Modbus/DeviceNet Gateway
MD-210
User Manual

REV 1.2

April，2009

Sibotech Automation Co., Ltd

Technical Support: +86-021-5102 8348

E-mail: support@sibotech.net

Table of Contents

1 About This Document

1.1 General

This document describes every parameter of the gateway MD-210 and provides using methods and some announcements that help users use the gateway. Please read this document carefully before using the gateway.

For further information, documentation etc., please visit the SiboTech website: http://www.sibotech.net/En/

1.2 Important user information

The data and examples in this manual can not be copied without authorization. Sibotech maybe upgrades the product without notifying users.

SiboTech is the registered trade mark of SiboTech Automation Co., Ltd.

The product has many applications. The users must make sure that all operations and results are in accordance with the safety of relevant field, and the safety includes laws, rules, codes and standards.

1.3Terms

DeviceNet: DeviceNet protocol, in accordance with GB/T18858.1,GB/T18858.3 and DeviceNet Protocol Release2.0.

RS485/RS232: Hardware specifications of serial interface

Modbus: MODICOM Modbus Protocol PI-MBUS-300 Rev.J

RS-25: RS232 /RS485 converter

2 About the Gateway

2.1 Function

The gateway can connects multiple devices with Modbus (RS485/RS232) interface to DeviceNet network, act as a master at the side of Modbus network, and a slave at the side of DeviceNet network. It supports RS485 and RS232 at the serial interface. Comparing with other previous products, MD-210 adds the debugging function, and the function makes the application so convenient in industrial field. RS485 interface is used for communication while RS232 interface is used to debug, and RS232 interface is used for communication while RS485 interface is used to debug. RS232 interface is the special interface of configuration mode.

MD-210 works through the data mapping between networks, mapping Modbus parameters to DeviceNet I/O data.

2.2 Feature

- Act as a DeviceNet slave: Group 2 Only Slave.
- Support all the baudrate which accords with the DeviceNet protocol, and support intercepting baud rate automatically function.
- Act as a Modbus master, and support the 1, 2, 3, 4, 5, 6, 15, 16 function codes.
- The range of input-voltage is 8~30V, and the standard working voltage is 24V DC.
- Free configuration software GT-123.
- Support the debugging without PLC.

2.3 Technical specification

[1]Communication rate

- DeviceNet interface supports: 125kbit/s, 250kbit/s, and 500kbit/s;
- The default parameters of Modbus interface are 192000bps, 8bit, no parity, 1 stop bit;

  The range of Modbus baud rate: 300, 600, 1200, 2400, 9600, 19200, 38400, 57600, 115200bps

[2]DeviceNet topology:

- Trunk lines

Thick cable and thin cable both can be used to build the trunk lines. When the thick cable and thin cable are mixed to build the trunk lines, the longest cable length can be calculated through the following formulas:

$L_{thick}+5*L_{thin}=500m$ 125kbit/s

$L_{thick}+2.5*L_{thin}=250m$ 250kbit/s

$L_{thick}+L_{thin}=100m$ 500kbit/s

Here L_{thick} is the length of thick cable, and L_{thin} is the length of thin cable.

- Drop lines

The length of drop lines is the distance from tap of the trunk lines to the transceiver of every device, and it should be less than 6m. The length of drop lines is related to baud rate, and the longest length with different baud rate shows in Table1.

Table1-The length of drop lines

Baud Rate	Length of Cable
125kbit/s	156m
250kbit/s	78m
500kbit/s	39m

[3] Working mode: DeviceNet interface only support: Group 2 Only Slave.

[4] Working environment:

- Relative Humidity: 5% to 95%(No condensation)
- Temperature: -20℃ to 60℃
- Pollution level: 3

[5] EMC testing standard compliant

[6] Power: 24VDC (11V~30V), maximum 80mA (24V)

Within the module, it uses DC to DC conversion, and the conversion efficiency is not less than 70%

[7] Mechanical size: 125mm (H)*110mm (W)*45mm (D)

2.4 Attention

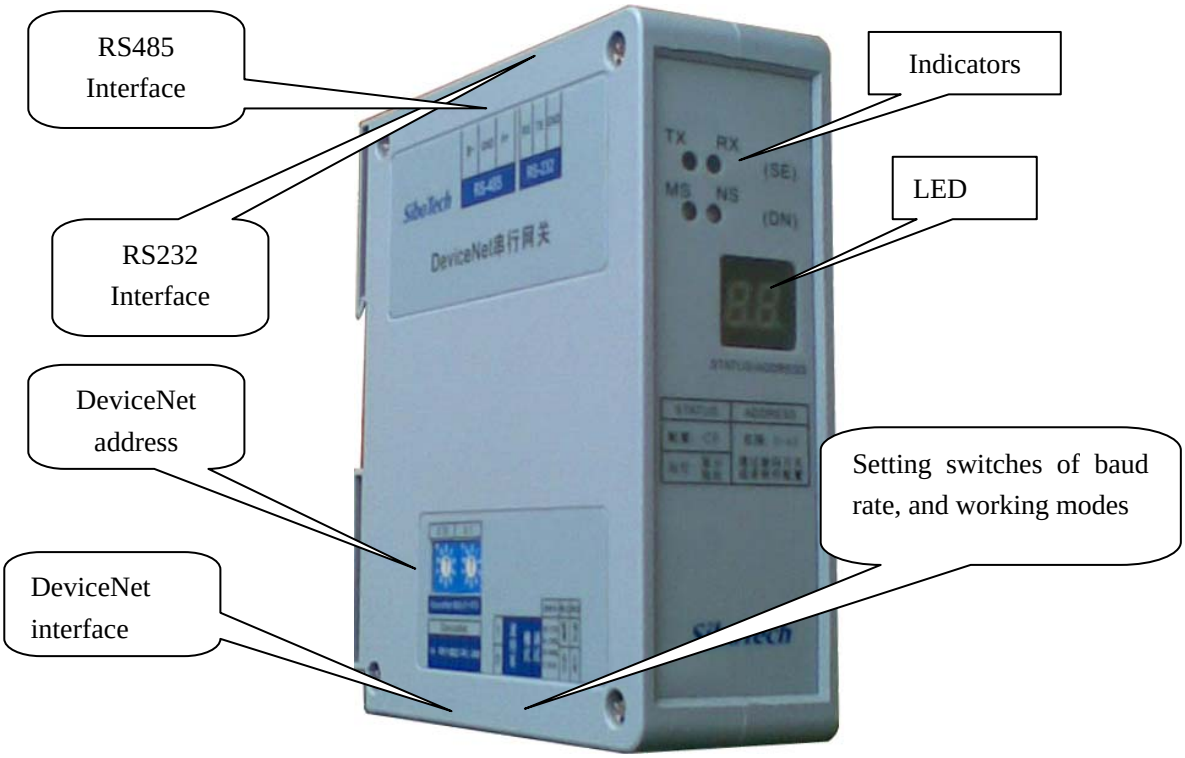
- To prevent stress, prevent module panel damage;
- To prevent bump, module may damage internal components;
- Power supply voltage control in the prospectus, within the scope of the requirements to burn module;
- To prevent water, water module will affect the normal work;
- Please check the wiring, before any wrong or short circuit.

2.5 Related products

There are other products: MD-21U, MD-21T, PM-160 and so on.

If you want to get more information about these products, please visit the SiboTech website: www.sibotech.net/en , or call the technical support number: +86-021-5102 8348.

3 External View



Note: This picture is only for reference, the product appearance should take the material object as a standard.

3.1 Indicators

The explanation of indicators show as Table 2, Table3 and Table4:

Table 2-Indicators of Module Status (MS)

Indicators	Description
Off	No power supply or broken indicators
Always Green	Work normally
Green blinking	Not correctly configured
Red blinking	Recoverable faults, Modbus communication faults(such as not find the slave station)
Always Red	Unrecoverable faults

Red-Green blinking	Self-testing is ongoing

Table 3-Indicators of DeviceNet network Status (NS)

Indicators	Description
Off	The repetitive MAC ID detection is not successful or no power supply
Green blinking	The devices are online but there are not connections established
Always Green	The devices are online and there are connections established
Red blinking	One or more I/O connections have been timeout
Always Red	The device detects unrecoverable faults and can not communicate, such as there is repetitive DeviceNet address on net.

Table 4-Indicators of Serial interface Status (TX, RX)

Indicators	Status	Description
RX (Green)	Blinking	Serial port is receiving data
	Off	Serial port is not receiving data
TX (Red)	Blinking	Serial port is transmitting data
	Off	Serial port is not transmitting data

3.2 Status setting switches and LED

3.2.1 Status setting switches

Status setting switches have three functions:

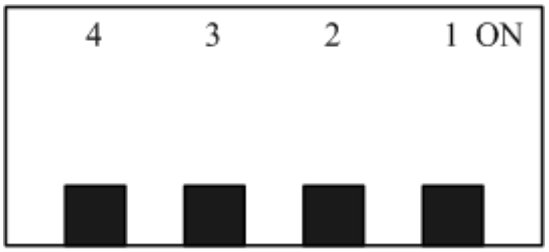
1) Modify DeviceNet baud rate

2) Set working mode: Configuration and run are optional. At the status of configuration, the LED shows “CF”.

3) Set debugging: Debugging and normal are optional.MD-210 has the function of debugging, and it provides users with easy way for debugging Modbus network data communications. At the status of debugging, the LED shows “db”.

Note: The priority of configuration mode is higher than debugging. When configuring the gateway, the debug switch should be dial to normal. When debugging the gateway, mode switch should be dial to run.

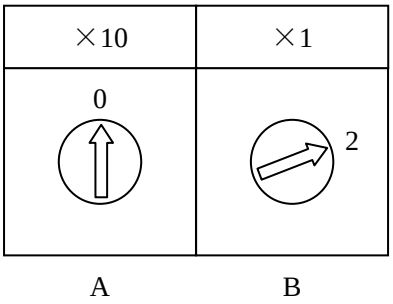
Status setting switches are below the product:



	4 3	2	1
	Baud rate	Mode	Debug
1	00:125K 01:250K	Configure	Debug
0	10:500K 11: Automatic	Run	Normal

Note: If you reset the status switches, you should restart MD-210 to make the settings take effect.

3.2.2 Setting switches of DeviceNet address



According to the above, the DeviceNet address is calculated as follow:

DeviceNet address = （ A × 10 ） + （ B × 1 ）

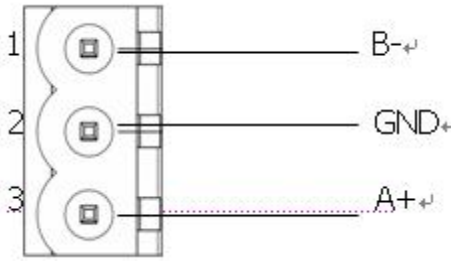
3.2.3 LED

The main contents of LED include: current baud rate (only show at startup), current DeviceNet address (show at running)

3.3 Communication interface

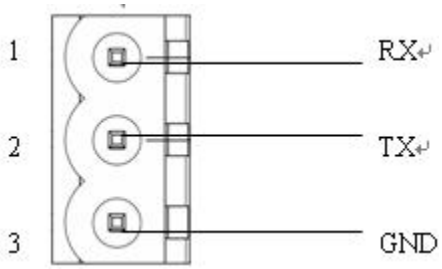
3.3.1 Modbus interface

Modbus interface use open 3 pin pluggable terminal, users could accord to the panel instruction to wire:



Pin	Function
1	B-，RS485
2	GND
3	A+，RS485

RS-485 interface

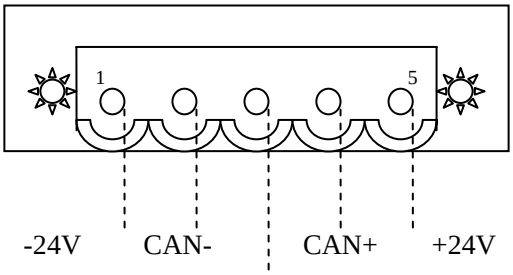


Pin	Function
1	RX, connect to RX of user’s device RS232
2	TX, connect to TX of user’s device RS232
3	GND, connect to GND of user’s device RS232

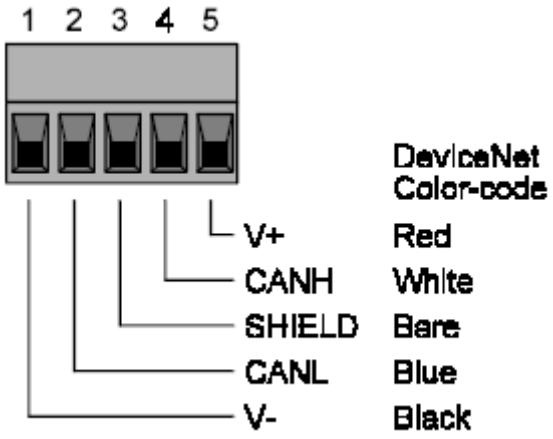
RS-232 interface

3.3.2 DeviceNet interface

5-pin connector:



Open 5-pin connector at the side of DeviceNet:



Pin	Wiring
1	GND(24V)
2	CAN-
3	shielding
4	CAN+
5	+24V

4 Use Method

4.1 Quick start

1 Setting DeviceNet address manually

Before connecting DeviceNet master, sets DeviceNet address by moving code switch of the gateway's side. The range of DeviceNet effective address is 0 to 63. The calculation method of DeviceNet address is shown at chapter 3.2.2 of the document. To supply power to this module, the LED displays blinking "bT", and then shows "12" or "25" or "50", respectively indicate "125K", "250K", "500K".Finally, it shows the DeviceNet address you have set. When baud rate is set to automatically baudrate intercept state, if there is no other CAN node on the network to send data, it shows "bT".

Note: When use DIP switch to modify the DeviceNet address, re-power the gateway to take the new address effect.

2 In the configuration mode, set DeviceNet address, Modbus parameters and commands through gateway configuration software GT-123.

The method of MD-210 entering configuration mode: Dial the mode setting switch to "1", and power on the module, the module can enter the configuration mode. Then connect with GT-123 for configuration. After configuration, restart the gateway to run normally.

Note: RS232 interface is specialized configuration interface, in configuration mode, please pay attention to the wiring exactly.

3 Setting DeviceNet baud rate manually

Set DeviceNet baud rate manually though the baud rate setting switches below the gateway, baud rate setting switches have four combinations, respectively indicate 125K, 250K, 500K, automatically intercepting baud rate. See chapter 3.2.1 of the document.

