later on.

15.3 Control Register Contents (Control System to Device)

15.3.1 General Information

The Control Register is (if enabled) located at 0x200–0x201 and constitutes a bit-field as follows:

bit(s)	Name	Description
15	Confirm (CR_HS_CONFIRM)	These bits control the handshaking towards the device. See also... - “Handshaking Procedure” on page 72 - “Status Register Contents (Device to Control System)” on page 74
14	Send (CR_HS_SEND)	
13	Data Valid	This bit controls data consistency (see “Data Consistency” on page 73). 1:Output Area valid; exchange data on the subnetwork 0:Output Area not valid; do not exchange data on the subnetwork **Note:** This bit is only relevant if the Control/Status Registers are set as “Enabled”
12	Execute Command	If set, the specified command will be executed by the device (see below).
11... 8	Control Code	This field holds commands which can be executed by the device (see below). See also... - “Control Codes in Master Mode and DF1 Master Mode” on page 76 - “Control Codes in Generic Data Mode” on page 76
7... 0	Data	

Note: Internally, this is treated as a Motorola-format word (i.e. MSB first). If the higher level network uses a different byte order, the upper and lower bytes will appear to be swapped.

15.3.2 Control Codes in Master Mode and DF1 Master Mode

(This table is valid only in Master Mode and DF1 Master Mode).

Code	Instruction	Data	Description
0x00	Disable Node	Actual node address	Disables the specified node.
0x01	Enable Node	Actual node address	Enables a previously disabled node.
0x02	Enable Nodes	Actual number of nodes to enable	Enables the specified number of nodes, starting from the first node in the configuration. Remaining nodes will be disabled.

15.3.3 Control Codes in Generic Data Mode

(No Control Codes are currently supported in this mode).

A. File System

A.1 General

General

The EtherNet/IP to Serial linking device features a built-in file system, which is used to store information such as web files, network communication settings, e-mail messages etc.

Storage Areas

The file system consists of the different storage areas:

- **Nonvolatile area (approx. 1.4 Mb)**
  This section is intended for static files such as web files, configurations files etc.
- **Volatile area (approx. 1 Mb)**
  This area is intended for temporary storage; data placed here will be lost in case of power loss or reset.

Conventions

- '\' (backslash) is used as a path separator
- A 'path' originates from the system root and as such must begin with a '\'
- A 'path' must not end with a '\'
- Names may contain spaces (' ') but must not begin or end with one.
- Names may not contain the following characters: '\ / : * ? " < > |'
- Names cannot be longer than 48 characters (plus null termination)
- A path cannot be longer than 256 characters (filename included)
- The maximum number of simultaneously open files is 40
- The maximum number of simultaneously open directories is 40

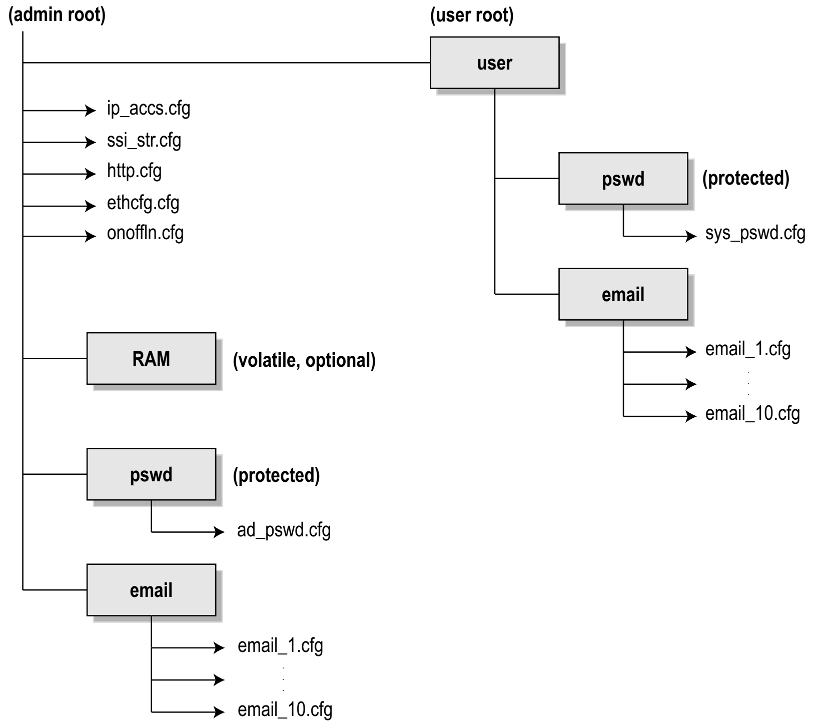
Important Note:

The nonvolatile storage is located in FLASH memory. Each FLASH segment can be erased approximately 100 000 times.

The following operations will erase one or more FLASH segments:

- Deleting, moving or renaming a file or directory
- Writing or appending data to an existing file
- Formatting the file system

A.2 File System Overview



A.3 System Files

The file system contains a set of files used for system configuration. These files, known as “system files”, are regular ASCII files that can be altered using a standard text editor (such as the Notepad in Microsoft Windows™). Note that some of these files may also be altered by the device itself, e.g. when using SSI (see “Server Side Include (SSI)” on page 44).

The format of the system files are based on the concept of ‘keys’, where each ‘key’ can be assigned a value, see example below.

Example:

```
[Key1]
value of key1

[Key2]
value of key2
```

The exact format of each system file is described in detail later in this document.

The contents of the above files can be redirected:

Example:

In this example, the contents will be loaded from the file ‘here.cfg’.

```
[file path]
\i\put\it\over\here.cfg
```

Note: Any directory in the file system can be protected from web access by placing the file web-accs.cfg in the directory, see “Authorization” on page 84.

B. FTP Server

B.1 General

The built-in FTP server provides a way to access the file system using a standard FTP client.

The following port numbers are used for FTP communication:

- TCP, port 20 (FTP data port)
- TCP, port 21 (FTP command port)

Security Levels

The FTP server features two security levels; admin and normal.

- **Normal level users**
  The root directory will be '\user'.

- **Admin level users**
  The root directory will be '\', i.e. the user has unrestricted access to the file system.

