

**SIMPLIFY
ACCESS &
CONTROL**

**Modbus
Panel Guide**

Intelli-Site Security Management Software

Modbus Panel Guide

For Windows 7 Professional and Ultimate,
Windows 8.1 Pro and Enterprise,
Windows 10 Pro and Enterprise,
Server 2008 R2, and
Server 2012 R2

Modbus Panel Guide

Copyright

Copyright © 2019 OSSI, LLC All rights reserved.

Information in this document is subject to change without notice. The software described in this document is furnished under a license agreement or nondisclosure agreement. The software may be used or copied only in accordance with the terms of those agreements. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or any means electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's use without the written permission of OSSI.

OSSI, LLC
10125 S 52nd St.
Franklin, WI 53132-8677
ossi-usa.com

Telephone: (262) 522-1870
Toll Free: (888) 488-2623
Fax: (262) 522-1872

Trademarks

Intelli-Site® is registered in U.S. Patent & Trademark Office.

All other registered and unregistered trademarks are the sole property of their respective owners.

Technical Support Assistance

OSSI, LLC Headquarters

OSSI, LLC
10125 S 52nd St.
Franklin, WI 53132-8677
(262) 522-1870

(888) 488-2623

Technical Support

Technical support is available via telephone or email. Contact Technical Support 8:00 AM to 5:00 PM Central Standard time. If calling after hours, please leave a detailed voice mail message, and someone will return your call as soon as possible.

Email: support@ossi-usa.com
Phone: (888) 488-2623

When calling, please be at the computer prepared to provide the following information:

- Product version number, found by selecting the  **About** button from the Intelli-Site Application Menu.
- The type of computer being used including, operating system, processor type, speed, amount of memory, type of display, etc.
- Exact wording of any messages that appear on the screen.
- What was occurring when the problem was detected?
- What steps have been taken to reproduce the problem?
- It is highly recommended that the user generate a support package for transmission to Intelli-Site technical support staff. To generate the package, run the Intelli-Site Configuration Utility. *Create Support Package...* is the last option in the **Tools** menu.

Modbus Panel Guide

Table of Contents

Copyright	3
Trademarks.....	3
Technical Support Assistance.....	4
Table of Contents	5
1 Introduction	7
2 Installation Guide.....	8
3  Hardware Management View	9
3.1 Setup.....	9
3.1.1 Add a Modbus Driver	9
3.1.2 Add a Communications Method	12
3.1.3 Add an Modbus Panel.....	13
3.1.4 Configure the Modbus Panel.....	15
3.1.5 Enable the Modbus Driver.....	21
3.2 Panel Control Screen.....	22
4  Design View	24
4.1 Modbus Driver Node	24
4.2 Communication Method Node.....	26
4.2.1 TCP/IP Communication Method Node	27
4.2.2 Rs232 Communication Method Node	28
4.3 Modbus Panel Node	29
4.3.1 Modbus Panel Quick Config	29
4.3.2 Modbus Panel Node Properties	33

Modbus Panel Guide

4.3.3 Modbus Child Nodes.....	37
4.4 Project Programming	43
4.4.1 Evaluation Grids and Action Grids.....	43
4.4.2 Automatically Created Screen Objects.....	45
4.4.3 The Panel Control Screen	55
4.4.4 Transaction Types Programming.....	60
Revision History	65

Modbus Panel Guide

1 Introduction

The Intelli-Site Modbus panel integration was written using the Modicon Modbus Protocol Reference Guide (PI-MBUS-300 Rev. J) and the Modbus Application Protocol Specification V1.1b.

This guide explains using the Modbus with Intelli-Site Security Management Software.

2 Installation Guide

The Modbus drivers in Intelli-Site must be installed. No external software is needed.

If Intelli-Site has been installed without the Modbus driver, run the Intelli-Site installer again and select **Modify**.

During the installation or modify installation process, you are presented with the **Select the drivers you want to install** window.

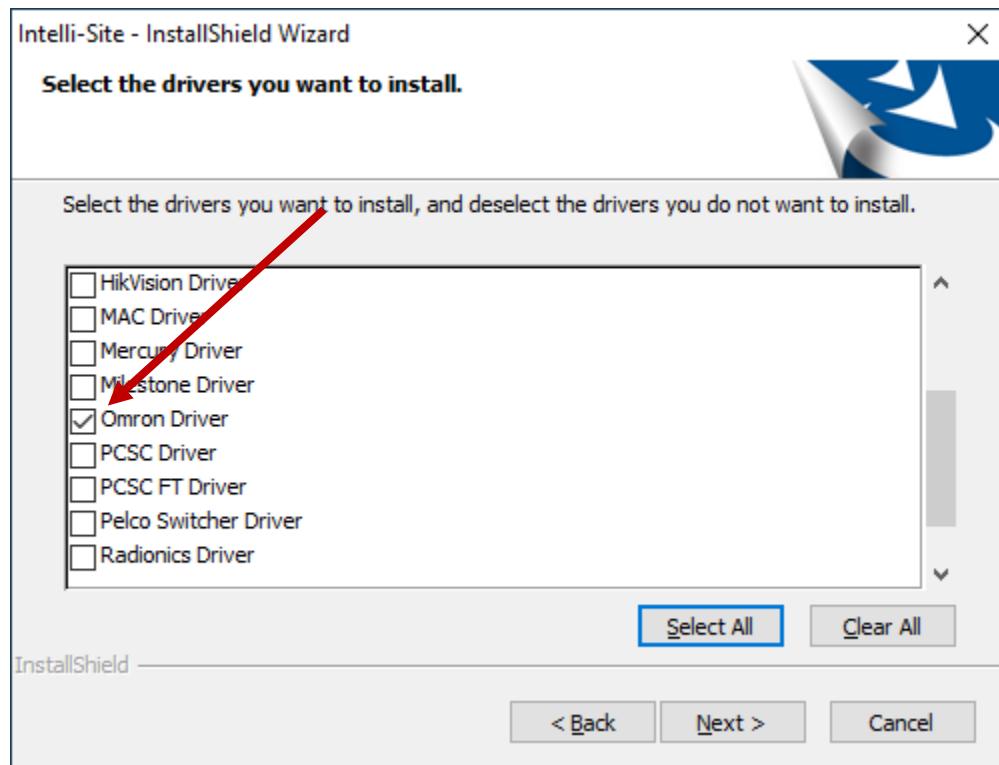


Figure 1 - Select the drivers you want to install

Ensure the **Modbus Driver** option is checked then continue with the installation or modification.

Note: It may be necessary to scroll down to locate the desired driver.

3 Hardware Management View

The Modbus driver and Modbus panel are best configured in  **Hardware Management View**. Once configured, the panels can be easily monitored and controlled using their panel control screens.

3.1 Setup

The Modbus driver and Modbus panel are best configured in  **Hardware Management View**. Before we begin though, it is a good idea to have the IP address and Port number of each of the Omron panels written down.

Once the above information is obtained, the setup process is straight forward.

1. Add an Modbus driver
2. Add a Communication Method to the driver
3. Add an Modbus Panel
4. Configure the Panel
5. Enable the Driver

3.1.1 Add a Modbus Driver

Adding a Modbus Driver is simple, but important. Without it, no communication with the Modbus panels can occur.

Launch the Desktop Client and login.

Hardware is managed in the  **Hardware Management View**. If you are not in  **Hardware Management View**, click on  and select .

Modbus Panel Guide

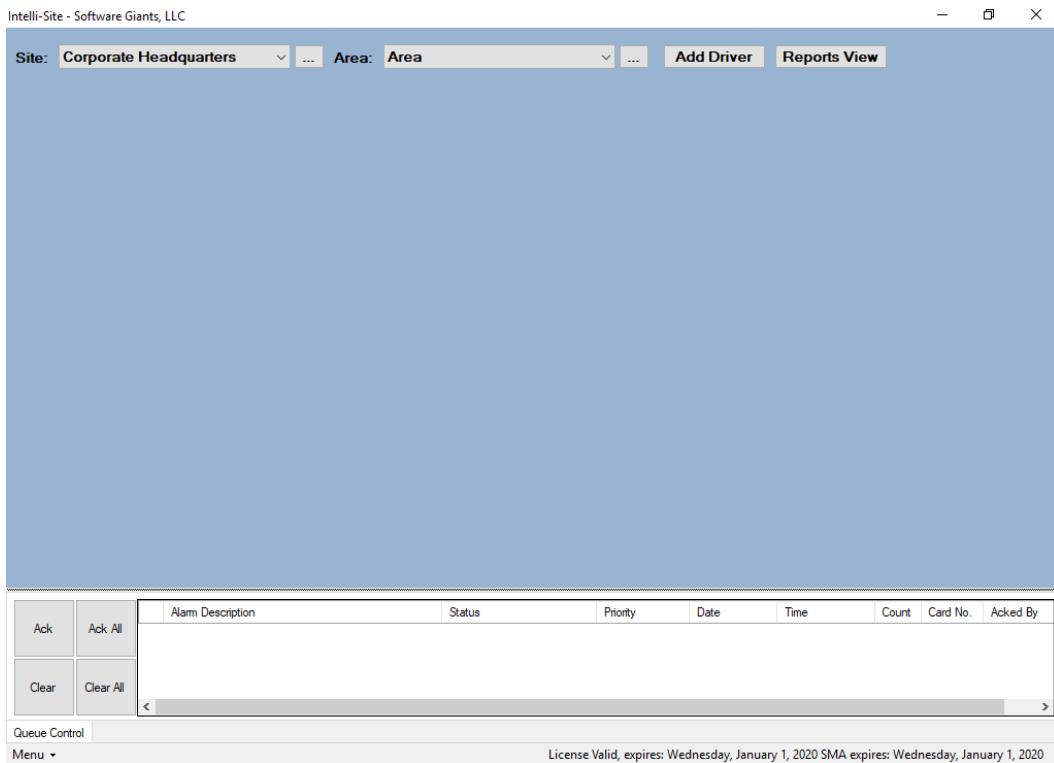


Figure 2 - Hardware Management View

To add the Modbus driver, click **Add Driver**. The **Choose Driver Type** dialog displays.

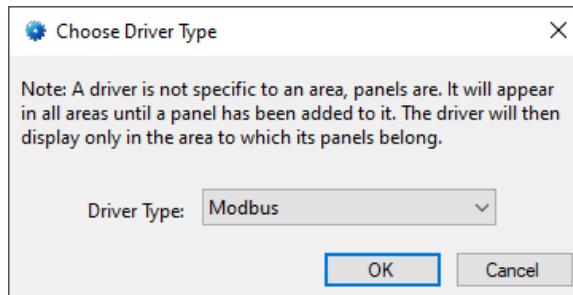


Figure 3 - Choose Driver Type Dialog

Select *Modbus* from the **Driver Type** drop-down menu in the **Choose Driver Type** dialog. Click **OK**. A new ModbusModbus driver icon is added to the screen and its properties dialog opens.

Note: If only one driver is installed, the *Choose Driver Type* dialog does not display. The driver is automatically added and the properties dialog is opened.

