

Tool Software

xAscender Studio **User's Manual**

Communication

WUME-XASCEN-COM-03

Disclaimer

© 2022 Panasonic Industrial Devices SUNX Co., Ltd.

Subject to change without notice

The information contained in this document is provided for informational purposes only. While efforts were made to verify the accuracy of the information contained in this documentation, it is provided 'as is' without warranty of any kind.

- The copyright of this manual is owned by Panasonic Industrial Devices SUNX Co., Ltd.
- Unauthorized reproduction of this manual is strictly prohibited.
- Windows is a registered trademark of Microsoft Corporation in the U.S. and other countries.
- Ethernet is a registered trademark of FUJIFILM Business Innovation Co., Ltd. and Xerox Corporation.
- QR Code is a registered trademark of DENSO WAVE INCORPORATED.
- Other company and product names are trademarks or registered trademarks of their respective companies.

Third-party brands and names are the property of their respective owners.

The example companies, organizations, products, domain names, e-mail addresses, logo, people, places, and events depicted herein are fictitious. No association with any real company, organization, product, domain name, e-mail address, logo, person, place or event is intended or should be inferred.

This products/software contains software licensed under the GNU General Public License, Version 2.0 (GPL V2.0), software licensed under the GNU LESSER General Public License, Version 2.1 (LGPL V2.1), and/or open source software other than the software licensed under the GPL V2.0 and/or LGPL V2.1. The software open source included is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY, without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

Contents

A-B DF1	
A-B DH-485	14
A-B ENET	27
ABB Mint Controller HCP	
BACnet	
Beckhoff ADS	87
Client System Variables	103
CODESYS V2 ETH	105
CODESYS V2 SER	118
CODESYS V3 ETH	127
Control Techniques Modbus TCP	138
Delta Modbus RTU	142
Direct Serial	152
Direct Socket	160
DMX512 Digital Multiplex	171
Eaton Suconet-K	176
Environment Variables	181
Ethernet/IP CIP	182
Fatek FACON ETH	207
Fatek FACON SER	214
GE Intelligent Platforms SNP	220
GE Intelligent Platforms SRTP	231
GE SRTP	241
Hitachi SER	252
Hitachi ETH	257
IDEC Maintenance	261
Jetter Ext ETH	271
Keyence KV	279
Koyo DL	289
Koyo DL ETH	295

Modbus RTU	300
Modbus RTU Server	315
Modbus TCP	330
Modbus TCP Server	347
Mitsubishi FX ETH	358
Mitsubishi FX SER	372
Mitsubishi iQ/Q/L ETH	381
Mitsubishi iQ/Q/L SER	394
NMEA 0183	403
Omron FINS ETH	427
Omron FINS SER	441
OPC UA Client	450
Panasonic FP/FP7	464
Ping	472
ROBOX BCC/31	477
SAIA S-BUS	486
SAIA S-BUS ETH	495
Simatic S7 PPI	502
Siemens S7 Optimized	509
Simatic S7 ETH	523
Simatic S7 MPI	562
System Variables	598
Variables	600

A-B DF1

The A-B DF1 communication driver has been designed to connect HMI devices to a Allen-Bradley controllers through serial communication.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the **PLC** list.

The protocol configuration dialog is displayed.

	X
	Comm OK
0	Cancel
BCC	•
	*
	Ξ
	-

Element	Description
Node ID	Serial node associated to the PLC.
Checksum type	It can be BCC or CRC , depending on PLC settings.
PLC Models	PLC models available:
	 PLC3 PLC5/10/12/15/25 PLC5/40/40L PLC5/60/60L SLC500 Fixed I/O

ment	Description	
	 SLC500 Mod Micrologix 10 Micrologix 15 Ultra5000 	00
mm	If clicked displays the	e communication parameters setup dialog.
	Comm Parameter Dia Port Baudrate Parity Data bits Stop bits Mode	alog ОК Сот Сот 19200 • еven 8 • КS-232 •
	Element	Parameter
	Port	 Serial port selection. COM1: device PLC port. COM2: computer/printer port on panels with 2 serial ports or optional Plug-In module plugged on Slot 1/2 for panels with 1 serial port on-board. COM3: optional Plug-In module plugged on Slot 3/4 for panels with 1 serial port on-board.
	Baudrate, Parity, Data Bits, Stop bits	Serial line parameters.
	Mode	 Serial port mode. Available modes: RS-232. RS-485 (2 wires). RS-422 (4 wires).

Tag Editor Settings

In Tag Editor select the protocol A-B DF1.

Add a tag using [+] button. Tag setting can be defined using the following dialog:

A-B DF1	×	J
A-B DF1		
File Type	Element Subindex	
Disc In 👻	0 💌 0 👻	
File Num	Data Type Arraysize	
7	short	
Conversion	Sub Element	
	+/- 0	
	OK Cancel Apply Help	

Element	Description				
Memory Type	Memory Type	Description			
туре	Disc Out	Discrete output value. O resource on PLC.			
	Disc In	Discrete input value. I resource on PLC.			
	Status	Status value. S resource on PLC.			
	Bit	Bit value. B resource on PLC.			
	Timer	Timer value. T resource on PLC.			
	Counter	Counter value. C resource on PLC.			
	Control	Control value. R resource on PLC.			
	Integer	Integer value. N resource on PLC.			
	Float	Float value. F resource on PLC.			
Element	Represents the line of the res	ource while monitoring PLC values.			
Subindex	Represents the column of the	resource while monitoring PLC values.			
File Num	Instance of resource of the PLC.				
Data Type	Available data types:				
	• boolean				
	byteshort				
	• int				
	unsignedByte				
	unsignedShort				
	unsignedInt				
	float				
	double				
	• string				
	binary				
	See "Programming concepts"	section in the main manual.			
	Note: To define arra brackets (byte[], sh	ays, select one of Data Type format followed by square ort[]…).			
Arraysize	In case of array tag, thi	s property represents the number of array elements.			
	 In case of string tag, the available in the string tag 	is property represents the maximum number of bytes ag.			

Element	Description						
	Note: number of bytes corresponds to number of string characters if Encoding property i set to UTF-8 or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2 bytes.						
Sub Element	Allows to point	to specific part of a resource:					
	 0 (entire resource) PRE ACC LEN POS 						
Conversion	Conversion to I	be applied to the tag.					
	Conversion						
	inv,swap2 Depending on o	Allowed Configured BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits Cancel OK Cancel OK					
	Value	Description					
	Inv bits	inv: Invert all the bits of the tag.					
		<i>Example:</i> 1001 \rightarrow 0110 (in binary format) 9 \rightarrow 6 (in decimal format)					
	Negate	neg: Set the opposite of tag value.					
		<i>Example:</i> 25.36 → -25.36					
	AB -> BA	swapnibbles: Swap nibbles in a byte.					
		<i>Example:</i> 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)					
	ABCD -> CDAB	swap2: Swap bytes in a word.					

Element	Description	
	Value	Description
		<i>Example:</i> 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)
	ABCDEFGH	swap4: Swap bytes in a double word.
	-> GHEFCDAB	<i>Example:</i> 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)
	ABCNOP -	swap8: Swap bytes in a long word.
	> OPMDAB	Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) $0\ 10000000110$ 0001110010111011010000101101000011100101
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001 \ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)
	Select conversi	on and click +. The selected item will be added to list Configured .
	If more convers Configured).	ions are configured, they will be applied in order (from top to bottom of list
	Use the arrow b	outtons to order the configured conversions.

Tag Import

Exporting Tags from PLC

The A-B DF1 tag import filter accepts symbol files with extension ".csv" created by the Rockwell RSLogix 500.

To create the file select Tool > Database > ASCII Export

RSLogix 500 Pro - UNTITLED		
File Edit View Search Comms	Tools Window Help	
D 🖻 🖬 🎒 🐰 🖻 🖻	Options	🔽 🔽 🖁 🖌 🖌
OFFLINE 🛓 No Forces	Delete Unused Memory	∃E ∃∕E <> <0> <00> abl abs
No Edits 🛃 Forces Disable	Database 🔹 🕨	ASCII Export
Driver: AB_ETHIP-1	Security 🕨 🕨	ASCII Import
UNTITLED		Native Import Delete DB Delete Unused Addr. Edit Using Excel Edit Device Codes Adjust Rung Offset Convert Rung Attachment Rebuild DB

From **CSV** tab select the data to be exported and give a name to the output csv file.

Documentation Database ASCII	Export	×
RSLogix 500 A.I. AB APS CS	SV	
Data to be exported :	Destination file names and extensions :	
Addr/Symbol Desc. 🔽	file1 .CSV	
Instruction Comments	file2 .CSV	
Page Title / Rung Desc. 🛛 🗖		
Symbol Groups 📃	file3 .CSV	
Program File Names 🗖	UNTITLED	
- AI/AB Address and Instruction dea	scription formatting :	
Characters per line in target data	abase 👤 20	
Treat Source Description as 5 line	nes (truncating each line if necessary)	
C Treat Source Description as 1 lin	ne (truncating from the end if necessary)	
 OK	Cancel Help	

Importing Tags in Tag Editor

Select the driver in Tag Editor and click on the Import Tags button to start the importer.

	Tags	×								
+	_	X	đ	ß	>]	₽	A 9B	B>	樹	1
Data	1		^		-	Та	g URI			

The following dialog shows which importer type can be selected.

HMIStudio					×
Multiple tag imp	orters are availab	le for this protocol.	. Please select	the importer type	e and continue.
Version	Туре				
RSLogix500 v1.0	Linear				
Tag Editor exported xn	nl General				
				ОК	Cancel

Importer	Description								
RSLogix500 v1.0 Linear	Requires an .csv file. All variables will be displayed at the same level.								
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.								
	Tags × + - > © > 1 > to to by the set of th								

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags × Protocols					
+ - 👗 🕲 🖄 🔰) 🚯 🕅	R Search Tilter by: Data	▼ Ite	ms used:6/10000 Protocol: Show all	Show all tags 🔅 🗖
Data	Туре	Comment	^	Property	Value
Modbus TCP:prot1	Container			Y Driver	
Model: Modicon Modbus(1-based)				Model	Modicon Modbus(1-based)
	unsignedShort			Protocol	Modbus TCP:prot1
	unsignedShort unsignedShort			✓ Dictionary	
	unsignedShort			Array	false
- MRTU2	unsignedShort			Array size	0
MRTU3	unsignedShort			Arravindex.Subindex	400003
	unsignedShort			Comment	100000
- MRTU5	unsignedShort			Data type	unsignedShort
	1-1-1			Data type	unaigneuarion

Toolbar item	Description
R	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
督	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.
	Example of both checked and unchecked result:
	Tage* Image* + - Image* - Image* Image* - Image
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Logical I/O addressing

When addressing Allen Bradley I/O data, the panel uses logical addressing rather than physical addressing. While physical addressing refers to the element number as the slot number, logical addressing refers to the first element for the first I/O card of a specific file type.

xAscender Protocols addressing depends on the mapping of the PLC CPU memory and not on the slot number, therefore you should be careful when changing the configuration in order to avoid remapping.

Use the RSLogix 500 I/O Configuration tool layout of the PLC I/O to configure I/O as in the example.

		Current Cards A	Vailable Filter All IO 💌
PowerSupply # Part # 0 Bul.1766 1 762-1016 2 1762-1016 3 1762-1016 3 1762-104 4 1762-1F4 5 1762-0F4 6 7	Read 10 Config.	Part # 1762-IA8 1762-IF20F2 1762-IF4 1762-IQ8 1762-IQ80W6 1762-IQ16 1762-0A8 1762-0B8 1762-0B16 1762-0W18 1762-0W16 1762-1F4 1762-0F4	16-Input 10/30 VDC 8-Output 120/240 VAC 8-Output (TRANS-SRC) 10/50 VDC 16-Output (TRANS-SRC) 10/50 VDC 8-Output (RLY) 240 VAC 4-Channel Thermocouple Input Module 4-Channel RTD/Resistance Input Module
Adv Config	Help Hide All Cards	1762-074 1762-0X6I	4-Channel Analog I/V Output Module 6-Ch High Current Isolated Relay Outputs Other Requires I/O Card Type ID

Note: When using a module with a configurable I/O size (for example, Devicenet Scanner) make sure you configure it to the largest possible size or you will have to remap it if you need to allocate more space.

Use the Data File Browser to see how the PLC allocates memory.

1

This example shows how to configure the xAscender Protocols Tag for pointing to PLC resource O:1/19 (O1:1.1/3 in word terms).

Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		WORD 0
1:0.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A <	
1:0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:1.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IQ16 - 16-Input 10/30 VDC	
1:2.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IQ16 - 16-Input 10/30 VDC	
1:4.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
1:4.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
I:4.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
I:4.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
I:4.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
I:4.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
1:4.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
1:5.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-OF4 - 4-Channel Analog I/V Output Module	WORD 18
I:5.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-OF4 - 4-Channel Analog I/V Output Module 🔫	in one no
•																	<u>▶</u> -	
Ŀ	:0/0																Radix: Binary	
Symbol:																	Columns: 16 💌	
Desc:																		

The following figure shows the xAscender Protocols Tag configuration.

A-B ENET	X
A-B ENET	
File Type Disc Out ▼	Element Subindex
File Num	Data Type Arraysize boolean Image: Constraint of the second s
Conversion	Sub Element +/- 0
	OK Cancel Apply Help

The xAscender Protocols Tag configured in the example above points on the element shown in the following figure.

	Data Fil	le O	0 (I	bin)		ou	ITPU	s	ubi	nde	x		_				٥	X
	Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
1	0:1.0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	^
	0:1.1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	
	0:1.2 🥄	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	Ξ
	0:1.3	ele	eme	ent	0	0	0	0	0	0	0	0	0	0	0	1	1	
I	0:1.4	Cic	-		0	0	0	0	0	0	0	0	0	0	1	0	0	
I	0:1.5	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	
I	0:1.6	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	
l	0:1.7	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	
	0:1.8	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
I	0:1.9	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	
۱	0:1.10	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	~
	•																•	Ð
	0:1	/19											Rad	B	inarj	,		•
ł	Symbol:		_												Colu	mns:	16	Y
1	Desc:																	
1			Prop	pertie	s		<u>U</u> :	age	•			Eorc	es			Hel	p	

Examples

I:0/19 (I1:0.1/3 in word terms) – 20^{th} Input on CPU

Parameter	Setting
File Type	Disc In
File Num	1

Parameter	Setting
Data Type	Boolean

In the Data File Browser, word 0.1 is Word 1:

Element	1
Sub Index	3

I:1/15 (I1:1.0/15 in word terms) - Last Input on Slot 1 Input Card

Parameter	Setting
File Type	Disc In
File Num	1
Data Type	Boolean

In the Data File Browser, word 1.0 is Word 8:

Element	8
Sub Index	15

I:4.0 (I1:4.0 in word terms) - First Analog Input

Parameter	Setting
File Type	Disc In
File Num	1
Data Type	Short

In the Data File Browser, word 4.0 is Word 10:

Element	10
Sub Index	-

A-B DH-485

The A-B DH-485 communication driver has been designed to connect HMI devices to a Allen-Bradley controllers through serial communication.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the **PLC** list.

The protocol configuration dialog is displayed.

A-B DH-485		X
PLC Network	Comn	n OK
Panel ID	2	Cancel
Slave ID	1	
MaxID	2	
PLC Models		
SLC500 Fixed I/O SLC500 Modular I/O Micrologix 1000 Micrologix 1500		

Element	Description	
Panel ID	Serial node associated to the HMI.	
Slave ID	Serial node associated to the PLC.	
MaxID	Represent the maximum ID available in the serial network.	
PLC Models	PLC models available:	
	SLC500 Fixed I/OSLC500 Modular I/O	

Element	Description		
	Micrologix 100	00	
	Micrologix 1500		
Comm	If clicked displays the communication parameters setup dialog.		
	Comm Parameter Dia	log	
		ОК	
	Port	com1 👻	
	Baudrate	19200 -	
	Parity	even 🔻	
	Data bits	8	
	Stop bits	1	
	Mode	RS-485 -	
	Element	Parameter	
	Port	Serial port selection.	
		COM1: device PLC port.	
		 COM2: computer/printer port on panels with 2 serial ports or optional Plug-In module plugged on Slot 1/2 for panels with 1 serial port on-board. 	
		 COM3: optional Plug-In module plugged on Slot 3/4 for panels with 1 serial port on-board. 	
	Baudrate, Parity, Data Bits, Stop bits	Serial line parameters.	
	Mode	Serial port mode. Available modes:	
		• RS-232.	
		• RS-485 (2 wires).	
		• RS-422 (4 wires).	
PLC Networ k	IP address for all contended of the second contended o	trollers in multiple connections. PLC Network must be selected to ections.	

Element	Description		
	A-B DH-485		
	PLC Network	Comm OK	
	Panel ID	2 Cancel	
	Slave ID	-1	
	MaxID	A-B DH-485	×
	PLC Models SLC500 Fixed 1/O SLC500 Modular I/O Micrologix 1000 Micrologix 1500	OK Slave ID 1 PLC Models	
	Slaves Slave Id	SLC500 Fixed I/O SLC500 Modular I/O Micrologix 1000 Micrologix 1500	

Tag Editor Settings

In Tag Editor select the protocol **A-B DH-485**.

Add a tag using [+] button. Tag setting can be defined using the following dialog:

A-B DH-485	×
File Type Disc In	Element Subindex ● ● ●
File Num	Data Type Arraysize short 0
Conversion	Sub Element
	OK Cancel Apply Help

Element	Description	
Memory Type	Memory Type	Description
i ypc	Disc Out	Discrete output value. O resource on PLC.
	Disc In	Discrete input value. I resource on PLC.
	Status	Status value. S resource on PLC.
	Bit	Bit value. B resource on PLC.
	Timer	Timer value. T resource on PLC.
	Counter	Counter value. C resource on PLC.
	Control	Control value. R resource on PLC.
	Integer	Integer value. N resource on PLC.
	Float	Float value. F resource on PLC.
	String	String value. STR resource on PLC.
Element	Represents the line of the resource while monitoring PLC values.	
Subindex	Represents the column of the resource while monitoring PLC values.	
File Num	Instance of resource of the PLC.	

Element	Description		
Data Type	Available data types:		
	• boolean		
	• byte		
	• short		
	• int		
	unsignedByteunsignedShort		
	unsigned int		
	• float		
	• double		
	• string		
	• binary		
	See "Programming concepts" section in the main manual.		
	Note: To define arrays, select one of Data Type format followed by square brackets (byte[], short[]).		
Arraysize	In case of array tag, this property represents the number of array elements.		
	 In case of string tag, this property represents the maximum number of bytes available in the string tag. 		
	Note: number of bytes corresponds to number of string characters if Encoding property is set to UTF-8 or Latin1 in Tag Editor.		
	If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2 bytes.		

Element	Description		
Sub Element	Allows to point to specific part of a resource:		
	 0 (entire PRE ACC LEN POS 	resource)	
Conversion	Conversion to be applied to the tag.		
	Conversion		
	inv,swap2	Allowed Configured BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits Cancel OK	
	Depending on d	lata type selected, the list Allowed shows one or more conversion types.	
	Value	Description	
	Inv bits	inv : Invert all the bits of the tag.	
		Example: $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)	
	Negate	neg: Set the opposite of tag value.	
		<i>Example:</i> 25.36 → -25.36	
	AB -> BA	swapnibbles : Swap nibbles in a byte. <i>Example:</i> $15D4 \rightarrow 514D$ (in hexadecimal format) $5588 \rightarrow 20813$ (in decimal format)	
	ABCD -> CDAB	swap2 : Swap bytes in a word. <i>Example:</i> $9ACC \rightarrow CC9A$ (in hexadecimal format) $39628 \rightarrow 52378$ (in decimal format)	
	ABCDEFGH	swap4 : Swap bytes in a double word.	

Element	Description		
	Value	Description	
	-> GHEFCDAB	<i>Example:</i> 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)	
	ABCNOP -	swap8: Swap bytes in a long word.	
	> OPMDAB	Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) $0\ 10000000110$ $0001\ 1100101110110100001011010000111001010101$	
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9)	
		Example: $23 \rightarrow 17$ (in decimal format) $0001 \ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)	
	Select conversi	on and click +. The selected item will be added to list Configured .	
	If more conversions are configured, they will be applied in order (from top to bottom Configured). Use the arrow buttons to order the configured conversions.		

Tag Import

Exporting Tags from PLC

The A-B DF1 tag import filter accepts symbol files with extension ".csv" created by the Rockwell RSLogix 500.

To create the file select Tool > Database > ASCII Export

RSLogix 500 Pro - UNTITLED											
⁻ ile Edit View Search Comms	Tools Window Help										
D 🖻 🖬 🎒 🐰 🖻 🖻	Options	💽 🖌 🖓 🖓 🛃 🛃									
OFFLINE 🛓 No Forces	Delete Unused Memory	∃E∃∕E <> <>> <0> ABL ABS									
No Edits 🛃 Forces Disable	Database 🕨 🕨	ASCII Export									
Driver: AB_ETHIP-1	Security 🕨 🕨	ASCII Import									
UNTITLED	Visual Basic Compare FactoryTalk Diagnostics Add-In Manager	Native Import Delete DB Delete Unused Addr. Edit Using Excel Edit Device Codes									
V Processor Status W IO Configuration ₩ Channel Configuratio	n	Adjust Rung Offset Convert Rung Attachment									
 Multipoint Monitor □ □ Program Files □ SYS 0 - □ SYS 1 		Rebuild DB									

From **CSV** tab select the data to be exported and give a name to the output csv file.

Documentation Database ASCII Export	×								
RSLogix 500 A.I. AB APS CSV									
Data to be exported: Destination file names and extensions:									
Addr/Symbol Desc. 🔽 file1 .CSV									
Instruction Comments 🔲 file2 .CSV	-								
Page Title / Rung Desc. 🗖									
Symbol Groups 🗖 file3 .CSV									
Program File Names 🔽 UNTITLED									
AI/AB Address and Instruction description formatting :									
Characters per line in target database 🗾 🚽 20									
Treat Source Description as 5 lines (truncating each line if necessary)									
C Treat Source Description as 1 line (truncating from the end if necessary)									
OK Cancel He	elp								

Importing Tags in Tag Editor

Select the driver in Tag Editor and click on the Import Tags button to start the importer.

	Tags	×								
+	—	X	D	ß	>]	₽	A 9B	B>	ŧ i ł	1
Data	1		^		-	Та	g URI			_

The following dialog shows which importer type can be selected.

MIStudio				
Multiple tag impo	rters are available for this p	rotocol. Please selec	ct the importer type and	d continue.
Version	Туре			
RSLogix500 v 1.0	Linear			
Tag Editor exported xm	General			
			ОК	Cancel

Importer	Description												
RSLogix500 v1.0	Requires an .csv file.												
Linear	All variables will be displayed at the same level.												
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.												
	Tags x												
	🕂 — 🎽 🕲 🖉 🚺 🚺 🕹 🖬												
	Data Tag URI												

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags × Protocols						•		
+ - 👗 🛍 🖄 🔰	D 🗞 🕅 🕅	R 🔎 - Search 🍸 Filter by: Data	▼ 1	Item.	as used:6/10000 Protocol: Show all	Show all tags 💮 🗖		
Data	Туре	Comment	^	•	Property	Value		
Modbus TCP:prot1	, Container				✓ Driver			
Model: Modicon Modbus(1-bas	ea)				Model	Modicon Modbus(1-based)		
Holding Registers 1	unsignedShort				Protocol	Modbus TCP:prot1		
Holding Registers 2	unsignedShort				 Dictionary 			
 Holding Registers 3 MRTU1 	unsignedShort unsignedShort				Array	false		
- MRTU2	unsignedShort				Array size	0		
MRTU3	unsignedShort				Arrayindex.Subindex	400003		
- MRTU4	unsignedShort				Comment	100000		
- MRTU5	unsignedShort			11	Data type	unsignedShort		
	1			00 L.	Data type	unaigneuarion		

Toolbar item	Description
ka	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
「「「」	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.
	Example of both checked and unchecked result:
	Tops: Image: Image: </th
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Logical I/O addressing

When addressing Allen Bradley I/O data, the panel uses logical addressing rather than physical addressing. While physical addressing refers to the element number as the slot number, logical addressing refers to the first element for the first I/O card of a specific file type.

xAscender Protocols addressing depends on the mapping of the PLC CPU memory and not on the slot number, therefore you should be careful when changing the configuration in order to avoid remapping.

Use the RSLogix 500 I/O Configuration tool layout of the PLC I/O to configure I/O as in the example.

		Current Cards A	Vailable Filter All IO
EowerSupply # Part # 0 Bul.1766 1 1762-IQ16 2 1762-IQ16 3 1762-IQ16 3 1762-IV16 4 1762-IF4 5 1762-0F4 6	Read IO Cogfig. Description MicroLogix 1400 Series A 16-Input 10/30 VDC 16-Input 10/30 VDC 16-Output (RLY) 240 VAC Analog 4 Chan. Input 4-Channel Analog I/V Output Module	Part # 1762-IA8 1762-IF20F2 1762-IF4 1762-IQ8 1762-IQ8 1762-Q88 1762-Q88 1762-Q88 1762-Q816 1762-QW16 1762-QW16 1762-QW16 1762-IT4 1762-IF4	Description 8-Input 79/132 VAC Analog 2 Chan. Input, 2 Chan. Output Analog 4 Chan. Input 8-Input 10/30 VDC
(1762-0F4	4-Channel Analog I/V Output Module
Adv Config	Help Hide All Cards	1762-0×61	6-Ch High Current Isolated Relay Outputs Other Requires I/O Card Type ID

Note: When using a module with a configurable I/O size (for example, Devicenet Scanner) make sure you configure it to the largest possible size or you will have to remap it if you need to allocate more space.

Use the Data File Browser to see how the PLC allocates memory.

This example shows how to configure the xAscender Protocols Tag for pointing to PLC resource O:1/19 (O1:1.1/3 in word terms).

Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		WORD 0
I:0.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A \prec	
I:0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:1.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IQ16 - 16-Input 10/30 VDC	
1:2.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IQ16 - 16-Input 10/30 VDC	
1:4.0	0	0	0	0	0	0	٥	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
1:4.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
I:4.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
I:4.3	0	0	0	0	0	0	٥	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
I:4.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
I:4.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
I:4.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
1:5.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-OF4 - 4-Channel Analog I/V Output Module	WORD 18
1:5.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-OF4 - 4-Channel Analog I/V Output Module 🔫	
•																	<u>ا</u> -	
	:0/0	_	_	_	_	_		_		_	_		_	_			Radix: Binary	
Symbol:				_								_		_	_		Columns: 16 💌	

The following figure shows the xAscender Protocols Tag configuration.

i

-B ENET	Door Annua Hannat	X
A-B ENET		
File Type	Element Subindex	
Disc Out	1 3 -	
File Num	Data Type Arraysize	
0	boolean	
Conversion	Sub Element	
	+/- 0 -	
	OK Cancel Apply He	lp

The xAscender Protocols Tag configured in the example above points on the element shown in the following figure.

Data F	ile O	0 (I	bin)		OL	JTPL	J s	ubi	nde	x		_					X
Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0:1.0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	^
0:1.1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	
0:1.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	
0:1.3		eme	ant	0	0	0	0	0	0	0	0	0	0	0	1	1	
0:1.4	ele		eni	0	0	0	0	0	0	0	0	0	0	1	0	0	-
0:1.5	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	
0:1.6	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	
0:1.7	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	
0:1.8	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
0:1.9	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	
0:1.10	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	~
																•	Ē
 [0:	1/19	_									٦	Radi	B	inary	,	-	•
Symbol:																	
Desc:		_	_	_	_		_		_	_	_	-	_	_		,	-
		Prop	pertie	es		Ŭ	sage	9			Eorc	es			Hel	p	

Examples

I:0/19 (I1:0.1/3 in word terms) -20^{th} Input on CPU

Parameter	Setting
File Type	Disc In
File Num	1

Parameter	Setting
Data Type	Boolean

In the Data File Browser, word 0.1 is Word 1:

Element	1
Sub Index	3

I:1/15 (I1:1.0/15 in word terms) - Last Input on Slot 1 Input Card

Parameter	Setting
File Type	Disc In
File Num	1
Data Type	Boolean

In the Data File Browser, word 1.0 is Word 8:

Element	8
Sub Index	15

I:4.0 (I1:4.0 in word terms) - First Analog Input

Parameter	Setting
File Type	Disc In
File Num	1
Data Type	Short

In the Data File Browser, word 4.0 is Word 10:

Element	10
Sub Index	-

A-B ENET

The A-B ENET communication protocol is normally used on the Allen-Bradley controllers via Ethernet communication.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the **PLC** list.

The protocol configuration dialog is displayed.

A-B ENET		×
PLC Network		ОК
Alias	PLC1	Cancel
IP address	192 . 168 . 10 . 5	
Port	44818	
Timeout	1000	
PLC Models		
PLC5 via NET-ENI SLC500 via NET-ENI Micrologix 1000/1200, Micrologix 1100/1400	/1500 via NET-ENI	

Element	Description
Alias	Name identifying nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node.
IP Address	Ethernet IP address of the controller.
Port	Port number used by the Ethernet interface.

Element	Description			
Timeout	Time delay in milliseconds between two retries in case of missing response from the controller.			
PLC Network	Enable access to multiple networked controllers. For every controller (slave) set the proper option.			
	OK Alias PLC1 IP address 192.168.10.1 Port 44818 Timeout 1000 PLC Models PLC Svia NET-ENI Micrologix 1000/1200/1500 via NET-ENI Micrologix 1100/1400			

Controller configuration

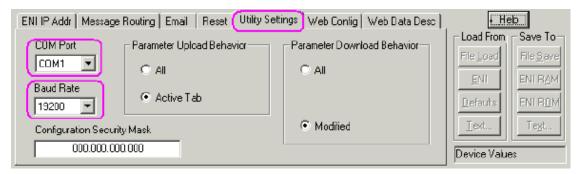
The PLC has to be correctly configured to match the IP address configured in the Protocol Editor. Normally the PLC configuration can be left as default.

Channel Configuration	X
General Chan. 1 - System Chan. 0 - System Chan. 0 - Us	ser
Driver Ethernet	
Broadcast Address: 0.0.00	DHRIO Link ID 0
Hardware Address: 00:00:BC:1D:D1:FC IP Address: 192.168.0.140	Pass Thru Routing
Subnet Mask: 255.255.255.0	
Gateway Address: 192.168.0.199	
	sg Connection Timeout (x 1mS): 15000
	Msg Reply Timeout (x 1mS): 3000
	Inactivity Timeout (x Min): 30
Contact:	
Location:	
OK Ca	ancel Apply Help

Configuring 1761-NET-ENI

Here is the procedure to configure the 1761-NET-ENI module using the Allen Bradley's ENI/ENIW Utility. The procedure requires a 1761-CBL-PM02 communication cable.

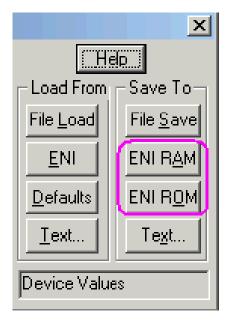
- 1. Connect the 8 pin din to the port 2 on the NET-ENI device and the 9 pin female D-shell to the computer COM port.
- 2. Connect the SLC 5/0x controller and go online.
- 3. In the Utility Settings tab, set COM Port and Baud Rate.



4. In the ENI IP Addr tab, select the correct ENI Series from the list and set ENI IP Address, Subnet Mask and Baud Rate, if needed.

ENI IP Add Message Routin	g Email Reset	Utility Setting: Web Config Web Data Desc	<u>H</u> e	lp
ENI Serie: D	232 Baud Rate	Auto 🔽 CompactLogix Routing 🗔	Load From	-Save To-
Obtain via BootP			File Load	File <u>S</u> ave
Always	ENI IP Address	003.058.137.092	<u>E</u> NI	ENI R <u>A</u> M
Falback 🗖	Subnet Mask	255.255.252.000	Delaults	ENI R <u>O</u> M
Dbtain via DHCP	Gateway	000.000.000	Text	Te <u>x</u> t
Ethernet Speed/Duplex	Security Mask 1	000.000.000.000		
Auto Negotiate 📃	Security Mask 2	000.000.000.000	Device Value	s

5. Save the configuration to the NET-ENI device.



Two separate memory areas are reserved for saving the configuration : **ENI/RAM** (for temporary configurations) and **ENI/ROM** (for permanent configurations).

Logical I/O addressing

When addressing Allen Bradley I/O data, the panel uses logical addressing rather than physical addressing. While physical addressing refers to the element number as the slot number, logical addressing refers to the first element for the first I/O card of a specific file type.

xAscender Protocols addressing depends on the mapping of the PLC CPU memory and not on the slot number, therefore you should be careful when changing the configuration in order to avoid remapping.

Use the RSLogix 500 I/O Configuration tool layout of the PLC I/O to configure I/O as in the example.

		Current Cards A	Vailable Filter All IO 💌
PowerSupply # Part # 0 Bul.1766 1 762-1016 2 1762-1016 3 1762-1016 3 1762-104 4 1762-1F4 5 1762-0F4 6 7	Read 10 Config.	Part # 1762-IA8 1762-IF20F2 1762-IF4 1762-IQ8 1762-IQ80W6 1762-IQ16 1762-0A8 1762-0B8 1762-0B16 1762-0W18 1762-0W16 1762-1F4 1762-0F4	16-Input 10/30 VDC 8-Output 120/240 VAC 8-Output (TRANS-SRC) 10/50 VDC 16-Output (TRANS-SRC) 10/50 VDC 8-Output (RLY) 240 VAC 4-Channel Thermocouple Input Module 4-Channel RTD/Resistance Input Module
Adv Config	Help Hide All Cards	1762-074 1762-0X6I	4-Channel Analog I/V Output Module 6-Ch High Current Isolated Relay Outputs Other Requires I/O Card Type ID

Note: When using a module with a configurable I/O size (for example, Devicenet Scanner) make sure you configure it to the largest possible size or you will have to remap it if you need to allocate more space.

Use the Data File Browser to see how the PLC allocates memory.

1

This example shows how to configure the xAscender Protocols Tag for pointing to PLC resource O:1/19 (O1:1.1/3 in word terms).

Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		WORD 0
I:0.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A \prec	
I:0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
1:0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1766 MicroLogix 1400 Series A	
I:1.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IQ16 - 16-Input 10/30 VDC	
1:2.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IQ16 - 16-Input 10/30 VDC	
1:4.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
I:4.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
I:4.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
I:4.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
1:4.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
I:4.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
1:4.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-IF4 - Analog 4 Chan. Input	
1:5.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-OF4 - 4-Channel Analog I/V Output Module	WORD 18
1:5.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1762-OF4 - 4-Channel Analog I/V Output Module 🔫	WORD TO
•																	▶-	
1:	:0/0	_	_		_		_		_		_	_	_	_	_		Radix: Binary	
Symbol:		_	_		_		_		_		_	_	_	_	_		Columns: 16 💌	
Desc:																		

The following figure shows the xAscender Protocols Tag configuration.

A-B ENET	X
A-B ENET	
File Type Disc Out ▼	Element Subindex
File Num	Data Type Arraysize boolean
Conversion	Sub Element +/- 0
	OK Cancel Apply Help

The xAscender Protocols Tag configured in the example above points on the element shown in the following figure.

	Data Fi	le O	0 (I	bin)		ou	ITPU	s	ubi	nde	x		_				٥	X
	Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
1	0:1.0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	^
	0:1.1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	
1	0:1.2 🥄	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	Ξ
	0:1.3	ele	eme	ent	0	0	0	0	0	0	0	0	0	0	0	1	1	
I	0:1.4	Cic	-		0	0	0	0	0	0	0	0	0	0	1	0	0	
I	0:1.5	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	
I	0:1.6	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	
l	0:1.7	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	
	0:1.8	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
I	0:1.9	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	
I	0:1.10	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	~
	•																•	Ð
	0:1	/19		_	_	_		_	_	_	_		Rad	ix B	inarj	,		•
ł	Symbol:		_												Colu	mns:	16	Y
1	Desc:																	
1			Prop	pertie	es		U	sage	,			Eorc	es			Hel	p	

Examples

I:0/19 (I1:0.1/3 in word terms) – 20^{th} Input on CPU

Parameter	Setting
File Type	Disc In
File Num	1

Parameter	Setting
Data Type	Boolean

In the Data File Browser, word 0.1 is Word 1:

Element	1
Sub Index	3

I:1/15 (I1:1.0/15 in word terms) - Last Input on Slot 1 Input Card

Parameter	Setting
File Type	Disc In
File Num	1
Data Type	Boolean

In the Data File Browser, word 1.0 is Word 8:

Element	8
Sub Index	15

I:4.0 (I1:4.0 in word terms) - First Analog Input

Parameter	Setting
File Type	Disc In
File Num	1
Data Type	Short

In the Data File Browser, word 4.0 is Word 10:

Element	10
Sub Index	-

Node Override IP

The protocol provides the special data type Node Override IP which allows you to change the IP address of the target controller at runtime.

This memory type is an array of 4 unsigned bytes, one per each byte of the IP address.

The Node Override IP is initialized with the value of the controller IP specified in the project at programming time.

Node Override IP	PLC operation
0.0.0.0	Communication with the controller is stopped, no request frames are generated anymore.
Different from 0.0.0.0	It is interpreted as node IP override and the target IP address is replaced runtime with the new value.

If the HMI device is connected to a network with more than one controller node, each node has its own Node Override IP variable.



Note: Node Override IP values assigned at runtime are retained through power cycles.

Hostname DNS or mDNS

In addition to the array of bytes, string memory type can be selected to be able use the DNS or mDNS hostname as an alternative to the IP Address.

A-B ENET		×
A-B ENET		
File Type	Element Subindex	
Node Override IP 👻	0 🔹 0 👻	
File Num	Data Type	Arraysize
7	unsignedByte [] 🛛 👻	4
Conversion	Sub Element	
	+/-	*
	ОК	Cancel Apply Help

Tag Import

Exporting Tags from PLC

The A-B Ethernet tag import filter accepts symbol files with extension ".csv" created by the Rockwell RSLogix 500.

To create the file select Tool > Database > ASCII Export

RSLogix 500 Pro - UNTITLED		
⁻ ile Edit View Search Comms	Tools Window Help	
D 🖻 🖬 🎒 🐰 🖻 🖻	Options	💽 🖌 🖓 🖓 🛃 🛃
OFFLINE 🛓 No Forces	Delete Unused Memory	∃E∃∕E <> <>> <0> ABL ABS
No Edits 🛃 Forces Disable	Database 🕨 🕨	ASCII Export
Driver: AB_ETHIP-1	Security 🕨 🕨	ASCII Import
UNTITLED	Visual Basic Compare FactoryTalk Diagnostics Add-In Manager	Native Import Delete DB Delete Unused Addr. Edit Using Excel Edit Device Codes
V Processor Status W IO Configuration ₩ Channel Configuratio	n	Adjust Rung Offset Convert Rung Attachment
 Multipoint Monitor □ □ Program Files □ SYS 0 - □ SYS 1 		Rebuild DB

From **CSV** tab select the data to be exported and give a name to the output csv file.

Documentation Database ASCI	Documentation Database ASCII Export				
RSLogix 500 A.I. AB APS	sv				
Data to be exported :	Destination file names and extensions :—				
Addr/Symbol Desc. 🔽	file1 .CSV	-			
Instruction Comments	file2 .CSV	-			
Page Title / Rung Desc. 🛛 🗖					
Symbol Groups 📃	file3 .CSV				
Program File Names 🛛 🗖	UNTITLED				
- AI/AB Address and Instruction de	scription formatting :				
Characters per line in target data	abase 🔽 20				
Treat Source Description as 5 li	ines (truncating each line if necessary)				
C Treat Source Description as 1 li	ine (truncating from the end if necessary)				
ОК	Cancel He	elp			
C Treat Source Description as 1 li	ine (truncating from the end if necessary)				

Importing Tags in Tag Editor

Select the driver in Tag Editor and click on the Import Tags button to start the importer.

	Tags	×								
+	_	Z	D	ß	>]	₽	A 9B	B>	ŧ3	1
Data	1		^		-	Та	g URI			_

The following dialog shows which importer type can be selected.

HMIStudio	×				
Multiple tag impo	orters are available for this protocol. Please select the importer type and continue.				
Version	Туре				
RSLogix500 v 1.0	Linear				
Tag Editor exported xm	Tag Editor exported xml General				
	OK Cancel				

Importer	Description		
RSLogix500 v1.0 Linear	Requires an .csv file. All variables will be displayed at the same level.		
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.		
	Tags × + → ▲		

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags 🗙 Protocols						•
+ - 👗 🛍 🗈 🔰	D 🗞 🖬 🕅	R O- Search Tilter by: Data	▼ 1	tem.	as used:6/10000 Protocol: Show all	🗸 🗹 Show all tags 🖉 🗖
Data	Туре	Comment	^	-	Property	Value
Modbus TCP:prot1	Container				✓ Driver	
Model: Modicon Modbus(1-based)			16	Model	Modicon Modbus(1-based)
- Holding Registers 1	unsignedShort				Protocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort				 Dictionary 	
- Holding Registers 3 - MRTU1	unsignedShort unsignedShort			11	Array	false
- MRTU2	unsignedShort				Array size	0
MRT02	unsignedShort		_	11	Arrayindex.Subindex	400003
- MRTU4	unsignedShort				Comment	100003
- MRTU5	unsignedShort			ar.		unsignedShort
	and any second for the		_	a.	Data type	unsignedShort

Toolbar item	Description		
ka	Import Tag(s).		
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project		
「「「」	Update Tag(s).		
	Click on this icon to update the tags in the project, due a new dictionary import.		
R	Check this box to import all sub-elements of a tag.		
	Example of both checked and unchecked result:		
	Tops: Image: Image: </th		
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.		

Communication status

Current communication status can be displayed using system variables. See "System Variables" section in the main manual.

Error	Cause	Action
NAK	The controller replies with a not acknowledge.	-
Timeout	A request is not replied within the specified timeout period.	Check if the controller is connected and properly configured to get network access.
Invalid response	The device did received a response with invalid format or contents from the controller.	Check if the data programmed in the project are consistent with the controller resources.
General Error	Unidentifiable error. Should never be reported.	Contact technical support.

Codes supported for this communication driver:

ABB Mint Controller HCP

This communication protocol allows the HMI devices to connect to the ABB motion and servo drive devices using the HCP and HCP2 communication protocols.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the **PLC** list.

The protocol configuration dialog is displayed.

		×
PLC Network	Comm	ОК
Node ID	1	ancel
Protocol Type	HCP2	
Timeout (ms)	200	
Retry count	2	
PLC Models		
ABB Mint Controller		

Element	Description
Node ID	Node ID assigned to the controller device.
Protocol Type	Two protocols are available: HCP HCP2
Timeout (ms)	Time delay in milliseconds between two retries in case of missing response from the server device.
Retry count	Number of times a certain message will be sent to the controller before reporting the communication error status.
PLC Models	PLC model you are going to connect to.

Element	Description
PLC Network	The protocol allows the connection of multiple controllers to one HMI device. To set-up multiple connections, check "PLC network" checkbox and enter the node ID per each slave you need to access.
	Image: Stave Id Model Slaves Add Slave Id Model
Comm	If clicked displays the communication parameters setup dialog.
	Comm Parameter Dialog Port Port Baudrate 9600 Parity even Data bits 8 Stop bits 1 Mode RS-485

Element	Description		
	Element	Description	
	Port	Serial port selection.	
		COM1= device PLC port.	
		COM2= computer/printer port.	
	Baudrate, Parity, Data Bits, Stop bits	Serial line parameters.	
	Mode	Serial port mode. Available modes:	
		• RS-232.	
		• RS-485 (2 wires).	
		• RS-422 (4 wires).	

Data types

The ABB Mint Controller HCP driver provides the support for two Memory Types which are referring to the same physical memory area in the Mint controller:

- **Comms**: should only be used with floating point values. The Mint program on the ABB controller should use COMMS to access this data.
- CommsInteger: allows a variety of integer-based data types to be selected.

If the Mind controller program uses	then
COMMS keyword for a tag setup to use the Commsinteger memory type	only the bottom 23 bits will be accurate (due to floating point precision of the COMMS keyword).
COMMSINTEGER keyword for a tag setup to use the Commsinteger memory type	the value is precise for the full 32 bits.

ABB Mint Controller HCP	
ABB Mint Controller HCP	
Memory Type CommsInteger -	Offset SubIndex
Data Type int boolean byte short int unsignedByte unsignedShort	Arraysize Conversion
unsignedInt string	OK Cancel Apply Help

See "Programming concepts" section in the main manual.

Tag Conversion

Conversion to be applied to the tag.

Conversion

Conversion			
inv,swap2	Allowed		Configured
	BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits	+ - ~ ~	Inv bits ABCD->CDAB
		(Cancel OK

Depending on data type selected, the list **Allowed** shows one or more conversion types.

Value	Description
Inv bits	inv : Invert all the bits of the tag.
	<i>Example:</i> 1001 \rightarrow 0110 (in binary format) 9 \rightarrow 6 (in decimal format)
Negate	neg : Set the opposite of tag value.
	Example:

Value	Description
	$25.36 \rightarrow \textbf{-}25.36$
AB -> BA	swapnibbles: Swap nibbles in a byte.
	<i>Example:</i> 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)
ABCD -> CDAB	swap2: Swap bytes in a word.
	<i>Example:</i> 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)
ABCDEFGH -> GHEFCDAB	swap4 : Swap bytes in a double word. <i>Example:</i> $32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format) $855441236 \rightarrow 1426062386$ (in decimal format)
ABCNOP -> OPMDAB	swap8: Swap bytes in a long word.Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) $0 1000000110$ $0001110010111011010000101101000011100000$
BCD	bcd: Separate byte in two nibbles, read them as decimal (from 0 to 9)Example: $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ $0001 = 1$ (first nibble) $0111 = 7$ (second nibble)

Select conversion and click +. The selected item will be added to list **Configured**.

If more conversions are configured, they will be applied in order (from top to bottom of list Configured).

Use the arrow buttons to order the configured conversions.

Tag Import

Select the driver in Tag Editor and click on the **Import Tags** button to start the importer.

	Tags	×								
+	_	X	đ	ß	>]	₽	₿ ₿	۵X	ŧ3	1
Data	1		^		-	Та	g URI			_

Locate the .xml file exported from Tag Editor and click Open.

imp	orter	- Select a file to open		x
Com V Com	puter	Local Disk (C:) Folder1 ✓	Search Folder1	٩
Organize 🔻 New	folde		:== :==	• 🔳 🔞
	*	Name	Date modified	Туре
Desktop	_	Exported Tags.xml	03/06/2016 16:42	XML File
Documents Documents Music Fictures Videos Homegroup User User Volser Computer Network Control Panel	E			
		<		
F	ile na	me: Exported Tags.xml 🔻	Tag Editor exporter	d file (*.xml) ▼ Cancel

Tags included in the symbol file are listed in the tag dictionary. The tag dictionary is displayed at the bottom of the screen.

Tags × Protocols					•
+ - 👗 🕲 🖄 🔰 [3	> \$ ₿ ₪ ∰	R 🔎 - Search YFilter by: Data -	Ite	ems used:6/10000 Protocol: Show all	Show all tags 🔅 🗌
Data	Туре	Comment	^	Property	Value
Modbus TCP:prot1	Container			✓ Driver	
Model: Modicon Modbus(1-based)				Model	Modicon Modbus(1-based)
	unsignedShort			Protocol	Modbus TCP:prot1
	unsignedShort unsignedShort			 Dictionary 	·
	unsignedShort			Array	false
	unsignedShort			Array size	0
MRTU3 U	unsignedShort			Arrayindex.Subindex	400003
- MRTU4	unsignedShort			Comment	
- MRTU5	unsignedShort			Data type	unsignedShort

Toolbar item	Description
ka	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
K 湖	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.

Toolbar item	Description				
	Example of both checked and unchecked result:				
	Tags* ×				
	+-¥©©刈▷\$=>>> +-¥©©刈▷\$=>>>				
	Data Type Con Data Type Con				
	CODESYS V3 ETH:prot2 Container CODESYS V3 ETH:prot2 Container CODESYS V3 ETH:prot2 Container				
	Application Container Application Container				
	PLC_PRG Container A PLC_PRG Container A testArrayTag Array[11]: INT A testArrayTag Array[11]: INT				
	- [1] INT - [2] INT - [2] INT				
	- [2] INT - [3] INT - [3] INT				
	- [6] INT - [6] INT				
	ти [7] Тип [7] Тип				
	TAI [9] - 10				
	[10] INT [10] INT - testTag1 INT				
	testTag1 INT testTag1 INT testTag2 INT				
	testTag3 BYTE testTag3 BYTE				
	Tagi DINT Tagi DINT				
	Tag2 DINT Tag2 DINT				
	Tag3 DINT Tag3 DINT				
P - Search Y Filter by: Tag name ▼	Tag2 DINT Tag2 DINT				

Communication status

Current communication status can be displayed using system variables. See "System Variables" section in the main manual.

Codes supported by this communication driver:

Error	Cause	Action
NAK	The controller replies with a not acknowledge.	-
Timeout	A request is not replied within the specified timeout period.	Check if the controller is connected and properly configured to get network access.
Line Error	An error on the communication parameter setup is detected (parity, baud rate, data bits, stop bits).	Check if the communication parameter settings of the controller is compatible with the device communication setup.
Invalid response	The device did received a response with invalid format or contents from the controller.	Ensure the data programmed in the project are consistent with the controller resources.
General Error	Unidentifiable error. Should never be reported.	Contact technical support.

BACnet

The BACnet communication driver has been designed to connect HMI devices to BACnet networks and supports IP and MS/TP communication.

The HMI device operates as a BACnet device.

Implementation details

This implementation of the BACnet communication protocol allows integrating HMIs in a BACnet network and exchange data between HMI and other devices connected to the BACnet network. HMIs provide client capability for displaying properties of BACnet objects in real time using BACnet/IP or MS/TP network types.

BACnet communication protocol can be:

- Configured as BACnet IP: communication with BACnet devices is established over Ethernet using HMI Ethernet port;
- Configured as BACnet MS/SP: communication with BACnet devices is established over serial line, using HMI serial port;

Communication protocol configuration allows defining HMI BACnet ID and object name used to identify HMI in BACnet network.

BACnet object properties are reachable from HMI using explicit Tag configuration. A single Tag represents a single property for a BACnet object.

Using the property Present_Value (85) in Tag configuration, the Tag will be connected to the current value of a specific object (for example in the case of analog values, it will be the measured value).

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In Config node double-click Protocols.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the PLC list.

The protocol configuration dialog is displayed.

net			×
	Comm		
Panel Device ID	262000	Analog Value Count	0
Object Name	DEV262000	Binary Value Count	0
Description	НМІ	Multi State Value Cour	0
Media	MS/TP 👻	Notification Class Cou	0
Timeout (ms)	5000	IP UDP Port	47808
Panel Node	1	Local IP	
COV Lifetime (s)	60		
COV Confirmed			
Max Master	127		
Max Info Frames	1		
max MS/TP APDU	480		
max IP APDU	1476		
Time Sync Interval (s)	0		
Time Sync UTC			
PLC Models			
default			
			OK Cancel

Element	Description
Panel Device ID	Identifies the HMI device in the network.
Object Name	BACnet Object Name for the HMI device.
Description	HMI device description, for documentation purposes.
Media	Type of communication of the protocol.
	• MS/TP : Master-Slave/Token-Passing communication (RS-485).

Element	Description							
	IP: based on standard UDP/IP communication.							
Timeout (ms)	Time delay in milliseconds between two retries in case of missing response from the BACnet device.							
Panel Node *	MS/TP address. Physical device address on the link; it is not passed through routers.							
COV Lifetime (s)	Desired lifetime of the subscription in seconds before the it shall be automatically cancelled. A value of zero indicates an indefinite lifetime, without automatic cancellation.							
Max Master *	Highest allowable address for master nodes. Must be less than or equal to 127.							
Max Info Frames *	Maximum number of information frames the node may send before it must pass the token. Max Info Frames may have different values on different nodes and may be used to allocate more or less of the available link bandwidth to particular nodes.							
Max MS/TP APDU *	Maximum length of APDU (Application Layer Protocol Data Unit), which means the actual packet length on BACnet network. This value cannot exceed 480 (default value).							
Max IP APDU **	Maximum length of APDU (Application Layer Protocol Data Unit), which means the actual packet length on BACnet network. This value cannot exceed 1476 (default value).							
Time Sync Interval (s)	Represent the interval between every time synchronization, in seconds. If left to 0, time synchronization is disabled.							
Time Sync UTC	Option to synchronize time in UTC format. If disabled, local time format used.							
PLC Models	Reserved for future use.							
Comm *	If clicked displays the communication parameters setup dialog.							
	Comm Parameter Dialog							
	ОК							
	Port com1							
	Baudrate 9600 -							
	Parity none							
	Data bits 8							
	Stop bits							
	Mode RS-485							

Element	Description	
	Element	Description
	Port	Communication port.
	Baudrate, Parity, Data bits, Stop bits	Communication parameters.
	Mode	Communication mode. Available modes:
		• RS-232
		• RS-485
		• RS-422
Analog Value Count ***	Number of Analog Value objects to be insta Min: 0 Max: 200	anced in BACnet Server.
Binary Value Count ***	Number of Binary Value objects to be insta Min: 0 Max: 200	inced in BACnet Server.
Multi State Value Count ***	Number of Multi State Value objects to be i Min: 0 Max: 200	instanced in BACnet Server.
Notification Class Count ***	Number of Notifications Class objects to be Min: 0 Max: 200	e instanced in BACnet Server.
IP UDP Port **	Port number for IP communication.	
Local IP **	IP Address of the network adapter to use for only one Ethernet adapter.	or protocol. Not required if the device has
Note *: A	vailable only if media is set to MS/TP .	



Note **: Available only if media is set to IP.



i

Note ***: Check Using BACnet Server chapter.

Tag Editor Settings

Path: ProjectView> Config > double-click Tags

- 1. To add a tag, click +: a new line is added.
- 2. Select **BACnet** from the **Driver** list: the tag definition dialog is displayed.

Object Type	Device ID	Data Type	
Analog Value 🔻	508	float 👻	
Arraysize Conv	version	Object Instance	
0	+/	3000015	
bject Property	Array Index	Write Priority	
85	-1	0	×
COV			

Elemen t	Description
Object Type	Type of BACnet object to be referenced. Available object types: • Device • Analog Input • Analog Output • Analog Value • Binary Input • Binary Output • Binary Value • Multi-state Input • Multi-state Output • Multi-state Value • Integer Value • Positive Integer Value • Large Analog Value
Device ID	ID of the device containing the object.
Data Type	Data type for display presentation. Available data types: boolean

Elemen t	Description								
	 int unsignedInt float double string binary boolean[] These data types are data types 								
	The equivalence with BACnet data types is shown in the table: BACnet data type Software data type Notes								
	BOOLEAN	Boolean	-						
	INTEGER	Int	-						
	UNSIGNED_INTEGER	unsignedInt	-						
	REAL	Float	-						
	BIT_STRING	boolean-x	x = size						
	CHARACTER_STRING	string-x	x = size						
	OCTET_STRING	binary-x	x = size						
	DATE	int or unsignedInt	-						
	TIME	int or unsignedInt	-						
	BACnetObjectIdentifier	int or unsignedInt	Use conversions instance and objType for proper display						
Arraysi ze	 In case of string tag, the string tag. Note: number of bytes corres UTF-8 or Latin1 in Tag Editor 	nis property repre ponds to numbe	esents the number of array elements. esents the maximum number of bytes available in or of string characters if Encoding property is set to -2LE, UTF-16BE or UTF-16LE one character						
Conver sion	Conversion to be applied to the second secon	he tag.							

lemen	Description					
	Conversion					
	inv,swap2	Allowed BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits Cancel OK				
		ata type selected, the list Allowed shows one or more conversion types.				
	Value	Description				
	Inv bits	inv : Invert all the bits of the tag. <i>Example:</i> $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)				
	Negate	eg: Set the opposite of tag value.				
		<i>Example:</i> 25.36 → -25.36				
	AB -> BA	swapnibbles : Swap nibbles in a byte. <i>Example:</i> 15D4 → 514D (in hexadecimal format) 5588 → 20813 (in decimal format)				
	ABCD ->	swap2: Swap bytes in a word.				
	CDAB	Example: $9ACC \rightarrow CC9A$ (in hexadecimal format) $39628 \rightarrow 52378$ (in decimal format)				
	ABCDEFGH -	swap4: Swap bytes in a double word.				
	> GHEFCDAB	Example: 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)				
	ABCNOP ->	swap8: Swap bytes in a long word.				
	OPMDAB	Example: 142.366 → -893553517.588905 (in decimal format) 0 1000000110				

Elemen t	Description												
	Value	D	escription										
		00	01110010111	101101	10010001011	010000	1110010101100	00001					
		10	→ 10000011100 010101000010100010110110110										
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9)											
	Example: $23 \rightarrow 17$ (in decimal format) $0001 \ 0111 = 23$ $0001 = 1$ (first nibble) $0111 = 7$ (second nibble)												
	Select conve	ersion a	nd click +. The	selecte	ed item will be	added t	o list Configure	ed.					
	If more conv Configured		are configure	d, they v	vill be applied	l in order	(from top to bo	ttom of list					
	Use the arro	w butto	ns to order the	configu	ared conversion	ons.							
Object Instanc e	BACnet ID o	f the ob	ject to be refer	enced.									
Object Propert y	most standa	rd objec					alue 85 means µ	present-va					
	Property	Val ue	Property	Val ue	Propert y	Val ue	Property	Val ue					
	accepted- modes	175	effective- period	32	max- info- frames	63	reason-for- halt	100					
	acked- transitions	0	elapsed- active- time	33	max- master	64	recipient- list	102					
	ack- required	1	error-limit	34	max- pres- value	65	records- since- notification	140					
	action	2	event-	35	max-	167	record-	141					

Elemen t

Description

Property	Val ue	Property	Val ue	Propert y	Val ue	Property	Val ue
		enable		segment s- accepte d		count	
action- text	3	event- state	36	member- of	159	reliability	103
active-text	4	event- time- stamps	130	minimu m-off- time	66	relinquish- default	104
active-vt- sessions	5	event- type	37	minimu m-on- time	67	required	105
active- cov- subscripti ons	152	event- paramet ers	83	minimu m-output	68	resolution	106
adjust- value	176	exceptio n- schedule	38	minimu m-value	136	scale	187
alarm- value	6	fault- values	39	minimu m-value- timesta mp	150	scale- factor	188
alarm- values	7	feedbac k-value	40	min- pres- value	69	schedule- default	174
all	8	file- access- method	41	mode	160	segmentati on- supported	107
all-writes- successfu I	9	file-size	42	model- name	70	setpoint	108
apdu- segment- timeout	10	file-type	43	modifica tion-date	71	setpoint- reference	109

E	lei	m	e
t			

en Description

Property	Val ue	Property	Val ue	Propert y	Val ue	Property	Val ue
apdu- timeout	11	firmware- revision	44	notificati on-class	17	slave- address- binding	171
applicatio n- software- version	12	high-limit	45	notificati on- threshol d	137	setting	162
archive	13	inactive- text	46	notify- type	72	silenced	163
attempte d- samples	124	in- process	47	number- of- APDU- retries	73	start-time	142
auto- slave- discovery	169	input- reference	181	number- of-states	74	state-text	110
average- value	125	instance- of	48	object- identifier	75	status- flags	111
backup- failure- timeout	153	integral- constant	49	object- list	76	stop-time	143
bias	14	integral- constant- units	50	object- name	77	stop-when- full	144
buffer- size	126	last- notify- record	173	object- property- referenc e	78	system- status	112
change- of-state- count	15	last- restore- time	157	object- type	79	time-delay	113
change- of-state- time	16	life- safety- alarm- values	166	operatio n- expecte d	161	time-of- active- time-reset	114

Elemen t

Description

Property	Val ue	Property	Val ue	Propert y	Val ue	Property	Val ue
client-cov- increment	127	limit- enable	52	optional	80	time-of- state- count-reset	115
configurat ion-files	154	limit- monitorin g-interval	182	out-of- service	81	time- synchroniz ation- recipients	116
controlle d- variable- reference	19	list-of- group- members	53	output- units	82	total- record- count	145
controlle d- variable- units	20	list-of- object- property- referenc es	54	polarity	84	tracking- value	164
controlle d- variable- value	21	list-of- session- keys	55	prescale	185	units	117
count	177	local- date	56	present- value	85	update- interval	118
count- before- change	178	local- time	57	priority	86	update- time	189
count- change- time	179	location	58	pulse- rate	186	utc-offset	119
cov- increment	22	log-buffer	131	priority- array	87	valid- samples	146
cov- period	180	log- device- object- property	132	priority- for- writing	88	value- before- change	190
cov- resubscri	128	log- enable	133	process- identifier	89	value-set	191

E	Ele	en	ne	2
t				

en Description

Property	Val ue	Property	Val ue	Propert y	Val ue	Property	Val ue
otion- nterval							
database- revision	155	log- interval	134	profile- name	168	value- change- time	192
date-list	23	logging- object	183	progra m- change	90	variance- value	151
daylight- savings- status	24	logging- record	184	progra m- location	91	vendor- identifier	120
deadband	25	low-limit	59	progra m-state	92	vendor- name	121
derivativ e- constant	26	maintena nce- required	158	proportio nal- constant	93	vt-classes- supported	122
derivativ e- constant- units	27	manipula ted- variable- reference	60	proportio nal- constan t-units	94	weekly- schedule	123
descriptio 1	28	manual- slave- address- binding	170	protocol- object- types- supporte d	96	window- interval	147
descriptio n-of-halt	29	maximu m-output	61	protocol- revision	139	window- samples	148
device- address- binding	30	maximu m-value	135	protocol- services- supporte d	97	zone- members	165

Elemen t	Description	1								
	Property	Val ue	Property	Val ue		Propert y	Val ue	Property	Val ue	
	device- type	31	maximu m-value- timestam p	149		protocol- version	98			
	direct- reading	156	max- apdu- length- accepted	62		read- only	99			
Array Index	• -1 me	 Index for subscribing elements in BACnet arrays. -1 means read all elements 0 to n means read the specified element 								
	To read a pr to refer to the	Priority Array example To read a priority array object it is necessary to set Object Property = 87 and Array Index has to refer to the priority item to be read. The following figure shows how to read the 16th item of a priority array.								

Elemen t	Description	
	BACnet	x
	BACnet	
	Object Type Device ID Data Type	
	Binary Value	
	Arraysize Conversion Object Instance	
	0 +/- 3000108	
	Object Property Array Index Write Priority	
	87 16 0	
	□ cov	
		1
		1
		Help
Write Priority	Write requests priority level. The value is in the range 1-16. 0 is interpreted as 16.	
COV	Enable the Change Of Value notification.	

Clear/Set Priority

The system offers actions for a more flexible handling of Write Priority.

Action	Description	
BACnetClearPriority	Clears the priority array at the position associated to the BACnet tag passed as parameter.	
	This action has immediate effect on the BACnet device.	
BACnetClearAllPriorities	Clears all positions in the priority array.	
	This action has immediate effect on the BACnet device.	
BACnetSetPriority	Overrides the Write Priority value configured in the BACnet tag definition	
	This action has two parameters:	
	TagName: name of the BACnet tag.	
	 TagPriority: new value of Write Priority for the BACnet tag passed as parameter. 	
	This action only overrides the value of Write Priority in the BACnet tag definition and does not perform any communication with the BACnet device. Any write command that will be performed to the Present Value property of the BACnet device identified by the tag, will be performed using the new Write Priority value.	
	The priority value will be valid until:	
	 A new call to the BACnetSetPriority action changes it. 	
	 The HMI device is restarted. The value of WritePriority defined in the project is valid in this case. 	

Tag Import

BACnet object information can be imported from BACnet EDE (Engineering Data Exchange) files. The EDE file must have the .csv extension.

The importer uses the characters "," and ";" as delimiters. They are considered as reserved characters and you cannot use them in file name.

Use the hierarchical importer to have a ordered list of BACnet objects and properties.

Tags will be created using the string specified in the column object-name of the EDE file. The importer will add the device ID as a prefix to avoid duplication of tag names.



Note: The importer will ask to locate the State-Texts, Unit-Texts and Object-Types files. Click Cancel to ignore.

Select the driver in Tag Editor and click on the Import Tags button to start the importer.

	Tags	×								
+	-	X	D	ß	>]	₽	A 9B	B>	ŧ i ł	1
Data			^		-	Та	g URI			_

The following dialog shows which importer type can be selected.

HMIStudio		X
Multiple tag	importers are available for this protocol. Please select the importer type and co	ontinue.
Version	Туре	
BACnet EDE v1.0	Linear	
BACnet EDE v1.0	Hierarchical	
Tag Editor exported	i xml General	
	OK	ncel

Importer	Description		
BACnet EDE v1.0	Requires a .csv file.		
Linear	All variables will be displayed at the same level.		
BACnet EDE v1.0	Requires a .csv file.		
Hierarchical	All variables will be displayed according to BACnet EDE Hierarchical view.		
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.		
	Tags x		
	🕂 — 🎽 🔞 🖉 🔰 🚺 🌡 🚯 🖬		
	Data Tag URI		
	1		

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags × Protocols					•
+ - 🗸 🕲 🖉 🔰 🕻	♣ ⓑ ⓑ	R 🔎 - Search 🍸 Filter by: Data	▼ Ite	ms used:6/10000 Protocol: Show all	🗹 Show all tags 🔅 🗖
Data ^	Туре	Comment	^	Property	Value
Modbus TCP:prot1	Container			✓ Driver	
Model: Modicon Modbus(1-based)				Model	Modicon Modbus(1-based)
	unsignedShort			Protocol	Modbus TCP:prot1
	unsignedShort unsignedShort			✓ Dictionary	
	unsignedShort			Array	false
	unsignedShort			Array size	0
	unsignedShort			Arrayindex.Subindex	400003
MRTU4 u	unsignedShort			Comment	
MRTU5 U	unsignedShort			Data type	unsignedShort

Toolbar item	Description
BA	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
督	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag. Example of both checked and unchecked result:
P - Search	Searches tags in the dictionary basing on filter combo- box item selected.

For tags referring to BACnet objects of type Calendar or Schedule the tag refresh rate is set to "Manual".

The following BACnet object properties are required for operation of the widgets.

Object	Tags to import
Calendar	Date_List
Schedule	Weekly_Schedule
	Exception_Schedule
	Default_Value
	Effective_Period

DEVICE Object Properties

A BACnet network scanner can detect properties when exploring the network and obtaining data from HMI device.

This are the supported DEVICE object properties:

Property	Description
Object_Identifier	BACnetObjectIdentifier
Object_Name	CharacterString
Object_Type	BACnetObjectType
System_Status	BACnetDeviceStatus
Vendor_Name	CharacterString
Vendor_Identifier	Unsigned16
Model_Name	CharacterString
Firmware_Revision	CharacterString
Application_Software_Version	CharacterString
Protocol_Version	Unsigned
Protocol_Revision	Unsigned
Protocol_Services_Supported	BACnetServicesSupported
Protocol_Object_Types_Supported	BACnetObjectTypesSupported
Object_List	BACnetARRAY[N]of BACnetObjectIdentifier
Max_APDU_Length_Accepted	Unsigned
Segmentation_Supported	BACnetSegmentation
APDU_Timeout	Unsigned
Number_Of_APDU_Retries	Unsigned
Device_Address_Binding	List of BACnetAddressBinding
Database_Revision	Unsigned

BACnet Alarm Events

The special "protAlarm:BACN" trigger mode, available from the Alarms Editor, give the possibility to receive alarm events from the BACnet native alarms module.

limitAlarm bitMaskAlarm deviationAlarm		^
	401	6
valueAlarm protAlarm:BACN[prot1]	onClassID 300	0015
activeMo	nday true	
activeTu	esday true	
activeWe	dnesday true	
activeTh	ursdav truo	×

Property	Description							
deviceID	Identifies the BACnet device in the network.							
notificationClassID	Notification Class ID to subscribe for the alarm events retrieving							
processID	Not used							
activeMonday activeTuesday activeWednesday activeThrusday activeFriday activeSaturday activeSunday	 Define in which days keep active the alarm events subscription False Subscription not active True Subscription active 							
startHour startMinute startSecond endHour endMinute endSecond	Define the time window where the alarm events subscription will be active							

The alarm widgets will report the alarm information that are provided from the BACnet device.

Select	Name	State Val		Time	Description		
	SISMI3NCE/Programming.4016.SUMMER-SP-SUPPLY:toOffNormal	Triggered Not Acked	90	13/02/2017 04:09:42	SUMMER ALARM		
	SISMI3NCE/Programming.4016.WINTER-SP-SUPPLY:toOffNormal	Triggered Not Acked	5	13/02/2017 04:10:06	WINTER ALARM		
			-				
4					•		
1	Check/Uncheck All Filter : Hide Not Triggered		•	Ack Ack	set) [Save]		



When the special "protAlarm:BACN" trigger mode is used, the widget of the active alarms show the timestamp provided from the BACnet device while the widget of the historical alarms show the timestamp of when the alarm events are received from the HMI device. Generally, both timestamps are the same but if you need to show the timestamp from the BACnet device even inside the widget of the historical alarms you can add a new column configured to use the "allSourceTimestamp" value from the alarm history widget.

P Search Name ▷ _AlarmsMgr ▷ _EventMgr ▷ _MultiLangMgr ▲ AlrmHstry ▲ AlrmHstry ▲ AlrmHstry.evntwgt.evntbfrwgt ■ alAlarmID ■ alBGColor ■ alBGColor ■ alBGcolor ■ alCustomField1 ■ alCustomField2 ■ alDescription ■ alFGColor ■ alFGColor ■ alFGColor ■ alState ■ alValue ■ alValue ■ alValue		
Name _AlarmsMgr _EventMgr _MultiLangMgr _AlrmHstry _AlrmHstry.evntwgt.evntbfrwgt _alAlarmID _alBGColor _alCustomField1 _alCustomField2 _alDateTime _alDescription _alFGColor _alFGColor _alFGColor _alSourceTimestamp _alState _alValue _alValue	field1.value	
Name	Source: O Tag O Alias O System Widget O Recipe	
> _AlarmsMgr > _EventMgr > _MultiLangMgr AlrmHstry > AlrmHstry.evntwgt.evntbfrwgt > alAlarmID > alBGColor > alCustomField 1 > alCustomField 2 > alDateTime > alEventType > alFGColor > alFdColor > alValue > alValue > evDate	P- Search	
EventMgr _MultiLangMgr AlrmHstry AlrmHstry.evntwgt.evntbfrwgt alAlarmID alBGColor alCustomField1 alCustomField2 alDateTime alEventType alFGColor alFGColor alFocolor alValue alValue alValue	Name	_
EventMgr _MultiLangMgr AlrmHstry AlrmHstry.evntwgt.evntbfrwgt alAlarmID alBGColor alCustomField1 alCustomField2 alDateTime alEventType alFGColor alFGColor alFocolor alValue alValue alValue evDate	▷ AlarmsMor	
MultiLangMgr AlrmHstry AlrmHstry.evntwgt.evntbfrwgt alAlarmID alBGColor alCustomField1 alCustomField2 alDateTime alEventType alFGColor alFGColor alSourceTimestamp alState alState alValue evDate		
 AlrmHstry.evntwgt.evntbfrwgt alAlarmID alBGColor alCustomField 1 alCustomField2 alDateTime alDescription alEventType alFGColor alGroups alName alSurceTimestamp alState alValue evDate 		
alAlarmID alBGColor alCustomField 1 alCustomField2 alDateTime alDescription alEventType alFGColor alGroups alName alSourceTimestamp alState alValue evDate	▲ AlrmHstry	
alBGColor alCustomField 1 alCustomField 2 alDateTime alDescription alEventType alFGColor alGroups alName alSourceTimestamp alState alValue evDate	 AlrmHstry.evntwgt.evntbfrwgt 	
alCustomField 1 alCustomField 2 alDateTime alDescription alEventType alFGColor alGroups alName alSourceTimestamp alState alValue evDate		
alCustomField2 alDateTime alDescription alEventType alFGColor alGroups alName alSourceTimestamp alState alValue evDate		
alDateTime alDescription alEventType alFGColor alGroups alName alSourceTimestamp alState alValue evDate		
alDescription alEventType alFGColor alGroups alName alSourceTimestamp alState alValue evDate		
alEventType alFGColor alGroups alName alSourceTimestamp alState alValue evDate		
alFGColor alGroups alName alSourceTimestamp alState alValue evDate		
alGroups alName alSourceTimestamp alState alValue evDate		
alName alSourceTimestamp alState alValue evDate		
alSourceTimestamp alState alValue evDate		
alState alValue evDate		
evDate		
	alValue	
evTime	evDate	
C. C. C. M. C.	evTime	



BACnet alarm is a special alarm that require a double space to be stored inside the events buffer. This means, for example, if the events buffer is configured to contain 1.000 events only the last 500 BACnet events will be stored.

BACnet Trend Buffer

To use a BACnet trend object as a trend buffer:

- 1. Open the Trends Editor
- 2. Click the "Add PLC Trend" button (This button is enabled only when at least one BACnet protocol is configured)
- 3. Configure the below parameters to identify the BACnet trend object to use.

Trends X	· · · · · · · · · · · · · · · · · · ·
Add trend Add PLC trend	X Delete trend
Trend1	Active PLC source BACN:prot1 -
Property V Location	Value
Device ID	0
Object ID Bit Index	0 -1
✓ Data	
Data type	float
✓ Time	
Time Specification	local

Property	Description							
Device ID	Identifies the BACnet device in the network.							
Object ID	BACnet ID of the trend object to be referenced.							
Bit Index	When the data type is boolean, it is the index to select the bit to use inside the BACnet bit_string. It is not used with the other data types.							
Data type	Specify the type of data of the BACnet trend object. The supported data types are: • boolean • int • unsignedInt • float							
Time Specification	Time format used inside the selected BACnet trend object local global (UTC) 							

The trend buffer thus configured can then be used inside any trend widgets.

BACnet Calendar Widget

Use Calendar widget to display content of a BACnet Calendar object.

Property	Description					
Date_List	Connect to the "Date_List" tag of a BACnet calendar object in ReadOnly or Read/Write.					
	Note: it can be connected to an alias which indexes a list of BACnet calendar Date_List(s), in order to use one calendar widget for more than one calendar object.					

Operation of Calendar Widget

The widget shows data for one month.

	MON	TUE	WED	THU	FRI	SAT	SUN
52	26	27	28	29	30	31	1
1	2	3	4	5	6	7	8
2	9	10	11	12	13	14	15
3	16	17	18	19	20	21	22
4	23	24	25	26	27	28	29
5	30	31	1	2	3	4	5
<	01/2	2017	>	Ne	ew Cle	ear All	Refresh

Use the < and > buttons to select the month to be displayed. The date of first day of the month is shown.

Swing gesture can be used on the widget to select the date.

New

Press the button "New" to enter a new calendar item. The button is active only if the tag associated to the calendar has been configured as Read/Write.

Calendar item	Description						
Single	Click on a day to select a single day into the calendar						
	MON TUE WED THU FRI SAT SUN 52 26 27 28 29 30 31 1 1 2 3 4 5 6 7 8 2 9 10 11 12 13 14 15 3 16 17 18 19 20 21 22 4 23 24 25 26 27 28 29 5 30 31 1 2 3 4 5 << 01/2017 >> Cancel Prev OK X						
Range	 Click on the first day and on the last day to select a range of days into the calendar. Single click on a day to change previous selected last day of the range. Double click on a day to change previous selected first selected day of the range. 						

Calendar item	Desci	riptior	า				
	Selec	t a date i	ange				
		MON T			FRI	SAT	SUN
	52	26		28 2			1
	1	2	3		5 6		8
	2	9		11 1			
	3	16		18 1			
	4	23		25 2	6 27	28	29
	5	30	31	1	2 3	8 4	5
	Prev	01/2017	>>		Cancel	Prev	ОК
MWD	Select	t a Da	y or a	Week	for e	ach y	ear c
	Selec	t a MWD					
		MON TI	JE WED	THU	FRI	SAT	SUN
	1-7						
	8-14						
	15-21						
	22-28						
	29-31						
	last 7 d.						

Clear All

Press the button "Clear All" to clear the content of the calendar object. The button is active only if the tag associated to the calendar has been configured as Read/Write. The button is configured to react to an onMouseHold event, to reduce risk of data loss.

Cancel Prev OK

•

January

Refresh

Press the "Refresh" button to start a manual refresh of the data of the widget. Always press the Refresh button after entering data in the calendar.

BACnet Schedule Widget

Use Schedule widget to display content of BACnet Schedule object.

Property	Description							
Туре	Select the type of BACnet object controlled by the schedule.							
	Options are:							
	• Binary							
	• Real							
	Multistate							
Weekly_Schedule	Attach to the Weekly_Schedule tag of the schedule object. The tag can be Read Only or Read/Write.							
Exception_Schedule	Optionally attach to the Exception_Schedule tag of the schedule object. The tag can be Read Only or Read/Write. Only attach this property if exceptions are used.							

Property	Description								
Default_Value	Optionally attach to the Default_Value tag of the schedule object. The tag can be Read Only or Read/Write. Only attach this property if default values are used.								
Cal. 0 (Date_List)	Optionally attach to the Date_List tag of the schedule widget in Read Only mode. Use this options to show the "calendar reference" exceptions.								
	 Note: An exception can be a single date, a date range, a mwd or a calendar reference. In this last case, exception_list does not contain the date information, but only time-value-priority and a reference to the calendar. The date_list needed to show the scheduling into the widget is stored into the relative BACNCalendar, and this is why we need this datalink. If there is no need to show calendar exceptions in the schedule, this property can be left void. Note: If it is not attached to a calendar, it is not possible to insert calendar exception. See BACNSchedKeypad for details. 								
Cal. 0 (Object_Name)	Optionally attach to the property of the calendar. This name is used to identify the calendar in the BACNSchedKeypad used to insert calendar exceptions. If Object_Name is not attached, the calendar is identified with its instance number. This property is used only if a Cal. 0 (Date_List) is attached to a calendar.								
Cal. 1 (Date_List)	Option for a second calendar.								
Cal. 1 (Object_Name)	Option for a second calendar.								
Value-color-text Map	Defines the association value – Color/Text shown in the schedule. Use this option to define all possible values available in the BACNSched keypad.								
	Value-Color Dialog ? ×								
	Tag value Mapped color Text								
	1 1 #00aaff Saving								
	2 2 #ffaa7f Confort								
	3 3 #55ff7f Normal								
	Ok Cancel								

Operation of Schedule Widget

The widget shows data for one week.

Default Value: Normal			New Clear All Refresh				
	MON	TUE	WED	THU	FRI	SAT	SUN
00:00							
04:00		E, 04:00 Normal					
08:00						E, 08:00 Confort	
12:00		E, 12:00 Confort					
16:00							
20:00		E, 20:00 Saving				E, 20:00 Saving	
<	16/01	/2017 ·	- 22/01	/2017	>		

Use the < and > buttons to select the week to be displayed. The date of first day and last day of the week is shown.

Swing gesture can be used on the widget to select the date.

New

Press the button "New" to enter a new schedule item. The button is active only if the tag associated to Weekly Schedule or Exception Schedule has been configured as Read/Write.

Schedule item	Description			
Weekly	Select the day and click Weekly button, the following dialog box appears. Then select the desired value and the time when it should be set. Press OK to confirm the new item.			
Exception Single	Click on a day to select a single day into the calendar. On the next dialog select the time window, the desired value and its priority. Image: Select a day Image: Select a day <tr< th=""></tr<>			
Exception Range	Click on the first day and on the last day to select a range of days into the calendar.			

Schedule item	Description				
	 Single click on a day to change previous selected last day of the range. Double click on a day to change previous selected first selected day of the range. 				
	On the next dialog select the time window, the desired value and its priority.				
Exception MWD	Select a Day or a Week for each year or each month. On the next dialog select the time window, the desired value and its priority.				
Exception Cal Ref	This option is available only if scheduler is linked to a calendar (configured as Read/Write) Select the time window, the desired value and its priority. Value will set on all days defined from the calendar. If there are more calendars associated with Scheduler widget, select the calendar to use.				

Clear All

Press the button "Clear All" to clear the content of the schedule object. The button is active only if the tag associated to the calendar has been configured as Read/Write. The button is configured to react to onMouseClick and onMouseHold events. The onMouseHold event will clear all data in the schedule. The onMouseClick event will recall a dialog box for selection of data to clear. It is needed to choice to clear weekly data or exception data.

Clear Weekly or Exception						
Monday -						
Clear Weekly	Clear All Exc.					
	Close					

Refresh

Press the "Refresh" button to start a manual refresh of the data of the widget. Always press the Refresh button after entering data in the schedule.

BACnet Effective Period Widget

Use the Effective Period widget to feed information to the Effective_Period tag of a Schedule object, if this is requested.

Property	Description
BACnet Effective_ Period	Attach to the Effective_Period tag of the Schedule object

01/10/2017 - 01/13/2017 Refresh

Operation of Effective Period Widget

The widget shows starting date and end date for the period.

Click on the area showing the dates to activate the data entry procedure showing the keypad BACNDateRange.

Select a date range								
	Always		All m	All month			year	
	MON	TUE	WED	THU	FRI	SAT	SUN	
52	26	27	28	29	30	31	1	
1	2	3	4	5	6	7	8	
2	9	10	11	12	13	14	15	
3	16	17	18	19	20	21	22	
4	23	24	25	26	27	28	29	
5	30	31	1	2	3	4	5	
< O1/2017 Esc Enter								

The keypad shows data for one month.

Use the < and > buttons to select the month to be displayed. The date of first day of the month is shown.

You may use the swing gesture on the widget to select the date.

Select the period clicking of first day and last day of the period. The Effective_Period is show with a different color. The keypad offers three predefined options:

Option	Description						
Always	The schedule will be always active.						
	//**** - **/**/**** Refresh						
All Month	The selected period will be extended to all months.						
	**/03/2017 - **/12/2017 Refresh						
All Year	The selected period will be extended to all years.						
	01/03/**** - 01/12/**** Refresh						

Refresh

Press the "Refresh" button to start a manual refresh of the data of the widget. Always press the Refresh button after entering data in the widget.

BACnet Keypads

Keypad	Description
BACNCal	Keypad for BACnet Calendar.
BACNDateRange	Keypad for BACnet Effective_Period.
BACNDefVal	Keypad for default value (embedded in the BACnet Schedule).
BACNSched	Keypad for BACnet Schedule.
	This keypad is context sensitive. It will show different options depending on the type of schedule.

BACnet widgets require dedicated keypads for data entry.

The system is configured to recall the appropriate keypad for each BACnet widget.

Using BACnet Server

BACnet protocol is capable to act as BACnet Server, by exposing BACnet objects.

To properly setup BACnet Server, it is needed to execute the following steps:

1. Configure objects to expose from **Protocol Editor Settings.**

BACnet			×
	Comm]	
Panel Device ID	262000	Analog Value Count	12
Object Name	DEV262000	Binary Value Count	11
Description	НМІ	Multi State Value Count	18
Media	[IP ▼	Notification Class Count	5
Timeout (ms)	5000	IP UDP Port	47808
Panel Node	1	Local IP	
COV Lifetime (s)	60		
COV Confirmed			
Max Master	127		
Max Info Frames	1		
max MS/TP APDU	480		
max IP APDU	1476		
Time Sync Interval (s) 0		
Time Sync UTC			
PLC Models			
default			
			OK Cancel



Note: Objects configured in above image can be discovered by BACnet clients:

Image: Subscriptions, Periodic Polling, Events/Alarms Image: Devices ObjectId Name Value Image: Devices Image: Devices	Time	Stat
Devices Devices Devices DeviceObjectId Name Value Device ObjectId Name Value	Time	Stat
Devices Udp:47808 DEV262000 [262000] Address Space Address Space ANALOG VALUE 0 ANALOG_VALUE:1 ANALOG_VALUE:2 ANALOG_VALUE:3 ANALOG_VALUE:4	Time	Staf
Address Space Address Space ANALOG VALUE 0 ANALOG_VALUE:1 ANALOG_VALUE:2 ANALOG_VALUE:3 ANALOG_VALUE:4	Time	Stat
Address Space		
DEV262000 ANALOG VALUE 0 ANALOG_VALUE:1 ANALOG_VALUE:2 ANALOG_VALUE:3 ANALOG_VALUE:4		
ANALOG VALUE 0 ANALOG_VALUE:1 ANALOG_VALUE:2 ANALOG_VALUE:3 ANALOG_VALUE:4		
ANALOG_VALUE:1 ANALOG_VALUE:2 ANALOG_VALUE:3 ANALOG_VALUE:4		
ANALOG_VALUE:2 ANALOG_VALUE:3 ANALOG_VALUE:4		
ANALOG_VALUE:3		
ANALOG_VALUE:4		
ANALOG_VALUE:5		
ANALOG_VALUE:6		
ANALOG_VALUE:7		
ANALOG_VALUE:8		
ANALOG_VALUE:9		
ANALOG_VALUE:10		
ANALOG_VALUE:11		
BINARY_VALUE:0		
BINARY_VALUE:1		
BINARY_VALUE:2		
BINARY_VALUE:3		
BINARY_VALUE:4		
BINARY_VALUE:5		
BINARY_VALUE:6		
BINARY_VALUE:7		
BINARY_VALUE:8		
BINARY_VALUE:9		
BINARY_VALUE:10		
NOTIFICATION_CLASS:0		
NOTIFICATION_CLASS:1		
NOTIFICATION_CLASS:2		
NOTIFICATION_CLASS:3		
MULTI_STATE_VALUE:0		
MULTI_STATE_VALUE:1		
MULTI_STATE_VALUE:2		
MULTI_STATE_VALUE:3		
MULTI_STATE_VALUE:4		
MULTI_STATE_VALUE:5		
MULTI_STATE_VALUE:6		

2. Create Tags that points to local BACnet objects, setting Device ID as the Device ID configured in Protocol Editor Settings:

ACnet BACnet			X
Object Type Analog Value	Device ID 262000	Data Type float 🗸	
Arraysize Cor	version +/-	Object Instance	
Object Property	Array Index	Write Priority	
COV			
	ОК	Cancel Apply	Help

Device objects description

Property Name	Code	Default value	Permanent	Note	Data Type
APDU timeout	11	Parameter	Yes		UnsignedInt
Application software version	12		Read-only		String
Database version	155		Read-only		UnsignedInt
Daylight saving status	24		Read-only		Boolean
Read-only	28	Parameter	Yes		String
Device address binding	30		Read-only		String
Firmware revision	44		Read-only		String
Local date	56		Read-only		UnsignedInt
Local time	57		Read-only		UnsignedInt

Property Name	Code	Default value	Permanent	Note	Data Type
Location	58	Parameter	Yes		String
Max APDU length accepted	62		Read-only		UnsignedInt
Max info frames	63	Parameter	Yes	Only if MSTP	String
Max master	64	Parameter	Yes	Only if MSTP	String
Model name	70		Read-only		String
Number of APDU retries	73	Parameter	Yes		UnsignedInt
Object identifier	75	Parameter	Yes		UnsignedInt + Conversion
Object list	76		Read-only		UnsignedInt + Conversion
Object name	77	Parameter	Yes		String
Object type	79		Read-only		UnsignedInt
Protocol object types supported	96		Read-only		Boolean(51)
Protocol revision	139		Read-only		UnsignedInt
Protocol services supported	97		Read-only		Boolean(40)
Protocol version	98		Read-only		UnsignedInt
Segmentation supported	107		Read-only		UnsignedInt
System status	112		Read-only		UnsignedInt
UTC offset	119		Read-only		Int
Vendor identifier	120		Read-only		UnsignedInt
Vendor name	121		Read-only		String

Analog Value objects description

Property Name	Code	Default value	Permanent	Note	Data Type
Acked transitions	0		Read-only		Boolen(3)
COV increment	22	0	Yes		Float
Deadband	25	0	Yes		Float
Description	28	"ANALOG	Yes		String

Property Name	Code	Default value	Permanent	Note	Data Type
		VALUE n"			
Event enable	35	0	Yes		Boolean(3)
Event state	36	0	Read-only		UnsignedInt
Event time stamps	130		Yes		UnsignedInt(3)
High limit	45	0	Yes		Float
Limit enable	52	0	Yes		Boolean(2)
Low limit	59	0	Yes		Float
Notification class	17	4194303	Yes		UnsignedInt
Notify type	72	0	Yes		UnsignedInt
Object identifier	75	2:n	Read-only		UnsignedInt + Conversion
Object name	77	"ANALOG VALUE n"	Yes		String
Object type	79	2	Read-only		UnsignedInt
Out of service	81	0	Yes		Boolean
Present value	85	0			Float
Priority array	87		Read-only		16 Single tag String
Reliability	103	0	Yes		UnsignedInt
Relinquish default	104	0	Yes		Float
Status flags	111		Read-only		Boolean(4)
Time delay	113	0	Yes		UnsignedInt
Units	117	98	Yes		Units

Binary Value objects description

Property Name	Code	Default value	Permanent	Note	Data Type
Acked transitions	0		Read-only		Boolean(3)
Active text	4		Yes		String
Alarm value	6	0	Yes		Boolean
Description	28	"BINARY VALUE n"	Yes		String

Property Name	Code	Default value	Permanent	Note	Data Type
Event enable	35	0	Yes		Boolean(3)
Event state	36	0	Read-only		UnsignedInt
Event time stamps	130		Yes		UnsignedInt(3)
Inactive text	46		Yes		String
Notification class	17	4194303	Yes		UnsignedInt
Notify type	72	0	Yes		UnsignedInt
Object identifier	75	5:n	Read-only		UnsignedInt + Conversion
Object name	77	"BINARY VALUE n"	Yes		String
Object type	79	5	Read-only		UnsignedInt
Out of service	81	0	Yes		Boolean
Polarity	84	0	Yes		UnsignedInt
Present value	85	0			Boolean
Priority array	87		Read-only		16 Single tag String
Reliability	103	0	Yes		UnsignedInt
Relinquish default	104	0	Yes		Boolean
Status flags	111		Read-only		Boolean(4)
Time delay	113	0	Yes		UnsignedInt

Multi State Value objects description

Property Name	Code	Default value	Permanent	Note	Data Type
Acked transitions	0		Read-only		Boolean(3)
Alarm values	7		Yes	Defines number of array elements	UnsignedInt
				Array of alarm values (0:n)	UnsignedInt(n)
Description	28	"MULTI STATE VALUE n"	Yes		String
Event enable	35	0	Yes		Boolean(3)

Property Name	Code	Default value	Permanent	Note	Data Type
Event state	36	0	Read-only		UnsignedInt
Event time stamps	130		Yes		UnsignedInt(3)
Fault values	39		Yes	Defines number of array elements	UnsignedInt
				Array of fault values (0:n)	UnsignedInt(n)
Number of states	74	1	Yes		UnsignedInt
Notification class	17	4194303	Yes		UnsignedInt
Notify type	72	0	Yes		UnsignedInt
Object identifier	75	19:n	Read-only		UnsignedInt + Conversion
Object name	77	"MULTI STATE VALUE n"	Yes		String
Object type	79	19	Read-only		UnsignedInt
Out of service	81	0	Yes		Boolean
Present value	85	0			UnsignedInt
Priority array	87		Read-only		16 Single tag String
Reliability	103	0	Yes		UnsignedInt
Relinquish default	104	0	Yes		UnsignedInt
State text	110		Yes		UnsignedInt
Status flags	111		Read-only		Boolean(4)
Time delay	113	0	Yes		UnsignedInt

Notification Class objects description

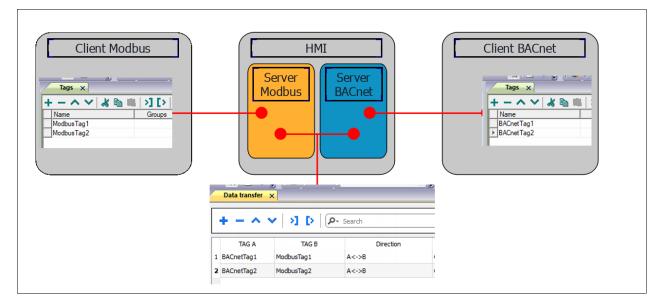
Property Name	Code	Default value	Permanent	Note	Data Type
Ack required	1	0	Yes		Boolean(3)
Description	38	"NOTIFICATION CLASS n"	Yes		String
Notification class	17	4194303	Yes		UnsignedInt
Object identifier	75	15:n	Read-only		UnsignedInt +

Property Name	Code	Default value	Permanent	Note	Data Type
					Conversion
Object name	77	"NOTIFICATION CLASS n"	Yes		String
Object type	79	15	Read-only		UnsignedInt
Priority	86	255,255,255	Yes		UnsignedInt(3)
Recipient list	102		Yes		UnsignedInt(n)

Example of usage

Once BACnet Server Tags are configured, they can be used in combination with Data Transfer feature.

Example: Modbus TCP/RTU Tags can be transferred to BACnet Tags (with same data type). In this way, all BACnet clients can reach BACnet Server and see actual value of Modbus Tags, using BACnet Tags as interface.



JavaScript Interface

Beside Tag interface the user can access the protocol via JavaScript.

Although defined Tags can be accesses by JavaScript too, JavaScript can access directly to a Command interface implemented in protocol. This interface does not require the definition of Tags and is direct to protocol resulting in more efficiency.

The following commands are supported:

Command	Description
scan (minID, maxID, <timeout>)</timeout>	Executes a scan for devices in the given range.
scan_status	Get the scanning result.
devices	Get the list of devices.
objectCount (deviceID, objectType)	Get the object count of given object types in given device.
objectNames (start, count)	Get the part of object names asked by previous objectCount.
properties (deviceID, objectType, objectInstance)	Get the properties of given device/object.

scan

Scan the bus to find all present devices having ID in the range minID – maxID.

To scan the whole network use 0 and 999999 ad minID and maxID.

The optional timeout can be indicated in milliseconds. Default value is 2000 ms.

The function starts the scan operation; the function scan_status can be used to know the status of the operation. The result of the operation is "**scanning**".

scan_status

Get the status of last started scan operation. It returns "**scanning**" or "**finished**". Scan operation finishes when the timeout time is expired

devices

Get the list of devices found by latest scan operation. The result is a JSON string containing of each device:

- device name
- model name
- vendor name
- vendor ID

Example:

```
{"minID":0,"maxID":999999,"devices":[262000,1101],"deviceNames":
["DEV262000","S01101"],"modelNames":["HMI model","EY-AS525F001"],"vendorNames":
["Company Name","SAUTER"],"vendorIDs":[262,80]}
```

objects

Get the list of all objects from the devices having the given ID. The list is returned as a JSON string containing for each object

- type
- instance number

type can be:

OBJECT_ANALOG_INPUT = 0,

OBJECT_ANALOG_OUTPUT = 1,

OBJECT_ANALOG_VALUE = 2,

OBJECT_BINARY_INPUT = 3,

OBJECT_BINARY_OUTPUT = 4,

OBJECT_BINARY_VALUE = 5,

OBJECT_CALENDAR = 6,

OBJECT_COMMAND = 7,

OBJECT_DEVICE = 8,

OBJECT_EVENT_ENROLLMENT = 9,

OBJECT_FILE = 10,

OBJECT_GROUP = 11,

OBJECT_LOOP = 12,

OBJECT_MULTI_STATE_INPUT = 13,

OBJECT_MULTI_STATE_OUTPUT = 14,

OBJECT_NOTIFICATION_CLASS = 15,

OBJECT_PROGRAM = 16,

OBJECT_SCHEDULE = 17,

OBJECT_AVERAGING = 18,

OBJECT_MULTI_STATE_VALUE = 19,

OBJECT_TRENDLOG = 20,

OBJECT_LIFE_SAFETY_POINT = 21,

OBJECT_LIFE_SAFETY_ZONE = 22,

OBJECT_ACCUMULATOR = 23,

OBJECT_PULSE_CONVERTER = 24,

OBJECT_EVENT_LOG = 25,

OBJECT_GLOBAL_GROUP = 26,

OBJECT_TREND_LOG_MULTIPLE = 27,

OBJECT_LOAD_CONTROL = 28,

OBJECT_STRUCTURED_VIEW = 29,

OBJECT_ACCESS_DOOR = 30,

OBJECT_TIMER = 31,

OBJECT_ACCESS_CREDENTIAL = 32,

OBJECT_ACCESS_POINT = 33,

OBJECT_ACCESS_RIGHTS = 34,

OBJECT_ACCESS_USER = 35,

OBJECT_ACCESS_ZONE = 36,

OBJECT_CREDENTIAL_DATA_INPUT = 37,

OBJECT_NETWORK_SECURITY = 38,

OBJECT_BITSTRING_VALUE = 39,

OBJECT_CHARACTERSTRING_VALUE = 40,

OBJECT_DATE_PATTERN_VALUE = 41,

OBJECT_DATE_VALUE = 42,

OBJECT_DATETIME_PATTERN_VALUE = 43,

OBJECT_DATETIME_VALUE = 44,

OBJECT_INTEGER_VALUE = 45,

OBJECT_LARGE_ANALOG_VALUE = 46,

OBJECT_OCTETSTRING_VALUE = 47,

OBJECT_POSITIVE_INTEGER_VALUE = 48,

OBJECT_TIME_PATTERN_VALUE = 49,

OBJECT_TIME_VALUE = 50,

OBJECT_NOTIFICATION_FORWARDER = 51,

OBJECT_ALERT_ENROLLMENT = 52,

OBJECT_CHANNEL = 53,

OBJECT_LIGHTING_OUTPUT = 54,

OBJECT_BINARY_LIGHTING_OUTPUT = 55,

OBJECT_NETWORK_PORT = 56,

Other types are manufacturer specific.

objectCount

Returns the number of objects of a defined type in the device having the indicated ID. If specified type is -1 the command will return the number of all objects.

```
Example:
objectCount 1101 -1
77
objectCount 1101 0
1
objectCount 1101 1
1
objectCount 1101 3
2
```

objectCount 1101 29 16

```
    objectNames
```

Returns a part of the objects listed by a previous **objectCount** command, from start index. The list contains only counted objects according to filter previously used

The list is returned as a JSON string containing for each object

- type
- instance number
- name

Example:

```
{"deviceID":1101,"objects":[{"type":29,"instance":0,"name":"0x7400000"},
{"type":29,"instance":16,"name":"0x7400010"},
{"type":29,"instance":18,"name":"0x7400012"},
{"type":29,"instance":19,"name":"0x7400013"},
{"type":29,"instance":20,"name":"0x7400014"},
{"type":29,"instance":21,"name":"0x7400015"},
{"type":29,"instance":22,"name":"0x7400016"},
{"type":29,"instance":23,"name":"0x7400017"},
{"type":29,"instance":24,"name":"0x7400018"},
{"type":29,"instance":25,"name":"0x7400019"},
{"type":29,"instance":26,"name":"0x740001a"},
{"type":29,"instance":27,"name":"0x740001b"},
{"type":29,"instance":28,"name":"0x740001c"},
{"type":29,"instance":29,"name":"0x740001d"},
{"type":29,"instance":30,"name":"0x740001e"},
{"type":29,"instance":31,"name":"0x740001f"}]}
```

properties

Returns the list of properties available for object with given type and instance number in device having the given ID.

The list is returned as a JSON string containing for each object

- deviceID
- object type
- object instance
- list of available properties

Example:

```
{"deviceID":1101,"objectType":2,"objectInstance":1,
"properties":
[22,28,36,65,69,75,77,79,81,85,87,103,104,111,117,168,8309,8314,8332,8333]}
```

Example of usage:

```
var tagMgr = project.getWidget("_TagMgr");
var protID = "prot2"; // to be set according to protocol numbering
var params = String(fromId) + " " + String(toId) + " " + String
(timeout); // fromID and toID are min and max IDs
var json_str = tagMgr.invokeProtocolCommand(protID , "scan", params, state); //json_
str containts JSON string with scanned devices.
```

Communication status

Current communication status can be displayed using system variables. See "System Variables" section in the main manual.

Error	Cause
Cannot bind to the device_id	Cannot establish communication with the Device ID provided for this tag.
Cannot read the property data type	The type of the property to write cannot be determined.
write conversion error	A conversion associated to this tag has failed.
Cannot write ICOM type BACnet type	A datatype selected for this tag is not compatible with the BACnet property to set.
Timeout on COV subscription	A request for COV subscription for this tag has timed out.
Timeout on waiting COV update	A COV notification has not been received for this tag whithin timeout.
Can't get COV for this property	The selected property for COV notification is unsupported.
datagramItem conversion error	A conversion associated to a tag that is part of a datagram has failed.
Timeout waiting on response	No response for a request of read or write property within timeout.
datagram element, no data available	No data available for a tag that is part of datagram.
datagram element, Unsupported BACnet data type	Read datagram element is of unsupported BACnet type.
datagram element, can't convert BACnet type to	A Data Type selected for a tag which is part of a datagram is not compatible with the BACnet property to read.

Codes supported by this communication driver:

Error	Cause
No data in response	No data available for a tag.
Datagram element 'element_ URI' error: 'error_class': error_code	The reading of indicated datagram element 'element_URI' was reported as error. The error descriptions error_class and error_code are included in the message.
datagram object does not match	The object of the received datagram item does not match the asked object.
datagram property does not match	The property of the received datagram item does not match the asked property.
BACnet abort: reason_of abort	BACnet abort message was received. The reason of abort is given.
BACnet reject: reason_of_ rejection	BACnet reject message was received. The reason of rejection is given.
BACnet error: error_class: error_code	BACnet error message was received. The error description is given as combination of error_class and error_code .
parameter 'parameter_name' out of range	The protocol parameter parameter_name value is out of range.

Beckhoff ADS

Beckhoff ADS protocol driver is used for communication with Beckhoff controllers through Ethernet connection. This implementation of Beckhoff ADS protocol driver is based on the information published by Beckhoff.

Protocol Editor Settings

Add (+) a driver in the Protocol editor and select the protocol "Beckhoff ADS" from the list of available protocols.

Beckhoff ADS		×
PLC Network		ОК
Alias		Cancel
Panel AMS Net ID	0.0.0.0.0	
Panel AMS Port	32976	
Target IP Address	0.0.0.0	
Target TCP Port	48898	
Target AMS Net ID	0.0.0.0.0	
Target AMS Port	801	
Timeout	3000	
PLC Models		
BC/BX		
PC/CX		

Element	Description
Alias	Name to be used to identify nodes in the plc network configuration. The name will be added as a prefix to each tag name imported for each network node.
Panel AMS Net ID	Specifies the AMS net ID of the panel; the first 4 bytes must match the panel IP address assigned to the HMI device. If panel has IP address 192.168.10.100 then AMS Net ID could be 192.168.10.100.1.1
Panel	Specifies the panel AMS port number to be used on panel.

Element	Description					
AMS Port	Using TwinCAT2, default Panel AMS Port is 32976.					
	Using TwinCAT3, default Panel AMS Port is 32844.					
Target IP Address	Specifies the IP address of the target controller.					
Target AMS Net ID	Specifies the Target AMS net ID of the target controller.					
Target AMS Port	Specifies the port number dedicated to the communication on target device. Using TwinCAT2, default Target AMS Port is 801. Using TwinCAT3, default Target AMS Port is 851.					
Timeout	The number of milliseconds between retries when communication fails.					
PLC models	Select the model which corresponds to the device to be connected. Model selection is very important to be set properly.					
PLC Network	The protocol allows the connection of multiple controllers to one operator panel. To set-up multiple connections, check "PLC network" checkbox and enter the Target Controller settings for every node.					

TwinCAT2 Route Settings

Beckhoff controllers require some specific settings to allow connection from HMI devices.

In TwinCAT2 System Manager you need to configure Static Route.

First of all the system must be reset in Configuration Mode using the toolbar button as showed in the following figure.

File Edit Actions View Options	Help
: D 🖻 📽 🖬 🎼 🗛 🕺	🖻 🖥 📾 🖊 ð 黒 📾 🗸 🌌 🇶 🧶 🗮 🌾 🏵 🗣 🖹 Q 🧕
SYSTEM - Configuration Image: Configuration <t< td=""><td>Current Routes Static Routes Project Routes Netle M. nagement</td></t<>	Current Routes Static Routes Project Routes Netle M. nagement
Route Settings	Route AmsNetId Address Type Commer
Configuration	
PLC - Configuration	
I/O Devices	
🔤 🚰 Mappings	
and the second s	Version of the second s

Then confirm to Restart TwinCAT2 System in Config Mode as in the figure below.

TwinCAT System Manager			
Restart TwinCAT System in Config Mode			
OK Cancel			
N N			

Once restarted, as in the next figure, follow these steps to add a new Route:

- 1. Open Route Settings.
- 2. Select Static Routes tab.
- 3. Click on [Add] button.

File Edit Actions View Options	Help
<u>:</u> 🗅 🚅 📽 🖬 🎒 🗟 🛛	🖻 🖬 🛃 🦄 🖢 🔚 🖬 🗸 🍏 🏡 🌺 🌂 🔪
SYSTEM - Configuration System - Configuration Real-Time Settings	Current Routes Static Routes Project Routes NetId Managem
ditional Tasks	Route AmsNetId Address
Route Settings COM Objects	
PLC - Configuration	
I/O - Configuration I/O Devices	A 100 A 1
🚰 Mappings	<u> </u>
	.
	l l
	2
	Add
and a second	and and the second second

Into Add Route Dialog user must set:

1. Route Name: a name useful to indentify the Route i.e. "HMI", AmsNetId: The Panel AMS Net ID as configured into Beckhoff ADS protocol, Transport Type: TCP/IP.

Address Info: Type in the Panel IP Address with "IP Address" option selected.

- 2. Target Route: Static.
- 3. Click on [Add Route] button. Note: no warning or message will be shown.
- 4. Click on [Close] button.

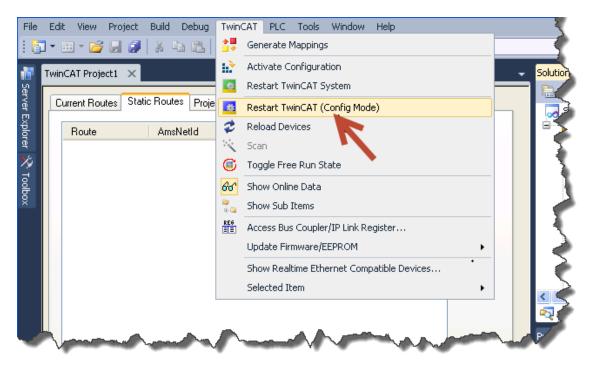
Add Route Dialog] [Refresh Status	Broadcast Search
Host Name	Connected Address	AMS NetId	TwinCAT OS V	ersion Comment
<	III]	>
Route Name (Target): AmsNetId: Transport Type: Address Info: Host Name IP	JMobile HMI 192.168.10.100.1.1 TCP/IP 192.168.10.100 Address	2	Route Name (Remote): Target Route Project Static Temporary	TS-VM-XP2 Remote Route O None O Static O Temporary
Connection Timeout (s):	5	3	Add Route	4 Close

Then the route will appear under Static Routes list.

TwinCAT3 Route Settings

Beckhoff controllers require some specific settings to allow connection from HMI devices. In TwinCAT3 XAE you need to configure a Static Route.

First of all TwinCAT3 system must be reset in Configuration Mode using the toolbar button as showed in the following figure.



Then confirm to Restart TwinCAT3 System in Config Mode.

TwinCAT XAE				
Restart TwinCAT System in Config Mode				
OK Cancel				

Once restarted, as in the next figure, follow these steps to add a new Route:

- 1. Open Routes.
- 2. Select Static Routes tab.
- 3. Click on [Add] button.

File		View	Project	Build	Debug	TwinCAT	Tools • 🖳				vinCAT	32 🔹 💌
🏪 Server Explorer 😤 Toolbox	Cur	rent Rou Route		ic Route	NetId	ct Routes	Manage	Type	F	Comment		Solution Explorer

Into Add Route Dialog user must set:

- Route Name: a name useful to indentify the Route i.e. "HMI", AmsNetId: The Panel AMS Net ID as configured into Beckhoff ADS protocol, Transport Type: TCP/IP. Address Info: Type in the Panel IP Address with "IP Address" option selected.
- 2. Target Route: Static.
- 3. Click on [Add Route] button. Note: no warning or message will be shown.
- 4. Click on [Close] button.

Add Route Dialog				
Enter Host Name / IP:]	Refresh Status	Broadcast Search
Host Name	Connected Address	AMS NetId	TwinCAT OS Ve	ersion Comment
٢	Ш			٢
Route Name (Target):	JMobile HMI	2	Route Name (Remote):	TS-VM-XP2
AmsNetId:	192.168.10.100.1.1	4	Target Route	Remote Route
Transport Type:	TCP/IP 🔽		Project Static	 None Static
Address Info:	192.168.10.100		Temporary	 Temporary
🔾 🔿 Host Name 🛛 💿 IF	Address			
Connection Timeout (s):	5	3	Add Route	4 Close

Then the route will appear under Static Routes list.

Tag Import

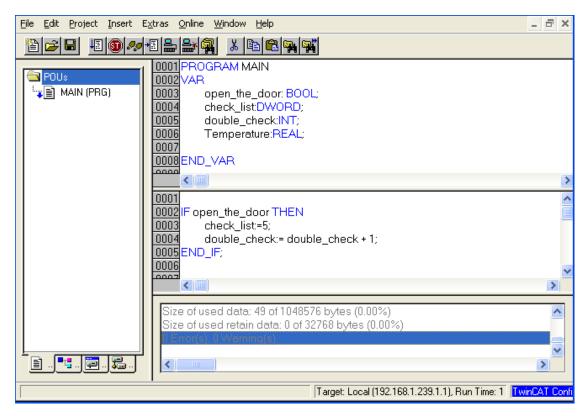
Exporting Tags from PLC

The data in the Beckhoff system is based on tags.

The organization of the internal memory of the controller is not fixed but it is configured by the user at development time. Each data item can be identified by a string called "tag".

The TwinCAT development environment generates the list of tags created for each controller in the configuration of the application.

The project in the panel must refer to the tag names assigned in the TwinCAT PLC Control programming software at development time. The Designer Tag Editor supports direct import of the tag file generated by the Beckhoff software.



To export tags defined for the selected controller, click on Project > Build as shown.

File Edi	t Project	Insert	Extras	Online	Window	Help	- 8 ×
🖹 🚅 🗲	Build				Ctrl+F8		
	Rebu						
🔁 POUs	Clear		id informa				
— Цар МА	JN	downioa	ia informa	3000		— r: BOOL;	
	Obje					DRD;	
	Proje	ct databa	ase			▶ INT:	
	Optic)ns				EAL;	
	Trans	slate into	other lar	nguages		•	
	Docu	ment					>
	Expo	rt					
	Impo	rt				HEN	
	Merg					= double_check + 1;	
		bare				= double_check + 1,	
		ct Info					
		al Search. al Replace					>
		я керіас	e				
		Instance				n	~
		/ Call Tree	-			'MAIN'	
	Show	Cross R	eference			Standard'	
Ē. 📑	. Chec	k				•	>
		Action					
Compiles the	llser	Group Pa	acciulorde			Target: Local (192.168.1.239.1.1), Run Time:	TwinCAT Confi

The TwinCAT PLC Control software will create a file with extension TPY.

Importing Tags in Tag Editor

Select the driver in Tag Editor and click on the Import Tags button to start the importer.

	Tags	×								
+	—	X	D	ß	>]	₽	A 9B	B>	ŧ i ł	1
Data	1		^		-	Та	g URI			

The following dialog shows which importer type can be selected.

MIStudio	×				
Multiple tag importers are available for this protocol. Please select the importer type and continue.					
Version	Туре				
TwinCAT v1.0	Linear				
Tag Editor exported	xml General				
	OK Cancel				

Importer	Description				
TwinCAT v1.0	Requires a .tpy file.				
Linear	All variables will be displayed at the same level.				
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.				
	Tags ×				
	+ - 👗 🕲 🔊 🚺 🥵 🖻 🚮 🗉				



Note: the Beckhoff driver supports direct access to the PLC tags using the handles; this means that if no tags are added to the PLC and the PLC program is just re-compiled, you do not need to re-import tags as the access to them does not depend from the offset, but only from name.

Tag URI

Once the importer has been selected, locate the symbol file and click **Open**.

Data

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags × Protocols					
+ - 🎽 🙆 🔊	D 🗞 🖬 🚮	R 🔎 - Search Tilter by: Data	▼ Ite	ems used:6/10000 Protocol: Show a	all 🛛 🗹 Show all tags 🔅 🗖
Data	Туре	Comment	^	Property	Value
Modbus TCP:prot1	Container			✓ Driver	
Model: Modicon Modbus(1-base	1)			Model	Modicon Modbus(1-based)
- Holding Registers 1	unsignedShort			Protocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			✓ Dictionary	
Holding Registers 3	unsignedShort			Array	false
- MRTU1 - MRTU2	unsignedShort unsignedShort			Array size	0
MRTU2	unsignedShort		_	Arravindex.Subindex	400003
MRTU4	unsignedShort				00005
- MRTU5	unsignedShort			Comment	i let i
1	an argine a an for e		_	Data type	unsignedShort

Toolbar item	Description
ka	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
βă.	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.
	Example of both checked and unchecked result:
	Tops* Image Tops Image Image Tops Image Tops Image Tops Image Image Tops Image Tops Image Tops Image Image <th< th=""></th<>
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Using TwinCAT v1.0 Import Filter

When importing tags, the user can decide to import all the tags from the **.tpy** file or apply a filter importing only a subset of them.

The figure below shows how to specify the filter. The filter consist in a string (no wildcards are supported). The import filter will import only the tags having the specified string in the description.

If the description is applied to an "instance declaration" of a Function Block, all the tags within the block will be imported.

If the string is contained only as comment of some variables inside the Function Block, only that variables will be imported.

Beckhoff ADS importer - Filter selection	8 ×
 Import all Import HMI 	OK Cancel

As an example for the use of the import filter, please see the following case.

```
FUNCTION BLOCK FB Motor
VAR INPUT
     bStartMotor: BOOL;
    bReset: BOOL;
END VAR
VAR OUTPUT
    bMotorOn: BOOL;
    bAlarm: BOOL; (* HMI Thermal alarm *)
END VAR
VAR
     sData: STRING;
    bResetStatistics: BOOL; (* HMI Reset statistics *)
END VAR
VAR PERSISTENT
     stStat: ST_MotorStats; (* HMI Motor statistics *)
END VAR
Function block instances declaration:
VAR
     fbMotor1: FB Motor;
     fbMotor2: FB Motor; (* HMI only show Motor 2!! *)
END VAR
```

The following tags will be imported:

- MAIN/fbMotor2/bAlarm
- MAIN/fbMotor2/bResetStatistics
- MAIN/fbMotor2/ST_MotorStats

Override Data Types

The protocol provides special data types which allow you to change the protocol configuration at runtime. If added in the project, these variables are initialized with the value specified in the project at programming time.

1:Page1* Protocols	Tags* 🗙		
- ^ ~ 🖌 🖻 📖	>] [> Beckhoff A	DS:prot1	- 5
Name	Groups	Driver 🛆 📗	Address
Beckhoff ADS Beckhoff ADS			×
deviceCode Boolean	dataType boolean	Arraysize	
Real Long Real Time Time Of Day Date DT String Boolean Array Byte Array Unsigned short integer Word Array Unsigned Integer Array Double word Array Unsigned Double Integer Short Integer Array			varName
Integer Array Double integer Array Real Array Long Real Array Time Array	≡	Cancel	Apply Help
Time Of Day Array Date Array DT Array String Array IP Override AMS NetID Override Panel AMS NetID Override	de		
Port Override AMS Port Override Panel AMS Port Override			

The table below shows which data type to use for any protocol parameter to override at runtime.

Override Data Type	Protocol Parameter		Description
IP Override	Target IP Address	0 . 0 . 0 . 0	Overrides the PLC IP address. It is an unsignedByte array of 4 elements, one per each byte of IP address.
AMS NetID Override	Target AMS Net ID	0.0.0.0.0	Overrides the PLC AMS NetID. It is an unsignedByte array of 6 elements, one per each byte of AMS NetID.
Panel AMS NetID Override	Panel AMS Net ID	0.0.0.0.0	Overrides the PLC AMS NetID. It is an unsignedByte array of 6 elements, one per each byte of AMS NetID.

Override Data Type	Protocol Parameter		Description
Port Override	Target TCP Port	48898	Overrides the PLC TCP port. It is an unsignedShort .
AMS Port Override	Target AMS Port	801	Overrides the PLC AMS port. It is an unsignedShort .
Panel AMS Port Override	Panel AMS Port	32976	Overrides the Panel AMS port. It is an unsignedShort .

Aliasing Tag Names in Network Configurations

Tag names must be unique at project level; it often happens that the same tag names are to be used for different controller nodes (for example when the HMI is connected to two devices that are running the same application). Since tags include also the identification of the node and Tag Editor does not support duplicate tag names, the import facility in Tag Editor has an aliasing feature that can automatically add a prefix to imported tags. With this feature tag names can be done unique at project level.

The feature works when importing tags for a specific protocol. Each tag name will be prefixed with the string specified by the "Alias". As shown in the figure below, the connection to a certain controller is assigned the name "Node1". When tags are imported for this node, all tag names will have the prefix "Node1" making each of them unique at the network/project level.

⊦-^~│∦『	b 🖷 🔁	Beckhoff ADS:prot1	 §		
Name 🛆		Group	Driver	Address	Comment
Node's Gal_hodnote		(#1212) Witcold	Bue TCP pet1 1	TT Durwigned/Shoft	
Node1/Cala_hodnote		/H212 / Wend	Bue TCP pet1 1	12 Dunsigned/Shoft	
Node1/IN_W/X7ER_law	al i	/41212 Wend	Bue TCP pet1 1	D/D-unsigned/DhoH	
Note: CHOCHERON		/4212 Week	Bue TCP pet1 1 2	245/Dunsigned/Shot	
Nede1/CRUT_BRT_HouP	with a	(H212) West	Bue TCP pet1 1	1 (Dunnigmed/Short)	
Webert/FE_D/KTA_headro	fia	(H212) Wend	Bue TCP polt1 1	2 (Dumsigned Short	
NAME TO MAKE THE PRO		/H212 (West	Bue TCP pet1 1	3 @unwigned/Short	
Node1/Water_level		ABB Calls	whe Efficient 1	10 0 unsigredShort	
		lode id as defined in im Gelect Network node id	port file		
		Slave Id	Model	Alias	
	-	K-K-K	Nullion multip	Node1	
		X X X Z	Nullion multius	Node2	
taqname Water_level	me			O ^l t	Cancel
tagname 🔽	me			Ok	Cancel
tagname 🗸	me			Ok	Cancel
tagname T	me			Ok	Cancel
tagname T	me	245	ő	Ok	Cancel
taoname C		245	0		
Water_level		245 0 12	0	unsignal/Diot	



Note: Aliasing tag names is only available when tags can be imported. Tags which are added manually in the Tag Editor do not need to have the Alias prefix in the tag name.

The Alias string is attached to the tag name only at the moment the tags are imported using Tag Editor. If you modify the Alias string after the tag import has been completed, there will be no effect on the names already present in the dictionary. When the Alias string is changed and tags are imported again, all tags will be imported again with the new prefix string.

Communication Status

The current communication status can be displayed using the dedicated system variables. Please refer to the User Manual for further information about available system variables and their use.

The codes supported for this communication driver are:

Error	Notes
NAK	Returned in case the controller replies with a not acknowledge
Timeout	Returned when a request is not replied within the specified timeout period; ensure the controller is connected and properly configured to get network access
Invalid response	The panel did receive from the controller a response, but its format or its contents is not as expected; ensure the data programmed in the project are consistent with the controller

Error	Notes
	resources
General Error	Error cannot be identified; should never be reported; contact technical support

Client System Variables

Client System Variables communication driver allows to create Tags that point to system information.

Refer to <u>Client System Variables > Protocol</u> chapter of User's Manual.

Protocol Editor Settings

Client System Variables communication driver allows to create Tags that point to system information.

Refer to <u>Client System Variables > Protocol</u> chapter of User's Manual.

CODESYS V2 ETH

CODESYS V2 ETH communication driver for supports communication through Ethernet connection with controllers based on the CODESYS V2.3 version.

Protocol Editor settings

Adding a protocol

To configure the protocol:

- 1. In Config node double-click Protocols.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the PLC list.

The protocol configuration dialog is displayed.

CoDeSys ETH		×
PLC Network		ОК
Alias		Cancel
IP address	0.0.0.0	
Port	1200	
BlockSize	128	
Timeout (ms)	10000	
Protocol type	Level2 -	
Source Address	0	
Destination Address	0	
PLC Models		
Intel Motorola		

Element	Description
Alias	Name identifying nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node.
IP address	Ethernet IP address of the controller.
Port	Port number used by the CODESYS V2 Ethernet driver. The default value is set to 1200 , which is also the default setting of CODESYS-based controllers.
Block Size	Maximum block size supported by your controller (limit is 1024 KB).

Element	Description		
Timeout (ms)	Time delay in milliseconds between two retries of the same message when communication fails.		
Protocol type	Protocol variant to be used. Please make sure you check which protocol variant is supported by the CODESYS runtime you want to connect.		
Source Address, Destination Address	Available only when TCP/IP Level 2 Route is selected in Protocol Type . The Destination is the node of the PLC and allows the protocol to read variables in a subnetwork. The address is used to read variables when multiple PLCs are connected in a sub-network (serial network) but only one have the Ethernet interface.		
PLC Models	Two PLC models are available. Intel Motorola		
PLC Network	<complex-block></complex-block>		

CODESYS V2 Ethernet driver supports connection to multiple controllers starting from version V1.60.

Note: CODESYS V2 Ethernet driver is recommended when creating projects for the internal controller iPLC CODESYS. To use the CODESYS V2 Ethernet driver with iPLC, configure the IP address of the PLC as localhost (127.0.0.1).

iPLC CODESYS supports communication with CODESYS V2 Ethernet driver with symbol based support starting from V1.55 and above.

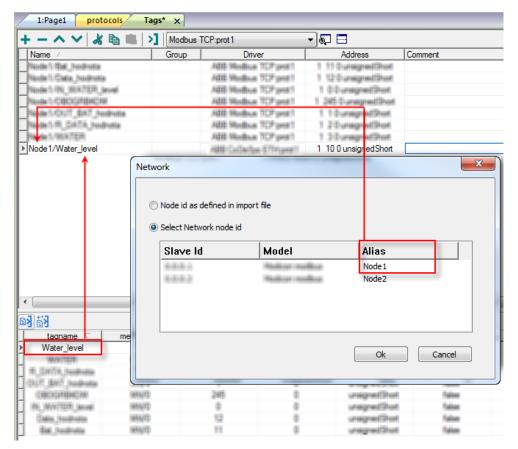
i

Adding an alias name to a protocol

Tag names must be unique at project level, however, the same tag names might need to be used for different controller nodes (for example when the HMI device is connected to two devices running the same application).

When creating a protocol you can add an alias name that will be added to tag names imported for this protocol.

In the example, the connection to a certain controller is assigned the name **Node1**. When tags are imported for this node, all tag names will have the prefix **Node1** making each of them unique at the network/project level.





Note: Aliasing tag names is only available for imported tags. Tags added manually in the Tag Editor cannot have the Alias prefix in the tag name.

The Alias string is attached at the time of tag import. If you modify the Alias string after the tag import has been completed, there will be no effect on names already present in the dictionary. When the Alias string is changed and tags are re-imported, all tags will be re-imported with the new prefix string.

CODESYS software settings

When creating the project in CODESYS, select Download symbol file.

Target Settings		x		
Configuration: EXOR/SITEK CoDeSys for JMobileWCE (WCE/ARM UN31)				
Target Platform Memory Layout Gener	Target Platform Memory Layout General Network functionality Visualization			
- 1/0-Configuration				
Con <u>f</u> igurable				
	No <u>a</u> ddress check			
	Download <u>symbol</u> file symbol coming from two me			
Byte addressing mode	PLC Browser	Load bootproject automatically		
	✓ Irace			
✓ <u>O</u> nline Change	Cycle independent forcing	🗖 Retain for <u>c</u> ing 🔲 S <u>a</u> ve		
Update unused I/Os				
		Default OK Cancel		

Note: CODESYS V2 Ethernet communication driver supports the automatic symbol file (SDB) upload from the PLC; any change in the tag offset due to new compilation of the PLC program does not require a symbol file re-import. Tag file has to be re-imported only in case of tag rename or definition of new tags.

When the option **Download symbol file** is not available or cleared, the protocol can work only if the **ProjectId** tag is imported. If the tag offset changes because of a new compilation of the PLC program, the symbol file must be re-imported.

Name	Gro	oup Driv	ver	Address	Enc
ProjectId		CODESYS	V2 ETH:p 0 Pro	jectld PROJ_ID 14784	
蓜					
	1-1-1	1			
tagname	datatype	array	arraysiz		offset
rojectId	unsignedInt	false	0	PROJ_ID	147842
March	CONTRACTOR OF STREET				-98
Mann	CONTRACTOR OF	1000	-		

Data types

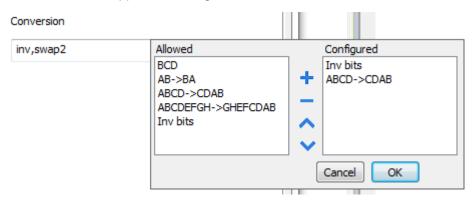
i

The import module supports variables of standard data types and user defined data types.

_						
Supported	• BOOL					
data types	• WORD					
	• DWORD					
	• INT					
	• UINT					
	UDINT					
	• DINT					
	STRING *					
	• REAL					
	• TIME					
	DATE & TIME					
	and 1-dimensional ARRAY of the types above. See "Programming concepts" section in the main manual.					
	Note *: String length for a STRING variable in PLC should be max 80 characters. Declare a STRING variable either with a specified size (str: STRING(35) or default size (str: STRING) which is 80 characters.					
Unsupported	• LWORD					
data types	• LINT					
	• LREAL					

Tag conversion

Conversion to be applied to the tag.



Depending on data type selected, the list **Allowed** shows one or more conversion types.

Value	Description
Inv bits	inv: Invert all the bits of the tag.
	Example: $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)
Negate	neg : Set the opposite of tag value.
	<i>Example:</i> 25.36 → -25.36
AB -> BA	swapnibbles: Swap nibbles in a byte.
	Example: $15D4 \rightarrow 514D$ (in hexadecimal format) $5588 \rightarrow 20813$ (in decimal format)
ABCD -> CDAB	swap2: Swap bytes in a word.
	Example: 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)
ABCDEFGH ->	swap4: Swap bytes in a double word.
GHEFCDAB	Example: $32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format) $855441236 \rightarrow 1426062386$ (in decimal format)
ABCNOP ->	swap8: Swap bytes in a long word.
OPMDAB	Example: 142.366 → -893553517.588905 (in decimal format) 0 10000000110 0001110010111011001000101101
	→ 1 10000011100 101010000101000101101101100101101100001001111
BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9)
	Example: $23 \rightarrow 17$ (in decimal format) $0001 \ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)

Select conversion and click +. The selected item will be added to list Configured.

If more conversions are configured, they will be applied in order (from top to bottom of list **Configured**).

Use the arrow buttons to order the configured conversions.

Node Override IP

The protocol provides the special data type Node Override IP which allows you to change the IP address of the target controller at runtime.

This memory type is an array of 4 unsigned bytes, one per each byte of the IP address.

The Node Override IP is initialized with the value of the controller IP specified in the project at programming time.

Node Override IP	PLC operation
0.0.0.0	Communication with the controller is stopped, no request frames are generated anymore.
Different from 0.0.0.0	It is interpreted as node IP override and the target IP address is replaced runtime with the new value.

If the HMI device is connected to a network with more than one controller node, each node has its own Node Override IP variable.



Note: Node Override IP values assigned at runtime are retained through power cycles.

Hostname DNS or mDNS

In addition to the array of bytes, string memory type can be selected to be able use the DNS or mDNS hostname as an alternative to the IP Address.

DDESYS V2 ETH				
Address Space Node Override IP	Tag Name		Offset	
SubIndex Data Type 0 v BYTE[]	-	Arraysize 4		
Conversion	+/-			
		ок	Annulla Ap	plica ?

Tag Import

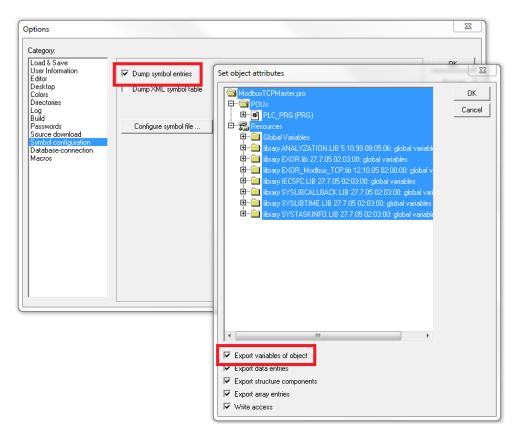
Exporting Tags from PLC

When configuring PLC using the manufacturer's configuration software, enable Symbol file (.sym extension) creation under the CODESYS programming software:

- 1. In the **Project** menu, click **Options**.
- 2. Click Symbol configuration.
- 3. Select Dump symbol entries.
- 4. Click OK.

i

Note: Click then **Configure symbol file...** and select **Export variables of object**. We recommend to clear the check box and re-select to be sure about the proper settings.



In some cases, duplication of symbols for variables associated to integrated I/O modules in the ".sym" file may be experienced. To remove the duplication selected the "PLC Configuration" voice from the objects list and uncheck the option "Export variables of object".

	oles iable
<	•
Export variables of object	
L Export data entries	
Export structure components	
Export array entries	
✓ Write access	

Importing Tags in Tag Editor

Select the driver in Tag Editor and click on the **Import Tags** button to start the importer.

	Tags	×								
+	_	Z	đ	ß	>]	₽	A 9B	B>	ŧ i ł	1
Data			^		-	Та	g URI			_

The following dialog shows which importer type can be selected.

HMIStudio	×
Multiple tag import	ers are available for this protocol. Please select the importer type and continue.
Version	Туре
CODESYS2 sym v1.1	Linear
CODESYS2 sym v1.1	Hierarchical
CODESYS2 sym_xml v1.0	Hierarchical
Tag Editor exported xml	General
	OK Cancel

Importer	Description		
CODESYS2 sym v1.1 Linear	Requires a .sym file.		
	All variables will be displayed at the same level.		
CODESYS2 sym v1.1	Requires a .sym file.		
Hierarchical	All variables will be displayed according to CODESYS V2 Hierarchical view.		
CODESYS2 sym_xml	Requires a .sym_xml file.		
v1.0 Hierarchical	All variables will be displayed according to CODESYS V2 Hierarchical view.		
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.		
	Tags × + → Image: Second seco		

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags × Protocols					
+ - 🎽 🕲 🔰	() 🗞 📾 🖏	R 🔎 - Search Tilter by: Data	▼ Ite	ms used:6/10000 Protocol: Show	al 🛛 🗹 Show all tags 💮 🗔
Data	Type	Comment	^	Property	Value
Modbus TCP:prot1	Container			✓ Driver	
Model: Modicon Modbus(1-based				Model	Modicon Modbus(1-based)
- Holding Registers 1	unsignedShort			Protocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			✓ Dictionary	
 Holding Registers 3 MRTU1 	unsignedShort unsignedShort			Array	false
-MRTU2	unsignedShort			Array size	0
MRTU3	unsignedShort			Arravindex.Subindex	400003
- MRTU4	unsignedShort			Comment	
- MRTU5	unsignedShort			Data type	unsignedShort

Toolbar item	Description
₽ X	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
「「」	Update Tag(s). Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag. Example of both checked and unchecked result:
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Exporting tag arrays

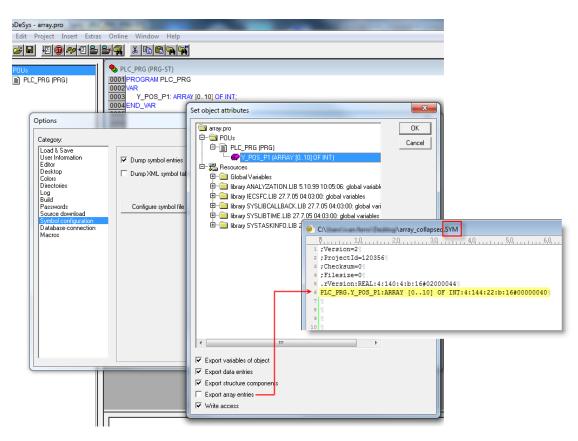
In CODESYS V2 program tag arrays are split into individual elements and one tag for each element is created. In the following example one array with 10 elements.

DeSys - array.pro Edit Project Insert Extras Online Window Help Image: Second	
POUs Open Comparison PIC_PRG (PRG) 0001 PROGRAM PLC_PRG 00002 VAR 00002 VAR 00003 Y_POS_P1: ARRAY [010] OF INT; 00004 END_VAR Set object attributes	
POUs Open Comparison PIC_PRG (PRG) 0001 PROGRAM PLC_PRG 00002 VAR 00002 VAR 00003 Y_POS_P1: ARRAY [010] OF INT; 00004 END_VAR Set object attributes	
Category: Image: Configure symbol entries Desktop Dump symbol entries Desktop Dump symbol entries Buda wordt Dump symbol file Swide Source Sources Image: Configure symbol file Image: Symbol file Image: Configure symbol file Image: Symbol file Image: Configure symbol file Image: Symbol file Image: Configure symbol file Image: Configure sy	



Note: If **Export array entries** is selected, a tag for each element will be created and exported into the .sym file. The entire tag list will be automatically imported into the Tag editor.

By clearing **Export array entries** only one tag for each one array can be created.





Note: When **Export array entries** has been cleared, only one tag is created and exported into the .sym file. The array is not automatically imported in the Tag editor and tags need to be manually configured in Tag editor.

+ - ^ ∨ ≵ ₪ ■ > Name PLC_PRG/Y_POS_P1	CoDeSys ETH:prot 1 Group	Driver CoDeSys ETH:prot1	Address	Comment			
	Group			Comment	1		
PLC_PRG/Y_POS_P1		CoDeSys ETH:prot1	A DUC DDC AV DOC D1 4 14		R/W	Active	T
			UPLC_PRG/T_POS_PT 4 14		R/W	false	Va
	CoDeSys ETH						
	CoDeSys ETH Address Space Local/Global Var	Tag Name PLC_PRG/Y_POS_P1	Offset				
	SubIndex Data Typ	e Arraysize	C.\ 	2,0,,3,0,,4,0,,,,		<u>6</u> 0	
< tagname ∧ //Version PLC PRG/Y POS P1(0)	Conversion	+/-	<pre>4 ;Filesize=0¶ 5 .rVersion:REAL:4:</pre>	140:4:b:16#02000044 ARRAY [010] OF INT:4:144:2	2:b:16#0000	0401	
PLC_PRG/Y_POS_P1[10] PLC_PRG/Y_POS_P1[11] PLC_PRG/Y_POS_P1[2] PLC_PRG/Y_POS_P1[3] PLC_PRG/Y_POS_P1[4]	short	OK CK false		148 4 150 4 152			l

All tag elements can be referenced in the editor using TagIndex in the Attach to Tag dialog.

ld1.value	
Tag)	KForms
Source:	
Tag	🔘 System 🔘 Widget 🔘 Recipe
Tag:	PLC_PRG/Y_POS_P1
	CoDeSys ETH:prot1
	PLC_PRG/Y_POS_P1
Read	d Only Read/Write Write Only TagIndex:
	OK Cancel

Communication status

Current communication status can be displayed using system variables. See "System Variables" section in the main manual.

Codes supported by this communication driver:

Error	Cause and action
Symbols file not present	Check Symbol file and download again the PLC program.
"tag" not present in Symbols files	Check if the Tag is present into the PLC project.
Time out on Acknoledge	Controller didn't send acknowledge.
Time out on last Acknoledge	Controller didn't sent last ack.
Time out on data reciving	Controller does not reply with data.
Connection timeout	Device not connected.

CODESYS V2 SER

The CODESYS V2 SER communication driver has been designed for serial communication with controllers based on CODESYS V2.3.

Please note that changes in the controller protocol or hardware, which may interfere with the functionality of this driver, may have occurred since this documentation was created. Therefore, always test and verify the functionality of the application. To accommodate developments in the controller protocol and hardware, drivers are continuously updated. Accordingly, always ensure that the latest driver is used in the application.

Limitations

Max block size is 1024 byte.

Protocol Editor Settings

Add (+) a driver in the Protocol editor and select the protocol called "CODESYS Serial" from the list of available protocols.

CODESYS V2 SER	×
	Comm OK
Alias	Cancel
BlockSize	128
Timeout (ms)	10000
Num of repeats	2
PLC Models	
Intel Motorola	

Element	Description
Alias	Name to be used to identify nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node
Block Size	Enter the max block size supported by your controller (limit is 1024)
Timeout	The number of milliseconds between retries when communication fails
Num of repeats	This parameter defines the number of times a certain message will be sent to the controller before reporting the communication error status.
	A value of 1 for the parameter "No of repeats" means that the panel will eventually report the communication error status if the response to the first request packet is not

Element	Description		
	correct.		
PLC Model	Defines the byte order that will be used by the communication driver when sending communication frames to the PLC Comm Parameter Dialog Port Com1 Parity none I 19200 Stop bits 1		
Port	Serial port selection.		
	 COM1 is the PLC port. COM2 is PC/Printer port on panels with 2 serial ports or refers to the optional plug-in module plugged in Slot 1/2 for panels with 1 serial port on-board. COM3 refers to the optional plug-in module plugged in Slot 3/4 for panels with 1 serial port on-board. 		
Baudrate, Parity, Data bits, Stop bits	Communication parameters for the serial line.		

CODESYS Software Settings

When creating the project in CODESYS, the option Download Symbol File (in Target Settings/General) must be checked.

Target Settings			×
Configuration: EXOR/SITEK CoDeSys	for JMobileWCE (WCE/ARM UN31)	•	
Target Platform Memory Layout Genera	al Network functionality Visualizatio	n	
- 1/O-Configuration			
Configurable			
	No <u>a</u> ddress chec	k	
	Download <u>symbol</u> file Symbol coning non-nyr nie		
Byte addressing mode	✓ PLC Browser ✓ Irace	Load <u>b</u> ootproject automatically	
🔽 <u>O</u> nline Change	<u>Cycle independent forcing</u>	🗖 Retain for <u>c</u> ing 🔲 S <u>a</u> ve	
Update unused I/Os			
		Default OK Ca	ncel

Note: CODESYS Serial communication driver supports the automatic symbol file (SDB) upload from the PLC; any change in the tag offset due to new compilation of the PLC program does not require a symbol file reimport. Tag file has to be re-imported only in case of tag rename or definition of new tags.

When the option Download symbol file is not available or not checked, the protocol can work only if the ProjectId tag is imported. Any change in the tag offset due to new compilation of the PLC program requires that symbol file is imported again.

+ - ^ ~	🔏 🖻	• >	CODESYS	V2 SER:p	prot1		ş 🔤	
Name		Group	Driv	er		Address		Encoding
ProjectId			CODESYS	V2 SER:p	0 Projectio	PROJ_ID 14	47842 UE	
	1							
•								
8ā Ka								
tagname	dataty	pe	агтау	an	raysize	addressSp	ace	offset
ProjectId	unsignedIn	t fals	e	0		PROJ_ID	147	7842
199603-1000	100000000000000000000000000000000000000	1999 100					100	
19882	- maintent	1.10	£1					
- HARRY	- TONATORINA	1.10	E	-			10.0	

Standard Data Types

The following data types in the CODESYS programming tool are considered standard data types by the import module:

BOOL

i

WORD

DWORD

INT

UINT

UDINT

DINT

STRING

REAL

TIME

DATE & TIME

and 1-dimensional ARRAY of the types above.

The 64-bit data types LWORD, LINT and LREAL are not supported.

String length for a STRING variable in PLC should be max 80 characters. Declare a STRING variable either with a specified size (str: STRING(35)) or default size (str: STRING) which is 80 characters.

Tag Conversion

Conversion to be applied to the tag.

Conversion		
inv,swap2	Allowed	Configured
	BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits	+ ABCD->CDAB
		Cancel OK

Depending on data type selected, the list **Allowed** shows one or more conversion types.

Value	Description
Inv bits	inv : Invert all the bits of the tag.
	Example: $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)
Negate	neg : Set the opposite of tag value.
	<i>Example:</i> 25.36 → -25.36
AB -> BA	swapnibbles: Swap nibbles in a byte.
	Example: $15D4 \rightarrow 514D$ (in hexadecimal format) $5588 \rightarrow 20813$ (in decimal format)
ABCD -> CDAB	swap2: Swap bytes in a word.
	Example: 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)
ABCDEFGH ->	swap4: Swap bytes in a double word.

Value	Description
GHEFCDAB	Example: $32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format) $855441236 \rightarrow 1426062386$ (in decimal format)
ABCNOP -> OPMDAB	swap8: Swap bytes in a long word. Example: 142.366 \rightarrow -893553517.588905 (in decimal format) 0 10000000110 0001110010111011001000101101000011100100101
BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)

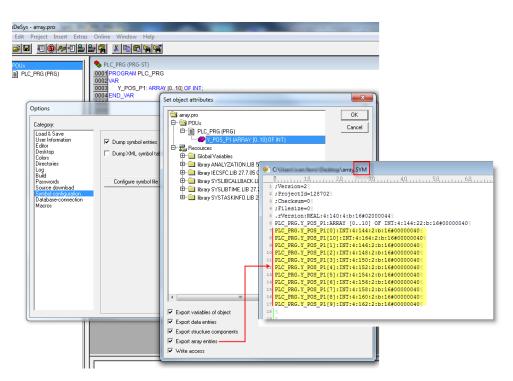
Select conversion and click +. The selected item will be added to list Configured.

If more conversions are configured, they will be applied in order (from top to bottom of list Configured).

Use the arrow buttons to order the configured conversions.

Tag Array

Tag Arrays are split into individual elements and one Tag for each element is created. The figure below shows an example of one Array with 10 elements

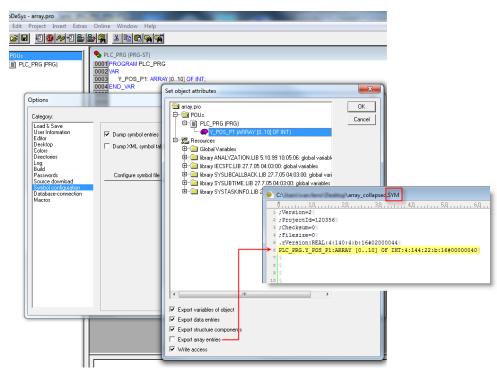


1

1

Note: When "Export array entries" is set, a tag for each element is created and exported into the SYM file. The entire tag list is automatically imported into Tag Editor.

The amount of tags can be reduced and only one Tag for each one array can be created by removing the checkbox "Export array entries", see figure below.



Note: When "Export array entries" is not set, only one tag is created and exported into the SYM file. The Array will not be automatically imported in Tag Editor and Tags need to be manually configured in Tag Editor

tagname / //Version PLC_PRGX/POS_P1[0] 0 PLC_PRGX/POS_P1[1] 0K C 148 PLC_PRGX/POS_P1[2] shot FLC_PRGX/POS_P1[3] shot FLC_PRGX/POS_P1[4] shot false 0 4 152	1:Page1 protocols*	Tags* ×				
Name Group Driver Address Connent R.W Active > PLC_PRG/Y_POS_P1 CoDeSys ETH CoDeSys ETH Image: CoDeSys ETH Image: CoDeSys ETH CoDeSys ETH CoDeSys ETH Image: CoDeSys ETH Image: CoDeSys ETH CoDeSys ETH Image: CoDeSys ETH Image: CoDeSys ETH Image: CoDeSys ETH Image: CoDeSys ETH CoDeSys ETH Image: CoDeSys ETH Image: CoDeSys EtH Image:		CoDeSys ETH:prot1	- 52 🗖			
CoDeSys ETH Address Space Tag Name Iccal/Globel Var PLC_PRG/Y_POS_P1 144 Itag Subindex Data Type Image: State of the state o			Driver	Address	Comment	R/W Active
CoDeSys ETH Address Space Tag Name Orfset Local/Global Var PLC_PRG/Y_POS_P1 144 Image: Statistic p2 Image: Statistic p2 Image: Statistic p2 Image: Statistic p	▶ PLC_PRG/Y_POS_P1		CoDeSys ETH:prot1	0 PLC_PRG/Y_POS_P1 4 14		R/W false
	Image: bit is a constraint of the constrain	CoDeSys ETH Address Space Local/Global Var • SubIndex 0 • Conversion	PLC_PRGYY_POS_P1 Pe Arraysze 1 10 +f OK Talse	144 0 1 0 1 1 2 10 2 10 2 10	\array_collapsed 5YM2030,40	
	PLC_PRG/Y_POS_P1[4]	short	false	0	4 152	

All Tag elements can be referenced in the editor using "TagIndex" in the "Attach to Tag" dialog

Tag :	Scale XForms	
Source:		
Tag	🔘 System 🔘 Widget 🔘 Recipe	
Tag:	PLC_PRG/Y_POS_1	٩
	CODESYS V2 SER:prot1	
	PLC_PRG/Y_POS_1	
Read	l Only 🔘 Read/Write 🔘 Write Only	TagIndex: 0
		OK Cancel

Aliasing Tag Names in Network Configurations

Tag names must be unique at project level; it often happens that the same tag names are to be used for different controller nodes (for example when the HMI is connected to two devices that are running the same application). Since tags include also the identification of the node and Tag Editor does not support duplicate tag names, the import facility in Tag Editor has an aliasing feature that can automatically add a prefix to imported tags. With this feature tag names can be done unique at project level.

The feature works when importing tags for a specific protocol. Each tag name will be prefixed with the string specified by the "Alias".

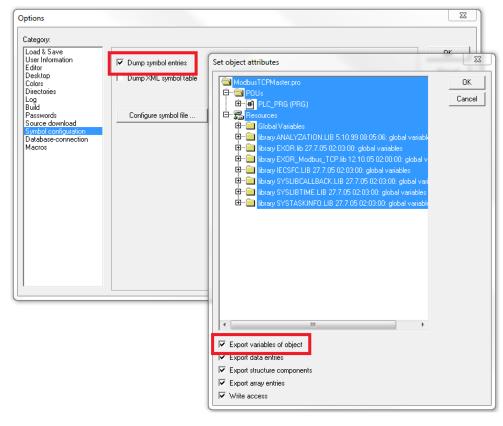


i

Note: An Aliasing tag name is only available when tags can be imported. Tags which are added manually in the Tag Editor do not need to have the Alias prefix in the tag name. The Alias string is attached to the tag name only at the moment the tags are imported using Tag Editor. If you modify the Alias string after the tag import has been completed, there will be no effect on the names already present in the dictionary. When the Alias string is changed and tags are imported again, all tags will be imported again with the new prefix string.

Tag Import

When configuring PLC using the manufacturer's configuration software, make sure to enable Symbol file creation (file with .SYM extension). It can be done under the CODESYS programming software, by selecting "Project\Option\Symbol configuration" and mark the check box "Dump symbol entries" as shown in the picture below.



Note: Click then on the "Configure symbol file..." button and make sure the "Export variables of object" check box is marked as shown in the following picture. We recommend to un-check the check box and mark it again to be sure about the proper settings.

Select the driver in the Studio tag editor and click on the "Import tag" button to start the importer.

ಮ	ProjectView 📮 🗙	1:Page1 protocols Tags ×	
	+ - ₫ ^ ∨	🕂 — 🔨 🖌 🐚 📖 🖂 CoDeSys ETH	prot 1 🗸 🚽 💭
ObjectView	⊡	Name Grou	
ŝ.			
ew	🖕 🖓 Pages		
	1 : Page1		
	🖕 🗁 Config		×
	Protocols	Tag Import	
	🛅 Tags		
		Shite Strategy	
	🔔 Alarms	Select Controller	CoDeSys ETH 👻
	📌 Events Buffer		
	Scheduler	Select Import Type	*.sym 👻
	😡 MultiLanguage		
	Security		
	🖳 🙀 UserGroups	ОК	Cancel
	Users		
	🖃 🦳 AuditTrail		
	📄 🖻 AuditTesil		

Locate the ".sym" file and confirm.

The tags present in the exported document are listed in the tag dictionary from where they can be directly added to the project using the add tags button as shown in the following figure.

Kā Ka							
tegname	memorytype	arrayindex.subin	index	datatype	array	arraysize	
str	MW0	8	0	string-16	true	16	
ARRAY_WORD[1]	MW0	0	0	unsignedShort	false	0	
ARRAY_WORD[2]	MW0	1	0	unsignedShort	false	0	
ARRAY_WORD[3]	MW0	2	0	unsignedShort	false	0	
ARRAY_WORD[4]	MW0	3	0	unsignedShort	false	0	
MDW2	MD0	2	0	unsignedInt	false	0	
MDW3	MD0	3	0	unsignedInt	false	0	

Communication Status

The current communication status can be displayed using the dedicated system variables. Please refer to the User Manual for further information about available system variables and their use.

The codes supported for this communication driver are:

Error	Notes
Symbol file not present	Check Symbol file and download again the PLC program
"tag" not present in Symbol file	Check if the Tag is present in the PLC project
Time out on Acknowledge	Controller didn't send acknowledge
Time out on last Acknoledge	Controller didn't send last acknowledge
Time out on data receiving	Controlled does not reply with data
Connection timeout	Device not connected

CODESYS V3 ETH

The CODESYS V3 ETH communication driver supports communication thought Ethernet connection with controllers based on the CODESYS V3 PLC software by the company 3S.



Note: To accommodate developments in the controller protocol and hardware, drivers are continuously updated. Make sure the latest driver is used in the application.



Note: Changes in the controller protocol or hardware may have occurred since this documentation was created. This may interfere with the functionality of this driver. Therefore, always test and verify the functionality of the application.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the **PLC** list.

The protocol configuration dialog is displayed.

CODESYS V3 ETH			Х
PLC Network		ОК	
Alias		Cancel	
Protocol	TCP ~		
IP address	0.0.0.0		
Timeout (ms)	1000		
Variable list count	255		
PLC Models			
CODESYS 3			

Element	Description	
Alias	Name to be used to identify nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node.	
Protocol Select between TCP and UDP protocol type.		
IP	Ethernet IP address of the controller	

Element	Description						
address							
Variable list count	Variable List is the best method to achieve higher performance in the CODESYS V3 communication protocol, as it allows requesting multiple data items in a single protocol session.						
	Since some implementations of CODESYS V3 at runtime have a limited number of Variable Lists that can be allocated, this parameter allows you to set the maximum number of Variable Lists the communication driver tries to create in the PLC.						
PLC Model	Byte order that will be used by the communication driver when sending communication frames to the PLC.						
Timeout	Number of milliseconds between retries when communication fails.						
PLC Network	Enable access to multiple networked controllers. For every controller (slave) set the proper option.						

i

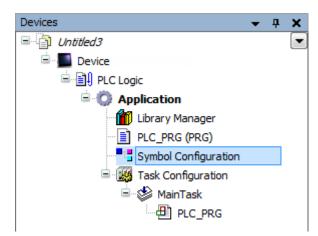
Note: Refer to the controller documentation to verify required values for the parameters **Full node address** or **Variable list count**.

Tag Import

Exporting Tags from PLC

When creating the project using CODESYS V3, properly configure the symbol file to contain the required variables.

1. To add the Symbol configuration in CODESYS V3 project, right click on the Application item from the project tree, then into the context menu select Add Object > Symbol configuration. The symbol configuration item will be added to the project tree.



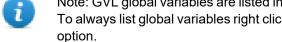
2. Double click on Symbol configuration item, then click on "Build" button.

Symbol Configuration 🗙									
📉 View 📲 🎬 Build 🛛 🛱 Settings 👻									
	Execute "Build" command to be able to select variables (you need an error-free build). Details Changed symbol configuration will be transferred with the next download or online change								
Symbols	Access Rights	Maximal	Attribute	Туре	Members	Comment			

3. Symbol configuration item contains a list of all the variables available into the CODESYS V3 project, single variables or groups of variables can be selected by checking the corresponding item in the list.

changed symbol configuration will be transf	erred with the next of	lownload or o	online change			
Symbols	Access Rights	Maximal	Attribute	Туре	Members	Comment
🗐 🗐 Constants						
👘 📄 🧳 CompilerVersion		*		VERSION		Does the target support an FPU
🔄 🖉 RuntimeVersion		~ @		VERSION		Does the target support an FPI
🗟 🐨 📄 GVL						
🔤 📝 🔶 MyVARGlobal1	*	*		INT		
📝 🔌 MyVARGlobal2		*		INT		
=						
📃 🔌 nIoConfigTaskMapCount		*		DINT		
📃 🔌 pIoConfigTaskMap		*		POINTER TO IoConfigTaskMap		
- 🔲 📄 PLC_PRG						
wyVAR1	*	*		INT		
myVAR2		N		BOOL		
V 🖉 myVAR3	*	-		REAL		
E BPLog						

4. After the symbols have been configured, download the project or use the Generate code function (Build > Generate code) to create an .xml file containing all the variables read to be imported in the Tag Editor.



Note: GVL global variables are listed in Symbols Configuration only if they are used in PLC program. To always list global variables right click on GVL and select "Properties". From "Build" tab check "Link Always" option.

Properties - GVL [Device: PLC Logic: Application]
Common Link To File Access control Network properties Build
Exclude from build
External implementation (Late link in the runtime system)
Enable system call
🕼 Link Always
Compiler defines:
OK Cancel Apply

Importing Tags in Tag Editor

Select the driver in Tag Editor and click on the Import Tags button to start the importer.



The following dialog shows which importer type can be selected.

ŀ	HMIStudio	×
	Multiple tag impo	orters are available for this protocol. Please select the importer type and continue.
	Version	Туре
	CODESYS3 xml v1.0	Linear
	CODESYS3 xml v1.0	Hierarchical
	Tag Editor exported xm	l General
		OK Cancel

Importer	Description		
CODESYS3 xml v1.0	Requires an .xml file.		
Linear	All variables will be displayed at the same level.		
CODESYS3 xml v1.0	Requires an .xml file.		
Hierarchical	All variables will be displayed according to CODESYS V3 Hierarchical view.		
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.		
	Tags x		
	+ - 🕺 🔞 🗈 >] [> 🔩 🖻 📅 🗉		
	Data Tag URI		

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags × Protocols					
+ - 👗 🕲 🖉 🔰	[> \$ _B ₪] ∰	R 🔎 - Search Tilter by: Data	▼ Ite	ems used:6/10000 Protocol: Show all	Show all tags 💮 🗔
Data	Туре	Comment	^	Property	Value
Modbus TCP:prot1	Container			Y Driver	
Model: Modicon Modbus(1-bas	ea)			Model	Modicon Modbus(1-based)
- Holding Registers 1	unsignedShort			Protocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			✓ Dictionary	
Holding Registers 3	unsignedShort				
- MRTU1	unsignedShort			Array	false
-MRTU2	unsignedShort			Array size	0
MRTU3	unsignedShort			Arrayindex.Subindex	400003
-MRTU4	unsignedShort			Comment	
- MRTU5	unsignedShort			Data type	unsignedShort
	- 1-1 ·			and the	an angi readition e

Toolbar item	Description
∎ <mark>></mark>	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
ăکا ا	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag. Example of both checked and unchecked result:
	Later product of both of collected and an encloced a resolution. Tage Tage Container
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Adding an alias name to a protocol

Tag names must be unique at project level, however, the same tag names might need to be used for different controller nodes (for example when the HMI device is connected to two devices running the same application).

When creating a protocol you can add an alias name that will be added to tag names imported for this protocol.

In the example, the connection to a certain controller is assigned the name **Node1**. When tags are imported for this node, all tag names will have the prefix **Node1** making each of them unique at the network/project level.

+ - ^ ~ * [🗈 📖 >] Ma	odbus TCP:prot1		-) 52 🗔		
Name 🛆	Gro	up	Driver		35	Comment
Node 1/ Eat_hodrotia		/#1212 Wind	Isue TOP point1	1 11 Dunkip	william.	
Node1/Oata_hodrotia		(81212) Wind	bue TCP pet1	1 12 Durwig	with wi	
Node1/IN W/ATER law	val .	(X1222) Wend	bue TCP pet1	1 D.D.umaige	and Short	
Note1/080678HCW		(X1232) William	Isua TCP peak1	1 245/Dumin	nedShort	
Nede1/OUT_BXT_Ned	rulia.	(X4232) (West	Isue TCP pret1	1 1 Durwige	North The	
Node1/R_D/RTA_hode	dia.	/H212 (West	Isue TCP pet1	1 2 Durwige	HorfElia	
No. 6e 1, 100/8,71238		/H212 (West	Isue TCP pet1	1 3 Durwige	HarfElia	
Node1/Water_level		AND CAD	State (114) and 1	1 10 0 unsign	edShort	
		id as defined in im t Network node id	port file			
		ve Id	Model		Alias	
	216	YC IU	MOUCI		MIIdo	
			Model		Node1	
	(6.6			ius-		
taoname Vater_level	(6.6	#-	Holicer mol	ius-	Node1	Cancel
tagname Water_level		#-	Holicer mol	ius-	Node1 Node2	Cancel
taoname Water_level		#-	Holicer mol	ius-	Node1 Node2	Cancel
tagname Water_level		#-	Holicer mol	har har	Node1 Node2	Cancel
tagname Water_level	me		Holicer mol	las las une une	Node1 Node2 Ok	Cancel
Image: state	me		Holicer mol	las las uns uns uns	Node 1 Node 2 Ok	Cancel



Note: Aliasing tag names is only available for imported tags. Tags added manually in the Tag Editor cannot have the Alias prefix in the tag name.

The Alias string is attached at the time of tag import. If you modify the Alias string after the tag import has been completed, there will be no effect on names already present in the dictionary. When the Alias string is changed and tags are re-imported, all tags will be re-imported with the new prefix string.

Data Types

The import module supports variables of standard data types and user defined data types.

Supported data types	 BOOL INT SINT UINT UDINT DINT STRING* REAL LREAL BYTE ULINT LINT
	and 1-dimensional ARRAY of the types above. See "Programming concepts" section in the main manual.
	Note *: String length for a STRING variable in PLC should be max 80 characters. Declare a STRING variable either with a specified size (str: STRING(35) or default size (str: STRING) which is 80 characters.
Unsupported data types	LWORDLINT

Tag conversion

Conversion to be applied to the tag.

Conversion			
inv,swap2	Allowed	_	Configured
	BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits	+ - < >	Inv bits ABCD->CDAB
		(Cancel OK

Depending on data type selected, the list **Allowed** shows one or more conversion types.

Value	Description
Inv bits	inv: Invert all the bits of the tag.
	Example: $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)
Negate	neg: Set the opposite of tag value.
	Example: $25.36 \rightarrow -25.36$
AB -> BA	swapnibbles: Swap nibbles in a byte.
	Example: 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)
ABCD -> CDAB	swap2: Swap bytes in a word.
	Example: 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)
ABCDEFGH ->	swap4: Swap bytes in a double word.
GHEFCDAB	Example: $32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format) $855441236 \rightarrow 1426062386$ (in decimal format)
ABCNOP ->	swap8: Swap bytes in a long word.
OPMDAB	Example: 142.366 \rightarrow -893553517.588905 (in decimal format) 0 10000000110 0001110010111011001000101101
	→ 1 10000011100 1010101000010100010110110110010110110000
BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9)
	Example: $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)

Select conversion and click +. The selected item will be added to list **Configured**.

If more conversions are configured, they will be applied in order (from top to bottom of list Configured).

Use the arrow buttons to order the configured conversions.

Node Override IP

The protocol provides the special data type Node Override IP which allows you to change the IP address of the target controller at runtime.

This memory type is an array of 4 unsigned bytes, one per each byte of the IP address.

The Node Override IP is initialized with the value of the controller IP specified in the project at programming time.

Node Override IP	PLC operation
0.0.0.0	Communication with the controller is stopped, no request frames are generated anymore.
Different from 0.0.0.0	It is interpreted as node IP override and the target IP address is replaced runtime with the new value.

If the HMI device is connected to a network with more than one controller node, each node has its own Node Override IP variable.



Note: Node Override IP values assigned at runtime are retained through power cycles.

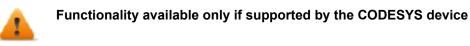
Hostname DNS or mDNS

In addition to the array of bytes, string memory type can be selected to be able use the DNS or mDNS hostname as an alternative to the IP Address.

CODESYS V3 ETH			×
CODESYS V3 ETH			
memType	Data type	Arraysize	
Node override IP V		 ✓ 4 	
Conversion	Tag name		
	+/-		
	ОК	Cancel Apply H	elp

Application Status

The protocol provides the special data type Application Status which allows you to check or change the applications status.



The tags pointing to Application Status must contains into field "**Tag name**" the name of the PLC application (frequently the default name is "Application")

If the HMI device is connected to a network with more than one controller node, each node has its own Application Status variable.

CODESYS V3 ETH			×
CODESYS V3 ETH			
memType	Data type	Arraysize	
Application Status \sim	BYTE	~ 0	
Conversion	Tag nan		
		auon	
	ОК	Cancel Apply Help	
Application Status	Descript	ion	
0	RUNNIN	G	
1	STOPPE	D	
2	HALTED	ON BreakPoint	
	0	It is not possible to write 2 as new status	6
251	Reboot C	ODESYS device	
252	Shutdow	n CODESYS	
253	Reset OF	RIGIN	

Reset COLD

Reset WARM

Communication Status

254

255

Current communication status can be displayed using System Variables. See "System Variables" section in the main manual.

Control Techniques Modbus TCP

Control Techniques Unidrive M Series are using Modbus TCP protocol where the device id should be always set to 0 or 255. This communication protocol is known as Control Techniques Modbus TCP. The HMI protocol identifies Control Techniques Modbus TCP devices using their IP addresses

You should take note of these addresses as you assign them because you will need them later in the set-up phase of the user interface application. The HMI protocol can be set to access to a different menu range

Different physical media, gateways, routers and hubs can be used in the communication network. Also, other devices can independently make simultaneous use of the network. However, it is important to ensure that the traffic generated by these devices does not degrade the communication speed (round-trip time) to an unacceptable level.

The implementation of the protocol operates as a Modbus TCP client only.

The HMI Control Techniques Modbus TCP protocol uses the standard port number 502 as the destination port.

The HMI Control Techniques Modbus TCP protocol supports the standard commonly referred as "Ethernet II".

Protocol Editor Settings

Add (+) a new driver in the Protocol editor and select the protocol called "Control Techniques Modbus TCP" from the list of available protocols.

The driver configuration dialog is shown in figure.

Control Techniques Modbu	is TCP	Х
PLC Network	ОК	
Alias	Cancel	
IP address	0.0.0.0	
Port	502	
Modbus Mode	Standard ~	
Timeout (ms)	2000	
PLC Models		
default		

Element	Description				
Alias	Name to be used to identify nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node				
IP address	Ethernet IP address of the controller				
Port	Port number used by the Modbus TCP driver; the default value can be changed when the communication goes through routers or Internet gateways where the default port number is already in use				
Modbus Mode	This parameter define the communication protocol used and needs to be set in according with the setting made on the drive (parameter S.15.013). Modified mode is provided to allow register numbers up to 255 to be addressed. If any menus with numbers above 63 should contain more than 99 parameters, then these parameters cannot be accessed via Modbus.ProtocolRegister address				
	Standard	(menu number * 100) + parameter number - 1 where menu number ≤ 162 and parameter number ≤ 99			
	Modified(menu number * 256) + parameter number – 1where menu number ≤ 63 and parameter number ≤ 255				
Timeout (ms)		Defines the time inserted by the protocol between two retries of the same message in case of missing response from the server device. Value is expressed in milliseconds.			

Element	Description						
PLC	Selection of device models that may affect operation of the protocol.						
Models	Currently only one model is available						
PLC Network	The protocol allows the connection of multiple controllers to one operator panel. To set-up multiple connections, check "PLC network" checkbox and enter IP Address for all controllers.						
	Control Techniques Modbus TCP ×						
	PLC Network OK						
	Alias Control Techniques Modbus TCP X						
	IP address			ОК			
	Port	Alias		Cancel			
	Modbus M	IP address	0.0.0.1				
	Timeout (r	Port	502				
	PLC Model	Modbus Mode	Standard ~				
		Timeout (ms)	2000				
		PLC Models default					
	Slaves						
	Slave Id						

Communication Status

The current communication status can be displayed using the dedicated system variables. Please refer to the User Manual for further information about available system variables and their use.

The error codes supported by this communication driver are:

Error	Notes
No response	Returned when a request is not replied within the specified timeout period; ensure the controller is connected and properly configured to get network access
Incorrect node address in response	The panel did receive from the controller a response with invalid node address

Error	Notes
The received message too short	The panel did receive from the controller a response with invalid format
Incorrect writing data acknowledge	Controller did not accept write request; ensure the data programmed in the project are consistent with the controller resources

Delta Modbus RTU

Delta Modbus RTU communication driver has been designed to connect HMI devices to Delta PLC through Serial connection.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the $\ensuremath{\text{PLC}}$ list.

The protocol configuration dialog is displayed.

Delta Modbus RTU					
PLC Network	Comm OK				
Alias	Cancel				
Node ID	1				
Timeout(ms)	1000				
delay	0				
Num of repeats	2				
PLC Models	RTU				
DELTA DVP-PLC					

Element	Description				
Alias	Name identifying nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node.				
Node ID	Serial node associated to the PLC.				
Timeout (ms)	Time delay in milliseconds between two retries in case of missing response from the server device.				
delay	Time delay in milliseconds between the end of the last received frame and the starting of a new request. If set to 0, the new request will be issued as soon as the internal system is able to reschedule it.				
Num of repeats	Number of times a certain message will be sent to the controller before reporting the communication error status.				
	When set to 1 the panel will report the communication error if the response to the first request packet is not correct.				
Transmission Mode	 RTU: use RTU mode ASCII: use ASCII mode Note: When PLC network is active, all nodes will be configured with the same Transmission Mode. 				
PLC Models	PLC model available:				
	DELTA DVP-PLC				

Element	Description	
PLC Network	IP address for all controlle enable multiple connectic	ers in multiple connections. PLC Network must be selected to ons.
	Delta Modbus RTU	23
	PLC Network	Comm OK
	Alias	Cancel
	Node ID -1	Delta Modbus RTU
	Timeout(ms)	OK Cancel
	delay	
	Num of repeats	Node ID 1 Timeout(ms) 1000
	Transmission Mode RTU	
	PLC Models DELTA DVP-PLC	0 0 Num of repeats 2
		PLC Models
	Slaves	Add
	Slave Id Model	Alias
Comm	If clicked displays the con	nmunication parameters setup dialog.
	Comm Parameter Dialog	
		ОК
	Port	com1 🔻
	Baudrate	9600 👻
	Parity	none
	Data bits	8
	Stop bits	1
	Mode	RS-485 -

Element	Description				
	Element	Parameter			
	Port	Serial port selection.			
		• COM1 : On-board port			
		 COM2: Optional Plug-in module plugged on slot#1 or slot#2 			
		 COM3: Optional Plug-in module plugged on slot#3 or slot#4 			
	Baudrate, Parity, Data Bits, Stop bits	Serial line parameters.			
	Mode	Serial port mode. Available modes:			
		• RS-232.			
		• RS-485 (2 wires).			
		• RS-422 (4 wires).			

Tag Editor Settings

In Tag Editor select **Delta Modbus RTU** protocol.

Add a tag using [+] button. Tag setting can be defined using the following dialog:

Delta Modbus RTU		X
Delta Modbus RTU		
Memory Type Input	Offset subindex ▼ 0 0	
Data Type boolean	Arraysize Conversion	
	OK Cancel Apply He	elp

Element	Description						
Memory Type	Memory Type	Description	Description				
l y pc	Input	X resources. Corresponding to internal digital Input point.					
	Output	Y resources. Corresponding to internal digital Output point.					
	Auxiliary Relay	M resources. Corresponding to PLC in	nternal memory.				
	Step Relay	S resources.					
	Timer Contact	T resources.					
	Counter Contact	C resources.					
	Timer Value	TV resources.					
	Counter Value	CV resources.					
	Counter 32bit Value	CV32 resources.					
	Data Register	D resources.					
	Node Override ID	see Special Data Types for mode details					
Offset	Starting address for the Tag. The possible range depend on PLC model selected.						
Subindex	This allows resource offset	selection depending on the selected data t	ype.				
Data Type	Data Type	Memory Space	Limits				
	boolean	1-bit data	0 1				
	byte	8-bit data	-128 127				
	short	16-bit data	-32768 32767				
	int	32-bit data	-2.1e9 2.1e9				
	int64	64-bit data	-9.2e18 9.2e18				
	unsignedByte	8-bit data	0 255				
	unsignedShort	16-bit data	0 65535				
	unsignedInt	32-bit data	0 4.2e9				
	uint64	64-bit data	01.8e19				

Element	Description	escription						
	Data Type		Memory Space	Limits				
			point type	3.4e38				
	double		IEEE double-precision 64-bit floating point type	2.2e-308 1.79e308				
	string		Array of elements containing character code defined b selected encoding					
	binary		Arbitrary binary data					
		e: to define array kets like "byte[]"	rs. select one of Data Type format followe ', "short[]"…	ed by square				
Arraysize	 In case available Note: number of set to UTF-8 or If Encoding pro 	 In case of array tag, this property represents the number of array elements. In case of string tag, this property represents the maximum number of bytes available in the string tag. e: number of bytes corresponds to number of string characters if Encoding property is to UTF-8 or Latin1 in Tag Editor. ncoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one racter requires 2 bytes. 						
Conversion	Conversion to	be applied to the	e tag.					
	Conversion							
	inv,swap2		A ->CDAB EFGH->GHEFCDAB					
	Depending on data type selected, the list Allowed shows one or more conversion ty							
	Value	Description						
	Inv bits	inv : Invert all the bits of the tag.						
			(in binary format)					
		$9 \rightarrow 6$ (in deci	mal format)					

Element	Des

Value	Description
	<i>Example:</i> 25.36 → -25.36
AB -> BA	swapnibbles: Swap nibbles in a byte.
	<i>Example:</i> 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)
ABCD ->	swap2: Swap bytes in a word.
CDAB	<i>Example:</i> 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)
ABCDEFGH	swap4: Swap bytes in a double word.
-> GHEFCDAB	Example: $32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format) $855441236 \rightarrow 1426062386$ (in decimal format)
ABCNOP -	swap8: Swap bytes in a long word.
> OPMDAB	Example: 142.366 → -893553517.588905 (in decimal format) 0 10000000110 0001110010111011001000101101000011100101
	\rightarrow 1 10000011100 1010101000010100010110110110
BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9)
	Example: $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)

Select conversion and click +. The selected item will be added to list **Configured**.

