INTELART

Robust in Automation

I5H1000 V1.0

HMC (HMI+PLC) Device

Technical Manual

www.intelart.ir

08/2021

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

DANGER
Indicates that death or severe personal injury will result if proper precautions are not taken.

WARNING
Indicates that death or severe personal injury may result if proper precautions are not taken.

NOTICE
Indicates that property damage can result if proper precautions are not taken.

Qualified personnel

The product/system described in this documentation may be operated only by personnel qualified for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions.

Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems

Contents

1 Technical Specifications

1.1 Hardware Parameters

The following table specifies the hardware information of the module.

Table 1 Hardware parameters

Performance	Processor	Arm Cortex M7, 480 MHz
	Bit operation	6 µs
	Word operation	6 µs
	Single Floating-point Operation	6 µs
	Double Floating-point Operation	6 µs
	CPU startup modes	Cold start, Warm start
	Configurable startup mode	Yes
Memory	Load memory	16 MB
	Application memory	128 KB
	Retentive memory	4 KB
	M area memory	32 KB
	I area memory	16 KB
	Q area memory	16 KB
	G area memory	32 KB
	Permanent memory	16 KB
	SD card memory	Up to 32 GB
I/O Expansion	Supported modules	8, IM2XX Series
	Max digital I/O count	128
	I/O integrated in CPU	No
	Emergency stop trigger	Yes
Device Elements	IEC-61131 data types	Yes
	Arrays	Yes
	Structures	Yes
	User custom data type	Yes
	Number of Tags	2048

	Number of Organization Blocks (OBs)	• Cyclic Programs: 1 • Periodic Interrupts: 4 • Startup Programs: 1 • Stop Interrupts: 1 • Emergency Stop Interrupts: 1 • Time Of Day Interrupts: 1
	Number of functions (FCs)	32
	Number of function blocks (FBs)	16
	Number of screens	16
	Number of elements per screen	64
	Basic drawings	Yes
	Image	Yes
	Button	Yes
	Textbox	Yes
	ComboBox	Yes
	CheckBox	Yes
	Switch	Yes
	LightIndicator	Yes
	LevelIndicator	Yes
	Slider	Yes
	MultiImage	Yes
	Datagrid	Yes
	Trend	Yes
	HMI element event handlers	Yes
Engineering	Configuration / programming Software	Intelart Studio
	Programming languages	LAD, FBD
Display	Design of display	TFT
	Number of colors	65000
	MTBF backlighting (at 25 °c)	80000 h
	Backlight dimmable	Yes, 0-100 %
	Screen diagonal	10 in
	Resolution WxH(pixels)	1024 x 600
	Display size WxH	222 x 126 mm
	Design as touch screen	Yes
Acoustics	Buzzer	Yes
	Speaker	No
Interfaces	Number of RS-485 interfaces	1 (Isolated) 115200 bps
	Number of RS-232 interfaces	1, 115200 bps
	Number of CAN interfaces	1, 1 Mbps
	Number of Ethernet interfaces	1, 10/100 Mbps
	Programming port	Ethernet
Isolated Power	Voltage range	18 – 30 V
	Current range	500 – 900 mA
Dimensions	Front panel WxH	263 x 173 mm
	Installation cutout WxHxD	255 x 165 x 48 mm
	Storage temperature	-15 to 75 ˚C

Ambient Conditions	Operating temperature	0 to 55 ˚C
	Relative humidity	Max 90 %, No Condensation
Miscellaneous	Weight	Approx. 900 g
	Led indications	Yes. RUN: Green, STOP: Orange, FAULT: Red
	Front panel protection	IP 65
	Rear protection	IP 20
	Backup battery lifetime	5 Years typical

2 Configurations

2.1 Digital Inputs

There is one digital input on the device as an emergency stop trigger. The EmergencyStop OB will execute (if present) when a 24V signal applied to this input.

2.2 Digital Outputs

The device has no digital output.

2.3 Memory Storages

This device uses several memory storages for different purposes. Each storage has a specific duty and should be considered by user.