Modbus Panel Guide

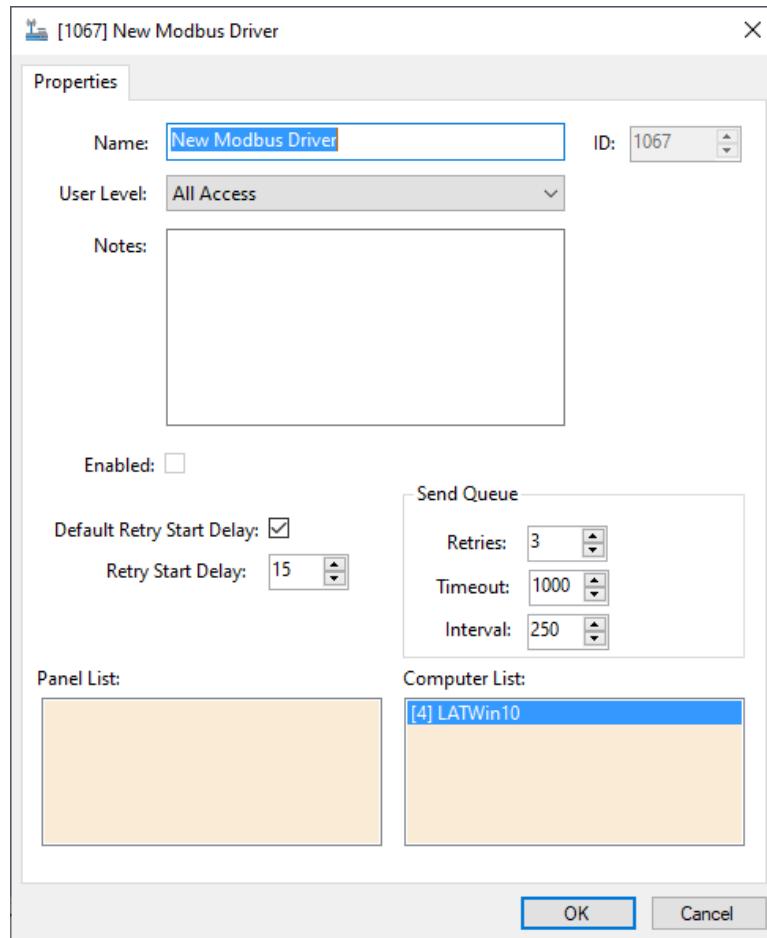


Figure 4 - Modbus Driver Properties Dialog

Please change the name of the driver to reflect the use and/or location of the panels that this driver will manage. It is also necessary to set the **Node Address** field. If Intelli-Site is going to initiate the connection with the Modbus panel, set **Node Address** to the auto-allocated FINS Node number. If the Omron PLC panel will be initiating the communication, then set **Node Address** to the last number of the IP Address of the computer hosting the Intelli-Site Driver Service. For the example project, Intelli-Site will initiate the connection. Therefore, the **Node Address** is set to 241.

Note: For an explanation of all of the driver properties, see [Modbus Driver Node](#).

Modbus Panel Guide



Figure 5 - Modbus Driver Icon

Notice the warning icon, . This appears when the driver requires a communication method and one does not exist yet. So, let's add one.

3.1.2 Add a Communications Method

Each Modbus driver must have a communication method defined and configured. Without one, the Intelli-Site Driver Service does not know where or how to communicate with the panel.

The communication method is added using the right-click context menu on the driver. Right-click on the driver icon and hover over *Comm Method*. The side menu opens. Currently, it only has one option, *Add Comm Method*. Select it. The **Choose Communication Method Type** dialog appears.

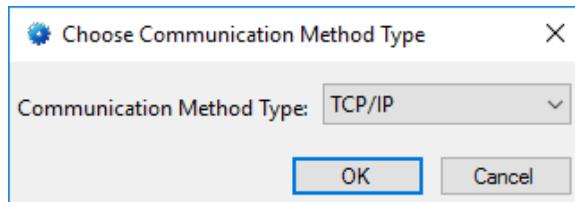


Figure 6 - The Choose Communication Method Type dialog

There are two (2) possible communication methods for any single Radionics device:

- TCP/IP – the Driver will connect directly to the panel using TCP/IP
- Rs232 – the Driver will connect to the panel using a COM port

For this example, we will select *TCP/IP* then click . A new TCP/IP communication method node is added to the driver and its properties dialog opens.

Modbus Panel Guide

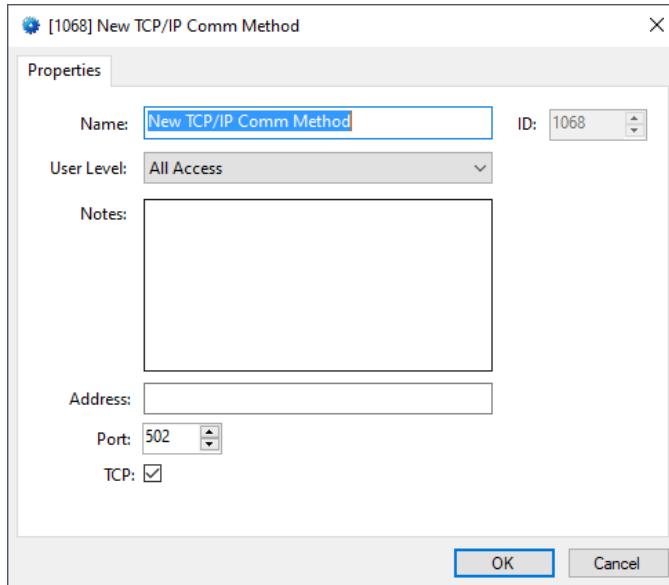


Figure 7 - The TCP/IP Communication Method properties dialog

Note: For a full explanation of all the fields on this dialog and each of the communication methods, please read [Communication Method Node](#).

Enter the IP address of the Modbus in the **Address** field. When the Communication Method is added, the **Port** number is the already set to the default port number for the panel. If for some reason, the panel is set up at a different port, enter the new port here as well. Please change the name and click **OK**.

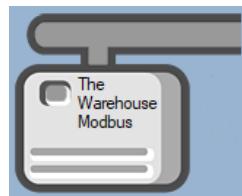


Figure 8 - The driver icon with a communication method

Notice the warning icon is no longer on the front of the driver icon.

It's now time to add a panel to the driver.

3.1.3 Add an Modbus Panel

Once an Modbus driver has been added, it's time to add an Modbus panel to it.

Modbus Panel Guide

Right-click on the Modbus Driver and select *Add Panel*. The **Add Panel** dialog displays.

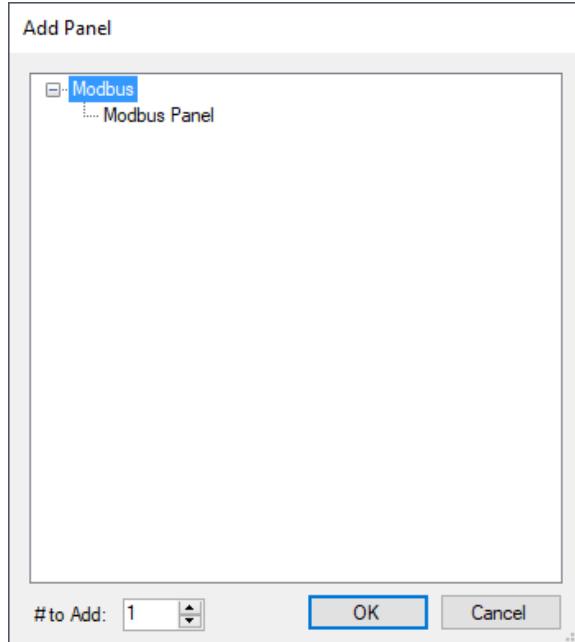


Figure 9 - Add Panel Dialog

This dialog only displays the panels that are appropriate for the target driver. Select the Modbus panel. Change **# to Add** to the desired number of panels at the same IP Address and port. Additional panels can be added later. Click **OK**. A **New Modbus** icon is added to the target Modbus driver.



Figure 10 - A new Modbus panel attached to an Modbus driver

Next comes configuring the panel.

Modbus Panel Guide

3.1.4 Configure the Modbus Panel

The Modbus panel is configured using the **Quick Config** dialog. To open the **Quick Config** dialog, right-click on the Modbus icon and select *Quick Config*.

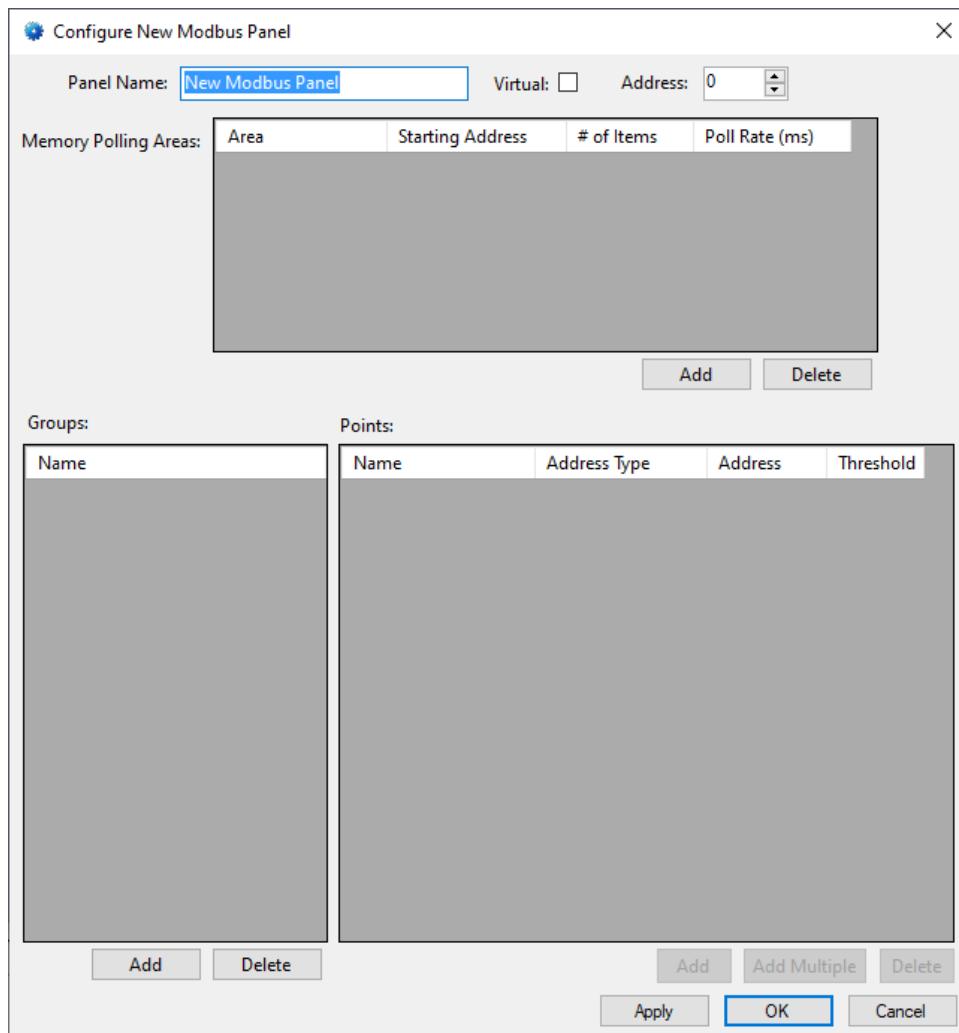


Figure 11 – Modbus Panel Quick Config Dialog

Change the **Panel Name** to describe its use or location. In our example, it is the control panel for The Warehouse. Set the **Address** to match the panel.

Note: *It is assumed that the user is aware of the programming of the PLC and, therefore, knows the memory areas and the points needed.*

Next add the **Memory Polling Areas** that the Driver will need to read from and/or write to.

Modbus Panel Guide

3.1.4.1 Add Memory Polling Areas

The **Memory Polling Areas** are the memory areas of the Modbus that the Driver will read from and/or write to.

Memory Polling Areas:	Area	Starting Address	# of Items	Poll Rate (ms)

Figure 12 - Memory Polling Areas table

Before adding any polling areas, a little thought needs to go into deciding the configuration of each of them. You'll need at least one (1) row for each of the panel's memory areas that need to be accessed and managed by the software. **For the sake of efficiency, define polling areas such that a minimum number of poll commands are required.** A separate poll command is required by each **Memory Polling Area** line in the table. Consider both the memory area where points of interest to the Project are located as well as the size of the memory area. The maximum size of a **Memory Polling Area** is 125 holding registers according to Modbus Application Protocol Specification V1.1b.

Click the button to add a row to the **Memory Polling Areas** table. A new row is added.

Memory Polling Areas:	Area	Starting Address	# of Items	Poll Rate (ms)
	Coil	0	1	250

Figure 13 - Memory Polling Areas table with a new row

Click on the **Area** cell and select the desired area. Set the **Starting Address** and the **# of Items** for this area. The default value for **Polling Rate** is fine for most cases.