User Accounts

The user accounts are stored in two files, which are protected from web access:

- **'\user\pswd\sys_pswd.cfg'**
  This file holds the user accounts for normal level users.

- **'\pswd\ad_pswd.cfg'**
  This file holds the user accounts for admin level users.

File Format:

The format of these files are as follows:

```
Username1:Password1
Username2:Password2
Username3:Password3
```

Note: If no valid user accounts have been defined, the gateway will grant admin level access to all users. In such cases, the FTP accepts any username/password combination, and the root directory will be '\'.

B.2 FTP Connection Example (Windows Explorer)

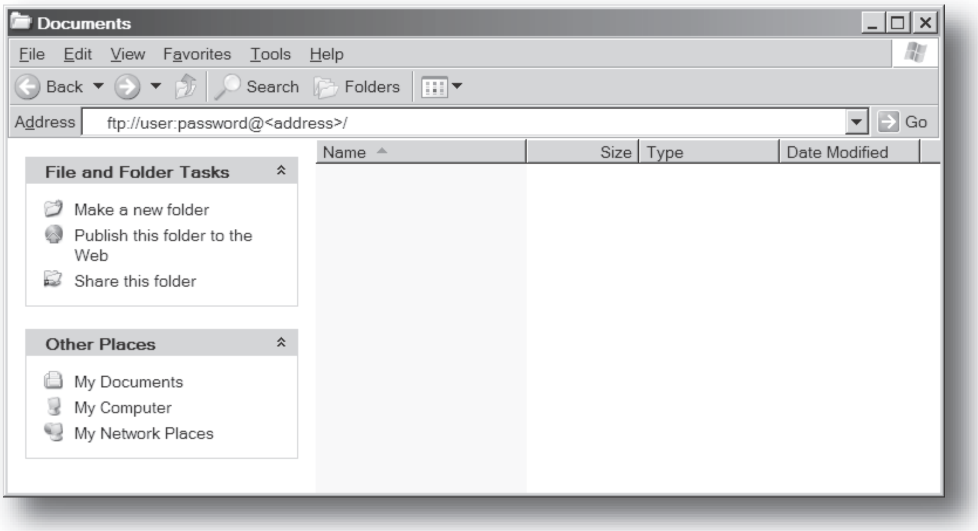
The built-in FTP client in Windows Explorer can be used to access the file system as follows:

1. Open the Windows Explorer by right-clicking on the 'Start' button and selecting 'Explore'.

2. In the address field, type FTP://<user>:<password>@<address>

- Substitute <address> with the IP address of the device
- Substitute <user> with the username
- Substitute <password> with the password

3. Press enter. The Explorer will now attempt to connect to the device using the specified settings. If successful, the built-in file system is displayed in the Explorer window.



Address ftp://user:password@<address>/

C. Advanced Network Configuration

C.1 Ethernet Configuration File (‘ethcfg.cfg’)

C.1.1 General

To exist on the network, the linking device needs a valid TCP/IP configuration. These settings are stored in the system file ‘\ethcfg.cfg’. Note that if TCP/IP settings are enabled in the configuration manager, then the IP address, gateway and subnet settings in ethcfg.cfg will be overwritten every time the device is restarted. All other settings are unaffected.

Parameter[a]	Default	Description/Comment
IP address	0.0.0.0	
Subnet mask	0.0.0.0	
Gateway address	0.0.0.0	
DHCP/BOOTP	OFF	DHCP/BOOTP Value: Meaning: ON Enabled OFF Disabled
Comm1	auto	Comm1, settings for port 1 Value: Meaning: auto Auto negotiation will be used 100FDX Forces port 1 of the device to operate only at 100 Mbit, full-duplex 100HDX Forces port 1 of the device to operate only at 100 Mbit, half-duplex 10FDX Forces port 1 of the device to operate only at 10 Mbit, full-duplex 10HDX Forces port 1 of the device to operate only at 10 Mbit, half-duplex
Comm2	auto	Comm2, settings for port 2 Value: Meaning: auto Auto negotiation will be used 100FDX Forces port 2 of the device to operate only at 100 Mbit, full-duplex 100HDX Forces port 2 of the device to operate only at 100 Mbit, half-duplex 10FDX Forces port 2 of the device to operate only at 10 Mbit, full-duplex 10HDX Forces port 2 of the device to operate only at 10 Mbit, half-duplex
HICP Password	“” (empty string)	
SMTP address	“” (empty string)	SMTP login/server settings. Username and password are only necessary if required by the server.
SMTP username	“” (empty string)	
SMTP password	“” (empty string)	
DNS1 address	0.0.0.0	Primary and secondary DNS. Required for resolving host names.
DNS2 address	0.0.0.0	
Domain name	“” (empty string)	Default domain name for not fully qualified host names (Max. 48 char.)
Host name	“” (empty string)	Host name (Max. 64 char.)
Mcast TTL	1	Multicast Time-To-Live
Mcast Alloc Control	0	Multicast address allocation
Mcast Num Mcast	1	Number of IP multicast addresses
Mcast Start Addr	0	Starting multicast address
802.1Q Enable	0	Enable 802.1Q Value: Meaning: 1 Enabled 0 Disabled

Parameter[a]	Default	Description/Comment
DSCP Urgent	55	Priority for CIP transport class 1 messages of different categories.
DSCP Scheduled	47	
DSCP High	43	
DSCP Low	31	
DSCP Explicit	27	Priority for CIP UCMM and CIP class 3 messages.
Select ACD	1	Enable Address Conflict Detection Value: Meaning: 1 Enabled 0 Disabled
ACD activity[b]	0	State of Address Conflict Detection activity.
Remote MAC[b]	00:00:00:00:00:00	MAC address of the remote node for which an address conflict is detected. Retrieved from the ARP PDU.
ARP PDU[b]	0	Address Resolution Protocol, Protocol Data Unit.

[a] When used in the configuration file, a parameter must be enclosed by brackets.

[b] Read-only parameter. Used to e.g. detect why a restart was required.

See also...

- "FTP Server" on page 79
- "Logix Network Interface" on page 36

C.2 IP Access Control

It is possible to specify which IP addresses are permitted to connect to the linking device. This information is stored in the system file '\ip_accs.cfg'.