If more conversions are configured, they will be applied in order (from top to bottom of list **Configured**).

Use the arrow buttons to order the configured conversions.

Node Override ID

The protocol provides the special data type Node Override ID which allows you to change the node ID of the slave at runtime. This memory type is an unsigned byte.

The node Override ID is initialized with the value of the node ID specified in the project at programming time.

Node Override ID	Modbus operation
0	Communication with the controller is stopped. In case of write operation, the request will be transmitted without waiting for a reply.
1 to 254	It is interpreted as the value of the new node ID and is replaced for runtime operation.
255	Communication with the controller is stopped; no request messages are generated.

Note: Node Override ID value assigned at runtime is retained through power cycles.

Delta Modbus RTU		×
Delta Modbus RTU		
	Official and the day	
Memory Type	Offset subindex	
Node Override ID 👻	0	
Data Type	Arraysize Conversion	
unsignedByte 👻	0 +/-	
	OK Cancel Apply H	Help

Tag Import

Select the driver in Tag Editor and click on the **Import Tags** button to start the importer.

	Tags	×								
+	-	Z	ß	ß	>]	₽	A 9B	B>	63	1
Data			^		-	Та	g URI			_

The following dialog shows which importer type can be selected.

HMIStudio	×
Multiple tag imp	orters are available for this protocol. Please select the importer type and continue.
Version	Туре
DELTA PLC v1.0	Linear
Tag Editor exported xr	nl General
	OK Cancel

Туре	Description		
DELTA PLC v1.0	Requires a .csv file.		
Linear	All variables will be displayed at the same level.		
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.		
	Tags x + - × © > > + + + + × + + + + + + + + + + + + +		

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags 🗙 Protocols					
+ - 👗 🕲 🖄	D 🗞 🖬	R 🔎 - Search YFilter by: Data	▼ Ite	tems used:6/10000 Protoco	l: Show all tags 🎯 🗔
Data	Туре	Comment	^	Property	Value
Modbus TCP:prot1	Container			Y Driver	
Model: Modicon Modbus(1-bas	ea)			Model	Modicon Modbus(1-based)
- Holding Registers 1	unsignedShort			Protocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			✓ Dictionary	
 Holding Registers 3 	unsignedShort				
- MRTU1	unsignedShort			Array	false
- MRTU2	unsignedShort			Array size	0
MRTU3	unsignedShort			Arrayindex.Subin	dex 400003
- MRTU4	unsignedShort			Comment	
- MRTU5	unsignedShort			Data type	unsignedShort

Toolbar item	Description
₽ X	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
「「「」	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.
	Example of both checked and unchecked result:
	Tops: Image: Image: </th
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Direct Serial

Direct Serial communication driver is a generic protocol that allows low level access to serial functions.

Using this protocol the application itself can realize some serial based protocol (RS-232/485/422) without requirement for a development of a dedicated protocol.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the $\ensuremath{\text{PLC}}$ list.

The protocol configuration dialog is displayed.

Direct Serial	×
	Comm OK
Rx Token Prefix	C0 Cancel
Rx Token Suffix	
Token Gap	20
Tx Token Prefix	
Tx Token Suffix	
Hexadecimal Tokens	s
Token Queue Size	100
PLC Models	
default	

Element	Description	
Rx Token Prefix	Indicates the prefix for read token, as string specified by hexadecimal characters.	
Rx Token Suffix	Rx Token Suffix Indicates the suffix for read token, as string specified by hexadecimal characters.	
Token Gap	en Gap Indicates the period between tokens, in milliseconds.	
Tx Token Prefix	Fx Token Prefix Indicates the prefix for sent token, as string specified by hexadecimal characters.	
Tx Token Suffix Indicates the suffix for sent token, as string specified by hexadecimal characters.		
Hexadecimal	checked = tokens are in hexadecimal	
Tokens	not checked = tokens are not in hexadecimal	
Talan Osan Ola		

Token Queue SizeIndicates the number of tokens in the queue, as an integer value from 1 to 10000 (default: 100)



These parameters are determining the behavior of the driver during RX and TX operations, as defined in next paragraphs. In addition the standard communication parameters are available.



All protocols parameters can be overwritten at runtime using the appropriate memory types, so the complete setup can be achieved during runtime using Tags. Settings using memory types are saved to permanent storage using standard procedures. The "Serial Done" memory type is used in order that all set parameters are transferred to usage at once. If any of the serial parameter is changed the serial driver is re-programmed.

Tag Editor Settings

Path: ProjectView> Config > double-click Tags

- 1. To add a tag, click +: a new line is added.
- 2. Select **Direct Serial** from the protocol list: tag definition dialog is displayed.

Direct Serial					×
Direct Serial					
Memory Type Token To Send	no •	de		Data Type string	•
Arraysize 0	Conversion		+/-		
		ОК	Cancel	Apply	Help

Element	Description				
Memory Type	Name	Datatype	Description		
, Abo	Token To Send	string	Write only. Writing given string to cor	g on this memory type sends the nmunication.	
	Token Received	string		ng from this memory type gets m the receiving queue.	
	Length of Token Received	unsignedInt	Read only. Returns the length in bytes of the fit token from the receiving queue.		
	Tokens Available	unsignedInt	Read only. Gives the number of tokens in the receiving queue.		
	Token Acknowledge	boolean		g to this memory type removes m the receiving queue.	
	Serial Baudrate	unsignedInt	Overrides serial b	audrate parameter.	
	Serial Bits	unsignedByte	Overrides serial bits parameter.		
	Serial Stop Bits	unsignedByte	Overrides serial stop bit parameter.		
	Serial Parity	unsignedByte	Overrides serial parity parameter.		
	Serial Mode	unsignedByte	Overrides serial mode parameter.		
	Rx Token Prefix	string	Overrides protocol parameters. Check " <i>Protoco</i> <i>Editor Settings</i> " from details.		
	Rx Token Suffix	string			
	Token Gap	unsignedInt			
	Tx Token Prefix	string			
	Tx Token Suffix	string			
	Hexadecimal Tokens	boolean			
	Token Queue Size	unsignedInt			
	Serial Done	boolean	values written in tl	mory type transfers all new he other tags to protocol o permanent storage.	
Data Type	Data Type	Memory Space		Limits	
	boolean	1-bit data		01	
	unsignedByte	8-bit data		0 255	

Element	Description					
	Data Type	Memory Space		Limits		
	unsignedInt	32-bit data		0 4.2e9		
	string	Array of elements encoding	s containing character	code defined by selected		
	Note: to d	fine arrays. select one of Data Type format followed by square brackets like "b 				
Arraysize	 size In case of array tag, this property represents the number of array elements. In case of string tag, this property represents the maximum number of bytes availabl string tag. 					
	or Latin1 in Tag Edit	or.	-	coding property is set to UTF-8 I6LE one character requires 2		
Conversion	Conversion to be ap	plied to the tag.				
	Conversion					
	inv,swap2	Allowed	Configured			
		BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits	+ ABCD->CDAB			
	Depending on data	type selected, the list Allowe		conversion types.		
	Value	Description				
	Inv bits	inv : Invert all the bits of th	e tag.			
		Example: $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)				
	Negate	neg: Set the opposite of t	neg: Set the opposite of tag value.			
		<i>Example:</i> 25.36 → -25.36				
		20:00 / 20:00				

Element	Description		
	Value	Description	
		<i>Example:</i> 15D4 → 514D (in hexadecimal format) 5588 → 20813 (in decimal format)	
	ABCD -> CDAB	swap2: Swap bytes in a word.	
		Example: 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)	
	ABCDEFGH ->	swap4: Swap bytes in a double word.	
	GHEFCDAB	Example: $32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format) $855441236 \rightarrow 1426062386$ (in decimal format)	
	ABCNOP -> OPMDAB	swap8: Swap bytes in a long word. Example: 142.366 \rightarrow -893553517.588905 (in decimal format) 0 1000000110 0001110010111011010000101101000011100000	
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 2201 = 1	
		0001 = 1 (first nibble) 0111 = 7 (second nibble) d click +. The selected item will be added to list Configured .	
	If more conversions are configured, they will be applied in order (from top to bottom of list Conf		

Use the arrow buttons to order the configured conversions.

Implementation Details

Receiving algorithm

The protocol applies a separate thread that receives the characters from specified serial port.

When tokens (substrings) are identified they are put into the receiving queue (as strings).

Both ASCII and binary mode are available. When binary data can be present into receiving stream the **Hexadecimal Tokens** parameter can be set. In this case tokens are stored in queue using hex string coding (each byte is stored using two chars representing the hex value 0 to F). When defining the tags used to read tokens the appropriate string length should be computed considering the binary mode.

The **Token Queue Size** parameter specifies the maximum number of tokens saved into the queue. When the queue becomes full the oldest token is discarded.

The token identification is as follows:

- if the parameters specify a rx-prefix all characters before detecting the prefix are ignored
- if protocol specifies a rx-suffix it is used to detect the token end
- if rx-suffix is specified the parameter 'gap' specifies the timeout after which the token receiving is restarted
- if rx-suffix is not specified the parameter gap specifies the timeout that terminates the token (anything received up to this interval). If within this time the rx-prefix is detected again the token is ended and stored and reception of a new token is started

In summary we can have four combinations:

- a. No rx-prefix and rx-suffix: the incoming stream is divided in tokens according to gap detection
- b. Rx-prefix specified but no suffix: all the received chars before prefix are ignored. All the chars after prefix are stored in a token till the gap detection
- c. Rx-prefix and Rx-suffix specified: all the chars between prefix and suffix are stored in a token. All the chars received before prefix or after suffix till the gap detection or till a new prefix are ignored
- d. Rx-suffix specified but not RX-prefix: all the chars received till suffix are stored in a token. All the chars received after suffix till the gap detection are ignored

The rx-prefix and rx-suffix parameters are specified as hex strings, so any characters can be specified (like DLE STX CR LF etc...). i.e. to define the string "STR" as prefix the string "535452" must be used.

Before putting string to the receiving queue the prefix and suffix are removed (only 'payload' saved).

Transmission algorithm

The strings to be transmitted are prepared adding the "Tx-prefix" in front and the "Tx-suffix" in the end, if defined. Then the whole string is transmitted immediately.

Interface to user project

Reading a tag defined as **Token Received** gets the front string from the queue. If there are no new tokens an empty string is returned.

Reading a tag defined as Length of Token Received gets the length in bytes of the token.

Reading a tag defined as Tokens Available gets the number of tokens currently stored in the queue.

Writing to a tag defined as **Token Acknowledge** removes the token from queue and makes available the next token if present.

Writing to a tag defined as Token To Send means immediate sending, without any queue used.

JavaScript Interface

Beside Tag interface the user can access the protocol via JavaScript.

Although defined Tags can be accesses by JavaScript too, JavaScript can access directly to a Command interface implemented in protocol. This interface does not require the definition of Tags and is direct to protocol resulting in more efficiency.

This interface provides the access to token queue and sending function. The following commands are supported:

Command	Description	
put	Put the token to send contained in string parameter.	
get	Get the received token.	
get_token_length	ength Get the length of received token.	
tokens_available Get number of tokens received.		
token_ack	Acknowledge reading token.	

Using the command interface the following JS code should receive data:

```
var tagMgr = project.getWidget("_TagMgr");
var protID = "prot2"; // to be set according to protocol numbering
var avail = tagMgr.invokeProtocolCommand(protID, "tokens_available", "");
while (parseInt(avail) > 0)
{
        var str = tagMgr.invokeProtocolCommand(protID, "get", ""); // get the next
token
        var status = tagMgr.invokeProtocolCommand(protID, "token_ack",""); //
acknowledge current token
        avail = tagMgr.invokeProtocolCommand(protID, "tokens_available",""); // get
number of available tokens in queue
}
```

VCS access

The protocol supports the remote (virtual com port) access in exclusive mode.

When VCS is enabled the serial line usage is suspended and serial line becomes available for remote user. At the end the protocol is restarted. The content of the token queue is lost.

Tag Import

Select the driver in Tag Editor and click on the Import Tags button to start the importer.



Locate the Tag Editor Exported symbol file and click **Open**.

Tags included in the symbol file are listed in the tag dictionary. The tag dictionary is displayed at the bottom of the screen.

Tags 🗙 Protocols						
+ - 👗 🕲 🖉 🔰	D & D D	R 🔎 - Search 🍸 Filter by: Data	▼ Ite	ems used:6/	/10000 Protocol: Show	all 🛛 🗹 Show all tags 🔅 🗖
Data	Туре	Comment	^	Property		Value
Modbus TCP:prot1	Container			✓ Driver	r	
Model: Modicon Modbus(1-bas	ea)			M	Iodel	Modicon Modbus(1-based)
- Holding Registers 1	unsignedShort			P	rotocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			✓ Diction	narv	
 Holding Registers 3 MRTU1 	unsignedShort unsignedShort			A	rrav	false
- MRTU2	unsignedShort			Α	rray size	0
MRTU3	unsignedShort				rravindex.Subindex	400003
- MRTU4	unsignedShort				omment	100000
- MRTU5	unsignedShort			-	ata type	unsignedShort
					ata type	unaigneubriorit

Toolbar item	Description
	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
Rā	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.
	Example of both checked and unchecked result:
	Tags* x + - X > > > > > > > > > > > > > > > > > > >
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Direct Socket

Direct Socket protocol is a generic protocol that allows low level access to socket functions.

Using this protocol the application itself can realize some IP based protocol without requirement for a development of a dedicated protocol.

Direct Socket protocol can be used as a standard (tag interface) protocol but also there is the appropriate implementation of DoCommand interface to enable using protocol from JavaScript.

The protocol can be used only with client socket type.

The protocol supports just one client socket. In case that application requires many sockets there could be many protocols installed, as the protocol supports multi-instance.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In Config node double-click Protocols.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the PLC list.

The protocol configuration dialog is displayed.

Direct Socket		x
		ОК
Socket type	UDP 👻	Cancel
Remote IP address	127.0.0.1	
Remote port	0	
Local IP address	0.0.0.0	
localPort	0	
Broadcast type	Global 👻	
Rx Token Prefix		
Rx Token Suffix		
Token Gap	0	
Tx Token Prefix		
Tx Token Suffix		
Hexadecimal Token	S	
Token Queue Size	100	
PLC Models		

Protocol parameters define a way how the connection is set and how the tokens are exchanged. The parameters are generally defined by the project. Many parameters can be accessed also as variables, allowing the runtime changes.

Element	Description			
Socket typeType of socket used for communication. Possible choices are UDP or TCP.				
Remote IP Address String. Indicates the IP address of remote device.				
Remote Port Integer. Indicates the port used by remote device.				
Local IP Address String. Indicates the IP address of local device. Mandatory for UDP usage.				

Element	Description			
Local Port Integer. Indicates the port used by local device. Mandatory for UDP usage.				
Broadcast Type of broadcast used. Possible choices are Global or Local.				

The following parameters are determining the behavior of the driver during RX and TX operations, as defined *Implementation Details* chapter.

Element	Description				
Rx Token Prefix	Indicates the prefix for read token, as string specified by hexadecimal characters.				
Rx Token Suffix	Suffix Indicates the suffix for read token, as string specified by hexadecimal characters.				
Token Gap	Indicates the period between tokens, in milliseconds.				
Tx Token Prefix	Indicates the prefix for sent token, as string specified by hexadecimal characters.				
Tx Token Suffix	Indicates the suffix for sent token, as string specified by hexadecimal characters.				
Hexadecimal	checked = tokens are in hexadecimal				
Tokens	not checked = tokens are not in hexadecimal				
Tokon Quouo Sizo	Indicates the number of tokens in the queue, as an integer value from 1 to 10000 (default: 100)				

Token Queue Size | Indicates the number of tokens in the queue, as an integer value from 1 to 10000 (default: 100)

All protocols parameters can be overwritten at runtime using the appropriate memory types, so the complete setup can be achieved during runtime using Tags. Settings using memory types are saved to permanent storage using standard procedures. The "Done" memory type is used in order that all set parameters are transferred to usage at once. If any parameter is changed the driver is re-programmed.

Tag Editor Settings

i

Path: ProjectView> Config > double-click Tags

- 1. To add a tag, click +: a new line is added.
- 2. Select **Direct Socket** from the protocol list: tag definition dialog is displayed.

Direct Socket		×
Direct Socket		
Memory Type	Data Type	Arraysize
Token To Send	✓ string	▼
Conversion	+/-	
	ОК	Cancel Apply Help

Element	Description						
Memory Type	Name	Datatype	Description				
.,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	Token To Send	string	Write only. Writing on this memory type sends the given string to communication.				
	Token Received	string	Read only. Reading from this memory type gets the front token from the receiving queue.				
	Length of Token Received	unsignedInt	Read only. Returns the length in bytes of the front token from the receiving queue.				
	Tokens Available	unsignedInt	Read only. Gives the number of tokens in the receiving queue.				
	Token Acknowledge	boolean	Write only. Writing to this memory type removes the front token from the receiving queue.				
	Connect	boolean	Write only. Writing 1 to this variable enables the connection.				
	Connection Status	boolean	Read only. Gives the status of the connection In TCP mode it reflects effective connection with the peer. In UDP mode it is TRUE as soon as Connect is TRUE				
	Socket type	string	Overrides protocol parameters. Check				
	Remote IP Address	string	"Protocol Editor Settings" from details.				
	Remote Port	unsignedShort					
	Local IP Address	strgin					
	Local Port	unsignedShort					
	Broadcast Type	string					
	Rx Token Prefix	string					
	Rx Token Suffix	string					
	Token Gap	unsignedInt					
	Tx Token Prefix	string					
	Tx Token Suffix	string					
	Hexadecimal Tokens	boolean					
	Token Queue Size	unsignedInt					
	Done	boolean	Writing to a tag of this memory type transfers all new values written in the other tags to protocol parameters, and to permanent storage.				

Element	Description					
Data Type	Data Type	Memory Space	Limits			
	boolean	1-bit data	01			
	unsignedByte	8-bit data	0 255			
	unsignedShort	16-bit data	0 65535			
	unsignedInt	32-bit data	0 4.2e9			
	string	Array of elements containing character code defined by selected encoding				
	Note: to define array []", "short[]"…	s. select one of Data Type format followed by square brackets like "byte				
Arraysize	 In case of string tag, this string tag. Note: number of bytes corresp or Latin1 in Tag Editor. 	g, this property represents the number of array elements. Ig, this property represents the maximum number of bytes available in the prresponds to number of string characters if Encoding property is set to UTF-8 et to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2				
Conversion		d Configured A ->CDAB EFGH->GHEFCDAB				
	Depending on data type select	ed, the list Allowed shows one or more o	conversion types.			

Element	Description	
	Value	Description
	Inv bits	inv : Invert all the bits of the tag.
		Example: $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)
	Negate	neg: Set the opposite of tag value.
		<i>Example:</i> 25.36 → -25.36
	AB -> BA	swapnibbles: Swap nibbles in a byte.
		<i>Example:</i> 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)
	ABCD -> CDAB	swap2: Swap bytes in a word.
		Example: 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)
	ABCDEFGH ->	swap4: Swap bytes in a double word.
	GHEFCDAB	<i>Example:</i> $32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format) $855441236 \rightarrow 1426062386$ (in decimal format)
	ABCNOP ->	swap8: Swap bytes in a long word.
	OPMDAB	Example: 142.366 → -893553517.588905 (in decimal format) 0 1000000110
		0001110010111011011001000101101000011100101
		→ 1 10000011100 101010000101000101101101100101101100001001111
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9)
		Example: $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)

Select conversion and click +. The selected item will be added to list **Configured**.

Element	Description
	If more conversions are configured, they will be applied in order (from top to bottom of list Configured).
	Use the arrow buttons to order the configured conversions.

Implementation Details

Principle of operation

Protocol is parameterized by number of protocols parameters. The parameters define which socket type is used and the host address.

The data access is based on 'tokens'. Token is data string that can be surrounded by prefix and suffix.

The protocol receiving process reads data from the specified IP/port and identifies tokens. Identified tokens are put to the queue from where they can be read by application. In the sending direction the application writes the token to protocol.

Protocol adds the defined tx_prefix/tx_suffix and sends data to the defined host.

Token extraction

The token extraction is slightly different for UDP and TCP sockets.

UDP protocols starts searching for tokens at the start of the received datagram. The search ends at the datagram end. If no rx_prefix is specified the token starts at datagram start. If no rx_suffix is specified the token ends on the datagram end. By specifying neither prefix nor suffix the whole datagram is delivered as a token. When both prefix and suffix are specified there can be many tokens extracted from a single datagram.

TCP protocol starts searching for tokens immediately after the previous rx_prefix. The search ends either when suffix is found or if the time gap without data is detected. If neither prefix nor suffix is specified the tokens will be all received data separated by time gaps.

The tokens can be plain ASCII strings, or hexadecimal strings. This is defined by the parameter 'hex_tokens'.

The prefix/suffix strings must always be in hexadecimal format.

Common behavior

Both ASCII and binary mode are available. When binary data can be present into receiving stream the **Hexadecimal Tokens** parameter can be set. In this case tokens are stored in queue using hex string coding (each byte is stored using two chars representing the hex value 0 to F). When defining the tags used to read tokens the appropriate string length should be computed considering the binary mode.

The **Token Queue Size** parameter specifies the maximum number of tokens saved into the queue. When the queue becomes full the oldest token is discarded.

The token identification is as follows:

- if the parameters specify a rx-prefix all characters before detecting the prefix are ignored
- if protocol specifies a rx-suffix it is used to detect the token end
- if rx-suffix is specified the parameter 'gap' specifies the timeout after which the token receiving is restarted

• if rx-suffix is not specified the parameter gap specifies the timeout that terminates the token (anything received up to this interval). If within this time the rx-prefix is detected again the token is ended and stored and reception of a new token is started

In summary we can have four combinations:

- a. No rx-prefix and rx-suffix: the incoming stream is divided in tokens according to gap detection
- b. Rx-prefix specified but no suffix: all the received chars before prefix are ignored. All the chars after prefix are stored in a token till the gap detection
- c. Rx-prefix and Rx-suffix specified: all the chars between prefix and suffix are stored in a token. All the chars received before prefix or after suffix till the gap detection or till a new prefix are ignored
- d. Rx-suffix specified but not RX-prefix: all the chars received till suffix are stored in a token. All the chars received after suffix till the gap detection are ignored

The rx-prefix and rx-suffix parameters are specified as hex strings, so any characters can be specified (like DLE STX CR LF etc...). i.e. to define the string "STR" as prefix the string "535452" must be used

Before putting string to the receiving queue the prefix and suffix are removed (only 'payload' saved).

Interface to user project

Reading a tag defined as **Token Received** gets the front string from the queue. If there are no new tokens an empty string is returned.

Reading a tag defined as Length of Token Received gets the length in bytes of the token.

Reading a tag defined as Tokens Available gets the number of tokens currently stored in the queue.

Writing to a tag defined as **Token Acknowledge** removes the token from queue and makes available the next token if present.

Writing to a tag defined as Token To Send means immediate sending, without any queue used.

Data traffic control

The TCP sockets can be controlled by variables "Connect" and "Connection Status". If the bool variable "Connect" is set the protocol will permanently try to make the connection to the specified host. If the TCP connection breaks it will be reestablished automatically. If the variable "Connect" is false the protocol will wait. The state of connection can be read by variable Connection Status".

For UDP there is no connection control. The socket is always connected and sends/receives data.

JavaScript Interface

Beside Tag interface the user can access the protocol via JavaScript.

Although defined Tags can be accesses by JavaScript too, JavaScript can access directly to a Command interface implemented in protocol. This interface does not require the definition of Tags and is direct to protocol resulting in more efficiency.

This interface provides the access to token queue and sending function. The following commands are supported:

Command	Description	
set_ip_port <ipaddress> <port></port></ipaddress>	Specify the remote IP/port couple to use for connection.	
	If protocol is already connected it is disconnected from current peer and re- connected to new one.	
	Example of usage in JavaScript:	
	<pre>var tagMgr = project.getWidget("_TagMgr"); var protID = "prot2"; // to be set according to protocol numbering tagMgr.invokeProtocolCommand(ProtID,"set_ip_ port","127.0.0.1 502");</pre>	
connect <on off></on off>	Enables/disables the connection.	
get_stat	Status of connection <connected disconnected>.</connected disconnected>	
put <string></string>	Put the token to send contained in string parameter.	
get	Get the received token.	
get_token_length	Get the length of received token.	
tokens_available	Get number of tokens received.	
token_ack	Acknowledge reading token.	

Using the command interface the following JS code should receive data:

```
var tagMgr = project.getWidget("_TagMgr");
var protID = "prot2"; // to be set according to protocol numbering
var avail = tagMgr.invokeProtocolCommand(protID, "tokens_available", "");
while (parseInt(avail) > 0)
{
    var str = tagMgr.invokeProtocolCommand(protID, "get", ""); // get the next
token
    var status = tagMgr.invokeProtocolCommand(protID, "token_ack",""); //
acknowledge current token
    avail = tagMgr.invokeProtocolCommand(protID, "tokens_available",""); // get
number of available tokens in queue
}
```

Tag Import

Select the driver in Tag Editor and click on the Import Tags button to start the importer.

	Tags	×								
+	-	X	đ	ß	>]	₽	A 9B	B>	ŧ3	1
Data			^		-	Та	g URI			_

Locate the Tag Editor Exported symbol file and click **Open**.

Tags included in the symbol file are listed in the tag dictionary. The tag dictionary is displayed at the bottom of the screen.

Tags × Protocols					•
+ - 🎽 🕲 🔊	() 🗞 🕅 🖏	R 🔎 - Search 🍸 Filter by: Data	▼ Ite	ems used:6/10000 Protocol	: Show all 🛛 😧 Show all tags
Data	Туре	Comment	^	Property	Value
Modbus TCP:prot1	Container			Y Driver	
Model: Modicon Modbus(1-based)				Model	Modicon Modbus(1-based)
	unsignedShort			Protocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			Dictionary	
- Holding Registers 3	unsignedShort			Array	false
- MRTU1	unsignedShort				
- MRTU2	unsignedShort			Array size	0
- MRTU3	unsignedShort			Arrayindex.Subing	dex 400003
- MRTU4	unsignedShort			Comment	
- MRTU5	unsignedShort			Data type	unsignedShort

Toolbar item	Description
	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
K 澍	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.
	Example of both checked and unchecked result:
	Tags* Tags* <th< th=""></th<>
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

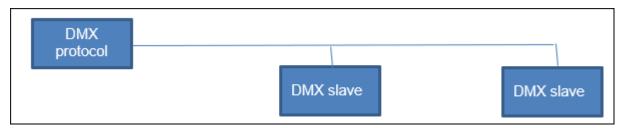
DMX512 Digital Multiplex

This document describes and specifies the implementation of DMX512 Digital Multiplex communication driver.

Purpose of implementation is to allow driving up to 512 channels connected to a RS485 serial line, or to merge additional channels, or to overwrite existing channels to an existing DMX controller.

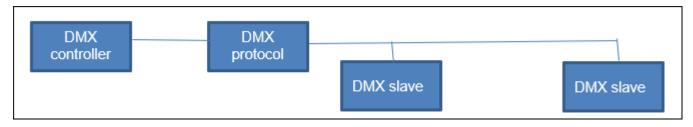
Possible topologies

Normal mode



In normal mode only Tx signal of the serial line is connected.

Merge mode



In merge mode the existing serial line must be opened and the origin line must be connected to Rx input.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In Config node double-click Protocols.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the PLC list.

The protocol configuration dialog is displayed.

DMX512 Digital Multiplex		×
Port		OK Cancel
	com1 👻	
Number of Channels	512	
Frame Delay (ms)	100	
Merge Mode		
PLC Models default		

Element	Description
Port	COM port to be used. Serial line parameters are fixed.
Number of Channels	1 - 512. Defines the number of channels transmitted in the multiplex flame.
Frame Delay (ms)	10 - 1000. Defines inter-frame delay to adapt to specifications of slaves. Delay is applied at the end of frame so the real frame rate is determined by formula: (approx)Time (microsec) = $120 + 20 + 40 \times (nr of channels) + Frame Delay * 1000$
Merge Mode	Selects the Merge Mode in which the unit receives a frame from an external controller and substitutes the values of some of the channels or add other channels in the end of the frame
PLC Models	Only "default" is available.

Tag Editor Settings

In Tag Editor select DMX512 Digital Multiplex protocol.

Add a tag using [+] button. Tag setting can be defined using the following dialog:

DMX512 Digital Multiplex	×
DMX512 Digital Multiplex	
memtype	index
channel -	
datatype	
short 👻	
	OK Cancel Apply Help

Each channel can be assigned to a Tag.

Element	Description		
memtype	Memory Type Description		
	channel	Only available memory type.	
index	Refer to channel number to point to.		
datatype	Data Type	Memory Space	Limits
	short	16-bit data	-32768 32767

Channel behavior

Only available DataType is short (signed 16-bit data) so a Tag can assume values from -32768 to 32767. Anyway the protocol uses only values from 0 to 255.

Other values are used in Merge Mode: when the channel overwrites an existing channel the negative values are used to disable overwriting.

Value	Normal Mode	Merge Mode
0 to 255	0 to 255	0 to 255
> 255	255	255
< 0	0	original value of channel in the incoming frame

Tag Import

Select the driver in Tag Editor and click on the Import Tags button to start the importer.



It is possible to import a Tag Editor exported xml

Туре	Description		
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.		
	Tags ×		
	+ - 🕺 🕲 🔎 >] 🚺 🕼 🖬 🕅 🗉		

Once the importer has been selected, locate the symbol file and click **Open**.

Data

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tag URI

Tags × Protocols						
+ - 🎽 🕲 🔊	D & D D	R 🔎 - Search Tilter by: Data	▼ Ite	ems used:6/10000	Protocol: Show all	🗸 🗹 Show all tags 🖉 🗍
Data	Туре	Comment	^	Property		Value
Modbus TCP:prot1	Container			✓ Driver		
Model: Modicon Modbus(1-base	a)			Model		Modicon Modbus(1-based)
- Holding Registers 1	unsignedShort			Protocol		Modbus TCP:prot1
- Holding Registers 2	unsignedShort			✓ Dictionary		
 Holding Registers 3 	unsignedShort					
- MRTU1	unsignedShort			Array		false
- MRTU2	unsignedShort			Array siz	e	0
MRTU3	unsignedShort			Arrayind	ex.Subindex	400003
- MRTU4	unsignedShort			Commen	t	
- MRTU5	unsignedShort			Data typ	e	unsignedShort

Toolbar item	Description
	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.
	Example of both checked and unchecked result:

Toolbar item	Description
	Tage: N Image: N
P → Search Tag name →	Searches tags in the dictionary basing on filter box item selected.

Eaton Suconet-K

The Eaton Suconet-K communication driver has been designed to connect HMI devices to a Suconet-K network with a Möeller PLC.

Protocol Editor Settings

Eaton Suconet-K		X
		ОК
Port	com1 👻	Cancel
slave ID	2	
Input Buffer Size	6	
Output Buffer Size	7	
timeout	3000	
PLC Models		
PS4 PS341 PS306-316 PS416		

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the **PLC** list.

The protocol configuration dialog is displayed.

Element	Description
Port	Serial port selection.
	COM1: device PLC port.
	 COM2: computer/printer port on panels with 2 serial ports or optional Plug-In module plugged on Slot 1/2 for panels with 1 serial port on-board.
	 COM3: optional Plug-In module plugged on Slot 3/4 for panels with 1 serial port on- board.
slave ID	node of the slave device.
Input Buffer Size	Size of Input Buffer. Input data length must be exactly the same as in PLC configuration.
Output Buffer Size	Size of Output Buffer. Output data length must be exactly the same as in PLC configuration.
timeout	Time delay in milliseconds between two retries in case of missing response from the server device.
PLC	Two PLC models are available:
Models	• PS4
	• PS341
	• PS306-316
	• PS416

Tag Editor Settings

In Tag Editor select the protocol Eaton Suconet-K.

Add a tag using [+] button. Tag setting can be defined using the following dialog:

Eaton Suconet-K			×
Eaton Suconet_K			
Memory Type	Offset	SubIndex	
Internal Relay	▼ 0 2	0 🔻	
Data Type	Arraysize	Conversion	
boolean 👻	0		+/-
	OK	Cancel	Apply Help

Element	Description		
Memory Type	Memory Type	Description	
.,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	Internal relay	Internal memory of PLC. It can be addressed using Offset and Data Type.	
Offset	Starting address for the Tag. The possible range depend on PLC model selected.		
SubIndex	This allows resource offset selection depending on the selected data type.		
Data Type	Available data types: • boolean • byte • short • int • unsignedByte • unsignedShort • unsignedInt • float • double • string • binary See "Programming concepts" section in the main manual. () Note: To define arrays, select one of Data Type format followed by square brackets (byte[], short[]).		

Element	Description							
Arraysize	In case of array tag, this property represents the number of array elements.							
	 In case of string tag, this property represents the maximum number of bytes available in the string tag. 							
	Note: number of bytes corresponds to number of string characters if Encoding property is set to UTF-8 or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2 bytes.							
Conversion	Conversion to	be applied to the tag.						
	inv,swap2	Allowed Configured						
	BCD AB->BA ABCD->CDAB ABCD=>CDAB ABCD=>CDAB ABCD=>CDAB Cancel OK Cancel OK							
	Value Description							
	Inv bits	inv: Invert all the bits of the tag.						
		Example: 1001 \rightarrow 0110 (in binary format)						
		$9 \rightarrow 6$ (in decimal format)						
	Negate							
	Negate	$9 \rightarrow 6$ (in decimal format)						
	Negate AB -> BA	$9 \rightarrow 6$ (in decimal format) neg : Set the opposite of tag value. <i>Example:</i>						
		$9 \rightarrow 6$ (in decimal format) neg : Set the opposite of tag value. <i>Example:</i> $25.36 \rightarrow -25.36$						
	AB -> BA ABCD ->	$9 \rightarrow 6$ (in decimal format)neg: Set the opposite of tag value.Example: $25.36 \rightarrow -25.36$ swapnibbles: Swap nibbles in a byte.Example: $15D4 \rightarrow 514D$ (in hexadecimal format) $5588 \rightarrow 20813$ (in decimal format)swap2: Swap bytes in a word.						
	AB -> BA	$9 \rightarrow 6$ (in decimal format)neg: Set the opposite of tag value. <i>Example:</i> $25.36 \rightarrow -25.36$ swapnibbles: Swap nibbles in a byte. <i>Example:</i> $15D4 \rightarrow 514D$ (in hexadecimal format) $5588 \rightarrow 20813$ (in decimal format)						

Element	Description					
	Value	Description				
	-> GHEFCDAB	<i>Example:</i> 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)				
	ABCNOP - > OPMDAB	swap8 : Swap bytes in a long word. Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) $0\ 10000000110$ 0001110010111011001000101101000011100101				
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)				
	Select conversion and click +. The selected item will be added to list Configure If more conversions are configured, they will be applied in order (from top to bot Configured).					

Use the arrow buttons to order the configured conversions.

Environment Variables

This protocol gives the possibility to copy the environment variables of the hosting Operative System inside tags. All variables will be read only, namely, is not possible to modify them.



Environment Variables communication driver is not counted as physical protocol. Refer to **Table of functions and limits** from main manual in "Number of physical protocols" line.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In the **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the Environment Variables protocol from the PLC list.

Tag Editor Settings

Path: ProjectView> Config > double-click Tags

- 1. To add a tag, click +: a new line is added.
- 2. Select Environment Variables from the protocol list: tag definition dialog is displayed.

Environment Variables				×
Environment Variables				
Data Type string ~	Arraysize	Conversion	+/-	
Name PATH				
	OK	Cancel	Apply	Help

Element	Description
Name	Name of the environment variable that you want to read.
Data Type	System variables are of type string, but if a different type is chosen, e.g. int, casting to the chosen type will be made.
Arraysize	This property represents the maximum number of bytes available in the string or in the array Tag.

Ethernet/IP CIP

The protocol has been implemented according to the published Ethernet/IP specifications (available from www.odva.org).

The Ethernet/IP CIP driver has been designed to provide the best performance with the least amount of impact on the system's overall performance. Although the Ethernet/IP CIP driver is fast, we suggest to use short Tag names. Tags are read from and written to the device by specifying their symbolic name in the communications request, therefore the longer the tag name is, the larger the request will be.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the **PLC** list.

The protocol configuration dialog is displayed.

Ethernet/IP CIP		×
PLC Network		ОК
Alias		Cancel
IP Address	0.0.0.0	
Slot	0	
PLC Models		
Logix 5000 Micro800 Omron NJ Serie Omron CJ Serie		

Field	Description
Alias	Name identifying nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node.
IP Address	Ethernet IP address of the controller.
Slot	CPU slot number for Logix 5000 models (typically 0). Refer to the controller documentation for further details.

Field	Description
PLC Models	PLC model used to import tags file.
PLC Network	Enable access to multiple networked controllers. For every controller (slave) set the proper option.

Controller Model Logix 5000

The Ethernet/IP CIP driver allows to connect Allen-Bradley ControlLogix and CompactLogix Ethernet controllers.

Communication with ControlLogix® 5500 controllers can be accomplished through an Ethernet/IP communication module for Ethernet such as the 1756-EN2T or 1756-ENET.

Ethernet communication with CompactLogix[™] 5300 controllers requires a processor with a built-in Ethernet/IP port such as the 1769-L32E.

All trademarks are the property of their respective owners.

The internal memory organization of the Logix CPUs is not fixed but configured by the user at development time. Each data item can be identified by a string called "Tag". The RSLogix 5000 software can then export to the application the list of Tags created for each controller.

The project loaded on the HMI device must refer to Tag names assigned in RSLogix 5000 software at development time. The Tag Editor supports direct import of the Tag file generated by RSLogix 5000 software in .CSV format.

The implementation of the Ethernet/IP driver also supports access to structured data types which can be imported from .L5X files.

The driver supports access to both Controller and Program Tags.

Export CSV and L5X files using RSLogix5000

To export the .CSV Tag file:

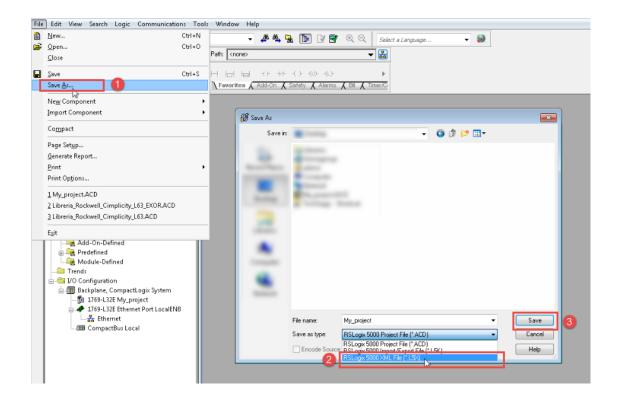
- 1. From the Controller Organizer pane, right-click on Controller Tags.
- 2. Select Export Tags: the Export dialog is displayed.

File Edit View Search Logic Communic	ations Tools Window Help		
🗎 🖆 🖬 🎒 X 🖻 💼 🗠 억	- <u>A</u> ≊ A <u>6</u> , G	🛦 📭 📝 🗣 🍳 🔍 🛛 Select a Language	
Offline RUN No Forces C BAT	B ← \ Favorites Add-On .	✓ ↓ ()> -(0)> -(0.)> ↓ Safety & Alarms & Bt & Timer/Q	
Controller Organizer Controller My project Controller My project Controller Fau Power-Up Har Monitor Tags MinProgr Unscheduled Export Tags Motion Group: Unscheduled Export Tags Motion Group: Print Add-On Instructions Data Types Module-Defined Module-Defined Module-Defined Module-Defined Module-Defined Module-Defined Trends CompactBus Local	ENB	File name: My_project-Controller-Tags Save as tyr A RSLogix 5000 Import/Export File (*.C Controller Programs Equipment Phases Add On Instructions	Export CSV)

- 3. Choose **All** from the **Tags** list to export all Tags.
- 4. Select the Save as type option to .CSV.
- 5. Click **Export**: all the Tags are exported to an **.CSV** file.

To export the .L5X data type file:

- 1. Choose File > Save As.
- 2. Select the Save as type option to .L5X.
- 3. Click **Save**: all the Tags are exported to an **.L5X** file.



Import Files in Tag Editor

Select the driver in Tag Editor and click on the Import Tags button to start the importer.



The following dialog shows which importer type can be selected.

MIStudio		×
Multiple tag importer	are available for this protocol. Please select the importer type and c	ontinue.
Version	Туре	^
Allen-Bradley L5X v1.1	Hierarchical	
Allen-Bradley RSLogix5000 v	15 Linear	E
Allen-Bradley CCW v15	Linear	-
Omron Sysmac v15	Linear	
Omron CX-One v15	Linear	
Tag Editor exported xml	General	-
	ОК Са	ncel

Select Allen-Bradley RSLogix5000 v15 option.

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

	Tags	• × [Protocols										
+	-	×	60	>]	[> & _B	6 8	3	R 🔎 Search	Filter by: Data	•	Ite	ms used:6/10000 Protocol: Show all	Show all tags 🔅 🗖
Data			^		Туре			Comment			^	Property	Value
		us TCP:			Containe							✓ Driver	
- I			on Modbus(1-based)								Model	Modicon Modbus(1-based)
			egisters 1		unsigned							Protocol	Modbus TCP:prot1
			egisters 2		unsigned							✓ Dictionary	
		olaing к IRTU1	egisters 3		unsigned unsigned							Array	false
		RTU2			unsigned							Array size	0
		RTU3			unsigned							Arravindex.Subindex	400003
		RTU4			unsigned							Comment	100000
	-M	RTU5			unsigned	Short						Data type	unsignedShort
						-				_		Data type	unsignedation

Toolbar item	Description
	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
K ä	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.
	Example of both checked and unchecked result:

Toolbar item	Description
	Togs* x Togs* x • • • • • • • • • • • • •
P- Search	Searches tags in the dictionary basing on filter of box item selected.

Note: When importing the array data types, the importer is expanding them creating individual Tags per each array element; this is valid for all the data types, except for arrays of boolean. In this case they are imported as "boolean-32" and the single array element can be addressed using "Tag Index" parameter from "Attach to…" dialog.

Module-Defined and User-Defined data types

RSLogix 5000 allows you to define Tags with several data types.

Data type group	Description
Predefined	Standard data types such as BOOL, DINT, SINT, INT and other less common data types such as PID, COUNTER, TIMER.
Module-Defined	Data type associated with I/O optional modules usually referenced by aliases.
User-Defined	Custom data type defined by user

In order to import Predefined (with the exception of standard data types which are always imported) and Module-Defined data type you need to edit the ETIPSpecialDataTypes.xml file located under *languages\shared\studio\tagimport* or *studio\tagimport* depending on installed version.

In RSLogix5000 software:

1. From the Controller Organizer pane, select Controller Tags.

Logic 5000 File Edit View Search Logic Com	munications Tools Window Help		_ 7
ine can new centri coge can	- 🐣 🗣 🙀 🚺 📝 💇 🔍		- 9
			• 🕺
fline 🛛 🗸 🔲 RUN	Path: <none></none>	▼ 🖁	
Forces	4		
Edits 🔒 🖬 🖓		•	
dundancy 0.0	Favorites & Add-On & Safety & Alarms & E	it 🖌 Timer/Ci	
ontroller Organizer 🛛 👻 🔻 🗙	Scope: 1 G1_PCS - Show: All Tags	🗸 🏹 Enter Deven	ption Filter
Controller G1 PCS	Name III	△ Data Type	Style ^
- Controller Tags	T-HYS_1_SEC_PLS	TIMER	
Controller Fault Andler Power-Up Handler	T-HYS_ALM_TON	TIMER	Decimal
Tasks	HYS_ALM_TWR_DE	BOOL	Decimal
- Gana Tasks	HYS_BKT_ALL_STP	BOOL	Decimal
Recipe_Task	HYS_BKT_CONF_CYL_OE	BOOL	Decimal
B-G Recipe_Handling	HYS_BKT_CONF_DWN_DI	BOOL	Decimal
- Servo_Program	HYS_BKT_CONF_UP_DI	BOOL	Decimal
PCS_Cassette_Servo_Control	HYS_BKT_CONV_HME_T0_SRV_OE	BOOL	Decimal
- Unscheduled Programs / Phases	⊞ HYS_BKT_DIST_1_2_CALC_SP	DINT	Decimal
- Canada Motion Groups	HYS_BKT_DIST_1_3 CALC_SP	DINT	Decimal
- 🔁 Add-On Instructions	HYS_BKT_HME_DI	BOOL	Decimal
🛓 📲 Jog_Axis	THYS_BKT_HME_DIST_1_2_OFST_SP	DINT	Decimal
🖂 Data Types	⊞ HYS_BKT_HME_DIST_1_3_OFST_SP	DINT	Decimal
🕀 🖳 User-Defined	± HYS_BKT_LNG_SP	DINT	Decimal
🕫 🚰 Strings	E HYS_BKT_MAM_ACC_SP	DINT	Decimal
🗄 🚂 Add-On-Defined	⊞ HYS_BKT_MAM_HGH_ACC_SP	DINT	Decimal
🕀 🙀 Predefined	⊞ HYS_BKT_MAM_HGH_JRK_SP	DINT	Decimal
💼 🕞 Module-Defined	HYS_BKT_MAM_HGH_SPD_SP	DINT	Decimal
Trends	THYS_BKT_MAM_JRK_SP	DINT	Decimal
- 🗀 I/O Configuration	Monitor Tags) Edit Tags /		* *

2. Filter tags to display only **Module-Defined** Tags.

Define Tag Filter	-
Filter On:	ОК
<al> ▼</al>	Cancel
Include Data Types:	
User-Defined	Clear Filter
⊕-Strings ⊕-Add-On-Defined	
F-Predefined	1
Module-Defined	J
	Help

Only tags (alias) with data type belonging to optional I/O Modules will be displayed.

Name	Data Type	그림 스	Style	^
. HYS_Point_I0_Rack_20:1	AB:1734_3SLOT:1:0			
I HYS_Point_I0_Rack_20:0	AB:1734_3SL0T:0:0			
. HYS_Point_IO_Rack_1:I	AB:1734_13SLOT:1:0			
HYS_Point_I0_Rack_1:0	AB:1734_13SLOT:0:0			
HYS_Point_I0_Rack_1:2:C	AB:1734_DI8:C:0			=
HYS_Point_I0_Rack_1:3:C	AB:1734_DI8:C:0			
───HYS_Point_IO_Rack_1:4:C	AB:1734_DI8:C:0			
HYS_Point_IO_Rack_1:5:C	AB:1734_DI8:C:0			
HYS_Point_IO_Rack_1:6:C	AB:1734_DI8:C:0			
HYS_Point_I0_Rack_1:7:C	AB:1734_DI8:C:0			
HYS_Point_IO_Rack_1:8:C	AB:1734_DI8:C:0			
HYS_Point_I0_Rack_20:1:C	AB:1734_DI8:C:0			
HYS_Point_IO_Rack_1:9:C	AB:1734_D08_NoDiag:C:0)		
HYS_Point_I0_Rack_1:10:C	AB:1734_D08_NoDiag:C:0)		
HYS_Point_I0_Rack_1:11:C	AB:1734_D08_NoDiag:C:0)		
HYS_Point_I0_Rack_1:12:C	AB:1734_D08_NoDiag:C:0)		
HYS_Point_I0_Rack_20:2:C	AB:1734_D08_NoDiag:C:0)		
HYS_Point_I0_Rack_1:1:C	AB:1734_VHSC:C:0			
────HYS_Point_IO_Rack_1:1:I	AB:1734_VHSC:1:0			

In this example alias HYS_Point_IO_Rack_20:I refers to data type AB:1734_3SLOT:I:0. Expand this tag to see how this data type is structured:

S	cope: Show:	▼ ▼. Enter Description F	ilter
	Name	Data Type 🔤 🛆	Style
	- HYS_Point_IO_Rack_20:1	AB:1734_3SLOT:1:0	
	HYS_Point_I0_Rack_20:I.SlotStatusBits0_31	DINT	Binary
	HYS_Point_I0_Rack_20:I.SlotStatusBits32_63	DINT	Binary
	HYS_Point_I0_Rack_20:I.Data	SINT[3]	Binary
	5 811, Page 3, Page 311	and the second second	

To make sure that HYS_Point_IO_Rack_20:I, and all his sub-tags, will be imported into the project, open the ETIPSpecialDataTypes.xml file in any text editor and check if the AB:1734_3SLOT:I:0 data type is included. If so you can proceed with the following data type. If not, you need to add it manually.

The structure is as in this example:

```
<DataType Name="aaa">
</Members>
</Member Name="bbb" DataType="ccc" Dimension="ddd" Radix="eee"/>
</Members>
</DataType>
```

where:

- aaa = Alias/Tag data type
- bbb = Sub-tag Name (it's sub-tag name part after dot)
- ccc = Sub-tag data type
- ddd = Array dimension (0 if it is not an array)
- eee = Style

In the example above:

🔚 ETIPSpecialDataTypes.xml 🔀

```
238
239 CataType Name="AB:1734_3SLOT:I:0">
240 Ammbers>
241 Ammber Name="SlotStatusBit0_31" DataType="DINT" Dimension="0" Radix="Binary"/>
242 Ammber Name="SlotStatusBit32_63" DataType="DINT" Dimension="0" Radix="Binary"/>
243 Ammber Name="Data" DataType="SINT" Dimension="3" Radix="Binary"/>
244 Ammbers>
245
```

- 3. Repeat step 2 for all Module-Defined data types.
- 4. Repeat the procedure from step 2, filtering Tags to display only **Predefined** Tags.

Controller Model Omron Sysmac

Data in NJ and CJ controllers can be accessed via CIP protocol.

Each data item can be identified by a string called "Tag". Use appropriate programming tools for controller to export the list of Tags.

NJ series controller are programmed using Sysmac Studio:

- NJ301-xxxx
- NJ501-xxxx

CJ series controller are programmed using CX-One:

- CJ2M CPU-3x
- CJ2H CPU 6x-EIP
- Any CPU with a CJ1W-EIP21 attached.

The project loaded on the HMI device must refer to the Tag names assigned in the programming software at development time. The Tag Editor supports direct import of the Tag file generated by Sysmac Studio software in .NJF format or generated by CX-One in the .CJF format.

All Tags to be accessed by the HMI device must be declared as Global Variables.

Export NJF files using Sysmac Studio

To export the **.NJF** Tag file:

- 1. In Sysmac Studio declare Tags as Global Variables.
- 2. Set the Network Publish attribute to Publish Only.

Edit View Insert Project	Controller Simulation To	ools Help							
(🖷 🗎 🖄 🍮 🖒 🖬	81	🖶 A 🖲 🤻 🔺 🔌	6 8 5 9	0 %	2				
o Case 🚽 🖣					_	_		/	
sster_PLC 🔻	Programming			_	_	_		_	_
	Global Variables X								
Configurations and Setup	Name	I Data Type	I Initial Value	I AT	I Retain	I Constan		I Comment	1
EtherCAT	N1_Input_Bit_00	BOOL		ECAT://no			Publish Only		
CPU/Expansion Racks E == CPU Rack	N3_REM_SF_RST	BOOL		ECAT://no			Publish Only		
I/O Map	HML_SAFETY_RST	BOOL					Publish Only	SAFETY RESET FROM HMI	
▼ R Controller Setup	PLC_BYTE	BYTE			8		Do not publish		
□ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □	HMLBYTE	BYTE					Publish Only		
⊢ d [#] Built-in EtherNet/IP	PLC_DINT	DINT					Do not publish		
View State And Setup	HMI_DINT	DINT					Publish Only		
L 🔅 Axis Settings	PLC_DWORD	DWORD					Do not publish		
L 🕼 Axes Group Setting	HMI_DWORD	DWORD					Publish Only		
⊥ & Cam Data Settings ↓ ► Event Settings	PLC_INT	INT			E.	E	Do not publish		
 In Task Settings 	HMLINT	INT					Publish Only		
L E Data Trace Settings	PLC_UNT	LINT					Do not publish		
Programming	HMILLINT	LINT					Publish Only	•	
V II POUs	PLC LREAL	LREAL					Do not publish Publish Only		
▼ 🕄 Programs	HMI LREAL	LREAL					Publish Only Input	2	
🔻 🖂 Main	PLC LWORD	LWORD			-		Output	-	
L 🖑 Safety	HMI LWORD	LWORD				Ē	Publish Only		
CommsTest	PLC REAL	REAL				Ē	Publish Only		
Im Functions Im W Function Blocks	HMLREAL	REAL					Publish Only		
I Data	PLC_SINT	SINT			-		Publish Only		
⊥ 5t Data Types	HML SINT	SINT			-	- E	Publish Only		
 L Im Global Variables 	PLC_UDINT	UDINT			-		Publish Only		
▶ 🗈 Tasks	HMI UDINT	UDINT			8		Publish Only		
	PLC UINT	UINT			- E	- H	Publish Only		
	HMLUINT	UINT					Publish Only		
	PLC_ULINT	UUNT			8		Publish Only		
	HMLUUNT	UUNT				- 🗄	Publish Only Publish Only		
	PLC USINT	USINT					Publish Only Publish Only		
	PLC_USINI LILAR UPINIT	USINI					Publish Only Bublish Only		
							and the second second		

2. From the **Tools** menu, choose **Export Global Variables > CX-Designer**.

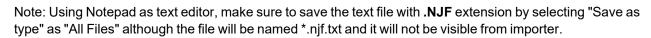
File Edit View Insert Project	Controller Simula	Tools Help	
V AL B m to at A	ា ៤ < % ធ	Troubleshooting	A ≤ 0 0 0 0
		Backup •	
Demo Case 🔹 👎	2	Export Global Variables	Network Configurator
	Programming	Comments for Variables and Data Types 🕨	CX-Designer 3
Master_PLC 🔻	Global Variables Safe	Import ST Program	
▼ Configurations and Setup	Variables	Option	
► 跚 EtherCAT	0 Reset		X N3 REM SE
▼ ISt CPU/Expansion Racks	Sysmac	Studio	N3_REM_SF
► == CPU Rack			
L 4* I/O Map ▼ ℝ Controller Setup			
↓ ☐ Operation Settings		Global variables are copied to t	the clipboard.
⊢ d [#] Built-in EtherNet/IF		Paste them to the symbol table	e of the CX-Designer.
🔻 🕸 Motion Control Setup			
L - 他 Axis Settings			
L 🇠 Axes Group Setting			
⊢ & Cam Data Settings			(4) ОК
Event Settings			
L M Data Trace Settings			

- 3. Click **OK** to confirm.
- 4. Cut and paste the content of the clipboard in any text editor.

File Edit View Insert Project	Controller Simulation Tools	Help			_	_			
X A A A S C A		从◎ 戌 <u>∧</u> ≫ &	a 68 % %	0 2	9				
Demo Case 🚽 🦞	Programming								
Master_PLC 🔻			_	_	_	_	_		
	Global Variables X								
 Configurations and Setup 	Name	I Data Type	Initial Value	I AT		Constan		I Comment	1
► 肼 EtherCAT	N1_Input_Bit_00	BOOL		ECAT://no			Publish Only		
ISt CPU/Expansion Racks ISt CPU Rack	N3_REM_SF_RST	BOOL		ECAT://no			Publish Only		
► == CPU Rack	HMI_SAFETY_RST	BOOL					Publish Only	SAFETY RESET FROM HMI	
▼ R Controller Setup	PLC_BYTE	BYTE					Do not publish		
Operation Settings	HMI_BYTE	Export.NJF - Blocco not							
🗉 🎁 Built-in EtherNet/IF	PLC_DINT		e						
V 🕸 Motion Control Setup	HMI_DINT	File Modifica Formato	Visualizza	?					
⊢ t‡ Axis Settings	PLC_DWORD	HOST NAME DA	TATYPE	Δ	DRESS	COMM	ENT TAGLINK	RW POU	
L the Axes Group Setting L the Cam Data Settings	HMI_DWORD	N1_Input_E						Bit 8 bits/Inpu	t Bit 00
□ B Cam Data Settings	PLC_INT	N3_REM_SF_		DL EC	AT://	node#	[1,3]/REM_S	F_RST	TRUE RW
L Task Settings	HMI_INT	HMI_SAFETY				SAFE	TY RESET FR		RW
Data Trace Settings	PLC_LINT	HMI_BYTE	BY				TRUE	RW	
▼ Programming	HMI_LINT	HMI_DINT	DIN				TRUE	RW RW	
V 🔮 POUs	PLC_LREAL	HMI_DWORD HMI_INT IN		JKD		TRUE	RW	ĸw	
▼ 3€ Programs	HMI_LREAL	HMI_LINT		т		TROL	TRUE	RW	
V 🖂 Main	PLC_LWORD	PLC_LREAL	LRE				TRUE	RW	
∟ 🔄 Safety ∟ 🗟 CommsTest	HMI_LWORD	HMI_LREAL	LRE				TRUE	RW	
L 💓 Functions	PLC_REAL	PLC_LWORD	LWC				TRUE	RW	
L I Function Blocks	HMI_REAL	HMI_LWORD PLC_REAL	LWO				TRUE	RW RW	
🔻 🎞 Data	PLC_SINT	HMI_REAL	RE/				TRUE	RW	
Data Types	HMI_SINT	PLC_SINT	SI				TRUE	RW	
Clobal Variables	PLC_UDINT	HMI_SINT	SI				TRUE	RW	
Tasks	HMI_UDINT	PLC_UDINT	UD				TRUE	RW	
	PLC_UINT	HMI_UDINT	UD				TRUE	RW	
	LINE LENT	PLC_UINT	UIU	T			TRUE	RW	

4. Save the file as .NJF.

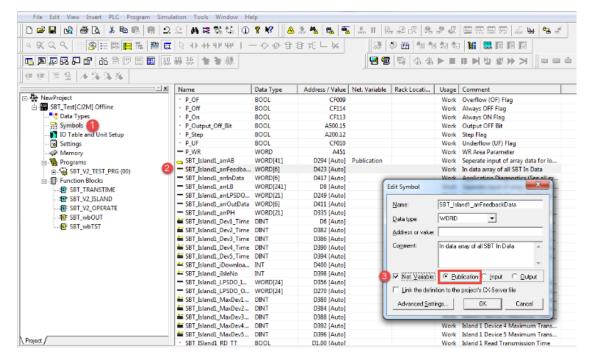
Ť



Export CJF file using CX-One

To export the **.CJF** Tag file:

- 1. In CX-One open the Symbols file in the project.
- 2. In the Edit Symbol dialog set the Net. Variables attribute to Publication.



3. Copy and paste all the Tags in any text editor.

Untitled - Notepad			
File Edit Format View Help			
SBT_Is]and1_LPSD0_OUT_Buff	WORD[24]		• 0
SBT_Island1_MaxDev1_Time			0
SBT_Island1_MaxDev2_Time			0
SBT_Island1_MaxDev3_Time		and 1 Device 3 Maximum Transmission Time	0
SBT_Island1_MaxDev4_Time	DINT IS	Save As	X
SBT_Island1_MaxDev5_Time	DINT IS	Jave As	
SBT_ISland1_RD_TT BOOL	Island 1 Re		
SBT_ISLAND1_TT FUNCTION BLOCK SBT_ISland1_wAddDiagCode		(-) < + CJ export	earch CJ export 🔎
	WORD FB		
SBT_ISland1_wAppAck WORD SBT_ISland1_wAppDiag WORD	Application Application	Organize 🔻 New folder	= - @
SBT_ISTand1_WAPpDTag WORD	FB Diag Cod		J U
SBT_ISTand1_wDTagCode word SBT_ISTand1_wDTagCodeDev1	WORD Dia	^ Name	 Date modified Type
SBT_ISTand1_wDTagCodeDev1	WORD Dia	Parantes Twome	Duce mounded Type
SBT_ISTand1_wDTagCodeDev2	WORD Dia	Contract of the second	
SBT_ISTand1_wDTagCodeDev4	WORD Dia	No items match you	ir search.
SBT_ISland1_wDiagCodeDev5	WORD Dia	Computer	
SBT_ISland1_wDiagCodeLPSD0	WORD Dia		
SBT Island1 xAcceptDiffLogic	BOOL ACC	 Devricadi 	
SBT_Island1_xAckDev1 BOOL	Acknowledge	C Drophes	
SBT_Island1_xAckDev2 BOOL	Acknowledge		
SBT_ISland1_xAckDev3 BOOL	Acknowledge	30. Recent Places	
SBT Island1 xAckDev4 BOOL	Acknowledge		
SBT_Island1_xAckDev5 BOOL	Acknowledge		
5BT_Island1_xAckLPSD0 BOOL	Acknowledge	Libraries	
5BT_Island1_xAckOp BOOL	Operator Ão	C Incoments	
5BT_Island1_xActivate BOOL	Activate FE	C. Large and C. La	
5BT_Island1_xActive BOOL	FB Active	Munic .	
SBT_ISland1_XCOK BOOL	Communicati		
5BT_Island1_xDevError BOOL	Operator Ac	💼 Pictures 🔻 K 👘 🕅	>
5BT_Island1_xDiffLogicDetected	BOOL Dif		
SBT_Is]and1_xError BOOL	Error	File name: *.CJF	-
SBT_Island1_xOpAckReq BOOL	Power Up Re		
SBT_ISland1_XPUR BOOL	Device Erro	Save as type: All Files (*.*)	•
5BT_Island1_xRunLogic BOOL	SBT LOGIC R		
SBT_ISLAND_1 FUNCTION BLOCK			
SBT_V2_SFPRG FUNCTION BLOCK	SBT V2 Safe	ANSI -	Save Cancel
strFilename STRING[40]			
TT_TMR_PV UINT	Island 1 Transmissi		
udtSBT_Island1 STRUCT D5000	Island 1 Safety Dat	a uuusbi_vz	
			-

4. Save the file as .CJF.

ž

Note: Using Notepad as text editor, make sure to save the text file with **.CJF** extension by selecting "Save as type" as "All Files" although the file will be named *.cjf.txt and it will not be visible from importer.

Export User Defined structures

To export the .CJS Tag file:

1. In CX-One open the Data Types file in the project.

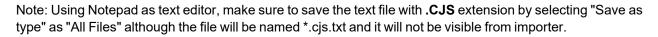
Tile Edit View Insert PLC Program Simul	ation Tools Window H	elp		
0 🛎 🖬 🖓 🖨 🖪 🕷 📾 📾 🕮 2	오 🛛 🗛 🕱 📽 🖉 🛈	? ₩ 2 _6) 🎄 🍓 🙇 🖣	5 . L B C L P C
<u>■</u> # # 8	l≳ -1114-412-442-1	-001	母毛上昇] 📰 🕸 🛗 🖿 🐄 🖬
B B C C C C C C C C C C	おお 書書 録			🛛 🔁 🖉 🖓 🖓 🕨 🔳
律律 国皇 本%%%				
<u></u>	Name	Data Type	Array Size	Comment
B MewProject . (2	🛃 udtSBT_V2			
E - SBT_Test[CJ2M] Offline	DEV_1_IN_Buff	WORD	[4]	
T Data Types	 DEV_2_IN_Buff 	WORD	[4]	
🔗 Symbols	DEV_3_IN_Buff	WORD	[4]	
10 Table and Unit Setup	DEV_4_IN_Buff	WORD	[4]	
🐻 Settings	DEV_5_IN_Buff	WORD	[4]	
	DEV_1_OUT_Buff	WORD	[4]	
🗄 🙀 Programs	DEV_2_OUT_Buff	WORD	[4]	
E- SBT_V2_TEST_PRG (00)	DEV_3_OUT_Buff	WORD	[4]	
E Function Blocks	DEV_4_OUT_Buff	WORD	[4]	
	DEV_5_OUT_Buff	WORD	[4]	
	iProjRead	INT		
	iBlockNum	INT		
41 SBT_wbOUT	* xBlockRead	BOOL		
	 xDiffLogicDetected xLPSDO_RUN 	BOOL BOOL		
	* xActivate	BOOL		
	* xActivate	BOOL		
	* xTransmitSnd	BOOL	[6]	
	xtransmicand	BOOL	[0]	
Project /				

2. Copy and paste all the Tags in any text editor.

File Edit Format View Help							
BT_Island1_LPSD0_OUT_Buff	WORD[24	4]	Seperate in	out of array da	ta for loop	0	
SBT_Island1_MaxDev1_Time	DINT		and 1 Device 1 Ma	aximum Transmis	sion Time	0	
5BT_Island1_MaxDev2_Time	DINT		and 1 Device 2 Ma			0	
5BT_Island1_MaxDev3_Time	DINT		and 1 Device 3 Ma	aximum Transmis	sion Time	0	
5BT_Is]and1_MaxDev4_Time	DINT	IS	Save As	and the second sec	a contra de contra de la contra d		×
SBT_Island1_MaxDev5_Time	DINT	IS	Jave As	COMPANY OF STREET, ST.	and the second sec		
BT_ISland1_RD_TT BOOL		Island 1 Re					
BT_ISLAND1_TT FUNCTION BLOCK			(C) < 🚺 🗸 (C)	export		Search CJ export	Q
BT_Island1_wAddDiagCode	WORD	FB				<u> </u>	
BT_ISland1_wAppAck WORD		Application	Organize 🔻 Ne	w folder		9==	- (2)
SBT_Island1_wAppDiag WORD SBT_Island1_wDiagCode WORD		Application FB Diag Cod					•
SBT_ISTand1_wDTagCode word	WORD	FB Diag Cod		 Name 	<u>^</u>	 Date modified 	Type
5BT_Island1_wDiagCodeDev1	WORD	Dia	Tevorites	Indifie		Date mounieu	Type
BT_Island1_wDiagCodeDev3	WORD	Dia	Contract of the local division of the local				
BT_Island1_wDiagCodeDev4	WORD	Dia			No items match	your search.	
BT_Island1_wDiagCodeDev5	WORD	Dia	(# Computer	=			
BT_Island1_wDiagCodeLPSD0	WORD	Dia	and the second second	-			
BT_Island1_xAcceptDiffLogic	BOOL	Acc					
SBT_ISland1_xAckDev1 BOOL		Acknowledge	C Drophess				
SBT_Island1_xAckDev2 BOOL		Acknowledge	12 Recent Places				
SBT_Island1_xAckDev3 BOOL		Acknowledge	and second second				
BT_Island1_xAckDev4 BOOL		Acknowledge					
BT_Island1_xAckDev5 BOOL		Acknowledge					
SBT_IS]and1_xAckLPSDO BOOL		Acknowledge	and comments				
BT_ISland1_XACKOP BOOL		Operator Ad	Documents				
BT_ISland1_xActivate BOOL BT_ISland1_xActive BOOL		Activate FE FB Active					
BT_ISTANDI_XACUTVE BOOL BT_ISTANDI_XCOK BOOL		Communicati	a militar				
BT Island1 xDevError BOOL		Operator Ac	Pictures	* *	III		•
BT_Island1_xDiffLogicDetected	BOOL	Dif					
BT_ISland1_xError BOOL	BOOL	Error	File name:	*.CJS			-
BT_Island1_x0pAckReg BOOL		Power Up Re	The nume.				·
BT_ISland1_XPUR BOOL		Device Erro	Save as type:	All Files (*.*)			•
BT_Island1_xRunLogic BOOL		SBT LOGIC R					
BT_ISLAND_1 FUNCTION BLOCK							
		SBT V2 Safe	Hide Folders	Encoding:		Save	Cancel
SBT_V2_SFPRG FUNCTION BLOCK				Encounty.		Juve	concer
strFilename STRING[40]			- Inde Folders				
BT_V2_SFPRG FUNCTION BLOCK strFilename STRING[40] TT_TMR_PV UINT dtSBT Island1 STRUCT D5000		1 Transmissi 1 Safety Dat					

3. Save the file as **.CJS**.

1



Import Files in Tag Editor

Select the driver in Tag Editor and click on the **Import Tags** button to start the importer.

	Tags	×								
+	—	X	đ	ß	>]	Þ	A 9B	B>	* *	1
Data	1		^		-	Та	g URI			

The following dialog shows which importer type can be selected.

HMIStudio		×
Multiple tag importers	are available for this protocol. Please select the importer typ	oe and continue.
Version	Туре	<u>^</u>
Allen-Bradley L 5X v 1. 1	Hierarchical	
Allen-Bradley RSLogix5000 v	15 Linear	=
Allen-Bradley CCW v15	Linear	_
Omron Sysmac v15	Linear	
Omron CX-One v15	Linear	
Tag Editor exported xml	General	-
	ОК	Cancel

Select Omron Sysmac to import a .NJF Tags file or Omron CX-One to import a .CJF Tags file.

Once the importer has been selected, locate the Tags file and click **Open**. The system will ask for User Defined structures **.CJS** file. If not required, skip the dialog by clicking on Cancel button.

Tags included in the symbol file are listed in the tag dictionary. The tag dictionary is displayed at the bottom of the screen.

	Tags ×	Protocols						•
+	- 👗] [> 💲 🖻 🕅	R Or Search	Filter by: Data	▼ Ite	ms used:6/10000 Protocol: Show a	I 💽 Show all tags 🔅 🗖
Data		^	Туре	Comment		^	Property	Value
	Modbus TCP:		Container				✓ Driver	
- N		on Modbus(1-ba	iseu)				Model	Modicon Modbus(1-based)
		egisters 1	unsignedShort				Protocol	Modbus TCP:prot1
		egisters 2	unsignedShort				V Dictionary	
		egisters 3	unsignedShort				Array	false
-	- MRTU1		unsignedShort unsignedShort				Array size	0
	MRTU2		unsignedShort				Arravindex.Subindex	400003
	- MRTU4		unsignedShort					-00005
	- MRTU5		unsignedShort				Comment	
	Piletos		anaighteachtaí e				Data type	unsignedShort

Toolbar item	Description
	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
K 湖	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.
	Example of both checked and unchecked result:

Toolbar item	Description
	Tops* Type Octoberry 13 Type • • Octoberry 13 Container • • Octoberry 13 Ontriner • • Octoberry 14 Ontriner <t< th=""></t<>
P - Search	Searches tags in the dictionary basing on filter combox item selected.

i

Note: When importing the array data types, the importer is expanding them creating individual Tags per each array element; this is valid for all the data types, except for arrays of boolean. In this case they are imported as "boolean-32" and the single array element can be addressed using "Tag Index" parameter from "Attach to…" dialog.

Controller Model Micro800

The Ethernet/IP CIP driver provides an easy and reliable way to connect to Allen-Bradley Micro800 controllers.

The scope of variables into a Micro800 controller can be local to a program or global:

Scope	Description
Local Variables	Program-scoped Tags. Tags are assigned to a specific program in the project and available only to that program.
	These Tags are not supported within this driver.
Global Variables	Controller-scoped Tags. Tags belong to the controller in the project and are available to any program in the project.
	These Tags are supported within this driver.

Export ISAXML file using Connected Component Workbench

To export .ISAXML global variables including I/O tags:

- 1. Select Device tab.
- 2. Expand Export item.
- 3. Select Export Device.

demo 🌈	nnect	ted Compo	nents Workbench	Standard Edi	tion		
File Edit View			Communication		Help	p	
i 🔁 🚅 🔒 🐰	:=	Configure				- 🏼 😤 😒 :	🚽 🛹 Disc
<mark>Project Organizer</mark> Name: demo	2 2	Build Connect		F7			
Micro850	业 1	Download Upload			D		Remote Mode:
		Diagnose Secure		Þ	1 Dioad	Tiagnose Secure	Rais Monito
Glot	Ş.	Change Co	ontroller	•		Alex-Bradier	• <u>aaaaa</u>
User	2	Import Export		•		Export Device	3
		Document	: Generator (print)		X	Variables to Excel	

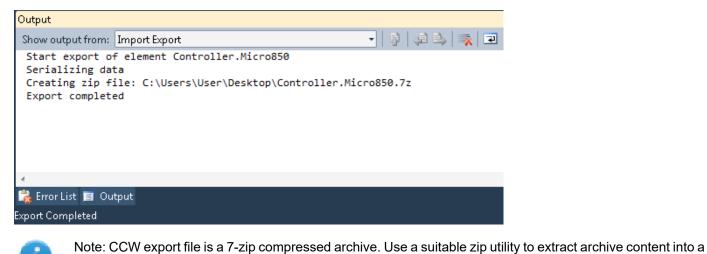
- 4. Click on Export Exchange File tab.
- 5. Click **Export** button.

Import Export	▼ □ ×
📑 Import Exchange File 🔛 Export Exchange File	
Set Password	
Password	
Password	
Confirm Password	
Element Exported Controller.Micro850	
5 Export CI	ose

6. Choose a location where to save the export file and click **Save**.

Save As		—
😋 🗢 💻 Deskti	op 🕨 👻 🗸 Search Desktop	٩
Organize 🔻 Ne	w folder	:: • 🕡
 ✓ Favorites Desktop Downloads Recent Places ∠ibraries Documents Music Pictures Videos 	 ▲ Libraries ▲ Homegroup ▲ Admin ▲ Computer ▲ Network 	
Þ 🤣 Homegroup	🗣 👘 Controllect Acou (50, co.)s	
File name: Save as type:	Controller.Micro850 SevenZip files (*.7z)	•
) Hide Folders	Save	Cancel

7. When the export is completed successfully the output information is displayed:





local folder.

Select the driver in Tag Editor and click on the Import Tags button to start the importer.

	Tags	×								
+	—	X	D	ß	>]	₽	A 9B	B>	ŧ i ł	1
Data			^		-	Та	g URI			_

The following dialog shows which importer type can be selected.

HMIStudio			×
Multiple tag importers	are available for this pro	otocol. Please select the importer type and co	ontinue.
Version	Туре		-
Allen-Bradley L5X v1.1	Hierarchical		
Allen-Bradley RSLogix5000 v	15 Linear		=
Allen-Bradley CCW v15	Linear		
Omron Sysmac v15	Linear		
Omron CX-One v15	Linear		
Tag Editor exported xml	General		-
		OK Car	ncel

Select Allen-Bradely CCW v15 option.

Directory structure extracted from 7z file is something like: "..\<folder_name>\Micro8xx\Micro8xx\" Inside this last folder, select the Micro8xx.isaxml file as shown below:

	cro850\Micro850\Micro850	✓ Searce	h Micro850		_
Organize 🔻 📄 Open				•== -	0
🔆 Favorites	Name	Date modified	Туре	Size	
Nesktop	Demo_logic.annex	18/06/2015 09:58	ANNEX File	8 KB	
🐌 Downloads	Demo logic.isaxml	18/06/2015 09:58	ISAXML File	6 KB	
Recorded TV	Micro850.isaxml	18/06/2015 09:58	ISAXML File	21 KB	
🚹 Google Drive	Mod_Message.annex	18/06/2015 09:58	ANNEX File	16 KB	
Recent Places	Mod_Message.isaxml	18/06/2015 09:58	ISAXML File	8 KB	
	oee_calcs.annex	18/06/2015 09:58	ANNEX File	1 KB	
🥃 Libraries	oee_calcs.isaxml	18/06/2015 09:58	ISAXML File	7 KB	
	PID_Feedback.annex	18/06/2015 09:58	ANNEX File	1 KB	
🜏 Homegroup	PID_Feedback.isaxml	18/06/2015 09:58	ISAXML File	3 KB	
	PID_OutputRegulator.annex	18/06/2015 09:58	ANNEX File	1 KB	
🖳 Computer	PID_OutputRegulator.isaxml	18/06/2015 09:58	ISAXML File	3 KB	
🟭 Local Disk (C:)	PID_PWM_LD.annex	18/06/2015 09:58	ANNEX File	7 KB	
Controller.Micro850	PID_PWM_LD.isaxml	18/06/2015 09:58	ISAXML File	5 KB	
Micro850	PID_PWM_SCALE.annex	18/06/2015 09:58	ANNEX File	1 KB	
Micro850	PID_PWM_SCALE.isaxml	18/06/2015 09:58	ISAXML File	4 KB	
	PIDWaterLvl.annex	18/06/2015 09:58	ANNEX File	9 KB	

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

				-			
🕇 🗕 🖉 🗋 👌 🎝 🚯 🖄 R 🔎 Search 🛛 🍸 Filter by: Data 🔹 Items used: 6/10000 Protocol: Show all 💽 Show all tags 🚳 🗖							
Туре	Comment	^	Property	Value			
Container			✓ Driver				
			Model	Modicon Modbus(1-based)			
			Protocol	Modbus TCP:prot1			
			Y Dictionary				
				false			
unsignedShort			Array size	0			
unsignedShort			Arrayindex.Subindex	400003			
unsignedShort			Comment				
unsignedShort			Data type	unsignedShort			
	Container unsignedShort unsignedShort	Type Comment Container unsignedShort Unsigne	Type Comment Container UnsignedShort Unsign	Type Comment Property Container UnsignedShort UnsignedShort UnsignedShort UnsignedShort UnsignedShort UnsignedShort UnsignedShort UnsignedShort UnsignedShort Comment UnsignedShort Comment			

Toolbar item	Description
B ≹	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
「「」	Update Tag(s).

Toolbar item	Description		
	Click on this icon to update the tags in the project, due a new dictionary import.		
R	Check this box to import all sub-elements of a tag. Example of both checked and unchecked result:		
	Tags* x + - X Data Type Obta Type Data Type Application Container Opto Container Opto Application Container PC PRG Container Container Image: X Image: X Image: X Image: X Image: X Application Container Opto Container Container Container Image: X Image: X Image: X Image: X Image: X Image: X Application Container Opto Container Container Container Container Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X Image: X		
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.		

Adding an alias name to a protocol

Tag names must be unique at project level, however, the same tag names might need to be used for different controller nodes (for example when the HMI device is connected to two devices running the same application).

When creating a protocol you can add an alias name that will be added to tag names imported for this protocol.

In the example, the connection to a certain controller is assigned the name **Node1**. When tags are imported for this node, all tag names will have the prefix **Node1** making each of them unique at the network/project level.

^ ~ * [🖻 🖷 🔁	Modbus TCP:prot1		-] 🗊 🚍		
Name 🛆 Group		Group	Driver		ss	Comment
Node'/ Gal_hodrotia		/X212 (West	Bue TCP pet 1	1 11 Durnig	with wi	
Node1/Cala, Nodrola		(HERE) William	Bue TCP pot1	1 12 Durwig	nel@hot	
Node1/IN_W/RTER_law	and in the second s	(X1232) Witted	Bue TCP pet 1	1 DDurwigr	with with	
Note1/OBOGFERICH			Bue TCP petril	1 245 Dunie	real Short	
Note1/OUT_BAT_Note	Profile.	Add West	Bue TOP pet 1	1 1 Durwigr	w/Short	
NedleT/FE_D/RTX_bodies	dia .		Bue TCP pet 1	1.2 Durwigr	with the	
No de 1, MA/H (TELE)		/X212 Wind	Bue TCP pet 1	1 3 Durwigr	with with	
Node1/Water_level		A000 (240)	alfas Effitiansfi	1 10 0 unsig	edShort	
	۲	Select Network node id				
	۲	Select Network node id	Model		Alias	
	۲		Model		Alias Node1	
	۲	Slave Id		ikua:		
	۲	Slave Id	Hullor mul	ikua:	Node1	
	•	Slave Id	Hullor mul	ikua:	Node1	
	•	Slave Id	Hullor mul	ikua:	Node1	
		Slave Id	Hullor mul	ikua:	Node1	
taaname Water Jevel	me	Slave Id	Hullor mul	ikua:	Node1 Node2	
tagname 🗸		Slave Id	Hullor mul	ikua:	Node1	Cancel
taoname Water_level		Slave Id	Hullor mul	ikua:	Node1 Node2	Cancel
tagname Water_level		Slave Id	Hullor mul	ikua:	Node1 Node2	Cancel
tagname Water_level		Slave Id	Hullor mul		Node1 Node2	Cancel
tagname Water_level		Slave Id	Hullor mul	ha ha	Node1 Node2 Ok	Cancel
Water_level		Slave Id	Hullor mul		Node 1 Node 2 Ok	Cancel



Note: Aliasing tag names is only available for imported tags. Tags which are added manually in the Tag Editor do not need to have the Alias prefix in the tag name.

The Alias string is attached on the import. If you modify the Alias string after the tag import has been completed, there will be no effect on the names already present in the dictionary. When the Alias string is changed and tags are re-imported, all tags will be re-imported with the new prefix string.

Node Override IP

The protocol provides the special data type Node Override IP which allows you to change the IP address of the target controller at runtime.

This memory type is an array of 4 unsigned bytes, one per each byte of the IP address.

The Node Override IP is initialized with the value of the controller IP specified in the project at programming time.

Node Override IP	PLC operation
0.0.0.0	Communication with the controller is stopped, no request frames are generated anymore.
Different from 0.0.0.0	It is interpreted as node IP override and the target IP address is replaced runtime with the new value.

If the HMI device is connected to a network with more than one controller node, each node has its own Node Override IP variable.



Note: Node Override IP values assigned at runtime are retained through power cycles.

Hostname DNS or mDNS

In addition to the array of bytes, string memory type can be selected to be able use the DNS or mDNS hostname as an alternative to the IP Address.

Ethernet/IP CIP			x
Ethernet/IP CIP			
Memory Type Node Override IP 🔻	ArrayIndex SubIn	ndex T	
Data Type	Arraysize Conver	rsion +/-	
Folder Name	Structure Name	Tag Name	_
	ОК	Cancel Apply	Help

Tag Import

Select the driver in Tag Editor and click on the Import Tags button to start the importer.



The following dialog shows which importer type can be selected.

HMIStudio		x
Multiple tag importers	are available for this protocol. Pl	ease select the importer type and continue.
Version	Туре	^
Allen-Bradley L5X v1.1	Hierarchical	
Allen-Bradley RSLogix5000 v	15 Linear	E
Allen-Bradley CCW v15	Linear	=
Omron Sysmac v15	Linear	
Omron CX-One v15	Linear	
Tag Editor exported xml	General	
		OK Cancel

Importer	Description		
Allen-Bradley L5X v1.1	Requires a .L5X file.		
Hierarchical	Check Controller Model Logix 5000 for more details.		
	All variables will be displayed according to RSLogix5000 Hierarchical view.		
Allen-Bradley	Requires a .CSV and .L5X (optional) files.		
RSLogix5000 v15 Linear	Check Controller Model Logix 5000 for more details.		
	All variables will be displayed at the same level.		
Allen-Bradley CCW v15	Requires a .ISAXML file.		
Linear	Check Controller Model Micro800 for more details.		
	All variables will be displayed at the same level.		
Omron Sysmac v15	Requires a .NJF file.		
Linear	Check Controller Model Omron Sysmac for more details.		
	All variables will be displayed at the same level.		

Importer	Description		
Omron CX-One v15	Requires a .CJFand .CJS (optional) files.		
Linear	Check Controller Model Omron Sysmac for more details.		
	All variables will be displayed at the same level.		
Tag Editor exported xml Select this importer to read a generic XML file exported from Tag appropriate button.			
	Tags x		
	🕂 — 🎽 🔞 🔊 🚺 🚺		
	Data Tag URI		

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

🕨 — 🎽 🕲 🖄	D 🔂 🚯	R 🔎 - Search 🍸 Filter by: Data	▼ Ite	ems used	6/10000 Protocol: Show a	ll 🛛 🗹 Show all tags 🔅
ata	Туре	Comment	^	Proper	ty	Value
Modbus TCP:prot1	Container			Y Dri	iver	
Model: Modicon Modbus(1-ba	sea)				Model	Modicon Modbus(1-based)
Holding Registers 1	unsignedShort				Protocol	Modbus TCP:prot1
- Holding Registers 2	unsignedShort			V Die	tionary	
 Holding Registers 3 	unsignedShort					
- MRTU1	unsignedShort				Array	false
- MRTU2	unsignedShort				Array size	0
MRTU3	unsignedShort				Arrayindex.Subindex	400003
- MRTU4	unsignedShort				Comment	
- MRTU5	unsignedShort				Data type	unsignedShort

Toolbar item	Description
	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
₩.	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.
	Example of both checked and unchecked result:

Toolbar item	Description
	Tage* x Tage* x + - ∠ ③ □ >1 ▷ ♣ □ ∅ ∅3 ℝ
	Data Type Con Data Type Con 4 CODESYS V3 Entropol2 Container 4 Application Application Application Application Applica
	• test/ray/lag Array(1) INT -01 901 911 101 -03 901 911 101 -13 911 11 101 -13 911 -13 101 -13 911 -13 101 -13 911 -13 101 -13 911 -13 101 -14 911 911 101 -15 911 101 101 -15 911 101 101 -15 911 101 101 -15 911 101 101 -15 911 101 101 -15 911 101 101 -15 911 101 101 -15 911 101 101 -15 911 101 101 -15 911 101 101 -15 911 101
P - Search	Searches tags in the dictionary basing on filter of box item selected.