Modbus Panel Guide

Memory Polling Areas:	Area	Starting Address	# of Items	Poll Rate (ms)
	Holding Register	806	5	500
	Holding Register	826	5	500

Add **Delete**

Figure 14 - Example Memory Polling Area

Now add **Groups** and **Points** to map the **Memory Polling Areas** for use.

3.1.4.2 Add Groups and Points

With the **Memory Polling Areas** are defined, points need to be added to access the data found in those areas.

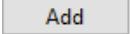
Groups:	Points:					
<table border="1"><thead><tr><th>Name</th></tr></thead><tbody></tbody></table>	Name	<table border="1"><thead><tr><th>Name</th><th>Address Type</th><th>Address</th><th>Threshold</th></tr></thead><tbody></tbody></table>	Name	Address Type	Address	Threshold
Name						
Name	Address Type	Address	Threshold			

Add **Delete** **Add** **Add Multiple** **Delete**

Figure 15 - Groups and Points tables

Modbus Panel Guide

No point can be added until a group is added and selected. A group is a method to organize the points into logical collections. It is an artifact of Intelli-Site and not the panel. How a point is used should dictate the group it belongs to. The example being used has points that are inputs and points that are outputs. We will group those points as **Inputs**, and **Outputs**. Locating those points in the Project Node Tree when programming screen objects will be easier to find since we know the type of points they are. But another way to group them could be based on how those points are used. If those points are associated with a door as a DPS, REX, lock, and timers, one could create groups for each door and assign the points associated with that door to it. It's about how you think about it. Do you see the points as their type or as their usage? This example groups points by their type.

Add a group by clicking the  button under the **Groups** table.

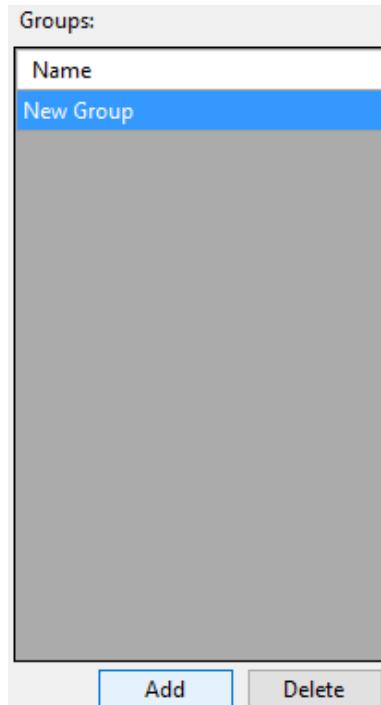


Figure 16 - New Group added to the Groups table

Click on the group in the table to rename it. Notice that the buttons under the **Points** table are enabled. You can now add points to the selected group.

Modbus Panel Guide

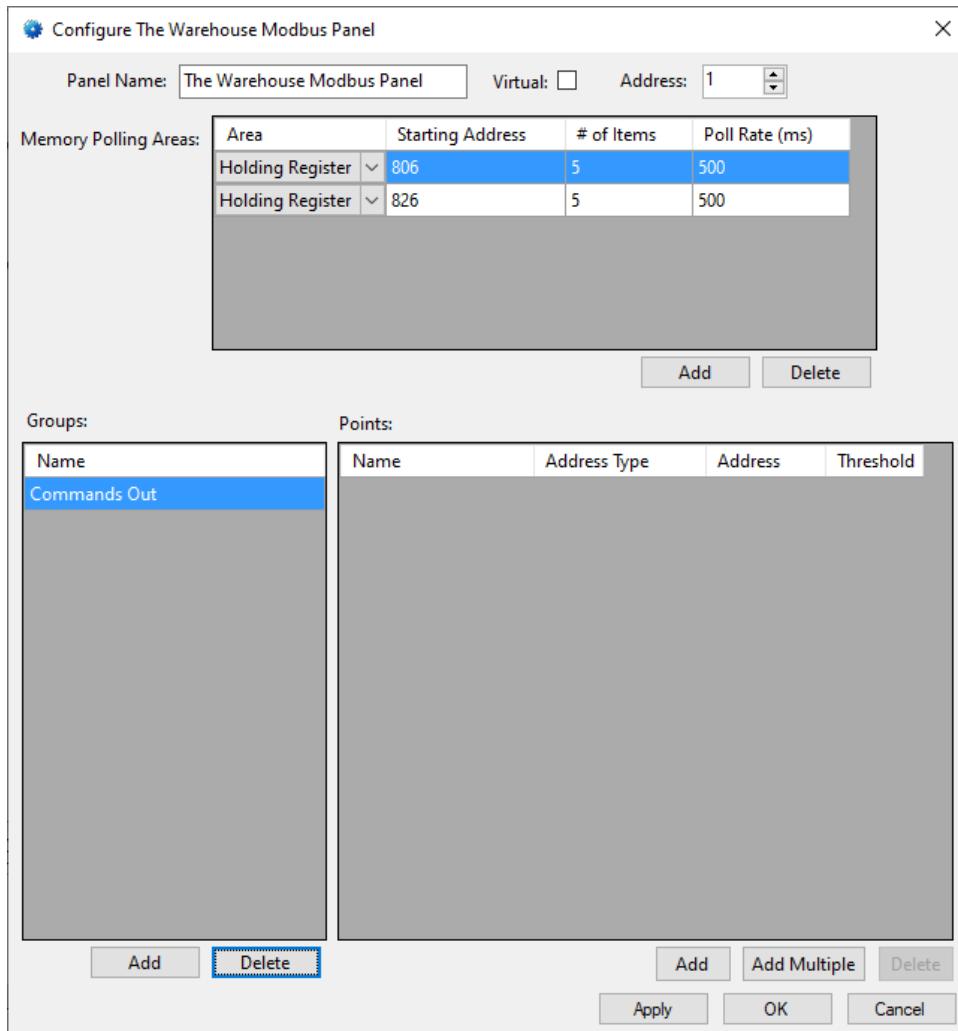


Figure 17 - Partially configured Quick Config

You can add points one at a time using the **Add** button or you can add many points in one fell swoop using the **Add Multiple** button. The example project has five (5) input points. The **Add Multiple** button opens the **Add Multiple** dialog.

Modbus Panel Guide

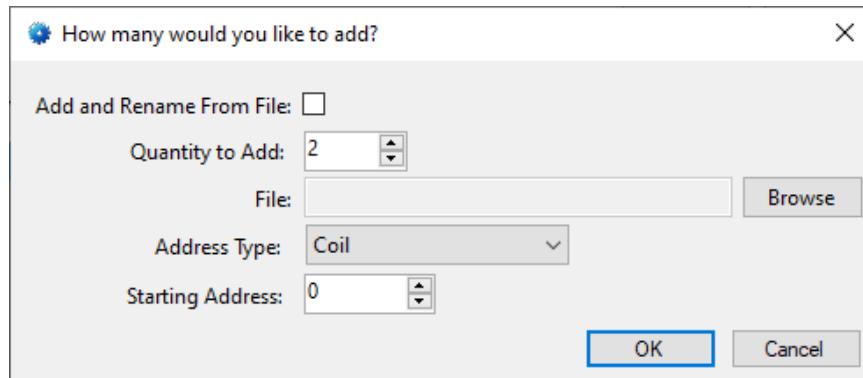


Figure 18 - The Add Multiple dialog

Set **Quantity to Add**, the **Address Type**, and the **Starting Address**. Then click the **OK** button.

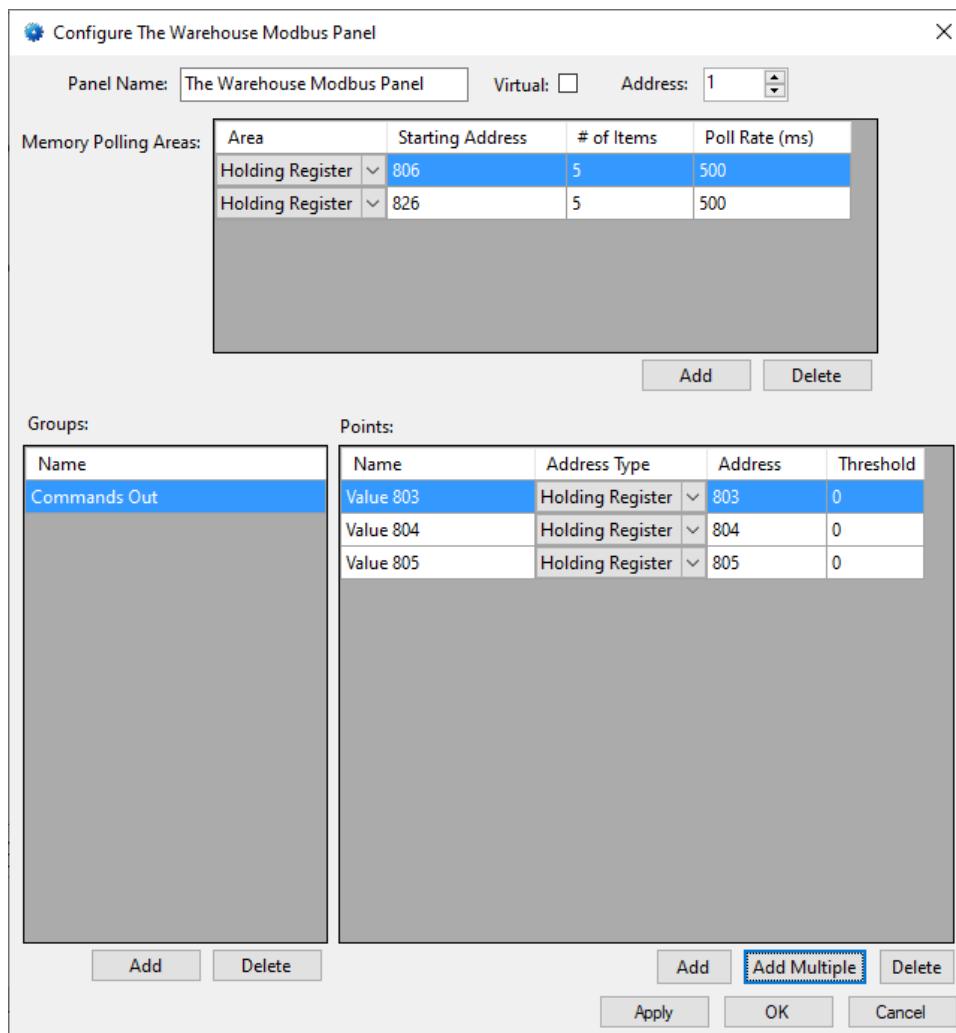


Figure 19 - Newly added points when using the **Add Multiple** button

Modbus Panel Guide

Ensure the **Name**, **Address Type**, **Address**, and **Threshold** for each of the points in the group is correct. Add any additional groups and points needed for your installation.

Note: When a large number of points is added, using the Rename From File... tool is probably the better option. See section 9.4.11 Rename From File in the Intelli-Site User's Guide.

Once all the **Memory Polling Areas**, **Groups**, and **Points** have been added, click the  button to save the changes and close the **Quick Config** dialog.

Note: Points in the Input Register and Holding Register memory polling areas can have sub-values. These sub-values can be used to act on the point or to refer to specific bits within the point. See [Value Sub-Nodes](#).

3.1.5 Enable the Modbus Driver

At this point, the Modbus panel and the Modbus driver are configured, but the driver is not online. A quick way to know this is that the communication indicator is grey.

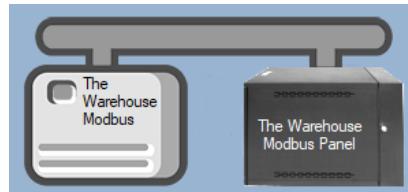


Figure 20 – Disabled Modbus Driver

Right-click on the driver to open the context menu; select *Enable Driver*. The communications indicator will change color to green when it's online.

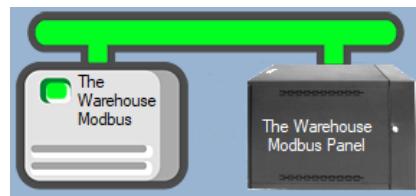


Figure 21 - Enabled Modbus Driver

Modbus Panel Guide

If for some reason the Engine cannot connect to the Driver Service or the Driver cannot connect to the Modbus, the communication indicator will be red.

