FREE Studio v3.9

Release Notes

Software / Firmware Version:

EVD75●●	Firmware msk: 423.26
EVC75●●	Firmware msk: 477.26
EVP3●●●	Firmware msk: 489.19
EVK1000	Firmware msk: 476.19
AVD126●●/AVC126●●/AVD84●●/AVC84●●	Firmware msk: 596.06
AVD62●●/AVC62●●/AVD30●●/AVC30●● (first official release)	Firmware msk: 640.03
AVP1●●●	Firmware msk: 659.04
FREE Studio	v 3.9.1

This document contains important information about the FREE Studio software. Please read the complete document before you run the product.

This documentation is not intended as a substitute for and is not to be used for determining suitability or reliability of these products for specific user applications. It is the duty of any such user or integrator to perform the appropriate and complete risk analysis, evaluation and testing of the products with respect to the relevant specific application or use thereof. Neither Eliwell nor any of its affiliates or subsidiaries shall be responsible or liable for misuse of the information contained herein. All pertinent state, regional, and local safety regulations must be observed when installing and using this product. When devices are used for applications with technical safety requirements, the relevant instructions must be followed. Failure to use Eliwell software or approved software with our hardware products may result in injury, harm, or improper operating results. Failure to observe this information can result in injury or equipment damage. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us. No part of this document may be reproduced in any form or by any means, electronic or mechanical, including photocopying, without express written permission of Eliwell.

Table of Contents

1. SYSTEM REQUIREMENTS

Eliwell FREE Studio can be installed on a personal computer having the following characteristics:

Operative System

- Windows 7 Home Premium 32 Bit & 64 Bit
- Windows 7 Professional 32 Bit & 64 Bit
- Windows 7 Ultimate 32 Bit & 64 Bit
- Windows 8 / 8.1 64 Bit
- Windows 10 64 Bit

Hardware requirements

- Processor — Pentium 1.6 GHz or greater
- RAM Memory — 1 GB; 2 GB preferred
- Hard Disk — 500 MB of free space
- Peripherals — Mouse or compatible pointing device
- Peripherals — USB interface
- Web access — Web registration requires Internet access

Please note the following information from Microsoft:

Windows XP support has ended

As of April 8, 2014, support and updates for Windows XP are no longer available.

FREE Studio requires Administrator rights to be installed.

For further information, contact your Eliwell support center.

2. IMPORTANT INFORMATION

2.1. Provided templates and project examples

**********Disclaimer of Warranty**********

THE INFORMATION CONTAINED HEREIN, AND WITHIN ANY RELATED ELIWELL DOCUMENTATION, IS PROVIDED "AS IS" WITHOUT WARRANTIES OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION, ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT OR OTHER VIOLATION OF RIGHTS. ELIWELL DOES NOT WARRANT OR MAKE ANY REPRESENTATIONS REGARDING THE USE, VALIDITY, ACCURACY, OR RELIABILITY OF, OR THE RESULTS OF THE USE OF, OR OTHERWISE, RESPECTING THE MATERIALS, SPECIFICATIONS, CHARACTERISTICS OR OTHER INFORMATION SPECIFIED HEREIN. FURTHERMORE, ALL WARRANTIES, CONDITIONS, REPRESENTATIONS, INDEMNITIES AND GUARANTEES WITH RESPECT TO THE ACCURACY, OPERATION, CAPACITY, SPEED, FUNCTIONALITY, QUALIFICATIONS, OR CAPABILITIES OF THE SOFTWARE, SYSTEMS AND SERVICES COMPRISING OR UTILIZED IN THE COURSE OF APPLYING THIS INFORMATION, SPECIFICATIONS, OR MATERIALS, WHETHER EXPRESS OR IMPLIED, ARISING BY LAW, CUSTOM, PRIOR ORAL OR WRITTEN STATEMENTS BY ELIWELL, OR OTHERWISE (INCLUDING, BUT NOT LIMITED TO ANY WARRANTY OF SATISFACTORY QUALITY, MERCHANTABILITY, FITNESS FOR PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT) ARE HEREBY EXPRESSLY EXCLUDED AND DISCLAIMED.

**********Disclaimer of Liability**********

UNDER NO CIRCUMSTANCES (INCLUDING NEGLIGENCE OR FORESEEABLE MISUSE) WILL ELIWELL BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION, BUSINESS INTERRUPTION, DELAYS, LOSS OF DATA OR PROFIT) ARISING OUT OF THE APPLICATION OF THE INFORMATION, SPECIFICATIONS, CHARACTERISTICS OR MATERIALS CONTAINED HEREIN EVEN IF ELIWELL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

****

The information, program and configuration contained herein, as in other Eliwell documentation, are provided as general training on the products concerned, in part, by way of theoretical and/or hypothetical examples. Only you, the user, can be aware of all the conditions and factors present during setup, operation, and maintenance of your machine and its application; therefore, only you, the user, can determine the automation equipment and the related safeties and interlocks which can be properly used. When selecting automation and control equipment and related software for a particular application, you must refer to the applicable local and national standards and regulations.

****

3. NEW FEATURES IN FREE STUDIO V3.9.1

- Availability of new targets:
  - Controllers AVD62●●/AVC62●●/AVD30●●/AVC30●●
  - Display Touch AVP1000, in portrait and landscape mode.

- New features for AVD126●●/AVC126●●/AVD84●●/AVC84●● targets:
  - Management of Modbus/RTU Master on event via PLC programming (see below, 10.1);
  - Available Address Conflict Detection (ACD) to avoid duplicate TCP-IP address (see 10.2)
  - Modbus TCP/IP improvements/fixes:
    - Increased from 3 to 4 the number of Modbus/TCP-IP sockets opened by default in listening
    - If Bridge function is not enabled the controller answers to any Unit Identifier
    - Unit Identifier of the request is now copied to the response and not fixed to 255
    - Correct management of “Protocol Identifier” field in Modbus TCP/IP frame. If the “Protocol Identifier” doesn’t correspond to Modbus protocol, target doesn’t answer
  - TCP/IP improvements:
    - Delayed ACK has been activated

- New features for AVP1●●● targets:
  - Management of Modbus/RTU Master on event via PLC programming (see below, 10.3);
  - specific keyboard and behavior for edit with custom format HH:MM

- Solved following issues on AV●126●●/AV●84●● and EV●75●● target
  - Target generate the Lonworks XIF with a custom or standard profile (8030, 8040, 8051, 8610) with a incorrect indication in the field 3 of line 8 (0 should be 1)
  - Incorrect type of nviSpaceRH in 8030 (Rooftop) Lonworks profile, now TEMP_P, must be LEV_PERCENT
  - Modified EVO.JS to solve issue with Chrome 64.0.3282.119
  - Solved problem in saving EEPROM data referred to BACnet Objects

- New features in FREE Studio
  - The hardware/software user guides are now available from the help menu of FREE Studio Application
  - Link to registration page in registration window

- Solved following issues in FREE Studio
  - on User Interface tool, if you attempt to save a project while ST editor (script editor) is open, FREE Studio UI ceases to function and project is not saved
  - In FBD, if links are pending (red links) the add of a function block and of EN/ENO delete all pending links
  - Incorrect behavior of SHL and ROL operators
  - Missing error emission on nested function call
  - Parameter on message inoperable: the use of a set inside a message page result into a compiler error

- Solved a collection of minor issues; contact your Eliwell representatives for any additional information you may need.

