# *460MSDS-N34*
# *Protocol Gateway*
## Product User Guide

Firmware Version 8.9.39

**Trademarks**
CompactLogix, ControlLogix, & PLC-5 are registered trademarks of Rockwell Automation, Inc. EtherNet/IP is a trademark of the ODVA. MicroLogix, RSLogix 500, and SLC are trademarks of Rockwell Automation, Inc. Microsoft, Windows, and Internet Explorer are registered trademarks of Microsoft Corporation. BACnet® is a registered trademark of American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE). All other trademarks and registered trademarks are the property of their holders.

**Limited Warranty**
Real Time Automation, Inc. warrants that this product is free from defects and functions properly.

EXCEPT AS SPECIFICALLY SET FORTH ABOVE, REAL TIME AUTOMATION, INC. DISCLAIMS ALL OTHER WARRANTIES, BOTH EXPRESSED AND IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR APPLICATION. THIS LIMITED WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS. YOU MAY ALSO HAVE OTHER RIGHTS, WHICH VARY FROM STATE TO STATE.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular application, Real Time Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams. Except as specifically set forth above, Real Time Automation and its distributors and dealers will in no event be liable for any damages whatsoever, either direct or indirect, including but not limited to loss of business profits, income, or use of data. Some states do not allow exclusion or limitation of incidental or consequential damages; therefore, the limitations set forth in this agreement may not apply to you.

No patent liability is assumed by Real Time Automation with respect to use of information, circuits, equipment, or software described in this manual.

**Government End-Users**
If this software is acquired by or on behalf of a unit or agency of the United States Government, this provision applies: The software (a) was developed at private expense, is existing computer software, and was not developed with government funds; (b) is a trade secret of Real Time Automation, Inc. for all purposes of the Freedom of Information Act; (c) is “restricted computer software” submitted with restricted rights in accordance with subparagraphs (a) through (d) of the Commercial “Computer Software-Restricted Rights” clause at 52.227-19 and its successors; (d) in all respects is proprietary data belonging solely to Real Time Automation, Inc.; (e) is unpublished and all rights are reserved under copyright laws of the United States. For units of the Department of Defense (DoD), this software is licensed only with “Restricted Rights”: as that term is defined in the DoD Supplement of the Federal Acquisition Regulation 52.227-7013 (c) (1) (ii), rights in Technical Data and Computer Software and its successors, and: Use, duplication, or disclosures is subject to restrictions as set forth in subdivision (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at 52.227-7013. If this software was acquired under GSA schedule, the U.S. Government has agreed to refrain from changing or removing any insignia or lettering from the Software or documentation that is provided or from producing copies of the manual or media. Real Time Automation, Inc.

© 2026 Real Time Automation, Inc. All rights reserved.

Overview ..... 6
Hardware Platforms ..... 7
Hardware – N34 ..... 8
Powering The Gateway ..... 9
Port Configuration ..... 11
CAN / RS232 ..... 12
CAN / RS485 ..... 13
Dual RS485 ..... 13
Mounting with a DIN Rail ..... 14
Installing ..... 14
Removing ..... 14
Accessing the Main Page ..... 15
Committing Changes to the Settings ..... 16
Main Page ..... 17
Device Configuration ..... 18
Network Configuration ..... 19
Modbus TCP/IP Server Configuration ..... 20
Modbus TCP/IP Server Configuration-Data Groups ..... 21
Auto-Configure Group by Device vs. Auto-Configure Group by Data Type ..... 22
Group by Device (Default Method) ..... 22
Group by Data Type ..... 22
Modbus TCP/IP Server Data Group Configuration: Auto-Configure ..... 23
Modbus TCP/IP Server Data Group Configuration: Manual Mode ..... 24
Configure Read and Write Data Groups ..... 25
DeviceNet Slave Configuration ..... 26
DeviceNet Slave Data Group Configuration ..... 27
DeviceNet Slave Data Group Configuration: Auto-Configure ..... 28
Auto-Configure Group by Device vs. Auto-Configure Group by Data Type ..... 29
Group by Device (Default Method) ..... 29

Group by Data Type ..... 29
DeviceNet Slave Data Group Configuration: Manual Mode ..... 30
DeviceNet Slave Data Group Configuration: Using Strings ..... 31
DeviceNet Slave Data Group Configuration-Connection Parameters ..... 32
Mapping - Transferring Data Between Devices ..... 33
Display Mapping and Values ..... 33
Display Data ..... 33
Display String ..... 36
Display String use case ..... 38
Data and String Mapping – Auto-Configure ..... 39
Data Mapping – Explanation ..... 40
Data Mapping – Adding Diagnostic Information ..... 41
String Mapping – Explanation ..... 46
Mapping – Auto-Configure Mode to Manual Configure Mode ..... 47
Mapping – Manual Configure Mode to Auto-Configure Mode ..... 48
View as Text ..... 49
Data Mapping ..... 49
String Mapping ..... 49
Base Triggering – Data Validiation Triggering ..... 50
Security Configuration ..... 52
Security Configuration-Security Levels ..... 53
Security - Log In ..... 54
Security - Log Out ..... 54
Email Configuration ..... 55
Alarm Configuration ..... 56
Diagnostics – Alarm Status ..... 58
Alarms – Active ..... 59
Alarms – Clear ..... 59
Change of State (COS) Configuration ..... 60
Diagnostics Info ..... 61
Diagnostics Mapping ..... 61

Diagnostics – Modbus TCP/IP Server ..... 62
Diagnostics – DeviceNet Slave ..... 64
LED Configuration ..... 67
Configuration Files ..... 68
Export Configuration ..... 68
Import Configuration ..... 68
Save and Replace Configuration Using SD Card ..... 70
Saving Configuration Using SD Card ..... 70
Replacing Configuration Using SD Card ..... 70
Utilities ..... 71

# Overview

The 460MSDS-N34 gateway Connects a Modbus TCP client with a DeviceNet master.  By following this guide, you will be able to configure the 460MSDS-N34 gateway.

Number of ASCII devices is dependent on the Hardware and Product number of the 460 gateway.

For further customization and advanced use, please reference the appendices located online at: http://www.rtautomation.com/product/460-gateway-support/.

If at any time you need further assistance, do not hesitate to call Real Time Automation support. Support Hours are Monday-Friday 8am-5pm CST

Toll free: 1-800-249-1612
Email: support@rtautomation.com

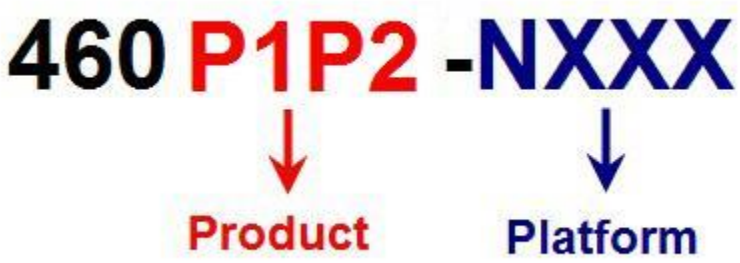
# Hardware Platforms

The 460 Product Line supports a number of different hardware platforms. There are differences in how they are powered, what serial settings are supported, and some diagnostic features supported (such as LEDs). For these sections, be sure to identify the hardware platform you are using.

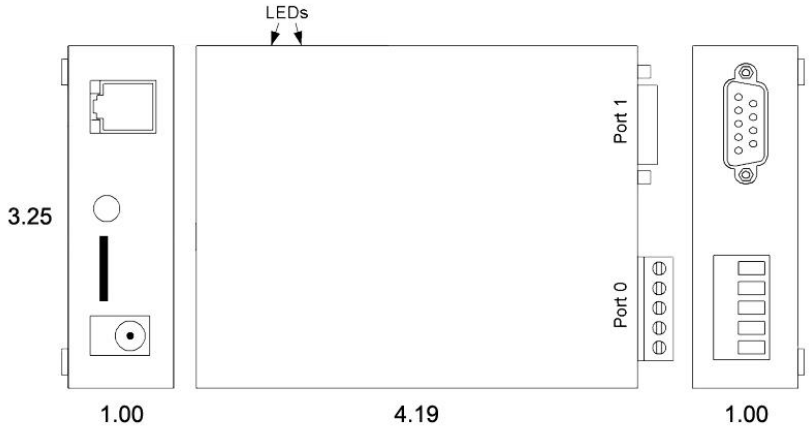
To find which hardware platform you are using:

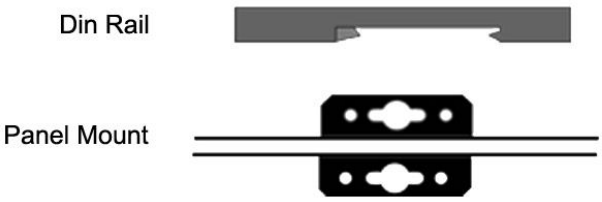
1) Look on the front or back label of the unit for the part number.
2) On the webpage inside the gateway, navigate to the dropdown menu under **Other** and select **Utilities**. Click the **Listing of Revisions** button. The full part number is displayed here.

Once you have the full part number, the platform will be the number following the "-N":



## Hardware – N34

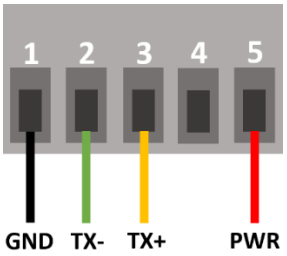


## Powering The Gateway

The following steps will allow you to properly and safely power the gateway.

**Warning improper wiring will cause unit failure! Use the Barrel Connector OR the Screw Terminal's power connection, NOT both!**




1) Connect a 24 VDC power source to the gateway.
    a. The unit draws 125 mA at 24 VDC
    b. The gateway has a voltage operating range from 8-28 VDC, 24 VDC is recommended.

### Hazardous Environment Power & Installation Instructions

This equipment is suitable for use in Class I, Division 2, Groups A, B, C and D, or non-hazardous locations only.

**WARNING – EXPLOSION HAZARD -** Do not disconnect equipment unless power has been removed or the area is known to be non-hazardous.

**WARNING – EXPLOSION HAZARD -** Substitution of components may impair suitability for Class I, Division 2.

**THIS EQUIPMENT IS AN OPEN-TYPE DEVICE AND IS MEANT TO BE INSTALLED IN AN ENCLOSURE SUITABLE FOR THE ENVIRONMENT SUCH THAT THE EQUIPMENT IS ONLY ACCESSIBLE WITH THE USE OF A TOOL.**

**WARNING** – POWER JACK (Screw Terminals, J7) IS FOR MAINTENANCE USE ONLY AND MAY ONLY BE USED WHILE THE AREA IS KNOWN TO BE FREE OF IGNITIBLE CONCENTRATIONS OF FLAMMABLE GASES OR VAPORS. IT IS NOT TO BE CONNECTED UNDER NORMAL OPERATION.

In Hazardous Environments the unit must be powered with between 8-35 VDC, 8 VDC @ 900 mA (7.2 W) max. Supervised. The unit is certified to be operated at -40°C to 50°C.

## Instructions d’alimentation et d’installation pour environnement dangereux

Cet équipement est conçu pour être utilisé uniquement dans des lieux de classe I, division 2, groupes A, B, C et D, ou non dangereux.

**AVERTISSEMENT** - **RISQUE D'EXPLOSION** - Ne débranchez pas l'équipement à moins que le courant ne soit coupé ou que la zone ne présente aucun danger.

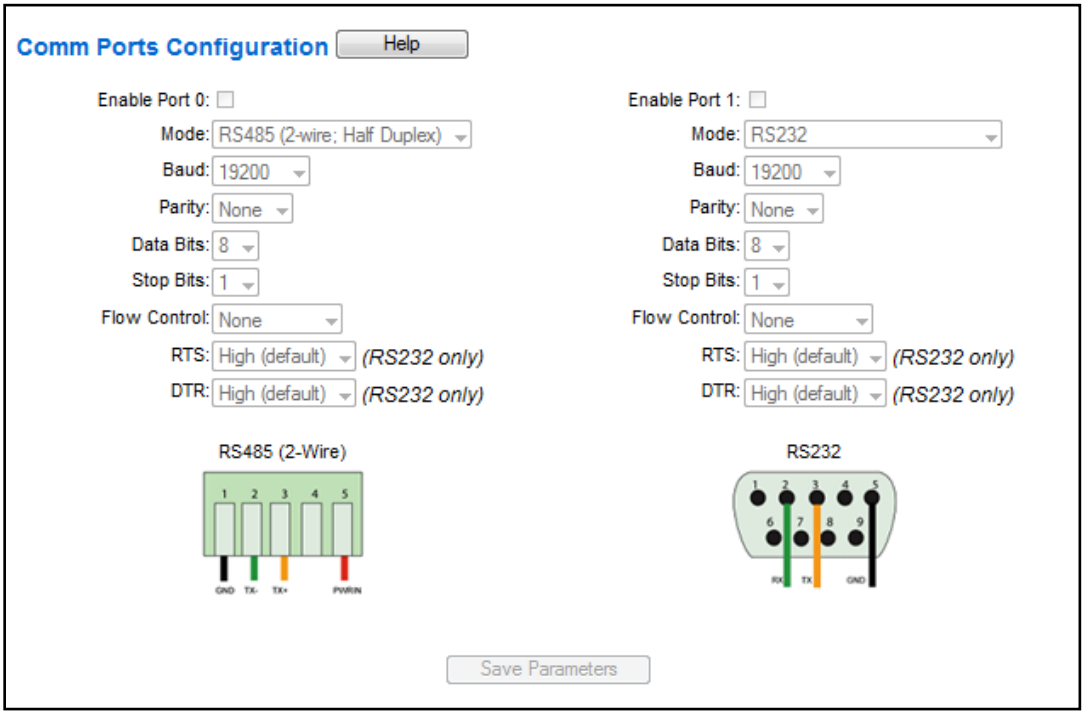
**AVERTISSEMENT** - **RISQUE D'EXPLOSION** - La substitution de composants peut compromettre l'adéquation à la classe I, division 2.

**CET APPAREIL EST UN DISPOSITIF DE TYPE OUVERT ET IL FAUT L'INSTALLER DANS UN ENCEINTE ADAPTÉ À L'ENVIRONNEMENT TEL QU'IL N'EST ACCESSIBLE À L'UTILISATION D'UN OUTIL.**

# Port Configuration

The Port Configuration page is where you set port specific parameters. These settings must match the settings of the device(s) that you are connecting to.

**NOTE:** Either port can be used for RS-232 communications but only on physical port can be used at a time. When you have completed your port configuration, click the **Save Parameters** button.




**The default jumper configurations for most gateway models are setup for the following serial modes:**

- **Port 0 – RS485**
- **Port 1 – RS232**

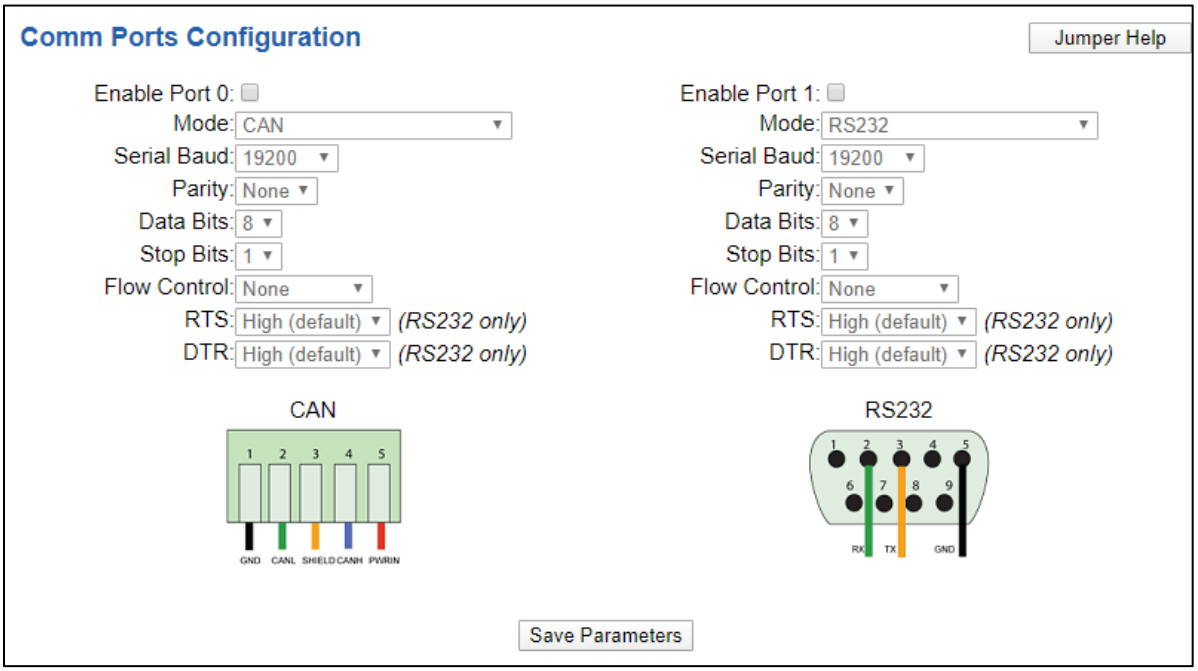
**A list of part numbers with different defaults can be found on the next page.**

**If you require a different serial mode, please refer to the N34_Hardware_Jumper_Configuration.pdf on our website** https://www.rtautomation.com/460-gateway-support/ **to make jumper changes.**

## CAN / RS232

The following products are configured for CAN on Port 1 and RS232 on Port 2 by default:

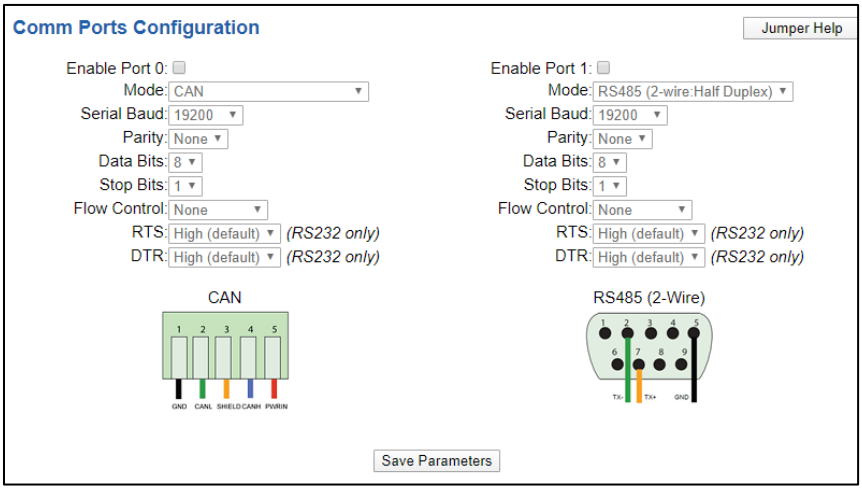
| | | | |
|---|---|---|---|
| 460DMA-N34 | 460DMBC-N34 | 460DMBS-N34 | 460DMDFM-N34 |
| 460DMTCP-N34 | 460DSA-N34 | 460DSBC-N34 | 460DSBS-N34 |
| 460DSDFM-N34 | 460DSTCP-N34 | 460ECDM-N34 | 460ECDS-N34 |
| 460ESDM-N34 | 460ESDS-N34 | 460ETCDM-N34 | 460ETCDS-N34 |
| 460MCDM-N34 | 460MCDS-N34 | 460MSDM-N34 | 460MSDS-N34 |
| 460PSDM-N34 | 460PSDS-N34 | 460DMSC-N34 | 460DSSC-N34 |



## CAN / RS485

The following products are configured for CAN on Port 1 and RS485 on Port 2 by default:

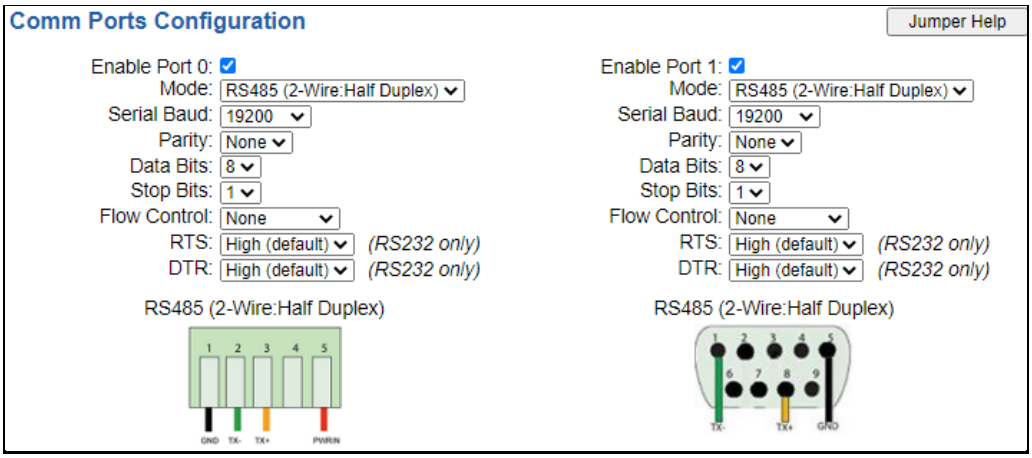
| | | | |
|---|---|---|---|
| 460DMBM-N34 | 460DMBMS-N34 | 460DSBM-N34 | 460DSBMS-N34 |
| 460MMDM-N34 | 460MMDS-N34 | 460MRSDM-N34 | 460MRSDS-N34 |




## Dual RS485

The following products are dual RS485 by default:

| | | | |
|---|---|---|---|
| 460MMBM-N34 | 460MMBMS-N34 | 460MMMRS-N34 | 460MRSBM-N34 |
| 460MRSBMS-N34 | 460MRSMRS-N34 | | |

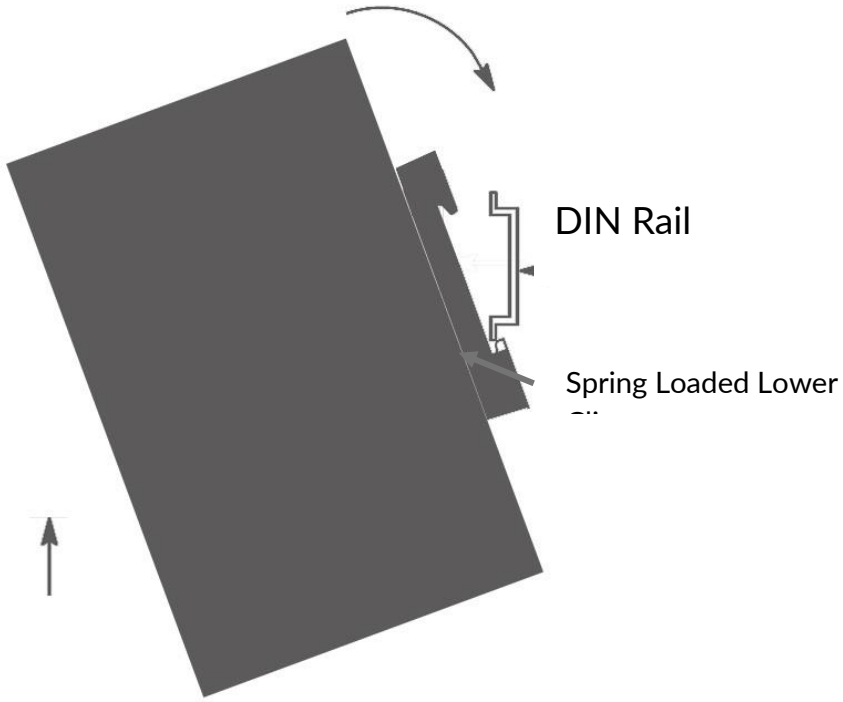


# Mounting with a DIN Rail

## Installing

Follow these steps to install your interface converter.

1) Mount your DIN Rail.
2) Hook the bottom mounting flange under the DIN Rail.
3) While pressing the 460MSDS-N34 against the rail, press up to engage the spring loaded lower clip and rotate the unit parallel to the DIN Rail.
4) Release upward pressure.




## Removing

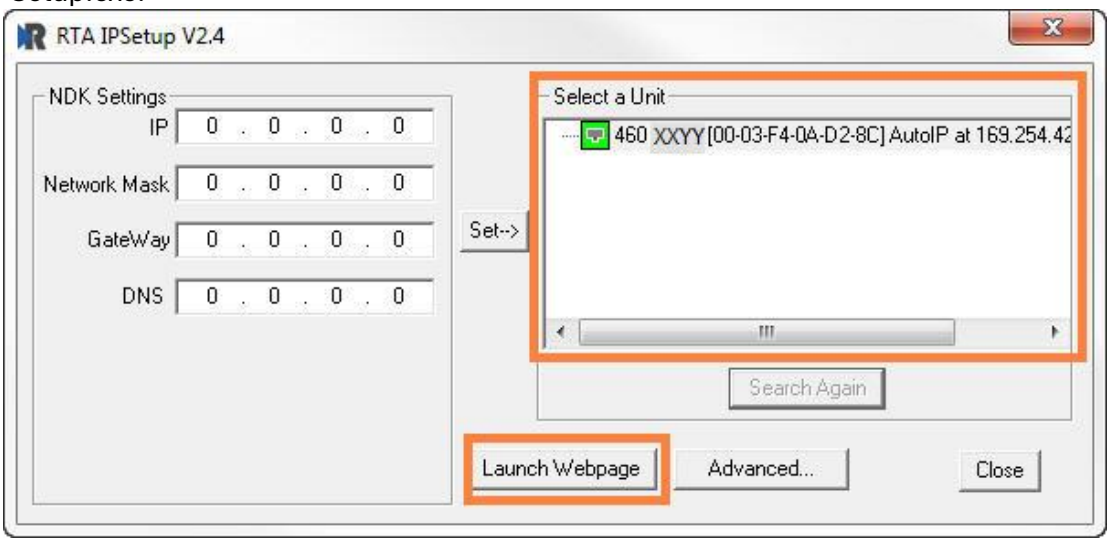
Follow these steps to remove your interface converter.

1) Press up on unit to engage the spring loaded lower clip.
2) Swing top of the unit away from DIN Rail.

# Accessing the Main Page

The following steps will help you access the browser based configuration of the gateway. By default, DHCP is enabled. If the gateway fails to obtain an IP address over DHCP it will Auto IP with 169.254.X.Y. For more information on your Operating system network setting refer to the Accessing Browser Configuration document from our support web site.

1) Scan the QR code on the back of the unit or navigate to www.rtautomation.com/460-gateway-support and download IPSetup.exe.



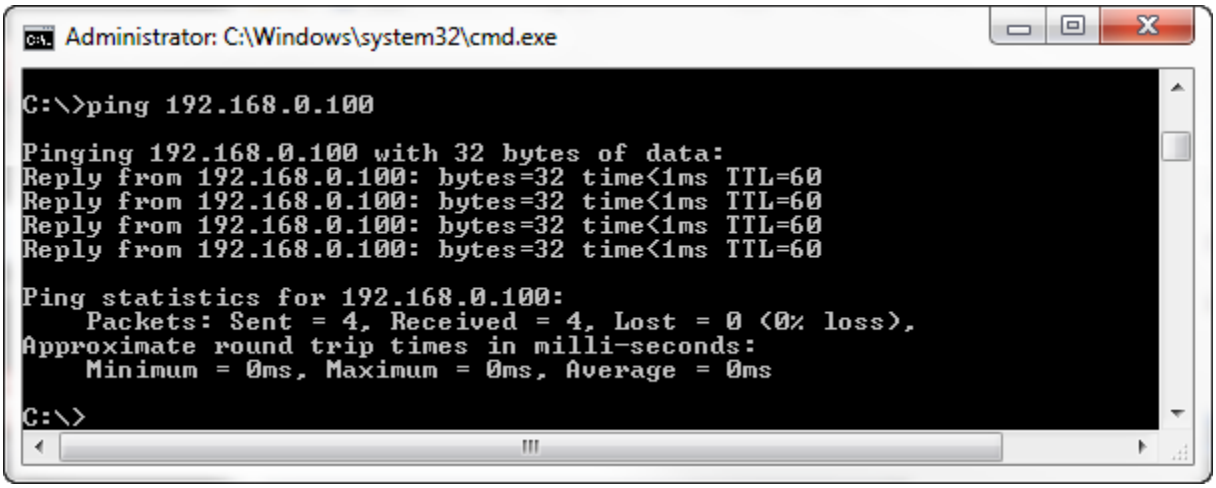

2) Run the IPSetup.exe program.
3) Find unit under “Select a Unit”.
   a. Change Gateway’s IP address to match that of your PC if DHCP has failed.
      i. You will know DHCP has failed if the gateway’s IP address is AutoIP at 169.254.X.Y.
      ii. If successful, it will say DHCP’d at ex: 192.168.0.100 or however your DCHP Client is set up.
   b. If you do not see the gateway in this tool, then your PC is most likely set up as a static IP.
      i. Change your PC’s network settings to be DHCP. If DHCP fails, then it will change to be on the 169.254.x.y network.
      ii. Relaunch the IP Setup tool to see if gateway can be discovered now.
4) Click **Launch Webpage**. The Main page should appear.

**Default setting is set to DHCP. If DHCP fails, default IP Address is 169.254.x.y**

## Error: Main Page Does Not Launch

If the Main Page does not launch, please verify the following:

1) Check that the PC is set for a valid IP Address
   a. Open a MS-DOS Command Prompt
   b. Type "ipconfig" and press enter
   c. Note the PC's IP Address, Subnet, and Default Gateway
2) The gateway must be on the same Network/Subnet as the PC whether it's setup for DHCP or Static. Once you have both devices on the same network, you should be able to ping the gateway using a MS-DOS Command Prompt.



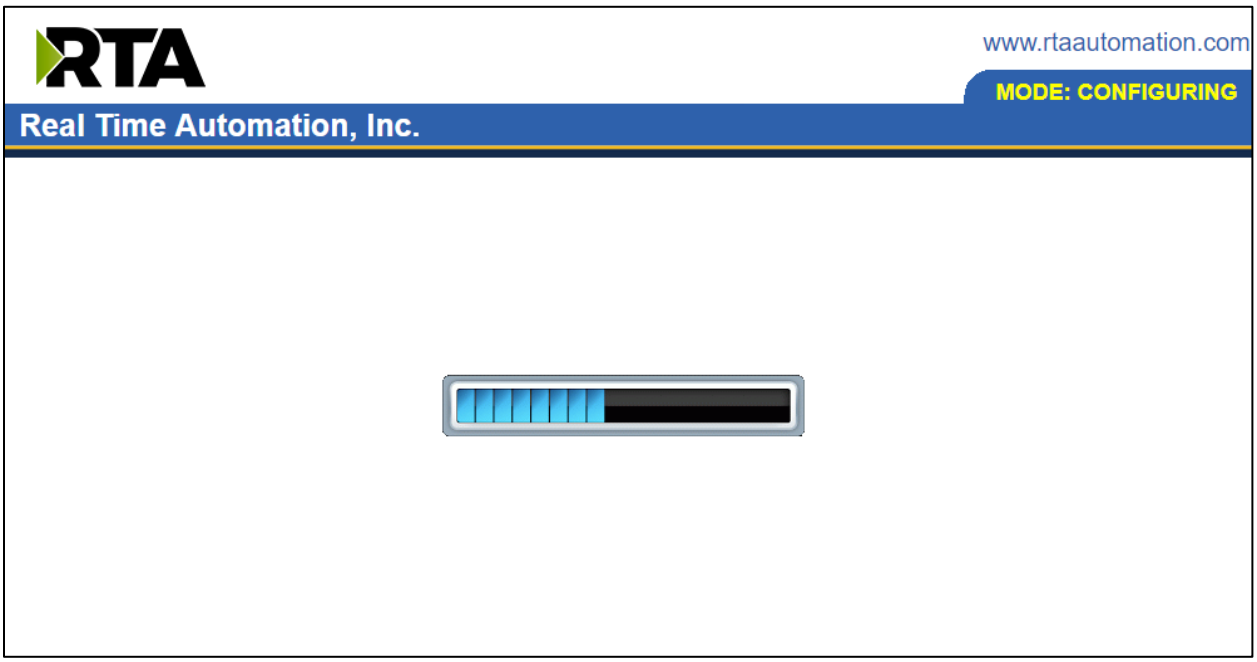

The Screenshot above shows a gateway that is currently set to a static IP Address of 192.168.0.100. If you are able to successfully ping your gateway, open a browser and try to view the main page of the gateway by entering the IP Address of the gateway as the URL.

# Committing Changes to the Settings

All changes made to the settings of the gateway in Configuration Mode will not take effect until the gateway is restarted via the webpage. Changes will not be stored if the gateway's power is removed prior to a reboot.

**NOTE:** The gateway does not need to be restarted after every change. Multiple changes can be made before a restart, but they will not be committed until the gateway is restarted.

When all desired changes have been made, press the **Restart Now** button.
The webpage will redirect to our rebooting page shown below:




The reboot can take up to 20 seconds.
If the IP address has not been modified, the gateway will automatically redirect to the main page.
If the IP address was modified, a message will appear at the top of the page to instruct the user to manually open a new webpage at that new IP.

# Main Page

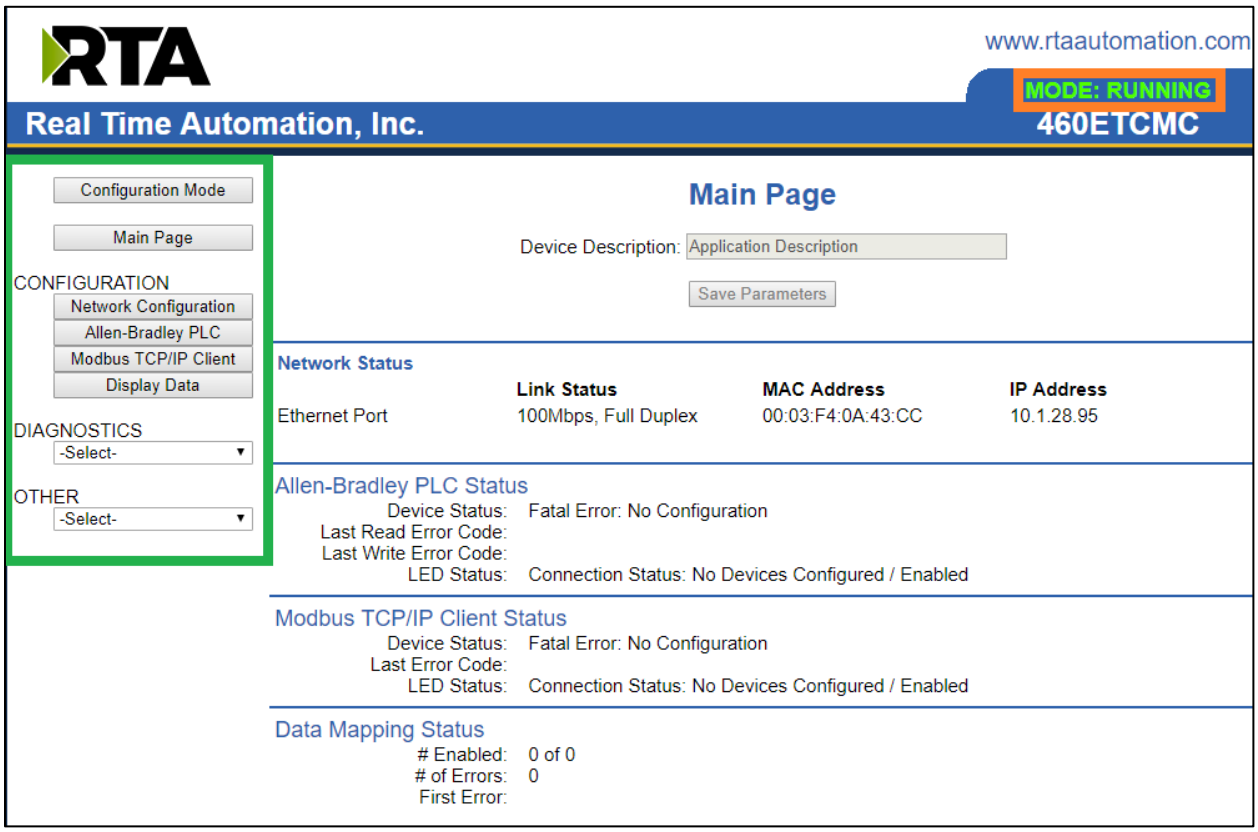
The main page is where important information about your gateway and its connections are displayed.
Mode (orange box below):
Running Mode:

- Protocol communications are enabled
- Configuration cannot be changed during Running Mode. If changes are needed, click the **Configuration Mode** button shown in the green box below

Configuring Mode:

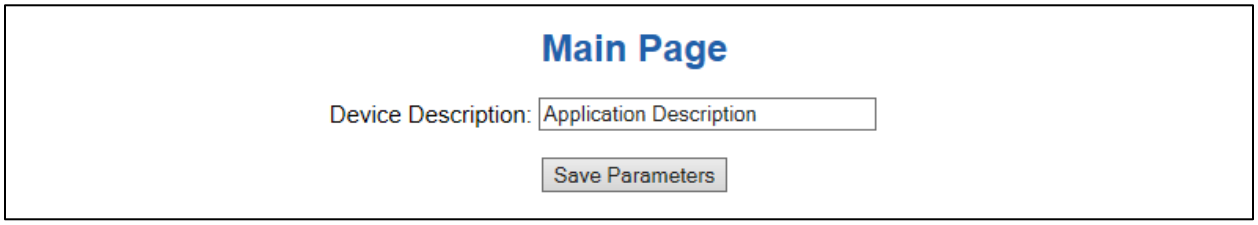
- Protocol communication is stopped and no data is transmitted
- Configuration is allowed

Navigation (green box below):
You can easily switch between modes and navigate between pages (Configuration, Diagnostics, and Other pages) using the buttons on the left hand side.




# Device Configuration

The device configuration area is where you assign the device description parameter. Changes can only be made when the gateway is in Configuration Mode.



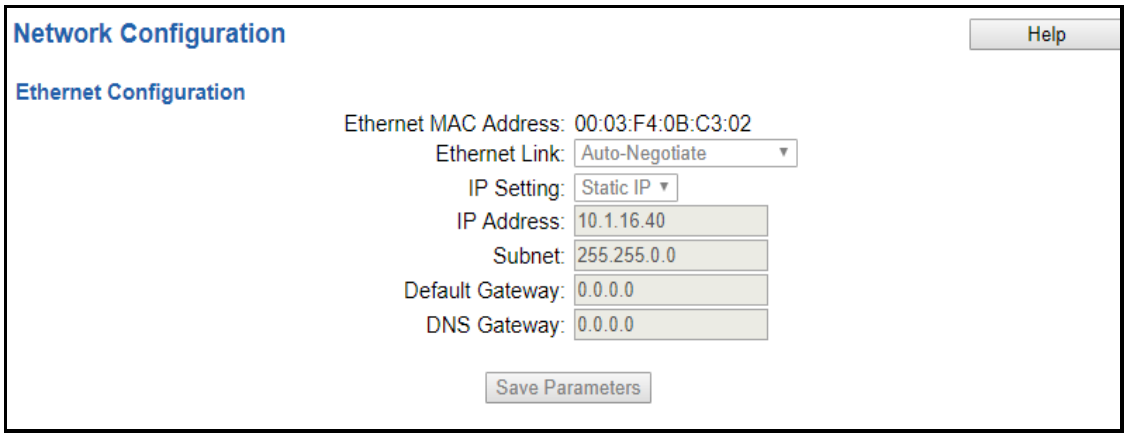

Once you are done configuring the Description, click the **Save Parameters** button.

# Network Configuration

The network configuration area is where you assign the IP address and other network parameters. Changes can only be made when the gateway is in Configuration Mode.

Once you are done configuring the Network Settings, click the **Save Parameters** button.

If you are changing the IP Address of the gateway, the change will not take effect until the unit has been rebooted. After reboot, you must enter the new IP Address into the URL.

