

# Onyxx<sup>®</sup> Modbus to BACnet<sup>®</sup> Bridge USER GUIDE



Onyxx-MB311

# Table of Contents

INTRODUCTION .....	3
FIRST TIME USERS .....	3
USER INTERFACE CONVENTIONS.....	4
Buttons .....	4
Field Editors.....	4
Table Rows.....	4
USER ACCESS.....	5
User Types .....	5
Login.....	5
Managing User Access.....	6
Adding a User.....	6
Editing a User .....	6
Editing the Default Admin User .....	6
Deleting a User .....	6
TCP/IP SETTINGS .....	7
Bridge TCP/IP Setup .....	7
Factory Default TCP/IP Settings.....	8
MODBUS RTU NETWORK SETTINGS.....	9
Network Setup.....	9
Serial Port Config.....	11
Poll Scheduler.....	11
MODBUS TCP NETWORK SETTINGS.....	12
Network Setup.....	12
Poll Scheduler.....	14
BACNET IP/ETHERNET NETWORK SETTINGS.....	15
Network Setup.....	15
BACnet Alarming .....	16
DEVICE SETUP SETTINGS.....	17
Advanced Setup Options.....	17
Allow Changes to the Device Byte Orders.....	18
Allow Changes to Modbus Address Format.....	19
Allow Changes to the BACnet Object Instance Number .....	20
Allow Changes to the BACnet Object Name.....	20

- MODBUS RTU DEVICE MANAGEMENT .....21**
  - Adding Devices to the Network ..... 21
    - Importing a New Device ..... 21
    - Manually Adding a New Device..... 22
    - Offline Status Indicator ..... 22
  - Managing Devices Added to the Network..... 23
    - Mass Updates..... 23
    - Reordering Devices..... 23
    - Duplicating a Device..... 24
    - Exporting a Device for Use on Another Bridge ..... 24
- MODBUS TCP DEVICE MANAGEMENT .....25**
  - Adding Devices to the Network ..... 25
    - Importing a New Device ..... 25
    - Manually Adding a New Device..... 26
    - Offline Status Indicator ..... 26
  - Managing Devices Added to the Network..... 27
    - Mass Updates..... 27
    - Reordering Devices..... 27
    - Duplicating a Device..... 28
    - Exporting a Device for Use on Another Bridge ..... 28
- MODBUS POINT MANAGEMENT .....29**
  - Adding Points to Devices..... 29
    - Mass Updates..... 29
    - Register Settings..... 29
    - BACnet Config ..... 30
    - BACnet Facet Setup ..... 31
  - Managing Points Added to Modbus Slave Devices..... 33
    - Reordering Devices..... 33
    - Duplicating a Point ..... 33
- BACNET EXPORT TABLE.....34**
  - Managing the BACnet Export Table..... 34
    - Export the BACnet Export Table ..... 34
- POINT LIMITATIONS.....35**
- RESOURCE LIMITATIONS.....36**
- LYNXSPRING COPYRIGHT AND TRADEMARKS .....37**

## INTRODUCTION

The Onyx<sup>®</sup> Modbus to BACnet<sup>®</sup> Bridge provides protocol translation between various building automation system (BAS) networks. It is built on top of the Onyx Platform with a customized and simplified user interface which is accessible through a standard web browser.

## FIRST TIME USERS

The web application that runs inside the bridge is divided into three main sections:

- ✓ **Device Management** – Provides set-up and configuration of the devices connected to the bridge.
- ✓ **Device Settings** – Provides configuration of the network settings required by the bridge.
- ✓ **Help** – Provides documentation on the features and limitations of the bridge.

Follow these steps to configure the bridge:

**Note:** All Device Settings pages require the **SAVE** button be clicked for settings to take effect.

1. Configure the TCP/IP settings according to the TCP network(s) that will be attached to the bridge.  
*(if required, reconfigure your PC settings to work with the new settings after saving)*

**The bridge will require a reboot if settings are changed.**

2. Configure the Modbus RTU network to meet your specific network requirements.
3. Configure the Modbus TCP network to meet your specific network requirements.
4. Configure the BACnet network to meet your specific network requirements.
5. Configure the Device Setup Settings if advanced Modbus device and/or point setup is required.
6. Add the required Modbus RTU devices and their associated points.
7. Add the required Modbus TCP devices and their associated points.
8. Export the BACnet Export Table to provide the BACnet integrator the points list.

Quick Help is provided in the navigation area when in the Device Management and Device Settings sections. Quick help provides definitions and instructions while setting up the bridge application.

# USER INTERFACE CONVENTIONS

Colors are used throughout the UI as indicators.

## Buttons



- enabled

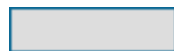


- disabled

## Field Editors

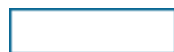


- enabled

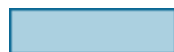


- disabled

## Table Rows



- default (not selected)



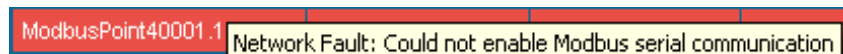
- selected



- error detected (*hover row for error message*)



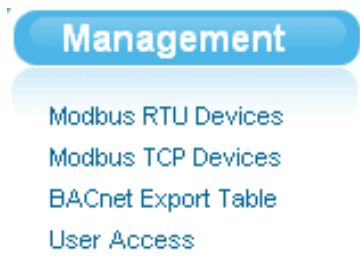
- selected/error detected (*hover row for error message*)



*Example (above) of an error message received while hovering over a row with an error detected.*

# USER ACCESS

From the **Home** page select USER ACCESS to manage user profiles with permissions to access the Onyx<sup>®</sup> Modbus to BACnet<sup>®</sup> Bridge. Always press the **SAVE** button to complete these settings.



## User Types

There are two types of users

1. Users with write capabilities (*the default admin user*)
2. Users with read only access (*the default basic user*)

**Note:** The default admin user cannot be deleted nor changed to a read only user.

**Note:** Users without **read only** checked must understand that they have full configuration and write capabilities within the bridge.

**Note:** Users with **read only** checked are very well suited for troubleshooting and accessing information.


## Login

To gain entry into DEVICE MANAGEMENT or DEVICE SETTINGS you must first log into the bridge.

A login form titled 'Please log in'. It contains two input fields: 'Username' and 'Password'. Below the fields is a blue 'Login' button.

Factory default usernames are **admin** and **basic**; both users have a default password of '**password**'. It is highly recommended that the password for the admin and basic user be set to a strong password and stored in a reliable location.

## Managing User Access

**User Access** 

Name	Write Access
admin	true
basic	false

[Add](#) [Edit](#) [Delete](#)

### Adding a User

As the default *admin* user or a user without read only limitations, you may add another user to the bridge. The following table lists the fields for adding a new user, only a Name is required.

**Adding New User**

Name	<input type="text"/>
Password	<input type="password"/>
Confirm	<input type="password"/>
Read Only	<input checked="" type="checkbox"/>

[Save](#) [Cancel](#)

- ✓ **Name:** User Name for logging into the Bridge (*case sensitive*)
- ✓ **Password:** Password used for logging into the Bridge (*case sensitive*)
- ✓ **Confirm:** Confirmation of the password used for logging into the Bridge (*case sensitive*)
- ✓ **Read Only:** Check this box if this user should NOT have write capabilities (*default is checked*)

You must click the **SAVE** button to retain changes.

[Save](#) [Cancel](#)

### Editing a User

As the default *admin* user or a user without read only limitations, you may edit another user's credentials. The fields available are the same as when Adding a User (*see above*).

A user with read only limitations can only edit their own credentials. The fields available are the same as when Adding a User (*see above*) with the exception of the read only box (*a read only user cannot give themselves write permissions*).

You must click the **SAVE** button to retain changes.

[Save](#) [Cancel](#)

### Editing the Default Admin User

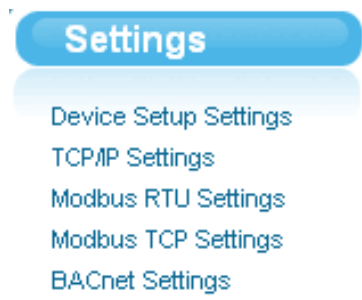
The credentials for the original default *admin* user can be changed by that user or another user without read only limitations. However; the default *admin* user is fixed into its position and the read only capability cannot be selected nor can the User Name be changed. This is true for the default *admin* user only.

### Deleting a User

A user may be deleted by any user operating without read only limitations. However, the default *admin* user cannot be deleted from the bridge.

# TCP/IP SETTINGS

From the **Home** page select TCP/IP SETTINGS to configure the communication settings used by the Bridge. Always press the **SAVE** button to complete these settings.