4. NEW FEATURES IN AVD126●●/AVC126●●/AVD84●●/AVC84●● (FIRMWARE MSK 596.06)

- Management of Modbus/RTU Master on event via PLC programming (see below, 10.1);
- Available Address Conflict Detection (ACD) to avoid duplicate TCP-IP address (see 10.2)
- Modbus TCP/IP improvements/fixes:
  - Increased from 3 to 4 the number of Modbus/TCP-IP sockets opened by default in listening
  - If Bridge function is not enabled the controller answers to any Unit Identifier
  - Unit Identifier of the request is now copied to the response and not fixed to 255
  - Correct management of “Protocol Identifier” field in Modbus TCP/IP frame. If the “Protocol Identifier” doesn’t correspond to Modbus protocol, target doesn’t answer
- TCP/IP modifications:
  - Delayed ACK has been activated
- Solved the following issues on BACnet stack:
  - continuous creation of COV events when the Present Value of an AI is negative
  - missed creation of the event Fault-to-Normal when the Present Value goes from -32768 to a “normal” value
  - Fixed problems on Object Calendar when writing dates
- Solved the following issues on LONWORKS stack:
  - Target generate the Lonworks XIF with a custom or standard profile (8030, 8040, 8051, 8610) with a incorrect indication in the field 3 of line 8 (0 should be 1)
  - Incorrect type of nviSpaceRH in 8030 (Rooftop) Lonworks profile, now TEMP_P, must be LEV_PERCENT
- Modified EVO.JS to solve issue with Chrome 64.0.3282.119
- Solved problem in saving EEPROM data referred to BACnet Objects

5. NEW FEATURES IN AVP1●●● (FIRMWARE MSK 659.04)

- Management of Modbus/RTU Master on event via PLC programming (see below, 10.3);

6. NEW FEATURES IN EVP3●●● (FIRMWARE MSK 489.19)

- Solved the following issues on BACnet stack:
  - continuous creation of COV events when the Present Value of an AI is negative
  - missed creation of the event Fault-to-Normal when the Present Value goes from -32768 to a “normal” value
  - Fixed problems on Object Calendar when writing dates
  - Fixed communications problems in BACnet MS/TP at 76800 b/s
- Modified EVO.JS to solve issue with Chrome 64.0.3282.119
- Solved problem in saving EEPROM data referred to BACnet Objects

7. NEW FEATURES IN EVK1000 (FIRMWARE MSK 476.19)

- Alignment of FW version for maintenance reasons.

8. NEW FEATURES IN EVD75●● (FIRMWARE MSK 423.26)

- Solved the following issues on BACnet stack:
  - continuous creation of COV events when the Present Value of an AI is negative
  - missed creation of the event Fault-to-Normal when the Present Value goes from -32768 to a “normal” value
  - Fixed problems on Object Calendar when writing dates
- Solved the following issues on LONWORKS stack:
  - Target generate the Lonworks XIF with a custom or standard profile (8030, 8040, 8051, 8610) with a incorrect indication in the field 3 of line 8 (0 should be 1)
  - Incorrect type of nviSpaceRH in 8030 (Rooftop) Lonworks profile, now TEMP_P, must be LEV_PERCENT
- Modified EVO.JS to solve issue with Chrome 64.0.3282.119
- Solved problem in saving EEPROM data referred to BACnet Objects

9. NEW FEATURES IN EVC75●● (FIRMWARE MSK 477.26)

- Solved the following issues on BACnet stack:
  - continuous creation of COV events when the Present Value of an AI is negative
  - missed creation of the event Fault-to-Normal when the Present Value goes from -32768 to a “normal” value
  - Fixed problems on Object Calendar when writing dates
- Solved the following issues on LONWORKS stack:
  - Target generate the Lonworks XIF with a custom or standard profile (8030, 8040, 8051, 8610) with a incorrect indication in the field 3 of line 8 (0 should be 1)
  - Incorrect type of nviSpaceRH in 8030 (Rooftop) Lonworks profile, now TEMP_P, must be LEV_PERCENT
- Modified EVO.JS to solve issue with Chrome 64.0.3282.119
- Solved problem in saving EEPROM data referred to BACnet Objects

10. DOCUMENTATION UPDATE

- FREE Advance Logic Controller Hardware User Guide has been updated in order to include the new devices of the offer: AVD62●●/AVC62●●/AVD30●●/AVC30●● and AVP1000 (flush mounting version)

11. NEW FEATURES DESCRIPTION

11.1. Management of Modbus/RTU Master on event via PLC programming for AVD●●●●/AVC●●●●

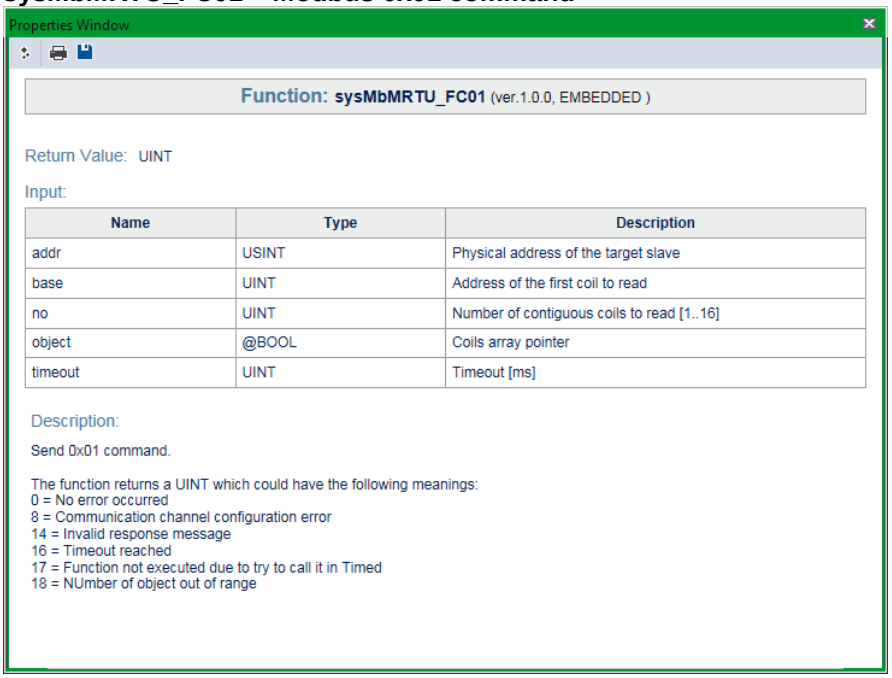
The following target functions can be used to send Modbus commands via PLC programming.

