IOMOD 8AI

IOMod 8AI – industrial 8 analog inputs module.

- Firmware version 1
  - IOMOD 8AI User Manual Modbus
  - IOMOD 8AI User Manual IEC 60870-5-103
  - IOMOD 8AI User Manual IEC 60870-5-101
- Firmware version 2
  - IOMOD 8AI User Manual

Firmware version 1

IOMOD 8AI User Manual Modbus

Introduction

IOMod 8AI is a compact-sized stand-alone Modbus (RTU) or IEC-60870-5-103 analog input controller. IOMod can be used for industrial applications where digital signaling is used and robust communication is needed. IOMod is an ideal solution for applications such as data acquisition, observation, process monitoring, testing and measurement at remote places. It is controlled over Modbus or IEC 60870-5-103 protocol, and can be used with any SCADA system.

Features

● 8 analog inputs, each configurable separately
● All inputs are capable of measuring electrical voltage or current
● Inputs can be recalibrated according to the needs of a user
● RS485 communication
● LED input indications, + Data transmission (Rx and Tx) indication.
● Configurable over USB
● Drag-and-Drop firmware upgrade over USB
● A small sized case with removable front panel
● DIN rail mount
● Operating temperature: from -30 to +70°C
● Power Requirements: 12-24 VDC

Operational Information

IOMod 8AI uses Modbus (RTU) or IEC 60870-5-103 protocol over RS485 interface. A protocol used by a device can be changed by uploading corresponding firmware. Default communication settings: 9600 baud/s baud rate, 8N1 port configuration, Slave address - 1.

Common configuration information

1. Measurement type. A user can select measurement type (electrical current or voltage) on each channel individually.
2. Sensitivity selection. To increase a resolution of input measurement capture, a user can define in which range measurement will occur. The best resolution will be achieved within a shortest selected range. For all possible configurations refer to Technical Information and Configuration over USB described later in this document.
3. Measurement limits can be selected, which in turn set thresholds on underflow or overflow error statuses. Also, if a scaled integer data type is selected, these limits will be converted to values using limits stated in Casting data range for Modbus protocol.
4. Data type selection. A device can output float or scaled integer data types for each input individually for Modbus. Modbus input register read will then show raw float value (in milliamperes or millivolts) or scaled integer types. When a float data type is selected, each input will be represented as two Modbus registers (32-bits). These values can be later
   converted to IEEE-754 standard-compatible float values.
5. Casting data range. A device is capable of converting measurements into desired decimal values in a linear manner. For Modbus, a user can select between floating point data type – which will return a raw measurement of either voltage or current and scaled integer type – which conversion is freely definable. Note that the specification for IEC-60870-5-103 only lets 13-bit signed measurands, therefore only cast representation of true value can be used with absolute limits being [-4095, 4095].
   Fig. 3.2 shows how raw measurement (current in this case) is converted to an integer type. Conversion is done by defining measurement limits (from -25mA to 25mA) and casting data range (from 0 to 100). Measurement limits define thresholds for overflow and underflow errors respectively; casting data range defines limits (from 0 to 100) that value should be between after recalculation. Such a recalculation can be used in applications that require measurements in relative units such as percentage.
   A device can round measurements to lower conversion limit (in this case – to zero) if a measurement is near it: ±15% of measurement value (Fig. 3.2). Live zero correction is a useful option for eliminating noise. As signals always have some noise level, such an option can compensate it to get a true zero value.

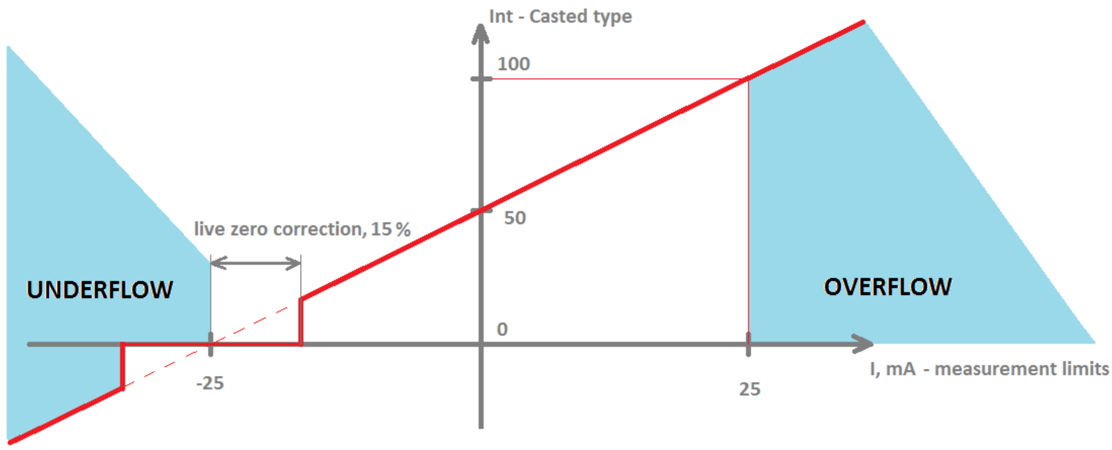

Fig. 3.2. Conversion graph from raw measurement to cast integer

MODBUS operational information

Each input can be configured to represent 16-bit signed integer value or 32-bit float value. When the float data type is selected, the value will be shown as two registers for one input. When the 16-bit integer data type is selected, a value will be shown as one register. This means addresses of individual input and the maximum number of readable registers can differ according to user configuration.
Configure device over USB terminal. Modbus commands that can be used are shown in the table below.

Supported MODBUS functions

02 (0x02) Read Input Status
Used to read analog input overflow and underflow statuses. First 8 inputs show each input underflow statuses (according to measurement limits option) and second 8 inputs show overflow statuses.

04 (0x04) Read Input Registers
Used to read measurements of at most 8 analog inputs. IOMod 8AI has 8 analog inputs from address 0 to address 15 (0000Fh). Different analog inputs can be cast in different data types configured over a USB interface.

02 (0x02) Read Input Status
Used to read analog input overflow and underflow statuses. First 8 inputs show each input underflow statuses (according to measurement limits option) and second 8 inputs show overflow statuses. 04 (0x04) Read Input Registers Used to read measurements of at most 8 analog inputs. IOMod 8AI has 8 analog inputs from address 0 to address 15 (0000Fh). Different analog inputs can be cast in different data types configured over a USB interface.

Modbus register table

Modbus register table

Function	**Register**	**Name**	**Description**	**Values range**
02	00000-0000F	Read Input Status	Returns overflow and underflow statuses of each input. First 8 shows underflow status, last 8 shows overflow	0 – OK, 1 - error
04	00000-0000F	Read Input Registers	Returns Input values	0-65535 (decimal) of one Register

Technical information

System	
Dimensions	101 x 119 x 17.5 mm
Case	ABS, black
Working environment	Indoor

Working temperature	-30°C +70°C
Recommended operating conditions	5 – 60°C and 20 – 80%RH;
Configuration	USB
Firmware upgrade	USB – mass storage device
Electrical specifications	
Inputs	16-bit resolution; Channel-Independent Programmable Input Ranges: ● Voltage input: ○ Bipolar: ±10.24 V, ±5.12 ±2.56 V ○ Unipolar: 10.24 V, 5.12 ● Current input: ○ Bipolar: ±45.5 mA, ±22.75. and ±11.38 mA ○ Unipolar: 45.5 mA, 22.75 Overvoltage protection up to ±20V;
Power	
Power Supply	9V to 33V
Current consumption	40mA @ 12VDC, 20mA @ 24VDC

Mounting and installation guide

IOMod 8AI RS485 interface

IOMod 8AI has integrated 120Ω termination resistor which can be enabled or disabled over USB configuration. It is recommended to use termination at each end of the RS485 cable. See typical connection diagram in Fig. 5.1.

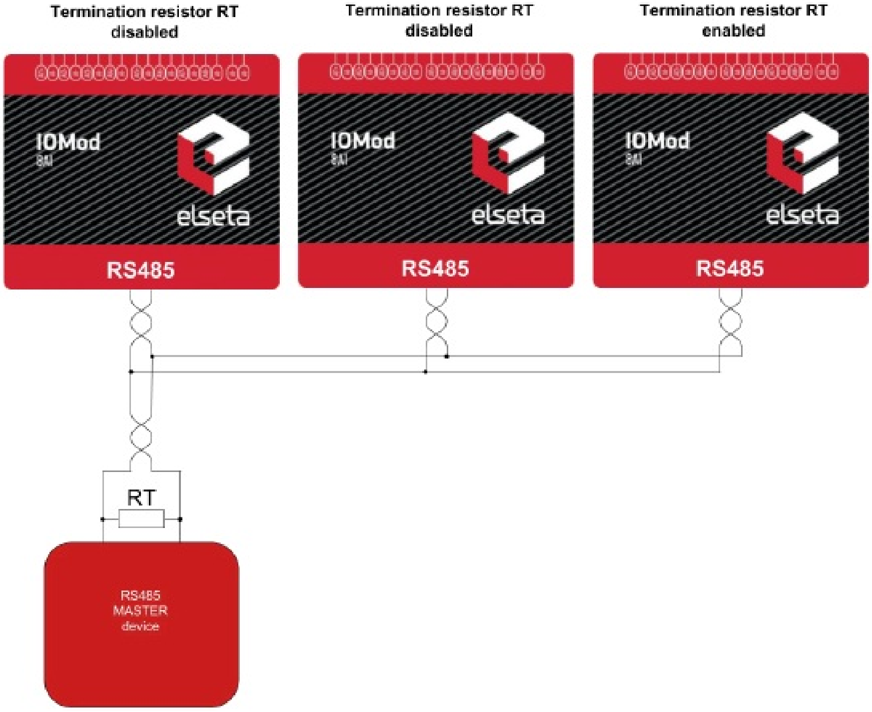

Fig. 5.1. Typical IOMod connection diagram

IOMod 8AI has 1/8 Unit load receiver which allows having up to 255 units on a single line (compared to standard 32 units). To reduce reflections keep the stubs (cable distance from main RS485 bus line) as short as possible.

IOMod 8AI inputs

A typical application of IOMod 8AI unipolar and bipolar voltage inputs is shown in Fig. 5.2. See Configuration over USB chapter for instructions for analog input configuration.

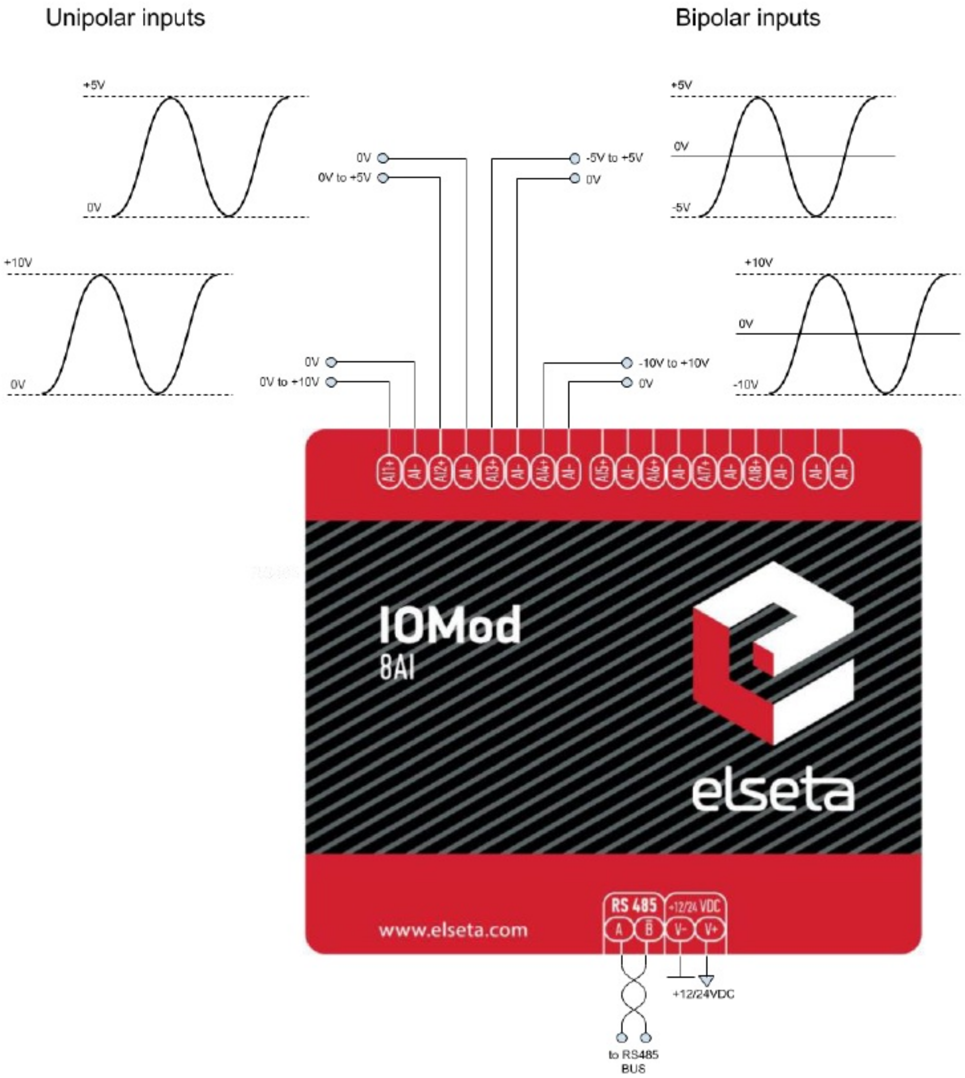

Fig. 5.2. Voltage input connection diagram for IOMod device

All analog inputs can be configured as current inputs to connect 0-20 mA, 4-20 mA or other current output sensors. Typical application of IOMod 8AI unipolar and bipolar current inputs is shown in

Fig. 5.3. See Configuration over USB chapter for instructions for analog input configuration.

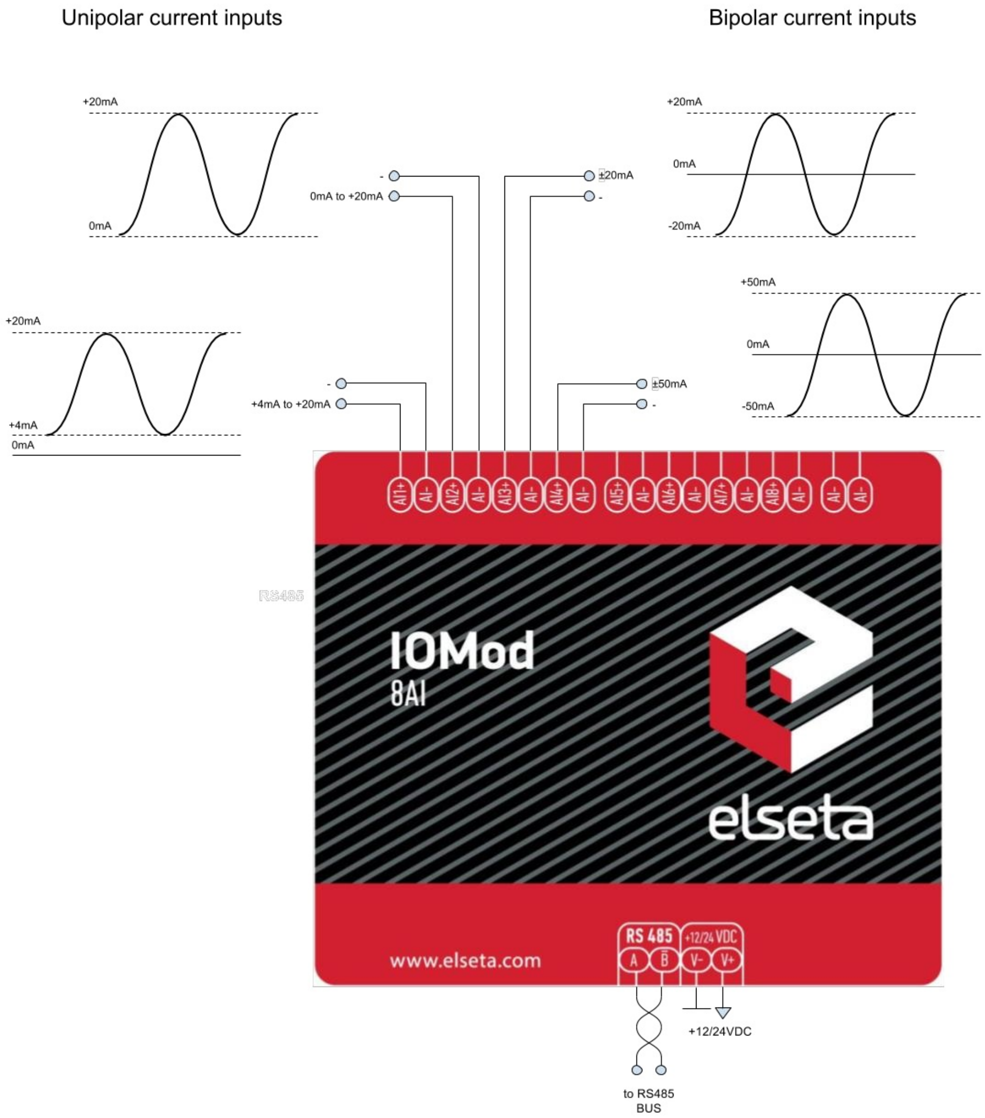

Fig. 5.3. Current input connection diagram for IOMod device

Configuration over USB

Driver installation

The device requires USB drivers to work as a Virtual COM port. The first-time connection between device and computer could result in “Device driver software was not successfully installed” error (as in Fig. 6.1).

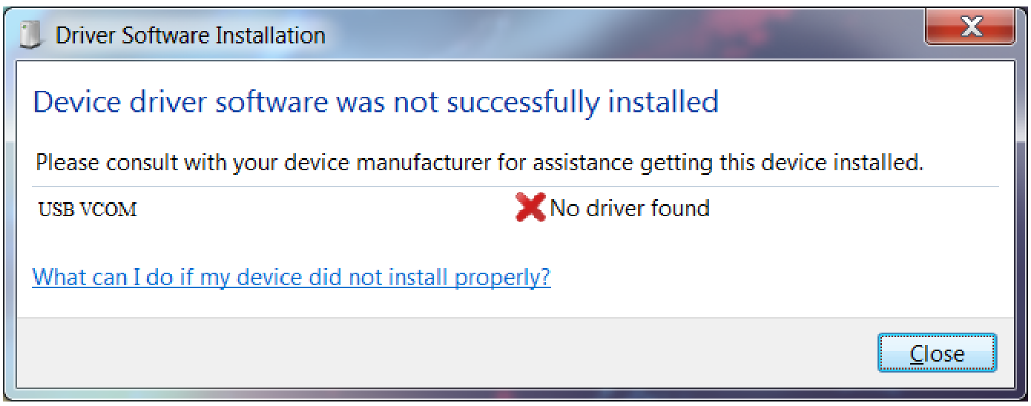

Fig. 6.1. Unsuccessful device software installation error

A user then should manually install drivers by selecting a downloaded driver folder:

- Go to Control Panel -> Device Manager;
- Select a failing device;

- Press “Update driver software”; screen as in Fig. 6.2. should appear:

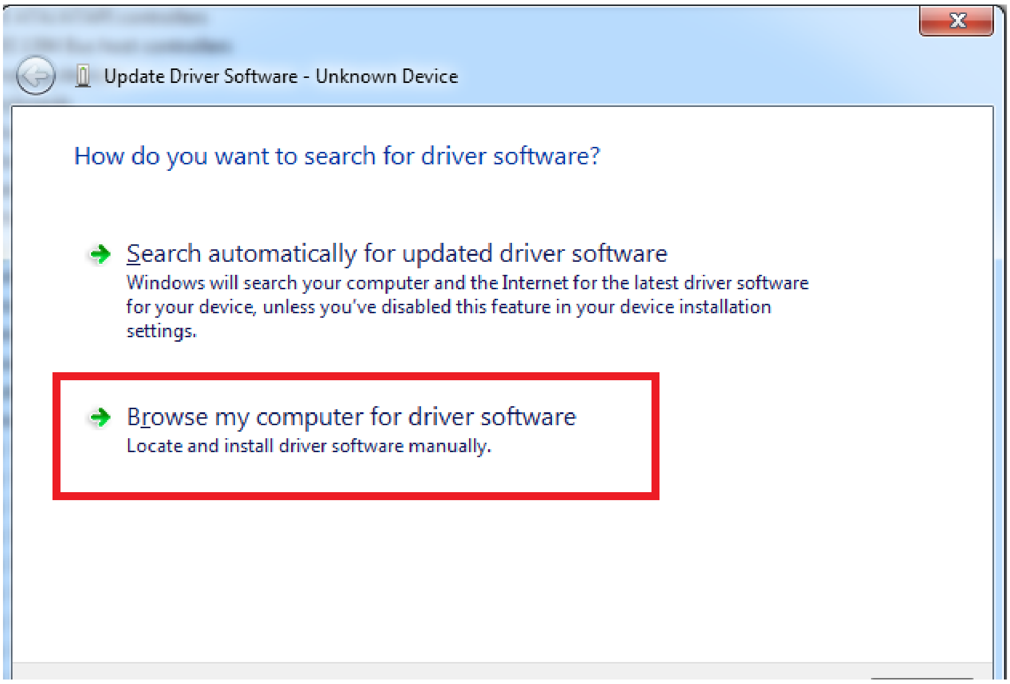

Fig. 6.2. Device driver software update message

- Select “x86” driver for a 32-bit machine or x64 for a 64-bit machine. If not sure, select a root folder (folder in which x64 and x86 lay inside, as in Fig. 6.3).