File Format:

`[Web]`
`xxx.xxx.xxx.xxx`
	Nodes listed here may access the web server

`[FTP]`
`xxx.xxx.xxx.xxx`
	Nodes listed here may access the FTP server

`[Modbus-TCP]`
`xxx.xxx.xxx.xxx`
	Nodes listed here may access the device via Modbus-TCP

`[EtherNet/IP]`
`xxx.xxx.xxx.xxx`
	Nodes listed here may access the device via EtherNet/IP

`[All]`
`xxx.xxx.xxx.xxx`
	Fallback setting, used by the device when one or several of the keys above are omitted

Note: '*' may be used as a wildcard to select IP series.

D. Web Server

D.1 General

The linking device features a flexible web server with SSI capabilities. The built-in web pages can be customized to fit a particular application and allow access to I/O data and configuration settings.

The web server communicates through port 80.

See also...

- "Server Side Include (SSI)" on page 44
- "IP Access Control" on page 82

Protected Files

For security reasons, the following files are protected from web access:

- Files located in '\user\pswdcfg\pswd'
- Files located in '\pswd'
- Files located in a directory which contains a file named 'web_accs.cfg'

Default Web Pages

The linking device contains a set of virtual files which can be used when building a web page for configuration of network parameters. These virtual files can be overwritten (not erased) by placing files with the same name in the root of disc 0.

This makes it possible to, for example, make links from a web page to the virtual configuration page. In such case the link shall point to '\config.htm'.

These virtual files are:

```
\index.htm                    - Points to the contents of config.htm
\config.htm                   - Configuration frame page
\configform.htm               - Configuration form page
\configform2.htm              - Configuration form page
\store.htm                    - Configuration store page
\configuration.gif            - Configuration picture
\boarder.bg.gif               - picture
\boarder_m_bg.gif             - picture
\index.htm       l            - Points to the contents of config.htm
\eth_stat.html                - Configuration frame page
\cip_stat.html                - Configuration form page
\ip_config.shtm               - Configuration form page
\smtp_config.shtm             - Configuration store page
\style.css                    - HMS logo
\arrow_red.gif                - Configuration picture
```

D.2 Authorization

Directories can be protected from web access by placing a file called 'web_accs.cfg' in the directory to protect. This file shall contain a list of users that are allowed to access the directory and its subdirectories.

File Format:

```
Username1:Password1
Username2:Password2
...
UsernameN:PasswordN

[AuthName]
(message goes here)
```

- List of approved users.
- Optionally, a login message can be specified by including the key [AuthName]. This message will be displayed by the web browser upon accessing the protected directory.

The list of approved users can optionally be redirected to one or several other files.

Example:

In this example, the list of approved users will be loaded from the files 'here.cfg' and 'too.cfg'.

```
[File path]
\i\put\it\over\here.cfg
\i\actually\put\some\of\it\over\here\too.cfg

[AuthName]
Please enter password
```

Note that when using this feature, make sure to put the user/password files in a directory that is protected from web access, see "Protected Files" on page 83.

D.3 Content Types

By default, the following content types are recognized by their file extension:

Content Type	File Extension
text/html	*.htm, *.html, *.shtm
image/gif	*.gif
image/jpeg	*.jpeg, *.jpg, *.jpe
image/x-png	*.png
application/x-javascript	*.js
text/plain	*.bat, *.txt, *.c, *.h, *.cpp, *.hpp
application/x-zip-compressed	*.zip
application/octet-stream	*.exe, *.com
text/vnd.wap.wml	*.wml
application/vnd.wap.wmlc	*.wmlc
image/vnd.wap.wbmp	*.wbmp
text/vnd.wap.wmlscript	*.wmls
application/vnd.wap.wmlscriptc	*.wmlsc
text/xml	*.xml
application/pdf	*.pdf

It is possible to configure/reconfigure the reported content types, and which files that shall be scanned for SSI. This is done in the system file '\http.cfg'.

File Format:

```
[FileTypes]
FileType1:ContentType1
FileType2:ContentType2
...
FileTypeN:ContentTypeN

[SSIFileTypes]
FileType1
FileType2
...
FileTypeN
```

Note: Up to 50 content types and 50 SSI file types may be specified in this file.

E. E-mail Client

E.1 General

The built-in e-mail client can send predefined e-mail messages based on trigger-events in input and output data areas. The client supports SSI, however note that some SSI functions cannot be used in e-mail messages (specified separately for each SSI function).

See also...

- “Server Side Include (SSI)” on page 44

Server Settings

The linking device needs a valid SMTP server configuration in order to be able to send e-mail messages. These settings are stored in the system file ‘\ethcfg.cfg’.

See also...

- “Advanced Network Configuration” on page 81

Event-Triggered Messages

As mentioned previously, the e-mail client can send predefined messages based on events in the input and output data areas. In operation, this works as follows:

1. The trigger source is fetched from a specified location
2. A logical AND is performed between the trigger source and a mask value
3. The result is compared to a reference value
4. If the result is true, the e-mail is sent to the specified recipient(s).

Which events that shall cause a particular message to be sent, is specified separately for each message. For more information, see “E-mail Definitions” on page 87.

Note that the input and output data areas are scanned twice per second, i.e. to ensure that an event is detected by the device, it must be present longer than 0.5 seconds.

E.2 E-mail Definitions

The e-mail definitions are stored in the following two directories:

- **'\user\email'**
  This directory holds up to 10 messages which can be altered by normal level FTP users.

- **'\email'**
  This directory holds up to 10 messages which can be altered by admin level FTP users.

E-mail definition files must be named 'email_1.cfg', 'email_2.cfg'... 'email_10.cfg' in order to be properly recognized by the device.

File Format:

```
[Register]
Area, Offset, Type

[Register Match]
Value, Mask, Operand

[To]
recipient

[From]
sender

[Subject]
subject line

[Headers]
Optional extra headers

[Message]
message body
```

Key	Value	Scanned for SSI
Area	Source area. Possible values: 'IN' (Input Data area) or 'OUT' (Output Data area).	No
Offset	Source offset, written in decimal or hexadecimal.	
Type	Source data type. Possible values are 'byte', 'word', and 'long'.	
Value	Used as a reference value for comparison.	
Mask	Mask value, applied on the trigger source prior to comparison (logical AND).	
Operand	Possible values are '<', '=' or '>'.	
To	E-mail recipient.	Yes
From	Sender e-mail address.	
Subject	E-mail subject. One line only.	
Headers	Optional; may be used to provide additional headers.	
Message	The actual message.	