4 Connect Modbus interface and DeviceNet interface accurately and check up the wiring.

5 Power on, the module enter the running mode. Before powering on, you should dial the mode setting switch to

running status, debugging setting switch to normal.

4.2 Hardware wiring

1 In accordance with the third chapter about DeviceNet interface instructions, wire every pin of 5-pin terminal exactly, no power on at this time.

2 In accordance with the third chapter about Modbus interface, properly wiring.

3 Check out all wirings.

4 Power on the module, the module enter running mode. Pay attention to settings of mode switch and debugging switch.

4.3 Software configuration

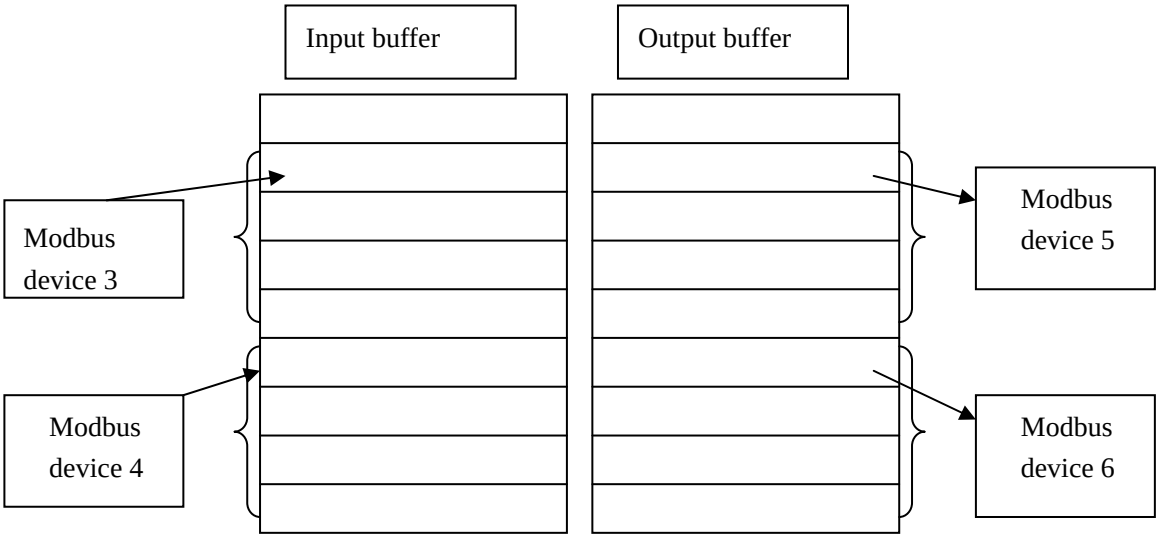
User connect MD-210 gateway to PC. Set gateway's Modbus parameters, commands, and DeviceNet parameters through gateway configuration software-GT-123. Before powering on the gateway, you should dial the mode setting switch to "1", make the gateway be at the status of configuration.

See GT-123 Software configuration instruction.

4.4 Run

4.4.1 Data exchange mode

The data exchange between MD-210 Modbus and DeviceNet is established by "mapping". MD-210 has two data buffers, one is DeviceNet network input buffer, and the other is DeviceNet network output buffer. Modbus read commands read the data and then write the data into input buffer for DeviceNet network reading. Modbus write commands get data from network output buffer, and then output the data to Modbus devices by writing commands.



Users can configure 48 commands at most, and can use a Modbus command to read a serial Modbus registers.

- **Note 1:** If there is something wrong with Modbus communication, DeviceNet I/O data can not be effectively collected, and the data got through I/O scanning is zero.
- **Note 2:** When DeviceNet interface of MD-210 receives network output-data, Modbus interface will send writing-commands. That is to say that after DeviceNet master station sending data, Modbus interface of MD-210 will send writing-commands, and transmit data to Modbus slave devices. If AB's PLC in programming mode, there will be no the network output-data.
- **Note 3:** During the configuration of MD-210, when polling mode of output commands is set to "Change of State", the function of the local data exchange can not be used.

  Local data exchange: Configurate writing-commands to input-area (0000~3FF0)

4.4.2 Terminal Resistor

DeviceNet network requires a 120ohm terminal resistor respectively at the two farthest terminals of the network. Modbus requires terminal resistors too. MD-210 has a terminal resistor at the side of Modbus interface, users only need add a 120ohm terminal resistor at the other side of the Modbus network.

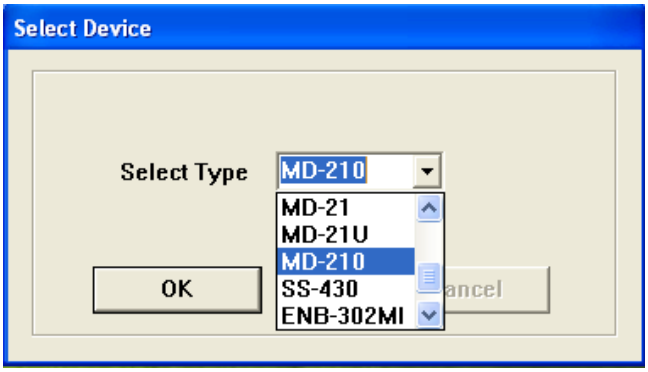
5 Configuration software instructions

5.1 Points for attention before configuration

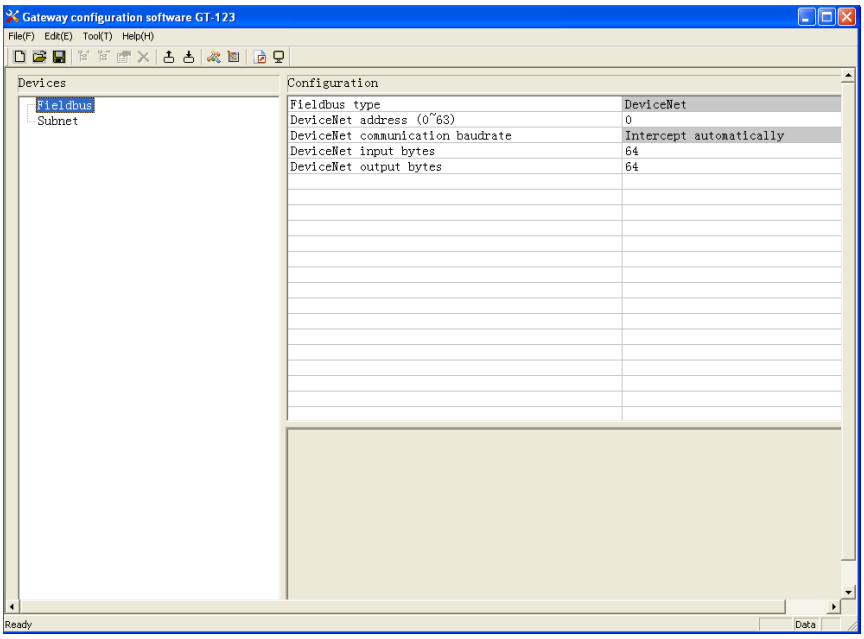
GT-123 is software which bases on Windows platform, and used to configure a variety of fieldbus gateways, including MD-210, MD-21U, PM-120, SS-430, PM-160 and so on. Users could set Modbus and fieldbus parameters and commands.

This document introduces the using method of MD-210.

After installed, double-click on the icon, you can see:



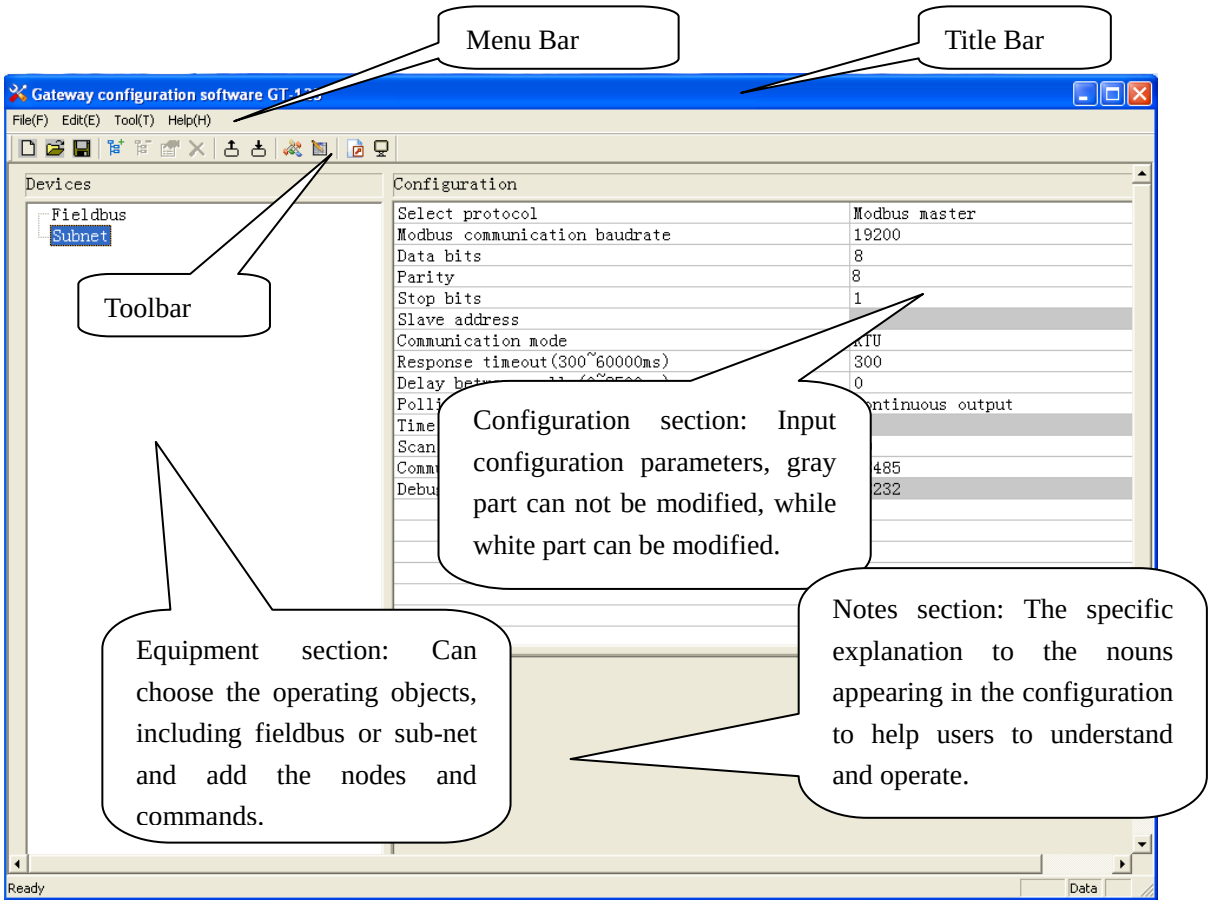
Choose “MD-210” and then enter Configuration interface.



5.2 User Interface

The interface of GT-123 includes: Title bar, Menu bar, Toolbar, Status bar, Equipment section, Configuration section and Notes section.

Remark: In this software, all the gray parts can't be modified.



Toolbar:

Toolbar is shown as below:

From left to right : New, Open, Save, Add node, Delete node, Add command, Delete command, Upload Configuration information, download Configuration information, Collision detection, Export XLS and Debug.

New: Create a new configuration project

Open: Open a configuration project

Save: Save the current configuration

Add node: Add a Modbus node

Delete node: Delete a Modbus node

Add command: Add a Modbus command

Delete command: Delete a Modbus command

Upload configuration information: Read the configuration information from the module and display them

Download configuration information: Download the configuration information from GT-123 to the module.

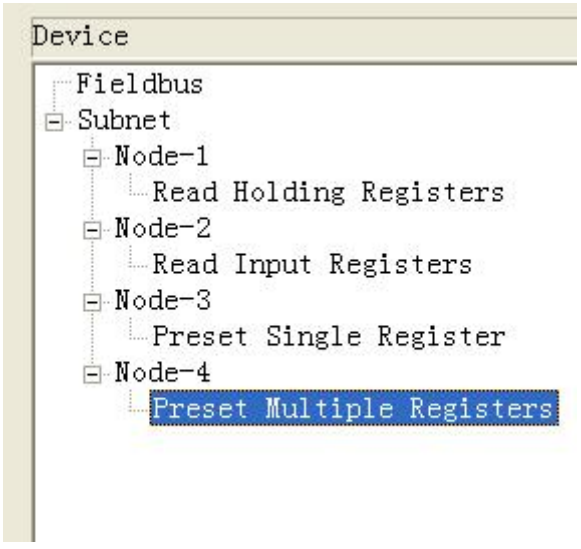
Collision detection: Detect whether there is conflict in memory data buffer of the gateway.

Export XLS：Output the current configuration to local hard disk and save it as .xls file

Debug: Debug the Modbus network communications, and define the network fault.

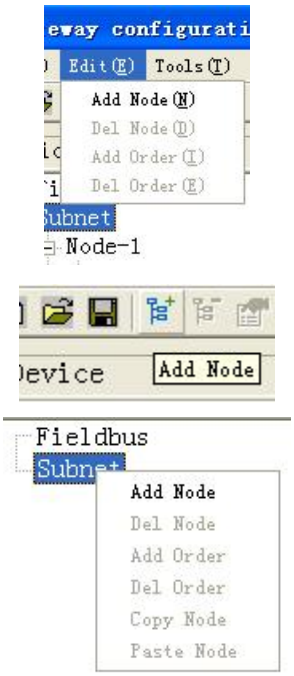
5.3 Device View Operation

5.3.1 Device View Interface



5.3.2 Device View Operation

The device view supports three types of operation: Edit Menu, Edit Toolbar and Right-click to edit Menu.



