

# ***460TCPAZR-NNA1*** ***Protocol Gateway***

## **Product User Guide**

*Firmware Version 8.8.37*

## 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.

© 2024 Real Time Automation, Inc. All rights reserved.

Revision History .....	6
Overview .....	8
Hardware Platforms .....	9
Hardware – NNA1 .....	10
Powering the Gateway.....	10
Mounting with a DIN Rail.....	11
Installing .....	11
Removing.....	11
Accessing the Main Page .....	12
Committing Changes to the Settings.....	14
Main Page.....	15
Device Configuration .....	16
Network Configuration .....	17
TCP/IP Configuration .....	18
ASCII Configuration.....	19
Receive Data .....	20
Receive Data – Operation Mode .....	21
Transmit Data .....	22
Transmit Data – Triggering Methods.....	23
ASCII Configuration – Technology Triggering Method.....	25
ASCII Configuration – ASCII Parsing.....	27
ASCII Configuration – ASCII Parsing Examples .....	28
ASCII Configuration – ASCII Concatenating.....	30
ASCII Configuration – ASCII Concatenating Examples.....	32
ASCII Configuration – ASCII Message Counter .....	34
MQTT Client Configuration .....	35
Microsoft Azure (AZR) Configuration .....	35
Configuring Subscribe and Publish Topics .....	36
Microsoft Azure Service Setup.....	38
Testing Microsoft Azure Communication.....	48
Send data from Microsoft Azure to RTA gateway.....	50
Send data from RTA gateway to Microsoft Azure (Publish Topics) .....	52

QT Publish Trigger .....	54
Mapping - Transferring Data Between Devices .....	55
Display Mapping and Values .....	56
Display Data .....	56
Display String .....	59
Display String use case .....	61
Data and String Mapping – Auto-Configure .....	62
Data Mapping – Explanation .....	63
Data Mapping – Adding Diagnostic Information .....	64
String Mapping – Explanation .....	69
Mapping – Auto-Configure Mode to Manual Configure Mode .....	70
Mapping – Manual Configure Mode to Auto-Configure Mode .....	71
View as Text .....	72
Data Mapping .....	72
String Mapping .....	72
Base Triggering – Data Validation Triggering .....	73
Security Configuration .....	75
Security Configuration-Security Levels .....	76
Security - Log In .....	77
Security - Log Out .....	77
Email Configuration .....	78
Alarm Configuration .....	79
Diagnostics – Alarm Status .....	81
Alarms – Active .....	81
Alarms – Clear .....	82
Change of State (COS) Configuration .....	83
Diagnostics Info .....	84
Diagnostics Mapping .....	84
Diagnostics – ASCII .....	85
Diagnostic – MQTT Client .....	90
LED Configuration .....	92
Configuration Files .....	93
Export Configuration .....	93
Import Configuration .....	93

Save and Replace Configuration Using SD Card .....	95
Saving Configuration Using SD Card .....	95
Replacing Configuration Using SD Card .....	95
Intelligent Reset Button .....	96
Utilities .....	97

## Revision History

Version	Date	Notes
<b>8.4.5</b>	11/18/2019	<p>Features Added</p> <ol style="list-style-type: none"> <li>Released OPC UA Server (US) Protocol</li> <li>Ability to now Import/Export Template Files with out an FTP session</li> </ol> <p>Bug Fixes</p> <ol style="list-style-type: none"> <li>Updated Profinet Server (PS) on N34 hardware Platform</li> <li>Updated Wi-Fi software</li> </ol>
<b>8.6.0</b>	2/28/20	<p>Bug Fixes</p> <ol style="list-style-type: none"> <li>Omron Plc Communication fixes for EtherNet/IP</li> <li>Profinet GSDML Substitute values fix</li> </ol>
<b>8.7.4</b>	9/1/20	<p>Features Added:</p> <ol style="list-style-type: none"> <li>BMS, BM, DFM, DS, DM, TCP, USB, PBS have been ported to the latest base software</li> <li>TCP,BMS,BM now Available on N2E and N2EW hardware Platform</li> <li>New ASCII Mode Available on TCP/A/USB/WI protocols</li> <li>User Guides updated with more examples</li> </ol> <p>Bug Fixes:</p> <ol style="list-style-type: none"> <li>Improved Data Mapping and String Mapping performance</li> <li>Improved functionality/performance on EC,ETC,ES,MC,MS,BS,BC, A,,WI,PS protocols</li> </ol>
<b>8.7.22</b>	4/6/21	<p>Features Added:</p> <ol style="list-style-type: none"> <li>Support for RSLogix Versions 32 + with unsigned data type support</li> <li>ETC now support Long integer files (L files) for MicroLogix PLCS that support them</li> <li>SC now supports data block (DB) access</li> </ol>
<b>8.7.53</b>	4/28/21	<p>Features Added:</p> <ol style="list-style-type: none"> <li>Added support for the NNBU hardware platform</li> <li>Improved RFIDEas scanner support</li> <li>Updated MM and MRS to use Modbus RTU Client and Modbus RTU Server terminology</li> </ol>

Version	Date	Notes
8.9.22	2/5/24	<p>Features Added:</p> <ol style="list-style-type: none"> <li>1. Added priority-based reads for client protocols</li> <li>2. Added improved diagnostic timers for client protocols</li> <li>3. Reduced minimum delay between messages to zero ms on client protocols</li> <li>4. Added support for USB serial connections</li> <li>5. Added support for multiple connections on EtherNet/IP Adapter</li> <li>6. Added 100ms and 1000ms heartbeat values for diagnostic use</li> <li>7. Added configurable data size to EtherNet/IP adapter and DeviceNet Slave</li> <li>8. Added support for TTL communications on N34, NNA1, NNA4, N2E, and N2EW hardware</li> <li>9. Added support for JSON payloads to MQTT</li> <li>10. Added Network Bitmap Status to ASCII, USB, and TCP protocols</li> </ol> <p>Bug Fixes:</p> <ol style="list-style-type: none"> <li>11. Fixed COV Subscription Issues on BACnet MS/TP</li> <li>12. Fixed timing issues affecting gateway performance</li> <li>13. Fixed a bug where the Run Idle Header on the output instance for EtherNet/IP Scanner was not checked by default</li> </ol>
8.9.29	4/1/24	<p>Features Added:</p> <ol style="list-style-type: none"> <li>14. Added ability to do raw HEX byte copy when receiving data over ASCII, TCP, or USB.</li> </ol> <p>Bug Fixes:</p> <ol style="list-style-type: none"> <li>15. Fixed bug where function code 15 did not work on MM/MC.</li> <li>16. Fixed bug relating to writing zeros on start up on BS.</li> <li>17. Fixed bug where MQTT client did not appear in display data page when MQTT was paired with BACnet</li> </ol>
8.9.37	7/30/24	<p>Bug Fixes:</p> <ol style="list-style-type: none"> <li>18. EIP IO Communication fixes</li> <li>19. Timing fixes</li> <li>20. USB Fixes <ol style="list-style-type: none"> <li>a. Inactivity Timeout</li> <li>b. Inactivity Timeout Logging</li> <li>c. Port Restart Logging</li> <li>d. Webpage fixes</li> </ol> </li> <li>21. ProfiNet Timing Fix</li> <li>22. EIP PanelView Fixes <ol style="list-style-type: none"> <li>a. Support for Explicit Messaging</li> </ol> </li> </ol>

## Overview

The 460TCPAZR-NNA1 gateway Easily and securely connect Ethernet ASCII data packets to cloud applications . By following this guide, you will be able to configure the 460TCPAZR-NNA1 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](mailto: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.

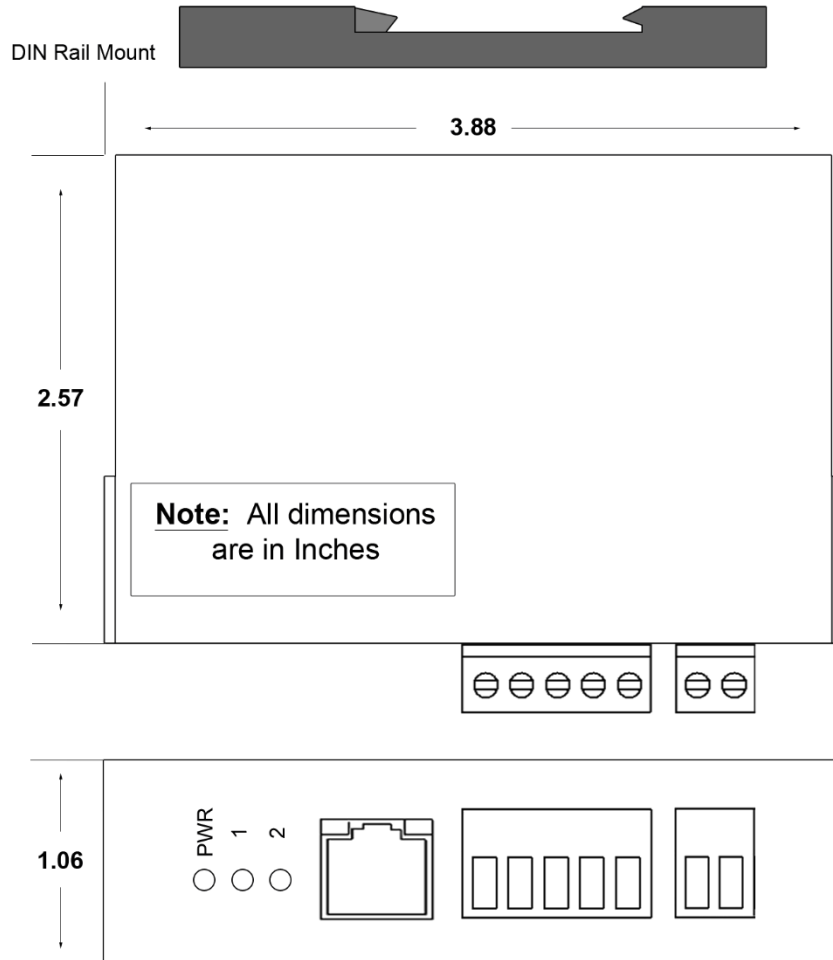
To find which hardware platform you are using:

- 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 – NNA1



## Powering the Gateway

- 1) Connect a 12-24 VDC power source to the gateway, Red Wire = (+) Black Wire = (-).
  - a) The unit draws 175mA @ 12 V.

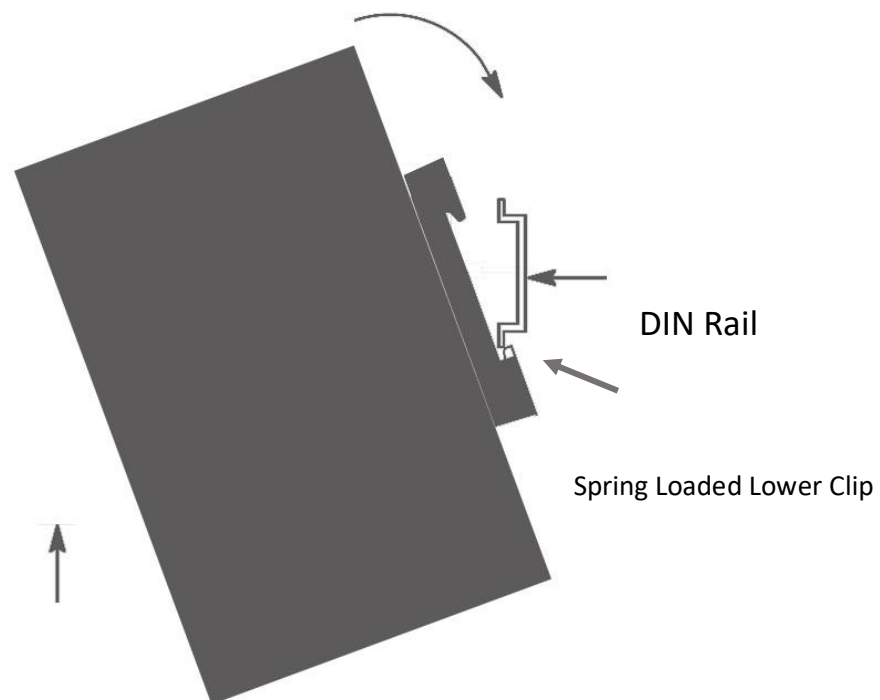


## Mounting with a DIN Rail

### 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 460TCPAZR-NNA1 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

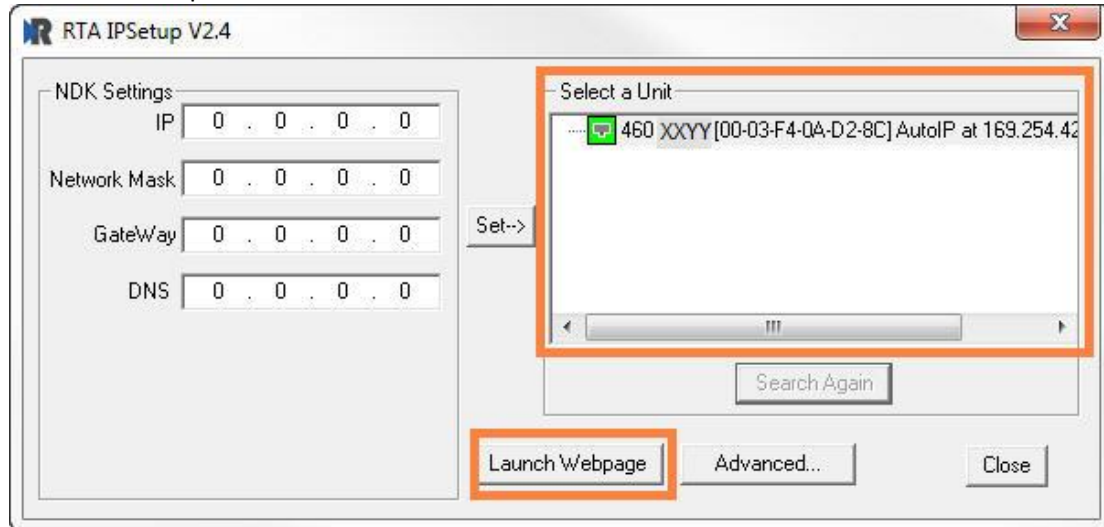
Follow these steps to remove your interface converter.

- 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](http://www.rtautomation.com/460-gateway-support) and download IPSetup.exe.