Note: Hexadecimal values must be written with the prefix '0x' in order to be recognized by the linking device.

F. CIP Object Implementation

F.1 General

The following CIP objects are implemented in this product:

Mandatory Objects

Object	Page
Identity Object, Class 01h	88
Message Router, Class 02h	90
Assembly Object, Class 04h	90
Port Object, Class F4h	97
TCP/IP Interface Object, Class F5h	98
Ethernet Link Object, Class F6h	99

Vendor Specific Objects

Object	Page
DLR Object, Class 47h	92
Parameter Data Input Mapping Object, Class B0h	95
Parameter Data Output Mapping Object, Class B1h	96

F.2 Identity Object, Class 01h

F.2.1 General Information

Object Description

-

Supported Services

Class services:	Get Attribute All Get Attribute Single
Instance services:	Get Attribute All Get Attribute Single Reset

F.2.2 Class Attributes

#	Access	Name	Type	Value	Description
1	Get	Revision	UINT	0001h	Revision 1

F.2.3 Instance Attributes

#	Access	Name	Type	Value	Description
1	Get	Vendor ID	UINT	Default: 005Ah	HMS Industrial Networks AB
2	Get	Device Type	UINT	Default: 000Ch	Communication Adapter
3	Get	Product Code	UINT	Default: 0054h	Anybus Communicator
4	Get	Revision	Struct of:		-
			USINT		Major fieldbus version
			USINT		Minor fieldbus version
5	Get	Status	WORD	-	Device status, see table below
6	Get	Serial Number	UDINT	Serial number	(set at production)
7	Get	Product Name	SHORT_STRING	Anybus Communicator	Name of product

Device Status

bit(s)	Name
0	Module Owned
1	(reserved)
2	Configured
3	(reserved)
4... 7	Extended Device Status:
	Value:Meaning: 0000b Unknown 0010b Faulted I/O Connection 0011b No I/O connection established 0100b Non-volatile configuration bad 0110b Connection in Run mode 0111b Connection in Idle mode (other) (reserved)
8	Set for minor recoverable faults
9	Set for minor unrecoverable faults
10	Set for major recoverable faults
11	Set for major unrecoverable faults
12... 15	(reserved)

F.3 Message Router, Class 02h

F.3.1 General Information

Object Description

-

Supported Services

Class services:	-
Instance services:	-

F.3.2 Class Attributes

-

F.3.3 Instance Attributes

-

F.4 Assembly Object, Class 04h

F.4.1 General Information

Object Description

This object provides access to the I/O Data in the input and output data areas in the linking device.

See also...

- "Linking Device IP Address Configuration" on page 20
- "Logix Network Interface" on page 36

Supported Services

Class services:	Get Attribute Single
Instance services:	Get Attribute Single Set Attribute Single

F.4.2 Class Attributes

#	Access	Name	Type	Value	Description
1	Get	Revision	UINT	0002h	Revision 2
2	Get	Max Instance	UINT	-	The highest initiated instance no.

F.4.3 Instance 64h (100) Attributes

This instance corresponds to I/O data (input) in the device.

Note: If the I/O input data size is set to 0 this instance will NOT be initialized.

#	Access	Name	Type	Value	Description
3	Get	Data	Array of BYTE	-	Data produced by the gateway

F.4.4 Instance 96h (150) Attributes

Note: If the I/O output data size is set to 0 this instance will NOT be initialized.

#	Access	Name	Type	Value	Description
3	Set	Data	Array of BYTE	-	Data consumed by the gateway[a]

[a] Rockwell Automation PLCs have the first four bytes consumed by a device defined as status information. This behavior is specific to devices from Rockwell Automation and is not defined in the EtherNet/IP specification. However, since all known PLCs are implemented this way, the linking device adopts this behavior and strips off the corresponding four bytes from the consumed data.

F.4.5 Instance C6h (198) Attributes (Heartbeat Input-Only)

This instance is used as heartbeat for input-only connections, and does not carry any data.

F.4.6 Instance C7h (199) Attributes (Heartbeat, Listen-Only)

This instance is used as heartbeat for listen-only connections, and does not carry any data.

F.5 DLR Object, Class 47h

General Information

Object Description

Information about the Device Level Ring (DLR) can be read from this object. An Announced-based ring participant is supported.

Note: The module will only act as an Announce-based DLR participant, not as a DLR ring supervisor.

Supported Services

Class services:	Get Attributes All Get Attribute Single
Instance services:	Get Attributes All Get Attribute Single

F.5.1 Class Attributes

#	Access	Name	Type	Value
1	Get	Revision	UINT	0002h

F.5.2 Instance Attributes, Instance 01h

#	Access	Name	Type	Description
01h	Get	Network Topology	USINT	Gives network topology. 0 - Linear 1 - Ring
02h	Get	Network Status	USINT	Gives network status. 0 - Normal 1 - Ring Fault
10h	Get	Active Supervisor Address	Struct of:	
			UDINT	Supervisor IP address
			Array of BYTE	Supervisor MAC address
12h	Get	Capability Flags	DWORD	Describes the DLR capabilities of the device, see below Value: 01h

F.5.3 Capability Flags

Bit #	Name	Definition
0	Announce-based Ring Node	Set if Announce frames are used[a]
1	Beacon-based Ring Node	Set if Beacon frames are used[a]
2 - 4	Reserved	Set to zero
5	Supervisor capable	Set if device is supervisor capable. Set to 0 as this module is not supervisor capable.
6 - 31	Reserved	Set to zero

a. Either bit 0 or bit 1 shall be set.

F.6 QoS Object, Class 48h

General Information

Object Description

This object sets up QoS services for the module. Quality of service is the ability to provide different priority to different applications on a congested network in order to guarantee a certain level of performance to a data flow. In the absence of congestion QoS is not needed.