Communication status

Current communication status can be displayed using System Variables. See "System Variables" section in the main manual.

Error	Cause	Action
NAK	The controller replies with a not acknowledge.	-
Timeout	A request is not replied within the specified timeout period.	Check if the controller is connected and properly configured to get network access.
Invalid response	The device did received a response with invalid format or contents from the controller .	Ensure the data programmed in the project are consistent with the controller resources.
General Error	Unidentifiable error. Should never be reported.	Contact technical support.

Codes supported for this communication driver:

Fatek FACON ETH

The Fatek FACON ETH communication driver has been designed to connect HMI devices to a Fatek FACON PLC through Ethernet connection.

Protocol Editor Settings

Fatek FACON ETH		×
PLC Network		ОК
IP address	0.0.0.0	Cancel
Port	500	
station	1	
Timeout	2000	
PLC Models		
FB Series		

Adding a protocol

To configure the protocol:

- 1. In Config node double-click Protocols.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the PLC list.

The protocol configuration dialog is displayed.

Element	Description
IP Address	Ethernet IP address of the PLC.
Port	Port number used to communicate with PLC.
station	station number according to PLC configuration.
Timeout	Time delay in milliseconds between two retries in case of missing response from the PLC.
PLC Models	PLC model available:

Element	Description				
	FB Series				
PLC Networ k	IP address for all controllers ir enable multiple connections.	n multiple connections	PLC Network must be so	elected to	
	Fatek FACON ETH				
	PLC Network		ОК		
	IP address	0.0.0.0	Cancel		
	Port 5	00			
	station Fat	ek FACON ETH			3
	Timeout			ОК	
	PLC Models FB Series	IP address	0.0.0.1	Cancel	
		Port	500		
		station	1		
	Slaves	Timeout	2000		
	Slave Id	PLC Models			
		FB Series			

Tag Editor Settings

In Tag Editor select the protocol Fatek FACON ETH.

Add a tag using [+] button. Tag setting can be defined using the following dialog:

Fatek FACON ETH		×
Fatek FACON ETH		
Memory Type	Offset SubIndex	
Input Discrete 👻	0 • 0 •	
Data Type	Arraysize Conversion	
boolean 👻	0	-/-
	OK Cancel A	Help

Element	Description	
Memory Type	Memory Type	Description
,,	Input Discrete	X resources. Corresponding to External Digital Input Point.
	Output Relay	Y resources. Corresponding to External Digital Output Point.
	Internal Relay	M resources. Corresponding to PLC internal memory.
	Step Relay	S resources.
	Timer Discrete	T resources.
	Counter Discrete	C resources.
	Timer Register	Current Time Value Register.
	Counter Register	Current Counter Value Register.
	Data Register - HR	R resources.
	Data Register - DR	D resources.
	Run	Boolean value. Corresponding to PLC status.
	Node Override IP	See Special Data Types for specifications.
Offset	Starting address for the Tag. The possible range depend on PLC model selected.	
SubIndex	This allows resource offset selection depending on the selected data type.	
Data Type	Available data types:	
	• boolean	
	• byte	
	• short	
	intunsignedByte	
	unsignedShort	
	unsignedInt	
	• float	
	double	
	• string	
	binary	
	See "Programming concepts"	section in the main manual.
	Note: To define arra brackets (byte[], sho	ys, select one of Data Type format followed by square prt[]…).

	Description				
Arraysize	In case of array tag, this property represents the number of array elements.				
	 In case of string tag, this property represents the maximum number of bytes available in the string tag. 				
	set to UTF-8 o	of bytes corresponds to number of string characters if Encoding property is r Latin1 in Tag Editor. operty is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one uires 2 bytes.			
Conversion	Conversion to	be applied to the tag.			
	inv,swap2	Allowed Configured			
	Depending on	BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits Cancel OK data type selected, the list Allowed shows one or more conversion types.			
	Value	Description			
	Inv bits	inv: Invert all the bits of the tag.			
		Example: $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)			
	Negate	$1001 \rightarrow 0110$ (in binary format)			
	Negate	$1001 \rightarrow 0110 \text{ (in binary format)} \\9 \rightarrow 6 \text{ (in decimal format)}$			
	Negate AB -> BA	$1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)neg: Set the opposite of tag value.Example:			
		$1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)neg: Set the opposite of tag value. <i>Example:</i> $25.36 \rightarrow -25.36$			
	AB -> BA	$1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format) neg : Set the opposite of tag value. <i>Example:</i> $25.36 \rightarrow -25.36$ swapnibbles : Swap nibbles in a byte. <i>Example:</i> $15D4 \rightarrow 514D$ (in hexadecimal format) $5588 \rightarrow 20813$ (in decimal format) swap2 : Swap bytes in a word.			
	AB -> BA	$1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format) neg : Set the opposite of tag value. <i>Example:</i> $25.36 \rightarrow -25.36$ swapnibbles : Swap nibbles in a byte. <i>Example:</i> $15D4 \rightarrow 514D$ (in hexadecimal format) $5588 \rightarrow 20813$ (in decimal format)			

Element	Description		
	Value	Description	
	-> GHEFCDAB	<i>Example:</i> 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)	
	ABCNOP - > OPMDAB	swap8 : Swap bytes in a long word. Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) $0\ 10000000110$ 0001110010111011010001011010000111001010	
	BCD Select conversi	 bcd: Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> 23 → 17 (in decimal format) 0001 0111 = 23 0001 = 1 (first nibble) 0111 = 7 (second nibble) on and click +. The selected item will be added to list Configured. 	
	If more convers Configured).	ions are configured, they will be applied in order (from top to bottom of list outtons to order the configured conversions.	

Node Override IP

The protocol provides the special data type Node Override IP which allows you to change the IP address of the PLC at runtime.

This memory type is an array of 4 unsigned bytes, one per each byte of the IP address.

The Node Override IP is initialized with the value of the PLC IP specified in the project at programming time.

Node Override IP	Modbus operation
0.0.0.0	Communication with the controller is stopped, no request frames are generated anymore.
Different from 0.0.0.0	It is interpreted as node IP override and the target IP address is replaced runtime with the new value.

If the HMI device is connected to a network with more than one PLC node, each node has its own Node Override IP variable.



Note: Node Override IP values assigned at runtime are retained through power cycles.

Fatek FACON ETH				x
Fatek FACON ETH				
Memory Type	Offset	SubIndex		
Data Type	0	Conversion		
unsignedByte [] 🔻	4		+/-	
	C	Cancel	Apply	Help

Fatek FACON SER

The Fatek FACON SER communication driver has been designed to connect HMI devices to a Fatek FACON PLC through Serial connection.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the **PLC** list.

The protocol configuration dialog is displayed.

Fatek FACON SER				x
PLC Network		Comm	ОК	
Node ID	1		Cancel	
PLC Models FB Series				

Element	Description
Node ID	Serial node associated to the PLC.
PLC Models	PLC model available:
	FB Series
PLC Network	IP address for all controllers in multiple connections. PLC Network must be selected to enable multiple connections.

Element	Description	
	Fatek FACON SER	
	PLC Network	Comm OK
	Node ID	-1 Cancel
	PLC Models FB Series	
	C	Fatek FACON SER
		ОК
	Slaves	Node ID Cancel
	Slave Id	PLC Models FB Series
Comm	Comm Parameter Dialo	communication parameters setup dialog.
		ОК
	Port	
	Baudrate	com1
	Parity	9600 V
	Data bits	8
	Stop bits	
	Mode	RS-485

Element	Description	
	Element	Parameter
	Port	Serial port selection.
		COM1: device PLC port.
		 COM2: computer/printer port on panels with 2 serial ports or optional Plug-In module plugged on Slot 1/2 for panels with 1 serial port on-board.
		 COM3: optional Plug-In module plugged on Slot 3/4 for panels with 1 serial port on-board.
	Baudrate, Parity, Data Bits, Stop bits	Serial line parameters.
	Mode	Serial port mode. Available modes:
		• RS-232.
		• RS-485 (2 wires).
		• RS-422 (4 wires).

Tag Editor Settings

In Tag Editor select the protocol Fatek FACON SER.

Add a tag using [+] button. Tag setting can be defined using the following dialog:

Fatek FACON SER		
Fatek FACON SER		
Memory Type	Offset SubIndex	
Input Discrete 🔻	0 •	
Data Type	Arraysize Conversion	
boolean 👻	0 +/-	
	OK Cancel Apply Help	

Element	Description		
Memory Type	Memory Type	Description	
iype	Input Discrete	X resources. Corresponding to External Digital Input Point.	
	Output Relay	Y resources. Corresponding to External Digital Output Point.	
	Internal Relay	M resources. Corresponding to PLC internal memory.	
	Step Relay	S resources.	
	Timer Discrete	T resources.	
	Counter Discrete	C resources.	
	Timer Register	Current Time Value Register.	
	Counter Register	Current Counter Value Register.	
	Data Register - HR	R resources.	
	Data Register - DR	D resources.	
	Run	Boolean value. Corresponding to PLC status.	
Offset	Starting address for the Tag. The possible range depend on PLC model selected.		
SubIndex	This allows resource offset selection depending on the selected data type.		
Data Type	Available data types:		
	• boolean		
	• byte		
	• short		
	 int unsignedByte 		
	unsignedShort		
	unsignedInt		
	• float		
	• double		
	• string		
	binary		
	See "Programming concepts" section in the main manual.		
	Note: To define arra brackets (byte[], sho	ys, select one of Data Type format followed by square ort[]…).	

	Description			
Arraysize	In case	In case of array tag, this property represents the number of array elements.		
	 In case of string tag, this property represents the maximum number of bytes available in the string tag. 			
	Note: number of bytes corresponds to number of string characters if Encoding property is set to UTF-8 or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2 bytes.			
Conversion	Conversion to	be applied to the tag.		
	Conversion			
	inv,swap2	Allowed Configured BCD Inv bits		
	Depending on	AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits Cancel OK data type selected, the list Allowed shows one or more conversion types.		
	Value	Description		
	Value Inv bits	Description inv: Invert all the bits of the tag.		
		inv: Invert all the bits of the tag. <i>Example:</i> $1001 \rightarrow 0110$ (in binary format)		
	Inv bits	inv: Invert all the bits of the tag. <i>Example:</i> $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)		
	Inv bits	inv: Invert all the bits of the tag. $Example:$ $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)neg: Set the opposite of tag value. $Example:$		
	Inv bits Negate	inv: Invert all the bits of the tag. $Example:$ $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)neg: Set the opposite of tag value. $Example:$ $25.36 \rightarrow -25.36$		
	Inv bits Negate AB -> BA ABCD ->	inv: Invert all the bits of the tag. $Example:$ $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)neg: Set the opposite of tag value. $Example:$ $25.36 \rightarrow -25.36$ swapnibbles: Swap nibbles in a byte. $Example:$ $15D4 \rightarrow 514D$ (in hexadecimal format)		
	Inv bits Negate AB -> BA	inv: Invert all the bits of the tag. $Example:$ $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)neg: Set the opposite of tag value. $Example:$ $25.36 \rightarrow -25.36$ swapnibbles: Swap nibbles in a byte. $Example:$ $15D4 \rightarrow 514D$ (in hexadecimal format) $5588 \rightarrow 20813$ (in decimal format)		

Element	Description		
	Value	Description	
	-> GHEFCDAB	<i>Example:</i> 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)	
	ABCNOP -	swap8: Swap bytes in a long word.	
	> OPMDAB	Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) $0\ 10000000110$ $0001\ 1100101110110100001011010000111001010101$	
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)	
	Select conversi	on and click +. The selected item will be added to list Configured .	
	If more convers Configured).	ions are configured, they will be applied in order (from top to bottom of list	
	Use the arrow b	outtons to order the configured conversions.	

GE Intelligent Platforms SNP

The GE Intelligent Platforms SNP driver can be used to connect the HMI device to the GE controllers through serial connection using the native and proprietary SNP communication protocol.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the $\ensuremath{\text{PLC}}$ list.

The protocol configuration dialog is displayed.

GE	Intelligent Platforms	SNP	x
	PLC Network	Comm OK	
	Alias	Cancel	
	Protocol type	SNP 👻	
	PLC Models		
	90-20 90-30 311		
	90-30 331 90-70 731/732		
	90-70 771/772 90-70 780/781/782 90-30 340/341 90-30 Micro 14 poir 90-30 313/321/323 90-30 351/352/360 90-70 788 90-30 Micro 28 poir 90-30 350/374 VersaMax Nano (NI VersaMax Micro 14 VersaMax Micro 14 VersaMax Micro 28 VersaMax CPU001 VersaMax CPU002 VersaMax CPU002 PACSystem RX3i VersaMax Micro 64 Fanuc CNC	3 0/363/364 nts DD 101, NDR001) points points 5, CPUE05)	-

Eleme nt	Description
Alias	Name identifying nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node.
PLC Model s	PLC models available.

Eleme nt	Descriptior	cription	
Proto col ype	Allows to se	lows to select between SNP and SNP-X protocol.	
Com n	If clicked displays the communication parameters setup dialog.		
	Comm Parameter Dialog		
	Port Baudr Parity Data t Stop t Mode	odd	
	Element Parameter		
	Port	Serial port selection.	
	 COM1: On-board port COM2: Optional Plug-in module plugged on slot#1 or slot#2 COM3: Optional Plug-in module plugged on slot#3 or slot#4 		
	Baudrat e, Parity, Data Bits, Stop bits	Serial line parameters.	

n		
Parameter		
 Serial port mode. Available modes: RS-232. RS-485 (2 wires). RS-422 (4 wires). 		
• RS-422 (4 Wires). Multiple controllers can be connected to one HMI device. To set-up multiple connections, select PLC networkand click Add to configure each slave GE Intelligent Platforms SNP Image: Platfor		

Tag Editor Settings

In Tag Editor select the protocol GE Intelligent Platforms SNP.

Add a tag using [+] button. Tag setting can be defined using the following dialog:

GE Intelligent Platform	s SNP			
Memory Type	Offset	SubIndex		
Register	• 1	0 -		
Data Type	Arraysize	Conversion		
boolean	•		+/-	
	_	OK Cance	Apply	

Element	Description		
Memory Type	Memory Type	Description	
	Register	R resource on PLC.	
	Discrete Input	I resource on PLC.	
	Discrete Output	Q resource on PLC.	
	Discrete Global	G resource on PLC.	
	Internal Coil	M resource on PLC.	
	Temporary Coil	T resource on PLC.	
	System Status	S resource on PLC.	
	Analog Input	AI resource on PLC.	
	Analog Output	AQ resource on PLC.	
	Clear I/O Fault	IOF resource on PLC.	
	Clear PLC Fault	PLF resource on PLC.	
Offset	Offset address where tag is located. Offset range depends on specific memory type and PLC model selected.		

Element	Description					
Data Type	Available data types:					
	• boolean					
	• byte					
	• short					
	• int					
	 unsignedByte unsignedShort 					
	unsignedInt					
	• float					
	• double					
	• string					
	• binary					
	See "Programming concepts" section in the main manual.					
	Note: To define arrays, select one of Data Type format followed by square brackets (byte[], short[]).					
Arraysize	In case of array tag, this property represents the number of array elements.					
	 In case of string tag, this property represents the maximum number of bytes available in the string tag. 					
	Note: number of bytes corresponds to number of string characters if Encoding property is set to UTF-8 or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2 bytes.					
Conversion	Conversion to be applied to the tag.					
	Conversion					
	inv,swap2 Allowed Configured					
	BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits Cancel OK					
	Depending on data type selected, the list Allowed shows one or more conversion types.					

Element	Description				
	Value Description				
	Inv bits	inv : Invert all the bits of the tag.			
		Example: 1001 \rightarrow 0110 (in binary format) 9 \rightarrow 6 (in decimal format)			
	Negate	neg: Set the opposite of tag value.			
		<i>Example:</i> 25.36 → -25.36			
	AB -> BA	swapnibbles: Swap nibbles in a byte.			
		Example: 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)			
	ABCD ->	swap2: Swap bytes in a word.			
	CDAB	<i>Example:</i> 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)			
	ABCDEFGH	swap4: Swap bytes in a double word.			
	-> GHEFCDAB	Example: 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)			
	ABCNOP -	swap8: Swap bytes in a long word.			
	> OPMDAB	Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) $0\ 1000000110$ 0001110010111011010001011010000111001010			
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format)			
		0001 0111 = 23 0001 = 1 (first nibble) 0111 = 7 (second nibble)			

L

Element	Description
	Select conversion and click +. The selected item will be added to list Configured .
	If more conversions are configured, they will be applied in order (from top to bottom of list Configured).
	Use the arrow buttons to order the configured conversions.

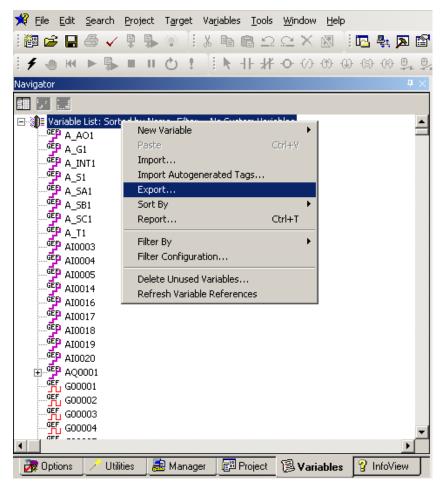
Tag Import

Exporting Tags from PLC

The GE Intelligent Platforms SRTP Ethernet driver support the Tag Import facility.

Variables can be exported by the controller programming software Proficy Machine Edition,

selecting "Variables" tab, then right mouse click and from context menu select the Export option as shown in following figure.



In the following dialog select then the file name and the file location on the computer.

📌 Eile Edit Search Project Target Variables Tools Window Help	
i 📴 🖆 💭 🖉 🍹 🖡 🔍 i 🐰 🖿 🛍 🗅 🗠 🗙 📗 📲	, 🔊 🔊
╡ ≠ @ ⋈ ▶ \$} ■ Ⅱ ᠔ ! │ ト ┼┼ ↩ ⊘ @ @ @) <u>0</u> , <u>0</u> ,
Navigator	Т ×
Image List: Sorted by Address, Filter = No System Variables Target1 GEP V_pr_AI0001 GEP AI0003	
Export ?	
Speichem 🤷 Eigene Dateien 💌 🗢 🖻 📸 🕶	
Eigene Bilder	
Dateiname: Machine_Edition_Variables Speichern]
Dateityp: Comma Separated Values(*.csv) Abbrechen	
- GEP AQ0001[008] GEP AQ0001[007] GEP AQ0001[008]	
📑 Options 🦯 Utilities 😹 Manager 🛱 Project 🔞 Variables 💡 Info	View

Importing Tags in Tag Editor

Select the driver in Tag Editor and click on the Import Tags button to start the importer.



The following dialog shows which importer type can be selected.

	×
Multiple tag impor	ters are available for this protocol. Please select the importer type and continue.
Version	Туре
Proficy Machine v1.0	Linear
Tag Editor exported xml	General
	OK Cancel

Importer	Description				
Proficy Machine v1.0	Requires an .csv file.				
Linear	All variables will be displayed at the same level.				
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.				
	Tags x				
	+ - 🎽 🕲 🔎 🔰 🚺 🕨 🗟 🛛				
	Data Tag URI				

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags × Protocols						
+ - 👗 🕲 🔊	D 🗞 🖬 🚮	R 🔎 - Search 🌱 Filter by: Data	▼ Ite	ems u	used:6/10000 Protocol: Show all	🗹 Show all tags 🔅 📋
Data	Туре	Comment	^	Pro	operty	Value
Modbus TCP:prot1	Container			~	Driver	
Model: Modicon Modbus(1-based)					Model	Modicon Modbus(1-based)
- Holding Registers 1	unsignedShort				Protocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			~	Dictionary	
- Holding Registers 3 - MRTU1	unsignedShort unsignedShort				Array	false
- MRTU2	unsignedShort				Array size	0
- MRTU3	unsignedShort				Arravindex.Subindex	400003
- MRTU4	unsignedShort				Comment	100005
-MRTU5	unsignedShort				Data type	unsignedShort

Toolbar item	Description			
ka	Import Tag(s).			
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project			
B	Update Tag(s). Click on this icon to update the tags in the project, due a new dictionary import.			
R	Check this box to import all sub-elements of a tag. Example of both checked and unchecked result:			
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.			

GE Intelligent Platforms SRTP

The GE Intelligent Platforms SRTP driver can be used to connect the HMI device to the GE controllers through Ethernet connection using the native and proprietary SRTP communication protocol.

Protocol Editor Settings

Add (+) a driver in the Protocol editor and select the protocol called "GE Intelligent Platforms SRTP" from the list of available protocols.

GE Intelligent Platforms SRT	ſP	—
PLC Network		ОК
Alias		Cancel
IP address	0 . 0 . 0 . 0	
Port	18245	
Timeout	2000	
PLC Models		
90-30 311 90-30 331 90-70 731/732 90-70 771/772 90-70 780/781/782/78 90-30 340/341	9/914/915/924/925/928/935	•

Element	Description			
Alias	Name to be used to identify nodes in network configurations. The name will be added as prefix to each tag name imported for each network node			
IP Address	The IP address of the Ethernet interface of the controller			
Port	Communication Port number for the Ethernet interface			
Timeout	The time the protocol waits the answer from the controller before issuing a new retry.			

Element	Description					
PLC Models	List of compatible controller models. Make sure to select the right model in this list when configuring the protocol.					
PLC Network	The protocol supports connection to multiple controllers. To enable this, check the "PLC Network" check box and provide the configuration per each node.					
	GE Intelligent Platforms S	RTP	8			
	V PLC Network		GE Intelligent Platforms S	SRTP	×	1
	Alias		Alias		OK Cancel	
	IP address Port	0.0.0.0	IP address	0.0.1		
	Timeout	2000	Port	18245		l
	PLC Models		Timeout	2000		
	90-30 311 90-30 331 90-70 731/732 90-70 771/772 90-70 780/781/782/ 90-30 340/341	789/914/915/924/925/928/935	PLC Models 90-30 311 90-30 331 90-70 731/732 90-70 771/772	1780 /014 /015 /026 /025	^	
	Slaves	Add	90-70 780/781/782/ 90-30 340/341	/789/914/915/924/925/928/935	-	l
	Slave Id	Model Alias				

Data Types

The import module supports variables of standard data types as per the following list.

- BOOL
- BYTE (8-bits unsigned integers)
- DINT (32-bits signed integers)
- DWORD (32-bit bit strings, displayed as unsigned integers)
- INT (16-bit signed integers)
- REAL (32-bit floating point data)
- STRING (character string)
- UINT (16-bit unsigned integers)
- WORD (16-bit bit strings, displayed as unsigned integers)



Note: User defined structure and predefined structures are not supported. 64-bit data are also not supported

Tag Conversion

Conversion to be applied to the tag.

Conversion			
inv,swap2	Allowed		Configured
	BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits	+ - < >	Inv bits ABCD->CDAB
		[Cancel OK

Depending on data type selected, the list **Allowed** shows one or more conversion types.

Value	Description
Inv bits	inv : Invert all the bits of the tag.
	Example: $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)
Negate	neg : Set the opposite of tag value.
	<i>Example:</i> 25.36 → -25.36
AB -> BA	swapnibbles: Swap nibbles in a byte.
	<i>Example:</i> 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)
ABCD -> CDAB	swap2: Swap bytes in a word.
	Example: 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)
ABCDEFGH ->	swap4: Swap bytes in a double word.
GHEFCDAB	Example: $32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format) $855441236 \rightarrow 1426062386$ (in decimal format)

Value	Description
ABCNOP -> OPMDAB	<pre>swap8: Swap bytes in a long word. Example: 142.366 → -893553517.588905 (in decimal format) 0 10000000110 0001110010111011010001011010000111001010</pre>
BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)

Select conversion and click +. The selected item will be added to list Configured.

If more conversions are configured, they will be applied in order (from top to bottom of list Configured).

Use the arrow buttons to order the configured conversions.

Special Data Types

The GE Intelligent Platforms SRTP driver provides one special data type called "Node Override IP".

The Node Override IP allows changing at runtime the IP address of the target controller you want to connect. This memory type is an array of 4 unsigned bytes, one per each byte of the IP address.

The Node Override IP is initialized with the value of the controller IP specified in the project at programming time.

If the IP Override is set to 0.0.0.0, all the communication with the node is stopped, no request frames are generated anymore.

If the IP Override has a value different from 0.0.0.0, it is interpreted as node IP override and the target IP address is replaced at runtime with the new value.

In case the panel has been configured to access to a network of controllers, each node has its own Override variable.



Note: the IP Override values assigned at runtime are retained through power cycles.

GE Intelligent Platforms SR	TP			×
GE Intelligent Platforms SRT	P			
Memory Type	Offset	SubIndex		
Node Override IP 👻	0	• •		
D. 1. T				
Data Type	Arraysize	Conversion		
unsignedByte 🛛 👻	8	1	+/-	
	C	Cancel	Apply Help	•

Aliasing Tag Names in Network Configurations

Tag names must be unique at project level; it often happens that the same tag names are to be used for different controller nodes (for example when the HMI is connected to two devices that are running the same application). Since tags include also the identification of the node and Tag Editor does not support duplicate tag names, the import facility in Tag Editor has an aliasing feature that can automatically add a prefix to imported tags. With this feature tag names can be done unique at project level.

The feature works when importing tags for a specific protocol. Each tag name will be prefixed with the string specified by the "Alias". As shown in the figure below, the connection to a certain controller is assigned the name "Node1". When tags are imported for this node, all tag names will have the prefix "Node1" making each of them unique at the network/project level.

1:Page1 protocols*	Tags* 🗙					
+ - ^ ~ & 🖻 🖷	>] GE Intel	ligent Platforms SRTP:p	rot2 🔻 👼 🗖			
Name	Group	Driver	Address	E	ncoding	Comment
Node1/C1_Imp	GE	E Intelligent Platforms S	192.168.0.1 1 bo	olean		
Node1/ZL_MVAutomat	GE	E Intelligent Platforms S	192.168.0.1 2 bo	olean		
Node1/C3_Imp		E Intelligent Platforms S				
Nod <mark>e1/Tuer_offen</mark>	011	Intelligent Platforme C		ologiti		
Noce1/Sek_PT		E Intelligent Platforms S				
Noce1/Stundenwechsel		E Intelligent Platforms S				
Note1/Vdat_akt		E Intelligent Platforms S				
▶ Node1/Vdat_reset	GE	EIntelligent Platforms S	192.168.0.1 M 93	boo		
	The Ular Di					
G	E Intelligent Pl	atforms SRTP				- ×-
	Network GE In	ntelligent Platforms SRT	P			
	GE I	realigence hadronna arch	F	_		
	Select PLC net	work Node			_	
	Slave Id	Slave Id Model Alias				
	192.168.0.1	90-30 3	311	Node 1		
	192, 168, 0, 2		311	Node2		
•						
Kā Ka						ʻ
tagname						
Vdat_reset M						
M00102 M						
_M00103 M						
_M00104 M						
Q00001 Q						
_Q00002 Q L						
_Q00003 Q			ОК	Cancel	Apply	Help
Q00004 Q						



Note: Aliasing tag names is only available when tags can be imported. Tags which are added manually in the Tag Editor do not need to have the Alias prefix in the tag name. The Alias string is attached to the tag name only at the moment the tags are imported using Tag Editor. If you modify the Alias string after the tag import has been completed, there will be no effect on the names already present in the dictionary. When the Alias string is changed and tags are imported again, all tags will be imported again with the new prefix string.

Tag Import

Exporting Tags from PLC

The GE Intelligent Platforms SRTP Ethernet driver support the Tag Import facility.

Variables can be exported by the controller programming software Proficy Machine Edition,

selecting "Variables" tab, then right mouse click and from context menu select the Export option as shown in following figure.

📌 Eile Edit Search	<u>Project Target Variables Tools Window H</u> elp
i 🔯 🖨 🖬 🏐 🗸	🖡 🗣 🖉 🛛 🕹 🖿 🖻 🗅 🗠 🗙 🕅 🗳 🗖 🗳
if 🕢 🖻 🕨 🖡	■Ⅱ❹!〗◣非凇ㅇળ◍◍◍◍▫▫▫
Navigator	₽ ×
□ Wariable List: Sort	New Variable Important Paste Ctrl+V Import Import Autogenerated Tags Export Sort By Filter By Import Filter Configuration Delete Unused Variables Refresh Variable References Export
	ies 🛃 Manager 🚰 Project 🔀 Variables 😵 InfoView

In the following dialog select then the file name and the file location on the computer.

×	<u>F</u> ile	<u>E</u> dit	<u>S</u> ea	rch	Pro	ject	T <u>a</u> rg	et	Vari	iable	s <u>T</u>	ools	<u>W</u> ir	ndow	He	lp				
e	3 🖻	; 日	6	\checkmark	ŧ	₽			Ж	Þ	Ē	Ω	2	\times	1		<u></u>	8 1	≽	8
1		- HH	►	₽.		П	Ċ	±.	1	k	-11-	, //	-0-	$\langle \rangle$	0	-(1)-	-(5)-	-(R)-	₽. 	
Navi	gator																		ņ	×
	i_;																			Í
	≬≣ V/ ⊡~ {	GEE		_AIO		y Ado	dress,	, Filto	er =	No S	ōyste	m Va	riabl	es						
	Еж	port																?	×	
	9	Speich	iern	<u>(</u>) Eig	jene	Datei	en					•	- 1		<u>*</u>	<u>;</u> -			
	D) ateina	ime:	N	1ach	ine_E	Editior	_Va	riabl	les					[Sp	eich	ern]	
	D) ateityj) :	0	Comn	na Se	eparat	ed V	alue	es(*.c	sv)			-		Ab	brecł	nen		
•		GEP	AQ00 AQ00 AQ00	001[0	007]														•	•
2	🔋 Opt	ions	2	Utili	ties		👌 Ma	nage	er j	[]	Proje	ect	Ø	Vari	able	:s]	<u>ଥ</u> ା	nfoV	iew	ŢĮ

Importing Tags in Tag Editor

Select the driver in Tag Editor and click on the Import Tags button to start the importer.



The following dialog shows which importer type can be selected.

	×
Multiple tag impor	ters are available for this protocol. Please select the importer type and continue.
Version	Туре
Proficy Machine v1.0	Linear
Tag Editor exported xml	General
	OK Cancel

Importer	Description						
Proficy Machine v1.0	Requires an .csv file.						
Linear	All variables will be displayed at the same level.						
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.						
	Tags x						
	+ - 🎽 🕲 🔎 🔰 🚺 🕨 🗟 🛛						
	Data Tag URI						

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags × Protocols						
+ - 👗 🕲 🔊	D 🗞 🖬 🚮	R 🔎 - Search 🌱 Filter by: Data	▼ Ite	ems u	used:6/10000 Protocol: Show all	🗹 Show all tags 🔅 📋
Data	Туре	Comment	^	Pro	operty	Value
Modbus TCP:prot1	Container			~	Driver	
Model: Modicon Modbus(1-based)					Model	Modicon Modbus(1-based)
- Holding Registers 1	unsignedShort				Protocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			~	Dictionary	
- Holding Registers 3 - MRTU1	unsignedShort unsignedShort				Array	false
- MRTU2	unsignedShort				Array size	0
- MRTU3	unsignedShort				Arravindex.Subindex	400003
- MRTU4	unsignedShort				Comment	100005
-MRTU5	unsignedShort				Data type	unsignedShort

Toolbar item	Description						
R	Import Tag(s).						
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project						
國	Update Tag(s). Click on this icon to update the tags in the project, due a						
	new dictionary import.						
R	Check this box to import all sub-elements of a tag. Example of both checked and unchecked result:						
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.						

Communication Status

The communication status can be displayed using the dedicated system variables. Please refer to the User Manual for further information about available system variables and their use.

Error	Notes
NAK	Controller replies with a not acknowledge.
Timeout	Request is not replied within the specified timeout period; ensure the controller is connected and properly configured for network access
Invalid response	The panel did receive from the controller a response, but its format or its contents or its length is not as expected; ensure the data programmed in the project are consistent with the controller resources.
General Error	Error cannot be identified; should never be reported; contact technical support

The status codes supported for this communication driver are:

GE SRTP

The GE SRTP communication driver has been designed to connect HMI devices to GE PLCs.

The driver allows symbolic communication with GE PLC model PacSystemRx3i.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In Config node double-click Protocols.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the **PLC** list.

The protocol configuration dialog is displayed.

GE SRTP		×
PLC Network		ОК
Alias		Cancel
IP address	0.0.0.0	
Port	18245	
Timeout	2000	
PLC Models		
90-30 311 90-30 331 90-70 731/732 90-70 771/772 90-70 780/781/782/7 90-30 340/341	789/914/915/924/925/928/935	

Element	Description	
Alias	Name identifying nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node.	
IP address	Ethernet IP address of the controller.	
Port	Port number used by the driver. The default value is 18245 .	
Timeout	Time delay in milliseconds between two retries in case of missing response from the server device.	
PLC Models	SAIA PLC models available:	

Element	Description
	• 90-30 311
	• 90-30 331
	• 90-70 731/732
	• 90-70 771/772
	 90-70 780/781/782/789/914/915/924/925/928/935
	• 90-30 340/341
	• 90-30 313
	 90-30 351/352/360/363/364
	• 90-70 788
	• 90-30 350/374
	VersaMax CPU001
	VersaMax CPU002
	 VersaMax (CPU005, CPUE05)
	PACSystem RX3i
PLC Network	Multiple controllers can be connected to one HMI device. To set-up multiple connections, select PLC network and click Add to configure each node

Tag Editor Settings

Path: ProjectView> Config > double-click Tags

- 1. To add a tag, click +: a new line is added.
- 2. Select **GE SRTP** from the **Driver** list: tag definition dialog is displayed.

SRTP		
Memory Type Register	Offset SubIndex ▼ 1 0 ▼	
Symbol Name	Data Type Arraysize unsignedShort 0	
Conversion	+/-	
	OK Cancel Apply	Help

Element	Description	Description		
Memory	Memory Type	Description		
Туре	Register	unsigned 16 bit data register (default)		
	Discrete Input	1 bit data input (default)		
Discrete Output		1 bit data output (default)		
	Discrete Global	1 bit data global (default)		
	Internal Coil	1 bit data coil (default)		
	Temporary Coil	1 bit data coil (default)		
	System Status	1 bit data status		
	System Status A	1 bit data status		
	System Status B	1 bit data status		
	System Status C	1 bit data status		
	Analog Input	unsigned 16 bit data input (default)		
	Analog Output	unsigned 16 bit data output (default)		
	SYMBOL	1 bit data symbol (default)		
	Node Override IP	unsigned 8 bit array (see Special Data Types for mode details)		
Offset	This parameter is the address on the physical memory of the controller. The range for any memory type depends on the PLC model.			
SubIndex	This allows resource offset selection within the register.			
Data Type	Available data types:			
	• boolean			
	• byte			
	shortint			
	unsignedByte			
	unsignedShort			
	 unsignedInt 			
	• float			
	 double string			
		epts" section in the main manual.		
	_			
	Note: To define arrays, select one of Data Type format followed by square brackets.			

Element	Description		
Arraysize	 In case of array tag, this property represents the number of array elements. In case of string tag, this property represents the maximum number of bytes available in the string tag. Note: number of bytes corresponds to number of string characters if Encoding property is set to UTF-8 or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2 		
Conversio n	bytes. Conversion to be app Conversion	plied to the tag.	
	inv,swap2	Allowed BCD AB->BA ABCD->CDAB ABCD=FGH->GHEFCDAB Inv bits ABCD=>CDAB Configured Inv bits ABCD->CDAB Cancel OK	
	Depending on data ty	pe selected, the list Allowed shows one or more conversion types.	
	Depending on data ty Value	pe selected, the list Allowed shows one or more conversion types. Description	
	Value	Descriptioninv: Invert all the bits of the tag. $Example:$ $1001 \rightarrow 0110$ (in binary format)	
	Value Inv bits	Descriptioninv: Invert all the bits of the tag. $Example:$ $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)neg: Set the opposite of tag value. $Example:$	
	Value Inv bits Negate	Descriptioninv: Invert all the bits of the tag. $Example:$ $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)neg: Set the opposite of tag value. $Example:$ $25.36 \rightarrow -25.36$ swapnibbles: Swap nibbles in a byte. $Example:$ $15D4 \rightarrow 514D$ (in hexadecimal format)	

Element	Description	
	Value	Description
	GHEFCDAB	<i>Example:</i> 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)
	ABCNOP -> OPMDAB	swap8: Swap bytes in a long word. Example: 142.366 \rightarrow -893553517.588905 (in decimal format) 0 10000000110 0001110010111011010001011101000011100000
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)
		click +. The selected item will be added to list Configured . re configured, they will be applied in order (from top to bottom of list Configured).

Use the arrow buttons to order the configured conversions.

Node Override IP

The protocol provides the special data type Node Override IP which allows you to change the IP address of the target controller at runtime.

This memory type is an array of 4 unsigned bytes, one per each byte of the IP address.

The Node Override IP is initialized with the value of the controller IP specified in the project at programming time.

Node Override IP	PLC operation
0.0.0.0	Communication with the controller is stopped, no request frames are generated anymore.
Different from 0.0.0.0	It is interpreted as node IP override and the target IP address is replaced runtime with the new value.

If the HMI device is connected to a network with more than one controller node, each node has its own Node Override IP variable.



Note: Node Override IP values assigned at runtime are retained through power cycles.

Hostname DNS or mDNS

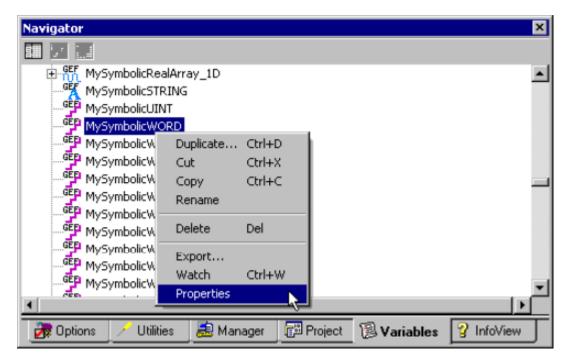
In addition to the array of bytes, string memory type can be selected to be able use the DNS or mDNS hostname as an alternative to the IP Address.

GE SRTP		X
GE SRTP		
Memory Type	Offset SubIndex	
Node Override IP 👻	0 • 0 •	
Symbol Name	Data Type Arraysize unsignedByte []	
Conversion	+/-	
	OK Cancel Apply	Help

Tag Import

For GE PLC model PacSystemRx3i it is possible to create symbolic variables.

To create a new variable, right-click on the **Variables View** and select **New Variable**. To edit an existing variable, rightclick on it and then select **Properties**.



In both cases, the variable's Properties Inspector dialog will appear as shown below.

Inspector ×		
Variable [Target1]		
Name	MySymbolicWORD	
Description		
Publish	External	
Array Dimension 1	0	
Data Source	GE FANUC PLC	
Ref Address		
Data Type	WORD	
Current Value	0	
Initial Value	0	
Default Display Format	Decimal	
Retentive	True	
General		

Important: In order for a symbolic variable to be visible to this driver, **Publish** must be set to **External**. The access must be set to **Read/Write**.

To export these variables from **PACSystem** programming software, right click on **Variable list** (or on selected variables) and click **Export**.

Navigator			×
□-:	New Variable	•	. A
MyMappedWord1	Paste	⊂trl+V	
MyMappedWordArray_1D	Import		
	Export Sort By	,	
MySymbolicBool01	Report	Ctrl+T	
^{GET} MySymbolicBool02 ^{GET} MySymbolicBool03 ^{GET} MySymbolicBool04	Filter By Filter Configuration	•	
MySymbolicBool05	Delete Unused Variables Refresh Variable Reference	s	
MySymbolicBool07 MySymbolicBool08 MySymbolicBool09			
MySymbolicBool10			
MySymbolicBool101			
MySymbolicBool103			
MySymbolicBool104			
🛃 Options 🦯 Utilities 😹 Manage	er 🗊 Project 🔞 Variabl	les 💡 Info	View

In the **Save as Type** drop-down list, select **Comma Separated Variable** (*.csv) as the export file type. The dialogs should appear as shown below.

Export			? ×
Savejn: 险	ExportData	- 🔁 🖻	
-			_
File <u>n</u> ame:	MyExportedVariables		<u>S</u> ave
Save as type:	Comma Separated Values(".csv)		Cancel
			///

Select the driver in the Studio tag editor and click on the "Import tag" button to start the importer.

1:Page1 Protocols* Tags* ×				
+- ~~ & 雪	GE SRTP:prot1	- • • •		
Name	Groups Driver	Address 🛆		
HMIStudio		×		
Multiple tag impor	ters are available for this protocol. Please select the i	importer type and continue.		
Version	Туре			
GeSrtp v1.0	Linear			
Tag Editor exported xml	General			
		OK Cancel		

Select Linear and locate the .csv file, then confirm.

The tags present in the exported document are listed in the tag dictionary from where they can be directly added to the project using the add tags button as shown in the following figure.

B¥	🕅 🔝 🗖 Recursive		₽• Search	P - Search				
Data			Туре		-	Property	Value	
	GE SRTP:prot1 Model: 90-30 331 AI0031 AI0032 AI0033		Container			Model	90-30 331	
					E	Protocol	GE SRTP:prot1	
			short			-		
			short			111		
			short			200		
	- AI00	34	short					
	-C1_I	Imp	boolean					
	-C2_I	Imp	boolean					
	-C3_I	Imp	boolean					
	- I000	04	boolean					
	- M001	102	boolean		-			
	M00.	102	haalaan		*			
Ready					[Dictionary Name:	[GE SRTP prot1] ME_Tag_Export	Protocol Name: GE SRTP



In case of **Online Changes** performed on PLC side, the tag database must be updated manually to correctly **Read** from PLC.

Write operations do not need a database update.

Communication status

Current communication status can be displayed using system variables. See "System Variables" section in the main manual.

Codes supported for this communication driver:

Error	Cause	Action
NAK	The controller replies with a not acknowledge.	-
Timeout	A request is not replied within the specified timeout period.	Check if the controller is connected and properly configured to get network access.
Invalid response	The device did received a response with invalid format or contents from the controller.	Check if the data programmed in the project are consistent with the controller resources.
General Error	Unidentifiable error. Should never be reported.	Contact technical support.

Hitachi SER

HMI devices can be connected to a Hitachi EH/EHV PLC as the network master using this communication driver.

This driver has been designed for serial connection to the programming port of the PLC.

Protocol Editor Settings

Add (+) a driver in the Protocol editor and select the protocol called "Hitachi SER" from the list of available protocols.

The driver configuration dialog box is shown in figure.

Hitachi SER		×
PLC Network	Comm OK	
Alias	Cancel	
PLC Models		_
EH150 CPU104		A
EH150 CPU208		
EH150 CPU316	-	
EH150 CPU516		
EH150 CPU548		
EHV CPU16.32.64		T

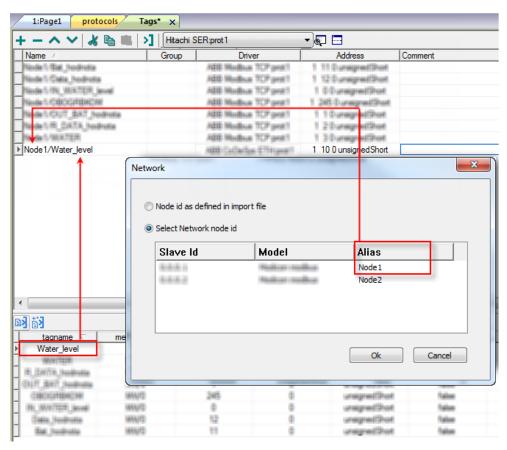
Element	Description
Alias	Name to be used to identify nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node
PLC Models	Select from the list the PLC model you are going to connect to. The selection will influence the data range offset per each data type according to the specific PLC memory resources.
PLC Network	The protocol allows the connection of multiple controllers to one HMI. To set-up multiple connections, check "PLC network" checkbox and create the list of controllers pressing the "Add" button. You must specify the node ID for each device you want to connect.

Element	Description		
	EH150 EH150 EH150	CK Cancel	
Comms.	Opens the serial	port configuration paramete	rs as shown in figure.
		ОК	
	Port	com1 •	
	Baudrate	19200	
	Parity Data bits	even •	
	Stop bits		
	Mode	RS-232	
Port	Serial port selec	tion	
		Series 400	Series 500
	com1	PLC Port	Serial Port
	com2	PC/Printer Port	Option Module
Baud rate, Parity, Data bits, Stop bits	Communication	parameters for serial commu	inication
Mode	Serial port mode	; available options:	
	• RS-232,		
	• RS-485 (2		
	• RS-422 (4	4 wires)	

Tag Name Aliasing in Network Configurations

Tag names must be unique at project level; it often happens that the same tag names are to be used for different controller nodes (for example when the HMI is connected to two devices that are running the same application). Since tags include also the identification of the node and Tag Editor does not support duplicate tag names, the import facility in Tag Editor has an aliasing feature that can automatically add a prefix to imported tags. With this feature tag names can be done unique at project level.

The feature works when importing tags for a specific protocol. Each tag name will be prefixed with the string specified by the "Alias". As shown in the figure below, the connection to a certain controller is assigned the name "Node1". When tags are imported for this node, all tag names will have the prefix "Node1" making each of them unique at the network/project level.





Note: Tag name aliasing is only available when tags can be imported. Tags which are added manually in the Tag Editor do not need to have the Alias prefix in the tag name.

The Alias string is attached to the tag name only at the moment the tags are imported using Tag Editor. If you modify the Alias string after the tag import has been completed, there will be no effect on the names already present in the dictionary. When the Alias string is changed and tags are imported again, all tags will be imported again with the new prefix string.

Tag Import

The Hitachi SER communication driver supports importing tags from the PLC programming software. The tag import filter accepts symbol files with extension ".txt" created by the Actwin-H programming tool.

In the Actwin-H Software, click on the menu "Operations" then "Export symbol list" and then select the project which should be exported as shown in figure.

File Edit View Insert Ope	rations Communication Tools	Help
12 🖻 🔒 🖪 🕼 [Force I/O	99 1 7 + + + E
▶ ■ @@ ≞ ↓ ⊟† 8	Force output	
	Release all forced	
	Move symbol addresses	Program main
🖹 Program main	Exchange addresses	Main
⊕	Export symbol list	Project U_65535_2
Hardware config Hardware config Finds Settings	Edit data memory	Lees de actue de wat

In the Tag Editor select the driver and click on the "Import tag" button to start the importer

52	ProjectView	ά×	1:P	age1 Tags 🗙			
	+ - @ ^ ∨		+ -	~~ & m m	2	Hitachi SER:prot1	
ObjectView	Project3		Name			Group	0
2							
8W	🖶 🚰 Pages			Tag Import			— × —
	🔜 🖬 1 : Page1						
	🖕 🗁 Config						
	Protocols			Select Controller		Hitachi SER	-
	🔛 Tags						
	- Trends			Select Import Type		*.txt	-
	🔔 Alarms			beleet import rype			
	Scheduler				6	ок	Cancel
	NultiLanguage						Curren
	Security						

Locate the ".TXT" file and confirm.

The tags present in the exported document are listed in the tag dictionary from where they can be directly added to the project using the add tags button as shown in figure.

Rā Ko						
tegname	memorytype	arrayindex.subin	index	datatype	array	arraysize
str	MW0	8	0	string-16	true	16
ARRAY_WORD[1]	MW0	0	0	unsignedShort	false	0
ARRAY_WORD[2]	MW0	1	0	unsignedShort	false	0
ARRAY_WORD[3]	MW0	2	0	unsignedShort	false	0
ARRAY_WORD[4]	MW0	3	0	unsignedShort	false	0
MDW2	MD0	2	0	unsignedInt	false	0
MDW3	MD0	3	0	unsignedInt	false	0

Communication Status

The current communication status can be displayed using the dedicated system variables. Please refer to the User Manual for further information about available system variables and their use.

The codes supported for this communication driver are:

Error	Notes
NAK	Returned in case the controller replies with a not acknowledge
Timeout	Returned when a request is not replied within the specified timeout period; ensure the controller is connected and properly configured for communication

Error	Notes
Line Error	Returned when an error on the communication parameter setup is detected (parity, baud rate, data bits, stop bits); ensure the communication parameter settings of the controller is compatible with panel communication setup
Invalid response	The panel did receive from the controller a response, but its format or its contents is not as expected; ensure the data programmed in the project are consistent with the controller resources

Hitachi ETH

This communication driver has been designed to support communication to Hitachi controllers with Ethernet connection. Hitachi controllers must either have an on-board Ethernet port (EHV CPU) or be equipped with an appropriate Ethernet interface (EH-ETH, ET-ETH2 or OB- ETH).

The communication driver supports both TCP/IP and UDP/IP communication protocols.

Protocol Editor Settings

Add (+) a driver in the Protocol editor and select the protocol called "Hitachi ETH" from the list of available protocols.

The driver configuration dialog is shown in figure.

Hitachi ETH		×
PLC Network		ОК
Alias		Cancel
IP address	0.0.0.0	
Port	3004	
enable waiting		
time between request	300	
Connection Type	UDP 👻	
PLC Models		
EH150 CPU316 EH150 CPU516 EH150 CPU548 EHV CPU16.32.64		۰ E
EHV CPU128 EH-W 10.23		-

Element	Description
Alias	Name to be used to identify nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node
IP address	Ethernet IP address of the controller
Port	Port number used for the communication. Default value 3004 and it corresponds to the default setting of Hitachi controllers.
Enable waiting	Introduces a wait time between two communication requests
Time	Wait time between two requests if enable waiting option has been activated

Element	Description
between request	
Connection type	UDP: use communication based on UDP/IP protocol TCP: use communication based on TCP/IP protocol
PLC Models	Select from the list the PLC model you are going to connect to. The selection will influence the data range offset per each data type according to the specific PLC memory resources.
PLC Network	To set-up multiple connections, check "PLC network" checkbox and create the list of controllers pressing the "Add" button. The IP address for each device you want to connect must be specified.

Controller Configuration

The PLC must to be properly configured to support either UPD/IP or TCP/IP communication using port numbers 3004, 3005, 3006 or 3007.

Tag Name Aliasing in Network Configurations

Tag names must be unique at project level; it often happens that the same tag names are to be used for different controller nodes (for example when the HMI is connected to two devices that are running the same application). Since tags include also the identification of the node and Tag Editor does not support duplicate tag names, the import facility in Tag Editor has an aliasing feature that can automatically add a prefix to imported tags. With this feature tag names can be done unique at project level.

The feature works when importing tags for a specific protocol. Each tag name will be prefixed with the string specified by the "Alias". As shown in the figure below, the connection to a certain controller is assigned the name "Node1". When tags are imported for this node, all tag names will have the prefix "Node1" making each of them unique at the network/project level.

トー ^ く ば	🖻 🖷 💙	Hitachi ETH:prot1		- 51 🖿	
Name 🛆		Group	Driver	Address	Comment
Node 1 (End _ hodroite		(HE12) West	disus TCP petril	1 11 Dunkigned/Short	
Node's Onte Androite		(HE12) West	disus TCP peak1	1 12 Dunsigned Short	
Node1/IN W/KTER in	web l	(HEEE) WHEE	disus TCP pret1	1.0.0 unsignat/Short	
Node1/0806780007		(HE12) West	disus TCP prot1	1 245 Dunkgreet/Shot	
Note1/OUT_BAT_host	Profile	(H212) West	disus TCP petril	1 1 E-unsignal/Short	
Node1/R_DX7X_hode	villa	/HERE / West	disus TCP post1	1.2/Dunsignal/Short	
No. de 1/10/871238		(HE12) West	disus TCP post1	1.3/Dunkigned/Short	
Node1/Water_level		AND CAD	artys ETTELaneT	1 10 0 unsignedShort	
	1 🔘	Node id as defined in in	nport file		
	2 (Select Network node id	1	Alias	
		Slave Id	Model	Alias	
	[1	Node1	
taoname 🗁 Water_level	[Slave Id	Model	Node1 Node2	
taoname Water_level		Slave Id	Model	Node1	Cancel
tagname Water_level		Slave Id	Model	Node1 Node2	Cancel
tagname Water_level		Slave Id	Model	Node1 Node2 Ok	
tagname T		Slave Id	Model	Node1 Node2	et faile
tagname Water_level		Slave Id	Model	Node1 Node2	et faile
tagname T		Slave Id	Model	Node1 Node2	et Tala et Tala et Tala



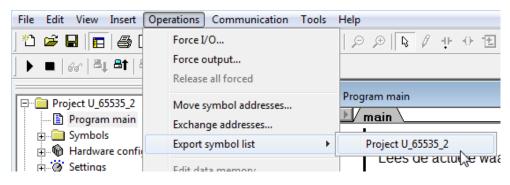
Note: Tag name aliasing is only available when tags can be imported. Tags which are added manually in the Tag Editor do not need to have the Alias prefix in the tag name.

The Alias string is attached to the tag name only at the moment the tags are imported using Tag Editor. If you modify the Alias string after the tag import has been completed, there will be no effect on the names already present in the dictionary. When the Alias string is changed and tags are imported again, all tags will be imported again with the new prefix string.

Tag Import

The Hitachi ETH communication driver supports importing tags from the PLC programming software. The tag import filter accepts symbol files with extension ".txt" created by the Actwin-H programming tool.

In the Actwin-H Software, click on the menu "Operations" then "Export symbol list" and then select the project which should be exported as shown in figure.



In the tag editor select the driver and click on the "Import tag" button to start the importer

ProjectView	4 × 1:Pa	ge1 protocols*	Tags 🗙	
	+ -	へ 🗸 🖥 🖷	Hitachi ETH:prot1	
Project55	Name		Group Driver	,
Project55				
😤 🚊 🚰 Pages				~
🔚 1 : Page1		Tag Import		×
DialogPages				
Page Templates				
Mobile Pages		Select Controller	Hitachi ETH	-
Config				
Protocols		Select Import Typ	ne *.txt	-
Tags		Deleter inport i /	-	
Trends				
Reports			ОК	Cancel
Alarms			UK	Cancer
🚽 Curris				
Scheduler				
Scheduler				

Locate the ".TXT" file and confirm.

The tags present in the exported document are listed in the tag dictionary from where they can be directly added to the project using the add tags button as shown in figure.

Rā R						
tagname	memorytype	arrayindex.subin	index	datatype	array	arraysize
str	MW0	8	0	string-16	true	16
ARRAY_WORD[1]	MW0	0	0	unsignedShort	false	0
ARRAY_WORD[2]	MW0	1	0	unsignedShort	false	0
ARRAY_WORD[3]	MW0	2	0	unsignedShort	false	0
ARRAY_WORD[4]	MW0	3	0	unsignedShort	false	0
MDW2	MD0	2	0	unsignedInt	false	0
MDW3	MD0	3	0	unsignedInt	false	0

Communication Status

The current communication status can be displayed using the dedicated system variables. Please refer to the User Manual for further information about available system variables and their use.

Error	Notes
NAK	Returned in case the controller replies with a not acknowledge
Timeout	Returned when a request is not replied within the specified timeout period; ensure the controller is connected and properly configured for communication
Invalid response	The panel did receive from the controller a response, but its format or its contents is not as expected; ensure the data programmed in the project are consistent with the controller resources
General Error	Error cannot be identified; should never be reported; contact technical support

The codes supported for this communication driver are:

IDEC Maintenance

IDEC Maintenance communication driver has been designed to connect HMI devices to IDEC PLC through Serial or Ethernet connection.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In Config node double-click Protocols.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the **PLC** list.

The protocol configuration dialog is displayed.

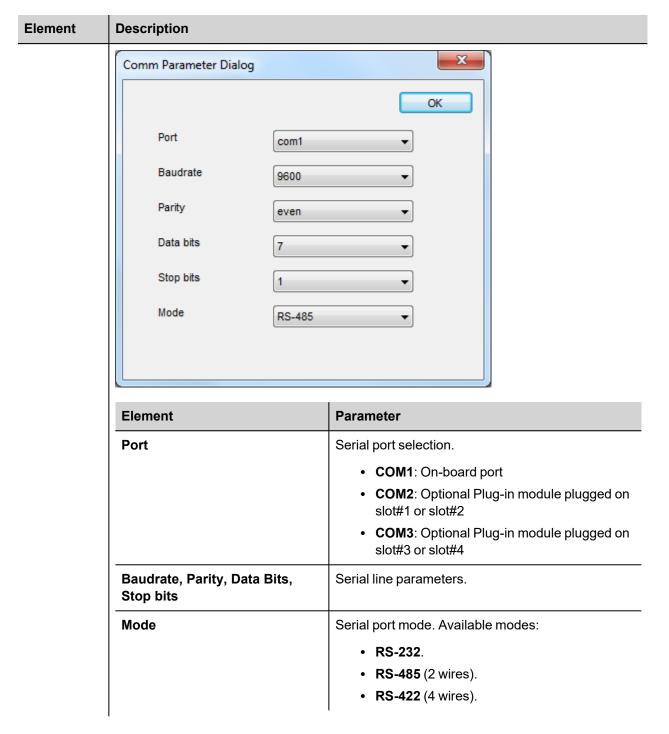
IDEC Maintenance	×
PLC Network	Comm OK
Alias	Cancel
Node ID	1
Media	Serial 🔹
IP address	0.0.0.0
Port	2101
Timeout(ms)	500
PLC Models	
FC6A - MicroSmart P FC6A - MicroSmart	rius
FC5A - MicroSmart F	Pentra
FC4A - MicroSmart FT1A - SmartAxis	
FT IA - Smar (AXIS	

Element	Description
Alias	Name identifying nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node.
Node ID	Serial node associated to PLC.
Media	Allows the selection of transport Media.

Element	Description		
	select Serial to connect via serial line		
	select Ethernet to connect via TCP/IP		
IP address	IP address of PLC (only available if Ethernet media is selected)		
Port	Port number of PLC		
Timeout (ms)	Time delay in milliseconds between retries in case of missing response		
PLC Models	PLC model available:		
	FC6A - MicroSmart Plus		
	FC6A - MicroSmart		
	FC5A - MicroSmart Pentra		
	FC4A - MicroSmart		
	FT1A - SmartAxis		
PLC Networ k	Enable configuration of multiple connections.		

Element	Description		
	IDEC Maintenance		
	PLC Network	Comm OK	
	Alias	Cancel	
	Node ID	1	
	Media	thernet	
	IP address	IDEC Maintenance	ر ۱
	Port 2	ОК	
	Timeout(ms)	Alias	
	PLC Models FC6A - MicroSmart Plus	Node ID 1	
	FC6A - MicroSmart FC5A - MicroSmart Pentra FC4A - MicroSmart	IP address 0 . 0 . 1	
	FT1A - SmartAxis	Port 2101	
	Slaves	PLC Models FC6A - MicroSmart Plus	
	Slave Id M	FC6A - MicroSmart FC5A - MicroSmart Pentra FC4A - MicroSmart FC4A - MicroSmart FT1A - SmartAxis	
Comm	If clicked displays the commu	unication parameters setup dialog (only available if Serial	

media is selected)



Tag Editor Settings

In Tag Editor select IDEC Maintenance protocol.

Add a tag using [+] button. Tag setting can be defined using the following dialog:

IDEC Maintenance				X
IDEC Maintenance				
Memory Type	Offset	SubIndex		
I -Input 🔻	0	• 0		
Data Type	Arraysize	Conversion		
unsignedByte 👻	0		+/-	
	0	K Cancel	Apply	Help

Element	Description				
Memory	Memory Type	Description			
Туре	I - Input	I resources. Corresponding to internal	digital Input point.		
	Q - Output	Q resources. Corresponding to internal digital Output point.			
	M - Internal Relay	M resources. Corresponding to PLC in	ternal memory.		
	R - Shift Register	S resources. Corresponding to PLC sh	ift registers.		
	T - Timer	T resources. Corresponding to PLC tin	iers.		
	TC - Timer Current Value	TC resources. Corresponding to PLC t values.	imer current		
	TP - Timer Preset Value	TP resources. Corresponding to PLC timer preset values.			
	C - Counter	C resources. Corresponding to PLC co	unters.		
	CC - Counter Current Value	CC resources. Corresponding to PLC counter current values.			
	CP - Counter Preset Value	CP resources. Corresponding to PLC counter preset values.			
	D - Data register	D resources. Corresponding to PLC data registers.			
Offset	Starting address for the Tag. T	he possible range depend on PLC mode	I selected.		
Subindex	This allows resource offset sel	ection depending on the selected data ty	pe.		
Data Type	Data Type	Memory Space	Limits		
	boolean	1-bit data	01		
	byte	8-bit data	-128 127		
	short	16-bit data	-32768 32767		
	int	32-bit data	-2.1e9 2.1e9		
	int64	64-bit data	-9.2e18 9.2e18		
	unsignedByte	8-bit data	0 255		
	unsignedShort	16-bit data	0 65535		
	unsignedInt	32-bit data	04.2e9		

Element	Description		
	Data Type	Memory Space	Limits
	uint64	64-bit data	0 1.8e19
	float	IEEE single-precision 32-bit floating point type	1.17e-38 3.4e38
	double	IEEE double-precision 64-bit floating point type	2.2e-308 1.79e308
	string	Array of elements containing character selected encoding	code defined by
	binary	Arbitrary binary data	
	Note: to define array brackets like "byte[]"	s. select one of Data Type format followe , "short[]"…	ed by square
Arraysize	 In case of array tag, this property represents the number of array elements. In case of string tag, this property represents the maximum number of bytes available in the string tag. Note: number of bytes corresponds to number of string characters if Encoding property is set to UTF-8 or Latin1 in Tag Editor. 		
	character requires 2 bytes.	CS-2BE, UCS-2LE, UTF-16BE or UTF-1	
Conversion	Conversion to be applied to the	tag.	
	Conversion		
		A >CDAB FGH->GHEFCDAB	
	Depending on data type selected	ed, the list Allowed shows one or more o	conversion types.

Element	Description	
	Value	Description
	Inv bits	inv : Invert all the bits of the tag. <i>Example:</i> $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)
	Negate	neg : Set the opposite of tag value. <i>Example:</i> $25.36 \rightarrow -25.36$
	AB -> BA	swapnibbles : Swap nibbles in a byte. <i>Example:</i> $15D4 \rightarrow 514D$ (in hexadecimal format) $5588 \rightarrow 20813$ (in decimal format)
	ABCD -> CDAB	swap2 : Swap bytes in a word. <i>Example:</i> $9ACC \rightarrow CC9A$ (in hexadecimal format) $39628 \rightarrow 52378$ (in decimal format)
	ABCDEFGH	swap4: Swap bytes in a double word.
	-> GHEFCDAB	<i>Example:</i> 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)
	ABCNOP - > OPMDAB	swap8: Swap bytes in a long word. Example: 142.366 \rightarrow -893553517.588905 (in decimal format) 0 1000000110 0001110010111011010001011010000111001010
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)

Element	Description
	Select conversion and click +. The selected item will be added to list Configured .
	If more conversions are configured, they will be applied in order (from top to bottom of list Configured).
	Use the arrow buttons to order the configured conversions.

Tag Import

Select the driver in Tag Editor and click on the Import Tags button to start the importer.

	lags	×								
+	-	z	đ	ß	>]	₽	A 9B	B>	63	1
Data			^		-	Та	g URI			

The following dialog shows which importer type can be selected.

HMIStudio					
Multiple tag in	Multiple tag importers are available for this protocol. Please select the importer type and continue.				
Version	Туре				
WindLDR v1.0	Linear				
Tag Editor exported	xml General				
	OK Cancel				
	OK Cancer				

Туре	Description			
WindLDR v1.0	Requires a .csv file.			
Linear	All variables will be displayed at the same level.			
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.			
	Tags X			
	+ - 🎽 🕲 🔊 🚺 🕩 🎭 🖬 👬			
	Data Tag URI			

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

I NBOILD					
👻 🔟 🖸 🚺 🖸	\$B ₪} ∰	R 🔎 - Search YFilter by: Data	▼ Ite	ms used:6/10000 Protocol: Show all	Show all tags 💮 🗖
Data Ty	ype	Comment	^	Property	Value
Modbus TCP:prot1	ontainer			Y Driver	
Model: Modicon Modbus(1-based)				Model	Modicon Modbus(1-based)
	nsignedShort			Protocol	Modbus TCP:prot1
	nsignedShort			✓ Dictionary	
- Holding Registers 3 un	nsignedShort				
- MRTU1 un	nsignedShort			Array	false
MRTU2 un	nsignedShort			Array size	0
MRTU3 un	nsignedShort			Arrayindex.Subindex	400003
MRTU4 un	nsignedShort			Comment	
MRTU5 un	nsignedShort			Data type	unsignedShort

Toolbar item	Description
ka	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
國	Update Tag(s) . Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag. Example of both checked and unchecked result:
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Jetter Ext ETH

The Jetter Ext ETH driver has been developed to communicate with Jetter devices using the PCOM7 protocol.

The HMI protocol identifies Jetter devices using their IP addresses. You should take note of these addresses as you assign them because you will need them later in the set-up phase of the user interface application.

Different physical media, gateways, routers and hubs can be used in the communication network. Also, other devices can independently make simultaneous use of the network. However, it is important to ensure that the traffic generated by these devices does not degrade the communication speed (round-trip time) to an unacceptable level. Too slow communication between the device and the Jetter device may result in low display update rate.

Protocol Editor Settings

Add (+) a new driver in the Protocol editor and select the protocol called "Jetter Ext ETH" from the list of available protocols.

The driver configuration dialog is shown in the following figure.

Jetter Ext ETH		×
PLC Network		ОК
Alias		Cancel
IP Address	0.0.0.0	
Local Port	50001	
PLC Models		
JetControl 3xx pcom7		

Element	Description
Alias	Name identifying nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node.
IP address	Ethernet IP address of the PLC.
Local Port	Allows to specify the source Port used from the HMI to communicate with PLC.

Element	Description			
PLC Models	An unique PLC model is available: JetControl 3xx pcom7.			
PLC Network	The protocol allows the connection of multiple controllers to one HMI device. To set-up multiple connections, check "PLC network" checkbox and enter IP Address for all PLCs.			

Tag Editor Settings

Into Tag editor select the protocol "Jetter Ext ETH" from the list of defined protocols and add a tag using [+] button.

Tag settings can be defined using the following dialog:

Jetter Ext ETH		X
Jetter Ext ETH		
Memory Type	Offset Subindex	
Input -	0 🔹 0 👻	
Туре	Arraysize Conversion	
boolean -	0 +/-	
	OK Cancel Apply	Help

Element	Description				
Memory Type	Area of PLC where ta	ag is located.			
Offset	Offset address where	e tag is located.			
SubInde x	This allows resource	offset selection within the re	egister.		
Туре	Data Type	Data Type Memory Space Limits			
	boolean	1 bit data	01		
	byte	8-bit data	-128 127		
	short	16-bit data	-32768 32767		
	int	32-bit data	-2.1e9 2.1e9		
	unsignedByte	8-bit data	0 255		

Element	Description				
	Data Type	Memory Space	Limits		
	unsignedShort	16-bit data	0 65535		
	unsignedInt	32-bit data	0 4.2e9		
	float	IEEE single-precision	1.17e-38 3.40e38		
		32-bit floating point type			
	string	Refer to "String data type chapt	er"		
	1 ike "byte[]", "	e arrays, select one of Data Type format followed by square brackets short[]"…			
Arraysiz e	 In case of array tag, this property represents the number of array elements. In case of string tag, this property represents the maximum number of bytes available in the string tag. Note: number of bytes corresponds to number of string characters if Encoding property is set to UTF-8 or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2 bytes. 				
Conversi	Conversion to be applie	d to the tag.			
on	Conversion				
	inv,swap2	BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits	nfigured v bits BCD->CDAB		
	Depending on data type selected, the Allowed list shows one or more converse below.				

Element	Description		
	Value	Description	
	Inv bits	Invert all the bits of the tag.	
		Example: $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)	
	Negate	Set the opposite of the tag value.	
		<i>Example:</i> 25.36 → -25.36	
	AB -> BA	Swap nibbles of a byte.	
		<i>Example:</i> 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)	
	ABCD -> CDAB	Swap bytes of a word.	
		Example: 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)	
	ABCDEFGH -> GHEFCDAB	Swap bytes of a double word. <i>Example:</i> $32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format) $855441236 \rightarrow 1426062386$ (in decimal format)	
	ABCNOP -> OPMDAB	Swap bytes of a long word. Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) 0.1000000110 0001110010111011001000101101000011100000	
	BCD	Separate the byte in two nibbles, and reads them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)	
	New Format	Jetter "string" data format	

Element	Description	
	Select the conversion and click on plus button. The selected item will be added on Configured list.	
	If more conversions are configured, they will be applied in order (from top to bottom of Configured list).	
	Use the arrow buttons to order the configured conversions.	

Special data types

The Jetter Ext ETH driver provides one special data type called "Node Override IP".

The Node override IP allows changing at runtime the IP address of the controller. This memory type is an array of 4 unsigned bytes, one per each byte of the IP address.

The Node Override IP is initialized with the value of the controller IP specified in the project at programming time.

If the Node Override IP is set to 0.0.0.0, all the communication with the slave is stopped, no request frames are generated anymore.

If the Node Override IP has a value different from 0.0.0.0, it is interpreted as node IP override and the controller IP address is replaced runtime with the new value.

In case the device has been configured to access to a network of controllers, each node has its own Node Override IP variable.



Note: the Node Override IP values assigned at runtime are retained through power cycles

Jetter Ext ETH				x
Network Jetter Ext ETH				
Mamanu Tuna	Offset	Subindex		
Memory Type	0			
Туре	Arraysize	Conversion		
unsignedByte [] 👻	8		+/-	
	0	K Cancel	Apply	Help

String data type

The Jetter devices allow to define within the programming software two different type of string variables: "Regstring" is the old format while "string" is the new format, both these formats are supported by the Jetter Ext ETH driver.

When "Regstring" format is used the corresponding Tag must be configured simply selecting string as data type as shown in the following figure, no further steps are required.

Jetter Ext ETH				×
Jetter Ext ETH				
Memory Type Register 👻	Offset	Subindex		
Type	Arraysize	Conversion	+/-	
		K Cancel	Apply	Help

When "string" format is used once selected the string data type in the Tag definition dialog it is necessary, as shown in the following figure, to add a New Format conversion.

Jetter Ext ETH			22	
Jetter Ext ETH				
Memory Type Register +	Offset	Subindex		
Туре	Arraysize	Conversion		
string 👻	8	newForm	Allowed AB->BA	Configured New Format
			ABCD->CDAB New Format	+
				Cancel OK
	ОК	Cancel	Apply Help	

Communication Status

The current communication status can be displayed using the dedicated system variables. Please refer to the User Manual for further information about available system variables and their use.

The codes supported for this communication driver are:

Error	Notes
No response	Returned when a request is not replied within the specified timeout period; ensure the controller is connected and properly configured to get network access
Incorrect node address in response	The device did receive from the controller a response with invalid node address
The received message too short	The device did receive from the controller a response with invalid format
Incorrect writing data acknowledge	Controller did not accept write request; ensure the data programmed in the project are consistent with the controller resources

Keyence KV

Keyence KV communication driver has been designed to connect HMI devices to KEYENCE PLCs through Serial or Ethernet connection.

Please note that changes in the communication protocol specifications or PLC hardware may have occurred since this documentation was created. Some changes may eventually affect the functionality of this communication driver. Always test and verify the functionality of your application. To fully support changes in PLC hardware and communication protocols, communication drivers are continuously updated. Always ensure that the latest version of communication driver is used in your application.

Setting-up the PLC for Communication

Keyence KV PLC's do not require any particular setup-up for communication at the programming port.

Protocol Editor Settings

Add (+) a driver in the Protocol Editor and select the protocol called "Keyence KV" from the list of available protocols.

The driver configuration dialog is shown in figure.

Comm OK
Cancel
Ethernet 👻
0.0.0.0
8501
600
2
A
E
-

Element	Description
Alias	Name identifying nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node.
Media	Allows the selection of transport Media.

Element	Description
	 select Serial to connect via serial line select Ethernet to connect via TCP/IP
IP address	IP Address of the controller. Only available for Ethernet Media.
Port	Port number used by PLC. The default value is 8501. Only available for Ethernet Media.
Timeout (ms)	Time delay in milliseconds between two retries in case of missing response from PLC.
Number of retries	Number of times a communication session is repeated before declaring reporting communication error.

Element	Description			
PLC Models	The list allows selecting the PLC model. The selection will influence the data range offset per each data type according to the specific PLC memory resources.			
	Available models:			
	• KV10/16			
	• KV24			
	• KV40/80			
	• KV300			
	KV700KV1000			
	 KV3000/5000/5500 			
	• KV7300/7500			
	• KV8000			
Comm	Opens the serial port configuration dialog box. Only ava	ailable for Serial Media.		
	Comm Parameter Dialog			
	ОК			
	Port com1 -			
	Baudrate 9600			
	Barity			
	Data bits 8			
	Stop bits			
	Element	Parameter		
	Port	Serial port selection.		
		COM1: On-board port		
		 COM2: Optional Plug-in module plugged on slot#1 or slot#2 		
		COM3: Optional Plug-in		
		module plugged on slot#3 or slot#4		
	Baudrate, Parity, Data Bits, Stop bits	Serial line parameters.		

Tag Editor Settings

Path: ProjectView> Config > double-click Tags

- 1. To add a tag, click +: a new line is added.
- 2. Select **ROBOX BCC/31** from the **Driver** list: tag definition dialog is displayed.

ROBOX BCC/31		×
ROBOX BCC/31		
Memory Type	Offset SubIndex	
Logic Input Bit 👻		
Axis Index	Data Type Arraysize boolean 0	
Conversion	+/-	
	OK Cancel Apply	Help

Element	Description			
Memory Type	Resource where tag is located on PLC.			
	Available resources are:			
	 Available resources are: Logic Input Bit Logic Output Word Logic Output Word Phis Input Bit Phys Input Word Phys Output Bit Phys Output Bit Phys Output Word Non Volatile I32 Non Volatile Double Non Volatile string Volatile I32 Volatile Double Volatile string Parameter I32 Parameter I32 Axis Parameter Double 			
	Alarm MaskAlarm Code			
	Alarm Code Alarm string			
Offset	Offset address where tag is loc Offset addresses are six digits	cated. composed by one digit data type prefix +	five digits resource address.	
SubIndex	This allows resource offset sel	ection within the selected memory type.		
Axis Index	Allows to select Axis index. Av	ailable only for Axis memory types.		
Data Type	Data Type	Memory Space	Limits	
	boolean	1-bit data	01	
	byte	8-bit data	-128 127	
	short	16-bit data	-32768 32767	
	int	32-bit data	-2.1e9 2.1e9	

Element	Description					
	Data Type	Memory Space	Limits			
	int64	64-bit data	-9.2e18 9.2e18			
	unsignedByte	8-bit data	0 255			
	unsignedShort	16-bit data	0 65535			
	unsignedInt	32-bit data	04.2e9			
	uint64	64-bit data	01.8e19			
	float	IEEE single-precision 32-bit floating point type	1.17e-38 3.4e38			
	double	IEEE double-precision 64-bit floating point type	2.2e-308 1.79e308			
	string	Array of elements containing character code defined by selected encoding				
	binary	Arbitrary binary data				
Arraysize	 Note: to define arrays. select one of Data Type format followed by s "byte[]", "short[]" In case of array tag, this property represents the number of array elements In case of string tag, this property represents the maximum number of array elements 					
	string tag. Note: number of bytes corresponds to number of string characters if Encod UTF-8 or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE 2 bytes.					
Conversion	Conversion to be applied to the tag.					
	Conversion					
	BCD AB->E ABCD ABCD	Allowed BCD AB->BA ABCD->CDAB ABCD=>CDAB ABCD=>GHEFCDAB Inv bits Cancel OK				
	Depending on data type select	ed, the list Allowed shows one or more	conversion types.			

Description		
Value	Description	
Inv bits	inv: Invert all the bits of the tag.	
	Example: 1001 \rightarrow 0110 (in binary format) 9 \rightarrow 6 (in decimal format)	
Negate	neg: Set the opposite of tag value.	
	<i>Example:</i> 25.36 → -25.36	
AB -> BA	swapnibbles: Swap nibbles in a byte.	
	Example: 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)	
ABCD -> CDAB	swap2: Swap bytes in a word.	
	Example: 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)	
ABCDEFGH ->	swap4: Swap bytes in a double word.	
GHEFCDAB	Example: 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)	
ABCNOP ->	swap8: Swap bytes in a long word.	
OPMDAB	Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) 0.10000000110 0001110010111011001000101101000011100101	
	\rightarrow 1 10000011100 101010000101000101101101100101101	
BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9)	
	Example: $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)	

Element	Description			
	If more conversions are configured, they will be applied in order (from top to bottom of list Configured).			
	Use the arrow buttons to order the configured conversions.			

Tag Import

Select the driver in Tag Editor and click on the **Import Tags** button to start the importer.

	Tags	×								
+	-	z	ß	ß	>]	Þ	A 9B	B>	š 3	1
Data	A Tag URI									

Locate the .xml file exported from Tag Editor and click Open.

importer - Select a file to open						
Com Com	puter 🕨 Local Disk (C:) 🕨 Folder1 🔍 🚽	Search Folder1				
Organize 🔻 New	Organize 🔻 New folder 🗄 🐨 🗍 🔞					
	Name	Date modified Type				
Desktop	Exported Tags.xml	03/06/2016 16:42 XML File				
Documents Documents Music Fictures Videos Homegroup User User Computer Network Control Panel	E					
Control Panel		+				
F	le name: Exported Tags.xml 🔹	Tag Editor exported file (*.xml) ▼ Open Cancel				

Tags included in the symbol file are listed in the tag dictionary. The tag dictionary is displayed at the bottom of the screen.

Tags × Protocols					
+ - 🎽 🕲 刘	[> \$a ₪ 🕅	R 🔎 - Search 🍸 Filter by: Data	▼ Ite	ems used:6/10000 Protoc	col: Show all 🛛 🕑 Show all tags 🔅 🗔
Data ^	Туре	Comment	^	Property	Value
Modbus TCP:prot1	. Container			✓ Driver	
Model: Modicon Modbus(1-base	a)			Model	Modicon Modbus(1-based)
- Holding Registers 1	unsignedShort			Protocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			✓ Dictionary	·
Holding Registers 3	unsignedShort			Array	false
- MRTU1 - MRTU2	unsignedShort unsignedShort			Array size	0
MRTU3	unsignedShort			Arrayindex.Sub	-
- MRTU4	unsignedShort			Comment	
- MRTU5	unsignedShort			Data type	unsignedShort
	- 1=1 · ·			Dota type	anagricaonore

Toolbar item	Description
R	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
K a Ka	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.
	Example of both checked and unchecked result:
	Tags* x + - X > 1 > 2 > 2 > 3
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Communication Status

The current communication status can be displayed using the dedicated system variables. Please refer to the User Manual for further information about available system variables and their use.

The codes supported for this communication driver are:

Error	Description
Timeout	Returned when a request is not replied within the specified timeout period; ensure the controller is connected and properly configured to get network access
Timeout receiving response characters	Returned when a request is not replied within the specified timeout period between chars in frame, should never be reported; contact technical support
Line Error	Returned when an error on the communication parameter setup is detected (parity, baud rate, data bits, stop bits); ensure the communication parameter settings of the controller is compatible with panel communication setup
Invalid response	The panel did receive from the controller a response, but its format or its contents is not as expected; ensure the data programmed in the project are consistent with the controller resources

Koyo DL

The Koyo DL driver has been developed for the communication with Koyo DL series controllers trough serial connection.

Protocol Editor Settings

Add (+) a new driver in the Protocol editor and select the protocol called "Koyo DL" from the list of available protocols.

The driver configuration dialog is shown in the following figure:

Koyo DL			x
PLC Network		Comm OK	
Node ID	1	Cancel	
PLC Models			
DL440 Net Port		A	
DL440 Prg.Port DL105, DL230			
DL240 Net.Port			
DL240 Prg.Port DL340		-	

Element	Description
Node ID	Controller Node ID
PLC Models	The driver supports communication with different DL controllers. Please check directly in the programming IDE software for a complete list of supported controllers.
PLC Network	The protocol allows the connection of multiple controllers to one operator panel. To set-up multiple connections, check "PLC network" checkbox and configure all controllers.

Element	Description		
Element	Description Koyo DL VPLC Network Comm OK Node D -1 Cancel PLC Models DL440 Prg.Port DL240 Net Port DL240 Net.Port DL240 Net.Port DL240 Net.Port DL340 Slaves Add Delete Modify Slave Id Koyo DL OK OK		
Comm	PLC Models DL440 Prg.Port DL440 Prg.Port DL240 Net.Port DL240 Prg.Port DL340 Gives access to the serial port configuration parameters as shown in the figure below. Comm Parameter Dialog		
	OK Port com1 Baudrate 9600 Parity even Data bits 8 Stop bits 1 Mode RS-485		
Port	Serial port selection		

Element	Description	
Baud rate, Parity, Data bits, Stop bits	Communication parameters for serial communication	
Mode	Serial port mode; available options:	
	• RS-232,	
	• RS-485 (2 wires)	
	• RS-422 (4 wires)	

Into Tag editor select the protocol "Koyo DL" from the list of defined protocols and add a tag using [+] button.

Tag settings can be defined using the following dialog:

Koyo DL		×
Koyo DL		
Memory Type	Offset SubIndex	
Timer (Acc.) 👻		
Data Type	Arraysize Conversion	
short -	0 +	6
	OK Cancel A	pply Help

Element	Description
Memory Type	Memory resource where tag is located.
Offset	Offset address where tag is located.
SubIndex	This allows resource offset selection within the register.

Element	Description				
Data Type	Data Type	Memory Space	Limits		
	boolean	1 bit data	01		
	byte	8-bit data	-128 127		
	short	16-bit data	-32768 32767		
	int	32-bit data	-2.1e9 2.1e9		
	unsignedByte	8-bit data	0 255		
	unsignedShort	16-bit data	0 65535		
	unsignedInt	32-bit data	04.2e9		
	float	IEEE single-precision	1.17e-38 3.40e38		
		32-bit floating point type			
	double	IEEE double-precision 64-bit floating point type	2.2e-308 1.79e308		
	string Array of elements containing character code defined by selected encoding.				
	binary Arbitrary binary data				
	Note: to define arrays, select one of Data Type format followed by square brackets like "byte[]", "short[]"				
Arraysize	 In case of array tag, this property represents the number of array elements. In case of string tag, this property represents the maximum number of bytes available in the string tag. 				
Note: number of bytes corresponds to number of string characters set to UTF-8 or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or U character requires 2 bytes.				ty is	
Conversion	Conversion to be applied to the tag.				

ent	Description			
	Conversion			
	inv,swap2	Allowed BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits Cancel OK		
	Depending on c	lata type selected, the list Allowed shows one or more conversion types		
	Value	Description		
	Inv bits	inv : Invert all the bits of the tag. <i>Example:</i> $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)		
	Negate	neg : Set the opposite of tag value.		
		<i>Example:</i> 25.36 → -25.36		
	AB -> BA	swapnibbles : Swap nibbles in a byte. <i>Example:</i> $15D4 \rightarrow 514D$ (in hexadecimal format) $5588 \rightarrow 20813$ (in decimal format)		
	ABCD -> CDAB	swap2 : Swap bytes in a word. <i>Example:</i> $9ACC \rightarrow CC9A$ (in hexadecimal format) $39628 \rightarrow 52378$ (in decimal format)		
	ABCDEFGH -> GHEFCDAB	swap4 : Swap bytes in a double word. <i>Example:</i> $32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format) $855441236 \rightarrow 1426062386$ (in decimal format)		
	ABCNOP - > OPMDAB	swap8 : Swap bytes in a long word. Example: 142.366 → -893553517.588905 (in decimal format) 0 10000000110 0001110010111011001000101101		

Element	Description		
	Value	Description	
		0001 → 1 10000011100 1010100001010001011011011001011011000010011 1101 (in binary format)	
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9)	
		Example: $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)	
	Select conversi	on and click +. The selected item will be added to list Configured .	
	If more convers Configured).	ions are configured, they will be applied in order (from top to bottom of list	
	Use the arrow b	puttons to order the configured conversions.	

Koyo DL ETH

The Koyo DL ETH driver has been developed for the connection of Koyo DL series controllers trough Ethernet.

Protocol Editor Settings

Add (+) a new driver in the Protocol editor and select the protocol called "Koyo DL ETH" from the list of available protocols.

The driver configuration dialog is shown in the following figure:

Koyo DL ETH		×
PLC Network		ОК
Address Type	IP •	Cancel
IP Address	0.0.0.0	
Device ID	1	
Device Name		
UDP Port	28784	
🔽 use new EBC Mode	:	
Timeout (ms)	100	
Retry count	3	
PLC Models ECOM		
EBC		

Element	Description
Address Type	Allow to select which address type to use
IP Address	When Address Type is "IP", define the controller IP Address

Element	Description		
Device ID	When Address Type is "ID", define the controller Device ID		
Device Name	When Address Type is "Name", define the controller name		
UDP Port	UDP port of controller		
use new EBC Mode	If PLC Model is "EBC" allow to use the new EBC Mode		
Timeout (ms)	Defines the time inserted by the protocol between two retries of the same message in case of missing response from the server device. Value is expressed in milliseconds.		
Retry count	Defines the number of times a certain message will be sent to the controller before reporting the communication error status. A value of 1 for this parameter means the HMI will eventually report the communication error		
	status if the response to the first request packet is not correct.		
PLC Models	The driver supports communication with different DL controllers. Please check directly in the programming IDE software for a complete list of supported controllers.		
PLC Network	The protocol allows the connection of multiple controllers to one operator panel. To set-up multiple connections, check "PLC network" checkbox and configure all controllers.		