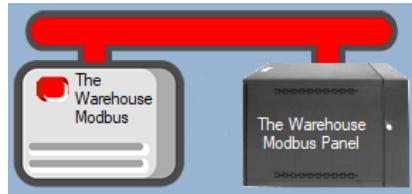


Figure 22 - Enabled Modbus Driver that is not communicating with the Driver Service

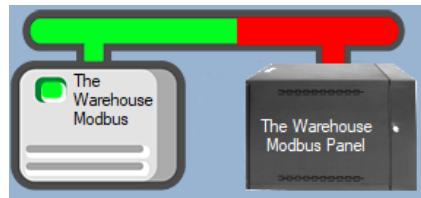


Figure 23 - Enabled Modbus Driver that is communicating with the Driver Service but NOT the panel

Congratulations! The Modbus is integrated and ready to use in screen design for **Live View**. Screen design occurs in **Design View**.

3.2 Panel Control Screen

Every Modbus panel has a **Panel Control Screen**. From this screen, the user can monitor the current state of the panel. Clicking on the panel icon in **Hardware Management View** opens the **Panel Control Screen** of the target panel. The content of the screen is dictated by the panel.

Modbus Panel Guide

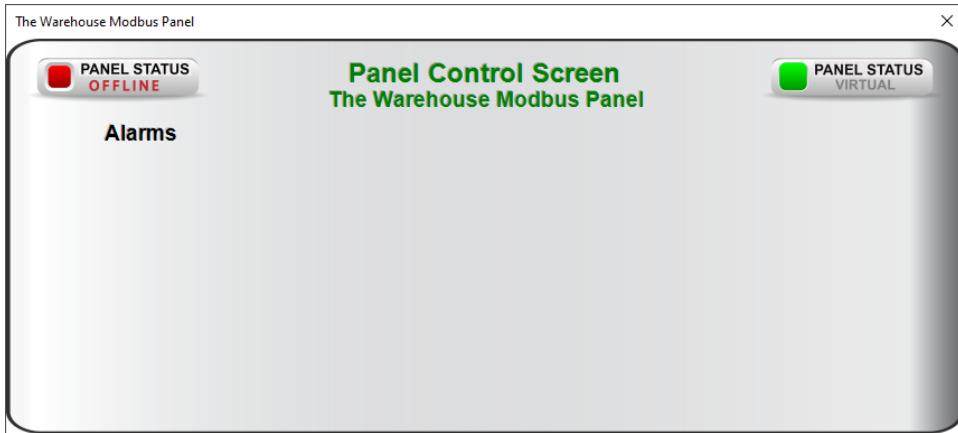


Figure 24 - Sample Modbus Panel Control Screen

Because there are no I/O Points associated with a Modbus panel when the panel is added, there are no points on this screen other than the Online Panel Status and the Virtual Panel Status.

Once the panel has been configured, screen objects can be added to this **Panel Control Screen** popup screen to give the user quick access to the point states and values. See [The Panel Control Screen](#) in Project Programming.

4 Design View

 **Design View** is the home of the Project Node Tree and the place where screens and screen objects are programmed. The following sections explain the Modbus Driver node properties, the Modbus node properties, and screen object programming especially the automatically created screen objects.

4.1 Modbus Driver Node

The Modbus driver node is found by expanding **Setup->Computer Setup->Drivers**.



Figure 25 - Modbus Driver node in the Project Node Tree

Right-click on the driver node and select *Properties* to open the properties dialog.

Modbus Panel Guide

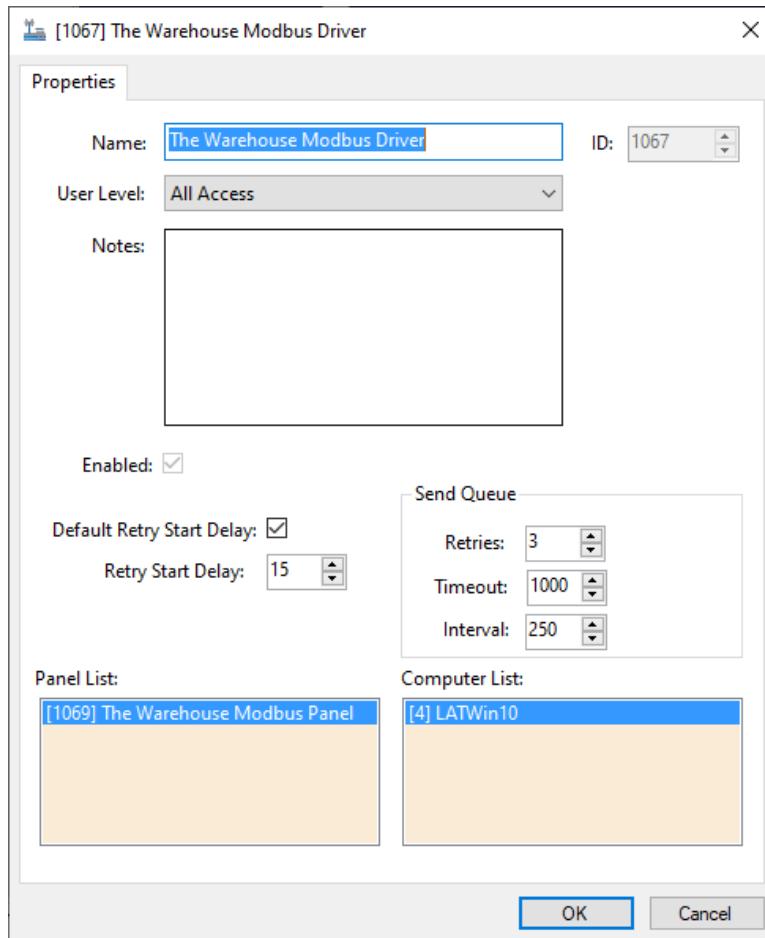


Figure 26 - Modbus Driver properties dialog

Name – edit box; the name for the node; the name doesn't have to be unique

ID – numeric (disabled); the unique identifier of this node; generated by the software

User Level – drop-down menu (default: All Access); the User Level a user must possess to open the properties for this node

Notes – multiline edit box; any notes the user may have for the node

Enabled – check box; when checked the driver is enabled

Default Retry Start Delay – check box (default: checked); if the driver did not connect, pause before attempting to connect again

Modbus Panel Guide

Retry Start Delay – numeric (default: 15); number of seconds to wait between retries on connection attempts

Send Queue – group box; fields associated with the Send Queue behavior

Retries – numeric (default: 3); number of retries for sending a packet

Timeout – numeric (default: 1000); number of milliseconds to wait for a response before assuming the packet was not received

Interval – numeric (default: 25); number of milliseconds to wait after a timeout before sending the packet again

Panel List – drop box; the Modbus nodes attached to this driver

Computer List – drop box (default: the computer on which the Engine is running); the computer node on which this driver is running

4.2 Communication Method Node

The communication method node is a child of the driver. It defines the properties needed to communicate with the panel.



Figure 27 - Communication Method node in the Project Node Tree

There are two (2) different possible communication types for the Modbus driver.

- TCP/IP
- Rs232

Modbus Panel Guide

4.2.1 TCP/IP Communication Method Node

This is the recommended method. With this communication method, Intelli-Site initiates the TCP/IP communication with the panel. When the driver is enabled or if for some reason communication is lost, the software actively attempts to connect with the panel instead of waiting for the panel.

Right-click on the node and select *Properties* to open the properties dialog.

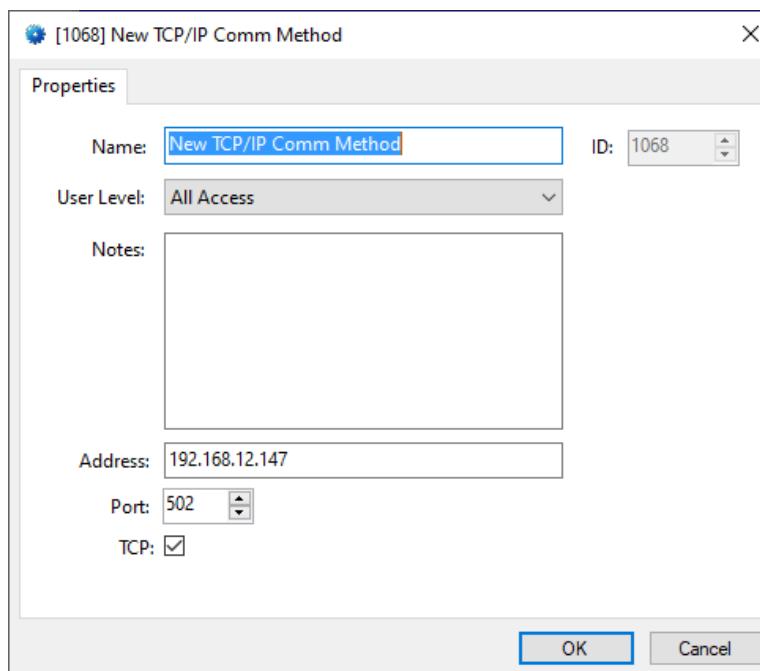


Figure 28 – TCP/IP Communication Method node properties dialog

Name – edit box; the name for the node; the name doesn't have to be unique

ID – numeric (disabled); the unique identifier of this node; generated by the software

User Level – drop-down menu (default: All Access); the User Level a user must possess to open the properties for this node

Notes – multiline edit box; any notes the user may have for the node

Address – edit box; the IP address of the panel

Modbus Panel Guide

Port – numeric (default: 502); the TCP or UDP port number

TCP - checkbox (default: checked); when checked, the driver will communicate with the panel using TCP; when not checked, the driver assumes UDP is the desired communication protocol

4.2.2 Rs232 Communication Method Node

Right-click on the node and select *Properties* to open the properties dialog.

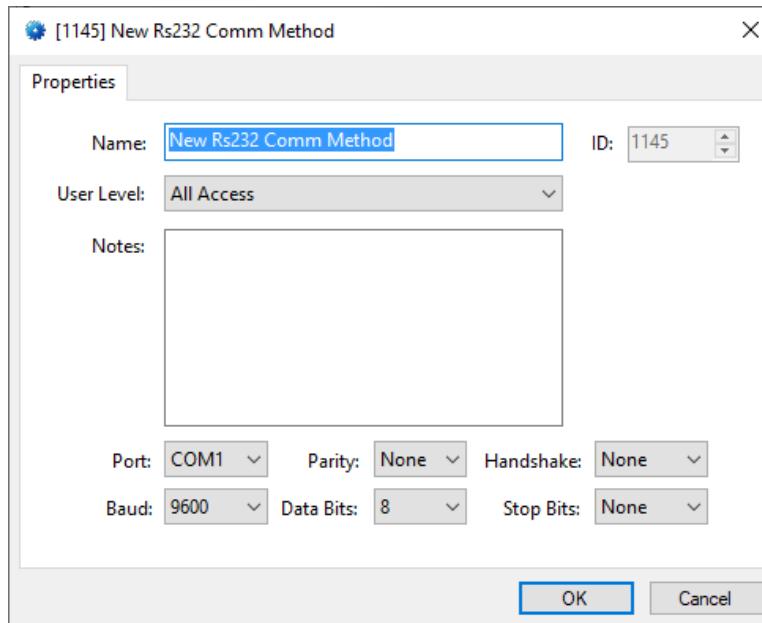


Figure 29 – Rs232 Communication Method node properties dialog

Name – edit box; the name for the node; the name doesn't have to be unique

ID – numeric (disabled); the unique identifier of this node; generated by the software

User Level – drop-down menu (default: All Access); the User Level a user must possess to open the properties for this node

Notes – multiline edit box; any notes the user may have for the node

Port – drop-down menu (default: COM1); the COM port

Modbus Panel Guide

Parity, Handshake, Baud, Data Bits, and Stop Bits – configuration parameters associated with Rs232 communication; for an explanation of Rs232 see [How RS232 Works](#).