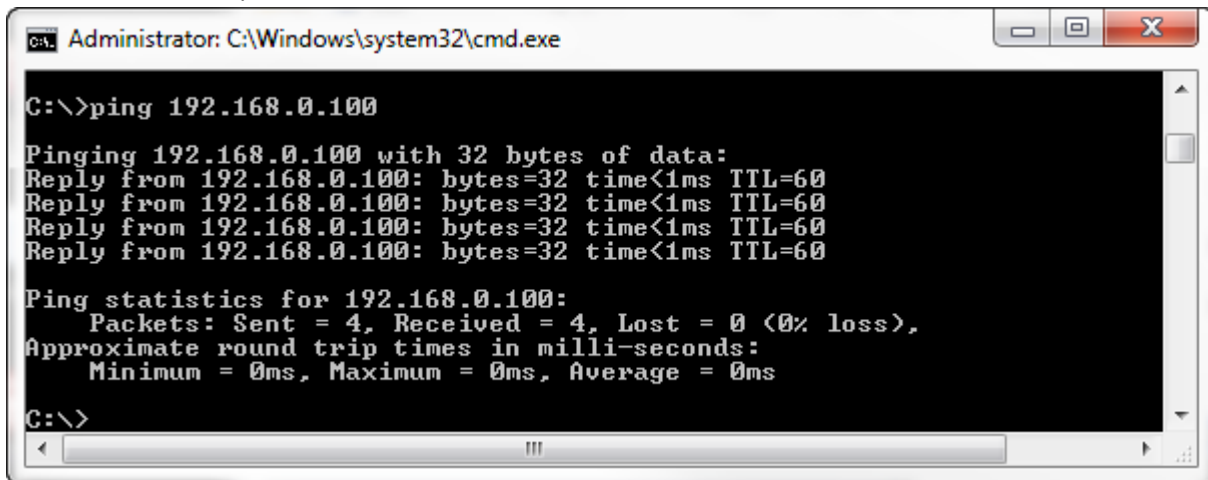
- 2) Run the IPSetup.exe program.
- 3) Find unit under "Select a Unit".
  - a. Change Gateway's IP address to match that of your PC if DHCP has failed.
    - i. You will know DHCP has failed if the gateway's IP address is AutoIP at 169.254.X.Y.
    - ii. If successful, it will say DHCP'd at ex: 192.168.0.100 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.



```
Administrator: C:\Windows\system32\cmd.exe

C:\>ping 192.168.0.100

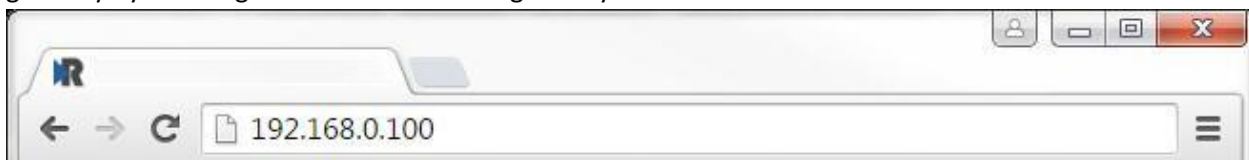
Pinging 192.168.0.100 with 32 bytes of data:
Reply from 192.168.0.100: bytes=32 time<1ms TTL=60
Reply from 192.168.0.100: bytes=32 time<1ms TTL=60
Reply from 192.168.0.100: bytes=32 time<1ms TTL=60
Reply from 192.168.0.100: bytes=32 time<1ms TTL=60

Ping statistics for 192.168.0.100:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

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.



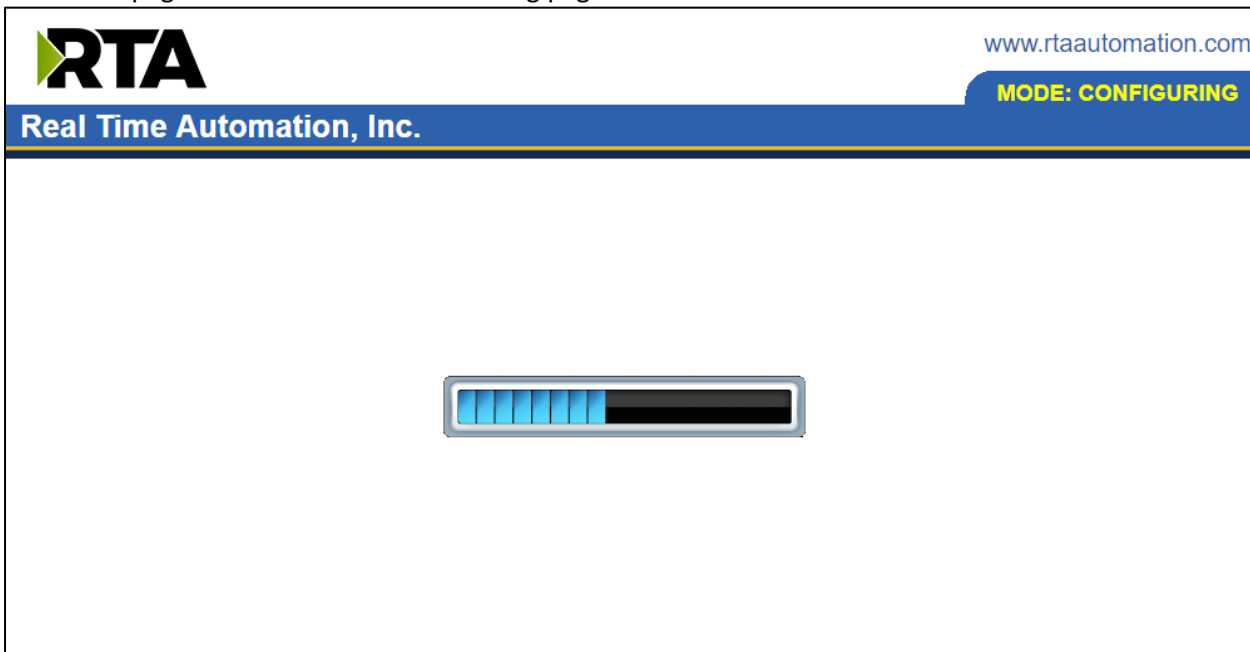
## Committing Changes to the Settings

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:

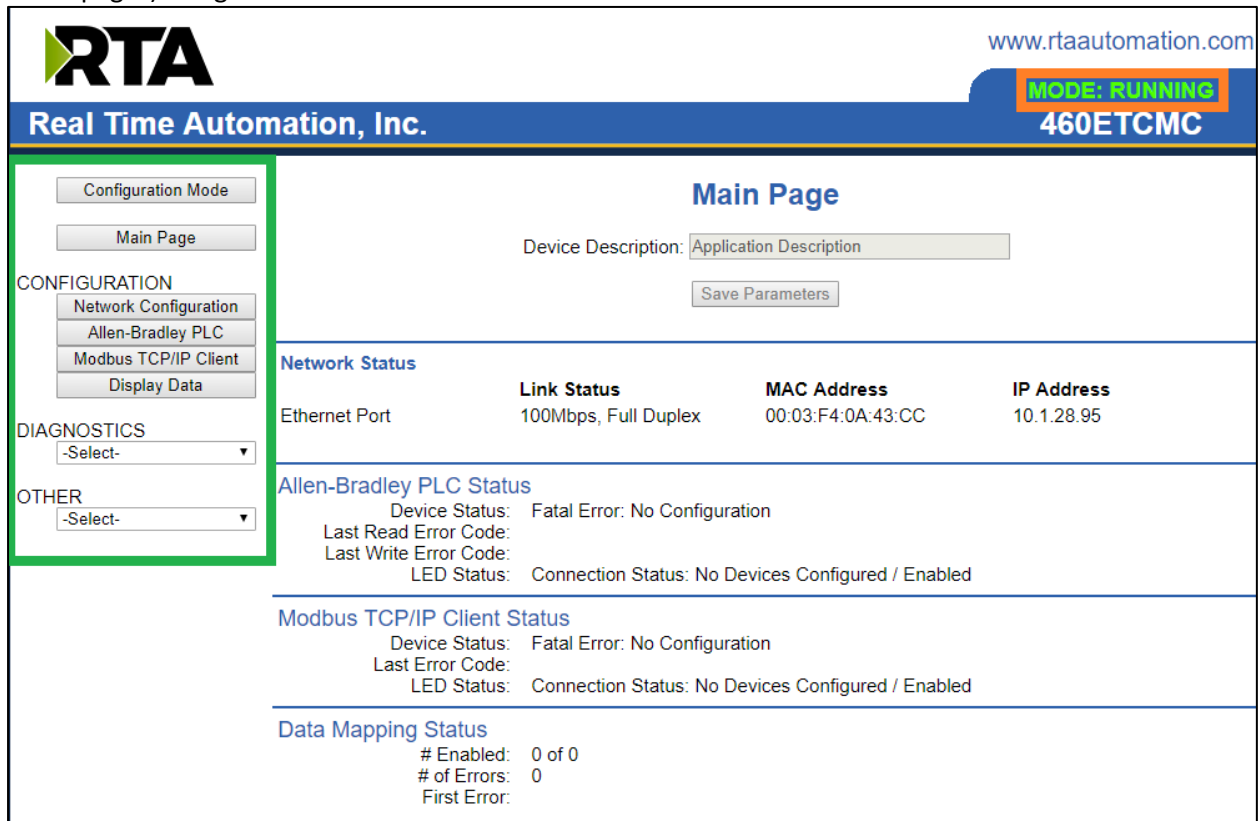
- Protocol communications are enabled
- Configuration cannot be changed during Running Mode. If changes are needed, click the **Configuration Mode** button shown in the green box below

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.



The screenshot shows the RTA Main Page interface. At the top left is the RTA logo and 'Real Time Automation, Inc.' At the top right is the website 'www.rtaautomation.com' and a status box indicating 'MODE: RUNNING' for device '460ETCMC'. On the left side, there is a navigation menu with buttons for 'Configuration Mode' (highlighted in green), 'Main Page', 'Network Configuration', 'Allen-Bradley PLC', 'Modbus TCP/IP Client', and 'Display Data'. Below these are dropdown menus for 'DIAGNOSTICS' and 'OTHER'. The main content area is titled 'Main Page' and includes a 'Device Description' field with the value 'Application Description' and a 'Save Parameters' button. Below this are several status sections: 'Network Status' with a table showing Ethernet Port, Link Status (100Mbps, Full Duplex), MAC Address (00:03:F4:0A:43:CC), and IP Address (10.1.28.95); 'Allen-Bradley PLC Status' with Device Status (Fatal Error: No Configuration), Last Read/Write Error Codes, and LED Status (Connection Status: No Devices Configured / Enabled); 'Modbus TCP/IP Client Status' with similar error and connection status information; and 'Data Mapping Status' showing 0 of 0 enabled items and 0 errors.

## 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.

**Main Page**

Device Description:

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.

### Network Configuration

Help  

#### Ethernet Configuration

Ethernet MAC Address: 00:03:F4:0B:C3:02

Ethernet Link:

IP Setting:

IP Address:

Subnet:

Default Gateway:

DNS Gateway:

**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.**

## TCP/IP Configuration

Click the **TCP/IP Configuration** button to access the primary configuration page for this device. If you have an N2E/N2EW hardware, then select which Network Interface ASCII comms will be on.

**TCP Configuration** Help

Network Interface: Ethernet Port (10.1.16.77) ▼

Port	Communication			Timeouts		
	TCP	IP Address	Port 0-65535	Inactivity 0-3600 sec	Connect 100-60000 ms	Reconnect 0-60 sec
10	Disabled ▼	0.0.0.0	0	0	100	0
11	Disabled ▼	0.0.0.0	0	0	100	0
12	Disabled ▼	0.0.0.0	0	0	100	0
13	Disabled ▼	0.0.0.0	0	0	100	0
14	Disabled ▼	0.0.0.0	0	0	100	0
15	Disabled ▼	0.0.0.0	0	0	100	0
16	Disabled ▼	0.0.0.0	0	0	100	0
17	Disabled ▼	0.0.0.0	0	0	100	0
18	Disabled ▼	0.0.0.0	0	0	100	0
19	Disabled ▼	0.0.0.0	0	0	100	0

Save Parameters

**To connect to a TCP server device on the network, enable a TCP client connection in the gateway for one of the available ports:**

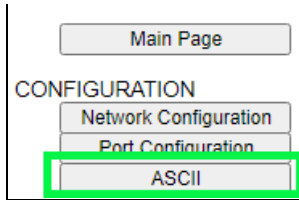
- 1) **TCP:** Under this dropdown select TCP Client.
- 2) **IP Address:** Enter the IP address of the TCP server to connect to.
- 3) **Port:** Enter the port to open a connection.
- 4) **Inactivity Timeout:** Enter the amount of time, in seconds, the gateway will wait for data to be transmitted to the gateway or sent from the gateway to the TCP device. If the timer expires, the gateway will close the connection.
- 5) **Connect Timeout:** Enter the amount of time, in milliseconds, the gateway will wait for the TCP server to accept the connection.
- 6) **Reconnect Timeout:** Enter the amount of time, in seconds, the gateway will wait after the Connect Timeout has expired before issuing another connect.

**To connect to a TCP client device on the network, enable a TCP server connection in the gateway for one of the available ports:**

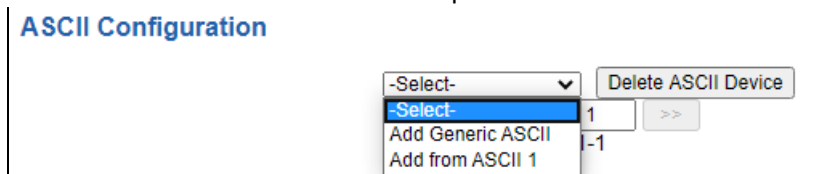
- 1) **TCP:** Under this dropdown select TCP server.
- 2) **IP Address:** Enter the IP address of the TCP client that will connect to the gateway.
- 3) **Port:** Enter the port to listen for a connection.
- 4) **Inactivity Timeout:** Enter the amount of time, in seconds, the gateway will wait for data to be transmitted to the gateway or sent from the gateway to the TCP device. If the timer expires, the gateway will close the connection.
- 5) **Connect Timeout:** Not used in this case (Client only).
- 6) **Reconnect Timeout:** Not used in this case (Client only).

## ASCII Configuration

After the port configuration has been completed, click the **ASCII** button to continue configuration.



- 1) To add an ASCII device or additional ASCII device, click the **-Select-** dropdown menu under ASCII Configuration and select **Add Generic ASCII** option.



- a) To remove a device, navigate to the ASCII device to delete using the **<<** and **>>** buttons and click the **Delete ASCII Device** button.
  - b) To create a new ASCII device with the same parameters already configured from another ASCII device, click the **-Select-** dropdown menu and select the **Add from ASCII X** option (where X represents the ASCII device you wish to copy parameters from).
  - c) Once created, you can make any additional changes needed to that new ASCII device.
- 2) The **Enable** check box should be selected for the device.
  - 3) **Port:** Select which port is being used for communication. This port must be configured on the Port/USB or TCP/IP (depending on your product) Configuration page. If it has not yet been configured, it will not display in this dropdown.
  - 4) Enter a **Device Label** to identify the device within the gateway.
  - 5) **LED Inactivity Timeout:** Enter the amount of time, in seconds, to wait before flashing the LED red indicating that no messages have been received or transmitted during this time.
  - 6) **Operation Mode:**
    - a. Mark Data New on Change of State: Send data to the mating technology, on a per point basis, upon a change of state. For more explanation see [the Receive Data – Operation Mode](#) section below.
    - b. Mark Data New on New Message: Send data to the mating technology for all data points, no matter change of state or not. For more explanation see the [Receive Data – Operation Mode](#) section below.

<input type="checkbox"/> <b>Enable</b>	<b>ASCII Device 1</b>		
Port	-Select- ▼	Device Label	ASCII01
LED Inactivity	0	0-60000 s	Operation Mode
			Mark Data New on Change of State ▼

## Receive Data

This side is configured to receive data from the ASCII device into the gateway.

**Receive Data (ASCII to 460ETCTCP)**

Enable:

Max Message Length:  1-1024 chars

Receive Character Timeout:  0-60000 ms

Delimiters

Start

End

Remove Delimiters from ASCII Message:

ASCII Parsing (Optional)

ASCII Conversion:

Gateway Hold Msg Timeout:  0-60000 ms

Queue Full Behavior:

Queue Size:  1-20 messages

Use the following fields to determine when a message has been received.

- 1) **Enable:** Check this box to configure the Receive Data section.
- 2) **Max Message Length:** Enter the max number of characters that can be received by the gateway.  
*Example:* Max Message Length is set to 5 and the message of "helloworld" was sent by the ASCII device. This will be sent to the other protocol as "hello" followed by "world" as two messages.
- 3) **Receive Character Timeout:** Enter the max amount of time (in ms) between characters that the gateway will wait before issuing a timeout and calling the message complete.  
*Example:* If Max Message Length varies in size, then use a timeout to call the message complete rather than message length. You can also use End Delimiters (below) to call a message complete.
- 4) **Number of Start Delimiters:** Select the number of delimiters that the gateway should look for before the gateway processes the data.
- 5) **Select Start Delimiters:** Select the Start Delimiters that the gateway should look for.
- 6) **Number of End Delimiters:** Select the number of delimiters that the gateway should look for to call a message complete.
- 7) **Select End Delimiters:** Select the End Delimiters that the gateway should look for.
- 8) **Remove Delimiters from ASCII Message:** If checked, the gateway will remove all delimiters that have been configured before sending it to the other protocol.
- 9) **ASCII Parsing (Optional):** Additional parsing can be performed on the string before being passed to the other protocol. See the [ASCII Configuration - ASCII Parsing](#) section for more information.

## 10) ASCII Conversion

- a. None - received string is treated as is
- b. ASCII Hex - received string must be Hex Digits 0 - F in ASCII
- c. ASCII Raw: received string will be mapped (memory copy) to the byte of data configured

11) **Gateway Hold Msg Timeout:** Enter the amount of time (in ms) to wait before sending a new message to the other protocol.

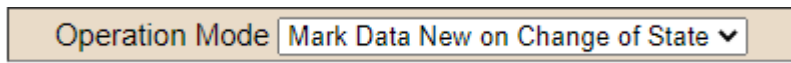
12) **Queue Full Behavior:** Select which message to discard when the queue is full. Once the queue is full, the gateway will discard either the oldest or newest message (Only used if Gateway Hold Msg Timeout is non-zero).

13) **Queue Size:** Select how many complete messages the gateway will hold before starting to discard (Only used if Gateway Hold Msg Timeout is non-zero).

## Receive Data – Operation Mode

### Mark Data New on Change of State (COS)

When data comes into the RTA gateway, then it will be sent over to the matting protocol only if the data has a different value.



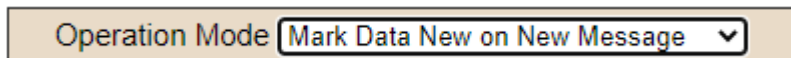
Operation Mode **Mark Data New on Change of State** ▼

*Example for 460ETCA*

Operator scans “HelloWorld” with a barcode scanner. That data is gathered in the ASCII side of the RTA gateway, and is then processed and sent over to the ETC side and written over to the Allen-Bradley PLC. The next time the operator scans the same barcode “HelloWorld”, the ASCII side gathers the data, but the data didn’t change so it will not be sent over to the ETC portion of the RTA gateway. If the operator scans “1234567890” with the barcode scanner, the ASCII side of the RTA gateway will process the data. Since the data has changed it will be sent over to ETC and sent over to the PLC.

### Mark Data New on New Message

When data comes into the RTA gateway, then it will be sent over to the matting protocol regardless if it’s the same data. This allows you to send the same data over again to the mating protocol.



Operation Mode **Mark Data New on New Message** ▼

*Example ETCA*

Operator scans “HelloWorld” with a barcode scanner. That data is gathered in the ASCII side of the RTA gateway, and is then processed and sent over to the ETC side and written over to the Allen-Bradley PLC. The next time the operator scans the same barcode “HelloWorld”, the ASCII side gathers the data, processes it, then sends it over to the ETC portion of the RTA gateway to be sent out. If the

operator scans “1234567890”, the ASCII side of the RTA gateway will process the data and send it over to ETC side and then to the PLC.

## Transmit Data

This side is configured to transmit data from the gateway into the ASCII device.

**Transmit Data (460ETCTCP to ASCII)**

Enable:

Max Message Length:  1-1024 chars

Transmit Timeout:  0-60000 ms

Delay Between Messages:  0-60000 ms

Add Delimiters to ASCII Message

Start

End

ASCII Conversion:

Use the following setup fields to help the 460 transmit an ASCII message.

- 1) **Enable:** Check this box to configure the Transmit Data section.
- 2) **Max Message Length:** Enter the max number of characters that can be transmitted by the gateway.
- 3) **Transmit Timeout:** Enter the amount of time (in ms) that the gateway waits before sending an ASCII message (0 Sends Immediately). If the data has changed before the time expires, the gateway immediately sends the message to the ASCII device.
- 4) **Delay Between Messages:** Enter the amount of time (in ms) that the gateway waits before verifying a Change of State of the ASCII message OR will start the Transmit Timeout.
- 5) **Number of Start Delimiters:** Select the number of delimiters that will be added onto the beginning of the ASCII string.
- 6) **Select Start Delimiters:** Select the Start Delimiters that should be added to the ASCII string.
- 7) **Number of End Delimiters:** Select the number of delimiters that will be added onto the end of the ASCII string.
- 8) **Select End Delimiters:** Select the End Delimiters that should be added to the ASCII string.
- 9) **ASCII Concatenating (Optional):** Additional concatenating can be performed on the string before being written to the ASCII device. See the [ASCII Configuration - ASCII Concatenating](#) section for more information.
- 10) **ASCII Conversion:**

- a. None - transmit string is treated as is.
- b. ASCII Raw - transmit string will be mapped (memory copy) from the byte of data configured.

## Transmit Data – Triggering Methods

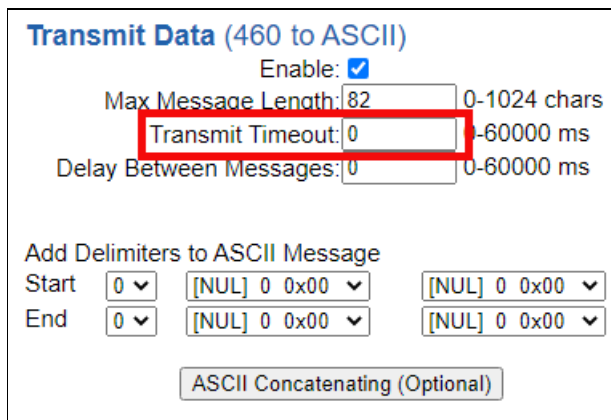
There are 3 methods that determine when the message is ready to be transmitted to your ASCII device:

- 1) **Cyclic** – This means that every x ms a new ASCII message will be transmitted, regardless of whether the data has changed or not.
- 2) **Triggering** – This means that a trigger event determines when a new ASCII message will be transmitted.

Some methods can co-exist with others. Here are the optional rules:

- a) **Option 1:** Change-Of-State is defaulted, so this method is chosen if the Transmit Timeout field is left at 0 and **ALL** data is new.

Example 1: From the PLC, send a message of “hello”, through the RTA gateway, the ASCII device sees “hello”. If you send “hello’ again, nothing will happen because of the RTA Change-of-State Rule.



**Transmit Data (460 to ASCII)**  
 Enable:   
 Max Message Length: 82 0-1024 chars  
 Transmit Timeout: 0 0-60000 ms  
 Delay Between Messages: 0 0-60000 ms

Add Delimiters to ASCII Message  
 Start 0 [NUL] 0 0x00 [NUL] 0 0x00  
 End 0 [NUL] 0 0x00 [NUL] 0 0x00

ASCII Concatenating (Optional)

- b) **Option 2:** Technology Triggering (A/USB/TCP/WI). This method is chosen if the Transmit Timeout field is left at 0 and the trigger variables (as described in ASCII Configuration – Technology Triggering Method section of this manual) are mapped. This will disable Change-of-State. This method is recommended if your product is **NOT** an 460ETC product.

**NOTE: If you have an 460ETC it’s high recommended you use the Optimization Triggering.**

*Example 2: Using the Technology Triggering Mappings (shown below in the Technology Triggering Method section) you can make the data new only with a trigger. If you want to send the same/new message of based on a trigger and NOT cyclically, then keep the Transmit Timeout at 0 **AND** use the 2 Trigger Mappings. See below for more examples in the ASCII Configuration – Technology Triggering Method.*

- c. **Option 3:** Cyclic and Trigger can co-exist. For this to happen, the Transmit Timeout field needs to be nonzero and the trigger variables (as described in ASCII Configuration – ASCII Message Triggering Method section of this manual) are also mapped.

*Example 3:* From the PLC, send a message of “hello”, based on whatever gets triggered first (the 3000ms Timeout or the trigger data point) the ASCII device will get updated. If the trigger data point is not updated, then the RTA gateway will send the data every x ms to the ASCII device. See below for more examples in the ASCII Configuration – Technology Triggering Method.

**Transmit Data (460 to ASCII)**

Enable:

Max Message Length:  0-1024 chars

**Transmit Timeout:**  0-60000 ms

Delay Between Messages:  0-60000 ms

Add Delimiters to ASCII Message

Start

End



## ASCII Configuration – Technology Triggering Method

This method allows the other protocol to signal when to send the next message using data handshakes. These “signals” are controlled using data variables (TransTrigger and TransHandshake) already in the mapping. The method will send the new/old data when triggered.

**NOTE:** These two data variables will need to be mapped manually on the Data Mapping webpage since it will not be mapped using Auto-Map.

While these two data variables are mapped, the Change-of-State method is disabled but messages can still be sent via the cyclic method, if configured. For more information on the ASCII triggering methods, please see the [Transmit Data – Triggering Methods](#) section of this user guide.

<input checked="" type="checkbox"/> Enable	Mapping 1	
<b>Source</b>	<input type="checkbox"/> Enable Manipulation	<b>Destination</b>
Group: rta-ps OUT_Slot12[0] (Int8) ▾ Start: OUT_Slot12[0] ▾ End: OUT_Slot12[0] ▾		Group: ASCII01 TransTrigger (UInt16) ▾ Start: TransTrigger ▾ End: TransTrigger
<input checked="" type="checkbox"/> Enable	Mapping 2	
<b>Source</b>	<input type="checkbox"/> Enable Manipulation	<b>Destination</b>
Group: ASCII01 TransHandshake (Ui) ▾ Start: TransHandshake ▾ End: TransHandshake ▾		Group: rta-ps IN_Slot2[0] (Int8) ▾ Start: IN_Slot2[0] ▾ End: IN_Slot2[0]

### How the triggering method works: The example shown below is our 460PS\* (\*A/TCP/USB)

- 1) The mating protocol sends a numbered value to the ASCII TransTrigger diagnostic variable. This value must be different from the previous value for a new message to be triggered. The following example is Slot 12[0] as the trigger for the PLC to update everything in Slot 11[0] which is the data.
- 2) Depending on the TransTigger value in the Display Data page, one of 4 things will occur:
  - a) If TransTrigger = 65535, then the triggering method is disabled. Usually on powerup.

#### RTA460 Display Data Example:

ASCII			460PSA ←←		Profinet IO	
Name	Value (Hex)		Manipulation	Name	Value (Hex)	
TransTrigger	65535	0xFFFF	←←	OUT_Slot12[0]	-1	0xFF

b) If TransTrigger = 0, then the triggering method is enabled, but no message will transmit.

**RTA460 Display Data Example:**

ASCII			460PSA ←←	Profinet IO		
Name	Value (Hex)		Manipulation	Name	Value (Hex)	
TransTrigger	0	0x0000	←←	OUT_Slot12[0]	0	0x00
Trans_Field01	11	0x000B	←←	OUT_Slot11[0]	11	0x0B

**TIA Portal Example:** The data will still go to the RTA gateway, however the RTA gateway will NOT transmit the data to the ASCII device until the Slot12[0] triggers the TransTrigger.

Name	Address	Display for..	Monitor ...	Modify va...	Comment
1	%QB80	DEC+/-		11	Data to RTA ASCII
2	%QB88	DEC+/-		0	Trigger data to RTA

c) If TransTrigger is between 1-65534 **AND** the value **IS** equal to the TransHandshake diagnostic variable, then no new message will transmit, until Slot 12[0] triggers again.

**RTA460 Display Data Example:**

ASCII			460PSA ←←	Profinet IO		
Name	Value (Hex)		Manipulation	Name	Value (Hex)	
TransTrigger	1	0x0001	←←	OUT_Slot12[0]	1	0x01

ASCII			460PSA →→	Profinet IO		
Name	Value (Hex)		Manipulation	Name	Value (Hex)	
TransHandshake	1	0x0001	→→	IN_Slot2[0]	1	0x01

d) If TransTrigger is between 1-65534 **AND** the value **IS NOT** equal to the TransHandshake diagnostic variable, then a new message will be transmitted. The value in TransTrigger will then be moved to TransHandshake.

**TIA Portal Example:** Once the Slot12[0] increments (data is sent from the PLC to the ASCII device) then the Slot2[0] will get updated with the handshake

Name	Address	Display format	Monitor value	Modify value	Comment
1	%QB80	DEC+/-		11	Data to RTA ASCII
2	%QB88	DEC+/-		1	Trigger data to RTA

Address	Display format	Monitor value	Modify value	Comment
%IB92	DEC+/-	0		Data From RTA
%IB100	DEC+/-	1		RTA Handshake

## ASCII Configuration – ASCII Parsing

The ASCII parsing feature allows you to break apart an incoming ASCII string by delimiter or character offset into multiple data fields. You can then apply a data type to the fields and deliver them to user defined locations in the mating protocol. Click the **ASCII Parsing (Optional)** button at the bottom of the ASCII Configuration page to access the ASCII Parsing Configuration page for this device.

**ASCII Parsing Configuration**
Help

<<  >>  
1-1

**ASCII Device 1 (ASCII01)**

Max Number of Fields:  1-50
Min Number of Fields:  1-50

Parsing Delimiter:

Field	Start Location	Length	Data Type	Internal Tag Name
1:	<input type="text" value="1"/>	<input type="text" value="0"/>	<input type="text" value="String"/>	<input type="text" value="Recv_Field01"/>

Sample/Test Data:

Field	Result
1:	Number of Fields Invalid

\* The length of result is greater than 64 characters

- 1) **Max Number of Fields:** This indicates the max number of fields the ASCII data will be parsed into (up to 50 values per message).
- 2) **Min Number of Fields:** This indicates the min number of fields that must be present in an ASCII string for the message to be considered valid. An error will be flagged if the actual number of fields is less than this value.
- 3) **Parsing Delimiter:** This defines the delimiter that will be used to parse an ASCII message. If delimiters are not present, select UNUSED and use the length fields to parse the message.
- 4) **Start Location & Length:**
  - a. If a parsing delimiter is used, the **Start Location** will be the first character of the data field. The **Length** will be the number of characters from the Start Location. If the **Length** is 0, the gateway will read the entire field.
  - b. If the parsing delimiter is unused, then the **Start Location** will be the first character of the string. The **Length** will be the number of characters from the Start Location. If the **Length** is 0, the gateway will read the entire message from the **Start Location** to the end of the ASCII string.
- 5) **Data Type:** Select the data type of the parsed value.
- 6) **Internal Tag Name:** Enter a name to reference this tag within the gateway's display and mapping pages.

## ASCII Configuration – ASCII Parsing Examples

### Example #1 - Parsing a message using the Parsing Delimiter option:

In this example, we are separating the string “12.25,SP100,temp setpoint” by a comma delimiter. The first value is being parsed into a float data type, the second and third values are being parsed into a string data type. Since the Min Number of Fields is 3, all 3 fields must be present for the message to be considered valid and processed. The output is seen below:

ASCII Device 1 (ASCII01)				
Max Number of Fields: <input type="text" value="3"/> 1-50		Min Number of Fields: <input type="text" value="3"/> 1-50		
Parsing Delimiter: <input type="text" value=","/> 44 0x2c				
<input type="button" value="Update Fields"/>				
Field	Start Location	Length	Data Type	Internal Tag Name
1:	<input type="text" value="1"/>	<input type="text" value="0"/>	<input type="text" value="32 Bit Float"/>	<input type="text" value="Recv_Field01"/>
2:	<input type="text" value="1"/>	<input type="text" value="0"/>	<input type="text" value="String"/>	<input type="text" value="Recv_Field02"/>
3:	<input type="text" value="1"/>	<input type="text" value="0"/>	<input type="text" value="String"/>	<input type="text" value="Recv_Field03"/>
<input type="button" value="Save Parameters"/>				
Sample/Test Data: <input type="text" value="12.25,SP100,temp setpoint"/>				<input type="button" value="Show Results"/>
Field	Result			
1:	<input type="text" value="12.25"/>			
2:	<input type="text" value="SP100"/>			
3:	<input type="text" value="temp setpoint"/>			
* The length of result is greater than 64 characters				

### Example #2 - Parsing a message without the Parsing Delimiter option:

In this example, we are separating the fields in the string “12.25,SP100,temp setpoint” using the start and length parameters. The first value is being parsed from the 1<sup>st</sup> character for a length of 5 and stored into a float data type. The second value is being parsed from the 7<sup>th</sup> character for a length of 5 characters and stored into a string data type. The third value is being parsed starting from the 13<sup>th</sup> character for the rest of the remaining characters and stored into a string. The fourth value contains the entire ASCII message and is stored into a string. Only the first field needs to be present for the data to be considered valid and will be processed. If less than field 1 is present, the message will not be parsed and will be flagged an error. The output is seen below:

ASCII Device 1 (ASCII01)				
Max Number of Fields: 4		1-50		Min Number of Fields: 1
				1-50
Parsing Delimiter: UNUSED				
Update Fields				
Field	Start Location	Length	Data Type	Internal Tag Name
1:	1	5	32 Bit Float	Recv_Field01
2:	7	5	String	Recv_Field02
3:	13	0	String	Recv_Field03
4:	1	0	String	Recv_Field04
Save Parameters				
Sample/Test Data: 12.25,SP100,temp setpoint				Show Results
Field	Result			
1:	12.25			
2:	SP100			
3:	temp setpoint			
4:	12.25,SP100,temp setpoint			

### Example #3 - Parsing a message using the Parsing Delimiter option and Start Location and Length:

In this example, we are separating the fields in the string "12.25,SP100,temp setpoint" using the comma delimiter, the start, and length fields. The first value is being parsed from the 1<sup>st</sup> character for a length of 2 and stored into an integer data type. The second value is being parsed from the 3<sup>rd</sup> character of the second comma-parsed field for the remainder of that field and stored into an integer data type. The third value is being parsed starting from the 1<sup>st</sup> character of the third comma-parsed field for that entire field and stored into a string. All 3 fields need to be present for the message to be valid. The output is seen below:

ASCII Device 1 (ASCII01)				
Max Number of Fields: 3		1-50		Min Number of Fields: 3
				1-50
Parsing Delimiter: , 44 0x2c				
Update Fields				
Field	Start Location	Length	Data Type	Internal Tag Name
1:	1	2	16 Bit Int	Recv_Field01