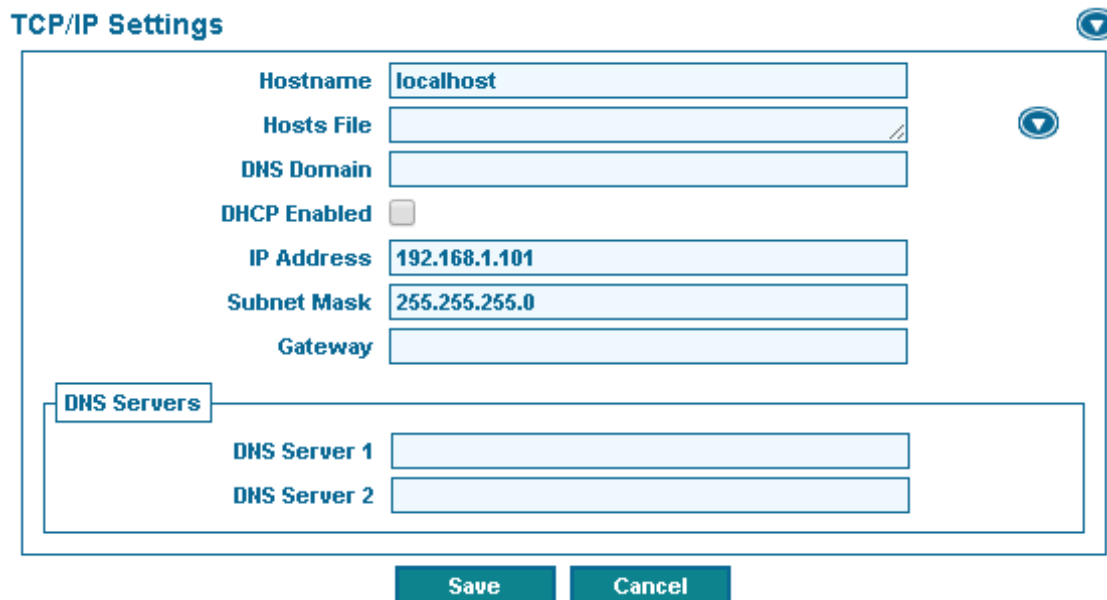


The Onyx<sup>®</sup> Modbus to BACnet Bridge provides a configurable Ethernet port providing multiple TCP/IP network configurations. Default settings are shown in the screen shot below. Please contact your network administrator for appropriate settings for your environment.

You must click the **SAVE** button to retain changes.



## Bridge TCP/IP Setup

A screenshot of the "TCP/IP Settings" configuration form. The form has a title bar "TCP/IP Settings" with a close button on the right. The fields are: "Hostname" (localhost), "Hosts File" (empty), "DNS Domain" (empty), "DHCP Enabled" (checkbox, unchecked), "IP Address" (192.168.1.101), "Subnet Mask" (255.255.255.0), and "Gateway" (empty). Below these is a section titled "DNS Servers" containing "DNS Server 1" and "DNS Server 2" (both empty). At the bottom are "Save" and "Cancel" buttons.

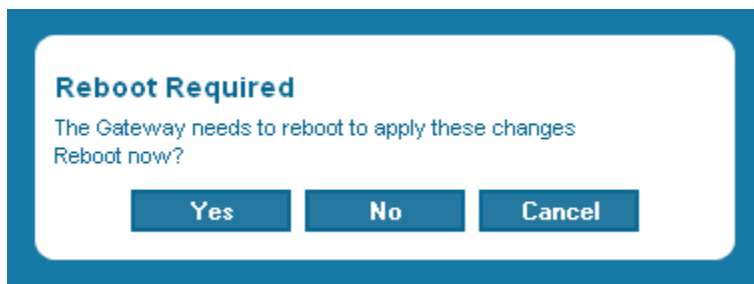


- ✓ **Hostname:** A unique name for the Bridge on the network
- ✓ **Hosts File:** Stores the information where to find a node on a computer network.  
This file maps hostnames to IP addresses
- ✓ **DNS Domain:** Domain this Bridge is associated with *(if any)*
- ✓ **Gateway:** Defines the gateway IP address to connect to an external LAN / WAN  
*(i.e. the World-Wide-Web)*
- ✓ **DNS Server 1:** Server to use for domain name resolution
- ✓ **DNS Server 2:** Backup server for domain name resolution
- ✓ **DHCP Enabled:** Disabled if this interface has a static IP address
- ✓ **IP Address:** The Internet Protocol Address for the selected interface on the Bridge
- ✓ **Subnet Mask:** Determines which subnet(s) the selected interface can communicate on

You must click the **SAVE** button to retain changes.



**Note:** Once changes have been saved, the Bridge will need to reboot for changes to take effect. The Bridge will redirect you to the Home page once the reboot is complete.



## Factory Default TCP/IP Settings

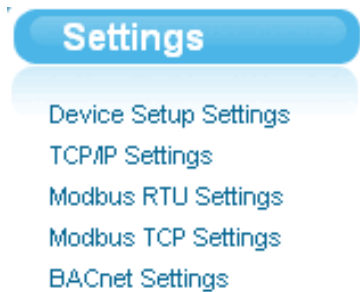
**Hostname:** localhost  
**Hosts File:** 127.0.0.1 localhost.localdomain localhost  
**IP Address:** 192.168.1.101  
**Subnet Mask:** 255.255.255.0

**Note:** All other fields are empty by default.

# MODBUS RTU NETWORK SETTINGS

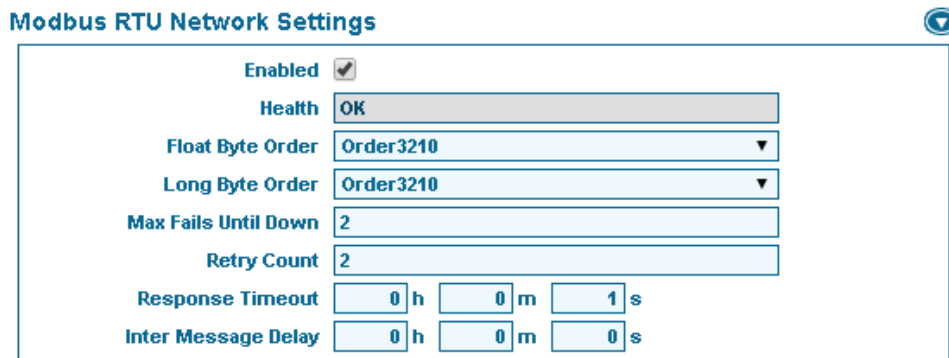
## Network Setup

From the **Home** page select MODBUS RTU SETTINGS to configure the Modbus RTU network. Because this is a multi-protocol bridge only focus on the Modbus RTU settings for this exercise. Always press the **SAVE** button to complete these settings.



In the Modbus RTU network you must set the serial port configuration to match the serial communications parameters used by the third party Modbus devices on the network. You can also configure network level protocol settings, such as byte orders, inter message delays, etc.

**Note:** When setting up large networks (*200 or more points*), it is highly recommended that the networks be disabled during setup. This will dramatically increase user interaction while adding devices and points. Re-enable the networks once setup is complete.



**Modbus RTU Network Settings**

Enabled

Health **OK**

Float Byte Order **Order3210**

Long Byte Order **Order3210**

Max Fails Until Down **2**

Retry Count **2**

Response Timeout **0** h **0** m **1** s

Inter Message Delay **0** h **0** m **0** s

### ENABLED

✓ Check this box to enable the Modbus RTU network.

You must click the **SAVE** button to retain changes.



The following network level properties, determine the “byte-order” that devices use for sending or receiving 4-byte (32-bit) data values in Modbus messages. Separate settings are for float (floating-point) and long (integer).

### *Float Byte Order*

Choices reflect two alternate methods, where numerals 0, 1, 2, and 3 represent the least significant byte to most significant byte, as follows:

- ✓ Order3210 – Most significant byte first, or “big-endian”, it is the default.
- ✓ Order1032 – Bytes transmitted in a 1,0,3,2 order, or “little-endian”.
- ✓ Order2301 – Swapped byte version of order 3210.
- ✓ Order0123 – Swapped byte version of order 1032.

### *Long Byte Order*

Choices reflect two alternate methods, where numerals 0, 1, 2, and 3 represent the least significant byte to most significant byte, as follows:

- ✓ Order3210 – Most significant byte first, or “big-endian”, it is the default.
- ✓ Order1032 – Bytes transmitted in a 1,0,3,2 order, or “little-endian”.
- ✓ Order2301 – Swapped byte version of order 3210.
- ✓ Order0123 – Swapped byte version of order 1032.

### *Max Fails Until Down*

Specifies the maximum number of communication failures (*i.e. failures from attempted polls/pings to the device*) that must occur before the device is considered failing (*down*). A value of zero means that the first communication failure for a device will cause the device to be considered failing (*down*). **Default value is 2.**

### *Retry Count*

Specifies the number of retries to perform after a null response to a basic message request. **Default value is 2.**

### *Response Timeout*

Specifies the maximum amount of time to wait for a response after a basic message request, prior to determining a failure. **Default is 1 second.**

### *Inter Message Delay*

Specifies the minimum amount of time to wait after a response message is received, prior to sending the next request message. **Default is 0 seconds.**

You must click the **SAVE** button to retain changes.



## Serial Port Config

For the Modbus RTU network, you must set the serial port configuration to match the serial communications parameters used by the Modbus devices on the network. Refer to third party device documentation to configure the network correctly.