4.3 Modbus Panel Node

The Modbus panel node is found by expanding **System Layout** then the Site and Area to which the Modbus panel was added in  **Hardware Management View**. In the example below, the panel was added to the site **Corporate Headquarters** and the area **Area**.



Figure 30 - The Modbus panel node in the Project Node Tree

The Modbus panel node is the root node for the panel. The following sections will explain:

- The configuration of the Modbus panel (a.k.a. **Quick Config**)
- The properties of the Modbus panel node
- The child nodes of the Modbus panel

4.3.1 Modbus Panel Quick Config

The **Quick Config** dialog is accessible either in  **Hardware Management View** or in  **Design View**. To open the **Quick Config** dialog, right-click on the Modbus panel node and select *Quick Config*.

Modbus Panel Guide

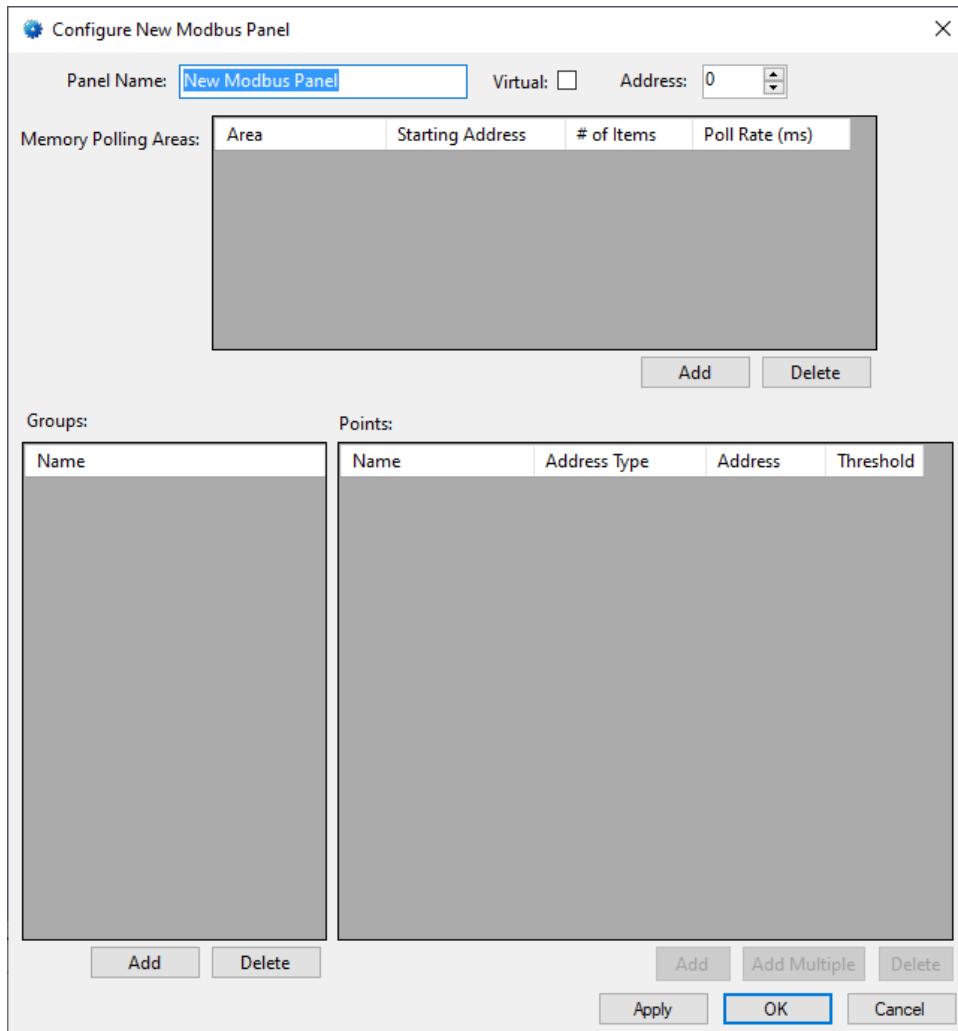


Figure 31 - Modbus panel Quick Config dialog

Panel Name – edit box; the name of the Modbus panel; updating this field will change the name of the node

Virtual – check box; when checked the panel is virtualized, allowing the driver to be brought online without having the physical panel available

Address – numeric (default: 0)

Apply – button; save all modifications but do not close the dialog

OK – button; save all modifications and close the dialog

Cancel – button; abandon any modifications and close the dialog

Modbus Panel Guide

Memory Polling Areas:	Area	Starting Address	# of Items	Poll Rate (ms)

Add **Delete**

Memory Polling Areas – table; lists the regions of panel memory that are polled and mapped

Area – drop-down menu (*Coil, Input Status, Input Register, and Holding Register*); the memory area on the panel

Starting Address – numeric; the offset in the memory are to begin copying; zero-based, the first word/bit/etc. at address 0

of Items – numeric; the number of units to request

Polling Rate – numeric (default: 250); microseconds; how often to request the data for this area

Add - button; add a row to the **Memory Polling Areas** table

Delete - button; delete the highlighted row from the **Memory Polling Areas** table

When defining Memory Polling Areas, please consider network efficiency. Each Memory Polling Area is a separate poll command.

For the sake of efficiency, define areas such that a minimum number of poll commands and responses are required. When the type is *Bits*, a different poll message is received for each bit. Words are the most efficient. Points can be mapped on a Memory Polling Area for the specific bits and specific words.

Modbus Panel Guide

When defining the **# of Items**, keep in mind that the size must be no bigger than the payload size for a single packet. If it is larger, it will require multiple packets. Ethernet has an MTU (Maximum Transmission Unit) of 1500 bytes. Subtracting off the headers, the TCP payload is about 1400 bytes which is 700 words. A Memory Polling Area larger than this payload size requires TCP/IP to break it up into multiple messages. Therefore, **# of Items** should be less than or equal to 700 words.

Groups:		Points:			
Name		Name	Address Type	Address	Threshold

Groups – table; lists the groups by which the points are organized; each group is a child node of the panel node in the Project Node Tree; use the buttons below the table to add and delete groups; click on the group in the table to edit the name of the group

Points – table; lists the memory mappings of different types of points to places in the memory polling areas

Name – edit box; the name for the point

Address Type – drop-down menu (default: *Coil*); the address area in which this point exists

Address – numeric; the offset in the **Address Type** area; offset is zero-based. The first word is at offset 0; the second word is at offset 1.

Threshold – numeric (default: 0); the point is set high when the value is at or above this number

Modbus Panel Guide

Add

- button; add a row to the **Points** table

Add Multiple

- button; add more than one point to the table; opens the **Add Multiple** dialog

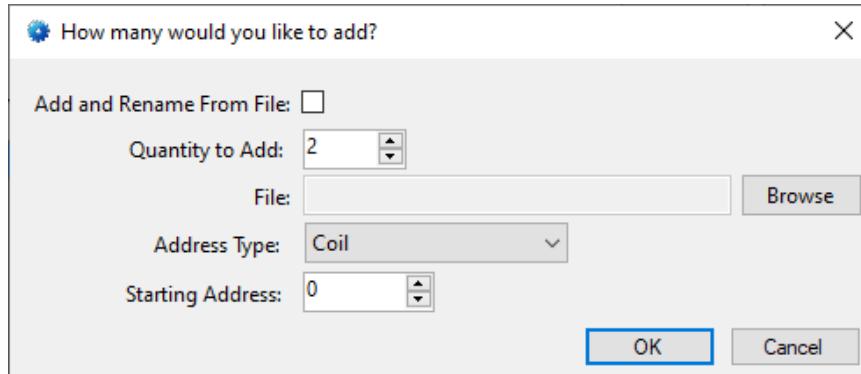


Figure 32 - The Add Multiple dialog

Delete

- button; delete the highlighted point

4.3.2 Modbus Panel Node Properties

Right-click on the Modbus panel node to open the properties. The **Quick Config** dialog is the recommended means to manage the panel and its child node properties, but Modbus allows the user to define transaction types and what each transaction type does.

Note: *Transaction Types only apply to devices that send status updates as a group of registers.*

Modbus Panel Guide

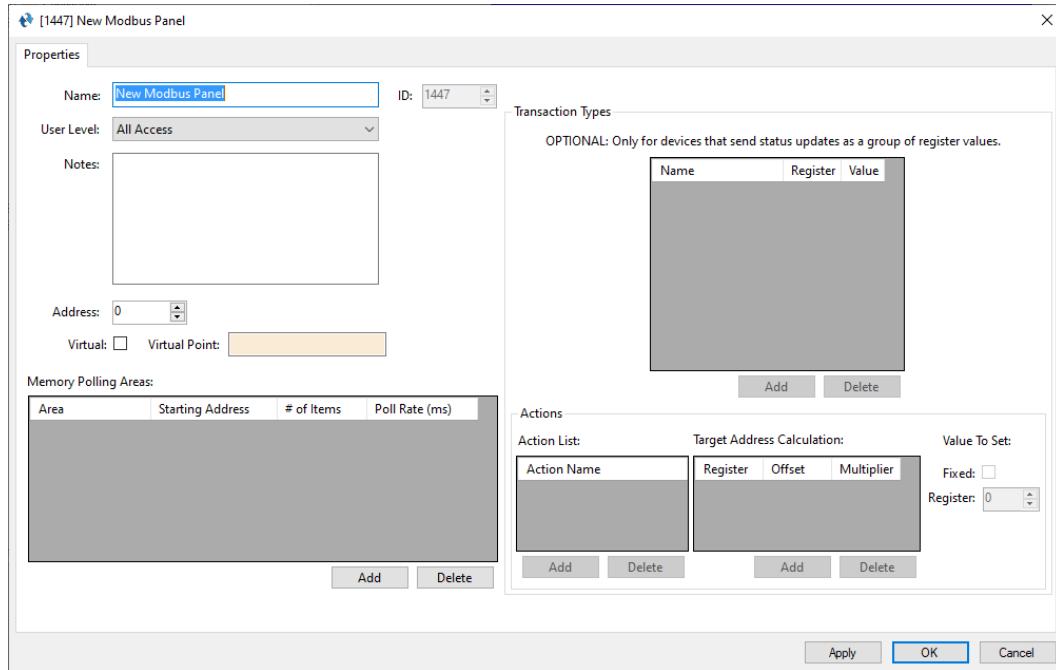


Figure 33 - Modbus node properties dialog

Name – edit box; the name for the node; the name doesn't have to be unique

ID – numeric (disabled); the unique identifier of this node; generated by the software

User Level – drop-down menu (default: All Access); the User Level a user must possess to open the properties for this node

Notes – multiline edit box; any notes the user may have for the node

Address – numeric (default: 0)

Virtual – check box; when checked the panel is virtualized, allowing the driver to be brought online without having the physical panel available

Virtual Point – drop box; this I/O Point will be set when the panel is virtualized, clear when it is not

Modbus Panel Guide

4.3.2.1 Memory Polling Areas Properties

Memory Polling Areas:			
Area	Starting Address	# of Items	Poll Rate (ms)
<input type="button" value="Add"/> <input type="button" value="Delete"/>			

Memory Polling Areas – table; lists the regions of PLC memory that are polled and mapped

Area – the memory area on the PLC

Starting Address – numeric; the offset in the memory are to begin copying; zero-based, the first word/bit/etc. at address 0

of Items – numeric; the number of units to copy

Polling Rate – numeric (default: 250); microseconds; how often to request the data for this area

When defining Memory Polling Areas, please consider network efficiency. Each Memory Polling Area is a separate poll command.

For the sake of efficiency, define areas such that a minimum number of poll commands and responses are required.

When defining the **# of Items**, keep in mind that the size must be no bigger than the payload size for a single packet. If it is larger, it will require multiple packets. Ethernet has an MTU (Maximum Transmission Unit) of 1500 bytes. Subtracting off the headers, the TCP payload is about 1400 bytes which is 700 words. A Memory Polling Area larger than this payload size requires TCP/IP to break it up into multiple messages. Therefore, **# of Items** should be less than or equal to 700 words.

4.3.2.2 Transaction Types Properties

When the Modbus panel sends status updates as a group of registers, the Transaction Types must be defined using the **Transaction Types** properties.

Modbus Panel Guide

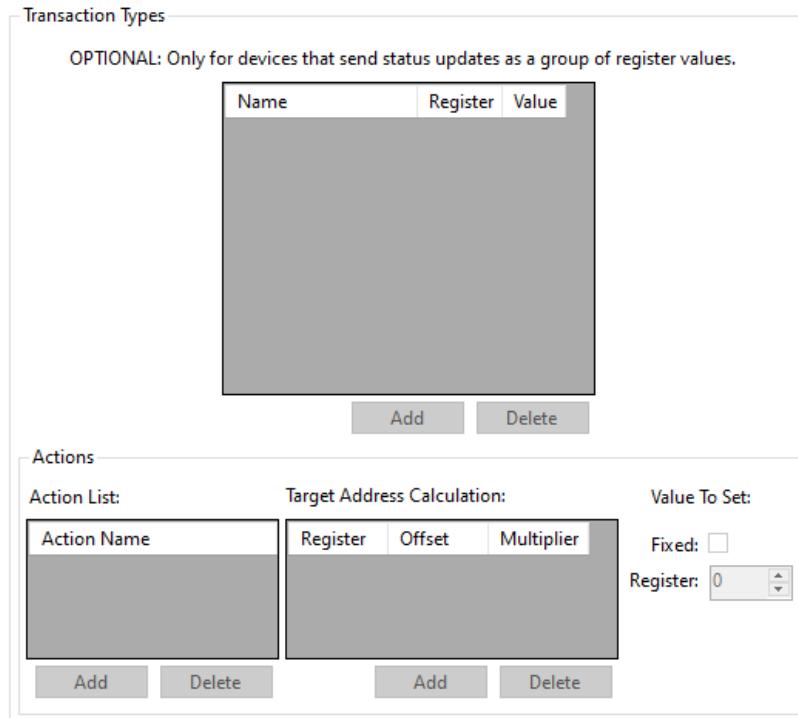


Figure 34 - Transaction Types properties

Note: To active the Add and Delete buttons for Transaction Types, a register polling area must be selected.

Transaction Types - table; a list of the transaction types defined for the selected register Memory Polling Area. Each row is a different transaction type. Each transaction type has a **Name**, a **Register**, and a **Value**. When the specified register in the selected Memory Polling Area equals the designated value, Intelli-Site will execute each of the defined actions for the transaction type.

Action List - table; a list of the actions to execute for the selected row of the **Transaction Types** table

Target Address Calculation – table; the method to calculate the address whose value will be modified. The value in the **Register** is multiplied by the **Multiplier** then the **Offset** is added to find the target address.

Modbus Panel Guide

Value To Set – two fields that define the value to assign to the target address

Fixed – checkbox; indicate how to interpret the **Value** field

Value – numeric; an integer used in assigning a value to the target address based on the **Fixed** field. When **Fixed** is checked, the calculated target address is set **Value**. When **Fixed** is clear, the number in **Value** is treated as an address. The current value of said address is assigned to the target address.

Note: A Point at the address designated by the Value field must exist. It would be pointless otherwise. This address does not have to be part of any polling area. It can be at an address that is beyond the # of Items or in a memory polling area that is not in the Memory Polling Areas table.

See [Transaction Types Programming](#) for an example illustrating using transaction types for an intercom system.

4.3.3 Modbus Child Nodes

The Modbus child nodes are the **Alarms** node and the group nodes. When a group is added in **Quick Config**, a node is added to the panel node in the Tree for that group. Any points added to the group in the **Quick Config** are added to the Tree as child nodes of the group node. The alarms are children of the **Alarms** node.

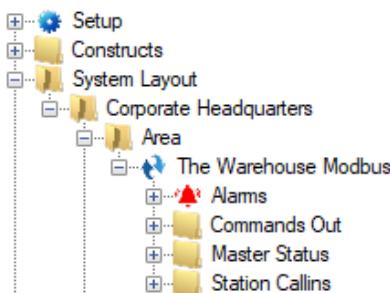


Figure 35 - Sample panel node and the Group nodes

4.3.3.1 Alarms

The alarm for the Modbus is located under the **Alarms** node. There is one (1) alarm, **Panel Status**.

Modbus Panel Guide

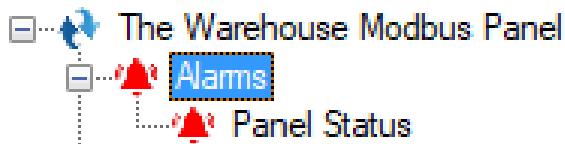


Figure 36 - Modbus panel Alarms child nodes

The **Panel Status** point is high when the panel is offline (assuming the driver is online, and the panel is not virtualized).

4.3.3.2 Point Nodes

The point nodes are the children of the group nodes. While they are referred to as points in general, they are specifically *Points* or *Values*. A *Point* can only be 0 or 1, on or off, high or low. A *Value* is an integer from 0 to 65535. The **Memory Polling Area** determines if a point is a *Point* or a *Value*. *Values* are in the *Input Register* or the *Holding Register Memory Polling Areas*. *Points* are in the *Coil* or *Input Status Areas*. Value nodes can also have child nodes (a.k.a. sub-nodes) added to them in **Design View**.

4.3.3.2.1 Point Node Properties

Right-click on the point node to open the properties. These fields are those specifically about the point. The **Quick Config** dialog is the recommended means to manage the point node properties. The point node is a special kind of I/O Point node called GenProto node. It has all the fields of an I/O Point with a few additional fields specific to the point. We will only discuss the additional fields. For an explanation of the I/O Point, see section [9.3 The Properties Dialog](#) of the Intelli-Site User's Guide.

Modbus Panel Guide

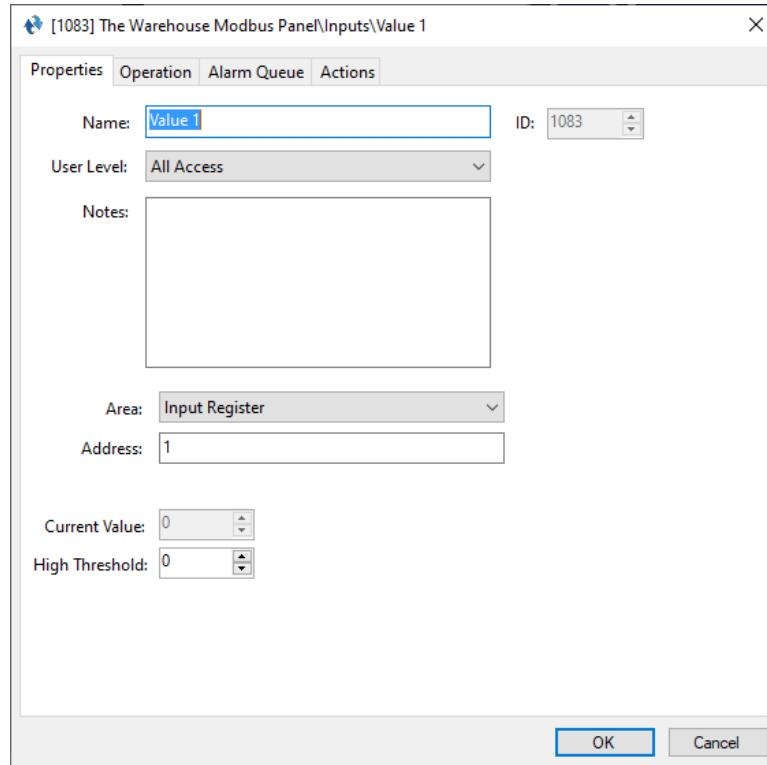


Figure 37 - Point node properties dialog

Area – drop-down menu; the memory polling area in which this point is located

Address – edit box; the address of the point in the **Area**; an integer defines a word address; a decimal number is a word and bit address

Area – drop-down menu (*Coil*, *Input Status*, *Input Register*, *Holding Register*); the Memory Polling Area in which this point is located; the node is a bit if equal to *Coil* or *Input Status*, else the node is a word

Current Value – numeric; the current value of the point; disabled when the **Type** is *Point*

High Threshold – numeric; when the **Current Value** is equal to or greater than this number, the point is high; disabled when the **Area** is *Coil* or *Input Status*

Modbus Panel Guide

4.3.3.2.2 Value Sub-Nodes

When a point node's **Area** is *Input Register* or *Holding Register*, child nodes can be added to it in  **Design View**. Like their parents, these sub-nodes are special GenProto nodes. They are like counter value nodes in that their state is tied to the current value of their parent node. And like a counter value, when a value sub-node is the target of a **SendCommand** action, the parent node is acted on by the value child node if possible.

To better understand these nodes, let's examine the properties dialog. Because a value sub-node is a GenProto node, we will only discuss the additional fields specific to a value sub-node. For an explanation of the rest of the fields and tabs, see section [9.3 The Properties Dialog](#) of the Intelli-Site User's Guide.

First, though, one needs to add a value child node. Locate the desired point node in the Tree. Right-click on it and select either *Add Node*, *Add Multiple...*, or *Add Node and Edit*.

Note: *If the point node is not a Value node, these menu options are disabled. In fact, Add Multiple... is not present.*

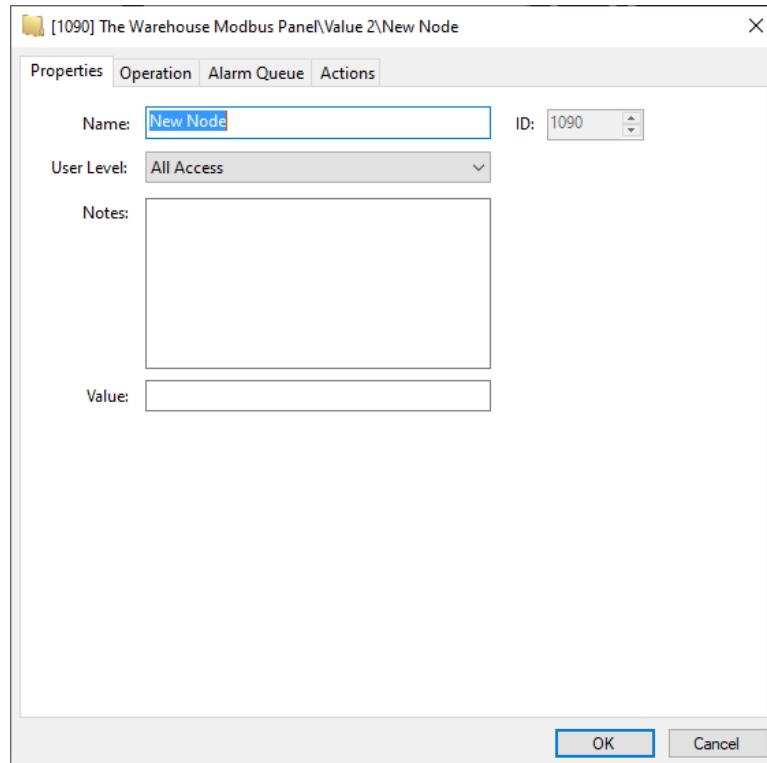


Figure 38 - Value Sub-Node properties dialog

Modbus Panel Guide

The Properties tab has the standard properties of **Name**, **ID**, **User Level**, and **Notes**. The only new field is **Value**. It is an optional field. If it is empty, this node is treated as group node. This means even value sub-nodes can have child nodes. There is no limit to the depth of child nodes. This gives the user flexibility in organizing the sub-values. Even if the value sub-node has a number in the **Value** field, it can still have child nodes of its own.

The **Value** field is a text field that can have one of three (3) types of data in it:

- a signed integer
- an unsigned integer
- B0-B15

4.3.3.2.2.1 Signed Value Sub-Node

When the number is signed, it can be used to increment or decrement the parent Value node's value using **SendCommand**. For example, the sub-node **Value** field is -2. **SendCommand** targeting this sub-node will decrement the parent Value node by 2. If the sub-node **Value** field is +1, then **SendCommand** will increment the parent Value node by 1.

4.3.3.2.2.2 Unsigned Value Sub-Node

When the **Value** field is unsigned, it is treated much like a counter value. When the parent Value node equals this sub-node, this sub-node is in the high state. The parent Value node can be set to the value of this sub-node using **SendCommand**.

4.3.3.2.2.3 b0-b15 Value Sub-Node

Each register Value point is a word in length. A word is 16 bits. Each bit can be either 0 or 1. Intelli-Site allows each bit to be monitored individually. Sub-nodes whose value is b0, b1, b2, b3, b4, b5, b6, b7, b8, b9, b10, b11, b12, b13, b14, or b15 (case-insensitive) correspond to the bits of the Value point.

Modbus Panel Guide

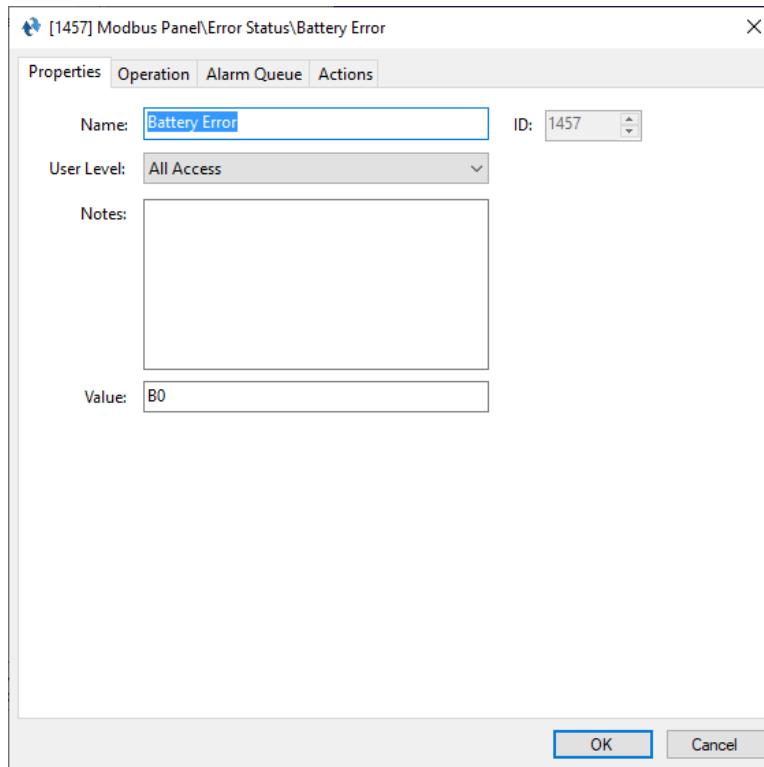
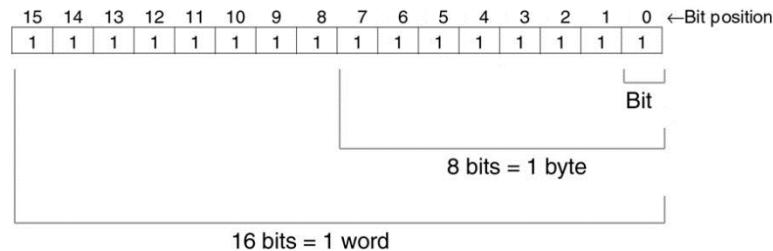


Figure 39 – Example Bit sub-node properties dialog

When the bit in the Value point is 1, the 'b'-node is 1. When the bit equals 0, the 'b'-node equals 0.



For example, the decimal number 10 is 0000000000001010 in binary. Using the above image as a guide, we see that bit 1 (b1) and bit 3 (b3) are 1 and the other bits are 0. Intelli-Site will set on the sub-nodes with the **Value** field equal to b1 and b3. The rest of the bit sub-nodes will be set off.

The internet has many sites that convert decimal to binary numbers. Check out [RapidTables decimal to binary converter](#).

Note: *It is not necessary to add all 16 of the bit sub-nodes. Only add the ones important to the Project.*

Modbus Panel Guide

4.4 Project Programming

The point nodes can be used in project programming in evaluation grids and in action grids. The point nodes can also be dragged and dropped onto screens to automatically create screen objects.

4.4.1 Evaluation Grids and Action Grids

The panel and point nodes can be used in evaluation and action grids.

4.4.1.1 The Panel Node

When a panel node is used in the evaluation grid, there are two **Selection** options: *Virtual* and *Driver Online*.

Point	Selection	Qual.	Oper.
[2683] The Warehouse Modbus Panel	Virtual	▼	▼
	Driver Offline	▼	▼
	Not Available	▼	▼
		▼	▼

The panel node can be used in action grids as the target of the following actions:

- VirtualizePanel
- UnvirtualizePanel

4.4.1.2 The Point Nodes

The point nodes are I/O Points. Therefore, they can be used in evaluation grids and action grids.

4.4.1.3 The Sub-nodes of Point Nodes

The sub-nodes of Point Nodes of **Type** value can be used in action grids to modify the Point Node's value if the Memory Polling Area is writeable.

4.4.1.3.1 Sub-nodes with Unsigned Values

To set the parent Value node to equal the sub-node, use **SendCommand** and target the sub-node.

Modbus Panel Guide

Action	Target
1 Play	[962] Click.wav
2 SendCommand	[2691] The Warehouse Modbus Panel\Ho...
3	
4	
5	
6	

Mouse Down Mouse Up Active Inactive Mouse Enter Mouse Leave

Insert Delete

Figure 40 - Example Action grid using an unsigned sub-value node

If the memory polling area at the address of Output 4 is writeable, it will be set to the **Value** of sub-node 1.

4.4.1.3.2 Sub-nodes with Signed Values

Below are the properties of a sub-node with a signed **Value** field.

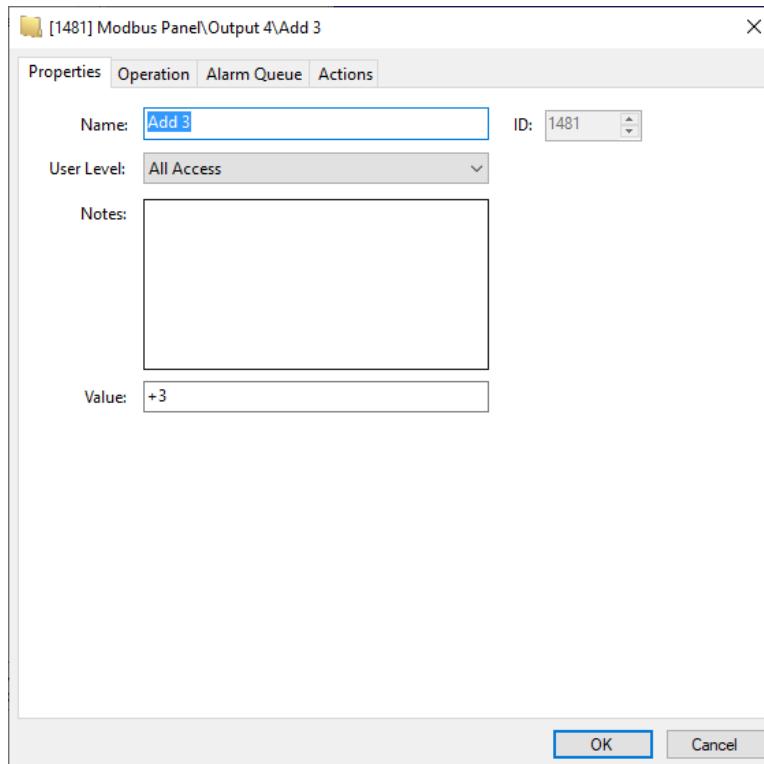


Figure 41 - Example sub-node with a signed Value field

When the **Value** field is signed (e.g., +1 or -2), this node can be used to increment or decrement the parent Value node. Use **SendCommand** and target this node in any action grid.

Modbus Panel Guide

The screenshot shows a software interface titled 'Modbus Panel Guide'. At the top is a toolbar with icons for 'File', 'Edit', 'View', 'Insert', 'Delete', 'Run', 'Stop', 'Help', and 'Exit'. Below the toolbar is a title bar with the text 'Modbus Panel Guide'.

The main area is a 'Action grid' window. It has two columns: 'Action' and 'Target'. The 'Action' column contains rows numbered 1 to 6. Row 1 is highlighted in blue and contains the text 'Play'. Row 2 contains 'SendCommand'. Rows 3 through 6 are empty. The 'Target' column contains the text '[962] Click.wav' for row 1 and '[1481] Modbus Panel\Output 4\Add 3' for row 2. Below the grid is a status bar with buttons for 'Mouse Down', 'Mouse Up', 'Active', 'Inactive', 'Mouse Enter', and 'Mouse Leave'.

	Action	Target
1	Play	[962] Click.wav
2	SendCommand	[1481] Modbus Panel\Output 4\Add 3
3		
4		
5		
6		

Figure 42 - Example Action grid using a signed sub-node

In this example, if the memory polling area at the address of Output 4 is writeable, the value of Output 4 will be incremented by 3 because the value of **Add 3** is +3.

4.4.2 Automatically Created Screen Objects

The point nodes can be used as evaluation points for the evaluation grids of screen objects. This allows the user to monitor and control the state of each point.

Instead of programming the screen objects by hand, drag and drop the nodes onto the screen to automatically create screen objects that are programmed based on the type of point it is.

4.4.2.1 Point Node Screen Objects

A node with an **Address Type** of either *Coil* or *Input Status* is a Point node. These nodes have two values. They are either 0 or 1, on or off, set or clear.

Dragging the node to the screen creates a button type screen object. The screen object is programmed to display the current state of the point. If the **Address Type** is *Coil*, clicking on the button will set the point on or off.



Figure 43 - Sample screen object created from a Point node

The following examines the properties of the button screen objects and the programming of the different states.

Modbus Panel Guide

4.4.2.1.1 Input Status Screen Object Properties

When a node with **Address Type** of *Input Status* is dragged and dropped onto the screen a button screen object is created. It has two states.

4.4.2.1.1.1 State 0 – Normal

State 0 is the base state. It displays when the evaluation grid on the other state does not evaluate to true. Stated more simply, when the point is off, this state displays.

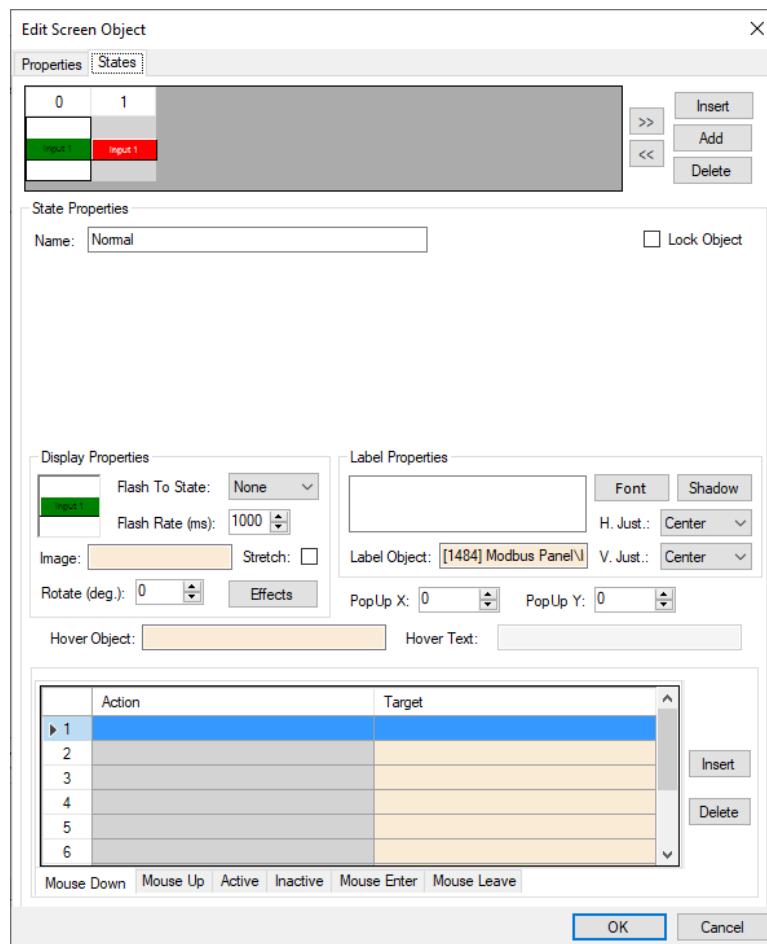


Figure 44 - Sample Input Status point button properties dialog: State 0

Notice the **Action** grid. The Input Status memory polling area is not writeable. Therefore, the state of the point cannot be set or cleared. Clicking on this screen object does nothing.