5.3.3 Kinds of device view operation

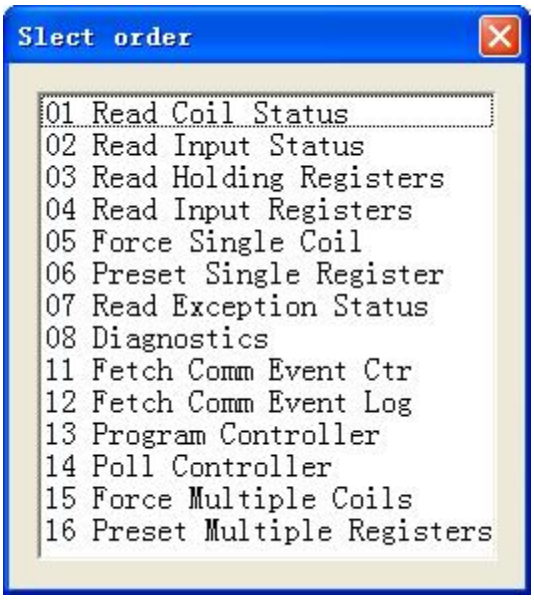
1) Add node operation: Right-click on subnet or existed nodes, and then you can add a new node named "new node" under subnet.

2) Delete node operation: Right-click on the node to be deleted, and then you can delete the node. The node and its all commands will be deleted.

3) Add command operation: Right-click on the node, and then you can add a command for the node. The commands dialog box is shown as follow：

Currently, MD-210 supports the commands: 01, 02, 03, 04, 05, 06, 15 and 16 commands

Select the command: Double click on the command



4) Delete command operation: Right-click on the command and then you can delete command.

5) Rename nodes: Left-click on the node to be renamed, and then the edit status will be shown and you can rename it.

5.4 Operation of configuration view

5.4.1 Interface of Fieldbus Configuration View

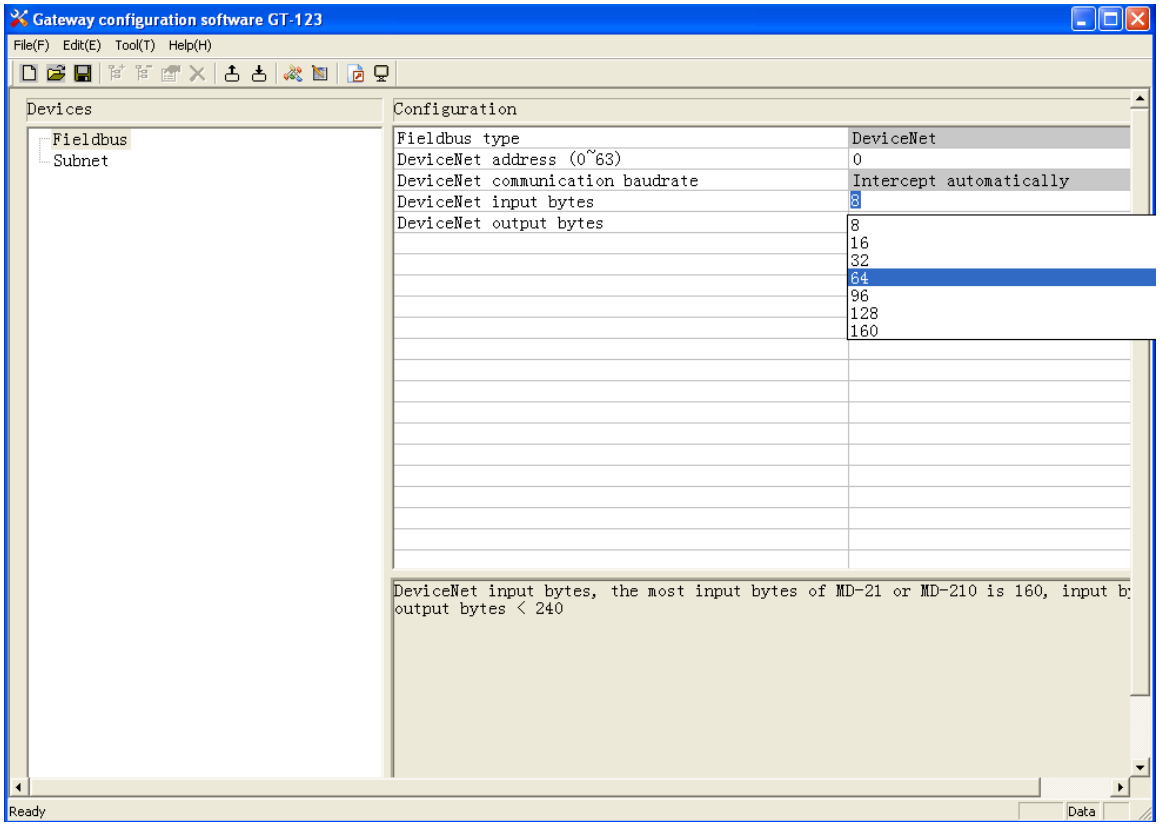
In the devices view, click on “fieldbus”, the configuration view is shown as follows:

Configurable items include: number of input-byte, output-byte and DeviceNet address.

Number of DeviceNet input-byte: There are 8, 16, 32, 64, 96, 128 and 160 to be selected.

Number of DeviceNet output-byte: There are 8, 16, 32, 64, 96 and 112 to be selected.

Note: The sum of input-byte and output-byte must be less than 240. (Expect 240). If the number of input-byte is 160, the number of output-byte is at most 64. If the number of output-byte is 112, the number of input-byte is at most 96. Or DeviceNet will not be connected.

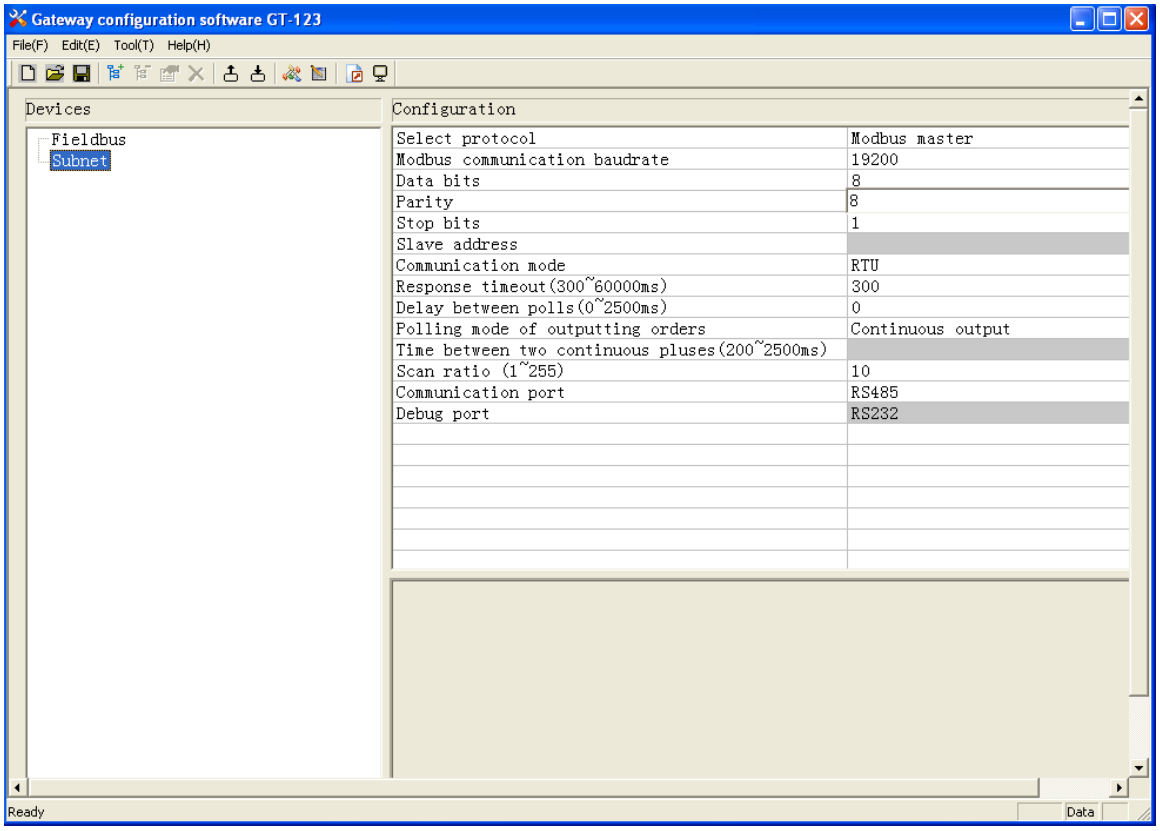


5.4.2 Interface of subnet configuration view

The protocol type is Modbus Master, and configurable parameters are shown as follows:

Modbus communication baud rate, Data bits, Parity check mode, Stop bit, Communication mode, Response timeout, Delay between polls, Polling mode of outputting commands, Time between two continuous pluses (the polling mode of outputting commands is “pulse output”), Scanning ratio.

Interface of configuration view is shown as follow:



Modbus communication baudrate: There are 300, 600, 1200, 2400, 9600, 19200, 38400, 57600 and 115200bps to be selected.

Data bits: 8 bits

Parity check mode: There are none, odd, even, mark and space to be selected.

Stop bits: There are 1 and 2 to be selected.

Transmission mode: There are RTU and ASCII to be selected.

Response timeout: After the Modbus master sending commands, the time waiting for response from the slaves, the range is 300~60000ms.

Delay between polls: After a command of Modbus having been sent and having received correct response, the time before next command being sent, the range is: 0 ~ 2500ms.

Polling mode of outputting command:

Modbus writing commands (output command) has 3 kinds of outputting modes: Continuous-output, Disable-0utput, Change-of–state-output

Continuous-output: The polling mode of outputting command is he same with Modbus read commands, and output commands according to the scanning ratio.

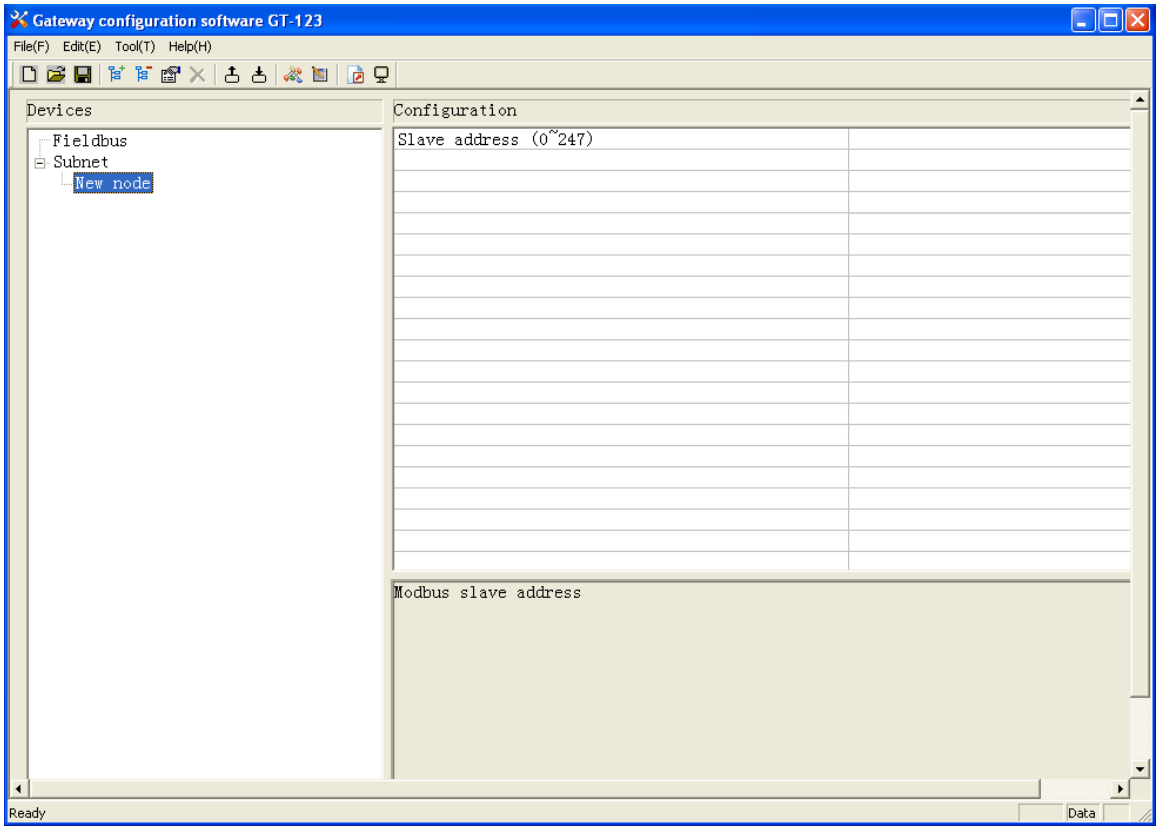
Disable-0utput: Prohibit outputting Modbus write commands.

Change of state output: When the output data has been changed, gateway output the write commands. The gateway stop outputting write commands after receiving correct response.

Scan ratio: Ratio of slow-scan and quick-scan. If the value is 10, it means that if the quick-scan command is send 10 times, slow-scan command is send 1 time.

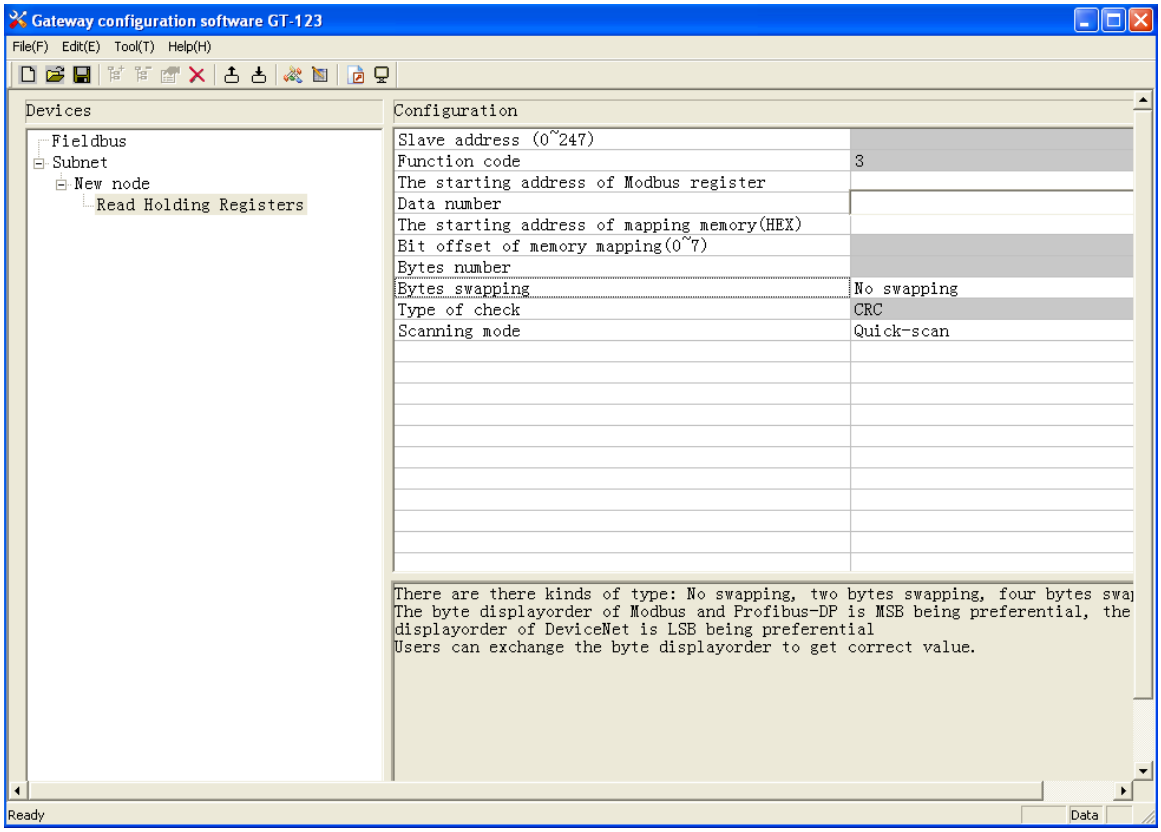
5.4.3 Interface of node configuration view

When the subset protocol is “Modbus master”, in the interface of device view, left-click on a node, the configuration interface is shown as follow, and you can set slave address in the interface:



5.4.4 Interface of order configuration view

In the interface of device view, left-click on a command, the configuration interface is shown as follow:



Starting address: The starting address of register or switching value or loop and so on of Modbus slave and the range is 0~65535.

Number of Data: The number of register or switching value or loop of Modbus slave.

Starting address of mapping memory (Hexadecimal): The starting address of data in memory buffer of the module

The address range of data mapping in the module memory:

Read command: 0x0002 ~ 0x009F

Write command: 0x4000 ~ 0x406F

When write commands are used as local data exchange, they also can use: 0x0000 ~ 0x006F

Bit offset of memory mapping (0 ~ 7): For the bit operation commands, the position of start-bit in a byte, and the range is: 0 ~ 7.

Byte exchange: There are three kinds of types:"No exchange",”Double bytes exchange”, and “Four bytes exchange”. Modbus byte sequence is that the Most Significant Byte (MSB) has the highest priority. DeviceNet byte sequence is that the Least Significant Byte (LSB) has the highest priority. For example, if a Modbus register value is 0x1234, the DeviceNet value is 0x3412 when not using “double-bytes exchange“. Users should exchange

the sequence of bytes so as to get right value. Usually, selecting “double-bytes exchange” is OK. If users use two continuous Modbus registers to express a four-byte value, they can use “four-byte exchange”, maybe it will achieve.

Scanning mode: There are two kinds of scanning mode: quick-scan and slow-scan. It is fit for user’ requests about quick-scan or slow-scan of different orders. Slow-scan is equal to quick-scan being multiplied by scan ratio. (Configure it in the interface of subnet configuration interface)

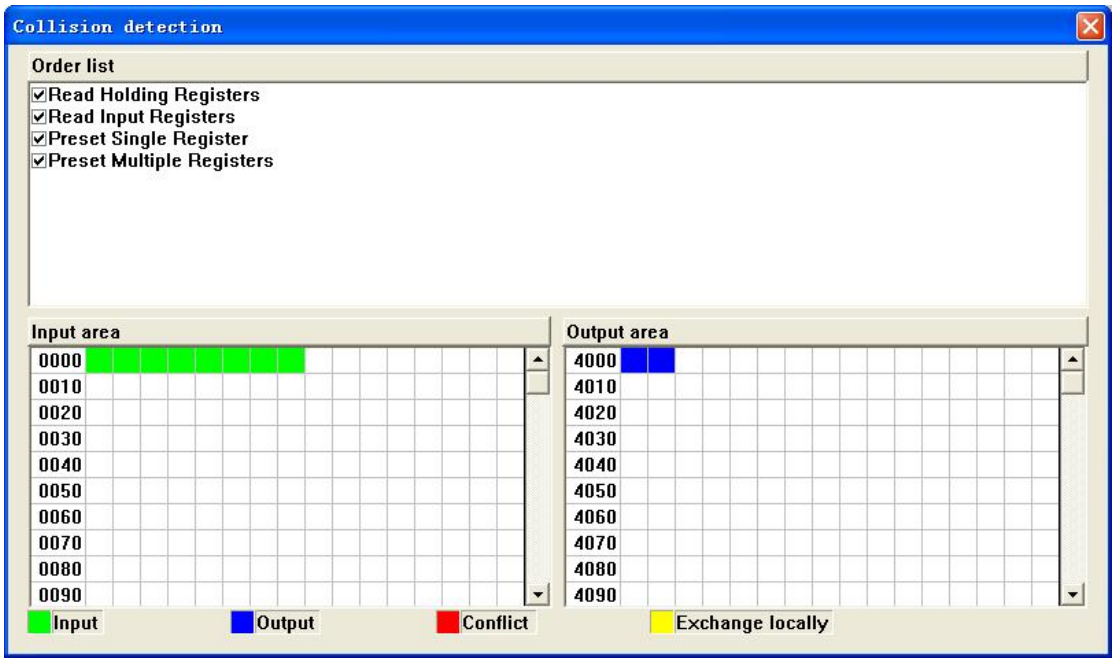
5.4.5 Notes View

Notes view displays the explanation of configuration items. For example the notes that show how to configurate the starting address of memory mapping is shown as follow:

```
The address range of data mapping in the module memory
MD21(MD-210): Read order: 0x0002~0x009F
 Write order: 0x4000~0x406F
 Input-mapping address 0 show the status of Modbus order, if the Nth order has no
response, the byte is N
  Input-mapping address 1 show the status of Modbus node, if the Mth node has no
response, the byte is M
  When write order is used exchanging locally, it also can use: 0x0000~0x009F
PM120(PM160)range: Read order: 0x0002~0x006F
 Write order: 0x4000~0x406F
 When write order is used exchanging locally, it also can use: 0x0000~0x006F
SS430range: Read order: 0x0002~0x0BFF
 Write order: 0x4000~0x4BFF
 When write order is used exchanging locally, it also can use: 0x0000~0x0BFF
```

5.5 Collision detection

For the detection of whether there is collision of "the starting address of memory mapping" or not. If there is collision then you can modify it quickly. The interface is shown as follow:



5.5.1 Operation of command list

All the configuration commands can be shown at the command list. Each select box before command is used for checking the memory-mapping location of the command. Click on the command can select the check box, and in the memory-mapping area it can show the corresponding share of spatial location. Click on the command again will remove the selected box and it doesn't show the mapping location. The function can be used to conflict detect ion of memory-mapping area.

5.5.2 Operation of memory mapping area

Memory mapping area is divided into two parts: input area and output area.

Input-mapping address: 0x0002 ~ 0x009F;

Output-mapping address: 0x4000 ~ 0x406F.

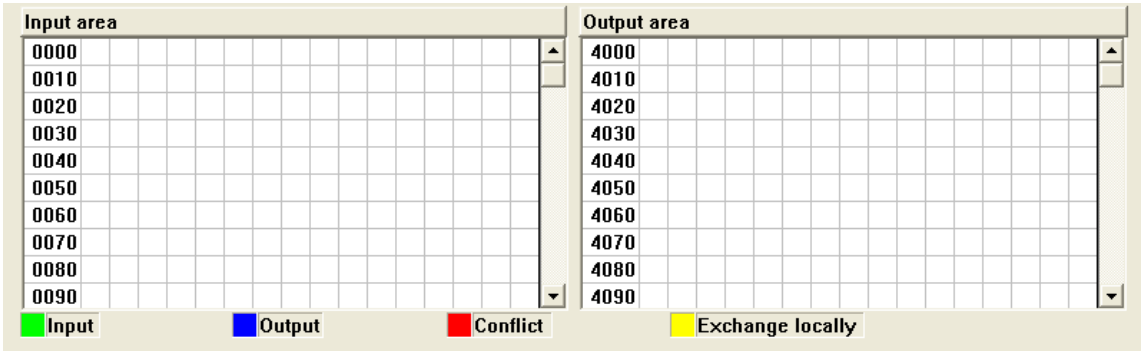
Each box represents a byte address.

Green: Read command show in the area; no conflict;

Yellow: Write command show in the input-mapping area; no conflict;

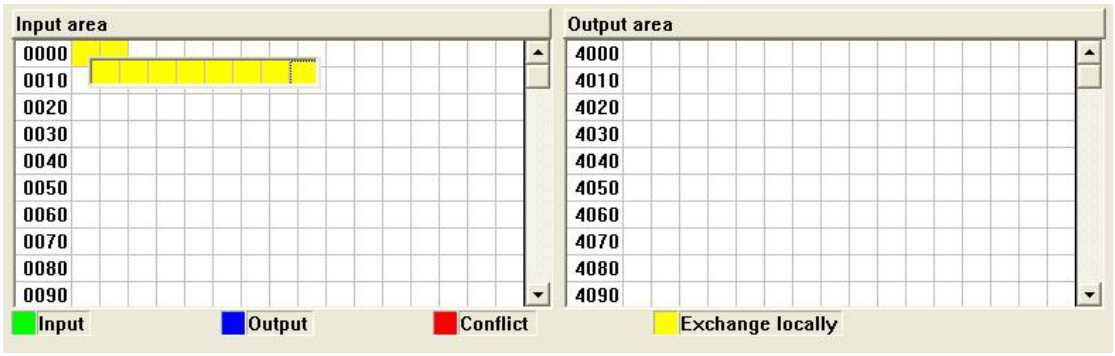
Blue: Write command show in the output-mapping area; no conflict;

Red: In output area or input area, different commands occupy the same byte address, the byte is shown as red.



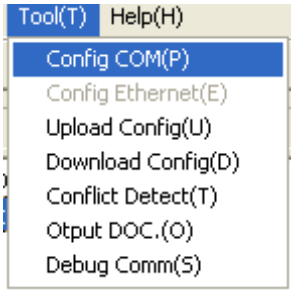
For bit operation commands, the meanings of all colors are also applicable.

Click on the input or output grid, whether the grid is occupied or not is shown as follows:



5.6 Hardware communication

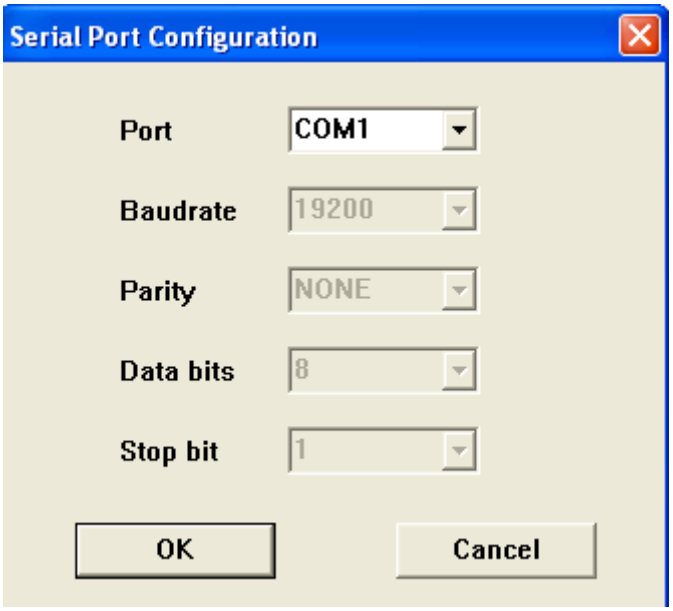
Hardware communications' menu items are shown as follow:



5.6.1 Serial Configuration

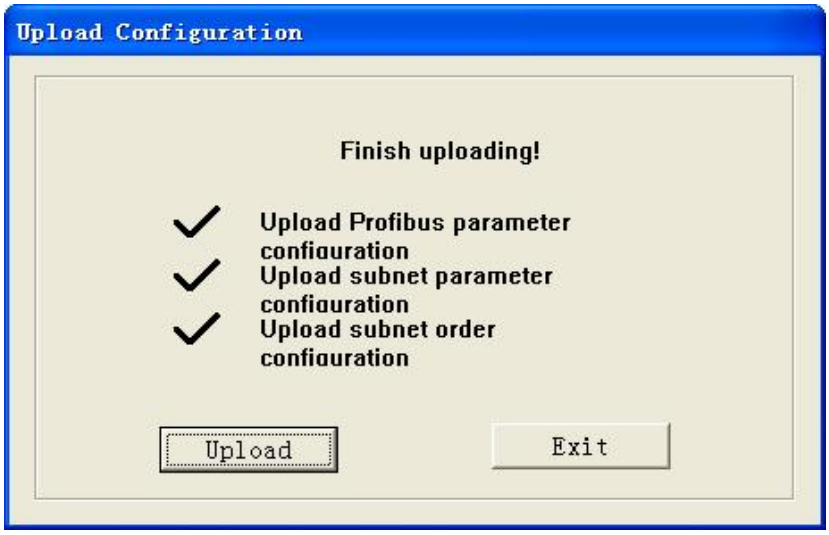
The software automatically scan the available serial port of system, and the available serial can be shown in serial list. After modifying the item, pressing "OK" to save your settings.

Note: Apart from the serial port, the other parameters are fixed values: 19200, 8, N, 1.



5.6.2 Upload configuration

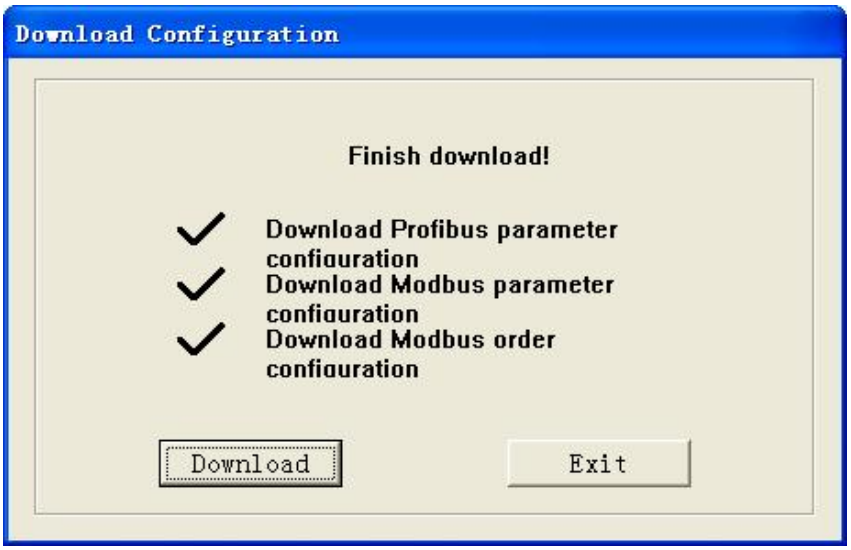
Choose upload configuration, upload the configuration from gateway to the software, the interface is shown as follows:



Note: Before uploading the configuration, please check whether the port is the available port.

5.6.3 Download configuration

Choose download configuration, download the configuration from software to the gateway, the interface is shown as follows:



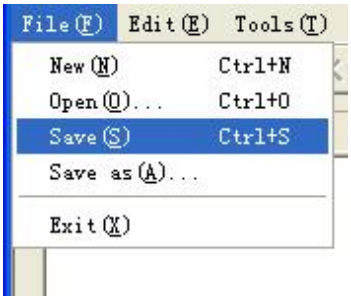
Note1: Before downloading the configuration, please check whether the port is the available port.

Note 2: Before downloading the configuration, make sure that all configurations has been completed.

5.7 Open and save configuration

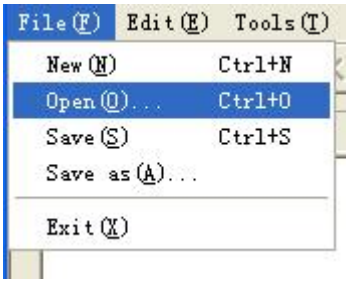
5.7.1 Save Configuration project

Choosing "Save" can save a project.



5.7.2 Open configuration project

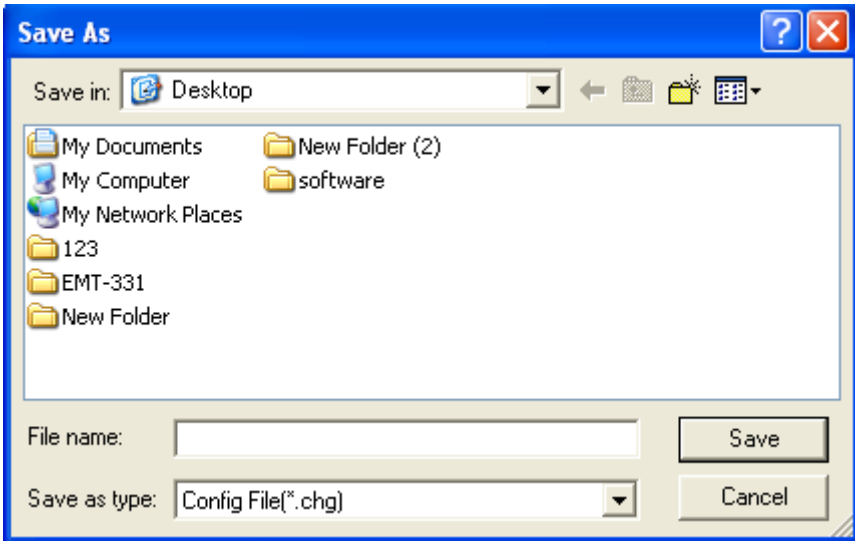
Choosing "Open" can open a saved project before.



5.8 Output excel document

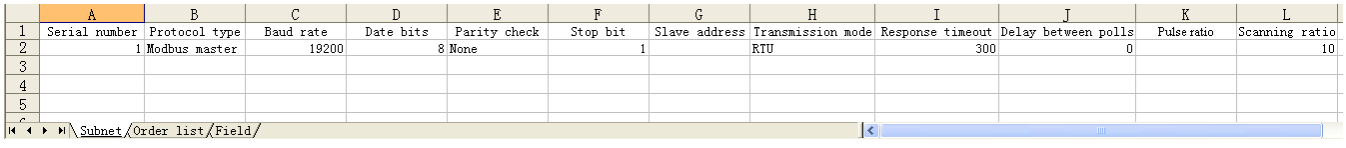
Excel document helps users to examine the configuration related.

Choose the icon, save the configuration as excel document and choose the right path.



Double-click to open the document, and it divided to three parts: "Command List", "Fieldbus", and "Subnet".

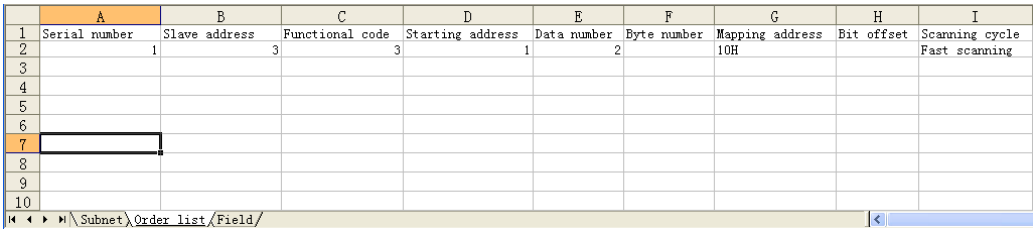
Subnet: Modbus subnet parameters. It is shown as follow:



	A	B	C	D	E	F	G	H	I	J	K	L
1	Serial number	Protocol type	Baud rate	Date bits	Parity check	Stop bit	Slave address	Transmission mode	Response timeout	Delay between polls	Pulse ratio	Scanning ratio
2	1	Modbus master	19200	8	None	1		RTU	300	0		10
3												
4												
5												

Subnet / Order list / Field /

Commands list: Modbus command list. It is shown as follow:



	A	B	C	D	E	F	G	H	I
1	Serial number	Slave address	Functional code	Starting address	Data number	Byte number	Mapping address	Bit offset	Scanning cycle
2	1	3	3	1	2		10H		Fast scanning
3									
4									
5									
6									
7									
8									
9									
10									

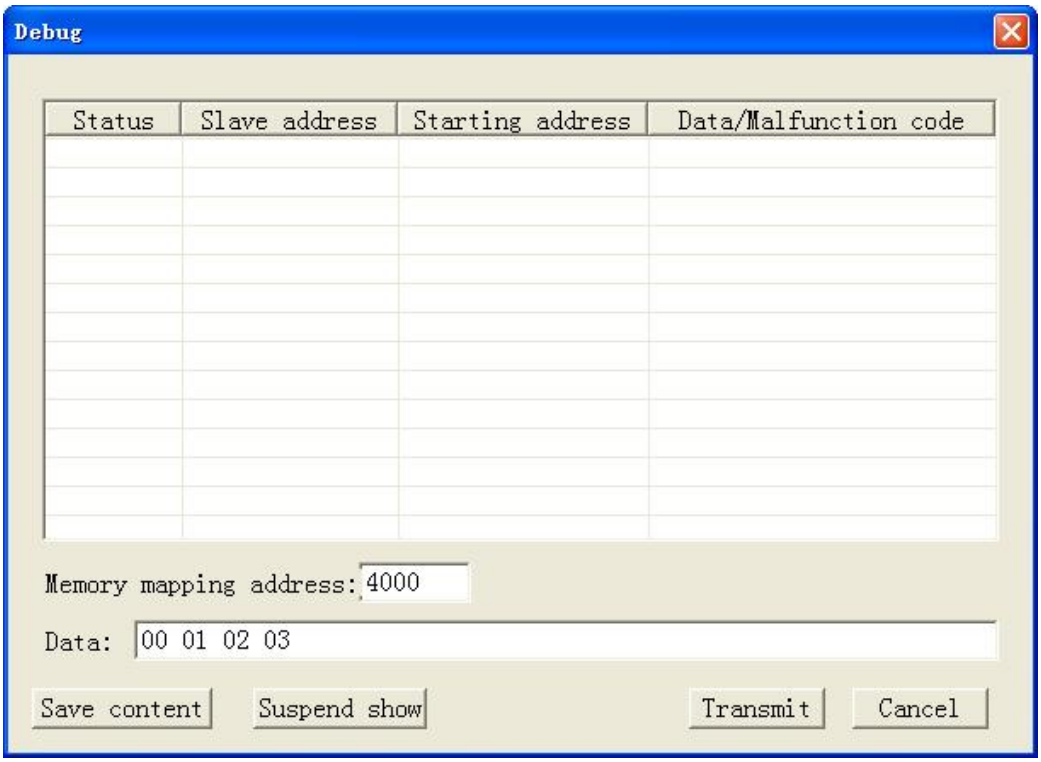
Subnet / Order list / Field /

Fieldbus: Fieldbus type and relevant parameters. It is shown as follow:

	A	B	C	D	E
1	Protocol type	Address	Communication baud	Input byte	Output byte
2	DeviceNet	5	125K	64	112
3					
4					
5					
6					
7					

5.9 Debug

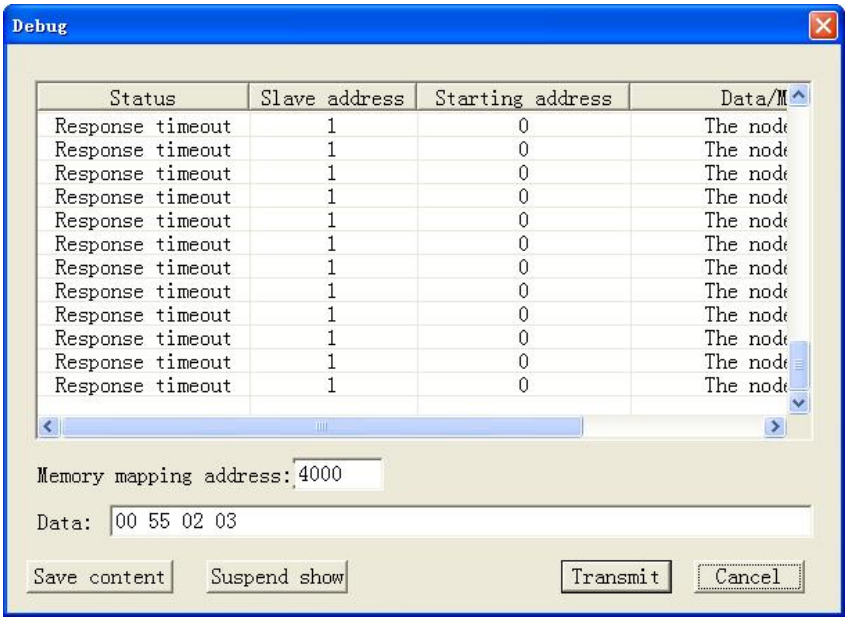
The function is for debugging Modbus network communications, the interface is shown as follows:



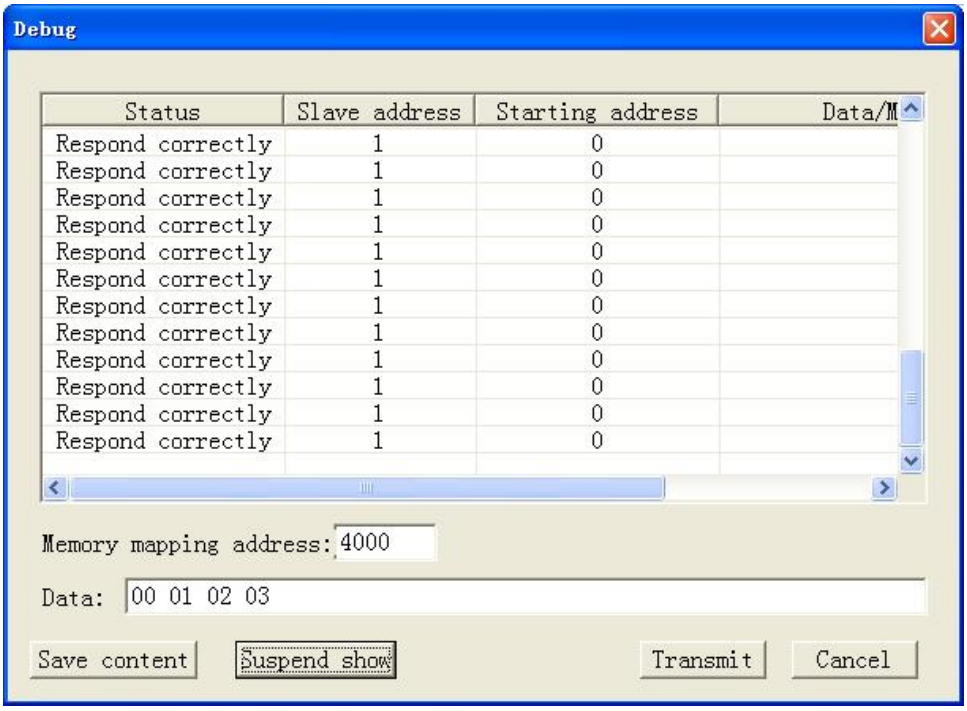
Memory mapping address: Starting address of data writing into the gateway memory

Data: Data writing into the gateway

When Modbus slave has no response or response timeout:

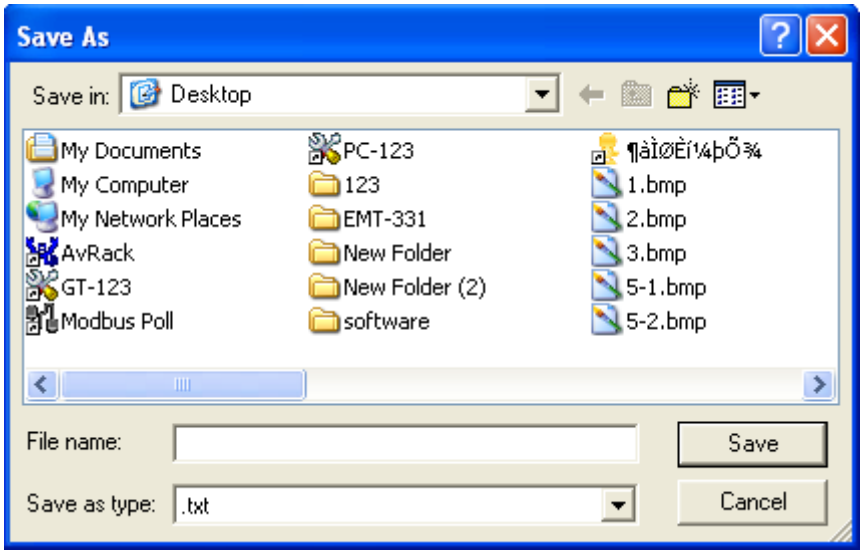


When Modbus responses are right:



After filling the "Memory mapping address" and "Data" rightly, users can click on "Transmit" button to transmit the packet.

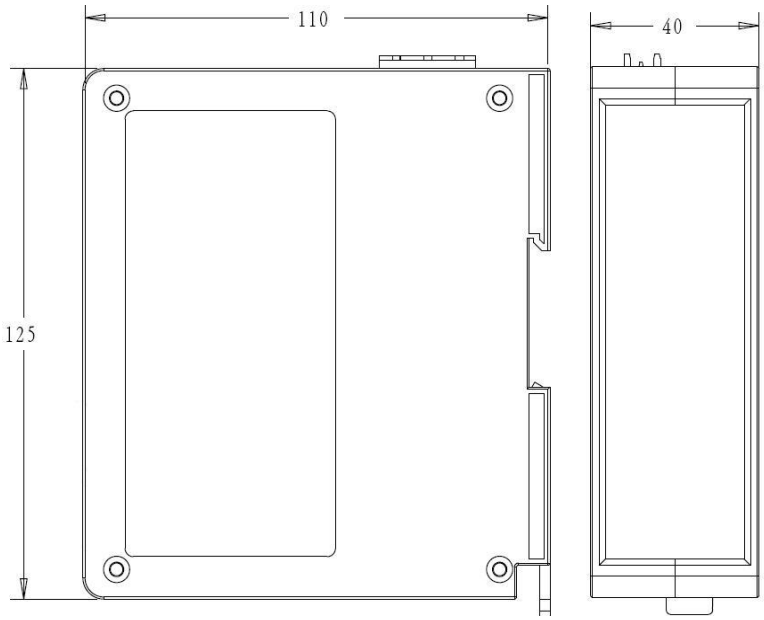
User clicks on the "Save content" button can save the received data to a computer's hard disk.


Save As
Save in:
Desktop
My Documents
My Computer
My Network Places
AvRack
GT-123
Modbus Poll
PC-123
123
EMT-331
New Folder
New Folder (2)
software
¶àÌØÈí¼þÕ¾
1.bmp
2.bmp
3.bmp
5-1.bmp
5-2.bmp
File name:
Save as type:
.txt
Save
Cancel

6 Installation

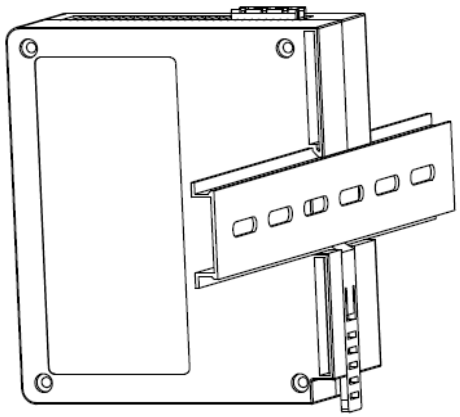
6.1 Mechanical Dimension

Mechanical Dimension: 110mm (W)*125mm (H)*40mm (D)



6.2 Installation

35mm DIN rail installation



7 Introduction to optional attachment

7.1 RS25——RS232/RS485 Isolated converters

RS25 is a product of Sibotech, and it is a RS232/RS485 isolated converter.

Function: Provides communication between RS232 and RS485.

Features: With 1000V photoelectric isolation, be applicable to the industrial scene with multivariate environment.

Note: There aren't special requires to communication with MS-210, and Most RS232-485 converters can be used. More information, please visit: www.sibotech.net/en

7.2 Communication lines

Users should select 3048A five core shielded wires (GB/T18858.3) to connect to DeviceNet drop lines; selecting cables non-compliant will shorten the communication distance.

3048A five core shielded wires

Select Class A shielded wires to connect Modbus device connectors. As follow:

Class A shielded wires

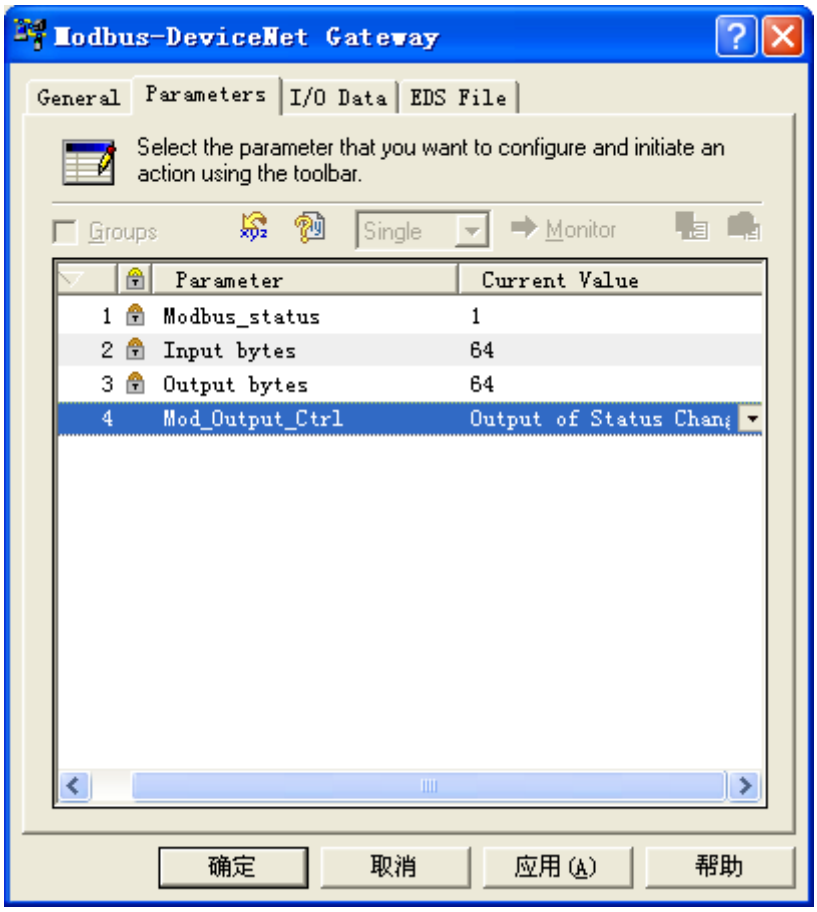
8 Instructions of DeviceNet I/O and parameters

8.1 I/O Configuration

DeviceNet input-bytes could be configured as 8 bytes, 16 bytes, 32 bytes, 64 bytes, 96 bytes, 128 bytes or 160bytes.

DeviceNet output-bytes could be configured as 8 bytes, 16 bytes, 32 bytes, 64 bytes, 96 bytes, 112 bytes.

8.2 DeviceNet parameters



Modbus_Status: The value shows the status of Modbus communications. If it keeps zero, the Modbus communication is OK. If it is a non-zero value, the communication of the command is fail.

Input-Bytes: Number of DeviceNet input bytes

Output-Bytes: Number of DeviceNet output bytes

The parameters of input-bytes and output-bytes must be the same with the numbers of input/output bytes in DeviceNet master scanning list of RSNetWorx and so on, or the connection can't be established.

Mod_output_ctrl: Modbus output control

Continuous Output

Disable Output

Output of Status Change: When the network output data has changed, Modbus commands can be sent.

Note: If the output mode is "Disable Output", though it has configured Modbus output commands, the gateway won't send Modbus output commands.

To ensure securities of output-data, if PLC hasn't effective output-data (For example, PLC in programming mode or DeviceNet device has not been connected), Modbus output commands will not be sent.

This parameter also could be modified through Modbus setting in GT-123

8.3 DeviceNet network configuration instructions

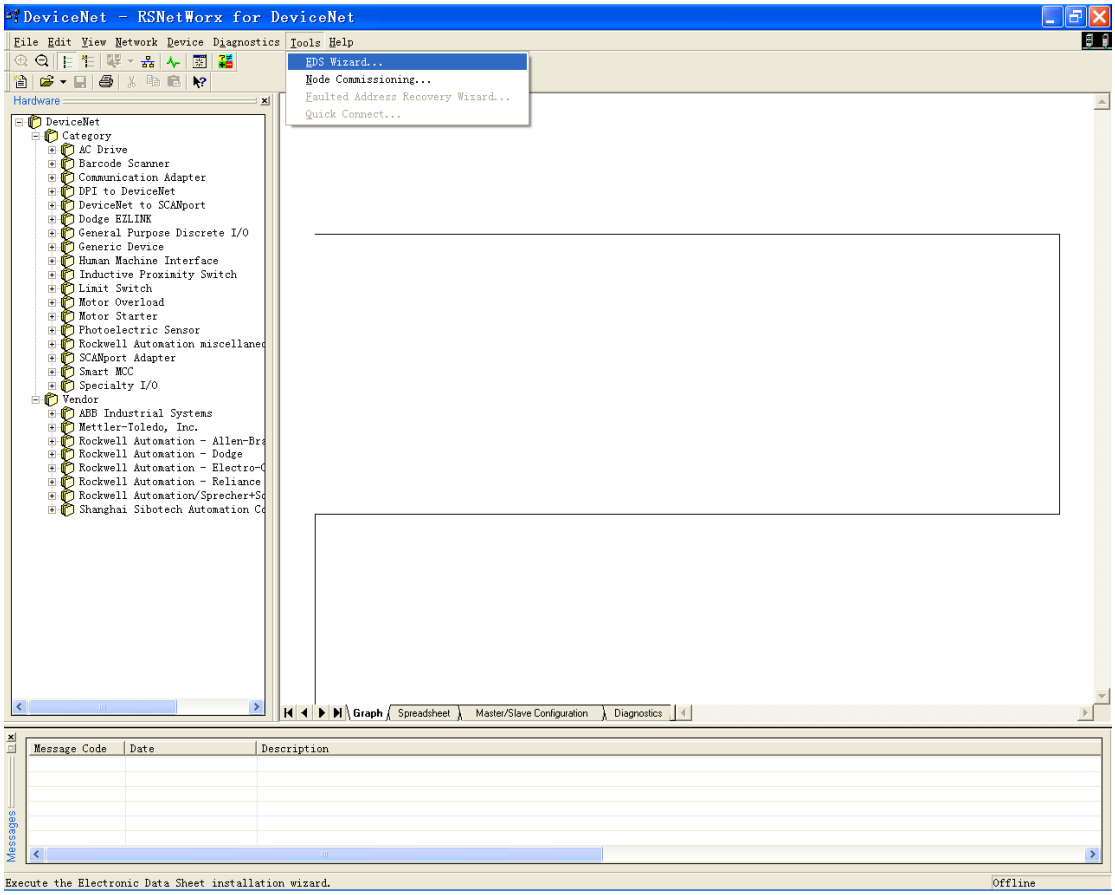
Users need install the *.EDS file in the disc to DeviceNet configuration software, then you can configure MD-210 through network configuration software.

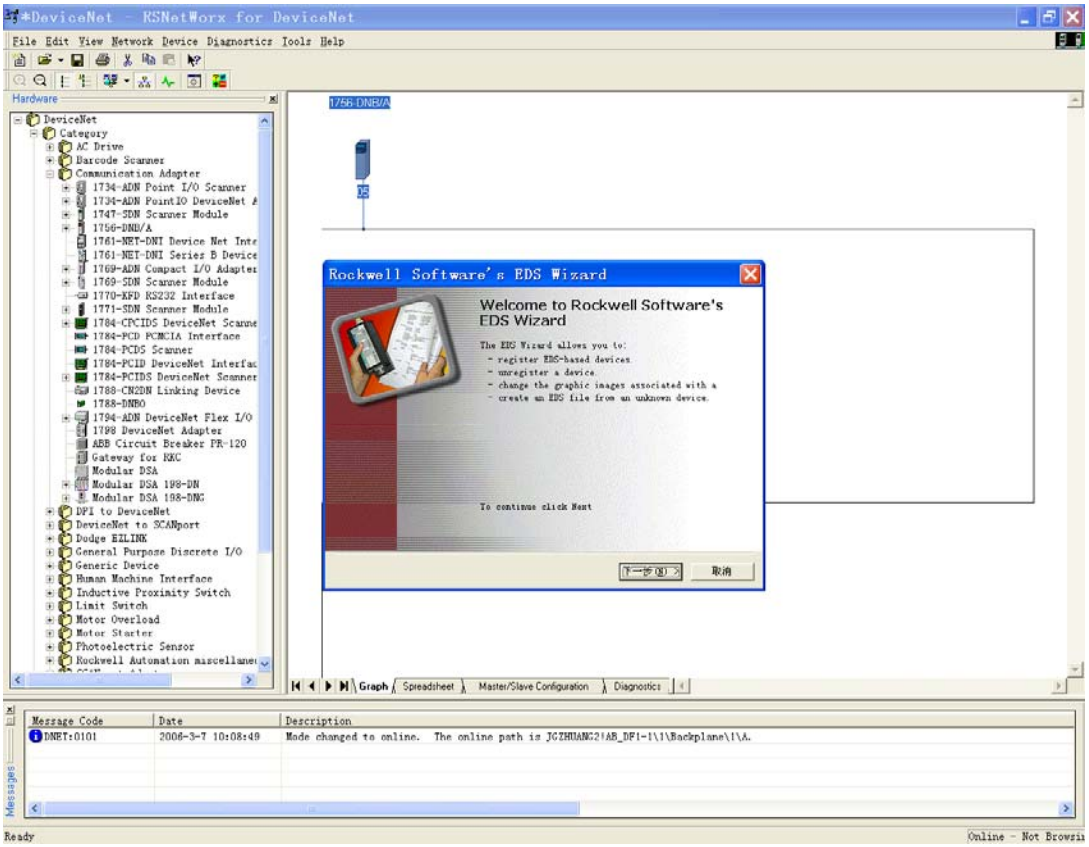
EDS（Electronic Data Sheet）is comprehensive description which supports DeviceNet network function. It equals to equipment's driver of Windows. Users need install EDS files to DeviceNet network configuration software, such as RsNetWorx and so on, and then the configuration can be going on through network configuration software.

Here we take Rockwell's RsNetWorx for example (edition 4.12.0), and explain how to install. For further details, please refer to the network configuration software instructions.

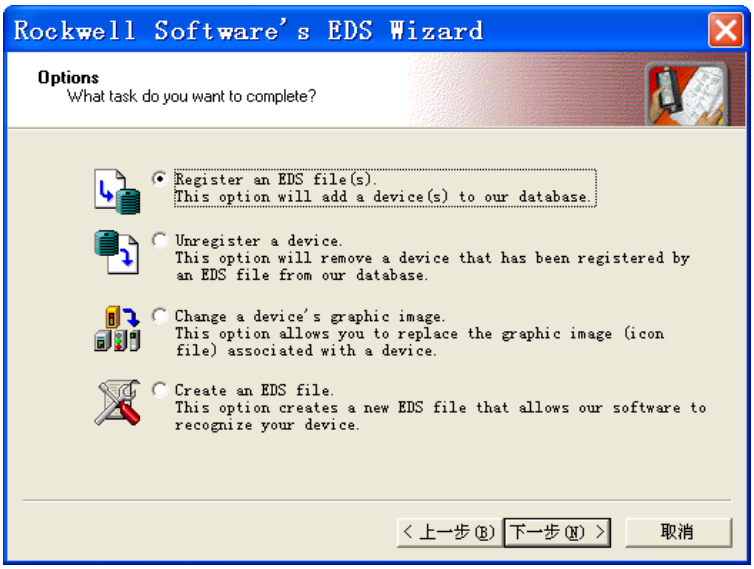
Step1: Create a new network configuration profile

Step2: Select EDS operation guide, select "Tools" and then "EDS-Wizard", you will see that:


DeviceNet - RSNetWorx for DeviceNet
File Edit View Network Device Diagnostics Tools Help
EDS Wizard...
Node Commissioning...
Faulted Address Recovery Wizard...
Quick Connect...
Hardware
DeviceNet
Category
AC Drive
Barcode Scanner
Communication Adapter
DPI to DeviceNet
DeviceNet to SCANport
Dodge EZLINK
General Purpose Discrete I/O
Generic Device
Human Machine Interface
Inductive Proximity Switch
Limit Switch
Motor Overload
Motor Starter
Photoelectric Sensor
SCANport Adapter
Smart MCC
Specialty I/O
Vendor
ABB Industrial Systems
Mettler-Toledo, Inc.
Rockwell Automation - Dodge
Rockwell Automation - Reliance
Shanghai Sibotech Automation
Graph
Spreadsheet
Master/Slave Configuration
Diagnostics
Message Code
Date
Description
Messages
Execute the Electronic Data Sheet installation wizard.
Offline


DeviceNet - RSNetWorx for DeviceNet
File Edit View Network Device Diagnostics Tools Help
Hardware
1756-DNB/A
Rockwell Software's EDS Wizard
Welcome to Rockwell Software's EDS Wizard
To continue click Next
Graph
Spreadsheet
Master/Slave Configuration
Diagnostics
Message Code
Date
Description
Messages
Ready
Online - Not Browsi

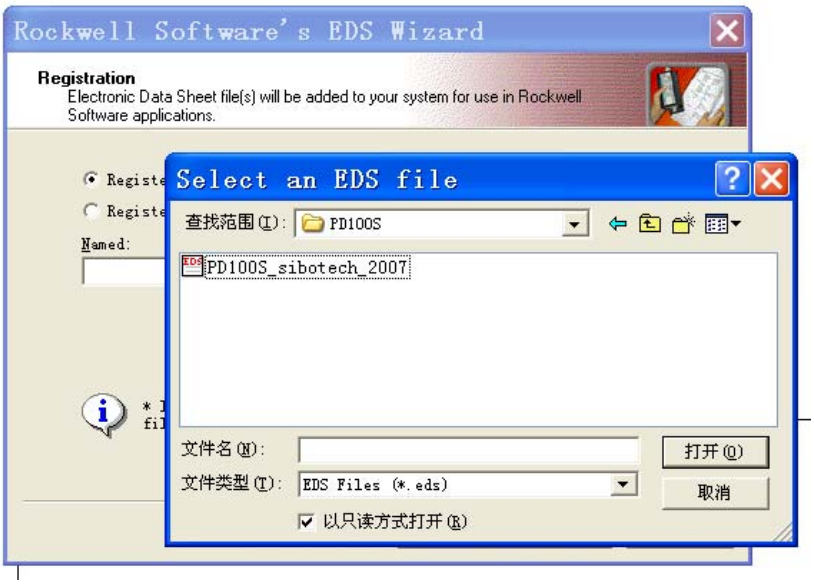
Step3: Select “NEXT”, as follow:



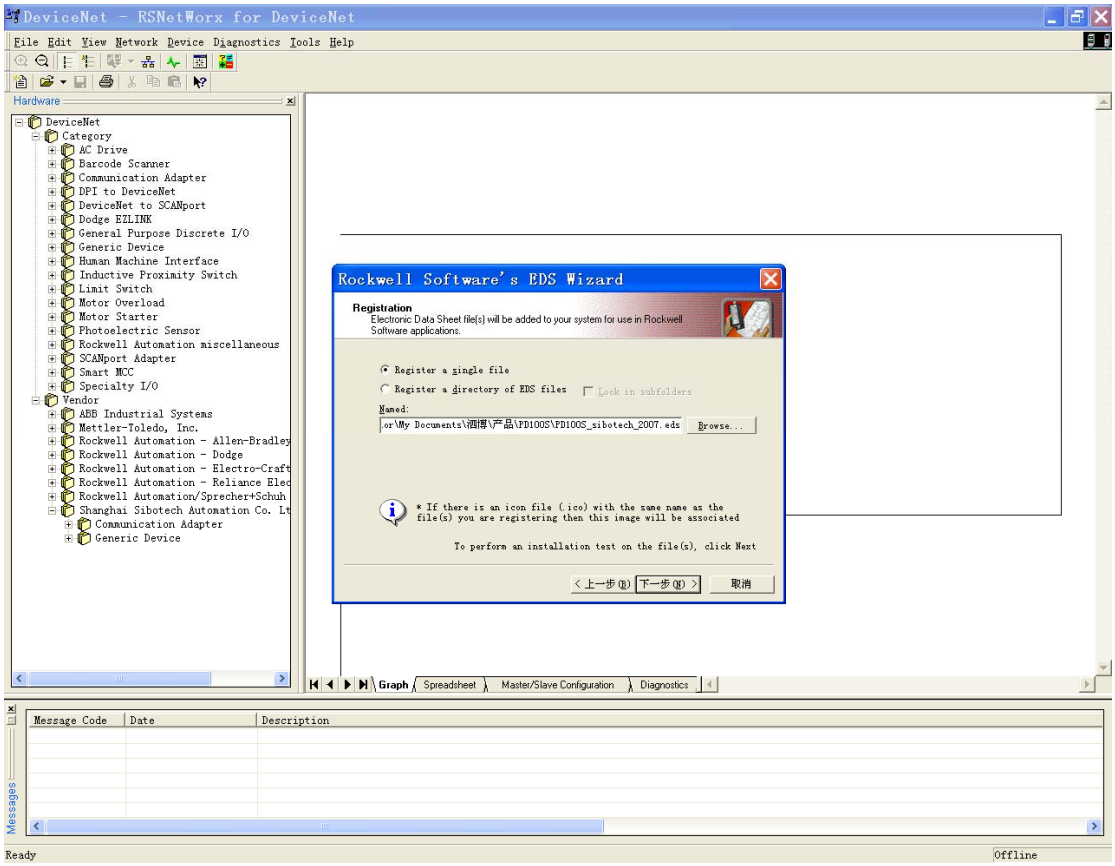
Step4: Install gateway MD-210

Shown as above, select “Register an EDS file”, as follow:

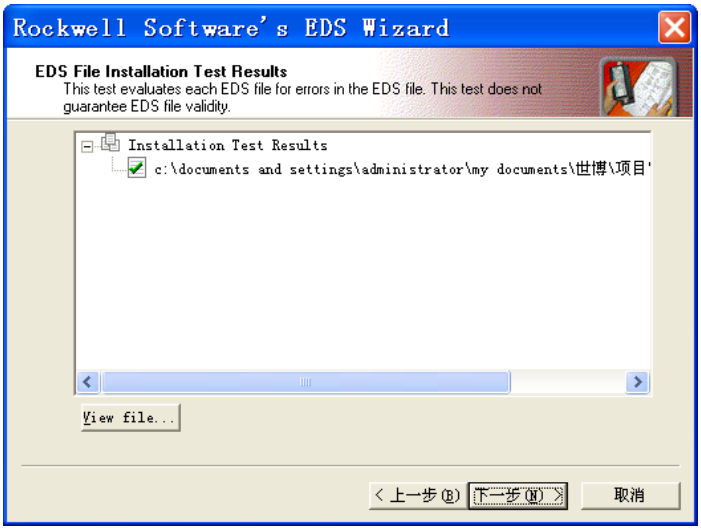
Please register MD-210.EDS file we provided, according to the place where you save EDS file, and select the file.



Step 5：Select the file registering to choose

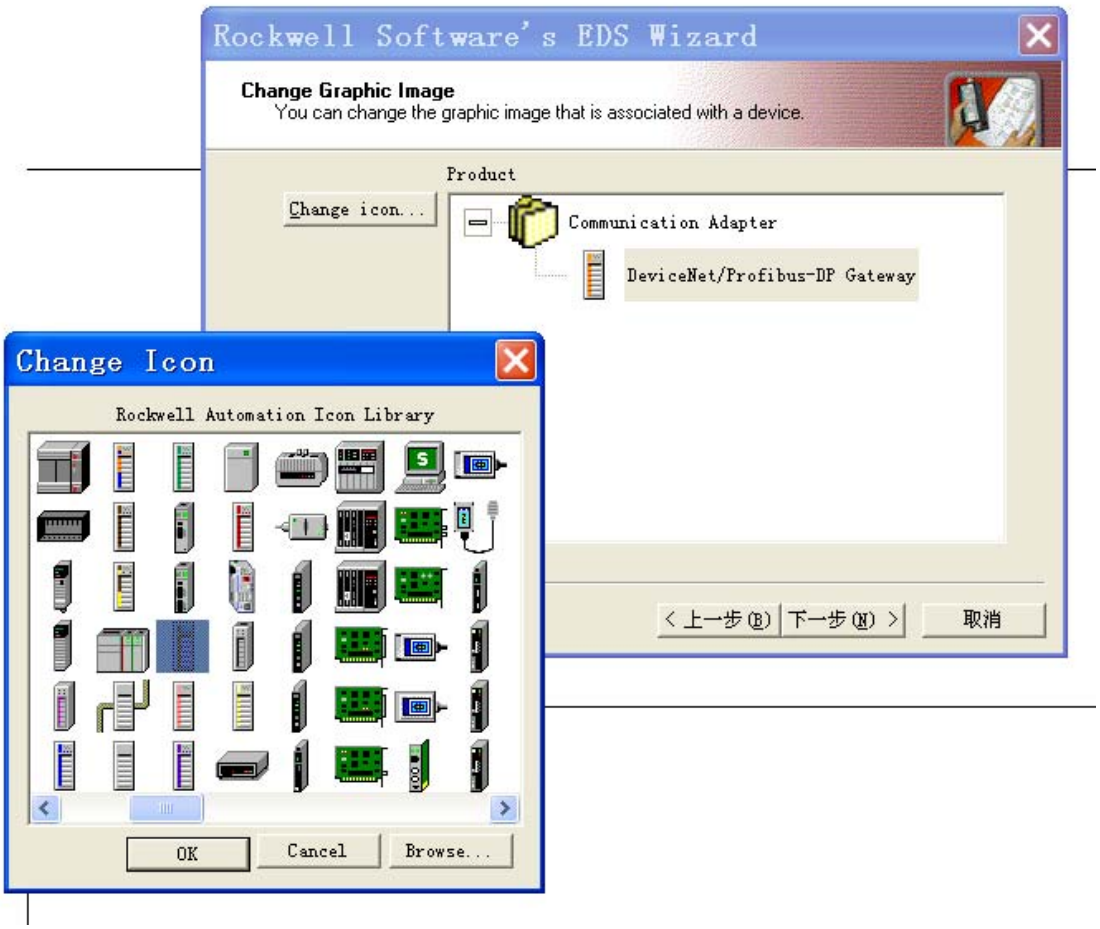


Click “NEXT” :

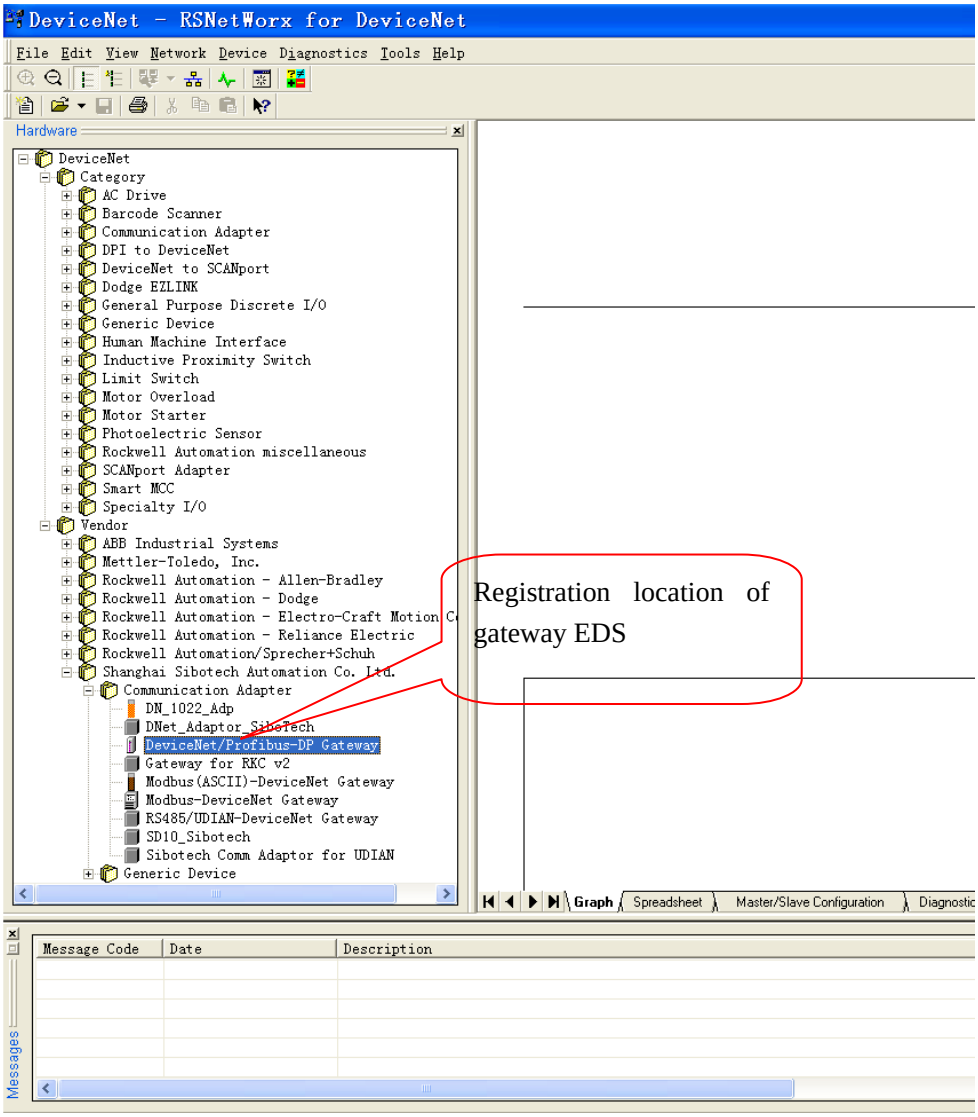


Step 6: Select the icon.