**Serial Port Config**

Baud Rate	Baud9600
Data Bits	Data Bits8
Stop Bits	Stop Bit1
Parity	None

Set the properties for the JENESys™ Bridge serial port used, where defaults are:

- ✓ **Baud Rate:** Baud9600 – Or choose different from selection list.
- ✓ **Data Bits:** Data Bits8 – Or choose different from selection list.
- ✓ **Stop Bits:** Stop Bit1 – Or choose different from selection list.
- ✓ **Parity:** None – Or choose different from selection list.

You must click the **SAVE** button to retain changes.

**Save** **Cancel**

## Poll Scheduler

The Poll Scheduler handles the polling and pinging of devices on the network. The *Poll Rate* determines how often the points on a device are polled. The *Ping Frequency* determines how often a device is pinged if polling has been stopped due to non-communication. The network ping will only “ping” the device if the time of last verification is older than the ping frequency. Therefore, in normal operation with most devices, the point polling mechanism actually alleviates the need for the monitor ping, providing that the ping frequency is long enough. Also, in most devices if a point poll request receives no response (*not even a “null”*) from a device, a “ping fail” condition is immediately noted, without waiting for the monitor ping interval.

While receiving any response from the device, including an exception response, this is considered proof of communication, and the Modbus device is no longer considered “down” if it had been previously marked “down”.

**Poll Scheduler**

Poll Rate	0	h	0	m	15	s
Ping Frequency	0	h	30	m	0	s

### *Poll Rate*

Specifies the interval between periodic polls of all devices. **Default value is 15 seconds.**

### *Ping Frequency*

Specifies the interval between periodic pings of all devices. **Default value is 30 minutes.**

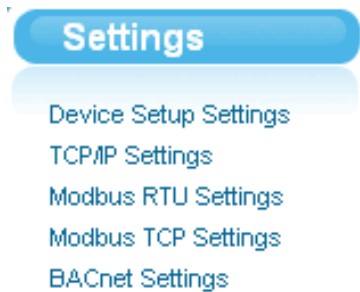
You must click the **SAVE** button to retain changes.

**Save** **Cancel**

# MODBUS TCP NETWORK SETTINGS

## Network Setup

From the **Home** page select MODBUS TCP SETTINGS to configure the Modbus TCP network. Because this is a multi-protocol bridge only focus on the Modbus TCP settings for this exercise. Always press the **SAVE** button to complete these settings.



In the Modbus TCP network you can configure network level protocol settings, such as byte orders, response timeouts, etc. used to communicate with the third party Modbus devices on the network.

**Note:** When setting up large networks (*200 or more points*), it is highly recommended that the networks be disabled during setup. This will dramatically increase user interaction while adding devices and points. Re-enable the networks once setup is complete.

A screenshot of the 'Modbus TCP Network Settings' configuration form. The form is titled 'Modbus TCP Network Settings' and has a dropdown arrow in the top right corner. The settings are as follows: 'Enabled' is checked; 'Health' is 'OK'; 'Float Byte Order' is 'Order3210'; 'Long Byte Order' is 'Order3210'; 'Max Fails Until Down' is '2'; 'Retry Count' is '2'; and 'Response Timeout' is set to '0 h', '0 m', and '1 s'.

### ENABLED

✓ Check this box to enable the Modbus TCP network.

You must click the **SAVE** button to retain changes.



The following network level properties, determine the “byte-order” that devices use for sending or receiving 4-byte (32-bit) data values in Modbus messages. Separate settings are for float (floating-point) and long (integer).

### *Float Byte Order*

Choices reflect two alternate methods, where numerals 0, 1, 2, and 3 represent the least significant byte to most significant byte, as follows:

- ✓ Order3210 – Most significant byte first, or “big-endian”, it is the default.
- ✓ Order1032 – Bytes transmitted in a 1,0,3,2 order, or “little-endian”.
- ✓ Order2301 – Swapped byte version of order 3210.
- ✓ Order0123 – Swapped byte version of order 1032.

### *Long Byte Order*

Choices reflect two alternate methods, where numerals 0, 1, 2, and 3 represent the least significant byte to most significant byte, as follows:

- ✓ Order3210 – Most significant byte first, or “big-endian”, it is the default.
- ✓ Order1032 – Bytes transmitted in a 1,0,3,2 order, or “little-endian”.
- ✓ Order2301 – Swapped byte version of order 3210.
- ✓ Order0123 – Swapped byte version of order 1032.

### *Max Fails Until Down*

Specifies the maximum number of communication failures (*i.e. failures from attempted polls/pings to the device*) that must occur before the device is considered failing (*down*). A value of zero means that the first communication failure for a device will cause the device to be considered failing (*down*). **Default value is 2.**

### *Retry Count*

Specifies the number of retries to perform after a null response to a basic message request. **Default value is 2.**

### *Response Timeout*

Specifies the maximum amount of time to wait for a response after a basic message request, prior to determining a failure. **Default is 1 second.**

You must click the **SAVE** button to retain changes.



## Poll Scheduler

The *Poll Scheduler* handles the polling and pinging of devices on the network. The *Poll Rate* determines how often the points on a device are polled. The *Ping Frequency* determines how often a device is pinged if polling has been stopped due to non-communication. The network ping will only “ping” the device if the time of last verification is older than the ping frequency. Therefore, in normal operation with most devices, the point polling mechanism actually alleviates the need for the monitor ping, providing that the ping frequency is long enough. Also, in most devices if a point poll request receives no response (*not even a “null”*) from a device, a “ping fail” condition is immediately noted, without waiting for the monitor ping interval.

While receiving any response from the device, including an exception response, this is considered proof of communication, and the Modbus device is no longer considered “down” if it had been previously marked “down”.

Poll Scheduler						
Poll Rate	<input type="text" value="0"/>	h	<input type="text" value="0"/>	m	<input type="text" value="15"/>	s
Ping Frequency	<input type="text" value="0"/>	h	<input type="text" value="30"/>	m	<input type="text" value="0"/>	s

### *Poll Rate*

Specifies the interval between periodic polls of all devices. **Default value is 15 seconds.**

### *Ping Frequency*

Specifies the interval between periodic pings of all devices. **Default value is 30 minutes.**

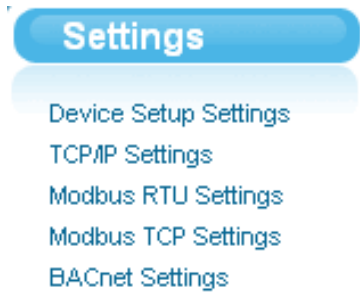
You must click the **SAVE** button to retain changes.

<input type="button" value="Save"/>	<input type="button" value="Cancel"/>
-------------------------------------	---------------------------------------

# BACNET IP/ETHERNET NETWORK SETTINGS

## Network Setup

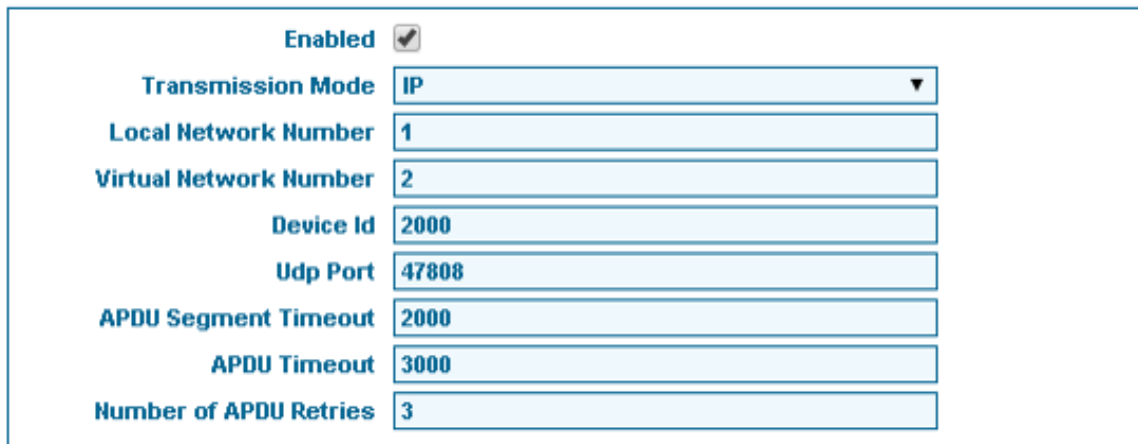
From the **Home** page select BACNET SETTINGS to configure the BACnet network. Because this is a multi-protocol bridge only focus on the BACnet settings for this exercise. Always press the **SAVE** button to complete these settings.



In the BACnet network you can configure network level protocol settings, such as Device Id, Udp Port, etc. used to communicate with the third party BACnet device(s) on the network.

**Note:** When setting up large networks (*200 or more points*), it is highly recommended that the networks be disabled during setup. This will dramatically increase user interaction while adding devices and points. Re-enable the networks once setup is complete.

### BACnet Network Settings



The image shows a form titled 'BACnet Network Settings' with a blue header and a blue arrow icon on the right. The form contains the following fields:

Enabled	<input checked="" type="checkbox"/>
Transmission Mode	IP
Local Network Number	1
Virtual Network Number	2
Device Id	2000
Udp Port	47808
APDU Segment Timeout	2000
APDU Timeout	3000
Number of APDU Retries	3

### ENABLED

✓ Check this box to enable the BACnet network.