These functions require that Modbus/RTU Master (for field) is enabled via FREE Studio Connection and can be used only in Background Task of the PLC.

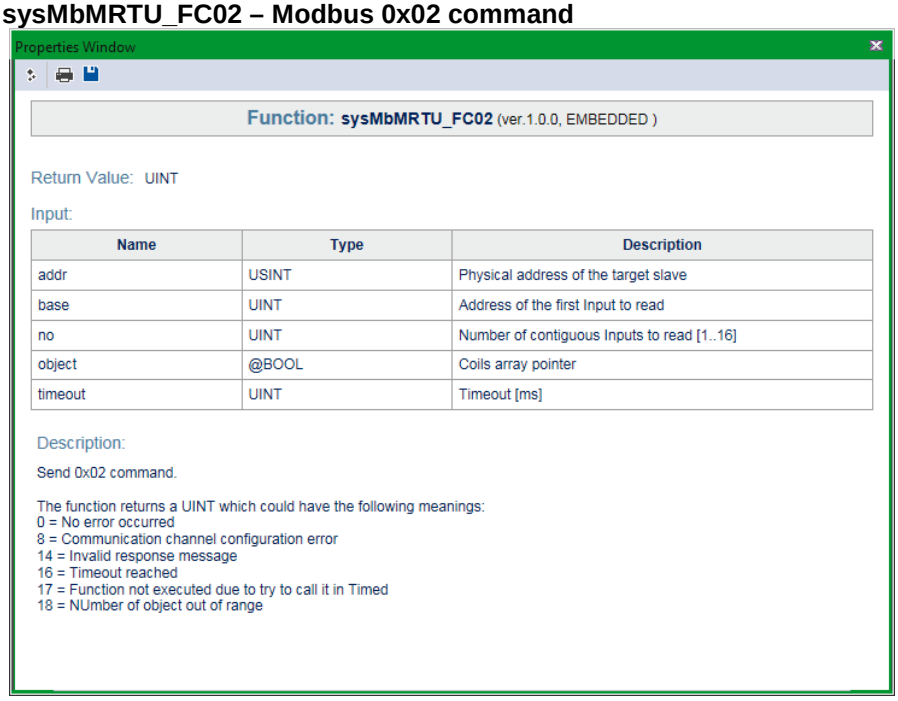
When one of these functions is executed the task is suspended till the answer from the field.

If a network of Modbus RTU slaves is configured in Connection, when the function is executed by the PLC it suspends the queue of commands programmed via Connection, sends the command, wait for the answer and finally resumes the queue.