Supported Services

Class services:	Get Attribute All Get Attribute Single
Instance services:	Get Attribute Single Set Attribute Single

Class Attributes

#	Access	Name	Type	Value
1	Get	Revision	UINT	0001h

Instance Attributes, Instance 01h

#	Access	Name	Type	Description
01h	Set	802.1Q Tag Enable	USINT	Enables or disables sending 802.1Q frames. 0 - Disabled (default) 1 - Enabled
04h	Set	DSCP Urgent	USINT	CIP transport class 1 messages with Urgent priority Default: 55
05h	Set	DSCP Scheduled	USINT	CIP transport class 1 messages with Scheduled priority Default: 47
06h	Set	DSCP High	USINT	CIP transport class 1 messages with High priority Default: 43
07h	Set	DSCP Low	USINT	CIP transport class 1 messages with Low priority Default: 31
08h	Set	DSCP Explicit	USINT	CIP UCMM and CIP class 3 Default: 27

F.7 Diagnostic Object, Class AAh

F.7.1 General Information

Object Description

This object groups diagnostic information for the network interface.

Supported Services

Class services:	Get Attribute All
Instance services:	Get Attribute Single

F.7.2 Class Attributes

#	Access	Name	Type	Value	Description
1	Get	Revision	UINT	0001h	Revision 1

F.7.3 Instance Attributes, Instance 01h

#	Access	Name	Type	Description
01h	Get	Module serial number	UDINT	Serial number
02h	Get	Vendor ID	UINT	Manufacturer Vendor ID
03h	Get	Fieldbus Type	UINT	Fieldbus Type
04h	Get	Module Software version	UINT	Module software version
0Ah	Get	Module Type	UINT	Module Type
0Fh	Get	IN cyclic I/O length	UINT	Size of I/O Input area (in bytes)
11h	Get	IN total length	UINT	Total number of IN bytes supported
12h	Get	OUT cyclic I/O length	UINT	Size of I/O Output area (in bytes)
14h	Get	OUT total length	UINT	Total number of OUT bytes supported

F.8 Parameter Data Input Mapping Object, Class B0h

F.8.1 General Information

Object Description

This object can be used to access input data acyclically, and is set up dynamically based on the Parameter Data Mailbox initialization (see “Parameter Data Initialization (Explicit Data)” on page 121).

See also...

- “Linking Device IP Address Configuration” on page 20
- “Logix Network Interface” on page 36
- “Parameter Data Output Mapping Object, Class B1h” on page 96
- “Parameter Data Initialization (Explicit Data)” on page 121

Supported Services

Class services: Get Attribute All

Instance services: Get Attribute Single

F.8.2 Class Attributes

#	Access	Name	Type	Value	Description
1	Get	Revision	UINT	0001h	Revision 1

F.8.3 Instance Attributes, Instance 01h

Each attribute corresponds to a block of Input Data. Note that the size and location of each block must be specified using the configuration manager.

For more information, see “Parameter Data Initialization (Explicit Data)” on page 121.

#	Access	Name	Type	Description
01h	Get	Data	Array of USINT	Mapped block if Input Data
02h	Get	Data	Array of USINT	Mapped block if Input Data
02h	Get	Data	Array of USINT	Mapped block if Input Data
02h	Get	Data	Array of USINT	Mapped block if Input Data
02h	Get	Data	Array of USINT	Mapped block if Input Data
02h	Get	Data	Array of USINT	Mapped block if Input Data
...	...	...	...	...
32h	Get	Data	Array of USINT	Mapped block if Input Data

F.9 Parameter Data Output Mapping Object, Class B1h

F.9.1 General Information

Object Description

This object can be used to access output data acyclically, and is set up dynamically based on the Parameter Data Mailbox initialization (see "Parameter Data Initialization (Explicit Data)" on page 121).

See also...

- "Linking Device IP Address Configuration" on page 20
- "Logix Network Interface" on page 36
- "Parameter Data Input Mapping Object, Class B0h" on page 95
- "Parameter Data Initialization (Explicit Data)" on page 121

Supported Services

Class services:	Get Attribute All
Instance services:	Get Attribute Single Set Attribute Single

F.9.2 Class Attributes

#	Access	Name	Type	Value	Description
1	Get	Revision	UINT	0001h	Revision 1

F.9.3 Instance Attributes, Instance 01h

Each attribute corresponds to a block of output data. Note that the size and location of each block must be specified using the configuration manager.

For more information, see "Parameter Data Initialization (Explicit Data)" on page 121

#	Access	Name	Type	Description
01h	Get/Set	Data	Array of USINT	Mapped block of Output Data
02h	Get/Set	Data	Array of USINT	Mapped block of Output Data
01h	Get/Set	Data	Array of USINT	Mapped block of Output Data
02h	Get/Set	Data	Array of USINT	Mapped block of Output Data
01h	Get/Set	Data	Array of USINT	Mapped block of Output Data
02h	Get/Set	Data	Array of USINT	Mapped block of Output Data
...	...	...	...	...
32h	Get/Set	Data	Array of USINT	Mapped block of Output Data

F.10 Port Object, Class F4h

F.10.1 General Information

Object Description

-

Supported Services

Class services:	Get Attribute All Get Attribute Single
Instance services:	Get Attribute All Get Attribute Single

F.10.2 Class Attributes

#	Access	Name	Type	Value	Description
1	Get	Revision	UINT	0001h	Revision 1
2	Get	Max Instance	UINT	0002h	2 is the highest instance number
3	Get	No. of instances	UINT	0001h	1 instance is implemented
8	Get	Entry Port	UINT	0002h	Returns the instance of the Port object that describes the port.
9	Get	All Ports	Array of STRUCT {UINT; UINT;}	0000h 0000h 0000h 0000h 0004h 0002h	Array of structure containing attributes 1 and 2 from each instance. Instance 1 is at byte offset 4. Instance 2 is at byte offset 8, etc. The 4 bytes at offset 0 shall be 0. (Default)

F.10.3 Instance Attributes, Instance 02h

#	Access	Name	Type	Value	Comments
1	Get	Port Type	UINT	0004h	TCP/IP
2	Get	Port Number	UINT	0002h	Port 2
3	Get	Port Object	Struct of:		
		Path Size	UINT	0002h	-
		Path	Padded EPATH	20 F5 24 01h	TCP class, Instance 1
4	Get	Port Name	SHORT_STIRNG	‘TCP/IP’	Name of port
8	Get	Node Address	Padded EPATH	-	-

F.11 TCP/IP Interface Object, Class F5h

F.11.1 General Information

Object Description

This object groups TCP/IP-related settings.