### *Transmission Mode*

Determines whether the device is to be BACnet IP or BACnet Ethernet.

### *Local Network Number*

The BACnet network number of the bridge itself.

### *Virtual Network Number*

The BACnet network number of the virtual devices on the bridge.

You must click the **SAVE** button to retain changes.

Save

Cancel

### *Device Id*

Perhaps the most important Bridge property is the Device ID for the application. Valid range is from 0 to 4194302. **Default value is 2000.**

### *Udp Port*

The Udp Port defines which HTTP Port the BACnet messages will use. The **default value is 47808** and can also be reference in its hex form as 0xBAC0.

The following network level properties refer to APDUs. APDU (*Application Protocol Data Unit*) is a unit of data within any of the BACnet link-layer protocols, consisting of protocol control information and possibly application user data.

### *APDU Segment Timeout*

Specifies the time waited before retransmission of an APDU segment (*only relevant if devices are doing segmentation*). **Default value is 2000 ms.**

### *APDU Timeout*

Specifies the time waited before retransmission of an APDU requiring acknowledgment, for which no acknowledgement has been received. **Default value is 3000 ms.**

### *Number of APDU Retries*

Defines the maximum number of times an APDU will be retransmitted. **Default value is 3.**

You must click the **SAVE** button to retain changes.

Save

Cancel

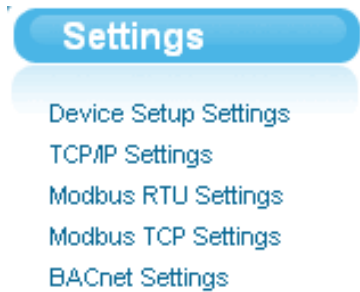
## BACnet Alarming

The Onyx Modbus to BACnet Bridge supports standard BACnet alarming. Whenever a device goes offline (*communication has ceased with the device*) a BACnet alarm is generated and sent to any clients that have registered with the BAS Bridge.

# DEVICE SETUP SETTINGS

## Advanced Setup Options

From the **Home** page select **DEVICE SETUP SETTINGS** to configure advanced Modbus setup options. Always press the **SAVE** button to complete these settings.



The Device Setup Settings provide advanced options required by some Modbus slave devices. When an option is selected, it will allow access to fields normally hidden when setting up the slave devices and their points. Each option and the fields it affects are detailed below. Also, the ability to upgrade, backup and restore the Bridge application can be found here.

A screenshot of the "Device Setup Settings" form. The title "Device Setup Settings" is at the top left, and a blue circular icon with a downward arrow is at the top right. The form is divided into two sections: "Device Setup" and "Point Setup".

- Device Setup**
  - Allow changes to the device Byte Orders
  - Allow changes to the device Ping Address
- Point Setup**
  - Allow changes to Modbus Address Format
  - Allow changes to the BACnet Object Instance Number
  - Allow changes to the BACnet Object Name

At the bottom of the form are six buttons: "Save", "Cancel", "Upgrade", "Backup", "Restore", and "Reboot".

## Allow Changes to the Device Byte Orders

This option allows users to access the Modbus Config section when adding/editing a Modbus device. This allows the Bridge to override the network configuration for specific devices.

### Adding New Rtu Device



The screenshot shows a web form titled "Adding New Rtu Device". It has two input fields: "Device Name" with the value "RtuDevice" and "Modbus Address" with the value "1". Below these is a section titled "Modbus Config" which is highlighted with a red border. Inside this section, there is an "Override Network" checkbox that is unchecked. Below the checkbox are two dropdown menus: "Float Byte Order" and "Long Byte Order", both of which are set to "Order3210".

### *Override Network*

This option must be selected to allow changes to the additional Modbus options and to override the default network settings.

The following network level properties, determine the “byte-order” that devices use for sending or receiving 4-byte (32-bit) data values in Modbus messages. Separate settings are for float (floating-point) and long (integer).

### *Float Byte Order*

Choices reflect two alternate methods, where numerals 0, 1, 2, and 3 represent the least significant byte to most significant byte, as follows:

- ✓ Order3210 – Most significant byte first, or “big-endian”, it is the default.
- ✓ Order1032 – Bytes transmitted in a 1,0,3,2 order, or “little-endian”.
- ✓ Order2301 – Swapped byte version of order 3210.
- ✓ Order0123 – Swapped byte version of order 1032.

### *Long Byte Order*

Choices reflect two alternate methods, where numerals 0, 1, 2, and 3 represent the least significant byte to most significant byte, as follows:

- ✓ Order3210 – Most significant byte first, or “big-endian”, it is the default.
- ✓ Order1032 – Bytes transmitted in a 1,0,3,2 order, or “little-endian”.
- ✓ Order2301 – Swapped byte version of order 3210.
- ✓ Order0123 – Swapped byte version of order 1032.

## Allow Changes to the Device Ping Address

This option allows users to access the device Ping Address when adding/editing a Modbus device. The **default value is address 40001** which represents Holding Register 0. This value can be changed if a specific device requires it (*very rare*). Ping requests are handled at the network level and are used to “Monitor” device status if normal polls are not being responded to.

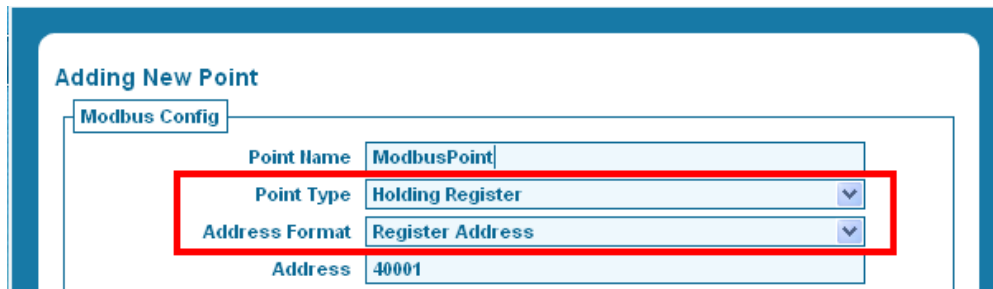
### Adding New Rtu Device



Device Name	RtuDevice
Modbus Address	1
Ping Address	40001

## Allow Changes to Modbus Address Format

This option allows users to access both the POINT TYPE and ADDRESS FORMAT fields when adding/editing a Modbus device. This option is required if extended Modbus addressing is required (*points above 9999*).



Point Name	ModbusPoint
Point Type	Holding Register
Address Format	Register Address
Address	40001

### Point Type

Defines the type of Modbus information this point represents. Available options are:

- **Coil** – Single-bit flag (*binary, On/Off*) representing the status of a discrete output on the slave device
  - The Bridge has the ability to both **read** from and **write** to coils
  - Valid addresses are 00001 – 09999 when using Register Address format, and 0-65535 when using either of the offset formats.
- **Input** – Single-bit flag (*binary, On/Off*) representing the status of a discrete input on the slave device
  - The Bridge can only **read** from inputs
  - Valid addresses are 10001 – 19999 when using Register Address format, and 0-65535 when using either of the offset formats.
- **Holding Register** – 16-bit value (*2 bytes*) representing general purpose data on the slave device
  - The Bridge has the ability to both **read** from and **write** to holding registers
  - Valid addresses are 40001 – 49999 when using Register Address format, and 0-65535 when using either of the offset formats.
  - Holding registers can be indexed to provide access to individual bit values
- **Input Register** – 16-bits value (*2 bytes*) representing general purpose data on the slave device
  - The Bridge can only **read** from inputs
  - Valid addresses are 30001 – 39999 when using Register Address format, and 0-65535 when using either of the offset formats.
  - Input registers can be indexed to provide access to individual bit values

## Address Format

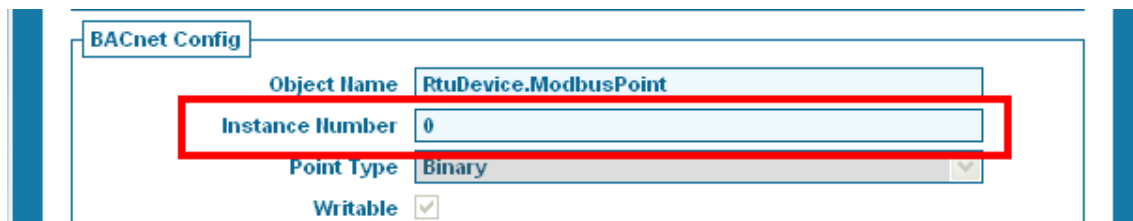
Defines the format the Address is defined in. The default is set to Register Address (*recommended*). Available options:

- **Register Address** – This address format is “one-based”, meaning that a coil addressed as Modbus 10110 has a “Decimal offset” of 109, and a “Hex offset” of 6D.
- **Hex Register Offset** – This address format is “zero-based”, meaning that a coil addressed as Modbus 109 has a “Hex address” of 6D.
- **Decimal Register Offset** – This address format is “zero-based”, meaning that a coil addressed as Modbus 109 has a “Decimal address” of 109.