Following network configuration software will prompt you the equipment category in equipment storehouse, you may choose icon in this process.



Here, the device has successfully registered to the icon library location of configuration software's equipment storehouse.



Then , you should connect gateway MD-210 to DeviceNet network, click on “SCAN” button of RsNetWorx , or select “Network-Online” in menu bar, your gateway will be scanned by system and identified exactly.

Appendix A: Modbus Protocol

Modbus-RTU Protocol:

Note: The equipments being connected with this product must have Modbus interface, and Modbus protocol of the equipments must be in line with the following rules. Our company provides customer-oriented services.

1 Description of protocol

Physical layer: transmission mode: RS485

Address: 0-247

Baud rate: Can configurate

Medium: STP

Transmission mode: Half-duplex mode

The connection is established throng one line with half-duplex mode, and that is to say that signals transmit through the only one line with opposite directions. Firstly, host computer find the only terminal and then the terminal transmits response signals on the opposite direction.

Protocol only allows the communication between host computers and terminals, while the communications between terminals are not allowed. Thus, they will not occupy communication line when they are in the status of being initialized and respond the polling signals which are transmitted to the local terminal only.

Format of a data frame: 1 bit start bit, 8 bits data and 1 bit stop bit.

Format of a data packet

Address	Function Code	Data	check code
8-Bits	8-Bits	N x 8-Bits	16-Bits

The protocol defines check code, data serial and so on in details, which are important contents when communicating specific data.

When data frames reach terminal, they access to the equipment through a simple entrance. The equipment delete the “envelop” of the data frame, and read the data. If there are no faults, the required tasks are executed. Then, it adds the data being generated by itself to the obtained “envelop”, and return the data frame to the sender.

The response data include: the address of terminal, the executed function, and the required data and a check code by executing the order. Any faults won't lead to respond successfully.

Address field

Address field is located at the beginning, composed of 8 bits (0-255). These bits indicate the address of the terminal being specified by users. The equipment will receive data from the host computer being connected with it. Every address of terminal must be the only one. Only the terminal being addressed will respond the polling with its address. when the terminal transmit a response, the response tell the host computer the terminal which is communicating with it.

Function field

Codes of function field show the function being executed by terminals which are addressed. Table 1-1 lists all the function codes, their meanings and their initial functions.

Table1-1 Function codes

code	meaning	action
03	Read data	obtain current binary value of single or multiple registers
06	Preset single register	Place one specific binary value into the single register
16	Preset multiple registers	Place specific binary values to a series of multiple registers

Data field

Data field includes data when specific functions are executed by terminals or data being collected when the terminal responds to query. The content of these data may be value of number, reference address or limit value. For instance, functional code indicates terminals read a register, data field, on the other hand, clearly show the register and the number of data to be read, the inside address, the type of data and different capacity of different computers.

Fault check code

The field allows checking the fault in the transmission between host computer and terminals. Sometimes, due to electrical noise and other interference, a set of data may change when transmitting from one equipment to another. The fault checking code can guarantee that host computer don't respond to the changed data in the transmission, which improves safety and efficiency of the system. The fault code apply 16 bit CRC.

Note: The transmitting serial is always the same- address, functional code, data and the fault checking code relating to direction.

The fault checking

CRC field occupy 2 bytes, including 16 bits binary value. The value of CRC is calculated by transmitting equipment, and then added to the data frame, the receiving equipment calculate the CRC value again while receiving data. Then it is compared with receiving CRC value. If the two values are not the same, the fault occurs.

When calculating CRC, preset a 16-bit register to one firstly, and then calculate 8-bit bytes in the data frame with the current value of the register. Only 8 data bits of each byte participate in the generation of CRC. The initial bit, final bit and occasional odd and even bit don't influence the value of CRC.

The process of generating a CRC:

Preset a 16-bit register to 0FFFFH, and the register is named CRC register.

When generating CRC, exclusive each 8 bit with the content of the register, and then shift result to the low byte, the high bit is filled with zero, the LSB is shifted out and checked. If it is one, the register exclusives with a presetting fixed value. If the lowest bit is zero, there is no settlement.

The above settlement is repeated before executing the shifting 8 times. After finishing shifting the last bit, the next 8-bit byte has the same exclusive calculation with the register, and the another 8-time shifting is carried out. When all the bytes are settled, the final value is CRC value.

The process of generating a CRC:

Preset a 16-bit register 0FFFFH (all one), and name it CRC register.

Exclusive the first 8-bit byte in the data frame with the low byte in the CRC register, and restore CRC register. Shift CRC register to the right bit, fill the highest bit with zero, shift the lowest bit out and check them.

If the lowest bit is zero, repeat the third step (next shift)

If the lowest if one, exclusive CRC register with a presetting fixed value (0A001H).

Repeat the third step and the fourth step until shifting eight times, which settle the entire eight bit down.

Repeat the second step to the fifth step to deal with the next eight-bit until all the bytes are settled down.

In the end, CRC register value is the CRC value.

2 Functions of application layer

The first chapter has described the protocol and data frame. The processors of the software can use the

following methods establish their specific application program via protocols without fault.

The protocol in this chapter use the follow format as many as possible, the format is shown as table 2-1(digital is set in hexadecimal)

Table 2-1

Address	Functional code	The high byte of the starting address of the variable	The low byte of the starting address of the variable	The high byte of the number of the variable	The low byte of the number of the variable	The low byte of checking code	The high byte of checking code
03H	03H	00H	01H	00H	03H	55H	E9H

Read Holding Registers (Function code 03)

Query

The Table 2-2 is an example that reading there collected data U1, U2, and U3 to the slave. The address of U1 is 0001H, the address of U2 is 0002H, and the address of U3 is 0003H.

Table 2-2

Address	Functional code	The high byte of the starting address of the variable	The low byte of the starting address of the variable	The high byte of the number of the variable	The low byte of the number of the variable	The low byte of checking code	The high byte of checking code
03H	03H	00H	01H	00H	03H	55H	E9H

Response

The response include: the address of slave, functional code, the number of the data and the CRC check.

The example table 2-3 is reading the response of U1, U2, and U3.

Table 2-3

Address	Functional code	The byte number of the variable	The high byte of the variable	The low byte of the variable	The high byte of the variable	The low byte of the variable	The high byte of the variable	The low byte of the variable	The low byte of checking code	The high byte of checking code
03H	03H	06H	01H	7CH	01H	7DH	01H	7CH	F9H	9BH

2.2 Preset Multiple Registers (Functional code 10)

Query

The functional code 10 allows users changing the content of multiple registers. The device can be set 16 values from any starting address. The controller work with the mode of dynamic scanning, and it can change the content of the register anytime.

The Table 2-4 is an example changing the action and delay setting values of the monitor 1 and monitor 2 of the slave. The address of the action setting value of the monitor 1 is 2AH, and the delay setting value is 2BH. The address of the action setting value of the monitor 2 is 2CH, and the delay setting value is 2DH

Table 2-4

Address	Functional code	The high byte of the starting address of the variable	The low byte of the starting address of the variable 节	The high byte of the number of the variable	The low byte of the number of the variable	The byte number of the variable	The high byte of the variable	The low byte of the variable	The high byte of the variable	The low byte of the variable	The high byte of the variable	The low byte of the variable	The high byte of the variable	The low byte of the variable	The low byte of checking code	The high byte of checking code

03H	10H	00H	2AH	00H	04H	08H	07H	D0H	00H	0AH	07H	0D0H	00H	0AH	25H	7CH

Response

The Table 2-5 is the response of changing the action and delay setting values of the monitor 1 and monitor 2.

Table 2-5

Address	Functional code	The high byte of the starting address of the variable	The low byte of the starting address of the variable	The high byte of the number of the variable	The low byte of the number of the variable	The low byte of checking code	The high byte of checking code
03	10H	00H	2AH	00H	04H	EBH	8DH

2.3 Preset Single Register (Functional code 06)

Query

The functional code 06 allows users changing the content of single register. Any single register of DAE system can use the order change the value. The controller work with the mode of dynamic scanning, and it can change the content of the register anytime.

The following example is changing the overload-action value Ir1. The address of Ir1 is 002EH.

Table 2-6

Address	Functional code	The high byte of the starting address of the variable	The low byte of the starting address of the variable	The high byte of the variable	The low byte of the variable	The low byte of checking code	The high byte of checking code
03H	06H	00H	2EH	07H	0D0H	EBH	8DH

Response

The normal response of preset single register is transmitting the receiving data after changing the value of the register.

Table 2-7

Address	Functional code	The high byte of the starting address of the variable	The low byte of the starting address of the variable	The high byte of the variable	The low byte of the variable	The low byte of checking code	The high byte of checking code
03H	06H	00H	2EH	07H	0D0H	EBH	8DH

Appendix B: EDS document

```
$ EZ-EDS Version 3.0 Generated Electronic Data Sheet
$       DeviceNet Electronic Data Sheet
$ Copyright (C) 2005-2009 Shanghai Sibotech Automation Co. Ltd.

[File]
        DescText = "Modbus-DeviceNet Gateway";
        CreateDate = 01-08-2008;
        CreateTime = 08:57:44;
        ModDate = 04-06-2009;
        ModTime = 10:06:05;
        Revision = 1.1;

[Device]
        VendCode = 1016;
        VendName = "Shanghai Sibotech Automation Co. Ltd.";
        ProdType = 12;
        ProdTypeStr = "DC Drives";
        ProdCode = 19;
        MajRev = 1;
        MinRev = 1;
        ProdName = "MD-210 Gateway";
        Catalog = "MD-210";

[IO_Info]
        Default = 0x0001;

        PollInfo =
                0x0001,
                4,
                4;

        Input1 =
                8,
                0,
                0x0001,
                "Run-time measurements and State",
                6,
```

```
            "20 04 24 74 30 03",
            "8 bytes";

Input2 =
            16,
            0,
            0x0001,
            "Network Input 2",
            6,
            "20 04 24 75 30 03",
            "16 Bytes";

Input3 =
            32,
            0,
            0x0001,
            "Network input 3",
            6,
            "20 04 24 76 30 03",
            "32 Bytes";

Input4 =
            64,
            0,
            0x0001,
            "Network input 4",
            6,
            "20 04 24 77 30 03",
            "64 Bytes";

Input5 =
            96,
            0,
            0x0001,
            "Network Input 5",
            6,
            "20 04 24 78 30 03",
            "96 Bytes";

Input6 =
            128,
            0,
```

```
            0x0001,
            "Network Input 6",
            6,
            "20 04 24 79 30 03",
            "128 Bytes";

Input7 =
            160,
            0,
            0x0001,
            "Network Input 7",
            6,
            "20 04 24 7A 30 03",
            "Input 7 160bytes";

Output1 =
            8,
            0,
            0x0001,
            "Network Output 1",
            6,
            "20 04 24 80 30 03",
            "8 Bytes";

Output2 =
            16,
            0,
            0x0001,
            "Network Output 2",
            6,
            "20 04 24 81 30 03",
            "16 Bytes";

Output3 =
            32,
            0,
            0x0001,
            "Network Output 3",
            6,
            "20 04 24 82 30 03",
            "32 Bytes";
```

```
        Output4 =
                64,
                0,
                0x0001,
                "Network Output 4",
                6,
                "20 04 24 83 30 03",
                "64 Bytes";

        Output5 =
                96,
                0,
                0x0001,
                "Network Output 5",
                6,
                "20 04 24 84 30 03",
                "96 Bytes";

        Output6 =
                112,
                0,
                0x0001,
                "Network Output 6",
                6,
                "20 04 24 85 30 03",
                "112 Bytes";

[ParamClass]
        MaxInst = 4;
        Descriptor = 0x0001;
        CfgAssembly = 0;

[Params]

        Param1 =
                0,
                6,"20 A2 24 01 30 64",
                0x0030,
                8,
                1,
                "Modbus_status",
                "",
```

```
            "Status of Modbus, OK or The time out error Command No.",
            0,50,0,
            ,,,,
            ,,,,
            0;

Param2 =
            0,
            6,"20 A2 24 01 30 65",
            0x0012,
            8,
            1,
            "Input bytes",
            "",
            "Number of poll input connection bytes",
            0,6,3,
            ,,,,
            ,,,,
            0;

Param3 =
            0,
            6,"20 A2 24 01 30 66",
            0x0012,
            8,
            1,
            "Output bytes",
            "",
            "Number of poll output connection bytes",
            0,5,3,
            ,,,,
            ,,,,
            0;

Param4 =
            0,
            6,"20 A2 24 01 30 67",
            0x0002,
            8,
            1,
            "Mod_Output_Ctrl",
            "",
```

```
                "Control of Modbus Output Commands",
                1,3,1,
                ,,,,
                ,,,,
                0;

[EnumPar]
        Param2 =
                "8",
                "16",
                "32",
                "64",
                "96",
                "128",
                "160";

        Param3 =
                "8",
                "16",
                "32",
                "64",
                "96",
                "112";

        Param4 =
                "Continous Output",
                "Disable Output",
                "Output of Status Change";

[Groups]
```