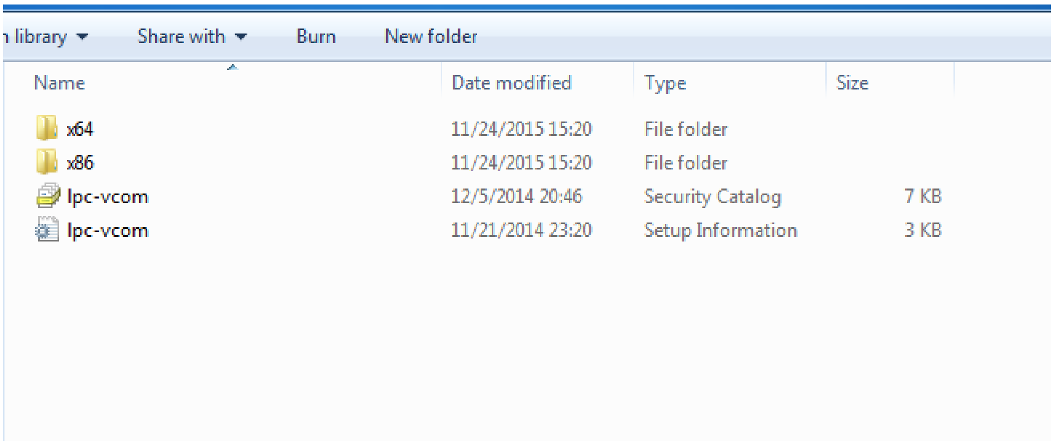

Fig. 6.3. Device driver folder content

IOMod configuration via PuTTY terminal

A configuration of IOMod device is done through CLI (Command Line Interface) on the virtual COM port. Drivers needed for Microsoft Windows to install VCOM will be provided. To open up CLI simply connect to specific V-COM port with terminal software (advised to use PuTTY terminal software. If other software is being used, a user might need to send <return> symbol after each command). When connected user should immediately see the main screen.

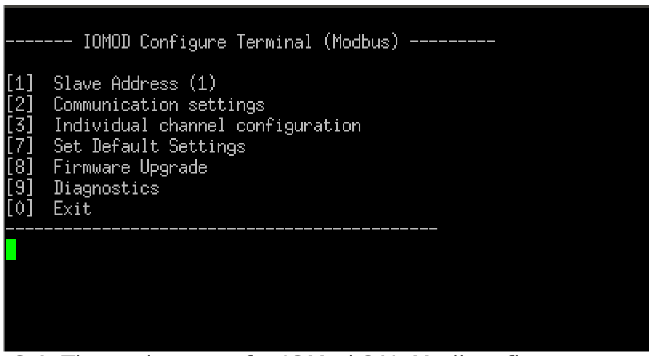

Fig. 6.4. The main menu for IOMod 8AI, Modbus firmware version

Navigation is performed by sending digits or other symbols to a terminal. A user then proceeds by following on-screen instructions. For example, to set baud rate, press [2] to enter Communication settings screen; press [1] to edit; enter new configuration; press [RETURN] to save, or [ESC] to cancel changes. When done, press [0] (exit) before disconnecting device. Default values are set by pressing [7] on the main screen, and confirming changes [1].

If the terminal window is closed accidentally, a user can connect the terminal program again, and press any key on a keyboard to show the main menu again.

Modbus Main menu

	Menu Name	Function	Values	Default Values
1.	Slave Address	Modbus Slave address / ID	1-247	1
2.	Communication settings	[1] Baud rate, [2] Data, Stop and Parity Bits, [3] RS485 Terminating resistor	[1] 100 - 256000, [2] 8 Data bits + 1/2 Stop bits, Even/None/Odd Parity [3] Enabled/Disabled	[1] 9600, [2] 8N1, [3] Enabled
3.	Individual channel configuration	Channel configuration screen (see Table "Individual Channel Configuration")		
7.	Set Default Settings	Sets Default Settings	(1 to confirm, 0 to cancel)	-
8.	Firmware Upgrade	Mass Storage Device Firmware Upgrade	(1 to confirm, 0 to cancel)	-
9.	Diagnostics	Input / Output states	-	-
0.	Exit	Exit and disconnect	-	-

Individual channel configuration

	MenuName	Function	Values	Default Values
[1]	Measure Current/Voltage	Selecting this option toggles measurement selection	-	-
[2]	Change sensitivity	Selecting a measurement range	[1] [-10.24 V, 10.24 V], [2] [-5.12 V, 5.12 V], [3] [-2.56 V, 2.56 V], [4] [0 V, 10.24 V], [5] [0 V, 5.12 V]	[1] [-10.24 V, 10.24 V]
[3]	Measurement limits	Entering floating point values to use them for marking overflow/underflow statuses	Floating values in a range defined by measuring sensitivity	-5 mV; 5mV
[4]	Cast to a scaled integer/float	Selecting this option toggles value casting selection		
[5]	Casting data range	Check **Common configuration information**		
[6]	Live zero tolerance			

0	Back	Back to last menu screen		

Measurements' calibration

Every device is shipped containing factory-predefined calibration coefficients. However, changing temperature and humidity conditions can affect the accuracy of measurements. To get the best accuracy, use the device at room temperature and if that is not possible, calibrate coefficients accordingly.
To enter the calibration screen (Fig. 6.6) from main menu press '@' symbol.

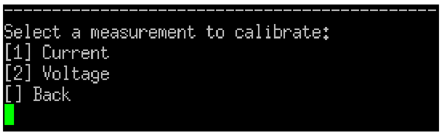

Fig. 6.6. Analog inputs' calibration screen

In the calibration screen, either current or voltage coefficients can be selected. Selecting current bypressing [1] should enter additional screen (Fig. 6.7). Calibrating voltage is an identical process therefore only current setting is explained.

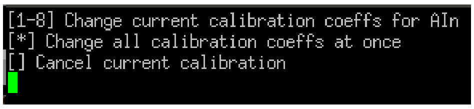

Fig. 6.7. Current calibration screen

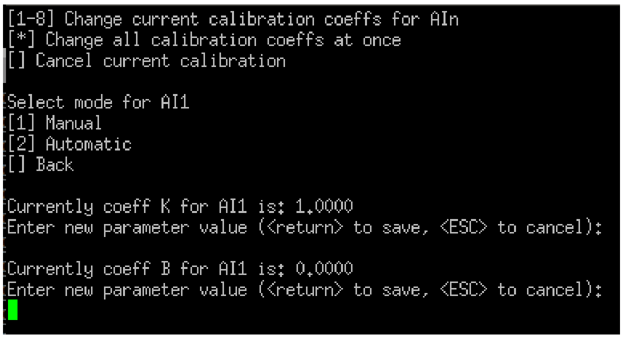

Fig. 6.8. Manual calibration coefficients' setting

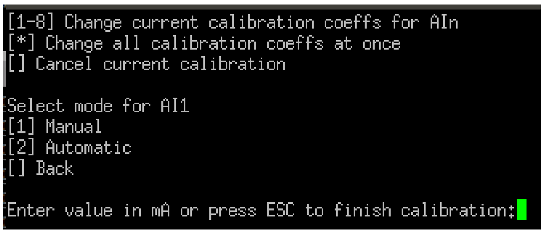

Fig.6.9. Automatic calibration coefficients' setting

All configuration coefficients can either be configured all at once or individually. The coefficient setting for all channels is done manually only. Tweaking individual channels, however, can be automatic (Fig. 6.9) as well as manual (Fig. 6.8). Selecting automatic calibration polls values that user inputs and tweaks coefficients to minimize the additive error. It is therefore advised to scan several values differing by the same value interval to get the best results. Pressing ESC finishes calibration and saves coefficients to non volatile memory.

Protocol simulation over USB

After entering the diagnostics screen, protocol simulator can be turned on by pressing [9]. When it is on, the device will communicate over USB port rather than a RS-485 line. Communication on a RS-485 line is closed and all Modbus or IEC-60870-5-103 commands will only be accepted from USB. To exit this mode device must be restarted.

Firmware upgrade over USB

To update device firmware user must enter main configuration menu.
Enter Firmware update screen by pressing [8];
Confirm update by pressing [1];
Device now enters Firmware Upgrade mode. Device reconnects as mass storage device (Fig 6.10.).

It is recommended to close terminal window after entering firmware upgrade mode.

Fig. 6.10. Mass storage device warning

User then must delete existing file “firmware.bin”, and simply upload new firmware file by drag and drop. (Fig 6.11.)

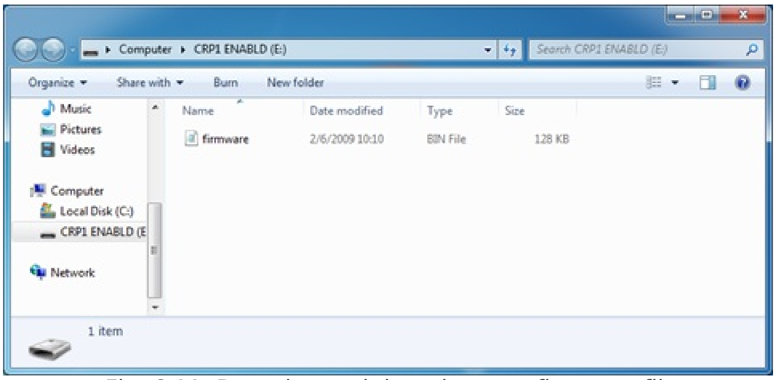

Fig. 6.11. Dragging and dropping new firmware file

Reconnect device and check firmware version. It should now represent the one it was updated to.

Testing With “THE VINCI” software

To test IOMod with default settings, a device should be connected through RS485 to Modbus master. Example using “The Vinci Expert” device as serial interface converter and adapter to PC with “The Vinci” software. Default settings for Modbus – 9600 baud/s baud rate; 8 data, no parity, 1 stop bit. When opening “The Vinci” software, choose Modbus serial – Master mode. In the Settings tab, choose station number (default – 1); configure tags (as described in section Modbus operational information); Press Start and go to “Statistic” tab as shown in Fig. 7.1.

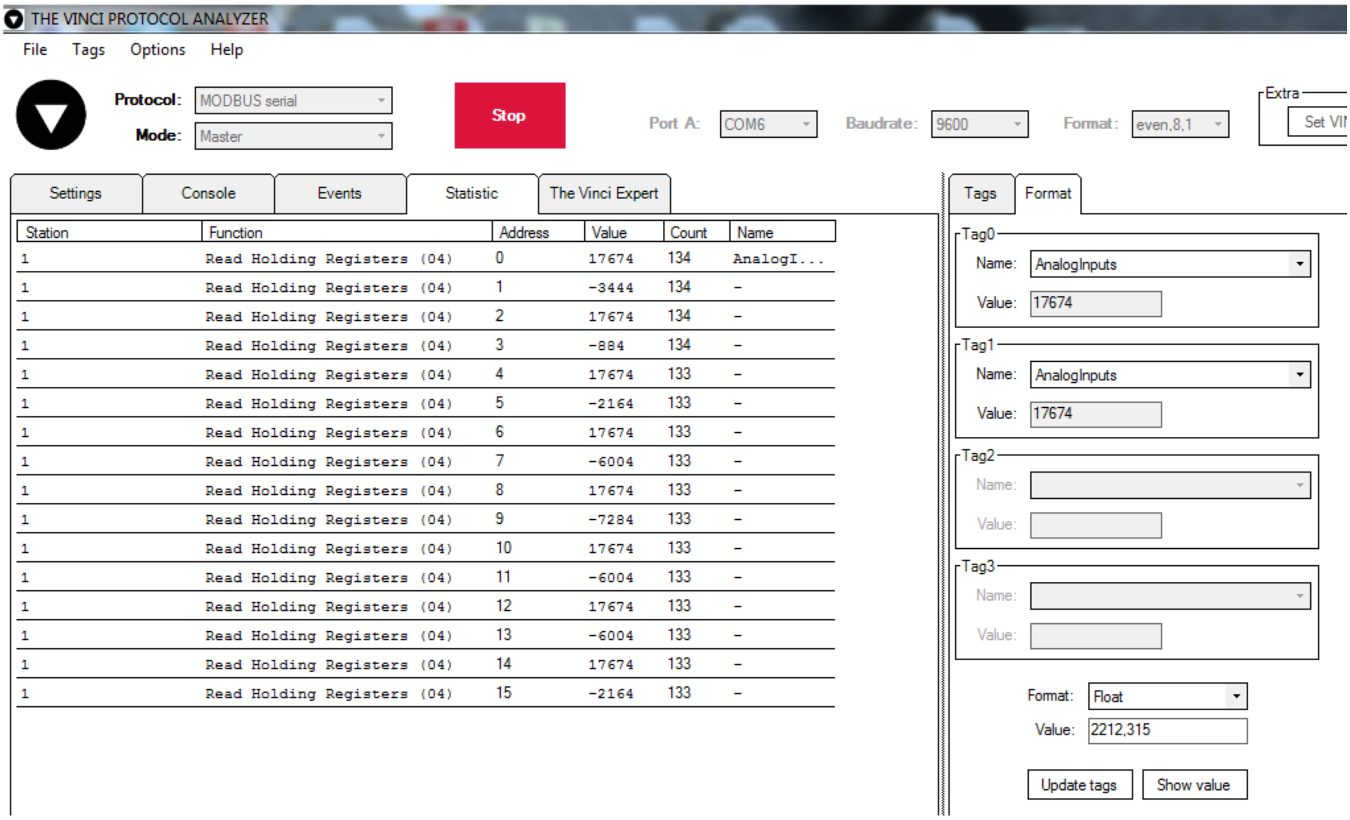

Fig. 7.1. Testing IOMod Modbus communication using “THE VINCI” software

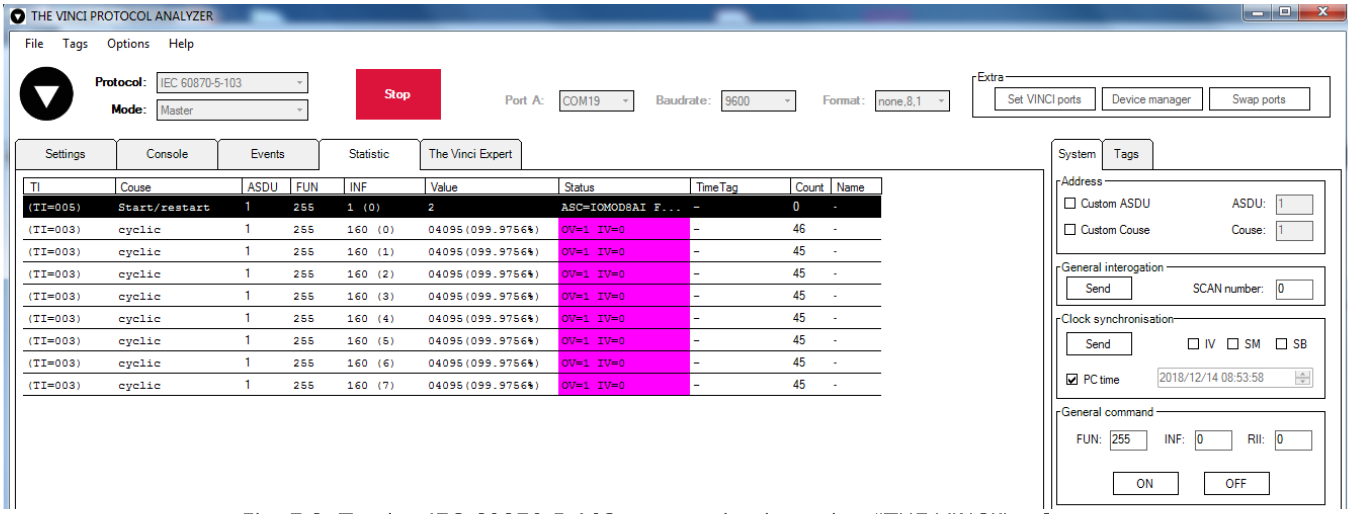

Fig. 7.2. Testing IEC-60870-5-103 communication using “THE VINCI” software

Firmware version 1

IOMOD 8AI User Manual IEC 60870-5-103

Introduction

IOMod 8AI is a compact-sized stand-alone Modbus (RTU) or IEC-60870-5-103 analog input controller. IOMod can be used for industrial applications where digital signaling is used and robust communication is needed. IOMod is an ideal solution for applications such as data acquisition, observation, process monitoring, testing and measurement at remote places. It is controlled over Modbus or IEC 60870-5-103 protocol, and can be used with any SCADA system.

Features

● 8 analog inputs, each configurable separately
● All inputs are capable of measuring electrical voltage or current
● Inputs can be recalibrated according to the needs of a user
● RS485 communication
● LED input indications, + Data transmission (Rx and Tx) indication.
● Configurable over USB
● Drag-and-Drop firmware upgrade over USB
● A small sized case with removable front panel
● DIN rail mount
● Operating temperature: from -30 to +70°C
● Power Requirements: 12-24 VDC

Operational Information

IOMod 8AI uses Modbus (RTU) or IEC 60870-5-103 protocol over RS485 interface. A protocol used by a device can be changed by uploading corresponding firmware. Default communication settings: 9600 baud/s baud rate, 8N1 port configuration, Slave address - 1.

Common configuration information

1. Measurement type. A user can select measurement type (electrical current or voltage) on each channel individually.
2. Sensitivity selection. To increase a resolution of input measurement capture, a user can define in which range measurement will occur. The best resolution will be achieved within a shortest selected range. For all possible configurations refer to Technical Information and Configuration over USB described later in this document.
3. Measurement limits can be selected, which in turn set thresholds on underflow or overflow error statuses. Also, if a scaled integer data type is selected, these limits will be converted to values using limits stated in Casting data range for Modbus protocol.
4. Data type selection. A device can output float or scaled integer data types for each input individually for Modbus. Modbus input register read will then show raw float value (in milliamperes or millivolts) or scaled integer types. When a float data type is selected, each input will be represented as two Modbus registers (32-bits). These values can be later converted to IEEE-754 standard-compatible float values.
5. Casting data range. A device is capable of converting measurements into desired decimal values in a linear manner. For Modbus, a user can select between floating point data type – which will return a raw measurement of either voltage or current and scaled integer type – which conversion is freely definable. Note that the specification for IEC-60870-5-103 only lets 13-bit signed measurands, therefore only cast representation of true value can be used with absolute limits being [-4095, 4095].
   Fig. 3.2 shows how raw measurement (current in this case) is converted to an integer type. Conversion is done by defining measurement limits (from -25mA to 25mA) and casting data range (from 0 to 100). Measurement limits define thresholds for overflow and underflow errors respectively; casting data range defines limits (from 0 to 100) that value should be between after recalculation. Such a recalculation can be used in applications that require measurements in relative units such as percentage.
   A device can round measurements to lower conversion limit (in this case – to zero) if a measurement is near it: ±15% of measurement value (Fig. 3.2). Live zero correction is a useful option for eliminating noise. As signals always have some noise level, such an option can compensate it to get a true zero value.