## Allow Changes to the BACnet Object Instance Number

This option allows users to access the INSTANCE NUMBER field when adding/editing a Modbus point. This option should rarely be used and is only provided as a convenience when specific Instance Numbers are required (*replacement installation*). A value of -1 indicates that the descriptor should be automatically numbered by object type and ensures uniqueness.

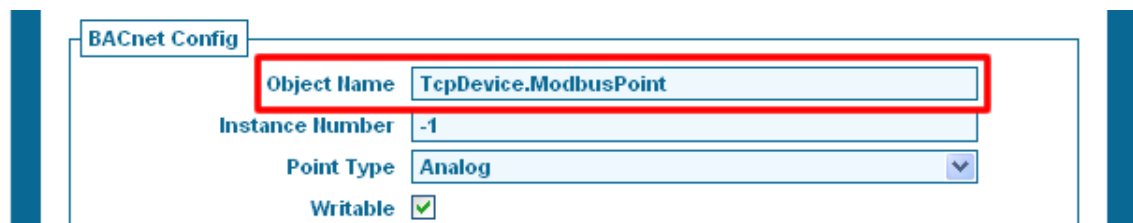
**Note:** The BACnet Object Instance Number must be unique among the available BACnet object types (*Boolean, Analog and Multi-State*).



The screenshot shows a 'BACnet Config' form with the following fields: 'Object Name' (RtuDevice.ModbusPoint), 'Instance Number' (0), 'Point Type' (Binary), and 'Writable' (checked). The 'Instance Number' field is highlighted with a red border.

## Allow Changes to the BACnet Object Name

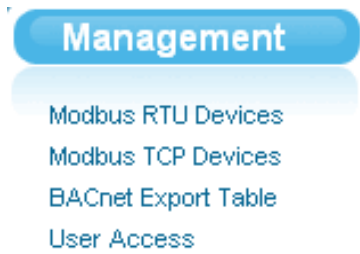
This option allows users to access the OBJECT NAME field when adding/editing a Modbus point. This option should rarely be used and is only provided as a convenience when specific Object Names are required. This value defaults to the device name, followed by a period (.), followed by the point name which ensures uniqueness.



The screenshot shows a 'BACnet Config' form with the following fields: 'Object Name' (TcpDevice.ModbusPoint), 'Instance Number' (-1), 'Point Type' (Analog), and 'Writable' (checked). The 'Object Name' field is highlighted with a red border.

# MODBUS RTU DEVICE MANAGEMENT

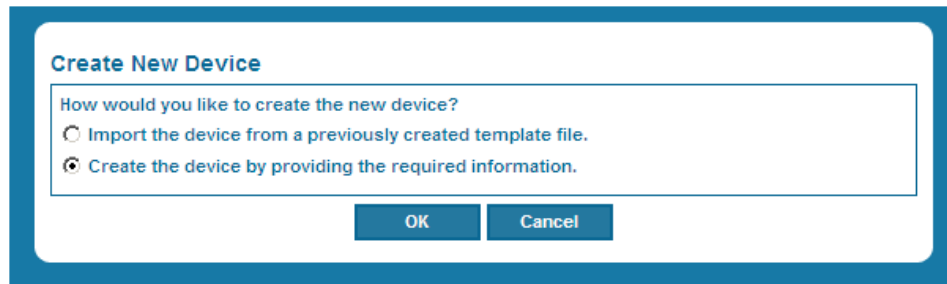
From the **Home** page select MODBUS RTU DEVICES to manage the devices attached to the Modbus RTU network. Because this is a multi-protocol bridge only focus on Modbus RTU devices for this exercise.



A vertical menu titled "Management" with a blue header. Below the header are four items: "Modbus RTU Devices", "Modbus TCP Devices", "BACnet Export Table", and "User Access".

## Adding Devices to the Network

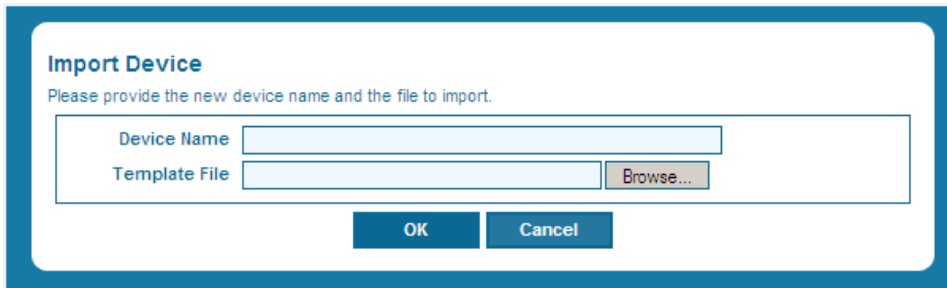
Click the **ADD** button to add a Modbus slave device to the Bridge. You can create devices by importing an existing template or by manual entry.



A dialog box titled "Create New Device". It contains the question "How would you like to create the new device?" and two radio button options: "Import the device from a previously created template file." and "Create the device by providing the required information." The second option is selected. At the bottom are "OK" and "Cancel" buttons.

## Importing a New Device

Devices that have been previously exported can be imported into the Bridge at any time. When importing a device, the device's Modbus Address is automatically chosen from the next available address (*not in use on the network*).



A dialog box titled "Import Device". It contains the instruction "Please provide the new device name and the file to import." Below this are two input fields: "Device Name" and "Template File". The "Template File" field has a "Browse..." button next to it. At the bottom are "OK" and "Cancel" buttons.

### *Device Name*

Defines the unique name that identifies this device on the network.

### *Template File*

Select the BROWSE button to locate the template file previously exported.

## Manually Adding a New Device

**Adding New Rtu Device**

Device Name: RtuDevice

Modbus Address: 1

Ping Address: 40001

**Modbus Config**

Override Network:

Float Byte Order: Order3210

Long Byte Order: Order3210

**Bacnet Config**

Device Name: Bacnet Device

Device Instance: -1

Device Description: Bacnet Description

Save Cancel

### *Device Name*

Defines the unique name that identifies this device on the network. Default value is RtuDevice, which can and should be changed to something meaningful.

### *Modbus Address*

Defines the numerical address of the device on the Modbus network. Valid range is from 1 to 247, default value is 1.

The remaining fields are considered advanced settings and are hidden by default in the Bridge. Please refer to Device Setup Settings for detailed descriptions of these fields.

You must click the **SAVE** button to retain changes.



## Offline Status Indicator

All devices contain an offline status point which is linked to the integrating network. This Boolean point is true when the parent device is in an offline state (*communication has ceased with the device*).

## Managing Devices Added to the Network

As each device is added, it is automatically selected in the device table to provide access to other device level management. You can also click on any device row to change the device you wish to manage. The active selected device is indicated by a faint blue highlight. When selected you can edit, delete, duplicate (*create 1-50 copies*), export for use on another Bridge, and manage the points on the selected device. When editing a device, the same options are available as when adding a device manually (*see above*).

**Modbus RTU Devices**

Select a device to manage its points.

Name	Status	Device Address
Generator1	{fault}	1
RTU7	{fault}	2
Power Meter	{fault}	3
import	{fault}	4

**Buttons:** Add, Reorder, Edit, Mass Updates, Duplicate, Export

**Points on selected device**

Name	Data Address	Export Value	Read Value	Write Value
ModbusPoint	modbus:40001	false	false	-

**Buttons:** Add, Reorder, Edit, Mass Updates, Duplicate

## Mass Updates

You can delete multiple devices as well as update all non-advanced settings through the Mass Updates functionality. Make all necessary changes at once and then click the **SAVE** button.

You must click the **SAVE** button to retain changes.

**Save** **Cancel**

## Reordering Devices

The ability to reorder the device table is available by clicking the **REORDER** button. This feature is for management purposes only and has no effect on the functionality of the Bridge.

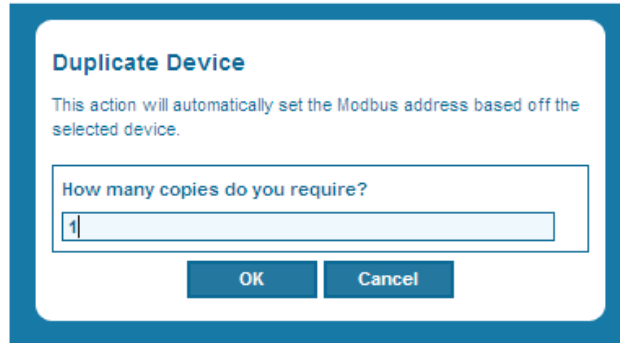
You must click the **SAVE** button to retain changes.

**Save** **Cancel**



## Duplicating a Device

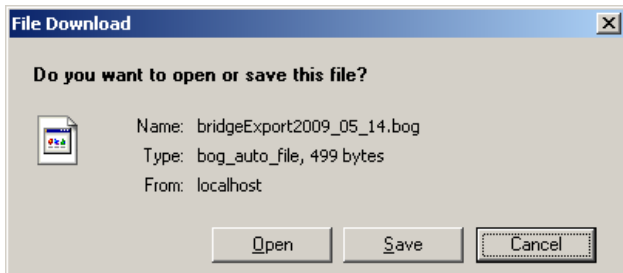
To duplicate a device, **highlight the device** and click the **DUPLICATE** button. Enter the number of copies and click **OK**. The device's Modbus Address is automatically chosen from the next available address (*not in use on the network*) when this action is selected.