Load memory

The load memory is a non-volatile memory for code blocks, user data, prebuild objects, media objects and hardware configuration. When these objects are downloaded to the CPU they are first saved into load memory. This memory is located on the device itself.

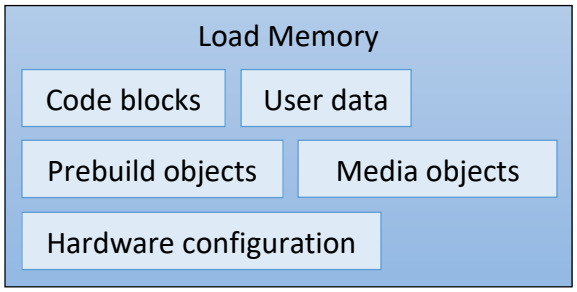


Figure 1 Load memory illustration

Application memory

The application memory is a volatile memory that contains the code and data structures. The application memory is integrated into the CPU and cannot be extended or moved. This memory also contains runtime-relevant parts of the program code.

Retentive memory

Retentive memory is a non-volatile memory for saving a limited quantity of data in the event of power failure. The tags that have been defined as retain are saved in retentive memory. These data is retained beyond a power-off or power failure. All other program tags are lost under these conditions and are set to their default values upon the operating mode transitions POWER ON to startup, and STOP to startup.

The content of retentive memory is deleted by the following actions:

- Memory reset because of a Cold Start
- Backup battery removal

NOTE
The low voltage in the backup battery causes the device to run in Cold Start mode.

M area memory

The **M** area memory is a volatile memory that contains all tags defined in external tag tables with an address starting by '%M' notation. This area mostly uses for communication protocols between devices such as Modbus, CAN etc. or some cases that a memory overlapping procedure is needed. Also this memory is usable for general programming.

External TagTable*

	Name	DataType	Address	Comment
1	Static			
2	Static_1	Bool	%M0.0	
3	Static_2	Byte	%MB1	
4	Static_3	Word	%MW2	
5	Static_4	Int	%MW4	
6	Static_5	DWord	%MD8	
7	Static_6	ULInt	%M12.0	
8	Static_7	Real	%MD20	
9	Static_8	LReal	%M24.0	
10	Static_9	TON	%M32.0	
11	PT	Time	%MD32	
12	ET	Time	%MD36	
13	IN	Bool	%M40.0	
14	Q	Bool	%M40.1	
15	<Add New Item>			

Figure 2 Defining tags in M area in an external tag table

I and Q area memory

The **I** and **Q** area memory is a volatile memory that contains all tags defined in external tag tables with an address starting by '%I' and '%Q' notation respectively. This area contains the values of all input/output areas of connected modules or the CPU itself. Addressing of I and Q area tags is a bit different from M area tags. All I and Q area tags address starts by a number that indicates the physical address of that area.

%I(Q) | Physical address | . | Tag address |

Example: the second digital input in the first module address will be %I0.0.1 that indicates:

first module → second bit in first input byte

NOTE
In cases where the device has input area the CPU index will be 0 and modules index will be started by 1.

After the device is configured, in programming software > Plant Explorer > Device Configuration window all installed modules will be accessible.

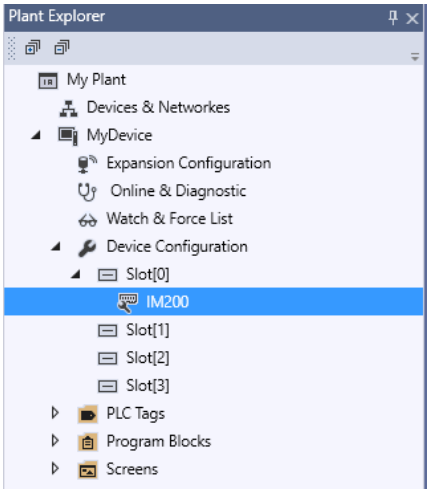


Figure 3 an installed module view in Plant Explorer window (The images belongs to I5H500 device)

Each module has a list that contains all I and Q tags with their physical address. By double click on the selected module the tags description list will be opened. There is a wizard in the list that can create all needed tags with their absolute address in an external tag table.