2:	3	0	16 Bit Int	Recv_Field02
3:	1	0	String	Recv_Field03
Save Parameters				
Sample/Test Data: 12.25,SP100,temp setpoint				Show Results
Field	Result			
1:	12			
2:	100			
3:	temp setpoint			
* The length of result is greater than 64 characters				

## ASCII Configuration – ASCII Concatenating

The ASCII concatenating feature allows you to combine multiple data points and locations, in the mating protocol, into a single ASCII string. Click the **ASCII Concatenating (Optional)** button at the bottom of the ASCII Configuration page to access the ASCII Concatenating Configuration page for this device.

**ASCII Concatenating Configuration**
Help

<< 1 >>  
1-1

**ASCII Device 1 (ASCII01)**

Number of Fields:  1-50
Concatenating Delimiter:  ▾

Field	Data Type	Internal Tag Name or Constant Name	Data Format	Max Characters	Padding	Add Delim
1:	String ▾	Trans_Field01	N/A ▾	<input type="text" value="0"/>	None ▾	<input checked="" type="checkbox"/>

**Sample Result**

XXXXXXXXXX

\* The length of result is greater than 64 characters

- 1) **Number of Fields:** This indicates how many values will be concatenated together to form a single ASCII message (up to 50 values per message).
- 2) **Concatenating Delimiter:** This adds a delimiter between data fields in the ASCII string. If a delimiter should not appear between each of the fields, select UNUSED.
- 3) **Data Type:** Select the data type of the parsed value.
  - a. Signed and Unsigned 8/16/32/64 Bit Integers
  - b. 32/64 Bit Floating Points
  - c. String – in order to use, a String data type must be selected in the other protocol. Cannot concatenate an Integer to a String.
  - d. Constant String
- 4) **Internal Tag Name/Constant Name:**
  - a. If data type other than Constant String is selected, then this will be the name to reference this tag within the gateway. This value is used on the display page and the mapping page.
  - b. If data type Constant String is selected, then this is the string value that will send.
- 5) **Data Format:**
  - a. %d – used for Signed Integers
  - b. %u – used for Unsigned Integers
  - c. %lf – used for Floating Points with no set decimal precision
  - d. %.1lf...%.6lf – used for Floating Points to show the offset of the decimal point value

- i. EX: 123.456789 set as %.3lf will display as 123.456
  - e. %e – used for Exponential Notation
  - f. %x – used to represent Hexadecimal values for Signed/Unsigned Integers or Floating points
  - g. String and Constant String data types do not use this field
- 6) **Max Characters:** This is the max number of characters that can be transmitted for a single field.
- Special Cases
- a. If set to 0, the entire field is transmitted.
  - b. If the length of the value is less than the Max Characters, then the Padding Character will be used (if set).
  - c. If the length of the value is greater than the Max Characters, then the value will be truncated.
- 7) **Padding:** If the length of the value is less than the Max Characters padding Zeroes, Spaces, or Nothing to the remaining character placeholders, the padding will occur to the left of the value.
- 8) **Add Delim:** Used when a concatenating delimiter is selected. Check to add the concatenating delimiter to the end of that field.
- 9) **Sample Result:** This will display an example of how the data will output. This will not display live data. It provides an example of the string structure.
- NOTE:** Sample Result field will only show the first 64 characters of the message.
- a. String data and constant data types will display as x's.
  - b. Any other data type will display as i's.
- NOTE:** For display purposes, if Max Characters is set to 0, only 10 characters will display for that field in the Sample Result section. The true value, if larger, will be processed correctly.
- EX: Field 1 is set for a String data type and Max Characters is set to 0, only 10 x's will display in the sample result even though the max character length is set to 50.





Below is a sample of the Transmit Data is set up with the following delimiters:

**Transmit Data (460 to ASCII)**

Enable:

Max Message Length:  0-1024 chars

Transmit Timeout:  0-60000 ms

Delay Between Messages:  0-60000 ms

Add Delimiters to ASCII Message

Start

End

**Example 1 Sample Result:** This use case is sending data via 5 PLC tags. Using the concatenating setup example and the transmit example, the ASCII data will be displayed within your ASCII device shown in the example below.

!123,0000003039      1234,0000123.45,ASCII Test,RTA\_MSG#\$


## ASCII Configuration – ASCII Message Counter

There is an additional ASCII variable that is very useful to access within the gateway’s mating protocol. This data variable will need to be added manually since it will not be mapped using Auto-Map.

**RecvCount**- indicates how many ASCII messages have been successfully read by the gateway for that device. A successful incoming message means that at least one of our three end cases (Max Length, Timeout or Delimiters) has been met. This will match the Diagnostic Variable Successful Receive Count for each ASCII device.

This variable can be mapped to the mating protocol using the Data Mapping webpage. It is mapped just like the Status\_XY variable described in the [Data Mapping- Adding Diagnostic Information](#) section of this user guide.

**Example:** For this example, the other protocol in the gateway is the Allen-Bradley PLC. As you can see from the picture below, the RecvCount for ASCII Device 1 is mapped to the first index of a PLC tag array called test\_cnt. The data type of this tag is an Int32 to match the data type of RecvCount. The tag test\_cnt[0] will now hold the number of successfully read messages from ASCII Device 1.

Mapping 1		
Source	Enable Manipulation	Destination
Group: ASCII01 RecvCount (Uint32) Start: RecvCount End: RecvCount	<input type="checkbox"/> Enable Manipulation 	Group: ETC01 test_cnt[0] (Int32) Start: test_cnt[0] End: test_cnt[0]

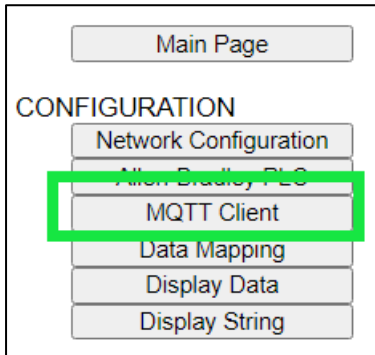
**Application Use:** This is particularly useful for applications connecting devices like barcode scanners and weigh scales. The gateway will cyclically update the mating protocol with the last ASCII message sent, a change in the RecvCount is the only way to identify a new message if the messages are identical.

## MQTT Client Configuration

You can configure up to three MQTT connections.

1. Configure up to one Microsoft Azure (ARZ) connection.

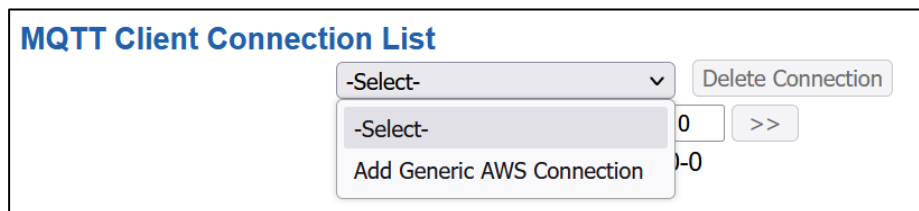
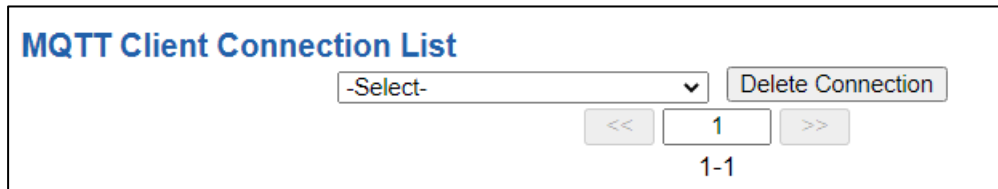
Click the **MQTT** button to continue configuration.



## Microsoft Azure (AZR) Configuration

You can only configure one Microsoft Azure connection with your RTA product.

- 1) To add an ARZ connection, click the -Select- dropdown menu under MQTT Client Connection List and select **Add Generic AZR Connection** option.
  - a. To remove a device, navigate to the AZR device to delete and click the **Delete Connection** button.



- 2) The **Enable** check box should be selected for the device.
- 3) Enter a **Device Label** to identify the device within the gateways mapping.
- 4) Select which **Network Interface** to use for Microsoft Azure connection. Option only available on the N2E hardware.
- 5) **Primary Connection String**: Enter the primary connection string for the device in Azure.
- 6) Enter the **TCP Port** for the MQTT broker to open a connection on. If this value doesn't match, the gateway will not open a connection.

- 7) **Keep Alive:** Enter in the amount of time that the gateway should attempt to ping the broker to keep the MQTT connection alive, 0 disables this feature.
- 8) **Client ID:** The MQTT client ID to be used when connecting to the Broker.
  - a. **NOTE:**This field is not user configurable and will be populated automatically based on the supplied primary connection string when the webpage is saved.
- 9) Check the **Add Timestamp to Publishes** checkbox to add a timestamp to the payload.

<input type="checkbox"/> Enable	<b>Azure 1</b>	
Device Label	QT01	Network Interface Ethernet Port (DHCP Assigned) ▾
Primary Connection String		
Client ID		<input type="checkbox"/> Add Timestamp to Publishes
TCP Port	8883 1-65535 (Default: 8883)	Keep Alive 60 0-200 sec (0 to Disable)
# of JSON Name/Value Pairs 0 0-500		

### Configuring Subscribe and Publish Topics

# of JSON Name/Value Pairs 3 0-500	
# of Publish Paths 1 0-250	# of Subscribe Paths 1 0-1
<input type="button" value="Generate Paths"/>	

### JSON Name/Value Pairs

Line #	Path	JSON Name	JSON Point Type
1	-No Path Defined- ▾		INT (8-bit) ▾
2	-No Path Defined- ▾		INT (8-bit) ▾
3	-No Path Defined- ▾		INT (8-bit) ▾

<< 1-3 >>

### Publish Paths (460PSQT to MQTT)

Line #	Enable	Path Name	QoS
1	<input checked="" type="checkbox"/>		0 ▾

<< 1-1 >>

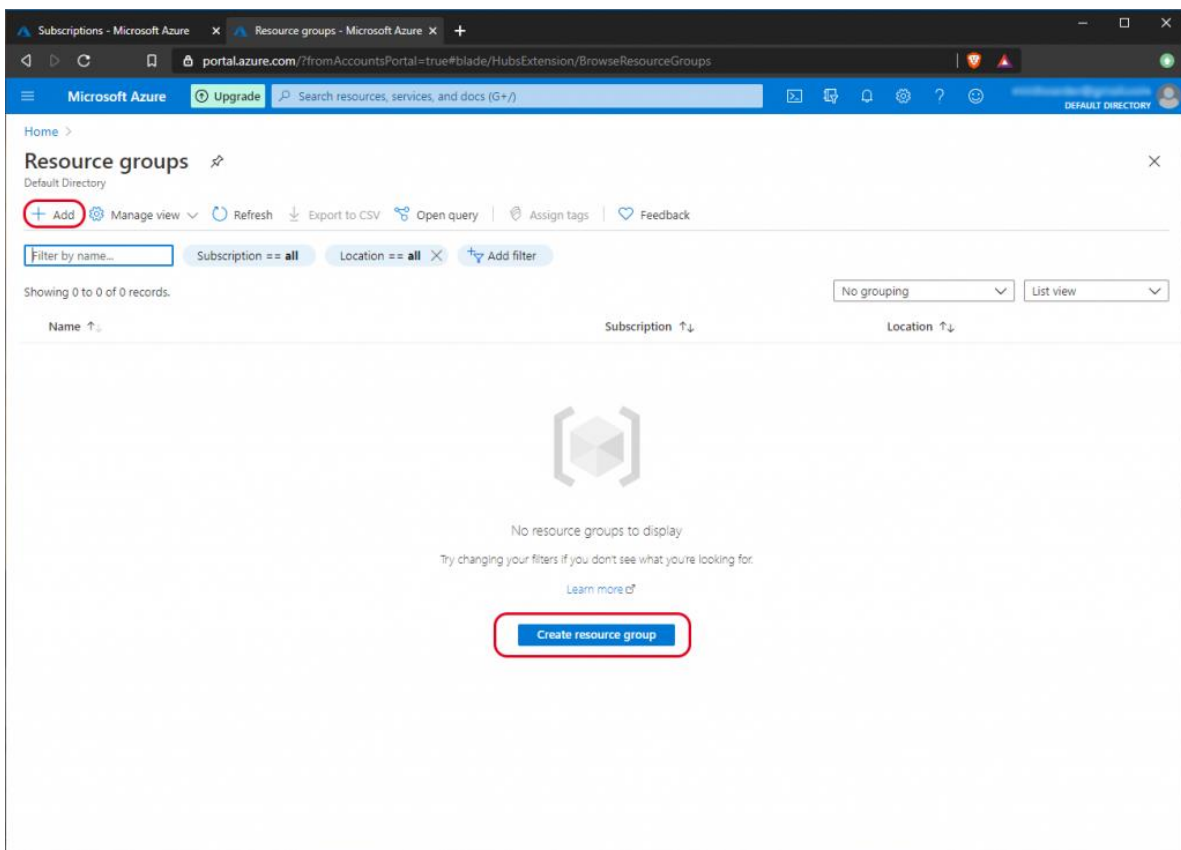
- 1) Enter in “# of Subscribe Paths” and/or “# of Publish Paths”.
- 2) Enter in “# of JSON Name/Value Pairs”
- 3) Click the **Generate Paths** button to have the lines generated for you.
- 4) **# of Subscribe Paths:** Enter in the number of topics to subscribe from the broker. Once the topics are subscribed to, the MQTT broker will publish the messages to the gateway.

- a) **NOTE:** Subscribe paths on Microsoft Azure are limited to 1 and are not user configurable. This is due to Microsoft Azure not utilizing a typical pub/sub broker and is instead using Cloud-to-Device messaging and expecting a pre-defined endpoint for these messages. The subscribe path may not populate immediately when generated, to resolve this, click view subscribe paths and then hit save parameters with the subscribe paths selected.
  - 5) **# of Publish Paths:** Enter the number of topics to publish to the broker from the mating protocol.
    - a) **NOTE:** Publish paths will be pre-pended with the defined Microsoft Azure Device-to-cloud message path when the page is saved while the View Publish Paths option is selected. This is because Microsoft Azure does not utilize a generic pub/sub broker and instead utilizes Device-to-Cloud messaging expecting a pre-defined endpoint for these messages.
  - 6) **# of JSON Name/Value Pairs:** Enter the number of JSON name/value pairs to configure, each name/value pair can be associated with a single publish or subscribe path.
  - 7) Select the publish or subscribe path each name/value pair should be associated with.
  - 8) Select the **Point Type** of the name/value pairs
- Click **Save Parameters** button when complete.

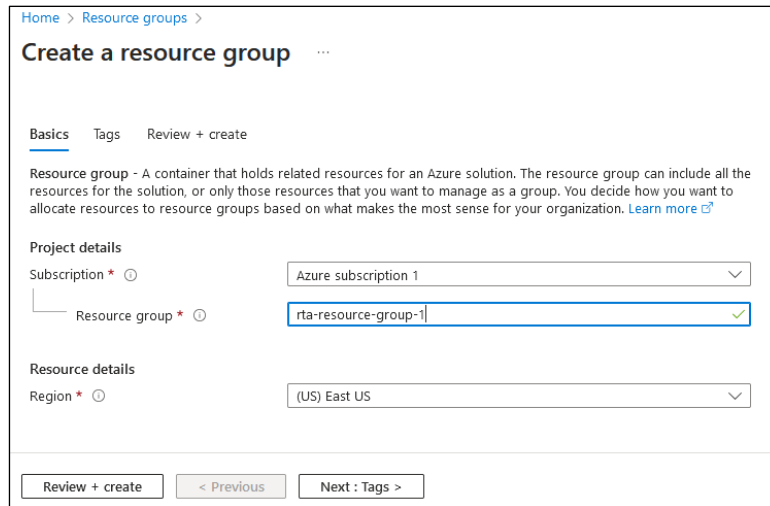
## Microsoft Azure Service Setup

Please note this section outlines the bare minimum configuration to get the RTA gateway connected to Azure and is not necessarily a recommended configuration in a production environment.

1. Create a Microsoft Azure subscription if you do not have one already
2. Create a resource group
  - a. In the search bar at the top type in “Resource Groups” and select the resource groups under services to navigate to the resource groups page.
  - b. Hit Add in the top left corner, there may also be a create resource group button in the middle if no resource groups exist.



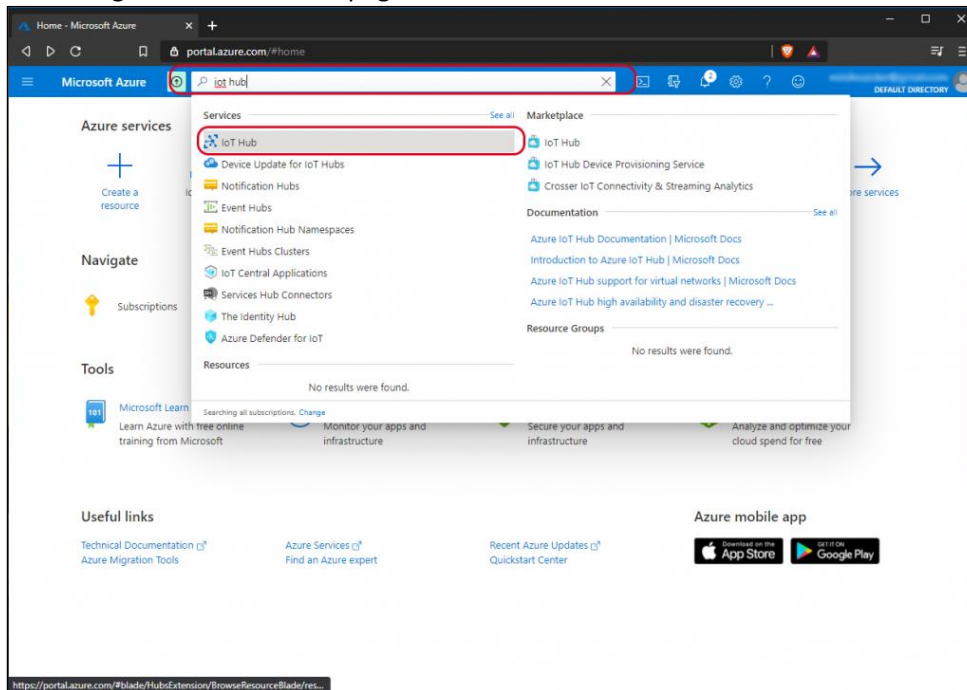
- c. In the create a resource group window give the resource group a name, the available Azure subscription should be selected by default.



- d. Hit Review + Create in the bottom left corner.
- e. In the review window, hit create in the bottom left corner to create the resource group.
- f. You should be re-directed back to the resource groups window where you can see the newly created resource group is listed.

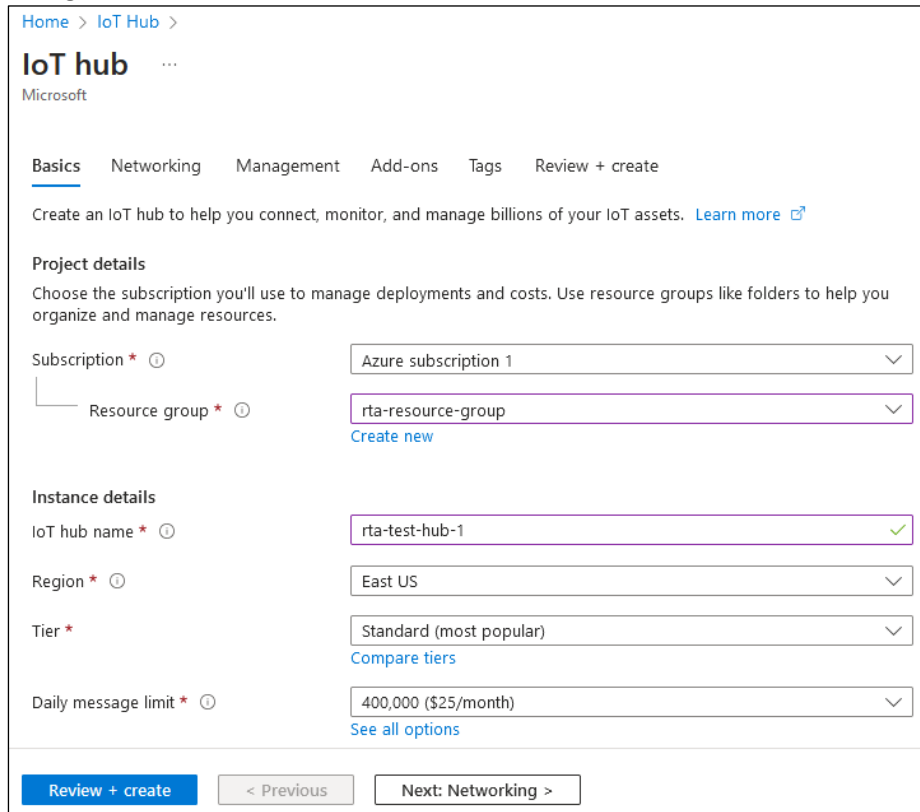
### 3. Create an IoT Hub

- a. In the search bar at the top type in "IoT Hub" and select the IoT under services to navigate to the IoT Hub page.



- a. Hit Add in the top left corner, there may also be a create IoT Hub button in the middle if no IoT Hubs exist.

- b. Select the resource group created previously, enter a name for the hub, and select a region.



Home > IoT Hub >

## IoT hub

Microsoft

Basics Networking Management Add-ons Tags Review + create

Create an IoT hub to help you connect, monitor, and manage billions of your IoT assets. [Learn more](#)

**Project details**  
Choose the subscription you'll use to manage deployments and costs. Use resource groups like folders to help you organize and manage resources.

Subscription \* ⓘ Azure subscription 1

Resource group \* ⓘ rta-resource-group  
[Create new](#)

**Instance details**

IoT hub name \* ⓘ rta-test-hub-1 ✓

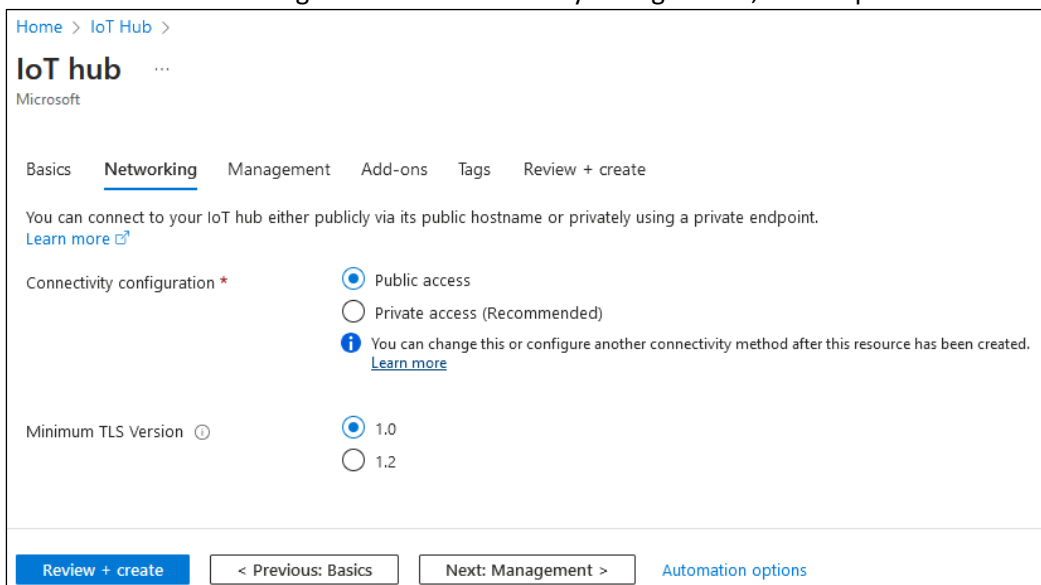
Region \* ⓘ East US

Tier \* Standard (most popular)  
[Compare tiers](#)

Daily message limit \* ⓘ 400,000 (\$25/month)  
[See all options](#)

[Review + create](#) < Previous Next: Networking >

- c. Hit Next at the bottom of the page to proceed to the networking tab.  
d. In the networking tab under connectivity configuration, ensure public access is selected.



Home > IoT Hub >

## IoT hub

Microsoft

Basics **Networking** Management Add-ons Tags Review + create

You can connect to your IoT hub either publicly via its public hostname or privately using a private endpoint. [Learn more](#)

Connectivity configuration \*  Public access  
 Private access (Recommended)  
**i** You can change this or configure another connectivity method after this resource has been created. [Learn more](#)

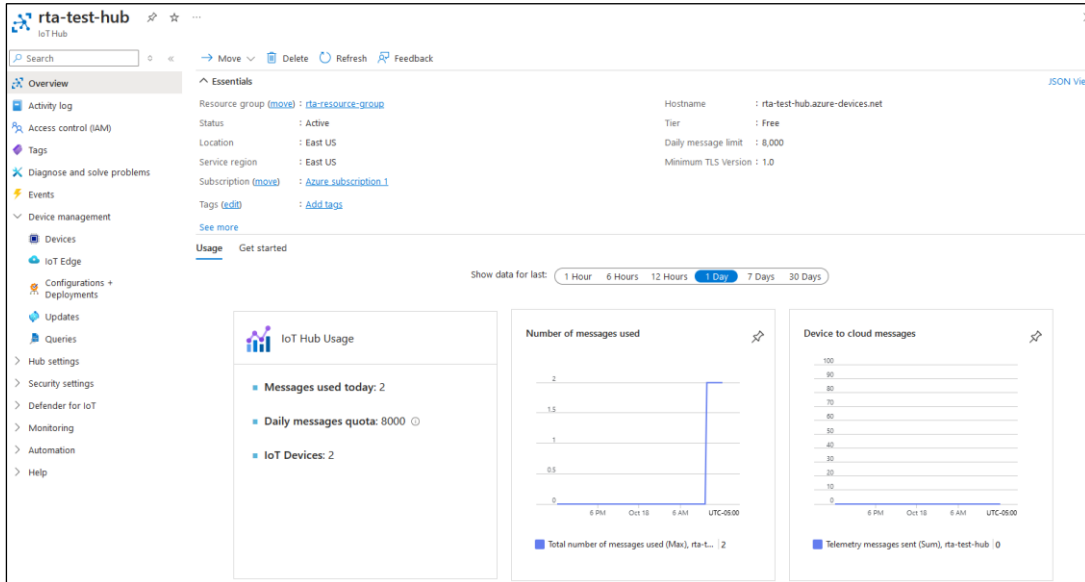
Minimum TLS Version ⓘ  1.0  
 1.2

[Review + create](#) < Previous: Basics Next: Management > [Automation options](#)

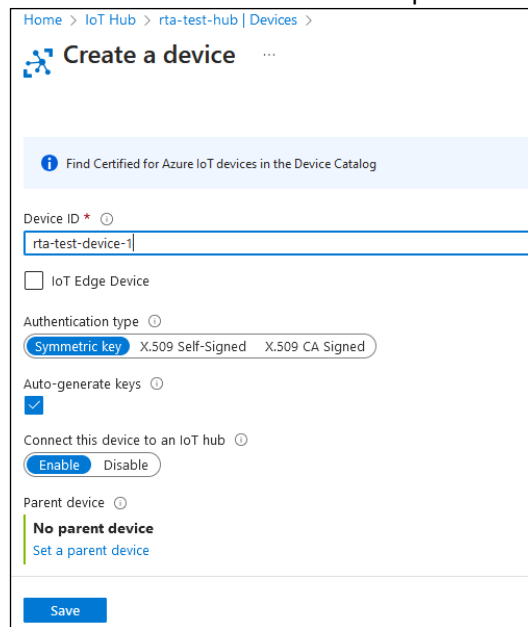
- e. Hit review and create in the bottom left corner of the page.  
f. On the review and create page, hit create in the bottom left corner of the page.



- g. One the deployment is completed click on “Go to Resource” to be re-directed to the newly created IoT Hubs overview page. If go to resource is not an option, navigate to the IoT hub and select your newly created IoT hub to access the overview page.

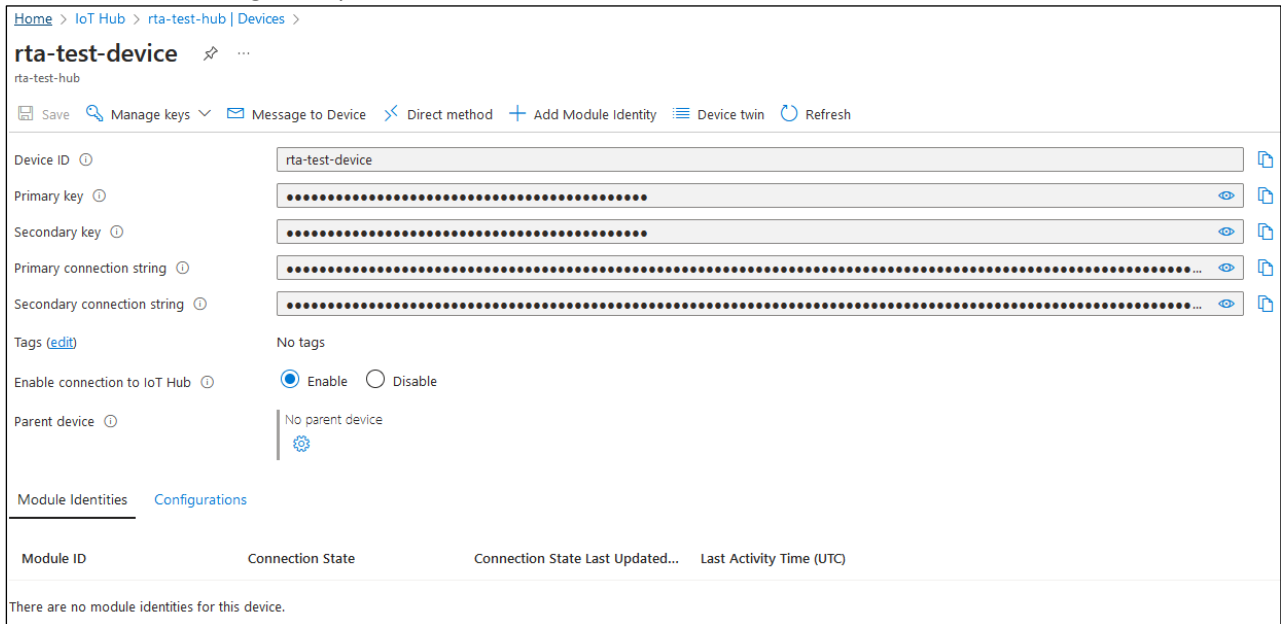


- h. In the left panel of the IoT Hub overview select Devices under the Device Management section.
- i. Click Add Device button in the top left corner of the page to create a new IoT device in this hub.
- j. In the create a device window enter a name for the device in the Device ID section and ensure the “Connect his device to an IoT hub” option is enabled.



- k. Hit Save in the bottom left corner of the window to create the device and be re-directed to the device list in the IoT Hubs overview window.
- l. Click on the newly created device in the device list to view the devices configuration.

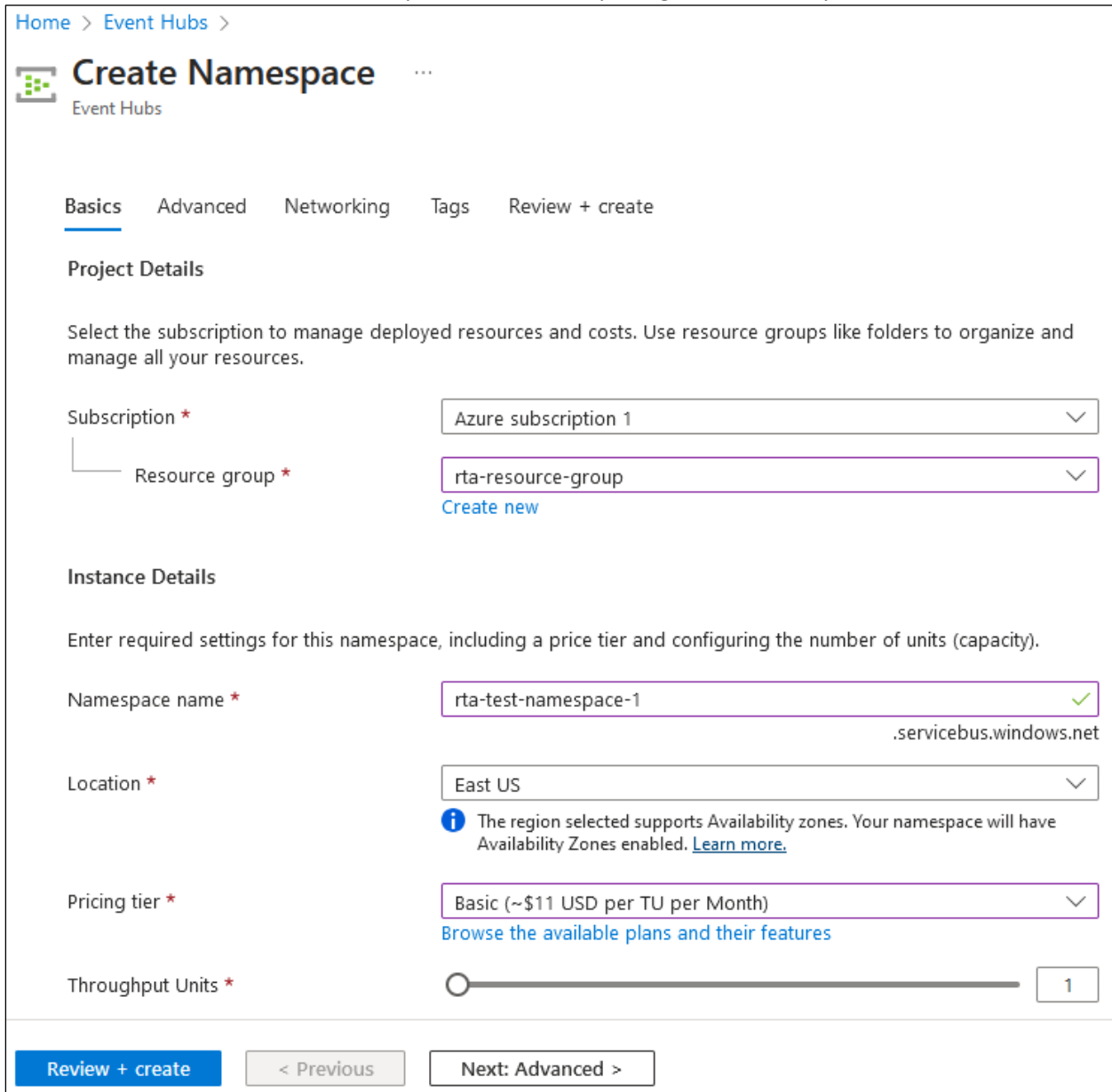
m. Copy the Primary connection string for the device, this will be used when connecting the RTA gateway to Azure.



The screenshot shows the Azure IoT Hub portal configuration for a device named 'rta-test-device'. The breadcrumb navigation is 'Home > IoT Hub > rta-test-hub | Devices >'. The device name 'rta-test-device' is displayed with a share icon and a menu icon. Below the name is the IoT Hub name 'rta-test-hub'. The main configuration area includes: 'Device ID' (rta-test-device), 'Primary key' (masked), 'Secondary key' (masked), 'Primary connection string' (masked), and 'Secondary connection string' (masked). There are 'Save', 'Manage keys', 'Message to Device', 'Direct method', 'Add Module Identity', 'Device twin', and 'Refresh' buttons. The 'Tags' section shows 'No tags'. The 'Enable connection to IoT Hub' section has 'Enable' selected. The 'Parent device' section shows 'No parent device'. Below the configuration are tabs for 'Module Identities' and 'Configurations'. A table header is visible with columns: 'Module ID', 'Connection State', 'Connection State Last Updated...', and 'Last Activity Time (UTC)'. The table content shows 'There are no module identities for this device.'

4. Create an Event Hub

- a. In the search bar at the top type in “Event Hubs” and select the Event Hubs under services to navigate to the Event Hubs page.
- b. Hit Add in the top left corner, there may also be a create Event Hub button in the middle if no Event Hubs Namespaces exist.
- c. In the Create Namespace window, select your previously created resource group, enter a name for the namespace, and select a pricing tier based on your needs.



Home > Event Hubs >

## Create Namespace

Event Hubs

Basics | Advanced | Networking | Tags | Review + create

### Project Details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \*

Resource group \*  [Create new](#)

### Instance Details

Enter required settings for this namespace, including a price tier and configuring the number of units (capacity).

Namespace name \*

Location \*    
*i* The region selected supports Availability zones. Your namespace will have Availability Zones enabled. [Learn more.](#)

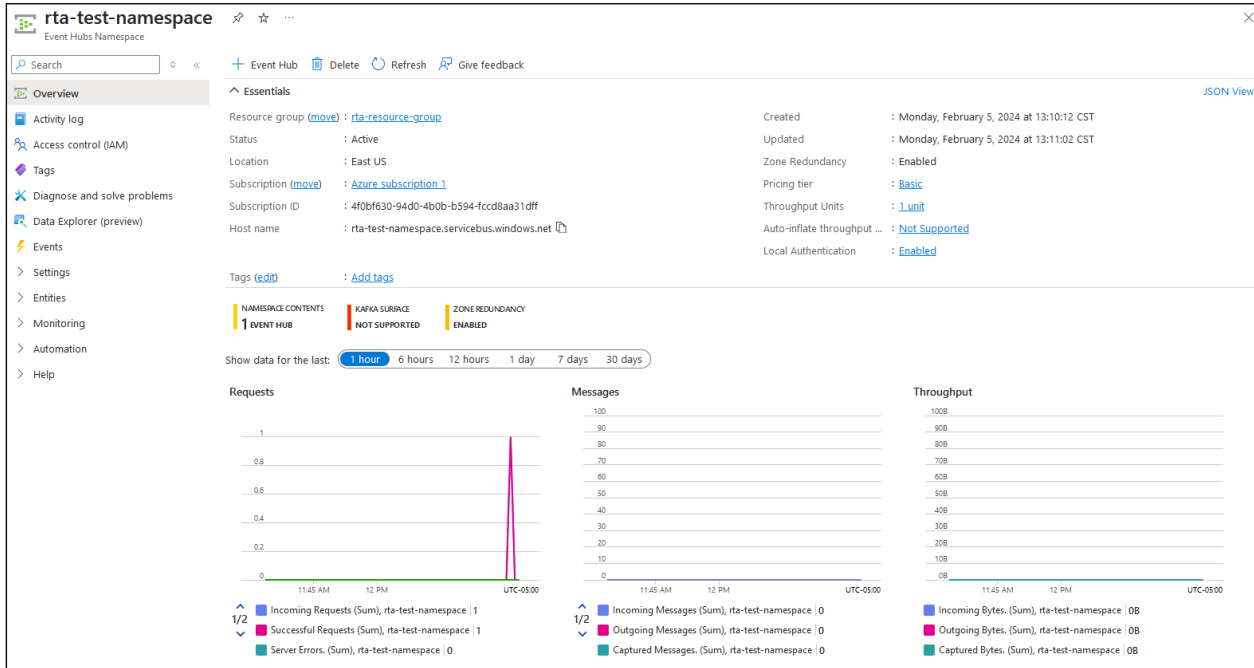
Pricing tier \*    
[Browse the available plans and their features](#)

Throughput Units \*

[Review + create](#) [< Previous](#) [Next: Advanced >](#)

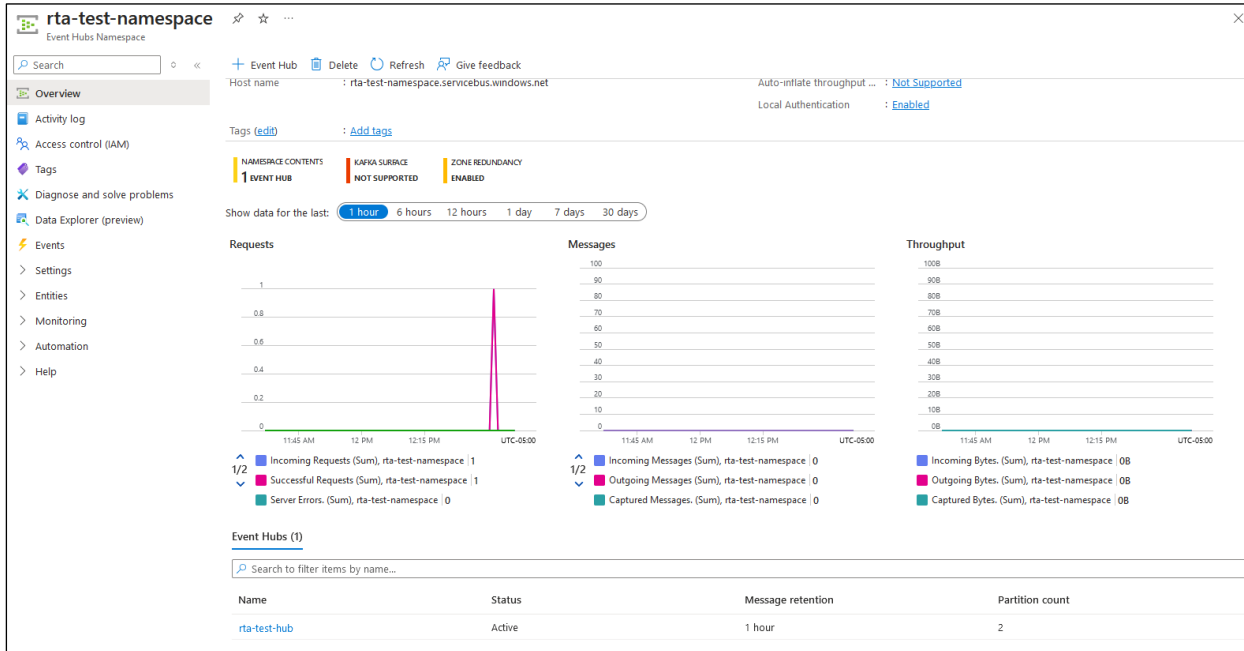
- d. Hit Review + Create in the bottom left corner of the page.
- e. Hit Create in the bottom left corner of the page to create the namespace.

- f. Once the namespace has finished initializing click on go to resource to navigate to the namespaces main panel. If go to resource is not an option navigate to Event Hubs and select the newly created namespace to open the main panel.



- g. Click Add Event Hub in the top left corner.  
 h. Enter in a name for the event hub  
 i. Click review + create in the bottom left corner.  
 j. On the review page hit create in the bottom left corner.

- k. You should be returned to the namespace overview, scroll down and you should see the newly created event hub at the bottom of the overview section.



## 5. Create an Event Grid System Topic

- In the search bar at the top type in "Event Grid System Topics" and select the event grid system topics option under services to navigate to the even grid system topics page..
- Click create in the top left corner to create a new event grid system topic.
- In the Create Even Grid System Topic window enter set the Topic Types to "Azure IoT Hub Accounts, Select your azure subscription, select your resource group, and enter a name for the topic

Home > Event Grid | System topics >

## Create Event Grid System Topic

Event Grid

Basics Tags Review + create

Topic Details

System topic resource is associated with an existing azure resource which allows customer to subscribe events emitted by that resource. System topic resource is created in the same subscription and resource group as the source.

Topic Types: Azure IoT Hub Accounts

Subscription: Azure subscription 1

Resource Group: rta-resource-group

Resource: rta-test-hub

System Topic Details

Enter required settings for this system topic.

Name: rta-test-topic

Location: eastus

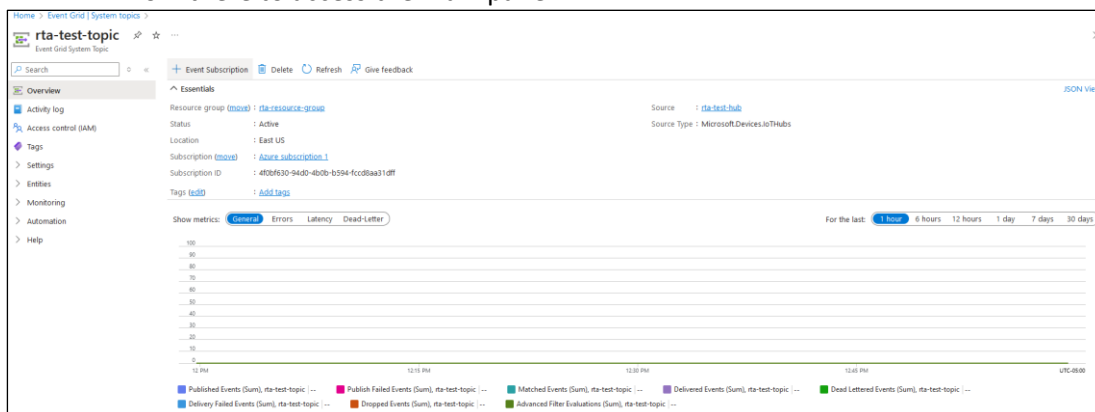
Identity

Managed identities are used to authenticate an Event Grid topic to Azure service instances when delivering events. [Learn more about Managed Identities](#)

System assigned identity

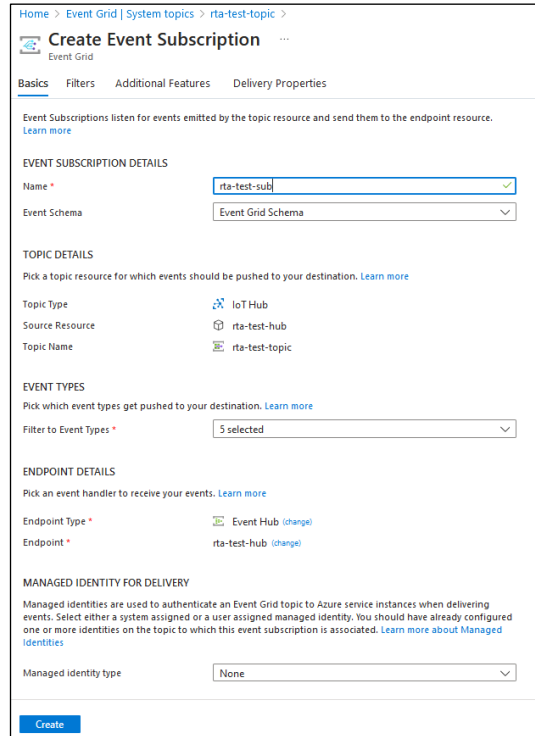
Review + create < Previous Next: Tags >

- d. Click Review + create in the bottom left corner.
- e. On the review + create page click create the create the Event Grid System Topic.
- f. Once the event grid system topic is initialized click go to resource to navigate to the Event Grid System Topic Panel. If go to resource is not an option navigate back to the even grid system topic section and select the newly created event grid system topic from there to access the main panel.



- g. In the your new event grid system topics main panel hit add event subscription in the top left corner.
- h. In the create Event Subscription window:
  - i. Enter a name for the subscription

- ii. Under event types, ensure that Device Created, Device Deleted, Device Connect, Device Disconnect, and Device Telemetry are all selected.
- iii. Under Endpoint Select Event Hub as the endpoint type.
- iv. Under endpoint details click “Configure an Endpoint”
- v. In the newly opened Select Event Hub window, select your azure subscription, resource group, namespace, and event hub and click confirm selection.



Home > Event Grid | System topics > rta-test-topic >

### Create Event Subscription

Event Grid

Basics Filters Additional Features Delivery Properties

Event Subscriptions listen for events emitted by the topic resource and send them to the endpoint resource. [Learn more](#)

**EVENT SUBSCRIPTION DETAILS**

Name \*

Event Schema

**TOPIC DETAILS**

Pick a topic resource for which events should be pushed to your destination. [Learn more](#)

Topic Type

Source Resource

Topic Name

**EVENT TYPES**

Pick which event types get pushed to your destination. [Learn more](#)

Filter to Event Types \*

**ENDPOINT DETAILS**

Pick an event handler to receive your events. [Learn more](#)

Endpoint Type \*

Endpoint \*

**MANAGED IDENTITY FOR DELIVERY**

Managed identities are used to authenticate an Event Grid topic to Azure service instances when delivering events. Select either a system assigned or a user assigned managed identity. You should have already configured one or more identities on the topic to which this event subscription is associated. [Learn more about Managed Identities](#)

Managed identity type

[Create](#)

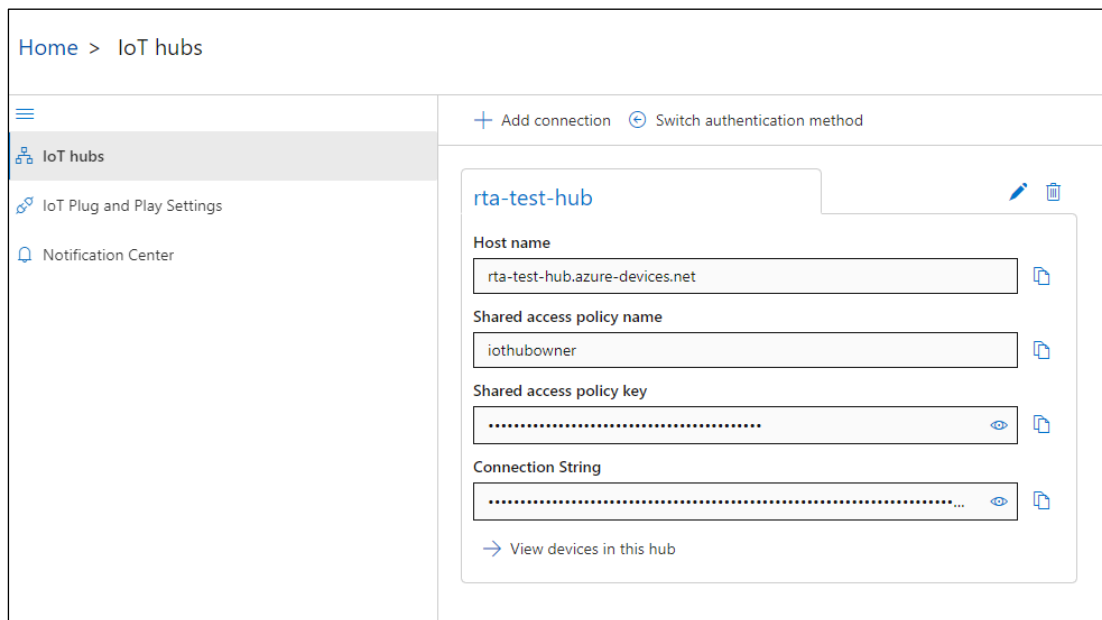
- i. Click create in the lower left corner of the webpage.

## Testing Microsoft Azure Communication

Once you have the Azure configured, you can use their [Azure IoT Explorer](#) utility to test the connection is working.

To set up Azure IoT Explorer:

1. open the utility and select the IoT Hubs tab on the left
2. Click the Add connection button
3. In the Add connection String window that appears enter the connection string for your IoT Hub.
  - a. This can be obtained by navigating to the IoT Hubs overview window in azure, selecting Shared access policies from the left pane, and opening the iothubowner policy.
4. Click save.
5. Your hub should now be listed in the IoT hubs section as shown below.



6. You can then click on the hub name to view devices available in the hub and see their current status.



Home > rta-test-hub > Devices

[+](#) New [↻](#) Refresh [🗑](#) Delete

Query by device ID...

Device ID <input type="text"/>	Status <input type="text"/>	Connectio... <input type="text"/>	Authentica... <input type="text"/>	Last status ... <input type="text"/>
<a href="#">mqtt-explorer-device</a>	Enabled	Disconnected	Sas	--
<a href="#">rta-test-device</a>	Enabled	Disconnected	Sas	--

7. Selecting a device within the hub will provide device specific information such as its Primary Connection string. It also provides useful tools for seeing data being from the device to Azure as well as send test messages from Azure to the device.

## Send data from Microsoft Azure to RTA gateway

Below is how the RTA Microsoft Azure Service is set up to receive data from Microsoft Azure.

<input checked="" type="checkbox"/> Enable	<b>Azure 1</b>		
Device Label	QT01	Network Interface	Switch Mode (DHCP Assigned) ▾
Primary Connection String	HostName=rta-test-hub.azure-devices.net;DeviceId=rta-test-device;SharedAcces		
Client ID	rta-test-device	<input type="checkbox"/> Add Timestamp to Publishes	
TCP Port	8883	1-65535 (Default: 8883)	Keep Alive
			60
			0-200 sec (0 to Disable)
# of JSON Name/Value Pairs			
		1	0-500
# of Publish Paths		0	0-250
# of Subscribe Paths		1	0-1
<input type="button" value="Generate Paths"/>			

**JSON Name/Value Pairs**

Line #	Path	JSON Name	JSON Point Type
1	devices/rta-test-device/devicebound/ ▾	message	INT (8-bit) ▾

<< 1-1 >>

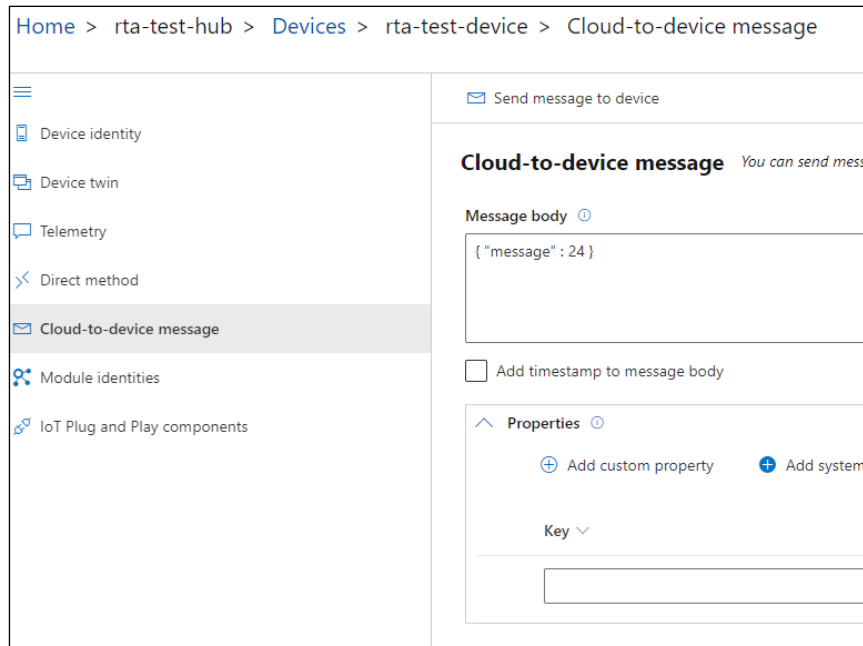
**Subscribe Paths (MQTT to 460ETCQT)**

Line #	Enable	Path Name	QoS
1	<input checked="" type="checkbox"/>	devices/rta-test-device/devicebound/ #/	0 ▾

<< 1-1 >>

In Azure IoT Explorer, navigate to the device you wish to receive data from Azure on. In the device configuration select the Cloud-to-device message tab on the left.

In the Cloud-to-device message tab, enter the message to be sent to the RTA in the message body section and click Send message to device at the top of the page. An example payload for the configuration above can be seen in the image below.



Navigate to the RTA display data page, view data from MQTT, refresh the page, and the value should be visible as shown below.

MQTT		460ETCQT		PLC	
Name	Value (Hex)	Manipulation	Name	Value (Hex)	
message	24 0x18	↔↔	N/A	Point Not Mapped	

Send data from RTA gateway to Microsoft Azure (Publish Topics)

This example shows a PLC writing data to the RTA gateway and presenting that data to the Publish topic on Microsoft Azure

Enable
 

**Azure 1**

Device Label

Network Interface

Primary Connection String

Client ID

Add Timestamp to Publishes

TCP Port  1-65535 (Default: 8883)

Keep Alive  0-200 sec (0 to Disable)

# of JSON Name/Value Pairs  0-500

# of Publish Paths  0-250

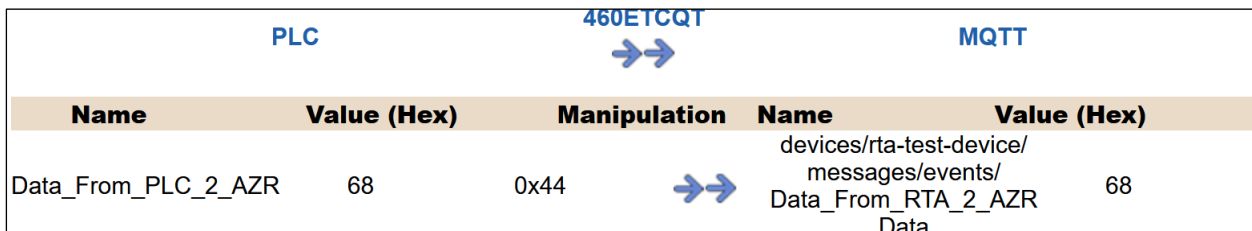
# of Subscribe Paths  0-1

### JSON Name/Value Pairs

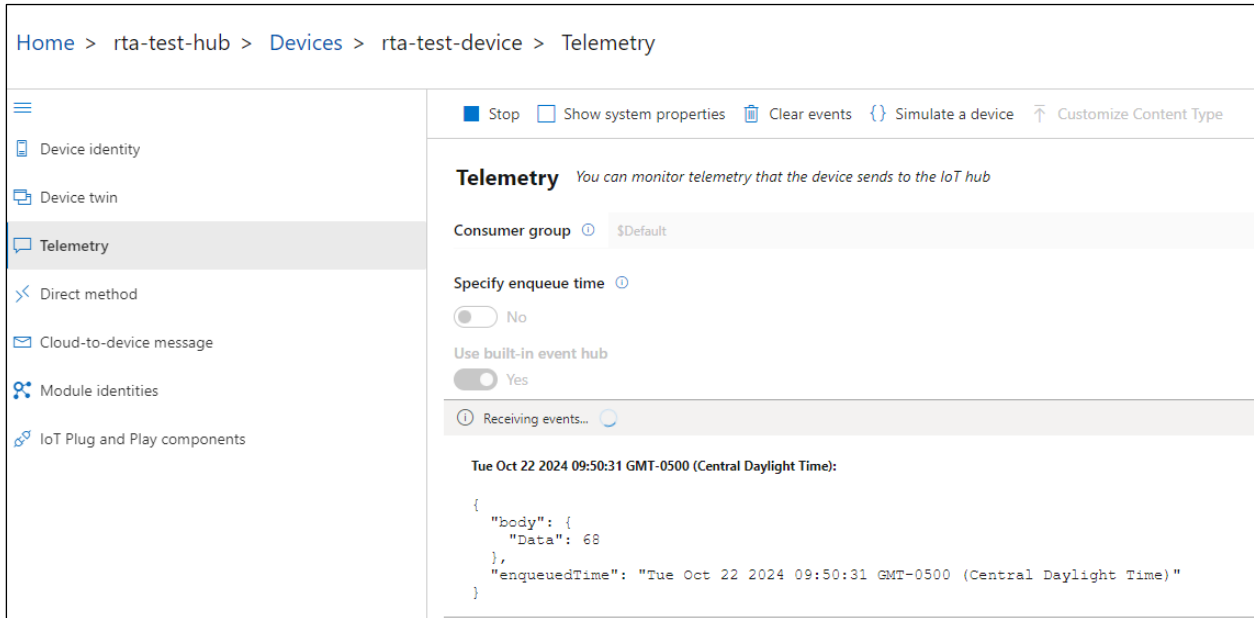
Line #	Path	JSON Name	JSON Point Type
1	<input type="text" value="devices/rta-test-device/messages/ev"/>	<input type="text" value="Data"/>	<input type="text" value="INT (8-bit)"/>

### Publish Paths (460ETCQT to MQTT)

Line #	Enable	Path Name	QoS
1	<input checked="" type="checkbox"/>	<input type="text" value="devices/rta-test-device/messages/events/Data_From_RTA_2_AZR"/>	<input type="text" value="0"/>



In Azure IoT Explorer, navigate to the device you would like to publish data to and select the telemetry tab on the left. In the telemetry window click Start in the top left corner. Any messages published to the selected device will now be shown in the telemetry window.



The screenshot shows the Azure IoT Explorer interface for a device named 'rta-test-device'. The left sidebar contains navigation options: Device identity, Device twin, Telemetry (selected), Direct method, Cloud-to-device message, Module identities, and IoT Plug and Play components. The main area displays the Telemetry tab with the following controls and data:


- Buttons: Stop, Show system properties, Clear events, Simulate a device, Customize Content Type
- Section: **Telemetry** *You can monitor telemetry that the device sends to the IoT hub*
- Consumer group: \$Default
- Specify enqueue time: No (toggle)
- Use built-in event hub: Yes (toggle)
- Status: Receiving events...
- Timestamp: Tue Oct 22 2024 09:50:31 GMT-0500 (Central Daylight Time):
- JSON payload:

```
{  "body": {    "Data": 68  },  "enqueuedTime": "Tue Oct 22 2024 09:50:31 GMT-0500 (Central Daylight Time)"}
```


## QT Publish Trigger

By default, the RTA gateway will publish to the broker based on change of state. This means a new publish to the broker will occur any time a value changes. In an application where the client is being charged per publish, such as with Azure, this is not ideal. In this situation the QT Publish trigger can be configured. When the QT Publish Trigger is configured a publish will only occur when the trigger value is incremented, allowing the user to control when data is published.

In the example below, it is shown how the trigger would be configured for MQTT device 1. A trigger value is being mapped from the mating protocol into the QT TriggerPublish1 destination. With this configuration all topics configured on MQTT device 1 will be published when the trigger value is incremented.

Mapping 2		
Source	<input type="checkbox"/> Enable Manipulation	Destination
Group: ETC01 Trigger_Value (Int16) ▾ Start: Trigger_Value ▾ End: Trigger_Value ▾		Group: QT TriggerPublish1 (UInt16) ▾ Start: TriggerPublish1 ▾ End: TriggerPublish1

To trigger for MQTT device 2 or 3, TriggerPublish2 or TriggerPublish3 should be selected under the “Start” dropdown in the destination configuration as shown below.

Mapping 2		
Source	<input type="checkbox"/> Enable Manipulation	Destination
Group: ETC01 Trigger_Value (Int16) ▾ Start: Trigger_Value ▾ End: Trigger_Value ▾		Group: QT TriggerPublish1 (UInt16) ▾ Start: TriggerPublish1 ▾ End: <ul style="list-style-type: none"> <li>TriggerPublish1</li> <li>TriggerPublish2</li> <li>TriggerPublish3</li> </ul>

## 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.

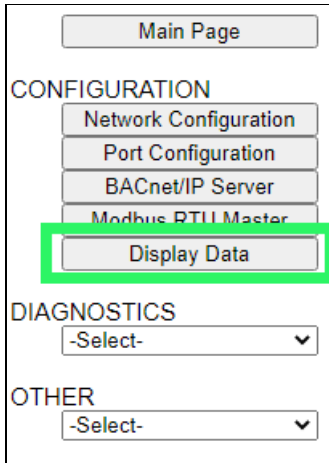
**Going from Manual Mapping to Auto-Mapping will delete ALL mappings and manipulations configured.**

## 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.



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.

Modbus RTU			BACnet/IP		
Name	Value (Hex)	Manipulation	Name	Value (Hex)	
400001	-- --	→→	AI1	-- --	
400002	-- --	→→	AI2	Mapping Disabled for Point	
400003	-- --	→→	AI3	-- --	

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.

### Display Data Edit Mapping View as Text

Select a Device Modbus TCP Server IP Address: 10.1.16.16 View

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

<< 1 >>  
 Displaying 1-7 of 7

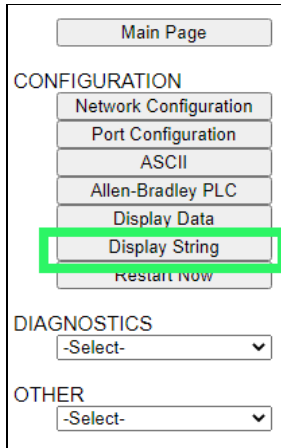
Modbus TCP/IP			460ETCMC ↔↔	PLC		
Name	Value (Hex)		Manipulation	Name	Value (Hex)	
400001	15	0x000F	↔↔	ETC01 MC2PLC_INT[0]	15	0x000F
400002	1495	0x05D7	↔↔	ETC01 MC2PLC_INT[1]	1495	0x05D7
400003	1	0x0001	↔↔	ETC01 MC2PLC_INT[2]	1	0x0001
400004	23	0x0017	↔↔	ETC01 MC2PLC_INT[3]	23	0x0017
400005	3	0x0003	↔↔	ETC01 MC2PLC_INT[4]	3	0x0003
400011	--	--	↔↔	ETC01 ETC01_G2N0_INT[0]	--	--
400012	--	--	↔↔	ETC01 ETC01_G2N0_INT[1]	--	--

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.



Main Page

CONFIGURATION

- Network Configuration
- Port Configuration
- ASCII
- Allen-Bradley PLC
- Display Data
- Display String**
- Restart Now

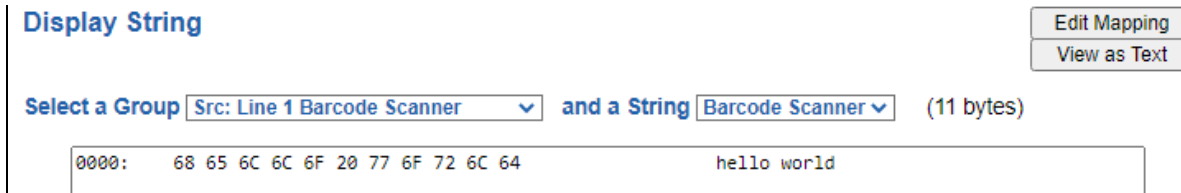
DIAGNOSTICS

-Select-

OTHER

-Select-

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.

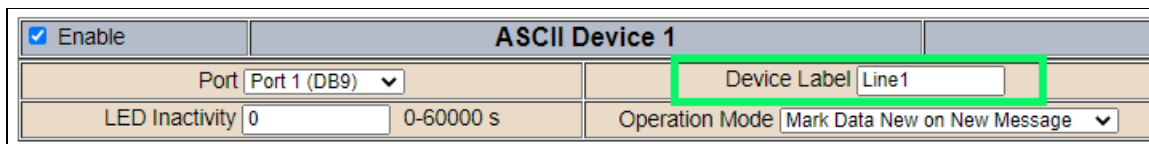


**Display String** Edit Mapping  
View as Text

Select a Group **Src: Line 1 Barcode Scanner** and a String **Barcode Scanner** (11 bytes)

0000: 68 65 6C 6C 6F 20 77 6F 72 6C 64      hello world

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



Enable      **ASCII Device 1**

Port **Port 1 (DB9)**      Device Label **Line1**

LED Inactivity **0**      0-60000 s      Operation Mode **Mark Data New on New Message**

Field	Start Location	Length	Data Type	Internal Tag Name
1:	1	0	String	<b>Barcode Scanner</b>

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.

**Display String** Edit Mapping  
View as Text

Select a Group Src: Line 1 Barcode Scanner and a String Barcode Scanner (0 bytes)

Data Not Valid

**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.

**Diagnostics**

ASCII View

Port 1 (DB9) View

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

**Display String** Edit Mapping  
View as Text

Select a Group Src: Line 1 Barcode Scanner and a String Barcode Scanner (11 bytes)

0000: 68 65 6C 6C 6F 20 77 6F 72 6C 64      hello world

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

**String Mapping Configuration** Help

Manual Configure ▼

# of Mappings to Configure: 1 0-250

Set Max # of Mappings

<< 1 >>

Source	Mapping 1	Destination
Group: <span style="border: 1px solid gray; padding: 2px;">Line 1 Barcode Scanner</span> String: <span style="border: 1px solid gray; padding: 2px;">Barcode Scanner</span>		Group: <span style="border: 1px solid gray; padding: 2px;">ETC01 ETC01_G2N0_STRIN</span> String: <span style="border: 1px solid gray; padding: 2px;">ETC01_G2N0_STRING</span>

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.

ASCII Device 1 (Line1)				
Max Number of Fields:	3	1-50	Min Number of Fields:	1
Parsing Delimiter:		, 44 0x2c		
Update Fields				
Field	Start Location	Length	Data Type	Internal Tag Name
1:	1	0	String	Header 1
2:	1	0	String	Header 2
3:	1	0	String	Header 3

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

**Display String** Edit Mapping  
View as Text

Select a Group  and a String  (3 bytes)

0000: 52 54 41 RTA

**Display String** Edit Mapping  
View as Text

Select a Group  and a String  (7 bytes)

0000: 53 75 70 70 6F 72 74 Support

**Display String** Edit Mapping  
View as Text

Select a Group  and a String  (5 bytes)

0000: 52 6F 63 68 73 Rocks

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.

**Diagnostics** Last Message Sent (17 bytes)

0000: 52 54 41 2C 53 75 70 70 6F 72 74 2C 52 6F 63 68 RTA,Support,Rock  
0016: 73 s

## 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.

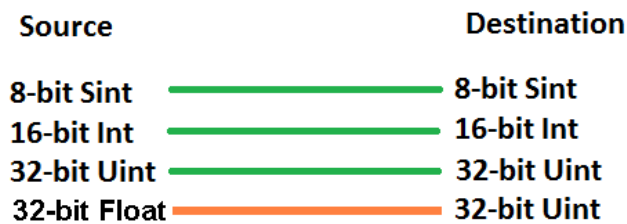
The following are examples of the Auto-Configure function.

- 1) This example shows a common valid setup.



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

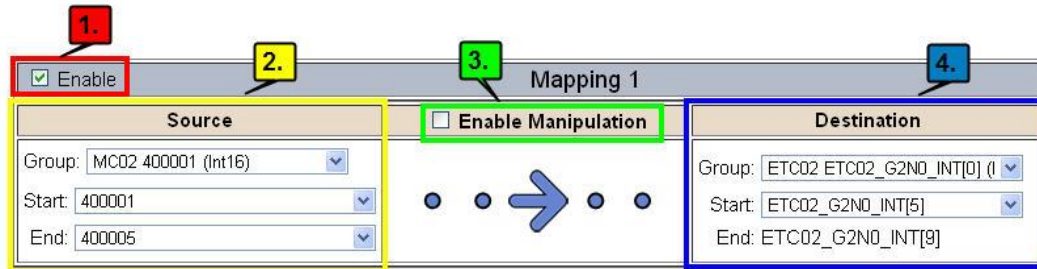
- 2) This example shows how Auto-Configure will make its best guess.



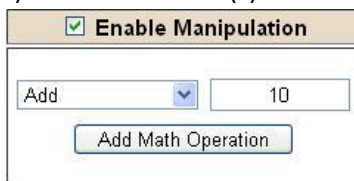
- a. The 32-bit Float from the Source location could not find a matching Destination data-type. After all other like data types were mapped, the only data type available was the 2<sup>nd</sup> 32-bit Uint data type. Auto-Configure was completed even though the data in the Float will be truncated.

## Data Mapping – Explanation

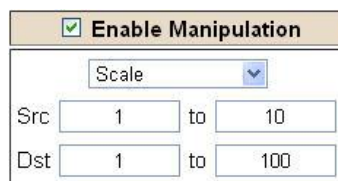
Below are the different parts that can be modified to make up a data mapping.



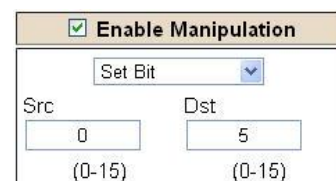
- 1) Enable (red box above): Check to enable mapping. If not checked, this mapping is skipped.
- 2) Source Field (yellow box above):
  - a) Group - Select the data group you set up in the protocol config to use for this mapping.
  - b) Start - This is the starting point for this mapping.
  - c) End - 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 0<sup>th</sup> source bit and copy it into the value of the 5<sup>th</sup> 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 necessary.

### 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.

<input checked="" type="checkbox"/> Enable Mapping 1		
Source	<input checked="" type="checkbox"/> Enable Manipulation	Destination
Group: Temporary Ram0 (Int64) <span style="float: right;">▼</span>	Scale <span style="float: right;">▼</span>	Group: Temporary Ram0 (Int64) <span style="float: right;">▼</span>
Start: Ram0 <span style="float: right;">▼</span>	Src: 1 to 10	Start: Ram1 <span style="float: right;">▼</span>
End: Ram0 <span style="float: right;">▼</span>	Dst: 1 to 100	End: Ram1
<input checked="" type="checkbox"/> Enable Mapping 2		
Source	<input checked="" type="checkbox"/> Enable Manipulation	Destination
Group: Temporary Ram0 (Int64) <span style="float: right;">▼</span>	Add <span style="float: right;">▼</span> 5	Group: Temporary Ram0 (Int64) <span style="float: right;">▼</span>
Start: Ram1 <span style="float: right;">▼</span>	<input type="button" value="Add Math Operation"/>	Start: Ram2 <span style="float: right;">▼</span>
End: Ram1 <span style="float: right;">▼</span>		End: Ram2

*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.

<input checked="" type="checkbox"/> Enable Mapping 1		
Source	<input type="checkbox"/> Enable Manipulation	Destination
Group: Ticks Since Powerup (UInt32) <span style="float: right;">▼</span>		Group: BS01 AI1 (Float) <span style="float: right;">▼</span>
Start: Since Powerup <span style="float: right;">▼</span>		Start: AI1 <span style="float: right;">▼</span>
End: Since Powerup <span style="float: right;">▼</span>		End: AI1




#### 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.

Mapping 1		
Source	<input type="checkbox"/> Enable Manipulation	Destination
Group: Heartbeat 100ms Update (Uir) ▾ Start: 100ms Update ▾ End: 100ms Update ▾		Group: ETC01 Heartbeat (Int32) ▾ Start: Heartbeat ▾ End: Heartbeat

#### 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.

Mapping 1		
Source	<input type="checkbox"/> Enable Manipulation	Destination
Group: Heartbeat 1000ms Update (U) ▾ Start: 1000ms Update ▾ End: 1000ms Update ▾		Group: ETC01 Heartbeat (Int32) ▾ Start: Heartbeat ▾ End: Heartbeat

#### 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>Modbus RTU Master</b>	
<b>Device Status</b>	
Connected and Running	
<b>LED Status</b>	
Connection Status:	Connected
<b>Variables</b>	
<b>Network Bitmap Status:</b>	0x0000001f

- 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.

Mapping 1		
Source	Enable Manipulation	Destination
<input checked="" type="checkbox"/> Enable Group: MM NetBmpStat (Uint32) Start: NetBmpStat End: NetBmpStat	<input type="checkbox"/> Enable Manipulation 	Group: BS01 AI1 (Float) Start: AI1 End: AI1

**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.

Mapping 1		
Source	Enable Manipulation	Destination
<input checked="" type="checkbox"/> Enable Group: MM NetBmpStat (Uint32) Start: NetBmpStat End: NetBmpStat	<input checked="" type="checkbox"/> Enable Manipulation Set Bit Src: 1 (0-31) Dst: 0 (0)	Group: BS01 BI1 (Bit1) Start: BI1 End: BI1

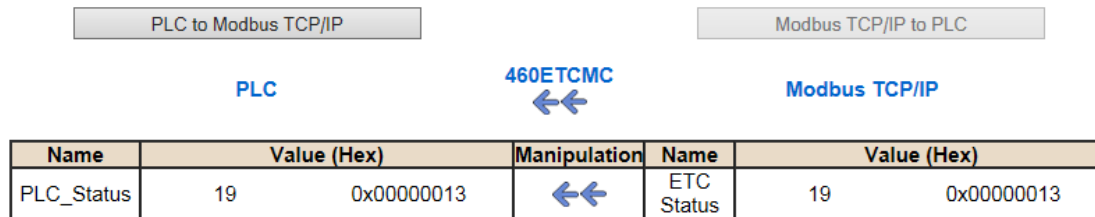
## 7) Status\_XY

- 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) 1<sup>st</sup> 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



**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) 2<sup>nd</sup> 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

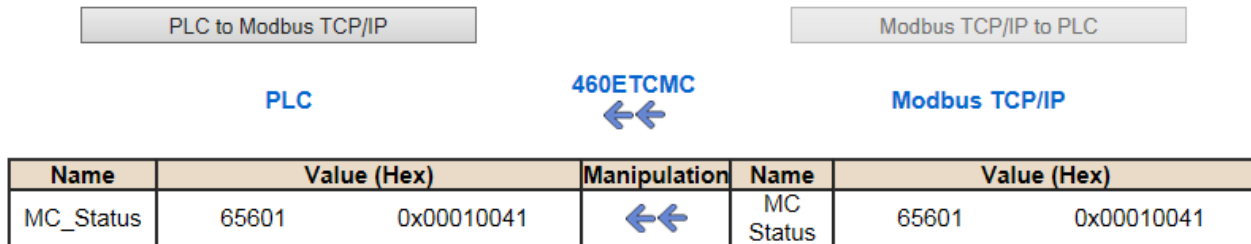
**Recoverable Faults: 0x00FF0000 (bit 16-23) 3<sup>rd</sup> byte**

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) 4<sup>th</sup> 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



**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:**

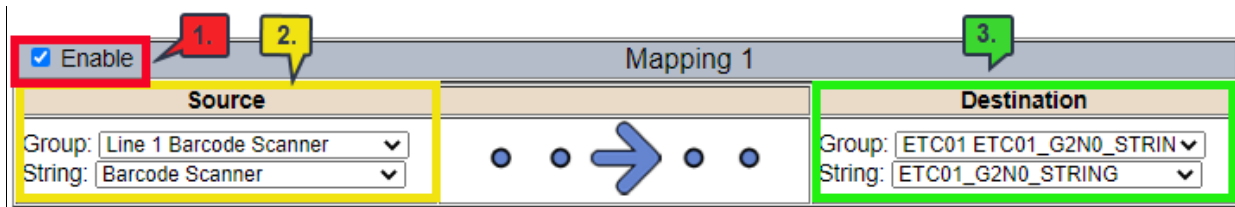
Hex:	Bit:	Decimal:	Explanation:
0x01	16	65,536	recoverable fault - timed

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.



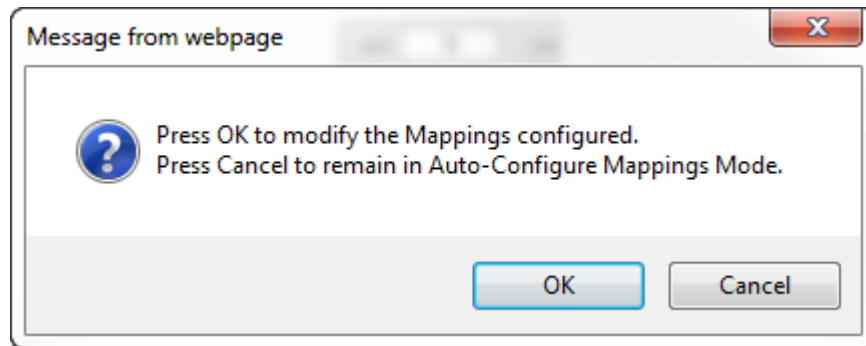
Mapping 1	
<input checked="" type="checkbox"/> Enable	
<b>Source</b>	<b>Destination</b>
Group: Line 1 Barcode Scanner	Group: ETC01 ETC01_G2N0_STRIN
String: Barcode Scanner	String: ETC01_G2N0_STRING

- 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

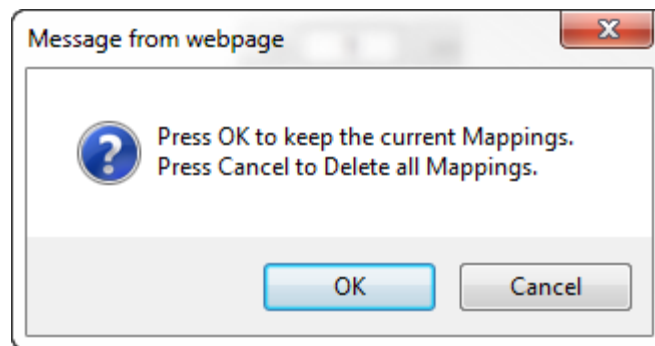
To transition from Auto-Configure Mapping Mode to Manual Configure Mode, click the dropdown at the top of the Mapping Configuration page and select Manual Configure.

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.

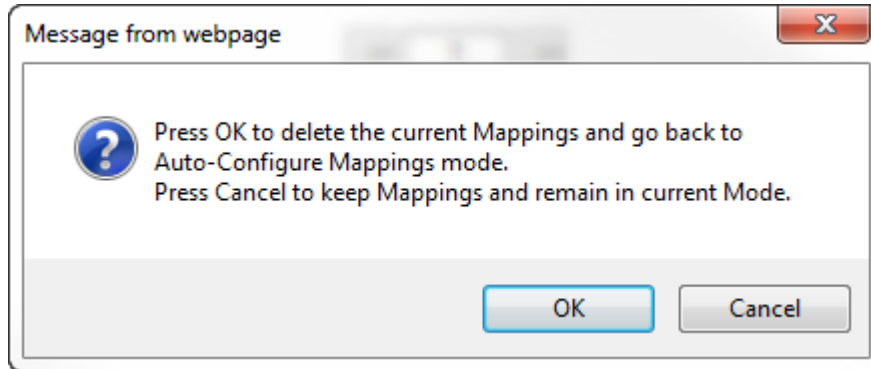


- 1) To keep the mappings that are already configured press **OK**.
  - a) You would want this option if you are adding additional mappings or you want to modify the mapping(s) that already exist.
- 2) To delete the mappings that are already there and start over press **Cancel**.

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.

```


### Data Mapping


|            |                |        |                        |    |                |
|------------|----------------|--------|------------------------|----|----------------|
| Mapping 1: | Temporary Ram0 | Len: 1 | -> 1:10 Scale to 1:100 | -> | Temporary Ram1 |
| Mapping 2: | Temporary Ram1 | Len: 1 | -> Add 5               | -> | Temporary Ram2 |


```

### 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.

```


### String Mapping


|            |                   |         |                |
|------------|-------------------|---------|----------------|
| Mapping 1: | Logix Test_String | -> Copy | -> MC02 400001 |
|------------|-------------------|---------|----------------|


```



## Base Triggering – Data Validation 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.

#### Write Data Groups (BACnet/IP to 460MCBS)

Data Group	Object Type	Starting Object	# of Objects
1	Analog Output (32 Bit Float)	1	21
2	Binary Output	1	0
3	CharacterString Value	51	0

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

Setup BACnet Names, Units, and COV


21	G01	Data Validation Trigger	Other	no-units	1.000000
----	-----	-------------------------	-------	----------	----------

- 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.

Mapping 2		
Source	<input type="checkbox"/> Enable Manipulation	Destination
Group: BS01 AO1 (Float) Start: AO21 End: AO21		Group: MC Trigger 0 (Uint16) Start: Trigger 1 End: Trigger 1

- 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.

Mapping 3		
Source	<input type="checkbox"/> Enable Manipulation	Destination
Group: MC Handshake 0 (Uint16) Start: Handshake 1 End: Handshake 1		Group: BS01 AI1 (Float) Start: AI40 End: AI40

## 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 event a password is forgotten.

**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** Help

Log Out Timer:  0-15 min

**Admin Configuration**

Admin	Username	Password	Re-enter Password	Email	Hint
1	<input type="text"/>	<input type="text"/>	<input type="text"/>	Not Configured	<input type="text"/>
2	<input type="text"/>	<input type="text"/>	<input type="text"/>	Not Configured	<input type="text"/>
3	<input type="text"/>	<input type="text"/>	<input type="text"/>	Not Configured	<input type="text"/>

**Admin Contact Information**

**User Configuration**

User	Username	Password	Re-enter Password	Email	Hint
1	<input type="text"/>	<input type="text"/>	<input type="text"/>	Not Configured	<input type="text"/>
2	<input type="text"/>	<input type="text"/>	<input type="text"/>	Not Configured	<input type="text"/>
3	<input type="text"/>	<input type="text"/>	<input type="text"/>	Not Configured	<input type="text"/>
4	<input type="text"/>	<input type="text"/>	<input type="text"/>	Not Configured	<input type="text"/>
5	<input type="text"/>	<input type="text"/>	<input type="text"/>	Not Configured	<input type="text"/>

## 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.

User 1: <input type="button" value="View"/>	
Web Page	Security
All Web Pages	No Access <input type="button" value="Set"/>
Web Page	Security
Main Page	Full Access <input type="button" value="v"/>
Device Configuration	Full Access <input type="button" value="v"/>
Port Configuration	Full Access <input type="button" value="v"/>
BACnet/IP Server	Full Access <input type="button" value="v"/>
Modbus RTU Master	Full Access <input type="button" value="v"/>
View Mapping	Full Access <input type="button" value="v"/>
Mapping	Full Access <input type="button" value="v"/>
Setup LED's	Full Access <input type="button" value="v"/>
Diagnostic Info	Full Access <input type="button" value="v"/>
Logging	Full Access <input type="button" value="v"/>
Display Data	Full Access <input type="button" value="v"/>
Export Configuration	Full Access <input type="button" value="v"/>
Import Configuration	Full Access <input type="button" value="v"/>
Save As Template	Full Access <input type="button" value="v"/>
Load From Template	Full Access <input type="button" value="v"/>
Utilities	Full Access <input type="button" value="v"/>
Email Configuration	Full Access <input type="button" value="v"/>
Alarm Configuration	Full Access <input type="button" value="v"/>
String Mapping	Full Access <input type="button" value="v"/>
View String Mapping	Full Access <input type="button" value="v"/>
Display String	Full Access <input type="button" value="v"/>

## Security - Log In

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

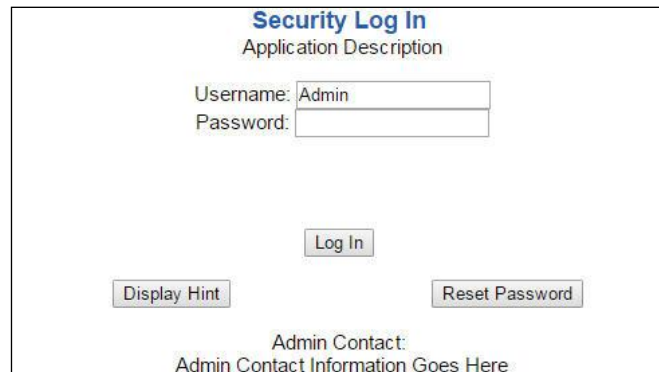
**Password:** Password of the user to login.

**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.



The screenshot shows a web interface titled "Security Log In" with the subtitle "Application Description". It contains a form with two input fields: "Username:" with the value "Admin" and "Password:". Below the form are three buttons: "Log In", "Display Hint", and "Reset Password". At the bottom, there is a label "Admin Contact:" followed by the text "Admin Contact Information Goes Here".

## 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.



The banner features the RTA logo on the left, the text "Welcome Admin [logout](#)" in the center, and the website URL "www.rtaautomation.com" on the right. A blue bar at the bottom contains "Real Time Automation, Inc." on the left and "MODE: RUNNING 460" on the right.

**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.

**Email Configuration**

Number of Emails to Configure:  0-10

User	SMTP Mail Username	SMTP Mail Password	SMTP Server	From Email	To Email	Email Group
1	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	Group A ▼

## 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.

**Alarm Configuration**
Help

Alarm Delay upon Powerup:  0-3600 s

# of Alarms to Configure:  0-100

<input checked="" type="checkbox"/> Enable	Alarm 1			
Data Point	Set Error	Clear Error	Alarm Name	Email
<input type="checkbox"/> Ticks Since Powerup (Uint32) <input type="checkbox"/> Ticks Since Powerup	<input type="text" value="&gt;="/> <input type="text" value="1000"/>	<input type="text" value="None"/> <input type="text" value="0"/>	<input type="text" value="Gateway_test"/>	<input type="text" value="Group A"/>

- 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.

**Alarm Status**

# Alarms Enabled: 1  
 # Alarms Active: 0  
 Last Active Alarm:

---

Alarm#	Name	Status	# of Times Active
1	Alarm Example	OK	0

## 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.


www.rtaautomation.com

Real Time Automation, Inc.
Alarms Active
MODE: RUNNING

460

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.

**Alarm Status**

# Alarms Enabled: 1  
 # Alarms Active: 1  
 Last Active Alarm: Alarm 1 is Set. Actual: 0 < Limit: 20

---

Alarm#	Name	Status	# of Times Active
1	Alarm Example	Alarm	1

### 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 performance.**

- 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.

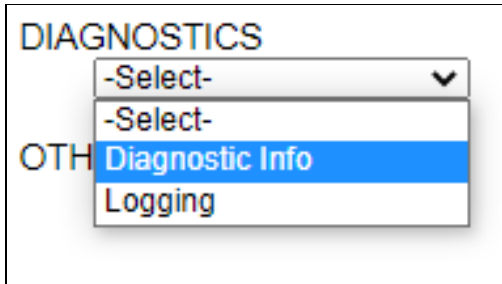
**Change of State Configuration**
Help

Stale Data Timer:	0	0-3600 s
Production Inhibit Timer:	0	0-60000 ms
Writes Before Reads:	10	0-255
Reads Before Writes:	1	1-255
Enable Data Integrity:	<input checked="" type="checkbox"/>	
Enable Mark Whole Entry New:	<input type="checkbox"/>	

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 – ASCII

Select ASCII in the top dropdown menu on the Diagnostics Page to view a breakdown of the diagnostics that are displayed on the page. You may also view individual ASCII device counters and messages by selecting the device in the *All ASCII* dropdown and clicking **View**. Additional diagnostic information can be found by clicking the **Help** button.

### Diagnostics

ASCII View

All ASCII View

**Device Status**  
Connected and Running

Clear All Values

Help

Clear Buffers

**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.

Example: If viewing ASCII – Port #, this will only clear the values for Port #. This will reduce the *All ASCII* values indirectly.

Variables	
Network Bitmap Status:	0x00000001
Successful Transmit Count:	4
Successful Receive Count:	1
Received due to Length:	0
Received due to Delimiters:	0
Received due to Timeout:	1
Received but Discarded:	0
Successful Parsed Messages:	1
Failed Parsed Messages:	0
Status Strings	
Queued Messages:	See Device Level
Last Parsed Error:	

**Clear Buffers** - This will clear the Next Message to Send from Queue buffer and Current Message being received from ASCII buffer and any message stored in the Queue.

**Next Message to Send from Queue (17 bytes)**