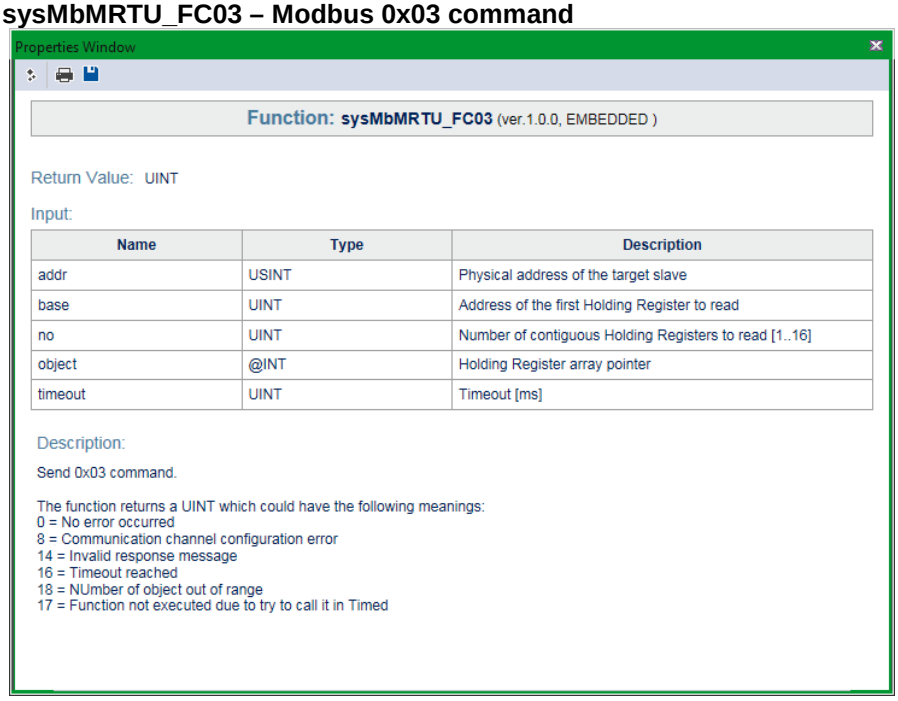
sysMbMRTU_FC01 – Modbus 0x01 command



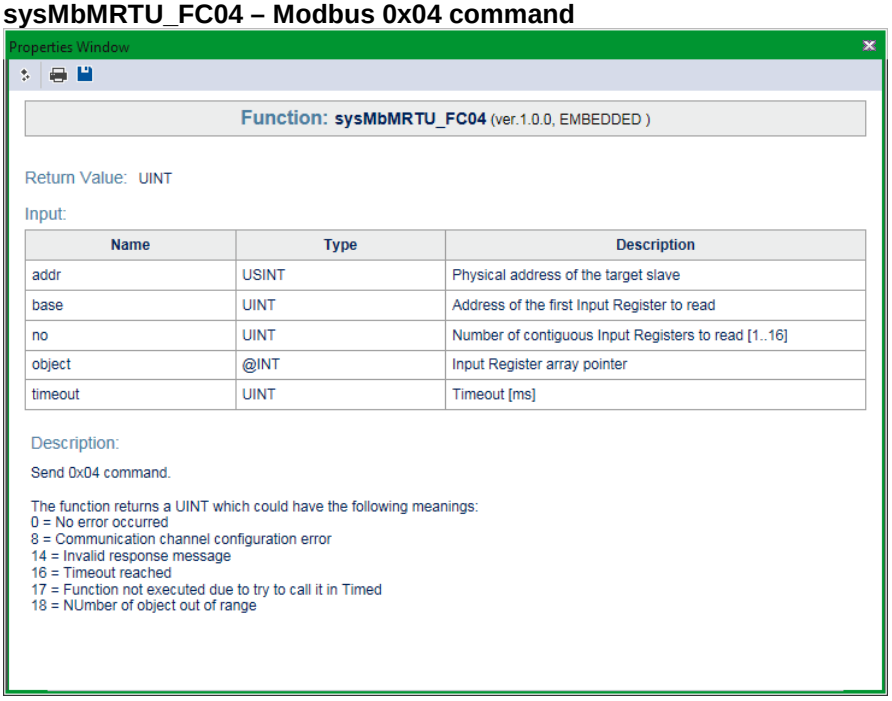
sysMbMRTU_FC02 – Modbus 0x02 command



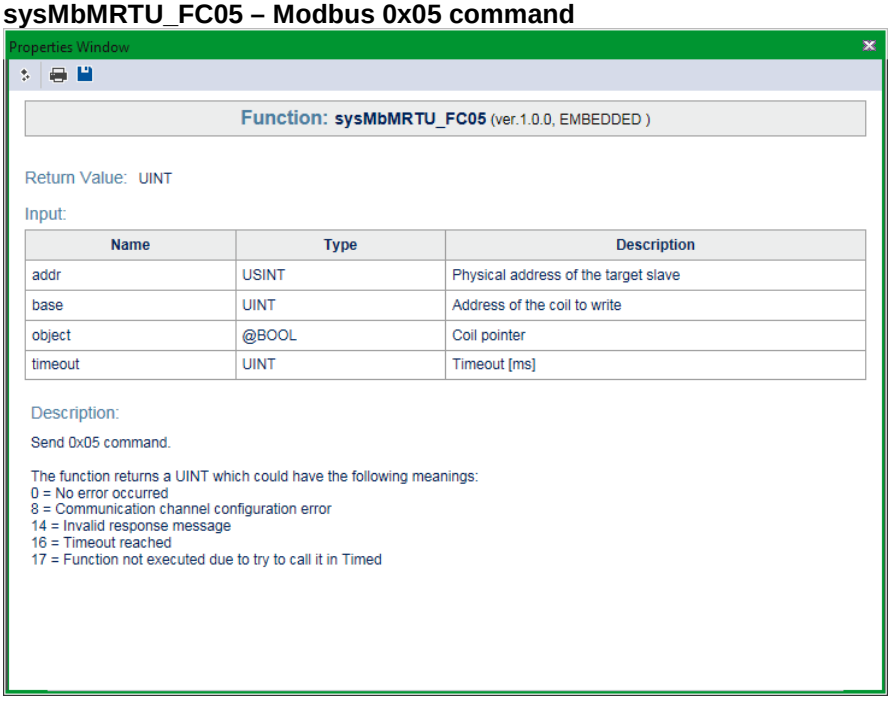
sysMbMRTU_FC03 – Modbus 0x03 command



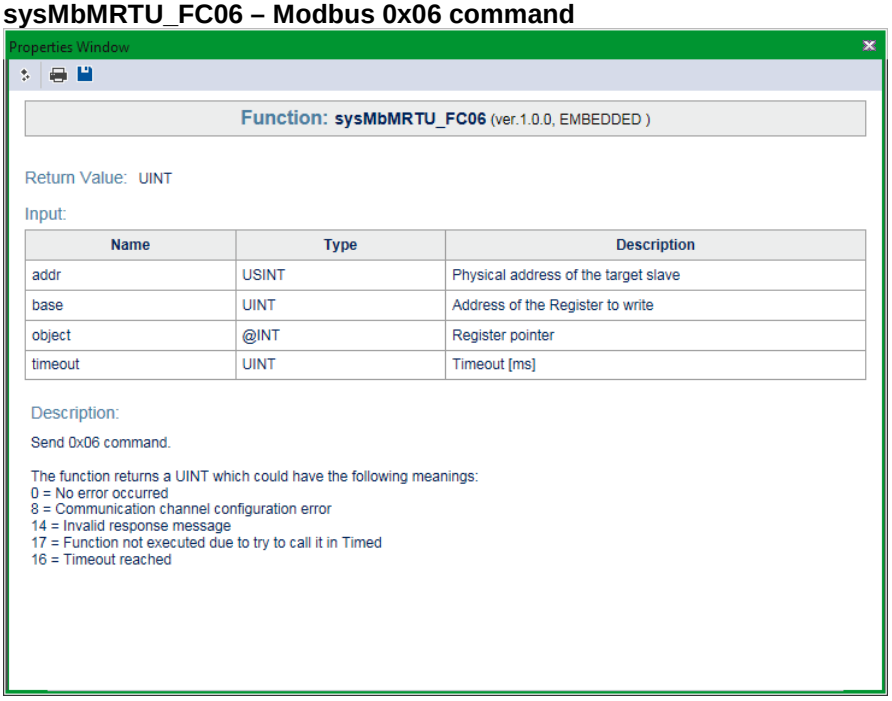
sysMbMRTU_FC04 – Modbus 0x04 command



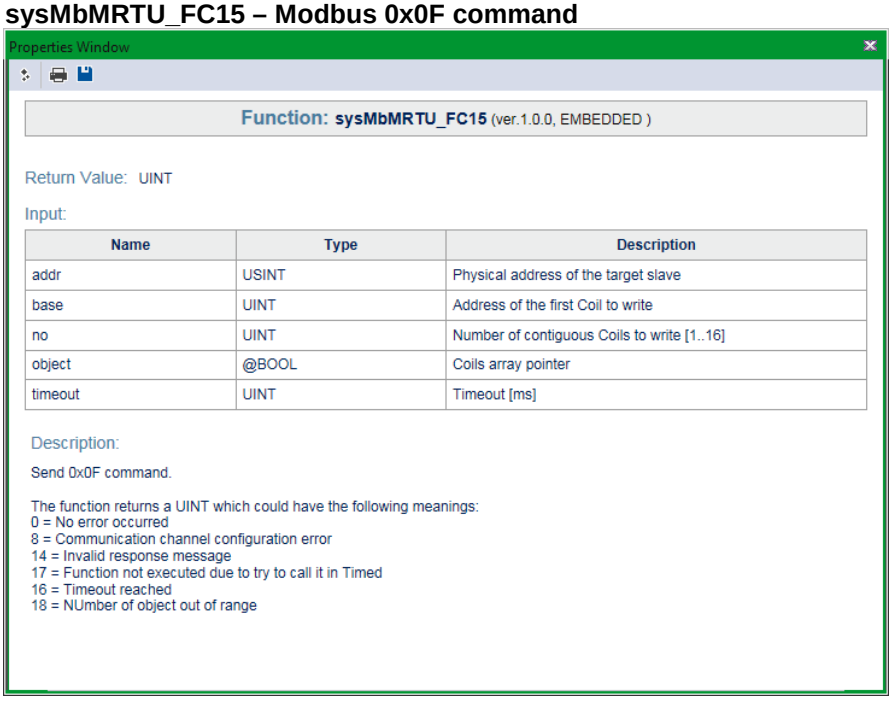
sysMbMRTU_FC05 – Modbus 0x05 command



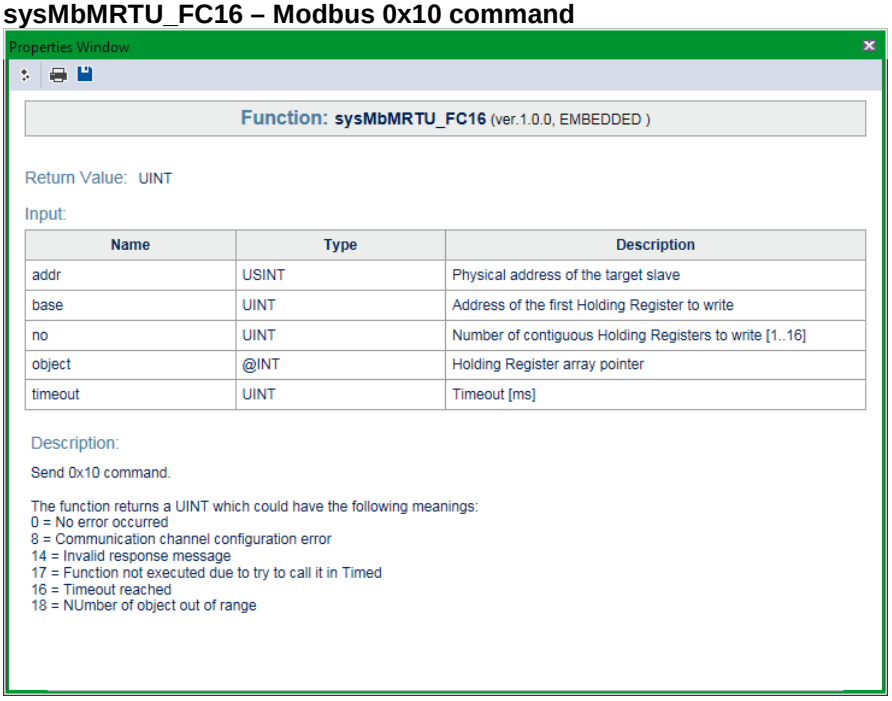
sysMbMRTU_FC06 – Modbus 0x06 command



sysMbMRTU_FC15 – Modbus 0x0F command



sysMbMRTU_FC16 – Modbus 0x10 command



11.2. Available Address Conflict Detection (ACD) to avoid duplicate TCP-IP

If the controller has the RJ45 connector for ethernet communication, the controller itself verifies if its IP address is already used into the local network.

From Power on/reset
If DHCP is enabled, the controller first waits for a new IP address and then it verifies if there is another device that has the same IP.
If DHCP is not enabled or the controller doesn't receive any IP address from the DHCP provider, the controller verifies if its static IP address is already used.

In both cases if the IP address is already used:

- Yellow, Red and Green LEDs flash
- **"IP conflict!"** appears on the Display
- USB-device runs
- IOs are off (but can be driven by USB-device) + RTC run
- USB-host run
- PLC doesn't start
- HMI doesn't start
- HTTP is disabled
- FTP is disabled
- BACnet is disabled
- Modbus slave TCP/IP is disabled
- Modbus slave RTU is disabled on RS485 serials
- Modbus master TCP/IP is disabled (even if programmed)
- Modbus master RTU is disabled (even if programmed)
- CAN networks inoperable
- DataBlock %MX4100.0 is set to TRUE
- Modbus 16bit register 8904 is set to one

Periodically the controller verifies if the conflict is still active.
If the conflict disappears:

- Yellow, Red and Green LEDs turn off
- **“IP conflict!”** is removed from the Display
- Controller starts in nominal mode
- DataBlock %MX4100.0 is set to FALSE
- Modbus 16bit register 8904 is set to zero

Running Controller already connected to a network
Periodically controller verifies if its IP address is used by other devices on the network.
If yes, the following actions are forced:

- Yellow, Red and Green LEDs flash
- **IP conflict!** appears on the Display if there is no HMI running
- HTTP is disabled
- FTP is disabled
- BACnet is disabled
- Modbus slave TCP/IP is disabled
- If BACnet IP active it is disabled
- DataBlock %MX4100.0 is set to TRUE
- Modbus 16bit register 8904 is set to one

Periodically the controller verifies if the conflict is still active.
If not, the following actions are forced:

- Yellow, Red and Green LEDs turn off or under control of the PLC or HMI
- If **IP conflict!** appeared on Display it is removed
- If HTTP was disabled due to IP conflict it is re-activated
- If FTP was disabled due to IP conflict it is re-activated
- All Modbus slave TCP/IP connections are re-opened
- If Modbus master TCP/IP was deactivated due to IP conflict it is re-activated
- If BACnet IP was deactivated due to IP conflict it is re-activated
- DataBlock %MX4100.0 is set to FALSE
- Modbus 16bit register 8904 is set to zero

11.3. Management of Modbus/RTU Master on event via PLC programming for AVP1●●●

The following target functions can be used to send Modbus commands via PLC programming.

These functions require that Modbus/RTU Master (for field) is enabled via SoM HVAC Connection and can be used only in Background Task of the PLC.

When one of these functions is executed the task is suspended until the answer is received from the field.

Parameter Proto_RS485_OB must be configured as Modbus/RTU Master.

sysMbMRTU_FC01 – Modbus 0x01 command

Properties Window

Function Block: sysMbMRTU_FC01 (ver.1.0.0, EMBEDDED)

Input:

Name	Type	Description
addr	USINT	Physical address of the target slave
base	UINT	Address of the first coil to be read
no	UINT	Number of contiguous coils to be read
pollingTime	UINT	Polling time [ms] (Reserved for future use)
waitBeforeSend	UINT	Time to wait before sending the message [ms]
timeout	UINT	Timeout [ms]

Output:

Name	Type	Description
errorCode	MBM_ERRORS	Exception Code
error	BOOL	TRUE = error occurred; FALSE = no error
object1	BOOL	1st Coil
object2	BOOL	2nd Coil
object3	BOOL	3rd Coil
object4	BOOL	4th Coil
object5	BOOL	5th Coil
object6	BOOL	6th Coil
object7	BOOL	7th Coil
object8	BOOL	8th Coil
object9	BOOL	9th Coil
object10	BOOL	10th Coil
object11	BOOL	11th Coil
object12	BOOL	12th Coil
object13	BOOL	13th Coil
object14	BOOL	14th Coil
object15	BOOL	15th Coil
object16	BOOL	16th Coil

Description:

Read Coil Status - Function 01 (0x01).
This function block can be used in the Background task only.
You have to assure that all input variables are properly set before invoking the function block. In particular, the addr input variable must be set to the physical address of an existing Modbus Slave target of this operation.
After function block execution, unless an error occurred, you can use a subset of its output variables (from object1 to objectN, where N equals the value of no input variable) to get the Modbus objects read from the communication bus. If an error did occur, you can read its code from the errorCode output variable.

sysMbMRTU_FC02 – Modbus 0x02 command

Properties Window

Function Block: sysMbMRTU_FC02 (ver.1.0.0, EMBEDDED)

Input:

Name	Type	Description
addr	USINT	Physical address of the target slave
base	UINT	Address of the first Input to be read
no	UINT	Number of contiguous Inputs to be read
pollingTime	UINT	Polling time [ms] (Reserved for future use)
waitBeforeSend	UINT	Time to wait before sending the message [ms]
timeout	UINT	Timeout [ms]

Output:

Name	Type	Description
errorCode	MBM_ERRORS	Exception Code
error	BOOL	TRUE = error occurred; FALSE = no error
object1	BOOL	1st Input
object2	BOOL	2nd Input
object3	BOOL	3rd Input
object4	BOOL	4th Input
object5	BOOL	5th Input
object6	BOOL	6th Input
object7	BOOL	7th Input
object8	BOOL	8th Input
object9	BOOL	9th Input
object10	BOOL	10th Input
object11	BOOL	11th Input
object12	BOOL	12th Input
object13	BOOL	13th Input
object14	BOOL	14th Input
object15	BOOL	15th Input
object16	BOOL	16th Input

Description:

Read Inputs Status - Function 02 (0x02).
This function block can be used in the Background task only.
You have to assure that all input variables are properly set before invoking the function block. In particular, the addr input variable must be set to the physical address of an existing Modbus Slave target of this operation.
After function block execution, unless an error occurred, you can use a subset of its output variables (from object1 to objectN, where N equals the value of no input variable) to get the Modbus objects read from the communication bus. If an error did occur, you can read its code from the errorCode output variable.

sysMbMRTU_FC03 – Modbus 0x03 command

Properties Window

Function Block: sysMbMRTU_FC03 (ver.1.0.0, EMBEDDED)

Input:

Name	Type	Description
addr	USINT	Physical address of the target slave
base	UINT	Address of the first Holding Register to be read
no	UINT	Number of contiguous Holding Registers to be read
pollingTime	UINT	Polling time [ms] (Reserved for future use)
waitBeforeSend	UINT	Time to wait before sending the message [ms]
timeout	UINT	Timeout [ms]

Output:

Name	Type	Description
errorCode	MBM_ERRORS	Exception Code
error	BOOL	TRUE = error occurred; FALSE = no error
object1	UINT	1st Holding Register
object2	UINT	2nd Holding Register
object3	UINT	3rd Holding Register
object4	UINT	4th Holding Register
object5	UINT	5th Holding Register
object6	UINT	6th Holding Register
object7	UINT	7th Holding Register
object8	UINT	8th Holding Register
object9	UINT	9th Holding Register
object10	UINT	10th Holding Register
object11	UINT	11th Holding Register
object12	UINT	12th Holding Register
object13	UINT	13th Holding Register
object14	UINT	14th Holding Register
object15	UINT	15th Holding Register
object16	UINT	16th Holding Register

Description:

Read Holding Registers - Function 03 (0x03).
This function block can be used in the Background task only.
You have to assure that all input variables are properly set before invoking the function block. In particular, the addr input variable must be set to the physical address of an existing Modbus Slave target of this operation.
After function block execution, unless an error occurred, you can use a subset of its output variables (from object1 to objectN, where N equals the value of no input variable) to get the Modbus objects read from the communication bus. If an error did occur, you can read its code from the errorCode output variable.

sysMbMRTU_FC04 – Modbus 0x04 command

Properties Window

Function Block: sysMbMRTU_FC04 (ver.1.0.0, EMBEDDED)

Input:

Name	Type	Description
addr	USINT	Physical address of the target slave
base	UINT	Address of the first Input Register to be read
no	UINT	Number of contiguous Input Registers to be read
pollingTime	UINT	Polling time [ms] (Reserved for future use)
waitBeforeSend	UINT	Time to wait before sending the message [ms]
timeout	UINT	Timeout [ms]

Output:

Name	Type	Description
errorCode	MBM_ERRORS	Exception Code
error	BOOL	TRUE = error occurred; FALSE = no error
object1	UINT	1st Input Register
object2	UINT	2nd Input Register
object3	UINT	3rd Input Register
object4	UINT	4th Input Register
object5	UINT	5th Input Register
object6	UINT	6th Input Register
object7	UINT	7th Input Register
object8	UINT	8th Input Register

Description:

Read Input Registers - Function 04 (0x04).
This function block can be used in the Background task only.
You have to assure that all input variables are properly set before invoking the function block. In particular, the addr input variable must be set to the physical address of an existing Modbus Slave target of this operation.
After function block execution, unless an error occurred, you can use a subset of its output variables (from object1 to objectN, where N equals the value of no input variable) to get the Modbus objects read from the communication bus. If an error did occur, you can read its code from the errorCode output variable.

sysMbMRTU_FC05 – Modbus 0x05 command

Properties Window

Function Block: sysMbMRTU_FC05 (ver.1.0.0, EMBEDDED)

Input:

Name	Type	Description
addr	USINT	Physical address of the target slave
base	UINT	Address of the Coil to write
object	BOOL	Coil value
pollingTime	UINT	Polling time [ms] (Reserved for future use)
waitBeforeSend	UINT	Time to wait before sending the message [ms]
timeout	UINT	Timeout [ms]

Output:

Name	Type	Description
errorCode	MBM_ERRORS	Exception Code
error	BOOL	TRUE = error occurred; FALSE = no error

Description:

Write Single Coil - Function 05 (0x05).
This function block can be used in the Background task only.
You have to assure that all input variables are properly set before invoking the function block. In particular, the addr input variable must be set to the physical address of an existing Modbus Slave target of this operation. The value of the object input variable is the value sent on the communication bus to the slave node.
After function block execution, if an error did occur, you can read its code from the errorCode output variable.

sysMbMRTU_FC06 – Modbus 0x06 command

Properties Window

Function Block: sysMbMRTU_FC06 (ver.1.0.0, EMBEDDED)

Input:

Name	Type	Description
addr	USINT	Physical address of the target slave
base	UINT	Address of the Register to write
object	UINT	Register value
pollingTime	UINT	Polling time [ms] (Reserved for future use)
waitBeforeSend	UINT	Time to wait before sending the message [ms]
timeout	UINT	Timeout [ms]

Output:

Name	Type	Description
errorCode	MBM_ERRORS	Exception Code
error	BOOL	TRUE = error occurred; FALSE = no error

Description:

Write Single Register - Function 06 (0x06).
This function block can be used in the Background task only.
You have to assure that all input variables are properly set before invoking the function block. In particular, the addr input variable must be set to the physical address of an existing Modbus Slave target of this operation. The value of the object input variable is the value sent on the communication bus to the slave node.
After function block execution, if an error did occur, you can read its code from the errorCode output variable.

sysMbMRTU_FC15 – Modbus 0x0F command

Properties Window

Function Block: sysMbMRTU_FC15 (ver.1.0.0, EMBEDDED)

Input:

Name	Type	Description
addr	USINT	Physical address of the target slave
base	UINT	Address of the first Coils to write
no	UINT	Number of contiguous Coils to write
object1	BOOL	1st Coil value
object2	BOOL	2nd Coil value
object3	BOOL	3rd Coil value
object4	BOOL	4th Coil value
object5	BOOL	5th Coil value
object6	BOOL	6th Coil value
object7	BOOL	7th Coil value
object8	BOOL	8th Coil value
object9	BOOL	9th Coil value
object10	BOOL	10th Coil value
object11	BOOL	11th Coil value
object12	BOOL	12th Coil value
object13	BOOL	13th Coil value
object14	BOOL	14th Coil value
object15	BOOL	15th Coil value
object16	BOOL	16th Coil value
pollingTime	UINT	Polling time [ms] (Reserved for future use)
waitBeforeSend	UINT	Time to wait before sending the message [ms]
timeout	UINT	Timeout [ms]

Output:

Name	Type	Description
errorCode	MBM_ERRORS	Exception Code
error	BOOL	TRUE = error occurred; FALSE = no error

Description:

Write Multiple Coils - Function 15 (0x0F).
This function block can be used in the Bakcground task only.
You have to assure that all input variables, except any objectN variable, where N is greater than the value of no input variable, are properly set before invoking the function block. In particular, the addr input variable must be set to the physical address of an existing Modbus Slave target of this operation. The values of all input variables ranging from object1 to objectN, where N equals the value of no input variable are the values sent on the communication bus to the slave node.
After function block execution, if an error did occur, you can read its code from the errorCode output variable.

sysMbMRTU_FC16 – Modbus 0x10 command

Properties Window

Function Block: sysMbMRTU_FC16 (ver.1.0.0, EMBEDDED)

Input:

Name	Type	Description
addr	USINT	Physical address of the target slave
base	UINT	Address of the first Holding Register to write
no	UINT	Number of contiguous Holding Registers to write
object1	UINT	1st Holding Register value
object2	UINT	2nd Holding Register value
object3	UINT	3rd Holding Register value
object4	UINT	4th Holding Register value
object5	UINT	5th Holding Register value
object6	UINT	6th Holding Register value
object7	UINT	7th Holding Register value
object8	UINT	8th Holding Register value
object9	UINT	9th Holding Register value
object10	UINT	10th Holding Register value
object11	UINT	11th Holding Register value
object12	UINT	12th Holding Register value
object13	UINT	13th Holding Register value
object14	UINT	14th Holding Register value
object15	UINT	15th Holding Register value
object16	UINT	16th Holding Register value
pollingTime	UINT	Polling time [ms] (Reserved for future use)
waitBeforeSend	UINT	Time to wait before sending the message [ms]
timeout	UINT	Timeout [ms]

Output:

Name	Type	Description
errorCode	MBM_ERRORS	Exception Code
error	BOOL	TRUE = error occurred; FALSE = no error

Description:

Write Multiple Registers - Function 16 (0x10).
This function block can be used in the Background task only.
You have to assure that all input variables, except any objectN variable, where N is greater than the value of no input variable, are properly set before invoking the function block. In particular, the addr input variable must be set to the physical address of an existing Modbus Slave target of this operation. The values of all input variables ranging from object1 to objectN, where N equals the value of no input variable are the values sent on the communication bus to the slave node.
After function block execution, if an error did occur, you can read its code from the errorCode output variable.

12. RELEASE HISTORY

12.1. FREE Studio v3.8.0

- Availability of new targets:
  - Expansion module EVE6000 EVE10200
  - Display Touch AVP1●●●, in portrait and landscape mode.
- New features for AV●126●●/AV●84●● target:
  - Management of Modbus/TCP Master (see below, 11.6);
  - Available Modbus Master command 0x05 (Write Single Coil) (see 11.7)
  - Available new Target Blocks for HTTP, FTP, SNTP and Modbus/TCP Slave
  - Available information on Max time of tasks (see 11.9);
- Management of Chinese text in web pages;
- Automatic declaration of variables;
- Solved issue: Password parameter reading error in simulation mode for FREE Smart;
- Solved issue: A Branch of a net in FBD cause an incorrect calculation
- Solved following issues on Application Function Blocks:
  - ThreePointActuator, override 110% of *uiMaxRunTime* at reset, calibration command given with transition to TRUE of xCalibration.
  - EnergyTrend added back to the library, solved issue on EEPROM writing.
  - Minor fixes: contact your Eliwell representative for information.