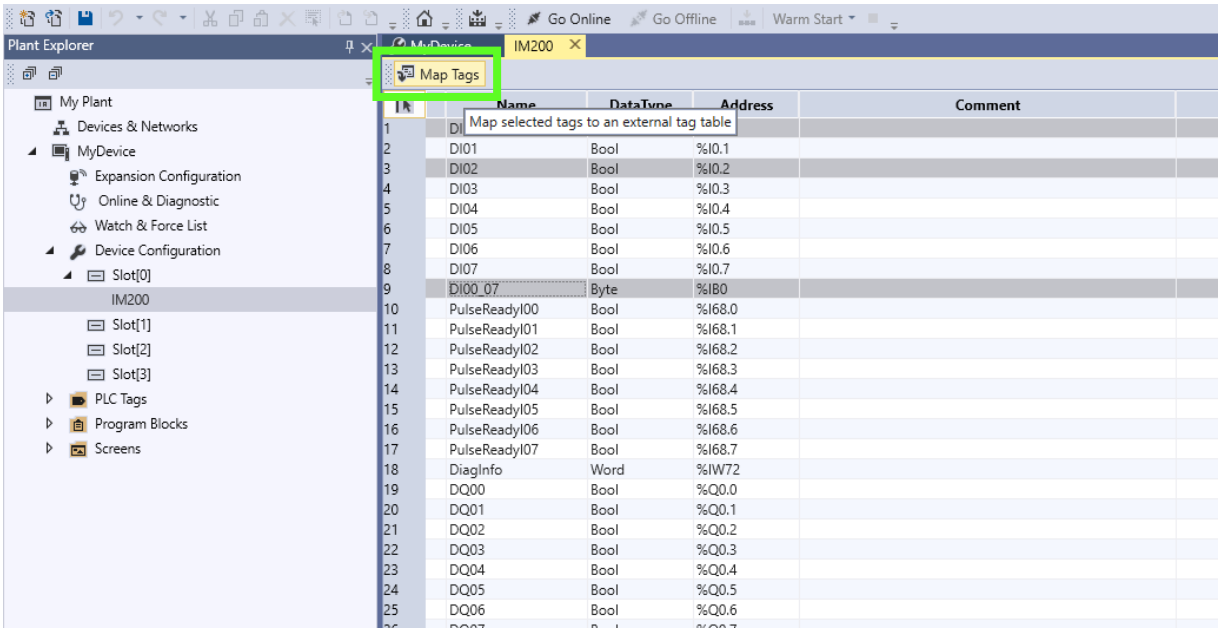


Figure 4 A typical module tags list (The images belongs to I5H500 device)