See also...

- "Advanced Network Configuration" on page 81
- "Logix Network Interface" on page 36

Supported Services

Class services:	Get Attribute All Get Attribute Single
Instance services:	Get Attribute All Get Attribute Single Set Attribute Single

F.11.2 Class Attributes

#	Access	Name	Type	Value	Description
1	Get	Revision	UINT	0001h	Revision 3

F.11.3 Instance Attributes

#	Access	Name	Type	Value	Comments
1	Get	Status	DWORD	00000001h	Attribute #5 contains valid information.
2	Get	Configuration Capability	DWORD	00000014h	Attribute #5 is settable Capable of obtaining network configuration via DHCP.
3	Get/Set	Configuration Control	DWORD	-	Value:Meaning: 0 Configuration from non-volatile memory 2 Configuration from DHCP
4	Get	Port Object	Struct of:		
		Path Size	UINT	0002h	2 words
		Path	Padded EPATH	20 F6 24 01h	Path to Ethernet Class, Instance 1
5	Get/Set	Interface Configuration	Struct of:		
		IP Address	UDINT	-	IP address
		Subnet Mask	UDINT	-	Subnet mask
		Gateway Address	UDINT	-	Gateway Address
		Name Server 1	UDINT	-	Primary DNS
		Name Server 2	UDINT	-	Secondary DNS
		Domain Name	STRING	-	Default domain name
6	Get/Set	Host Name	STRING	-	Host name

F.12 Ethernet Link Object, Class F6h

F.12.1 General Information

Object Description

This object groups diagnostic information for the Ethernet interface.

See also...

- "Advanced Network Configuration" on page 81

Supported Services

Class services:	Get Attribute All Get Attribute Single
Instance services:	Get Attribute All Get Attribute Single

F.12.2 Class Attributes

#	Access	Name	Type	Value	Description
1	Get	Revision	UINT	0001h	Revision 3
2	Get	Max Instance	UINT	0001h	2 is the highest instance number
3	Get	No. of instances	UINT	0001h	2 instances are implemented

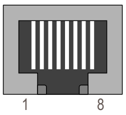
F.12.3 Instance Attributes

#	Access	Name	Type	Value	Comments
1	Get	Interface Speed	UDINT	10 or 100	Actual Ethernet interface speed
2	Get	Interface Flags	DWORD	-	-
3	Get	Physical Address	Array of 6 USINTS	(MAC ID)	Physical network address
4	Get	Interface Counters	Struct:		
		In Octets	UDINT	-	Octets received on the interface
		In Ucast Packets	UDINT	-	Unicast packets received on the interface
		In NUcast Packets	UDINT	-	Non-unicast packets received on the interface
		In Discards	UDINT	-	Inbound packets with unknown protocol
		In Errors	UDINT	-	Inbound packets that contain errors (does not include discards)
		In Unknown Protos	UDINT	-	Inbound packets with unknown protocol
		Out Octets	UDINT	-	Octets sent on the interface
		Out Ucast Packets	UDINT	-	Unicast packets sent on the interface
		Out NUcast Packets	UDINT	-	Non-unicast packets sent on the interface
		Out Discards	UDINT	-	Outbound packets with unknown protocol
		Out Errors	UDINT	-	Outbound packets that contain errors (does not include discards)
5	Get	Media Counters	Struct:		
		Alignment Errors	UDINT	-	Frames received that are not an integral number of octets in length
		FCS Errors	UDINT	-	Frames received that do not pass the FCS check
		Single Collisions	UDINT	-	Successfully transmitted frames which experienced exactly one collision
		Multiple Collisions	USINT	-	Successfully transmitted frames which experienced more than one collision
		SQE Test Errors	UDINT	0	-
		Deferred Transmissions	UDINT	-	Frames for which first transmission attempt is delayed because the medium is busy
		Late Collisions	UDINT	-	Number of times a collision is detected later than 512 bit-times into the transmission of a packet
		Excessive Collisions	UDINT	-	Frames for which a transmission fails due to excessive collisions
		MAC Transmit Errors	UDINT	-	Frames for which transmission fails due to an internal MAC sublayer receive error
		Carrier Sense Errors	UDINT	-	Times that the carrier sense condition was lost or never asserted when attempted to transmit a frame
		Frame Too Long	UDINT	-	Frames received that exceed the maximum permitted frame size
		MAC Receive Errors	UDINT	-	Frames for which reception on an interface fails due to an internal MAC sublayer receive error

G. Connector Pin Assignments

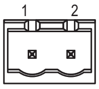
G.1 Ethernet Connector

Pin	Signal
Housing	Cable Shield
1	TD+
2	TD-
3	RD+
4	Termination
5	Termination
6	RD-
7	Termination
8	Termination



G.2 Power Connector

Pin	Description
1	+24 VDC
2	GND



Notes:

- Use 60/75 or 75 °C copper (Cu) wire only.
- Minimum terminal tightening torque: 5–7 lb-in (0.5–0.8 Nm).

G.3 Subnetwork Interface

G.3.1 General Information

The subnetwork interface provides for RS232, RS422 and RS485 communications. Depending on the configuration specified in the configuration manager, different signals are activated in the subnetwork connector.

G.3.2 Bias Resistors (RS485 Only)

When idle, RS485 enters an indeterminate state, which may cause the serial receivers to pick up noise from the serial lines and interpret this as data. To prevent this, the serial lines should be forced into a known state using pull-up and pull-down resistors, commonly known as bias resistors.

The bias resistors form a voltage divider, forcing the voltage between the differential pair to be higher than the threshold for the serial receivers, typically >200 mV.

Note that bias resistors shall only be installed on one node; installing bias resistors on several nodes may compromise the signal quality on the network and cause transmission problems.

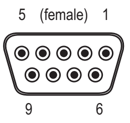
G.3.3 Termination (RS485 & RS422 Only)

To avoid reflections on the serial lines, it is important to properly terminate the subnetwork by placing termination resistors between the serial receivers near the end nodes.