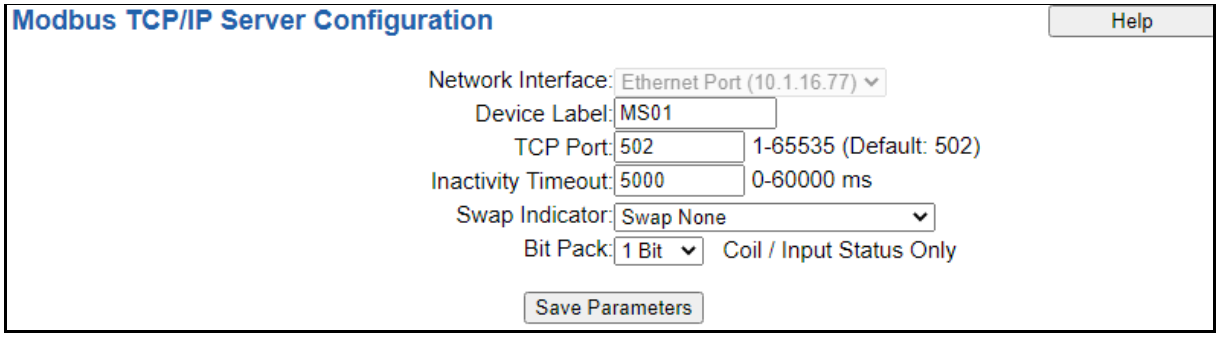



**It is recommended to leave the DNS Gateway set to 0.0.0.0 and the Ethernet Link as Auto-Negotiate. If configuring the gateway to use E-mail, the DNS Gateway must be set.**

# Modbus TCP/IP Server Configuration

Click the **Modbus TCP/IP Server** button to access the configuration page.

1) Select which **Network Interface** to use for this Modbus TCP Client connection. If using single port hardware, the Network Interface will default to Ethernet port only.
2) Enter a **Device Label** to identify the device within the gateway.
3) **TCP Port**: Enter a TCP Port for the Modbus TCP server to listen on. Default for Modbus TCP/IP is 502.
4) **Inactivity Timeout**: Amount of time the gateway will wait for a read/write request before issuing a timeout.
5) To enable data swapping, select the required **Swap Indicator**. If the bytes appear in the wrong order, enable swapping to change the data. This swapping does *NOT* change coils and their ordering inside the Bit Pack.
6) **Bit Pack**: Select the formatting of the Coil Status/Input Status. Automap will use this packing size to map coils to/from the other protocol. The bit pack selection here should match that of the other protocol. The starting address is considered Bit 0 and is the low-order bit.



# Modbus TCP/IP Server Configuration-Data Groups

The bottom area of the Modbus TCP/IP Server Configuration page lets you configure up to 100 data groups for both the read and write.

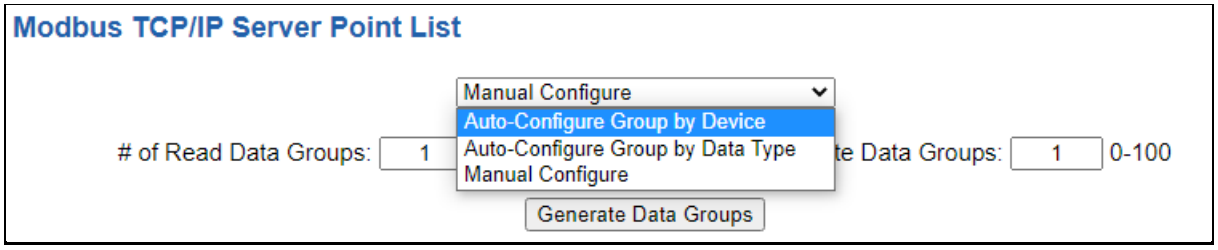
There are three ways to configure this protocol:

1) Auto-Configure Group by Device (Default)

2) Auto-Configure Group by Data Type

3) Manual Mode

**NOTE**: You may go back and forth between modes, but when reverting from Manual Mode to either of the two Auto-Configure modes, all changes made in Manual Mode will be discarded.



## Auto-Configure Group by Device vs. Auto-Configure Group by Data Type

There are two different methods for Auto-Configure: Group by Device or Group by Data Type.

There are a couple of rules to keep in mind when using Auto-Configure Mode:

1) If the other protocol inside the gateway is a server, slave, or adapter protocol, then there are no differences between the Auto-Configure modes.

## Group by Device (Default Method)

Group by Device goes through the other protocol on the gateway and auto-configures the data groups in the Modbus TCP/IP server for all the data points on the other protocol's first device. After it finishes with the first device, it will auto-configure all the points for the second device (if one is configured), and so on.

The data in this method is not optimized- there could potentially be a lot of wasted/unused data space, but it will be organized more logically from the master/client's point of view.

## Group by Data Type

Group by Data Type goes through the other protocol on the gateway and auto-configures the data groups in the Modbus TCP/IP server for all the data points within the other protocol.

Another way to view this option is to say that the data points allocated are packed together so there is very little wasted data space. The data is packed or optimized.

***Example:*** *Protocol A is a master/client protocol that has 2 devices with the same setup:*

*Device_1 has 1 integer scan line, 1 float scan line, 1 integer scan line- each for 1 point of data*
*Device_2 has 1 integer scan line, 1 float scan line, 1 integer scan line- each for 1 point of data*

Protocol B is a server/slave/adapter protocol that can be mapped as follows:

**Group by Device -** Protocol B will have 4 scan lines that will look like the following: Scan Line 1 and 2 will represent Device_1 and Scan Line 3 and 4 will represent Device_2.

Scan Line 1 => Type Integer, length of 2
Scan Line 2 => Type Float, length of 1
Scan Line 3 => Type Integer, length of 2
Scan Line 4 => Type Float, length of 1

**Group by Data Type -** Protocol B will have 2 scan lines that will look like the following: All like data types from Device_1 and Device_2 will be combined.

Scan Line 1 => Type Integer, length of 4
Scan Line 2 => Type Float, length of 2

## Modbus TCP/IP Server Data Group Configuration: Auto-Configure

While in either of the two Auto-Configure modes, the # of Data Groups and the actual data groups themselves cannot be edited. Auto-Configure Mode looks at the other protocol and then configures the data groups to match. The data formats will be defined after the other protocol is configured.

The data will be configured according to the following rules:

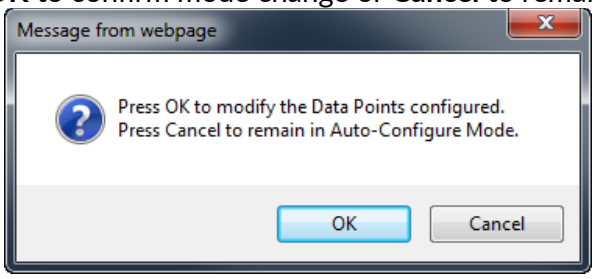
1) Any Coils, 8 Bit Signed/Unsigned, or 1/8/16/32 Bit Binary Packs data will be mapped as **0x Coil Status.**
2) Any 16 Bit Signed/Unsigned data will be mapped as **4x Hold Reg 16 Bit Int or 16 Bit Uint**, matching signs whenever possible.
3) Any 32 Bit Signed/Unsigned data will be mapped as **4x Hold Reg 32 Bit Int or 32 Bit Uint**, matching signs whenever possible.
4) Any 64 Bit Signed/Unsigned data will be mapped as **4x Hold Reg 64 Bit Int or 64 Bit Uint**, matching signs whenever possible.
5) Any 32 Bit Float will be mapped as **4x Hold Reg 32 Bit Float**.
6) Any 64 Bit Float will be mapped as **4x Hold Reg 64 Bit Float**.
7) Any String data types will be mapped as **4x Hold Reg String**.
8) The read or write direction depends on whether it is configured as a read or write on the other protocol.
9) If the other protocol exceeds the number of data groups supported, then nothing will be mapped. You will see the # of Data Groups remain at 0 and the main page will display the following error:

   a) To fix this error, simply decrease the amount of data you configured on the other protocol so that the max number of Data Groups is not exceeded or call customer support to increase the limits.

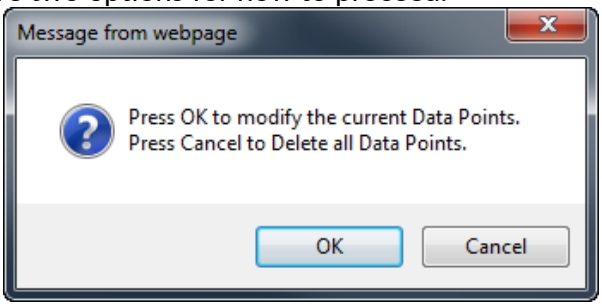
To add additional or edit existing data groups, you will need to go into Manual Configure Mode. If you go back to Auto-Configure Mode, you will lose **ALL** manual edits.

## Modbus TCP/IP Server Data Group Configuration: Manual Mode

1) To transition from either of the two Auto-Configure modes to Manual Configure Mode, click the dropdown at the top of the Modbus TCP/IP server Configuration page and select Manual Configure.
   a) When prompted, click **OK** to confirm mode change or **Cancel** to remain in Auto-Configure Mode.




2) Once OK is clicked, there are two options for how to proceed.

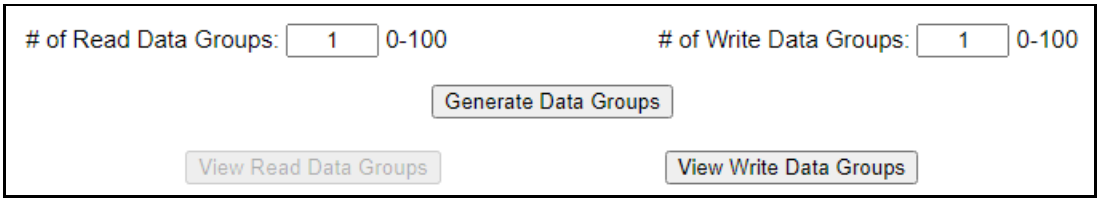



3) To keep the data groups that are already configured, press **OK**.

   a) You would want this option if you are adding additional data groups or you want to modify the data group(s) that already exist.

4) To delete the data groups that are already there and start over, press **Cancel**.

5) Enter the number of Read Data Groups and/or Write Data groups.

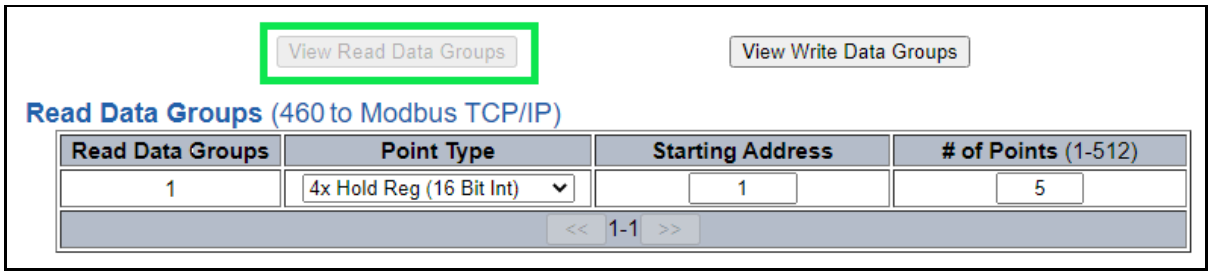


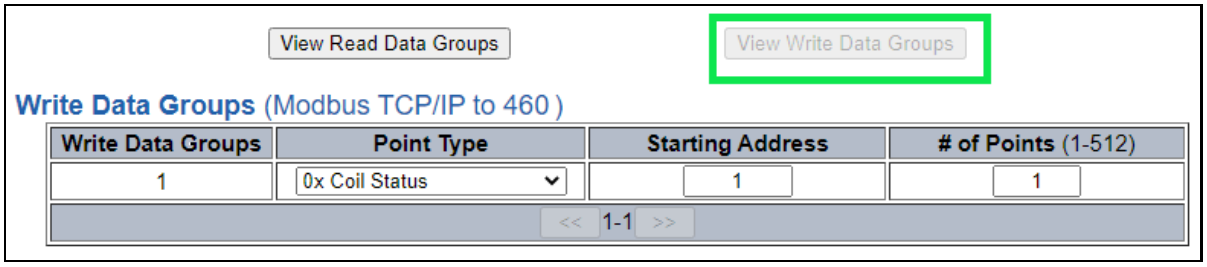

6) Click the **Generate Data Groups** button to have the read and write data groups auto-generate for you. You may manually configure the read and write data groups after they have been generated.

## Configure Read and Write Data Groups

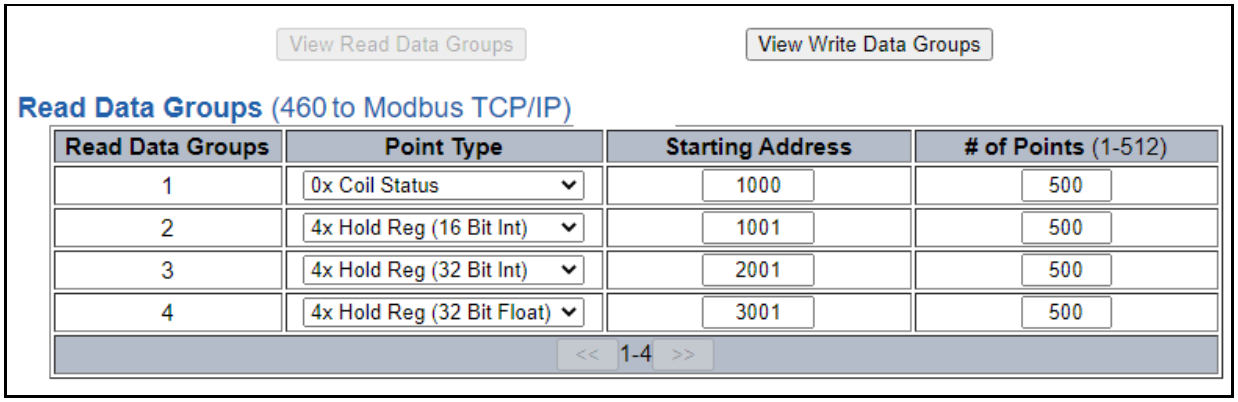
Follow these steps to manually configure read or write data groups.

1) Select **View Read Data Groups** or **View Write Data Groups** if not already selected.



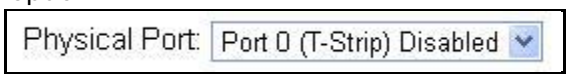

2) Select a **Point Type** for each scan line. Options include: Coil Status, Input Status, Input Registers, and Holding Registers. **Note**: Input/Holding Registers have a data type associated with them.
   a) String Point Type- If the mating protocol supports strings, you may select string as a point type in Modbus. With this point type, 2 characters will be packed into a single register and the first register will be set aside for the length.
   b) EX: 4x Hold Reg (String) with a Starting Address of 1 for a length of 5 Registers. This means that Register 1 will hold the length of the string and Registers 2-5 will hold the string contents. This string can contain a max of 8 characters.
3) Enter a **Starting Address** (1-based).
4) Enter the **# of Points** to read or write. This will allocate the number of the data type selected.



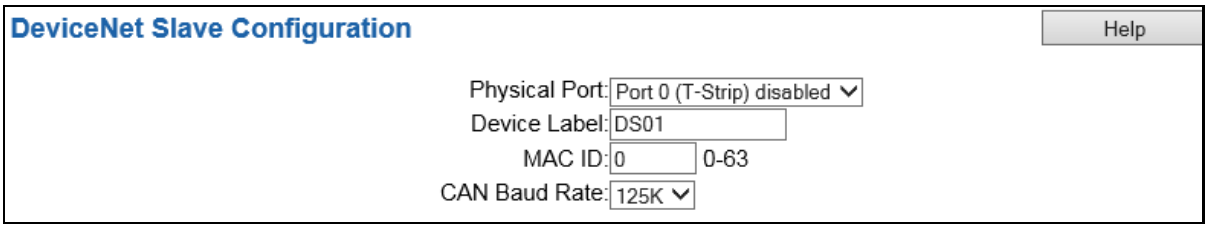
# DeviceNet Slave Configuration

Click the **DeviceNet Slave** button to access the configuration page.

1) **Physical Port**: Select which physical port is being used for communication. This port must be configured on the Port Configuration page. If it has not yet been configured, it will display *Disabled* after the Port descriptions in this dropdown.




2) **Device Label**: Enter a label to identify the device within the gateway.
3) **MAC ID:** Enter a unique DeviceNet MAC ID for the device on the network.
4) **CAN Baud Rate**: Select a CAN baud rate for the network. This must match the other devices on the network.



# DeviceNet Slave Data Group Configuration

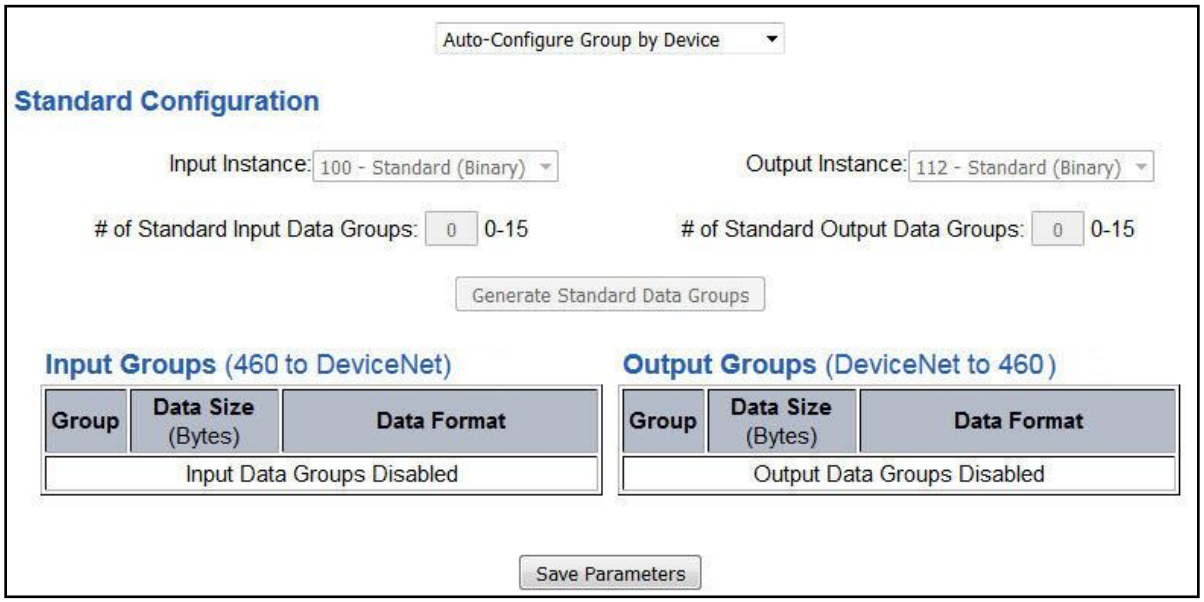
The middle area of the DeviceNet Slave Configuration page lets you configure multiple Input and Output Groups.

1) **Input Instance:** Select the input instance to communicate to.
2) **Output Instance:** Select the output instance to communicate to.
3) Data Size is configurable from 0-480 bytes per group. The total bytes in a single direction (Input/Output) cannot exceed 480 bytes across all data groups.
4) Data format sets the formatting of the data. Automap will use this packing size to map data to/from the other protocol.

There are three ways to configure this protocol:

- Auto-Configure Group by Device (Default)
- Auto-Configure Group by Data Type
- Manual Mode

**NOTE**: You may go back and forth between modes, but when reverting from Manual Mode to either of the two Auto-Configure Modes, all changes made in Manual Mode will be discarded.



## DeviceNet Slave Data Group Configuration: Auto-Configure

While in either of the two Auto-Configure Modes, the # of Data Groups and the actual Data Groups themselves cannot be edited. Auto-Configure Mode looks at the other protocol and then configures the Data Groups to match. The Data Formats will be defined after the other protocol is configured.

The data will be configured according to the following rules:

1) Any 8 Bit Signed/Unsigned data will be mapped as **8 Bit Int or 8 Bit Uint**, matching signs whenever possible.
2) Any 16 Bit Signed/Unsigned data will be mapped as **16 Bit Int or 16 Bit Uint**, matching signs whenever possible.
3) Any 32 Bit Signed/Unsigned data will be mapped as **32 Bit Int or 32 Bit Uint**, matching signs whenever possible.
4) Any 64 Bit Signed/Unsigned data will be mapped as **64 Bit Int or 64 Bit Uint**, matching signs whenever possible.
5) Any 32-Bit Float will be mapped as **32-Bit Float**.
6) Any 64-Bit Float will be mapped as **64-Bit Float**.
7) Any coils or 1/8/16/32 Bit Binary Packs will be mapped as **Binary 8 Bit Pack/Binary 16 Bit Pack/Binary 32 Bit Pack**, matching bit sizes whenever possible.
8) The read or write direction depends on whether it is configured as a read or write on the other protocol.
9) If the other protocol exceeds the number of data groups supported, then nothing will be mapped. You will see the # of Data Groups remain at 0 and the main page will display the following error:

   a) To fix this error, simply decrease the amount of data you configured on the other protocol so that the max number of Data Groups is not exceeded or call customer support to increase the limits.

To add additional or edit existing data groups you will need to go into Manual Configure Mode.

## Auto-Configure Group by Device vs. Auto-Configure Group by Data Type

There are two different methods for Auto-Configure: Group by Device or Group by Data Type.

There are a couple of rules to keep in mind when using Auto-Configure Mode:

1) If the other protocol inside the gateway is a server, slave, or adapter protocol, then there are no differences between the Auto-Configure Modes.

## Group by Device (Default Method)

Group by Device goes through the other protocol on the gateway and auto-configures the data groups in the DeviceNet slave for all the data points on the other protocol's first device. After it finishes with the first device, it will auto-configure all the points for the second device (if one is configured), and so on.

The data in this method is not optimized- there could potentially be a lot of wasted/unused data space, but it will be organized more logically from the master/client's point of view.

## Group by Data Type

Group by Data Type goes through the other protocol on the gateway and auto-configures the data groups in the DeviceNet slave for all the data points within the other protocol.

Another way to view this option is to say that the data points allocated are packed together so there is very little wasted data space. The data is packed or optimized.

***Example:*** *Protocol A is a master/client protocol that has 2 devices with the same setup:*

*Device_1 has 1 integer scan line, 1 float scan line, 1 integer scan line- each for 1 point of data*
*Device_2 has 1 integer scan line, 1 float scan line, 1 integer scan line- each for 1 point of data*

Protocol B is a server/slave/adapter protocol that can be mapped as follows:

**Group by Device -** Protocol B will have 4 scan lines that will look like the following: Scan Line 1 and 2 will represent Device_1 and Scan Line 3 and 4 will represent Device_2.

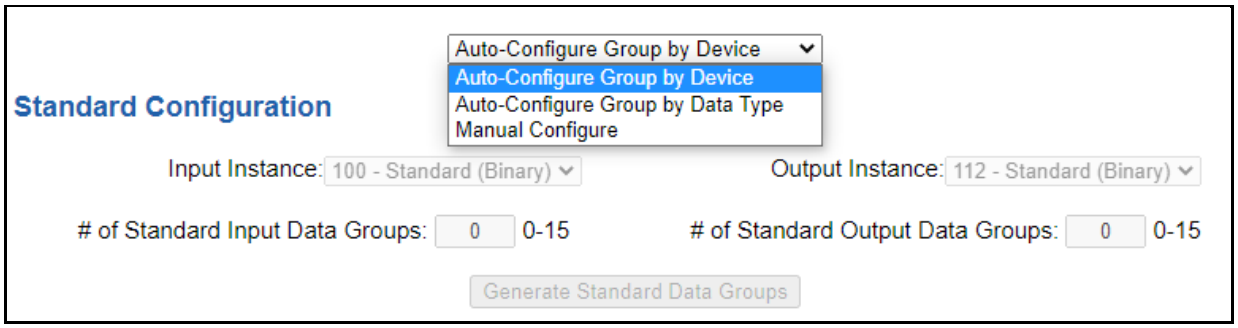
Scan Line 1 => Type Integer, length of 2
Scan Line 2 => Type Float, length of 1
Scan Line 3 => Type Integer, length of 2
Scan Line 4 => Type Float, length of 1

**Group by Data Type -** Protocol B will have 2 scan lines that will look like the following: All like data types from Device_1 and Device_2 will be combined.

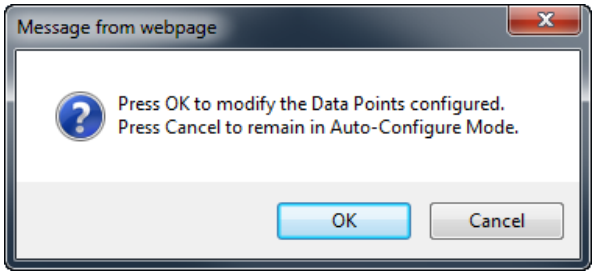
Scan Line 1 => Type Integer, length of 4
Scan Line 2 => Type Float, length of 2

## DeviceNet Slave Data Group Configuration: Manual Mode

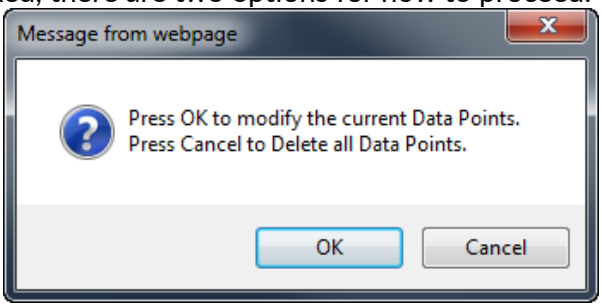
1) To transition from either of the two Auto-Configure Modes to Manual Configure Mode, click the dropdown at the top of the DeviceNet Slave Configuration page and select Manual Configure.




   a. When prompted, click **OK** to confirm mode change or **Cancel** to remain in Auto-Configure Mode.




   b. Once OK is clicked, there are two options for how to proceed.




      i. To keep the data groups that are already configured, press **OK**.
         - You would want this option if you are adding additional data groups or you want to modify the data group(s) that already exist.

      ii. To delete the data groups that are already there and start over, press **Cancel**.

2) Select the Input Instance and Output Instance to communicate to.
**NOTE**: If you selected Input Instance 101 (String) or 102 (String), then you do not need to configure the Standard Data Groups. Skip steps 3-5 below.
**NOTE**: If you selected Output Instance 113 (String) or 114 (String), then you do not need to configure the Standard Data Groups. Skip steps 3-5 below.

3) Enter the number of Input Data Groups and Output Data groups.
4) Click the **Generate Standard Data Groups** button to have the read and write data groups auto-generate for you. You may manually configure the read and write data groups after they have been generated.
5) Select a Data Format for each Data Group. Options include: 8 Bit Int/Unit, 16 Bit Int/Uint, 32 Bit Int/Uint, 64 Bit Int/Uint, 32/64 Bit Float, Binary 8/16/32 Bit Pack.
   **NOTE:** The Connection Parameters and Input/Output Data Mapping tables will update to match the values entered in the Data Groups after **Save Parameters** is pressed.

## DeviceNet Slave Data Group Configuration: Using Strings

Any String Data Types will be mapped to a String Assembly and will not display in the Input/Output Data Groups.

When using a String Data Type, the first byte is set aside for the length. Each String may have 99 characters.

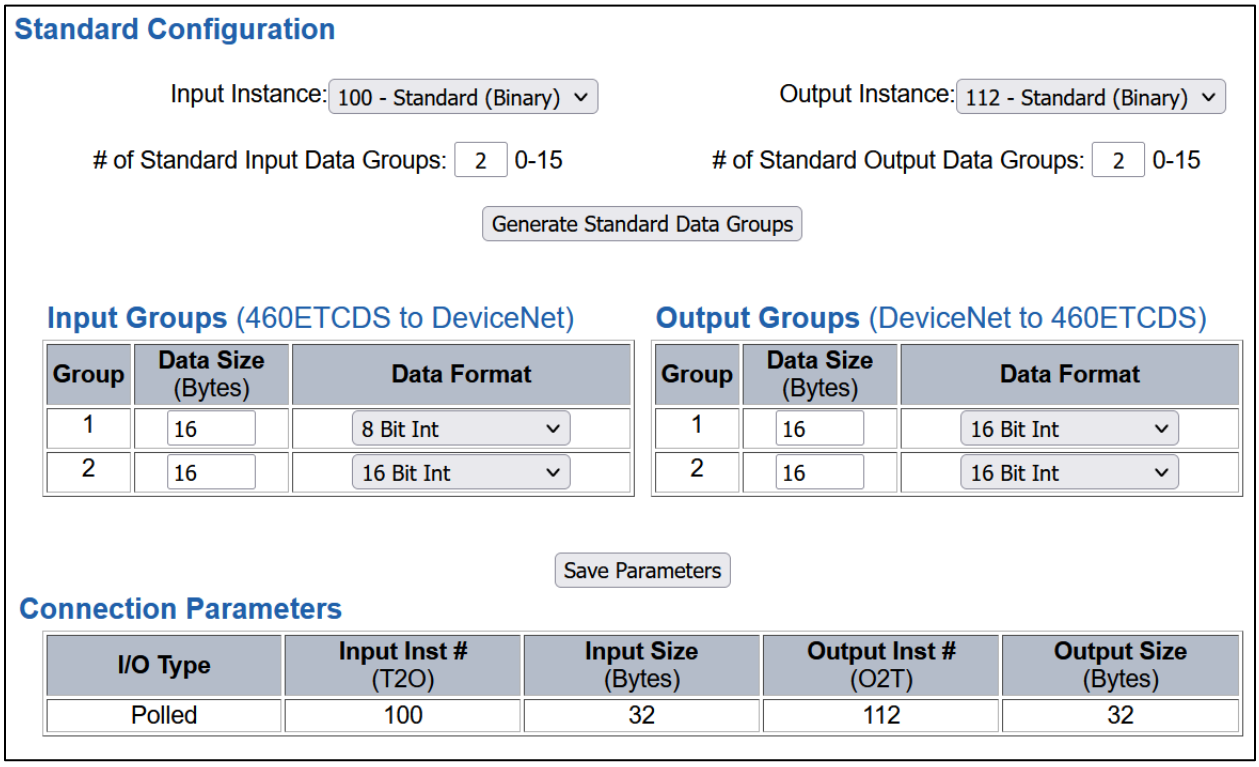
When using 1 String, use Input Assembly 101/Output Assembly 113 where Byte 0 would be the length and Byte 1-99 would contain the data.

When using 2 Strings, use Input Assembly 102/Output Assembly 114 where Byte 0 would be the length of String 1 and Byte 1-99 would contain the data of String 1 and Byte 100 would be the length of String 2 and Byte 101-199 would contain the data of String 2.

## DeviceNet Slave Data Group Configuration-Connection Parameters

The lower area of the DeviceNet Slave Configuration page gives a list of parameters to be used to connect to the gateway from your DeviceNet master device. After data is configured for the data groups and the appropriate input/output instances are selected from the dropdowns, the applicable connection will display below in the table. Use the instance # and size next to the connection for information on how to connect to a DeviceNet master.

**Example Setup:**




To establish communications with the above example settings, configure the DeviceNet master with the following:

- **Input Instance** or Destination for Target to Originator as 100.
- **Output Instance** or Destination for Originator to Target as 112.
- Data **Input Size** for Target to Originator as 32 Bytes.
- Data **Output Size** for Originator to Target as 32 Bytes.

# Mapping - Transferring Data Between Devices

There are 5 ways to move data from one protocol to the other. You can combine any of the following options to customize your gateway as needed.

**Option 1 – Data Auto-Configure Mappings:** The gateway will automatically take the data type (excluding strings) from one protocol and look for the same data type defined in the other protocol. If there isn't a matching data type, the gateway will map the data to the largest available data type. See Data Auto-Configure section for more details.

**Option 2 – String Auto-Configure:** The gateway will automatically take the string data type from one protocol and map it into the other. See String Auto-Configure section for more details.

**Option 3 – Manual Configure Mappings:** If you don't want to use the Auto-Configure Mappings function, you must use the manual mapping feature to configure translations.

**Option 4 – Manipulation/Scaling:** You can customize your data by using math operations, scaling, or bit manipulation. See Data Mapping-Explanation section for more details.

**Option 5 – Move Diagnostic Information:** You can manually move diagnostic information from the gateway to either protocol. Diagnostic information is not mapped in Auto-Configure Mappings Mode. See Diagnostic Info section for more details.

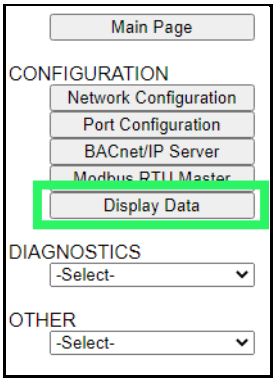
# Display Mapping and Values

The Display Data and Display String pages are where you can view the actual data for each mapping that is set up.

## Display Data

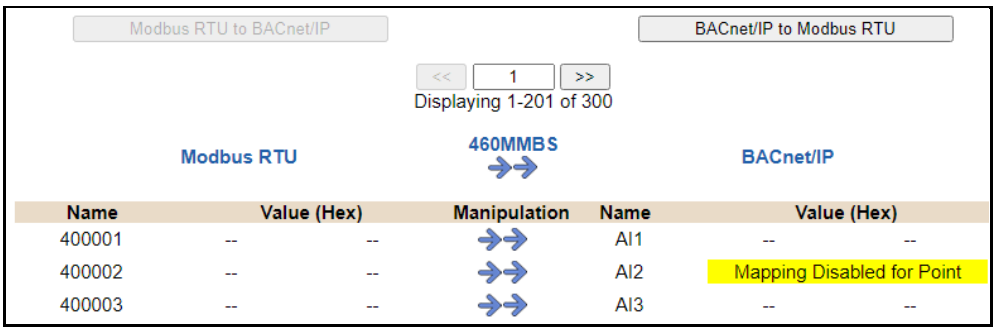
Click the **Display Data** button to view how the data is mapped and what the values of each mapping are.

**Going from Manual Mapping to Auto-Mapping will delete ALL mappings and manipulations configured.**




Here you will see how each data point (excluding strings) is mapped. To view, select the device from the dropdown menu and click **View** to generate the information regarding that device. Then select either the ***Protocol 1 to Protocol 2*** or ***Protocol 2 to Protocol 1*** button, correlating to the direction you wish to see the data.

This page is very useful when verifying that all data is mapped somehow from one protocol to another. If a data point is not mapped, it will display on this page in a yellow highlighted box. The Display Data page will display up to 200 mappings per page, simply navigate to the next page for the additional mapping to display.




In the above example, we see the following:

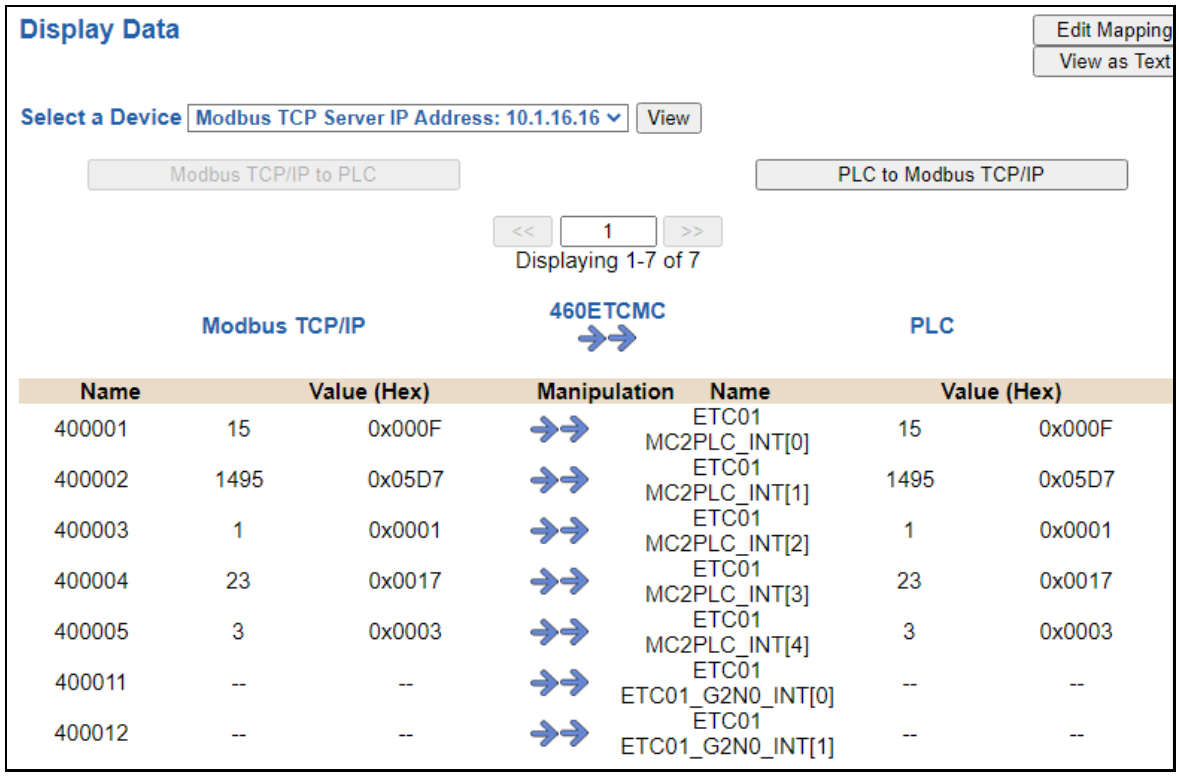
- Modbus register 400001 from Slave 1 is being mapped to AI1 on BACnet
- Nothing is being moved from Modbus register 400002 to AI2 on BACnet because the mapping is disabled
- Modbus register 400003 from Slave 1 is being mapped to AI3 on BACnet

**NOTE**: If a data point is mapped twice, only the first instance of it will show here. EX: If Modbus 400001 & 400040 from Slave 1 are both mapped to AI1, only 400001 will show as being mapped to AI1.

If there are values of "- - "on this page, it indicates that the source has not yet been validated and no data is being sent to the destination.

The example below reflects the Modbus to PLC flow of data. The Modbus (left side) is the source and the PLC (right side) is the destination.

- The 460 gateway has received valid responses from Modbus registers 400001- 400005 and therefore can pass the data on to the PLC tag called MC2PLC_INT.
- The 460 gateway has NOT received valid responses from Modbus register 400011 & 400012. As a result, the data cannot be passed to the PLC tag ETC01_GN0_INT2 and indicates so by using "- - "in the value column of the table.

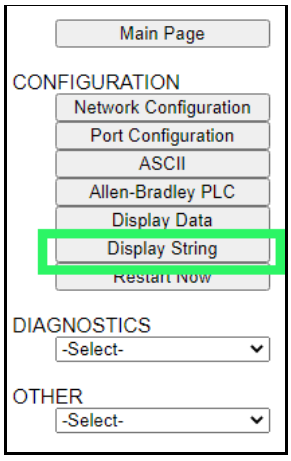



To view the actual data mappings, click the **Edit Mapping** button. For more details, see the Data Mapping-Explanation section.

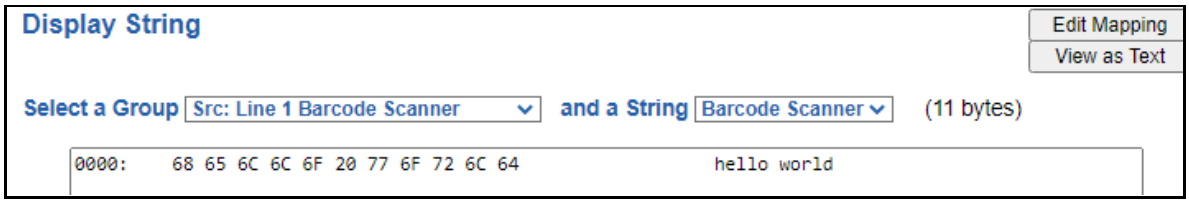
To view the data mappings purely as text, click the **View as Text** button. For more details, see the View Data Mapping as Text section.

## Display String

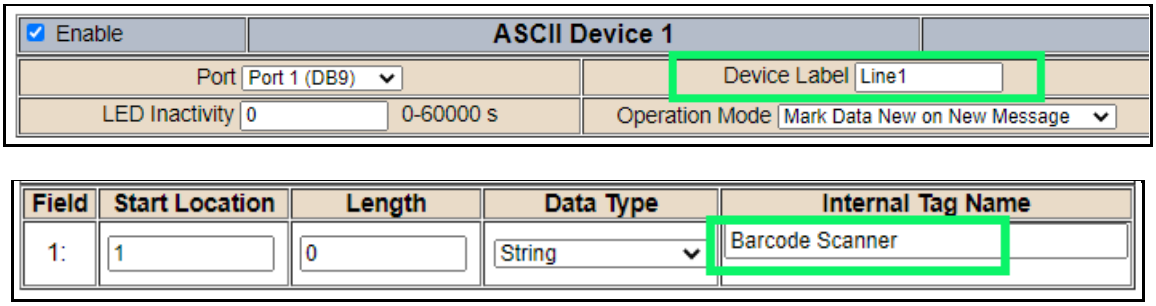
Click the **Display String** button to view what the values of each Parsing and/or Concatenating strings are, you can also click on the Edit Mapping to view the mapping of each string.




To view the source or destination groups from a string, click the dropdown menu to generate the information regarding that device. The string data will be displayed in both Hex and ASCII (only the ASCII data is sent). The example below shows data that is coming from the source device. A group will be displayed for each Parsing/Concatenating String field that is configured.

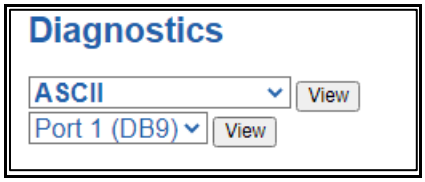



In the Group drop down, “Line1” is defined on the ASCII Device configuration page and “Barcode Scanner” is defined in the ASCII Parsing configuration.

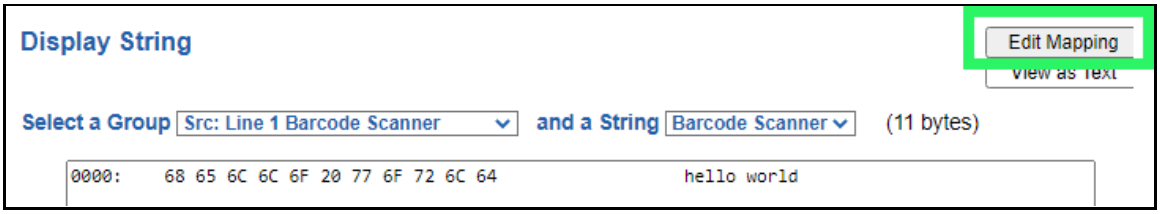


If there are values of “Data Not Valid “on this page, it indicates that the source has not been validated yet and no data is being sent to the destination.

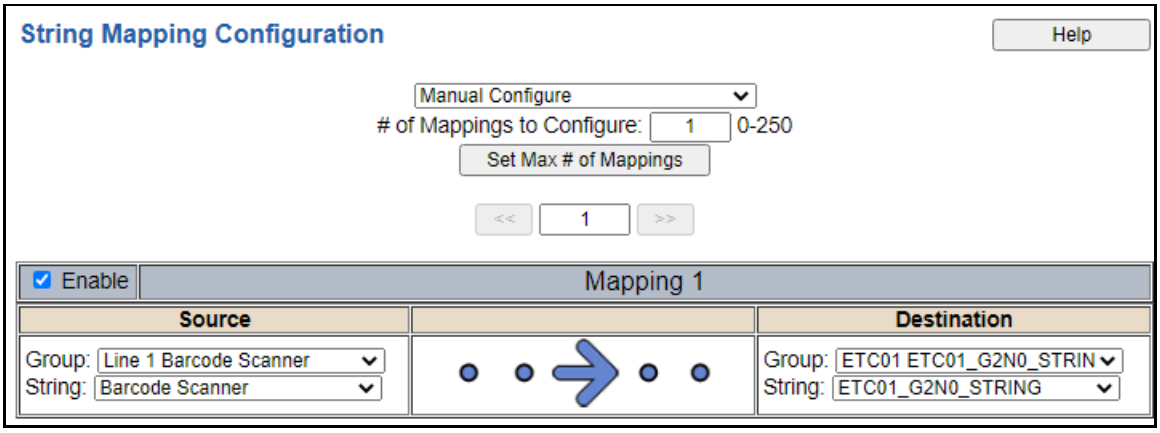
**NOTE:** You can view the whole string data by clicking on **Diagnostics Info** drop down and navigating to ASCII Diagnostics page. You will also have to select the port you want to view in the dropdown below ASCII.




To view the string mappings, click the **Edit Mapping** button. For more details see the **String Mapping-Explanation** section.



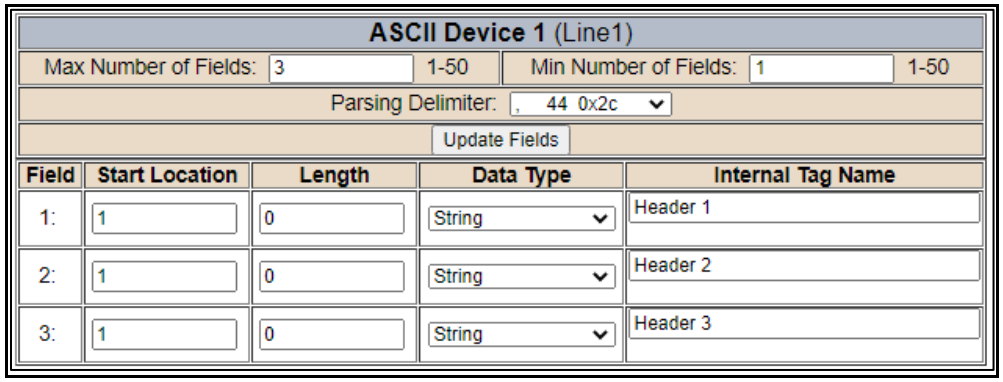

**NOTE: Only String data types can be mapped to another String data type.**




To view the string mappings purely as text, click the **View as Text** button. For more details see the **View String Mapping** as Text section.

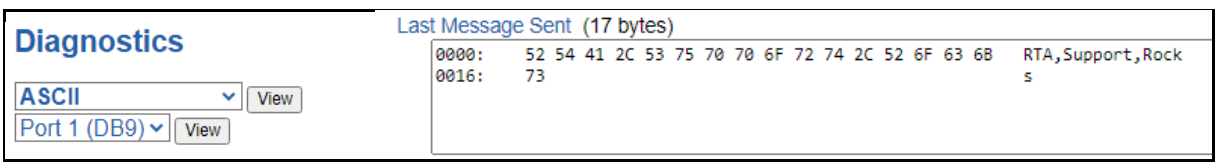
# Display String use case

Sending a message of “RTA,Support,Rocks” from an ASCII device to the RTA unit. The ASCII Parsing Configuration would look like my example below. There are more detailed examples of what all the fields represent in the ASCII Parsing section.




The message is broken up into 3 “Groups” or Parsing fields.

To view the Entire message, click on the Diagnostic drop down, select Diagnostics Info. Select ASCII, click view, select your Port. Whole data will be in the Last Message Sent Diagnostic box.



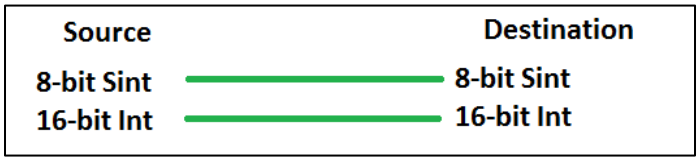
# Data and String Mapping – Auto-Configure

The Auto-Configure function looks at both protocols and will map the data between the two protocols as best as it can so that all data is mapped. Inputs of like data types will map to outputs of the other protocols like data types first. If a matching data type cannot be found, then the largest available data type will be used. Only when there is no other option is data truncated and mapped into a smaller data type.

If the Auto-Configure function does not map the data as you want or you want to add/modify the mappings, you may do so by going into Manual Configure mode.

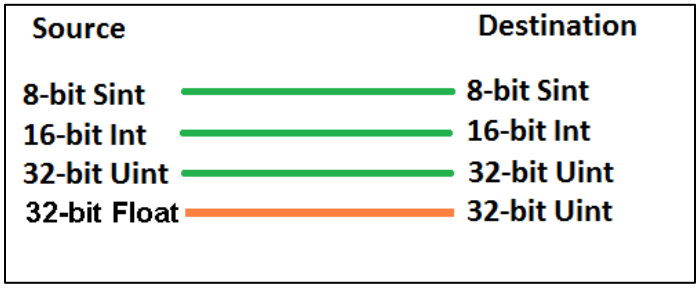
The following are examples of the Auto-Configure function.

1) This example shows a common valid setup.




   a. Both Source values were able to be mapped to a corresponding Destination value.

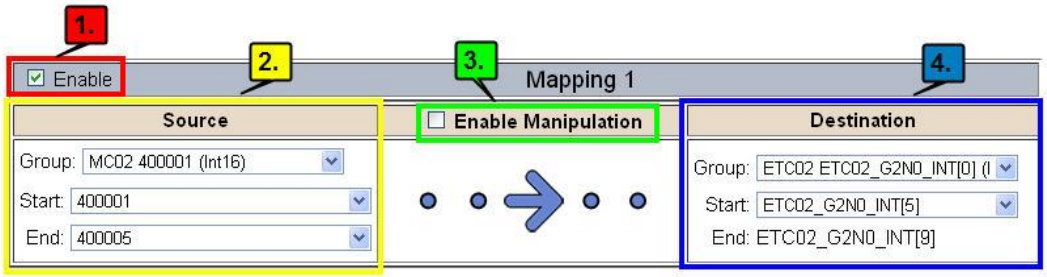
2) This example shows how Auto-Configure will make its best guess.



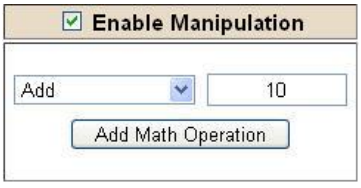

   a. The 32-bit Float from the Source location could not find a matching Destination data-type. After all other like data types were mapped, the only data type available was the 2nd 32-bit Uint data type. Auto-Configure was completed even though the data in the Float will be truncated.

# Data Mapping – Explanation

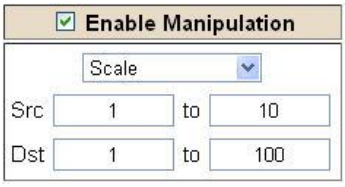
Below are the different parts that can be modified to make up a data mapping.



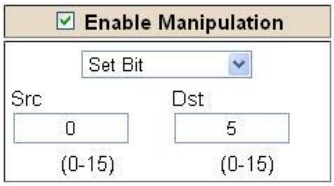

1) Enable (red box above): Check to enable mapping. If not checked, this mapping is skipped.
2) Source Field (yellow box above):
   a) Group - Select the data group you set up in the protocol config to use for this mapping.
   b) Start - This is the starting point for this mapping.
   c) End - This is the final point to be included for this mapping.
3) Manipulation Area (green box above):
   a) Enable the Data Manipulation. This can be enabled for any mapping.
   b) Click **Add Math Operation** for each operation needed. Up to 3 are allowed unless you are using the Scale, Set Bit, or Invert Bit functions. If using Scale, Set Bit, or Invert Bit, then only 1 operation is allowed.
   c) Select the Operation(s) to perform.
      i) Math Operations are performed in the order they are selected.
      ii) If more than one point is selected on the source, the Math Operations will be performed on every point.
   d) Enter the value(s) for the operation.




*Example of Add (similar for Subtract, Multiple, Divide, and MOD). This will add a value of 10 to the source field before it is written to the destination field.*




*Example of Scale. This will scale the source values from 1-10 into 1-100 for the destination.*




*Example of Set Bit (similar to Invert Bit). This will take the value of the 0th source bit and copy it into the value of the 5th destination bit.*

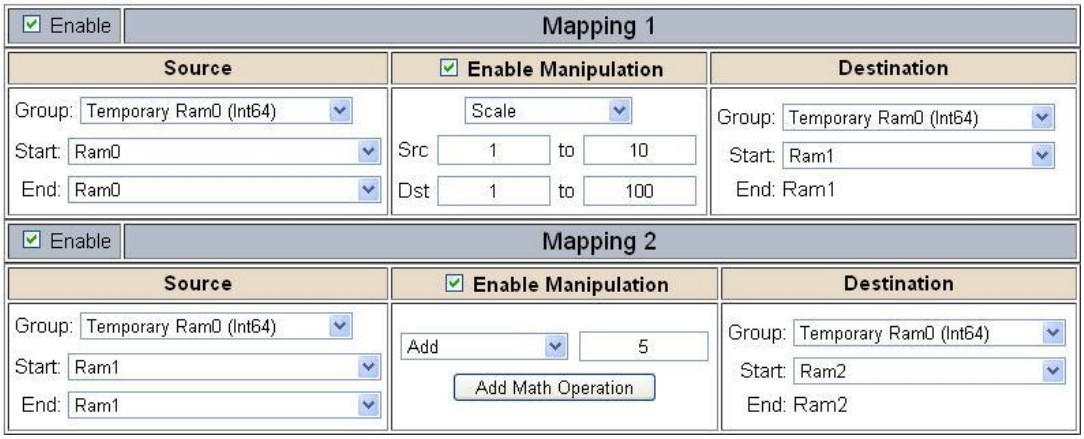
4) Destination Field (blue box above):
   a) Group - Select the data group you set up in the protocol config to use for this mapping.
   b) Start - This is the starting point for where the data is being stored.
   c) End - The End point is derived from the length of the source and cannot be modified.

# Data Mapping – Adding Diagnostic Information

Data Mapping offers 5 different types of information in addition to any scan lines specified for each protocol.

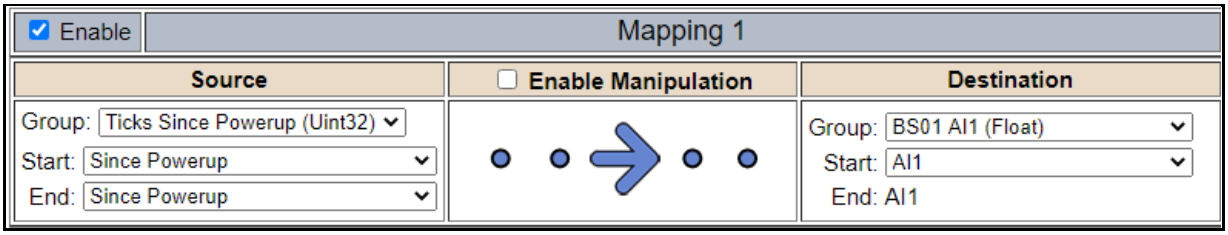
**IMPORTANT NOTE:** Only add Diagnostic Information **AFTER** both sides of the gateway have been configured. If changes to either protocol are made after diagnostic information has been added to the mapping table, it is necessary to verify all mappings. Remapping may be

1) **Temporary Ram (Int64)**
   a) This offers five levels of 64bit Integer space to assist in multiple stages of math operations. For example, you may wish to scale and then add 5. You can set up a single translation to scale with the destination as the temporary ram. Then another translation to add 5 with the source as the temporary ram.
   b) The gateway will automatically convert the Source to fit the Destination, so there is no need for Int 8, 16, 32 since the 64 may be used for any case.

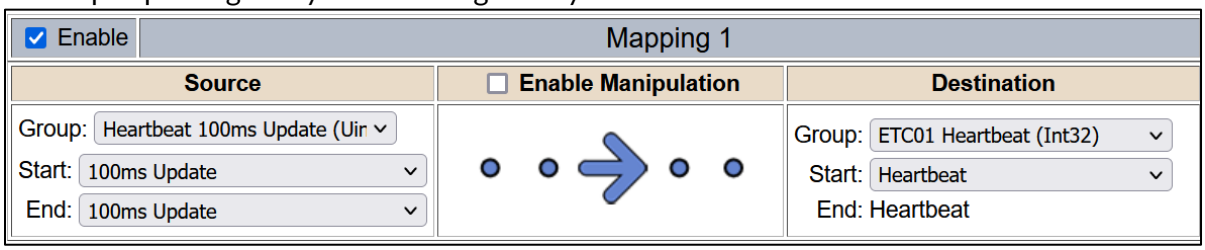



*In this example, Ram0 is scaled into Ram1. Ram1 is then increased by 5 and stored into Ram2. Ram0 and Ram2 could be considered a source or destination group.*

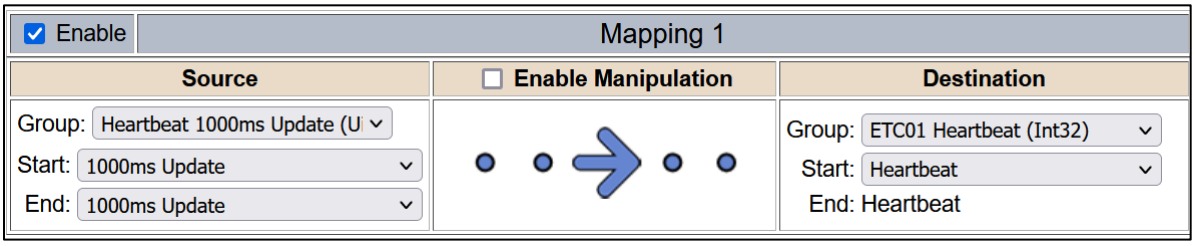
2) **Temporary Ram (Double)**
   a) This is like the Temporary Ram (Int 64), except manipulations will be conducted against the 64bit floating point to allow for large data.
3) **Ticks Per Second**
   a) The gateway operates at 200 ticks per second. This equates to one tick every 5ms. Thus, mapping this to a destination will give easy confirmation of data flow without involving one of the two protocols. If data stops on the destination end, then the RTA is offline.



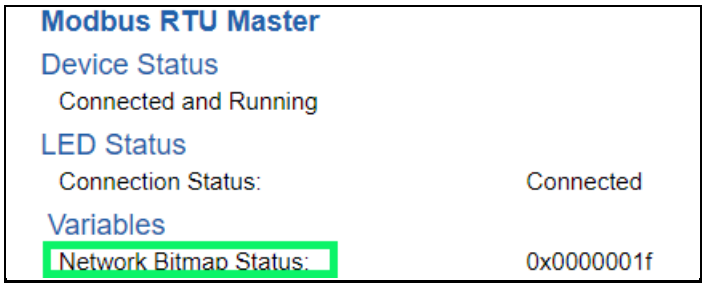
4) **Heartbeat 100ms Update**
   a) The Heartbeat 100ms Update variable can be used as a heartbeat that updates once every 100ms. The variable starts at 0 on gateway startup and increments by 1 every 100ms. This can be mapped into a destination on one of the available protocols to monitor the gateways connection status. If the value stops updating every 100ms the gateway is offline.




5) **Heartbeat 1000ms Update**
   a) The Heartbeat 1000ms Update variable can be used as a heartbeat that updates once every 1000ms. The variable starts at 0 on gateway startup and increments by 1 every 1000ms. This can be mapped into a destination on one of the available protocols to monitor the gateways connection status. If the value stops updating every 1000ms the gateway is offline.



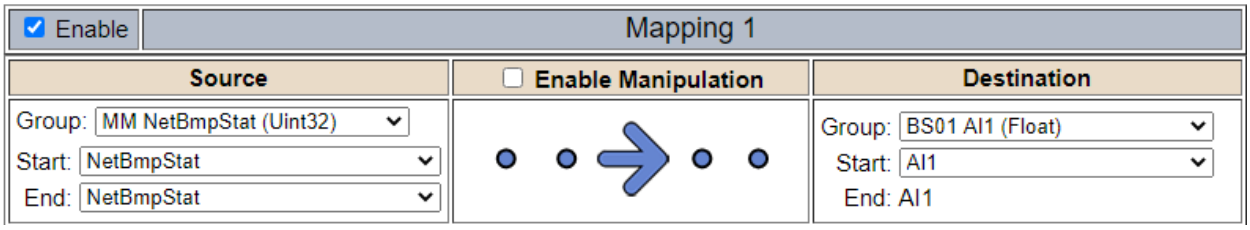

6) **XY_NetBmpStat**
   a) If a protocol is a Client/Master, there is a Network Bitmap Status that is provided on the Diagnostics Info page under the Variables section.



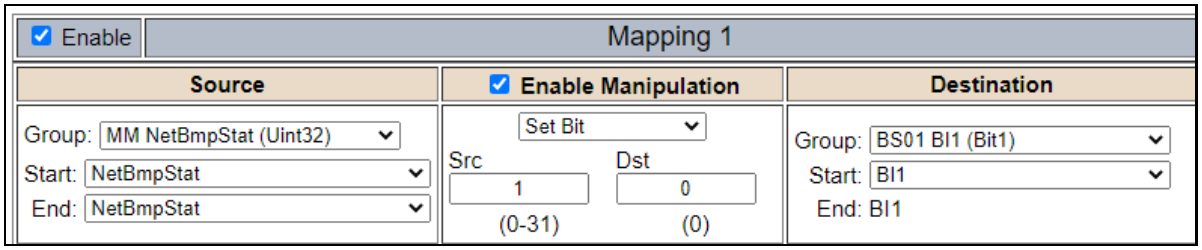

   b) Since a Client/Master may be trying to communicate with multiple devices on the network, it may be beneficial to know if a Server/Slave device is down. By using this Network Bitmap Status, you can expose the connection statuses of individual devices. **Values shown are in HEX.**
      i) 0x00000002 shows that only device 2 is connected
      ii) 0x00000003 shows that only devices 1 and 2 are connected
      iii) 0x0000001f shows that all 5 devices are connected (shown in image above)

c) There are multiple ways to map the NetBmpStat.

**Option 1:** Map the whole 32bit value to a destination. Example below shows the NetBmpStat is going to an Analog BACnet object. Using a connection of 5 Modbus Slave devices AI1 will show a value of 31.0000. Open a calculator with programmer mode and type in 31, this will represent bits 0 – 4 are on. This mean all 5 devices are connected and running.
If using an AB PLC with a Tag defined as a Dint, then expand the tag within your RSlogix software to expose the bit level and define each bit as a description such as device1, device2, etc.




**Option 2:** You can extract individual bits from the NetBmpStat by using the Set Bit Manipulation and map those to a destination. You'll need a mapping for each device you want to monitor. Example below shows Modbus device 2 (out of 5) is being monitor to a BACnet Binary Object. You can define the object in the BACnet Name configuration.



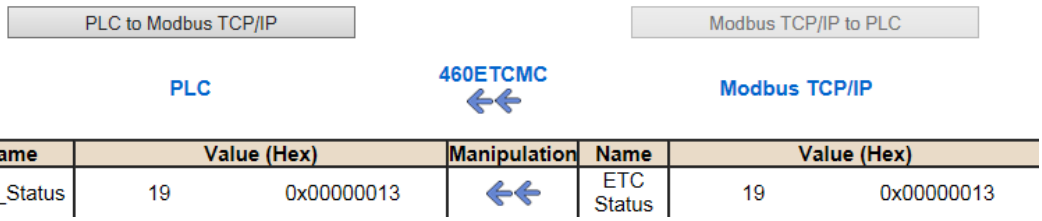
7) **Status_XY**

a) There are two Statuses provided, one for each protocol. This gives access to the overall status of that Protocol. Each Bit has its own meaning as follows:

***Common Status: 0x000000FF (bit 0-7) 1st byte***

| Hex: | Bit Position: | Decimal: | Explanation: |
|---|---|---|---|
| 0x00 | 0 | 0 | if we are a Slave/Server |
| 0x01 | 0 | 1 | if we are a Master/Client |
| 0x02 | 1 | 2 | connected (0 not connected) |
| 0x04 | 2 | 4 | first time scan |
| 0x08 | 3 | 8 | idle (usually added to connected) |
| 0x10 | 4 | 16 | running (usually added to connected) |
| 0x20 | 5 | 32 | bit not used |
| 0x40 | 6 | 64 | recoverable fault |
| 0x80 | 7 | 128 | nonrecoverable fault |

For this example, the ETC Status is mapped to a PLC tag called PLC_Status




| Name | Value (Hex) | | Manipulation | Name | Value (Hex) | |
|---|---|---|---|---|---|---|
| PLC_Status | 19 | 0x00000013 | ← | ETC Status | 19 | 0x00000013 |

**Example:** ETC Status is 0x00000013 (19 decimal), here is the break down

| Hex | Bit | Decimal | Explanation |
|---|---|---|---|
| 0x01 | 0(on) | 1 | if we are a Master/Client |
| 0x02 | 1(on) | 2 | connected (0 not connected) |
| 0x10 | 4(on) | 16 | running (usually added to connected) |
| Total: | 0x13 | 19 | |

***External Faults: 0x0000FF00 (bit 8-15) 2nd byte***

| Hex: | Bit Position: | Decimal: | Explanation: |
|---|---|---|---|
| 0x00 | 8 | 0 | local control |
| 0x01 | 8 | 256 | remotely idle |
| 0x02 | 9 | 512 | remotely faulted |
| 0x04 | 10 | 1,024 | idle due to dependency |
| 0x08 | 11 | 2,048 | faulted due to dependency |

***Recoverable Faults: 0x00FF0000 (bit 16-23) 3rd byte***

| Hex: | Bit Position: | Decimal: | Explanation: |
|---|---|---|---|
| 0x01 | 16 | 65,536 | recoverable fault - timed out |
| 0x02 | 17 | 131,072 | recoverable fault - Slave err |

***Non-Recoverable Faults  0xFF000000 (bit 24-31) 4th byte***

| Hex: | Bit Position: | Decimal: | Explanation: |
|---|---|---|---|
| 0x01 | 24 | 16,777,216 | nonrecoverable fault – task fatal err |
| 0x02 | 25 | 33,554,432 | nonrecoverable fault – config missing |
| 0x04 | 26 | 67,108,864 | nonrecoverable fault – bad hardware port |
| 0x08 | 27 | 134,217,728 | nonrecoverable fault – config err |
| 0x10 | 28 | 268,435,456 | Configuration Mode |
| 0x20 | 29 | 536,870,912 | No Ethernet Cable Plugged In |

For this example, the MC Status is mapped to a PLC tag called MC_Status

PLC to Modbus TCP/IP  Modbus TCP/IP to PLC

PLC  460ETCMC  Modbus TCP/IP

| Name | Value (Hex) | | Manipulation | Name | Value (Hex) | |
|---|---|---|---|---|---|---|
| MC_Status | 65601 | 0x00010041 | ← | MC Status | 65601 | 0x00010041 |

**Example:** MC Status is 0x00010041 (65601 decimal), here is the break down, we know that bytes 1 and 3 are being used, so here is the break down,

***Common Status:***

| Hex: | Bit: | Decimal: | Explanation: |
|---|---|---|---|
| 0x01 | 0(on) | 1 | if we are a Master/Client |
| 0x40 | 6(on) | 64 | recoverable fault |

***Recoverable Faults:***

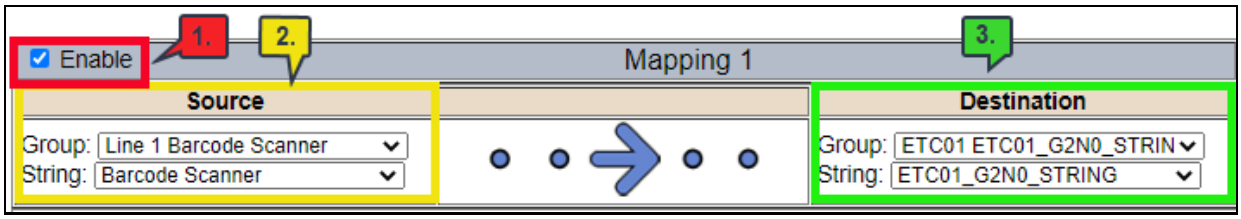
| Hex: | Bit: | Decimal: | Explanation: |
|---|---|---|---|
| 0x01 | 16 | 65,536 | recoverable fault – timed |

| Total: | 0x010041 | 65,601 |
|---|---|---|

# String Mapping – Explanation

Below are the different parts that can be modified to make up a string mapping.

String data types can only be mapped to other string data types. There is no manipulation that can be done on the string.

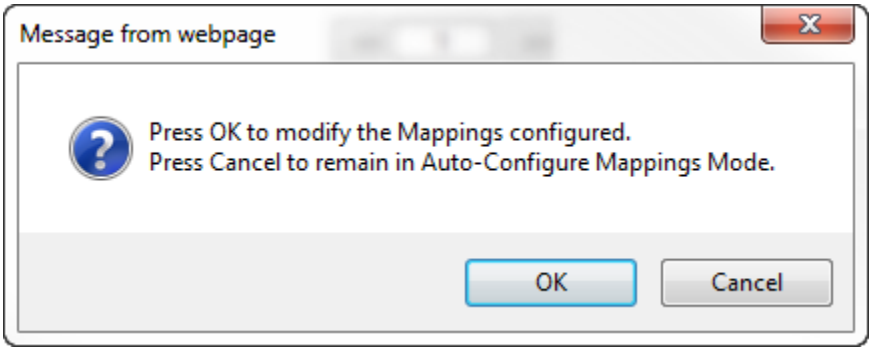



1) Enable (red box above): Check to enable mapping. If not checked, this mapping is skipped.
2) Source Field (yellow box above):
   a) Group - Select the string data group you set up in the protocol config to use for this mapping.
   b) String - This is the string used for this mapping.
3) Destination Field (green box above):
   a) Group - Select the string data group you set up in the protocol config to use for this mapping.
   b) String - This is the string where the data is being stored.

# Mapping – Auto-Configure Mode to Manual Configure Mode

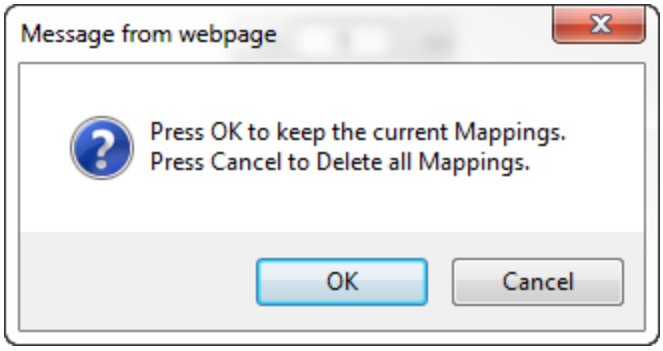
To transition from Auto-Configure Mapping Mode to Manual Configure Mode, click the dropdown at the top of the Mapping Configuration page and select Manual Configure.

After you click this button, you will be prompted to confirm if this is really what you want to do.




Click **OK** to proceed to Manual Configure Mode or click **Cancel** to remain in Auto-Configure Mappings Mode.

Once OK is clicked, there are 2 options on how to proceed from here.

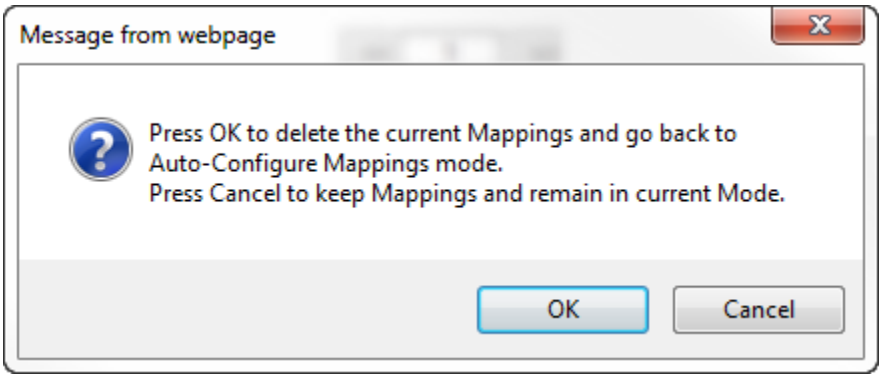



1) To keep the mappings that are already configured press **OK**.
    a) You would want this option if you are adding additional mappings or you want to modify the mapping(s) that already exist.
2) To delete the mappings that are already there and start over press **Cancel**.

To modify the number of mappings, enter a number in the text field next to **# of Mappings to Configure** and click the **Set Max # of Mappings** button. You can always add more mappings if needed.

# Mapping – Manual Configure Mode to Auto-Configure Mode

To transition from Manual Configure Mode to Auto-Configure Mapping Mode, click the dropdown menu at the top of the Mapping Configuration page and select Auto-Configure Mappings.




Click **OK** to proceed to delete all current mappings and go back to Auto-Configure Mappings Mode. Click **Cancel** to keep all mappings and remain in Manual Configure Mode.

**NOTE**: Once you revert to Auto-Configure Mapping Mode there is no way to recover the mappings you lost. Any mappings you previously have added will be deleted as well.

# View as Text

## Data Mapping

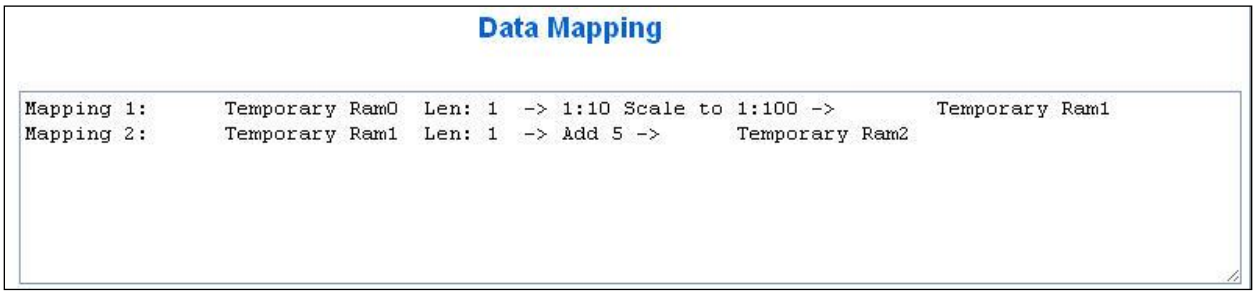
The View as Text page displays the point to point mapping(s) you set up in the Data Mapping section. This will also display any manipulation(s) that are configured.

Each line on this page will read as follows:

> **Mapping** *number*: *source point* **Len:** *Number of points mapped* **->** *manipulation (if blank then no manipulation)* **->** *destination point*

If you are looking for a specific point to see if it is mapped, you can do a find in this text box for your point in question. Example: you defined 20 Registers starting at register 1 and want to see if 400011 is mapped. If it is not in this text box, then it is not mapped, and no data will be transferred.

This is the text display for the example shown under the *Data Mapping- Adding Diagnostic Information* section.



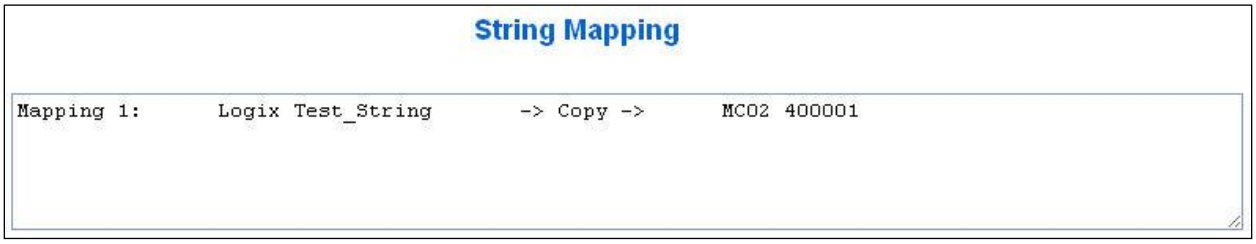

## String Mapping

The View as Text page displays the string mapping(s) you set up in the String Mapping section.

Each line on this page will read as follows:

> **Mapping** *number*: *source point* **-> Copy ->** *destination point*

If you are looking for a specific point to see if it is mapped, you can do a find in this text box for your point in question. Example: you defined 20 String Tags in the PLC and want to see if “Test_String” in the Logix PLC is mapped. If it is not in this text box, then it is not mapped, and no data will be transferred.




## Base Triggering – Data Validiation Triggering

With Base Triggering, you will be marking data as "Invalid" and force RTA Master/Controller/Client protocols to read all the read data points sources until ALL source protocols data is valid. You will be able to utilize the Handshake to map over to Technology Trigger and/or back over to your source protocol for reference.

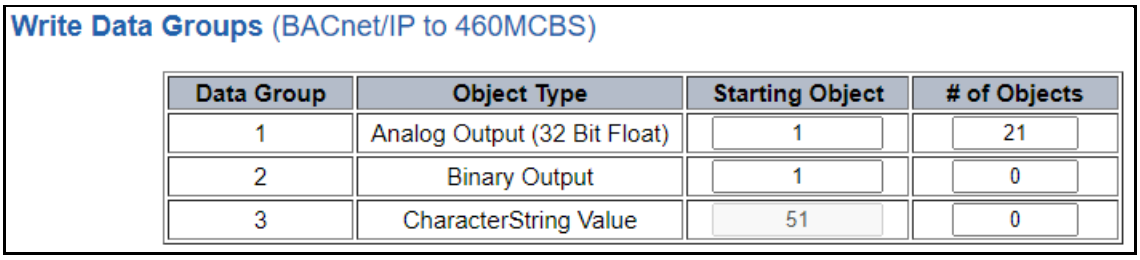
## How does this work?

1) Map the Triggering Variable (Source) over to Trigger **#** (Dest).
2) If Trigger **#** value changes states mark all Trigger **#** protocols read data as "Invalid".
3) Read all source read data points until ALL source read data is valid.
4) Handshake **#** value is set equal to Trigger **#** value.
5) Map Handshake **#** to reference data point.
   **Note**: **#** is an internal reference to the Server/Slave number you are settings up. **ex**. RTA Server/Slave products can only be Trigger 1 and Handshake 1 since we are only 1 device. If RTA is a Master/Client, then you can have a Trigger# for each server/slave connected too.

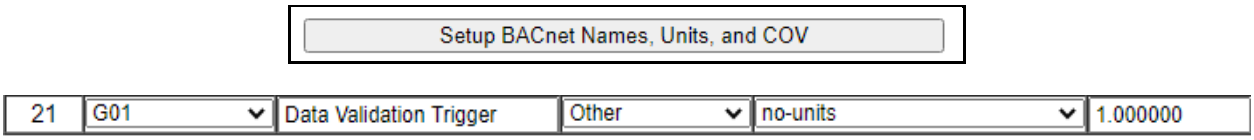
## How do you set this up?

In this example I'm using a 460MCBS. My Building Automation System wants to verify that all data read from Modbus TCP/IP Server is valid.

1) Add an extra Analog Output for your Trigger. This tells the RTA to mark all data invalid.




a) You can define AI21 as your validation name in the Setup BACnet Names Configuration.

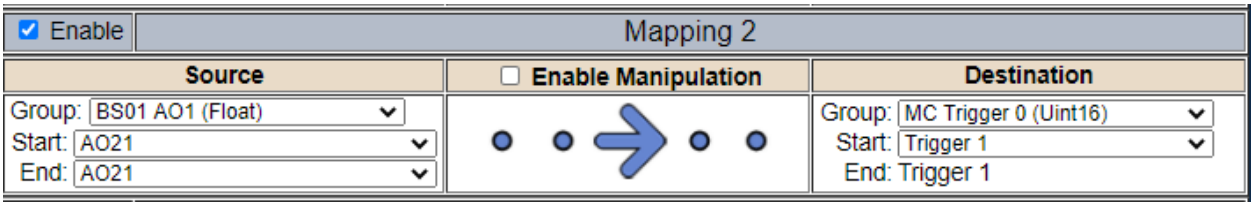



2) Add another Analog Input as reference for when data has been validated. When you write from AO21 to validate data, the RTA will reply to AI40 saying "validation complete".

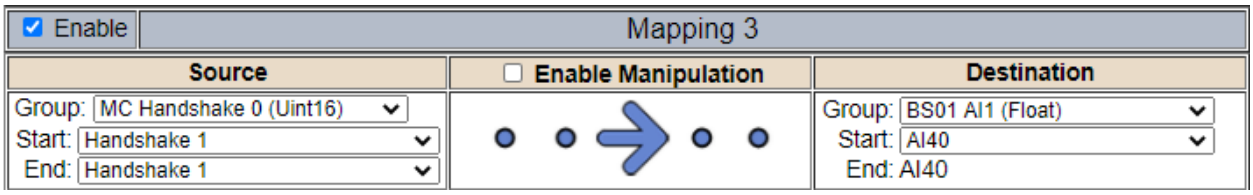
| Data Group | Object Type | Starting Object | # of Objects |
|---|---|---|---|
| 1 | Analog Input (32 Bit Float) | 1 | 40 |
| 2 | Binary Input | 1 | 0 |
| 3 | CharacterString Value | 1 | 0 |

| | | | | | |
|---|---|---|---|---|---|
| 40 | G01 | Data Validation Result | Other | no-units | 1.000000 |

3) Within the Data Mapping page manually add 2 additional mappings.
4) The first mapping is going to be the Data Validation Triggering. AO21 will write to the RTA, MC Trigger 1 will mark data invalid.




5) The second mapping, the MC Handshake will increment that all data is validated and write to AI21 “all data is validated”. The value of AI40 and AO21 should be the same.



# Security Configuration

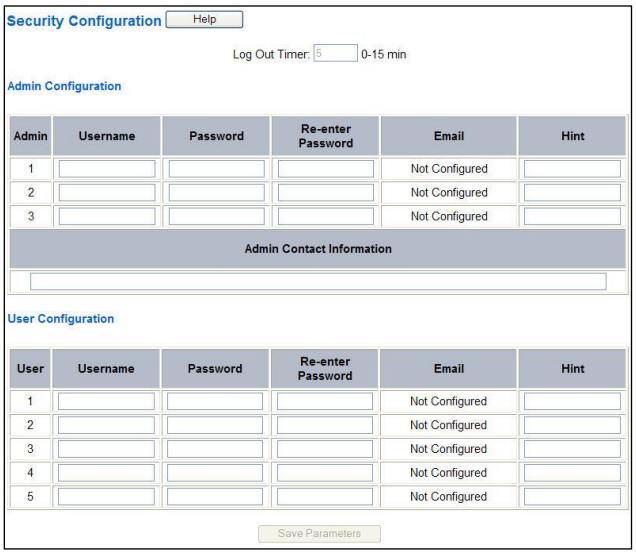
To setup security on the 460 gateway, navigate to **Other->Security Configuration**. You can configure Security for 3 administrators, 5 users, and 1 guest.

## THIS IS NOT A TOTAL SECURITY FEATURE

The security feature offers a way to password protect access to diagnostics and configuration on the network. The security feature does not protect against "Air Gap" threats. If the gateway can be physically accessed, security can be reset. All security can be disabled if physical contact can be made. From the login page, click the Reset Password button twice. You will be forced to do a hard reboot (power down) on the gateway within 15 minutes of clicking the button. This process should be used in the

**Note:** Only Admins have configuration access to all web pages.

1) Log Out Timer: The system will automatically log inactive users off after this period of time. **NOTE**: A time of 0 means that the user will not be automatically logged off. Instead, they must manually click the **Logout** button.
2) Username: Enter a username, max of 32 characters.
3) Password: Enter a password for the username, max of 32 characters, case sensitive.
    a. Re-enter the Password
4) E-mail: In case the password was forgotten, a user can have their password e-mailed to them if e-mail was configured.
5) Hint: A helpful reminder of what the password is.



## Security Configuration-Security Levels

Each webpage in the gateway can have a separate security level associated with it for each user.

Security Levels:

1) **Full Access**: Capability to view and configure a web page.
2) **View Access**: Capability to view a web page, but cannot configure parameters.
3) **No Access**: No capability of viewing the web page and page will be removed from Navigation.

## Security - Log In

**Username**: Name of the user to login.

**Password**: Password of the user to login.

**Log In:** If login is successful, the user will be redirected to the Main Page.

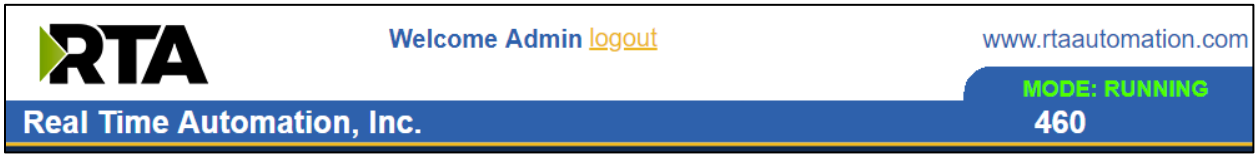
**Send Password to Email:** Sends the specified User's Password to the email configured for that user.

**Display Hint:** Displays the hint specified for the User if one was set up.

**Reset Password:** This is used to reset security settings. Confirm reset password must be selected to confirm this action. Once confirmed, there is a 15 minute window to do a hard reset of the gateway by physically removing and restoring power from the gateway. Once power is restored, you may navigate to the IP address of the gateway as normal.

## Security - Log Out

Once a user is done with a session they may click **logout** at the top of any page. The user may also be logged out for inactivity based off of the Log Out Timer specified during the configuration.




**Closing the browser is not sufficient to log out.**

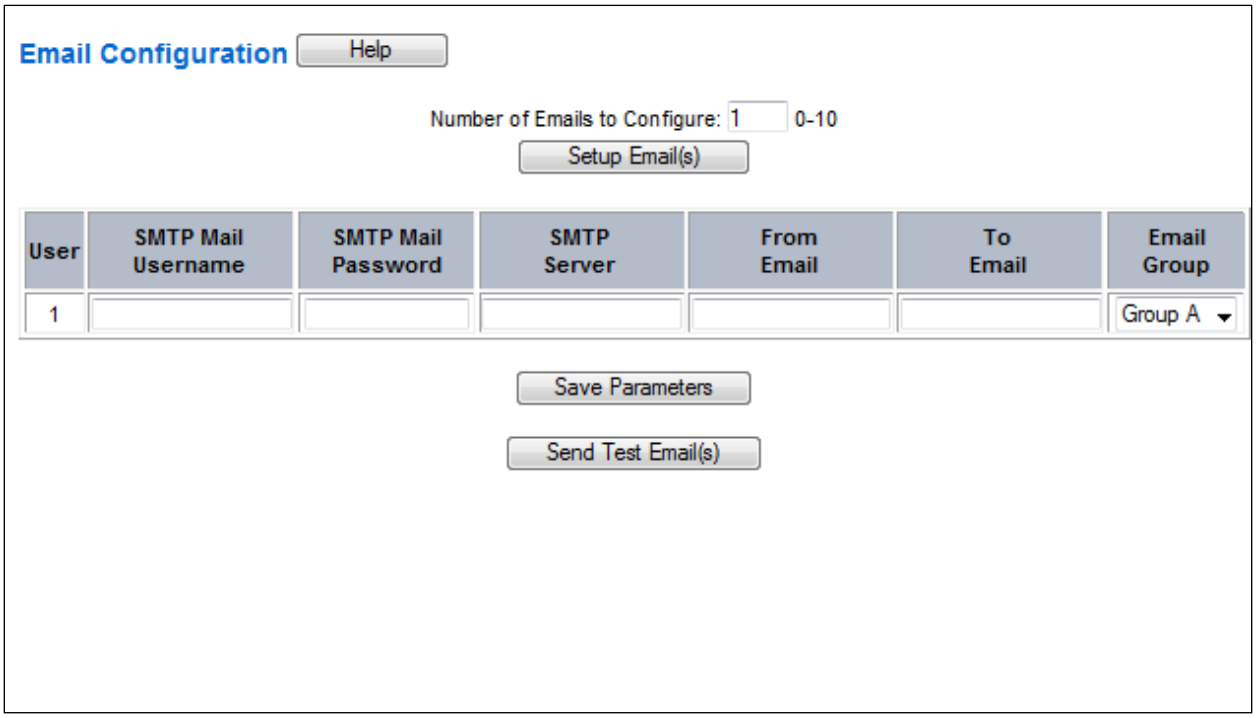
# Email Configuration

To setup e-mails on the 460 gateway, navigate to **Other->Email Configuration**.

You can configure up to 10 email addresses.

1) SMTP Mail Username: The email address that the SMTP server has set up to use.
2) SMTP Mail Password: If authentication is required, enter the SMTP Server's password (Optional).
3) SMTP Server: Enter the Name of the SMTP Server or the IP Address of the Server.
4) From E-mail: Enter the e-mail that will show up as the sender.
5) To E-mail: Enter the e-mail that is to receive the e-mail.
6) E-mail Group: Choose a group for the user. This is used in other web pages.

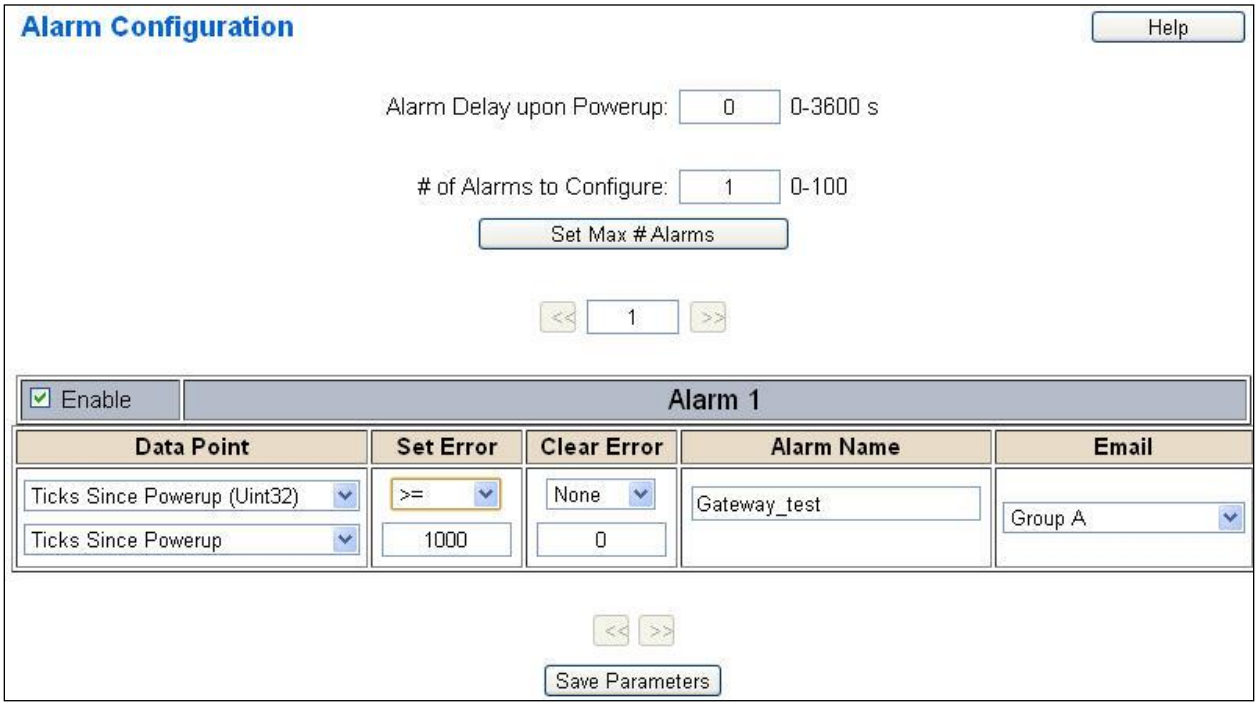
Click the **Save Parameters** button to commit the changes and reboot the gateway.



# Alarm Configuration

To setup alarms on the 460 gateway, navigate to **Other->Alarm Configuration**.

1) Alarm Delay upon Powerup: At Powerup, the gateway will have values of '0' stored for all data. This may cause alarms to trigger before these values are updated by the mating protocols. Set this field to provide needed time to update fields before considering values for alarms.




2) Enter the number of alarms to configure and click **Set Max # Alarms** to generate those lines.
3) In the Data Point Section:
    a. Top dropdown: select the Data Group. This dropdown menu will contain all groups that go from the gateway to the network.
    b. Lower dropdown: select the Data Point's Specific Point. This is used to select which point in the group will be monitored for alarms.
4) In the Set Error Section:
    a. Select the Set Error Operation in the top dropdown menu. Available options are <, >, <=, >=, !=, ==, and Change of State (COS). This is the operation that will be used to compare the Data Point value against the Error Value to determine if the alarm needs to be set.
    b. Select the Set Error Value. This value is used as: 'Data Point's Value' 'Operation' 'Value.' Ex: Ticks Since Powerup >= 1000. This will set the alarm after 1000 ticks have elapsed since the unit powered up.

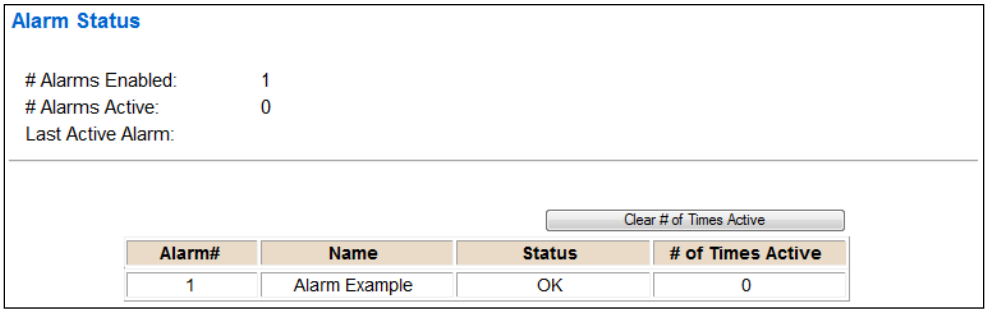
5) In the Clear Error Section:
    a. Select the Clear Error Operation. Available options are <, >, <=, >=, !=, ==, and Change of State (COS). This is the operation that will be used to compare the Data Point value against the Error Value to determine if the alarm needs to be cleared.
    b. Select the Clear Error Value.
    -Ex: Ticks Since Powerup >= 5000. This will clear the alarm after 5000 ticks have elapsed since the unit powered up.
6) Enter an Alarm Name. This will make the alarm unique and will be available in the Alarm Status page as well as in the email generated by the alarm.
7) Select an email to associate this alarm with. When an alarm is set, it sends an email. When an alarm is cleared, it will also send an email.

Click the **Save Parameters** button to commit the changes to memory and reboot the gateway.

## Diagnostics – Alarm Status

Alarm Status will only display under the Diagnostic menu tab if at least 1 Alarm is enabled.

1) # Alarms Enabled: This is a count of enabled alarms.
2) # Alarms Active: This is how many alarms are presently active (set).
3) Last Active Alarm: This is the last alarm that the gateway detected.
4) **Clear # of Times Active**: This will reset all alarms '# of Times Active' to 0.
5) Alarm #: The reference number to the given alarm on the alarm setup page.
6) Name: The name of the alarm.
7) Status: The current status of the alarm, either OK or ALARM.
8) # of Times Active: This count represents the number of times this alarm has become active. If an alarm is triggered, this count will increment.

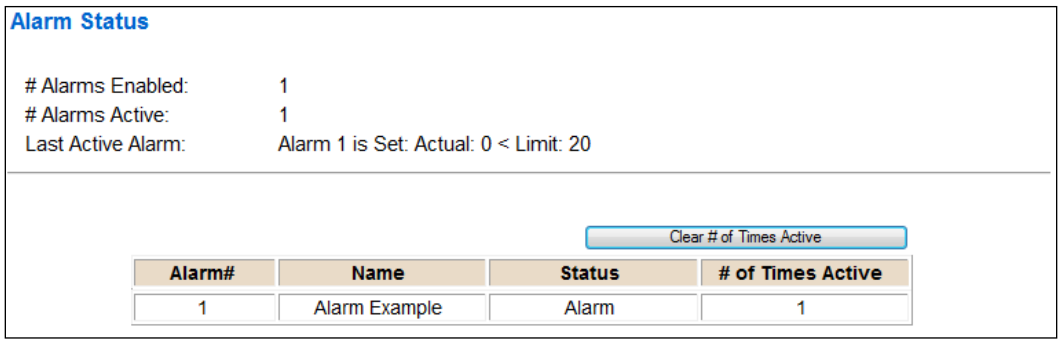


## Alarms – Active

While one or more alarms are active, every page will display 'Alarms Active' at the top of the page. This will no longer be displayed if all active alarms have been cleared.

When an alarm is activated, the following will occur:

1) A one-time notification will be sent out to the email associated with the alarm.
2) For duplicate emails to occur, the alarm must be cleared and then become active again.
3) *# Alarms Active* and *# of Times Active* will be incremented.
4) Status of the Individual Alarm will be set to *Alarm*.
5) *Last Active Alarm* field will be populated with details on what triggered the alarm.




## Alarms – Clear

When an alarm is cleared, the following will occur:

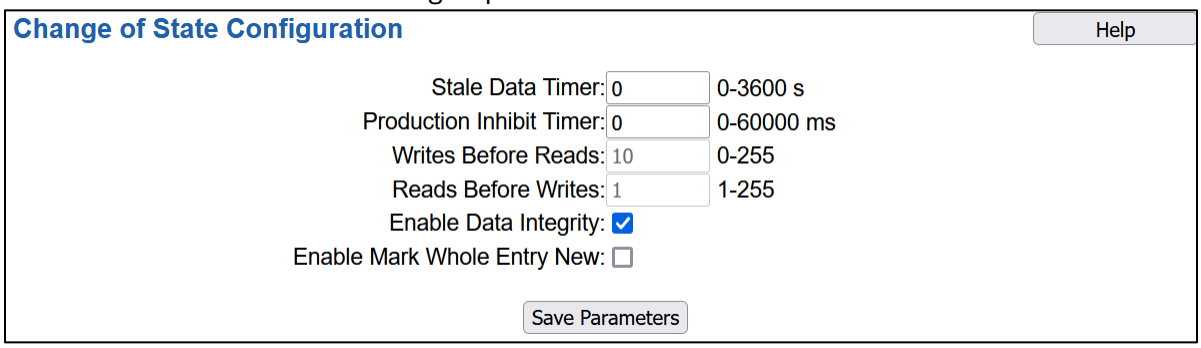
1) A one-time notification will be sent to the email associated with the alarm.
    a. For duplicate emails to occur, the alarm must become active and then be cleared again.
2) Total *# Alarms Active* will decrement. *Last Active Alarm* will not be changed.
3) Status of the Individual Alarm will be reset to *OK*.

# Change of State (COS) Configuration

To access the configuration files in the 460 gateway, navigate to dropdown **Other->COS Configuration**. The gateway, by default only writes when data has changed. The gateway also waits to write any data to the destination until the source protocol is successfully connected.

**Default values should fit most applications. Change these values with caution as they affect**

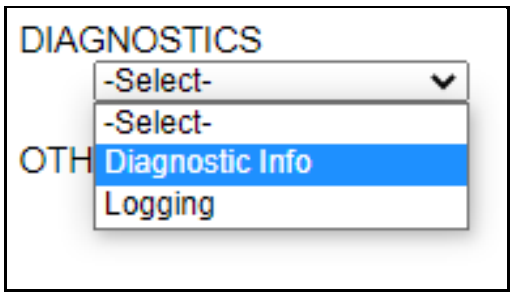
1) **Stale Data Timer**: If the data has not changed within the time allocated in this Stale Data Timer, the data will be marked as stale within the gateway and will force a write request to occur. This timer is to be used to force cyclic updates in the gateway, since data will only be written if it has changed by default. There is a separate timer per data mapping.
   **Gateway behavior**:
   - If time = 0s => (DEFAULT) The gateway will write out new values on a Change of State basis.
   - If time > 0s => The gateway will write out new values whenever the timer expires to force cyclic updates (write every *x* seconds).
2) **Production Inhibit Timer:** Amount of time after a Change of State write request has occurred before allowing a new Change of State to be written. This is to be used to prevent jitter. Default value is 0ms. This timer takes priority over the Stale Data Timer. There is a separate timer per data mapping. This timer is active only after the first write goes out and the first COS event occurs.
3) **Writes Before Reads**: If multiple writes are queued, execute # of Writes Before Reads before the next read occurs. Default is 10 and should fit most applications.
   **Warning:** A value of 0 here may starve reads if a lot of writes are queued. This may be useful in applications where a burst of writes may occur and you want to guarantee they all go out before the next set of reads begin.
4) **Reads Before Writes:** If multiple writes are queued, the # of Writes Before Reads will occur before starting the # of Reads Before Writes. Once the # of Reads Before Writes has occurred, the counter for both reads and write will be reset. Default is 1 and should fit most applications.
5) **Enable Data Integrity**: If enabled, do not execute any write requests to the destination until the source data point is connected and communicating. This prevents writes of 0 upon power up.
6) **Enable Mark Whole Entry New**: If Enabled, mark the entire scan line or data group new upon 1 data element within the scan line or data group to be new.




Click the **Save Parameters** button to commit the changes to memory and reboot the gateway.

# Diagnostics Info

The Diagnostics page is where you can view both protocols' diagnostics information, # of Data Mappings, # of String Mapping and # Alarm Mappings.




For protocol specific diagnostic information, refer to the next few pages.

## Diagnostics Mapping

This section displays the number of mappings that are enabled, Data Mapping and String Mapping will show the # of Errors and First Errors. Alarms will show # active and Last Alarm that was active.

**Common Errors:**

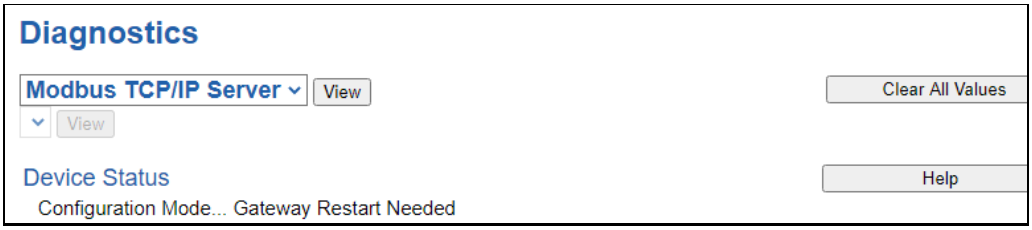
1) Destination or Source Point does not exist
   a) Solution: Re-map the mapping
2) Source or Destination Pointer too small
   a) There is not enough space on either the Source, or the Destination for the data you want to copy. This is typically seen when the Destination is smaller than the amount of data being transferred to it.
3) Range Discard, Min or Max Value
   a) The actual data value is outside of the defined range
4) Math Error
   a) Operation value cannot be 0
5) Scaling Error
   a) Source Min must be smaller than Source Max
   b) Destination Min must be smaller than Destination Max

| | |
|---|---|
| **Data Mapping** | |
| # Enabled: | 5 of 5 |
| # of Errors: | 0 |
| First Error: | |
| **String Mapping** | |
| # Enabled: | 2 of 2 |
| # of Errors: | 0 |
| First Error: | |
| **Alarms** | |
| # Enabled: | 3 |
| # Active: | 0 |
| Last Active: | |

**Note:** you can also view this information on the Main Page.

# Diagnostics – Modbus TCP/IP Server

Select the **Modbus TCP/IP Server** in the dropdown menu on the Diagnostics Page to view a breakdown of the diagnostics and common strings that are displayed on the page. Additional diagnostic information can be found by clicking the **Help** button.

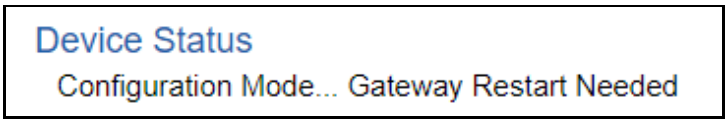



**NOTE**: This page will auto-refresh every five seconds with the latest data.

**Clear All Values** - This will only affect displayed values.

1) This will reset all displayed values back to zero.
2) If viewing Modbus TCP/IP Server, this will only clear the values for the Modbus TCP/IP Server section of the gateway.

**Device Status**




1) Connected - A Modbus TCP/IP client has a connection for the gateway.
2) Not Connected:
   a) Ethernet Cable not plugged in.
   b) The Modbus TCP/IP client has not initiated communication to the gateway.
   **c)** The Modbus TCP/IP client has not communicated to the gateway in "x" milliseconds, where "x" is the inactivity timeout specified in the Modbus TCP/IP Server Configuration.

**LED Status:**

1) Solid Green (Connected and Running) – The gateway is connected to a Modbus TCP/IP client and communicating as expected.
2) Flashing Green (Connection not yet attempted) – The Modbus TCP/IP client has never attempted to connect to the gateway.
3) Flashing Red (Nodes Missing Timeout) - The gateway has lost a connection to the Modbus TCP/IP client.
4) Off:
   a) No power.
   b) No Ethernet cable plugged in.

**Variables:**

| Variables | |
|---|---|
| FC01 Read Coil Status: | 0 |
| FC02 Read Input Status: | 0 |
| FC03 Read Holding Registers: | 0 |
| FC04 Read Input Registers: | 0 |
| FC05 Force Single Coil: | 0 |
| FC06 Preset Single Register: | 0 |
| FC15 Force Multiple Coils: | 0 |
| FC16 Preset Multiple Registers: | 0 |
| FC23 Read/Write 4X Registers: | 0 |
| Successful Responses Sent: | 0 |
| Error Responses Sent: | 0 |
| Timeout, TCP Closed: | 0 |
| TCP Connections Open: | 0 |

1) FC01 Read Coil Status – Modbus Function Code 1: Number of Read Coil Status requests received.
2) FC02 Read Input Status – Modbus Function Code 2: Number of Read Input Status requests received.
3) FC03 Read Holding Registers – Modbus Function Code 3: Number of Read Holding Registers requests received.
4) FC04 Read Input Registers – Modbus Function Code 4: Number of Read Input Registers requests received.
5) FC05 Force Single Coil – Modbus Function Code 5: Number of Write Coil Status requests received.
6) FC06 Preset Single Register – Modbus Function Code 6: Number of Write Holding Register requests received.
7) FC15 Force Multiple Coils – Modbus Function Code 15: Number of Write Multiple Coil Status requests received.
8) FC16 Preset Multiple Registers – Modbus Function Code 16: Number of Write Multiple Holding Register requests received.
9) FC23 Read/Write 4X Register – Modbus Function Code 23: Number of Read/Write Holding Registers requests received.
10) Successful Responses Sent – Total number of Read/Write messages sent by the gateway.
11) Error Responses Sent - Total number of Read/Write errors sent by the gateway.
12) Timeouts TCP Closed – Total number of Read/Write timeouts that cause the Modbus TCP/IP connection to close.
13) TCP Connections Open – Number of Modbus TCP/IP connections that have been opened to the gateway.

# Diagnostics – DeviceNet Slave

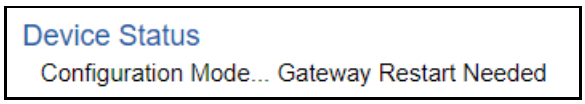
Select the **DeviceNet Slave** in the dropdown menu on the Diagnostics Page to view a breakdown of the diagnostics and common strings that are displayed on the page. Additional diagnostic information can be found by clicking the **Help** button.

**NOTE**: This page will auto-refresh every five seconds with the latest data.

**Clear All Values** - This will only affect displayed values.

1) This will reset all displayed values back to zero and clear the Status Strings.
2) If viewing DeviceNet Slave, this will only clear the values for the DeviceNet Slave section of the gateway.

**Device Status**

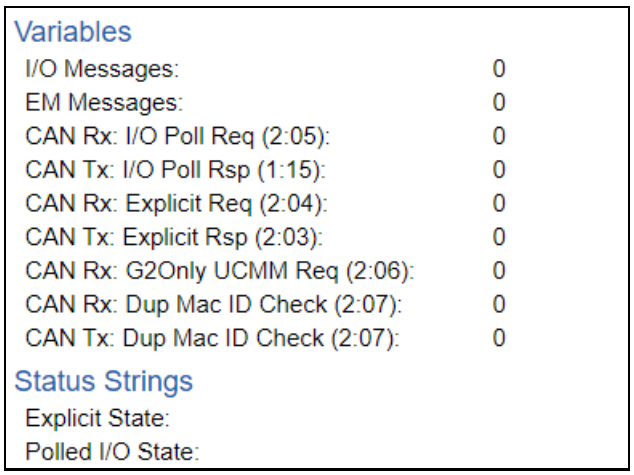



1) Connected and Running - A DeviceNet master has an I/O connection to the gateway and outputs are being updated.
2) Connected and Idle - A DeviceNet master has an I/O connection to the gateway, but the outputs are not being updated.
3) Connected – A DeviceNet master has an explicit connection to the gateway (no I/O).
4) Not Connected - The DeviceNet master has not connected to the gateway.
5) Error: Timeout - The gateway has lost a polled I/O connection to the DeviceNet master.
6) Fatal Error: Couldn't Open Hardware Port – The physical port selected on the DeviceNet Slave Configuration page is not configured.

### LED Status

1) Solid Green (Connected) - The gateway is connected to a DeviceNet master and communicating via Explicit or Polled I/O.
2) Flashing Green (Not Connected) - Not connected currently to a DeviceNet master, but Baud Rate and MAC ID are valid.
3) Flashing Red (I/O Timeout) - The gateway has lost a Polled I/O connection to the DeviceNet master.
4) Solid Red (Fatal Error)
   a) Duplicate MAC ID check
   b) BUS OFF
   c) Hardware Port - The physical port selected on the DeviceNet Slave Configuration page is not configured.
   d) Verify that the DeviceNet slave has an enabled port selected. If needed, configure port settings.
5) Off - Duplicate MAC ID check in progress (no other CAN devices at our baud rate on the bus).

### Variables




1) I/O Messages - Number of Polled I/O Messages received.
1) EM Messages - Number of Explicit Message received.
2) CAN Rx: I/O Poll Req (Group 2: Message ID 5) - Primary form of active communications. This number should increment by 1-58 for every I/O Output Message received, depending on the I/O Output Size (7 bytes per fragment).
3) CAN Tx: I/O Poll Rsp (Group 1: Message ID 15) - Primary form of active communications. This number should increment by 1-58 for every I/O Input Message sent, depending on the I/O Input Size (7 bytes per fragment).
4) CAN Rx: Explicit Req (Group 2: Message ID 4) - Typically only used during connection allocation and should be small (<10).
5) CAN Tx: Explicit Rsp (Group 2: Message ID 3) - Typically only used during connection allocation and should be small (<10).
6) CAN Rx: G2Only UCMM Req (Group 2: Message ID 6) - Typically only used during connection allocation and should be small (if no errors, should be 1).

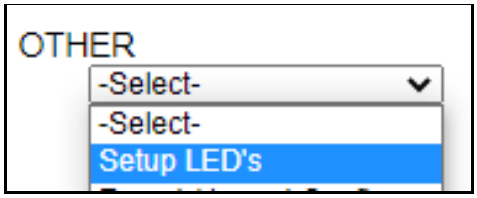
7) CAN Rx: Dup Mac ID Check (Group 2: Message ID 7) - Should be zero. If higher, then another node with the ID is on the network.
8) CAN Tx: Dup Mac ID Check (Group 2: Message ID 7) - Should be two normally. If higher, then another node with the ID is on the network.

**Status Strings**

1) Explicit State:
   a) Non-Existent – No connection
   b) Established – Connection is properly allocated and communicating
   c) Deferred Delete – Connection is timed out, but remains allocated until the polled I/O connection is removed
2) Polled I/O State:
   a) Non-Existent – No connection
   b) Configuring – Connection is allocated, but configuration is not complete
   c) Established – Connection is properly allocated and communicating
   d) Timed Out – Connection is timed out

# LED Configuration

To modify the behavior of the LEDs on the 460 gateway, navigate to **Other->Setup LEDs**.

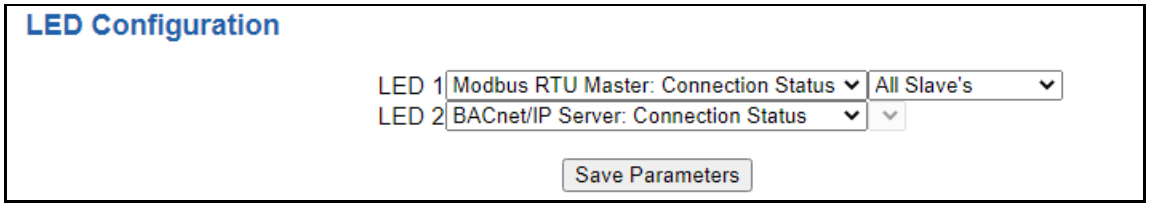



Each LED may be set to Disabled, Protocol 1, or Protocol 2. If either protocol is a master/client, you may set the LED to represent either all slaves/servers configured in the gateway or a slave/server device.

To select a slave/server device:

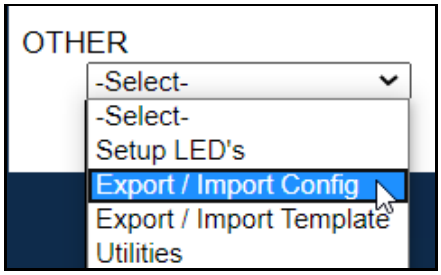
1) Select the protocol in the left dropdown menu.
2) Click **Save Parameters** to generate the second dropdown menu.
3) Select the individual slave/server in the right dropdown menu.

Click the **Save Parameters** button to commit the changes and reboot the gateway.



# Configuration Files

To access the configuration file in the 460 gateway, select the dropdown **Other->Export/Import Config**.




## Export Configuration

The Export Configuration allows you to save your configuration file for backup or to be imported into another gateway. This file is named *rta_cfg.rtax* by default.

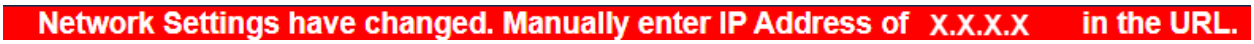
Upon clicking the **Save Configuration to File** button, you will be prompted to select a location to save the file. Different web browsers will yield different looks.

## Import Configuration

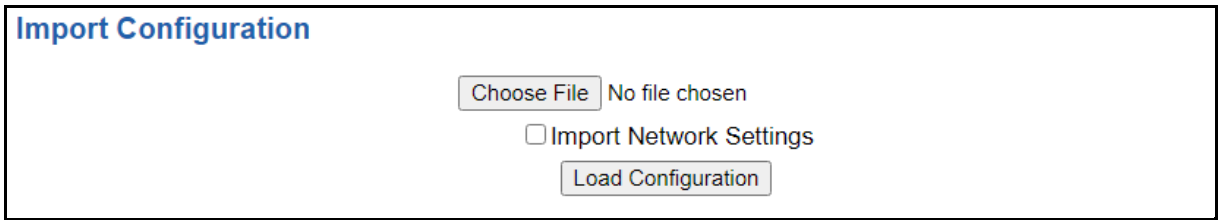
You can import a previously exported configuration file or a configuration file from another device into the 460 gateway, whenever it is in Configuration Mode.

Upon clicking the **Choose File** button, you will be prompted to select a location from which to load the saved file. Once the location is selected, you can choose the **Import Network Settings** checkbox if you want to load the network settings of the configuration file or just load the configuration without the network setting.

If you choose to Import Network Settings, this will override your current gateway's network setting with the settings in the configuration file. After you click on the Load Configuration button, a banner will display your gateway's new IP address.



If the configuration has successfully loaded, the gateway will indicate that it was successful, and a message will appear under the Load Configuration button indicating Restart Needed.




If it encountered an error while trying to load the saved configuration, the gateway will indicate the first error it found and a brief description about it under the Load Configuration button. Contact RTA Support with a screenshot of this error to further troubleshoot.

# Save and Replace Configuration Using SD Card

## Saving Configuration Using SD Card

This function saves the gateway's configuration automatically to an SD Card each time the gateway is rebooted via the **Restart Now** button on the web page. If this unit should fail in the future, the last configuration stored on the SD card and can be used for a new gateway to get the application back up and running quickly.

This SD Card replaces every configurable field in the gateway, **EXCEPT** for IP Address, Subnet Mask, and Default Gateway.

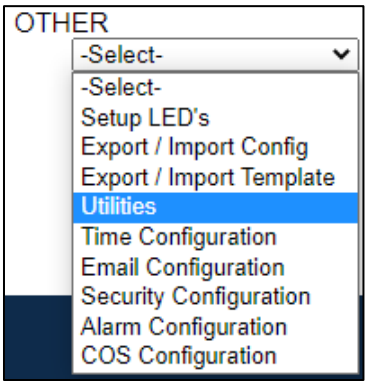
## Replacing Configuration Using SD Card

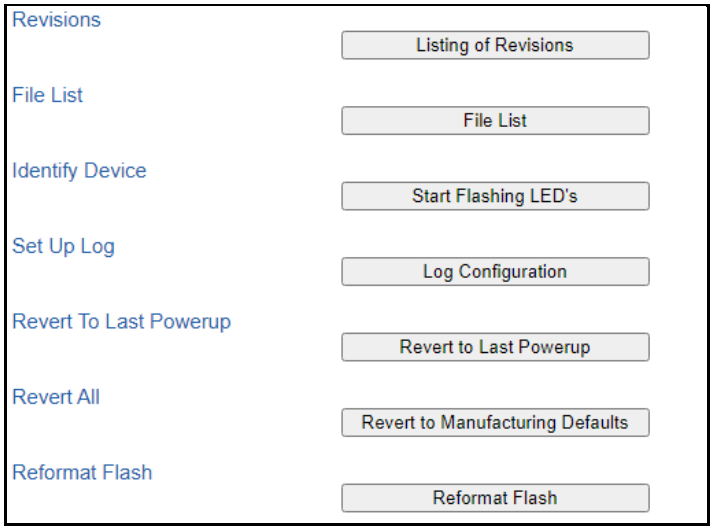
To replace a configuration in a gateway using the SD Card, a specific sequence of events must be followed for the replacement to happen correctly:

1) Extract SD Card from gateway you wish to copy the configuration from.
2) Power up the gateway you wish to copy the configuration to. DO NOT INSERT SD CARD YET.
3) Navigate to the webpage inside the unit.
4) Navigate to the dropdown **Other->Utilities**.
5) If you are not currently in *Mode: Configuration*, go into Configuration Mode by clicking the **Configuration Mode** button at the top left-hand side of the screen.
6) Press the **Revert to Manufacturing Defaults** button on the Utilities Page. The Configuration will ONLY be replaced by the SD Card if the gateway does not have a configuration already in it.
7) When the unit comes back in *Mode: Running*, insert the SD Card.
8) Do a hard power cycle to the unit by unplugging power. DO NOT RESET POWER VIA WEB PAGES.
    a. It will take an additional 30 seconds for the unit to power up while it is transferring the configuration. During this time, the gateway cannot be accessed via the web page.
9) When the unit comes back up, the configuration should be exactly what was on the SD Card.

# Utilities

To access the Utilities page in the 460 gateway, navigate to **Other->Utilities**. The Utilities screen displays information about the gateway including Operation Time, File System Usage, Memory Usage, and Memory Block Usage.




Here you can also:

- View the full revision of the software.
- View all the files stored in the Flash File System within the gateway.
- Identify your device by clicking the **Start Flashing LEDs** button. By clicking this button, the two diagnostic LEDs will flash red and green. Once you have identified which device you are working with, click the button again to put the LEDs back into running mode.
- Configure the size of the log through the Log Configuration.
- Bring the device back to its last power up settings.
- Bring the device back to its original manufacturing defaults.
- Remove the Configuration File and Flash Files within the gateway.