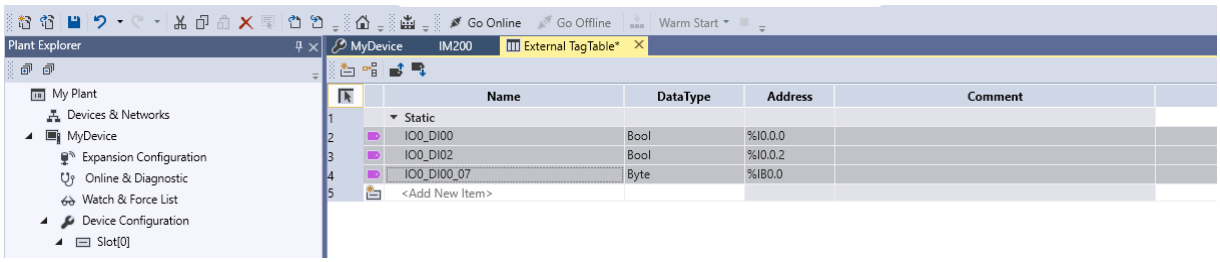


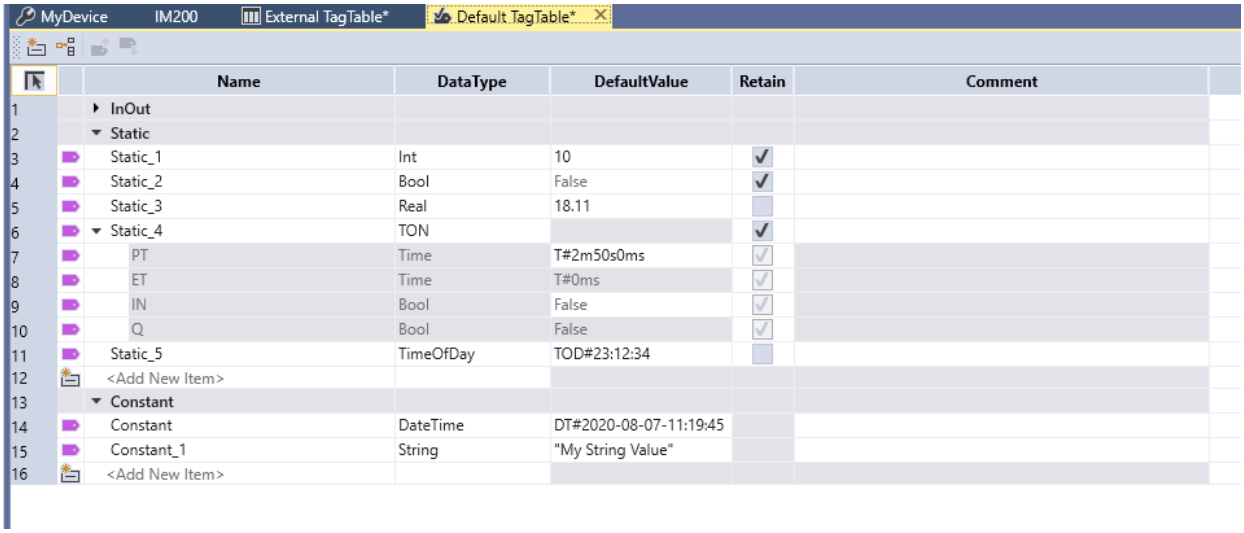
Figure 5 Created I area tags that refering the first physical module

G area memory

The **G** area memory is a volatile memory that contains tags with different features from M area memory. These tags define in a global tag table (such as Default Tag Table) and can start with a user specified default value. There is no need to assign an address for a tag that defined in this area. The compiler calculates the correct address in compile time for G area and retained tag storages. Any tag of any data type in a global tag table can be marked as **Retain**. The retain tags value will be stored in retentive storage.

NOTE
Any change in retain storage arrangement such as add/remove a retain tag or changing default value of a retain tag will be caused a **Cold Start** at the next startup of device after downloading user program to the CPU.

Constant tags store in **G** area and their values never change after downloading program to the CPU.



	Name	DataType	DefaultValue	Retain	Comment
1	InOut				
2	Static				
3	Static_1	Int	10	✓	
4	Static_2	Bool	False	✓	
5	Static_3	Real	18.11		
6	Static_4	TON		✓	
7	PT	Time	T#2m50s0ms	✓	
8	ET	Time	T#0ms	✓	
9	IN	Bool	False	✓	
10	Q	Bool	False	✓	
11	Static_5	TimeOfDay	TOD#23:12:34		
12	<Add New Item>				
13	Constant				
14	Constant	DateTime	DT#2020-08-07-11:19:45		
15	Constant_1	String	"My String Value"		
16	<Add New Item>				

Figure 6 a typical global tag table

NOTE
All defined tags in **M**, **I** and **Q** areas will start by IEC-61131 default values on a CPU start until the user program or CPU change those tag values.

Permanent memory

The permanent memory is a storage space for using in applications that need to keep the value of tags permanently and not be erased even on a Cold Start condition. Reading from or writing to this memory is different because of its EEPROM nature. There is an instruction in order to read from or write to this memory. You can write any type of data on this memory and read them back again.