Into Tag editor select the protocol "Koyo DL" from the list of defined protocols and add a tag using [+] button.

Tag settings can be defined using the following dialog:

yo DL ETH Koyo DL ETH			×
Memory Type	Offset	SubIndex	
Data Type boolean 💌	Arraysize	Conversion	+/-
		OK Cancel	Apply Help

Element	Description			
Memory Type	Memory resource	Memory resource where tag is located.		
Offset	Offset address wh	Offset address where tag is located.		
SubIndex	This allows resour	This allows resource offset selection within the register.		
Data Type	Data Type	Limits		
	boolean	1 bit data	01	
	byte	8-bit data	-128 127	
	short	16-bit data	-32768 32767	
	int	32-bit data	-2.1e9 2.1e9	
	unsignedByte	8-bit data	0 255	
	unsignedShort	16-bit data	0 65535	
	unsignedInt	32-bit data	0 4.2e9	
	float	IEEE single-precision	1.17e-38 3.40e38	

Element	Description			
	Data Type	Memory Space	Limits	
		32-bit floating point type		
	double	IEEE double-precision 64-bit floating point type	2.2e-308 1.79e308	
string Array of elements conta selected encoding.		Array of elements containing cha selected encoding.	aracter code defined by	
	binary	Arbitrary binary data		
		TE: to define arrays, select one of Data ckets like "byte[]", "short[]"…	Type format followed by square	
Arraysize	 In case availab Note: number set to UTF-8 c 	of array tag, this property represents the number of array elements. If string tag, this property represents the maximum number of bytes in the string tag. If bytes corresponds to number of string characters if Encoding property is Latin1 in Tag Editor. Derty is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one		
Conversion		be applied to the tag.		
Conversion	Conversion			
	inv,swap2	Allowed	Configured	
		BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits	Inv bits ABCD->CDAB	
		✓	Cancel OK	
	Depending on	a data type selected, the list Allowed sh		
	Depending on Value	a data type selected, the list Allowed si		
	Value	Description		

Element	Description	
	Value	Description
		<i>Example:</i> 25.36 → -25.36
	AB -> BA	swapnibbles: Swap nibbles in a byte.
		<i>Example:</i> $15D4 \rightarrow 514D$ (in hexadecimal format) $5588 \rightarrow 20813$ (in decimal format)
	ABCD -> CDAB	swap2: Swap bytes in a word.
	CDAB	<i>Example:</i> 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)
	ABCDEFGH	swap4: Swap bytes in a double word.
	-> GHEFCDAB	<i>Example:</i> 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)
	ABCNOP -	swap8: Swap bytes in a long word.
	> OPMDAB	Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) $0\ 10000000110$ 0001110010111011010001011010000111001010
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9)
		Example: $23 \rightarrow 17$ (in decimal format) $0001 \ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)
	Select conversi	on and click +. The selected item will be added to list Configured .
	If more convers Configured).	ions are configured, they will be applied in order (from top to bottom of list
	Use the arrow b	uttons to order the configured conversions.

Use the arrow buttons to order the configured conversions.

Modbus RTU

The operator panels can be connected to a Modbus network as the network master using this communication driver.

Implementation details

The Modbus RTU implementation supports only a subset of the Modbus standard RTU function codes.

Code	Function	Description
01	Read Coil Status	Reads multiple bits in the device Coil area
02	Read Input Status	Read the ON/OFF status of the discrete inputs (1x reference) in the slave
03	Read Holding Registers	Read multiple Registers
04	Read Input Registers	Reads the binary contents of input registers (3x reference) in the slave
05	Force Single Coil	Forces a single Coil to either ON or OFF
06	Preset Single Register	Presets a value in a Register
16	Preset Multiple Registers	Presets value in multiple Registers



Note: Communication speed with controllers is supported up to 115200 baud.



Note: Floating point data format is IEEE standard compliant.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the PLC list.

The protocol configuration dialog is displayed.

Modbus RTU	×
PLC Network	Comm OK
Alias	Cancel
Node ID	1
Timeout (ms)	2000
Delay (ms)	0
Num of repeats	2
Max read block	250
Max read bit block	2000
Write Holding Register	1 6 •
Write Coils	15 🔹
Transmission Mode	RTU 🔻
PLC Models	
Modicon Modbus(1-bas Generic Modbus(0-base Enron Modbus(1-based Enron Modbus(0-based	ed) I) with 32bit registers

Element	Description
Alias	Name identifying nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node.
Node ID	Modbus node of the slave device.
Timeout (ms)	Time delay in milliseconds between two retries in case of missing response from the server device.
Delay (ms)	Time delay in milliseconds between the end of the last received frame and the starting of a new request. If set to 0, the new request will be issued as soon as the internal system is able to reschedule it.
Num of repeats	Number of times a certain message will be sent to the controller before reporting the communication error status.
_	When set to 1 the panel will report the communication error if the response to the first request packet is not correct.

Element	Description		
Max read block	Maximum length in bytes of a data block request. It applies only to read access of Holding Registers.		
Max read bit block	Maximum length in bits of a block request. It applies only to read access of Input Bits and Output Coils.		
Write Holding Register	Modbus function for write operations to Holding Registers. Select between the function 06 (preset single register) and function 16 (preset multiple registers).		
	If function 06 is selected, the protocol will always use function 06 for writing to the controller, even when writing to multiple consecutive registers.		
	If function 16 is selected, the protocol will always use function 16 to write to the controller, even for a single register write request and the Max read block size parameter of the query is set to 2 . The use of function 16 may result in higher communication performance.		
Write Coils	Modbus function for write operations to Output Coils. Select between the function 05 (write single coil) and function 15 (write multiple coils).		
	If Modbus function 05 is selected, the protocol will always use function 05 for writing to the controller, even when writing to multiple consecutive coils.		
	If Modbus function 15 is selected, the protocol will always use function 15 to write to the controller, even for a single coil write request. The use of function 15 may result in higher communication performance.		
Transmission	RTU: use RTU mode ASCII: use ASCII mode		
Mode			
	Note: When PLC network is active, all nodes will be configured with the same Transmission Mode.		
PLC Models	Allows to select between different PLC models:		
	 Modicon Modbus (1-based): Modbus implementation where all resources starts with offset 1. 		
	 Generic Modbus (0-based): Modbus implementation where all resources starts with offset 0. 		
	 Enron Modbus (1-based): Extends Modicon Mobdus implementation with 32 bit registers memory area. 		
	 Enron Modbus (0-base): Extends Generic Modbus implementation with 32 bit registers memory area. 		
	Note: The address range used in the Modbus frames is always between 0 and 65535 for the Holding Registers and between 0 and 65535 for Coils.		
Comm	If clicked displays the communication parameters setup dialog.		

Element	Description	
	Comm Parameter Dialog Port Port Baudrate 19200 Parity Even Data Bits 8 Stop bits 1 Mode	
	Element	Parameter
	Port	 Serial port selection. COM1: On-board port COM2: Optional Plug- in module plugged on slot#1 or slot#2 COM3: Optional Plug- in module plugged on slot#3 or slot#4
	Baudrate, Parity, Data Bits, Stop bits	Serial line parameters.
	Mode	Serial port mode. Available modes: • RS-232. • RS-485 (2 wires). • RS-422 (4 wires).
PLC Network	Multiple controllers can be connected to one HMI devic connections, select PLC network and click Add to con	

Path: ProjectView> Config > double-click Tags

- 1. To add a tag, click +: a new line is added.
- 2. Select **Modbus RTU** from the protocol list: tag definition dialog is displayed.

+/-

Element	Description				
Memory	Modbus resource where tag is located.				
Туре	Memory Type	Description			
	Coil Status	Coils			
	Input Status	Discrete Input			
	Input Registers	Input Registers	Input Registers		
	Holding Registers	Holding Registers			
	32 bit Registers	32 bit registers men	nory area.		
		Available only for E	nron Modbus PLC Mode	s	
	Node Override ID				
	Modicon Mode				
	Serial Baudrate		protocol parameter (see Special Data Types for mode details)		
	Serial Parity	protocol parameter details)			
	Serial Stop Bits				
	Serial Mode				
	Serial Done				
Offset	Offset address where tag is	ocated.			
	Offset addresses are six digi address.	ts composed by one dig	git data type prefix + five d	igits resource	
	Memory Type	Studio Offset range	Modicon Offset range	Generic Modbus Offset range	
	Coil Status	0 – 65535			
	Input Status	100000 – 165535]		
	Input Registers	300000 - 365535	1 – 65536	0-65535	
	Holding Registers	400000 - 465535			
	32 bit Registers	0 – 65535			
SubIndex	This allows resource offset selection within the register.				

Element	Description				
Data Type	Data Type	Memory Space	Limits		
	boolean	1-bit data	01		
	byte	8-bit data	-128 127		
	short	16-bit data	-32768 32767		
	int	32-bit data	-2.1e9 2.1e9		
	int64	64-bit data	-9.2e18 9.2e18		
	unsignedByte	8-bit data	0 255		
	unsignedShort	16-bit data	0 65535		
	unsignedInt	32-bit data	04.2e9		
	uint64	64-bit data	01.8e19		
	float	IEEE single-precision 32-bit floating point type	1.17e-38 3.4e38		
	double	IEEE double-precision 64-bit floating point type	2.2e-308 1.79e308		
	string	Array of elements containing character selected encoding	code defined by		
	binary	Arbitrary binary data			
	Note: to define arrays. select one of Data Type format followed by square brackets like "byte[]", "short[]"				
Arraysize	 In case of array tag, this property represents the number of array elements. In case of string tag, this property represents the maximum number of bytes available in the string tag. 				
	Note: number of bytes corresponds to number of string characters if Encoding property is set to UTF-8 or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2 bytes.				
Conversion	Conversion to be applied to the	e tag.			

Description			
Conversion			
inv,swap2	Allowed BCD AB->BA ABCD->CDAB ABCD=FGH->GHEFCDAB Inv bits Cancel OK Cancel OK		
Depending on d	lata type selected, the list Allowed shows one or more conversion		
Value Description			
Inv bits	inv : Invert all the bits of the tag. <i>Example:</i> $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)		
Negate	neg: Set the opposite of tag value.		
	<i>Example:</i> 25.36 → -25.36		
AB -> BA	swapnibbles : Swap nibbles in a byte. <i>Example:</i> $15D4 \rightarrow 514D$ (in hexadecimal format) $5588 \rightarrow 20813$ (in decimal format)		
ABCD -> CDAB	swap2: Swap bytes in a word.Example: $9ACC \rightarrow CC9A$ (in hexadecimal format) $39628 \rightarrow 52378$ (in decimal format)swap4: Swap bytes in a double word.Example: $32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format) $855441236 \rightarrow 1426062386$ (in decimal format)		
ABCDEFGH -> GHEFCDAB			
ABCNOP -	swap8: Swap bytes in a long word.		
> OPMDAB	Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) 0.10000000110 0001110010110110100001011010000111001010		

Element	Description		
	Value Description		
		001 \rightarrow 1 10000011100 101010000101000101101101101101000010010	
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9)	
		Example: $23 \rightarrow 17$ (in decimal format) $0001 \ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)	
	Select conversion	on and click +. The selected item will be added to list Configured .	
	If more convers Configured).	ions are configured, they will be applied in order (from top to bottom o	of list
	Use the arrow b	outtons to order the configured conversions.	

Node Override ID

The protocol provides the special data type Node Override ID which allows you to change the node ID of the slave at runtime. This memory type is an unsigned byte.

The node Override ID is initialized with the value of the node ID specified in the project at programming time.

Node Override ID	Modbus operation
0	Communication with the controller is stopped. In case of write operation, the request will be transmitted without waiting for a reply.
1 to 254	It is interpreted as the value of the new node ID and is replaced for runtime operation.
255	Communication with the controller is stopped; no request messages are generated.



Note: Node Override ID value assigned at runtime is retained through power cycles.

Memory Type	Offset	SubIndex		
Node Override ID 🔻	0	0 -		
Data Type	Arraysize	Conversion		
unsignedByte 👻	0		+//	

Modicon Mode

The protocol provide a special data type that can be used to override the Modicon Mode parameter at runtime.

Modicon Mode	Description	
0	Generic Modbus (0-based). Register indexes start from 0.	
1	Modicon Modbus (1-based). Register indexes start from 1.	



Note: Modicon Mode parameter value assigned at runtime is retained through power cycles.

Modbus RTU	×
Modbus RTU	
Memory Type	Offset SubIndex
Modicon Mode 👻	
Data Type	Arraysize Conversion
boolean 👻	0 +/-
	OK Cancel Apply Help

Serial Parameters Override

The protocol provide special data types that can be used to override the serial parameters at runtime.

Parameter	Description			
Serial Baudrate	unsigned 32 bit value for baudrate overriding. Possible values are 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200.			
Serial Parity	unsigned 8 bit value for parity overriding. Possible values are described in the following list.			
	Value Description			
	0 none parity 1 even parity			
2 odd parity				
Serial Stop Bits	unsigned 8 bit value for stop bits overriding. Possible values are 1, 2.			
Serial Mode	unsigned	unsigned 8 bit value for serial mode overriding. Possible values are described in the following list.		

Parameter	Description		
	Value	Description	
	0	RS-232 mode	
	1	RS-485 mode	
	2	RS-422 mode	
Serial Done	Set to 1 to overwrite the communication line parameters. The parameters are processed all together only when this variable is set to value 1		

Modbus RTU		×
Modbus RTU		
Memory Type	Offset SubInde	×
Serial Baudrate	0	v
Input bits Output coils Holding registers Input registers 32 bit Long Integer 32 bit Floating Point Node Override ID Modicon Mode	Arraysize Conversio	on +/-
Serial Baudrate Serial Parity Serial Stop Bits Serial Mode Serial Done		
	ОК	Cancel Apply Help

Tag Import

Select the driver in Tag Editor and click on the Import Tags button to start the importer.

	Tags	×								
+	—	X	D	ß	>]	₽	A 9B	B>	ŧ i ł	1
Data			^		-	Та	g URI			

The following dialog shows which importer type can be selected.

HMIStudio	×
Multiple tag import	ters are available for this protocol. Please select the importer type and continue.
Version	Туре
Modbus Generic csv v1.0	Linear
DELTA PLC csv v1.0	Linear
Schneider Unity v1.0	Linear
Tag Editor exported xml	General
	OK Cancel

Туре	Description		
Modbus Generic csv v1.0 Linear	Requires a .csv file. All variables will be displayed at the same level.		
DELTA PLC csv v1.0	Requires a .csv file. All variables will be displayed at the same level.		
Schneider Unity v1.0 Linear	Requires a .uny file. The file containing symbols must be exported in .txt format and later renamed as .uny . The importer considers only variables located at fixed address and disregards arrays of strings. All other arrays, except for boolean type, are expanded.		
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.		

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

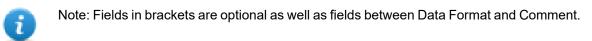
Tags × Protocols					
+ - 🎽 🙆 🔊	D 🗞 🖬 🚮	R 🔎 - Search Tilter by: Data	▼ Ite	ems used:6/10000 Protocol: Show a	all 🛛 🗹 Show all tags 🔅 🗖
Data	Туре	Comment	^	Property	Value
Modbus TCP:prot1	Container			✓ Driver	
Model: Modicon Modbus(1-base	1)			Model	Modicon Modbus(1-based)
- Holding Registers 1	unsignedShort			Protocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			✓ Dictionary	
Holding Registers 3	unsignedShort			Array	false
- MRTU1 - MRTU2	unsignedShort unsignedShort			Array size	0
MRTU2	unsignedShort		_	Arravindex.Subindex	400003
MRTU4	unsignedShort				00005
- MRTU5	unsignedShort			Comment	i let i
1	an argine a an for e		_	Data type	unsignedShort

Toolbar item	Description
B ≱	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.
	Example of both checked and unchecked result:
	Tags* × Tags* × + × · · · · · · · · · · · · · · · · ·
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Modbus Generic csv file structure

This protocol supports the import of tag information when provided in .csv format according to the following format:

NodeID, TagName, MemoryType, Address, DataFormat,..., [Comment]



Field	Description
NodelD	Node the tag belongs to
TagName	Tag description
MemoryType	 OUTP INP IREG HREG
Address	Offset compatible with Modbus notation
DataFormat	Data type in internal notation. See "Programming concepts" section in the main manual.
Comment	Optional additional description.

Tag file example

Example of .csv line:

2, Holding Register 1, HREG, 400001, unsignedShort,



Note: This line has no comment. When the Comment is missing, the comma as a terminator character is mandatory.

Communication status

Current communication status can be displayed using System Variables. See "System Variables" section in the main manual.

Codes supported for this communication driver:

Error	Cause	Action
No response	No reply within the specified timeout.	Check if the controller is connected and properly configured to get network access.
Incorrect node address in response	The device received a response with an invalid node address from the controller .	-
The received message too short	The device received a response with an invalid format from the controller .	-
Incorrect writing data acknowledge	The controller did not accept a write request.	Check if project data is consistent with the controller resources.

Modbus RTU Server

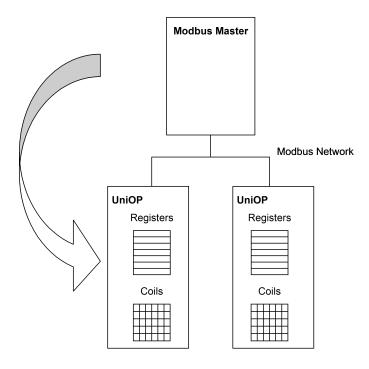
Modbus RTU Server communication driver allows connecting the HMI device as a slave in a Modbus RTU network. Standard Modbus messages are used for information exchange.

This approach allows connecting HMI devices to SCADA systems through the universally supported Modbus RTU communication protocol.

Principle of operation

This communication driver implements a Modbus RTU slave unit in the HMI device. A subset of the complete range of Modbus function codes is supported. The available function codes allow data transfer between the master and the slave.

The following diagram shows the system architecture.



The HMI device is actually simulating the communication interface of a PLC: Coils and Registers are respectively boolean and 16 bit integers.

The device always access data in its internal memory. Data can be transferred to and from the Modbus Master only on initiative of the Master itself.

Implementation details

This Modbus RTU slave implementation supports only a subset of the standard Modbus function codes.

Code	Function	Description
01	Read Coil Status	Reads multiple bits in the device Coil area.
03	Read Holding Registers	Read multiple device Registers.

Code	Function	Description
05	Force Single Coil	Forces a single device Coil to either ON or OFF.
06	Preset Single Register	Presets a value in a device Register.
08	Loopback Diagnostic Test	Only sub function 00 (Return Query Data) is supported.
15	Force Multiple Coils	Forces multiple device Coils to either ON or OFF.
16	Preset Multiple Registers	Presets value in multiple device Registers.
17	Report Slave ID	Returns diagnostic information of the controller present at the slave address.
23	Read Write Multiple Registers	Read & presets values in multiple device Registers

Exception Codes

Code	Description
01	Illegal Function. the function code received in the query is not supported
02	Illegal Data Address . Data Address received in the query exceeds the predefined data range (see Tag Definition for detailed ranges of all types).
03	Illegal Data Value . A sub function other than 00 is specified in Loopback Diagnostic Test (Code 08).

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the $\ensuremath{\text{PLC}}$ list.

The protocol configuration dialog is displayed.

Modbus RTU Server		×
	Comm OK	
Modbus ID	1 Cancel	
🔲 Enron 32bit regis	ters	
32bit reg Start	0	
32bit reg Size	0	
PLC Models		
Modicon Modbus(1- Generic Modbus(0-t Enron Modbus(1-ba Enron Modbus(0-ba		

Eleme nt	Description		
Modbu s ID	Modbus node ID. Every Modbus server device in the network must have its own Modbus ID.		
Enron 32bit registe rs	If selected, allows to define the first register address and the number of registers for 32 bit registers memory area. Note: 32 bit registers are available only for Enron Modbus PLC Models.		
32bit reg Start 32bit reg Size	 32 bit registries memory area definition. Start value represents the first register address. Size value represents the number of registries. Note: A request to one of the registries inside this area gives a 4 byte answer. 		
PLC Models	 Allows to select between different PLC models: Modicon Modbus (1-based): Modbus implementation where all resources starts with offset 1. Generic Modbus (0-based): Modbus implementation where all resources starts with offset 0. Enron Modbus (1-based): Extends Modicon Mobdus implementation with 32 bit registers memory area. Enron Modbus (0-base): Extends Generic Modbus implementation with 32 bit registers memory area. 		

Eleme nt	Description			
		nge used in the Modbus frames is always between 0 and 65535 ters and between 0 and 65535 for Coils.		
Com	If clicked, displays the communication parameters setup dialog.			
m	You have to set parameters acc	cording to the values programmed in Modbus Master.		
	Comm Parameter Dialog			
	ОК			
	Uart com1	_		
	Baudrate 9600	▼		
	Parity even			
	Data bits 8	▼		
	Stop bits	▼		
	Mode RS-485	▼		
	Element	Description		
	Uart	Serial port selection.		
		COM1: On-board port		
		 COM2: Optional Plug-in module plugged on slot#1 or slot#2 		
		 COM3: Optional Plug-in module plugged on slot#3 or slot#4 		
Baudrate, Parity, Data bits, Serial line parameters. Stop bits		Serial line parameters.		
	Mode	Serial port mode. Available options:		
		• RS-232		
		• RS-485 (2 wires)		
		• RS-422 (4 wires)		

Path: ProjectView> Config > double-click Tags

- 1. To add a tag, click +: a new line is added.
- 2. Select **Modbus RTU Server** from the protocol list: tag definition dialog is displayed.

Modbus RTU Server		x
Modbus RTU Server		
Memory Type	Offset SubIndex	
Coil status	▼ 1 ▲ 0 ▼	
Data Type	Arraysize Conversion	
boolean	▼ 0 +/-	
	OK Cancel Apply Hel	<u> </u>

Mamaria						
Memory	Modbus resource where tag is located.					
Туре	Memory Type	Modbus Resource	Modbus Resource			
	Coil Status	Coils				
	Input Status	Discrete Input				
	Input Registers	Input Registers				
	Holding Registers	Holding Registers				
	32 bit Registers	32 bit registers men	32 bit registers memory area.			
		Available only for Enron Modbus PLC Models				
	Node Override ID					
	Modicon Mode					
	Serial Baudrate					
	Serial Parity	protocol parameter (see Special Data Types for mo details)		for mode		
	Serial Stop Bits					
	Serial Mode					
	Serial Done					
Offset	Offset address where tag is located.					
	Offset addresses are six digits composed by one digit data type prefix + five digits resource address.					
	Memory Type	Studio Offset range	Modicon Offset range	Generic Modbus Offset range		
	Coil Status	0 – 65535				
	Input Status	100000 – 165535				
	Input Registers	300000 - 365535	1 – 65536	0 – 65535		
	Holding Registers	400000 - 465535				
	32 bit Registers	0 – 65535				
SubIndex	This allows resource offset selection within the register.					

Element	Description				
Data type	Data Type	Memory Space	Limits		
	boolean	1-bit data	01		
	byte	8-bit data	-128 127		
	short	16-bit data	-32768 32767		
	int	32-bit data	-2.1e9 2.1e9		
	int64	64-bit data	-9.2e18 9.2e18		
	unsignedByte	8-bit data	0 255		
	unsignedShort	16-bit data	0 65535		
	unsignedInt	32-bit data	0 4.2e9		
	uint64	64-bit data	0 1.8e19		
	float	IEEE single-precision 32-bit floating point type	1.17e-38 3.4e38		
	double	IEEE double-precision 64-bit floating point type	2.2e-308 1.79e308		
	string	Array of elements containing characte selected encoding	Array of elements containing character code defined by selected encoding		
	binary	Arbitrary binary data			
Arraysize	Note: to define arrays. select one of Data Type format followed by square brackets like "byte[]", "short[]" When configuring array or string tags, this option define the amount of array elements or characters of the string.				
Conversion	Conversion to be applied to the tag.				
	Conversion				
		Allowed Configured			
		BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits			
		Cancel OK			

Element	Description	
	Depending on data type selected, the list Allowed shows one or more conversion	
	Value	Description
	Inv bits	inv : Invert all the bits of the tag.
		<i>Example:</i> 1001 \rightarrow 0110 (in binary format) 9 \rightarrow 6 (in decimal format)
	Negate	neg: Set the opposite of tag value.
		<i>Example:</i> 25.36 → -25.36
AB -> BA swapnibbles : Swap nibbles in a byte.		swapnibbles: Swap nibbles in a byte.
		<i>Example:</i> 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)
	ABCD -> CDAB	swap2 : Swap bytes in a word. <i>Example:</i> $9ACC \rightarrow CC9A$ (in hexadecimal format) $39628 \rightarrow 52378$ (in decimal format)
	ABCDEFGH	swap4: Swap bytes in a double word.
	-> GHEFCDAB	<i>Example:</i> 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)

Element	Description		
	Value	Description	
	ABCNOP - > OPMDAB	swap8 : Swap bytes in a long word. Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) $0\ 10000000110$ 0001110010111011010001011010000111001010	
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)	
		on and click +. The selected item will be added to list Configured . ions are configured, they will be applied in order (from top to bottom of list	
	Use the arrow b	uttons to order the configured conversions.	

Node Override ID

The protocol provides the special data type Node Override ID which allows you to change the node ID of the slave at runtime. This memory type is an unsigned byte.

The node Override ID is initialized with the value of the node ID specified in the project at programming time.

Node Override ID	Modbus operation
0	Communication with the slave is stopped. In case of write operation, the device will not respond to request frames.
1 to 255	It is interpreted as the value of the new node ID and is replaced for runtime operation.



Note: Node Override ID value assigned at runtime is retained through power cycles.

10dbus RTU Server			
Memory Type	Offset	SubIndex	
Node Override ID 👻	0		
Data Type	Arraysize	Conversion	
unsignedByte 🗸	0	+/-	

Modicon Mode

The protocol provide a special data type that can be used to override the Modicon Mode parameter at runtime.

Modicon Mode	Description	
0 Generic Modbus (0-based). Register indexes start from 0.		
1 Modicon Modbus (1-based). Register indexes start from 1.		



Note: Modicon Mode parameter value assigned at runtime is retained through power cycles.

Modbus RTU Server	X	
Modbus RTU Server		
Memory Type	Offset SubIndex	
Modicon Mode 👻	0 • •	
Data Type	Arraysize Conversion	
boolean -	0 +/-	
	OK Cancel Apply Help	

Serial Parameters Override

The protocol provide special data types that can be used to override the serial parameters at runtime.

Parameter	Description				
Serial Baudrate	unsigned 32 bit value for baudrate overriding. Possible values are 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200.				
Serial Parity	unsigned 8 bit value for parity overriding. Possible values are described in the following list.				
	Value	Value Description			
	0	0 none parity 1 even parity 2 odd parity			
	1				
	2				
Serial Stop Bits	unsigned 8 bit value for stop bits overriding. Possible values are 1, 2.				
Serial Mode	unsigned 8 bit value for serial mode overriding. Possible values are described in the following list.				

Parameter	Descript	Description		
	Value	Description		
	0 RS-232 mode 1 RS-485 mode			
	2	RS-422 mode		
Serial Done	Set to 1 to overwrite the communication line parameters. The parameters are processed all together only when this variable is set to value 1			

Modbus RTU Server				
Modbus RTU Server				
Memory Type	Offset SubIndex			
Serial Baudrate 🗢	0 • 0 •			
Coil status Input status Input registers Holding registers 32 bit Long Integer 32 bit Floating Point Node Override ID Modicon Mode	Arraysize Conversion 0 +/-			
Serial Baudrate Serial Parity Serial Stop Bits Serial Mode Serial Done				
	OK Cancel Appl	y Help		

Tag Import

Select the driver in Tag Editor and click on the Import Tags button to start the importer.



The following dialog shows which importer type can be selected.

HMIStudio				X
Multiple t	ag importers are available for th	s protocol. Please selec	ct the importer type and cont	inue.
Version	Туре			
Modbus Generic	csv v1.0 Linear			
Tag Editor exported xml General				
			OK Cance	1

Туре	Description		
Modbus Generic csv v1.0 Linear	Requires a .csv file. All variables will be displayed at the same level.		
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.		
	Tags X + - X C V S B D I I Data		

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

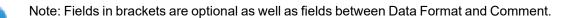
Tags × Protocols						•
+ - 🕉 🕲 刘	(> 🛟 🖬	R 🔎 - Search Tilter by: Data	▼ Ite	ms used:6/1000	9 Protocol: Show all	🗹 Show all tags 🖉 🗔
Data	Туре	Comment	^	Property		Value
Modbus TCP:prot1	Container			✓ Driver		
Model: Modicon Modbus(1-based)			Model		Modicon Modbus(1-based)
- Holding Registers 1	unsignedShort			Protoc	ol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			✓ Dictionary		
- Holding Registers 3	unsignedShort			Array		false
- MRTU1	unsignedShort					
- MRTU2	unsignedShort			Array s	size	0
MRTU3	unsignedShort			Arrayin	ndex.Subindex	400003
- MRTU4	unsignedShort			Comme	ent	
- MRTU5	unsignedShort			Data to	/ne	unsignedShort
	and the second sec				P.4	anogradorere

Toolbar item	Description
R	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
B ä	Update Tag(s). Click on this icon to update the tags in the project, due a
	new dictionary import.
R	Check this box to import all sub-elements of a tag. Example of both checked and unchecked result:
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Modbus Generic csv file structure

This protocol supports the import of tag information when provided in **.csv** format according to the following format:

NodeID, TagName, MemoryType, Address, DataFormat, ..., [Comment]



Field	Description	
NodelD	Node the tag belongs to	
TagName	Tag description	
MemoryType	• OUTP	
	• INP	
	• IREG	
	• HREG	
Address	Offset compatible with Modbus notation	

ĩ

Field	Description	
DataFormat	Data type in internal notation. See "Programming concepts" section in the main manual.	
Comment	Optional additional description.	

Tag file example

Example of .csv line:

2, Holding Register 1, HREG, 400001, unsignedShort,



Note: This line has no comment. When the Comment is missing, the comma as a terminator character is mandatory.

Communication status

Current communication status can be displayed using system variables. This communication protocol acts as server and doesn't return any specific Protocol Error Message.

See "System Variables" section in the main manual.

Modbus TCP

Various Modbus TCP-capable devices can be connected to HMI devices. To set-up your Modbus TCP device, please refer to the documentation you have received with the device.

The implementation of the protocol operates as a Modbus TCP client only.

Implementation details

This Modbus TCP implementation supports only a subset of the Modbus TCP standard function codes.

Code	Function	Description	
01	Read Coil Status	Reads multiple bits in the HMI device Coil area.	
02	Read Input Status	Reads the ON/OFF status of the discrete inputs (1x reference) in the slave.	
03	Read Holding Registers	Reads multiple registers.	
04	Read Input Registers	Reads the binary contents of input registers (3x reference) in the slave.	
05	Force Single Coil	Forces a single coil to either ON or OFF.	
06	Preset Single Register	Writes a value to one register.	
15	Write Multiple Coils	Writes each coil in a sequence of coils to either ON or OFF.	
16	Preset Multiple Registers	Writes values to a block of registers in sequence.	

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the PLC list.

The protocol configuration dialog is displayed.

Modbus TCP		×
PLC Network		ОК
Alias		Cancel
IP address	0.0.0.0	
Port	502	
use UDP/IP		
Encapsulated RTU		
Timeout (ms)	2000	
Modbus ID	1	
Max read block	250	
Max read bit block	2000	
Write Holding Register	1 6 •	
Write Coils	15 💌	
PLC Models		
Modicon Modbus(1-bas	ed)	
Generic Modbus(0-bas	ed)	
Enron Modbus(1-based Enron Modbus(0-based	d) with 32bit registers d) with 32bit registers	

Element	Description
Alias	Name identifying nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node.
IP address	Address of the controller.
Port	Port number used by the Modbus TCP driver. The default value is 502 and can be changed when the communication goes through routers or Internet gateways where the default port number is already in use.
use UDP/IP	If selected, the protocol will use connectionless UDP datagrams.
Encapsulate d RTU	If selected, the protocol will use serial RTU protocol over Ethernet instead of Modbus TCP protocol, independently from TCP or UDP usage.
Timeout (ms)	Time delay in milliseconds between two retries in case of missing response from the server device.

Element	Description	
Modbus ID	Usually used when communicating over Ethernet-to-serial gateways and then interpreted as the Slave ID. This value is simply copied into the Unit Identifier field of the Modbus TCP communication frame. This must correspond to server configuration. In most cases, server answers to Modbus ID 1, so this parameter can be left 1.	
Max read block	Maximum length in bytes of a data block request. It applies only to read access of Holding Registers.	
Max read bit block	Maximum length in bits of a block request. It applies only to read access of Input Bits and Output Coils.	
Write Holding	Modbus function for write operations to Holding Registers. Select between the function 06 (preset single register) and function 16 (preset multiple registers).	
Register	If 06 is selected, the protocol will always use function 06 for writing to the controller, even when writing to multiple consecutive registers.	
	If 16 is selected, the protocol will always use function 16 to write to the controller, even for a single register write request and the Max read block size parameter of the query is set to 2 . The use of function 16 may result in higher communication performance.	
	If Auto is selected, the protocol will use both function 06 or function 16 depending on number of registries to be written.	
Write Coils	Modbus function for write operations to Output Coils. Select between the function 05 (write single coil) and function 15 (write multiple coils).	
	If Modbus function 05 is selected, the protocol will always use function 05 for writing to the controller, even when writing to multiple consecutive coils.	
	If Modbus function 15 is selected, the protocol will always use function 15 to write to the controller, even for a single coil write request. The use of function 15 may result in higher communication performance.	

Element	Description			
PLC Models	Allows to select between different PLC models:			
	 Modicon Modbus (1-based): Modbus implementation where all resources starts with offset 1. 			
	 Generic Modbus (0-based): Modbus implementation where all resources starts with offset 0. 			
	• Enron Modbus (1-based): Extends Modicon Mobdus implementation with 32 bit registers memory area.			
	 Enron Modbus (0-base): Extends Generic Modbus implementation with 32 bit registers memory area. 			
	Note: The address range used in the Modbus frames is always between 0 and 65535 for the Holding Registers and between 0 and 65535 for Coils.			
PLC Network	IP address for all controllers in multiple connections. PLC Network must be selected to enable multiple connections.			

lement	Description		
	Modbus TCP	Modbus TCP	٢
	PLC Network	ОК	
	Alias	Alias	
	IP address	IP address 0 . 0 . 1	
	Port	Port 502	
	use UDP/IP	use UDP/IP	
	Encapsulated R	Encapsulated RTU	
	Timeout (ms)	Timeout (ms) 2000	
	Modbus ID	Modbus ID 1	
	Max read block	Max read block 250	
	Max read bit block	Max read bit block 2000	
	Write Holding Regis	Write Holding Register 16	
	Write Coils	Write Coils	
	PLC Models Modicon Modbus(1) Generic Modbus(0- Enron Modbus(1-b Enron Modbus(0-b	Enrop Modbus(1-based) with 32bit registers	
	Slaves		
	Slave Id	Model Alias	

Tag Editor Settings

Path: ProjectView> Config > double-click Tags

- 1. To add a tag, click +: a new line is added.
- 2. Select **Modbus TCP** from the **Driver** list: tag definition dialog is displayed.

Nodbus TCP			
Memory Type Input bits	 Offset ▼ 100001 ▲ 	SubIndex	
Data Type	Arraysize	Conversion	
boolean	•		+/-

Element	Description			
Memory	Modbus resource where tag is located.			
Туре	Memory Type	Modbus Resource		
	Coil Status	Coils		
	Input Status	Discrete Input		
	Input Registers	Input Registers		
	Holding registers	Holding Registers		
	32 bit Registers	32 bit registers memory area.		
		Available only for Enron Modbus PLC Models		
	Node Override IP			
	Node Override Port	protocol parameter (see Special Data Types for mode details)		
	Node Override ID	protocorparameter (see Special Data Types for mode details)		
	Modicon Mode			
Offset	Offset address where tag is	located.		
	Offset addresses are six digits composed by one digit data type prefix + five digits resource addres			

Element	Description					
	Memory Type	Studio Offset range	Modicon Offset rang	e	Generic Modbus Offset range	
	Coil Status	0 – 65535				
	Input Status	100000 – 165535				
	Input Registers	300000 – 365535	1 – 65536		0 – 65535	
	Holding Registers	400000 - 465535				
	32 bit Registers	0 – 65535		1		
SubIndex	This allows resource offset	set selection within the register.				
Data Type	Data Type	Memory Space		Limits		
	boolean	1-bit data		0 1		
	byte	8-bit data	8-bit data -128 127			
	short	16-bit data -32768 327		2767		
	int	32-bit data -2.1		-2.1e9 2.′	.1e9 2.1e9	
	int64	64-bit data -9.2e		-9.2e18 9	e18 9.2e18	
	unsignedByte	8-bit data	8-bit data 0 255			
	unsignedShort	16-bit data 0 65535				
	unsignedInt	32-bit data 0		04.2e9		
	uint64	64-bit data 0 1.		0 1.8e19	3e19	
	float	IEEE single-precision point type			1.17e-38 3.4e38	
	double	IEEE double-precision point type			2.2e-308 1.79e308	
	string	Array of elements containing character code defined by selected encoding			by selected	
	binary	Arbitrary binary data				
	Note: to define arrays. select one of Data Type format followed by square brackets like "byte []", "short[]"					
Arraysize		is property represents the number of array elements. his property represents the maximum number of bytes available in the				

Element	Description			
	string tag. Note: number of bytes corresponds to number of string characters if Encoding property is set to UTF-8 or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2 bytes.			
Conversion	Conversion to be appli	ied to the tag.		
	Conversion			
	inv,swap2	Allowed Configured BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits Cancel OK		
	Depending on data type selected, the list Allowed shows one or more conversion types.			
	Value	Description		
	Inv bits	inv: Invert all the bits of the tag. <i>Example:</i> $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)		
	Negate	neg : Set the opposite of tag value. <i>Example:</i> $25.36 \rightarrow -25.36$		
	AB -> BA	swapnibbles : Swap nibbles in a byte. <i>Example:</i> $15D4 \rightarrow 514D$ (in hexadecimal format) $5588 \rightarrow 20813$ (in decimal format)		
	ABCD -> CDAB	swap2 : Swap bytes in a word. <i>Example:</i> $9ACC \rightarrow CC9A$ (in hexadecimal format) $39628 \rightarrow 52378$ (in decimal format)		
	ABCDEFGH -> GHEFCDAB	swap4 : Swap bytes in a double word. <i>Example:</i> $32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format)		

Description	
Value	Description
	$855441236 \rightarrow 1426062386$ (in decimal format)
ABCNOP -> OPMDAB	swap8: Swap bytes in a long word.Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) $0 10000000110$ $0001110010111011010000101101000011100000$
BCD Select conversion and	 bcd: Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> 23 → 17 (in decimal format) 0001 0111 = 23 0001 = 1 (first nibble) 0111 = 7 (second nibble) click +. The selected item will be added to list Configured.
	ABCNOP -> OPMDAB

If more conversions are configured, they will be applied in order (from top to bottom of list Configured).

Use the arrow buttons to order the configured conversions.

Node Override IP

The protocol provides the special data type Node Override IP which allows you to change the IP address of the target controller at runtime.

This memory type is an array of 4 unsigned bytes, one per each byte of the IP address.

The Node Override IP is initialized with the value of the controller IP specified in the project at programming time.

Node Override IP	PLC operation
0.0.0.0	Communication with the controller is stopped, no request frames are generated anymore.
Different from 0.0.0.0	It is interpreted as node IP override and the target IP address is replaced runtime with the new value.

If the HMI device is connected to a network with more than one controller node, each node has its own Node Override IP variable.



Note: Node Override IP values assigned at runtime are retained through power cycles.

Hostname DNS or mDNS

In addition to the array of bytes, string memory type can be selected to be able use the DNS or mDNS hostname as an alternative to the IP Address.

Modbus TCP			×
Modbus TCP			
Memory Type	Offset	SubIndex	
Node Override IP 🔻	0	0 -	
Data Type	Arraysize	Conversion	
unsignedByte [] 🔻	4		+/-
	C	Cancel	Apply Help

Node Override Port

The protocol provides the special data type Node Override Port which allows you to change the network Port of the target controller at runtime.

This memory type is unsigned short.

Node Override Port is initialized with the value of the controller Port specified in the project at programming time.

Node Override Port	t Modbus operation		
0	Communication with the controller is stopped, no request frames are generated anymore.		
Different from 0	It is interpreted as the value of the new port and is replaced for runtime operation.		

If the HMI device is connected to a network with more than one controller node, each node has its own Node Override Port variable.



Note: Node Override Port values assigned at runtime are retained through power cycles.

Modbus TCP	×
Modbus TCP	
Memory Type	Offset SubIndex
Node Override Port 👻	
Data Type	Arraysize Conversion
unsignedShort 👻	0 +/-
	OK Cancel Apply Help

Node Override ID

The protocol provides the special data type Node Override ID which allows you to change the node ID of the slave at runtime. This memory type is an unsigned byte.

The node Override ID is initialized with the value of the node ID specified in the project at programming time.

Node Override ID	Modbus operation
0	Communication with the controller is stopped. In case of write operation, the request will be transmitted without waiting for a reply.
1 to 254	It is interpreted as the value of the new node ID and is replaced for runtime operation.
255	Communication with the controller is stopped; no request messages are generated.



Note: Node Override ID value assigned at runtime is retained through power cycles.

Modbus TCP		X
Modbus TCP		
Memory Type	Offset SubIndex	
Node Override ID 🔻	0 • •	
Data Type	Arraysize Conversion	
unsignedByte 👻	0 +/-	
	OK Cancel Apply	Help

Modicon Mode

The protocol provide a special data type that can be used to override the Modicon Mode parameter at runtime.

Modicon Mode	Description
0	Generic Modbus (0-based). Register indexes start from 0.
1	Modicon Modbus (1-based). Register indexes start from 1.



Note: Modicon Mode parameter value assigned at runtime is retained through power cycles.

Nodbus TCP		×
Memory Type Modicon Mode	Offset SubIndex	
Data Type boolean	Arraysize Conversion	+/-
	OK Cancel	Apply Help

Tag Import

Select the driver in Tag Editor and click on the Import Tags button to start the importer.

	Tags	×								
+	-	Z	đ	ß	>]	₽	A 9B	B>	63	1
Data			^		-	Та	g URI			

The following dialog shows which importer type can be selected.

IMIStudio		x
Multiple tag import	ters are available for this protocol. Please select the importer type and conti	nue.
Version	Туре	-
Modbus Generic csv v1.0	Linear	
DELTA PLC csv v1.0	Linear	-
CODESYS2 sym v1.0	Linear	I
Kollmorgen txt v1.0	Linear	
Schneider Unity v1.0	Linear	
Tag Editor exported xml	General	-
	OK Cancel	

Туре	Description			
Modbus Generic csv	Requires a .csv file.			
v1.0 Linear	All variables will be displayed at the same level.			
DELTA PLC csv v1.0	Requires a .csv file.			
	All variables will be displayed at the same level.			
CODESYS2 sym v1.0	Requires a .sym file.			
Linear	All variables will be displayed at the same level.			
	After selecting the .sym file, the following dialog will appear for PLC model selection.			
	Modbus TCP importer - Filter selection			
	Available PLC Models OK ABB Cancel ABB OK			
	WAGO			
Kollmorgen txt v1.0	Requires a .txt file.			
Linear	All variables will be displayed at the same level.			
Schneider Unity v1.0 Linear	Requires a .uny file.			

Туре	Description	
	The file containing symbols must be exported in .txt format and later renamed as .uny . The importer considers only variables located at fixed address and disregards arrays of strings. All other arrays, except for boolean type, are expanded.	
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.	
	Tags x	
	Data Tag URI	

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

+ - 🎽 🕲 刘	D 🗞 🕞 📅	R P- Search Tilter by:	Data 🔻 Ite	ms used:6/10000 Protocol: Sho	w all 🛛 🗹 Show all tags 🖓 [
Data	Type	Comment	^	Property	Value
Modbus TCP:prot1	Container			Y Driver	
Model: Modicon Modbus(1-bas	ea)			Model	Modicon Modbus(1-based)
- Holding Registers 1	unsignedShort			Protocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			 Dictionary 	
- Holding Registers 3	unsignedShort			Array	false
- MRTU1	unsignedShort				
- MRTU2	unsignedShort			Array size	0
MRTU3	unsignedShort			Arrayindex.Subindex	400003
- MRTU4	unsignedShort			Comment	
- MRTU5	unsignedShort			Data type	unsignedShort

Toolbar item	Description
	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.
	Example of both checked and unchecked result:

Toolbar item	Description
	Heat Task Task <th< th=""></th<>
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Modbus Generic csv file structure

This protocol supports the import of tag information when provided in **.csv** format according to the following format:

NodeID, TagName, MemoryType, Address, DataFormat,..., [Comment]

Note: Fields in brackets are optional as well as fields between Data Format and Comment.

Field	Description
NodelD	Node the tag belongs to
TagName	Tag description
MemoryType	 OUTP INP IREG HREG
Address	Offset compatible with Modbus notation
DataFormat	Data type in internal notation. See "Programming concepts" section in the main manual.
Comment	Optional additional description.

Tag file example

Example of .csv line:

2, Holding Register 1, HREG, 400001, unsignedShort,



1

Note: This line has no comment. When the Comment is missing, the comma as a terminator character is mandatory.

Communication status

Current communication status can be displayed using system variables. See "System Variables" section in the main manual.

Codes supported for this communication driver:

Error	Cause	Action
No response	No reply within the specified timeout.	Check if the controller is connected and properly configured to get network access.
Incorrect node address in response	The device received a response with an invalid node address from the controller.	-
The received message too short	The device received a response with an invalid format from the controller .	-
Incorrect writing data acknowledge	The controller did not accept a write request.	Check if project data is consistent with the controller resources.

Modbus TCP Server

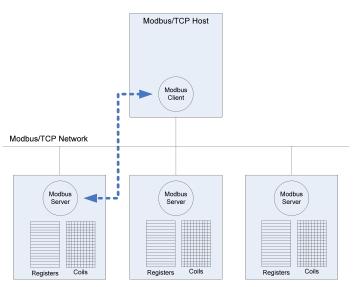
Modbus TCP Server communication driver allows connecting the HMI device as a server in a Modbus TCP network. It is possible for Modbus TCP clients to connect then to multiple HMI panels acting as servers. Standard Modbus TCP messages are used for information exchange.

This approach allows connecting HMI devices to SCADA systems through the universally supported Modbus TCP communication protocol.

Principle of operation

This communication driver implements a Modbus TCP Server unit in HMI device. A subset of the complete range of Modbus function codes is supported. The available function codes allow data transfer between clients on the TCP network and the server. The HMI device acts as a server in the network. It can exchange data with up to 32 clients. This means that up to 32 clients can be connected to the HMI device at the same time. If all the 32 available connections are in use, any further attempt to connect by a client will be refused by the system.

The following diagram shows the system architecture.



The device simulates the communication interface of a PLC: Coils and Registers data types are respectively boolean and 16 bit integers.

The device always access data in its internal memory. Data can be transferred to and from the Modbus Client only on the initiative of the client itself.

Implementation details

This Modbus TCP Server implementation supports only a subset of the Modbus standard function codes.

Code	Function	Description
01	Read Coil Status	Reads multiple bits in the device Coil area.
02	Read Input Status	Reads multiple bits in the device Coil area.
03	Read Holding Registers	Read multiple device Registers.

Code	Function	Description
04	Read Input Registers	Read multiple device Registers.
05	Force Single Coil	Forces a single device Coil to either ON or OFF.
06	Preset Single Register	Presets a value in a device Register.
15	Force Multiple Coils	Forces multiple device Coils to either ON or OFF.
16	Preset Multiple Registers	Presets value in multiple device Registers.
23	Read Write Multiple Registers	Read & presets values in multiple device Registers



Note: For both PLC models the Read Coil Status and Read Input Status function codes both access the same Coil memory area in the HMI device memory. The Read Holding Registers and Read Input Registers function codes both access the same Register area in the HMI device memory.

Exception Codes

Code	Description
01	Illegal Function. the function code received in the query is not supported
02	Illegal Data Address . Data Address received in the query exceeds the predefined data range (see Tag Editor Settings for detailed ranges of all types).
03	Illegal Data Value . A sub function other than 00 is specified in Loopback Diagnostic Test (Code 08).

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the **PLC** list.

The protocol configuration dialog is displayed.

Modbus TCP Server		×
		ОК
Modbus ID	1	Cancel
Port	502	
use UDP/IP		
Encapsulated RTU		
Enron 32bit register	s	
32bit reg Start	0	
32bit reg Size	0	
	ased) sed) ed) with 32bit registers ed) with 32bit registers	

Element	Description	
Modbus ID	Modbus node ID of the HMI device. Every Modbus server device in the network must have its own Modbus ID.	
Port	Port number used by the Modbus TCP protocol. Default value is 502 . Set the value accordingly to the port number used by your Modbus TCP Network.	
use UDP/IP	If selected, the protocol will use connectionless UDP datagrams.	
Encapsulated RTU	If selected, the protocol will use serial RTU protocol over Ethernet instead of Modbus TCP protocol, independently from TCP or UDP usage.	
Enron 32bit registers	If selected, allows to define the first register address and the number of registers for 32 bit registers memory area.	
	Note: 32 bit registers are available only for Enron Modbus PLC Models.	

Element	Description
	32 bit registries memory area definition.
32bit reg Start	Start value represents the first register address.
32bit reg Size	Size value represents the number of registries.
	Note: A request to one of the registries inside this area gives a 4 byte answer.
PLC Models	Allows to select between different PLC models:
	• Modicon Modbus (1-based) : Modbus implementation where all resources starts with offset 1.
	• Generic Modbus (0-based) : Modbus implementation where all resources starts with offset 0.
	• Enron Modbus (1-based): Extends Modicon Mobdus implementation with 32 bit registers memory area.
	 Enron Modbus (0-base): Extends Generic Modbus implementation with 32 bit registers memory area.
	Note: The address range used in the Modbus frames is always between 0 and 65535 for the Holding Registers and between 0 and 65535 for Coils.

Tag Editor Settings

Path: ProjectView> Config > double-click Tags

- 1. To add a tag, click +: a new line is added.
- 2. Select Modbus TCP Server from the protocol list: tag definition dialog is displayed.

lodbus TCP Server				X
Modbus TCP Server				
Memory Type	Offset	SubIndex		
Coil status	• 1	0 -		
Data Type	Arraysize	Conversion		
boolean -	• 0		+/-	
	ОК	Cancel	Apply	Help

Element	Description					
Memory	Modbus resource where tag is located.					
Туре	Memory Type	Modbus Resource				
	Coil Status	Coils				
	Input Status	Discrete Input				
	Input Registers	Input Registers				
	Holding Registers	Holding Registers				
	32 bit Registers	32 bit registers memory	area.			
		Available only for Enron	Modbus PLC Mo	dels.		
	Modicon Mode	protocol parameter (see	Special Data Typ	bes for mo	de details)	
Offset	Offset address where tag is	located.				
	Offset addresses are six dig address.	fset addresses are six digits composed by one digit data type prefix + five digits resource dress.			s resource	
	Memory Type	Studio Offset range	Modicon Offset range		Generic Modbus Offset range	
	Coil Status	0 – 65535				
	Input Status 100000 – 16553					
	Input Registers	300000 – 365535	1 – 65536		0 – 65535	
	Holding Registers	400000 - 465535				
	32 bit Registers	0 – 65535				
SubIndex	This allows resource offset	selection within the regist	er.			
Data type	Data Type	Memory Space		Limits		
	boolean	an 1-bit data		0 1		
	byte	e 8-bit data		-128 127		
	short 16-bit data -3276		-32768 .	32767		
	int 32-bit data		-2.1e9 2.1e9			
	int64	64-bit data		-9.2e18	9.2e18	

Element	Description				
	Data Type	Memory Space	Limits		
	unsignedByte	8-bit data	0 255		
	unsignedShort	16-bit data	0 65535		
	unsignedInt	32-bit data	04.2e9		
	uint64	64-bit data	01.8e19		
	float	IEEE single-precision 32-bit floating point type	1.17e-38 3.4e38		
	double	IEEE double-precision 64-bit floating point type	2.2e-308 1.79e308		
	string	Array of elements containing character selected encoding	r code defined by		
	binary	Arbitrary binary data			
Arraysize	 Note: to define arrays. select one of Data Type format followed by square brackets like "byte[]", "short[]" In case of array tag, this property represents the number of array elements. In case of string tag, this property represents the maximum number of bytes available. 				
	Note: number of bytes corres to UTF-8 or Latin1 in Tag Edi	in the string tag. lote: number of bytes corresponds to number of string characters if Encoding property is se o UTF-8 or Latin1 in Tag Editor. Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character equires 2 bytes.			
Conversio	Conversion to be applied to the	ne tag.			
n	Conversion				
		>BA D->CDAB DEFGH->GHEFCDAB			

Depending on data type selected, the list **Allowed** shows one or more conversion types.

Element	t Description		
	Value	Description	
	Inv bits	inv: Invert all the bits of the tag.	
		<i>Example:</i> 1001 \rightarrow 0110 (in binary format) 9 \rightarrow 6 (in decimal format)	
	Negate	neg : Set the opposite of tag value.	
		<i>Example:</i> 25.36 → -25.36	
	AB -> BA	swapnibbles: Swap nibbles in a byte.	
		<i>Example:</i> 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)	
	ABCD ->	swap2: Swap bytes in a word.	
	CDAB	<i>Example:</i> 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)	
	ABCDEFGH	swap4: Swap bytes in a double word.	
	-> GHEFCDAB	<i>Example:</i> $32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format) $855441236 \rightarrow 1426062386$ (in decimal format)	
	ABCNOP - > OPMDAB	swap8: Swap bytes in a long word. Example: 142.366 \rightarrow -893553517.588905 (in decimal format) 0 1000000110 000111001011101101000101101000011100001110000	
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)	

Element	Description
	Select conversion and click +. The selected item will be added to list Configured .
	If more conversions are configured, they will be applied in order (from top to bottom of list Configured).
	Use the arrow buttons to order the configured conversions.

Modicon Mode

The protocol provide a special data type that can be used to override the Modicon Mode parameter at runtime.

	Modicon Mode	Description	
-	0	Generic Modbus (0-based). Register indexes start from 0.	
-	1	Modicon Modbus (1-based). Register indexes start from 1.	



Note: Modicon Mode parameter value assigned at runtime is retained through power cycles.

Modbus TCP Server				×
Modbus TCP Server				
Memory Type	Offset	SubIndex		
Modicon Mode	▼ 0	▲ ▼ 0 ▼		
Data Type	Arraysize	Conversion		
boolean	▼ 0		+/-	
	OK	Cancel	Apply	Help

Tag Import

Select the driver in Tag Editor and click on the Import Tags button to start the importer.



The following dialog shows which importer type can be selected.

HMIStudio	X
Multiple tag impor	ters are available for this protocol. Please select the importer type and continue.
Version	Туре
Modbus Generic csv v1.0	Linear
CODESYS2 sym v1.0	Linear
Tag Editor exported xml	General
	OK Cancel

Importer	Description
Modbus Generic csv v1.0 Linear	Requires a .csv file. All variables will be displayed at the same level.
CODESYS2 sym v1.0 Linear	Requires a . sym file. All variables will be displayed at the same level. After selecting the . sym file, the following dialog will appear for PLC model selection. Modbus TCP Server importer - Filter s Available PLC Models Cancel ABB Cancel
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags × Protocols					
+ - 👗 🕲 🔊	D 🕼 📾 🖏	R 🔎 - Search 🍸 Filter by: Data	▼ Ite	ms used:6/10000 Protocol: Show all	Show all tags 💮 🗖
Data	Туре	Comment	^	Property	Value
Modbus TCP:prot1	Container			Y Driver	
Model: Modicon Modbus(1-based				Model	Modicon Modbus(1-based)
- Holding Registers 1	unsignedShort			Protocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			✓ Dictionary	
- Holding Registers 3 - MRTU1	unsignedShort			Array	false
- MRTU2	unsignedShort unsignedShort			Array size	0
MRTU3	unsignedShort		_	Arrayindex.Subindex	400003
- MRTU4	unsignedShort			Comment	100005
- MRTU5	unsignedShort			Data type	unsignedShort
	1-1-1			Data type	unsignedShort

Toolbar item	Description	
B	Import Tag(s).	
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project	
δ <u>a</u>	Update Tag(s). Click on this icon to update the tags in the project, due a	
	new dictionary import.	
R	Check this box to import all sub-elements of a tag. Example of both checked and unchecked result:	
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.	

Modbus Generic csv file structure

This protocol supports the import of tag information when provided in **.csv** format according to the following format:

NodeID, TagName, MemoryType, Address, DataFormat, ..., [Comment]



Note: Fields in brackets are optional as well as fields between Data Format and Comment.

Field	Description
NodelD	Node the tag belongs to
TagName	Tag description
MemoryType	• OUTP
	• INP
	• IREG
	• HREG
Address	Offset compatible with Modbus notation
DataFormat	Data type in internal notation. See "Programming concepts" section in the main manual.
Comment	Optional additional description.

Tag file example

Example of .csv line:

2, Holding Register 1, HREG, 400001, unsignedShort,



Note: This line has no comment. When the Comment is missing, the comma as a terminator character is mandatory.

Communication status

The HMI device is a server station in the Modbus TCP network. The current implementation of the protocol doesn't report any communication error code apart from standard communication error codes related to the proper driver loading.

See "System Variables" section in the main manual.

Mitsubishi FX ETH

Mitsubishi FX ETH implements the MELSEC-F (or MC) communication protocol that can be used with FX CPUs as described in the Mitsubishi document "FX3U-ENET USER'S MANUAL", chapter 8 "Communication using MC protocol".



Note: Mitsubishi FX3U controller must be equipped with the appropriate Ethernet module: FX3U-ENET

Protocol Editor Settings

Add [+] a driver in the Protocol editor and select the protocol called "Mitsubishi FX ETH" from the list of available protocols.

Mitsubishi FX ETH		x
PLC Network		ОК
Alias		Cancel
IP address	0.0.0.0	
Port	5551	
PLC Models		
FX1N		
FX2N FX3G		
FX3U		

Element	Description
IP address	Ethernet IP address of the controller
Port	Specifies the port number (decimal) used in the communication with the PLC.

Element	Description	
PLC Model	Defines the PLC n	nodel connected
PLC Network		s the connection of multiple controllers to one operator panel. To set-up ns, check "PLC network" checkbox and enter IP Address for all controllers.

Controller Settings with GX Developer

The Mitsubishi FX system must be properly configured for Ethernet communication using the Mitsubishi FX Configurator. Click on "Operational settings" as shown at point (1) in the following figure:

File View Help				
D 🛎 🖬 🎒				
Ethernet Mod	lule settings			
	Module O	▼		
1	Operational settings			
2	Initial settings			
3	Open settings			
	Router relay parameter			
	E-mail settings			
Necessary setting(No settin Set if it is needed(No settin Online		Check]	
Transfer setup	PLC remote operation	Diagnostics		
Write	Read	Verify		
			_	
l Ready				NUM

Into Operational Settings dialog, verify the "Communication data code" is set to "Binary code",

Then type-in the Controller IP Address and confirm with [End] button.

File View Help	
Communication data code Dinary code C ASCII code ASCII code C ASCII cod C ASC	
IP address	Send frame setting
Input format DEC.	€ Ethernet(V2.0)
IP address 192 168 1 254	C IEEE802.3
C Use the F C Use the F End Cancel	Ping
Ready	NUM

Click now on "Initial settings" as shown at point (2) of Figure below:

For proper communication between HMI and controller it is required to change "Destination existence confirmation starting interval" from the default value of 1200 to 10ms.

e View Help				
) 📽 🖬 🥌				
Timer setting				1
	Setting value		In units	
TCP ULP timer		60	X500ms	
TCP zero window timer		20	X500ms	
TCP resend timer		20	X500ms	
TCP end timer		40	X500ms	
IP assembly timer		10	X500ms	
Response monitoring timer		80	×500ms	
Destination existence confirmation starting interval	10		X500ms	
Destination existence confirmation interval times		20	X500mc	
Destination existence confirmation resend		3	Times	
DNS setting Input format DEC.	Cancel			

In case of communication error, this avoid controller keeps alive the connection for a too long time before to allow a new connection from the HMI.

Click now on "Open settings" as shown at point (3) of Figure below

le View Help			
) 🍃 🖬 🎒			
– Ethernet Mo	dule settings		
	Module 0	•	
0	Operational settings		
2	Initial settings		
3	Open settings		
	Router relay parameter		
	E-mail settings		
Necessary setting(No setti Set if it is needed(No setti Online	ng / Alreadyset)	Check	
Transfer setup	PLC remote operation	Diagnostics	
Write	Read	Verify	
dy			NUM

The next figure shows the "Ethernet open settings" configuration.

The detailed explanation of the meaning of each setting is available in Chapter 5.5 of the Mitsubishi "FX3U-ENET USER'S MANUAL".

"Host station Port No." defined here is the same must be used into Protocol Editor Settings chapter.

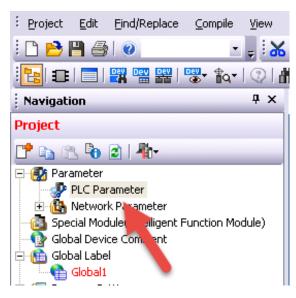
	Protocol	Open system		Fixed buffer	Fixed buffer communication procedure		Pairin; open	-	Existence confirmation	Host station Port No. (DEC.)	Transmission target device IP address	Transmission target device Port No. (DEC.)
-	тср 🔻	Unpassive	•	Send 🔻	. ,	_	Disable	_		5551		
2	-		•	-		-		•		·		
3	-		•	-		•		•		·		
4 5			• •	• •		-		• •		• •		
6		-	•			÷		÷				
7	-		•			÷		-		,		
8	-	-	•	-	-	Ŧ		•				
								_				

i

Note: the usage of more than one panel communicating with the same controller requires to define proper settings in the "Open settings" configuration dialog: one connection per each panel must be configured with proper properties

Controller Settings with GX Works2

The Mitsubishi FX system must be properly configured for Ethernet communication inside GX Works2 programming suite. FX Parameter dialog can be recalled with double-click on PLC Parameter:



Then select "Ethernet Port" tab where is possible to configure IP Address.

Verify the "Communication data code" is set to "Binary code" as shown below:

FX Parameter		
Memory Capacity PLC Name PLC S	ystem(1) PLC System(2) Positioning	Ethernet Port
Channel CH1 💌		
IP Address Setting		Open Setting
	Input Format DEC	
IP Address	192 168 1 250	Time Setting
Subnet Mask Pattern	255 255 255 0	Log Record Setting
Default Router IP Address		
Derault Router IP Address	192 168 1 254	Optional Settings (Default / Changed)
Communication Data Code	7	
Binary Code		
C ASCII Code		
Disable direct connection t		
Do not respond to search	for CPU on network	
Print Window Print Window	Preview Default	Check End Cancel

Then click on "Open Settings" button to recall the "Ethernet Port Open Setting" dialog.

FX Parameter								E
Memory Capacity	PLC N	lame PLC System()	I) PLC System(2)	Position	ing Ethernel	: Port		
Channel	CH1	•						
- IP Addres								a
	thern	et Port Open Se	tting					X
IP Add					Host Station	Destinati	on Destination	1
		Protocol	Open System	n	Port No.	IP Addre		
Subnel	1		MC Protocol	-	102	25		1
	2		MELSOFT Connection			_		1
Defaul	4		MELSOFT Connection					nged)
Communi	Inpul	t decimal value for th	ne Host Station Port I	Vo., Des	tination IP Ad	dress and Dest	ination Port No	
🔎 Bina			End	1	Cancel			
C ASCII	Code							_
Disable	direct	connection to MELS	OFT					
🗖 Do not	respor	nd to search for CPU	on network					
Driet Window		Print Window Preview	. [P-	fault	Chark	End	Cancel
Print Window		mine window Preview	<u>'</u>	De		Check	End	Cancel

"Host station Port No." defined here is the same must be used into Protocol Editor Settings chapter.



Note: For FX3GE Controller, the Open System must be set as "Data Monitor" and Port set to 1025.

Note: the usage of more than one panel communicating with the same controller requires to define proper settings in the "Open settings" configuration dialog: one connection per each panel must be configured with proper properties.

Tag Editor Settings

Into Tag editor select the protocol "Mitsubishi FX ETH" from the list of defined protocols and add a tag using [+] button.

Tag settings can be defined using the following dialog:

Mitsubishi FX ETH	x
Mitsubishi FX ETH	
Resources	Offset Subindex
Input	
Туре	Arraysize Conversion
boolean	▼ 0 +/-
	OK Cancel Apply Help

Element	Description							
Resources	Area of PLC where tag is located							
Offset	Offset address where tag is located.							
SubIndex	This allows resource offset selection within the register.							
Туре	Data Type	Memory Space	Limits					
	boolean	1 bit data	0 1					
	byte	8-bit data	-128 127					
	short	16-bit data	-32768 32767					
	int	32-bit data	-2.1e9 2.1e9					
	unsignedByte	8-bit data	0 255					
	unsignedShort	16-bit data	0 65535					

Element	Description							
	Data Type	Memory Space		Limits				
	unsignedInt	32-bit data		04.2e9				
	float	IEEE single-pred	sision	1.17e-38 3.40e38				
		32-bit floating po	int type					
	string	Refer to "String of	Refer to "String data type chapter"					
		lote: to define arrays, select one of Data Type format followed by rackets like "byte[]", "short[]"…						
Arraysize	In case of		•	e number of array elements e maximum number of byte				
	Note: number of bytes corresponds to number of string characters if Encoding prosent to UTF-8 or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2 bytes.							
Conversion	Conversion to be	applied to the tag.						
	Conversion							
	inv,swap2	Allowed		Configured				
		BCD AB->BA ABCD->CDAB ABCDEFGH->GHEF Inv bits	AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB					
	Depending on data type selected, the list Allowed shows one or more conversion types.							
	Value	Description	ription					
	Inv bits	nv : Invert all the bits of t	nvert all the bits of the tag.					
	Example: $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)							
	Negate	neg: Set the opposite of	tag value.					
		Example:						

Element	Description				
	Value	Description			
		25.36 → - 25.36			
	AB -> BA	swapnibbles: Swap nibbles in a byte.			
		<i>Example:</i> 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)			
	ABCD -> CDAB	swap2 : Swap bytes in a word. <i>Example:</i> $9ACC \rightarrow CC9A$ (in hexadecimal format) $39628 \rightarrow 52378$ (in decimal format)			
	ABCDEFGH	swap4: Swap bytes in a double word.			
	-> GHEFCDAB	<i>Example:</i> 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)			
	ABCNOP - > OPMDAB	swap8 : Swap bytes in a long word. Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) $0\ 10000000110$ 0001110010111011010001011010000111001010			
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)			
	Select conversi	on and click +. The selected item will be added to list Configured .			
	If more convers Configured).	ions are configured, they will be applied in order (from top to bottom of list			
	Lloo the arrow h	wittens to order the configured conversions			

Use the arrow buttons to order the configured conversions.

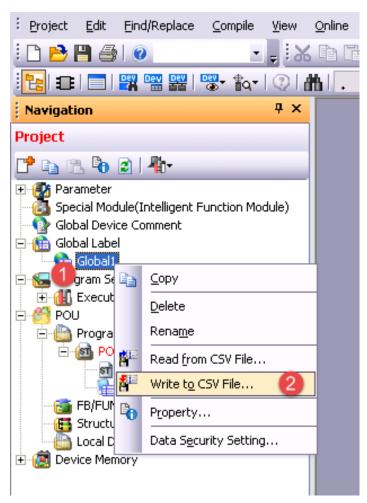
Tag Import

Exporting Tags from PLC

The Mitsubishi FX Ethernet tag import accepts symbol files with extension "csv" created by the Mitsubishi GX Works2 (Not from GX Developer).

The ".csv" file can be exported from the Project tree, as shown in the following figure.

- 1. Right-click on the Global variable list that need to be exported,
- 2. Select "Write to CSV File..."



Into following dialog select the file name and location:

Write to CSV Fi	ile	? 🔀
Save in:	🕼 Desktop 💽 🔶 💼 🕂 🎫 -	
My Recent Documents Desktop My Documents	My Documents My Computer My Network Places	
My Computer		
My Network Places	File name: Global1.csv 💽 👤	Save
FIGUES	Save as type: CSV(tab delimited)(*.csv)	Cancel

Importing Tags in Tag Editor

Select the driver in Tag Editor and click on the Import Tags button to start the importer.

/	Tags	×								
+	-	z	ß	ß	>]	₽	A 9B	B>	63	1
Data			^		-	Та	g URI			_

The following dialog shows which importer type can be selected.

ŀ	IMIStudio	×
	Multiple tag impo	orters are available for this protocol. Please select the importer type and continue.
	Version	Туре
	GX Works2 v1.0	Linear
	Tag Editor exported xm	l General
		OK Cancel

Importer	Description		
GX Works2 v1.0 Linear	Requires a .csv file. All variables will be displayed at the same level.		
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.		

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags × Protocols				•
+ - 👗 🔞 🖉 >] [> 🗞 🕅 🕅	R 🔎 - Search 🍸 Filter by: Data	▼ Ite	ems used:6/10000 Protocol: Sho	w all 🛛 🗹 Show all tags 🔅 🗌
Data Type	Comment	^	Property	Value
Modbus TCP:prot1			✓ Driver	
Model: Modicon Modbus(1-based)			Model	Modicon Modbus(1-based)
Holding Registers 1 unsignedShort			Protocol	Modbus TCP:prot1
Holding Registers 2 unsignedShort			✓ Dictionary	
Holding Registers 3 unsignedShort				6.1
- MRTU1 unsignedShort			Array	false
MRTU2 unsignedShort			Array size	0
MRTU3 unsignedShort			Arrayindex.Subindex	400003
MRTU4 unsignedShort			Comment	
MRTU5 unsignedShort			Data type	unsignedShort

Toolbar item	Description
	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
₩.	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.
	Example of both checked and unchecked result:

Toolbar item	Description
	Heat* Heat* <th< th=""></th<>
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Communication Status

The current communication status can be displayed using the dedicated system variables. Please refer to the User Manual for further information about available system variables and their use.

Error	Notes
NAK	Returned in case the controller replies with a not acknowledge
Timeout	Returned when a request is not replied within the specified timeout period; ensure the controller is connected and properly configured to get network access
Invalid response	The panel did receive from the controller a response, but its format or its contents is not as expected; ensure the data programmed in the project are consistent with the controller resources
General Error	Error cannot be identified; should never be reported; contact technical support

The codes supported for this communication driver are:

Mitsubishi FX SER

The HMI operator panels can be connected to Mitsubishi FX PLC as the network master using this communication driver.

The protocol has been designed to connect to the programming port of the PLC.

Please note that changes in the communication protocol specifications or PLC hardware may have occurred since this documentation was created. Some changes may eventually affect the functionality of this communication driver. Always test and verify the functionality of your application. To fully support changes in PLC hardware and communication protocols, communication drivers are continuously updated. Always ensure that the latest version of communication driver is used in your application.

Protocol Editor Settings

Add [+] a driver in the Protocol editor and select the protocol called "Mitsubishi FX SER" from the list of available protocols.

bishi FX SER	
	Comm OK
PLC Models	Cancel
FX	
FX0/FX0S	
FX0N	1
FX1N	
FX1S	
FX2N	

Element	Description
PLC Models	The list allows selecting the PLC model you are going to connect to. The selection will influence the data range offset per each data type according to the specific PLC memory resources.
Comm	Gives access to the serial port configuration parameters as shown in the figure below.

Element	Description			
	Comm Param	neter Dialog	×	
			ОК	
	Port	com1	_	
	Baudra	9600		
	Parity	even	-	
	Data bi	7	_	
	Stop bit	1	_	
	Mode	RS-422	—	
Port	Serial port se	election:		
	Port	Series 400	Series 500/600	
	com1	PLC Port	Onboard Serial Port	
	com2	PC/Printer Port	Optional Module on slot #1 or #2	
	com3	Not available	Optional Module on slot #3 or #4	
Baud rate, Parity, Data bits, Stop bits	Communication parameters for serial communication			
Mode	Serial port mode; available options:			
	RS-232,			
	RS-485 (2 w	ires)		
	RS-422 (4 wires)			

Tag Editor Settings

Into Tag editor select the protocol "Mitsubishi FX ETH" from the list of defined protocols and add a tag using [+] button.

Tag settings can be defined using the following dialog:

Mitsubishi FX SER	×	
Mitsubishi FX SER		
Resources	Offset Subindex	
Input		
Туре	Arraysize Conversion	
boolean	▼ 0 I +/-	
	OK Cancel Apply Help	

Element	Description				
Resources	Area of PLC where t	ag is located			
Offset	Offset address wher	e tag is located.			
SubIndex	This allows resource	e offset selection within the regis	ster.		
Туре	Data Type	Memory Space	Limits		
	boolean	1 bit data	01		
	byte	8-bit data	-128 127		
	short	16-bit data	-32768 32767		
	int	32-bit data	-2.1e9 2.1e9		
	unsignedByte	8-bit data	0 255		
	unsignedShort	16-bit data	0 65535		
	unsignedInt	32-bit data	0 4.2e9		
	float	IEEE single-precision	1.17e-38 3.40e38		
		32-bit floating point type			
	string	Refer to "String data type chapter"			

 Arraysize
 • In case of array tag, this property represents the number of array elements.

 • In case of string tag, this property represents the maximum number of bytes available in the string tag.

Element	Description						
	set to UTF-8 or	f bytes corresponds to number of string characters if Encoding property is Latin1 in Tag Editor. perty is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one res 2 bytes.					
Conversion	Conversion to b	be applied to the tag.					
	Conversion						
	inv,swap2	Allowed Configured					
		BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits Cancel OK					
	Depending on data type selected, the list Allowed shows one or more conversion types.						
	Value	Description					
	Inv bits	inv : Invert all the bits of the tag.					
		Example: 1001 \rightarrow 0110 (in binary format) 9 \rightarrow 6 (in decimal format)					
	Negate	neg: Set the opposite of tag value.					
		<i>Example:</i> 25.36 → -25.36					
	AB -> BA	swapnibbles: Swap nibbles in a byte.					
		Example: 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)					
	ABCD ->	swap2: Swap bytes in a word.					
	CDAB	Example: 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)					
	ABCDEFGH	swap4: Swap bytes in a double word.					
	-> GHEFCDAB	<i>Example:</i> 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)					

Element	Description			
	Value	Description		
	ABCNOP -	swap8: Swap bytes in a long word.		
	> OPMDAB	Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) $0\ 10000000110$ 0001110010111011010001011010000111001010		
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format)		
		0001 0111 = 23 0001 = 1 (first nibble) 0111 = 7 (second nibble)		
	Select conversion and click +. The selected item will be added to list Configured .			
	If more convers Configured).	ions are configured, they will be applied in order (from top to bottom of list		
	Use the arrow b	outtons to order the configured conversions.		

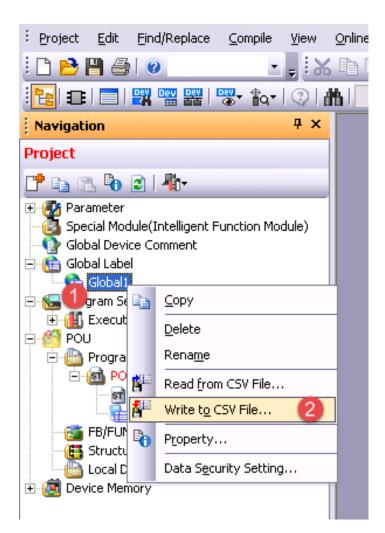
Tag Import

Exporting Tags from PLC

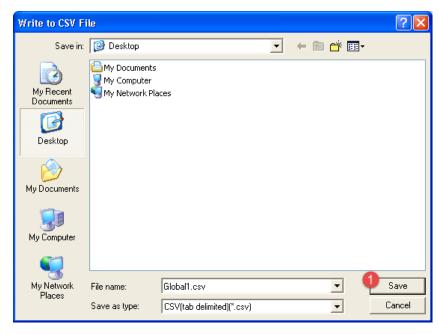
The Mitsubishi FX Serial tag import accepts symbol files with extension "csv" created by the Mitsubishi GX Works2 (Not from GX Developer).

The ".csv" file can be exported from the Project tree, as shown in the following figure.

- 1. Right-click on the Global variable list that need to be exported,
- 2. Select "Write to CSV File..."



Into following dialog select the file name and location:



Importing Tags in Tag Editor

Select the driver in Tag Editor and click on the Import Tags button to start the importer.

	Tags	×								
+	—	Z	D	ß	>]	₽	A 9B	B>	ŧ3	1
Data			^		-	Та	g URI			

The following dialog shows which importer type can be selected.

MIStudio					
Multiple tag importers are available for this protocol. Please select the importer type and continue.					
Version	Туре				
GX Works2 v1.0	Linear				
Tag Editor exported x	ml General				
	OK Cancel				

Importer	Description			
GX Works2 v1.0	Requires a .csv file.			
Linear	All variables will be displayed at the same level.			
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.			
	Tags x			
	+ - 🕺 🕲 🔎 🚺 🚺 🖬 🔂 🛛			
	Data Tag URI			

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags × Protocols					
+ - 🕺 🕲 🖉 🔰 [> \$ ₀ ₪ ∰	R 🔎 - Search 🍸 Filter by: Data	▼ Ite	ms used:6/10000 Protocol: Show a	ll 🛛 🗹 Show all tags 🔅 🗖
ata	Туре	Comment	^	Property	Value
Modbus TCP:prot1	Container			✓ Driver	
Model: Modicon Modbus(1-based)				Model	Modicon Modbus(1-based)
	unsignedShort			Protocol	Modbus TCP:prot1
	unsignedShort			✓ Dictionary	
	unsignedShort unsignedShort			Array	false
	unsignedShort			Array size	0
	unsignedShort			Arravindex.Subindex	400003
	unsignedShort			Comment	
- MRTU5	unsignedShort			Data type	unsignedShort

Toolbar item	Description
ka	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
ta Ka	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.
	Example of both checked and unchecked result:
	Tage: X + - X Type Construct Type Construct Type - Construct Type Construct - Model: Construct Type - Model: Construct Type - Model: Construct Type - Model: Construct Construct - Model: Construct Construct - Nor Nor Nor - Nor Nor Nor - Nor Nor Nor - Nor Nor Nor - Nor Nor Nor <t< th=""></t<>
P - Search	Searches tags in the dictionary basing on filter combo- box item selected.

Communication Status

The current communication status can be displayed using the dedicated system variables. Please refer to the User Manual for further information about available system variables and their use.

Error	Notes
NAK	Returned in case the controller replies with a not acknowledge
Timeout	Returned when a request is not replied within the specified timeout period; ensure the controller is connected and properly configured to get network access
Line Error	Returned when an error on the communication parameter setup is detected (parity, baud

The codes supported for this communication driver are:

Error	Notes
	rate, data bits, stop bits); ensure the communication parameter settings of the controller is compatible with panel communication setup
Invalid response	The panel did receive from the controller a response, but its format or its contents is not as expected; ensure the data programmed in the project are consistent with the controller resources
General Error	Error cannot be identified; should never be reported; contact technical support

Mitsubishi iQ/Q/L ETH

The Mitsubishi iQ/Q/L ETH driver supports communication with Mitsubishi controllers with integrated Ethernet port and with external Ethernet card (QJ71E71-100).

Protocol Editor Settings

Add (+) a driver in the Protocol editor and select the protocol called "Mitsubishi iQ/Q/L ETH" from the list of available protocols.

The driver configuration dialog is shown as in the following figure:

Mitsubishi iQ/Q/L ETH		×
PLC Network		ОК
IP address	0.0.0.0	Cancel
Port	5002	
PLC Models Q00J/Q00/Q01 Q02/Q02H/Q06H/Q12 QnU L02CPU L26CPU-BT Q170M-PLC CPU Q170M-Motion CPU iQ-FX5U iQ-R	2H/Q25H	

Element	Description
IP address	Ethernet IP address of the controller
Port	Specifies the port number (decimal) used in the communication with the PLC.

Element	Description			
PLC Model	The driver supports communication with different Mitsubishi iQ, Q and L controllers. Note: PLC Model selection has only effect on range values of variables. If a particular model is not present in the list, try selecting a similar one. If range values of variables are the same, the communication will be correctly established.			
PLC Network	The protocol allows the connection of multiple controllers to one HMI device. To set-up multiple connections, check "PLC network" checkbox and create your network using the command "Add" per each slave device you need to include in the network.			

Controller Settings

GX Works2

The Mitsubishi Q system must be properly configured for Ethernet communication using the Mitsubishi GX Developer software version 7 or higher, from GX Works2 software.

The Figure below shows an example of network configuration for Ethernet communication.

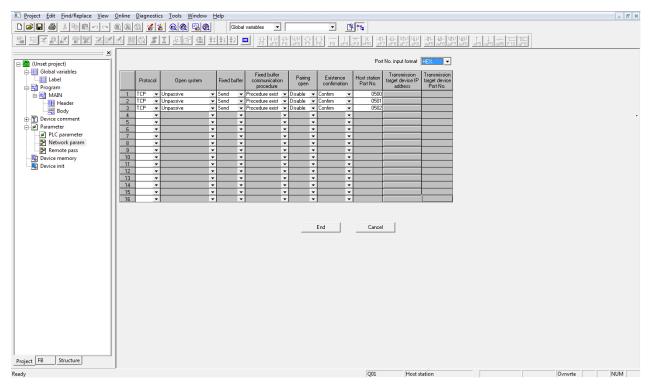
Ethernet operations	×
Communication data code Binary code C ASCII code ASCII code C ASCII	
IP address Input format DEC. IP address 192 168 10 132	Send frame setting • Ethernet(V2.0) • IEEE802.3
Enable Write at RUN time	

Please note that the communication protocol supports only Binary code communication.

The PLC system must be configured to accept incoming data from the external device.

In the GX Developer Software open "Parameters", "Network Param" and select Ethernet/ CC IE/ MELSECNET". Add the number of connections of the operator panels you want to configure in the network.

When using the Mitsubishi CPU with external Ethernet card (QJ71E71-100) the connections have to be configured according to the following figure as "Unpassive":



When the "Existence confirmation" setting has been set to Confirm, the TCP connection will be closed when it is not used (connection lost); by default the TCP port remains open and it is not possible to reconnect.



Note: The GX Developer software allows entering the conventional representation settings (decimal or hexadecimal) for the port number; in the above figure it is in hexadecimal.

In the next figures there are 2 examples about how to set "Initial settings" for 5 and 15 seconds timeout.

<u>F</u> ile <u>V</u> iew <u>H</u> elp					
Timer setting					
Module will operate with default values if setting is left b	lank				
	Setting value	Default value	In units		
TCP ULP timer	5	60	X500ms		
TCP zero window timer	2	20	X500ms		
TCP resend timer	2	20	X500ms		
TCP end timer	3	40	X500ms		
IP assembly timer	1	10	X500ms		
Response monitoring timer	5	60	X500ms		
Destination existence confirmation starting interval	8	1200	X500ms		
Destination existence confirmation interval timer	2	20	X500ms		
Destination existence confirmation resend	1	3	Times		
DNS setting					
Input format DEC.					
IP address of DNS server 1					
IP address of DNS server 2					
IP address of DNS server 3					
IP address of DNS server 4					
End	Cancel				
	Cancer				
l Ready				NUM	
Ready					
<u>F</u> ile <u>V</u> iew <u>H</u> elp					
Timer setting					
	lank				
Module will operate with default values if setting is left b					
	Setting value	Default value	In units		
TCP ULP timer	8	60	X500ms		
TCP zero window timer	4	20	X500ms		
TCP resend timer	4	20	X500ms		
TCP end timer	6	40	X500ms		
IP assembly timer	2	10	X500ms		
Response monitoring timer	8	60	X500ms		
Destination existence confirmation starting interval	20	1200	X500ms		
Destination existence confirmation interval timer	4	20	X500ms		
Destination existence confirmation resend					
	3	<u>э</u>			
	3	3			
DNS setting	3				
	3				
DNS setting	3				
	3				
Input format DEC.					
Input format DEC.					
Input format DEC.					
Input format DEC.					
Input format DEC.					
Input format DEC.	Cancel				
Input format DEC.					
Input format DEC.					
Input format DEC.				NUM	

When using Mitsubishi CPU with integrated Ethernet port the "Open System" settings should be changed to "MC connection"

							Port No. input forma	
	Protocol		Open system		TCP connection	Host station port No.	Transmission target device IP address	Transmission target device port No.
1	TCP	•	MC Protocol	-	•	0500		
2	TCP	•	MC Protocol	-	•	0501		
3	TCP	•	MC Protocol	•	•	0502		
4	TCP	•	MELSOFT connection	-	•			
5	TCP	Ŧ	MELSOFT connection	-	•			
6	TCP	۲	MELSOFT connection	-	-			
7	TCP	•	MELSOFT connection	-	•			
8	TCP	•	MELSOFT connection	-	-			
9	TCP	•	MELSOFT connection	-	-			
10	TCP	•	MELSOFT connection	•	-			
11	TCP	•	MELSOFT connection	-	•			
12	TCP	•	MELSOFT connection	•	-			
13		•	MELSOFT connection	•	-			
14		_	MELSOFT connection	•	-			
15		•	MELSOFT connection	•	•			
16	TCP	•	MELSOFT connection	-	-			
15	TCP	•	MELSOFT connection	╶	•			



Note: The number format for Host Station Port No. is hexadecimal, not decimal.