The resistor value should ideally match the characteristic impedance of the cable, typically 100–120 Ω.

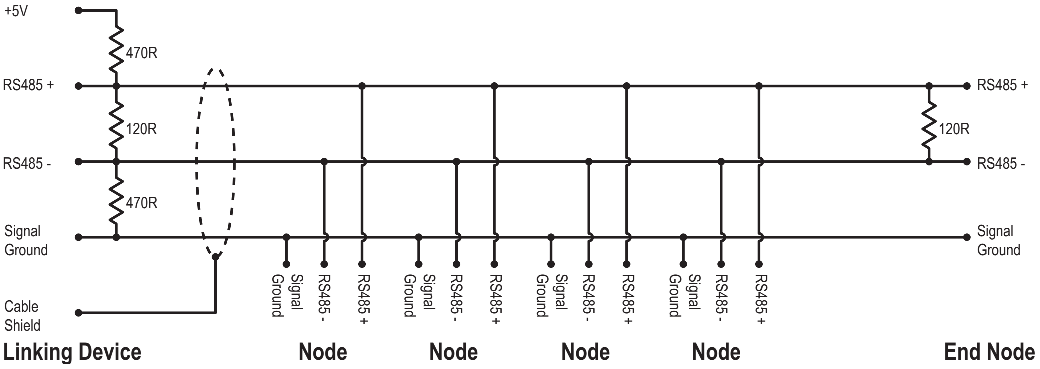
G.3.4 Connector Pinout (DB9F)

Pin	Description	RS232	RS422	RS485
1	+5 V Output(100 mA max)	✓	✓	✓
2	RS232 Rx	✓		
3	RS232 Tx	✓		
4	(reserved)			
5	Signal Ground[a]	✓	✓	✓
6	RS422 Rx +		✓	
7	RS422 Rx -		✓	
8	RS485 + / RS422 Tx+		✓	✓
9	RS485 - / RS422 Tx-		✓	✓
(housing)	Cable Shield	✓	✓	✓

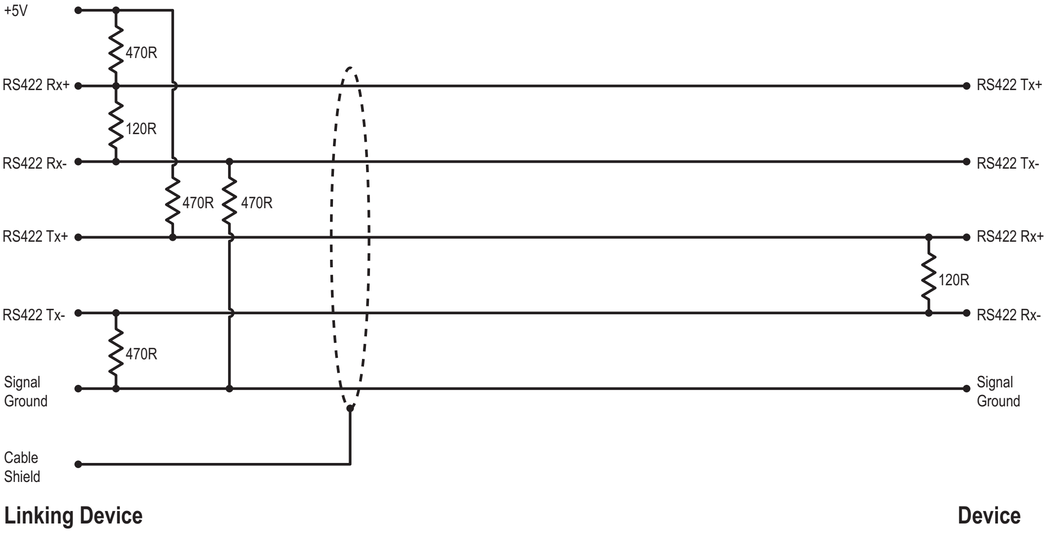


[a] Connecting this signal directly to Protective Earth (PE) of other nodes may, in case of grounding loops etc., cause damage to the on-board serial transceivers. It is therefore generally recommended to connect it only to Signal Ground (if available) of other nodes.

G.3.5 Typical Connection (RS485)

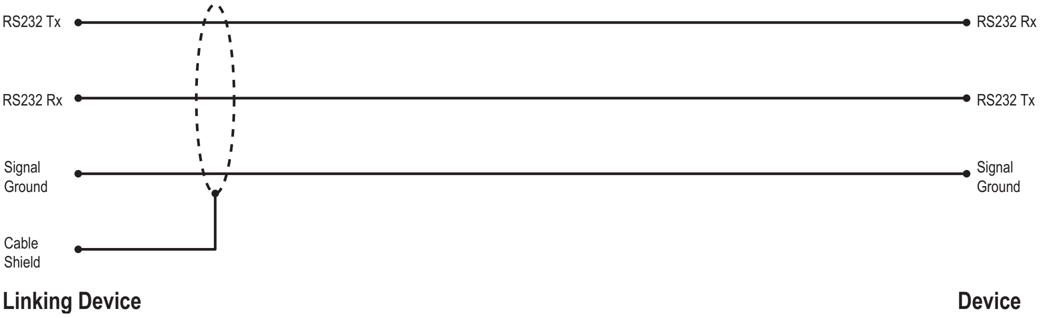


G.3.6 Typical Connection (RS422 & 4-Wire RS485)



Note: Bias resistors are normally not needed on RS422, but may be required when using 4-wire RS485.

G.3.7 Typical Connection (RS232)



H. Technical Specification

H.1 Mechanical Properties

Housing

Plastic housing with snap-on connection to DIN-rail, protection class IP20.

Dimensions (L x W x H)

120 mm x 75 mm x 27 mm (4.72” x 2.95” x 1.06”)

H.2 Electrical Characteristics

Power Supply

Power: 24 VDC ± 10%

Power Consumption

Maximum power consumption is 280 mA on 24 VDC. Typically around 100 mA.

H.3 Environmental Characteristics

Relative Humidity

The product is designed for a relative humidity of 0 to 95% non-condensing.