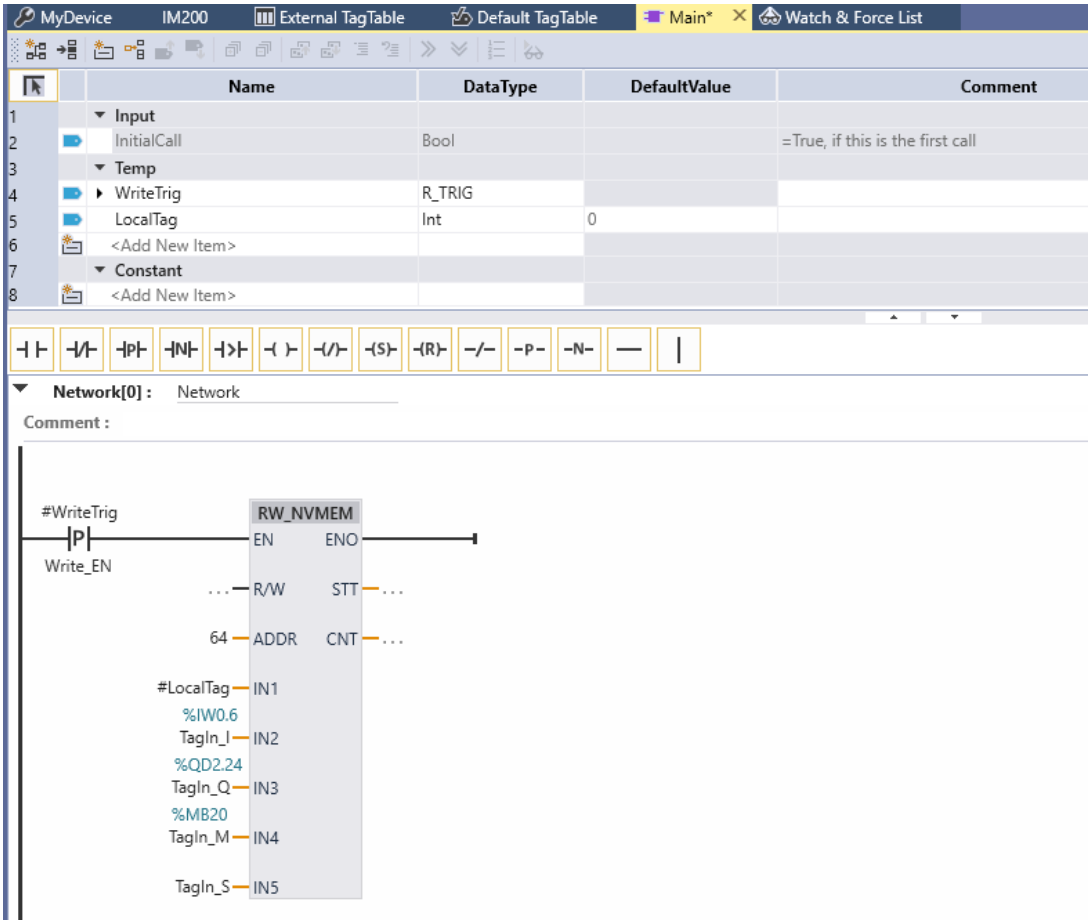


Figure 7 Reading/writing on permanent memory

WARNING
Writing on this memory cells is guaranteed until one million times. Use this memory only for saving configurations, important data or any type of data that does not need to be write frequently. For example, if the scan time of the CPU is 20 ms and a value was saved once per scan, the EEPROM would last a minimum of 20,000 seconds, which is less than 6 hours. On the other hand, if a value were saved once an hour, the EEPROM would last a minimum of 114 years. Typically, you should perform save operations at the occurrence of specific events that occur rather infrequently.

SD card memory

Your controller program can store process values in Data Logs using the Data Logging instructions. The Data Logs are saved in csv format in the directory of the SD memory card. The Data Logging instructions are used in your program to create, open and write Data Logs and to close Data Logs. You decide which tags are to be logged by incrementing or decrementing the inputs of the instruction. You can execute “F_LOG” instruction to add a row of new values to the csv file.