GX Works3

The Mitsubishi Q system must be properly configured for Ethernet communication using GX Works3 software.

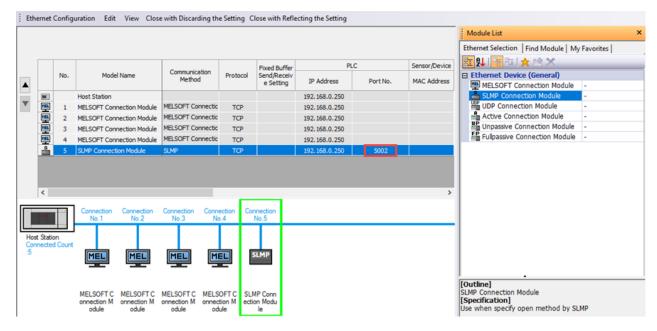
The communication driver is based on SLMP function.

SLMP (Seamless Message Protocol) is a protocol for accessing SLMP-compatible devices from an external device (such as HMI) using TCP or UDP through Ethernet.

From GX Works3 software, Ethernet port parameters must be set from **Module parameter > Ethernet Port > Basic** Settings > Own Node Settings.

Project Edit Find/Replace Convert View C	Online Debug Diagnostics Tool Window Help	
i 🗅 🔁 💾 🎒 👩 💿 💿 💽 🚦 👪 🗈	🔁 🗠 🗠 📴 🖼 🖛 🚚 🖉 👧 👧 🐘 🐘 💭 🦊	🔊 🔜 🖳 🕀 🕀 🦳 🔽 🚽 📰 🖬 🖉 🥝 Max.:
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	🎘 🎉 📅 👔 📮	
Navigation 🛛 🕹 🗸	🐽 ProgPou [PRG] [LD] (Read Only 👘 🐏 Global [Global Label Setting	g] 3 8 B Module Parameter Ethernet Port ×
noject Module Configuration	Contraction (Global Label Setting)	
FB/FUN	Setting Item List	Setting Item
■ ∰ Label ■ ∰ Global Label ∰ H-Global ∰ HHL_Global ■ ∰ Structured Data Types	Input the Setting Item to Search	Item IP Address IP Address IP Address UP Address IP Address IP Address Default Gateway 192.168.0.1
	Cown Node Settings Configuration	Communication Data Code Binary External Device Configuration External Device Configuration Cotailed Setting> Explanation
Module Parameter	tem List Find Result	Check Restore the Default Settings

SLMP Connection Module must be added in **Module parameter > Ethernet Port > Basic Settings > External Device Configuration > Detailed Settings > Ethernet Configuration (Built-in Ethernet Port)**. **Port No.** parameter must be the same as per **Port** parameter from Protocol Editor Settings (see images below).



Mitsubishi iQ/Q/L ETH		×
PLC Network		ОК
IP address	0.0.0.0	Cancel
Port	5002	
PLC Models		
iQ-FX5U		A
iQ-R Q00J/Q00/Q01 Q02/Q02H/Q06H/Q12 QnU	2H/Q25H	E
Q170M-PLC CPU		T

Note: To actually get communication with HMI it is necessary to initialize the PLC after the above settings have been applied.

To initialize the PLC it possibile to use the Run/Stop/Reset switch or by simply rebooting the PLC.

Tag Editor Settings

i

Path: ProjectView> Config > double-click Tags

- 1. To add a tag, click +: a new line is added.
- 2. Select **Mitsubishi iQ/Q/L ETH** from the protocol list: tag definition dialog is displayed.

Mitsubishi iQ/Q/L ETH				×
Mitsubishi iQ/Q/L ETH				
Resources	Offset	Subindex		
Internal Relay	• 0	• 0 •		
Туре	Arraysize	Conversion		
boolean 👻	0		+/-	
	ОК	Cancel	Apply	Help

Element	Description				
Element	DescriptionPLC resources. AvailableInternal RelayError RelayInput (hex)Output (hex)Latch RelayLink RelayData RegisterLink RegisterTimer (Current)Counter (Current)Timer (Switch)Timer (Coil)Counter (Coil)Special RelaySpecial RegisterFile Register	resources are:			
	 Input (oct) Output (oct) 				
Offset	Offset address where tag is located.				
Ulisel	Offset address where tag	is located.			
	Offset address where tag Allows resource offset sel				
SubIndex			Limits		
SubIndex	Allows resource offset sel	ection.	Limits 0 1		
SubIndex Type	Allows resource offset sel Data Type	Memory Space			
SubIndex	Allows resource offset sel Data Type boolean	Memory Space	0 1		
SubIndex	Allows resource offset sel Data Type boolean byte	Memory Space 1-bit data 8-bit data	0 1 -128 127 -32768		
SubIndex	Allows resource offset sel Data Type boolean byte short	Memory Space 1-bit data 8-bit data 16-bit data	0 1 -128 127 -32768 32767		
SubIndex	Allows resource offset sel Data Type boolean byte short int	Memory Space 1-bit data 8-bit data 16-bit data 32-bit data	0 1 -128 127 -32768 32767 -2.1e9 2.1e9 -9.2e18		
SubIndex	Allows resource offset sel Data Type boolean byte short int int64	Memory Space 1-bit data 8-bit data 16-bit data 32-bit data 64-bit data	0 1 -128 127 -32768 32767 -2.1e9 2.1e9 -9.2e18 9.2e18		

Element	Description				
	Data Type	Memory Space	Limits		
	uint64	64-bit data	0 1.8e19		
	float	IEEE single-precision 32-bit floating point type	1.17e-38 3.4e38		
	double	IEEE double-precision 64-bit floating point type	2.2e-308 1.79e308		
	string	Array of elements containing character code defined by selected encoding			
	binary	Arbitrary binary data			
	Note: to define array brackets like "byte[]"	s. select one of Data Type format followe , "short[]"…	ed by square		
Arraysize	 In case of array tag, this property represents the number of array elements. In case of string tag, this property represents the maximum number of bytes available in the string tag. Note: number of bytes corresponds to number of string characters if Encoding property is set to UTF-8 or Latin1 in Tag Editor. 				
	If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2 bytes.				
Conversion	Conversion to be applied to the	tag.			
	Conversion				
		A >CDAB FGH->GHEFCDAB			
	Depending on data type selected	ed, the list Allowed shows one or more o	conversion types.		

lement	Description	
	Value	Description
	Inv bits	inv : Invert all the bits of the tag. <i>Example:</i> $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)
	Negate	neg : Set the opposite of tag value. <i>Example:</i> $25.36 \rightarrow -25.36$
	AB -> BA	swapnibbles: Swap nibbles in a byte. <i>Example:</i> 15D4 → 514D (in hexadecimal format) 5588 → 20813 (in decimal format)
	ABCD -> CDAB	swap2 : Swap bytes in a word. <i>Example:</i> 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)
	ABCDEFGH	swap4: Swap bytes in a double word.
-> GHEFCDAB		<i>Example:</i> 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)
	ABCNOP -	swap8: Swap bytes in a long word.
	> OPMDAB	Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) 0.10000000110 0001110010111011010001011010000111001010
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)

Element	Description
	Select conversion and click +. The selected item will be added to list Configured .
	If more conversions are configured, they will be applied in order (from top to bottom of list Configured).
	Use the arrow buttons to order the configured conversions.

Tag Import

Select the driver in Tag Editor and click on the Import Tags button to start the importer.

	Tags	×								
+	-	z	ß	ß	>]	₽	A 9B	B>	63	1
Data			^		-	Та	g URI			

The following dialog shows which importer type can be selected.

HMIStudio	×
Multiple tag importe	ers are available for this protocol. Please select the importer type and continue.
Version	Туре
GX Works2/GX Works3 v 1.	0 Linear
Tag Editor exported xml	General
	OK Cancel

Importer	Description				
GX Works2/GX Works3 v1.0 Linear	Requires a .csv file. All variables will be displayed at the same level.				
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.				
	+ - 🎽 🕲 🔊 🏷 🖧 🖻 👬 1 Data				

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags × Protocols						
+ - 🗸 🕲 🖉 🔰 🕻	😽 🕼 🚓 🔇	R 🔎 - Search 🍸 Filter by: Data	▼ Ite	ems	used:6/10000 Protocol: Show all	🕑 Show all tags 💮 🗖
Data	Туре	Comment	^	P	roperty	Value
Modbus TCP:prot1	Container			~	Driver	
Model: Modicon Modbus(1-based)					Model	Modicon Modbus(1-based)
	unsignedShort				Protocol	Modbus TCP:prot1
	unsignedShort				Dictionary	
 Holding Registers 3 	unsignedShort			II-		
- MRTU1	unsignedShort				Array	false
- MRTU2	unsignedShort				Array size	0
MRTU3	unsignedShort			IE	Arrayindex.Subindex	400003
- MRTU4	unsignedShort				Comment	
- MRTU5	unsignedShort				Data type	unsignedShort

Toolbar item	Description
ka	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
K ₫	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag. Example of both checked and unchecked result:
	Tags* Tags* + - - - - -
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Communication Status

The current communication status can be displayed using the dedicated system variables. Please refer to the User Manual for further information about available system variables and their use.

The codes supported for this communication driver are:

Error	Notes		
NAK	Returned in case the controller replies with a not acknowledge		
Timeout	Returned when a request is not replied within the specified timeout period; ensure the controller is connected and properly configured to get network access		
Invalid response	The panel did receive from the controller a response, but its format or its contents is not as expected; ensure the data programmed in the project are consistent with the controller resources		
General Error	Error cannot be identified; should never be reported; contact technical support		

Mitsubishi iQ/Q/L SER

The Mitsubishi iQ/Q/L SER driver supports communication with Mitsubishi controllers with integrated serial port.

Protocol Editor Settings

Add (+) a driver in the Protocol editor and select the protocol called "Mitsubishi iQ/Q/L SER" from the list of available protocols.

The driver configuration dialog is shown as in the following figure:

Mitsubishi iQ/Q/L SER			×
	Comm	Ок	
Alias		Cancel	
timeout	500		
PLC Models			
Q02/Q02H/Q06H/Q12	1/Q25H		

Element	Description		
Alias	Name identifying PLC. The name will be added as a prefix to each tag name.		
timeout	Time delay in milliseconds between two retries in case of missing response from the device.		

Element	Description				
PLC	The driver supports communication with different Mitsu	ıbishi iQ, Q and L controllers.			
Model	Note: PLC Model selection has only effect on range values of variables. If a particular model is not present in the list, try selecting a similar one. If range of variables are the same, the communication will be correctly established.				
Comm	If clicked displays the communication parameters setu	p dialog.			
	Comm Parameter Dialog				
	Port com1 Baudrate 9600 • Parity even • Data bits 8 • Stop bits 1 • Mode RS-232 •	DK			
	Element	Parameter			
	Port	Serial port selection.			
		 COM1: On-board port COM2: Optional Plug-in module plugged on slot#1 or slot#2 COM3: Optional Plug-in module plugged on slot#3 or slot#4 			
	Baudrate, Parity, Data Bits, Stop bits	Serial line parameters.			
	Mode	 Serial port mode. Available modes: RS-232. RS-485 (2 wires). RS-422 (4 wires). 			

Tag Editor Settings

Path: ProjectView> Config > double-click Tags

- 1. To add a tag, click +: a new line is added.
- 2. Select Mitsubishi iQ/Q/L SER from the protocol list: tag definition dialog is displayed.

Mitsubishi iQ/Q/L SER	×
Mitsubishi iQ/Q/L SER	
Resources	Offset Subindex
Internal Relay	
Туре	Arraysize Conversion
boolean •	0 I +/-
	OK Cancel Apply Help

Element	Description		
Resources	PLC resources. Available resources. Avail	esources are:	
Offset	Offset address where tag is	located.	
SubIndex	Allows resource offset select	ction.	
Туре	Data Type	Memory Space	Limits
	boolean	1-bit data	0 1
	byte	8-bit data	-128 127
	short	16-bit data	-32768 32767
	int	32-bit data	-2.1e9 2.1e9
	int64	64-bit data	-9.2e18 9.2e18
	unsignedByte	8-bit data	0 255
	unsignedShort	16-bit data	0 65535
	unsignedInt	32-bit data	0 4.2e9

Element	Description			
	Data Type	Memory Space	Limits	
	uint64	64-bit data	0 1.8e19	
	float	IEEE single-precision 32-bit floating point type	1.17e-38 3.4e38	
	double	IEEE double-precision 64-bit floating point type	2.2e-308 1.79e308	
	string	Array of elements containing character selected encoding	code defined by	
	binary	Arbitrary binary data		
	Note: to define array brackets like "byte[]	/s. select one of Data Type format followe ', "short[]"…	ed by square	
Arraysize	 In case of array tag, this property represents the number of arra In case of string tag, this property represents the maximum num available in the string tag. Note: number of bytes corresponds to number of string characters if Elements 			
	set to UTF-8 or Latin1 in Tag E If Encoding property is set to U character requires 2 bytes.	ditor. ICS-2BE, UCS-2LE, UTF-16BE or UTF-1	I6LE one	
Conversion	Conversion to be applied to the	e tag.		
	Conversion			
		AA ->CDAB EFGH->GHEFCDAB ts Cancel OK		
	Depending on data type selec	ted, the list Allowed shows one or more o	conversion types.	

Element	Description	
	Value	Description
	Inv bits	inv : Invert all the bits of the tag.
		<i>Example:</i> 1001 \rightarrow 0110 (in binary format) 9 \rightarrow 6 (in decimal format)
	Negate	neg : Set the opposite of tag value.
		<i>Example:</i> 25.36 → -25.36
	AB -> BA	swapnibbles: Swap nibbles in a byte.
		<i>Example:</i> 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)
	ABCD ->	swap2: Swap bytes in a word.
	CDAB	<i>Example:</i> 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)
	ABCDEFGH	swap4: Swap bytes in a double word.
	-> GHEFCDAB	<i>Example:</i> 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)
	ABCNOP -	swap8: Swap bytes in a long word.
	> OPMDAB	Example: 142.366 → -893553517.588905 (in decimal format) 0 1000000110 0001110010111011010000101101000011100101
		→ 1 10000011100 1010100001010001011011011001011011000010011 1101 (in binary format)
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9)
		Example: $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)

Element	Description
	Select conversion and click +. The selected item will be added to list Configured .
	If more conversions are configured, they will be applied in order (from top to bottom of list Configured).
	Use the arrow buttons to order the configured conversions.

Tag Import

Select the driver in Tag Editor and click on the Import Tags button to start the importer.

	Tags	×								
+	_	X	D	ß	>]	₽	A 9B	B>	ŧ3	1
Data			^		-	Та	g URI			_

The following dialog shows which importer type can be selected.

HMIStudio	
Multiple tag impo	orters are available for this protocol. Please select the importer type and continue.
Version	Туре
Mitsubishi v 1. 1	Linear
Tag Editor exported xm	l General
	OK Cancel

Importer	Description		
Mitsubishi v1.1 Linear	Requires a .csv file generated by GX Works2/GX Works3 software. All variables will be displayed at the same level.		
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.		
	Data Tag URI		

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags × Protocols						
+ - 🗸 🕲 🖉 🤰 🕻	> \$ ∎ ₪ ∰	R 🔎 - Search 🍸 Filter by: Data	▼ Ite	ms used:6/10000 Protocol: Show all	🗸 🗹 Show all tags 🖉 🗔	
Data	Туре	Comment	^	Property	Value	
Modbus TCP:prot1	Container			✓ Driver		
Model: Modicon Modbus(1-based)				Model	Modicon Modbus(1-based)	
	unsignedShort			Protocol	Modbus TCP:prot1	
	unsignedShort			✓ Dictionary		
	unsignedShort				6 las	
- MRTU1	unsignedShort			Array	false	
- MRTU2	unsignedShort			Array size	0	
MRTU3	unsignedShort			Arrayindex.Subindex	400003	
- MRTU4	unsignedShort			Comment		
- MRTU5	unsignedShort			Data type	unsignedShort	

Toolbar item	Description
ka	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
樹	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag. Example of both checked and unchecked result:
	Image: Image:<
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Communication Status

The current communication status can be displayed using the dedicated system variables. Please refer to the User Manual for further information about available system variables and their use.

The codes supported for this communication driver are:

Error	Notes
NAK	Returned in case the controller replies with a not acknowledge
Timeout	Returned when a request is not replied within the specified timeout period; ensure the controller is connected and properly configured to get network access
Invalid response	The panel did receive from the controller a response, but its format or its contents is not as expected; ensure the data programmed in the project are consistent with the controller resources
General Error	Error cannot be identified; should never be reported; contact technical support

NMEA 0183

The NMEA 0183 driver has been developed to communicate with NMEA 0183 compatible devices trough the operator panel serial ports.

Protocol Editor Settings

Add (+) a new driver in the Protocol editor and select the protocol called "NMEA 0183" from the list of available protocols.

The driver configuration dialog is shown in the following figure.

NMEA 0183		×
	Comm OK	
Timeout (sec)	5 Cancel	
Cho Echo		
PLC Models		_
NMEA0 183 sentences Virtual variables		

Element	Description
Timeout (sec)	Defines the time inserted by the protocol between two retries of the same message in case of missing response from the server device. It is expressed in seconds.
Echo	If selected the NMEA messages received on the RX channel of serial port are sent out from the TX channel. This allows to continue the NMEA network downstream of the operator panel whether required.
PLC Models	Two PLC models are available: NMEA 0183 Sentences: when selected the Tags will point univocally to the specified NMEA sentence. Virtual variables: when selected the Tag will show the value coming from any NMEA sentence of the specified type, for example any NMEA sentence of Latitude type.

Tag Editor Settings

Into Tag editor select the protocol "NMEA 0183" from the list of defined protocols and add a tag using [+] button.

Tag settings can be defined using the following dialog:

NMEA 0183		
NMEA 0183		
Variable	Data Type	Arraysize
ALM_01_TotalNumber(👻	float	▼ 0
Conversion		
	+/-	
	ОК	Cancel Apply Help

Element	Description				
Variable	The NMEA Sentence or Virtual variable				
Data Type	Data Type	Memory Space Limits			
	boolean	1 bit data	01		
	byte	8-bit data	-128 127		
	short	-32768 32767			
	int	32-bit data	-2.1e9 2.1e9		
	unsignedByte	0 255			
	unsignedShort	16-bit data	0 65535		
	unsignedInt	32-bit data	0 4.2e9		
	float	IEEE single-precision	1.17e-38 3.40e38		
		32-bit floating point type			
	string	String data			
Arraysize	In case of strin	f array tag, this property represents the number of array elements. f string tag, this property represents the maximum number of bytes in the string tag.			
	Note: number of bytes corresponds to number of string characters if Encoding property is set to UTF-8 or Latin1 in Tag Editor.				

Element	Description				
	If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2 bytes.				
Conversion	Conversion to be applied to the tag.				
	Value Description				
Degrees Shows Degrees data only from coordinates sentence		Shows Degrees data only from coordinates sentence			
Minutes Shows Minutes data only from coordinates sentence		Shows Minutes data only from coordinates sentence			
	Seconds Shows Seconds data only from coordinates sentence				

List of supported NMEA 0183 commands

The NMEA 0183 commands supported from the communication protocol are the following:

- AAM_01_StatusArrivalCircle
- AAM_02_StatusPerpendicular
- AAM_03_ArrivalCircleRadius
- AAM_04_UnitsOfRadius
- AAM_05_WaypointID
- ACK_01_LocalAlarmNumber
- ALM_01_TotalNumberOfMessages
- ALM_02_MessageNumber
- ALM_03_SatelliteNumber
- ALM_04_WeekNumber
- ALM_05_SVhealth
- ALM_06_Eccentricity
- ALM_07_AlmanacReferenceTime
- ALM_08_InclinacionAngle
- ALM_09_RateOfRightAscension
- ALM_10_RootOfSemimajorAxis
- ALM_11_ArgumentOfPerigee
- ALM_12_LongitudeOfAscesionNode
- ALM_13_MeanAnomaly
- ALM_14_ClockParameter0
- ALM_15_ClockParameter1
- ALR_01_TimeOfAlarmConditionChange

ALR_02_LocalAlarmNumber

- ALR_03_AlarmCondition
- ALR_04_AlarmAcknowledgeState
- ALR_05_AlarmDescriptionState
- APB_01_StatusSNR
- APB_02_StatusLock
- APB_03_MagnitudeOfXTE
- APB_04_DirectionToStear
- APB_05_UnitsXTE
- APB_06_StatusArrivalCircle
- APB_07_StatusPerpendicular
- APB_08_BearingOriginToDestination
- APB_09_MagneticOrTrue
- APB_10_DestinatonWaypointID
- APB_11_Bearing
- APB_12_BearingMagneticOrTrue
- APB_13_HeadingToSteer
- APB_14_HeadingMagneticOrTrue
- APB_15_ModeIndicator
- BEC_01_ObservationUTC
- BEC_02_WaypointLatitude
- BEC_03_WaypointLatitudeInd
- BEC_04_WaypointLongitude
- BEC_05_WaypointLongitudeInd
- BEC_06_BearingTrue
- BEC_07_BearingTrueInd
- BEC_08_BearingMagnetic
- BEC_09_BearingMagneticInd)
- BEC_10_Distance
- BEC_11_DistanceUnits
- BEC_12_WaypointID
- BOD_01_BearingTrue
- BOD_02_BearingTrueInd
- BOD_03_BearingMagnetic

BOD_04_BearingMagneticInd BOD_05_DestinationWaypointID BOD_06_OriginWaypointID BWC_01_ObservationUTC BWC_02_WaypointLatitude BWC_03_WaypointLatitudeInd BWC 04 WaypointLongitude BWC_05_WaypointLongitudeInd) BWC_06_BearingTrue BWC_07_BearingTrueInd BWC_08_BearingMagnetic BWC 09 BearingMagneticInd BWC 10 Distance BWC 11 DistanceUnits BWC 12 WaypointID BWC_13_ModeIndicator BWR_01_ObservationUTC BWR_02_WaypointLatitude BWR_03_WaypointLatitudeInd BWR 04 WaypointLongitude BWR 05 WaypointLongitudeInd BWR_06_BearingTrue BWR_07_BearingTrueInd BWR_08_BearingMagnetic BWR 09 BearingMagneticInd BWR_10_Distance BWR_11_DistanceInd BWR 12 WaypointID BWR_13_ModeIndicator BWW 01 BearingTrue BWW_02_BearingTrueInd BWW_03_BearingMagnetic BWW_04_BearingMagneticInd BWW 05 ToWaypointID

BWW_06_FromWaypointID

- DBT_01_WaterDepthFeet
- DBT_02_WaterDepthFeetInd
- DBT_03_WaterDepthMeters
- DBT_04_WaterDepthMetersInd
- DBT_05_WaterDepthFathoms
- DBT_06_WaterDepthFathomsInd
- DCN_01_DeccaChainIdentifier
- DCN_02_RedZoneIdentifier
- DCN_03_RedLineOfPosition
- DCN_04_StatusRedMasterLine
- DCN_05_GreenZoneIdentifier
- DCN_06_GreenLineOfPosition
- DCN_07_StatusGreenMasterLine
- DCN_08_PurpleZoneIdentifier
- DCN_09_PurpleLineOfPosition
- DCN_10_StatusPurpleMasterLine
- DCN_11_RedLineNavigationUse, A=Valid
- DCN_12_GreenLineNavigationUse, A=Valid
- DCN_13_PurpleLineNavigationUse, A=Valid
- DCN_14_PositionUncertainty
- DCN_15_PositionUncertaintyInd
- DCN_16_FixDataBasis
- DPT_01_WaterDepth
- DPT_02_OffsetFromTransducer
- DPT_03_MaximumRangeScale
- DSC_01_FormatSpecifier
- DSC_02_Address
- DSC_03_Cattegory
- DSC_04_NatureOfDistress
- DSC_05_TypeOfCommunication
- DSC_06_PositionOrChannel
- DSC_07_TimeOrTelNo
- DSC_08_ShipMMSI

DSC_09_NatureOfDistress DSC_10_Acknowledgment DSC_11_ExpansionIndicator DSE_01_TotalNumberOfMessages DSE_02_MessageNumber DSE_03_Query_ReplyFlag DSE_04_Vessel_MMSI DSE_05_DataSet1Code DSE_06_Dataset1Data DSE_07_Dataset2Code DSE_08_Dataset2Data DSE_09_Dataset3Code DSE_10_Dataset3Data DSE_11_Dataset4Code DSE 12 Dataset4Data DSE_13_Dataset5Code DSE_14_Dataset5Data DSE_15_Dataset6Code DSE_16_Dataset6Data DSE_17_Dataset7Code DSE_18_Dataset7Data DSE_19_Dataset8Code DSE_20_Dataset8Data DSE_21_Dataset9Code DSE_22_Dataset9Data DSE_23_Dataset10Code DSE_24_Dataset10Data DSI_01_TotalNumberOfMessages DSI_02_MessageNumber DSI_03_Vessel_MMSI DSI_04_VesselCourse DSI_05_VesselType DSI_06_GeographicArea DSI 07 Commandset1Code

- DSI_08_Commandset1Data
- DSI_09_Commandset2Code
- DSI_10_Commandset2Data
- DSI_11_Commandset3Code
- DSI_12_Commandset3Data
- DSI_13_ExpansionIndicator
- DSR_01_TotalNumberOfMessages
- DSR_02_MessageNumber
- DSR_03_Vessel_MMSI
- DSR_04_Dataset1Code
- DSR_05_Dataset1Data
- DSR_06_Dataset2Code
- DSR_07_Dataset2Data
- DSR_08_Dataset3Code
- DSR_09_Dataset3Data
- DSR_10_ExpansionIndicator
- DTM_01_LocalDatumCode
- DTM_02_LocalDatumSubdivisioncode
- DTM_03_LatOffset
- DTM_04_LatOffsetInd
- DTM_05_LonOffset
- DTM_06_LonOffsetInd
- DTM_07_AltitudeOffset
- DTM_08_ReferenceDatumCode
- FSI_01_TransmitingFrequency
- FSI_02_ReceivingFrequency
- FSI_03_ModeOfOperation
- FSI_04_PowerLevel
- GBS_01_UTC
- GBS_02_ExpectedLatitudeError
- $GBS_03_ExpectedLongitudeError$
- GBS_04_ExpectedAltitudeError
- GBS_05_FailedSatelliteID
- GBS_06_ProbabilityOfMissedDetection

GBS_07_EstimateOfBiasMeters GBS_08_StandardDeviationOfBiasEstimate GGA_01_UTC GGA_02_Latitude GGA_03_LatitudeInd GGA_04_Longitude GGA_05_LongitudeInd GGA_06_QualityIndicator GGA_07_NumberOfSatellitesInUse GGA_08_HorizontalDilutionOfPrecision GGA_09_Altitude GGA_10_AltitudeInd GGA_11_GeoidalSeparation GGA_12_GeoidalSeparationInd GGA 13 AgeOfDifferentialData GGA_14_DifferentialReferenceID GLC_01_GRI GLC_02_MasterTOA GLC_03_SignalStatus1 GLC_04_TD1 GLC_05_SignalStatus2 GLC_06_TD2 GLC_07_SignalStatus3 GLC_08_TD3 GLC_09_SignalStatus4 GLC_10_TD4 GLC_11_SignalStatus5 GLC_12_TD5 GLC_13_SignalStatus6 GLL_01_Latitude GLL_02_LatitudeInd GLL_03_Longitude GLL_04_LongitudeInd GLL 05 UTC

- GLL_06_Status
- GLL_07_ModeIndicator
- GNS_01_UTC
- GNS_02_Latitude
- GNS_03_LatitudeInd
- GNS_04_Longitude
- GNS_05_LongitudeInd
- GNS_06_ModeIndicator
- GNS_07_NumberOfSatellitesInUse
- GNS_08_HDOP
- GNS_09_AntennaAltitude
- GNS_10_GeoidalSeparation
- GNS_11_AgeOfDifferentialData
- GNS_12_DifferentialStationID
- GRS_01_UTC
- GRS_02_Mode
- GRS_03_RangeResidual
- GRS_04_RangeResidual
- GRS_05_RangeResidual
- GRS_06_RangeResidual
- GRS_07_RangeResidual
- GRS_08_RangeResidual
- GRS_09_RangeResidual
- GRS_10_RangeResidual
- GRS_11_RangeResidual
- GRS_12_RangeResidual
- GRS_13_RangeResidual
- GRS_14_RangeResidual
- GSA_01_Mode
- GSA_02_Mode
- GSA_03_ID
- GSA_04_ID
- GSA_05_ID
- GSA_06_ID

GSA_07_ID

- GSA_08_ID
- GSA_09_ID
- GSA_10_ID
- GSA_11_ID
- GSA_12_ID
- GSA_13_ID
- GSA_14_ID
- GSA_15_PDOP
- GSA_16_HDOP
- GSA_17_VDOP
- GST_01_UTC
- GST_02_RMSvalueOfStandardDeviation
- GST_03_StandardDeviationOfSemiMajorAxis
- GST_04_StandardDeviationOfSemiMinorAxis
- GST_05_OrientationOfSemiMajorAxis
- GST_06_StandardDeviationOfLatitude
- GST_07_StandardDeviationOfLongitude
- GST_08_StandardDeviationOfAltitude
- GSV_01_NumberOfMessages
- GSV_02_MessageNumber
- GSV_03_NumberOfSatellitesInView
- GSV_04_SET1_SatelliteID
- GSV_05_SET1_Elevation
- GSV_06_SET1_Azimuth
- GSV_07_SET1_SNR
- GSV_08_SET2_SatelliteID
- GSV_09_SET2_Elevation
- GSV_10_SET2_Azimuth
- GSV_11_SET2_SNR
- GSV_12_SET3_SatelliteID
- GSV_13_SET3_Elevation
- $GSV_14_SET3_Azimuth$
- GSV_15_SET3_SNR

GSV_16_SET4_SatelliteID

GSV_17_SET4_Elevation

GSV_18_SET4_Azimuth

GSV_19_SET4_SNR

HDG_01_MagneticHeading

HDG_02_MagneticDeviation

HDG_03_MagneticDeviationInd

HDG_04_MagneticVariation

HDG_05_MagneticVariation

HDM_01_MagneticHeading

HDM_02_MagneticHeadingInd

HDT_01_Heading

HDT_02_HeadingInd

HMR_01_HeadingSensor1ID

HMR_02_HeadingSensor2ID

HMR_03_DifferenceLimit

HMR_04_HeadingSensorDifference

HMR_05_WarningFlag

HMR_06_HeadingReadingSensor1

HMR_07_StatusSensor1

HMR_08_TypeSensor1

HMR_09_DeviationSensor1

HMR_10_DeviationSensor1Ind)

HMR_11_HeadingReadingSensor

HMR_12_StatusSensor2

HMR_13_TypeSensor2

HMR_14_DeviationSensor2

HMR_15_DeviationSensor2Ind)

HMR_16_Variation

HMR_17_VariationInd)

HMS_01_HeadingSensor1ID

HMS_02_HeadingSensor2ID

HMS_03_MaximumDifference

HSC_01_CommandedHeading

414

- HSC_02_CommandedHeadingInd
- HSC_03_CommandedHeadingMagnetic
- HSC_04_CommandedHeadingMagneticInd
- HTC_01_Override
- HTC_02_CommandedRudderAngle
- HTC_03_CommandedRudderDirection
- HTC_04_SelectedSteeringMmode
- HTC_05_TurnMode
- HTC_06_CommandedRudderLimit
- HTC_07_CommandedOffHeadingLimit
- HTC_08_CommandedRadiusOfTurn
- HTC_09_CommandedRateOfTurn
- HTC_10_CommandedHeadingToSteer
- HTC_11_CommandedOffTrackLimit
- HTC_12_CommandedTrack
- HTC_13_HeadingReferenceInUse
- HTD_01_Override
- HTD_02_CommandedRudderAngle
- HTD_03_CommandedRudderDirection
- HTD_04_SelectedSteeringMode
- HTD_05_TurnMode
- HTD_06_CommandedRudderLimit
- HTD_07_CommandedOffHeadingLimit
- HTD_08_CommandedRadiusOfTurn
- HTD_09_CommandedRateOfTurn
- HTD_10_CommandedHeadingToSteer
- HTD_11_CommandedOffTrackLimit
- HTD_12_CommandedTrack
- HTD_13_HeadingReferenceInUse
- HTD_14_RudderStatus
- HTD_15_OffHeadingStatus
- HTD_16_OffTrackstatus
- HTD_17_VesselHeading
- LCD_01_GRI

- LCD_02_MasterSNR
- LCD_03_MasterECD
- LCD_04_Secondary1_SNR
- LCD_05_Secondary1_ECD
- LCD_06_Secondary2_SNR
- LCD_07_Secondary2_ECD
- LCD_08_Secondary3_SNR
- LCD_09_Secondary3_ECD
- LCD_10_Secondary4_SNR
- LCD_11_Secondary4_ECD
- LCD_12_Secondary5_SNR
- LCD_13_Secondary5_ECD
- MDA_01_BarometricPressureInchesOfMercury
- MDA_02_BarometricPressureInchesOfMercuryInd
- MDA_03_Barometric pressureBars
- MDA_04_Barometric pressureBarsInd
- MDA_05_AirTemperature
- MDA_06_AirTemperatureInd
- MDA_07_WaterTemperature
- MDA_08_WaterTemperatureInd
- MDA_09_RelativeHumidity
- MDA_10_AbsoluteHumidity
- MDA_11_DewPoint
- MDA_12_DewPointInd
- MDA_13_WindDirectionTrue
- MDA_14_WindDirectionTrueInd
- MDA_15_WindDirectionMagnetic
- MDA_16_WindDirectionMagneticInd
- MDA_17_WindSpeedKnots
- MDA_18_WindSpeedKnotsInd
- MDA_19_WindSpeedMs
- MDA_20_WindSpeedMsInd
- MLA_01_TotalNumberOfMessages
- MLA_02_MessageNumber

MLA_03_SatelliteID

MLA_04_CalendarDay

MLA_05_GeneralizedHealth

MLA_06_Eccentricity

MLA_07_DOT

MLA_08_ArgumentOfPerigee

MLA_09_SystemTimeScaleCorrectionMSB

MLA_10_CorrectionOfAverageValueDraconitic

MLA_11_TimeOfAscensionNode

MLA_12_GreenwichLongitude

MLA_13_CorrectionToAverageValueInclination

MLA_14_SystemTimeScaleCorrectionLSB

MLA_15_CourseValueOfTimeScaleShift

MSK_01_BeaconFrequency

MSK_02_Auto_Manual_Frequency

MSK_03_BeaconBitRate

MSK_04_Auto_Manual_BitRate

MSK_05_IntervalForSending

MSK_06_ChannelNumber

MSS_01_SignalStrength

MSS_02_SNR

MSS_03_BeaconFrequency

MSS_04_BeaconBitRate

MSS_05_ChannelNumber

MTW_01_Temperature

MTW_02_TemperatureInd

MWD_01_WindDirection

MWD_02_WindDirectionInd

MWD_03_WindDirectionMagnetic

MWD_04_WindDirectionMagneticInd

MWD_05_WindSpeedKnots

MWD_06_WindSpeedKnotsInd

 $\mathsf{MWD_07_WindSpeedMs}$

MWD_08_WindSpeedMsInd

MWV_01_WindAngle

MWV_02_Reference

MWV_03_WindSpeed

MWV_04_WindSpeedInd

MWV_05_Status

NMEA_Altitude

NMEA_Course

NMEA_Latitude

NMEA_LatitudeInd

NMEA_Longitude

NMEA_LongitudeInd

NMEA_SpeedKnots

NMEA_UTC

OSD_01_Heading

OSD_02_HeadingStatus

OSD_03_VesselCourse

OSD_04_CourseReference

OSD_05_VesselSpeed

OSD_06_SpeedReference

OSD_07_VesselSet

OSD_08_VesselDrift

OSD_09_SpeedUnits

RMA_01_Status

RMA_02_Latitude

RMA_03_LatitudeInd

RMA_04_Longitude

RMA_05_LongitudeInd

RMA_06_TimeDifferenceA

RMA_07_TimeDifferenceB

RMA_08_SpeedOverGroundKnots

RMA_09_CourseOverGround

RMA_10_MagneticVariation

RMA_11_MagneticVariationInd

RMA_12_ModeIndicator

RMB_01_DataStatus

RMB_02_CrossTrackError

RMB_03_DirectionToSteer

RMB_04_OriginWaypointID

RMB_05_DestinationwaypointID

RMB_06_DestinationwaypointLat

RMB_07_DestinationwaypointLatInd

RMB_08_DestinationWaypointLongitude

RMB_09_DestinationWaypointLongitudeInd

RMB_10_RangeToDestination

RMB_11_BearingToDestination

RMB_12_DestinationClosingVelocity

RMB_13_ArrivalStatus

RMB_14_ModeIndicator

RMC_01_UTC

RMC_02_Status

RMC_03_Latitude

RMC_04_LatitudeInd

RMC_05_Longitude

RMC_06_LongitudeInd

RMC_07_SpeedOverGround

RMC_08_CourseOverGround

RMC_09_Date

RMC_10_MagneticVariation

RMC_11_MagneticVariationInd

RMC_12_ModeIndicator

ROT_01_RateOfTurn

ROT_02_Status

RPM_01_SourceShaftEngine

RPM_02_EngineOfShaftNumber

RPM_03_Speed

RPM_04_PropellerPitch

RPM_05_Status

RSA_01_StarboardRudderSensor

RSA_02_StatusRudderSensor)

- RSA_03_PortRudderSensor
- RSA_04_StatusPortRudderSensor)
- RSD_01_Origin1Range
- RSD_02_Origin1Bearing
- RSD_03_VariableRangeMarker1
- RSD_04_BearingLine1
- RSD_05_Origin2Range
- RSD_06_Origin2Bearing
- RSD_07_VRM2
- RSD_08_EBL2
- RSD_09_CursorRange
- RSD_10_CursorBearing
- RSD_11_RangeScale
- RSD_12_RangeScaleUnits
- RSD_13_DisplayRotation
- RTE_01_TotalNumberOfMessages
- RTE_02_MessageNumber
- RTE_03_MessageMode
- RTE_04_RouteIdentifier
- RTE_05_WaypointIdentifier1
- RTE_06_WaypointIdentifier2
- RTE_07_WaypointIdentifier3
- RTE_08_WaypointIdentifier4
- RTE_09_WaypointIdentifier5
- RTE_10_WaypointIdentifier6
- RTE_11_WaypointIdentifier7
- RTE_12_WaypointIdentifier8
- RTE_13_WaypointIdentifier9
- RTE_14_WaypointIdentifier10
- SFI_01_TotalNumberOfMessages
- SFI_02_MessageNumber
- SFI_03_1stFrequency
- SFI_04_1stMode

SFI_05_2ndFrequency SFI_06_2ndMode SFI_07_3rdFrequency SFI_08_3rdMode SFI_09_4thFrequency SFI_10_4thMode SFI_11_5thFrequency SFI_12_5thMode SFI_13_6thFrequency SFI_14_6thMode STN_01_TalkerID TLB_01_TargetNumber TLB_02_LabelAssigned TLB 03 TargetNumber1 TLB 04 LabelAssigned1 TLB_05_TargetNumber2 TLB_06_LabelAssigned2 TLB_07_TargetNumber3 TLB_08_LabelAssigned3 TLB 09 TargetNumber4 TLB 10 LabelAssigned4 TLB_11_TargetNumber5 TLB_12_Labelassigned5 TLB_13_TargetNumber6 TLB 14 LabelAssigned6 TLB 15 TargetNumber7 TLB_16_LabelAassigned7 TLB_17_TargetNumber8 TLB_18_LabelAssigned8 TLB_19_TargetNumberReported TLB_20_TargetLabelAssigned TLL_01_TargetNumber TLL_02_TargetLatitude TLL 03 TargetLatitudeInd

TLL_04_TargetLongitude

TLL_05_TargetLongitudeInd

TLL_06_TargetName

TLL_07_UTC

TLL_08_TargetStatus

TLL_09_ReferenceTarget

TTM_01_TargetNumber

TTM_02_TargetDistance

TTM_03_Bearing

TTM_04_BearingInd

TTM_05_TargetSpeed

TTM_06_TargetCourse

TTM_07_TargetCourseInd

TTM_08_DistanceOfClosestPoint

TTM_09_TimeToCPA

TTM_10_SpeedAndDistanceUnits

TTM_11_TargetName

TTM_12_TargetStatus

TTM_13_ReferenceTarget

TTM_14_UTC

TTM_15_TypeOfAcquisition

TXT_01_TotalNumberOfMessages

TXT_02_MessageNumber

TXT_03_TextIdentifier

TXT_04_TextMessage

VBW_01_LongitudinalWaterSpeed

VBW_02_TransverseWaterSpeed

VBW_03_StatusWaterSpeed

VBW_04_LongitudinalGroundSpeed

VBW_05_TransverseGroundSpeed

VBW_06_StatusGroundSpeed

VBW_07_SternTransverseWaterSpeed

VBW_08_StatusSternWaterSpeed

VBW_09_SternTransverseGroundSpeed

VBW_10_StatusSternGroundSpeed

VDR_01_Direction

VDR_02_DirectionInd

VDR_03_DirectionMagnetic

VDR_04_DirectionMagneticInd

VDR_05_CurrentSpeed

VDR_06_CurrentspeedInd

VHW_01_Heading

VHW_02_HeadingInd

VHW_03_HeadingMagnetic

VHW_04_HeadingMagneticInd

VHW_05_SpeedKnots

VHW_06_SpeedKnotsInd

VHW_07_SpeedKmh

VHW_08_SpeedKmhInd

VLW_01_TotalCumulativeDistance

VLW_02_TotalCumulativeDistanceInd

VLW_03_DistanceSinceReset

VLW_04_DistanceSinceResetInd

VPW_01_SpeedKnots

VPW_02_SpeedKnotsInd)

VPW_03_SpeedMs

VPW_04_SpeedMsInd

VTG_01_CourseOverGround

VTG_02_CourseOverGroundInd

VTG_03_CourseOverGroundMagnetic

VTG_04_CourseOverGroundMagneticInd

VTG_05_SpeedOverGroundKnots

VTG_06_SpeedOverGroundKnotsInd

VTG_07_SpeedOverGroundKmh

VTG_08_SpeedOverGroundKmhInd

VTG_09_ModeIndicator

VWR_01_MeasuredWindAngle

VWR_02_VesselHeading

VWR_03_MeasuredWindSpeed

- VWR_04_MeasuredWindSpeedInd
- VWR_05_WindSpeedMeters
- VWR_06_WindSpeedMetersInd
- VWR_07_WindSpeedKmh
- VWR_08_WindSpeedKmhInd
- VWT_01_CalculatedWindAngle
- VWT_02_VesselHeading
- VWT_03_CalculatedWindSpeed
- VWT_04_CalculatedWindSpeedInd
- VWT_05_WindSpeedMeters
- VWT_06_WindSpeedMetersInd
- VWT_07_WindSpeedKmh
- VWT_08_WindSpeedKmhInd
- WCV_01_VelocityComponent
- WCV_02_VelocityComponentInd
- WCV_03_WaypointIdentifier
- WCV_04_ModeIndicator
- WNC_01_DistanceMiles
- WNC_02_DistanceMilesInd
- WNC_03_DistanceKm
- WNC_04_DisttanceKmInd
- WNC_05_WaypointIdentifierFrom
- WNC_06_WaypointIdentifierTo
- WPL_01_WaypointLatitude
- WPL_02_WaypointLatitudeInd
- WPL_03_WaypointLongitude
- WPL_04_WaypointLongitudeInd
- WPL_05_WaypointIdentifier
- XDR_01_Transducer1Type
- XDR_02_Measurmnt1Data
- XDR_03_UnitsOfMeasure1
- XDR_04_Transducer1
- XDR_05_Transducer2Type

XDR_06_Measurment2Data XDR_07_UnitsOfMeasure2 XDR_08_Transducer2 XDR_09_Transducer3Type XDR_10_Measurment3Data XDR_11_UnitsOfMeasure3 XDR_12_Transducer3 XDR_13_Transducer4Type XDR_14_Measurment4Data XDR_15_UnitsOfMeasure4 XDR_16_Transducer4 XDR_17_Transducer5Type XDR_18_Measurment5Data XDR_19_UnitsOfMeasure5 XDR 20 Transducer5 XDR_21_Transducer6Type XDR_22_Measurment6Data XDR_23_UnitsOfMeasure6 XDR_24_Transducer6 XDR 25 Transducer7Type XDR 26 Measurment7Data XDR_27_UnitsOfMeasure7 XDR_28_Transducer7 XDR_29_Transducer8Type XDR_30_Measurment8Data XDR_31_UnitsOfMeasure8 XDR_32_Transducer8 XTE_01_Status1 XTE_02_Status2 XTE_03_MagnitudeOfCrossTrackError XTE_04_DirectionToSteer XTE_05_Units XTE_06_ModeIndicator XTR 01 MagnitudeOfCrossTrackError XTR_02_DirectionToSteer

XTR_03_Units

ZDA_01_UTC

ZDA_02_Day

ZDA_03_Month

ZDA_04_Year

ZDA_05_LocalZoneHours

ZDA_06_LocalZoneMinutes

ZDL_01_TimeToPoint

ZDL_02_DistanceToPoint

ZDL_03_TypeOfPoint

ZFO_01_UTC

ZFO_02_ElapsedTime

ZFO_03_OriginWaypointID

ZTG_01_UTC

ZTG_02_TimeToGo

ZTG_03_DestinationWaypointID

Omron FINS ETH

This driver supports the FINS protocol via Ethernet connection. For a list of models that support the FINS Communications Service, refer to the manufacturer's website.

Protocol Editor Settings

Omron FINS ETH		×
PLC Network		ОК
Alias		Cancel
IP address	192 . 168 . 2 . 18	
port	9600	
panel network	0	
panel node	0	
panel unit	0	
network	0	
node	18	
unit	0	
PLC Models CJx/CS1x/CP1x		

Element	Description
Alias	Name to be used to identify nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node
IP address	The Ethernet IP address of the controller connected to the operator panel
Port	Defines the port number used in the communication with the PLC. The UDP Port number

Element	Description
	must match the value specified in the PLC configuration; the default value is 9600. Most applications will use the default value.
Network Node Unit	 Parameters that define the FINS address of the device. There is a conversion rule to determine the IP address of a device starting from the FINS address in the Omron network. When using the FINS communication service, it is necessary to specify the node addressing according to the FINS addressing scheme. Even in this case, data must be sent and received on the Ethernet network using IP addresses. Therefore, IP addresses are converted from FINS addresses. There are three ways to convert the FINS addresses into the corresponding IP address; they are: Automatic generation (default) IP address table Combined method (uses Automatic and IP address table) The Omron documentation contains all the details related to determine the IP address of the controller depending on the FINS address assigned to it. The next chapter shows an example of controller configuration based on IP address table.
Panel Network Panel Node Panel Unit	 The Panel Network/Node/Unit parameters assigned to HMI should be compatible with the ones assigned in the Omron network to the PLC: Network Number must match the one specified for the PLC Node Number should match the last number of the IP address of the HMI; in the figure above the panel has been configured with IP address 192.168.2.15. Unit represent the possible different network cards over the same node; for the HMI should be always set to zero since there is always only one communication unit.

The protocol supports the connections to multiple controllers.

To enable this, check the "PLC Network" check box and provide the configuration per each node.

Omron FINS ETH			22	
V PLC Network			ОК	
Alias			Cancel	
IP address	0	mron FINS ETH		x
port	9600			OK
panel network	0	Alias		Cancel
panel node	0	IP address	192 . 168 . 2 . 18	
panel unit	0	port	9600	
network	0	network	0	
node	0	node	18	
unit	0	unit	0	
PLC Models CJx/CS1x/CP1x		PLC Models		
GX/G31X/C+1X				
Slaves		Add	elete Modify	
Slave Id	Model	Alias		

Controller Settings

PLC must be properly configured to handle the communication with HMI.

Below an example of configuration based on a real scenario.



Configuration windows in this chapter are depending on PLC model. Following lines must be used as guidelines for any specific configuration.

Example Setup

HMI IP address = 192.168.2.16

PLC IP address = 192.168.2.18

In Ethernet configuration Tab:

1. Make sure that last number of IP address is the same of FINS Node No.

PLC Settings - CP1HCP1LModbusSinglePort			×
File Options Help			
Pulse Output 0 Pulse Output 1 Inverter Positionin	g 0 Inverter Positioning 1	Built-in Ethernet	• •
IP Address 192 . 168 . 2 . 18 Sub-net Mask 255 . 255 . 255 . 0	IP Router Table	Ins Del	
FINS Node No. Node 18	Broadcast	C All 0 (4.2BSD)	
TCP/IP keep-alive 0 min [0: Default(120)]			
FINS/TCP Setting FINS/UDP Setting	DNS Setting	Clock Auto Adjustment	
		CP1L-E EM	Monitor

In FINS/UDP Setting

- 2. Set Conversion to "IP address table"
- 3. Set FINS/UDP Options to "Destination IP is changed dynamically"
- 4. Insert HMI IP address

PLC Settings - CP1HCP1LI File Options Help	ModbusSinglePort		×
Pulse Output 0 Pulse C FI	FINS/UDP Port	FINS/UDP Option © Destination IP is changed dynamically	•
IP Address 19	C User defined	C Destination IP is Not changed dynamically	
Sub-net Mask 25 FINS Node No. Node 18 TCP/IP keep-alive	Conversion C Auto (dynamic) C Auto (Static) C Combined IP address table	IP Address Table	
FINS/TCP Setting	FINS/UDP Setting D	NS Setting Clock Auto Adjustment	
		CP1L-E EM Moni	tor

0

IP Address Table can contain more than one address. In these cases make sure that index of IP addresses is consecutive: 001 192.168.002.016

002 192.168.002.017 003 192.168.002.033



Add PC IP address in IP Address Table described above to allow communication between PLC and online Simulation.

In protocol editor

- 5. Set the IP address of PLC
- 6. Insert last number of HMI IP address in panel node parameter
- 7. Insert last number of PLC IP address in node parameter

Omron FINS ETH		×
PLC Network	ОК	
Alias	Cancel	
IP address	196 . 168 . 2 . 18	
port	9600	
panel network	0	
panel node	16 6	
panel unit	0	
network	0	
node	18 7	
unit	0	
PLC Models		
CJx/CS1x/CP1x		

Tag Import

Exporting Tags from PLC

The Omron FINS Ethernet driver can import tag information from CX-Programmer PLC programming software. The tag import filter accepts symbol files with extension ".cxr" created by the Omron programming tool.

The ".cxr" files can be exported from the symbol table utility.

See in figure how to access the Symbol Table (if configured) from the Omron programming software.

File Edit View Insert PLC Program Simulation	Tools Window Help			
📗 🗅 🚅 🖬 🙀 🚑 🖪 🗍 % 🖻 🛍 🔍 2	2 🗠 🖊 🚟 🕵 🕼	🤋 📢 🗌 🛆 🚴	🍇 🛛 🗸 🗍 💑 🔢	D. D.C.
」 < < < < < < < < < < < < <	[□ - - - - - - - - - - 		₿ 元 L ¥ 🗍	💀 l 🅸 🛗 🛙 🗉
🖪 🗖 🖗 🖓 🖓 😭 🖀 13	88 16 11 11 18] 🗷 🗐 📑 🖓	S ► ■ ■ 1	▲ 밤 ♥ >
	Name	Data Type	Address / Value	Rack Location Use
E 💑 NewProject	- MyData_01	WORD	DO	W
E	- MyData_02	WORD	D2	W
Symbols	- MyData_03	WORD	D3	W
Memory ⊡ ∰ Programs ⊡ ∰ NewProgram1 (00)				
1 ∰ Symb G Sectio				
🕞 🗊 Sectic 🔁 Insert Symbol				
CPU010 😡 Yalidate Symbols				
≞ Large Icons				
°⊡- S <u>m</u> all Icons				
a-a- List				
TITT Details				
X Cu <u>t</u>				
B Copy				
Paste				
Delete				
Project /				
Reusable <u>Fi</u> le	Add to Project			
NewPLC1/NewProgram1/	3 <u>S</u> ave As			
	-			
Hide				
<u>E</u> loat In Main Window				
Properties				
Compile Find Report Transfer				
Save as a reusable file		NewPLC1(Net:0,N	ode:0) - Offline	

Importing Tags in Tag Editor

Select the driver in Tag Editor and click on the Import Tags button to start the importer.



The following dialog shows which importer type can be selected.

HMIStudio		×
Multiple tag imp	orters are available for this protocol. Please select the importer type and	continue.
Version	Туре	
CX-Programmer v1.1	Linear	
Tag Editor exported xn	l General	
	ОК С	ancel

Importer	Description			
CX-Programmer v1.1 Linear	Requires a .cxr file. All variables will be displayed at the same level.			
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.			
	+ - ≱ ⓒ ▷ >] ▷ ♣ ▷] [3] ↓ Data			

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

+ - 🎽 🕲 🖄 🕨	€ Β Β3 Β3 β	R 🔎 Search Trilter by: Data	▼ Iter	ms used:6/10000 Protocol: Show all	🗹 Show all tags 🖉 🗐
Data Typ	/pe	Comment	^	Property	Value
Modbus TCP:prot1	ontainer			Y Driver	
Model: Modicon Modbus(1-based)				Model	Modicon Modbus(1-based)
	nsignedShort			Protocol	Modbus TCP:prot1
	nsignedShort			✓ Dictionary	
	nsignedShort			Array	false
	nsignedShort				
MRTU2 uns	nsignedShort			Array size	0
MRTU3 uns	nsignedShort			Arrayindex.Subindex	400003
MRTU4 uns	nsignedShort			Comment	
MRTU5 uns	nsignedShort			Data type	unsignedShort

Toolbar item	Description		
ka	Import Tag(s).		
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project		
「「「」	Update Tag(s).		
	Click on this icon to update the tags in the project, due a new dictionary import.		
R	Check this box to import all sub-elements of a tag.		
_	Example of both checked and unchecked result:		
	Tags: Image: Image: </th		
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.		

Override variables

The protocol provides the special data types to override the following protocol settings:

Omron FINS ETH		×
PLC Network		ОК
Alias		Cancel
IP address	0.0.0.0	1
port	9600	2
panel network	0	3
panel node	0	4
panel unit	0	6
network	0	6
node	0	7
unit	0	8
Timeout	1000	-
PLC Models		
CJx/CS1x/CP1x		

Tags can be created by manually add them from Tag Editor

Omron FINS ETH			X
Omron FINS ETH			
Memory Type	Offset	Subindex	
I/O area Auxilary area Holding area Timer completion flags Timer PVs DM area Counter completion flags Counter CVs EM area Work area Index registers Data registers Data registers Data registers Override IP Override Port Override Node Override Node Override VanelNet Override PanelNet Override PanelNode Override PanelUnit	Type unsignedSho	Arraysize rt 0	
	ОК	Cancel	Apply Help

Tag Name	Description
Override IP	Permits to override "IP address" property (1) in runtime.
	Data type: array unsigned bytes.
	Notes:
	- when address is set as "0.0.0.0" communication with the controller is stopped, no request frames are generated anymore.
	- when address is different than "0.0.0.0" it is interpreted as a real IP address to override and target PLC IP address is replaced in runtime with the new value.
Override	Permits to override "port" property (2) in runtime.
Port	Data type: unsignedShort.
Override	Permits to override "network" property (6) in runtime.
Network	Data type: unsignedByte.
Override	Permits to override "node" property (7) in runtime.
Node	Data type: unsignedByte.

Tag Name	Description
Override	Permits to override "unit" property (8) in runtime.
Unit	Data type: unsignedByte.
Override	Permits to override "panel network" property (3) in runtime.
PanelNet	Data type: unsignedByte.
Override	Permits to override "panel network" property (4) in runtime.
PanelNode	Data type: unsignedByte.
Override	Permits to override "panel unit" property (5) in runtime.
PanelUnit	Data type: unsignedByte.



Note: Override Tags are initialized with the value of properties specified in Protocol Editor. Override values assigned at runtime are retained through power cycles.

Aliasing Tag Names in Network Configurations

Tag names must be unique at project level; it often happens that the same tag names are to be used for different controller nodes (for example when the HMI is connected to two devices that are running the same application). Since tags include also the identification of the node and Tag Editor does not support duplicate tag names, the import facility in Tag Editor has an aliasing feature that can automatically add a prefix to imported tags. With this feature tag names can be done unique at project level.

The feature works when importing tags for a specific protocol. Each tag name will be prefixed with the string specified by the "Alias". As shown in the figure below, the connection to a certain controller is assigned the name "Node1". When tags are imported for this node, all tag names will have the prefix "Node1" making each of them unique at the network/project level.

Name Group Driver Address Comment Name Group Driver Address Comment Name Address Comment It is an any efforted Node1/Water_level It is an any efforted It is an any efforted Node1/Water_level It is an any efforted It is an any efforted Node1/Water_level It is an any efforted It is an any efforted Node1/Water_level It is an any efforted It is an any efforted Node1/Water_level Node id as defined in import file Node1 Select Network node id Node1 Node1 Nodez Node2 It is an any efforted Node2	F - ^ V 🔏 🛙	b 📖 🔰	Omron FIN	IS ETH:prot1	• 6	ş 🔜	
Add Hudina TCP peril 1 20 anegred line Note 1/Water Jevel Add Hudina TCP peril 2 20 anegred line Add Hudina TCP per			Group			Address	Comment
Add Heating T2 gent of the second sec	Node's Gal_hodrolia			ABE Wollow TOP	poli 1	11 Dunsigned/Shot	
Node 1/Water_level Network Node 1/Water_level 1 100 unsigned Short	Node1/Cala_hodrotia			ABB Worldson TCP	peak? 1	12 Dunsigned Shot	
Node 1/Water_level Node 1/Water_level Network Network Network node id Slave Id Model Node 1 Node 1 Node 2 Ok Cancel	Node 1/IN W/X7EP Jack	al l		ABB Worldson TCP	pest 1 1	(D.D.unsigned/DhoH	
Node 1/Water_level 1 10 0 unsigned Short Network Node id as defined in import file Select Network node id Slave Id Mode1 Node1 Node1 Node2 Ok Cancel	Note: CHOGEDHOW			ABE Worldow TCP	pest 1	245 Dunkigned Short	
Node 1/Water_level 1 10 0 unsign edShort	Note1/OUT_BRT_Note	nalis.		ABE Wollow TOP			
Node 1/Water_level Network Node id as defined in import file Select Network node id Slave Id Model Node 1 Node 1 Node 2 Ok Cancel	Node1/FL DATA_headre	fia		ABE Worldson TCP	pest 1 1	2 Dumigned/Short	
Network Node id as defined in import file Select Network node id Select Network node id Select Network node id Model Alias Node 2 Node 2 Node 2 Node 3 Node 4 Node 5 Node 5 Node 6 Node 7 Node 7 Node 7 Node 8 Node 9 Node 9 Node 9 Node 1 Node 2 Node 1 Node 2 Node 2	No de 1, MA/H TIER			ABB Worldson TCP	peak? 1	3 Dumigned Short	
Interview Interview Interview Node id as defined in import file Image: Select Network node id Image: Node 1 Image: Select Network node id Node 1 Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id Image: Select Network node id	Node1/Water_level			ABE Collective ETh	lani 1	10 0 unsigred Short	
Image: Section of the section of t		۲		1	udel	Alias	
Node2 taoname me taoname me Water_level Ok Cancel Martine Martine Martine Martine Martine Martine Martine Martine Ok Cancel Martine Martine Martine Martine Martine Martine Martine Martine Martine Martine Martine Martine Martine Martine Martine </th <th></th> <th></th> <th></th> <th></th> <th></th> <th></th> <th></th>							
Image:							
tagname me Water_jevel Ok Cancel			10000			Nouez	
tagname me Water_level Ok Water_level Ok Cancel Water_level Water_level <tr< th=""><th>_</th><th>_</th><th></th><th></th><th></th><th></th><th></th></tr<>	_	_					
Water_level Ok Cancel Water_level UNV122T t_24TT12_buildness CtdD122T12MCMM MV/2 24T5 0 unsignest2hort faile H_VV12T22T_level MV/2 0 0 unsignest2hort faile Date:_buildness Date:_buildness Dtd12 0 unsignest2hort faile	193						
Wat 12/h Ok Cancel 11_2/r17/r_1volifivitie 0		me					
Watterin T2m17x_holdholis SUT_Sm17_holdholis CSECETERCM MV/D DL_WV12ET_Javail MV/D Date_holdholis 0 Unity restriction 0 Date_holdholis MV/D Date_holdholis MV/D	_					Ok	Cancel
ALTT_Strift_Isolatelia Automatical CBDCSTREAGN MNVD 245 0 unsigned/Short false NL_WV/TETL_level MNVD 0 0 unsigned/Short false Etable_truthetia MNVD 12 0 unsigned/Short false							
CIBOLOTERICIAN MNV/D 245 D unsigned/Drivet false N_WY/TETL_leval MNV/D 0 0 unsigned/Drivet false Etable_Involvedian MNV/D 12 D unsigned/Drivet false							
N_W//TET_laval MN/O 0 0 unsigned/Shot false Date_holdrate MN/O 12 0 unsigned/Shot false		<u></u>		Total I			
Data Institute MIVO 12 0 unsigned Droit false	Contraction of the second second			245	0		
				0	0		
Ball holdvala MAVD 111 D unsigned/Short Salae				14	0		



Note: aliasing tag names is only available when tags can be imported. Tags which are added manually in the Tag Editor do not need to have the Alias prefix in the tag name.

The Alias string is attached to the tag name only at the moment the tags are imported using Tag Editor. If you modify the Alias string after the tag import has been completed, there will be no effect on the names already present in the dictionary. When the Alias string is changed and tags are imported again, all tags will be imported again with the new prefix string.

Communication Status

The current communication status can be displayed using the dedicated system variables. Please refer to the User Manual for further information about available system variables and their use.

The codes supported for this communication driver are:

Error	Notes
NAK	Returned in case the controller replies with a not acknowledge; can be returned also in case the network/node/unit parameters contained in the PLC response are not matching with panel configuration
Timeout	Returned when a request is not replied within the specified timeout period; ensure the controller is connected and properly configured to get network access

Error	Notes
Invalid response	The panel did receive from the controller a response, but its format or its contents is not as expected; ensure the data programmed in the project are consistent with the controller resources. The same error can be returned also in case the PLC could not complete the processing of the panel request and sent back to the panel and invalid/not completed response.
Cnt error	Returned when a specific control character in the protocol frame received does not match with the corresponding one in the request; verify the proper settings of the controller network configuration
General Error	Error cannot be identified; should never be reported; contact technical support

Omron FINS SER

This driver supports the FINS protocol via serial connetcion. For a list of models that support the FINS Communications Service, refer to the manufacturer's website.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In Config node double-click Protocols.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the **PLC** list.

The protocol configuration dialog is displayed.

Omron FINS SER	×
	Comm OK
PLC Models	Cancel
CJx/CS1x/CP1x	

Element	Description
PLC	PLC models available:
Models	• CJx/CSx/CP1x
Comm	If clicked displays the communication parameters setup dialog.

Element	Description	
	Comm Parameter Dia	alog
		ОК
	Uart	com1 👻
	Baudrate	9600 🗸
	Parity	even 🔻
	Data bits	7 👻
	Stop bits	2
	Mode	RS-485
	Element	Parameter
	Port	Serial port selection.
		COM1: device PLC port.
		 COM2: computer/printer port on panels with 2 serial ports or optional Plug-In module plugged on Slot 1/2 for panels with 1 serial port on-board.
		• COM3 : optional Plug-In module plugged on Slot 3/4 for panels with 1 serial port on-board.
	Baudrate, Parity, Data Bits, Stop	Serial line parameters.
	bits	
	Mode	Serial port mode. Available modes:
		Serial port mode. Available modes: • RS-232.

Tag Editor Settings

In Tag Editor select the protocol Omron FINS SER.

Add a tag using [+] button. Tag setting can be defined using the following dialog:

Mron FINS SER					×
Memory Type I/O area	Offset ▼ 0 ●	Subindex			
Data Block	Type boolean	T	Arraysize		
Conversion	+/-				
			Cancel	Apply	Help

Element	Description				
Memory Type	Memory Type	Description			
Type	I/O area	Corresponds to CIO resource on PLC			
	Auxiliary area	Corresponds to A resource on PLC			
	Holding area	Corresponds to H resource on PLC			
	Timer completion flags	Corresponds to T resource on PLC			
	Timer PVs	Corresponds to TPV resource on PLC			
	DM area	Corresponds to D resource on PLC			
	Counter completion area	Corresponds to C resource on PLC Corresponds to CVS resource on PLC			
	Counter CVs				
	EM area	Corresponds to E resource on PLC			
	Work area	Corresponds to W resource on PLC			
	Index registers	Corresponds to IR resource on PLC			
	Data registers	Corresponds to DR resource on PLC			
Offset	Starting address for the Tag. T	he possible range depend on memory type selected.			

Element	Description					
Subindex	This parameter allow to select a single part of the resource if the selected data type is shorter than the resource data type					
Data block	Instance of resource of the PLC.					
Data Type	Available data types:					
	 boolean byte short int unsignedByte unsignedShort unsignedInt float 					
	 double string 					
	• binary					
	See "Programming concepts" section in the main manual.					
	Note: To define arrays, select one of Data Type format followed by square brackets (byte[], short[]).					

Element	Description					
Arraysize	In case of array tag, this property represents the number of array elements.					
	 In case of string tag, this property represents the maximum number of bytes available in the string tag. 					
	set to UTF-8 o	of bytes corresponds to number of string characters if Encoding property is or Latin1 in Tag Editor. operty is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one uires 2 bytes.				
Conversion	Conversion to	be applied to the tag.				
	inv,swap2	Allowed Configured				
	Depending on	BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits Cancel OK data type selected, the list Allowed shows one or more conversion types.				
	Value	Description				
	Inv bits	inv : Invert all the bits of the tag.				
		Example: 1001 \rightarrow 0110 (in binary format)				
		$9 \rightarrow 6$ (in decimal format)				
	Negate					
	Negate	$9 \rightarrow 6$ (in decimal format)				
	Negate AB -> BA	$9 \rightarrow 6$ (in decimal format) neg : Set the opposite of tag value. <i>Example:</i>				
		$9 \rightarrow 6$ (in decimal format) neg : Set the opposite of tag value. <i>Example:</i> $25.36 \rightarrow -25.36$				
	AB -> BA ABCD ->	$9 \rightarrow 6$ (in decimal format)neg: Set the opposite of tag value.Example: $25.36 \rightarrow -25.36$ swapnibbles: Swap nibbles in a byte.Example: $15D4 \rightarrow 514D$ (in hexadecimal format) $5588 \rightarrow 20813$ (in decimal format)swap2: Swap bytes in a word.				
	AB -> BA	$9 \rightarrow 6$ (in decimal format)neg: Set the opposite of tag value. <i>Example:</i> $25.36 \rightarrow -25.36$ swapnibbles: Swap nibbles in a byte. <i>Example:</i> $15D4 \rightarrow 514D$ (in hexadecimal format) $5588 \rightarrow 20813$ (in decimal format)				

Element	Description	
	Value	Description
	-> GHEFCDAB	<i>Example:</i> 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)
	ABCNOP -	swap8: Swap bytes in a long word.
	> OPMDAB	Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) $0\ 10000000110$ $0001\ 1100101110110100001011010000111001010101$
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)
	Select conversi	on and click +. The selected item will be added to list Configured .
	If more convers Configured).	ions are configured, they will be applied in order (from top to bottom of list
	Use the arrow b	outtons to order the configured conversions.

Tag Import

Exporting Tags from PLC

The Omron FINS SER driver can import tag information from CX-Programmer PLC programming software. The tag import filter accepts symbol files with extension ".cxr" created by the Omron programming tool.

The ".cxr" files can be exported from the symbol table utility.

See in figure how to access the Symbol Table (if configured) from the Omron programming software.

File Edit View Insert PLC Program Simulation	Tools Window Help			
] D 🖻 🖬 🙀 🍜 & X 🖻 🛱 🕷 🖄	. 으 🛛 🏘 💐 📽 🗍 💡	🤁 🔊	🍇 👼 🛄 II	
< ≪ Q へ	<u> </u>	一つ必由い	87 元 └ ᅛ 📗	💀 🕸 🛗
🖪 🗖 🖓 🖓 🖓 😭 📸 🕺 ए 🚊 🛄 19	品 16 🔺 🖢 💀 🗍	3 5 5 4	S ► ■ ■ ■	▶발받₩
	Name	Data Type	Address / Value	Rack Location
⊡ 😽 NewProject	- MyData_01	WORD	D0	
NewPLC1[CJ1M] Offline	- MyData_02	WORD	D2	
Symbols	- MyData_03	WORD	D3	
Settings				
Memory				
🖃 🖓 Programs				
NewProgram1 (00)				
Symb Edit				
🕞 🛱 Section 🔁 Insert Symbol				
□ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □				
₽ <u></u> Large Icons				
tongo tenis "=- S <u>m</u> all Icons				
<u>Details</u>				
X Cut				
©ру				
🔁 <u>P</u> aste				
Project Delete				
Reusable File	• <u>A</u> dd to Project			
NewPLC1/NewProgram1/	3 <u>S</u> ave As			
Hide				
<u>F</u> loat In Main Window				
Properties				
Compile / Find Report / Transfer /				
Save as a reusable file		NewPLC1(Net:0,Ne	ode:0) - Offline	

Importing Tags in Tag Editor

Select the driver in Tag Editor and click on the **Import Tags** button to start the importer.

	Tags	×								
+	_	Z	D	ß	>]	₽	A 9B	B>	ŧ3	1
Data			^		-	Та	g URI			_

The following dialog shows which importer type can be selected.

HMIStudio				×
Multiple tag	mporters are available for this	protocol. Please sele	ct the importer type and	continue.
Version	Туре			
CX-Programmer v1	1 Linear			
Tag Editor exported	i xml General			
			ОКС	ancel

Importer	Description	
CX-Programmer v1.1 Linear	Requires a .cxr file. All variables will be displayed at the same level.	
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.	
	+ - ≱ ⓓ ℗ >] ▷ ✿ ▷ ▮ ▮ Data	

Once the importer has been selected, locate the symbol file and click **Open**.

Tags included in the symbol file are listed in the tag dictionary. The tag dictionary is displayed at the bottom of the screen.

Tags × Protocols						
+ - 👗 🕲 🔊	D & D D	R 🔎 - Search 🍸 Filter by: Da	ita 🔻 It	tems	used:6/10000 Protocol: Show a	I 💽 Show all tags 🔅 🗖
Data	Туре	Comment	^	Pr	roperty	Value
Modbus TCP:prot1	Container			~	Driver	
Model: Modicon Modbus(1-base	ea)				Model	Modicon Modbus(1-based)
Holding Registers 1	unsignedShort				Protocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			- V	Dictionary	
Holding Registers 3	unsignedShort				Array	false
- MRTU1	unsignedShort				Array size	0
- MRTU2	unsignedShort					
MRTU3	unsignedShort				Arrayindex.Subindex	400003
- MRTU4	unsignedShort				Comment	
- MRTU5	unsignedShort				Data type	unsignedShort

Toolbar item	Description
₽ X	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
「「「」	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.
	Example of both checked and unchecked result:
	Tops: Image: Image: </th
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

OPC UA Client

The OPC UA Client communication driver has been designed to connect HMI devices to OPC UA servers.

This implementation of the protocol operates as a client only.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In Config node double-click Protocols.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the $\ensuremath{\text{PLC}}$ list.

The protocol configuration dialog is displayed.

OPC UA Client		×
PLC Network		ОК
Alias		Cancel
IP address	0.0.0.0	
Port	4840	
Timeout (ms)	1000	
Security Policy	None 🗸	
Security Mode	None 🗸	
Username		
Password		
Server Certificate		
Client Certificate		
Client Private Key		
PLC Models		
Default		

Element	Description	
PLC Network	Enable access to multiple networked controllers. For every controller set proper options.	
Alias	Name identifying nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node.	
IP Address	IP address of the server.	
Port	Port number where the server is listening.	
Timeout (ms)	Time delay in milliseconds between two retries in case of no response from the server device.	
Security Mode	Type of authentication:	
	None: Certificates are not used	
	Sign: Certificates only used for authentication with server.	
	SignAndEncrypt: Certificates used for authentication with server and data encryption.	
Security Policy	Encryption level to use (used only when Security Mode is active).	
	• Basic256	
	Basic256Sha256	
Username Password	Authentication with user name and password	
Server Certificate	Certificate for OPC UA Server.	
	Server certificate can be downloaded using tag importer. See "Remote OPC UA Server certificate" on page 461	
Client Certificate	Certificate used by OPC UA client. If blank, a certificate is automatically generated.	
Client Private Key	Key used by OPC UA client. If blank, a key is automatically generated.	
PLC Models	No options available.	

Notes:

- Before choosing security options, be aware that not all security modes might be supported by the OPC UA server. Make sure to use security mode that is supported.
- When working within a private network you do not need to provide devices' certificates because you trust used devices. On a public network, instead, the certificate will give you a guarantee of the identity of devices.

External Certificate

ASCII version of the certificate (usually a file with .pem extension) is required.

Edit the certificate files and then copy and paste the full text of your certificate to the certificate fields.

Step 1: Remove header and footer lines

----BEGIN CERTIFICATE-----

MIIDNjCCAh4CCQCJtJgjqDDUqjANBgkqhkiG9w0BAQsFADBdMQswCQYDVQQGEwJJ
VDEPMA0GA1UEBwwGVmVyb25hMRQwEgYDVQQKDAtDb21wYW55TmFtZTERMA8GA1UE
CwwIUiZEIFRlYW0xFDASBgNVBAMMC0hNSURldmljZUlEMB4XDTE4MDMyNjA5MTAz
OFoXDTI4MDMyMzA5MTAzOFowXTELMAkGA1UEBhMCSVQxDzANBqNVBAcMB1Z1cm9u
YTEUMBIGA1UECgwLQ29tcGFueU5hbWUxETAPBgNVBAsMCFImRCBUZWFtMRQwEgYD
VQQDDAtITUlEZXZpY2VJRDCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEB
ALONtzGwlrGv6cXH8i7sNWbwmx9Xo4tp20khnt/VJnDLoYHv7ZvV1vQYHom3/HiC
IaWV/uUvYnXaNBlxHnPsQPV0bEEq26Np01ne8jXEHY6bcMVK3XBV3eno3adOwHA5
vio0MmF6fPQVWTfyVb4/MrcfqUke1gWk3sFlFxEtxXlRLOwNK1+G7Wbnb30j4oPL
Ev60VN3DwisDzvivpW7Nv4RPjNK9XJ2DVI+/+KDCNNLlP8GpD0xB1iIpj1S8BwqZ
om1+SUs10IM1cfv/AfArZj9QaIo3c2uPwkLncqQxfDvmlC1fCfsRVxm5N3bmimwC
2F6hbkZksLp7ovCx/haKhfkCAwEAATANBgkqhkiG9w0BAQsFAAOCAQEALVjkNEa/
40JnMZIVkSZZWGylHHGZ8rphcUPH4olbq7MkaHk7mKacYKqI/qorrIPhmKf7Y2x5
UcTN4Uff6NT0xjrMUg2Q6Lp+a/fBqOUvEebrtmd8NYbhjTs4iVYg3R/NBlgrfx9N
6Ipp060Jo0hYXjwDZU0HADnSXVABeBxzAESvLVK7mxgXypdB1D+kgcC6hL9Xv4u5
melNI24LNkRiBT35Exlo2YTu4I9YHFelc5iILvC6DpUYHeSlIEKiNmccL2DDGEBZ
TscRZykvWRi1Xpm2WMzjbf9HE0XNRM8DTCkOscxcrYZrcTVpm0a0WH50D2531LnF
XsH5sLPyOxtKFw==
END CERTIFICATE

Step 2: Remove all Newline characters

MIIDNjCCAh4CCQCJtJgjqDDUqjANBgkqhkiG9w0BAQsFADBdMQswCQYDVQQGEwJJVDEPMA0GA1.....

Step 3: Copy and paste the single text line of the certificate to the protocol dialog

Script to generate a Certificate

If you want to use your own certificate, note that the certificate must include the "Subject Alternative Name (SAN)" parameters as required by the OPC UA standard.

Here is an example of how to generate certificate files using a public OpenSSL-Win32 library (Reference: https://www.openssl.org/)

```
@echo off
set OpenSSL="C:\Program Files (x86)\OpenSSL-Win32\bin\openssl.exe"
set NodeName=HMI-Client
rem Generate an RSA key
   %OpenSSL% genrsa -out client-key.pem 2048
rem Creating Certificate Signing Requests
    %OpenSSL% req -new -key client-key.pem -out client.csr -subj "/ST=NY/C=US/L=New
York/O=CompanyName/OU=R&D Team/CN=OPCUAClient@%NodeName%"
rem Creating Certificate (.pem)
   echo subjectAltName=URI:urn:%NodeName%:CompanyName:OPCUAClient > san.txt
   echo
keyUsage=digitalSignature, nonRepudiation, keyEncipherment, dataEncipherment, keyCertSign
>> san.txt
   echo extendedKeyUsage=critical,serverAuth,clientAuth >> san.txt
   echo authorityKeyIdentifier=keyid,issuer >> san.txt
   echo basicConstraints=CA:TRUE >> san.txt
```

```
%OpenSSL% x509 -req -days 3650 -in client.csr -signkey client-key.pem -out
client.crt -extfile san.txt
rem Convert Certificate (.der)
  %OpenSSL% x509 -in client.crt -outform der -out client.der
rem Not necessary files
  del san.txt
```

pause

Tag Editor Settings

Path: ProjectView> Config > double-click Tags

- 1. Select OPC UA Client from the protocol list.
- 2. To add a tag, click +: tag definition dialog is displayed.

OPC UA Client				×
OPC UA Client				
Data type	Arraysize	Conversion		
INT	•		+/-	
Tag name				
	OK	Cancel	Apply	Help

Element	Description			
Data Type	Available data types:			
	• boolean			
	• byte			
	• short			
	• int			
	unsignedByte			
	unsignedShort			
	unsignedInt			
	• float			
	double			
	 time uint64 			
	• int64			
	• string			
	• binary			
	See "Programming concepts" section in the main manual.			
	Note: To define arrays, select one of Data Type format followed by square brackets.			
Arraysize	In case of array tag, this property represents the number of array elements.			
	 In case of string tag, this property represents the maximum number of bytes available in the string tag. 			
	Note: number of bytes corresponds to number of string characters if Encoding property is			
	set to UTF-8 or Latin1 in Tag Editor.			
	If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one			
	character requires 2 bytes.			
Conversion	Conversion to be applied to the tag.			
	Conversion			
	inv,swap2 Allowed Configured			
	BCD Inv bits AB->BA ABCD->CDAB			
	ABCD->CDAB			
	ABCDEFGH->GHEFCDAB Inv bits			
	Cancel			
	Depending on data type selected, the list Allowed shows one or more conversion types.			

t Description	n
Value	Description
Inv bits	inv: Invert all the bits of the tag.
	Example: $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)
Negate	neg: Set the opposite of tag value.
	<i>Example:</i> 25.36 → -25.36
AB -> BA	swapnibbles: Swap nibbles in a byte.
	Example: 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)
ABCD ->	swap2: Swap bytes in a word.
CDAB	Example: 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)
ABCDEFC	GH swap4: Swap bytes in a double word.
-> GHEFCDA	AB $Example:$ 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)
ABCNO	P - swap8: Swap bytes in a long word.
> OPMDA	B Example: 142.366 → -893553517.588905 (in decimal format) 0 10000000110 0001110010111011001000101101000011100101
	→ 1 10000011100 1010101000010100010110110110010110110000
BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9)
	Example: $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)

Element	Description
	Select conversion and click +. The selected item will be added to list Configured .
	If more conversions are configured, they will be applied in order (from top to bottom of list Configured).
	Use the arrow buttons to order the configured conversions.
Tag name	Name of tag to be used in communication.



Note: Tag properties result from import process. In most cases manual creation of new tags is not necessary.

Node Override IP

The protocol provides the special data type Node Override IP which allows you to change the IP address of the target controller at runtime.

This memory type is an array of 4 unsigned bytes, one per each byte of the IP address.

The Node Override IP is initialized with the value of the controller IP specified in the project at programming time.

Node Override IP	PLC operation
0.0.0.0	Communication with the controller is stopped, no request frames are generated anymore.
Different from 0.0.0.0	It is interpreted as node IP override and the target IP address is replaced runtime with the new value.

If the HMI device is connected to a network with more than one controller node, each node has its own Node Override IP variable.

Note: Node Override IP values assigned at runtime are retained through power cycles.

Hostname DNS or mDNS

In addition to the array of bytes, string memory type can be selected to be able use the DNS or mDNS hostname as an alternative to the IP Address.

OPC UA Client		×
OPC UA Client		
Memory Type	Data type Arraysize	
Node Override IP 🗸 🗸	string ~ 8	
Conversion	Tag name	
	OK Cancel	Apply Help

Node Override Port

The protocol provides the special data type Node Override Port which allows you to change the network Port of the target controller at runtime.

This memory type is unsigned short.

Node Override Port is initialized with the value of the controller Port specified in the project at programming time.

Node Override Port	Modbus operation	
0	Communication with the controller is stopped, no request frames are generated anymore.	
Different from 0	It is interpreted as the value of the new port and is replaced for runtime operation.	

If the HMI device is connected to a network with more than one controller node, each node has its own Node Override Port variable.



Note: Node Override Port values assigned at runtime are retained through power cycles.

OPC UA Client		×
OPC UA Client		
Momony Type	Data tuno	Arrausia
Memory Type	Data type	Arraysize
Node Override Port 🗸 🗸	unsignedShort V	0
Conversion	Tag name	
	+/-	
	OK Can	cel Apply Help

Adding an alias name to a protocol

Tag names must be unique at project level, however, the same tag names might need to be used for different controller nodes (for example when the HMI device is connected to two devices running the same application).

When creating a protocol you can add an alias name that will be added to tag names imported for this protocol.

In the example, the connection to a certain controller is assigned the name **Node1**. When tags are imported for this node, all tag names will have the prefix **Node1** making each of them unique at the network/project level.

	b 💼 刘	Madhua TCP:set1				
1 -				- 67 🗔		1
Name 🛆			Driver	Addre		Comment
Node 1/ Gal _ hodrotte			Bue TCP pet1	1 11 Dunieg		
Node1/Cate_hodrote			Bue TCP pet1	1 12 Dunes		
Node 1/19 W/ATER law	116		Bue TCP pet1	1 DDunwig		
Note 1/OBOGEDHOM			Bue TOPpet1	1 245/Dune		
Nederl/ORUT_BRIT_Head			Bue TCPpet1	1 1 Dunkig		
Node1/FL_D/RTA_houdry	1010		Bue TOPpet1	1 2 Dunkig		
Norther 1, 1987 (H. 77 EUR)			Bue TOPpet1	1 3 Dunkig		
Node1/Water_level		Hitti Cath	aftya Effiliana "	1 10 0 unsig	redShort	
↑	Netwo	rk				
	و ا	Select Network node id				
	•		1			
	e (e)	Select Network node id	Model		Alias	
			Model	-But	Alias Node1	
		Slave Id				
		Slave Id	Noticer (the		Node1	
		Slave Id	Noticer (the		Node1	
		Slave Id	Noticer (the		Node1	
		Slave Id	Noticer (the		Node1	
tagname 🔽		Slave Id	Noticer (the		Node1 Node2	
		Slave Id	Noticer (the		Node1	Cancel
tagname Water_level		Slave Id	Noticer (the		Node1 Node2	Cancel
tagname Water_level		Slave Id	Noticer (the		Node1 Node2	Cancel
tagname Water_level		Slave Id	Noticer (the	illus [Node1 Node2	Cancel
tagname C	me	Slave Id	Noticer (the	iin [Node1 Node2 Ok	
Water_level	me	Slave Id	Noticer (the	iliua (Node1 Node2 Ok	faine



Note: Aliasing tag names is only available for imported tags. Tags added manually in the Tag Editor cannot have the Alias prefix in the tag name.

The Alias string is attached at the time of tag import. If you modify the Alias string after the tag import has been completed, there will be no effect on names already present in the dictionary. When the Alias string is changed and tags are re-imported, all tags will be re-imported with the new prefix string.

Importing tags

Tags for OPC UA Client protocol must be imported from OPC UA servers.

Path: ProjectView> Config > double-click Tags

- 1. Select **OPC UA Client** in the list of available protocols.
- 2. Click Import Tags.
- 3. Select Hierarchical importer.
- 4. Enter address of the server.
- 5. Choose Security and Authentication mode.
- 6. Click **Browse** to connect and retrieve tag dictionary from the OPC UA server.
- 7. The OPC UA Server will provide its own certificate. You have to accept the certificate to continue and retrieve data.
- 8. When the discovery process has been completed, click **OK** to create the dictionary with the tags.

OPCUA Client imp	orter	×
Symbol disco	overy, click 'Browse' to pull symbols. Do you want to continue?	
opc.tcp://192.168.	.44.165:48010 ~	Browse
Security Settings		
Security Policy	None	-
Security Mode	None	-
Client Certificate	Client's own certificate	
Private Key	Client's private key	
Authentication Set	ttings	
Anonymous		
Username	Both security policy and mode should be none.	
Password	Both security policy and mode should be none.	
Password	bour security policy and mode should be none.	
-	0 / Remaining nodes to process: 0	
 ✓ ✓ ServerNan ✓ ✓ opc.tc 	ne :p://192.168.44.165:48010	
✓ Ø 01		
] Server] Tags	
	Tag1	
	☑ Tag2 ☑ Tag3	
>	✓ Tag4	
> ~	☑ Tag5 Protocols	
> 🗹 Ту	, pes	
		Canaal
	ОК	Cancel

Element	Description
Remote URI	Address of OPC UA Server in the form: <i>opc:tcp:<ipaddress>:<port></port></ipaddress></i>
	Example:
	• opc.tcp://192.168.44.165:4840
Security Mode	Type of authentication:
	None: No authentication with server and no data encryption.
	Sign: Certificates only used for authentication with server.
	• SignAndEncrypt : Certificates used for authentication with server and data encryption.
Security Policy	Encryption level to use (used only when Security Mode is active).

Element	Description	
	 Basic128Rsa15 Basic256 Basic256Sha256 	
Username Password	Authentication with user name and password	
Client Certificate	Certificate used by OPC UA client. If blank, a certificate is automatically generated. The certificate is used by the importer only if requested by the server	
Client Private Key	Key used by OPC UA client. If blank, a key is automatically generated.	



To be allowed to retrieve data from the OPC UA Server you must provide the required security parameters. Dialog will be filled automatically with the parameters provided by protocol editor settings (you can simply accept the proposed values)

Remote OPC UA Server certificate

C	ertificate details		×
D	o you want to trust <u>t</u> Certificate details	his server certificate permanently?	
	CommonN	OPCUAServer@HMI-Server	^
	Organization	CompanyName	
	Organizatio	R&D Team	
	Locality	Meronalities -	
	State	daily setting	
	Country		
	DomainCo		
	Issuer	OPCUAServer@HMI-Server	
	Valid from	2018-06-01T10:36:06.000Z	
	Valid to	2028-05-29T10:36:06.000Z	
	Serial Num	8785CE5BDC6B570D	
	Thursdamint	L-261L660-J-20-71644050100-JL-27L-0	*
	Accept permanently	Accept temporarily Copy to clipboard Can	el

When OPC UA Server provides its own certificate, you have the option to:

Accept temporarily

Certificate is accepted for current working session only.

Accept permanently

Certificate is accepted and copied to computer. Any future import request for the same OPC UA Server will be accepted automatically without asking confirmation.



The certificate file will be copied inside the folder: %AppData%\Roaming\...\studio\OPCUA\pki\trusted\certs

Copy to clipboard

ASCII format of the certificate is copied to the clipboard to allow you to verify its authenticity, save and insert it into protocol configuration (if required).



To verify a certificate, use a text editor to paste it from the clipboard to a text file with the extension .crt. You can then double-click the .crt file to allow Windows to view the properties of certificate.

🦗 Certificate	×
General Details Certification Path	
Certificate Information This CA Root certificate is not trusted. To enable trust, install this certificate in the Trusted Root Certification Authorities store.	
Issued to: HMI-Server	
Issued by: HMI-Server	
Valid from 24/05/2018 to 21/05/2028	
Install Certificate Issuer Stat	ement
	ОК

Cancel

Cancel the import operation

Communication status

Current communication status can be displayed using System Variables. See "System Variables" section in the main manual.

Codes supported for this communication driver:

Error	Description
Connecting < Error description>	Error during connection
Connection while reading: <error description=""></error>	Error encountered when connecting for read operation
Bad status while reading: <error description=""></error>	Error in read operation

Error	Description
Connection while writing: <error description=""></error>	Error encountered when connecting for write operation
Bad status writing: < Error description>	Error in write operation
OPC UA client for given node ID not found	Wrong node ID information

<Error description> can be one of the following:

Error	Notes
BadTimeout	Timeout error. No answer from server.
BadSecurityChecksFailed	Error during exchange of certificates. Typically occurs when the server does not accept the client certificate as trusted.
BadCertificatexxxInvalid	Error in client or server certificate.
BadNodeUnknown	The tag (node) does not exist.
BadAttributeNotFound	Attempt to access an invalid attribute.
BadNotWritable	Attempt to write to a read-only attribute.

Panasonic FP/FP7

The HMI devices can be connected to a Panasonic FP/FP7 PLC as the network master using this communication driver.

This driver has been designed for connection to the programming port of the PLC with serial or Ethernet connection.

Please note that changes in the communication protocol specifications or PLC hardware may have occurred since this documentation was created. Some changes may eventually affect the functionality of this communication driver. Always test and verify the functionality of your application. To fully support changes in PLC hardware and communication protocols, communication drivers are continuously updated. Always ensure that the latest version of communication driver is used in your application.

Protocol Editor Settings

Add (+) a driver in the Protocol editor and select the protocol called "Panasonic FP/FP7" from the list of available protocols.

Panasonic FP/FP7		×
PLC Network	Comm	ОК
Alias		Cancel
Node ID	1	
Force Read Single		
Media	Serial ~	
Request Delay (ms)	0	
IP address	0.0.0.0	
Port	0	
Timeout(ms)	500	
PLC Models		
FP		
FP7		•

The driver configuration dialog is shown in figure.

Element	Description	
Alias	Name to be used to identify nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node	
Node ID	Node number of the slave device	
Force Read Single	When enabled, data communication is performed according to the "Offset" and	

Element	Description	
	"Data Type" settings specified in the tag. When disabled, data communication of tag setting information is performed in one message.	
Media	Serial or Ethernet	
IP Address	Ethernet IP address of the controller (valid only when Media has been set to "Ethernet")	
Port	Using Panasonic FPWIN Pro 7 it is possible to connect to PLC through HMI. In this case HMI is connected to PLC via serial, and FPWIN Pro 7 is connected to HMI via Ethernet. Port option changes its meaning depending on Media option:	
	 If Media is set to Ethernet, Port is the PLC listening port used by the HMI to connect 	
	 If Media is set to Serial, Port is the HMI listening port used by the FPWIN Pro 7 to connect 	
	The default value can be changed when the communication goes through routers or Internet gateways where the default port number is already in use.	
Timeout (ms)	Defines the time inserted by the protocol between two retries of the same message in case of missing response from the server device. Value is expressed in milliseconds.	
PLC Models	The list allows selecting the PLC model you are going to connect to. The selection will influence the data range offset per each data type according to the specific PLC memory resources.	

Element	Description		
PLC Network	The protocol allows the connection of multiple controllers to one HMI device. To set-up multiple connections, check "PLC network" checkbox and create your network using the command "Add" per each slave device you need to include in the network.		
Comm	Recalls the serial port configuration parameters as shown in the figure.		
	Comm Parameter Dialog		
	OK		
	Port con	1 🔹	
	Baudrate 960		
	Parity odd	•	
	Data bits 8	•	
	Stop bits 1	_	
	Mode RS-	485 💌	
	Element	Parameter	
	Port	Serial port selection.	
		 COM1: device PLC port. COM2: computer/printer port. 	
	Baudrate, Parity, Data Bits,	Stop bits Serial line parameters.	
	Mode	Serial port mode. Available modes:	
		• RS-232.	
		• RS-485 (2 wires).	
		• RS-422 (4 wires).	

Data Types

_

When creating a tag you have to specify its properties. Data type are specific to xAscender Studio, memory type are specific to the selected protocol. Choose the value according to the internal representation you need for the selected controller address.

Note: arrays type use the same data type followed by "[]" (i.e.: boolean [])

FPWIN Pro	xAscender Studio	Description
BOOL	boolean	One bit data (01)
INT	short	Signed 16 bits data (-3276832767)
UINT	unsignedShort	Unsigned 16 bit data (065535)
DINT	int	Signed 32 bit data (-2.1e9 2.1e9)
UDINT	unsignedInt	Unsigned 32 bit data (0 4.2e9)
REAL	float	IEEE single-precision 32-bit floating point type (±1.17e-38 ±3.40e38)
STRING	string	Characters coded according to selected format
TIME		Unsigned 32 bit data (0 4.2e9). These PLCs supports only a 16 bit time value: FP1, FP3, FPC, FP5, FP10/10S. The area is T#0s–T#327.67s. ATTENTION: The HM panel uses a 32 bit access to the PLC.
TIME		Unsigned 32 bit data (0 4.2e9) theoretical. Used in PLC T#0s–T#21474836.47s
DATE_AND_TIME	unsignedInt	Unsigned 32 bit data (0 4.2e9) equates DT#2001-01-01- 00:00:00– DT#2099-12-31-23:59:59 in the PLC
TIME_OF_DAY		Unsigned 32 bit data (0 4.2e9) equates TOD#00:00:00– TOD#23:59:59 in the PLC
DATE		Unsigned 32 bit data (0 4.2e9) equates D#2001-01-01– D#2099-12-31 in the PLC

This table shows the tags, after a tag import from Panasonic FPWIN Pro tags to xAscender Studio tags.

Aliasing Tag Names in Network Configurations

Tag names must be unique at project level; it often happens that the same tag names are to be used for different controller nodes (for example when the HMI is connected to two devices that are running the same application). Since tags include also the identification of the node and Tag Editor does not support duplicate tag names, the import facility in Tag Editor has an aliasing feature that can automatically add a prefix to imported tags. With this feature tag names can be done unique at project level.

The feature works when importing tags for a specific protocol. Each tag name will be prefixed with the string specified by the "Alias". As shown in the figure below, the connection to a certain controller is assigned the name "Node1". When tags are imported for this node, all tag names will have the prefix "Node1" making each of them unique at the network/project level.

	b 🛋 🔰	Panasonic FP/FP7;p	rot1	- 5		
Name 🛆		Group Driver		Addre		Comment
Node 1/ Gal _hodrolia			Bue TCP pol(1	1 TT Durwig		
Node1/Cala_hodrolia			Isue TCP petri	1 12 Duneig		
Node 1/PN_W/R71EIR_Mean	al l		Bue TCP pet1	1 DDurwig	Heard Consult	
Node1/0806#BHOW			Isue TCP petri	1 245 Dune		
Nedet/OUT_BRT_Nede			Bue TCP pol(1	1 1 Durning	Hard Share	
Nedle1/R_D/KTX_headre	fia		Isue TCP petit	1 2 Durwig		
NAMES OF TAXABLE PARTY.		HEE West	Ibue TCP petri	1 3 Durwig	horethan	
Node1/Water_level		AND CALL	No. 2711 anti-	1 10 0 unsig	redShort	
		Node id as defined in im	port nie			
	() S		Model		Alias	
	• s	Select Network node id	Model	il a		
	• s	Select Network node id			Alias Node1 Node2	
) 출권 taoname 또	• s	ielect Network node id	Model		Node1	
Ra Ra	© S	ielect Network node id	Model		Node1	Cancel
tagname Water_level	© S	ielect Network node id	Model		Node1 Node2	Cancel
taoname Water_level	© S	ielect Network node id	Model		Node1 Node2	Cancel
tagname Water_level	© S	ielect Network node id	Model	iliua [Node1 Node2	Cancel
tagname T Water_level	me	ielect Network node id	Model	ikua (Node 1 Node 2 Ok	Cancel
tagname T Water_level	me	ielect Network node id	Model	iliua (1) (1) (1) (1) (1) (1) (1) (1) (1) (1)	Node 1 Node 2 Ok	Cancel



Note: Aliasing tag names is only available when tags can be imported. Tags which are added manually in the Tag Editor do not need to have the Alias prefix in the tag name.

The Alias string is attached to the tag name only at the moment the tags are imported using Tag Editor. If you modify the Alias string after the tag import has been completed, there will be no effect on the names already present in the dictionary. When the Alias string is changed and tags are imported again, all tags will be imported again with the new prefix string.

Tag Import

The Panasonic FP/FP7 driver can import tag information from Panasonic Control FPWIN Pro 7 PLC programming software. The tag import filter accepts symbol files with extension ".csv" created by the programming tool.

The ".csv" files can be exported from the symbol table utility.

See in figure how to export tags from the programming software.

11-1		1.7	· · ·		D	1		
	penannt - Control FPWIN Pro 7 - The IEC 6113				-			
Pro	oject Object Edit Tools Online N New	Monito		xtras	Wind PDHP ×			<u> </u>
6		ð			i 💖 🖞	⊴ 🕏 🖇 🖛 :	** *= 🕑 📴 🔒	
_	Open recent projects	μ×	中唱 Program					
	Save Ctrl+S		Clas			Identifier		Туре
	Save as							
	Close	μ×	1	1				
	Import •							
	Export 1	8	All objects					
		뙵	Objects selecte		avigato			
56			Variables as CS Program code		I C and	2 •	Global variables	- -
	Used memory	_	Program code		LC CON	riguration	All project varia	bles U
J.	Printer setup Print preview Ctrl+Q							· · · ·
	Print Ctrl+P		3					
2	Open cross-reference list							
1	-							
Êx				,				
×	Exit Alt+F4							
-			•					
	CV and a table to a sight to list							×
0	SV export global variable list							^
l r	<u></u>	_			_			
	Class Identifier		Assign	>>		Feld 1 (Class) Feld 2 (Identif		
	IEC address					Feld 3 (IEC ad	ddress)	
	FP address Type		<< Rem	ove		Feld 4 (FP ad Feld 5 (Type)		
	Initial					Feld 6 (Initial)		
	Comment					Feld 7 (Comm	ent)	
					I	Customize colu	umn title	
ſ	Export column title as first line					Save e	export configuration	
	Only export variables with explicit a	ddres	35			Load e	export configuration	
						Reset to	default configuration	n
							-	
1	Array variables							
	Export array variable declarations							
	Export array elements as single va	ariabl	es					
	Only for arrays with fewer that	n						
	1000 elements							

DUT variables

Export DUT variable declarations Export DUT elements as single variables

4

Export

Cancel

Help

Export to file			×
$\leftarrow \rightarrow \land \uparrow$	» > v ق	Topolo destantes	Q
Organisieren 🔻	Neuer Ordner		• ?
in the second	^ Name	Änderungsdatum	Тур ^
	and a	1000,000 (0.00	1000
	Submary .	100,000,000	Theory
	.	100 AT 100 A	Sec.
	burnt.	1000 000 000	the second s
	Trapellar .	10,0,000,000	
4	v <		>
Dateiname:	.CSV		~
Dateityp:	CSV files (Unicode) ([*] ,csv)		~
 Ordner ausblende 	n !	5 Speichern Abbr	echen

Select the driver "Panasonic FP/FP7..." in the Tag Editor and click on the "Import Tags" button to start the importer.

1:Page1* Tag	gs 🗙		
+-~~ *	: 🐂 📖 🔁 🚺	B Panasonic FP/FP7:pr	ot 1 🗸 🗸 🖉 [
Name	roups	Driver	Address
g_aiBasisDT100	6 Import Ta	hasonic FP/FP7:prot1	1 DT 100 unsignedShort
:D:-DT110	- mpore in	97	1 DT 110
HMWIN			×
Multiple tag impo	rters are available for this	protocol. Please select the im	porter type and continue.
Version	Туре		
FPWIN v1.0	Linear		
Tag Editor exported xml	l General		
		7	OK Cancel

In the file select window locate the ".csv" file and confirm.

The tags present in the exported document are listed in the tag dictionary from where they can be directly added to the project using the add tags button.

Recursive	₽- Search	🍸 Filter by: Data 🗸
Import Tag(s)	Туре	select for import
 Panasonic FP/FP7:prot1 Model: FP7 	Container	
g_tZeit_a	unsignedInt	
gDate	unsignedInt	
gDATE_AND_TIME	unsignedInt	
ATTME OF DAY	unsignedInt	

Communication Status

The current communication status can be displayed using the dedicated system variables. Please refer to the User Manual for further information about available system variables and their use.

The codes supported for this communication driver are:

Error	Notes
NAK	Returned in case the controller replies with a not acknowledge
Timeout	Returned when a request is not replied within the specified timeout period; ensure the controller is connected and properly configured to get network access
Line Error	Returned when an error on the communication parameter setup is detected (parity, baud rate, data bits, stop bits); ensure the communication parameter settings of the controller is compatible with panel communication setup
Invalid response	The panel did receive from the controller a response, but its format or its contents is not as expected; ensure the data programmed in the project are consistent with the controller resources
General Error	Error cannot be identified; should never be reported; contact technical support

Ping

Ping communication driver allows to send ping commands to a specific IP address.

The purpose of this communication driver are:

- test a connection between the HMI and another device in the same network
- check internet connectivity by executing ping commands to a public IP address (example 8.8.8.8)

In case it is needed to send ping commands to many IP addresses at the same time, it is possible to create many instances of Ping protocol:

Protocols x		
+ - ^ / 5	P	
PLC	Configuration	
Ping:prot1	CfgVer=1 model=1 ipAddress=8.8.8.8 protocol=ICMP port=80 timeout=5000	Ν
Ping:prot2	CfgVer=1 model=1 ipAddress=192.168.2.2 protocol=ICMP port=80 timeout=5000	Ν
Ping:prot3	CfgVer=1 model=1 ipAddress=192.168.2.3 protocol=ICMP port=80 timeout=5000	Ν



Ping communication driver is not counted as physical protocol. Refer to **Table of functions and limits** from main manual in "Number of physical protocols" line.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In Config node double-click Protocols.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the PLC list.

The protocol configuration dialog is displayed.

Ping		×
		ОК
IP address	8.8.8.8	Cancel
Protocol	ICMP 🔻	
Port	80	
Timeout (ms)	5000	
PLC Models		
default		

Element	Description	
IP address	Destination IP address to which ping commands are sent.	
Protocol	Network protocol used to send ping commands (default is ICMP).	
Port	Network port used for sending ping commands (fixed to 53 for ICMP Protocol).	
Timeout (ms) Polling time between each ping command sent.		
PLC Models	Fixed to default.	

Tag Editor Settings

Path: ProjectView> Config > double-click Tags

- 1. To add a tag, click +: a new line is added.
- 2. Select **Ping** from the protocol list: tag definition dialog is displayed.

Ping			×
Ping			
Memory Type	Data Type	Arraysize	
Node Override IP	✓ string	8	
Conversion			
	+/-		
	OK Car	Apply	Help

Element	Description				
Memory Type	Name Description				
Type	Node Override IP		fined, this Tag allows to ch ch ping commands are ser	nange the destination IP address to nt, at runtime.	
	Status	Rep	resents the result of last p	ing command:	
			• 0 = last ping command	failed	
			• 1 = last ping command	got response	
	Last ping time	Rep	resents the result of last p	ing time, expressed in milliseconds.	
Data Type	Data Type		Memory Space	Limits	
1900	boolean		1-bit data	01	
	unsignedByte[]		8-bit data	0 255	
	unsignedShort		16-bit data	0 65535	
	unsignedInt		32-bit data	04.2e9	
	string		Express the number of characters used to specify the destination IP address <i>Example: string[15]> xxx.xxx.xxx</i>		
Arraysiz e	This property represents the maximum number of bytes available in the string or in the array Tag.				
	UTF-8 or Latin1 in Tag E	ditor.	C C	chars if Encoding property is set to -16BE or UTF-16LE one char requires	

If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one char requires 2 bytes.

Tag Import

Select the driver in Tag Editor and click on the Import Tags button to start the importer.



The system will require a generic XML file exported from Tag Editor by appropriate button.

_ T	ags	×								
+	-	×	ß	C	>]	Þ	A 9B	B>	6 3	1
Data			^			Та	g URI			

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags × Protocols					
+ - 👗 🕲 🖉 🔰 [> \$ B ₪ ∰	R 🔎 - Search 🍸 Filter by: Data	▼ Ite	ms used:6/10000 Protocol: Show	all 💽 Show all tags 🔅 🗌
Data	Туре	Comment	^	Property	Value
Modbus TCP:prot1	Container			Y Driver	
Model: Modicon Modbus(1-based)				Model	Modicon Modbus(1-based)
	unsignedShort			Protocol	Modbus TCP:prot1
	unsignedShort unsignedShort			✓ Dictionary	
	unsignedShort			Array	false
	unsignedShort			Array size	0
	unsignedShort			Arrayindex.Subindex	400003
- MRTU4	unsignedShort			Comment	
- MRTU5	unsignedShort			Data type	unsignedShort

Toolbar item	Description
Ka	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
R ^a	Update Tag(s).
	Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag.
	Example of both checked and unchecked result:

Toolbar item	Description
	Tage* x Tage* x + - 2 0 >1 > 4 0
P + Search	Searches tags in the dictionary basing on filter of box item selected.

ROBOX BCC/31

ROBOX BCC/31 communication driver has been designed to connect HMI devices to ROBOX BCC/31 PLC through Ethernet connection.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the **PLC** list.

The protocol configuration dialog is displayed.

ROBOX BCC/31		×
PLC Network		ОК
Alias		Cancel
IP address	0 . 0 . 0 . 0	
Timeout (ms)	2000	
PLC Models		
Robox BBC/31		

Element	Description
Alias	Name identifying nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node.
IP address	Address of PLC.
Timeout (ms)	Time delay in milliseconds between two retries in case of missing response from the server device.
PLC Models	Allows to select between different PLC models:

Element	Description			
	Robox BBC/31			
PLC Network	IP address for all PLCs in multiple connections. PLC Network must be selected to enable multiple connections.			
	ROBOX BCC/31			
	PLC Network OK			
	Alias			
	IP addres ROBOX BCC/31			
	Timeout (
	PLC Mode Alias Cancel			
	IP address 0 . 0 . 1			
	Timeout (ms) 2000			
	Slaves PLC Models Robox BBC/31			
	Slave Id			

Tag Editor Settings

Path: ProjectView> Config > double-click Tags

- 1. To add a tag, click +: a new line is added.
- 2. Select **ROBOX BCC/31** from the **Driver** list: tag definition dialog is displayed.

BOX BCC/31 ROBOX BCC/31					
Memory Type	Offset	SubIndex			
Axis Index	Data Type	Ar	raysize		
Conversion	+/-				
	ОК	Cano	cel A	Apply	Help

Element	Description					
Memory Type	Resource where tag is located on PLC.					
	Available resources are:					
	Logic Input Bit					
	Logic Input Word					
	Logic Output Bit					
	Logic Output Word					
	 Phis Input Bit Phys Input Word					
	Phys Output Bit					
	Phys Output Word					
	Non Volatile I32					
	Non Volatile Double					
	Non Volatile string					
	Volatile I32					
	Volatile Double					
	Volatile string Decomptor 122					
	Parameter I32Parameter Double					
	Axis Parameter I32					
	Axis Parameter Double					
	Alarm Mask					
	Alarm Code					
	Alarm string					
Offset	Offset address where tag is located.					
	Offset addresses are six digits composed by one digit data type prefix + five digits resource address.					
SubIndex	This allows resource offset sel	ection within the selected memory type.				
Axis Index	Allows to select Axis index. Available only for Axis memory types.					
Data Type	Data Type	Memory Space	Limits			
	boolean	1-bit data	01			
	byte	8-bit data	-128 127			
	short	16-bit data	-32768 32767			
	int	32-bit data	-2.1e9 2.1e9			

Element	Description				
	Data Type	Memory Space	Limits		
	int64	64-bit data	-9.2e18 9.2e18		
	unsignedByte	8-bit data	0 255		
	unsignedShort	16-bit data	0 65535		
	unsignedInt	32-bit data	04.2e9		
	uint64	64-bit data	01.8e19		
	float	IEEE single-precision 32-bit floating point type	1.17e-38 3.4e38		
	double	IEEE double-precision 64-bit floating point type	2.2e-308 1.79e308		
	string Array of elements containing character code defined b encoding encoding				
	binary	Arbitrary binary data			
Arraysize	 In case of string tag, t string tag. 	his property represents the number of array his property represents the maximum num sponds to number of string characters if Er	ber of bytes available in the		
	UTF-8 or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character require 2 bytes.				
Conversion	Conversion to be applied to the tag.				
	Conversion				
	BCC AB- ABC ABC	>BA CD->CDAB CDEFGH->GHEFCDAB bits ABCD->CDAB Cancel OK			
	Depending on data type sele	ected, the list Allowed shows one or more	conversion types.		

Element Description

Value	Description
Inv bits	inv : Invert all the bits of the tag.
	<i>Example:</i> 1001 \rightarrow 0110 (in binary format) 9 \rightarrow 6 (in decimal format)
Negate	neg: Set the opposite of tag value.
	<i>Example:</i> 25.36 → -25.36
AB -> BA	swapnibbles: Swap nibbles in a byte.
	<i>Example:</i> 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)
ABCD -> CDAB	swap2: Swap bytes in a word.
	Example: 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)
ABCDEFGH ->	swap4: Swap bytes in a double word.
GHEFCDAB	<i>Example:</i> 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)
ABCNOP ->	swap8: Swap bytes in a long word.
OPMDAB	Example: 142.366 → -893553517.588905 (in decimal format) 0 10000000110 0001110010111011010000101101000011100101
	1 10000011100 1010101000010100010110110110
BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9)
	Example: $23 \rightarrow 17$ (in decimal format) $0001 \ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)

Element	Description
	If more conversions are configured, they will be applied in order (from top to bottom of list Configured).
	Use the arrow buttons to order the configured conversions.

Tag Import

Select the driver in Tag Editor and click on the **Import Tags** button to start the importer.

	Tags	×								
+	-	X	ß	ß	>]	Þ	A 9B	B>	63	1
Data			^		-	Та	g URI			

It is possible to import a Tag Editor exported xml

Туре	Description				
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.				
	Tags x				
	+ - 🎽 🕲 🔊 🚺 🥵 🖬 🗗				
	Data Tag URI				

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags × Protocols						•
+ - 🎽 🕲 🔌] [> 🗞 📦 📅	R 🔎 - Search 🍸 Filter by: Data	▼ Ite	ems u	used:6/10000 Protocol: Show all	🗹 Show all tags 🖉 🗐
Data	Туре	Comment	^	Pro	operty	Value
Modbus TCP:prot1	Container			~	Driver	
Model: Modicon Modbus(1-ba	iseu)				Model	Modicon Modbus(1-based)
- Holding Registers 1	unsignedShort				Protocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			~	Dictionary	
 Holding Registers 3 MRTU1 	unsignedShort unsignedShort				Array	false
- MRTU2	unsignedShort				Array size	0
MRTU3	unsignedShort				Arravindex.Subindex	400003
- MRTU4	unsignedShort				Comment	
- MRTU5	unsignedShort				Data type	unsignedShort
					Data type	unsignedation

Toolbar item	Description
ka	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
8	Update Tag(s) . Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag. Example of both checked and unchecked result:
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

JavaScript Interface

Beside Tag interface the user can access the protocol via JavaScript.

Although defined Tags can be accesses by JavaScript too, JavaScript can access directly to a Command interface implemented in protocol. This interface does not require the definition of Tags and is direct to protocol resulting in more efficiency.

The following commands are supported:

Command	Description
dir (node,path)	Get directory of node starting from path.
readFile (node,deviceFilePath,localFilePath)	Get file from node.
writeFile (node,deviceFilePath,localFilePath)	Write file to node.
deleteFile	Delete file into node.

Example of usage:

```
var tagMgr = project.getWidget("_TagMgr");
var protID = "prot2"; // to be set according to protocol numbering
var params = "0 /F@/file.ext /mnt/usbmemory/file.ext";
tagMgr.invokeProtocolCommand(protID, "writeFile", params, state);
```

SAIA S-BUS

The SAIA S-BUS communication driver has been designed to connect HMI devices to SAIA PLCs through serial connection.



HMIs from UN65 and UN70 platforms do not support PARITY mode on PLC configuration due hardware incompatibility.

DATA mode is supported in all HMI platforms.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the **PLC** list.

The protocol configuration dialog is displayed.

SAIA S-BUS		x				
PLC Network	Comm OK					
Node ID	1 Cancel					
Timeout (ms)	200					
Retry count	2					
data/parity protocol	data/parity protocol					
PLC Models		_				
PCD1 PCD2 PCD3						

Element	Description
Node ID SAIA PLC node on the serial network.	
Timeout (ms)	Time delay in milliseconds between two retries in case of missing response from the server device.
Retry count	Defines the number of times a certain message will be sent to the controller before reporting the communication error status.

Element	Description			
data/parity protocol	 SAIA protocol mode: unchecked (default): parity mode checked: data mode 			
PLC Models	SAIA PLC models available: • PCD1 • PCD2 • PCD3			
Comm	If clicked displays the communication parameters setup dialog.			
	Element Parameter			
	Port Serial port selection. • COM1: device PLC port. • COM2: computer/printer port on panels with serial ports or optional Plug-In module plugge on Slot 1/2 for panels with 1 serial port on-bo • COM3: optional Plug-In module plugged on 3/4 for panels with 1 serial port on-board. Baudrate Serial baudrate. Available speeds:	ed ard.		
	• 9600.			

Element	Description				
	Element Parameter				
		 19200. 38400. 57600. 			
	Mode	 Serial port mode. Available modes: RS-232. RS-485 (2 wires). RS-422 (4 wires). 			
PLC Network	Multiple controllers can be connected to one HMI device. To set-up multiple connections, select PLC network and click Add to configure each node				

Tag Editor Settings

Path: ProjectView> Config > double-click Tags

- 1. To add a tag, click +: a new line is added.
- 2. Select SAIA S-BUS from the Driver list: tag definition dialog is displayed.

SAIA S-BUS					×
SAIA S-BUS					
Memory Type	Offset	SubIndex			
R# Register -	0 📄	0 🔻	Arraysize		
1	boolean	•	0		
Conversion					
	+/-				
	Oł		Cancel	Apply	Help

Element	Description								
Memory Type	Memory Type	Description							
туре	R # -Register	unsigned 32 bit data register	unsigned 32 bit data register (default)						
	C # -Counter	unsigned 32 bit data counter (default)							
	T # -Timer	unsigned 32 bit data timer (c	unsigned 32 bit data timer (default)						
	F # -Flag	1 bit data flag	1 bit data flag						
	l # -Input	1 bit data input							
	O # -Output	1 bit data output							
	Data Block	unsigned 32 bit data block (d	default)						
	Real Time Clock	unsigned 8 bit real time cloc details)	unsigned 8 bit real time clock (default) (see Special Data Types for mode details)						
	Node Override	protocol parameter (see Spe	ecial Data Types for mode de	tails)					
Offset	Memory Type	Offset PCD1	Offset PCD2	Offset PCD3					
	R # -Register	0 – 4095	0 – 4095	0 – 16383					
	C # -Counter	0 – 1599	0 – 1599	0 – 1599					
	T # -Timer	0 – 1599	0 – 1599	0 – 1599					
	F # -Flag	0-8191	0-8191	0-8191					
	l # -Input	0-512	0-8192	0-5120					
	O # -Output	0-512	0-8192	0-5120					
	Data Block	0 – 3333	0 – 3333	0 – 16383					
	Real Time Clock	1-8	1-8	1-8					
	Node Override	0	0	0					
SubIndex	This allows resource	offset selection within the regi	ster.						
Data Type	Available data types:								
	• boolean								
	• byte								
	 short int 								
	unsignedByte	9							
	unsignedShort								
	unsignedInt								

Element	Description					
	float string See "Programming concepts" section in the main manual. Note: To define arrays, select one of Data Type format followed by square brackets.					
Arraysize	 In case of array tag, this property represents the number of array elements. In case of string tag, this property represents the maximum number of bytes available in the string tag. Note: number of bytes corresponds to number of string characters if Encoding property is set to UTF-8 or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2 bytes. 					
Conversio n	Conversion to be appl Conversion inv,swap2	Allowed BCD AB->BA ABCD->CDAB ABCD=FGH->GHEFCDAB Inv bits Cancel OK Cancel OK				
	Value	Description				
	Inv bits	inv: Invert all the bits of the tag. <i>Example:</i> $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)				
	Negate	neg : Set the opposite of tag value. <i>Example:</i> $25.36 \rightarrow -25.36$				
	AB -> BA	swapnibbles : Swap nibbles in a byte. <i>Example:</i> $15D4 \rightarrow 514D$ (in hexadecimal format)				

Element	Description				
	Value	Description			
		$5588 \rightarrow 20813$ (in decimal format)			
	ABCD -> CDAB	swap2: Swap bytes in a word.			
		Example: 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)			
	ABCDEFGH -> GHEFCDAB	swap4: Swap bytes in a double word.			
		Example: $32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format) $855441236 \rightarrow 1426062386$ (in decimal format)			
	ABCNOP -> OPMDAB	swap8: Swap bytes in a long word. Example: 142.366 \rightarrow -893553517.588905 (in decimal format) 0 10000000110 00011100101110110010001011101000011100101			
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)			
	Select conversion and	click +. The selected item will be added to list Configured .			
	If more conversions ar	e configured, they will be applied in order (from top to bottom of list Configured).			
	Use the arrow buttons	to order the configured conversions.			

Real Time Clock

The protocol provides the special data type Real Time Clock which allows you to change the date and time on PLC. This memory type is an unsigned byte.

Offset	Description
1	Number of week
2	Day of week

Offset	Description
3	Year
4	Month
5	Day
6	Hours
7	Minutes
8	Seconds

SAIA S-BUS		X
SAIA S-BUS		
Memory Type Real Time Clock	Offset SubIndex	
Data Block	Data Type Arraysize unsignedByte 0	
Conversion	+/-	
	OK Cancel Apply Help	

It is also possible to use the Date/Time control widget to directly write in Real Time Clock variable.

- 1) Define a Real Time Clock, as per above picture
- 2) Drag and drop the Date/Time control widget

		<u> </u>
	Widget Gallery 🔸 🗙	Ŵ
~	Basic	
	Controls 🔹	dget
	05/30/16 - 09:57:25	Widget Gallery
	05/30/16 - 09:57:25	
	Combo Box	

3) From Property Pane, click on the + button beside **Value** property. Then locate the Real Time Clock variable from Tag source, and select Read/Write option.

DataLink System Time:_SysPropMgr Number Forma MM/DD/YY - hh:mm:ss Time Spec local OnDataUpdate . 05/30/16 - 10:01:17 .		👻 Pr	operties			ф ,
Data Type Tag name Potecol Show all age Protocol Show al		5] 🐺 🤞	2		
Image: Source: Image: Type Tag Image: Source: Image: Source: <th></th> <th>- 0</th> <th>DateTime :</th> <th>dateTime1</th> <th></th> <th></th>		- 0	DateTime :	dateTime1		
DataLink System Time:_SysPropMgr Number Forms HM/DD/W* hh:mm:ss Time Spec Iccal OnDataLipdate Image: Source: © Tag Alas Source: Image: Source: Image: Source: Image: Source: <	D==		Value	-1		F
Number Forma HM/DD/V*- th:mm:ss Time Spec local OD5/30/16 - 10:01:17 Image: Special contained in the			Datal ink		1ar	
Time Spec local OnDataUpdate Itext Itext Text Itext Itext						
OnDataUpdate OnDataUpdate OnDataUpdate Text OnDa					55	
O5/30/16 - 10:01:17 dateTimeLvalue Source: @ Tag Alas System Widget Recipe Protocol: Show all tags Property Value Adate Type Tag name Model: Protocol: Show all tags Property Value Active faise Comment Data Type unsignedByte Deta VW Rate So Scaling None						
05/30/16 - 10:01:17 idateTimeLvalue Source: • Tag • Alas • System • Widget • Recipe • Filter by: Data • Protocol: Show all • I' Show all tags • Order • Model: PCD1 • Protocol: Show all • Order • Model: PCD1 • Protocol: Show all • Order • Model: PCD1 • Protocol: SALA S-BUSynot1 • Trier • Model: PCD1 • Protocol: SALA S-BUSynot1 • Tag • Active • false Comment • Defailt value • Filter ty: • Defailt value • Protocol: Sala S-BUSynot1 • Tag • Active • false Comment • Defailt value • Protocol: Sala S-BUSynot1 • Tag • Active • false Comment • Defailt value • Protocol: Sala S-BUSynot1 • Tag • Real Time Clock • Protocol: Sala S-BUSynot1 • Tag • Active • false Comment • Defailt value • Protocol: Sala S-BUSynot1 • Tag • Real Time Clock • Pictocol: Sala S-BUSynot1 • Tag • Active • false Comment • Defailt value • Pictocol:			·	te		
Source:	05/30/16 - 10:01:17	÷	Text			
P- Search ▼ Fiter by: Data Protocol: Show all ▼ ♥ Show all tags Data Type Tag name Property Value SALA S-8USignot1 Container Model PCD 1 Protocol SALA S-8USignot1 Protocol SALA S-8USignot1 Model PCD 1 Protocol SALA S-8USignot1 Protocol SALA S-8USignot1 Protocol SALA S-8USignot1 Maximum Maximum Min value Min value Min value Min value Min value	dateTime1.value					
Data Type Tag name Model: PCD1 Container Real Time Clock unsignedByte Real Time Clock Property Value Active false Comment Data Type unsignedByte Time Clock unsignedByte Real Time Clock Protocol SLA S-BUSynot1 Protocol SLA S-BUSynot1 Protocol Active false Comment Data Type UnsignedByte Defailt value Encoding Groups Max value Min value Min value Pic Ctag name R/W R/W Rate 500 Scaling None	jource: 💿 Tag 🔘 Alias 🔘 System 🔘 Widget 🔘 Recipe					
SAIA 5-8US;prot1 Container Model: PCD1 Protocol SAIA 5-8US;prot1 Protocol SAIA 5-8US;prot1 Protocol SAIA 5-8US;prot1 Protocol SAIA 5-8US;prot1 Protocol Yet Group Active false Comment Data Type Data Type unsignedByte Encoding Groups Max value Min value Min value Pic C tag name R_/W R_/W Rate 500 Scaling None	P- Search Titter by: Data	•	Protocol: Sho	w all 🔻	Show all tags	
Model: PCD1 Container Model: PCD1 Protocol SALAS-BUS:prot1 Table T	Data Tyroe Tag name					
Model PDD1 Protocol SAIA S-BUS;prot1 ✓ Tag Active false Comment Data Type unsignedByte Default value Encoding Groups Max value PLC tag name R_/W R_Ate S00 Scaling None				Property	Value	
Protocol SALA - Boosprot 1 Protocol SALA - Boosprot 1 Active false Comment Data Type unsignedByte Default value Encoding Groups Max value PLC tag name R/W Rate S00 Scaling None	SAIA S-BUS:prot1				Value	
Active false Comment Data Type unsignedByte Default value Encoding Groups Max value Min value PLC tag name R_/W R_Ate 500 Scaling None	SAIA S-BUS:prot1 Model: PCD1 Container			4 Driver		
Comment Data Type UnsignedByte Encoding Groups Max value Min value PLC tag name R_VW Rate S00 Scaling None	SAIA S-BUS:prot1 Model: PCD1 Container			Driver Model	PCD1	
Data Type unsignedByte Defrault value Encoding Groups Max value Min value PLC tag name R_/W R_ate 500 Scaling None	SAIA S-BUS:prot1 Model: PCD1 Container			Driver Model Protocol	PCD1	
Default value Encoding Groups Max value Min value PLC tag name R_/W R_/W R_ate 500 Scaling None	SAIA S-BUS:prot1 Model: PCD1 Container			Driver Model Protocol A Tag	PCD1 SAIA S-BUS:prot1	
Encoding Groups Max value Min value PLC tag name R/W R/W Rate 500 Scaling None	SAIA S-BUS:prot1 Model: PCD1 Container			Driver Model Protocol Tag Active	PCD1 SAIA S-BUS:prot1	
Groups Max value Min value PLC tag name R/W R/W Rate 500 Scaling None	SAIA S-BUS:prot1 Model: PCD1 Container			Driver Model Protocol Tag Active Comment	PCD1 SAIA S-BUS:prot1 false	
Max value Min value PLC tag name R,WV R,WV Rate 500 Scaling None	SAIA S-BUS:prot1 Model: PCD1 Container			Driver Model Protocol Tag Active Comment Data Type	PCD1 SAIA S-BUS:prot1 false	
Min value PLC tag name R/W R/W Rate 500 Scaling None	SAIA S-BUS:prot1 Model: PCD1 Container			Driver Model Protocol Tag Active Comment Data Type Default value	PCD1 SAIA S-BUS:prot1 false	
PLC tag name R/W R/W Rate 500 Scaling None	SAIA S-BUS:prot1 Model: PCD1 Container			Driver Model Protocol Tag Active Comment Data Type Default value Encoding	PCD1 SAIA S-BUS:prot1 false	
R/W R/W Rate 500 Scaling None	SAIA S-BUS:prot1 Model: PCD1 Container			Driver Model Protocol Tag Active Comment Data Type Default value Encoding Groups	PCD1 SAIA S-BUS:prot1 false	
R/W R/W Rate 500 Scaling None	SAIA S-BUS:prot1 Model: PCD1 Container			Driver Model Protocol Tag Active Comment Data Type Default value Encoding Groups Max value	PCD1 SAIA S-BUS:prot1 false	
Rate 500 Scaling None	SAIA S-BUS:prot1 Model: PCD1 Container			Driver Model Protocol Tog Active Comment Default value Encoding Groups Max value Min value	PCD1 SAIA S-BUS:prot1 false unsignedByte	
Scaling None	SAIA S-BUS:prot1 Model: PCD1 Container			Driver Model Protocol Tag Active Comment Data Type Default value Encoding Groups Max value Min value PC Ctag name	PCD1 SATA S-BUS:prot1 false unsignedByte	
	SAIA S-BUS:prot1 Model: PCD1 Container			A Driver Model Protocol A Tag Active Comment Data Type Default value Encoding Groups Max value Min value PLC tag name R/W	PCD1 SAIA S-BUS:prot1 false unsignedByte	
	SAIA S-BUS:prot1 Model: PCD1 Container			Driver Model Protocol Tag Active Comment Data Type Default value Encoding Groups Max value Min value PLC tag name R/W Rate	PCD 1 SATA S-BUS:prot1 false unaignedByte R/W S00	

Node Override

The protocol provides the special data type Node Override which allows you to change the node ID of the slave at runtime. This memory type is an unsigned byte.

The node Override is initialized with the value of the node ID specified in the project at programming time.

Node Override	Description
0	Communication with the controller is stopped. In case of write operation, the request will be transmitted without waiting for a reply.
1 to 254	It is interpreted as the value of the new node ID and is replaced for runtime operation.
255	Communication with the controller is stopped; no request messages are generated.



Note: Node Override ID value assigned at runtime is retained through power cycles.

Memory Type Offset SubIndex Node Override O O O Data Block Data Type Arraysize 1 unsignedByte O Conversion +/-	SAIA S-BUS	
1 unsignedByte 0 Conversion		
	Conversion	+/-

Communication Status

The current communication status can be displayed using the dedicated system variables. Please refer to the User Manual for further information about available system variables and their use.

The codes supported for this communication driver are:

Error	Notes
NAK	Returned in case the controller replies with a not acknowledge
Timeout	Returned when a request is not replied within the specified timeout period; ensure the controller is connected and properly configured for communication
Line Error	Returned when an error on the communication parameter setup is detected (parity, baud rate, data bits, stop bits); ensure the communication parameter settings of the controller is compatible with panel communication setup
Invalid response	The panel did receive from the controller a response, but its format or its contents is not as expected; ensure the data programmed in the project are consistent with the controller resources

SAIA S-BUS ETH

The SAIA S-BUS ETH communication driver has been designed to connect HMI devices to SAIA PLCs through ethernet connection.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In Config node double-click Protocols.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the PLC list.

The protocol configuration dialog is displayed.

SAIA S-BUS ETH		×
PLC Network		ОК
IP address	0.0.0.0	Cancel
Port	5050]
Slave ID	0]
Timeout (ms)	1000]
PLC Models		
PCD3		

Element	Description
IP address	Ethernet IP address of the controller.
Port	Port number used by the driver. The default value is 5050 .
Slave ID	ID if the controller.
Timeout (ms)	Time delay in milliseconds between two retries in case of missing response from the server device.

Element	Description
PLC Models	SAIA PLC models available:
	• PCD3
PLC Network	Multiple controllers can be connected to one HMI device. To set-up multiple connections, select PLC network and click Add to configure each node

Tag Editor Settings

Path: ProjectView> Config > double-click Tags

- 1. To add a tag, click +: a new line is added.
- 2. Select SAIA S-BUS ETH from the Driver list: tag definition dialog is displayed.

SAIA S-BUS ETH		x
SAIA S-BUS ETH		
Memory Type R# -Register -	Offset SubIndex	
Data Block	Data Type Arraysize unsignedInt ● 0	
Conversion	+/-	
	OK Cancel Apply Help	

Element	Description				
Memory Type	Memory Type	Description			
Type	R # -Register	unsigned 32 bit data register (default)			
	C # -Counter	unsigned 32 bit data counter (default)			
	T # -Timer	unsigned 32 bit data timer (default)			
	F # -Flag	1 bit data flag			
	l # -Input	1 bit data input			
	O # -Output	1 bit data output			
	Data Block	unsigned 32 bit data block (default)			
	Real Time Clock	unsigned 8 bit real time clock (default) (see Special Data Types for mode details)			
Offset	Memory Type Offset				
	R # -Register	0 – 16383			
	C # -Counter	0 – 1599			
	T # -Timer	0 – 1599			
	F # -Flag	0-8191			
	l # -Input	0-5120			
	O # -Output	0-5120			
	Data Block	0 – 16383			
	Real Time Clock	1-8			
SubIndex	This allows resource	offset selection within the register.			
Data Type	Available data types:				
	• boolean				
	• byte				
	 short int 				
	unsignedByte				
	unsignedShounsignedInt				
	float				
	• string				
	• string				

Element	Description						
	See "Programming concepts" section in the main manual.						
	Note: To define arrays, select one of Data Type format followed by square brackets.						
Arraysize		ay tag, this property represents the number of array elements.					
	• In case of string tag.	ng tag, this property represents the maximum number of bytes available in the					
	Note: number of bytes corresponds to number of string characters if Encoding property is set to UTF- or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2 bytes.						
Conversio	Conversion to be ap	plied to the tag.					
n	Conversion						
	inv,swap2	Allowed Configured					
		BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits					
		Cancel OK					
	Depending on data type selected, the list Allowed shows one or more conversion types.						
	Value	Description					
	Inv bits	inv : Invert all the bits of the tag.					
		Example: $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)					
	Negate	neg: Set the opposite of tag value.					
		<i>Example:</i> 25.36 → -25.36					
	AB -> BA	swapnibbles: Swap nibbles in a byte.					
		<i>Example:</i> 15D4 \rightarrow 514D (in hexadecimal format)					
		$5588 \rightarrow 20813$ (in decimal format)					

Element	Description				
	Value	Description			
		Example: $9ACC \rightarrow CC9A$ (in hexadecimal format) $39628 \rightarrow 52378$ (in decimal format)			
	ABCDEFGH -> GHEFCDAB	swap4 : Swap bytes in a double word. <i>Example:</i> $32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format) $855441236 \rightarrow 1426062386$ (in decimal format)			
	ABCNOP -> OPMDAB	swap8: Swap bytes in a long word. Example: 142.366 \rightarrow -893553517.588905 (in decimal format) 0 1000000110 0001110010111011010000101101000011100000			
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)			
	Select conversion and	click +. The selected item will be added to list Configured .			
	If more conversions a	re configured, they will be applied in order (from top to bottom of list Configured).			

Use the arrow buttons to order the configured conversions.

Real Time Clock

The protocol provides the special data type Real Time Clock which allows you to change the date and time on PLC. This memory type is an unsigned byte.

Offset	Description
1	Number of week
2	Day of week
3	Year
4	Month

Offset	Description
5	Day
6	Hours
7	Minutes
8	Seconds

AIA S-BUS ETH	×
Memory Type Real Time Clock 🔹	Offset SubIndex
Data Block	Data Type Arraysize unsignedByte 0
Conversion	+/-
	OK Cancel Apply Help

It is also possible to use the Date/Time control widget to directly write in Real Time Clock variable.

1) Define a Real Time Clock, as per above picture

2) Drag and drop the Date/Time control widget

		x
	Widget Gallery 🕒 🎽	Ŵ
-	Basic	
	Controls	idge
	05/30/16 - 09:57:25	Widget Gallery
	05/30/16 - 09:57:25	
	Combo Box	
E		

3) From Property Pane, click on the + button beside **Value** property. Then locate the Real Time Clock variable from Tag source, and select Read/Write option.

•	Pro	operties				ф×	Ī
▼ E ▼ ∰ ፼ ▼ H ▼	5) 🔂 🚳					
		DateTime : d	ateT	ïme1			
E =		Value	-1			+	
A		DataLink		tem Time:_SysPropl	Mar	-	
				1/DD/YY - hh:mm:			1
					55		
		Time Spec	loc	al			
		OnDataUpdat				+	
	÷	Text					
05/30/16 - 10:01:17 🔶							
dateTime1.value						- 2	3
Jacenmeilvalue							
Source: 🛛 🔘 Tag 🔘 Alias 🔘 System 🔘 Widget 🔘 Recipe							
P- Search Tilter by: Data	•	Protocol: Show		•	Show all tags	F	
Price by: Data		Protocol: Show	V dil	•	Show all tags		-
Data Type Tag name				Property	Value		4
SAIA S-BUS:prot1 Container				A Driver			
Model: PCD1			_	Model	PCD1		
Real Time Clock unsignedByte Real Time Clock				Protocol	SAIA S-BUS:prot1		
				▲ Tag			
			5	Active	false		
			5	Comment			
				Data Type	unsignedByte		1
				Default value			
			6	Encoding			
			1	Groups			
			ľ	Max value			
				Min value			
				PLC tag name			ł
				R/W	R/W		
				Rate	500		
				Scaling	None		
Read Only • Read/Write • Write Only Items used: 1/10000 Ar	ray i	ndex 0				A V	ł

Communication status

Current communication status can be displayed using system variables. See "System Variables" section in the main manual.

Codes supported for this communication driver:

Error	Cause	Action
NAK	The controller replies with a not acknowledge.	-
Timeout	A request is not replied within the specified timeout period.	Check if the controller is connected and properly configured to get network access.
Invalid response	The device did received a response with invalid format or contents from the controller.	Check if the data programmed in the project are consistent with the controller resources.
General Error	Unidentifiable error. Should never be reported.	Contact technical support.

Simatic S7 PPI

HMI devices can be connected to the Siemens Simatic S7-200 family of PLCs. The communication is performed via the PLC programming ports using the PPI and the PPI+ protocols.

This document describes the PPI+ protocol and includes the information needed for a successful connection.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the **PLC** list.

The protocol configuration dialog is displayed.

Comm OK
1 Cancel
2
1
1000
×
-

Element	Description
PLC Network	Enable access to multiple networked controllers. For every controller (slave) set the proper option.
Panel ID	Node number of the operator panel.
Slave ID	Node number of the connected PLC.
Max ID	Available only if PPI+ protocol is in use. Contains the highest node number in PPI+ network.

Element	Description
PPI+	Checked to use PPI+ protocol instead of PPI protocol.
Timeout (ms)	Time delay in milliseconds between two retries of the same message when no answer is received from the controller.

Element	Description							
PLC Models	Several Siemens controllers are su software for a complete list of supp	pported. Please check directly in the programming IDE orted controllers.						
Comm	If clicked displays the communication parameters setup dialog.							
	Comm Parameter Dialog							
		ОК						
	Port com1	▼						
	Baudrate 9600	•						
	Parity even	•						
	Data bits 8	•						
	Stop bits 1	•						
	Mode RS-485	•						
	Element	Parameter						
	Port	Serial port selection.						
		On UN20:						
		COM1: device PLC port.						
		COM2: PC/printer port						
		On UN31 or UN30:						
		 COM1: integrated serial port COM2: optional module plugged on Slot 1/2 						
		COM3: optional module plugged on Slot 3/4						
	Baudrate, Parity, Data Bits, Sto	p bits Serial line parameters.						
	Mode	Serial port mode. Available modes:						
		• RS-232.						
		• RS-485 (2 wires).						
		• RS-422 (4 wires).						

_

Tag Editor Settings

Simatic S7 PPI × Simatic S7 PPI Offset SubIndex Memory Type • Variable memory • 0 0 ÷ Data Type Arraysize Conversion Τ short • 0 +/-OK Cancel Apply Help

In the Tag Editor select Simatic S7 PPI from the list of defined protocols and click + to add a tag.

Element	Description
Memory Type	Area of PLC where tag is located.
Offset	Offset address where tag is located.
SubIndex	In case of Boolean data type, this is the offset of single bit.
Data Type	Available data types:
	 boolean byte short int unsignedByte unsignedShort unsignedInt float string See "Programming concepts" section in the main manual.
Arraysize	 In case of array tag, this property represents the number of array elements. In case of string tag, this property represents the maximum number of bytes available in the string tag. Note: number of bytes corresponds to number of string characters if Encoding property is set to UTF-8 or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character

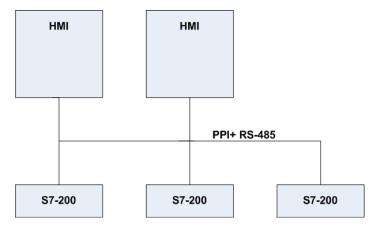
Element	Description						
	requires 2 bytes.						
Conversion	Conversion to be applied to the tag.						
	Conversion						
	inv,swap2	Allowed Configured					
		BCD AB->BA ABCD->CDAB ABCD->CDAB					
		ABCDEFGH->GHEFCDAB Inv bits					
		Cancel OK					
	Depending on da	ata type selected, the list Allowed shows one or more conversion types.					
	Value	Description					
	Inv bits	inv : Invert all the bits of the tag.					
		Example: 1001 \rightarrow 0110 (in binary format) 9 \rightarrow 6 (in decimal format)					
	Negate	neg : Set the opposite of tag value.					
		<i>Example:</i> 25.36 → -25.36					
	AB -> BA	swapnibbles: Swap nibbles in a byte.					
		<i>Example:</i> 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)					
	ABCD ->	swap2: Swap bytes in a word.					
	CDAB	<i>Example:</i> 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)					
	ABCDEFGH -	swap4: Swap bytes in a double word.					
	>	Example:					
	GHEFCDAB	$32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format) $855441236 \rightarrow 1426062386$ (in decimal format)					

Element	Description	
	Value	Description
		$\begin{array}{c} 142.366 \rightarrow -893553517.588905 \mbox{ (in decimal format)} \\ 0 \ 10000000110 \\ 0001110010111011010000101101000011100101$
	BCD	bcd: Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)
		on and click +. The selected item will be added to list Configured .
	Use the arrow b	uttons to order the configured conversions.

PPI+ Connectivity

HMI devices can be connected to more than one CPU S7-200, more than one operator panel can also be connected to the same PLC.

Operator panels will not interfere with PPI+ communication between the PLC's.



PPI+ protocol allows you to use more complex configurations than the standard PPI protocol.

Each PLC can execute read and write operations to and from other PLCs. At the same time more than one panel can be connected on the PPI network and can access all the variables from all the PLCs.

PLC programming software can be used and online programming can be performed without interfering with the panel-PLC communication .

Communication Status

Current communication status can be displayed using System Variables. See "System Variables" section in the main manual.

Codes supported for this communication driver:

Error	Cause	Action
NAK	The controller replies with a not acknowledge.	-
Timeout	A request is not replied within the specified timeout period.	Check if the controller is connected and properly configured to get network access.
Invalid response	The device did received a response with invalid format or contents from the controller .	Ensure the data programmed in the project are consistent with the controller resources.
General Error	Unidentifiable error. Should never be reported.	Contact technical support.

Siemens S7 Optimized

Siemens S7 Optimized communication driver has been designed to communicate with Siemens PLCs through Ethernet connection.

PLC must either have an on-board Ethernet port or be equipped with an appropriate Ethernet interface (either built-in or with a module).

This communication driver allows communication with PLCs which have been programmed using optimized Data Blocks.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In Config node double-click Protocols.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the **PLC** list.

The protocol configuration dialog is displayed.

Siemens S7 Optimized		×
PLC Network		ОК
Alias		Cancel
IP address	0.0.0.0	
Slot	2	
Timeout	1000	
PLC Models		
S7-1200/1500		

Element	Description
Alias	Name identifying nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node.
IP	Ethernet IP address of PLC.

Element	Description	
address		
Slot	Number of the slot	where the CPU is mounted.
PLC Models	List of compatible	PLCs.
PLC Network	Enable access to r	nultiple networked PLCs. For every PLC set the proper option. ized K K K K K K K K K K K K K K K K K K

S7-1200 and S7-1500 PLC configuration

S7-1200 (starting from firmware version 4.0) and S7-1500 PLC Series from Siemens have a built-in firewall; by default the maximum protection level is enabled. To establish communication with these PLC models it is necessary to enable S7 communication with 3rd party devices; this setting is available in TIA Portal programming software.

- 1. Open the PLC project in TIA Portal.
- 2. Select the PLC from the project tree and open PLC Properties.

:	·			s Window	Help		i		
	🛉 🎦 🔚 Save project ا 🍶	(🗉 ū	X 🎝 Ŧ (🔍 ± 🖬 🖥		16 🖳	🕞 🗯	🍠 Go o	nline 🧯
	Project tree		Project15	00 → PLC_1	1 [CPU	1511-	1 PN]		
	Devices								
	B O O E	1	Hr PLC_1	1		-		6 🗄	€ ±
brks									
& networ	🔻 🔄 Project1500	^			0	1	2	3	4
n n	💕 Add new device			Dell O	-		-	-	
Devices 8	Bevices & networks The plc_1 [CPU 1511-1 PN]			Rail_0		LANKIN			
e i	🛐 Device configuratio	Open			Ext	oort moo	lule lab	elina st	rips
	😼 Online & diagnostic	Open ir	new editor		_	_			·
	🕨 🔂 Program blocks	X Cut		Ctrl+X	Pro Pro	perties.)	Alt+	Enter
	🕨 🙀 Technology objects	🗓 Сору		Ctrl+C	-				
	External source file:	💼 Paste		Ctrl+∨					
	🕨 🔁 PLC tags	🗙 Delete		Del					
	🕨 🛅 PLC data types	Rename	e	F2					
	🕨 🥅 Watch and force tab	Sector to	pology view						
	Straces 🗹		etwork view						
	📴 Program info								
	🖂 PLC alarms	Compile		•					
	🔄 Text lists		ad to device	•					
	🕨 🛅 Local modules	🝠 Go onlii		Ctrl+K					
	🕨 🙀 Common data	🖉 Go offlir		Ctrl+M					
	Documentation setting		& diagnostic:	s Ctrl+D					
	🕨 🐻 Languages & resource	🖳 Start sii	nulation	Ctrl+Shift+X					
	🕨 🔚 Online access	🐴 Compa	e	•					
	🕨 🣴 Card Reader/USB memory	••••••••••••••••••••••••••••••••••••••		F 4 4					

3. In General > Protection choose a permission between the top three (make sure that the tick is present on HMI column).

General IO tags Syste	m con	stants	Texts					
General PROFINET interface [X1] General	^		ection					_
Ethernet addresses	_	PIC	Juction					
Time synchronization	- 11	5.	elect the access level for the PLC.					
Operating mode	- 11		need the beceasile verifier the rec.					
Advanced options								
Web server access			Access level		Access		Access permi	
Hardware identifier				HMI	Read	Write	Password	
DI 8/DQ 6			Full access (no protection)	✓	 ✓ 	✓	*	
AI 2	•		Read access	×	✓			
High speed counters (HSC)			HMI access	~				
Pulse generators (PTO/PWM)			 No access (complete protection) 					
Startup								
Cycle								
Communication load		Fu	Ill access (no protection):					
System and clock memory	_	π/	A Portal users and HMI applications will have	e access to all fun	ctions.			
Web server		No	password is required.					
User interface languages								
Time of day								
Protection								
Configuration control	~							

Note: If "No access" is selected, the communication with the panel will not be established.

4. Scroll down the page and check "Permit access with PUT/GET communication from remote partner".

PLC	_1 [CPU 1511-1	PN]	×
[General 10) tags Texts	
•	General		^
•	PROFINET inte		
	Startup		
	Cycle	Connection mechanisms	
	Communicati		
	System and clo	Permit access with PUT/GET communication from remote partner (PLC, HMI, OPC,)	
•	System diagn		
•	Web server		
	Display		
	User interface I		
	Time of day		
	Protection		
•	System powe		
	Connection res		-
	Overview of a	l I I I I I I I I I I I I I I I I I I I	
			~
<			
		OK Cancel	

Note: If variables are defined in "Program blocks", DB must configured as "Optimized".

To check or change DB optimization, open DB Properties:

i

😼 Online & diagno	DSTICS	рірро
▼ → Program blocks		pluto
Add new block 7		<add new=""></add>
💁 Main [OB1]		
DB2 [DB2]		
DB4 [DB4	Open	
	Generate source from blocks	5
External sou		
🕶 🚂 PLC tags 👘	Snapshot of the monitor valu	Jes
🔄 Show all	Apply snapshot values as sta	art values 🕨
📑 Add new 💥	Cut	Ctrl+X
💐 Default ta 🧾		Ctrl+C
🖳 Tag table 💼	Paste	Ctrl+V
💌 💽 PLC data typ	Copy as text	
📑 Add new		Del
🖪 my_data 🗙	Delete Rename	F2
🥵 my_data		FZ
🕨 🕨 📷 Watch and f	Compile	•
📴 Program info	Download to device	• •
🛅 Text lists 🛛 🐣	Go online Go offline	Ctrl+K
🕨 🍺 Local modu		Ctrl+M
🕨 🏹 Common data	Cross-reference information	Shift+F11
🔹 🕨 🛅 Documentation		F11
	Call structure	
🕨 🖬 Online access 📃 🛄	Assignment list	
 Details view 	Switch programming langua	ige 🕨 🕨
	Know-how protection	
Name	Print	Ctrl+P
<u>م</u>	Print preview	
<	Properties	AltNEnter
Portal view	inopenies	AR AR

In General > Attributes check "Optimized block access":

2 [DB2]	
General	
General Information Time stamps Compilation Protection Attributes	Attributes Only store in load memory Data block write-protected in the device Optimized block access
	OK Cancel

If check box "Optimized block access" is not available (grayed-out) it could be because DB is an "instance DB" linked to an "optimized access FB".

After compiling the project, tag offsets will be shown close to variable name.

These settings can be applied to TIA Portal programming software, S7-1200 PLC family starting from PLC firmware version 4.0 and S7-1500 PLC family.

Direct Import of TIA Portal project

It is possible to import TIA Portal variables directly from TIA Portal project, by selecting "TIA Portal Project v12 or newer" from import selection (refer to "Tag Import" chapter).

Data Blocks must be set as Optimized:

- 1. Configure the Data Block as **Optimized**.
- 2. Right-click on the Data Block and choose Properties:

😼 Online & diagno	stics	
💌 🖛 Program blocks		
📑 Add new blo	ck	
💁 Main [OB1]		
j DB2 (DB2	Open	
🧧 DB4 [DB4	•	
🔹 🕨 🙀 Technology 🐇	Cut	Ctrl+X
🕨 🔄 External sou 🚞	Сору	Ctrl+C Ctrl+V
🔻 浸 PLC tags 🛛 💻	Paste	CEN+V
🝇 Show all	Copy as text	
📑 Add new 🗙	Delete	Del
💐 Default ti	Rename	F2
🍕 Tag table	Compile	•
🔻 💽 PLC data typ	Download to device	, i i i i i i i i i i i i i i i i i i i
📑 Add new 💋	Go online	Ctrl+K
	Go offline	Ctrl+M
🕨 🕨 🔜 Watch and f	Snapshot of the monitor valu	es
Online back	Apply snapshot values as sta	
🕨 🔀 Traces		
· • • • • • • • • • • • • • • • • • • •	Generate source from blocks	
📴 Program info	Cross-reference information	Shift+F11
	Cross-references	F11
	Call structure	
	Assignment list	
Documentation	Switch programming langua	ge 🕨
Languages & re Online access	Know-how protection	
	Print	Ctrl+P
	Print preview	CLITT
9	Properties	Alt+Enter

3. In the General tab select Attributes and select Optimized block access.

082 [D82]	\$
General	
General Information Time stamps Compilation Protection Attributes	Attributes Only store in load memory Data block write-protected in the device Optimized block access
	OK Cancel



Note: If the options **Optimized block access** is not enabled (checkbox grayed out) this might mean that the Data Block is an "instance DB" linked to an "optimized access FB".

Tag Editor Settings

In the Tag Editor select "Simatic S7 ETH" from the list of defined protocols and click + to add a tag.

Siemens S7 Optimized	×
Siemens S7 Optimized	
Access Type PLC variable	Data Type Arraysize ✓ UnsignedByte ✓
Conversion	Variable name
	OK Cancel Apply Help

Element	Description			
Memory	Area of PLC where tag is located.			
Туре	Type Description PLC variable Variables im from TIA Point		on	
			•	
	Node Override IP Check "Spe type" chapter			
Data Type	Data Type Memory Space		Limits	
Type	boolean	n 1-bit data		01
	byte	8-bit data		-128 127
	short	16-bit data		-32768 32767
	int	32-bit data		-2.1e9 2.1e9
	unsignedByte 8-bit data			0 255
	unsignedShort	16-bit data		0 65535
	unsignedInt	32-bit data		0 4.2e9
	float	IEEE single-precision 32- point type	bit floating	1.17e-38 3.4e38
	double	IEEE double-precision 64 point type	-bit floating	2.2e-308 1.79e308
	string	Array of elements contain selected encoding	ing character	code defined by

	Description		
Arraysiz)	 In case of array tag, this property represents the number of array elements. In case of string tag, this property represents the maximum number of bytes a in the string tag. Note: number of bytes corresponds to number of string characters if Encoding properts to LUTE 0 and stind in Tag Editor. 		
	to UTF-8 or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2 bytes.		
Conversi	Conversion to be a	applied to the tag.	
on	Conversion		
	inv.swap2 Allowed Configured BCD AB->BA ABCD->CDAB ABCD=>CDAB ABCD->CDAB ABCDEFGH->GHEFCDAB Cancel Inv bits Cancel OK OK Depending on data type selected, the Allowed list shows one or more conversions, lister below. Value Description		
	Value	Description	
	Value Inv bits	Description Invert all the bits of the tag.	
		Invert all the bits of the tag. <i>Example:</i> $1001 \rightarrow 0110$ (in binary format)	
	Inv bits	Invert all the bits of the tag. <i>Example:</i> $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)	
	Inv bits	Invert all the bits of the tag. $Example:$ $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)Set the opposite of the tag value. $Example:$	

Element	Description	Description	
	Value	Description	
	ABCDEFGH -> GHEFCDAB	Swap bytes of a double word. <i>Example:</i> $32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format) $855441236 \rightarrow 1426062386$ (in decimal format)	
	ABCNOP -> OPMDAB	Swap bytes of a long word. Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) $0\ 10000000110$ 0001110010111011001000101101000011100100101	
	BCD	Separate the byte in two nibbles, and reads them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)	
	Select the conversion Configured list.	on and click on plus button. The selected item will be added on	
	If more conversions Configured list).	are configured, they will be applied in order (from top to bottom of	
	Use the arrow butto	ns to order the configured conversions.	

Node Override IP

The protocol provides the special data type Node Override IP which allows you to change the IP address of the target controller at runtime.

This memory type is an array of 4 unsigned bytes, one per each byte of the IP address.

The Node Override IP is initialized with the value of the controller IP specified in the project at programming time.

Node Override IP	PLC operation
0.0.0.0	Communication with the controller is stopped, no request frames are generated anymore.
Different from 0.0.0.0	It is interpreted as node IP override and the target IP address is replaced runtime with the new value.

If the HMI device is connected to a network with more than one controller node, each node has its own Node Override IP variable.



Note: Node Override IP values assigned at runtime are retained through power cycles.

Hostname DNS or mDNS

In addition to the array of bytes, string memory type can be selected to be able use the DNS or mDNS hostname as an alternative to the IP Address.

Siemens S7 Optimized		×
Siemens S7 Optimized		
Access Type Node Override IP	Data Type	Arraysize
Conversion	Variable name	
		ancel Apply Help

Tag Import

Select the driver in Tag Editor and click on the **Import Tags** button to start the importer.



The following dialog shows which importer type can be selected.

HMIStudio	×
Multiple tag importers are a	available for this protocol. Please select the importer type and continue.
Version	Туре
TIA Portal Project v12 or newer	Linear
Tag Editor exported xml 1.1	General
138 Periot v10, v11, v12	
	OK Cancel

Importer	Description	
TIA Portal Project v12 or newer Linear	Allows to import the whole TIA Portal project file using .apxx file (where "xx" is the TIA Portal version, example: for TIA Portal 13, file name is "project.ap13"). All variables will be displayed at the same level.	
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.	

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

+ - 🕺 🕲 🔊 🔸] [> 🗞 🖬 🛱	R 🔎 Search 🍸 Filter by: Da	ta 🔻 Ite	ems use	ed:6/10000 Protocol: Show a	ll 🛛 🗹 Show all tags 🔅
Data	Туре	Comment	^	Prop	perty	Value
Modbus TCP:prot1	, Container			× 0	Driver	
Model: Modicon Modbus(1-ba	ised)				Model	Modicon Modbus(1-based)
Holding Registers 1	unsignedShort				Protocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			V I	Dictionary	
 Holding Registers 3 	unsignedShort					false
- MRTU1	unsignedShort				Array	
- MRTU2	unsignedShort				Array size	0
MRTU3	unsignedShort				Arrayindex.Subindex	400003
- MRTU4	unsignedShort				Comment	
- MRTU5	unsignedShort				Data type	unsignedShort

Toolbar item	Description		
ka	Import Tag(s).		
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project		
B ä	Update Tag(s).		
	Click on this icon to update the tags in the project, due a new dictionary import.		
R	Check this box to import all sub-elements of a tag.		
	Example of both checked and unchecked result:		
	Tags* x + - X > > > > > > >		
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.		

Communication status

Current communication status can be displayed using system variables. See "System Variables" section in the main manual.

Codes supported by this	communication driver:
-------------------------	-----------------------

Error	Cause	Action		
NAK The controller replies with a not acknowledge.		-		
Timeout	A request is not replied within the specified timeout period.	Check if the controller is connected and properly configured to get network access.		
Invalid response	The device did received a response with invalid format or contents from the controller .	Ensure the data programmed in the project are consistent with the controller resources.		
General Error	Unidentifiable error. Should never be reported.	Contact technical support.		

Simatic S7 ETH

Simatic S7 ETH communication driver has been designed to communicate with Simatic controllers through Ethernet connection.

The Simatic controller must either have an on-board Ethernet port or be equipped with an appropriate Ethernet interface (either built-in or with a module).

Communication is based on the PG/OP (ISO on TCP) communication functions.

This documents describes the driver settings to be applied in programming IDE software and in S7 PLC programming software.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In Config node double-click Protocols.
- 2. To add a driver, click +: a new line is added.
- 3. Select the protocol from the **PLC** list.

The protocol configuration dialog is displayed.

Simatic S7 ETH		×
PLC Network		ок
Alias		Cancel
IP address	0.0.0.0	
Slot	2	
PLC Models		
S7-3xx S7-313/314 S7-315 S7-317 S7-318 S7-319		▲

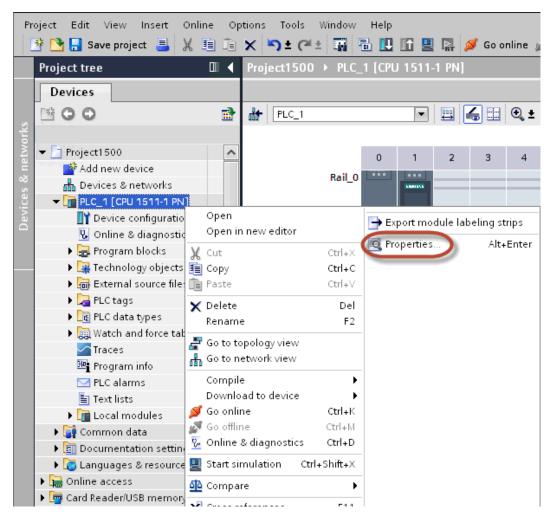
Element	Description
Alias	Name identifying nodes in network configurations. The name will be added as a prefix to each tag name imported for each network node.
IP address	Ethernet IP address of the controller.
Slot	Number of the slot where the CPU is mounted. 2 for S7-300, may take a higher value for S7-

Element	Description
	400 systems.
PLC Models	List of compatible controller models. Make sure to select the correct PLC model in this list when configuring the protocol.
PLC Network	Enable access to multiple networked controllers. For every controller (slave) set the proper option.

S7-1200 and S7-1500 PLC configuration

S7-1200 (starting from firmware version 4.0) and S7-1500 PLC Series from Siemens have a built-in firewall; by default the maximum protection level is enabled. To establish communication with these PLC models it is necessary to enable S7 communication with 3rd party devices; this setting is available in TIA Portal programming software.

- 1. Open the PLC project in TIA Portal.
- 2. Select the PLC from the project tree and open PLC Properties.



3. In General > Protection choose a permission between the top three (make sure that the tick is present on HMI column).

General IO tags	System cons	stants Texts	• • • • • • • • • • • • • • • • • • •					
General	^	Protection						
▼ PROFINET interface [X1]								
General		Protection						
Ethernet addresses								
Time synchronization		Select the access level for the PLC.						
Operating mode								
Advanced options			Access level		Access		Access permi	
Web server access				HMI	Read	Write	Password	
Hardware identifier		Full a	ccess (no protection)	~	×	······		w.
DI 8/DQ 6	=		access	×	1 V		L	
AI 2	-	O HMLa	ccess	×				
 High speed counters (HSC) 			ccess (complete protection)	· ·	-			
Pulse generators (PTO/PWM)	-	0	·····					
Startup								
Cycle								
Communication load			no protection):					
System and clock memory		TIA Portal users and HMI applications will have access to all functions. No password is required.						
Web server								
User interface languages								
Time of day								
Protection								
Configuration control	~							

Note: If "No access" is selected, the communication with the panel will not be established.

4. Scroll down the page and check "Permit access with PUT/GET communication from remote partner".

PL	C_1 [CPU 1511-1	PN]	×
	General 10	tags Texts	
	General		
	PROFINET inte		
	Startup		
	Cycle	Connection mechanisms	
	Communicati		
	System and clo	Permit access with PUT/GET communication from remote partner (PLC, HMI, OPC,)	
	System diagn		
	Web server		
	Display		
	User interface I	Fi i i i i i i i i i i i i i i i i i i	
	Time of day	•	
	Protection		
	System powe		
	Connection res		
	Overview of a		
	< >		
	<	K	
		OK Cancel	

Note: If variables are defined in "Program blocks", DB must configured as "Not optimized".

To check or change DB optimization, open DB Properties:

i

😼 Uniine & alagna	STICS	5 📲	рірро
🔻 🛃 Program blocks		6 🕣	pluto
📑 Add new blo	ck 🛛	7	<add new=""></add>
💶 Main [OB1]			
📕 DB2 [DB2]			
🥃 DB4 [DB4	Open		
🕨 🕨 🙀 Technology 📑	Generate source	from block	s
🕨 🖢 External sou	Snapshot of the I	monitor val	ues
🔻 🚂 PLC tags 👘	•		
li Show all	Apply snapshot v	alues as st	art values 🕨
📑 Add new	Cut		Ctrl+×
💐 Default ta 🧾	Сору		Ctrl+C
🖳 Tag table 💼	Paste		Ctrl+V
🔻 💽 PLC data typ	Copy as text		
📑 Add new 🧹	Delete		Del
🦉 my_data 🔨	Rename		F2
🥵 my_data			12
🕨 🕽 🙀 Watch and f	Compile		•
📴 Program info	Download to dev	lice	•
🛅 Text lists 🛛 🖉	Go online Go offline		Ctrl+K Ctrl+M
🕨 🍺 Local modu			
🕨 🏹 Common data	Cross-reference i		Shift+F11
🔹 🕨 🛅 Documentation 🗡			F11
	Call structure		
🕨 🔚 Online access 🔤 🛄	Assignment list		
 Details view 	Switch programm	ning langua	age 🕨 🕨
	Know-how protec	tion	
Name	Print		Ctrl+P
<u>دا</u>	Print preview		
	Properties		Alts Enter-
🔹 Portal view 🛛 🛤	hopenies		A

In General > Attributes uncheck "Optimized block access":

General	
Information	Attributes
Time stamps	
Compilation	Only store in load memory
Protection	Data block write-protected in the device
Attributes	Optimized block access

If check box "Optimized block access" is not available (grayed-out) it could be because DB is an "instance DB" linked to an "optimized access FB".

After compiling the project, tag offsets will be shown close to variable name.

These settings can be applied to TIA Portal programming software, S7-1200 PLC family starting from PLC firmware version 4.0 and S7-1500 PLC family.

Logo! PLC configuration

To configure communication with Logo! PLC:

- 1. Open the Logo!Soft Comfort project.
- 2. Select **Tools > Ethernet Connections**: the Configure Ethernet Connections dialog is displayed.

Configure Ethernet	Connections			
Module Address				
IP Address	192.168. 2.210			
Subnet Mask:	255.255.255. 0			
Default gateway	192.168. 20. 10			
Ethernnet connecti	onnections			
Connection1(Server)				
	OK Cancel	Help		

- 3. Right-click on Ethernet Connections and add a server connection.
- 4. Double-click on the newly created connection: the connection properties dialog is displayed.

Connection1(Server)
Local Properties (Server)
Connect with an Operator Panel (OP) Accept all connection requests
Only this connection:
Remote Properties (Client)
Keep Alive
Enable the Keep Alive function for this connection
Keep Alive Interval: 👥 🌐 Seconds
OK Cancel Help

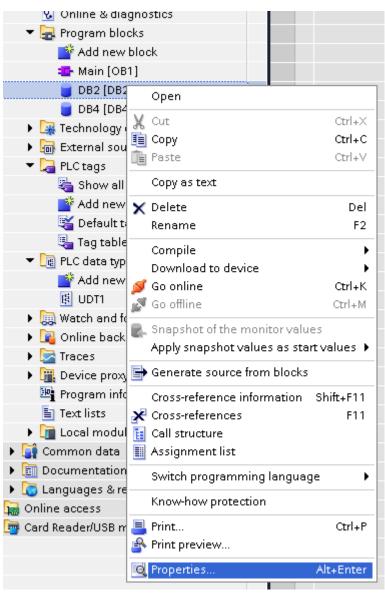
- 5. Select the Connect with an operator panel (OP) (0BA7 model only, do noth check for Logo! 0BA8 model)
- 6. Select Accept all connection requests options.
- 7. In the Remote Properties (Client) section, set TSAP to 02.00.

Direct Import of TIA Portal project

It is possible to import TIA Portal variables directly from TIA Portal project, by selecting "TIA Portal Project v12 or newer" from import selection (refer to "Tag Import" chapter).

Data Blocks must be set as Not optimized:

- 1. Configure the Data Block as **Not optimized**.
- 2. Right-click on the Data Block and choose Properties:



3. In the General tab select Attributes and unselect Optimized block access.

General	Attributes
Information Time stamps	
Compilation	Only store in load memory
Protection	
Attributes	Data block write-protected in the device
	Optimized block access
	•
	b

Note: If the options **Optimized block access** is not enabled (checkbox grayed out) this might mean that the Data Block is an "instance DB" linked to an "optimized access FB".

Export using TIA Portal v13, v14 or newer

Exporting Program blocks

ĩ.

These files refer to DB tags defined in **Program blocks**.

- 1. Configure the Data Block as Not optimized.
- 2. Right-click on the Data Block and choose Properties:

😼 Online & diagno	stics
🔻 🚘 Program blocks	
📑 Add new bloc	ck
💁 Main [OB1]	
🥃 DB2 [DB2	Open
🥃 DB4 [DB4	
🔹 🕨 🙀 Technology 🐇	Cut Ctrl+X
🕨 🔄 External sou 🚞	Copy Ctrl+C
🔻 浸 PLC tags 🛛 💻	Paste Ctrl+V
🍇 Show all	Copy as text
📑 Add new 🗙	Delete Del
💐 Default ti	Rename F2
🎭 Tag table	Compile •
💌 🔻 [🕞 PLC data typ	Download to device
💣 Add new 🧹	Go online Ctrl+K
🚯 UDT1 🖉	Go offline Ctrl+M
🔹 🕨 🥅 Watch and fe	Change at af the manitory values
🔹 🕨 🙀 Online back 🌑	Snapshot of the monitor values Apply snapshot values as start values
🕨 🔀 Traces 🔤	
🔹 🕨 强 Device proxy 📑	Generate source from blocks
📴 Program info	Cross-reference information Shift+F11
🛅 Text lists 🛛 🔀	Cross-references F11
🕨 🧎 Local modul 📔	Call structure
🕨 🙀 Common data 🔳	Assignment list
Documentation	Switch programming language
Languages & re	Know-how protection
Gonline access	· · · · · · · · · · · · · · · · · · ·
	Print Ctrl+P
5	Print preview
Q	Properties Alt+Enter

3. In the General tab select Attributes and unselect Optimized block access.

General General	
Information	Attributes
Time stamps	
Compilation	Only store in load memory
Protection	Data block write-protected in the device
Attributes	Optimized block access



Note: If the options **Optimized block access** is not enabled (checkbox grayed out) this might mean that the Data Block is an "instance DB" linked to an "optimized access FB".

4. Right-click on the Data Block and choose Generate source from blocks:

📥 Devices & networks			2		•	pippo
▼ 1 PLC_1 [CPU 1212C DQ/		DC]	3		•	pluto
🛐 Device configuratio						
😮 Online & diagnostic	cs					
🔻 🛃 Program blocks						
📑 Add new block						
🐴 Main [OB1]						
🥃 DB2 [DB2]		_				
🥃 DB4 [DB4]		Open				
🕨 🕨 🙀 Technology object	Ж	Cut				Ctrl+X
🕨 🕨 🖬 External source file	ŧ.	Сору				Ctrl+C
🔻 🚂 PLC tags	ì	Paste				Ctrl+V
🍇 Show all tags		Copy as text				
💣 Add new tag ta	×	Delete				Del
🍯 Default tag tab		Rename				F2
🔩 Tag table_1 [0] -		0 1				
🔻 🛅 PLC data types		Compile Devente e des de				
📑 Add new data t		Download to de	vice			
UDT1	-	Go online				Ctrl+K
🕨 🕨 🔜 Watch and force ta	127	Go offline				Ctrl+M
🕨 📴 Online backups	R-	Snapshot of the	e mor	nitor	r valu	es
🕨 🔄 Traces		Apply snapshot	valu	es a	is stai	t values 🕨
🕨 🕒 Device proxy data	₽	Generate sourc	e fro	m bl	locks	
📴 Program info						ohih saa
🛅 Text lists	-	Cross-reference		rma	tion	Shift+F11
Local modules		Cross-reference	:5			F11
🕨 📑 Common data		Call structure				
Documentation settin-		Assignment list				
🕨 🐻 Languages & resourc		Switch program	ming	g lar	nguag	le 🕨
🔚 Online access		Know-how prot	ectio	n		
🤄 Card Reader/USB memor		Print				Ctrl+P
	_	Print preview				
	Q	Properties				Alt+Enter

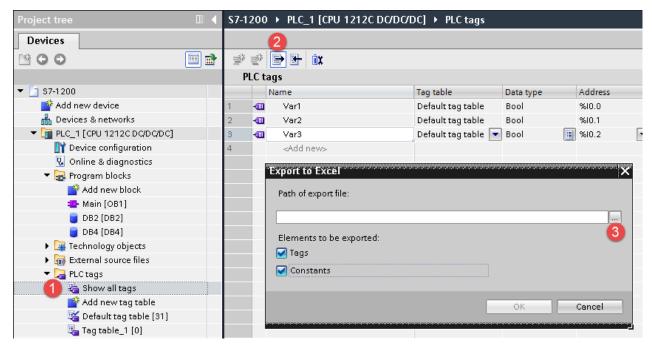
5. Save the file as DBxxx.db, where xxx=number of DB.

S7-1200	Name		Data type	Offset	Start	value	Retain	Accessible f
💣 Add new device 🦷								
🛔 Devices & networks	Save As							— ×
🛅 PLC_1 [CPU 1212C DC/DC/DC] 👘	OO V Use	r 🕨 My Docun	nents 🕨 Automation	•	44	Search Au	tomation	Q
🔢 Device configuration		-				L		
😨 Online & diagnostics	Organize 🔻 Nev	v folder					•	• 0
🔻 🛃 Program blocks	📜 Recent Places	 Name 	*		Date	e modified	Туре	
📑 Add new block	necent haces	Indrie			Date	mouneu	туре	
🔹 Main [OB1]	🥽 Libraries			No items match y	Zour se	earch.		
🥃 DB2 [DB2]								
🥃 DB4 [DB4]	Documents							
🕨 🙀 Technology objects	a) Music							
External source files	📔 Pictures							
🔻 浸 PLC tags	🛃 Videos	=						
🍇 Show all tags								
📑 Add new tag table	🍓 Homegroup							
💐 Default tag table [29]								
🖳 Tag table_1 [0]	🜉 Computer							
🔻 [PLC data types	1 computer							
📑 Add new data type	• ••••	+ +		111				- F
围 UDT1	F 1	DB2.db						
Watch and force tables	File name:	DB2.db						
🕨 📴 Online backups	Save as type:	DB files(*.db)						
🕨 🔄 Traces								
🕨 强 Device proxy data	🛋 Hide Folders					Save	Ci	ancel
📴 Program info	- Inde Folders							
🛅 Text lists								

Exporting PLC tags

An Excel file refers to PLC tags.