Temperature

Operating:	0 °C to +55 °C
Non-operating:	-25 °C to +85 °C

H.4 Regulatory Compliance

EMC Compliance (CE)

This product is in accordance with the EMC directive 89/336/EEC, with amendments 92/31/EEC and 93/68/EEC through conformance with the following standards:

- **EN 50082-2 (1993)**
  EN 55011 (1990) Class A
- **EN 61000-6-2 (1999)**
  EN 61000-4-3 (1996) 10 V/m
  EN 61000-4-6 (1996) 10 V/m (all ports)
  EN 61000-4-2 (1995) ±8 kV air discharge, ±4 kV contact discharge
  EN 61000-4-4 (1995) ±2 kV power port, ±1 kV other ports
  EN 61000-4-5 (1995) ±0.5 kV power ports (DM/CM), ±1 kV signal ports

UL/c-UL Compliance

C UL US LISTED

IND: CONT. EQ.
FOR HAZ LOC.
CL I, DIV 2
GP A,B,C,D
TEMP
CODE
E203225

WARNING - EXPLOSION HAZARD - SUBSTITUTION OF ANY COMPONENTS MAY IMPAIR SUITABILITY FOR CLASS I, DIVISION 2.

WARNING - EXPLOSION HAZARD - WHEN IN HAZARDOUS LOCATIONS, TURN OFF POWER BEFORE REPLACING OR WIRING MODULES.

WARNING - EXPLOSION HAZARD - DO NOT DISCONNECT EQUIPMENT UNLESS POWER HAS BEEN SWITCHED OFF OR THE AREA IS KNOWN TO BE NON-HAZARDOUS.

ATTENTION – RISQUE D’EXPLOSION – LE REMPLACEMENT DE TOUT COMPOSANTS INVALIDE LA CERTIFICATION CLASS I, DIVISION 2.

ATTENTION – RISQUE D’EXPLOSION – EN ZONE EXPLOSIVE, VEUILLEZ COUPER L’ALIMENTATION ÉLECTRIQUE AVANT LE REMPLACEMENT OU LE RACCORDEMENT DES MODULES.

ATTENTION – RISQUE D’EXPLOSION – NE PAS DÉCONNECTER L’ÉQUIPEMENT TANT QUE L’ALIMENTATION EST TOUJOURS PRÉSENTE OU QUE LE PRODUIT EST TOUJOURS EN ZONE EXPLOSIVE ACTIVE.

Additional installation and operating instructions

- Max Ambient Temperature: 55 °C (for Hazloc environments)
- Field wiring terminal markings (wire type (Cu only, 14–30 AWG)).
- Use 60/75 or 75 °C copper (Cu) wire only.
- Terminal tightening torque must be 5–7 lb-in (0.5–0.8 Nm).
- Use in overvoltage category 1 pollution degree 2 environment.
- Installed in an enclosure considered representative of the intended use.
- Secondary circuit intended to be supplied from an isolating source and protected by overcurrent protective devices installed in the field sized per the following:

Control circuit wire size		Maximum protective device rating
AWG	mm²	Amperes
22	0.32	3
20	0.52	5
18	0.82	7
16	1.3	10
14	2.1	20
12	3.3	25

Galvanic isolation on subnetwork interface

- **EN 60950-1 (2001)**

  Pollution Degree 2

  Material Group IIIb

  250 V_{RMS} or 250 VDC working voltage

  500 V secondary circuit transient rating

CIP Product Compliance

ODVA™
CONFORMANT

I. Troubleshooting

Problem	Solution
Problem during configuration Upload / Download. The Config Line “LED” turns red in the configuration manager.	- Serial communication failed. Try again
The serial port seems to be available, but it is not possible to connect to the linking device	- The serial port may be in use by another application. Exit the configuration manager and close all other applications including the ones in the system tray. Try again - Select another serial port Try again
Poor performance	- Right click “subnetwork” in the Navigation window and select “subnetwork Status” to see status / diagnostic information about the subnetwork. If the device reports very many retransmissions, check your cabling and/or try a lower baud rate setting for the subnetwork (if possible). - Is the Subnet Monitor in the configuration manager active? The subnetwork monitor has a negative influence on the overall performance of the device, and should only be used when necessary. - Is the Node Monitor in the configuration manager active? The node monitor has a negative influence on the overall performance of the device, and should only be used when necessary.
No subnetwork functionality	- Use the “Data logger”-functionality to record the serial data communication on the subnetwork. - If no data is being transmitted, check the configuration in the configuration manager. - If no data is received, check the subnetwork cables. Also verify that the transmitted data is correct.

J. ASCII Table

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0x	NUL 0	SOH 1	STX 2	ETX 3	EOT 4	ENQ 5	ACK 6	BEL 7	BS 8	HT 9	LF 10	VT 11	FF 12	CR 13	SO 14	SI 15
1x	DLE 16	DC1 17	DC2 18	DC3 19	DC4 20	NAK 21	SYN 22	ETB 23	CAN 24	EM 25	SUB 26	ESC 27	FS 28	GS 29	RS 30	US 31
2x	(sp) 32	! 33	" 34	# 35	$ 36	% 37	& 38	' 39	(40	) 41	* 42	+ 43	, 44	- 45	. 46	/ 47
3x	0 48	1 49	2 50	3 51	4 52	5 53	6 54	7 55	8 56	9 57	: 58	; 59	< 60	= 61	> 62	? 63
4x	@ 64	A 65	B 66	C 67	D 68	E 69	F 70	G 71	H 72	I 73	J 74	K 75	L 76	M 77	N 78	O 79
5x	P 80	Q 81	R 82	S 83	T 84	U 85	V 86	W 87	X 88	Y 89	Z 90	[91	\ 92	] 93	^ 94	_ 95
6x	` 96	a 97	b 98	c 99	d 100	e 101	f 102	g 103	h 104	i 105	j 106	k 107	l 108	m 109	n 110	o 111
7x	p 112	q 113	r 114	s 115	t 116	u 117	v 118	w 119	x 120	y 121	z 122	{ 123	| 124	} 125	~ 126	DEL 127

K. Copyright Notices

This product includes software developed by Carnegie Mellon, the Massachusetts Institute of Technology, the University of California, and RSA Data Security: