MDC-700 Series User Manual

May. 2025, Version 1.0.5

SUPPORT

MDC-711
MDC-714/MDC-714i
MDC-741
MDC-771

Written by Liam Lin
Edited by Sunny Chiu

Warranty

All products manufactured by ICP DAS are warranted against defective materials for a period of one year from the date of delivery to the original purchaser.

Warning

ICP DAS assumes no liability for damages consequent to the use of this product. ICP DAS reserves the right to change this manual at any time without notice. The information furnished by ICP DAS is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS for its use, nor for any infringements of patents or other rights of third parties resulting from its use.

Copyright

Contact Us

If you have any questions, please feel free to contact us via email at:

Service@icpdas.com

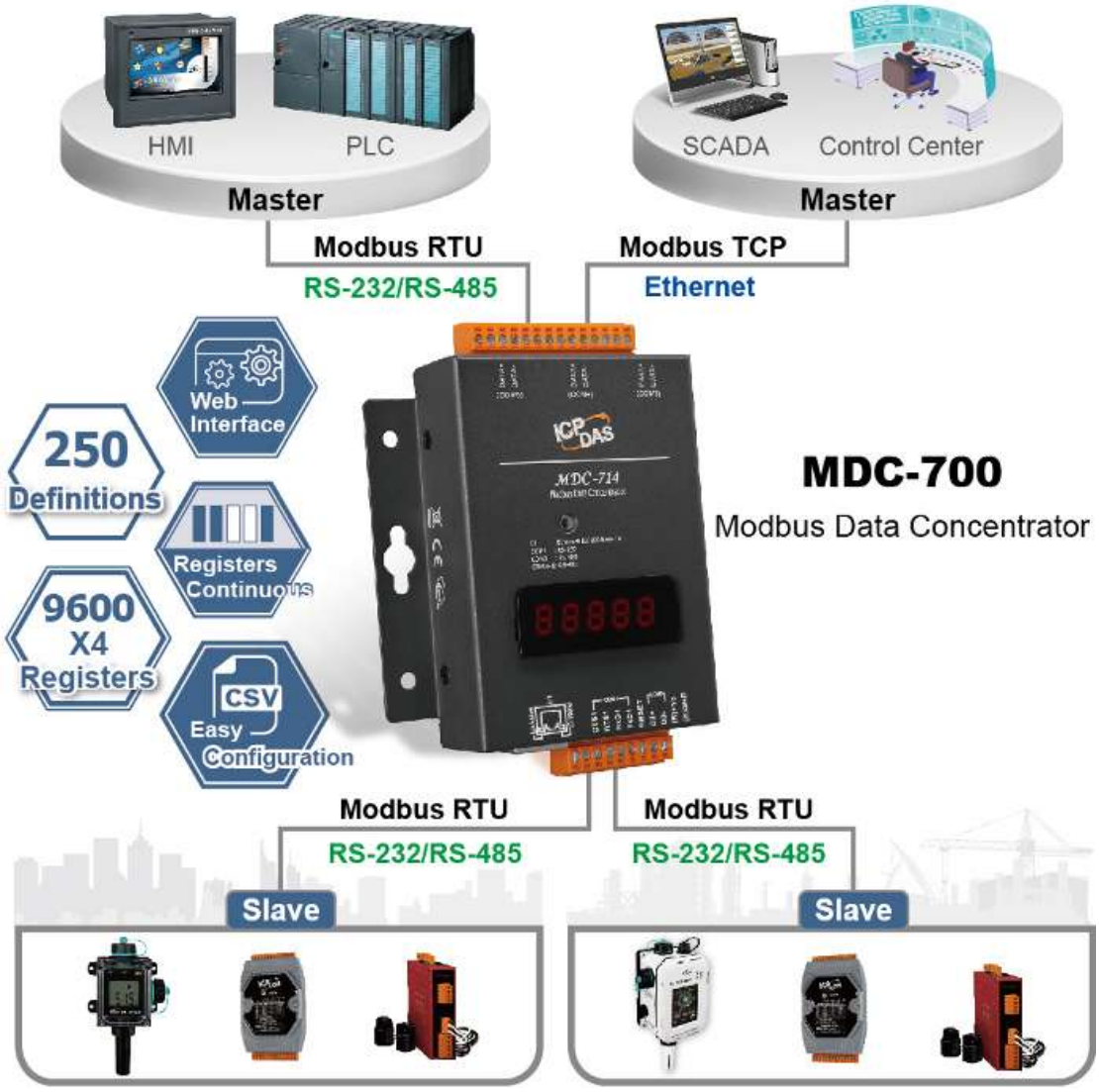
Contents

1. Introduction

The MDC-700 series is a Modbus Data Concentrator designed to aggregate data from multiple Modbus RTU slave devices via RS-232/RS-485 interfaces. It allows simultaneous read/write access by up to 8 Modbus TCP masters over Ethernet/LAN, enabling efficient and seamless data sharing.

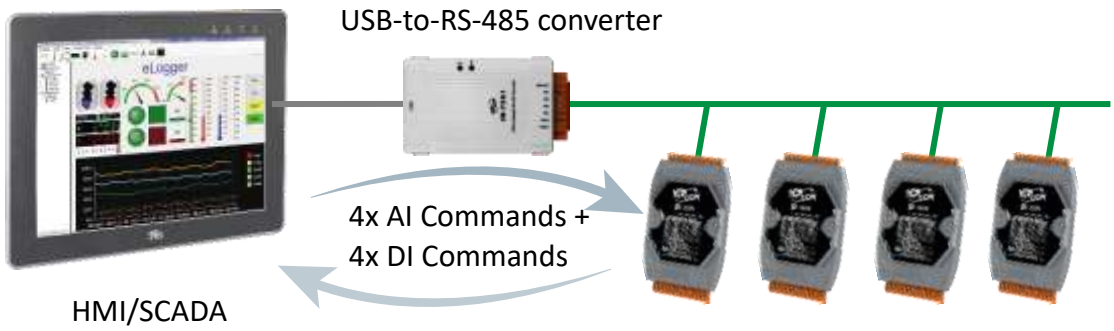
Data collected from various Modbus RTU slave devices is reorganized into a block of consecutive addresses within the MDC-700 module. This allows a Modbus TCP master to retrieve data on those Modbus RTU slave devices with a single Modbus command, significantly improving communication efficiency. In essence, the MDC-700 simplifies access to data from multiple Modbus RTU slaves over Ethernet, delivering enhanced performance.

The MDC-700 series is capable of executing up to 250 Modbus RTU commands to read from or write to RTU slave devices, and it supports up to 9,600 registers for each Modbus data type. With built-in support for the Modbus TCP protocol, the MDC-700 can be integrated smoothly into PC-based applications such as SCADA (Supervisory Control and Data Acquisition) systems and HMI (Human Machine Interface) software.

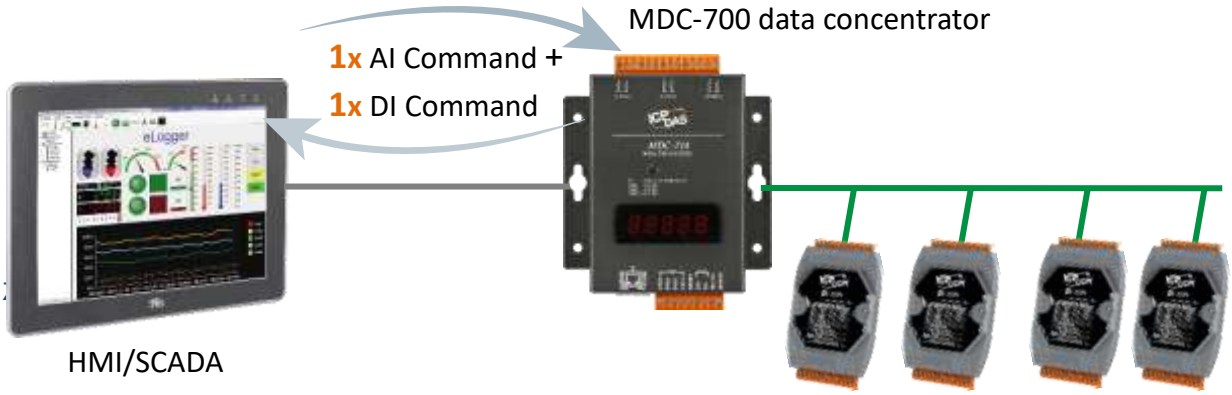


Speed up your Modbus system

When collecting data from multiple Modbus RTU devices, a USB-to-RS-485 converter is commonly used. For example, when reading from four M-7026 modules, each equipped with 6 analog input (AI) channels and 3 digital input (DI) channels. The HMI/SCADA host typically needs to issue a total of eight Modbus commands: four for AI data and four for DI data.



By using the MDC-700 data concentrator, the process is greatly streamlined. The MDC-700 reads the AI and DI data from all four M-7026 modules and organizes the results into a block of consecutive addresses. As a result, the HMI/SCADA host only needs to send one AI command and one DI command to retrieve data on the 4 M-7026 - 24 AI channels and 12 DI channels in total, significantly improving system performance and reducing communication overhead.

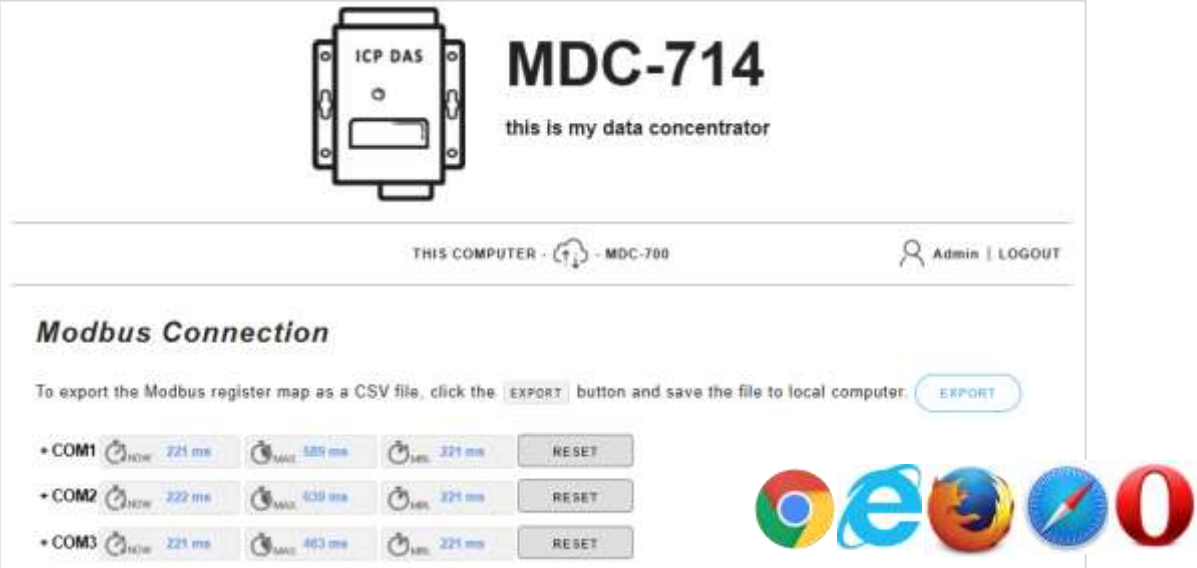


Features

■ HTML5 Web-based User Interface

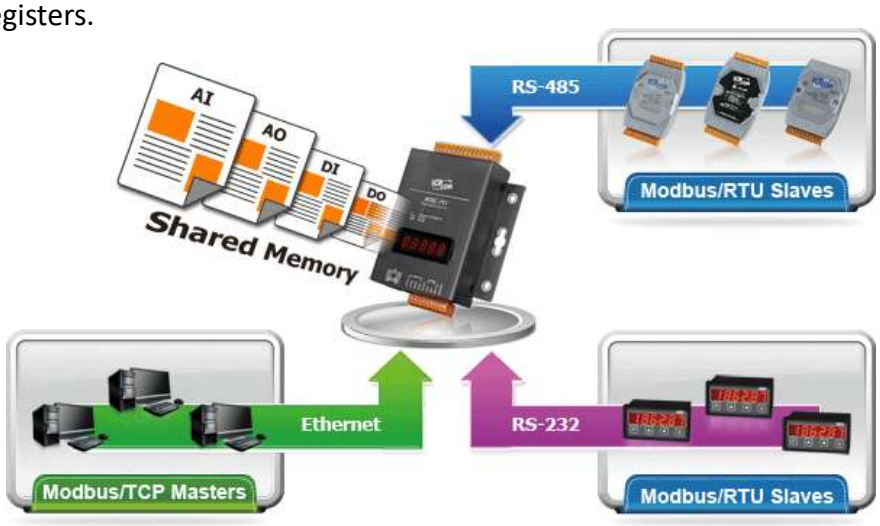
HTML5 is the latest version of the HTML markup language. It is supported by most current browsers including Mozilla Firefox, Apple Safari, Google Chrome and so on.

For the reason, the Web-based user interface of the MDC-700 is accessible from a wide variety of devices anywhere. Users can configure the module and monitor connection status of each polling definition through their smart phones, tablets or desktops without a line of code.



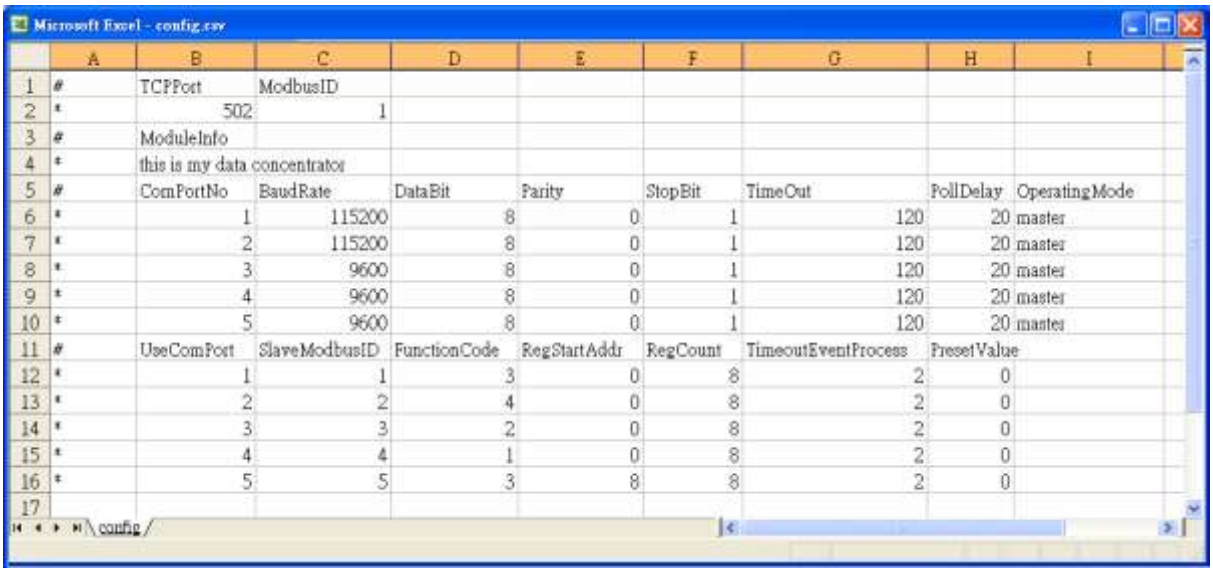
■ Great Capability of Shared Memory

The MDC-700 series module can perform up to 250 polling definitions. And the internal shared memory has four tables to store the polled AI, AO, DI and DO data. Each table can store up to 9600 registers.



Config.csv to Ease Hard Work of Editing a Lot of Definition

Editing and checking a lot of polling definitions is a hard work and it may make mistakes. Users can easy to manage so many definitions in a CSV format file with Excel and import or export the config.csv via a simple mouse-click action.

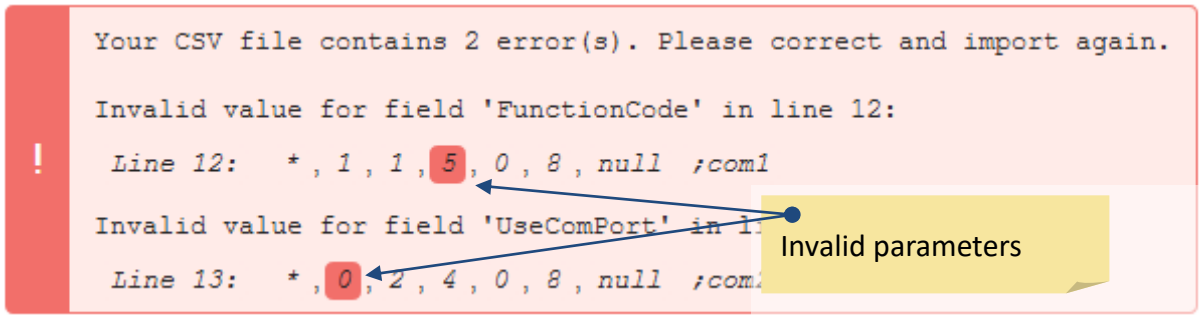


	A	B	C	D	E	F	G	H	I
1	#	TCPPort	ModbusID						
2	*	502	1						
3	#	ModuleInfo							
4	*	this is my data concentrator							
5	#	ComPortNo	BaudRate	DataBit	Parity	StopBit	TimeOut	PollDelay	OperatingMode
6	*	1	115200	8	0	1	120	20	master
7	*	2	115200	8	0	1	120	20	master
8	*	3	9600	8	0	1	120	20	master
9	*	4	9600	8	0	1	120	20	master
10	*	5	9600	8	0	1	120	20	master
11	#	UseComPort	SlaveModbusID	FunctionCode	RegStartAddr	RegCount	TimeoutEventProcess	PresetValue	
12	*	1	1	3	0	8	2	0	
13	*	2	2	4	0	8	2	0	
14	*	3	3	2	0	8	2	0	
15	*	4	4	1	0	8	2	0	
16	*	5	5	3	8	8	2	0	
17									

config

Built-in Definition Validation

One of the polling definitions may not be executed due to invalid parameters is given in the imported config.csv file. MDC-700 provides the function of validating and displaying invalid parameters with line information in config.csv file on its web interface.



Support for Modbus TCP Master and Modus RTU Master

The MDC-700 can be accessed by Modbus TCP Master and Modus RTU Master. Changing the mode for a COM port from Master to Slave allows a connected Modus RTU Master to read/write data from/to the Modbus RTU slave devices on the other COM ports.

Authentication and Access Control

The MDC-700 provides authentication and access control mechanisms designed to protect web-based system configurations and data. These features help ensure that only authorized users can access or interact with the device interface.

Administrator

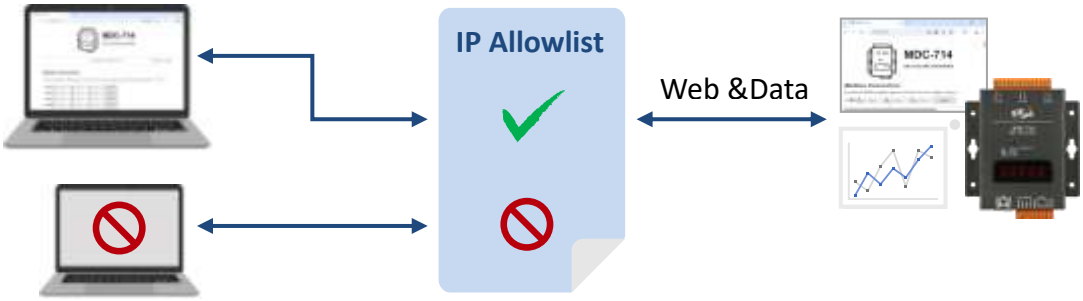
- Full access to configuration settings
- Create and manage user account
- Define access permissions for the user account

User

- View-only access to specific data
- No ability to modify device settings

IP Allowlist

The MDC-700 supports an **IP Allowlist** feature that enables administrator to restrict access to only trusted IP addresses. Once the IP Allowlist is enabled, only devices with IP addresses included in the list will be allowed to access the MDC-700's web interface and retrieve data via the Modbus TCP protocol. This helps prevent unauthorized access and potential system attacks, thereby enhancing overall system security.

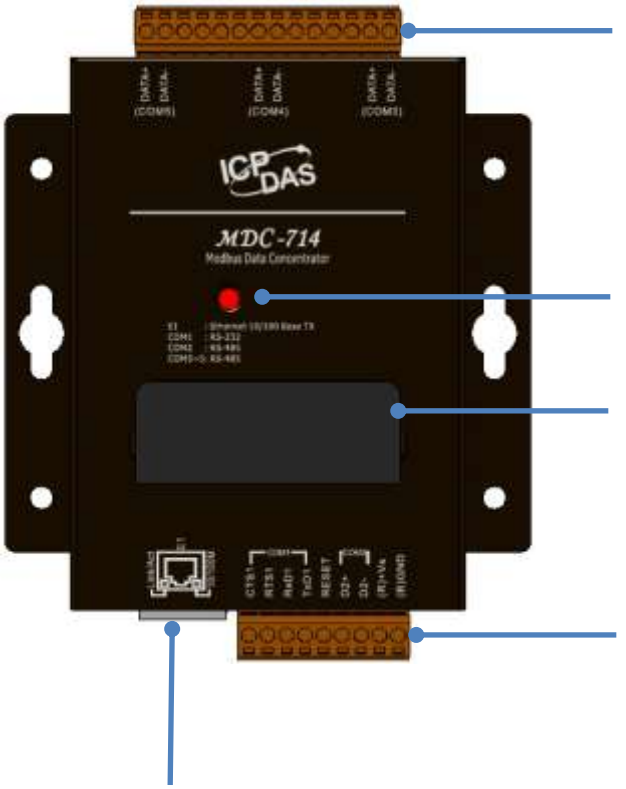


2. Hardware Information

2.1. Specifications

	MDC-711	MDC-714	MDC-714 i	MDC-741	MDC-771
Ethernet					
Port	RJ-45 x 1, 10/100 Base-TX				
Protocol	Modbus/TCP Slave				
Max. Connection	8				
COM Port					
RS-232	x 1 (5-wire)	x 1 (5-wire)	x 1 (5-wire)	x 4 (5-wire)	x 1 (5-wire) + x 6 (3-wire)
RS-485	x 1	x 4	x 4 (3 isolated ports)	x 1	x 1
Baud Rate	1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200 (bps)				
Data Format	N81, E81, O81				
Protocol	Modbus RTU Master/ Modbus RTU Slave				
Max. Node	32 nodes for each RS-485 port				
Polling Definition	250 definitions for all RS-232/485 ports				
Shared Memory	9600 registers for each of AI, AO, DI and DO Data				
System					
5-Digit 7 Segment LED Display	Yes, to display IP address and COM port configuration				
System LED Indicator	Yes, to display heartbeat				
Mechanical					
Casing	Metal				
Dimension	102 mm x 125 mm x 28 mm (W x H x D)				
Installation	DIN-rail /Wall mounting				
Power					
Input Range	+10 VDC ~ +30 VDC (non-regulated)				
Consumption	2.5 W				
Environmental					
Operating Temperature	-25°C ~ +75°C				
Storage Temperature	-30°C ~ +80°C				
Humidity	10 ~ 90% RH, non-condensing				

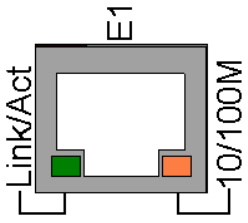
2.2. Appearance



■ LED Indicator

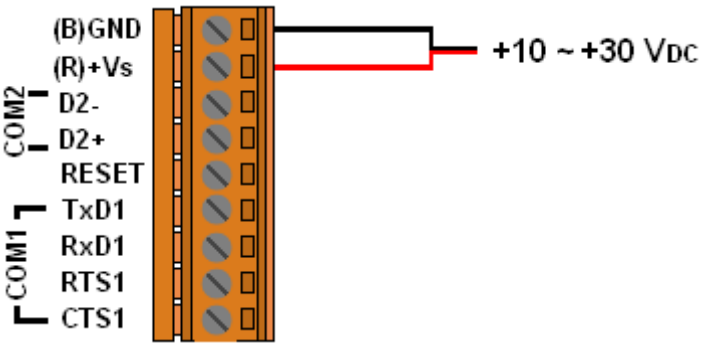
The LED is used as a heartbeat indicator and slows to approximately one flash per second.

■ Ethernet Port



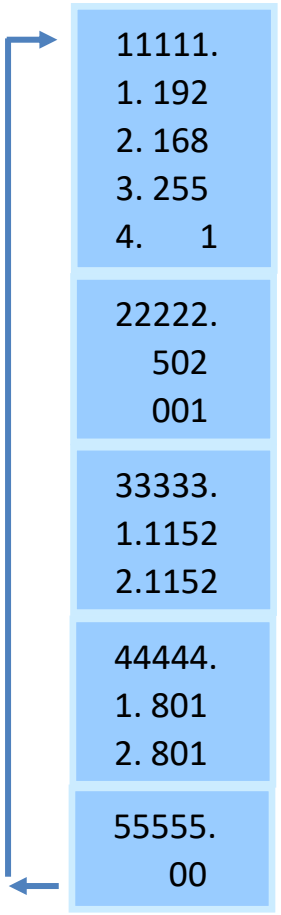
The MDC-700 is equipped with a RJ45 port for Ethernet LAN connection. When 100BASE-TX is operating, the 10/100M LED is lit orange. When 10BASE-T is operating or the machine is not connected to the network, it is turned off. When an Ethernet link is detected and an Ethernet packet is received, the Link/Act LED is lit green.

Power Connector



Configuration Display

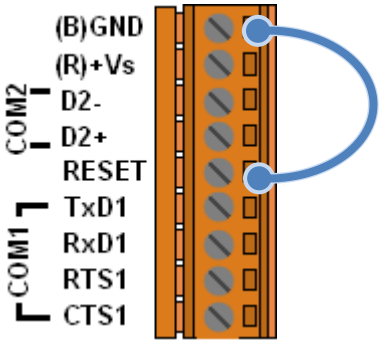
MDC-700 includes a 5-digit 7-Segment LED display to indicate configuration in a module as below:



- The IP address for the MDC-700
  (192.168.255.1)
- Modbus TCP communication settings
  Port: 502
  Net ID: 1
- Baud rate setting for each COM port
  COM1: 115200 bps
  COM2: 115200 bps
- Data format setting for each COM port
  COM1: 8N1
  COM2: 8N1
- The count of TCP/IP connection
  0: No TCP/IP connection

Reset

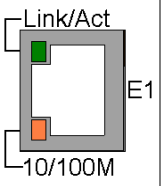
To restore the IP address/Subnet Mask/Gateway to the factory default values, short-circuit the RESET and GND pins for more than 3 seconds while the MDC-700 is powered on.



Over **3** seconds to restore factory default settings

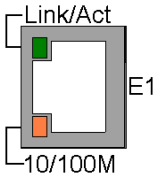
2.3. Pin Assignments

■ MDC-711

Terminal No.		Pin Assignment
E1		
COM1	01	CTS1
	02	RTS1
	03	RxD1
	04	TxD1
	05	RESET
COM2	06	D2+
	07	D2-
	08	(R)+Vs
	09	(B)GND

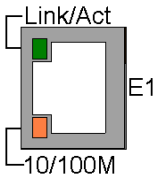
■ MDC-714/MDC-714i

COM3, COM4 and COM5 of MDC-714i are provided with 2500 VDC high voltage isolation protection.

Terminal No.		Pin Assignment
E1		
COM1	01	CTS1
	02	RTS1
	03	RxD1
	04	TxD1
	05	RESET
COM2	06	D2+
	07	D2-
	08	(R)+Vs
	09	(B)GND

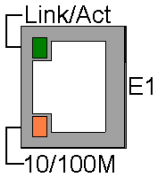
Terminal No.		Pin Assignment
COM5	23	DATA+
	22	DATA-
	21	
	20	
	19	
	18	
COM4	17	DATA+
	16	DATA-
	15	
	14	
	13	
	12	
COM3	11	DATA+
	10	DATA-

■ MDC-741

Terminal No.		Pin Assignment
E1		
COM1	01	CTS1
	02	RTS1
	03	RxD1
	04	TxD1
	05	RESET
COM2	06	D2+
	07	D2-
	08	(R)+Vs
	09	(B)GND

Terminal No.		Pin Assignment
COM5	23	RxD
	22	TxD
	21	RTS
	20	CTS
	19	GND
COM4	18	RxD
	17	TxD
	16	RTS
	15	CTS
	14	GND
COM3	13	RxD
	12	TxD
	11	RTS
	10	CTS

■ MDC-771

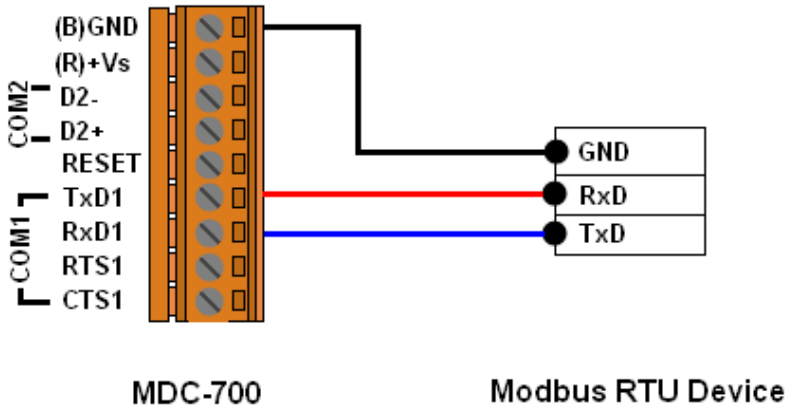
Terminal No.		Pin Assignment
E1		
COM1	01	CTS1
	02	RTS1
	03	RxD1
	04	TxD1
	05	RESET
COM2	06	D2+
	07	D2-
	08	(R)+Vs
	09	(B)GND

Terminal No.		Pin Assignment
COM8	23	TxD
	22	RxD
COM7	21	TxD
	20	RxD
	19	GND
COM6	18	TxD
	17	RxD
COM5	16	TxD
	15	RxD
	14	GND
COM4	13	TxD
	12	RxD
COM3	11	TxD
	10	RxD

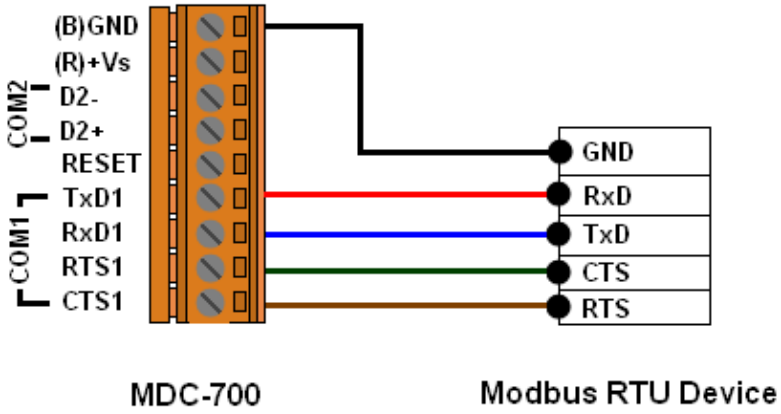
2.4. Wiring Connections

■ RS-232 Wiring

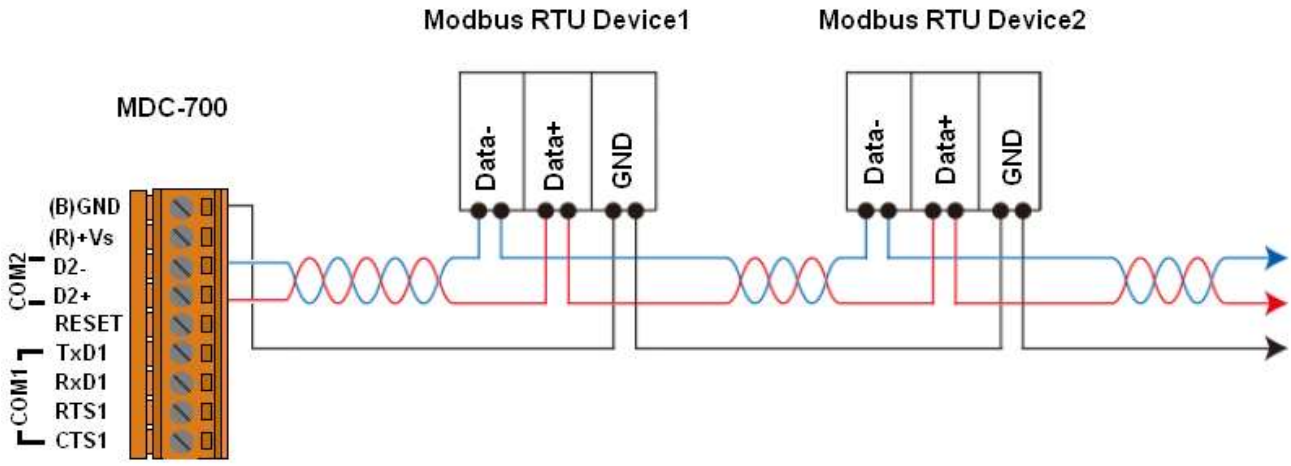
3-wire Connection Wiring



5-wire Connection Wiring

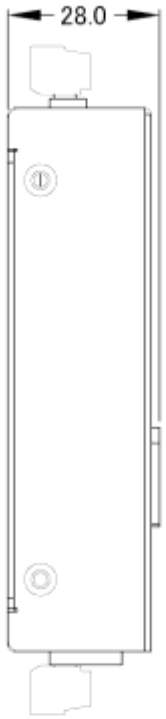


■ RS-485 Wiring

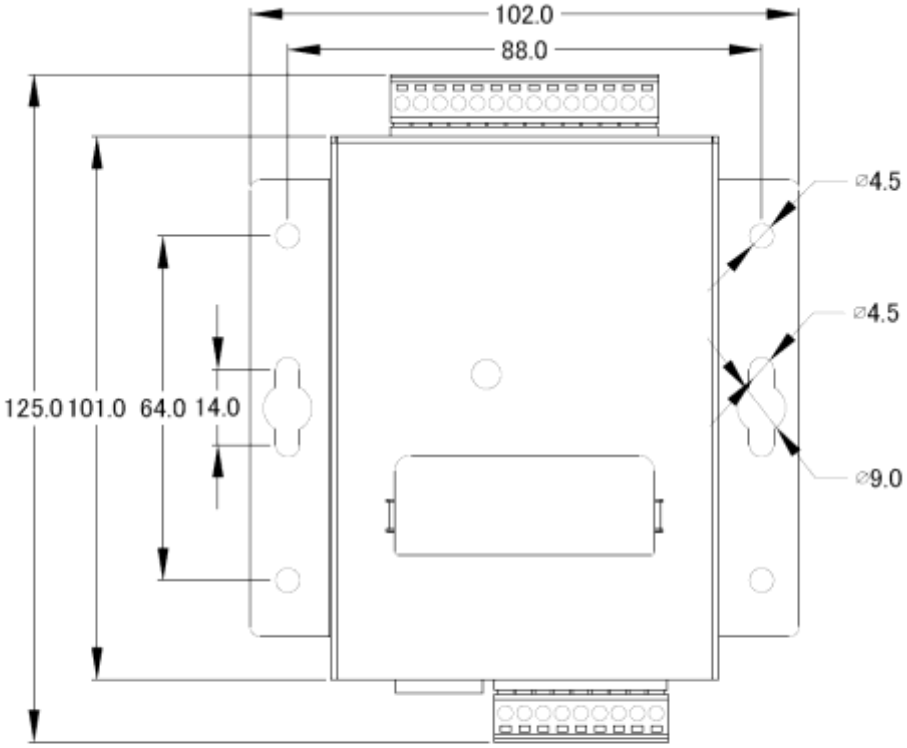


2.5. Dimensions

Unit: mm

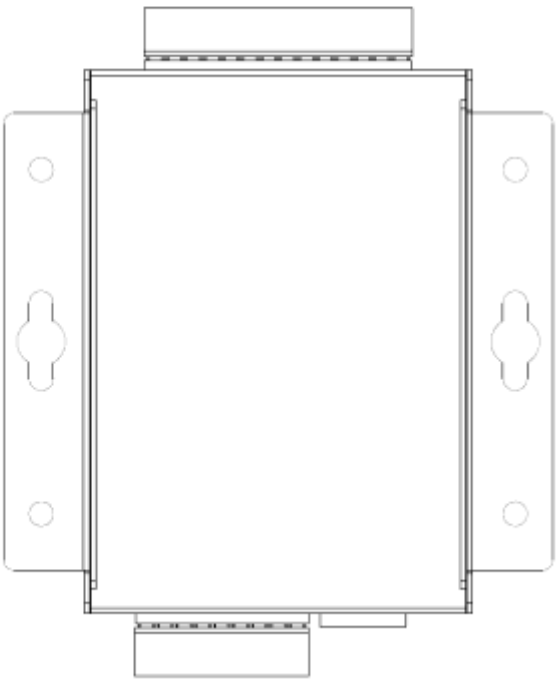


Left Side View

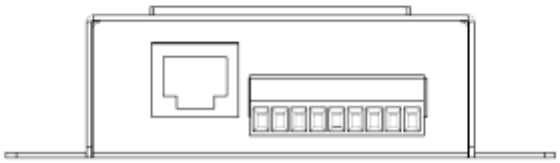


Front View

Top View



Rear View



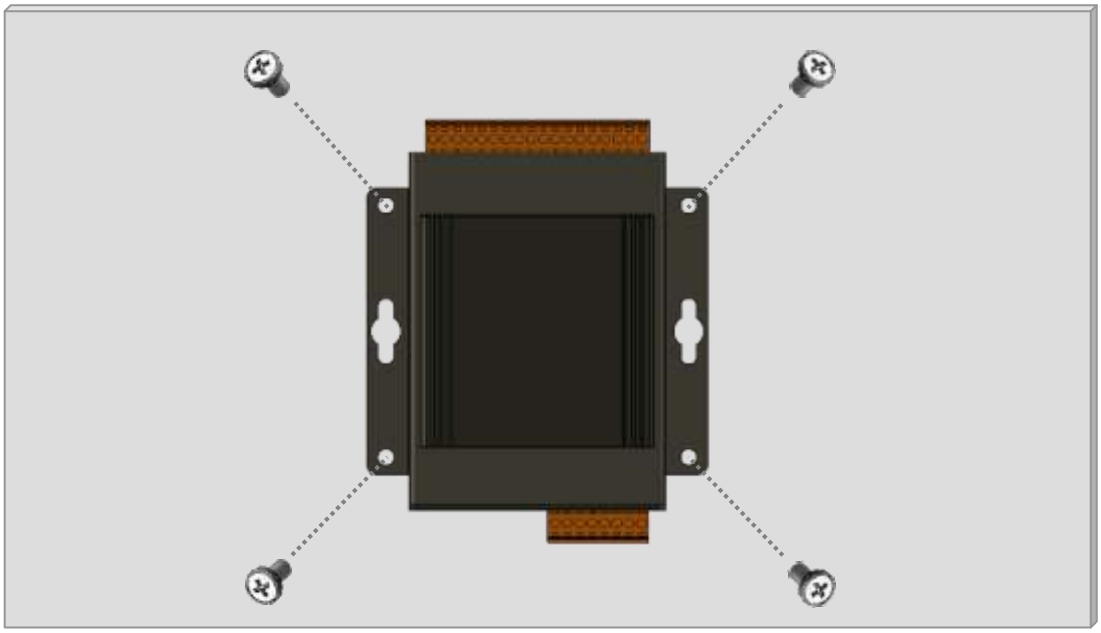
Bottom View

2.6. Mounting the Hardware

■ Wall/Panel mounting

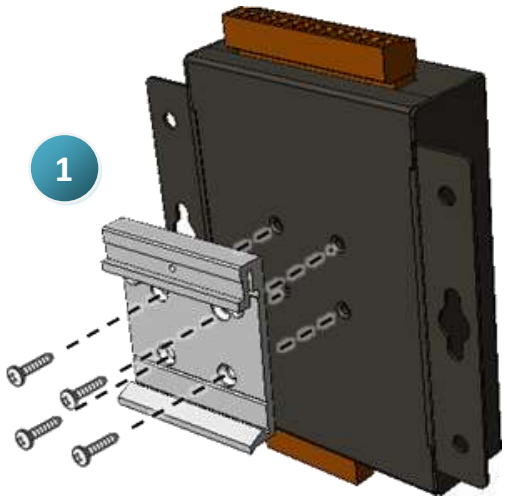
Step 1: Install the four mounting screws into the 4 keyhole mounting holes.

Step 2: Fasten the screws securely.



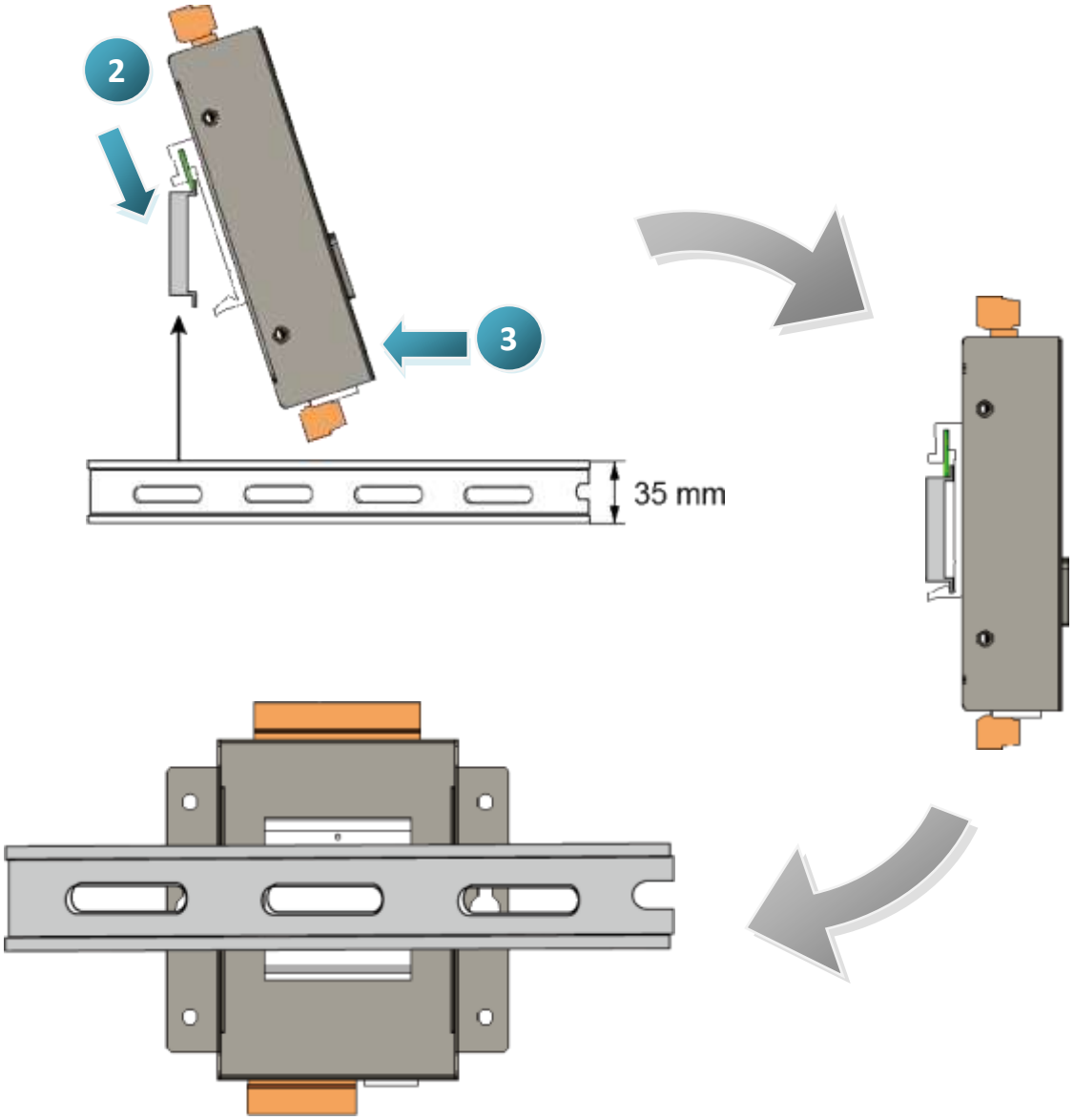
■ DIN Rail mounting

Step 1: Align the screw holes of the DIN-rail clip with the holes on the back side of the module, and then fasten the screws securely.



Step 2: Hook upper tab over upper flange of DIN rail.

Step 3: Tilt the module toward DIN rail until it snaps securely to DIN rail



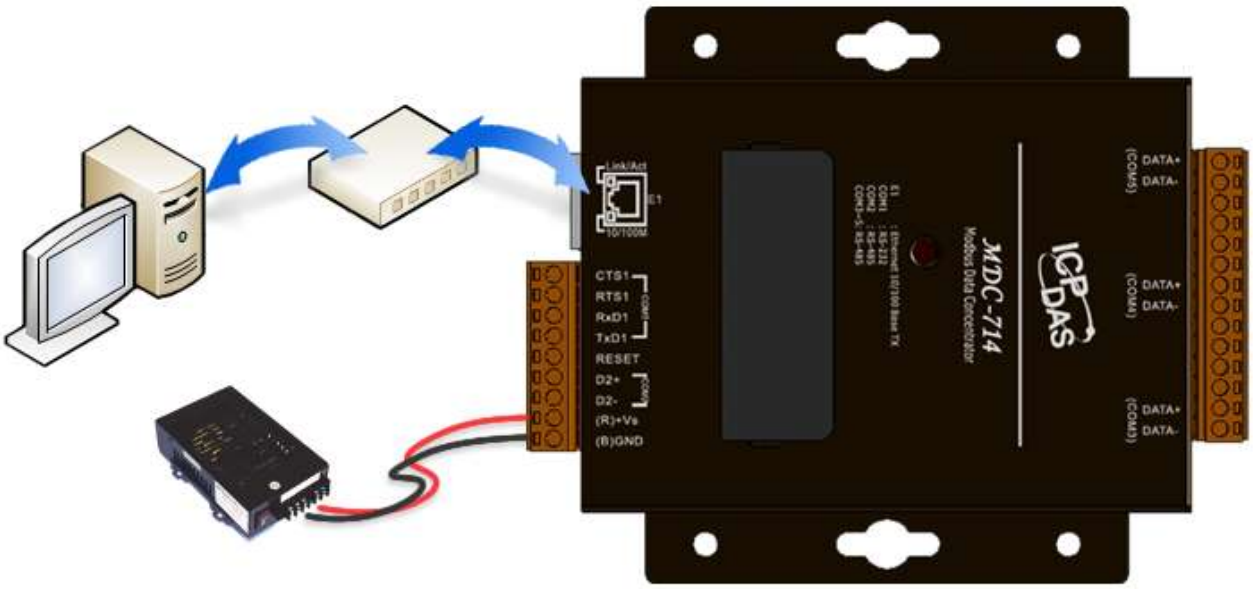
3. Getting Started

A new MDC-700 comes with a default IP address of 192.168.255.1; therefore, a valid IP address should be assigned for the module to join your network. Then you can configure the MDC module via its web interface.

The factory default settings

IP Address	Subnet Mask	Gateway
192.168.255.1	255.255.0.0	192.168.0.1

STEP 1: Connect the MDC module to the same network as your computer and power on all the devices. You can also connect the module to PC directly with an Ethernet cable.



STEP 2: Change the IP address of your computer

If the MDC module is new with default IP address of 192.168.255.1, your pc should pick up an IP address in the range of 192.168.255.2 to 192.168.255.253 that is not in use.

NOTE

1. Details on how to change the IP address of your computer depend upon the type architecture and operating system you are using. Use the Help and Support functionality on your computer and search for "IP Addressing".
2. After the MDC-700 IP settings are completed, the host computer needs to be set back to the original address.

STEP 3: Enter the IP address of the MDC-700 and press Enter.

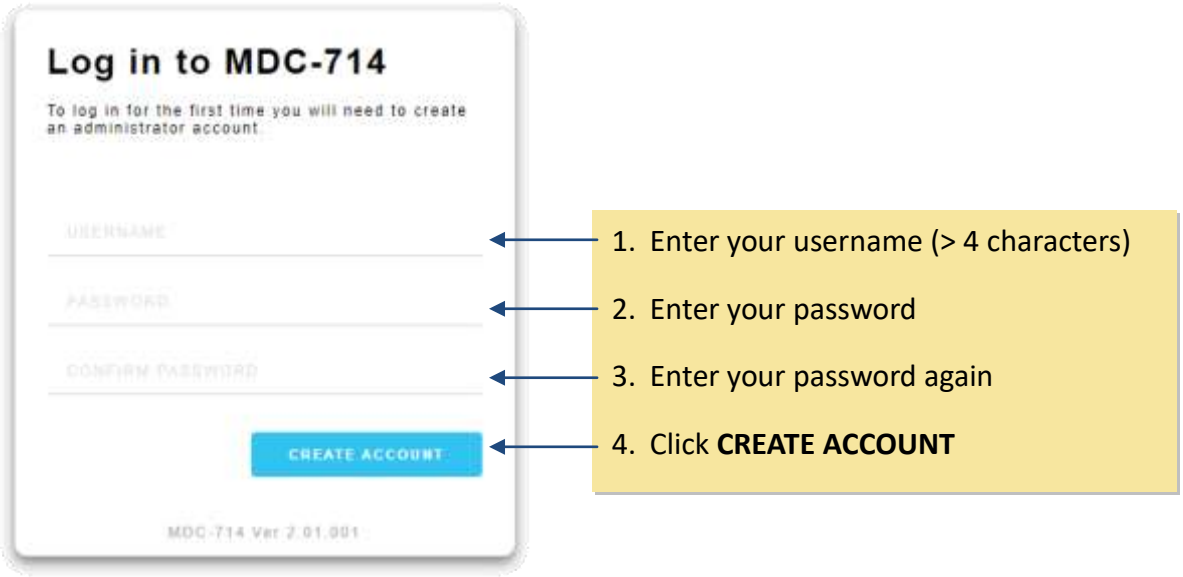
(For example, http://192.168.255.1)

STEP 4: Create your account (for the first time login)

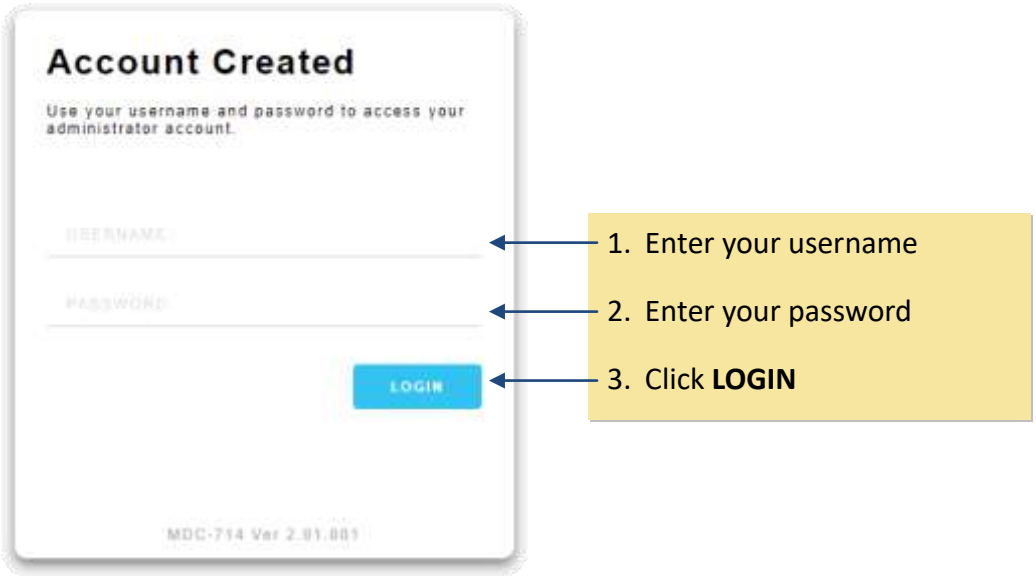
When connecting to the MDC module's web interface for the first time, you will be prompted to create your username and password as an administrator.

The username and password must be at least 4 characters long and may only include:

- Uppercase and lowercase English letters (A–Z, a–z; case-sensitive)
- Numbers (0–9)
- Special characters: dot (.), dash (-), underscore (_), and at sign (@).



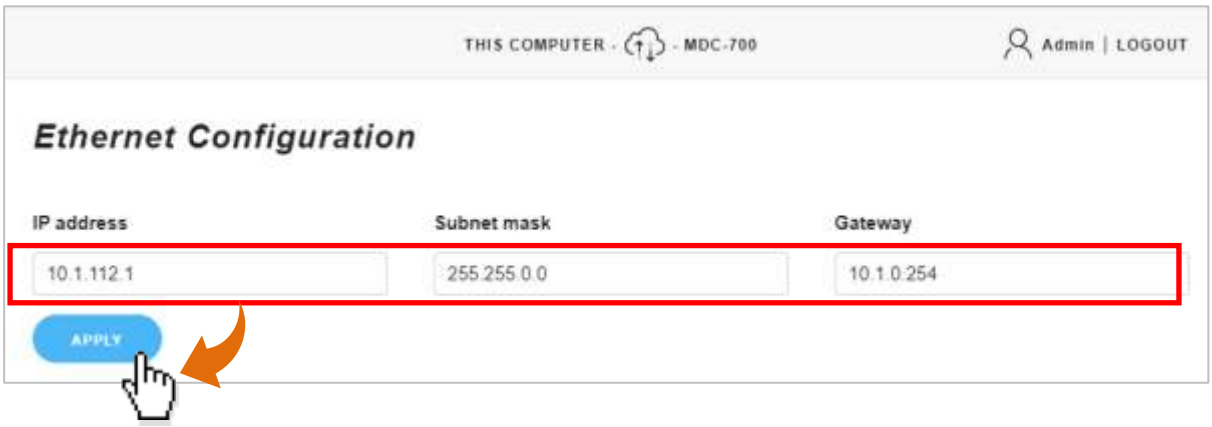
STEP 5: Enter your username and password to log in to the MDC module.



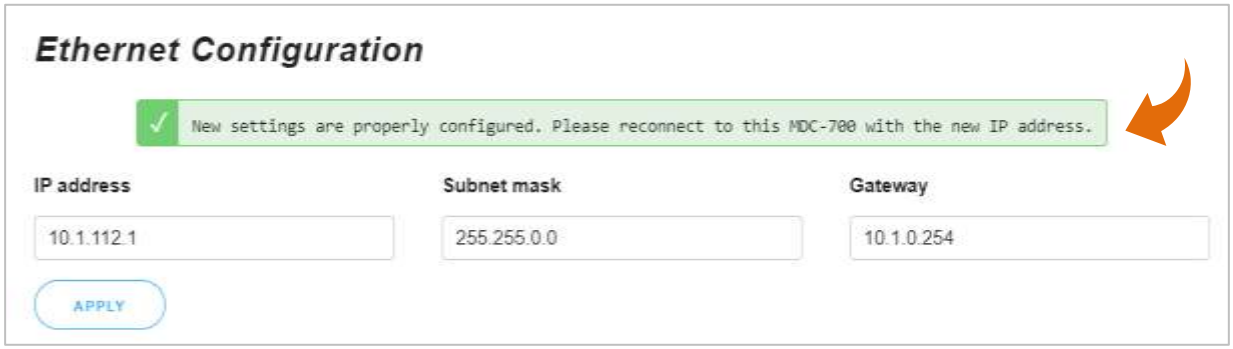
STEP 6: Choose a valid IP address of the network for your MDC-700 module

Scroll down to **Ethernet Configuration** section, input the IP/Subnet mask and Gateway addresses, and then click "**Apply**". Make sure that the IP address you pick is not currently in use by another device on the network.

.

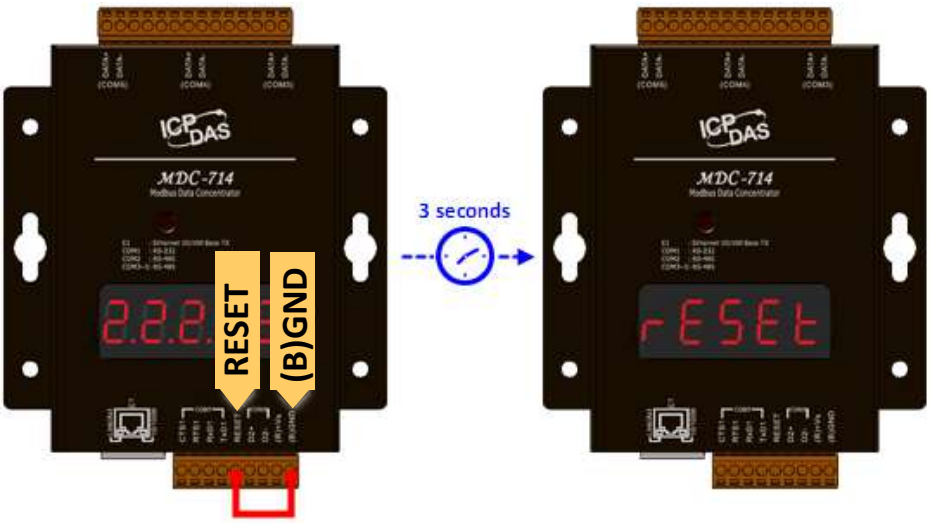


STEP 7: After the success message is displayed, restore the IP address of your computer, log in the MDC again via its new IP address.



NOTE

The IP/Subnet mask/Gateway modified in a MDC-700 can be reset to factory defaults by shorting the RESET pin to GND pin for more than 3 seconds. The LED display will show “RESET” as below and the network settings will be cleared and restored to their factory default values.



4. Configuration

The necessary configuration for Modbus TCP/Modbus RTU communication and polling definition is handled by a single file named “config.csv”. Just follow the easy-to-use format defined in the config.csv file to edit the configuration parameters and import the new file via a simple mouse-click, the data on connected Modbus RTU slave devices can be accessed via Ethernet.

Only the Function code 01/02/03/04 can be used in the config.csv file:

01: Read Coil Status (Read DO)
02: Read Input Status (Read DI)
03: Read Holding Registers (Read AO)
04: Read Input Registers (Read AI)

If you would like to write data to a digital or analog output channel on a Modbus RTU slave device, the output channel needs be mapped with a local register address in the MDC-700 by editing the polling definition with using corresponding read function code (01 or 03). Refer to section **FAQ-Q4** for more detailed steps.

The following section intends to guide you to set up your MDC-700 module. After completing the following steps, you can obtain configuration and other information related to the MDC module and associated slave devices in your browser.

Basic operating procedure

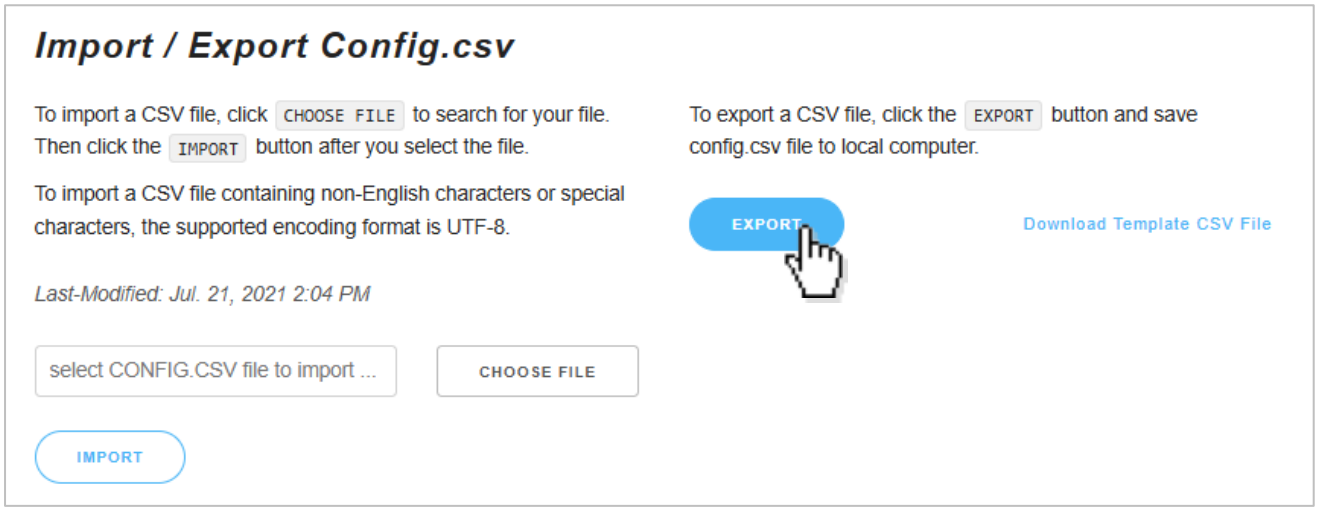
Step1: Export the config.csv file from MDC-700.
Step 2: Edit the config.csv file.
Note that before editing this file, you should confirm the parameter value for all associated slave devices.
Step 3: Import the config.csv file to the MDC-700.

4.1. Exporting and importing config.csv file

Open the web browser and enter the IP address of the MDC-700. Any standard browser such as Mozilla Firefox, Internet Explorer or Google Chrome can be used to access the web interface.

■ Exporting the config.csv file

STEP 1: Scroll down the web page to the "**Import/Export Config.csv**" section.
STEP 2: Click **Export** to export the config.csv file from the MDC-700. The config.csv file will be saved to the download directory specified in your web browser settings.



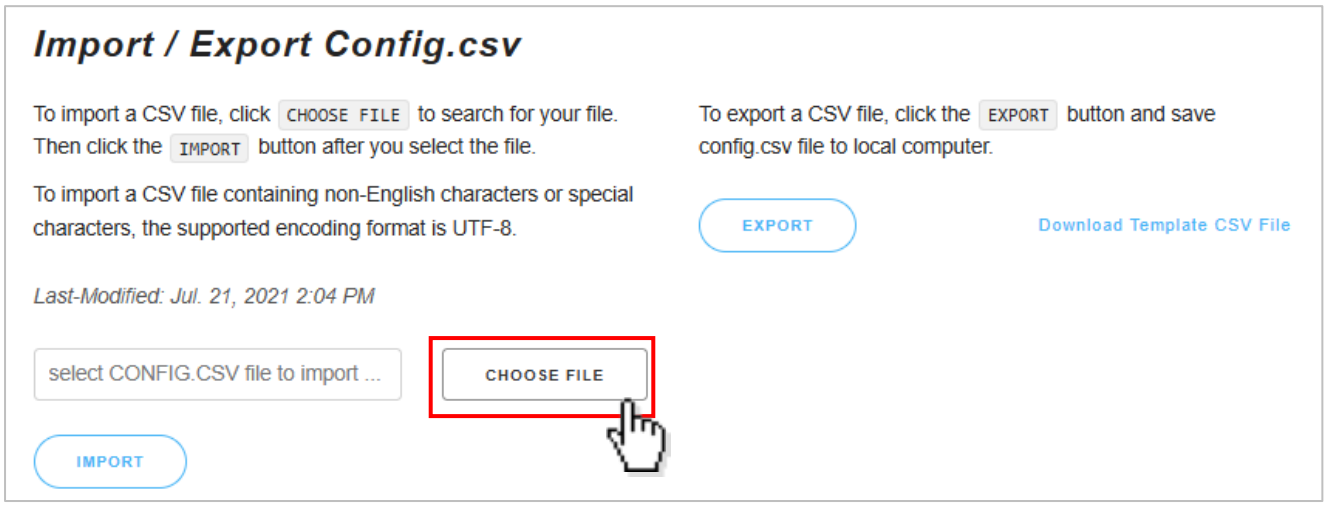
NOTE

- If you have not changed the default IP address in the MDC-700, refer to Section 3 to configure it.

■ Importing the config.csv file

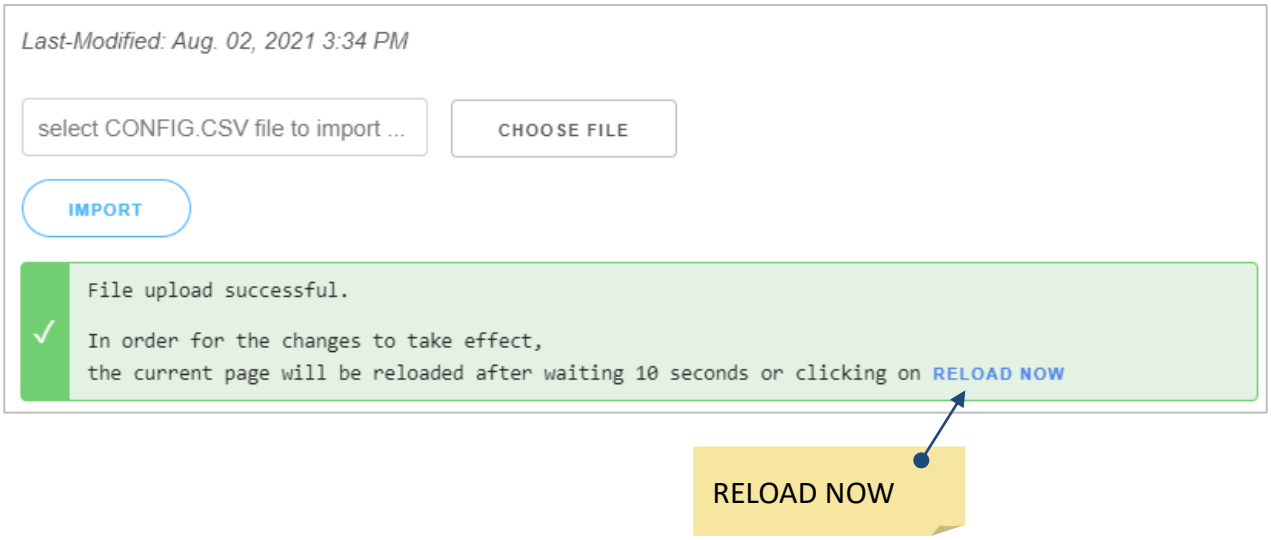
STEP 1: Scroll down the web page to the "**Import/Export Config.csv**" section.

STEP 2: Click **CHOOSE FILE**, then select your config.csv file.



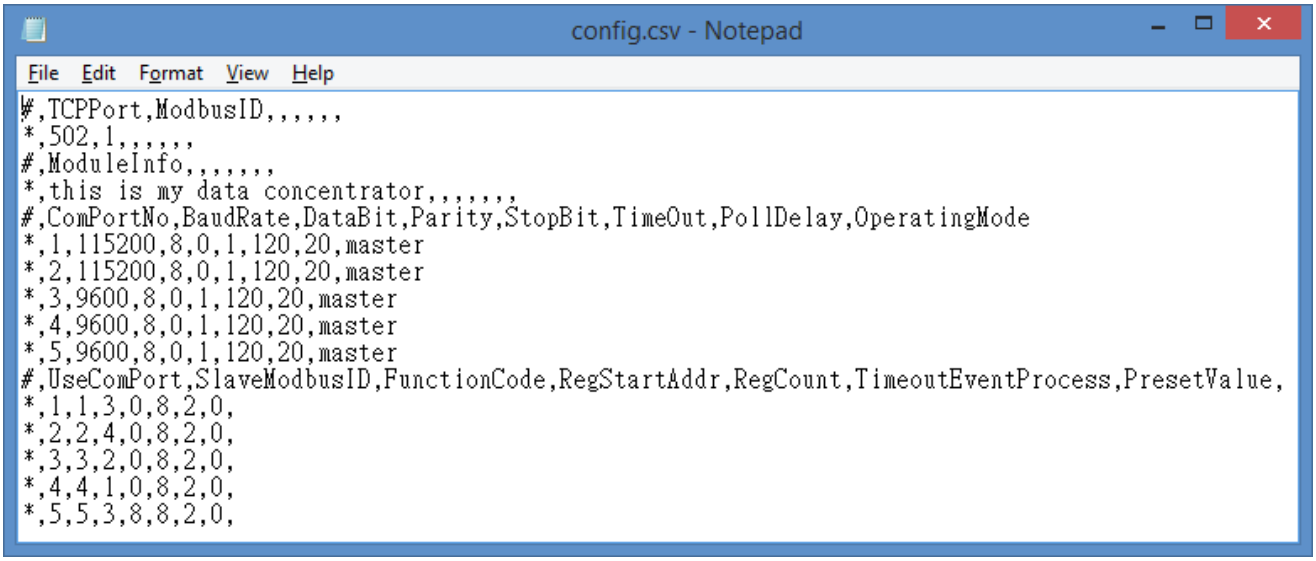
STEP 3: Click **IMPORT** to import the config.csv file into the MDC-700.

After the success message is displayed, waiting 10 seconds for reloading the web page or click **RELOAD NOW** to refresh the page immediately



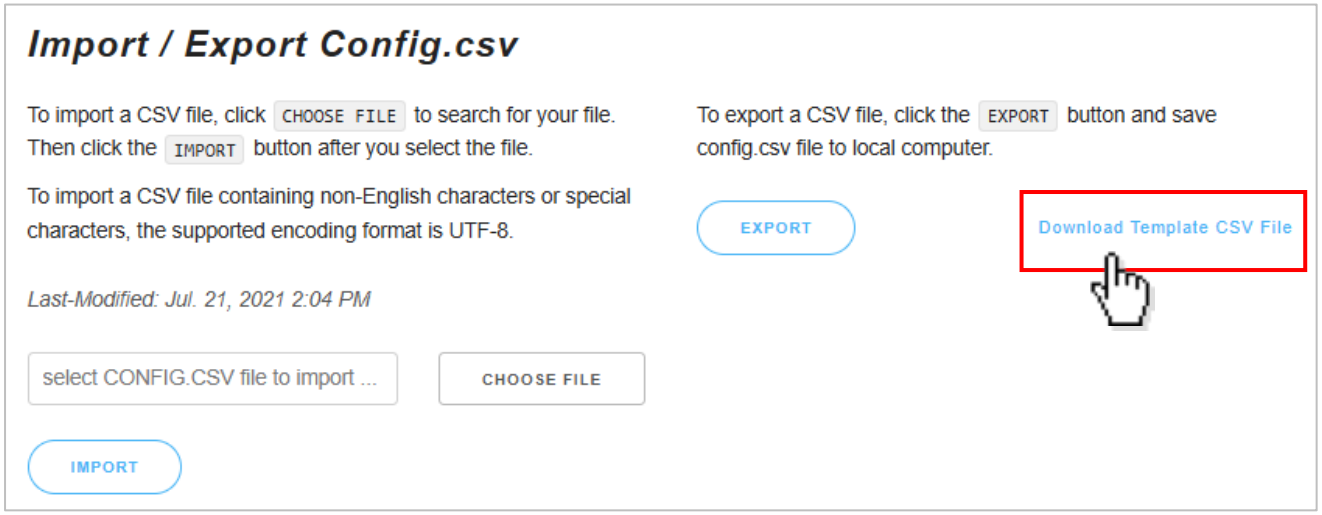
4.2. Editing the config.csv file

The MDC module is configured using a config.csv file, which allows it to communicate with your master and RTU slave devices. This Comma-Separated Values (CSV) file can be viewed and edited in spreadsheet applications such as Microsoft Excel, or in any plain text editor. In CSV format, each field is typically separated by a comma (,). When opened in a text editor, it appears like this:



4.2.1. Downloading the config.csv Template

The MDC-700 web interface provides a download link for a config.csv template in the **Import/Export Config.csv** section, allowing users to conveniently and efficiently modify the template for their own projects.



4.2.2. Editing config.csv

4.2.2. Editing config.csv

When opened in a plain text editor, the structure of the config.csv file looks similar to the example below. The first column represents the attribute markers.

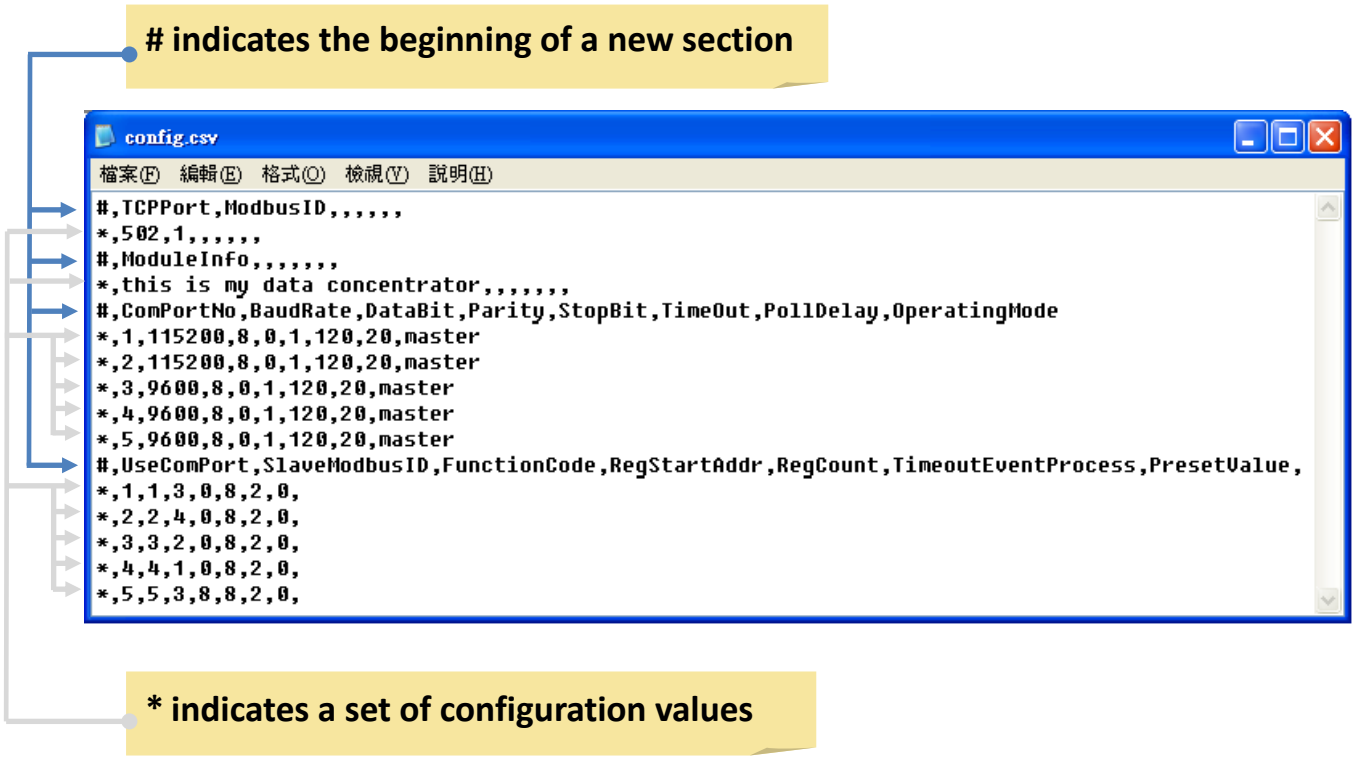
A line starting with “#” indicates the beginning of a new section, followed by the field names for the settings within that section, separated by commas.

A line starting with “*” represents a set of configuration values under that section, with each field also separated by commas.

NOTE

- The field names in each section must match the template (or the example below) exactly and must not be changed.

The config.csv file contains a total of four sections. The fields and their descriptions are presented in spreadsheet format in the following pages.



Modbus Connection

In Modbus Connection section, you can configure the Modbus ID of the MDC-700 and the TCP port number for Modbus TCP communication.

#	TCPPort	ModbusID
*	502	1

TCPPort: Defines the TCP/IP Port number, in the example set to 502. (Default value)

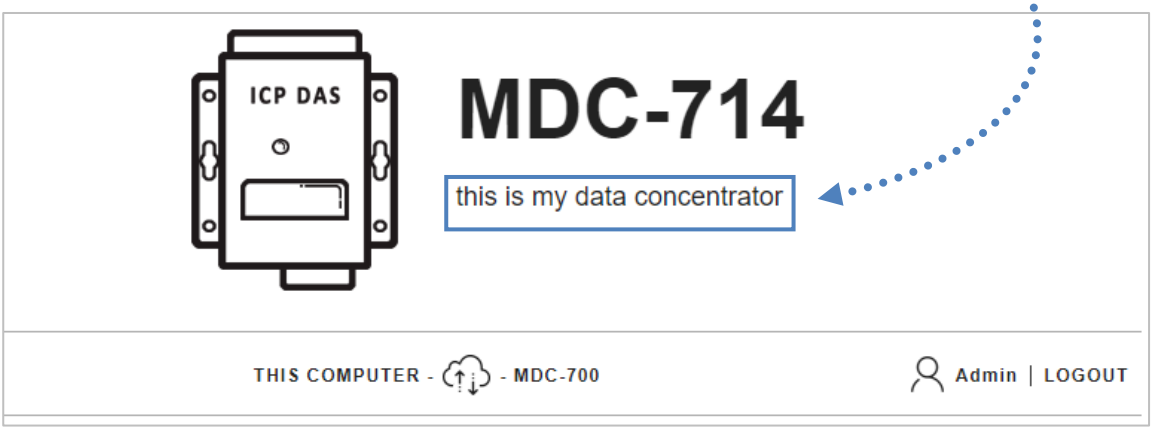
ModbusID: Defines the Modbus ID of the MDC-700, in the example set to 1. (Default value)

Module Information

A longer description or alias can be set for a MDC-700 in this Module Information section. It will be displayed on the main page of the MDC-700, and can be used to identify one MDC-700 from the others.

#	ModuleInfo
*	this is my data concentrator

ModuleInfo: Defines the auxiliary description for the MDC module.
The string constant has a maximum length of 32 characters.



■ COM Port Configuration

The COM Port Configuration is used to configure the parameters for Modbus communication connection between the MDC-700 and Modbus RTU slave devices.

NOTE Only one set of configuration settings is allowed for each COM port.

#	ComPortNo	BaudRate	DataBit	Parity	StopBit	Timeout	PollDelay	OperatingMode
*	1	115200	8	0	1	100	20	Master
*	2	115200	8	0	1	100	20	Master
*	3	115200	8	0	1	100	20	Master
*	4	115200	8	0	1	100	20	Master
*	5	115200	8	0	1	100	20	Master

The connection configuration for a COM port consists of 8 parameters defined as follows.

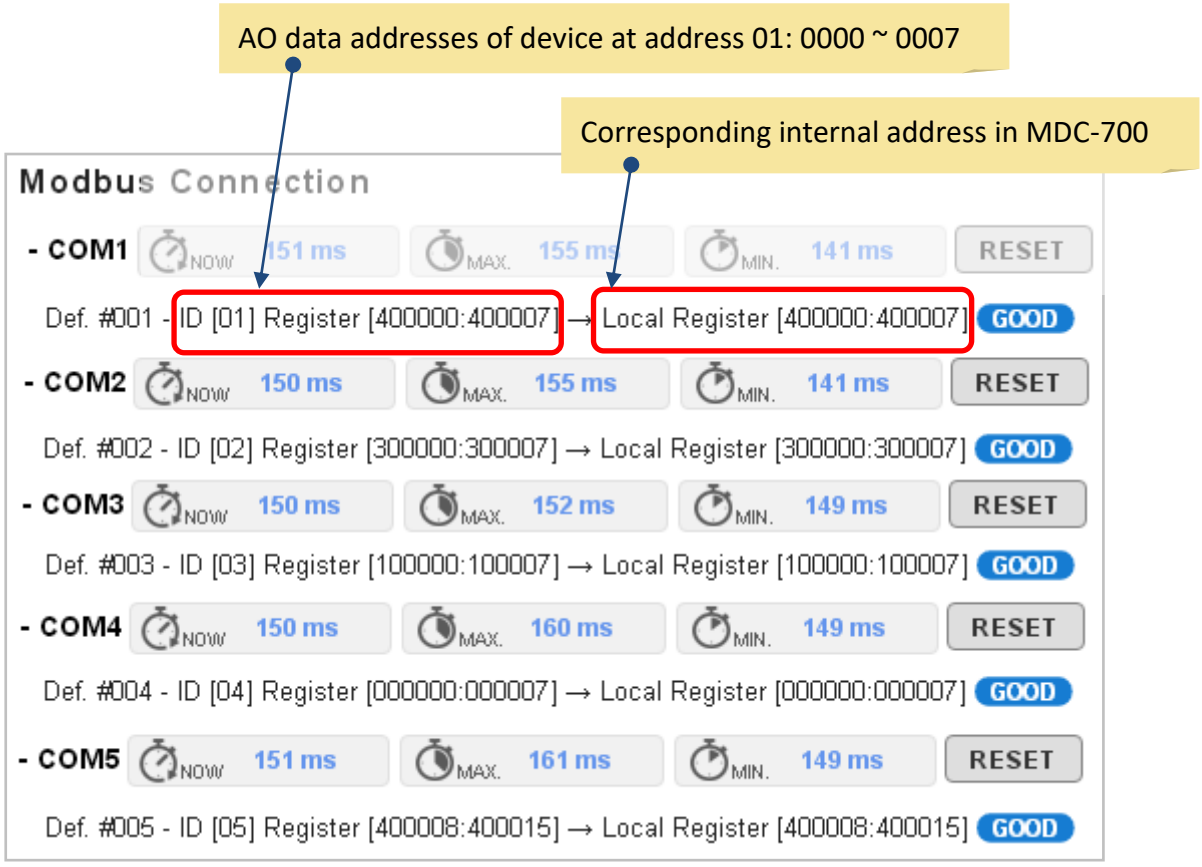
ComPortNo	Specifies the COM port number in MDC module. The COM port number can be 1 or 2 for MDC-711, 1 ~ 5 for MDC-714 and MDC-741 and so on.
BaudRate	Defines the transmission speed between the MDC module and the RTU slave devices. The BaudRate can be set to 1200/ 2400/ 4800/ 9600/ 19200/ 34800/ 57600/ 115200 (bps) depending on the RTU slave device being used.
DataBit	Defines the number of data bits in each character. It is fixed to 8 and the RTU slave devices need be set to 8-bit data, too.
Parity	Defines the Parity bit. The parity bit can be set to 0 (none), 1 (even) or 2 (odd).
StopBit	Defines the Stop bits. The stop bit can be set to 1 (1 stop bit) or 2 (2 stop bits).
Timeout	Defines the period of time that the MDC module will wait for a response from the RTU slave device. The available range is from 50 to 6000 (ms).
PollDelay	Defines the Poll Delay between each scan for Modbus RTU communication. The available range is from 20 to 6000 (ms).
OperatingMode	Defines the operating mode. - Master: the com port is used to connect Modbus RTU slave devices. The MDC-700 is acting as a master to send requests to slave devices. - Slave: the com port is used to connect Modbus RTU master devices. The master devices can read/write data from/to the MDC-700.

Polling Definition

The Polling Definition specifies the sequence of Modbus commands that the MDC-700 will use to request data from Modbus slave devices. The MDC-700 sends these requests in the user-defined order and organizes the retrieved data into continuous address spaces based on the data type - Analog Input (AI), Analog Output (AO), Digital Input (DI), and Digital Output (DO).

As long as the starting address and data length within the MDC-700 are known, the host computer can efficiently read from or write to multiple devices in a single operation.

The MDC-700's web interface will display how the data from each command maps to the internal register addresses, as illustrated in the diagram below.



Before attempting to configure the parameters for the Polling Definition, be sure to check the COM port number that the target device is connected to, the Modbus ID setting for the target device, and the function code, starting address, and the quantity for reading data.

#	UseComPort	SlaveModbusID	FunctionCode	RegStart Addr	RegCount	Timeout EventProcess	Preset Value
*	1	1	3	0	8	2	0
*	2	2	4	0	8	2	0
*	3	3	2	0	8	2	0
*	4	4	1	0	8	2	0
*	5	5	3	8	8	2	0

Each Polling Definition consists of 8 parameters listed as below:

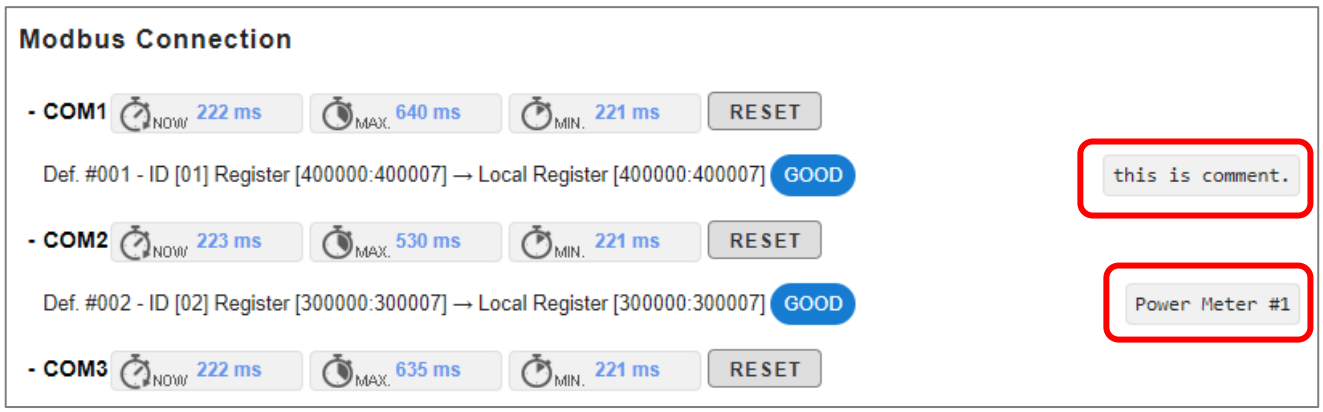
#	Defines the type for a polling definition. In the MDC-700, it provides three types: "*": Asterisk symbol means that this is a valid polling definition. The MDC-700 will assign local register for data defined in the definition and save the polled data to the mapping local register. "-": Minus sign means that this is a disabled polling definition. The MDC-700 will assign local register for data defined in the definition but will not poll data. "": Empty means that this is a null polling definition. The MDC-700 will neither assign local register for data defined in the definition nor poll data.
UseComPort	Defines the COM port number to which the slave device is connected. The COM port number is from 1 to the total number of COM ports on the MDC-700.
SlaveModbusID	Defines the identification of the remote slave. The valid range is from 1 to 255.
FunctionCode	Defines the request function code. A valid code can be 1 (Read DO), 2 (Read DI), 3 (Read AO) or 4 (Read AI) depending on the I/O features of the slave device.
RegStartAddr	Defines the starting address, i.e. the address of the first register specified. The available range is from 0 to 65535.
RegCount	Defines the quantity of registers to be read. The available range is from 1 to 125.
Timeout EventProcess	Defines which data will be read while a timeout error is occurred: 0: the exception code 1: the latest data before the timeout error occurred 2: a preset value
PresetValue	Defines the preset value applied when the ***TimeoutEventProcess*** is set to 2.

NOTE

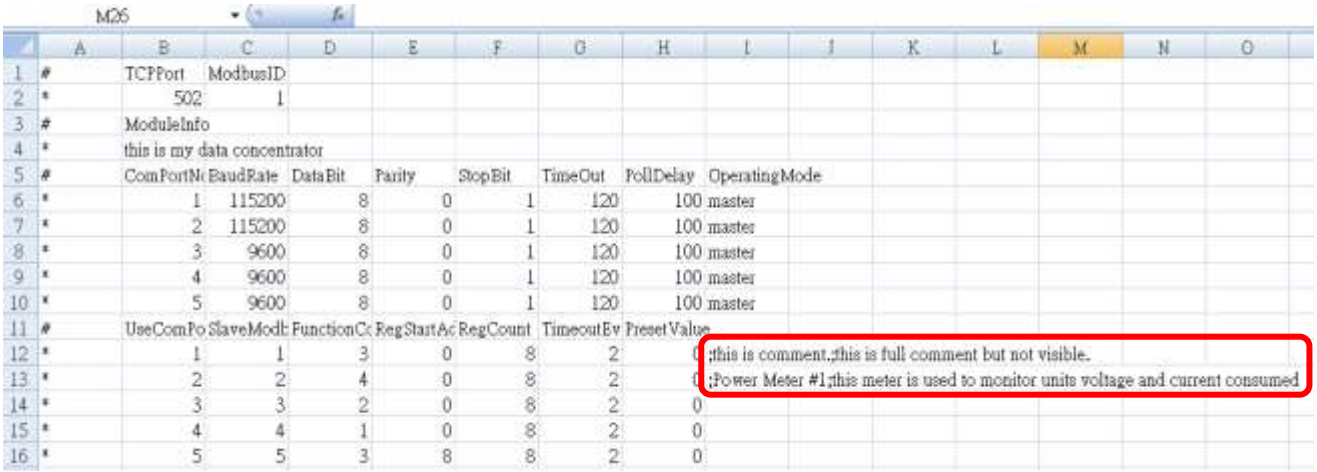
- The maximum number of all the polling definitions is 250.
- The MDC-700 provides 9600 internal Modbus registers each table (DI/DO/AI/AO) to hold data collected from the RTU slave devices.
- The Modbus ID for the MDC-700 is defined in Modbus Connection section.
- By setting different types for a polling definition to retain register space mapped for specific devices, or to release those space mapped but reserve the definition, the main program on the Modbus master device can be applied in different applications where users would like to change or stop some devices without modification or with minimum level of modification.
- The ***TimeoutEventProcess*** and the ***PresetValue*** parameters are only available to firmware version 1.08 and later. If a config.csv file for firmware version 1.06 or prior is imported to a MDC-700 with firmware version 1.08 or later, the ***TimeoutEventProcess*** parameter is auto set to 2, and the ***PresetValue*** parameter is set to 0.

■ Displaying Simple Comments for Polling Definition

After firmware 2.00.001 released in 2021, users can annotate polling definitions by adding comments in the field after each definition.



In spreadsheet software



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	#	TCPPort	ModbusID												
2	*	502	1												
3	#	ModuleInfo													
4	*	this is my data concentrator													
5	#	ComPortNo	BaudRate	DataBit	Parity	StopBit	TimeOut	PollDelay	OperatingMode						
6	*	1	115200	8	0	1	120	100	master						
7	*	2	115200	8	0	1	120	100	master						
8	*	3	9600	8	0	1	120	100	master						
9	*	4	9600	8	0	1	120	100	master						
10	*	5	9600	8	0	1	120	100	master						
11	#	UseComPo	SlaveModb	FunctionCo	RegStartAc	RegCount	TimeoutEv	PresetValue							
12	*	1	1	3	0	8	2	0	;this is comment.;this is full comment but not visible.						
13	*	2	2	4	0	8	2	0	;Power Meter #1;this meter is used to monitor units voltage and current consumed						
14	*	3	3	2	0	8	2	0							
15	*	4	4	1	0	8	2	0							
16	*	5	5	3	8	8	2	0							

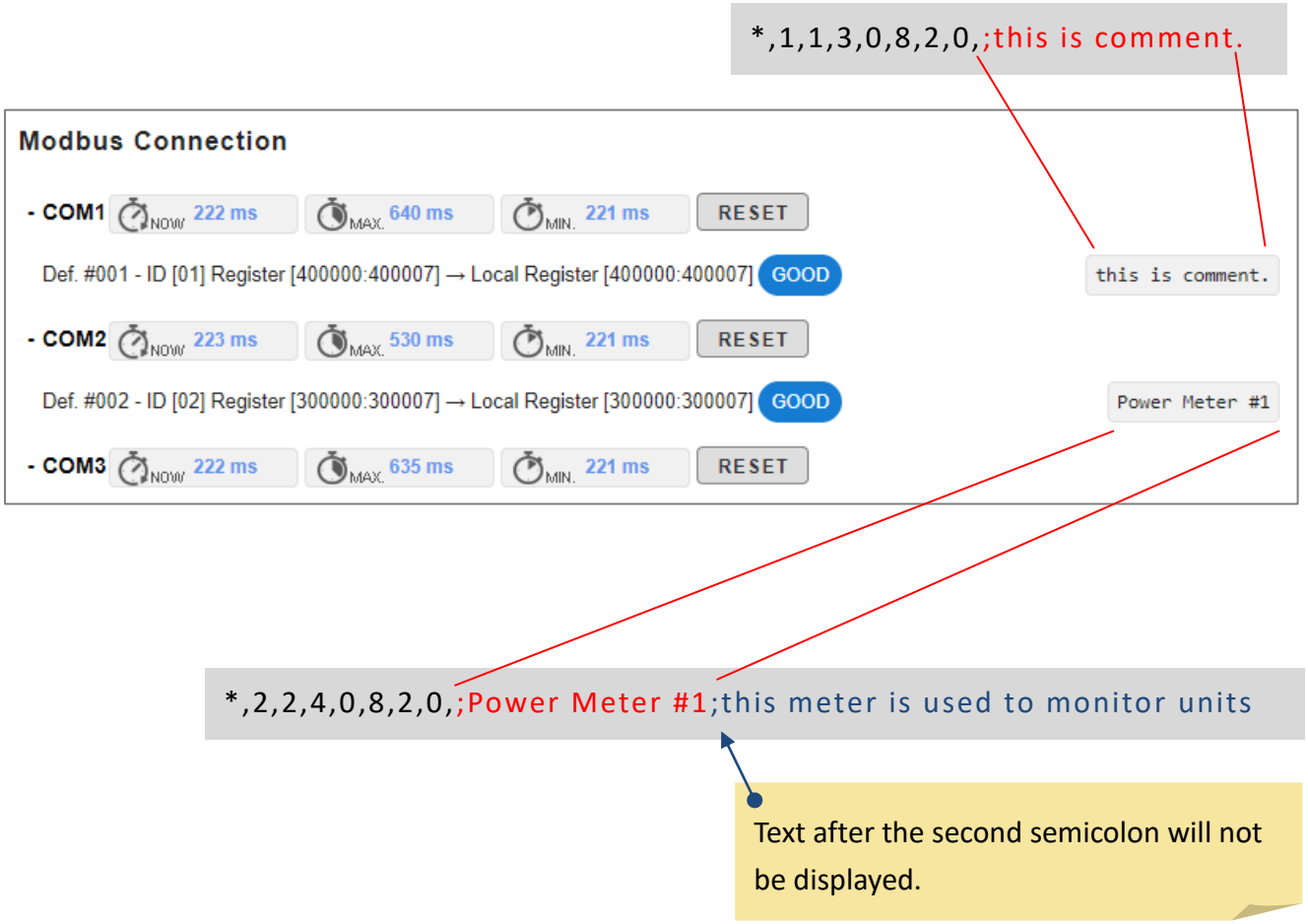
In text editor

```
#,TCPPort,ModbusID,,,,,
*,502,1,,,,,
#,ModuleInfo,,,,,,
*,this is my data concentrator,,,,,,
#,ComPortNo,BaudRate,DataBit,Parity,StopBit,TimeOut,PollDelay,OperatingMode
*,1,115200,8,0,1,120,100,master
*,2,115200,8,0,1,120,100,master
*,3,9600,8,0,1,120,100,master
*,4,9600,8,0,1,120,100,master
*,5,9600,8,0,1,120,100,master
#,UseComPort,SlaveModbusID,FunctionCode,RegStartAddr,RegCount,TimeoutEventProcess,PresetValue,
*,1,1,3,0,8,2,0,;this is comment.;this is full comment but not visible.
*,2,2,4,0,8,2,0,;Power Meter #1;this meter is used to monitor units voltage and current consumed
*,3,3,2,0,8,2,0,
*,4,4,1,0,8,2,0,
*,5,5,3,8,8,2,0,
```

Editing rules for displaying comments of Polling Definitions on MDC-700 web interface

You can add comments for each polling definition by entering text in the field following the Preset Value column. Up to 24 characters, including spaces, can be displayed. To support multilingual characters, ensure the config.csv file is saved with UTF-8 encoding.

1. Text beginning with a semicolon (;) will be displayed on the MDC-700 web interface after the corresponding polling definition.
2. Text that does not begin with a semicolon will not be shown on the web interface.
3. Any text following the second semicolon will be treated as an internal comment and will not appear on the web interface.



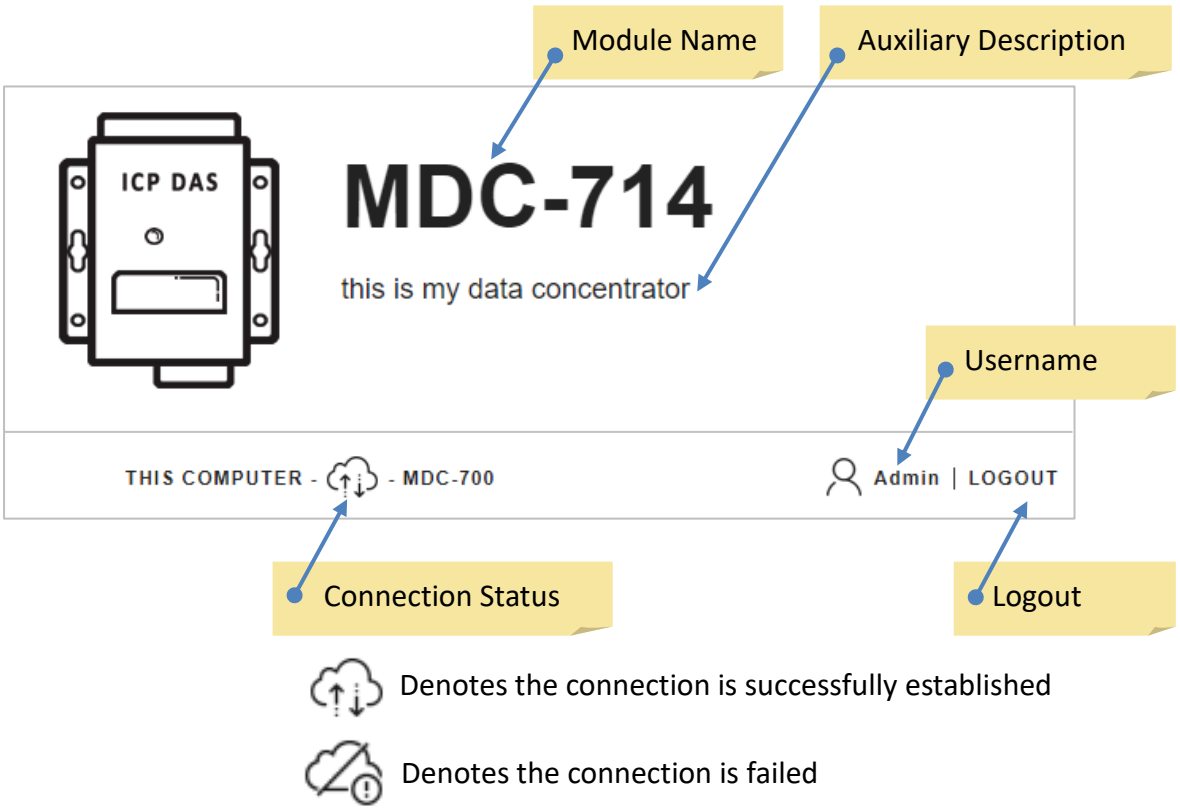
5. MDC-700 Web Interface

Go to the web interface at http://xxx.xxx.xxx.xxx, where xxx.xxx.xxx.xxx is the IP address in your MDC-700. Any standard browser such as Mozilla Firefox, Internet Explorer or Google Chrome can be used to access the MDC-700.

The MDC-700 web interface includes the following sections:

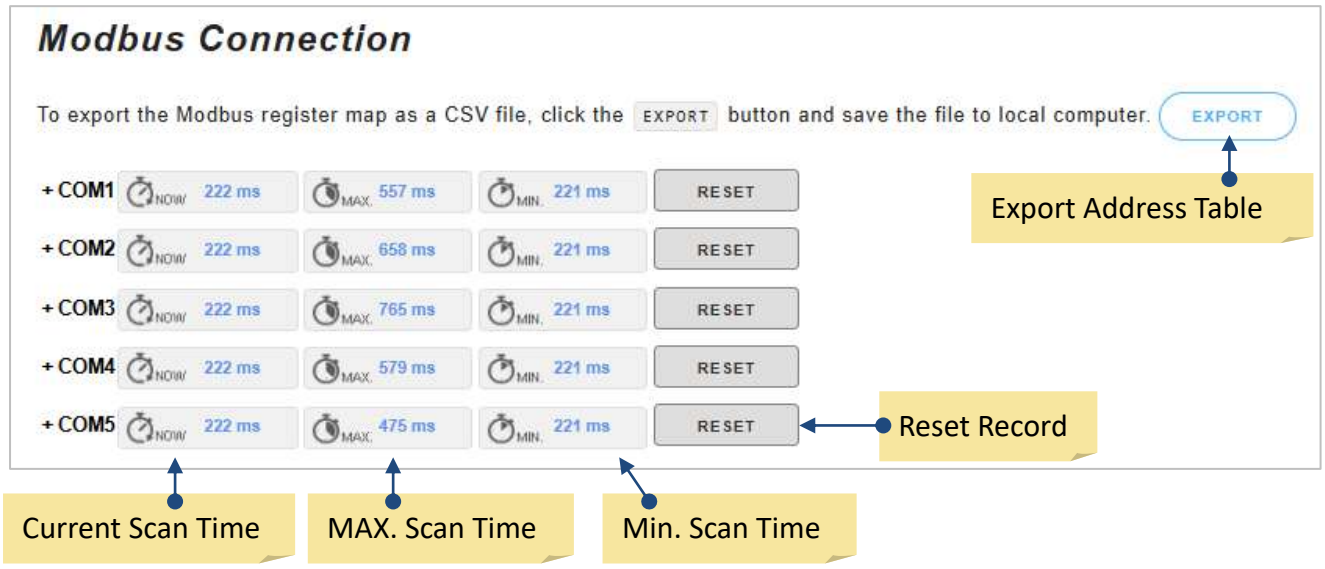
1. Connection status between the device and the MDC-700,
2. The connection information for each polling definition,
3. Export address mapping table,
4. COM port configuration in using. (Same as settings defined in the config.csv file),
5. Ethernet configuration – IP configuration and IP allowlist,
6. Authentication / User Management,
7. Importing/exporting the config.csv file and file validation,
8. OS version, firmware version and MAC address information

■ Connection status between your device and the MDC-700

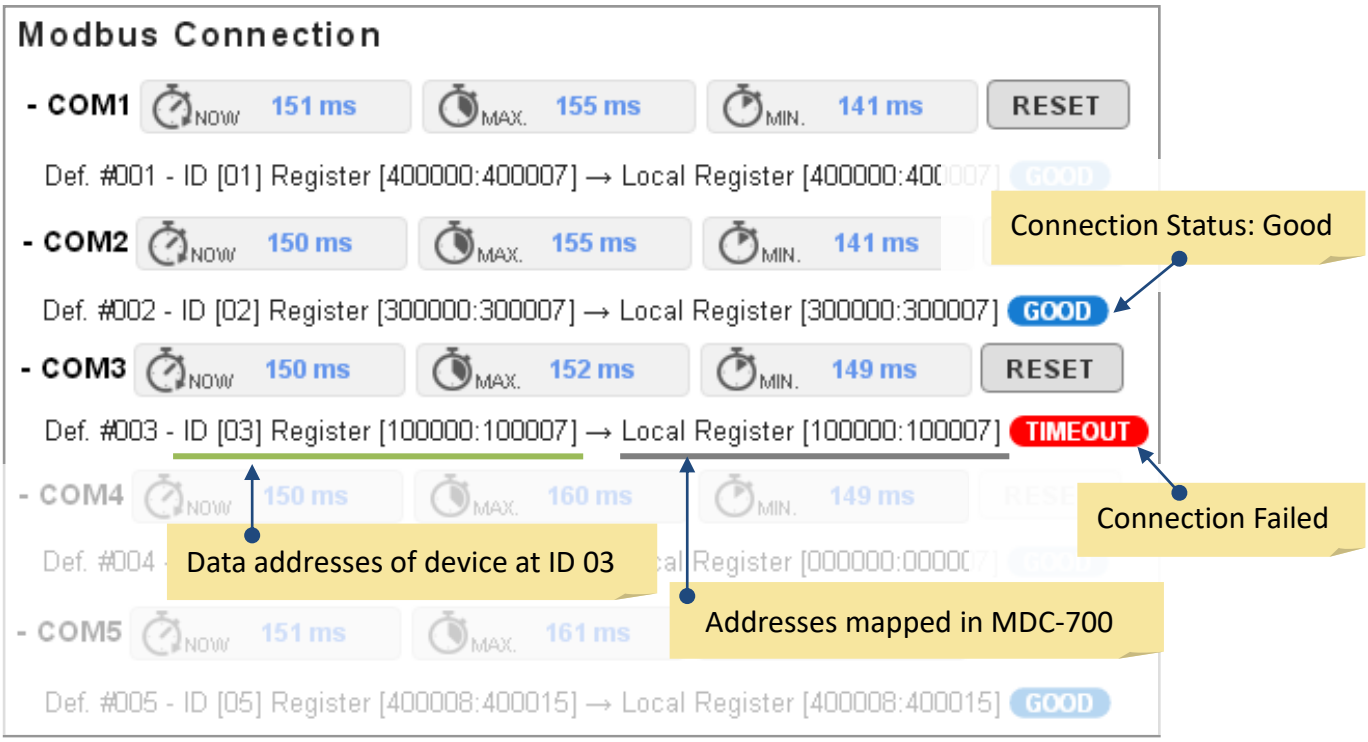


■ Modbus Connection

In the Modbus Connection section, it provides the scan time information for each COM port (available in firmware 1.08 and later). The Master device can refer to the scan time to extend or shorten the time interval for each requesting data command.

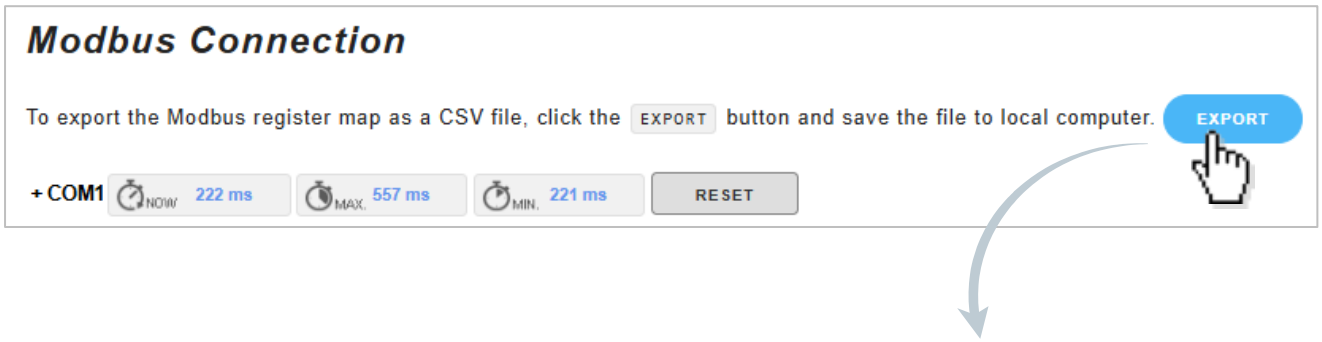


Expand the polling definitions by clicking [**+COMn**] item, information including the polling definition number, SlaveModbusID, Starting Address of Register and Count of Register on both slave client and MDC-700, and the connection status are provided. The content of this section may vary depending on the setting.



■ Address Table Export

Starting from firmware version V2.02, a new feature has been added to export the Modbus register address table. In the **Modbus Connection** section, clicking the **EXPORT** button allows users to export the address mapping table of all the polling definition. The register address table lists the address of each data in the order of definitions.



MDC-714	this is my data concentrator					
Last-Modified	Jul. 21, 2021 2:04 PM					
No.	COM Port	Modbus Sl	Register Address of the Slave	Mapped Register in MDC	Quantity of	Description
1	1	1	400000:400007	400000:400007	8	this is comment
	1	1	400000	400000	1	
	1	1	400001	400001	1	
		1	400002	400002	1	
		1	400003	400003	1	
	1	1	400004	400004	1	
	1	1	400005	400005	1	
	1	1	400006	400006	1	
	1	1	400007	400007	1	
2	2	2	300000:300007	300000:300007	8	Power Meter #1
	2	2	300000	300000	1	
	2	2	300001	300001	1	

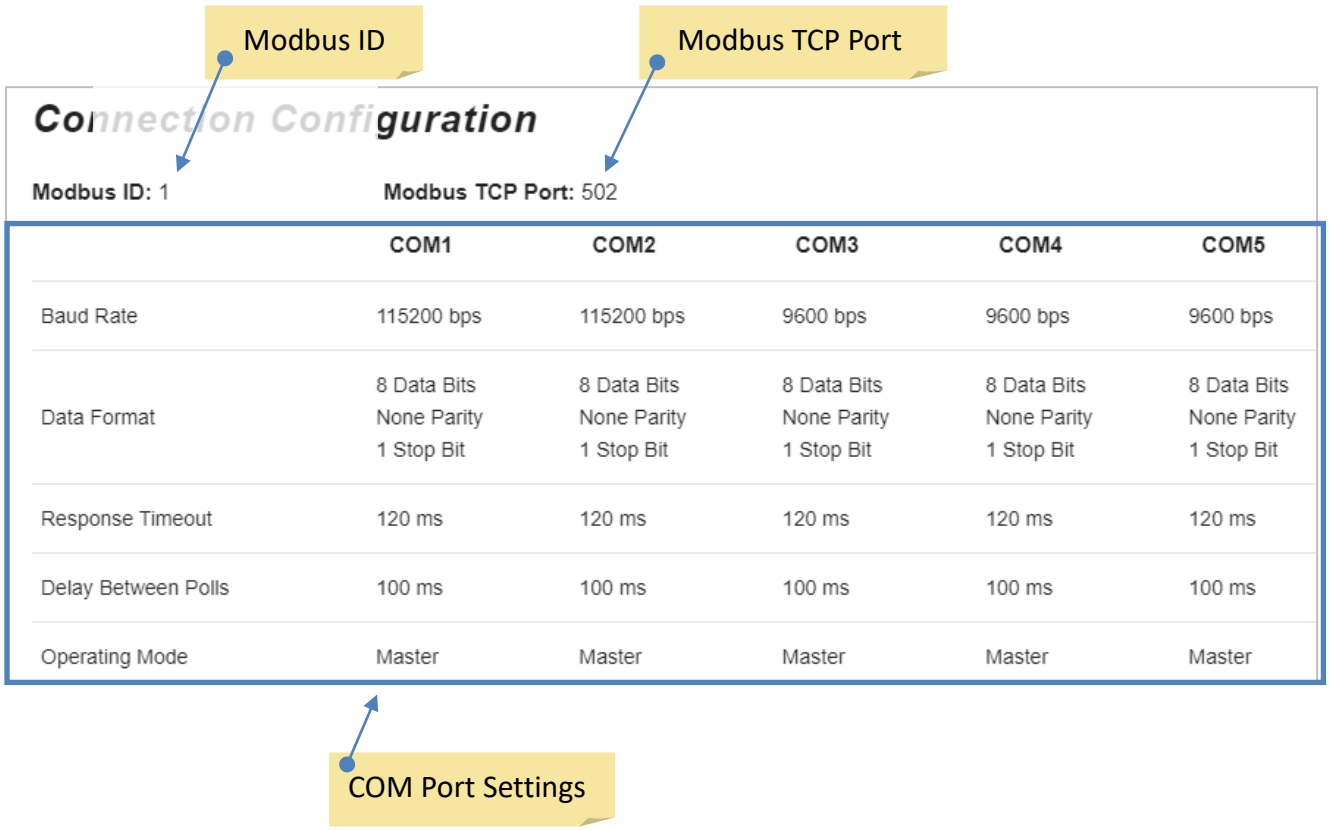
Definition number

Addresses in device

Addresses mapped in MDC

■ Connection Configuration

The **Connection Configuration** section provides the configuration information including Modbus ID, Modbus TCP port on the MDC-700, and Baud Rate. Data Format, Response Timeout, Delay Between Polls, Operation Mode settings for each COM port.

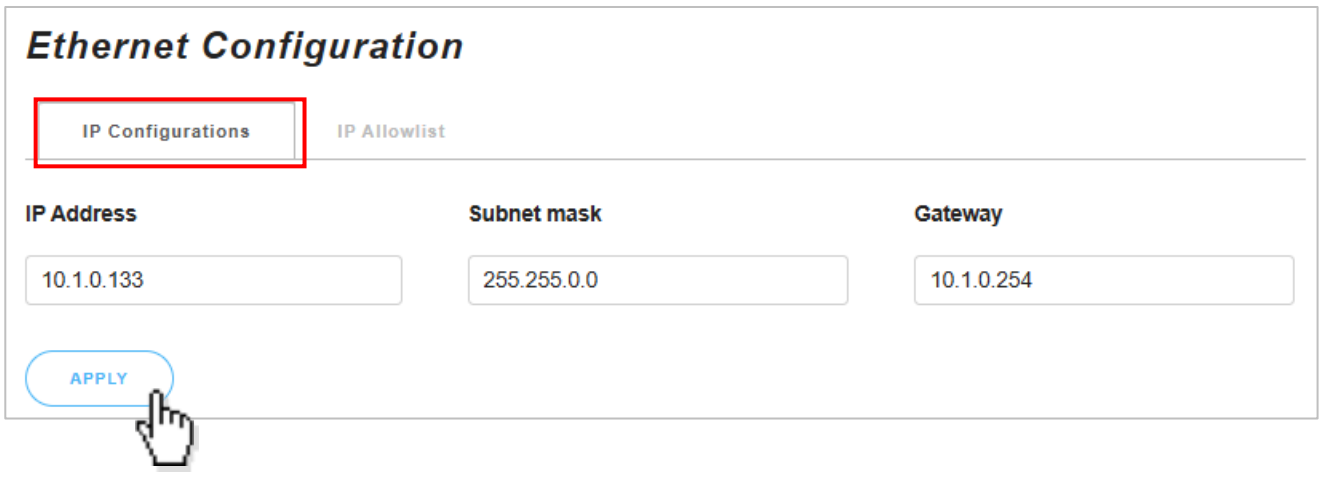


	COM1	COM2	COM3	COM4	COM5
Baud Rate	115200 bps	115200 bps	9600 bps	9600 bps	9600 bps
Data Format	8 Data Bits None Parity 1 Stop Bit	8 Data Bits None Parity 1 Stop Bit	8 Data Bits None Parity 1 Stop Bit	8 Data Bits None Parity 1 Stop Bit	8 Data Bits None Parity 1 Stop Bit
Response Timeout	120 ms	120 ms	120 ms	120 ms	120 ms
Delay Between Polls	100 ms	100 ms	100 ms	100 ms	100 ms
Operating Mode	Master	Master	Master	Master	Master

■ Ethernet Configuration - IP

The Ethernet Configuration section contains the IP settings of the MDC-700 and the IP allowlist settings for allowed access.

To change the Ethernet parameters, you just need to input the valid IP, Subnet mask and Gateway addresses and then click **APPLY**.

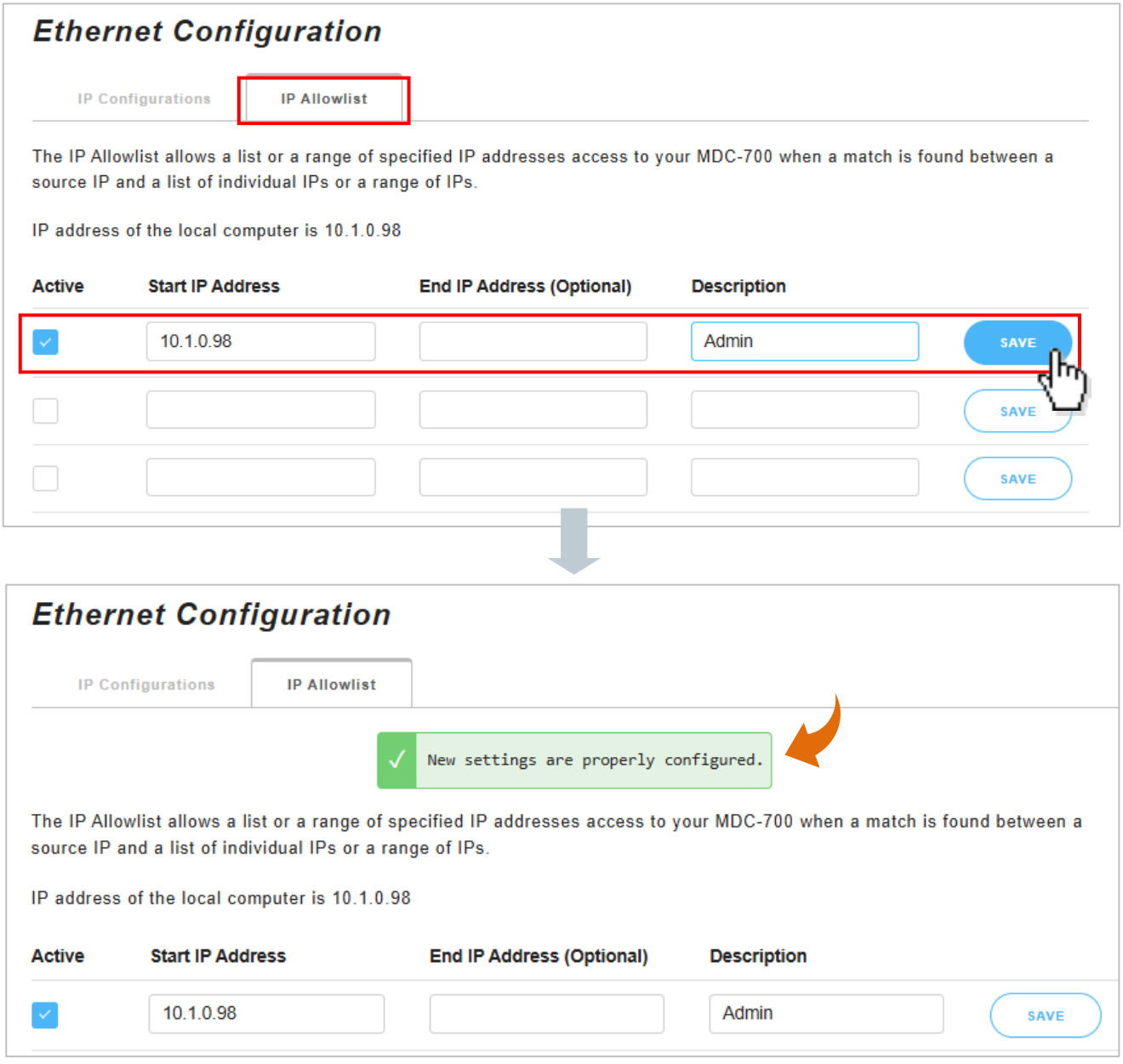


■ Ethernet Configuration - IP Allowlist

The IP Allowlist is used to restrict access to the MDC-700 by limiting connections to trusted IP addresses only. It helps prevent unauthorized access from distrusted IPs. The IP Allowlist provides up to 3 configurable entries, each of which can define a single IP address or a specific IP range. When any of the entries is enabled (by checking Active), only the IP addresses listed in the allowlist will be permitted to access the MDC-700's interface and retrieve data via the Modbus TCP protocol.

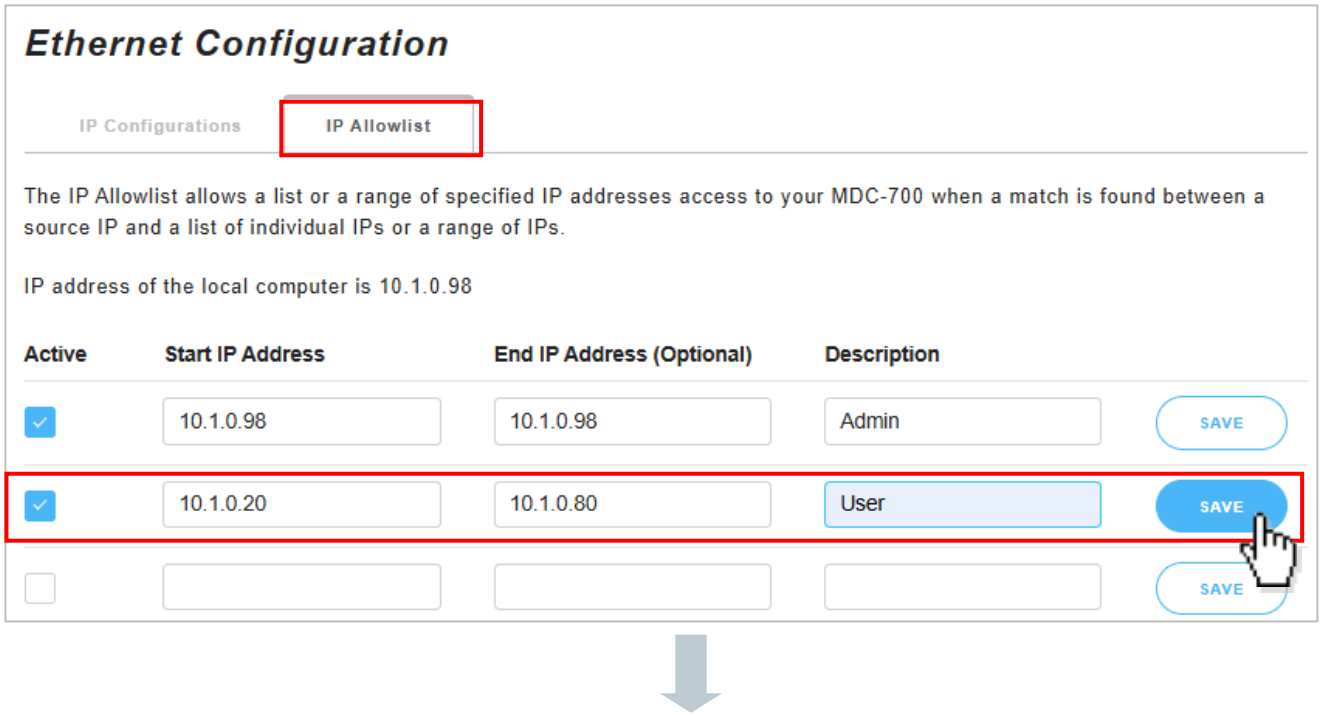
Configure a Single IP Address

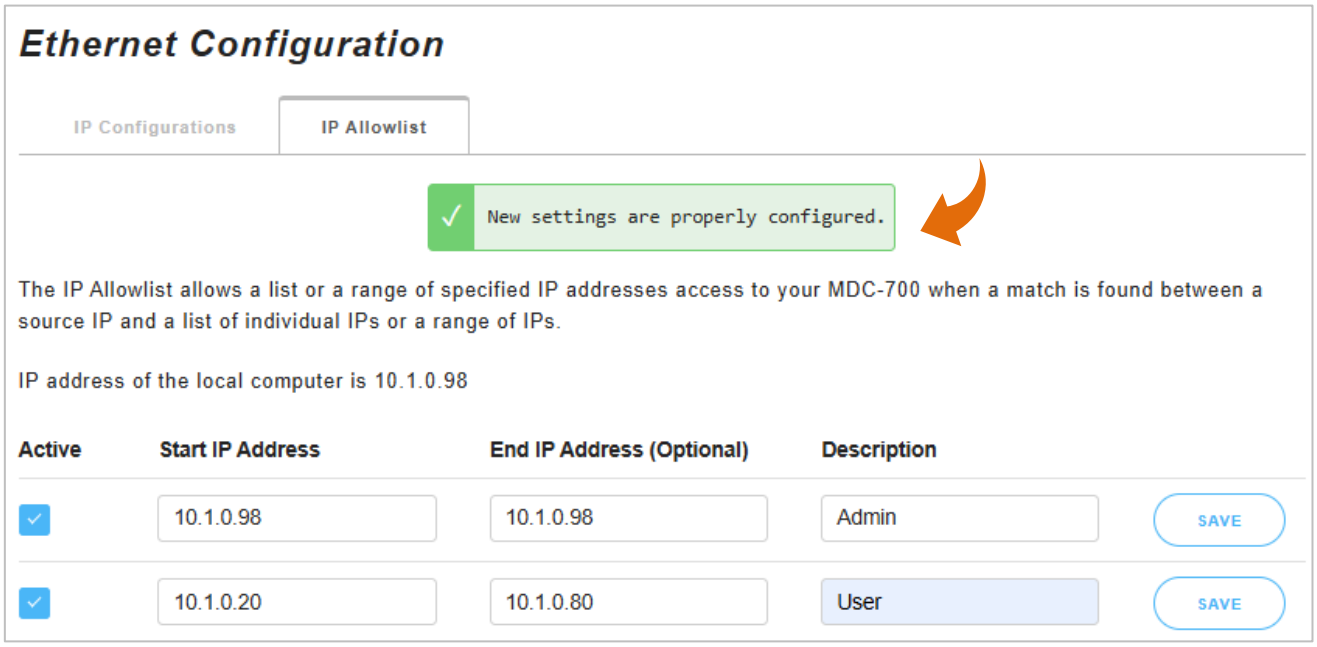
1. In the **Ethernet Configuration** section, select **IP Allowlist**.
2. Check the **Active** box to enable the entry, and enter the IP address in the Start IP Address field.
3. Enter a description in the Description field.
4. Click the Save button.
5. Verify that the web interface confirms the settings were saved successfully.



Configure an IP Range

1. In the **Ethernet Configuration** section, select **IP Allowlist**.
2. Check the **Active** box to enable the entry. Enter the starting IP address in the Start IP Address field and the ending IP address in the End IP Address field.
3. Enter a description in the Description field.
4. Click the Save button.
5. Verify that the web interface confirms the settings were saved successfully.



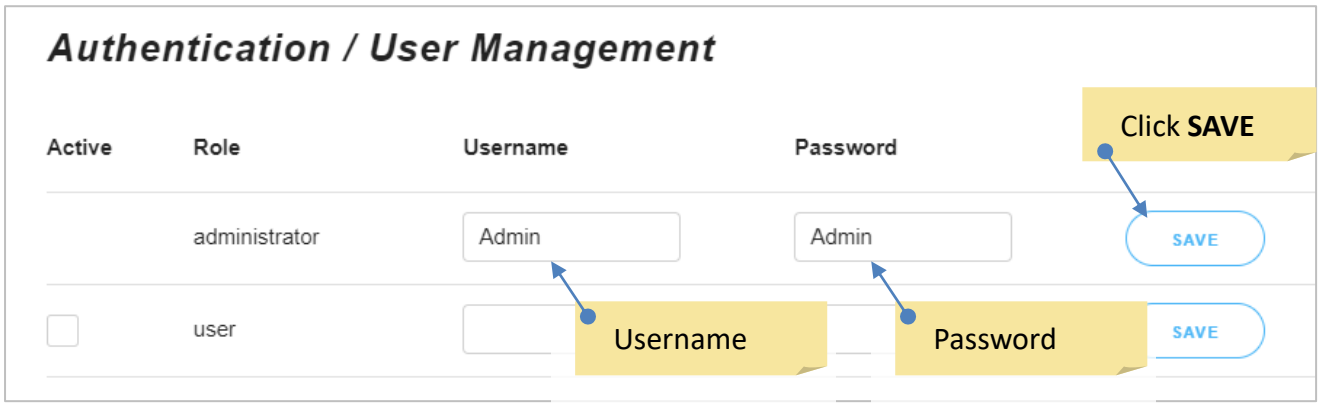


■ Authentication / User Management

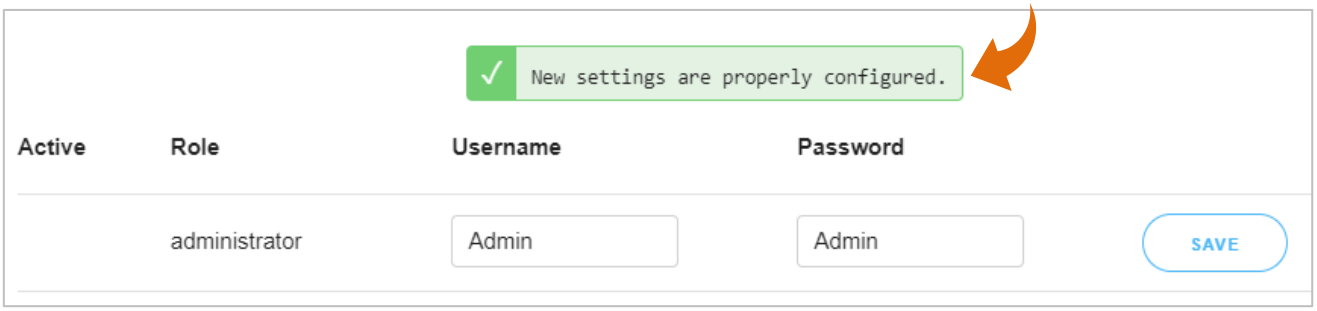
In the **Authentication / User Management** section, you can change the username and password of the administrator account, create a read only user account, and set security questions and answers for password recovery.

Changing the username and password of the administrator

Enter new username and/or password and click **SAVE**

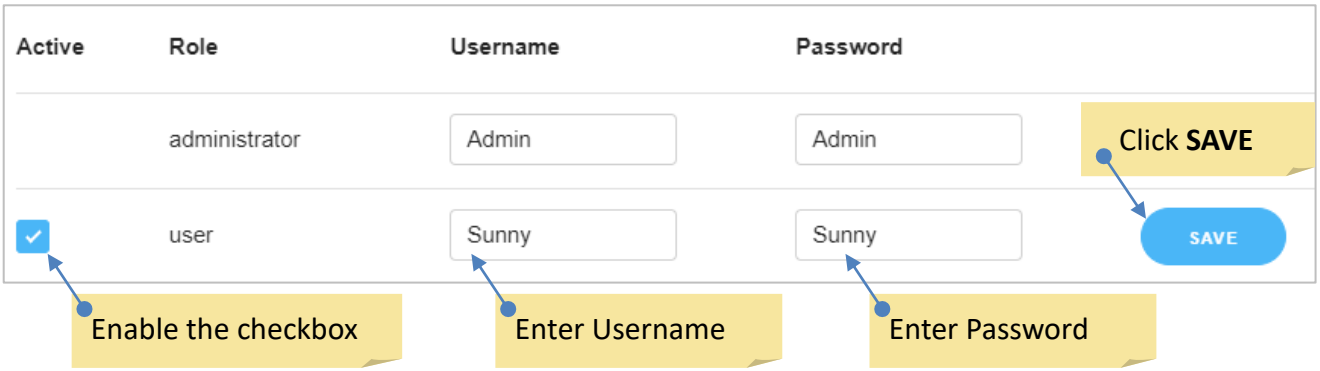


You will see a success message displayed.

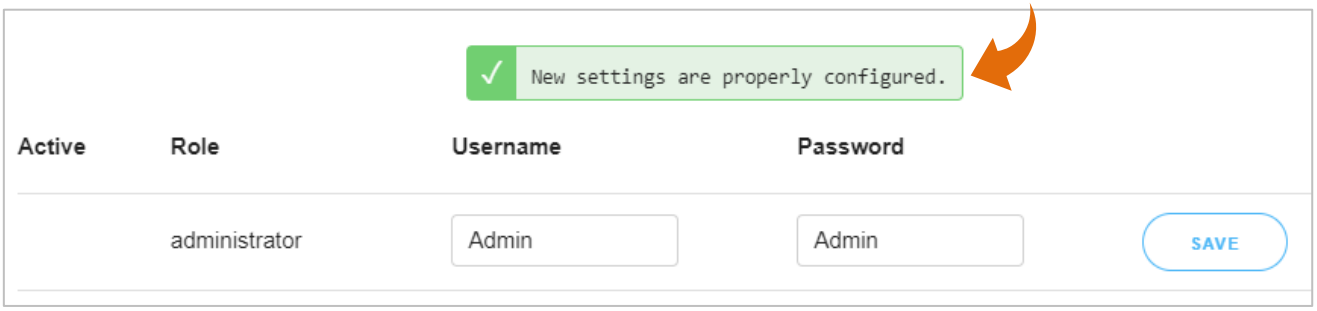


Creating a read only user account

In order to avoid unexpected changes to the settings of a running MDC-700 module, you can create a user account with only read permission, and specify which information can be accessed.

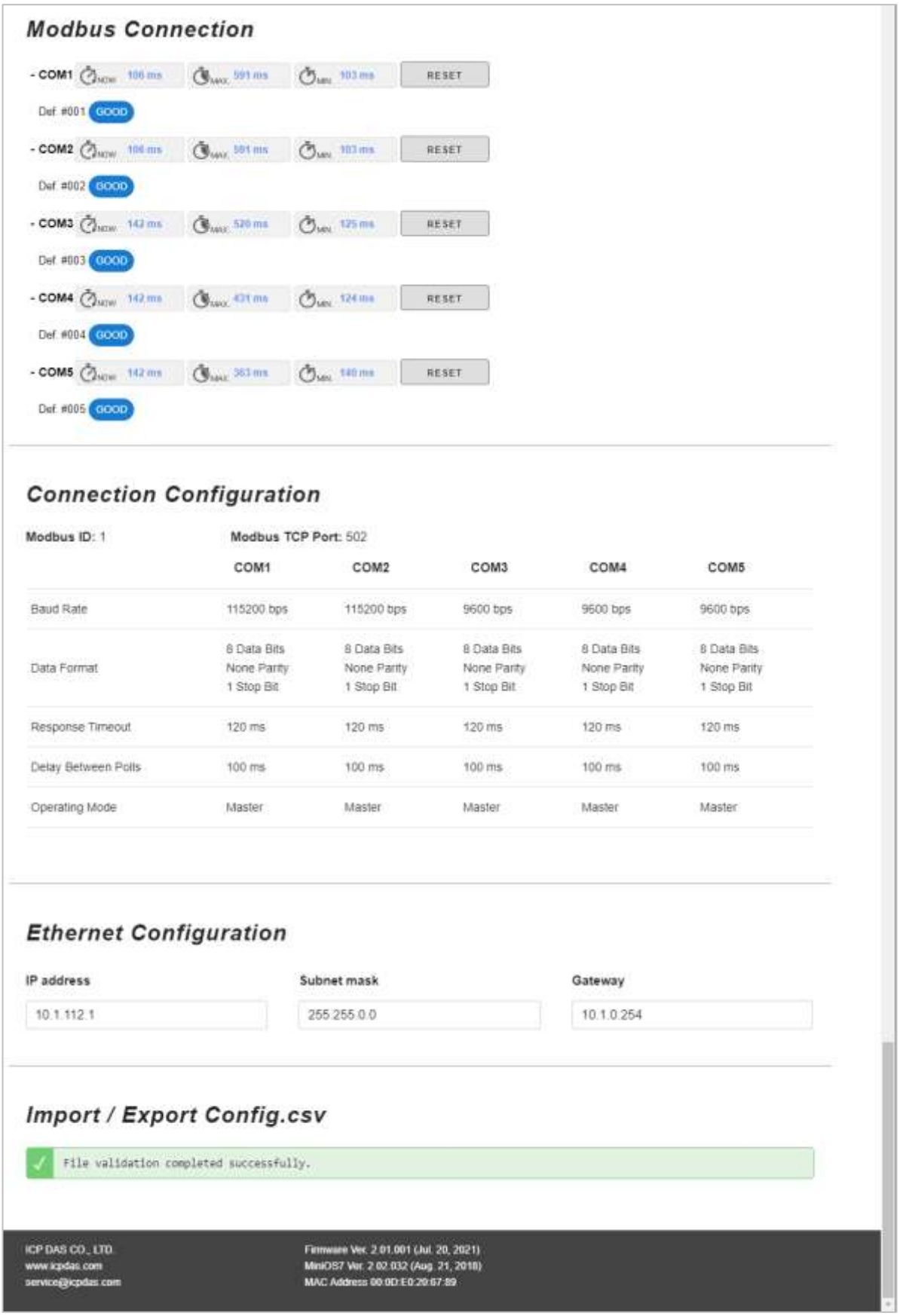


You will see a success message displayed.

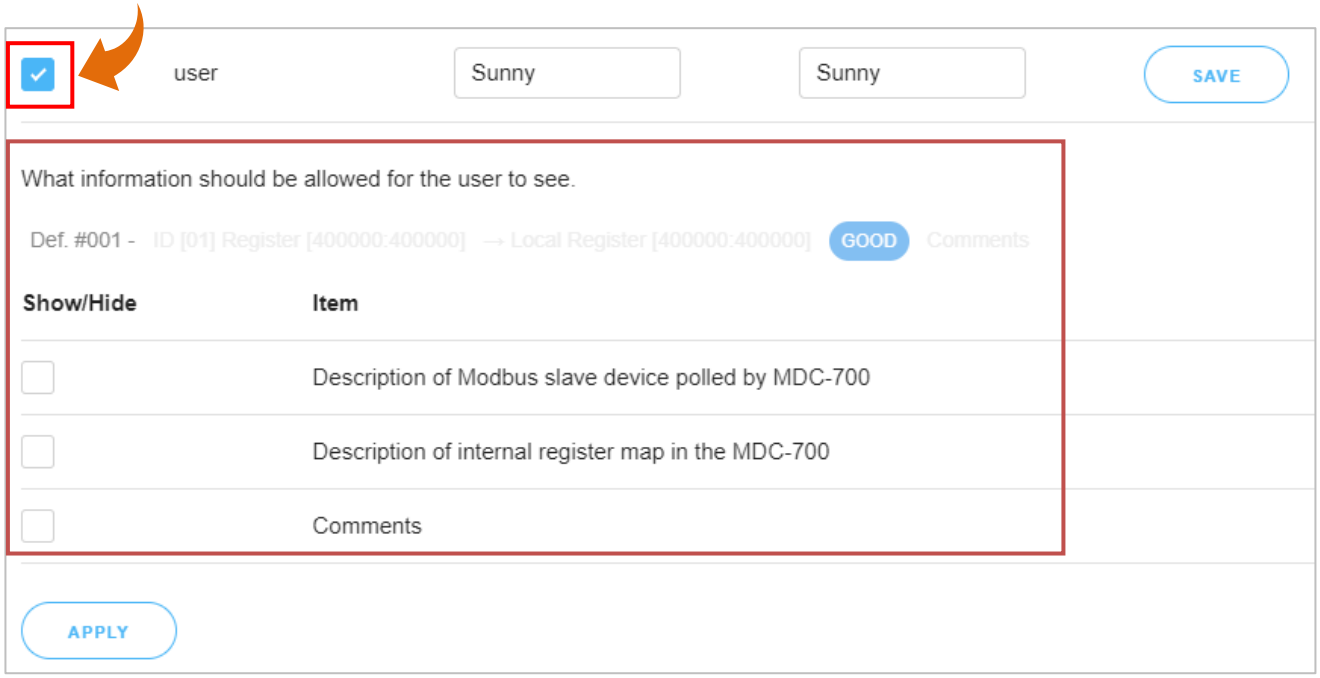


Specifying the information for the read-only user account

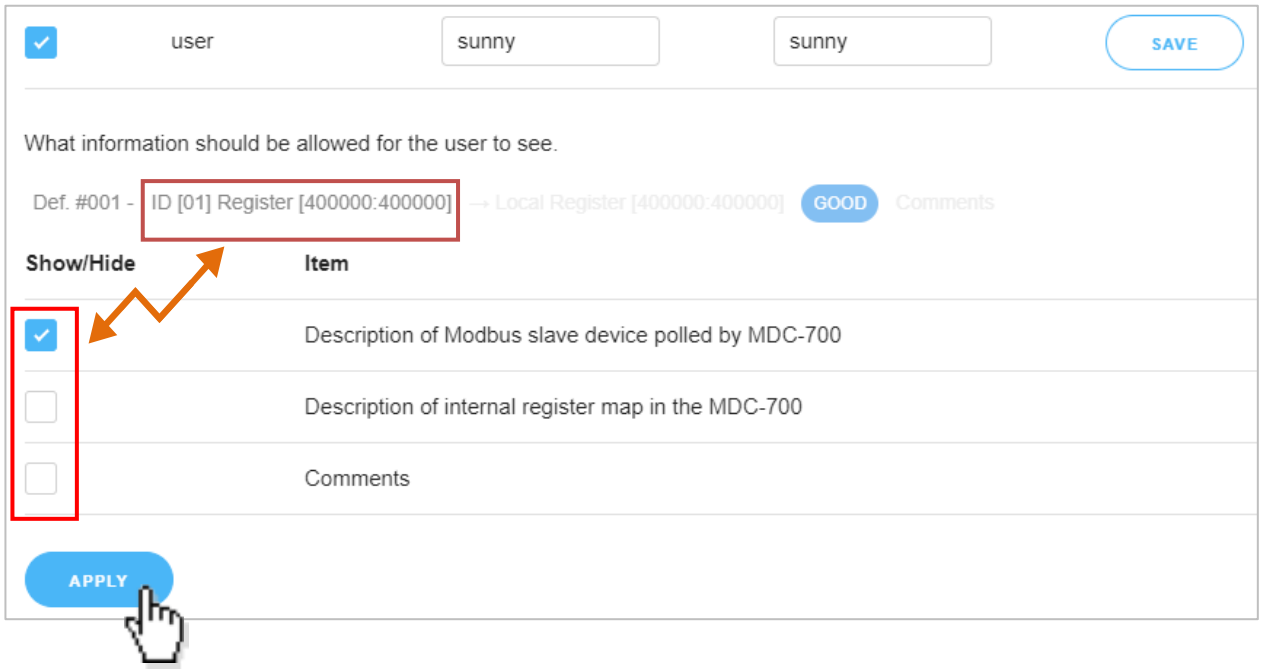
If the user account is created without specifying which information can be accessed, the content that can be accessed is shown as the picture below.



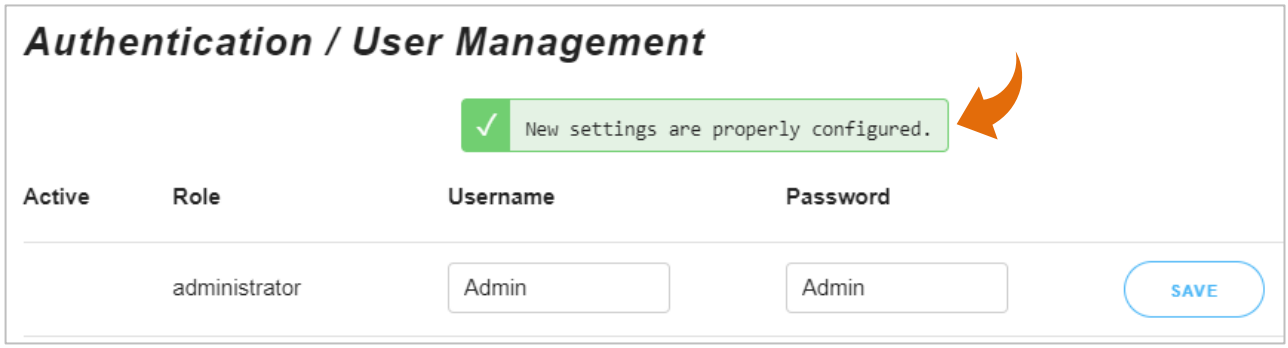
The section for specifying information for the read only user to access is enabled only when the checkbox for user account has been activated.



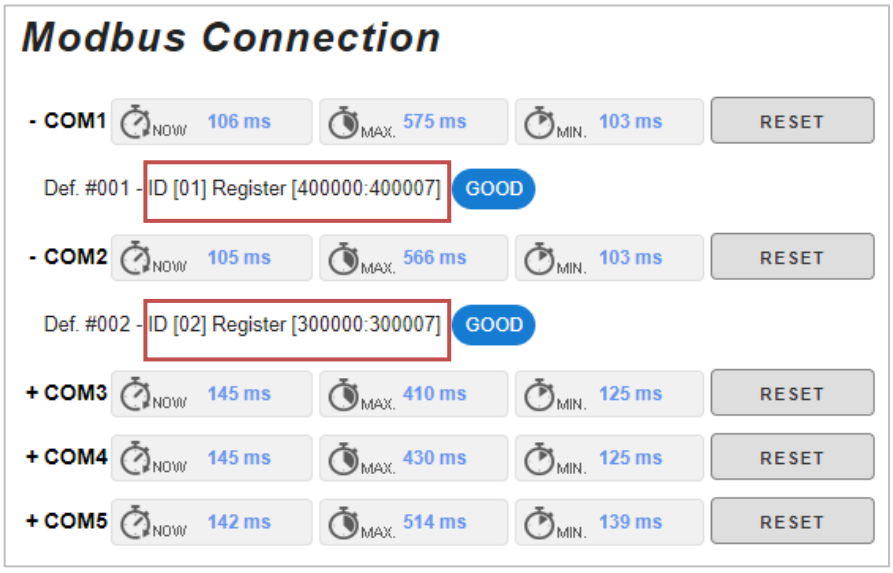
Check the checkbox for allowing the information to be accessed by the user account, the example of information checked will be shown in the next line of "**What information should be allowed for the user to see**". After completing the operation, click **APPLY** to make the settings take effect.



The success message will be displayed.



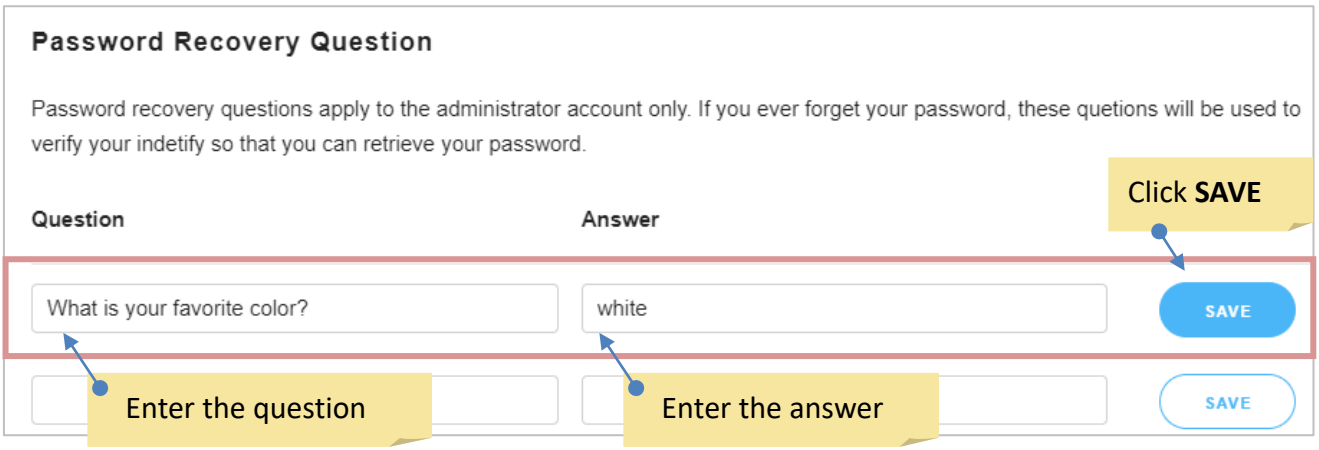
Log in with the user account, now the information checked is displayed on the page.



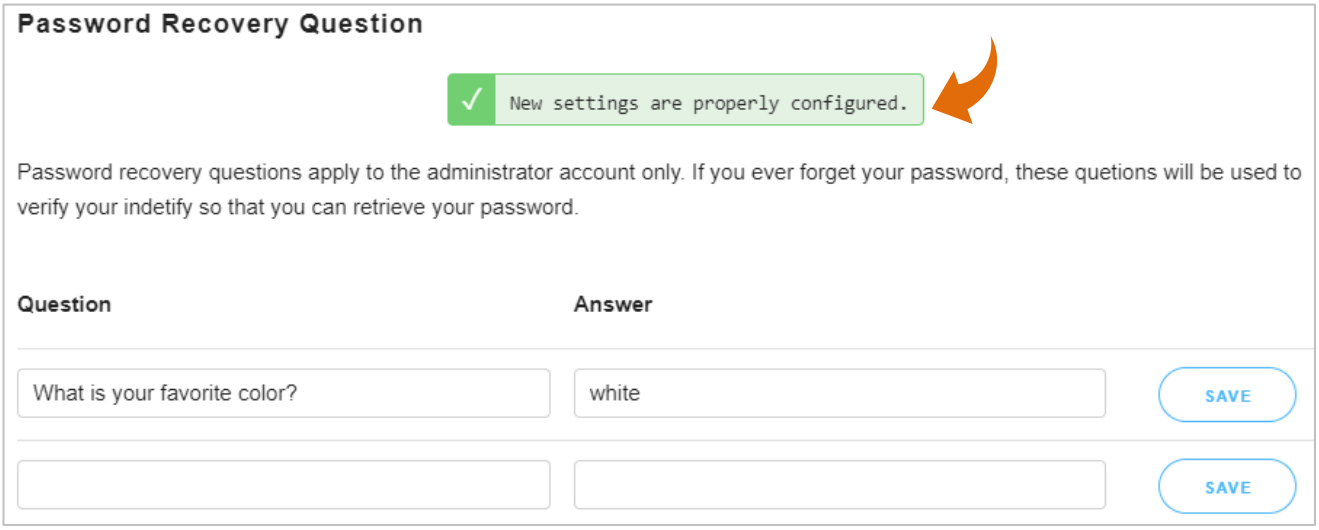
Setting security questions and answers

The MDC-700 allows you to set security questions and answers that you can use should you forget your password. Two sets of security questions and answers are provided. You can enter a maximum of 38 characters in the Question field and a maximum of 14 characters in the Answer field. Note that the answer is case sensitive when it is used to log in to the MDC module.

Enter the question and answer, and click **SAVE**.

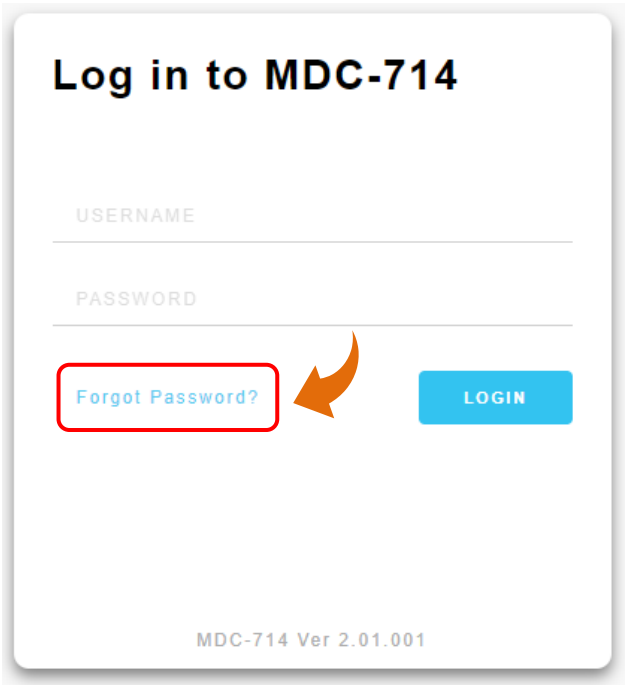


The success message will be displayed.

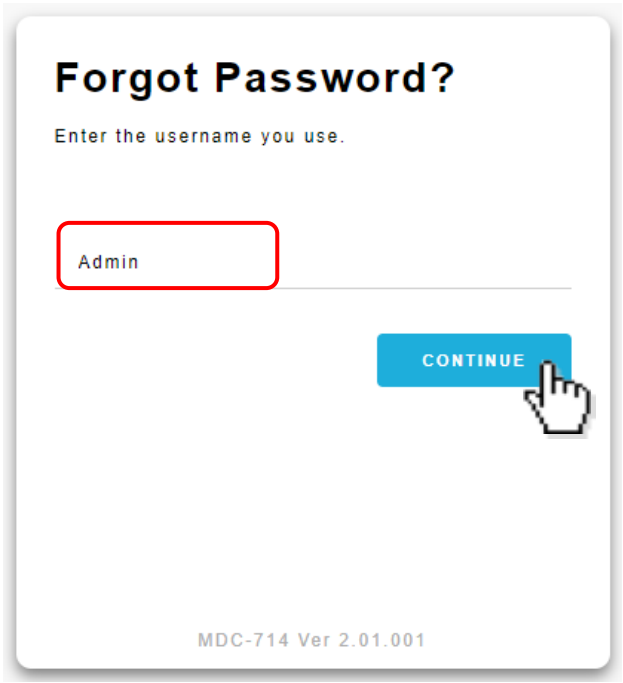


How to log in to the module when you forgot your password?

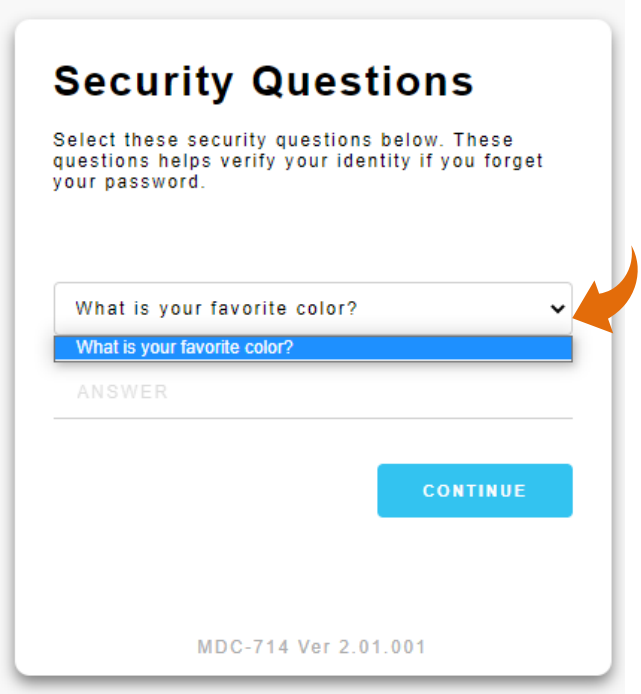
If you are an administrator and you have forgotten your password, click **Forgot Password?**



Enter your username and click **CONTINUE**

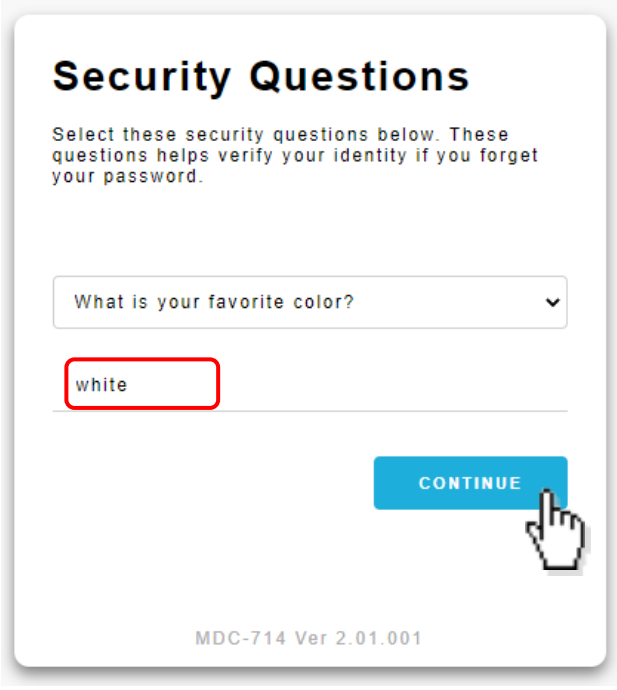


Select your question from the drop down menu



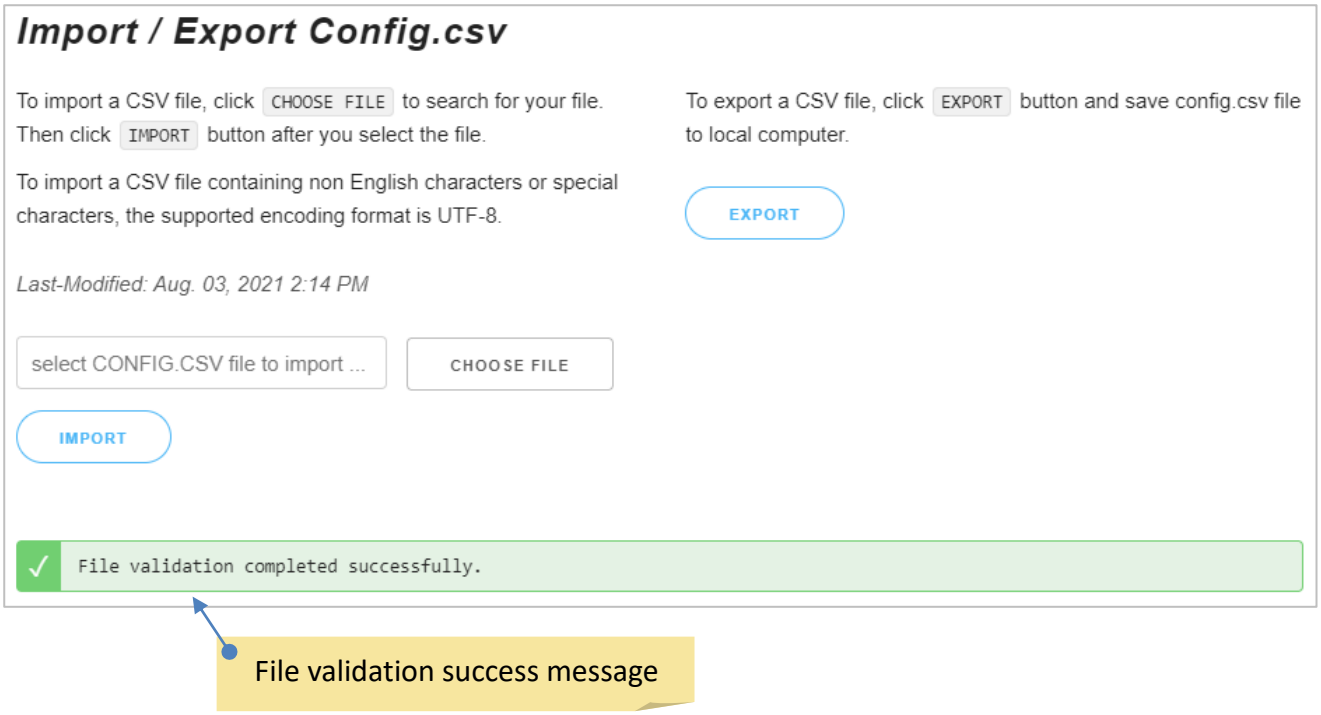
Enter the answer (case sensitive) and click **CONTINUE**.

Now you have logged into your account on the module.

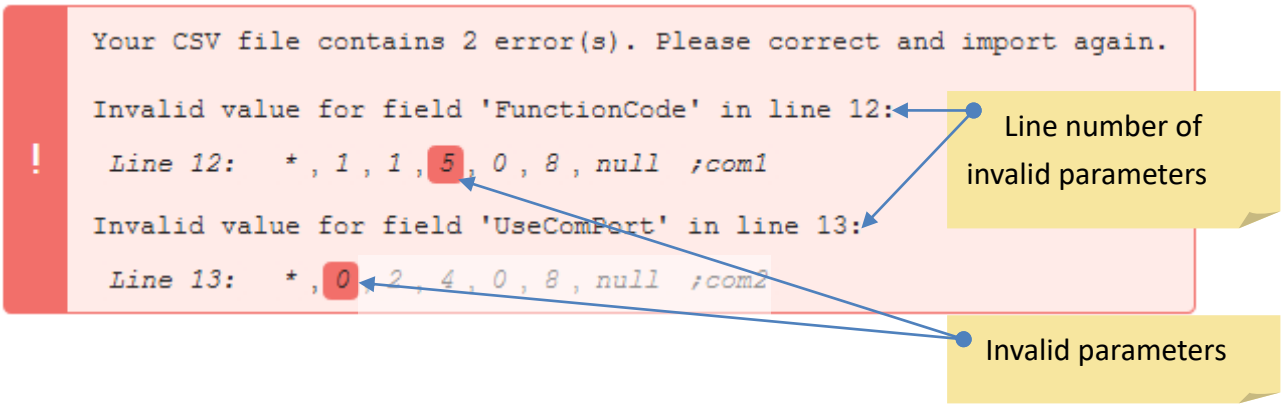


■ Import/Export Config.csv and file validation

You can import/export the config.csv file in this section. Refer to section 4.1 for the detailed steps.



After firmware 2.00.001, MDC-700 provides the function of validating the polling definitions in its config.csv file. If the validation is failed, the failure message with line number and position of invalid parameters will be shown as below.

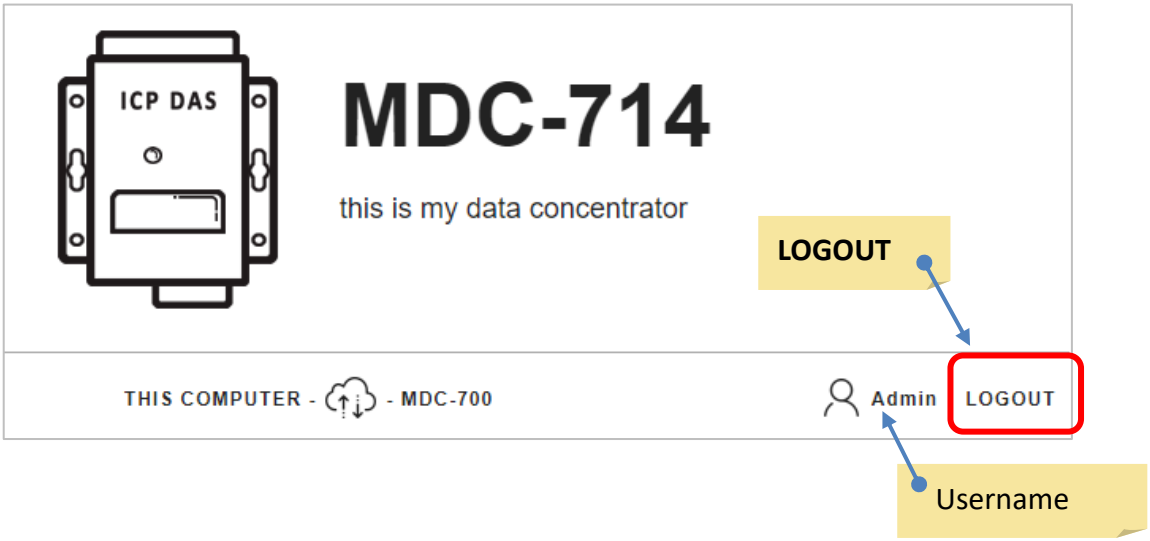


■ Fimware Version/OS Version and MAC Address

Information about Firmware version, OS version and MAC address is located in the footer.

■ Logging out

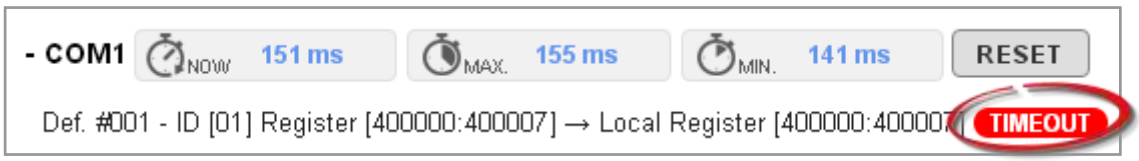
The current username is displayed at the right of the connection status. Click **LOGOUT** to log out from the MDC-700.



6. Troubleshooting

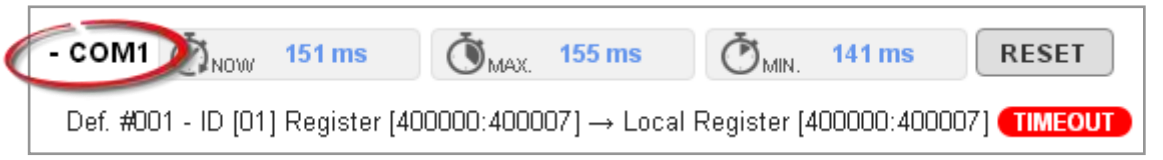
In this chapter, we will explain how to troubleshoot the communication problems.

■ Possible causes of TIMEOUT



◆ **Situation #1**: The slave device is not active or the transfer function of the slave site may fail.
Solution: Check the slave device is powered up and the communication function is enabled.

◆ **Situation #2**: The COM port number to which the slave device is connected is not the same with the UseComPort setting in the polling definition.
Solution: Connect the slave device to the COM port number that is defined in the polling definition, or fix the ***UseComPort*** parameter to the virtual COM port number that the slave device is connected to.



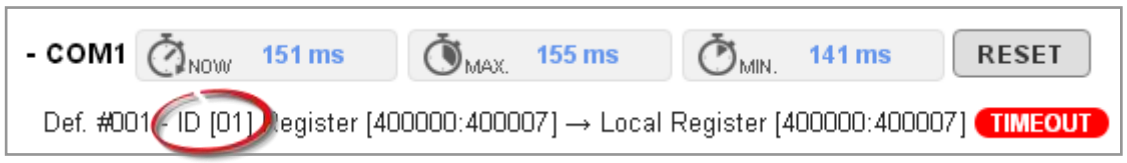
◆ **Situation #3**: The wiring for communication is wrong.
Solution: Exchange the D+ and D- wiring of RS-485 connection, or exchange the Rx and Tx wiring of RS-232 connection, and check the GND pin on the slave device is properly connected to the MDC-700.

◆ **Situation #4**: An incorrect Baud Rate or/and Data Format setting is being specified.
Solution: Check and fix the difference of the Baud Rate and Data Format settings between the polling definition and the slave device.

ComPortNo	*BaudRate*	*DataBit*	*Parity*	*StopBit*	*Timeout*	*PollDelay*	*Operating Mode*
1	9600	8	0	1	120	100	Master
2	9600	8	0	1	3000	1000	Master
3	9600	8	0	1	3000	1000	Master
4	9600	8	0	1	120	100	Master
5	9600	8	0	1	120	100	Master

◆ **Situation #5**: An incorrect ID of the Modbus slave device is being specified.

Solution: Check and fix the difference of ID number between the polling definition and the slave device.



◆ **Situation #6**: The Timeout or PollDelay setting is not long enough.

Solution: Lengthen the Timeout or PollDelay setting until it is suitable for communication with the slave device.

ComPortNo	*BaudRate*	*DataBit*	*Parity*	*StopBit*	*Timeout*	*PollDelay*	*Operating Mode*
1	9600	8	0	1	120	100	Master
2	9600	8	0	1	3000	1000	Master
3	9600	8	0	1	3000	1000	Master
4	9600	8	0	1	120	100	Master
5	9600	8	0	1	120	100	Master

7. FAQ

Q1: What are the maximum numbers of polling definition and local register?

A1: The maximum number of polling definition in a MDC-700 is 250, each definition can access up to 125 registers. Each of the four tables (DI/DO/AI/DO) can store up to 9600 registers for polled data.

Q2: What is the maximum number of registers can be accessed in one Modbus command?

A2: By following the Modbus protocol, the maximum amount of registers that one command can access is 255 of function code 01 and 02, and 126 of function code 03 and 04.

Q3: How are the local registers mapped to the polled data in a MDC-700?