```
0000: 52 54 41 2C 53 75 70 70 6F 72 74 2C 52 6F 63 6B RTA,Support,Rock
0016: 73 s
```

---

**Current Message being Received from ASCII (17 bytes)**

```
0000: 52 54 41 2C 53 75 70 70 6F 72 74 2C 52 6F 63 6B RTA,Support,Rock
0016: 73 s
```

**Device Status** - This will only display when viewing *All ASCII*.

- 1) Connected and Running– The gateway is connected to all the ASCII devices and data is being received/transmitted.
- 2) Not Connected – There have been no messages received or transmitted.
  - a. Verify that the serial /TCP/IP/USB settings match your device.
- 3) Fatal Error: Hardware Port Not Configured – The port selected on the ASCII Configuration page is not configured.

### Diagnostics

ASCII ▼ View

All ASCII ▼ View

**Device Status**

Fatal Error: Couldn't Open Hardware Port

**LED Status**

Connection Status: No Serial Port Configured

- a. Verify the ASCII device is enabled and configured.
- b. Verify the port configured matches the port enabled.

<input checked="" type="checkbox"/> Enable	<b>ASCII Device 1</b>		
Port <span style="border: 1px solid gray; padding: 2px;">-Select-</span> <span style="font-size: 0.8em;">▼</span>	Device Label <span style="border: 1px solid gray; padding: 2px;">Line1</span>		
LED Inactivity <span style="border: 1px solid gray; padding: 2px;">0</span>	<span style="border: 1px solid gray; padding: 2px;">0-60000 s</span>	Operation Mode <span style="border: 1px solid gray; padding: 2px;">Mark Data New on New Message</span> <span style="font-size: 0.8em;">▼</span>	

**LED Status** - This is the Status for *All ASCII* or the specific ASCII device selected.

- 1) Solid Green (Connected) – The gateway is receiving/transmitting data within the inactivity period for all the ASCII devices that are configured and enabled.
- 2) Flashing Green (Not Connected/First Time Scan) – Start up state. No messages have been received or transmitted, but port is connected.
- 3) Flashing Red (Connection Timeout) – The only way to exit this state is with a valid received message.
  - a. Data has been discarded due to the queue being full.
  - b. Data has not been received/transmitted within the inactivity period.
  - c. Port not opened.
  - d. Message parsing has failed.

### Diagnostics

ASCII ▼ View
Clear All Values

All ASCII ▼ View

**Device Status**

Connected and Running

**LED Status**

Connection Status: Connected

Help
Clear Buffers

**Variables** - These are the values for *All ASCII*, or the ASCII device selected.

- 1) Successful Transmit Count:
  - a) Number of messages that the gateway has transmitted to the ASCII device
- 2) Successful Receive Count:
  - a) Number of complete messages that the gateway has received from the ASCII device
- 3) Received due to Length:
  - a) Number of messages completed due to the Max Message Length being reached
- 4) Received due to Delimiters:
  - a) Number of messages completed due to the Start or End Delimiters being seen
- 5) Received due to Timeout:
  - a) Number of messages completed due to the Receive Character Timeout being reached
- 6) Received but Discarded:
  - a) Number of messages that are complete but discarded due to the queue being full
  - b) Change the Gateway Hold Msg Timeout to be less than what you currently have set
- 7) Successful Parsed Messages:
  - a) Number of messages that are complete and have been successfully parsed
- 8) Failed Parsed Messages:
  - a) Number of messages that are complete but have not been parsed successfully

Variables	
Network Bitmap Status:	0x00000001
Successful Transmit Count:	0
Successful Receive Count:	1
Received due to Length:	1
Received due to Delimiters:	0
Received due to Timeout:	0
Received but Discarded:	0
Successful Parsed Messages:	1
Failed Parsed Messages:	0
Status Strings	
Queued Messages:	
Last Parsed Error:	

**Status Strings** - These are the values for *All ASCII*, or the ASCII device selected.

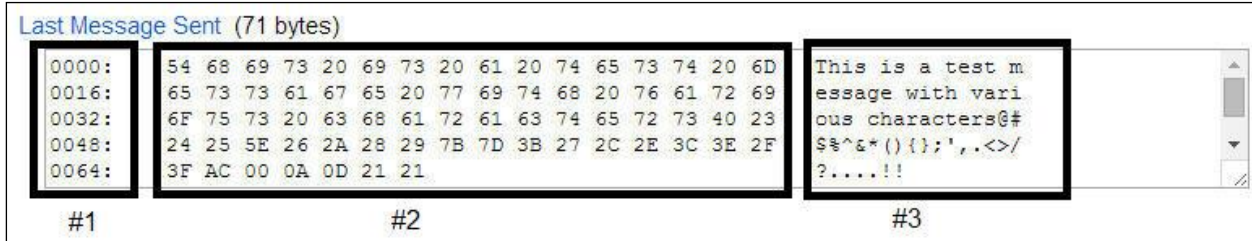
- 1) Queued Messages:
  - a) The gateway will hold up to 20 (configurable) complete messages to send to the other protocol
  - b) This will only increment if the Gateway Hold Msg Timeout is non-zero and messages are being received faster than we can send to the other protocol
- 2) Last Parsed Error:
  - a) Last parsed error the gateway encountered

**Common Error Messages:**

- 1) **Number of Fields Invalid:** The total number of parsed fields is greater than the number of fields the gateway was expecting
- 2) **Discard:** The Field has been discarded
- 3) **Invalid Length for Field:** Number of characters parsed is greater than the number of characters that the gateway is expecting

- 4) **Calculated Length of Data exceeds 255 Characters:** Number of characters parsed within a field exceeds 255 characters

### Buffers

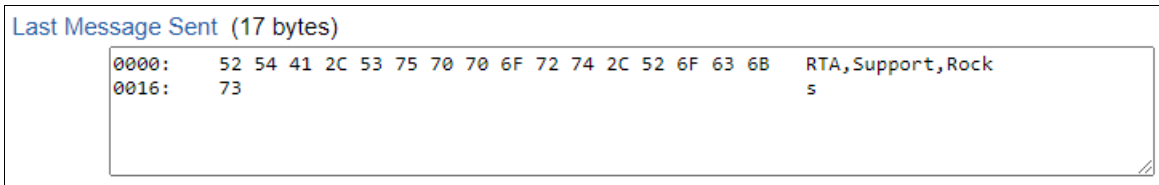


Each buffer text area is divided into three separate parts. Refer to screenshot above for labels.

- 1) Starting byte for that line
- 2) HEX character representation
- 3) ASCII character representation (Unprintable ASCII characters (like <CR>) will be displayed as '.' on the right-hand side of the buffer area.)

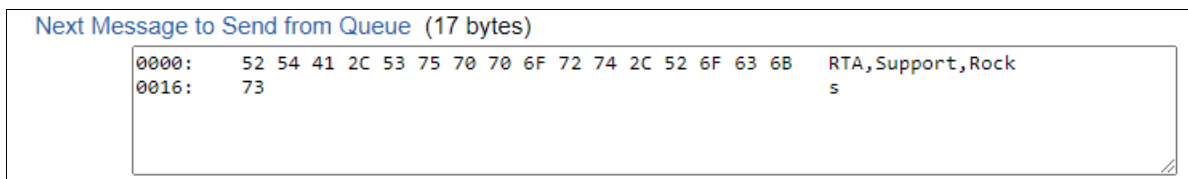
### Last Message Sent:

- 1) Last complete message the gateway received from the ASCII device, processed, and sent to the other protocol



### Next Message to Send from Queue:

- 2) This is the next complete message that the gateway has already received and processed from the ASCII device, and will be sent next to the other protocol



### Current Message being Received from ASCII:

- 1) The current data that the gateway is receiving. The data needs to hit one of the three end cases to call a message complete:
  - a) Max Message Length (max number of characters to receive)
  - b) Receive Character Timeout (max time to wait after a character to call the current message complete)
  - c) Delimiters (gateway has not received the delimiters that are specified)



Current Message being Received from ASCII (17 bytes)

```
0000: 52 54 41 2C 53 75 70 70 6F 72 74 2C 52 6F 63 6B RTA,Support,Rock
0016: 73 s
```

Last Message Sent to ASCII:

- 1) Last message that the gateway sent to the ASCII device

**Note:** The concatenated delimiters are not displayed in this message but will be transmitted with the message

Last Message Sent to ASCII (18 bytes)

```
0000: 52 54 41 20 53 75 70 70 6F 72 74 20 52 6F 63 6B RTA Support Rock
0016: 73 21 s!
```

Send Data from Gateway to ASCII: (Used for testing only, Character limit of 1024):

- 1) Enter a message to send to your ASCII device
- 2) Can be used to test communication and test formatting of messages

Send Data from Gateway to ASCII (Used for Testing Only)

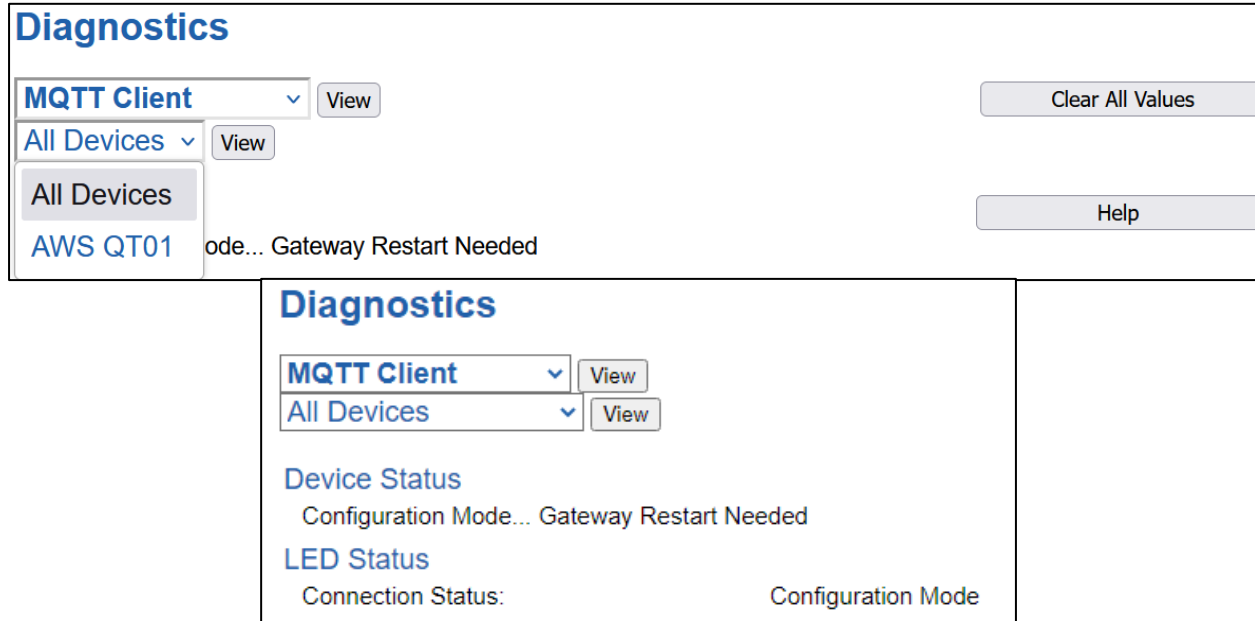
Enter Test Message Here

Send ASCII Message

## Diagnostic – MQTT Client

Select the MQTT Client in the dropdown menu on the Diagnostic page to view the breakdown of the diagnostics and common strings that are display on the page. You may also view the individual MQTT device counters by selecting the device in the All Devices drop down and clicking **View**.

**NOTE:** This page will auto-refresh every five seconds.



**Diagnostics**

MQTT Client View Clear All Values

All Devices View Help

All Devices

AWS QT01 ode... Gateway Restart Needed

---

**Diagnostics**

MQTT Client View

All Devices View

**Device Status**  
Configuration Mode... Gateway Restart Needed

**LED Status**  
Connection Status: Configuration Mode

**Clear All Values:** This will only affect the Variables values.

- This will return all values displayed to zero.

**Device Status:** This will only display when viewing All Devices.

**Connected:** All Devices configured/enabled are communicating

**Not Connected:** Fatal Error No Configuration

- No Devices that have been configured are enabled
- No Devices that have been configured and enabled have topics configured

**Not Connected:** Dependency Protocol is Faulted

- The Dependency Protocol has Faulted

**Error:** Timeout

- One or more enabled devices are missing
- Verify MQTT broker for correct IP address.

**LED Status**

**Solid Green (Connected):**

- The gateway is connected to all the MQTT devices that are configured and enabled

**Flashing Green (Not Connected):**

- No MQTT devices are configured / enabled. Go to the MQTT Client Device Configuration to configure a device

**Flashing Red (Not Connected):**

- One or more of the MQTT brokers configured are missing (nodes missing)
- One or more of the MQTT brokers configured do not have topics configured
- The Dependency Protocol has faulted

**Flashing Red (Communication not attempted yet):**

- No topics are configured and data needed for writes isn't valid yet

**Solid Red (Invalid Configuration):**

- No devices are enabled
- One or more of the MQTT devices have a conflicted IP address

**Off:**

- No Power
- No Ethernet cable connected

**Variables:** These are the values for all servers, or the specific server selected.

Variables	
Network Bitmap Status:	0x00000000
Published Messages to MQTT:	0
Published Messages from MQTT:	0
Subscribed Messages Actual:	0
Subscribed Messages Expected:	0

**Network Bitmap Status (Displayed in Hex):**

- Each bit corresponds to a MQTT device. If the bit is set, the MQTT device is connected, otherwise the bit is 0.
- Bit 0 corresponds to MQTT device 1 and Bit 2 is for MQTT device 3 and so on.

**Published Messages to MQTT:**

- Number of Write Topics which have been sent to the MQTT broker

**Published Messages from MQTT:**

- Number of Read Topics which have been sent from the MQTT broker to the gateway

**Subscribed Messages Actual:**

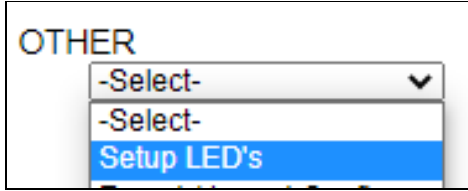
- Number of Successful Subscribed Topics
- This should equal the Subscribed Messages Expected

**Subscribed Messages Expected:**

- Number of Subscribed Topics that the gateway should have open

## LED Configuration

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

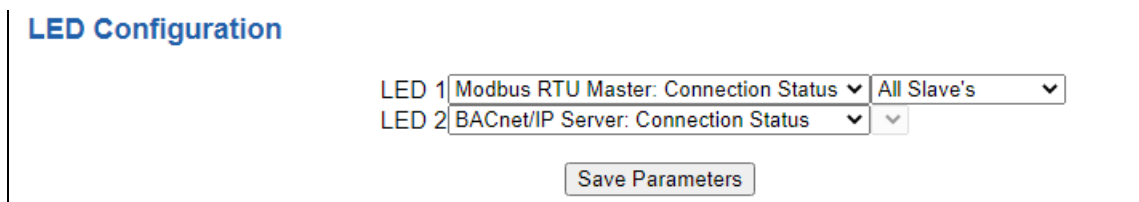


Each LED may be set to Disabled, Protocol 1, or Protocol 2. If either protocol is a master/client, you may set the LED to represent either all slaves/servers configured in the gateway or a slave/server device.

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.

A screenshot of the 'LED Configuration' page. It features two rows of dropdown menus. The first row is labeled 'LED 1' and has two dropdowns: 'Modbus RTU Master: Connection Status' and 'All Slave's'. The second row is labeled 'LED 2' and has two dropdowns: 'BACnet/IP Server: Connection Status' and an empty dropdown. Below the dropdowns is a 'Save Parameters' button.

**LED Configuration**

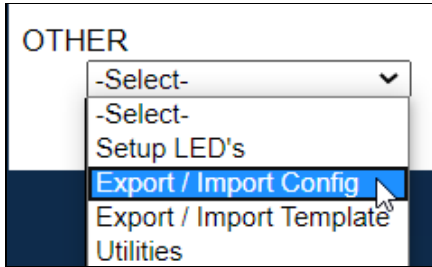
LED 1 | Modbus RTU Master: Connection Status | All Slave's

LED 2 | BACnet/IP Server: Connection Status |

Save Parameters

## 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.

**Network Settings have changed. Manually enter IP Address of X.X.X.X in the URL.**

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.

### Import Configuration

No file chosen

Import Network Settings

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.

## Intelligent Reset Button

If the IP Address of the gateway is forgotten or is unknown, there is an easy way to recover the IP Address using a reset button on the hardware.

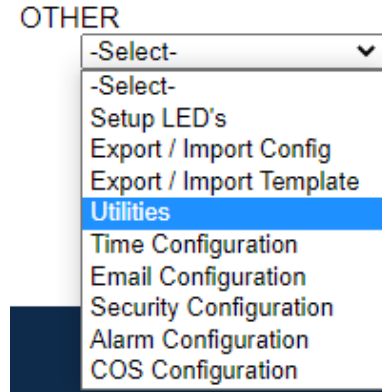


- 1) On the side of the gateway with the SD card slot, there is a small pinhole. Using a paperclip, press the button through this pinhole and hold the button for at least 5 seconds.
- 2) After 5 seconds, the unit will acknowledge the command and LED 1 and LED 2 will start an alternate Blink Green quickly pattern.
- 3) Release the button and the gateway will reset to default IP settings (DHCP).



## 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.

Revisions

Listing of Revisions

File List

File List

Identify Device

Start Flashing LED's

Set Up Log

Log Configuration

Revert To Last Powerup

Revert to Last Powerup

Revert All

Revert to Manufacturing Defaults

Reformat Flash

Reformat Flash