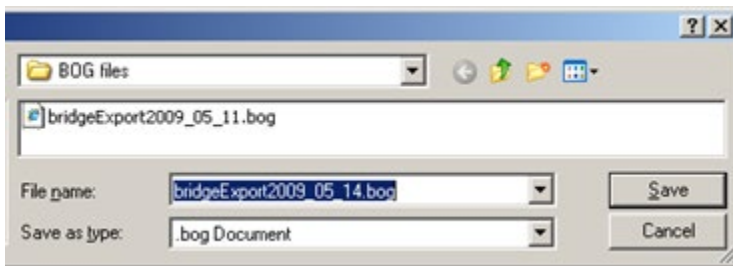
## Exporting a Device for Use on Another Bridge

To export a device, highlight the device and click the **EXPORT** button. The export feature creates a Niagara .bog file that can imported into other Bridge applications at a later time.

1. Confirm the download by clicking the **SAVE** button.

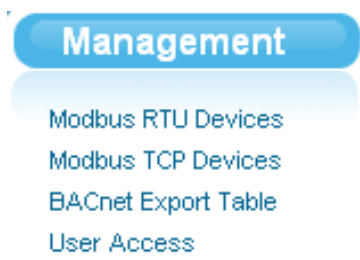


2. Browse to the folder you want to save the file and select the **SAVE** button.



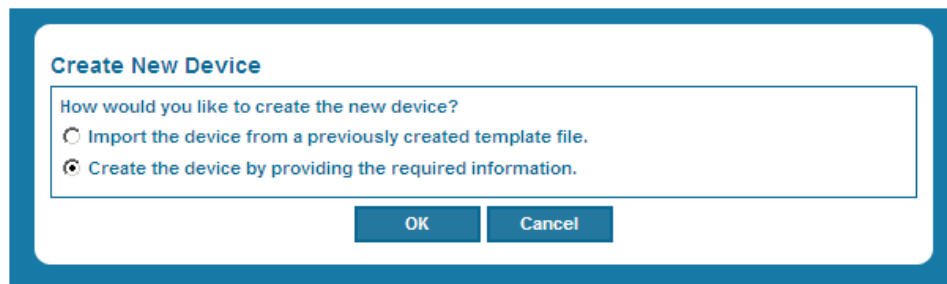
# MODBUS TCP DEVICE MANAGEMENT

From the **Home** page select MODBUS TCP DEVICES to manage the devices attached to the Modbus TCP network. Because this is a multi-protocol bridge only focus on Modbus TCP devices for this exercise.



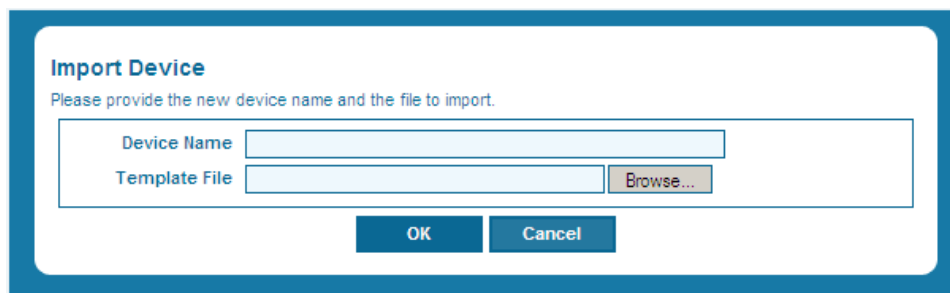
## Adding Devices to the Network

Click the **ADD** button to add a Modbus slave device to the Bridge. You can create devices by importing an existing template or by manual entry.



## Importing a New Device

Devices that have been previously exported can be imported into the Bridge at any time. When importing a device, the device's Modbus Address is automatically chosen from the next available address (*not in use on the network*). The IP Address is not updated and should be updated (*via the **EDIT** button*) after the device has been added.



### *Device Name*

Defines the unique name that identifies this device on the network

### *Template File*

Select the **BROWSE** button to locate the template file previously exported

## Manually Adding a New Device

**Adding New Tcp Device**

Device Name

Modbus Address

IP Address

Port

Ping Address

**Modbus Config**

Override Network

Float Byte Order

Long Byte Order

**Bacnet Config**

Device Name

Device Instance

Device Description

### *Device Name*

Defines the unique name that identifies this device on the network. Default value is *RtuDevice*, which can and should be changed to something meaningful.

### *Modbus Address*

Defines the numerical address of the device on the Modbus network. Valid range is from 1 to 247, default value is 1.

### *IP Address*

Defines the Internet Protocol Address of the TCP slave device.

### *Port*

Defines the HTTP Port used by the TCP slave device. Default value is 502.

The remaining fields are considered advanced settings and are hidden by default in the Bridge. Please refer to Device Setup Settings for detailed descriptions of these fields.

You must click the **SAVE** button to retain changes.



## Offline Status Indicator

All devices contain an offline status point which is linked to the integrating network. This Boolean point is true when the parent device is in an offline state (*communication has ceased with the device*).

## Managing Devices Added to the Network

As each device is added, it is automatically selected in the device table to provide access to other device level management. You can also click on any device row to change the device you wish to manage. The active selected device is indicated by a faint blue highlight. When selected you can edit, delete, duplicate (*create 1-50 copies*), export for use on another Bridge, and manage the points on the selected device. When editing a device, the same options are available as when adding a device manually (*see above*).

### Modbus TCP Devices

Select a device to manage its points.

Name	Status	Device Address	IP Address	Port
Generator1	{ok}	1	192.168.1.201	502
RTU7	{ok}	2	192.168.1.202	502
Power Meter	{ok}	3	192.168.1.203	502

Add Reorder Edit Mass Updates Duplicate Export

#### Points on selected device

Name	Data Address	Export Value	Read Value	Write Value
ModbusPoint	modbus:40001	0.0	0.0	-

Add Reorder Edit Mass Updates Duplicate

## Mass Updates

You can delete multiple devices as well as update all non-advanced settings through the Mass Updates functionality. Make all necessary changes at once and then click the SAVE button.

You must click the SAVE button to retain changes.



## Reordering Devices

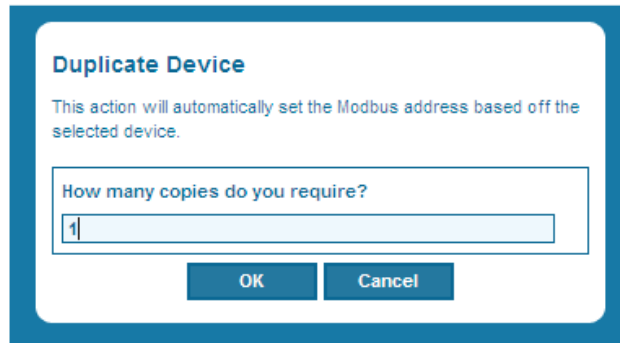
The ability to reorder the device table is available by clicking the REORDER button. This feature is for management purposes only and has no effect on the functionality of the Bridge.

You must click the SAVE button to retain changes.



## Duplicating a Device

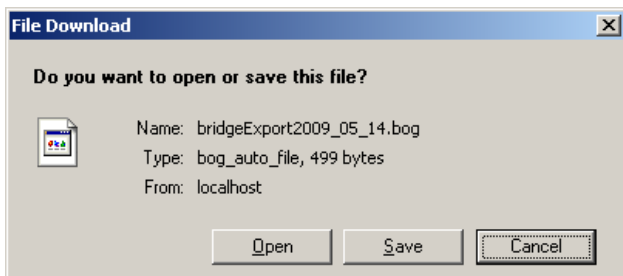
To duplicate a device, highlight the device and click the **DUPLICATE** button. Enter the number of copies and click **OK**. The device's Modbus Address is automatically chosen from the next available address (*not in use on the network*) when this action is selected. The device's IP Address is automatically incremented by 1 (*i.e. 192.168.1.201 becomes 192.168.1.202 when duplicated*).



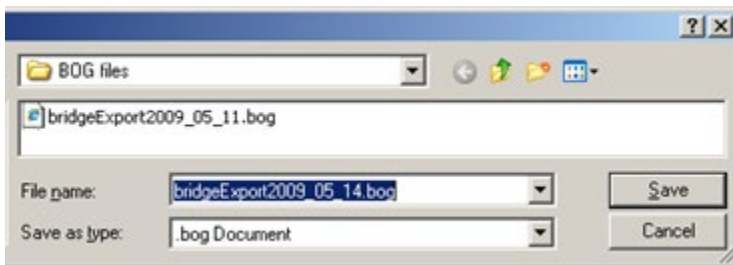
## Exporting a Device for Use on Another Bridge

To export a device, highlight the device and click the **EXPORT** button. The export feature creates a Niagara .bog file that can be imported into other Bridge applications at a later time.