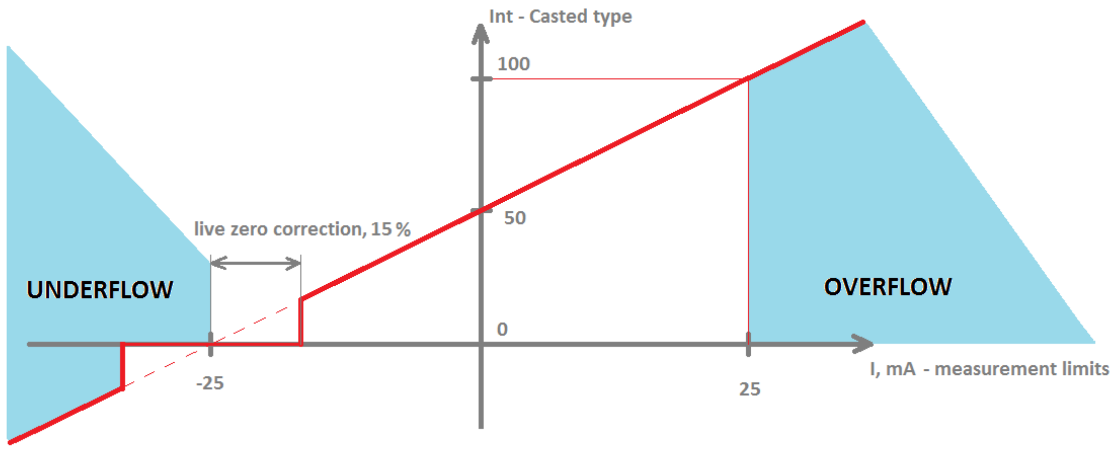

Fig. 3.2. Conversion graph from raw measurement to cast integer

IEC 60870-5-103 operational information

IOMod with IEC-60870-5-103 firmware uses a standard communication scheme. Initiation, control messages and queries are initiated by a master (controlling station), while IOMod device (controlled station) only answers requests and sends values. The first message sent by master should be RESET CU to restart communication. When acknowledge (ACK) packet is sent from a slave device, a master may proceed with acquiring General Interrogation and sending Time synchronization packets.
When this initialization is complete, master should poll IOMod device with Class 1 and Class 2 requests. Class 2 is used when master polls for cyclic data. The controlled device responds when spontaneous data exists and master then sends a request for Class 1. Controlled station responds with a time-tagged message.
As IOMod 8AI doesn't have any digital inputs, only analog ones, general interrogation returns nothing. Values of measurements are returned cyclically without any additional request therefore commands sent will be ignored.

Technical information

System	
Dimensions	101 x 119 x 17.5 mm
Case	ABS, black
Working environment	Indoor
Working temperature	-30°C +70°C
Recommended operating conditions	5 – 60°C and 20 – 80%RH;
Configuration	USB
Firmware upgrade	USB – mass storage device
Electrical specifications	
Inputs	16-bit resolution; Channel-Independent Programmable Input Ranges: ● Voltage input: ○ Bipolar: ±10.24 V, ±5.12 ±2.56 V ○ Unipolar: 10.24 V, 5.12 ● Current input: ○ Bipolar: ±51.2 mA, ±25. and ±12.8 mA ○ Unipolar: 51.2 mA, 25.6 Overvoltage protection up to ±20V;

Power	
Power Supply	9V to 33V
Current consumption	40mA @ 12VDC, 20mA @ 24VDC

Mounting and installation guide

IOMod 8AI RS485 interface

IOMod 8AI has integrated 120Ω termination resistor which can be enabled or disabled over USB configuration. It is recommended to use termination at each end of the RS485 cable. See typical connection diagram in Fig. 5.1.

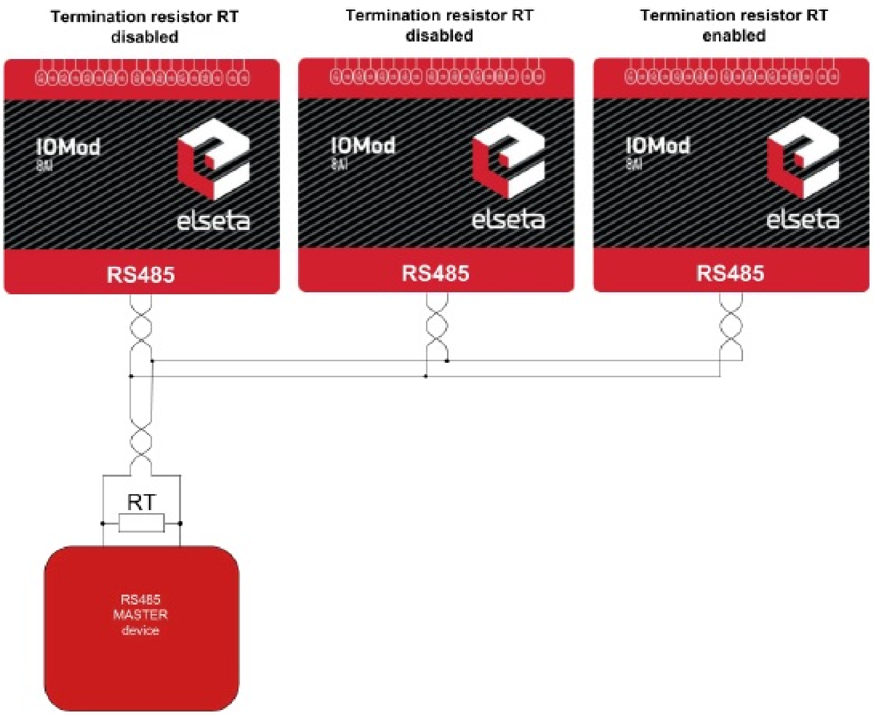

Fig. 5.1. Typical IOMod connection diagram

IOMod 8AI has 1/8 Unit load receiver which allows having up to 255 units on a single line (compared to standard 32 units). To reduce reflections keep the stubs (cable distance from main RS485 bus line) as short as possible.

IOMod 8AI inputs

A typical application of IOMod 8AI unipolar and bipolar voltage inputs is shown in Fig. 5.2. See Configuration over USB chapter for instructions for analog input configuration.

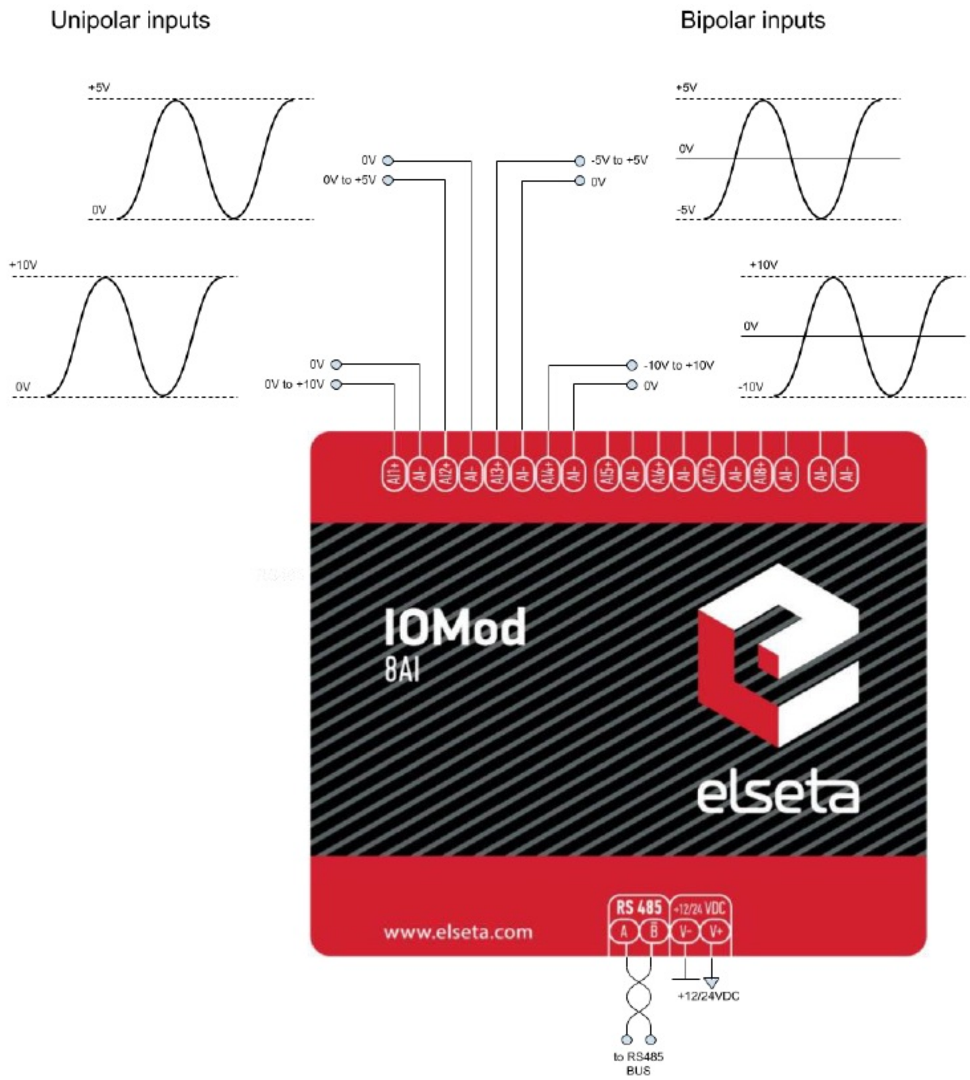

Fig. 5.2. Voltage input connection diagram for IOMod device

All analog inputs can be configured as current inputs to connect 0-20 mA, 4-20 mA or other current output sensors. Typical application of IOMod 8AI unipolar and bipolar current inputs is shown in

Fig. 5.3. See Configuration over USB chapter for instructions for analog input configuration.

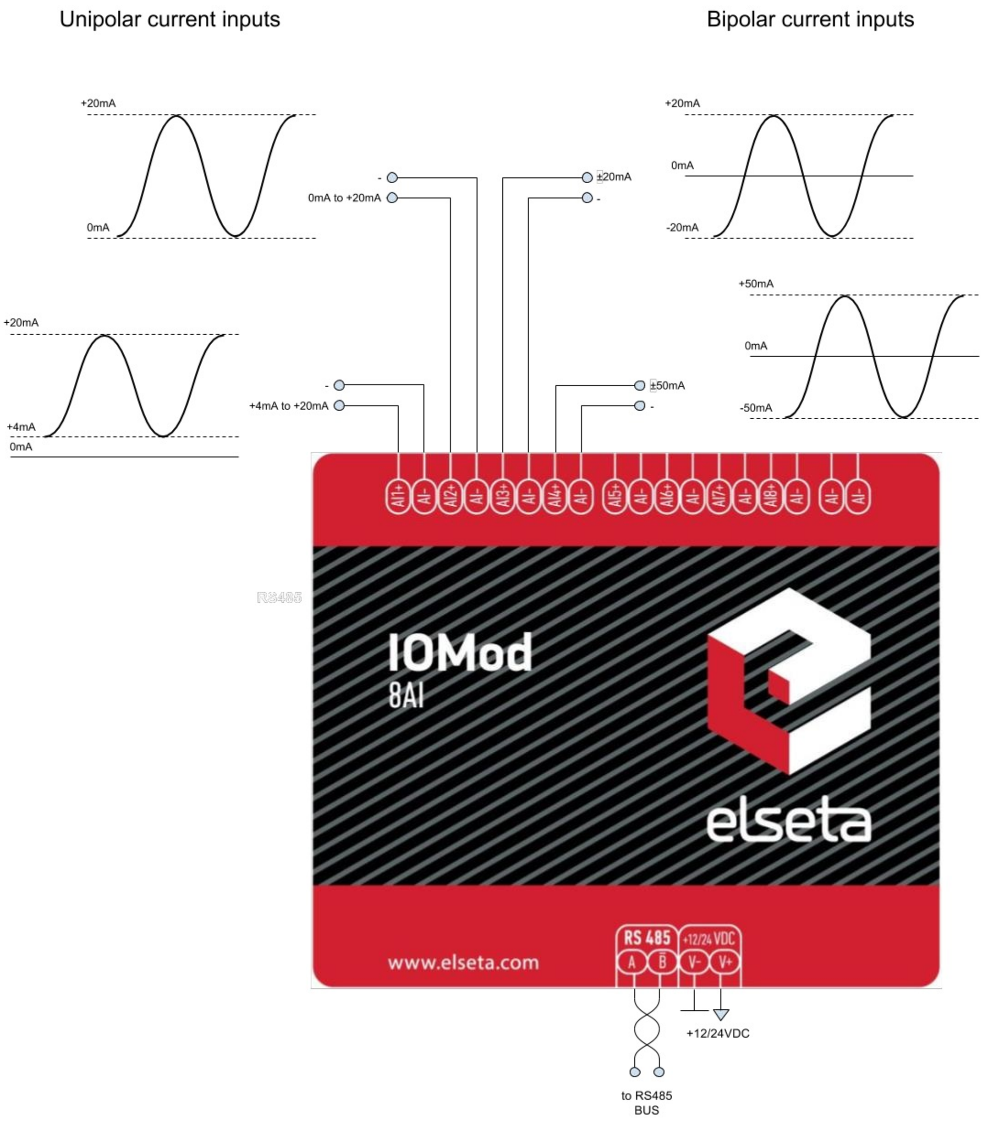

Fig. 5.3. Current input connection diagram for IOMod device

Configuration over USB

Driver installation

The device requires USB drivers to work as a Virtual COM port. The first-time connection between device and computer could result in “Device driver software was not successfully installed” error (as in Fig. 6.1).

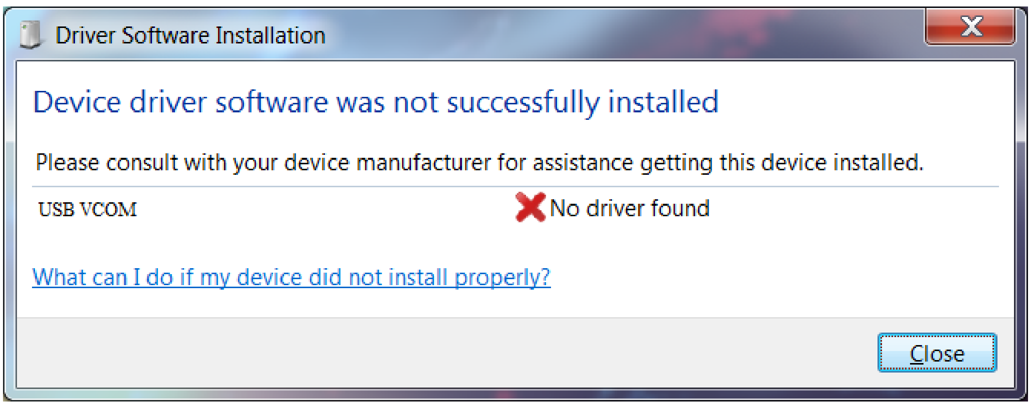

Fig. 6.1. Unsuccessful device software installation error

A user then should manually install drivers by selecting a downloaded driver folder:

- Go to Control Panel -> Device Manager;
- Select a failing device;

* Press “Update driver software”; screen as in Fig. 6.2. should appear:

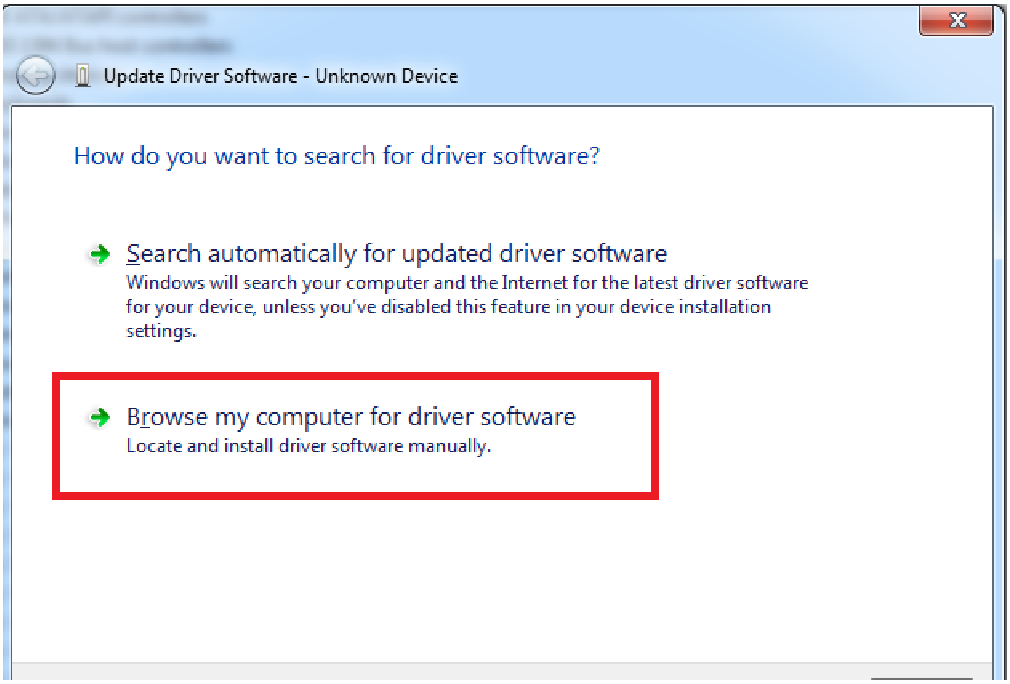

Fig. 6.2. Device driver software update message

* Select “x86” driver for a 32-bit machine or x64 for a 64-bit machine. If not sure, select a root folder (folder in which x64 and x86 lay inside, as in Fig. 6.3).

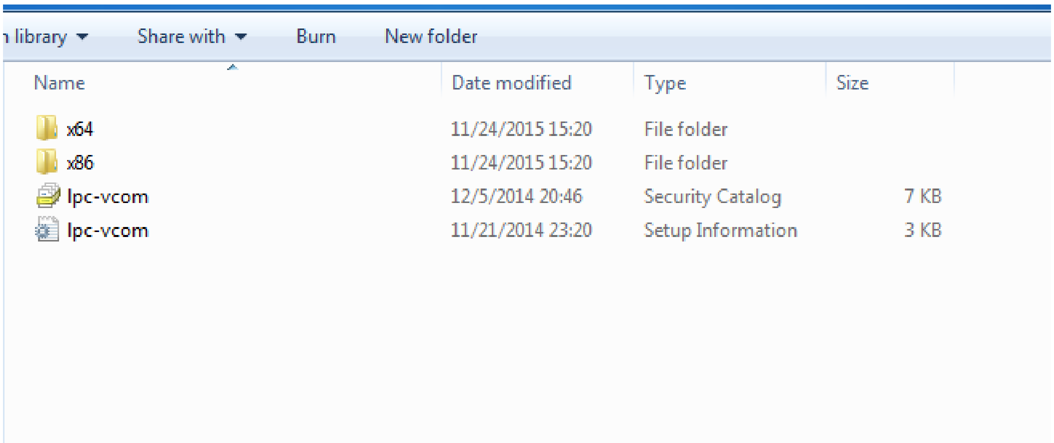

Fig. 6.3. Device driver folder content

IOMod configuration via PuTTY terminal

A configuration of IOMod device is done through CLI (Command Line Interface) on the virtual COM port. Drivers needed for Microsoft Windows to install VCOM will be provided. To open up CLI simply connect to specific V-COM port with terminal software (advised to use PuTTY terminal software. If other software is being used, a user might need to send <return> symbol after each command). When connected user should immediately see the main screen.