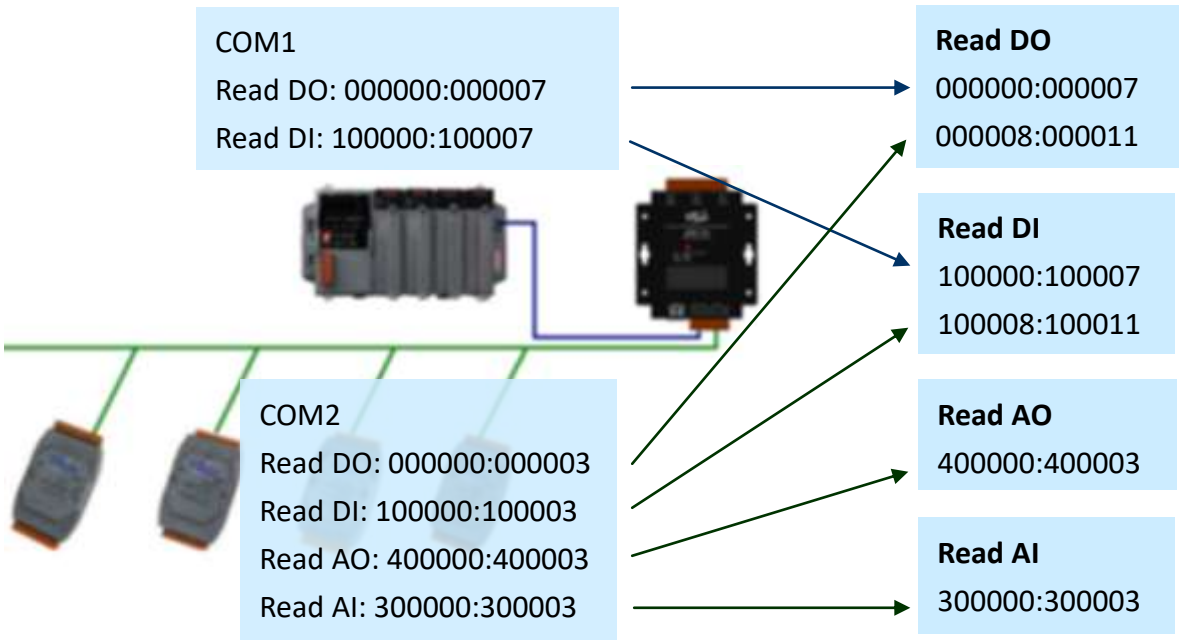
A3: Only the function code 01/02/03/04 can be used in the polling definition section

01: Read Coil Status (Read DO)
02: Read Input Status (Read DI)
03: Read Holding Registers (Read AO)
04: Read Input Registers (Read AI)

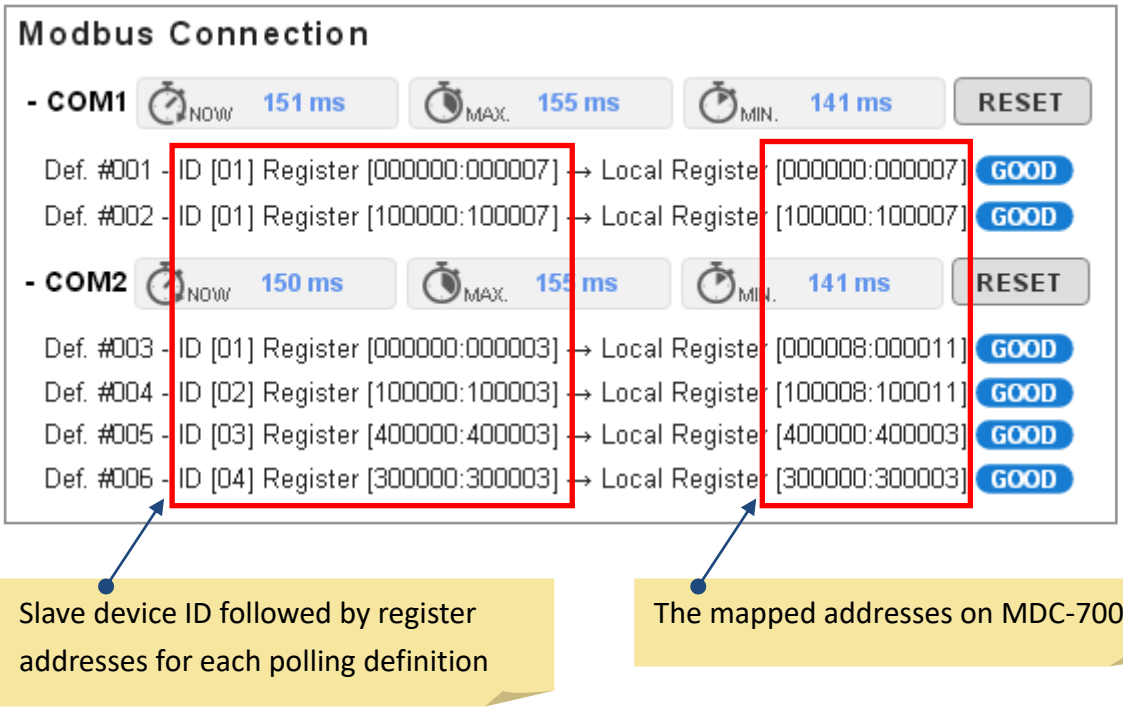
Refer to the example below,

#	UseComPort	SlaveModbusID	FunctionCode	RegStart Addr	RegCount	Timeout EventProcess	Preset Value
*	1	1	1	0	8	2	0
*	1	1	2	0	8	2	0
*	2	1	1	0	4	2	0
*	2	2	2	0	4	2	0
*	2	3	3	0	4	2	0
*	2	4	4	0	4	2	0

The MDC-700 will sort the order of polling data by COM port number and the sequence of polling definition; and then map the local registers corresponding to the data type (DI/DO/AI/AO) by the order of polling data. So the data comes from different slave devices with the same type will be saved in continuous registers, and a Modbus mater device and read the data on a variety of slave devices with one Modbus command.



The local registers mapping is listed on the main page of the MDC-700 module.



The MDC-700 allows users to enable/disable a polling definition by changing the first field of the polling definition section in the config.csv file. There are three types that users can use:

"*": Asterisk symbol means that this is a valid polling definition. The MDC-700 will assign local register for data defined in the definition and save the polled data to the mapping local register.

"-": Minus sign means that this is a disabled polling definition. The MDC-700 will assign local register for data defined in the definition but will not poll the data.

"": Empty means that this is a null polling definition. The MDC-700 will neither assign local register for data defined in the definition nor poll data.

#	UseComPort	SlaveModbusID	FunctionCode	RegStartAddr	RegCount
*	1	1	1	0	8
*	1	1	2	0	8
*	2	2	1	0	4

With the function of retaining register space mapped for specific devices, or releasing those spaces mapped but reserving the definition, the main program on the Modbus master device can be applied in similar applications where users would like to change or stop some devices without modification or with minimum level of modification.

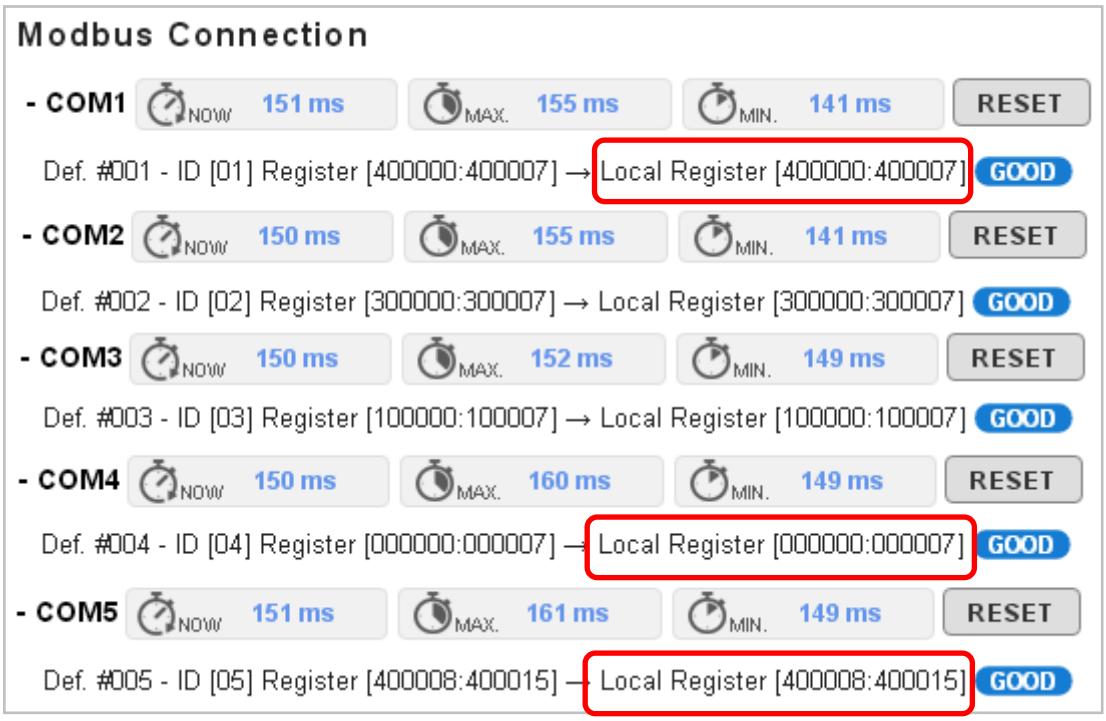
Q4: How to write data to output channels on a Modbus RTU slave device?

A4:

Step 1: Edit the polling definition for the output channels with read function code in the config.csv file. (For example, use 01 to read DO channels, 03 to read AO channels)

#	UseComPort	SlaveModbusID	FunctionCode	RegStart Addr	RegCount	Timeout EventProcess	Preset Value
*	1	1	3	0	8	2	0
*	2	2	4	0	8	2	0
*	3	3	2	0	8	2	0
*	4	4	1	0	8	2	0
*	5	5	3	8	8	2	0

Step 2: Import the config.csv file into the MDC-700, wait the MDC-700 reboot in 5 seconds, and then check the addresses for the local registers mapped to the output channels.



Step 3: Write data with corresponding function code (05/06/15/16) on your Modbus master device to the local registers mapped for the output channels, the MDC-700 will process writing operations to the slave devices.

Q5: How to read the status of each connection?

A5: The status for each connection is saved in the sequence of polling definition from local register address 39600. The maximum number of polling definition in the config.csv file is 250, so the available address for the connection status is from 39600 to 39849. A Modbus master use function code 04 to read the status, up to 126 register of status can be read in one command. For example, the status of the graph shown above is presented as the third column in the following table.

Def. number	Address	Status	Status display on web page
Def.#001	39600	0	GOOD
Def.#002	39601	0	GOOD
Def.#003	39602	0xFFFF	TIMEOUT
Def.#004	39603	0x8201	ERROR: ILLEGAL FUNCTION
Def.#005	39604	0	GOOD
Def.#006	39605	0x8402	ERROR: ILLEGAL DATA ADDRESS

The value of status:

0: Good

0xFFFF: Timeout

0x8XYY: Exception Rresponse. **X** - Modbus Function Code. **YY** - Exception Code.

Exception Code	Name	Meaning
01	Illegal Function	The function code received is not an allowable action.
02	Illegal Data Address	The data address received in the query is not an allowable address.
03	Illegal Data Value	A value contained in the query data field is not an allowable value.
04	Illegal response length	The request would generate a response with size bigger than that available for MODBUS protocol.

Q6: How to update firmware?

A6: The upgrade procedure of the firmware consists of the following main steps:

- Install the MiniOS7 Utility on your computer
- Upload the latest firmware to MDC-700 through the MiniOS7 Utility
- Check the firmware version and the configuration settings via web interface

Here we will introduce how to update firmware of the MDC-700 step-by-step.

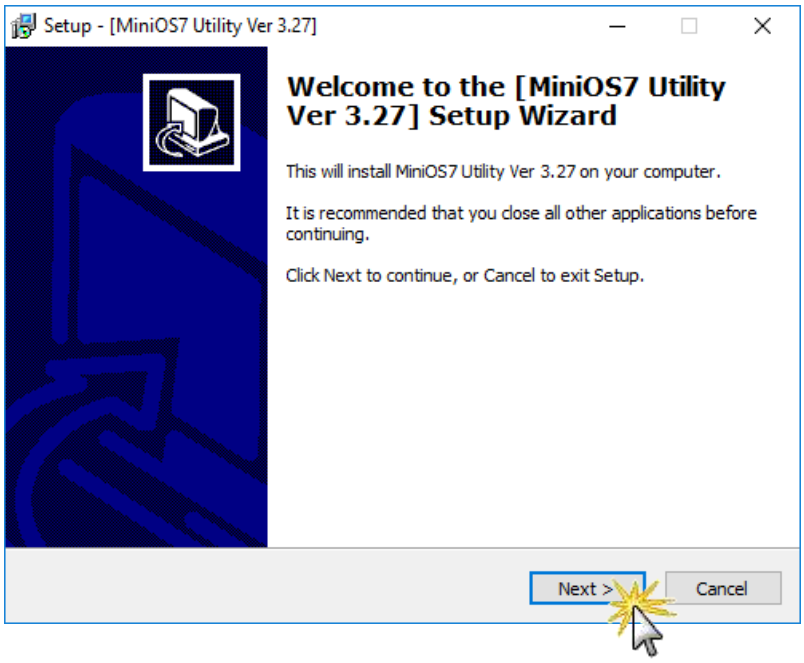
1. Install MiniOS7 Utility

STEP 1: Download the installation file of the MiniOS7 Utility to your computer

The installation file can be obtained from:

https://www.icpdas.com/en/download/show.php?num=1053

Step 2: Run the downloaded file to start the installation process. It will lead you through the installation step by step



Step 3: After the installation is finished, a “MiniOS7 Utility Ver 3.27” icon will appear on your desktop. You can run the program by double-clicking the icon or clicking MiniOS7 Utility Ver 3.27 item in the ICPDAS folder in the Start menu.

2. Upgrade Firmware using the MiniOS7

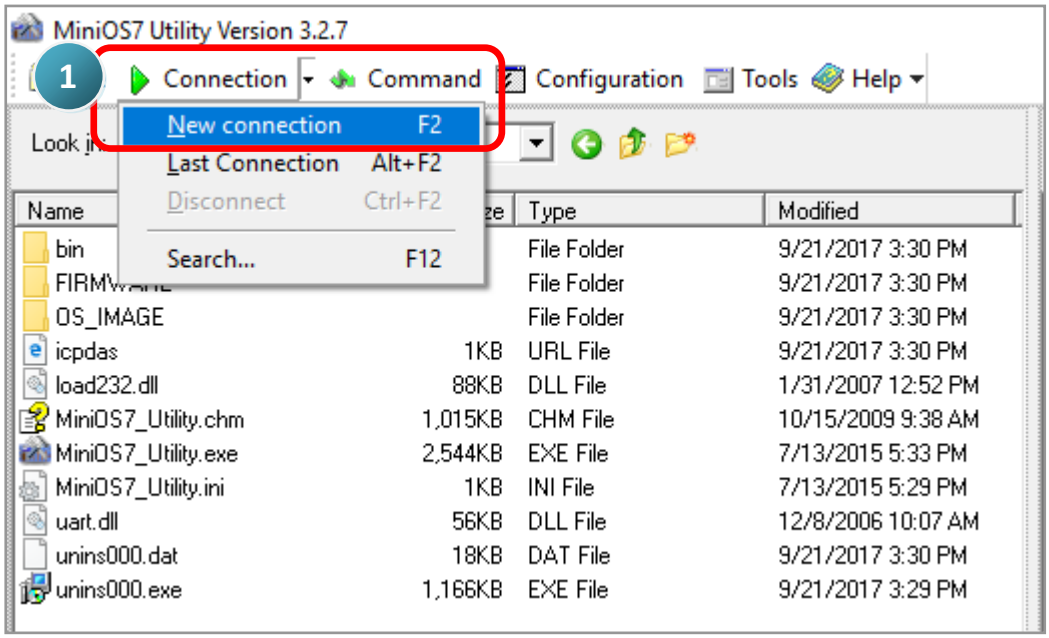
The firmware update requires a TCP/IP connection. Connect the MDC-700 to a network whenever possible.

Step 1: Use an Ethernet cable to connect the MDC-700 to the computer

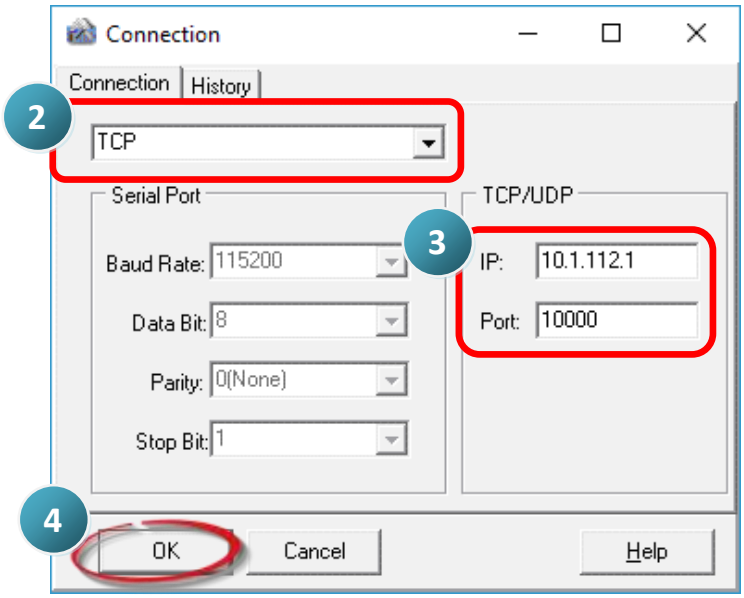
After plugging the Ethernet cable, the Link/Act and 10/100 indicator LEDs come on or start flashing to indicate a connection was made.

Step 2: Establishing a connection between the MiniOS7 Utility and the MDC-700

Launch the MiniOS7 Utility and then select **New Connection** on the Connection menu.



On the "**Connection**" tab of the "Connection" dialog, select "**TCP**" from the dropdown list, type the IP address of MDC-700, and then click OK button.



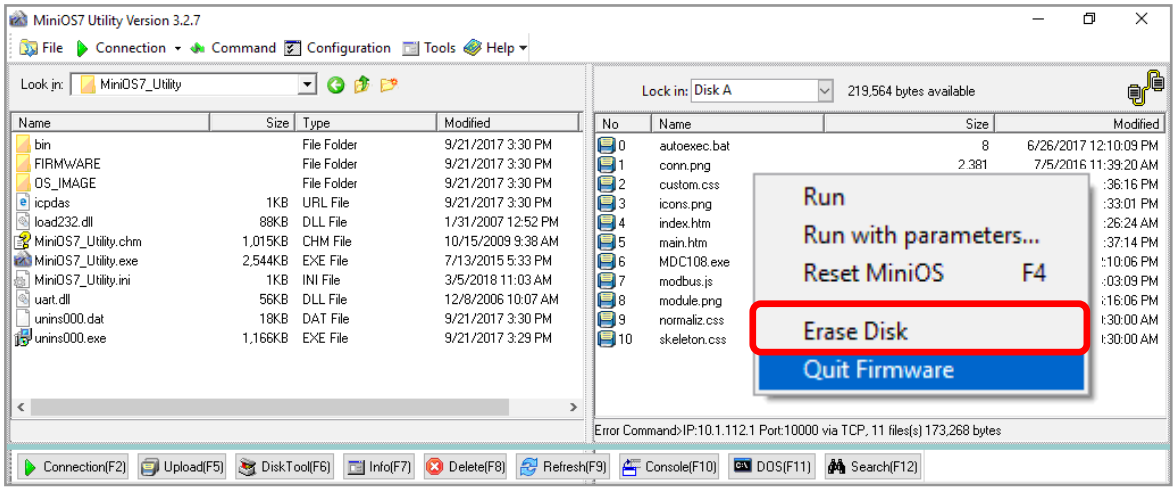
Step 3: Look for the connector symbol at the upper right-hand corner of the MiniOS7 Utility to ensure the connection is made

If the connection fails, make sure that:

- An Ethernet cable is connected securely to both the MDC-700 and your computer
- The MDC-700 is active (powered on)
- The IP address of MDC-700 is correct
- No firewall is blocking the connection

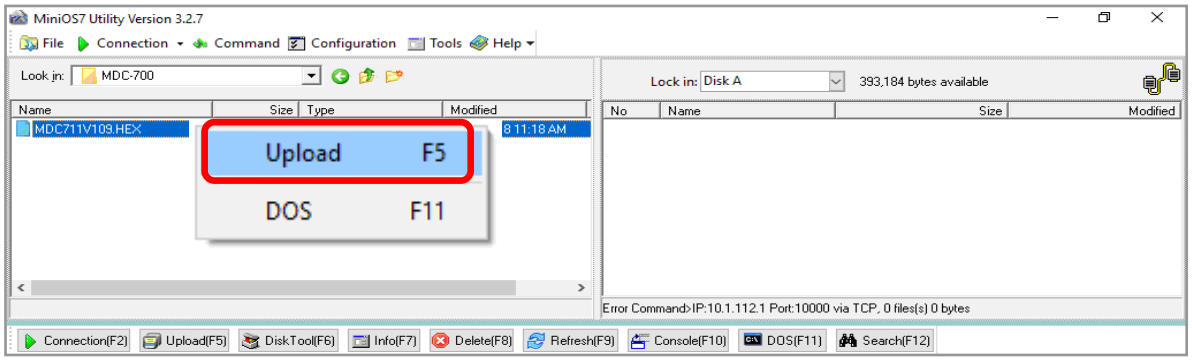
Step 4: Delete the original files from the MDC-700

After establishing a connection, select “Erase Disk” from Command menu (or right-click on the right of window) to delete all files existed on the MDC-700.



Step 5: Upload the firmware file to MDC-700

Right-click on the MDC7XXV109.HEX file and select Upload from the menu.

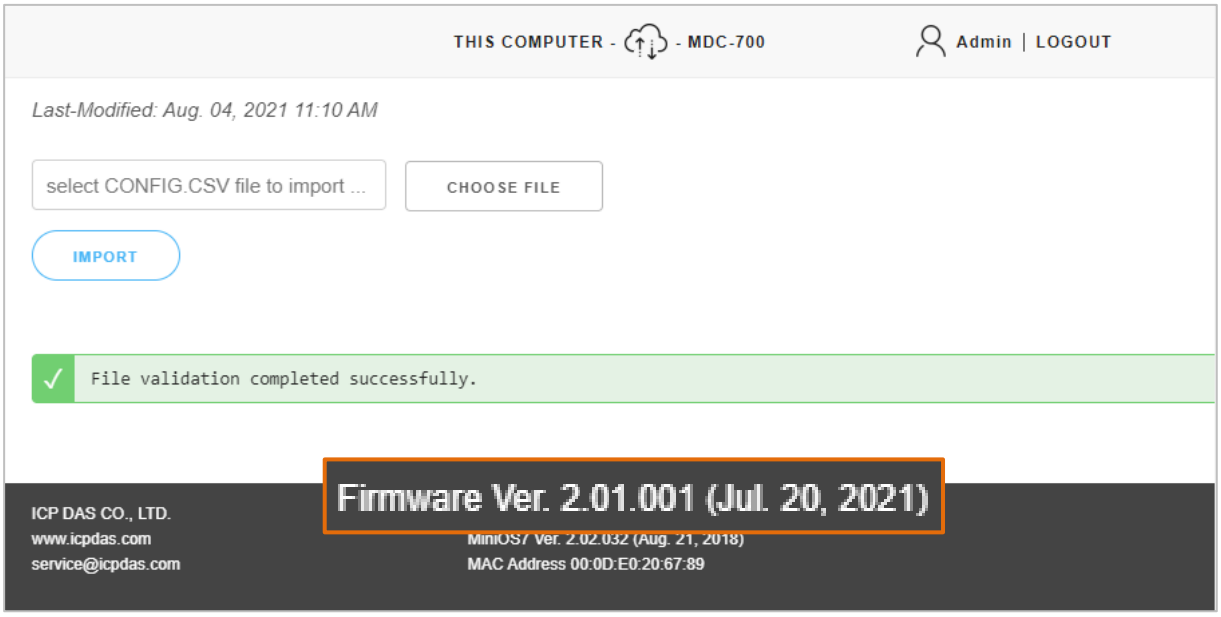


Step 6: Wait until the firmware update is finished, and then power cycle the MDC-700.

3. Check the Firmware Version

Step 1: Open a web browser and enter the IP address of the MDC-700 in the URL.

Step 2: Check the version information at the bottom of the page.



Q7: Why does the page not display correctly in my browser?

A7: After the firmware version 1.08 was released, the MDC module adopts HTML5 in place of Flash. HTML5 is supported in all modern browsers, but not the older browsers like IE8 and below. If your browser does not support the HTML5, it cannot render the page correctly. It is recommended to use a newer browser.

The browsers support HTML5:

- Windows Edge 14 or later
- Windows IE9/IE10/IE11 or later
- Google Chrome 55 or later
- Mozilla Firefox 50 or later
- Apple Safari 9.1 or later
- Opera 42 or later

If the MDC-700 module is running with firmware version 1.06 or earlier, the page requires the Adobe Flash Player to be installed. The latest version of the Adobe Flash Player can be downloaded by accessing the Adobe Systems Incorporated website. The following instructions will help you to install the Adobe Flash Player in your web browser.

STEP 1: Go to the Adobe Flash Player Download Center

The address for Adobe Flash Player Download Center is http://get.adobe.com/flashplayer/

NOTE The Adobe Flash Player is subject to change without notice; refer to http://www.adobe.com/support/flashplayer/debug_downloads.html for the latest version of this software.

STEP 2: Follow the instructions to download the installation file and install it on your PC.

Appendix

Differences between Firmware V. 1.08 and V. 2.00

	Firmware V. 1.08	Firmware V. 2.00
Authentication / User Management		
Security authentication	-	Account and password login Security question and answer login
Access permission management	-	One Full Access Administrator and one view-only user
Polling Definition		
Definition validation	-	Yes
Support for displaying definition comments	-	Yes

Differences between Firmware V. 1.06 and V. 1.08

	Firmware V. 1.06	Firmware V. 1.08
Modbus RTU		
Polling Definition	240 Max.	250 Max.
Max. Register Count in one Polling Definition	64 Max.	125 Max.
The data that Master will obtain while timeout error is occurred	Exception Code	Exception Code, the last correct data or the preset value selectable
Web Interface		
Web technique	Flash	HTML5
Scan Time for each COM port	-	Yes

Revision History

Revision	Date	Description
1.0.5	2025/05	- Added new function description for web interface: IP Allowlist, Template CSV File (configuration template) download, Export Modbus register table as CSV file.
1.0.4	2021/08	- Added specifications of MDC-714, MDC-714i and MDC-771i - Added the description for new functions in firmware v. 2.00
1.0.3	2018/02	- Modified the description for web page for firmware V1.08 - Added Section 2.5. Mounting the Hardware.
1.0.2	2015/11	Added dimensions, appearance information and Troubleshooting, FAQ sections.
1.0.1	2015/07	Added description for MDC-741.
1.0.0	2014/11	First released