3. Confirm the download by clicking the **SAVE** button.



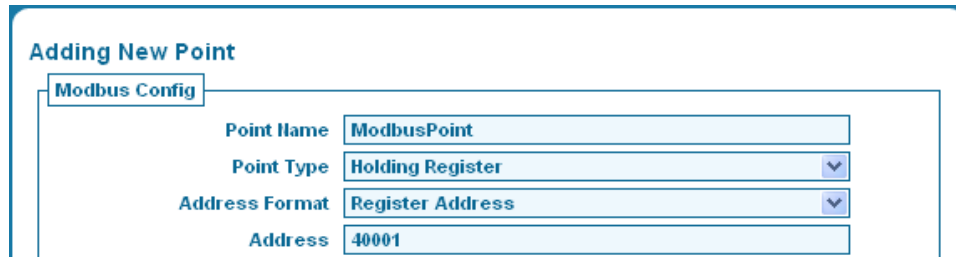
4. Browse to the folder you want to save the file and select the **SAVE** button.



# MODBUS POINT MANAGEMENT

## Adding Points to Devices

Click the **ADD** button located under the *Points on selected device* section to add a point to the selected Modbus slave device.



**Adding New Point**

**Modbus Config**

Point Name:

Point Type:

Address Format:

Address:

### Point Name

Defines the unique name that identifies this device on the Modbus slave device. The default value is **ModbusPoint** which can and should be changed to something meaningful.

### Address

Defines the numerical address of the point on the Modbus slave device. Valid ranges are 00001-19999 for Coils, 10001-19999 for Inputs, 30001-39999 for Input Registers and 40001-49999 for Holding Registers.

The remaining fields are considered advanced settings and are hidden by default in the Bridge. Please refer to **Device Setup Settings** for detailed descriptions of these fields.

## Mass Updates

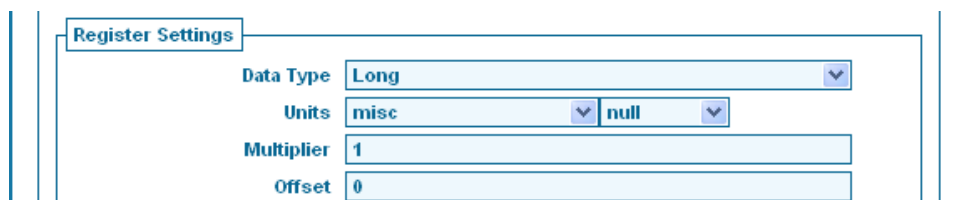
You can delete multiple points as well as update all non-advanced settings through the Mass Updates functionality. Make all necessary changes at once and then click the **SAVE** button.

You must click the **SAVE** button to retain changes.



## Register Settings

The register settings section is available whenever the selected Point Type is a register. By default the Point Type field is hidden and automatically updated via the Address field. Valid ranges for registers are 30001-39999 for Input Registers and 40001-49999 for Holding Registers.



**Register Settings**

Data Type:

Units:

Multiplier:

Offset:

## Data Type

Defines the type of data being read from the register(s). Available options are:

- ✓ **Integer** – Unsigned 16-bit integer ranging from 0 to 65,535 (*commonly referred to as a “word”*)
- ✓ **Signed Integer** – Signed 16-bit integer ranging from -32,768 to 32,767 (*commonly referred to as a “short”*)
- ✓ **Long** – Unsigned 32-bit integer ranging from 0 to 4,294,967,295
- ✓ **Signed Long** – Signed 32-bit integer ranging from -2,147,483,648 to 2,147,483,647
- ✓ **Float Type** – (*floating point*) 32-bit single precision. Very small and large numbers are possible (*commonly referred to as a “real”*)

**Note:** 32-bit data types require 2 consecutive registers in the Modbus slave device. This configuration is automated inside the Bridge. The Address defined should represent the lowest register address being accessed (*i.e. a Long data type at Address 40001 will access the data at Address 40002 automatically*).

## Units

Defines the units being represented by this point in the slave device.

## Multiplier

Defines the multiplier applied to the incoming value.

## Offset

Defines the offset applied to the incoming value.

**Note:** The Multiplier and Offset fields are used to scale the values being read from and written to the slave device. Unit conversion is also applied after scaling if the units between the slave point and the target point are convertible.

You must click the **SAVE** button to retain changes.



## BACnet Config

A screenshot of a web form titled 'BACnet Config'. It contains four input fields: 'Object Name' with the value 'TcpDevice.ModbusPoint', 'Instance Number' with the value '-1', 'Point Type' with a dropdown menu showing 'Analog', and 'Writable' with a checked checkbox.

## Point Type

Defines the type of BACnet Object exposed on the network. Available options are:

- ✓ **Binary** – Exposes a Binary Value Object on the BACnet network
- ✓ **Analog** – Exposes an Analog Value Object on the BACnet network
- ✓ **Multi-state** – Exposes a Multi-state Value Object on the BACnet network

## Writable

Defines whether this point is writable via a BACnet client. This option is not available if the selected Modbus Point Type is not writable (*Inputs and Input Registers*). Writable points expose the standard BACnet priority array to the BACnet network.

The Object Name and Instance Number fields are considered advanced settings and are hidden by default in the Bridge. Please refer to Device Setup Settings for detailed descriptions of this field.

## BACnet Facet Setup

Depending on the Point Type selected in the BACnet Config section different options are available for facet settings.

### Binary Facets

Binary Facets	
True Text	<input type="text" value="true"/>
False Text	<input type="text" value="false"/>
Bit Index	<input type="text" value="0"/>

### True Text

Defines what value is display for a true (On) value

### False Text

Defines what value is display for a false (Off) value

*Bit Index (Models AM-BIP, RM-BIP and TM-BIP only)*

Defines the bit index used to access binary data located in the selected Modbus Register. LSB format (*i.e. a value of 7 on the register will set bits 0-2 to true and bits 3-15 false*).

**Note:** The Bit Index field is only available if the selected Modbus Point Type is a register.

You must click the **SAVE** button to retain changes.

Save

Cancel



## Analog Facets

The screenshot shows a configuration window titled "Analog Facets". It contains four input fields: "Units" is a dropdown menu with "misc" selected; "Min" is a text input field; "Max" is a text input field; and "Precision" is a text input field. There is also a "null" dropdown menu next to the "Units" field.

### Units

Defines the units being represented by this point in the exported BACnet Object.

### Min

Defines the minimum value allowed for the point. Not enforced by the bridge, it is simply provided to the exported BACnet Object. Leave blank for no minimum value.

### Max

Defines the maximum value allowed for the point. Not enforced by the bridge, it is simply provided to the exported BACnet Object. Leave blank for no maximum value.

### Precision

Defines the number of fractional digits (*decimal places*) allowed for the point. Not enforced by the bridge, it is simply provided to the exported BACnet Object. Leave blank for no defined precision.

You must click the [Save](#) button to retain changes.

Two buttons: "Save" and "Cancel".

## Multi-State Facets

Multi-state BACnet Objects provide a textual representation for integer data defined as ordinals. The representations and ordinals together form the "range" of the multi-state object. In BACnet, the ordinals in the range must start at 1 and be continuous (*1=On,2=Off,3=Standby is valid but 1=On,2=Off,4=Standby is not valid*). Therefore, any blank State fields will be ignored and values will always be defined in continuous order (*if State 2 is blank, the ordinal for State 3 becomes 2*).

Click the [ADD STATE](#) button to add more State fields as needed.

The screenshot shows a configuration window titled "Multi-state Values". It contains three text input fields labeled "State 1", "State 2", and "State 3". Below these fields is a blue button labeled "Add State".

You must click the [SAVE](#) button to retain changes.

Two buttons: "Save" and "Cancel".

## Managing Points Added to Modbus Slave Devices

You can click on any point row to select the point you wish to manage. The active selected point is indicated by a faint blue highlight. When selected you can edit, delete, and duplicate (*create 1-50 copies*) the selected point. When editing a point, the same options are available as when adding a point (*see above*).

### Modbus TCP Devices

Select a device to manage its points.

Name	Status	Device Address	IP Address	Port
Generator1	{down}	1	192.168.1.201	502
RTU7	{down}	2	192.168.1.202	502
Power Meter	{down}	3	192.168.1.203	502
Power Meter4	{down}	4	192.168.1.204	502

**Add** **Reorder** **Edit** **Delete** **Duplicate** **Export**

### Points on selected device

Name	Data Address	Export Value	Read Value	Write Value
ModbusPoint	modbus:40001	0.0	0.0	-

**Add** **Reorder** **Edit** **Delete** **Duplicate**

## Reordering Devices

The ability to reorder the point table is available by clicking the **REORDER** button. This feature is for management purposes only and has no effect on the functionality of the Bridge.

You must click the **SAVE** button to retain changes.



## Duplicating a Point

To duplicate a point, highlight the device and click the **DUPLICATE** button. Enter the number of copies and click **OK**. The point's Modbus Address is automatically chosen from the next available address when this action is selected.

### Duplicate Device

This action will automatically set the Modbus address based off the selected device.

How many copies do you require?

**OK** **Cancel**

# BACnet Export Table

From the **Home** page select BACNET EXPORT TABLE to manage the BACnet Objects being exported.