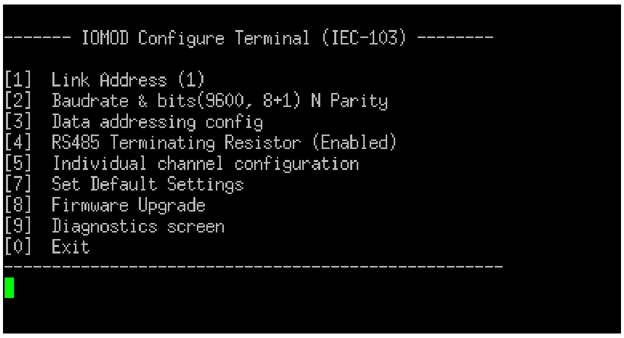

Fig. 6.4. The main menu for IOMod 8AI, IEC-60870-5-103 version

Navigation is performed by sending digits or other symbols to a terminal. A user then proceeds by following on-screen instructions. For example, to set baud rate, press [2] to enter Communication settings screen; press [1] to edit; enter new configuration; press [RETURN] to save, or [ESC] to cancel changes. When done, press [0] (exit) before disconnecting device. Default values are set by pressing [7] on the main screen, and confirming changes [1].

If the terminal window is closed accidentally, a user can connect the terminal program again, and press any key on a keyboard to show the main menu again.

IEC-60870-5-103 Main Menu

	Menu Name	**Function**	**Values**	**Default Values**
[1]	Link Address	Changing link address	1-247	1
[2]	Baudrate, parity and stopbits	[1] Set 8 Data bits + 1 Stop bit [2] Set 8 Data bits + 2 Stop bit [3] Configure baud rate [4] Configure Parity	[3] 100-256000 [4] None/ Odd /Even / Mark/ Space	3] 9600 [4] None
[3]	Data addressing config	Enters configuring screen for Input address (function type)	1-255	160
[4]	RS485 Terminating Resistor	RS485 120 Ohm Terminating Resistor	[1] Enable [2] Disable	Enabled
6.	Individual channel configuration	Channel configuration screen (see Table “Individual Channel Configuration”)		
7.	Set Default Settings	Sets Default Settings	(1 to confirm, 0 to cancel)	-
8.	Firmware Upgrade	Mass Storage Device Firmware Upgrade	(1 to confirm, 0 to cancel)	-
9.	Diagnostics	Input / Output states	-	-
0.	Exit	Exit and disconnect	-	-

Individual channel configuration

	MenuName	**Function**	**Values**	**Default Values**
[1]	Measure Current/Voltage	Selecting this option toggles measurement selection	-	-
[2]	Change sensitivity	Selecting a measurement range	[1] [-10.24 V, 10.24 V], [2] [-5.12 V, 5.12 V], [3] [-2.56 V, 2.56 V], [4] [0 V, 10.24 V], [5] [0 V, 5.12 V]	[1] [-10.24 V, 10.24 V]
[3]	Measurement limits	Entering floating point values to use them for marking overflow/underflow statuses	Floating values in a range defined by measuring sensitivity	-5 mV; 5mV
[0]	Back	Back to last menu screen		

Measurements’ calibration

Every device is shipped containing factory-predefined calibration coefficients. However, changing temperature and humidity conditions can affect the accuracy of measurements. To get the best accuracy, use the device at room temperature and if that is not possible, calibrate coefficients accordingly.
To enter the calibration screen (Fig. 6.6) from main menu press ‘@’ symbol.

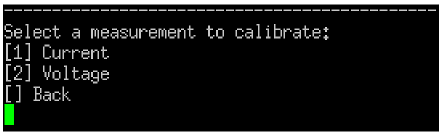

Fig. 6.6. Analog inputs’ calibration screen

In the calibration screen, either current or voltage coefficients can be selected. Selecting current by pressing [1] should enter additional screen (Fig. 6.7). Calibrating voltage is an identical process therefore only current setting is explained.

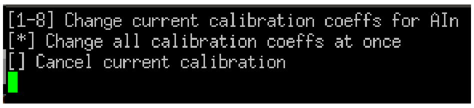

Fig. 6.7. Current calibration screen

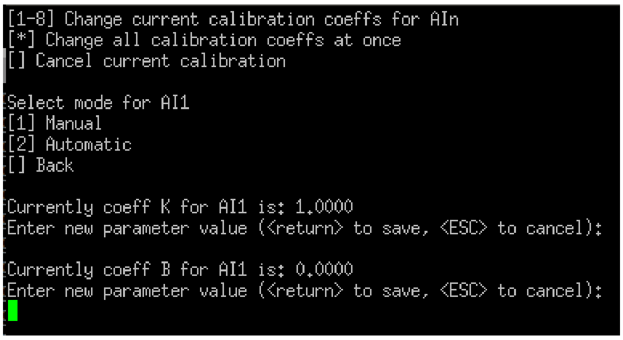

Fig. 6.8. Manual calibration coefficients’ setting

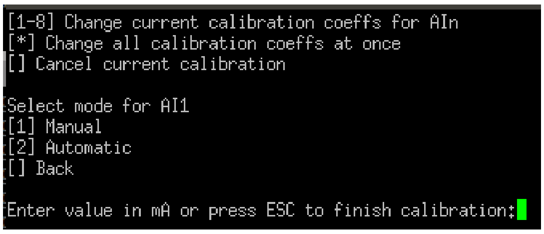

Fig.6.9. Automatic calibration coefficients’ setting

All configuration coefficients can either be configured all at once or individually. The coefficient setting for all channels is done manually only. Tweaking individual channels, however, can be automatic (Fig. 6.9) as well as manual (Fig. 6.8). Selecting automatic calibration polls values that user inputs and tweaks coefficients to minimize the additive error. It is therefore advised to scan several values differing by the same value interval to get the best results. Pressing ESC finishes calibration and saves coefficients to non volatile memory.

Protocol simulation over USB

After entering the diagnostics screen, protocol simulator can be turned on by pressing [9]. When it is on, the device will communicate over USB port rather than a RS-485 line. Communication on a RS-485 line is closed and all Modbus or IEC-60870-5-103 commands will only be accepted from USB. To exit this mode device must be restarted.

Firmware upgrade over USB

To update device firmware user must enter main configuration menu.
Enter Firmware update screen by pressing [8];
Confirm update by pressing [1];
Device now enters Firmware Upgrade mode. Device reconnects as mass storage device (Fig 6.10.).

It is recommended to close terminal window after entering firmware upgrade mode.

Fig. 6.10. Mass storage device warning

User then must delete existing file “firmware.bin”, and simply upload new firmware file by drag and drop. (Fig 6.11.)

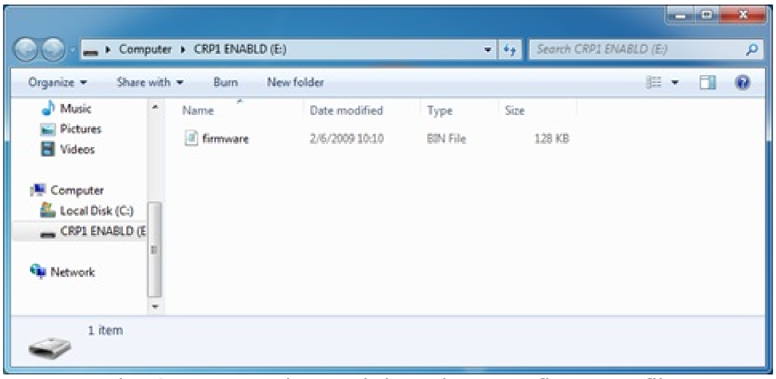

Fig. 6.11. Dragging and dropping new firmware file

Reconnect device and check firmware version. It should now represent the one it was updated to.

Testing With “THE VINCI” software

To test IOMod with default settings, a device should be connected through RS485 to IEC 60870-5-103 Master. Example using “The Vinci Expert” device as serial interface converter and adapter to PC with “The Vinci” software. Default settings for IEC-60870-5-103 – 9600 baud/s baud rate; 8 data, no parity, 1 stop bit. After opening “The Vinci” software choose IEC-60870-5-103 serial – Master mode. In “Settings” tab, choose station number (default – 1) and ASDU (default - 1); press “Start” and go to “Statistic” tab as shown in Fig. 7.2.

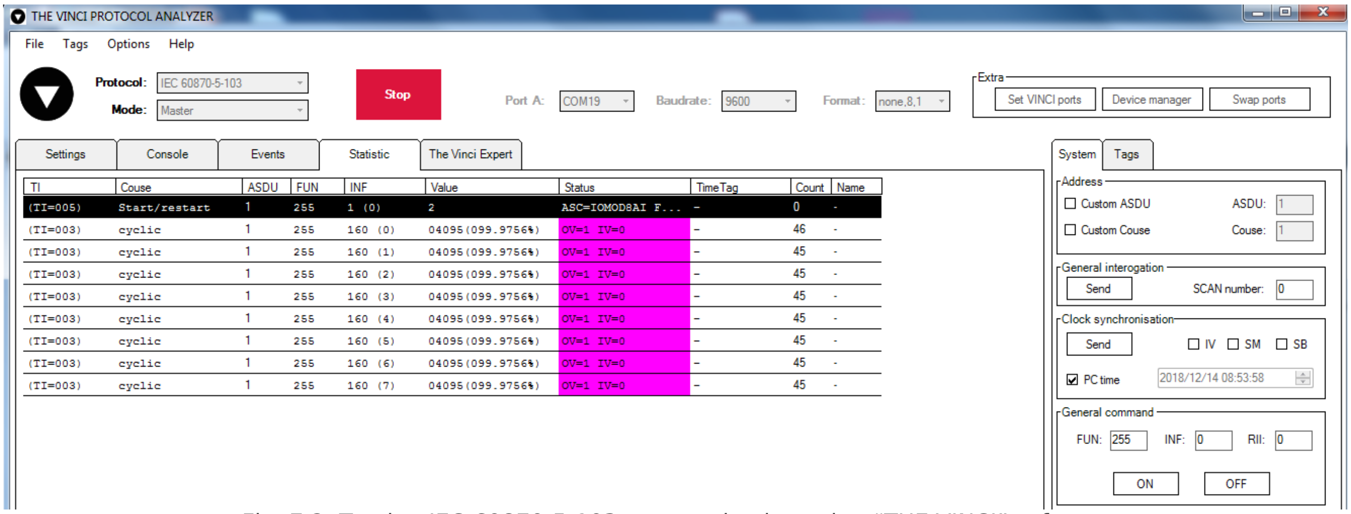

Fig. 7.2. Testing IEC-60870-5-103 communication using “THE VINCI” software

IOMOD 8AI User Manual IEC 60870-5-101

Introduction

IOMod 8AI is a compact-sized stand-alone Modbus (RTU) or IEC60870-5-101 analog input controller. IOMod can be used for industrial applications where digital signaling is used and robust communication is needed. IOMod is an ideal solution for applications such as data acquisition, observation, process monitoring, testing and measurement at remote places. It is controlled over Modbus or IEC 60870-5-101 protocol, and can be used with any SCADA system.

Features

● 8 analog inputs, each configurable separately
● All inputs are capable of measuring electrical voltage or current
● Inputs can be recalibrated according to the needs of a user
● RS485 communication
● LED input indications, + Data transmission (Rx and Tx) indication.
● Configurable over USB
● Drag-and-Drop firmware upgrade over USB
● A small sized case with removable front panel
● DIN rail mount
● Operating temperature: from -30 to +70°C
● Power Requirements: 12-24 VDC

Operational Information

IOMod 8AI uses Modbus (RTU) or IEC 60870-5-103 protocol over RS485 interface. A protocol used by a device can be changed by uploading corresponding firmware. Default communication settings: 9600 baud/s baud rate, 8E1 port configuration, Link address - 1.

Common configuration information

1. Measurement type. A user can select measurement type (electrical current or voltage) on each channel individually.
2. Sensitivity selection. To increase a resolution of input measurement capture, a user can define in which range measurement will occur. The best resolution will be achieved within a shortest selected range. For all possible configurations refer to Technical Information and Configuration over USB described later in this document.
3. Measurement limits can be selected, which in turn set thresholds on underflow or overflow error statuses. Also, if a scaled integer data type is selected, these limits will be converted to values using limits stated in Casting data range for Modbus protocol.
4. Data type selection. A device can output float or scaled integer data types for each input individually for Modbus. Modbus input register read will then show raw float value (in milliamperes or millivolts) or scaled integer types. When a float data type is selected, each input will be represented as two Modbus registers (32-bits). These values can be later
   converted to IEEE-754 standard-compatible float values.
5. Casting data range. A device is capable of converting measurements into desired decimal values in a linear manner. For Modbus, a user can select between floating point data type – which will return a raw measurement of either voltage or current and scaled integer type – which conversion is freely definable. Note that the specification for IEC-60870-5-101 only lets 13-bit signed measurands, therefore only cast representation of true value can be used with absolute limits being [-4095, 4095].
6. Fig. 1. shows how raw measurement (current in this case) is converted to an integer type. Conversion is done by defining measurement limits (from -25mA to 25mA) and casting data range (from 0 to 100). Measurement limits define thresholds for overflow and underflow errors respectively; casting data range defines limits (from 0 to 100) that value should be between after recalculation. Such a recalculation can be used in applications that require measurements in relative units such as percentage.
   A device can round measurements to lower conversion limit (in this case – to zero) if a measurement is near it: ±15% of measurement value (Fig. 1.). Live zero correction is a useful option for eliminating noise. As signals always have some noise level, such an option can compensate it to get a true zero value.

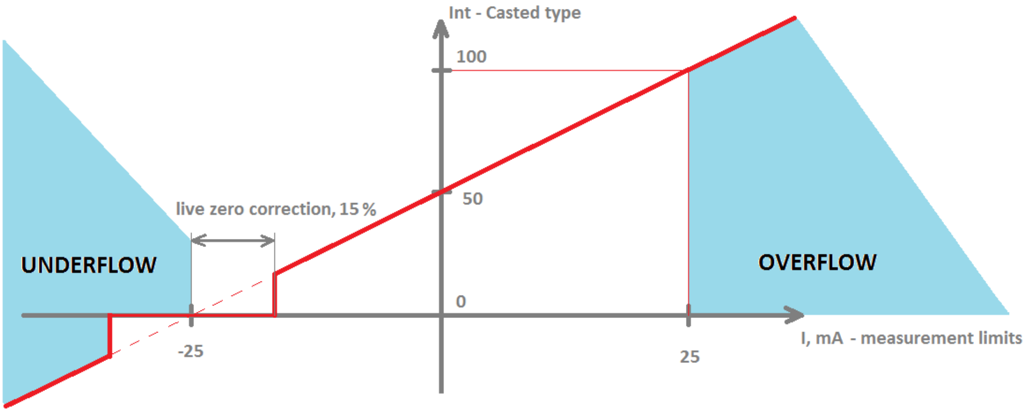

Fig. 1. Conversion graph from raw measurement to cast integer

IEC 60870-5-101 working information

Initialization

IOmod uses a standard IEC-60870-5-101 communication scheme. Initiation, control messages, and queries are initiated by the master (controlling station), while the IOMod device (controlled station) only answers these requests. Therefore, the first message should be sent by the master to start/restart communication (ResetOfRemoteLink). This message is answered by IOMod with an acknowledgment (ACK) to enable the master to proceed with sending other messages defined by the IEC-60870-5-101 protocol.

Data polling

When initialization is complete, the master may request data from the IOmod device with general interrogation. Although according to the protocol specification IOMod will send data on value change. The 8AI IOMod responds with type 30 (M_SP_TB_1) a single point value with a time tag.

Input messages

When input status changes, IOMod device filters input glitches through filters with a user-configurable filter time. When the filter is passed device sends a “Spontaneous” message with the 30 data types (M_SP_TB_1), and “IOA” as the input pin number shifted by 8.

Time synchronization

To initiate the time synchronization between devices the master must send a Clock Sync command. The command type is C_CS_NA_1 (103) and the Cause of Transmission (COT) has to be 6. The command has to be sent to the correct link address and CASDU, which is the same as the link address by default. If the sent frame is correct the IOMod will respond with a C_CS_NA_1 (103) type command with the COT (cause of transmission) of 7 and the **p/n** bit will be positive (0) also the command will be time-tagged with the **device** time. If the time synchronization feature is disabled or the command is sent to an undefined CASDU the response is the same except the **p/n** bit will be negative (1).

General interrogation

General Interrogation (GI) is initiated by the master sending the General Interrogation command. The command type is C_IC_NA_1 (100) and the Cause of Transmission (COT) has to be 6. The command has to be sent to the correct link address and CASDU, which is the same as the link address by default. If the sent frame is correct the IOMod will respond with a C_IC_NA_1 (103) type command with the COT (cause of transmission) of 7 and the **p/n** bit will be positive (0). Otherwise, it will respond with the same command just that the **p/n** bit will be negative (1). Then the device will begin to send all of its data. After that's done the IOMOD will also send another 100 type command with the COT (cause of transmission) of 10 (ActTerm) meaning the general interrogation is over.

IOAs [1,8] inputs.

Technical information

System	
Dimensions	101 x 119 x 17.5 mm
Case	ABS, black
Working environment	Indoor
Working temperature	-30 °C+70°C

Recommended operating conditions	5 – 60°C and 20 – 80%RH;
Configuration	USB
Firmware upgrade	USB – mass storage device
Electrical specifications	
Inputs	16-bit resolution; Channel-Independent Programmable Input Ranges: ● Voltage input: ○ Bipolar: ±10.24 V, ±5.12 ±2.56 V ○ Unipolar: 10.24 V, 5.12 ● Current input: ○ Bipolar: ±51.2 mA, ±25. and ±12.8 mA ○ Unipolar: 51.2 mA, 25.6 Overvoltage protection up to ±20V;
Power	
Power Supply	9V to 33V
Current consumption	40mA @ 12VDC, 20mA @ 24VDC

Mounting and installation guide

IOMod 8AI RS485 interface

IOMod 8AI has integrated 120Ω termination resistor which can be enabled or disabled over USB configuration. It is recommended to use termination at each end of the RS485 cable. See typical connection diagram in Fig. 2.

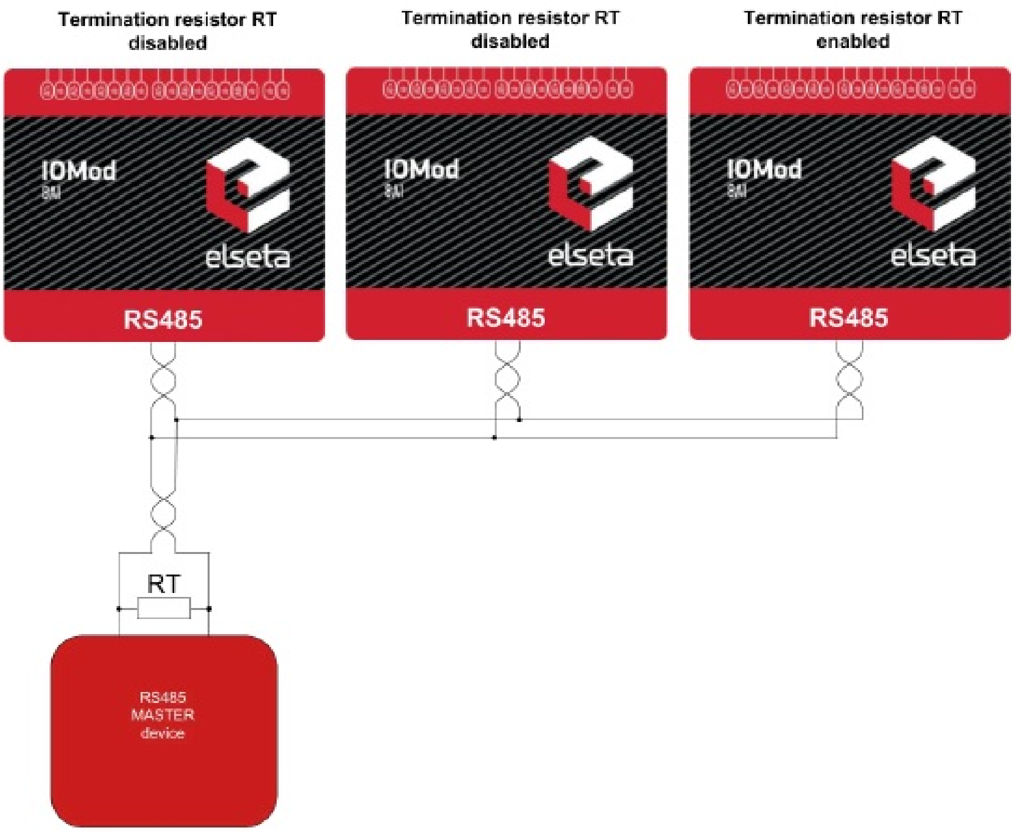

Fig. 2. Typical IOMod connection diagram

IOMod 8AI has 1/8 Unit load receiver which allows having up to 255 units on a single line (compared to standard 32 units). To reduce reflections keep the stubs (cable distance from main RS485 bus line) as short as possible.