12.2. FREE Studio v3.7.0

- 4 new Application Function Blocks (more information in FREE Studio Application Function HVAC Library Guide, EIO0000002057.02)
  - EC Fan Management (*EcFanMgmt*)
  - Calendars (*YearlyEvent* and *WeekSchedule*)
  - Compressor Management with VSD (*CompMgmtVS*)
  - QR Code Generator (*QRcodeGenerator*)
- Improvement of design of GUI.
- New tree library view.
- Complex constant variable initialization.
- Expression in FBD language.
- Custom image for POUs.
- New POU property view.
- SFC editing improvements.
- SFC transitions with expressions.
- Added *Exp* mathematical function in AV●126●●/AV●84●● simulator.
- Improved the export resources to XSLT functionality.
- FREE Studio Connection: limit the number of mapped variables in CAN.
- Auto-creation of variable in LD editor.
- Reduction of the opening time of large projects with many global variables and POUs.
- Application name and version used for source code id calculation.
- New colors for debugging of Boolean variables.
- Copy of structures allowed even when the source structure is accessed by variable index.
- Minor fixes.

12.3. AV●126●●/AV●84●●

12.3.1. Firmware msk 596.05

- Features related to HTTP server:
  a) Service enable/disable: with parameter, or with PLC function;
  b) White list of up to 3 IP-addresses for clients, settable by PLC function;

c) Introduction of 3 additional users besides administrator settable with PLC function, with lower permissions, possibility to change the starting web page displayed by each, and to filter up to 10 files to their access.

- Features related to Modbus Slave TCP/IP
  a) Service enable/disable by parameter, effective at power-on (reset needed);
  b) White list of up to 3 IP-addresses for clients, settable with PLC function (different from HTTP list)
- Features related to BACnet IP
  a) Service enable/disable by parameter, effective at power-on (reset needed)
- Features related to FTP Server
  a) PLC function for administrator and password setting
  b) Service enable/disable, with parameter, or with PLC function;
  c) White list of up to 3 IP-addresses for clients, settable with PLC function (different from lists above)
  d) Introduction of 3 additional users besides administrator settable with PLC function, with lower permissions, possibility to filter up to 10 files, in read or write, to their access;
- FTP Client: connection to a remote server set in PASSIVE MODE via PLC function.
- Introduction of Modbus/TCP Master, with possibility to change Unit ID of Slaves.
- Modbus Master command 0x05;
- Introduction of SNTP unicast;
- Improvements:
  - in BACnet communication (/IP and MS/TP), resulting in BTL compliance for MSK596.05 and greater versions;
  - Available information on Max time of tasks
- Solved the following issues on BACnet MS/TP stack:
  - Token pass delay out of specification in case of projects with high number of BACnet objects.
  - Network message timeout not managed at boot.
  - Minor fixes: contact your Eliwell representative for information.
- Solved an issue occurring on a particular configuration of device: FREE Advance Logic Controller with a Communication Module connected and an active Vdc digital input at power on, causing a blocking error until the next power cycle. Such issue was revealed in version 596.604 only.

12.3.2. Firmware msk 596.604

- Solved an issue occurring on particular configurations of Modbus/TCP networks, revealed in two typical scenarios:
  a) PLC Modbus/TCP client which uses several sockets to communicate with a FREE Advance Logic Controller may cause a blocking communication error until the next power cycle;
  b) in a binding network of FREE AdvanceLogic Controllers on Modbus/TCP, a temporary block of communication on some devices may happen.
- Improvements in BACnet communication (/IP and MS/TP).

12.3.3. Firmware msk 596.04

- Improved communication both on RS485 and TCP/IP.
- Change of Modbus Master RTU settings on the fly.
- Improved font management in embedded HMI.
- Copy of generic files (maximum 256) from a USB stick, both to internal memory and to microSD card.
- Visibility of IP address in DHCP mode.
- Turbo mode linked to EVK1000.

- Several improvements in BACnet communication (/IP and MS/TP), deriving from BTL pre-compliance tests.

12.4. EVP3●●● - Firmware msk 489.18

- Improvements in BACnet communication (/IP and MS/TP) for alignment with MSK596.05 and MSK423.25 for BTL compliance;
- Alignment of communication features with MSK596.604 and MSK596.804 (see above at 4.2 and 4.3)
- Solved the following issues on BACnet MS/TP stack:
    - Token pass delay out of specification in case of projects with high number of BACnet objects.
    - Network message timeout not managed at boot.
    - Minor fixes: contact your Eliwell representative for information.

12.5. EVP3●●● - Firmware msk 489.17

- Improved communication both on RS485 and TCP/IP.
- Introduction of Turbo mode:
    - Faster upload of HMI from CPU, when used with AVD/AVC.
    - Possibility to upload FW from CPU, when used with AVD/AVC .
    - To be functional this function must be present in display and CPU.
- Several improvements in BACnet communication, deriving from BTL pre-compliance tests.
- Fixed in Web Server the truncation of PLC strings longer than 24 characters.

12.6. EVD75●● - Firmware msk 423.25

- Improvements in BACnet communication (/IP and MS/TP), resulting in BTL compliance for MSK423.25 and greater versions;
- Alignment of communication features with MSK596.604 and MSK596.804 (see above at 4.2 and 4.3).
- Solved the following issues on BACnet MS/TP stack:
    - Token pass delay out of specification in case of projects with high number of BACnet objects.
    - Network message timeout not managed at boot.
    - Minor fixes: contact your Eliwell representative for information.

12.7. EVD75●● - Firmware msk 423.24

- Improved communication both on RS485 and TCP/IP
- Several improvements in BACnet communication, deriving from BTL pre-compliance tests.
- Fixed in Web Server the truncation of PLC strings longer than 24 characters.

12.8. EVC75●● - Firmware msk 477.25

- Improvements in BACnet communication (/IP and MS/TP) for alignment with MSK596.05 and MSK423.25 for BTL compliance;
- Alignment of communication features with MSK596.604 and MSK596.804 (see above at 4.2 and 4.3).

- Solved the following issues on BACnet MS/TP stack:
  - Token pass delay out of specification in case of projects with high number of BACnet objects.
  - Network message timeout not managed at boot.
  - Minor fixes: contact your Eliwell representative for information.

12.9. EVC75●● - Firmware msk 477.24

- Improved communication both on RS485 and TCP/IP
- Several improvements in BACnet communication, deriving from BTL pre-compliance tests.
- Fixed in Web Server the truncation of PLC strings longer than 24 characters.
- Minor fixes: contact your Eliwell representative for information.

12.10. Smart Modbus Master - Firmware msk 542.08

- Minor fixes: contact your Eliwell representative for information.

12.11. Smart Modbus Master - Firmware msk 542.07

- Fixed an issue causing occasional block of Modbus RTU communication.