**Management**

- Modbus RTU Devices
- Modbus TCP Devices
- BACnet Export Table**
- User Access

## Managing the BACnet Export Table

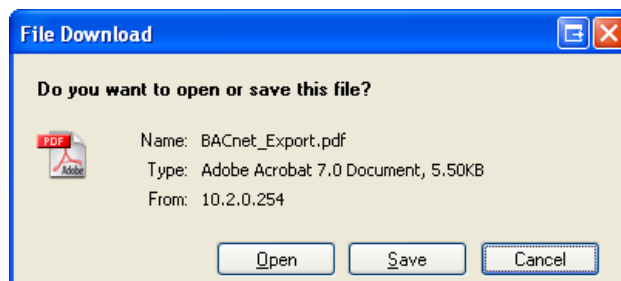
You can click on any BACnet Object row to select the object you wish to manage. The active selected object is indicated by a faint blue highlight. BACnet Export Objects are automatically added and configured when adding points to the Bridge. You can edit the object if required, but this should normally not be required.

Object Name	Object Type	Inst Num	Value	Status	Writable
TcpDevice.Integer	Analog Value	0	2.0	{fault}	true
TcpDevice.Signed Long	Analog Value	1	-2147483648.0	{fault}	false

[Edit](#) [Reorder](#) [Export](#)

## Export the BACnet Export Table

A CSV version of the BACnet Export Table (*for integration purposes*) is available by clicking the **EXPORT** button. When prompted either Open the CSV document or Save it to a location on your computer for later use.



## POINT LIMITATIONS

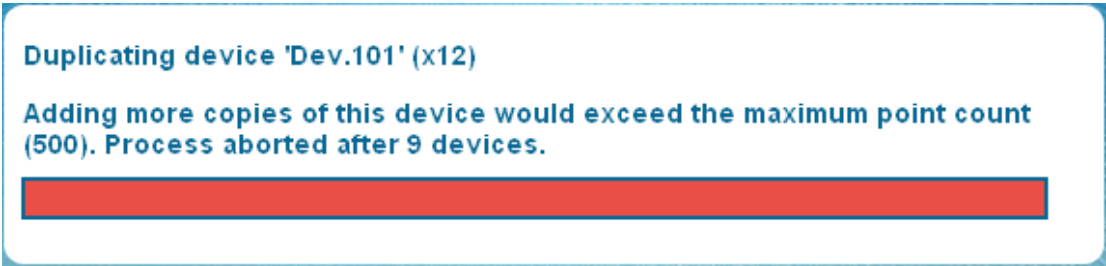
The Onyx Modbus to BACnet Bridge allows a maximum of 1000 Modbus points to be integrated and converted to BACnet. If the maximum point count is reached, various messages may be displayed:

- ✓ This message is displayed at the top of the Device Management page



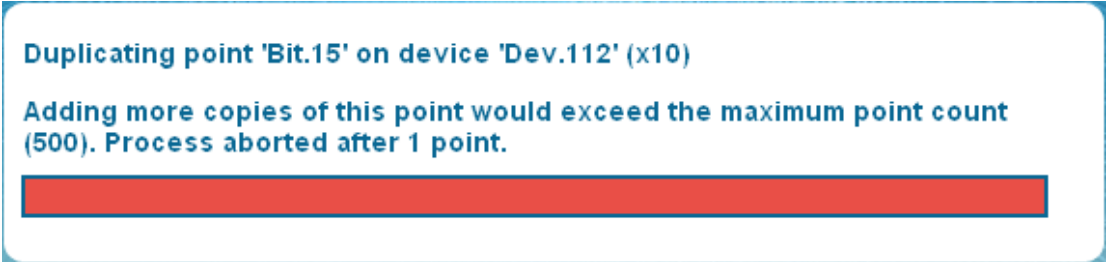
**Point limit of 500 reached on this device**  
**Add and duplicate functions will be unavailable**

- ✓ This message will be seen during device duplication



**Duplicating device 'Dev.101' (x12)**  
**Adding more copies of this device would exceed the maximum point count (500). Process aborted after 9 devices.**

- ✓ This message will be seen during point duplication



**Duplicating point 'Bit.15' on device 'Dev.112' (x10)**  
**Adding more copies of this point would exceed the maximum point count (500). Process aborted after 1 point.**

## RESOURCE LIMITATIONS

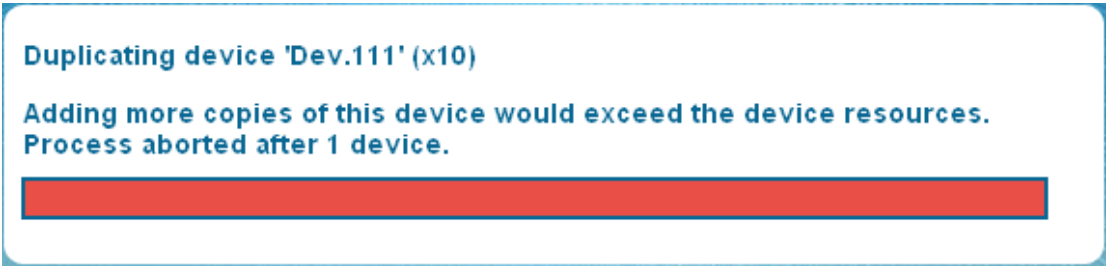
All models monitor their memory resources to help prevent overloading the system. If resources are running low on a device, various messages may be displayed:

- ✓ This message is displayed at the top of the Device Management page



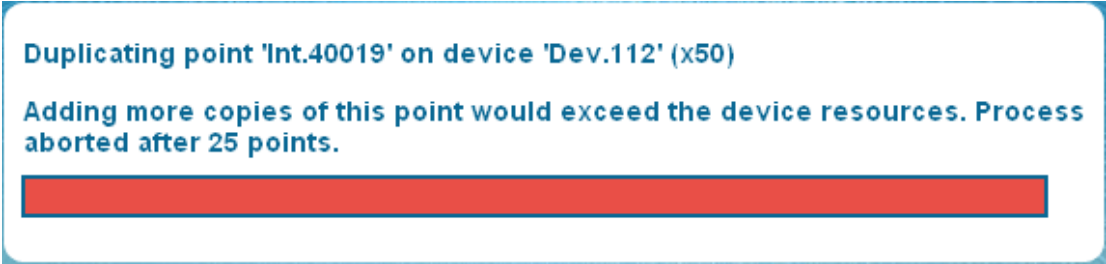
**Resource limit reached on this device**  
**Add and duplicate functions will be unavailable**

- ✓ This message will be seen during device duplication



**Duplicating device 'Dev.111' (x10)**  
**Adding more copies of this device would exceed the device resources.**  
**Process aborted after 1 device.**

- ✓ This message will be seen during point duplication



**Duplicating point 'Int.40019' on device 'Dev.112' (x50)**  
**Adding more copies of this point would exceed the device resources. Process aborted after 25 points.**

If resources have run low and more points are required, the following may help:

1. Reboot the device using the **REBOOT** button under **DEVICE SETUP SETTINGS**. This allows the device to clean up and recover from the build process (*this is especially true if mass duplications have occurred*). You must click the **SAVE** button prior to rebooting if any changes were made to the Device Setup Settings.
2. Adjust the Poll and Ping rates under the **Network Settings** page in **DEVICE SETTINGS**. Slowing these rates will help decrease the amount of active memory being used to process network communications.

# LYNXSPRING COPYRIGHT AND TRADEMARKS

## Trademarks

Lynxpring,<sup>®</sup> JENESys,<sup>®</sup> Helixx,<sup>®</sup> and Onyx<sup>®</sup> are registered trademarks of Lynxpring, Inc. JENESys Edge<sup>™</sup> is a trademark of Lynxpring, Inc.

Other brands and product names mentioned in this manual may be trademarks or registered trademarks of their respective companies and are hereby acknowledged.

## Copyright Notice

Copyright © 2017, Lynxpring, Inc.

All rights reserved.

This document contains proprietary information which is protected by copyright. No part of this document may be copied, photocopied, reproduced, translated, or converted to any electronic or machine-readable form in whole or in part without prior written approval of Lynxpring, Inc.

## Disclaimer

NO WARRANTY. This technical documentation is being delivered to you AS-IS, and Lynxpring makes no warranty as to its accuracy or use. Any use of the technical documentation or the information contained therein is at the risk of the user. Documentation may include technical or other inaccuracies or typographical errors. Lynxpring reserves the right to make changes in this document without prior notice, and the reader should in all cases consult Lynxpring to determine whether any such changes have been made. The information in this publication does not represent a commitment on the part of Lynxpring.

Lynxpring shall not be liable for incidental or consequential damages resulting from the furnishing, performance, or use of this material.

This guide contains links and references to third-party websites that are not under the control of Lynxpring, and Lynxpring is not responsible for the content of any reference material or linked websites. If you access a third party website mentioned in this guide, then you do so at your own risk. Lynxpring provides these links only as a convenience, and the inclusion of the link does not imply that Lynxpring endorses or accepts any responsibility for the content on those third-party sites.

## [Lynxpring Legal Documents](#)