IOMod 8AI inputs

A typical application of IOMod 8AI unipolar and bipolar voltage inputs is shown in Fig. 3. See Configuration over USB chapter for instructions for analog input configuration.

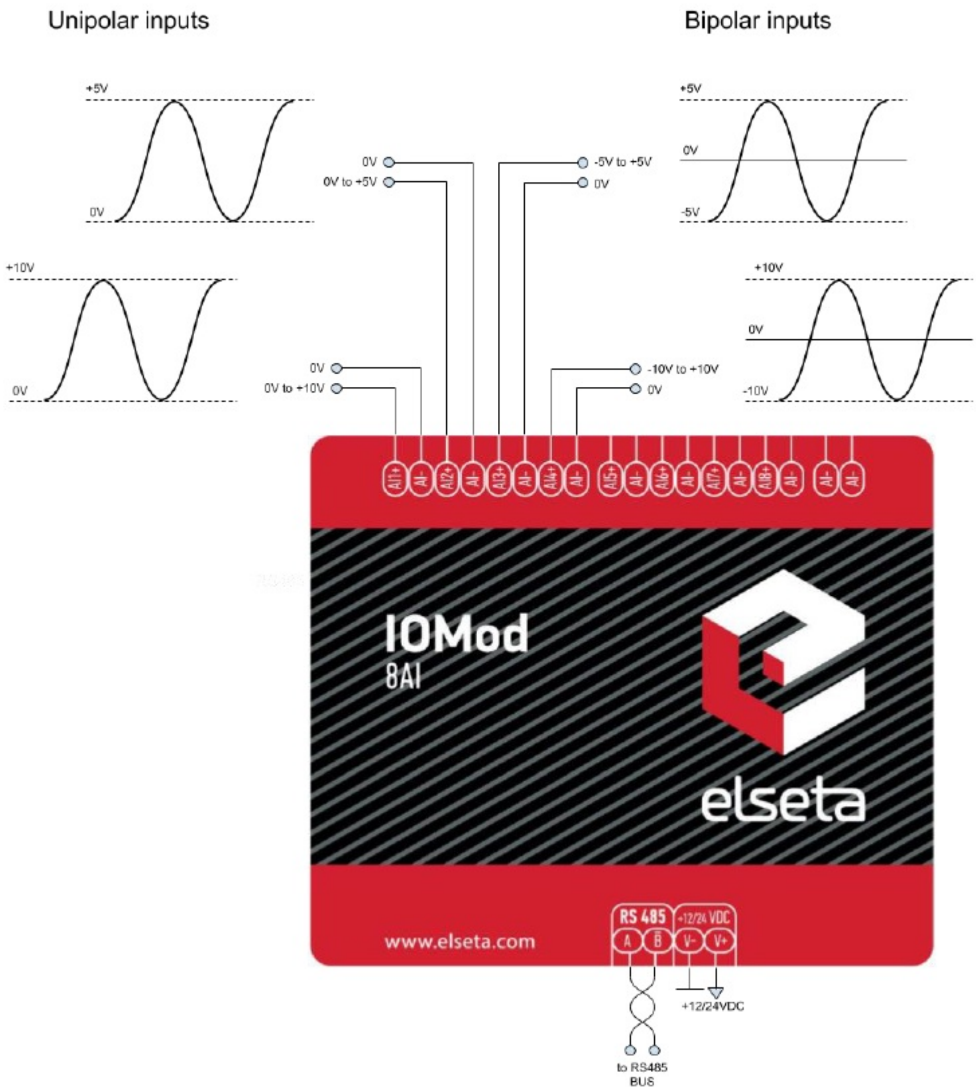

Fig. 3. Voltage input connection diagram for IOMod device

All analog inputs can be configured as current inputs to connect 0-20 mA, 4-20 mA or other current output sensors. Typical application of IOMod 8AI unipolar and bipolar current inputs is shown in Fig. 4. See Configuration over USB chapter for instructions for analog input configuration.

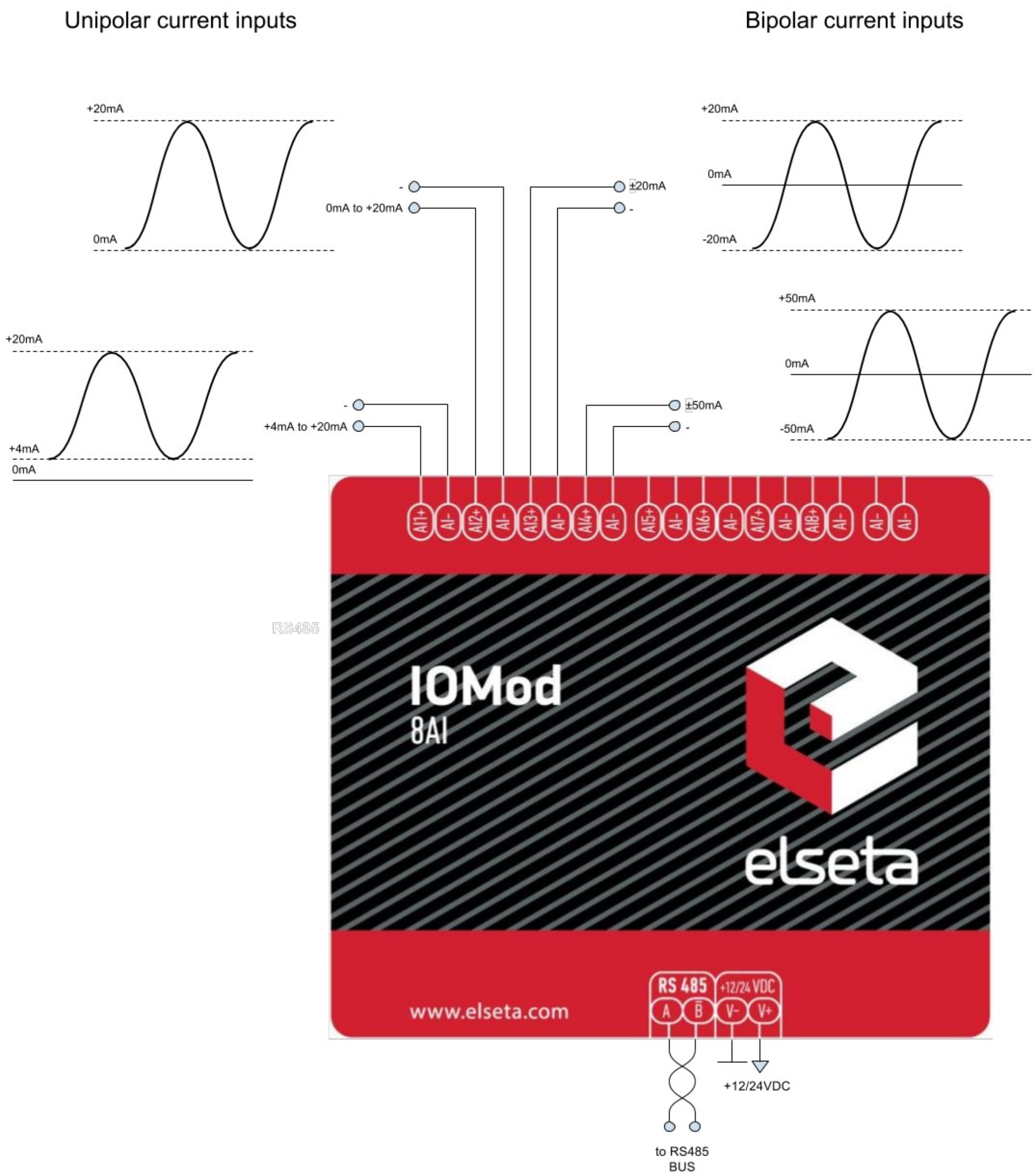

Fig. 4. Current input connection diagram for IOMod device

Configuration over USB

Driver installation

The device requires USB drivers to work as a Virtual COM port. The first-time connection between device and computer could result in “Device driver software was not successfully installed” error (as in Fig. 5.)

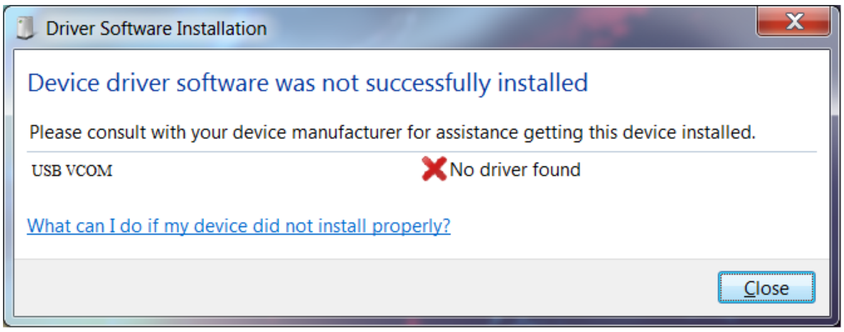

Fig. 5. Unsuccessful device software installation error

A user then should manually install drivers by selecting a downloaded driver folder:

- Go to Control Panel -> Device Manager;

- Select a failing device;
- Press “Update driver software”; screen as in Fig. 6. should appear:

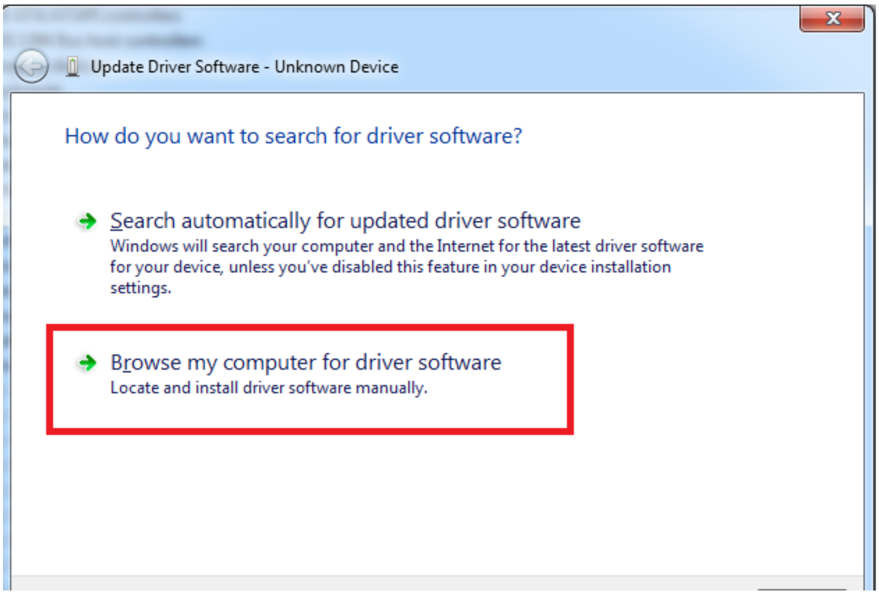

Fig. 6. Device driver software update message

- Select “x86” driver for a 32-bit machine or x64 for a 64-bit machine. If not sure, select a root folder (folder in which x64 and x86 lay inside, as in Fig. 7.)

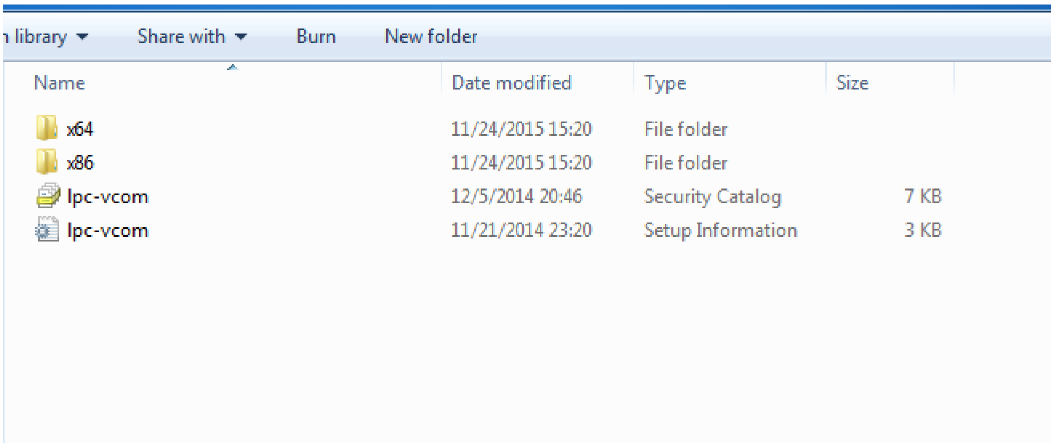

Fig. 7. Device driver folder content

IOMod configuration via PuTTY terminal

A configuration of IOMod device is done through CLI (Command Line Interface) on the virtual COM port. Drivers needed for Microsoft Windows to install VCOM will be provided. To open up CLI simply connect to specific V-COM port with terminal software (advised to use PuTTY terminal software. If other software is being used, a user might need to send <return> symbol after each command). When connected user should immediately see the main screen.

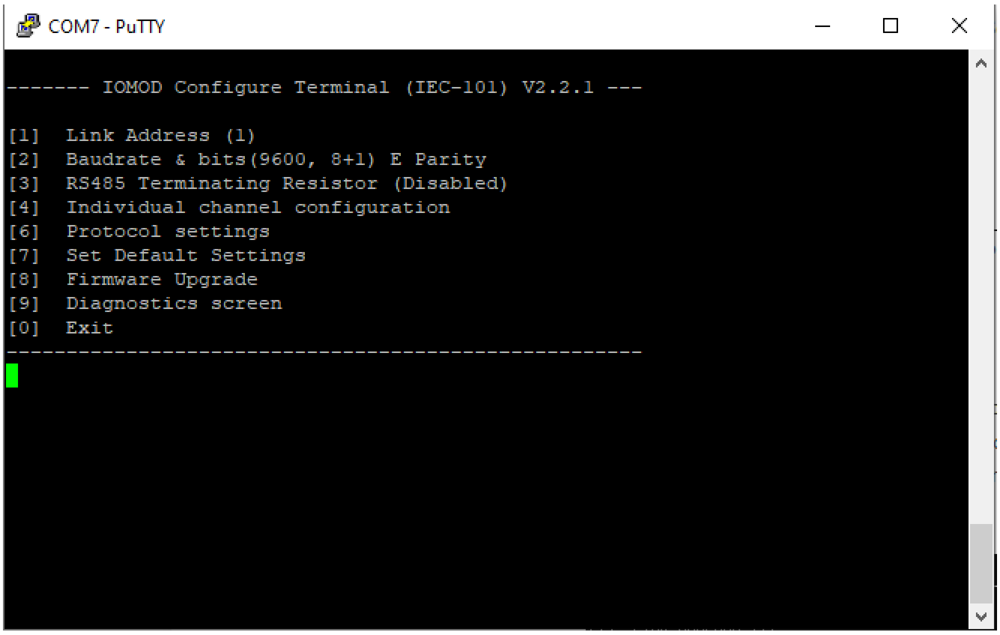

Fig. 8. The main menu for IOMod 8AI, IEC-60870-5-101 version

Navigation is performed by sending digits or other symbols to a terminal. A user then proceeds by following on-screen instructions. For example, to set baud rate, press [2] to enter Communication settings screen; press [3] to edit; enter new configuration; press [RETURN] to save, or [ESC] to cancel changes. When done, press [0] (exit) before disconnecting device. Default values are set by pressing [7] on the main screen, and confirming changes [1].

If the terminal window is closed accidentally, a user can connect the terminal program again, and press any key on a keyboard to show the main menu again.

IEC-60870-5-101 Main Menu

	Menu name	Function	Values	Default values
[1]	Link Address	Changing link address	1-247	1
[2]	Baudrate, parity and stopbits	[1] Set 8 Data bits + 1 Stop bit [2] Set 8 Data bits + 2 Stop bit [3] Configure baud rate [4] Configure Parity	[3] 100-256000 [4] None/ Odd /Even / Mark/ Space	[3] 9600 [4] Even
[3]	RS485 Terminating Resistor	RS485 120 Ohm Terminating Resistor	[1] Enable [2] Disable	Disabled
[4]	Individual channel configuration	Channel configuration screen (see Table “Individual Channel Configuration”)		
[6]	Protocol settings	-		
[6].[1]	Toggle 24/56 bit time	Changes time mode for info mode to 24 or 56 bits by pressing [1]	24/56	56 bit
[6].[2]	Change IOA size	[1] 1 [2] 2 [3] 3 [0] back	1-3	1
[6].[3]	Toggle measurements type	Changes measurement type to integer or float by pressing [3]	Integer/Float	Float
[6].[4]	Toggle measurement time	Enables or disables timed measurements by pressing [4]	Enabled/Disabled	Enabled
[7]	Set Default Settings	Sets Default Settings	(1 to confirm, 0 to cancel)	-
[8]	Firmware Upgrade	Mass Storage Device Firmware Upgrade	(1 to confirm, 0 to cancel)	-
[9]	Diagnostics screen	Input states	-	-
[0]	Exit	Exit and disconnect	-	-

Individual channel configuration

	Menu name	Function	Values	Default values

[1]	Measure Current/Voltage	Selecting this option toggles measurement selection	-	-
[2]	Change sensitivity	Selecting a measurement range	[1] [-10.24 V, 10.24 V], [2] [-5.12 V, 5.12 V], [3] [-2.56 V, 2.56 V], [4] [0 V, 10.24 V], [5] [0 V, 5.12 V]	[1] [-10.24 V, 10.24 V]
[3]	Measurement limits	Entering floating point values to use them for marking overflow/underflow statuses	Floating values in a range defined by measuring sensitivity	[-10240.00 mV; 10210.00mV]
[4]	Live zero tolerance	Enter live zero tolerance (percent)	Floating values in a range defined by measuring sensitivity	(0.0000) %
[0]	Back	Back to last menu screen		

Measurements' calibration

Every device is shipped containing factory-predefined calibration coefficients. However, changing temperature and humidity conditions can affect the accuracy of measurements. To get the best accuracy, use the device at room temperature and if that is not possible, calibrate coefficients accordingly.
To enter the calibration screen (Fig. 9.) from main menu press '@' symbol.