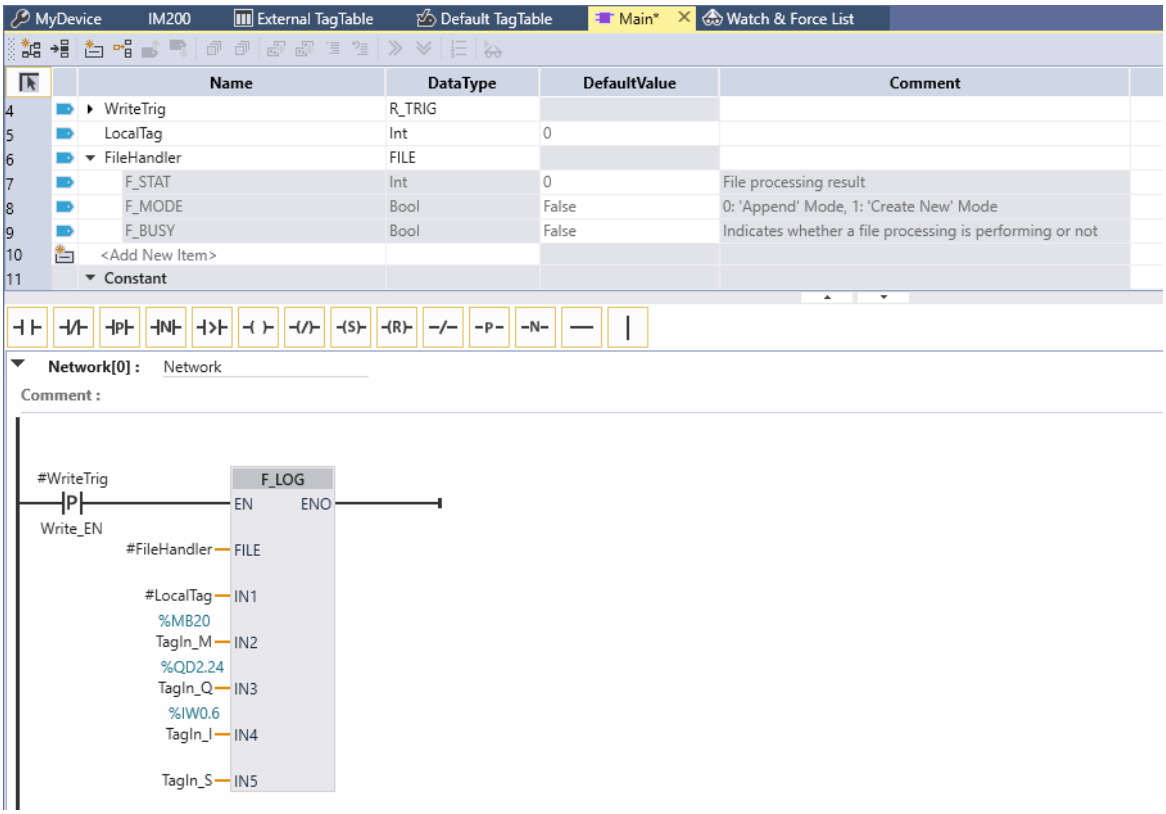


Figure 8 Data logging on SD card

2.4 Communications

2.4.1 RS-485

This device provides a single RS-485 port for connection to the network. The port is electrically isolated from bus. The maximum length of a network segment is determined by two factors:

Isolation (using an RS-485 repeater) and baud rate.

Isolation is required when you connect devices at different ground potentials. Different ground potentials can exist when grounds are physically separated by a long distance. Even over short distances, load currents of heavy machinery can cause a difference in ground potential.

Supported baud rates for the RS-485 port can vary from 1200 to 115200 b/s.

2.4.2 RS-232

This device provides a single RS-232 port for general purposes. Supported baud rates for the RS-232 port can vary from 1200 to 115200 b/s.

2.4.3 CAN

This device provides a single CAN port for connection to the network. You use this port for sending and receiving CAN messages on a multi master CAN network. Supported baud rates for the CAN port is a maximum of 1000 kb/s for a 40 m length CAN bus.