- 1. Double-click **Show all tags**: the tag table is displayed.
- 2. Click the **Export** button and browse for path file.
- 3. Define file name.



4. Click Save to confirm.

Save As		×
🔾 🕞 🗸 « User 🕨 My Documents 🕨 Automa	ation 🔻 🍫 Search Autor	nation 🔎
Organize 🔻 New folder		≣ ▼ 🔞
🛧 Favorites 🚔 Name	Date modified	Туре
💻 Desktop 🙀 👔	No items match your search.	
🖳 Recent Places 😑		
🥽 Libraries		
Documents		
J Music		
📄 Pictures 📕 Videos		
🛃 Videos		
🤞 Homegroup 🛛 👻 🗸	III	•
File name: PLCTags.xlsx		-
Save as type: Xlsx files (*.xlsx)		•
🔿 Hide Folders	4 Save	Cancel

5. Click **OK** to export.

Export to Excel ×
Path of export file:
C:\Users\User\Documents\Automation\PLCTags.xlsx
Elements to be exported:
🗹 Tags
Constants
5 OK Cancel
لئر. لئر

Exporting PLC data types

To create the file, expand **PLC data types** item from TIA Portal project tree and right click on the user defined structure. Then click on **Generate source from blocks**.

💐 Default tag table [31]	
🍓 Tag table_1 [0]		
🔻 🛅 PLC data types		
📑 Add new data type		
uDT1	Open	
🖪 UDT2	•	
Watch and force tables	🔏 Cut	Ctrl+X
🕨 🙀 Online backups	🛅 Сору	Ctrl+C
🕨 🔀 Traces	Paste	Ctrl+V
🕨 🛄 Device proxy data	Copy as text	
📴 Program info	🗙 Delete	Del
🔄 Text lists	Rename	F2
🕨 🛅 Local modules	Compile	
🕨 🙀 Common data	of Go online	Ctrl+K
Documentation settings	🖉 Go offline	Ctrl+M
🕨 🐻 Languages & resources	1	
🔚 Online access	Generate source from block:	5
🤄 Card Reader/USB memory	Cross-reference information	Shift+F11
	🔀 Cross-references	F11
	🛅 Call structure	
	🏢 Assignment list	
	昌 Print	Ctrl+P
	Print preview	
	國 Properties	Alt+Enter

In case of multiple PLC data types in PLC project, it is necessary to select them all from **PLC data types** list, right click and select **Generate source from blocks** to create the .UDT file that contains all the PLC data types defined.

tag table_1 [0] € ■ Cata types		
Add new data type		
UDT1	Open	
	🗶 Cut 🛅 Copy	Ctrl+X Ctrl+C
 Watch and force table 	💼 Paste	Ctrl+V
 Image: Online backups Image: Image: Ima	🗙 Delete Rename	Del F2
Device proxy data Program info	Compile 💋 Go online	► Ctrl+K
Text lists	🖉 Go offline	Ctrl+M
 Lim Local modules Common data 	Generate source from blocks	
 Documentation settings Constant Languages & resources Online access Card Reader/USB memory 	Cross-reference information Cross-references Call structure Assignment list	Shift+F11 F11
	💻 Print Print preview	Ctrl+P
	🖸 Properties	Alt+Enter

In the next step, give a name to the .UDT file and choose the path to where to save the file.

Save As			×
	ser 🕨 My Documents 🕨 Automati	on 🔻 🍫 Search Au	tomation 🔎
Organize 🔻 Ne	ew folder		:= • 🕡
📃 Recent Place:	s 🔺 Name	Date modified	Туре
 □ Libraries □ Documents □ Music □ Pictures ■ Videos ■ Homegroup 	E	No items match your search.	
🌉 Computer			
•	▼	III	4
File name:	myUDTfile		•
Save as type:	UDT files(*.udt)		•
🔿 Hide Folders		Save	Cancel

This file will content all the PLC data types and it can be used for importing tags in Tag Editor.

Check **Tag Import** chapter for more details.

Export using TIA Portal v10, v11, v12

Exporting Program blocks

These files refer to DB tags defined in **Program blocks**.

- 1. Configure the Data Block as **Not optimized**.
- 2. Right-click on the Data Block and choose Properties:

😼 Online & alagno	STICS	5 📲	рірро
🕶 🛃 Program blocks 🛛 🛛 🗧 🗠 🚳			pluto
📑 Add new bloc	Add new block 7 <		
💶 Main [OB1]			
📕 DB2 [DB2]	-		
🥃 DB4 [DB4	Open		
🕨 🕨 🙀 Technology 📑	Generate source i	from blocks	5
🕨 🔚 External sou	Snapshot of the n	nonitor valu	Jes
🔻 浸 PLC tags 🛛 🚽			
🍇 Show all	Apply snapshot va	alues as sta	art values 🕨
📑 Add new	Cut		Ctrl+X
💐 Default ta 🧾	Сору		Ctrl+C
🖳 Tag table 🛅	Paste		Ctrl+V
🗢 💽 PLC data typ	Copy as text		
📑 Add new 🧹	Delete		Del
🥫 my_data_ 🔨	Rename		F2
🖪 my_data			
🕨 🕨 🛄 Watch and f	Compile		•
📴 Program info	Download to devi	ce	Per la K
🛅 Text lists 🖉	Go online Go offline		Ctrl+K Ctrl+M
🕨 🕨 🚺 Local modu			
🕨 🙀 Common data	Cross-reference ir	nformation	
🔹 🕨 🛅 Documentation			F11
🕨 🏹 Languages & ri 🛅			
	Assignment list		
 Details view 	Switch programm	ing langua	ge 🕨 🕨
	Know-how protect	tion	
Name	Print		Ctrl+P
·	Print preview		
	Properties		AltNEnter
🖣 Portal view 🛛 🗳			- ht

3. In the General tab select Attributes and unselect Optimized block access.

General	
Information	Attributes
Time stamps	
Compilation	Only store in load memory
Protection	Data block write-protected in the device
Attributes	Optimized block access

i

Note: If the options **Optimized block access** is not enabled (checkbox grayed out) this might mean that the Data Block is an "instance DB" linked to an "optimized access FB".

Project Edit View Insert Online Optic	ons Tools Window Holp
📑 🎦 🔚 Save project ا 🐰 🗎 🗎	< 🔊 ± (🛎 🔹 🎁 🖳 📭 🖉 🖳 🖉 Go online 🧬 Go offline 🏻 🏪 🖪
Project tree 🔲 🖣	Tia_project_Live_IV11/ PLC_1 [CPU 1211C DC/DC/DC] > PLC tags
Devices	
E O O E	2 2 2 5 K 0x
	81.0.1

- Tia_project_Luca [V11] → PLC_1 [CPU 1211C DC/DC/DC] → Program b Devices 🖻 O O 🛫 🛫 🎭 🤛 🗰 🗛 📾 🔛 🖤 DB2 📥 Devices & networks Data type Offset Start value ~ Name 🕶 🛅 PLC_1 [CPU 1211C DC/DC/... 🕣 👻 Static 🚔 Insert row III Device configuration 2 - 1 tag1 🚔 Add row 😼 Online & diagnostics 3 tag2 ¥ Cut Ctrl+X 🕶 🔙 Program blocks 4 📶 = tag3 📑 Add new block 5 Static_1 💼 Сору Ctrl+C 💼 Paste W Ctrl+V Static_1[0] 💶 Main [OB1] 6 📶 . a) 🥃 DB2 [DB2] 7 -Static_1[1] 🗙 Delete Del 📕 DB4 [DB4] 8 Static 2 Rename F2 🕨 🙀 Tech 📕 Untitled - Notepad 🕨 词 External File Edit Format View Help 🚂 PLC tags 👆 Shov Static False False False Bool false false 0.0 1.0 2.0 tag1 tag2 Bool false True 💣 Add True True 4.0 False Char 🝯 Defai tag3 Byt Static_1 Static_1[0] Static_1[1] 16#0 Byte Array [0..1] of Bool 0.0 Bool 0.1 False Tr 🔩 Tag t True 🕶 🛅 PLC date False True
- 4. Build the project to make sure TIA Portal calculates the tags offset.

- 5. Double-click on a DB name.
- 6. Expand the view of program block selected.
- 7. Select all rows.
- 8. Copy and paste into any text editor.
- 9. Save the file as DBxxx.tia, where xxx=number of DB.



Note: Make sure you use the **Save As** function or the file will be named DB2.tia.txt and will not be visible from the importer.

10. Repeat from step 5 for all program blocks.



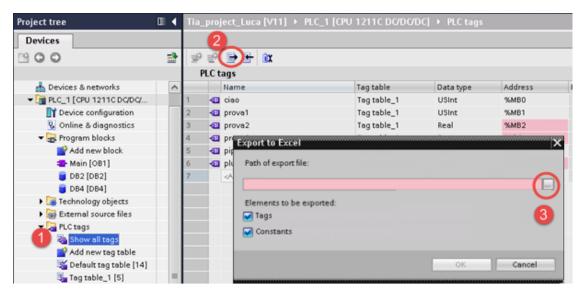
Note: Make sure that only the following columns are shown in DB editor before copying all data in the txt file

		✓ Find an	d replace
Accessible from HMI	Comment		
		Show/Hide 🕨 🕨	📝 Name
	CURRENT PAGE DISPLAYED ON C	Show all columns	📝 Data type
	PANEL PUSHBUTTON PB01	Optimize width	🛛 🗹 Offset
	PANEL PUSHBUTTON PB02	Optimize width of all columns	🔲 Default value
	PANEL PUSHBUTTON PB03		🛃 Start value
	PANEL PUSHBUTTON PB04	Find in s	
	FEEDBACK FOR FIELD COLOR	Find in H	📗 Monitor value
			Retain
		Use wild	🛚 🗹 Accessible from HMI
		Use reg	u 🔜 Visible in HMI
2			📃 Setpoint
		🔘 Whole o	d 🗹 Comment
		From ci	More

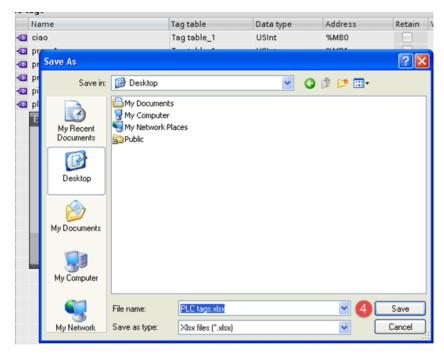
Exporting PLC tags

An Excel file refers to PLC tags.

1. Double-click **Show all tags**: the tag table is displayed.



- 2. Click the **Export** button and browse for path file.
- 3. Define file name.
- 4. Click Save to confirm.

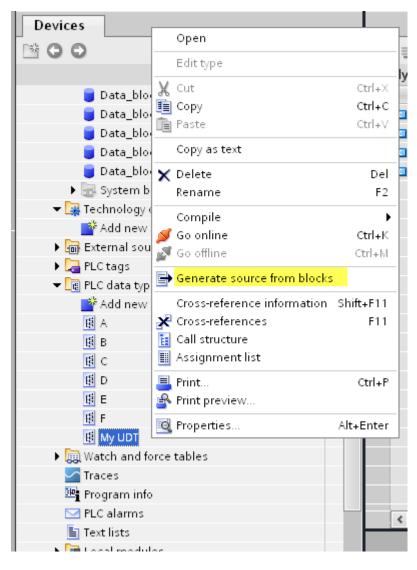


5. Click OK to export.

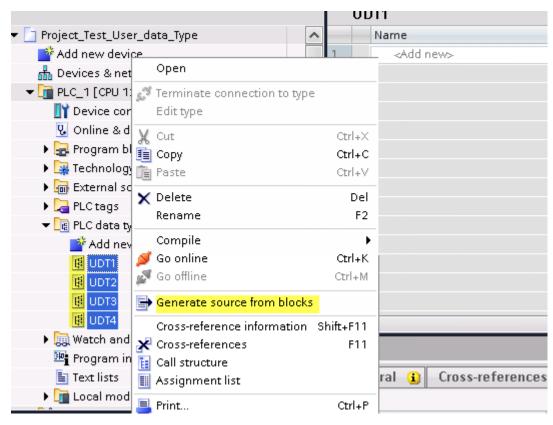
Path of export file:			×
C:\Documents and Settings\ad	min\Desktop\PLC tag:	s.xlsx	
Elements to be exported:			
🛃 Tags			
🛃 Constants			
	•		
	5	ОК	Cancel
	*****	******	

Exporting PLC data types

To create the file, expand **PLC data types** item from TIA Portal project tree and right click on the user defined structure. Then click on **Generate source from blocks**.



In case of multiple PLC data types in PLC project, it is necessary to select them all from **PLC data types** list, right click and select **Generate source from blocks** to create the .SCL file that contains all the PLC data types defined.



In the next step, give a name to the .SCL file and choose the path to where to save the file.

Save As						? 🛛
Save in:	🗁 exported		*	3 🕫 🖻	•	
My Recent Documents						
Desktop						
My Documents						
My Computer						
S	File name:	my_SCL_with_all_UDT		~		Save
My Network	Save as type:	scl files (*.scl)		*		Cancel

This file will content all the PLC data types and it can be used for importing tags in Tag Editor.

Check Tag Import chapter for more details.

Export using STEP7

The Simatic S7 ETH Tag importer accepts symbol files (ASCII format .asc) and source files (.awl extension) created by the Simatic Step7. The symbol file can be previously exported using the Step7 symbol table utility.

Exporting Symbols table

Symbol files (.asc) can be exported from the symbol table utility.

File Edit Insert PLC View Options Window Help	SIMATIC Manager - S7_Protoco	lTest		
S7_ProtocolTest C:\Program Files\Stemens\Step7\s7proj\\$7_Pro-1 S7_ProtocolTest SiMATIC 300 Station CPU315-2DP(1) S7 Program(1) Sources	File Edit Insert PLC View Optio	ons Window Help		
S7_ProtocolTest Sources Blocks Image: SIMATIC 300 Station CPU315-20P(1) Image: S7 Program(1) Sources	🗅 😅 📲 🛲 🕹 🛍 💼		💼 📔 < No Filter >	🔄 🏹 🞇 😂 🖷 🗖 📢
Image: SIMATIC 300 Station Im	S7_ProtocolTest C:\Program	m Files\Siemens\Step7\s7proj\	\$7_Pro~1	
	Gradien SIMATIC 300 Station Gradient CPU315-2DP(1) Gradient Sources	Sources 🗃 Blocks	Symbols	

- 1. From the **Symbol Table** menu in the Symbol Editor choose **Export**.
- 2. Assign a name and save the symbol table as ASCII file.

Symbol	Table E	dit Insert View Options	Window Help			
) 🗳 🛯	6	👗 🖻 💼 🗠 🖂	All Symbols	-	🏹 🕴 💦	
	Status	Symbol 🛆		Address	Data type	Comment
1		A234567890123456789012	234	MVV 65524	WORD	
2		BasicDataType_UDT		UDT 1	UDT 1	
3		bit_M_2_0		M 2.0	BOOL	
4		bit_M_32770_0		M 32770.0	BOOL	
5		bit_M_65522_0		M 05500.0	D00	
6		bit_M_7_5	Export			? 🔀
7		byte_MB_3				
8		byte_MB_32771	Save in: 🗁 ProtTe:	st		💌 🖛 🗈 📸 📰 =
9		byte_MB_65523	ProtTest_Symbols	01.555		
10		char_MB_7				
11		ComplexDataTypes	ProtTest_Symbols_	_uz.asc		
12		Cycle Execution				
13		date_MVV_24641				
14		dint_MD_32773				
15		dint_MD_5				
16		dint_MD_65525				
17		dword_MD_0				
18		dword_MD_32768	File name:	at Carebola 01		
19		dword_MD_65520	File name: ProtTe	st_Symbols_02	2.850	Save
20		int_MW_32774	Save as type: ASCII	Format (* ACC)		
21		int_MV_6	Dave as type. [ASUI	Format (.ASU)		
22		int_MV_65528		1999 00020	INT	
23		real_MD_32777		MD 32777	REAL	

Exporting Sources

These files are created exporting source code.

- 1. Open any program block in the editor, "OB1" in this example.
- 2. From the File menu choose Generate Source: the following dialog is displayed:

LAD/STL/FBD - [OB1 "Cycle Ex	ecution" S7_ProtocolTest\SIMATIC 300 Station\CPU315-2DP(1)\\OB1]
FB blocks FB blocks FB blocks FC blocks SFB blocks SFC blocks Multiple instances €- ∭ Libraries	Contents Of: 'Environment\Interface' Contents Of: 'Environment\Interface' Name TEMP New Entry point: View: Project View: Storage path:
	OB1 : S7_ProtocolTest C:\Program Files\Siemens\Step7\s7pr Browse E IIII OB1 : SMATIC 300 Station ProtTest_All ProtTest_UDT_etc Commer SP CPU3152DP(1) Sources Sources Sources Commer Blocks
	Object name: Sources Object type: STL Source
<u> </u>	DK Cancel Help

1. Assign a name, "Sources" in the example, and click **OK**: the **Generate source Sources** dialog is displayed.

Generate source Sources								
Note: Automatic generation of single sources Menu 'Options' > 'Customize' in the 'So	per block: urces' tab							
Path: S7_ProtocolTest\SIMATIC 300 Station\C	PU315-2DP(1)\							
Blocks Not Selected: Blocks Selected:								
> [A]]> (DB1 UDT Struct Test OB1 Cycle Execution UDT1 BasicDataType_UDT UDT2 ComplexDataTypes							
Name/Family:								
 Include reference blocks Sort according to program structure 	Addresses C Absolute © Symbolic							
ОК	Cancel Help							

- 2. Click **All >** to generate source for all blocks.
- 3. Select the following options:
- Include reference blocks
- Sort according to program structure
- Symbolic address
- 4. Click **OK** to confirm: the "Sources" object is generated in the Step7 project as in the example.

SIMATIC Manager - S7_ProtocolTest	
File Edit Insert PLC View Options Window Help	
🗅 🧀 🚼 🐖 👗 🖻 🕄 📩 😰 🗣 🏪 🏪 📜 🔠 🗰 🔁 🛛 < No Filter >	💽 🎾 🞇 🛞 🖷 🗖 🚺 📢
S7_ProtocolTest C:\Program Files\Siemens\Step7\s7proj\S7_Pro~1	
ST_ProtocolTest SIMATIC 300 Station Image: State of the state	

5. Right click on the object and select **Export Sources**.

📔 🔤 🕹	✓ ½	🖁 🗐 🖷		N
proj\\$7_Pro~1				
t_UDT_etc 🗎 Sources				
	Open Object	Ctrl+Alt+O		
	Cut	Ctrl+X		
	Сору	Ctrl+C		
	Paste	Ctrl+V		
	Delete	Del		
	Insert New Object		•	
_	PLC		۲	
	Compile	Ctrl+B		
	Export Source			
_	Print		۲	
	Rename	F2		
	Object Properties	Alt+Return		
	Special Object Properties		×	

The generated .awl file can be imported in the Tag Editor.

Note: The .awl file contains additional information not included in the .asc file exported from the symbol table.

Make sure that reference to all data blocks is inserted in the symbol table. The tags from a data block are imported only if the symbol table contains a line with the data block name and related comment.

\$7 Pi	ogram(2	?) (Symbols) CPU3	14C-2PNDP	_MPI_187K\SIMATIC	C \$7-300 Station 1\CPU 314C-2 PN/DP	
	Status	Symbol 🛆	Address	Data type	Comment	
1		CPU_FLT	OB 84	OB 84	CPU Fault	
2		I/O_FLT2	OB 83	OB 83	I/O Point Fault 2	
}		OBNL_FLT	08 85	08 85	OB Not Loaded Fault	
Ļ		Prova Data Block	DB 123	DB 123		
5		Prova MB0	MB 0	BYTE		
)		VAT_1	VAT 1			
,						

Each entry enables the import filter to import the tags related to the specified data block.

Tag Editor Settings

i

In the Tag Editor select "Simatic S7 ETH" from the list of defined protocols and click + to add a tag.

Simatic S7 ETH	×
Simatic S7 ETH	
Memory Type	Offset SubIndex
Internal Memory 👻	
Data Block	Data Type Arraysize
Conversion	
	+/-
	OK Cancel Apply Help

Element	Description				
Memory	Area of PLC where tag is located.				
Туре	Data Type	Simatic Type			
	Internal Memory M				
	Data Block	DB			
	Input	I (E)			
	Output	O (A)			
	Timer value	Т			
	Counter value	С			
Offset	Offset address where tag is located.				
SubInde x	Resource offset within the register.				
Data Block	Data block number for Data Block Memory Type.				
Data Type	Available data types: • boolean • byte • short • int • unsignedByte • unsignedShort • unsignedInt • float • string See "Programming concepts" section in the r Note: To define arrays, select one	nain manual. of Data Type format followed by square brackets.			

	Description			
Arraysiz Ə	 In case of st in the string Note: number of by to UTF-8 or Latin1 i If Encoding propert 	tes corresponds to number of string characters if Encoding property is set		
Conversi	requires 2 bytes. Conversion to be a	pplied to the tag.		
on	Conversion	Allowed Configured		
	Depending on data below.	BCD AB->BA ABCD->CDAB ABCD=>CDAB ABCD=>CDAB ABCD=>CDAB Cancel OK Cancel OK		
	Value	Description		
	Value Inv bits	DescriptionInvert all the bits of the tag. $Example:$ $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)		
		Invert all the bits of the tag. <i>Example:</i> $1001 \rightarrow 0110$ (in binary format)		
	Inv bits	Invert all the bits of the tag. <i>Example:</i> $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format) Set the opposite of the tag value. <i>Example:</i>		

DescriptionSwap bytes of a double word.Example: $32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format) $855441236 \rightarrow 1426062386$ (in decimal format)
<i>Example:</i> 32FCFF54 → 54FFFC32 (in hexadecimal format)
$32FCFF54 \rightarrow 54FFFC32$ (in hexadecimal format)
Swap bytes of a long word. Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) 0.10000000110 0001110010111011001000101101000011100001001110000
(in binary format)
Separate the byte in two nibbles, and reads them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)
Used to support S5timer. Check Simatic S5timer special data type for more details.
Legacy transformation for S5timer in binary format.

Use the arrow buttons to order the configured conversions.

Adding an alias name to a protocol

Tag names must be unique at project level, however, the same tag names might need to be used for different controller nodes (for example when the HMI device is connected to two devices running the same application).

When creating a protocol you can add an alias name that will be added to tag names imported for this protocol.

In the example, the connection to a certain controller is assigned the name **Node1**. When tags are imported for this node, all tag names will have the prefix **Node1** making each of them unique at the network/project level.

	a 📖 🔁	Modbus TCP:prot1	€				
Name 🛆			Driver	Addres	s	Comment	
Node 1/ Eat_hodrotia		(#1212) Witted	Bue TCP petril 1	11 Dunkig	with the		
Node1/Calia_hodnotia		(4212) West	Bue TCP pet 1 1	12 Dunkig	withwi		
kode 1/PN_W/X71EPR_larve	1	HEER WINN	Ibue TCP petril 1	0.0-uneign	nd Prof.		
WHEN CHOSE BHOW		(4212) West	Bue TCP petril 1	245 Durney	HerfCherr		
de1/01/1 BX7_holes		(8232) West	Bue TCP pet1 1	1 Dunwign	n#Short		
Indent/PE_C/RTX_hadron		(8232) West	Bue TCP pet1 1	2 Dunsign	North In		
(He T) 700/H (TE2H)		(H212) West	Bue TCP pret1 1	3 Dunxign	Hilling		
lode1/Water_level		ABD CAD	whe Efficient	10 0 unsigr	edShort		
		Select Network node id	Model	Alias			
	-					_	
		(元)天(元)	Noticer (notifica-		Node1		
		医关节	Notices - Intelline		Node2		
	=						
tagname	me						
	me				Ok		
tagname Water_level	me				Ok	Car	ncel
taoname Water_level	me				Ok	Car	ncel
tagname Vater_level	me					Car	ncel
tagname Water_level	me	245	ē		predShort	Car	
Water_level	me L	245	0	une		Car	



Note: Aliasing tag names is only available for imported tags. Tags added manually in the Tag Editor cannot have the Alias prefix in the tag name.

The Alias string is attached at the time of tag import. If you modify the Alias string after the tag import has been completed, there will be no effect on names already present in the dictionary. When the Alias string is changed and tags are re-imported, all tags will be re-imported with the new prefix string.

String data type

In Simatic S7 PLC two different types of tags manage string variables:

- as Array [1..xx] of characters,
- as String[xx].

Step7 string declaration is shown in this example:

Address	Name	Туре	Initial value	Comment	S7 String
0.0		STRUCT			or ounig
+0.0	Stringl	STRING[254]	'sample'		
+256.0	String2	ARRAY[110]			
*1.0		CHAR		Name of Concession, Name of Street, or other	String as array of char
=266.0		END_STRUCT			

TIA Portal string declaration is shown in this example:

	Image: Strate in the second secon								
	-	Na	ime	Data type	Uffset	Start value	Retain	Accessible	Visible in
1		-	Static						
2	-	•	String1	String		'sample'			
3		•	String2	Array [1 10] of Char					
						String as a	rray of cha	ar	

Note: When using String[xx] data type specific a conversion must be applied to the tag. If the tag dictionary is imported from TIA Portal or Step7 using the import tool, however, conversion of the string tags is performed automatically and no further action is required.

To add a string as an array of characters:

1. Press the + in the Tag Editor.

i

Simatic S7 ETH		x
Simatic S7 ETH	<u> </u>	
Memory Type Data Block	Offset SubIndex ▼ 114 0 ▼	
Data Block	Data Type Arraysize String 10	
Conversion	*/-	
	OK Cancel Apply Help	

- 2. Select string as Data Type.
- 3. Enter string length in Arraysize.
- 4. Click OK to confirm.

To add a string data type:

1. Press the + in the Tag Editor.

Simatic S7 ETH		
Simatic S7 ETH		_
Memory Type Data Block	Offset SubIndex ▼ 114 ▼ 0 ▼	
Data Block	Data Type Arraysize string 10	
Conversion	+/-	
	OK Cancel Apply Help	

- 2. Select string as Data Type.
- 3. Enter string length in Arraysize.
- 4. Click +/- to open the Conversion dialog.

Simatic S7 ETH			— ×-
Simatic S7 ETH			
Memory Type	Offset	SubIndex	
Data Block	▼ 114	0 -	
Data Block	Data Type	Arraysize	
1	string	▼ 10	
Conversion		_	
	+/-		
	ОК	Cancel	Apply Help

5. In the conversion dialog select the **S7 String** conversion type.

imatic S7 ETH					×
Simatic S7 ETH					
Memory Type Data Block	Offset ▼ 114 ▼	SubIndex			
Data Block	Data Type		Arraysize		
Conversion	B				
A —	Allowed AB->BA ABCD->CDAB S7 String	-	Configured S7 String		
			Cancel	ОК	Help

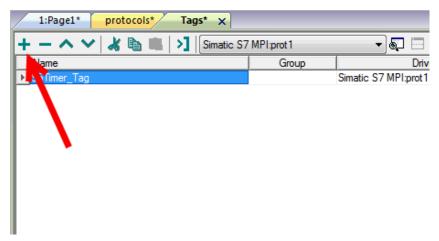
- 6. Click + to add the conversion: the conversion will be listed into the **Configured** list on the right.
- 7. Click OK to confirm.

Simatic S5Timer data type

Simatic drivers support a special data type, the S5Timer data type.

The tag must be configured with a specific data type and a conversion must be applied to the tag to correctly read/write a Simatic S5Timer Variable.

1. In the Tag Editor click + to add a tag.



2. Select unsignedInt as Data Type.

Simatic S7 MPI Memory Type Offset SubIndex Internal Memory V 50 V O V Data Block Data Type Arraysize 1 V O V Conversion V foolean byte short int unsignedInt float string	imatic S7 MPI	X
Internal Memory ▼ 50 ♥ 0 ♥ Data Block Data Type Arraysize 1 ♥ 0 Conversion ♥ 0 ↓ 0 boolean byte short unsignedInt float string	Simatic S7 MPI	
Data Block Data Type Arraysize 1 Image: State of the state of th	Memory Type	Offset SubIndex
1 Conversion Image: Conv	Internal Memory 🗸	50 0 -
Conversion +/- unsignedByte unsignedShort unsignedInt float string	Data Block	Data Type Arraysize
Conversion +/- byte short int unsignedByte unsignedInt float string	1	· · · · · · · · · · · · · · · · · · ·
+/- unsignedByte unsignedShort unsignedInt float string	Conversion	byte short
float string		+/- unsignedByte
		float
Cancer Appry Help		OK Cancel Apply Help

3. Click +/- to open the Conversion dialog.

matic S7 MPI		X
Simatic S7 MPI		
Memory Type	Offset SubIndex	
Data Block	Data Type ↓ unsignedInt ▼	Arraysize
Conversion		
	ОК Са	ancel Apply Help

- 4. In the conversion dialog select the **S5timer(BCD)** conversion type.
- 5. Click + to add the conversion: the conversion will be listed into the **Configured** list on the right.

imatic S7 MPI	N.	23
Simatic S7 MPI		
Memory Type	Offset SubIndex	
Internal Memory 🔹	50 🔺 0 🔻	
Data Block	Data Type Arraysize	
1	unsignedInt	
<u> </u>		
	B	
	Allowed Configur	
	S5timer(BCD) S5timer(BIN)	(BCD)
	—	
	^	
	<	Help
	Cancel	ОК

6. Click OK to confirm.

Node Override IP

The protocol provides the special data type Node Override IP which allows you to change the IP address of the target controller at runtime.

This memory type is an array of 4 unsigned bytes, one per each byte of the IP address.

The Node Override IP is initialized with the value of the controller IP specified in the project at programming time.

Node Override IP	PLC operation
0.0.0.0	Communication with the controller is stopped, no request frames are generated anymore.
Different from 0.0.0.0	It is interpreted as node IP override and the target IP address is replaced runtime with the new value.

If the HMI device is connected to a network with more than one controller node, each node has its own Node Override IP variable.



Note: Node Override IP values assigned at runtime are retained through power cycles.

Hostname DNS or mDNS

In addition to the array of bytes, string memory type can be selected to be able use the DNS or mDNS hostname as an alternative to the IP Address.

Simatic S7 ETH		x
Simatic S7 ETH		
Memory Type	Offset SubIndex	
Node Override IP 👻	0 •	
Data Block	Data Type Arraysize	
1	unsignedByte [] 🔻 4	
Conversion		
	+/-	
	OK Cancel Apply H	lelp

Tag Import

Select the driver in Tag Editor and click on the Import Tags button to start the importer.

	Tags	×								
+	-	z	Ē	ß	>]	₽	A 9B	B>	63	1
Data			^		-	Та	g URI			

The following dialog shows which importer type can be selected.

Н	MIStudio	
	Multiple tag importers are	available for this protocol. Please select the importer type and continue.
	Version	Туре
	TIA Portal Project v 12 or newer	Linear
	TIA Portal v13, v14 or newer	Linear
	TIA Portal v10, v11, v12	Linear
	Step7	Linear
	Tag Editor exported xml	General
	L	OK Cancel

Importer	Description
TIA Portal Project v12 or newer Linear	Allows to import the whole TIA Portal project file using .apxx file (where "xx" is the TIA Portal version, example: for TIA Portal 13, file name is "project.ap13").
	All variables will be displayed at the same level.
TIA Portal v13, v14 or newer Linear	 Allows to import: Program blocks using .db file PLC tags using .xlsx file PLC data types using .udt file Check Export using TIA Portal v13, v14 or newer for more details.
	All variables will be displayed at the same level.
TIA Portal v10, v11, v12 Linear Step7 Linear	 Allows to import: Program blocks using .tia file PLC tags using .xlsx file PLC data types using .scl file Check Export using TIA Portal v10, v11, v12 for more details. All variables will be displayed at the same level. Allows to import: Symbols table .asc file Sources using .awl file
	Check Export using STEP7 for more details.
	All variables will be displayed at the same level.
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags 🗙 Protocols					
+ - 🕺 🕲 👂] [> 🗞 📦 🛱	R 🔎 - Search 🍸 Filter by: Da	ita 🔻 Iter	ns used:6/10000 Protocol: Show a	all 🛛 🗹 Show all tags 🔅
Data	Type	Comment	^	Property	Value
Modbus TCP:prot1	, Container			Y Driver	
Model: Modicon Modbus(1-ba	ised)			Model	Modicon Modbus(1-based)
- Holding Registers 1	unsignedShort			Protocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			 Dictionary 	
 Holding Registers 3 MRTU1 	unsignedShort unsignedShort			Array	false
- MRTU2	unsignedShort			Array size	0
MRTU3	unsignedShort			Arravindex.Subindex	400003
- MRTU4	unsignedShort			Comment	
- MRTU5	unsignedShort			Data type	unsignedShort

Toolbar item	Description
B	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
E⊈ B	Update Tag(s). Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag. Example of both checked and unchecked result:
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Communication status

Current communication status can be displayed using system variables. See "System Variables" section in the main manual.

Codes supported by this communication driver:

Error	Cause	Action
NAK The controller replies with a not acknowledge.		-
Timeout	A request is not replied within the specified timeout period.	Check if the controller is connected and properly configured to get network access.
Invalid	The device did received a response with invalid	Ensure the data programmed in the project are

Error	Cause	Action
response	format or contents from the controller .	consistent with the controller resources.
General Error	Unidentifiable error. Should never be reported.	Contact technical support.

Simatic S7 MPI

HMI products support direct Siemens MPI communication without any additional module.

The driver supports the standard communication speed 187Kbit/s.

here is a minimum requirement also for the version of operating system running in the HMI (this is normally referenced as BSP version). See in user manual how to read the BSP version with the System Settings menu. The minimum requirements are shown in the following table.

Platform	BSP Version
UN30/31	v1.38 or newer
UN65/UN71	v1.0.300 or newer
UN60/UN70	v1.0.413 or newer
UN73	v1.0.142 or newer

Protocol Editor Settings

Add [+] a driver in the Protocol editor and select the "Simatic S7 MPI" protocol from the list of available protocols.

The protocol type can be selected from the dedicated combo box in the dialog.

Simatic S7 MPI	×
PLC Network	Comm OK
Alias	Cancel
Timeout (ms)	1000
Panel MPI address	1
Highest MPI address	15
PLC MPI address	2
PLC Models	
S7-3xx S7-313/314 S7-315 S7-317 S7-318 S7-319	

Element	Description
Alias	Name to be used to identify nodes in the plc network configuration. The name will be added as a prefix to each tag name imported for each network node.
Timeout (ms)	Defines the time inserted by the protocol between two retries of the
	same message in case of missing response from controller.
	Value is expressed in milliseconds.
Panel MPI Address	MPI node number assigned to the device.
Highest MPI Address	The highest node number in the MPI network where the device is operating and communicating.
PLC MPI Address	The MPI address of the controller to which the device needs to communicate.
PLC Models	List of compatible controller models. Make sure to select the correct PLC model in this list when configuring the protocol.
Comm	Click on this button to configure the serial port on the device to be used as MPI port (see example in the following figure) Comm Parameter Dialog Port communication parameters for Simatic S7 MPI are fixed at: Baud rate=187500 Parity=Even Data=bits8 Stop=bit1 On UN20: com1 is the HMI port labeled "PLC", com2 is the HMI port labeled "PLC",

Element	Description
	On UN31 or UN30:
	 com1 is the integrated serial port,
	 com2 is an add-on module plugged in Slot#1 or #2
	 com3 is an add-on module plugged in Slot#3 or #4
	The connection between device and PLC can be made:
	 Using a standard MPI cable with ADP-0001 "MPI wiring adapter"
PLC Network	The protocol supports connection to multiple controllers. To enable this option, check the "PLC Network" check box and enter the configuration per each controller node.

Direct Import of TIA Portal project

It is possible to import TIA Portal variables directly from TIA Portal project, by selecting "TIA Portal Project v12 or newer" from import selection (refer to "Tag Import" chapter).

Data Blocks must be set as Not optimized:

- 1. Configure the Data Block as **Not optimized**.
- 2. Right-click on the Data Block and choose Properties:

😼 Online & diagno	stics
🔻 🚘 Program blocks	
📑 Add new bloc	ck
💁 Main [OB1]	
🥃 DB2 [DB2	Open
🥃 DB4 [DB4	
🔹 🕨 🙀 Technology 🐇	Cut Ctrl+X
🕨 🔄 External sou 🚞	Copy Ctrl+C
🔻 浸 PLC tags 🛛 💻	Paste Ctrl+V
🍇 Show all	Copy as text
📑 Add new 🗙	Delete Del
💐 Default ti	Rename F2
🎭 Tag table	Compile •
💌 🔻 [🕞 PLC data typ	Download to device
💣 Add new 🧹	Go online Ctrl+K
🚯 UDT1 🖉	Go offline Ctrl+M
🔹 🕨 🥅 Watch and fe	Change at af the manitory values
🔹 🕨 🙀 Online back 🌑	Snapshot of the monitor values Apply snapshot values as start values
🕨 🔀 Traces 🔤	
🔹 🕨 强 Device proxy 📑	Generate source from blocks
📴 Program info	Cross-reference information Shift+F11
🛅 Text lists 🛛 🔀	Cross-references F11
🕨 🚺 Local modul 📔	Call structure
🕨 🙀 Common data 🔳	Assignment list
Documentation	Switch programming language
Languages & re	Know-how protection
Gonline access	· · · · · · · · · · · · · · · · · · ·
	Print Ctrl+P
5	Print preview
Q	Properties Alt+Enter

3. In the General tab select Attributes and unselect Optimized block access.

General General Information	Attributes
Time stamps	
Compilation	Only store in load memory
Protection	Data block write-protected in the device
Attributes	Optimized block access

Note: If the options **Optimized block access** is not enabled (checkbox grayed out) this might mean that the Data Block is an "instance DB" linked to an "optimized access FB".

Export using TIA Portal v13, v14 or newer

Exporting Program blocks

i

These files refer to DB tags defined in **Program blocks**.

- 1. Configure the Data Block as Not optimized.
- 2. Right-click on the Data Block and choose Properties:

😼 Online & diagno	stics	
💌 🖛 Program blocks		
📑 Add new blo	ck	
💁 Main [OB1]		
j DB2 (DB2	Open	
🧧 DB4 [DB4	•	
🔹 🕨 🙀 Technology 🐇	Cut	Ctrl+X
🕨 🔄 External sou 🚞	Сору	Ctrl+C Ctrl+V
🔻 浸 PLC tags 🛛 💻	Paste	CEN+V
🝇 Show all	Copy as text	
📑 Add new 🗙	Delete	Del
💐 Default ti	Rename	F2
🍕 Tag table	Compile	•
🔻 💽 PLC data typ	Download to device	, i i i i i i i i i i i i i i i i i i i
📑 Add new 💋	Go online	Ctrl+K
	Go offline	Ctrl+M
🕨 🕨 🔜 Watch and f	Snapshot of the monitor valu	es
Online back	Apply snapshot values as sta	
🕨 🔀 Traces		
· • • • • • • • • • • • • • • • • • • •	Generate source from blocks	
📴 Program info	Cross-reference information	Shift+F11
	Cross-references	F11
	Call structure	
	Assignment list	
Documentation	Switch programming langua	ge 🕨
Languages & re Online access	Know-how protection	
	Print	Ctrl+P
	Print preview	CLITT
9	Properties	Alt+Enter

3. In the General tab select Attributes and unselect Optimized block access.

General	
Information	Attributes
Time stamps	
Compilation	Only store in load memory
Protection	Data block write-protected in the device
Attributes	Optimized block access

Ð

Note: If the options **Optimized block access** is not enabled (checkbox grayed out) this might mean that the Data Block is an "instance DB" linked to an "optimized access FB".

4. Right-click on the Data Block and choose Generate source from blocks:

2 🕣 = pippo
VDC] 3 📶 = pluto
Open
Cut Ctrl+X
Copy Ctrl+C
Paste Ctrl+V
Copy as text
Delete Del Rename F2
Rename F2
Compile 🔰
Download to device
Goonline Ctrl+K
Go offline Ctrl+M
Snapshot of the monitor values
Apply snapshot values as start values 🕨
Generate source from blocks
Cuase veloces and information Chilt. E11
Cross-reference information Shift+F11
Cross-references F11
Cross-references F11 Call structure
Cross-references F11
Cross-references F11 Call structure
Cross-references F11 Call structure Assignment list
Cross-references F11 Call structure Assignment list Switch programming language Know-how protection
Cross-references F11 Call structure Assignment list Switch programming language Know-how protection Print Ctrl+P
Cross-references F11 Call structure Assignment list Switch programming language Know-how protection

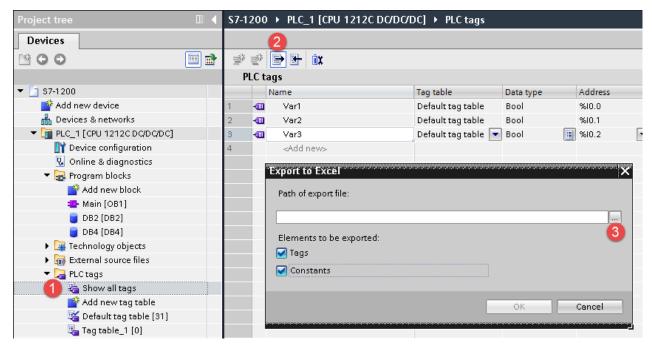
5. Save the file as DBxxx.db, where xxx=number of DB.

\$7-1200	Name			Data type	Offset	Start	t value	Retain	Accessible f.
🗳 Add new device 🦷									
h Devices & networks	Save As								×
🛅 PLC_1 [CPU 1212C DC/DC/DC]	Use	r ► M	ly Documents	 Automation 	-	44	Search Au	tomation	Q
🛐 Device configuration									
😧 Online & diagnostics	Organize 🔻 Nev	v folder	r					:== ·	• 🔞
 Program blocks 	🖳 Recent Places		Name	<u>^</u>		Date	e modified	Туре	
💣 Add new block			Tadiric .			Date	- mounica	13PC	
🖀 Main [OB1]	🔚 Libraries				No items match	/our se	earch.		
📕 DB2 [DB2]	Documents								
🗧 DB4 [DB4]									
Technology objects	🚽 Music								
External source files	E Pictures								
▼ 📮 PLC tags	🛃 Videos	=							
🖏 Show all tags									
🚔 Add new tag table	🤣 Homegroup								
🝯 Default tag table [29]									
💺 Tag table_1 [0]	🜉 Computer								
▼ 🕞 PLC data types	17. company								
📫 Add new data type	A	T	•		III				F.
UDT1	File name:	DB2.db	, ,						
Watch and force tables			-						
Online backups	Save as type:	DB files	s(*.db)						•
Traces									
Device proxy data	🗻 Hide Folders						Save	Ca	incel
📴 Program info	<u> </u>								
Text lists									

Exporting PLC tags

An Excel file refers to PLC tags.

- 1. Double-click **Show all tags**: the tag table is displayed.
- 2. Click the **Export** button and browse for path file.
- 3. Define file name.



4. Click Save to confirm.

Save As			×	
🔾 🕞 🗸 « User 🕨 My Documents 🕨 Automation	· •	← Search Auton	nation 🔎	>
Organize 🔻 New folder			:= • 🔞	
ravorites ▲ Name		Date modified	Туре	
📰 Desktop 🚺 Downloads 🖼 Recent Places 🔄	No items match yo	our search.		
 Libraries Documents Music Pictures Videos 				
🌏 Homegroup 👻 🤟				Þ
File name: PLCTags.xlsx			-	•
Save as type: XIsx files (*.xIsx)			•	·
) Hide Folders		4 Save	Cancel	

5. Click **OK** to export.

Export to Excel
Path of export file:
C:\Users\User\Documents\Automation\PLCTags.xlsx
Elements to be exported:
🗹 Tags
Constants
OK Cancel
ل

Exporting PLC data types

To create the file, expand **PLC data types** item from TIA Portal project tree and right click on the user defined structure. Then click on **Generate source from blocks**.

💐 Default tag table [31]	
🍓 Tag table_1 [0]		
🔻 🛅 PLC data types		
📑 Add new data type		
uDT1	Open	
🖪 UDT2		
🕨 🕨 📷 Watch and force tables	X Cut	Ctrl+X
🕨 📴 Online backups	🛅 Сору	Ctrl+C
🕨 🔀 Traces	Paste	Ctrl+V
🕨 🗽 Device proxy data	Copy as text	
📴 Program info	🗙 Delete	Del
🔄 Text lists	Rename	F2
🕨 🛅 Local modules	Compile	
🕨 🙀 Common data	🧭 Go online	Ctrl+K
Documentation settings	So offline	Ctrl+M
🕨 🐻 Languages & resources		
🔚 Online access	Generate source from block	<5
🤠 Card Reader/USB memory	Cross-reference informatior	n Shift+F11
	🔀 Cross-references	F11
	🛅 Call structure	
	🏢 Assignment list	
	昌 Print	Ctrl+P
	Print preview	
	國 Properties	Alt+Enter

In case of multiple PLC data types in PLC project, it is necessary to select them all from **PLC data types** list, right click and select **Generate source from blocks** to create the .UDT file that contains all the PLC data types defined.

🍇 Tag table_1 [0]			
🔻 [🔃 PLC data types			
📑 Add new data type	e		
团 UDT1		Open	
团 UDT2		open	
臣 UDT3	9.0	Cut	Ctrl+X
度 UDT4	_	Сору	Ctrl+C
🕨 🕨 🔜 Watch and force table		Paste	Ctrl+V
🕨 🙀 Online backups	×	Delete	Del
🕨 🔄 Traces		Rename	F2
🕨 🗽 Device proxy data		Compile	•
📴 Program info	ø	Go online	Ctrl+K
🛅 Text lists	1	Go offline	Ctrl+M
🕨 🛅 Local modules	-	Generate source from blocks	
🕨 🙀 Common data			
Documentation settings		Cross-reference information	
🕨 🚺 Languages & resources		Cross-references	F11
🔜 Online access		Call structure	
👼 Card Reader/USB memory 🚽		Assignment list	
		Print	Ctrl+P
	<u>a</u>	Print preview	
	Q	Properties	Alt+Enter
L			

In the next step, give a name to the .UDT file and choose the path to where to save the file.

Save As					×
Us 🔍 🗸 🔾	er 🕨 My Documents 🕨 Automation	- - - f - j	Search Autom	ation	٩
Organize 🔻 Ne	w folder			== -	0
📃 Recent Places	^ Name	Dat	e modified	Туре	
 ➢ Libraries ➢ Documents ➢ Music ➢ Pictures ☑ Videos ☑ Homegroup 	IL	No items match your s	earch.		
📑 Computer					
~	▼	III			÷.
File name:	myUDTfile				-
Save as type:	UDT files(*.udt)				•
) Hide Folders			Save	Cancel	

This file will content all the PLC data types and it can be used for importing tags in Tag Editor.

Check Tag Import chapter for more details.

Export using TIA Portal v10, v11, v12

Exporting Program blocks

These files refer to DB tags defined in **Program blocks**.

- 1. Configure the Data Block as **Not optimized**.
- 2. Right-click on the Data Block and choose Properties:

😼 Online & alagn	OSTICS	5	-	рірро
 Program blocks 	6		pluto	
Add new blo	7		<add new=""></add>	
📥 Main [OB1]				
DB2 [DB2]				
🗧 DB4 [DB4	Open			
🕨 🙀 Technology 📑	Generate source	from bl	ocks	5
🕨 🔚 External sou	Snapshot of the r	nonitor	vali	Jes
🔻 浸 PLC tags	•			
🍇 Show all	Apply snapshot v	alues a	s sta	art values 🕨
	Cut			Ctrl+X
🍯 Default ti 🏥	Сору			Ctrl+C
🖳 Tag table 🗊	Paste			Ctrl+V
🔻 🛅 PLC data typ	Copy as text			
📑 Add new	Delete			Del
🔢 my_data	Rename			F2
🦉 my_data				
🕨 🥅 Watch and fi	Compile Download to devi			
📴 Program info	Go online	CC .		Ctrl+K
🛅 Text lists 🎽	Co offline			Ctrl+M
🕨 🛄 Local modu 🖛				
🕨 🙀 Common data	Cross-reference in	ntormat	tion	Shift+F11
🕨 🛅 Documentation 🗶				F11
	Call structure Assignment list			
	_			
 Details view 	Switch programm	ning lar	igua	ige 🕨 🕨
	Know-how protec	tion		
Name	Print			Ctrl+P
	Print preview			
	Properties			AltNEnter
🔹 Portal view 🛛 🗎	, Hoperico			

3. In the General tab select Attributes and unselect Optimized block access.

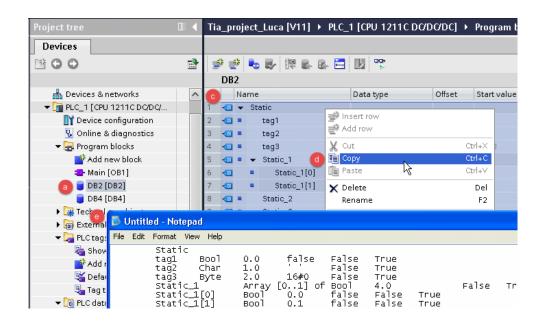
General General Information Time stamps Compilation	Attributes
Protection Attributes	Data block write-protected in the device Optimized block access

Ð

Note: If the options **Optimized block access** is not enabled (checkbox grayed out) this might mean that the Data Block is an "instance DB" linked to an "optimized access FB".

Project Edit View Insert	Online Optio	ns Tools Window Halo
📑 🎦 🛃 Save project 昌	X 🗈 🗈 🗙	(🔊 ± (주 ± 🧃 🗟 🏨 🏨 🖀 🖳 🦝 🎸 Go online 🧬 Go offline 🏻 🏪 🖪 🛛
Project tree	□ <	Tia_project_Live_JV11/ PLC_1 [CPU 1211C DC/DC/DC] > PLC tags
Devices		
00	B	2 2 5 6 0x

4. Build the project to make sure TIA Portal calculates the tags offset.



- 5. Double-click on a DB name.
- 6. Expand the view of program block selected.
- 7. Select all rows.
- 8. Copy and paste into any text editor.
- 9. Save the file as DBxxx.tia, where xxx=number of DB.



Note: Make sure you use the **Save As** function or the file will be named DB2.tia.txt and will not be visible from the importer.

10. Repeat from step 5 for all program blocks.



Note: Make sure that only the following columns are shown in DB editor before copying all data in the txt file

		✓ Find an	d replace
Accessible from HMI	Comment		
		Show/Hide 🕨 🕨	📝 Name
	CURRENT PAGE DISPLAYED ON C	Show all columns	📝 Data type
	PANEL PUSHBUTTON PB01	Optimize width	🗹 Offset
	PANEL PUSHBUTTON PB02	Optimize width of all columns	🔲 Default value
	PANEL PUSHBUTTON PB03		🛛 🗹 Start value
	PANEL PUSHBUTTON PB04	Find in s	su 🥅 Snapshot
	FEEDBACK FOR FIELD COLOR	Find in H	🚬 🔜 Monitor value
	TEEDDI IOITTOINTIEED COECIN		🛛 🛃 Retain
		Use wild	💷 🗹 Accessible from HMI
		🖂 Use reg	u 🔜 Visible in HMI
			🔲 Setpoint
		Whole @	d 🗹 Comment
		From cl	More
			MOIC

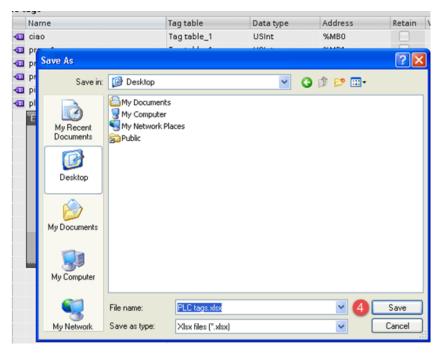
Exporting PLC tags

An Excel file refers to PLC tags.

1. Double-click **Show all tags**: the tag table is displayed.

Project tree	□ 4	Tia	_proj	ject_Luca [V11] → PLC	_1 [CPU 1211C DC/DC/I	DC] → PLC tags	
Devices			2				
B 0 0	2	Ð	# ₹()) ± (x				
		1	PLC t	ags			
📥 Devices & networks	^	_		Name	Tag table	Data type	Address
		1	-0	ciao	Tag table_1	USInt	%MB0
Device configuration		2	-83	prova1	Tag table_1	USInt	%MB1
😼 Online & diagnostics		3	-83	prova2	Tag table_1	Real	%MB2
🕶 🕁 Program blocks		4	-0	Pr Export to Excel			X
Add new block		5	-0	pig			^
🖀 Main [OB1]		6	-0	plu Path of export file:			
📒 DB2 [DB2]		7		<a< td=""><td></td><td></td><td></td></a<>			
🥃 DB4 [DB4]							
Technology objects				Elements to be export	rted:		\sim
External source files				Tags			3
PLC tags				Constants			-
1 🐁 Show all tags							
Add new tag table							
📽 Default tag table [14]						OK	Cancel
light Tag table_1 [5]	=						

- 2. Click the Export button and browse for path file.
- 3. Define file name.
- 4. Click Save to confirm.

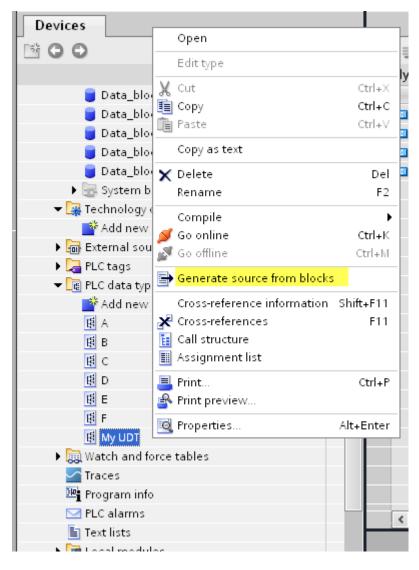


5. Click OK to export.

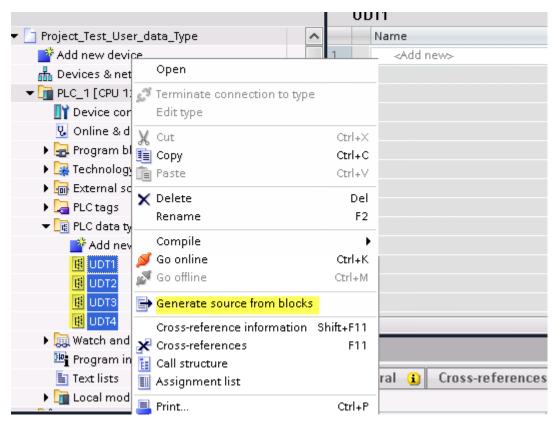
cport to Excel	
Path of export file:	
C:\Documents and Settings\admin\Desktop\PLC tags.xlsx	
Elements to be exported:	
✓ Tags	
Constants	
•	
5 OK Cance	

Exporting PLC data types

To create the file, expand **PLC data types** item from TIA Portal project tree and right click on the user defined structure. Then click on **Generate source from blocks**.



In case of multiple PLC data types in PLC project, it is necessary to select them all from **PLC data types** list, right click and select **Generate source from blocks** to create the .SCL file that contains all the PLC data types defined.



In the next step, give a name to the .SCL file and choose the path to where to save the file.

Save As					? 🔀
Save in:	🗁 exported		~	3 🦻 📂	 .
My Recent Documents					
Desktop					
My Documents					
My Computer					
S	File name:	my_SCL_with_all_UDT		~	Save
My Network	Save as type:	scl files (*.scl)		*	Cancel

This file will content all the PLC data types and it can be used for importing tags in Tag Editor.

Check Tag Import chapter for more details.

Export using STEP7

The Simatic S7 ETH Tag importer accepts symbol files (ASCII format .asc) and source files (.awl extension) created by the Simatic Step7. The symbol file can be previously exported using the Step7 symbol table utility.

Exporting Symbols table

Symbol files (.asc) can be exported from the symbol table utility.

SIMATIC Manager - S7_ProtocolTest	
File Edit Insert PLC View Options Window Help	
🗋 🗅 😅 🏭 📾 👗 🕿 🗣 🔛 🏥 🗰 💼 🔁 🔨 🔀) 🖷 🗖 🛄 🐶
S7_ProtocolTest C:\Program Files\Siemens\Step7\s7proj\S7_Pro-1	X
S7_ProtocolTest SIMATIC 300 Station CPU315-ZDP(1) Sources Blocks Blocks	

- 1. From the Symbol Table menu in the Symbol Editor choose Export.
- 2. Assign a name and save the symbol table as ASCII file.

Symbol Table Edit Insert View Options Window Help						
🖻	- 4	👗 🖻 💼 🗠 🗠	All Symbols	•	🏆 🛛 💦	
	Status	Symbol 🛆		Address	Data type	Comment
1		A234567890123456789012	234	MVV 65524	WORD	
2		BasicDataType_UDT		UDT 1	UDT 1	
3		bit_M_2_0		M 2.0	BOOL	
4		bit_M_32770_0		M 32770.0	BOOL	
5		bit_M_65522_0		N 05500.0	000	
6		bit_M_7_5	Export			? 🔀
7		byte_MB_3				
8		byte_MB_32771	Save in: 🗁 ProtTe:	st		- ⊡
9		byte_MB_65523	DeatTast Sumbola	01.555		
10		char_MB_7	ProtTest_Symbols_01.asc			
11		ComplexDataTypes	ProtTest_Symbols_	_uz.asc		
12		Cycle Execution				
13		date_MVV_24641				
14		dint_MD_32773				
15		dint_MD_5				
16		dint_MD_65525				
17		dword_MD_0				
18		dword_MD_32768	Elements D. J.			
19		dword_MD_65520	File name: ProtTe	st_Symbols_0	2.asc	Save
20		int_MVV_32774	Source have ACCIL	E (* A.C.C.)		▼ Cancel
21		int_MVV_6	Save as type: ASCII	Format (*.ASC)		Cancel
22		int_MVV_65528		1000 00020		
23		real_MD_32777		MD 32777	REAL	

Exporting Sources

These files are created exporting source code.

- 1. Open any program block in the editor, "OB1" in this example.
- 2. From the File menu choose Generate Source: the following dialog is displayed:

LAD/STL/FBD - [OB1 "Cy <u>cle Exec</u> u	ution" \$7_ProtocolTest\SIMATIC 300 S	tation\CPU315-2DP(1)\\OB1]
🕞 File Edit Insert PLC Debug View	v Options Window Help	
	60 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	□ ⊑ ﷺ <u>++++-○ः⊔⊐≻</u> №
	Contents 0	f: 'Environment\Interface'
	Interface Name TEMP	
	· · · · · · · · · · · · · · · · · · ·	
E blocks	New	
5FC blocks	Entry point:	View:
Multiple instances	Project 🗨	Component view C Online © Offline
	Name:	Storage path:
	S7_ProtocolTest	C:\Program Files\Siemens\Step7\s7pr Browse 🗈 📰
		ProtTest_All ProtTest_UDT_etc
	0B1 : SIMATIC 300 Station	ProtTest_All 📄 ProtTest_UDT_etc
	🖻 🔛 CPU315-2DP(1)	
	Commer S7 Program(1) Sources	
	Networ Blocks	
	Conner	
		Object name: Sources
		Object type:
		Ubject type: STL Source
	ΟΚ	Cancel Help
		Lancei Help
	+D T "dint MD 5"	

1. Assign a name, "Sources" in the example, and click **OK**: the **Generate source Sources** dialog is displayed.

Generate source Sources				
Note: Automatic generation of single sources per block: Menu 'Options' > 'Customize' in the 'Sources' tab				
Path: S7_ProtocolTest\SIMATIC 300 Station\CF	PU315-2DP(1)\			
Blocks Not Selected:	Blocks Selected:			
> [All > [<	DB1 UDT Struct Test OB1 Cycle Execution UDT1 BasicDataType_UDT UDT2 ComplexDataTypes			
Name/Family:				
	Addresses			
Include reference blocks	C Absolute			
Sort according to program structure	Symbolic			
ОК	Cancel Help			

- 2. Click **All >** to generate source for all blocks.
- 3. Select the following options:
- Include reference blocks
- Sort according to program structure
- Symbolic address
- 4. Click **OK** to confirm: the "Sources" object is generated in the Step7 project as in the example.

SIMATIC Manager - S7_ProtocolTest						
File Edit Insert PLC View Options Window Help						
🗅 😂 🔡 🐖 👗 🛍 🛍 🖆 😰 🏪 🕒 📴 🖽 🏛 🔍 < No Filter >	💽 🏹 器 😂 🖷 🖽 🕅 😵					
S7_ProtocolTest C:\Program Files\Siemens\Step7\s7proj\\$7_Pro~1						
Station ProtTest_All ProtTest_UDT_etc Sources SIMATIC 300 Station Sources Sources Sources Sources Blocks Blocks Sources						

5. Right click on the object and select **Export Sources**.

主 < No Filter >	- V P	🖁 🗐 🖷	
proj\\$7_Pro~1			
t_UDT_etc 🗎 Sources			
	Open Object	Ctrl+Alt+O	
	Cut	Ctrl+X	
	Сору	Ctrl+C	
	Paste	Ctrl+V	
-	Delete	Del	
	Insert New Object PLC		•
-	Compile	Ctrl+B	
	Export Source		
_	Print		•
	Rename	F2	
	Object Properties	Alt+Return	
	Special Object Properties		•

The generated .awl file can be imported in the Tag Editor.



Note: The .awl file contains additional information not included in the .asc file exported from the symbol table.

Make sure that reference to all data blocks is inserted in the symbol table. The tags from a data block are imported only if the symbol table contains a line with the data block name and related comment.

s	7 Pro	gram(2) (Symbols) CPU3	14C-2PNDP	_MPI_187K\SIMATIC	S7-300 Station 1\CPU 314C-2 PN/DP	
		Status	Symbol 🛆	Address	Data type	Comment	
1			CPU_FLT	OB 84	OB 84	CPU Fault	
2			I/O_FLT2	OB 83	OB 83	I/O Point Fault 2	
3		_	OBNL_FLT	OB 85	OB 85	OB Not Loaded Fault	
4		- T	Prova Data Block	DB 123	DB 123		
5			Prova MB0	MB 0	BYTE		
6			VAT_1	VAT 1			
7	·						

Each entry enables the import filter to import the tags related to the specified data block.

Tag Editor Settings

Into Tag editor select the protocol "Simatic S7 MPI" from the list of defined protocols and add a tag using [+] button.

Simatic S7 MPI					×
Simatic S7 MPI					
Memory Type Internal Memory 🗸	Offset	SubIndex			
Data Block	Data Type		Arraysize		
1	≜ boolean	•	0		
Conversion	+/-				
	0	K (Cancel	Apply	Help

Tag settings can be defined using the following dialog:

Element	Description						
Memory	Area of PLC where tag is	located.					
Туре	Data Type		Simatic Type				
	Internal Memory		М				
	Data Block		DB				
	Input		I (E)				
	Output		O (A)				
	Timer value		Т				
	Counter value		С				
Offset	Offset address where tag	g is located.					
SubInde x	In case of Boolean data	type, this is the offset o	of single bi	t.			
Data Block	If Memory Type is "Data	Block", this will identif	y the DB n	umber.			
Data Type	Data Type Memory Space			Limits			
туре	boolean	1 bit data		01			
	byte	8-bit data		-128 127			
	short	16-bit data		-32768 32767			
	int	32-bit data		-2.1e9 2.1e9			
	unsignedByte	8-bit data		0 255			
	unsignedShort	16-bit data		0 65535			
	unsignedInt	32-bit data		04.2e9			
	float	IEEE single-precisio	n	1.17e-38 3.40e38			
		32-bit floating point	уре				
	string	Refer to "String data	type chan	nel"			
	Note: to define arrays, select one of Data Type format followed by square brackets like "byte[]", "short[]"						
Arraysiz	In case of array ta	ag, this property repres	sents the n	umber of array elements.			
e	 In case of string taging the string tag. 	ag, this property repre	sents the n	naximum number of bytes available			

Element	Description				
	Note: number of bytes corresponds to number of string characters if Encoding property is set to UTF-8 or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2 bytes.				
Conversi	Conversion to be a	pplied to the tag.			
on	Conversion				
	inv,swap2 Depending on data below.	Allowed BCD AB->BA ABCD->CDAB ABCD=>CDAB ABCD=>CDAB ABCD=>CDAB ABCD=>CDAB Inv bits Cancel OK type selected, the Allowed list shows one or more conversions, listed			
	Value	Description			
	Inv bits	Invert all the bits of the tag. <i>Example:</i> $1001 \rightarrow 0110$ (in binary format) $9 \rightarrow 6$ (in decimal format)			
	Negate	Set the opposite of the tag value. <i>Example:</i> $25.36 \rightarrow -25.36$			
	AB -> BA	Swap nibbles of a byte. <i>Example:</i> $15D4 \rightarrow 514D$ (in hexadecimal format) $5588 \rightarrow 20813$ (in decimal format)			
		Swap bytes of a word. <i>Example:</i> $9ACC \rightarrow CC9A$ (in hexadecimal format)			
	ABCD -> CDAB	Example:			

	Description			
	$855441236 \rightarrow 1426062386$ (in decimal format)			
	Swap bytes of a long word. Example: $142.366 \rightarrow -893553517.588905$ (in decimal format) 0.1000000110 0001110010111011001000101101000011100101			
	Separate the byte in two nibbles, and reads them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)			
r(BCD)	Used to support S5timer. Check Simatic S5timer special data type for more details.			
r(BIN)	Legacy transformation for S5timer in binary format.			
r	NOP -> DAB			

Use the arrow buttons to order the configured conversions.

Aliasing Tag Names in Network Configurations

Tag names must be unique at project level; it often happens that the same tag names have to be used for different controller nodes (for example when the HMI is connected to two devices that are running the same application). Since tags include also the identification of the node and Tag Editor does not support duplicate tag names, the import facility in Tag Editor has an aliasing feature that can automatically add a prefix to imported tags. With this feature tag names can be done unique at project level.

The feature works when importing tags for a specific protocol. Each tag name will be prefixed with the string specified by the "Alias". As shown in the figure below, the connection to a certain controller is assigned the name "Node1". When tags are imported for this node, all tag names will have the prefix "Node1" making each of them unique at the network/project level.

F — 🗸 🗸 🛛	b 🖷 🚺	Simatic S7 ETH:prot	:1	- 5		
Name 🛆		Group	Driver	Addr	ess	Comment
Node 1/End_hodrotia		/HE212 WHA	ditua TCP petri	1 TT Dunk	palition	
Node1/Oata_hodrotia		(HE212) WHAT	dbue TCP petri	1 12 Dunk	predition	
Node1/IN_W/R7EPUlev	rel la	/H212 VHz	dbue TCP pet1	1 DDunkig	realShort	
Node1 / OBOX5#BHOW		/#1212 VHo	dbue TCP petri	1 245/Dune	grad that	
Note: CUT_BAT_Need		#411 We	ditua TCP petri	1 1 Dunkig		
Nedle 1/R. D/RTA Justin	dia	/H212 Wite	dbue TCP petri	1 2 Dunkig		
No. Secti MAX TELE		/H212 Wite	dbue TCP petri	1 3 Dunkig	matthat	
Node1/Water_level		ABECO	Telline ETH and T	1 10 0 unsi	gr <mark>edShort</mark>	
	() N	lode id as defined in ir	nport file			
	© S	lode id as defined in ir ielect Network node io Slave Id	Model		Alias	
	© S	ielect Network node ic Slave Id	Model		Node1	
	© S	ielect Network node ic Slave Id	Model			
Karame −	© 5	ielect Network node ic Slave Id	Model		Node1	
taoname 🔽	© S	ielect Network node ic Slave Id	Model		Node1 Node2	
	© S	ielect Network node ic Slave Id	Model		Node1	Cancel
taoname Water_level	© S	ielect Network node ic Slave Id	Model		Node1 Node2	Cancel
tagname Water_level	© S	ielect Network node ic Slave Id	Model		Node1 Node2	Cancel
tagname Water_level	© S	ielect Network node ic Slave Id	Model	- Thu	Node1 Node2	Cancel
tagname C	e s	ielect Network node id Slave Id	Model	- Fina (Node 1 Node 2 Ok	Cancel
tagname Vater_level	e s	ielect Network node id Slave Id	Model		Node1 Node2 Ok	Cancel



Note: Aliasing tag names are only available when tags can be imported. Tags which are added manually in the Tag Editor do not need to have the Alias prefix in the tag name.

The Alias string is attached to the tag name only at the moment the tags are imported using Tag Editor. If Alias string is modified after the tag import has been completed, there will be no effect on the names already present in the dictionary. When the Alias string is changed and tags are imported again, all tags will be imported again with the new prefix string.

String data type

In Simatic S7 PLC it's possible to define two different types of tags to manage string variables.

- as Array [1..xx] of Chars.
- as String[xx].

Step7 string declaration is showed in the following figure:

ddress	Name	Туре	Initial value	Comment	S7 String
0.0		STRUCT			Sr Stillig
+0.0	Stringl	STRING[254]	'sample'		
+256.0	String2	ARRAY[110]			
*1.0		CHAR		The subscription of the local division of the local division of the local division of the local division of the	String as array of char
=266.0		END_STRUCT			

TIA Portal string declaration is showed in the following figure:

_		🎭 🛃 🎼 🕾 🐼 🕻 _block_1		S7 S	String			
	_	ime	Data type	Uffset	Start value	Retain	Accessible	Visible in
1	-	Static						
2	1 -	String1	String		'sample'			
3	•	 String2 	Array [1 10] of Char					
					String as a	rrav of cha	ar 🗋 👘	
					L J	,		

Note: Usage of String[xx] data type is allowed but a specific Conversion must be applied to the tag. Anyway using tag importer to import tag dictionary from TIA Portal or Step7 string tags are automatically configured and no changes/conversion are needed.

To manually add an "Array [1..xx] of Chars" data type tag, press the [+] button in the Tag Editor, then select "string" as Data Type of the Tag and type the string length in the "Arraysize" field:

Simatic S7 ETH	
Simatic S7 ETH	
Memory Type	Offset SubIndex
Data Block	▼ 114 ● 0 ▼
Data Block	Data Type Arraysize
1	string 🗸 10
Conversion	
1	+/-
	OK Cancel Apply Help

and confirm with OK button.

ï

To manually add a "String[xx]" data type tag, press the [+] button in the Tag Editor, then select "string" as Data Type of the Tag and type the string length in the "Arraysize" field,

Simatic S7 ETH		
Simatic S7 ETH	1	_
Memory Type Data Block	Offset SubIndex ▼ 114 ♥ 0 ▼	
Data Block	Data Type Arraysize String 10	
Conversion	+/-	
	OK Cancel Apply Help	
		J

then click on [+/-] button to open the Conversion dialog.

Simatic S7 ETH			—X —
Simatic S7 ETH			
Memory Type	Offset	SubIndex	
Data Block	▼ 114 ₹	0 -	
Data Block	Data Type	Arraysize	
1	string	▼ 10	
Conversion			
1	+/-		
	ОК	Cancel	Apply Help

Into conversion dialog:

- select the "S7 String" conversion type
- click on [+] button to add the conversion.

matic S7 ETH			Σ
Simatic S7 ETH			
Memory Type	Offset S	ubIndex	
Data Block	▼ 114 ▲	0 -	
Data Block	Data Type	Arraysize	
1	string	▼ 10	
Conversion	B		
	Allowed	Configured	
	AB->BA ABCD->CDAB	S7 String	
	S7 String		
			р
		Cancel OK	_

The conversion will be listed into the Configured window on the right.

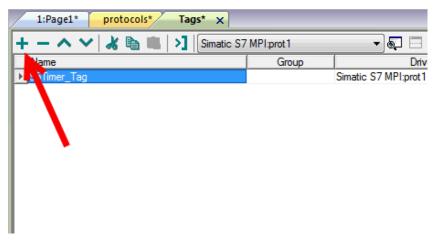
Confirm with OK button.

Simatic S5timer data type

Simatic drivers support a special data type, called S5Timer.

The tag must be configured with a specific data type and a conversion must be applied to the Tag to correctly read/write a Simatic S5Timer Variable.

Open the Tag Editor and add a Tag pressing the Plus button.



Select "unsignedInt" as Data Type of the Tag.

imatic S7 MPI	×
Simatic S7 MPI	
Memory Type	Offset SubIndex
Internal Memory 👻	50 A 0 V
Data Block	Data Type Arraysize
1	
Conversion	boolean byte short int
	+/- unsignedByte unsignedShort
	unsignedInt float string
	OK Cancel Apply Help

Click on +/- button to open the Conversion dialog.

1	t Data Type	SubInd	▼			
Internal Memory 50 Data Block 1	×		▼			
Internal Memory 50 Data Block 1	×		▼			
Data Block		0				
1						
1	Data Type		Ö PP ST			
·			Array	size		
	unsignedInt		• 0]	
Conversion						
I <u>+</u>						
	Ok	K	Cancel		Apply	Help

In the Conversion dialog select the S5timer(BCD) conversion type [A] then click on Plus button [B] to add the conversion, the configured conversion will be listed into the Configured window on the right. Then confirm with OK.

matic S7 MPI	1. Contract (1997)	×
Simatic S7 MPI		
Memory Type	Offset SubIndex	
Internal Memory -	50 💌 0 👻	
Data Block	Data Type Arraysize	
1	unsignedInt 🗸 0	
_		
Conversion A	B	
	Allowed Configur	
	S5timer(BCD) S5timer(S5timer(BIN)	BCD)
	^	
	< 📄 Þ 🗸	Help
	Cancel	ОК

Tag Import

Select the driver in Tag Editor and click on the Import Tags button to start the importer.

_	Tags	×								
+	_	X	D	ß	>]	₽	A 9B	B>	ŧ i ł	1
Data			^		-	Та	g URI			

The following dialog shows which importer type can be selected.

HMIStudio	
Multiple tag importers are	e available for this protocol. Please select the importer type and continue.
Version	Туре
TIA Portal Project v 12 or newer	Linear
TIA Portal v13, v14 or newer	Linear
TIA Portal v10, v11, v12	Linear
Step7	Linear
Tag Editor exported xml	General
	OK Cancel

Importer	Description			
TIA Portal Project v12 or newer Linear	Allows to import the whole TIA Portal project file using .apxx file (where "xx" is the TIA Portal version, example: for TIA Portal 13, file name is "project.ap13").			
	All variables will be displayed at the same level.			
TIA Portal v13, v14 or newer Linear	 Allows to import: Program blocks using .db file PLC tags using .xlsx file PLC data types using .udt file Check Export using TIA Portal v13, v14 or newer for more details. 			
	All variables will be displayed at the same level.			
TIA Portal v10, v11, v12 Linear Step7 Linear	 Allows to import: Program blocks using .tia file PLC tags using .xlsx file PLC data types using .scl file Check Export using TIA Portal v10, v11, v12 for more details. All variables will be displayed at the same level. Allows to import: Symbols table .asc file Sources using .awl file 			
	Check Export using STEP7 for more details.			
	All variables will be displayed at the same level.			
Tag Editor exported xml	Select this importer to read a generic XML file exported from Tag Editor by appropriate button.			

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Tags × Protocols					
+ - 🎽 🕲 刘	D \$8 ₪ 13	R 🔎 - Search Tilter by: Data	▼ Ite	ms used:6/10000 Protocol: Show all	🗹 Show all tags 🖉 🗖
Data	Туре	Comment	^	Property	Value
Modbus TCP:prot1	Container			Y Driver	
Model: Modicon Modbus(1-based)				Model	Modicon Modbus(1-based)
	unsignedShort			Protocol	Modbus TCP:prot1
Holding Registers 2	unsignedShort			✓ Dictionary	
 Holding Registers 3 MRTU1 	unsignedShort unsignedShort			Array	false
- MRTU2	unsignedShort			Array size	0
MRTU3	unsignedShort			Arravindex.Subindex	400003
- MRTU4	unsignedShort			Comment	
- MRTU5	unsignedShort			Data type	unsignedShort

Toolbar item	Description
₿ X	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
B 国	Update Tag(s). Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag. Example of both checked and unchecked result:
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Communication status

The communication status can be displayed using the dedicated system variables. Please refer to the User Manual for further information about available system variables and their use.

The status codes supported for this communication driver are:

Error	Notes
NAK	Controller replies with a not acknowledge.
Timeout	Request is not replied within the specified timeout period; ensure the controller is connected and properly configured for network access
Invalid	The device did receive from the controller a response, but its format or its contents or its

Error	Notes
response	length is not as expected; ensure the data programmed in the project are consistent with the controller resources.
General Error	Error cannot be identified; should never be reported; contact technical support

System Variables

System Variables communication driver allows to create Tags that point to system information.

Refer to <u>System Variables > Protocol</u> chapter of User's Manual.

Protocol Editor Settings

System Variables communication driver allows to create Tags that point to system information.

Refer to <u>System Variables > Protocol</u> chapter of User's Manual.

Variables

Variables communication driver allows to define Tags which points to HMI internal memory.

Variables Tags are not retentive: when the project starts, the starting value of any Variables Tag is 0 (or "" in case of string Tag).



Variables communication driver is not counted as physical protocol.

Refer to Table of functions and limits from main manual in "Number of physical protocols" line.

Protocol Editor Settings

Adding a protocol

To configure the protocol:

- 1. In the **Config** node double-click **Protocols**.
- 2. To add a driver, click +: a new line is added.
- 3. Select the Variables protocol from the PLC list.

Tag Editor Settings

Path: ProjectView> Config > double-click Tags

- 1. To add a tag, click +: a new line is added.
- 2. Select Variables from the protocol list: tag definition dialog is displayed.

Variables		×
Variables		
Data type	Arraysize Conversion	
boolean 👻	0	+/-
	OK Cancel	Apply Help

Element	Description					
Data Type	Data Type	Memory Space	Limits			
	boolean	1-bit data	01			
	byte	8-bit data	-128 127			
	short	16-bit data	-32768 32767 -2.1e9 2.1e9 -9.2e18 9.2e18 0 255			
	int	32-bit data				
	int64	64-bit data				
	unsignedByte	8-bit data				
	unsignedShort	16-bit data	0 65535			
	unsignedInt	32-bit data	0 4.2e9 0 1.8e19 1.17e-38 3.4e38 2.2e-308 1.79e308			
	uint64	64-bit data				
	float	IEEE single-precision 32-bit floating point type				
	double	IEEE double-precision 64-bit floating point type				
	string	Array of elements containing character code defined by selected encoding				
	binary Arbitrary binary data					
	Note: to define arrays. select one of Data Type format followed by square brackets like "byte[]", "short[]"					
Arraysize	 In case of array tag, this property represents the number of array elem In case of string tag, this property represents the maximum number of available in the string tag. 					
	Note: number of bytes corresponds to number of string characters if Encoding property is set to UTF-8 or Latin1 in Tag Editor. If Encoding property is set to UCS-2BE, UCS-2LE, UTF-16BE or UTF-16LE one character requires 2 bytes.					
Conversion	Conversion to be applied to the tag.					

Description			
Conversion			
inv,swap2 Depending on c	Allowed BCD AB->BA ABCD->CDAB ABCDEFGH->GHEFCDAB Inv bits Cancel OK Cancel OK		
Value	Description		
Inv bits	inv : Invert all the bits of the tag.		
	<i>Example:</i> 1001 \rightarrow 0110 (in binary format) 9 \rightarrow 6 (in decimal format)		
Negate	neg: Set the opposite of tag value.		
	<i>Example:</i> 25.36 → -25.36		
AB -> BA	swapnibbles: Swap nibbles in a byte.		
	<i>Example:</i> 15D4 \rightarrow 514D (in hexadecimal format) 5588 \rightarrow 20813 (in decimal format)		
ABCD ->	swap2: Swap bytes in a word.		
CDAB	<i>Example:</i> 9ACC \rightarrow CC9A (in hexadecimal format) 39628 \rightarrow 52378 (in decimal format)		
ABCDEFGH	swap4: Swap bytes in a double word.		
-> GHEFCDAB	Example: 32FCFF54 \rightarrow 54FFFC32 (in hexadecimal format) 855441236 \rightarrow 1426062386 (in decimal format)		
ABCNOP -	swap8: Swap bytes in a long word.		
> OPMDAB	Example: 142.366 → -893553517.588905 (in decimal format) 0 1000000110 0001110010111011001000101101000011100101		

Element	Description		
	Value	Description	
		0001 \rightarrow 1 10000011100 1010100001010001011011011001011011000010011 1101 (in binary format)	
	BCD	bcd : Separate byte in two nibbles, read them as decimal (from 0 to 9) <i>Example:</i> $23 \rightarrow 17$ (in decimal format) $0001\ 0111 = 23$ 0001 = 1 (first nibble) 0111 = 7 (second nibble)	
	Select conversi	on and click +. The selected item will be added to list Configured .	
	If more conversions are configured, they will be applied in order (from top to bottom of Configured).		
	Use the arrow b	outtons to order the configured conversions.	

Tag Import

Select the driver in Tag Editor and click on the Import Tags button to start the importer.



The system will require a generic XML file exported from Tag Editor by appropriate button.

	Tags	×							
+	-	×	C	>]	Þ	A 9B	B>	ŧ3	1
Data			^		Та	g URI			

Once the importer has been selected, locate the symbol file and click **Open**.

The tags available within the Dictionary but not imported into the project are gray and are visible only when the "Show all tags" check box is selected.

Т	Fags × Protocols					
+	- 🎽 🕲 🖉 🔰	(> \$8 D) 53	R 🔎 - Search 🍸 Filter by: Data	▼ Ite	ms used:6/10000 Protocol: Show all	Show all tags 💮 🗔
Data	^	Туре	Comment	^	Property	Value
	odbus TCP:prot1	Container			✓ Driver	
- Mo	odel: Modicon Modbus(1-based)				Model	Modicon Modbus(1-based)
		unsignedShort			Protocol	Modbus TCP:prot1
		unsignedShort			✓ Dictionary	
1	- Holding Registers 3 - MRTU1	unsignedShort unsignedShort			Array	false
	-MRTU2	unsignedShort			Array size	0
	MRTU3	unsignedShort			Arravindex.Subindex	400003
	- MRTU4	unsignedShort			Comment	100000
	MRTU5	unsignedShort			Data type	unsignedShort
		1-1-1			butu type	draight donor c

Toolbar item	Description
ka	Import Tag(s).
	Select tags to be imported and click on this icon to add tags from tag dictionary to the project
B a	Update Tag(s). Click on this icon to update the tags in the project, due a new dictionary import.
R	Check this box to import all sub-elements of a tag. Example of both checked and unchecked result: Image: A line of both checked and unchecked result: Image: A line of both checked and unchecked result: Image: A line of both checked and unchecked result: Image: A line of both checked and unchecked result: Image: A line of both checked and unchecked result: Image: A line of both checked and unchecked result: Image: A line of both checked and unchecked result: Image: A line of both checked and unchecked result: Image: A line of both checked and unchecked result: Image: A line of both checked and unchecked result: Image: A line of both checked and unchecked result: Image: A line of both checked and unchecked result: Image: A line of both checked and unchecked result: Image: A line of both checked and unchecked result: Image: A line of both checked and unchecked result: Image: A line of both checked and unchecked result: Image: A line of both checked and unchecked result: Image: A line of both checked and unchecked and unchecked result: Image: A line of both checked and unchecked and unchecked result: Image: A line of both checked and unchecked and unchecked result: Image: A line of both checked and unchecked and unchecked result: Image: A line of both checked and unchecked and unche
P- Search	Searches tags in the dictionary basing on filter combo- box item selected.

Record of changes

Manual number can be found at the bottom of the cover page.

Date	Manual No.	Record of Changes
Dec.2020	WUME-XASCEN-COM-01	1st Edition
May.2021	WUME-XASCEN-COM-02	2nd Edition Error correction
Sep.2022	WUME-XASCEN-COM-03	 3rd Edition Upgrading the version of xAscender Studio (Version: 4.5) Added description of Force Read Single configuration to Panasonic FP / FP7 protocol Added "Environment Variables" chapter Added "Client System Variables" chapter Remove all unsupported protocols below CAN Direct v2.0x CANopen HMI CANopen SDO J1939 KNX TP / IP Lenze CANopen NMEA 2000 Profibus DP Profibus DP S7

Panasonic Industry Co., Ltd. Panasonic Industrial Devices SUNX Co., Ltd. https://panasonic.net/id/pidsx/global

 Please visit our website for inquiries and about our sales network.

 Panasonic Industrial Devices SUNX Co., Ltd. 2022

 September, 2022
 WUME-XASCEN-COM-03