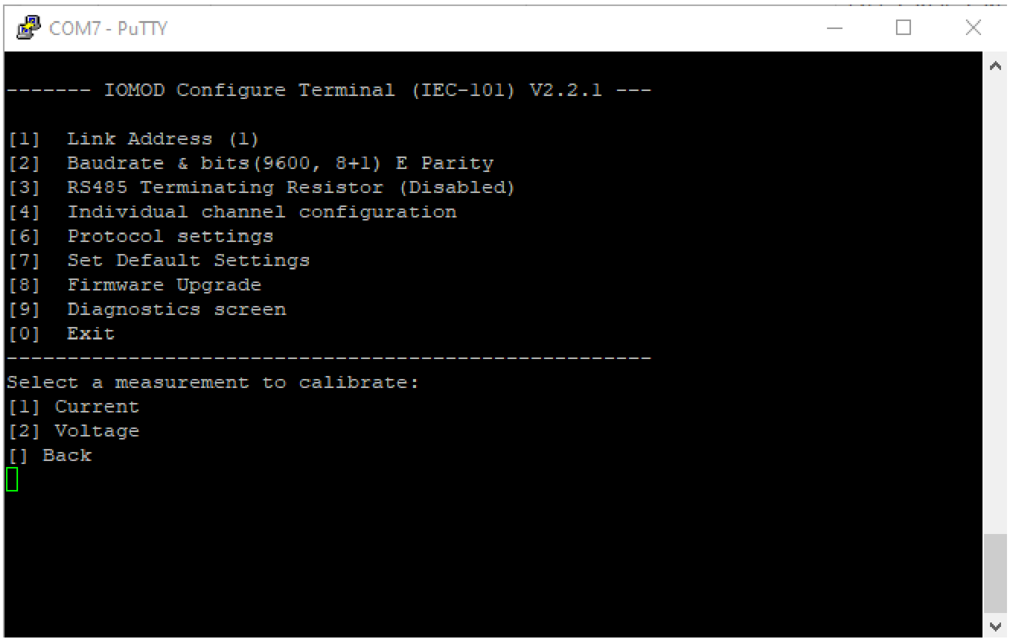

Fig. 9. Analog inputs' calibration screen

In the calibration screen, either current or voltage coefficients can be selected. Selecting current by pressing [1] should enter additional screen (Fig. 10.) Calibrating voltage is an identical process therefore only current setting is explained.

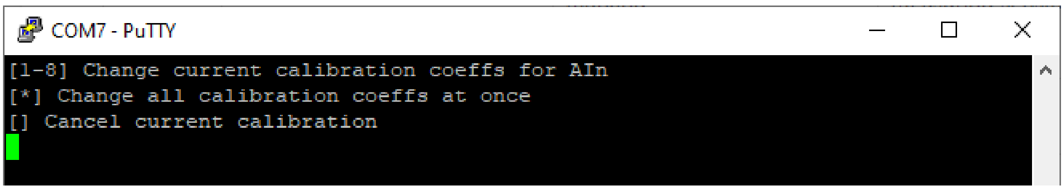

Fig.10. Current calibration screen

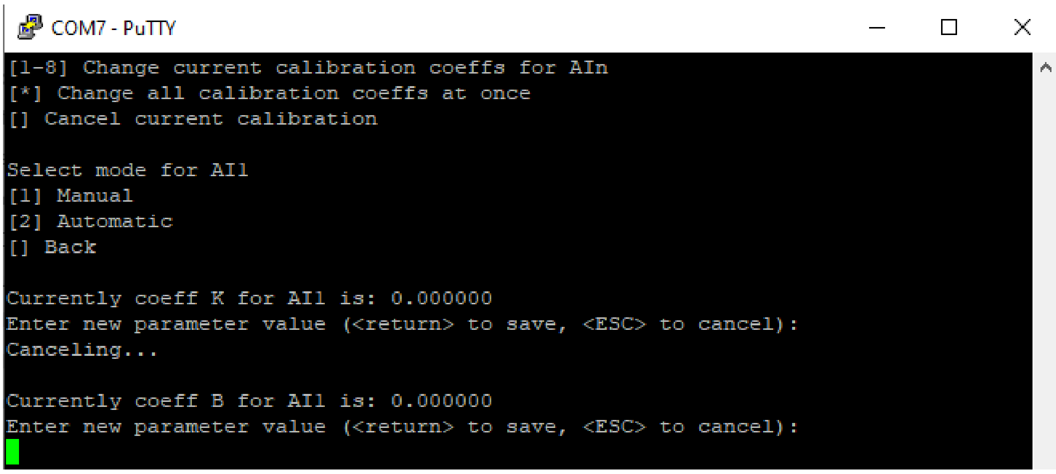

Fig. 10. Manual calibration coefficients' setting

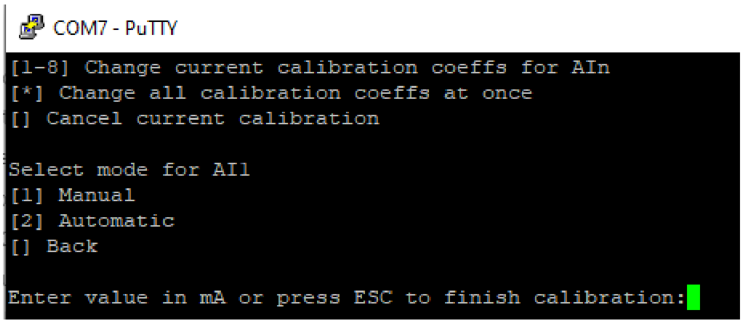

Fig. 11. Automatic calibration coefficients' setting

All configuration coefficients can either be configured all at once or individually. The coefficient setting for all channels is done manually only. Tweaking individual channels, however, can be automatic (Fig. 11.) as well as manual (Fig. 10.). Selecting automatic calibration polls values that user inputs and tweaks coefficients to minimize the additive error. It is therefore advised to scan several values differing by the same value interval to get the best results. Pressing ESC finishes calibration and saves coefficients to non volatile memory.

Protocol simulation over USB

After entering the diagnostics screen, protocol simulator can be turned on by pressing [9]. When it is on, the device will communicate over USB port rather than a RS-485 line. Communication on a RS-485 line is closed and all Modbus or IEC-60870-5-101 commands will only be accepted from USB. To exit this mode device must be restarted.

Firmware upgrade over USB

To update device firmware user must enter main configuration menu.
Enter Firmware update screen by pressing [8];
Confirm update by pressing [1];
Device now enters Firmware Upgrade mode. Device reconnects as mass storage device (Fig. 12.).

!	It is recommended to close terminal window after entering firmware upgrade mode.

Fig. 12. Mass storage device warning

User then must delete existing file “firmware.bin”, and simply upload new firmware file by drag and drop. (Fig. 13.)

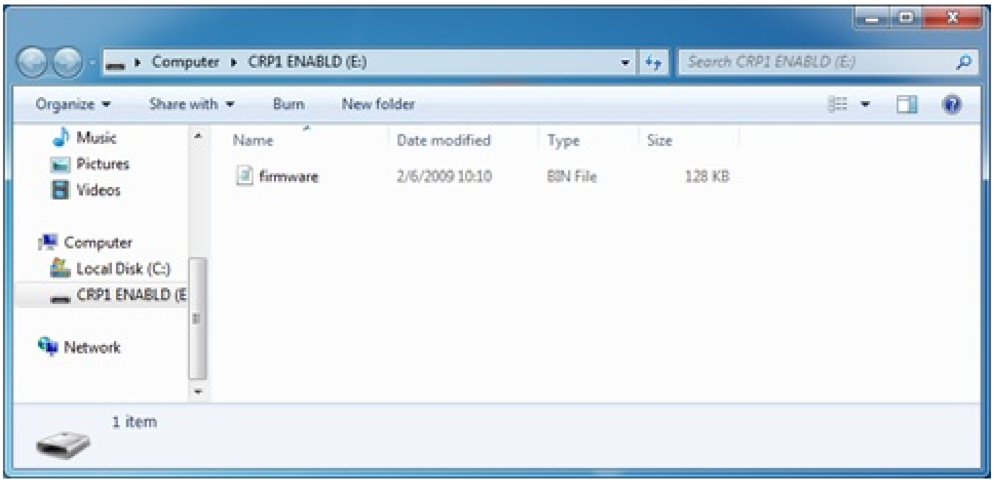

Fig. 13. Dragging and dropping new firmware file

Reconnect device and check firmware version. It should now represent the one it was updated to.

Firmware version 2

Firmware version 2

IOMOD 8AI User Manual

1. Introduction

IOMOD 8AI is a compact analog input module, designed for industrial applications where reliable digital signaling and robust communication are essential. IOMod 8AI can be used for numerous applications where user needs to log Voltage or Current changes. IOMod can be used to measure temperature, pressure, water level or weight with corresponding sensors (e.g. 4-20mA). IOMod 8AI input measurement resolution, data scaling and data casting can be configured by user for each channel individually. The IOMOD 8AI is controlled via **Modbus, IEC 60870-5-101, or IEC 60870-5-103** protocols, making it compatible with any SCADA system for seamless integration.

1.1 Features

- 8 analog inputs, galvanically isolated, each configurable separately.
- All inputs are capable of measuring electrical voltage or current.
- Firmware upgrade over USB, RS485.
- Configurable using the IOMOD Utility app for user-friendly setup.
- Compact case with a removable transparent front panel.
- DIN rail mounting for seamless integration into industrial systems.
- RS485 interface with a switchable terminating resistor.

1.2 Block Diagram

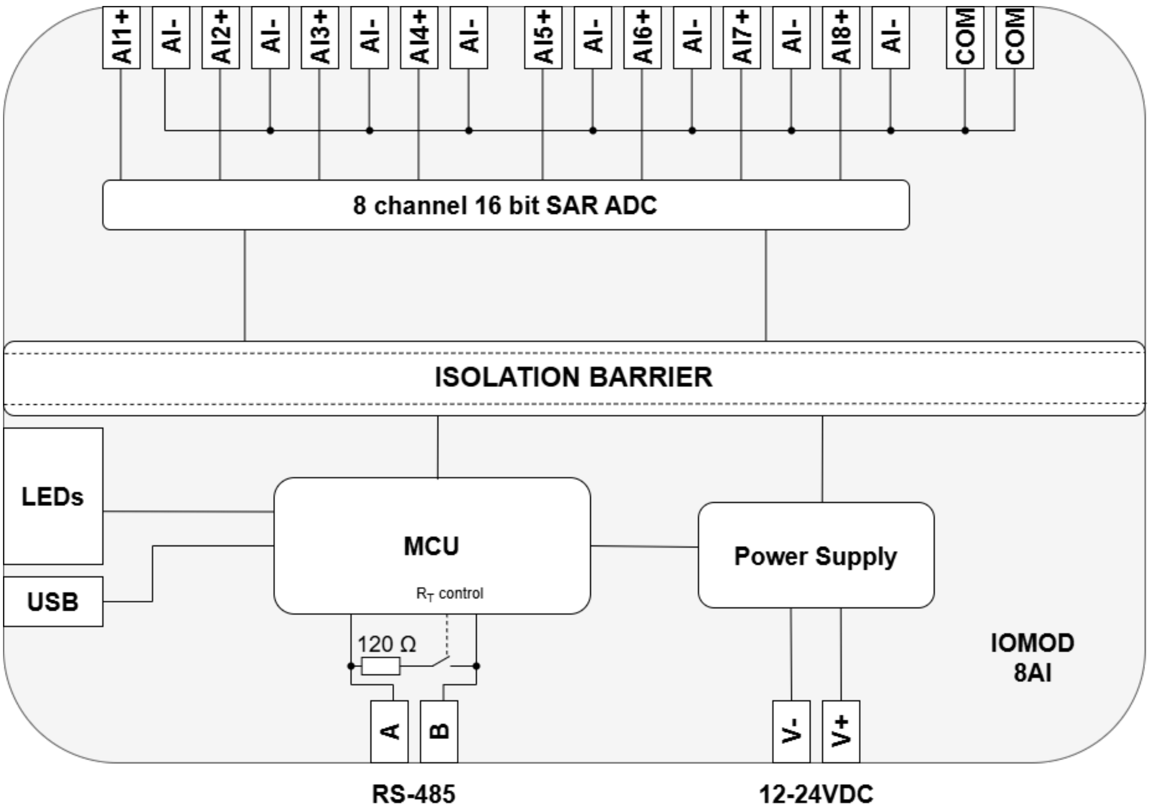

Fig. 1.2.1 IOMOD 8AI internal structure and block diagram

2. Hardware data

2.1 Mechanical drawings

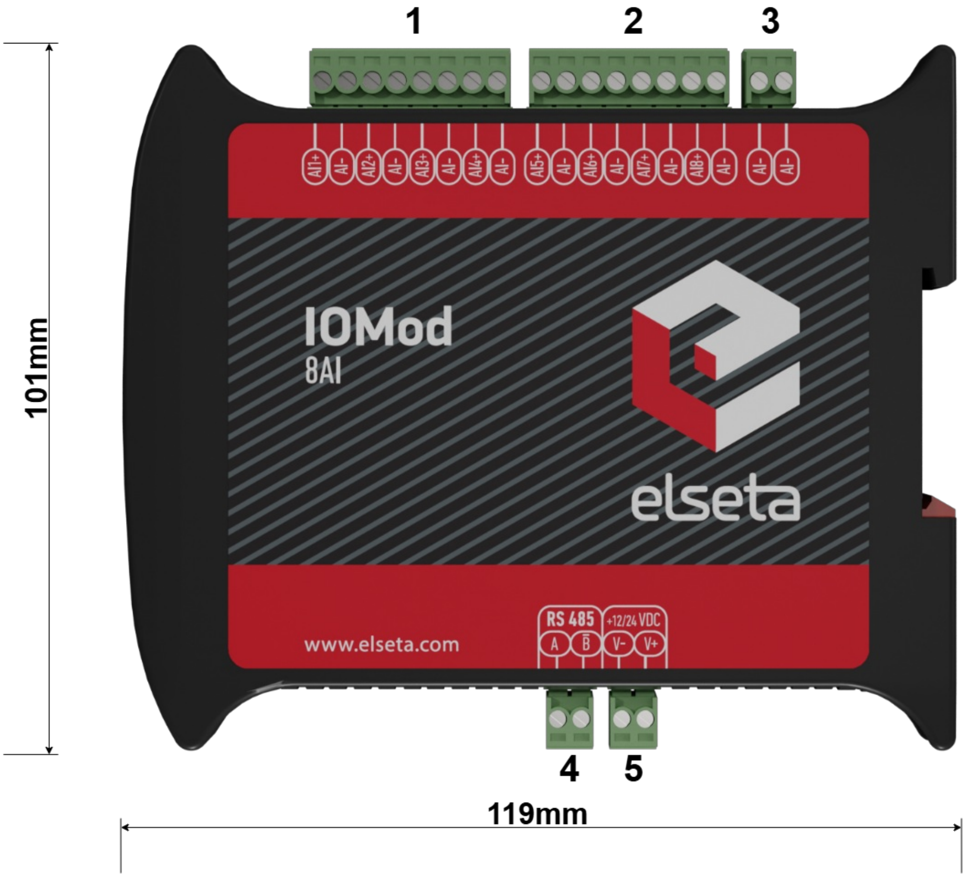

Fig. 2.1.1 IOMOD 8AI side view with dimensions and terminals description. 1 - 2 analog inputs with ground, 3 - Common, 4 - RS485 interface, 5 - power supply input

Fig. 2.1.2 IOMOD 8AI front view with measurements

2.2 Terminal connections

IOMod 8AI has 22 terminals, which are depicted below:

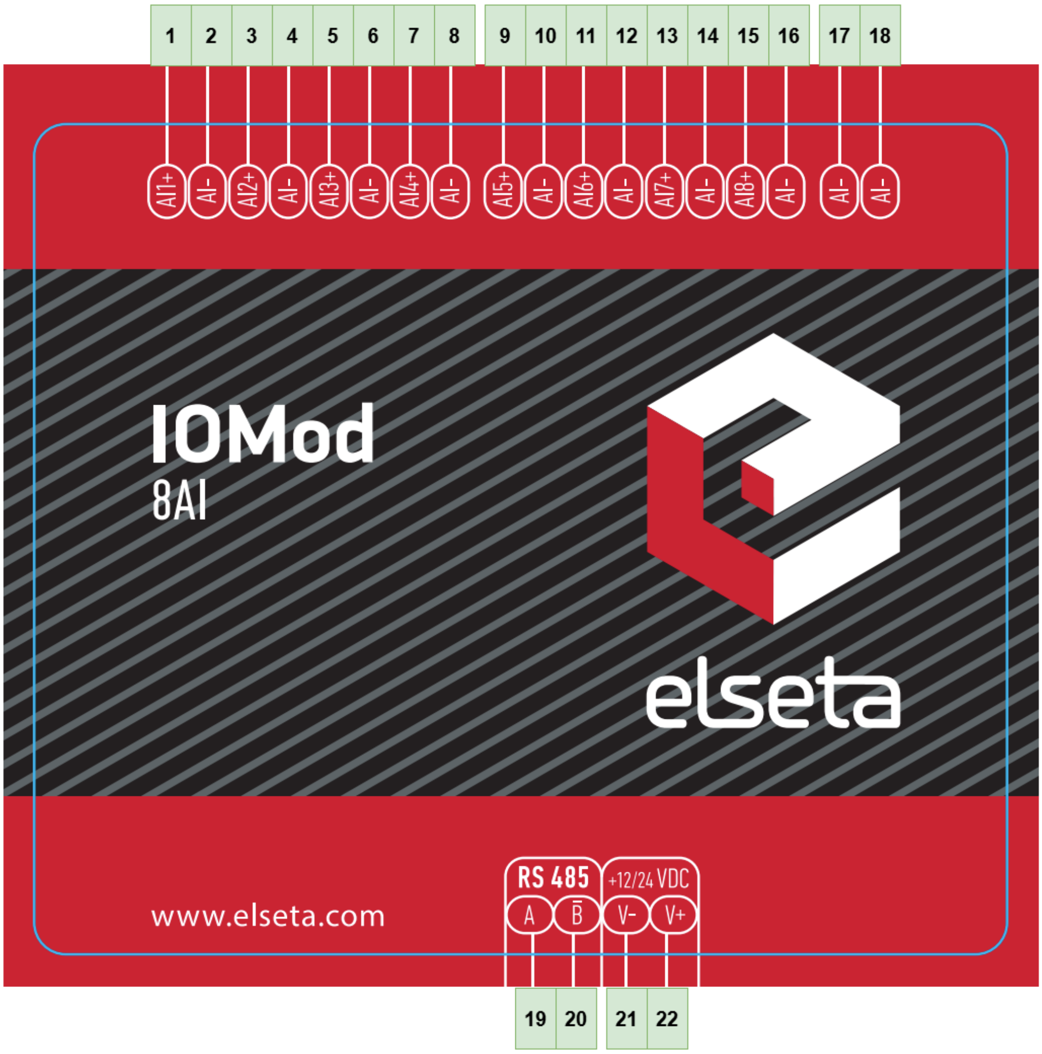

Fig. 2.2.1 IOMod 8AI terminal diagram

The description of each terminal can be found in the table below:

Table 2.2.1 Terminal Specifications

Terminal number	Terminal name	Description
1	AI1+	Analog input 1
2	AI-	Common
3	AI2+	Analog input 2
4	AI-	Common
5	AI3+	Analog input 3
6	AI-	Common
7	AI4+	Analog input 4
8	AI-	Common

Terminal number	Terminal name	Description
9	AI5+	Analog input 5
10	AI-	Common
11	AI6+	Analog input 6
12	AI-	Common
13	AI7+	Analog input 7
14	AI-	Common
15	AI8+	Analog input 8
16	AI-	Common
17	AI-	Common
18	AI-	Common
19	A	RS485 input
20	B̄	
21	V-	Power source input
22	V+	

2.3 Status indication

IOMod 8AI has two LEDs, which are used to indicate communication and power statuses.

Fig. 2.3.1 IOMod 8AI LEDs physical location

The description of each IOMod 8AI LED can be found in the table below:

Table 2.3.1 Description of LEDs

Name	LED color	Description
RX/TX	□ (green)	A blinking green light indicates active communication via the RS485 interface.
STAT	□ (green)	The power source is connected to the power supply input.
	□ (blue)	IOMod 8AI is connected to an external device via a USB mini cable.