2.4.4 Ethernet

The I5H500 can be remotely configured, programmed and diagnosed via Ethernet using Intelart Studio. Also the device can communicate with another devices via Ethernet. Ethernet can be implemented to offer a wide range of application specific uses, such as switching, high-speed SCADA, fast connects, and redundant networks.

3 Diagnostic and Wiring

There is 1 LED indicating the status of the device. The following table explains the states of the LED.

Table 2 Combination of "POWER" and "MAINT" LEDs

	Indicating	Solution
Off	Power missing or hardware failure.	• Check the main power supply • Verify that the power connector is installed correctly
On	The device is configured and is in RUN mode.	--
On	The device is in stop mode	--
On	The device has an error (communication error, configuration error, runtime error etc.)	• Verify that all communication and hardware configuration is identical in the defined values in Intelart Studio. • Check the user program if it generates a runtime error

The following block diagram shows you information about wiring of the device.

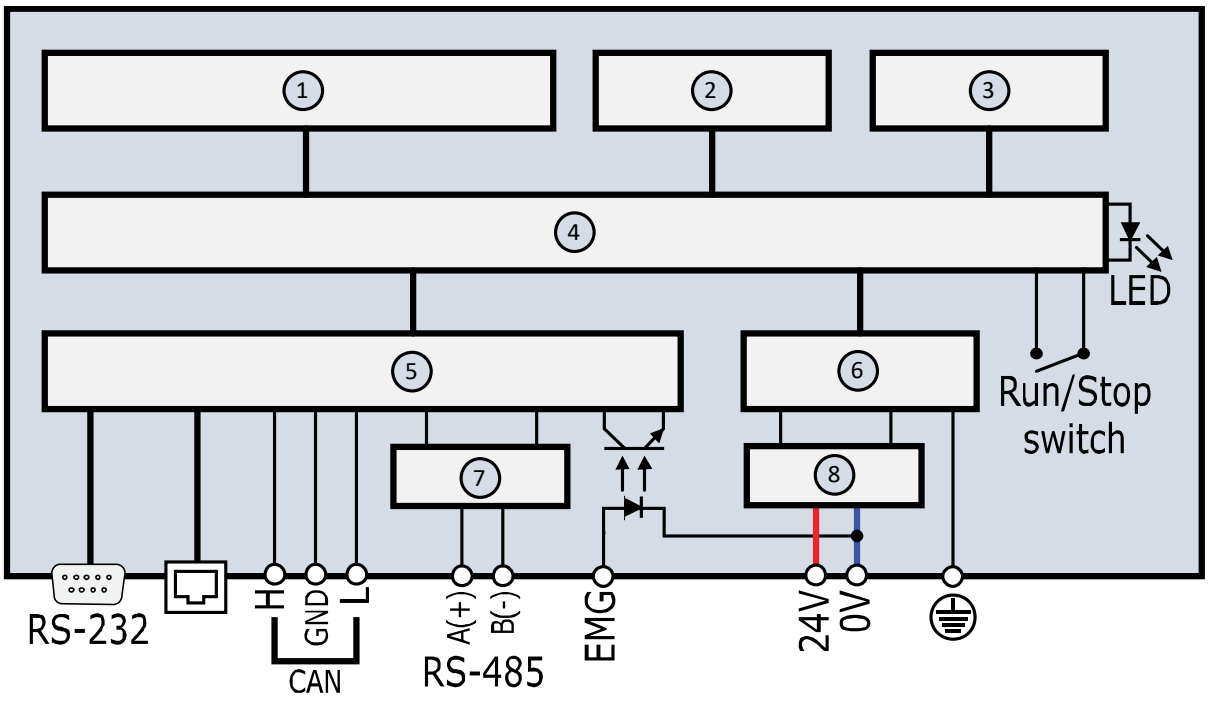


Figure 9 Wiring diagram and terminal assignments

1	I/O module slots	5	Communication unit
2	SD card driver	6	Power unit
3	LCD & touch panel	7	RS-485 magnetic isolator
4	CPU	8	Power isolator

4 Dimensional drawing

The dimensions of the module are available in this section. For install the module and its main device follow the below dimensional drawing.