3. Technical information

Table 3.1. Technical specifications.

System	
Dimensions	17.5 (H) x 101 (W) x 119 (L), mm
Case	ABS, black
Working environment	Indoor
Working temperature	-40°C ... +85°C
Recommended operating conditions	5 – 60°C and 20 – 80%RH;
Configuration	USB, RS485
Firmware upgrade	USB, RS485
Electrical specifications	
Inputs	16-bit resolution; Channel-Independent Programmable Input Ranges: ● Voltage input: ○ Bipolar: ±10.24 V, ±5.12 V, ±2.56 V ○ Unipolar: 10.24 V, 5.12 V ● Current input: ○ Bipolar: ±45.5 mA, ±22.75 and ±11.38 mA ○ Unipolar: 45.5 mA, 22.75 mA Overvoltage protection up to ±20V;
Power	
Power Supply	9 - 33 VDC
Current consumption	40mA @ 12VDC, 20mA @ 24VDC

4. Mounting and installation

4.1 IOMOD 8AI inputs

A typical application of IOMod 8AI unipolar and bipolar voltage inputs is shown in Fig. 4.1.1

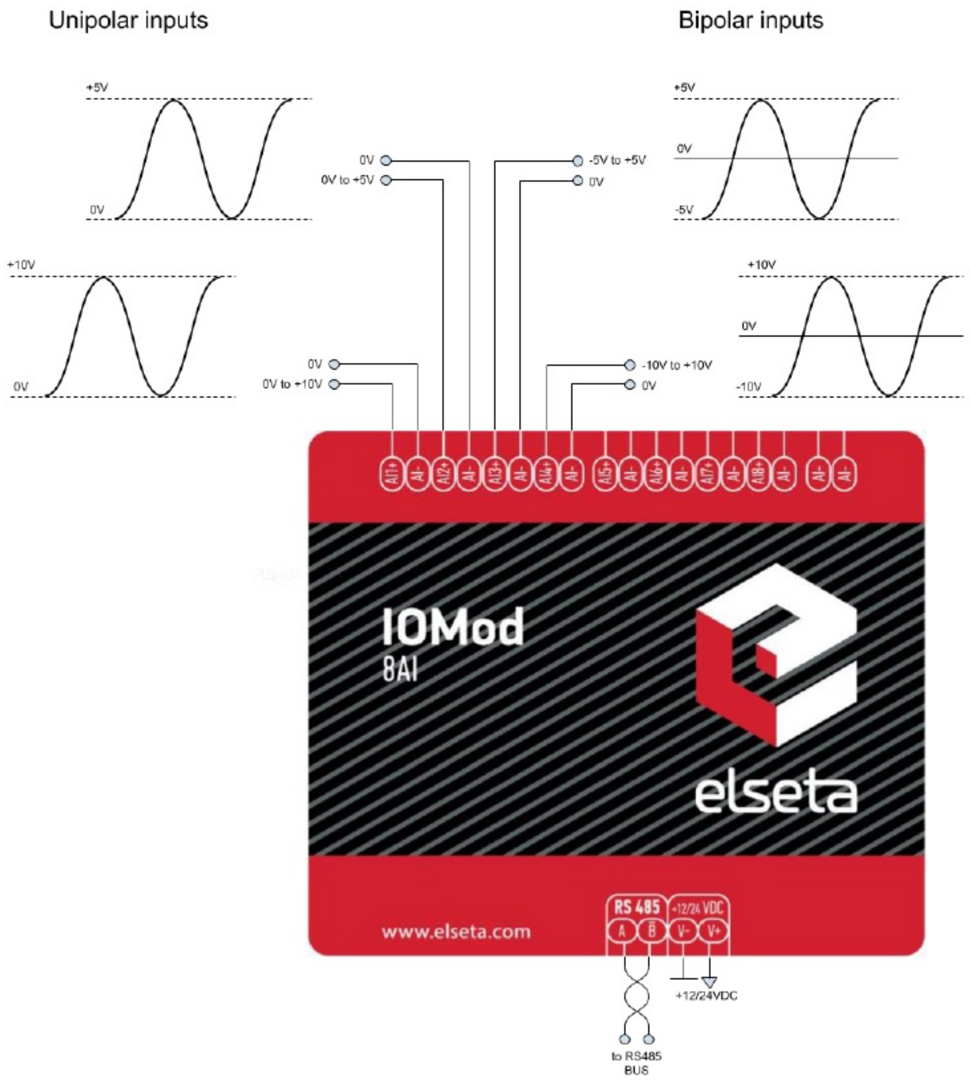

Fig. 4.1.1 Voltage input connection diagram for IOMod device

All analog inputs can be configured as current inputs to connect 0-20 mA, 4-20 mA or other current output sensors. Typical application of IOMod 8AI unipolar and bipolar current inputs is shown in Fig. 4.1.2.

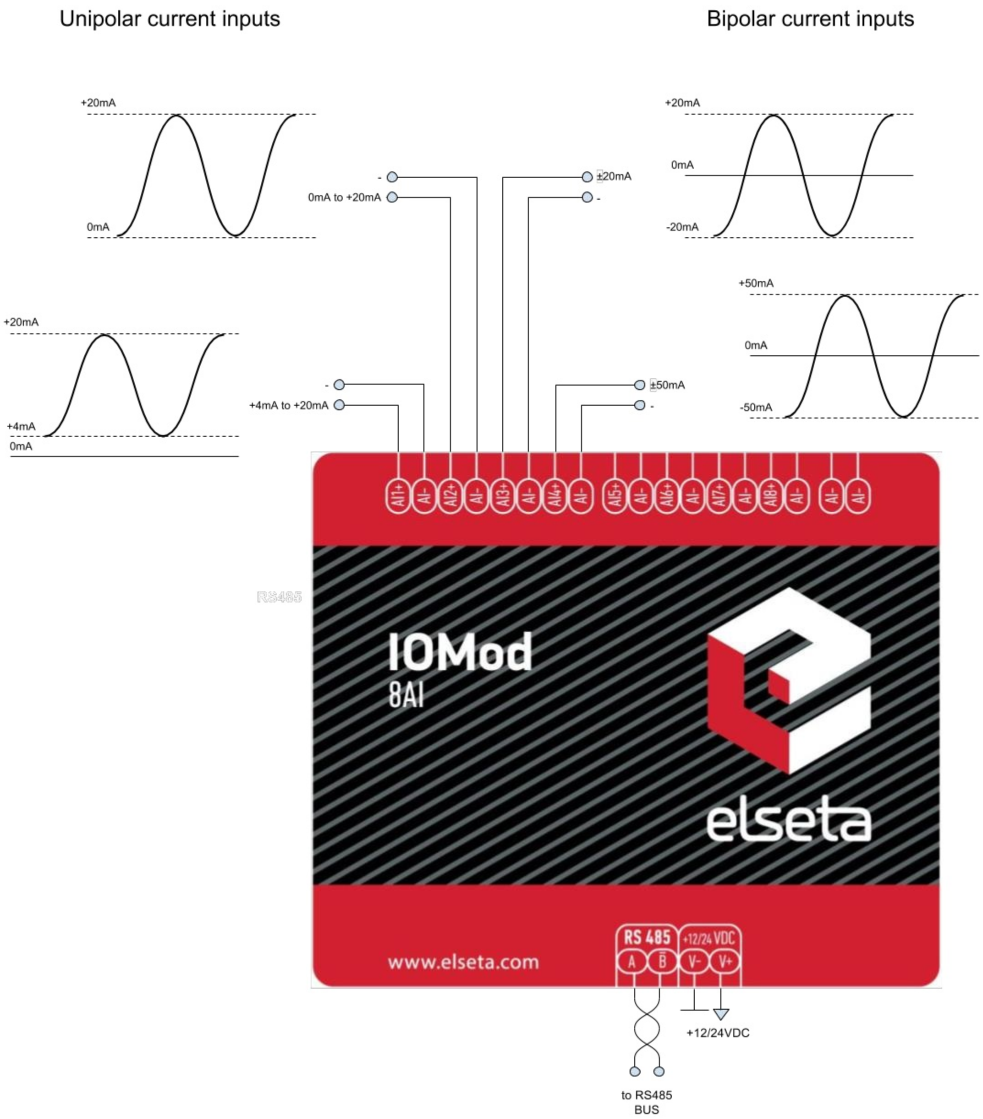

Fig. 4.1.2 Current input connection diagram for IOMod device

4.2 Power connection

IOmod 8AI can be powered through the main power connector +12/24 VDC or USB. Apply 12/24VDC power supply to the V+ and V- terminals. The device has a built-in reverse voltage polarity, overcurrent and overvoltage protection.

Fig. 4.2.1 Power supply inputs physical location

4.3 USB connection

IOMod 8AI device has a USB-mini connection port. Its primary function is the physical connection establishment between the IOMod and a PC. By selecting the USB interface and correct communication port in IOMod Utility (Fig. 4.3.1) a user can connect to the IOMod to control its parameters and monitor its measured data.

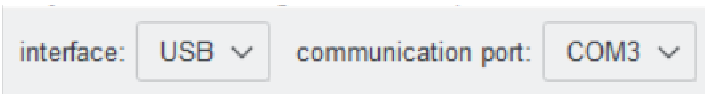

Fig. 4.3.1 IOMod Utility interface and communication port parameters

Fig. 4.3.2 IOMod 8AI USB connection port physical location

5. Parametrization

In this section, the IOMOD 8AI settings configuration is described. IOMOD 8AI configuration is performed via IOMod Utility (the manual can be accessed here). All IOMOD-related settings can be found in the "Iomod settings" tab (Fig. 5.1).

Fig. 5.1 IOMod settings tab

5.1 IOMod settings

In this section are described all the measurement parameters of the device. The first 3 subsections show all the parameters that are specific to each protocol and the forth one is for general parameters that all of them share.

5.1.1 IOMod Modbus measurement parameters

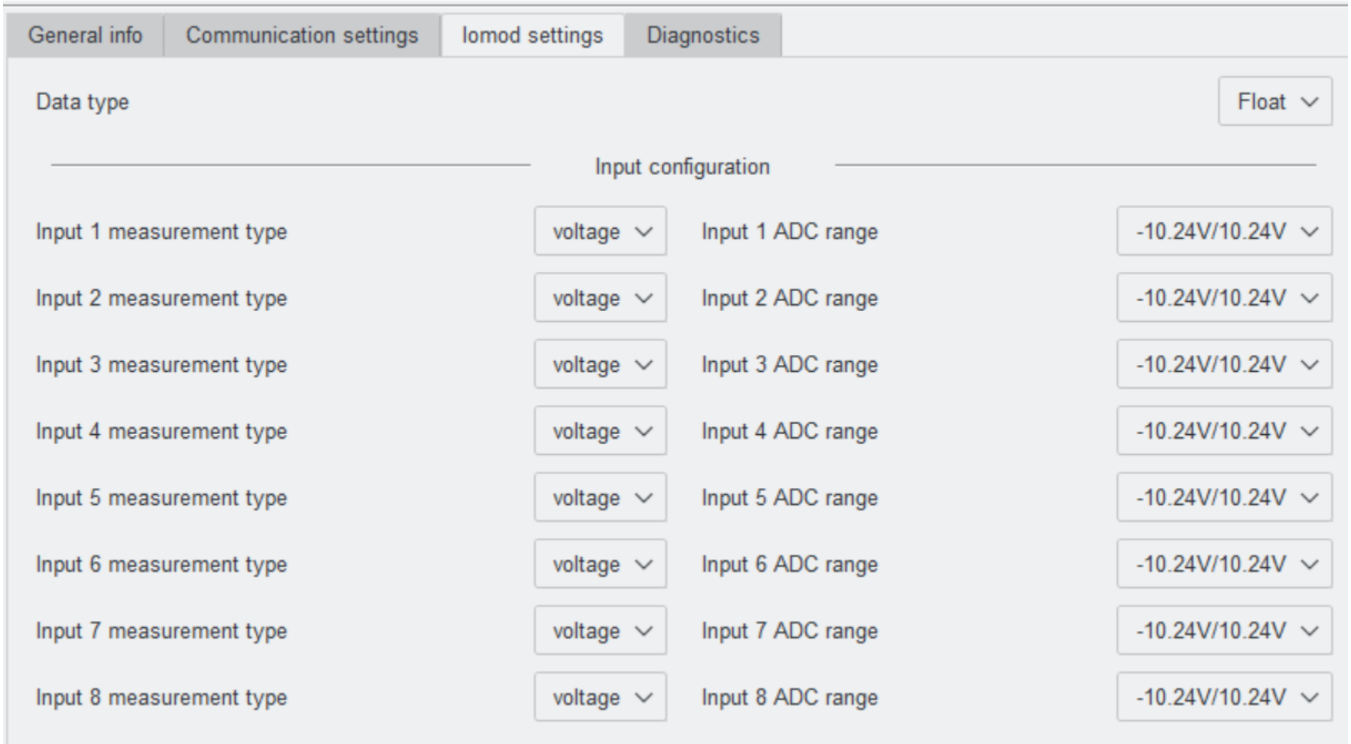

Fig. 5.1.1.1 IOMod settings for **Modbus**, showing data type and input configuration

Table 5.1.1.1 IOMod Modbus parameter ranges and default values

Parameter	Range	Default value
Data type	Float, Unsigned 16	Float

5.1.2 IOMod IEC 60870-5-101 measurement parameters

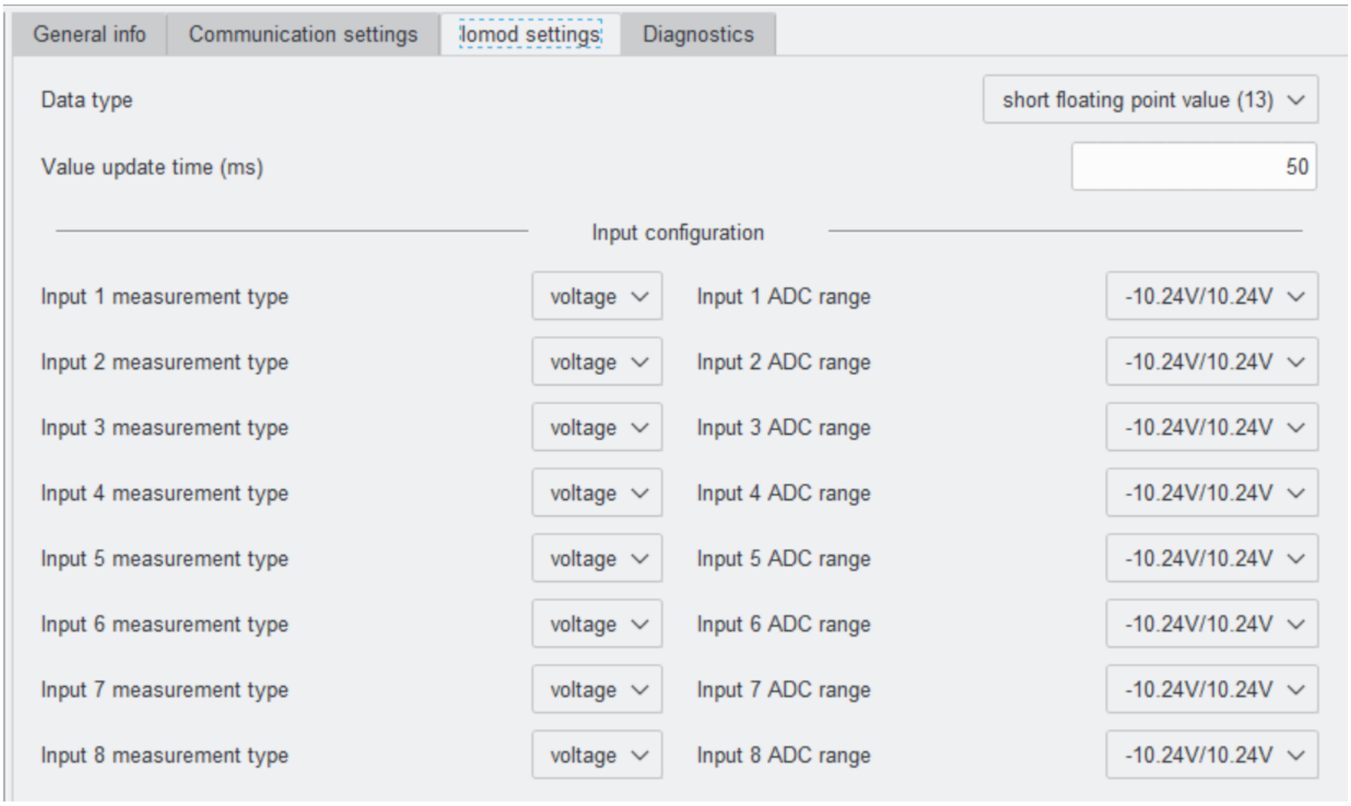

Fig. 5.1.2.1 IOMod settings for **IEC 60870-5-101,** showing data type and input configuration

Table 5.1.2.1 IOMod IEC 60870-5-101 parameter ranges and default values

Parameter	Range	Default value
Data type	scaled value (11), short floating point value (13)	short floating point value (13)
Value update time (ms)	10 - 65535	50

5.1.3 IOMod IEC 60870-5-103 measurement parameters

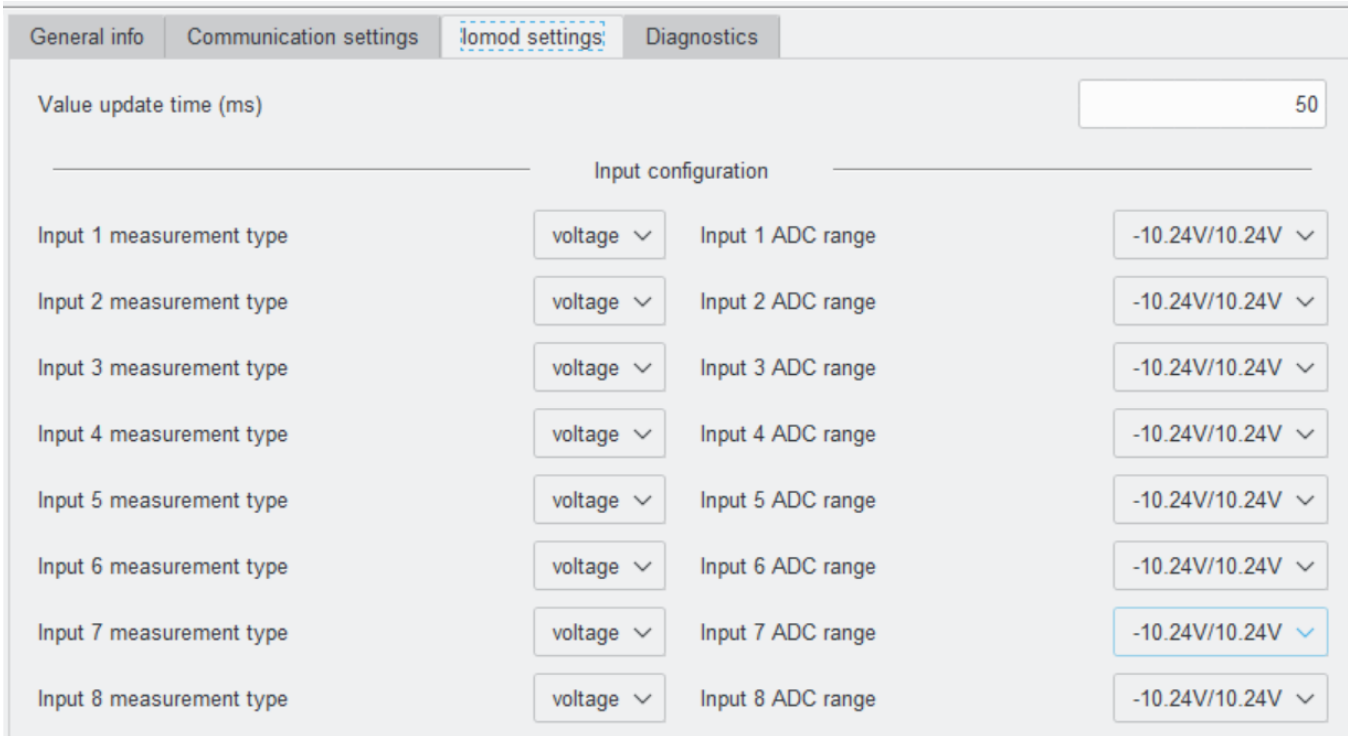

Fig. 5.1.3.1 IOMod settings for **IEC 60870-5-103**, showing data type and input configuration

Table 5.1.3.1 IOMod IEC 60870-5-103 parameter ranges and default values

Parameter	Range	Default value
Value update time (ms)	10 - 65535	50

5.1.4 IOMod general measurement parameters

Table 5.1.4.1 IOMod 8AI parameter ranges and default values

Parameter	Range	Default value
Input [ ] measurement type	voltage, current	voltage
Input [ ] ADC range (voltage)	-10.24V/10.24V, -5.12V/5.12V, -2.56V/2.56V, 0V/10.24V, 0V/5.12V	-10.24V/10.24V
Input [ ] ADC range (current)	-45.5mA/45.5mA, -22.75mA/22.75mA, -11.38mA/11.38mA, 0mA/45.5mA, 0mA/22.75mA	-45.5mA/45.5mA

Value range can be selected to set thresholds on underflow or overflow error statuses. Also, if a scaled integer data type is selected, these limits will be converted to values.

Fig. 5.1.4.1 IOMOD settings sections showing value range

Table 5.1.4.2 IOMOD 8AI parameter ranges and default values

Parameter	Range	Default value
Input [ ] min value limit	float limit	-10
Input [ ] max value limit	float limit	10

Value scaling is where an input value in one range is mapped to a new value in another range. The range for the input is being transformed to another range. Value range and value scaling are related.

$$\text{Mapped Output} = \frac{(\text{Input Value} - \text{Input Min})}{\text{Input Max} - \text{Input Min}} \times (\text{Output Max} - \text{Output Min}) + \text{Output Min}$$

The Input Min and Input Max is the**value range** and Output Min and Output Max is the**value scaling**. For example, if:

- Input Min = -10
- Input Max = +10
- Output Min = 0
- Output Max = 10

This mapping means:

- An **input value of +10** will map to **10** (the upper bound of the mapped range).
- An **input value of 0** will map to **5** (the middle of the range).
- An **input value of -10** will map to **0** (the lower bound of the mapped range).

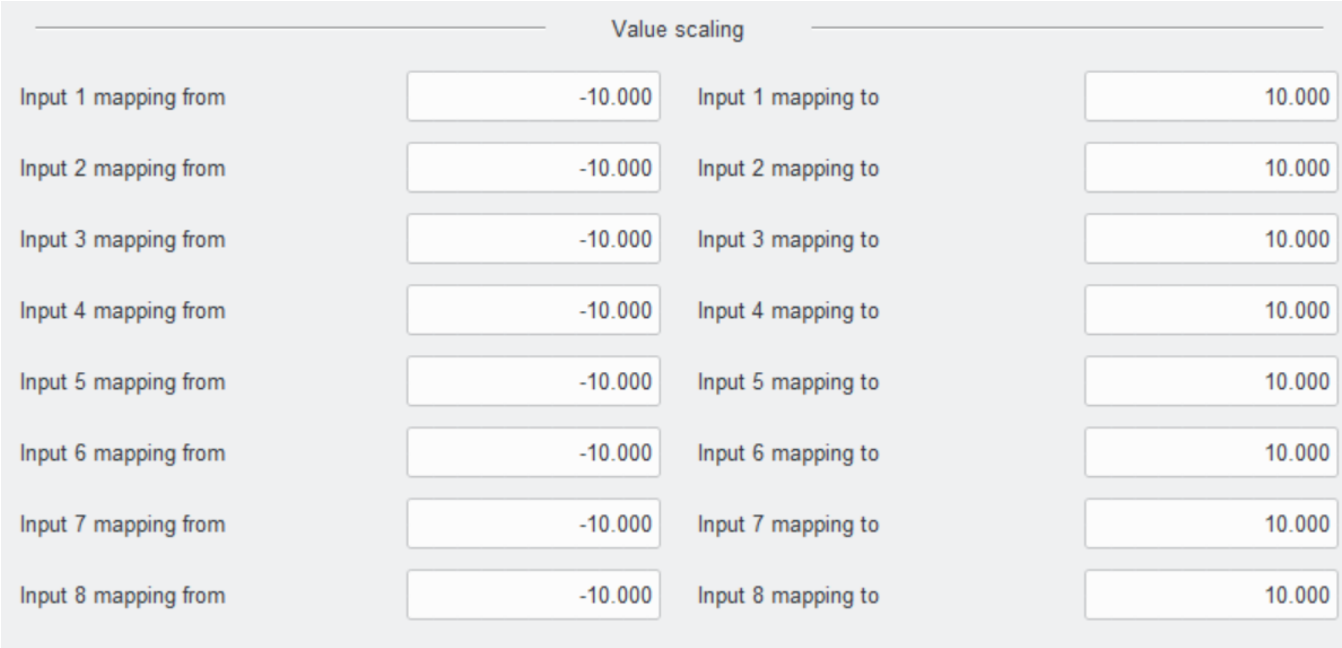

Fig. 5.1.4.2 IOMod settings sections showing value scaling

Table 5.1.4.3 IOMOD 8AI parameter ranges and default values

Parameter	Range	Default value
Input [ ] mapping from	float limit	-10
Input [ ] mapping to	float limit	10

Zero error (%) is the proportion of error (due to a baseline or zero offset) relative to the full-scale range of the signal, expressed as a percentage. For example, IOMOD 8AI will accept a zero error (or offset) of up to 1% (Fig. 5.1.4.3.) of the input signal range before attempting to correct or adjust it.

The **averaging coefficient** is used to control how many previous measurements contribute to the current value when averaging signals over time. This is typically applied to smooth out noise, reduce fluctuations, or improve the accuracy of readings by averaging multiple data points.

ADC Scan Rate is how frequently the ADC samples and converts the input signal to digital values. It's the rate at which the raw data is being captured by the ADC.

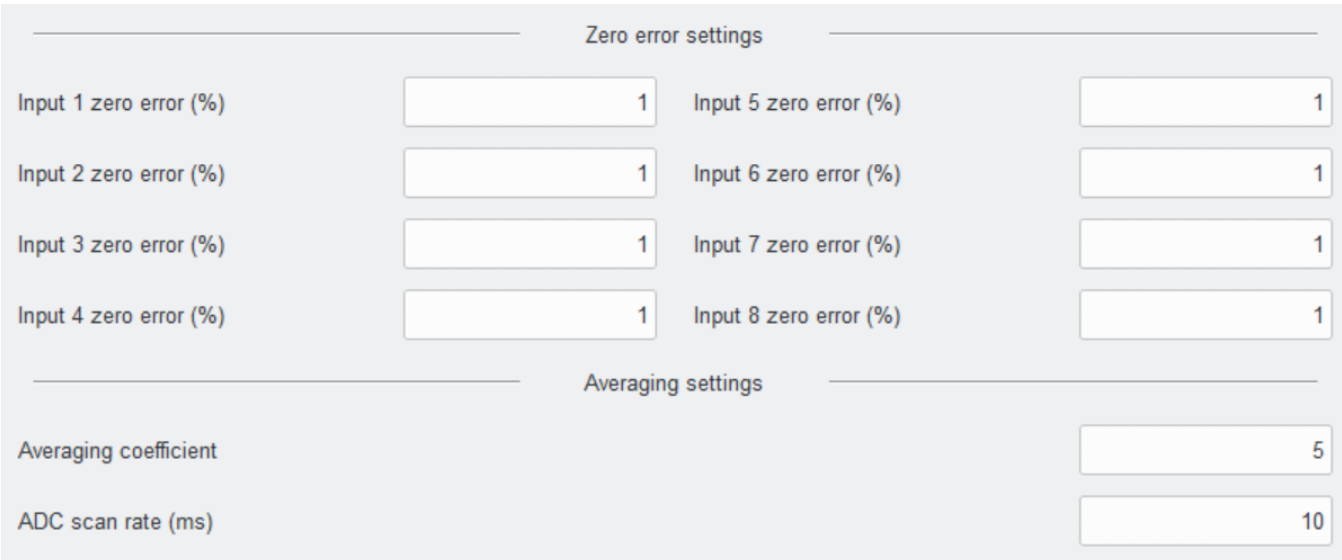

Fig. 5.1.4.3 IOMod settings sections showing zero error and averaging settings

Table 5.1.4.4 IOMOD 8AI parameter ranges and default values

Parameter	Range	Default value
Input [ ] zero error (%)	1 - 100	1
Averaging coefficient	> 0	5
ADC scan rate (ms)	>= 2	10

5.2 Diagnostics

The Utility diagnostics windows allow users to connect to IOMOD directly and observe the values in real-time. The 8AI diagnostics window shows the raw and scaled values.

To turn on real-time monitoring of both Diagnostics sections, the "Connect" button to the left of the "Offline" word designation needs to be pressed. After pressing the "Connect" button the word designation of Diagnostics mode changes to "Online", the black circle starts blinking and the button name changes to "Disconnect" (Fig. 5.2.1).

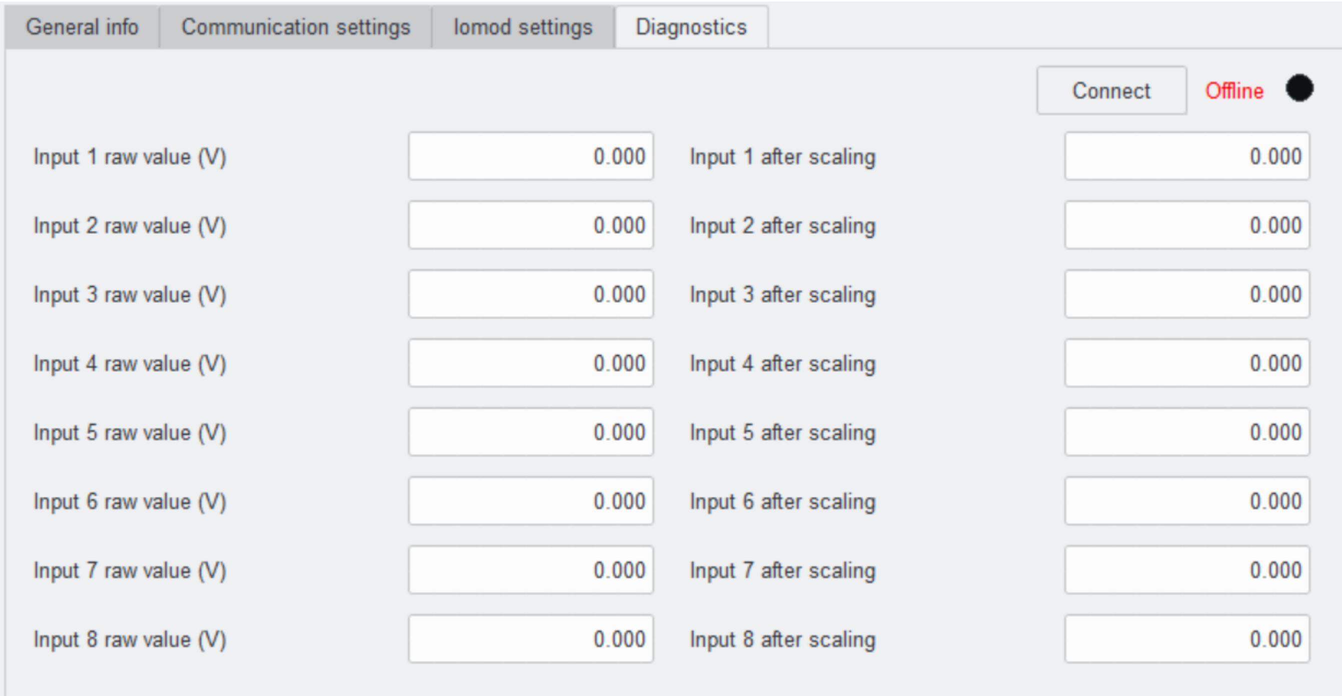

Fig. 5.2.1 IOMod Utility Diagnostics tab in offline mode

6. Communication protocols

The IOMod 8AI supports three communication protocols: **Modbus RTU, IEC 60870-5-101, and IEC 60870-5-103**. These protocols allow a user, via a master device, to read measured data from the IOMod. The desired communication protocol can be selected using the IOMod Utility application (Fig. 6.1) The Utility's interface allows users to connect to IOMOD via USB port and RS485. More information about this tool and its installation can be found on detailed IOMod Utility manual here.

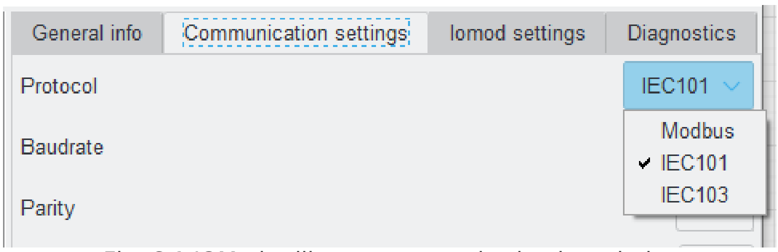

Fig. 6.1 IOMod utility app protocol selection window

Table 6.1 IOMod 8AI protocols default communication settings

Protocol	baudrate	parity	stop bits	wait byte count	slave address	link address size	ASDU size	COT size	IOA size	Input function
Modbus	19200	Even	1	8	1					
IEC 101	19200	Even	1	8	1	1	1	1	2	
IEC 103	19200	Even	1	8	1					253

*Default IOMod 8AI communication protocol is Modbus

6.1 Modbus RTU operational information

Each input can be configured to represent a 16-bit signed integer value or a 32-bit float value. When the float data type is selected, the value will be shown as two registers for one input. When the 16-bit integer data type is selected, a value will be shown as one register. This means addresses of individual input and the maximum number of readable registers can differ according to user configuration.
Configure the device over the IOMod utility. Modbus commands that can be used are shown in the tables below.

02 (0x02) Read Input Status
Used to read analog input overflow and underflow statuses. The first 8 inputs show each input underflow statuses (according to the measurement limits option) and the second 8 inputs show overflow statuses. 0 is the default and 1 means that there's an error (overflow or underflow).

04 (0x04) Read Input Registers
Used to read measurements of at most 8 analog inputs. IOMod 8AI has 8 analog inputs from address 0 to address 15. Different analog inputs can be cast in different data types configured over a USB interface.

Table 6.1.1 Modbus registers for function 2

Discrete Inputs FC02			
Address (Dec)	**Description**	**Data Type**	**Access**
0	Returns underflow status of input 01	BOOLEAN	R
1	Returns underflow status of input 02	BOOLEAN	R
2	Returns underflow status of input 03	BOOLEAN	R
3	Returns underflow status of input 04	BOOLEAN	R
4	Returns underflow status of input 05	BOOLEAN	R
5	Returns underflow status of input 06	BOOLEAN	R
6	Returns underflow status of input 07	BOOLEAN	R
7	Returns underflow status of input 08	BOOLEAN	R
8	Returns overflow status of input 01	BOOLEAN	R
9	Returns overflow status of input 02	BOOLEAN	R

10	Returns overflow status of input 03	BOOLEAN	R
11	Returns overflow status of input 04	BOOLEAN	R
12	Returns overflow status of input 05	BOOLEAN	R
13	Returns overflow status of input 06	BOOLEAN	R
14	Returns overflow status of input 07	BOOLEAN	R
15	Returns overflow status of input 08	BOOLEAN	R

Table 6.1.2 Modbus registers for function 4

Input Register FC04			
Address (Dec)	**Description**	**Data type***	**Access**
0	Input 1 value	UINT16	R
0-1		FLOAT	
2	Input 2 value	UINT16	R
2-3		FLOAT	
4	Input 3 value	UINT16	R
4-5		FLOAT	
6	Input 4 value	UINT16	R
6-7		FLOAT	
8	Input 5 value	UINT16	R
8-9		FLOAT	
10	Input 6 value	UINT16	R
10-11		FLOAT	
12	Input 7 value	UINT16	R
12-13		FLOAT	
14	Input 8 value	UINT16	R
14-15		FLOAT	

*Depends on Data type settings set on IOMod

6.2 IEC 60870-5-101 operational information

Table 6.2.1 IEC 60870-5-101 registers

IOA	Name	TI*
1	input 1 value	11 (M_ME_NB_1) or 13 (M_ME_NC_1)
2	input 2 value	11 (M_ME_NB_1) or 13 (M_ME_NC_1)
3	input 3 value	11 (M_ME_NB_1) or 13 (M_ME_NC_1)
4	input 4 value	11 (M_ME_NB_1) or 13 (M_ME_NC_1)
5	input 5 value	11 (M_ME_NB_1) or 13 (M_ME_NC_1)
6	input 6 value	11 (M_ME_NB_1) or 13 (M_ME_NC_1)
7	input 7 value	11 (M_ME_NB_1) or 13 (M_ME_NC_1)
8	input 8 value	11 (M_ME_NB_1) or 13 (M_ME_NC_1)

*Depends on data type settings set on IOMod: scaled value or short floating point value. Default is short floating value(13)

IOmod uses a standard IEC-60870-5-101 communication scheme. Initiation, control messages, and queries are initiated by the master (controlling station), while the IOMod device (controlled station) only answers these requests. Therefore, the first message should be sent by the master to start/restart communication (ResetOfRemoteLink). This message is answered by IOMod with an acknowledgement (ACK) to enable the master to proceed with sending other messages defined by the IEC-60870-5-101 protocol.

Time synchronization is critical for logging events. To synchronize time, the master sends a Time Sync command C_CS_NA_1 (103) with Cause of Transmission (COT) 6. According to the IEC 60870-5-101 protocol specification, time synchronization can be performed for multiple devices using broadcast messages. A master device sends a broadcast timesync command with a broadcast link address. This ensures consistent time-stamping for event recording and fault detection across the network.

General Interrogation (GI) is initiated by the master sending the General Interrogation command. The command type is C_IC_NA_1 (100) and the Cause of Transmission (COT) has to be 6. The command has to be sent to the correct link address and CASDU, which is the same as the link address by default. If the sent frame is correct the IOMod will respond with a C_IC_NA_1 (103) type command with the COT (cause of transmission) of 7 and the **p/n** bit will be positive (0). Otherwise, it will respond with the same command just that the **p/n** bit will be negative (1). Then the device will begin to send all of its data. After that's done the IOMOD will also send another 100 type command with the COT (cause of transmission) of 10 (ActTerm) meaning the general interrogation is over.

6.3 IEC 60870-5-103 operational information

Table 6.3.1 IEC 60870-5-103 registers

Type	INF	FUN	Description
3 (M_MEI_NA_3)	0	253	input 1 value
3 (M_MEI_NA_3)	1	253	input 2 value
3 (M_MEI_NA_3)	2	253	input 3 value
3 (M_MEI_NA_3)	3	253	input 4 value

3 (M_MEI_NA_3)	4	253	input 5 value
3 (M_MEI_NA_3)	5	253	input 6 value
3 (M_MEI_NA_3)	6	253	input 7 value
3 (M_MEI_NA_3)	7	253	input 8 value

As IOMod 8AI doesn’t have any digital inputs, only analog ones, general interrogation returns nothing. Values of measurements are returned cyclically without any additional request therefore commands sent will be ignored.

Time synchronization is critical for logging events. To synchronize time, the master sends a Time Sync command with function 0 and Cause of Transmission (COT) 8. According to the IEC 60870-5-103 protocol specification, time synchronization can be performed for multiple devices using broadcast messages. For broadcast time synchronization, the master device sends a periodic signal with a time stamp to synchronize the system time of slave devices. If synchronization fails, devices default to their local system time until they successfully resynchronize.