Modbus Panel Guide

4.4.2.1.1.2 State 1 - Alarm

This state displays when the point is high. See the evaluation grid. The evaluation grid is programmed to return true when the point's state evaluates as high.

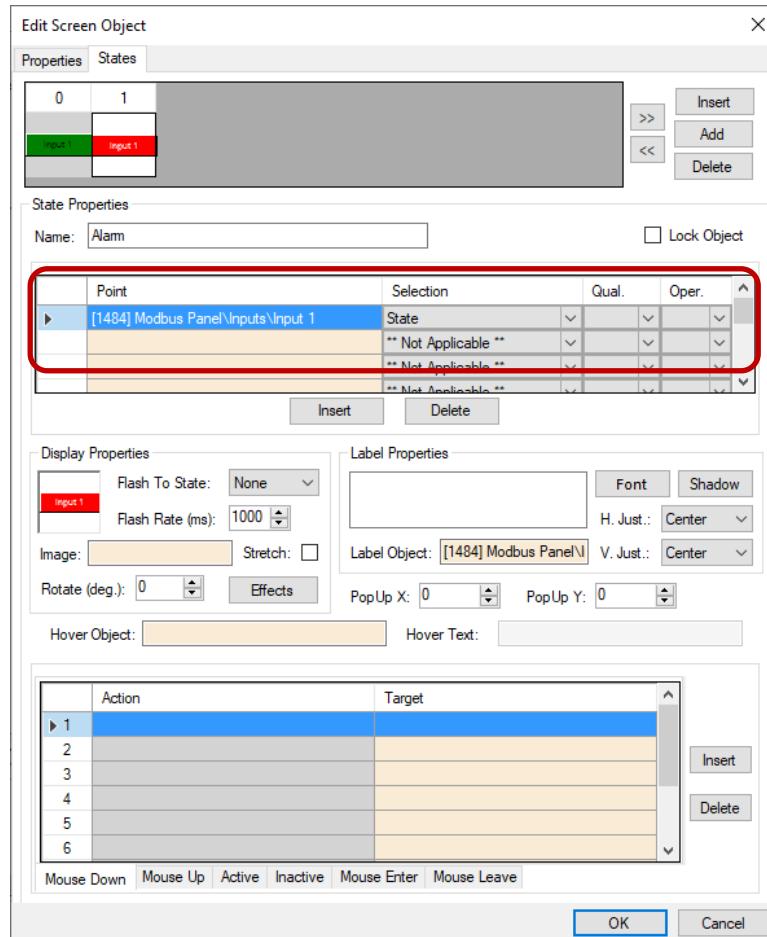


Figure 45 - Sample Input Status point button properties dialog: State 1

Notice the **Action** grid. It's empty. Clicking on this screen object does nothing.

4.4.2.1.2 Coil Screen Object Properties

When a node with **Address Type** of *Coil* is dragged and dropped onto the screen a button screen object is created. It has two states.

Modbus Panel Guide

4.4.2.1.2.1 State 0 – Normal

State 0 is the base state. It displays when the evaluation grid on the other state does not evaluate to true. Stated more simply, when the point is off, this state displays.

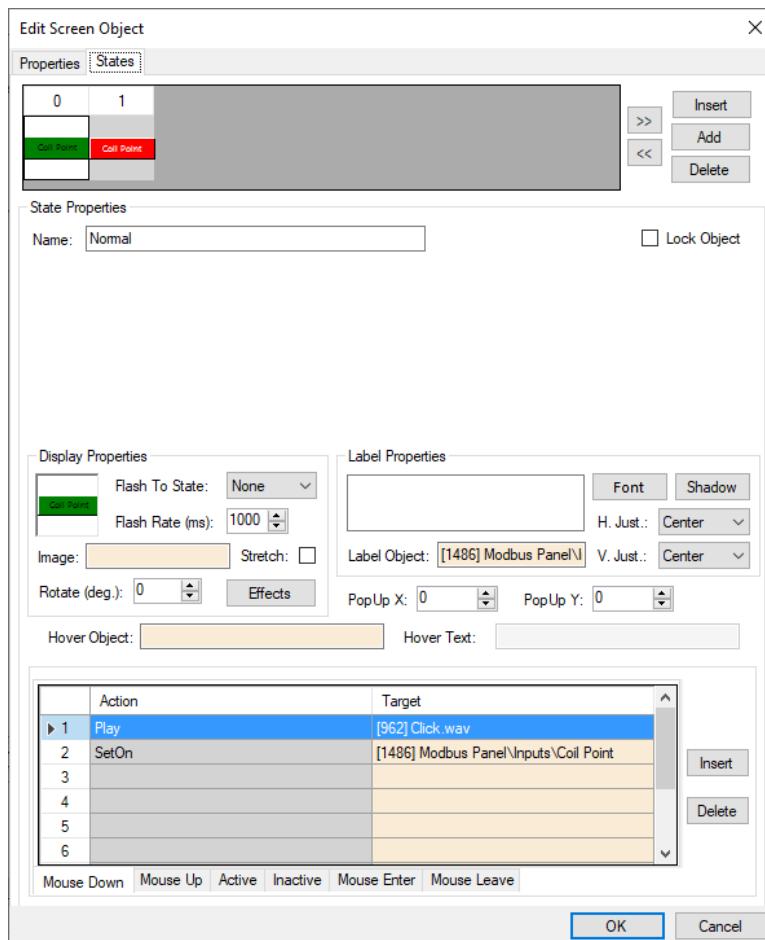


Figure 46 - Sample Coil point button properties dialog: State 0

Notice the **Action** grid. When this button is clicked in this state, a **SetOn** command is sent to the Modbus panel for this point. The panel will set the point if applicable. Intelli-Site will receive the updated state of the point in the next poll.

4.4.2.1.2.2 State 1 - Alarm

This state displays when the point is high. See the evaluation grid. The evaluation grid is programmed to return true when the point's state evaluates as high.

Modbus Panel Guide

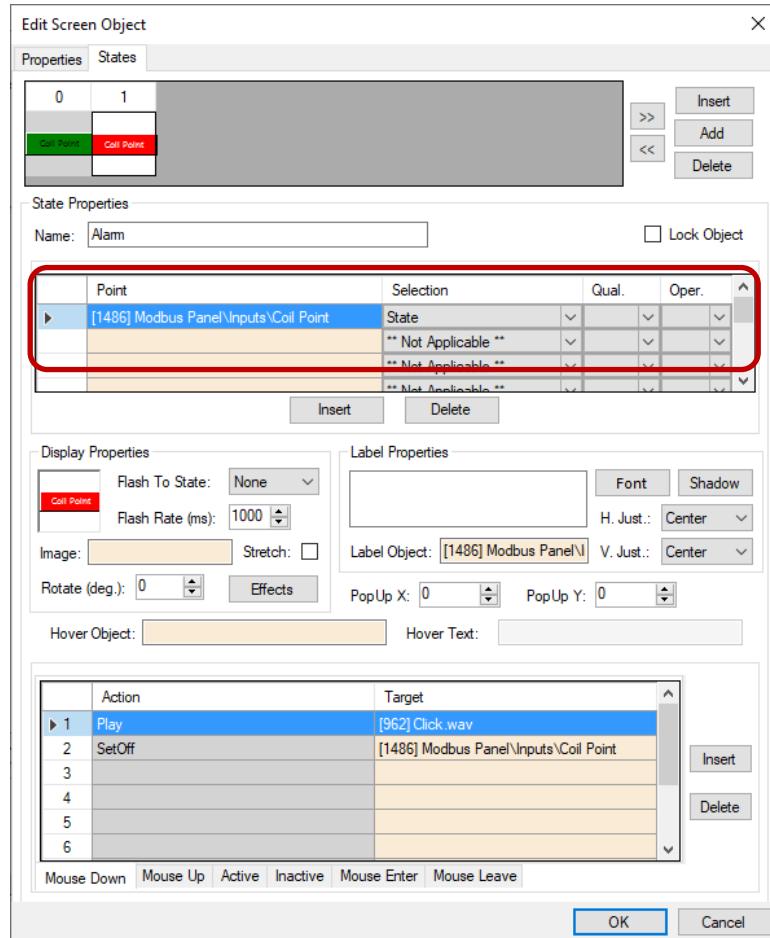


Figure 47 - Sample Coil point button properties dialog: State 1

Notice the **Action** grid. When this button is clicked in this state, a **SetOff** command is sent to the PLC panel for this point. The panel will turn off the point if applicable. Intelli-Site will receive the updated state of the point in the next poll.

4.4.2.2 Value Screen Objects

When the **Address Type** is *Input Register* or *Holding Register*, the node is a Value node. When it is dragged and dropped to the screen, a simple screen object is created that displays the current value of the object.

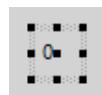


Figure 48 - Sample screen node created from a register node

Modbus Panel Guide

The screen object is programmed to display the current value of the point.

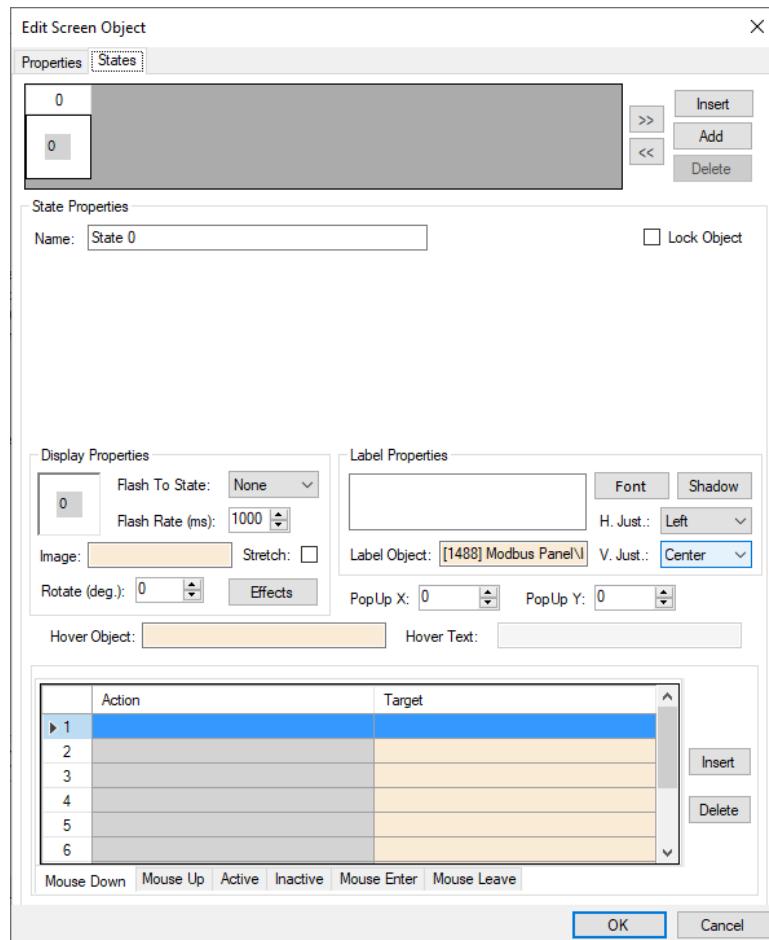


Figure 49 - Sample register screen object properties dialog: State 0

Values cannot be set on or off; they can only be displayed. Therefore, the **Action** grid has no programming.

4.4.2.3 Sub-Value Node Screen Objects

When a sub-value node is dragged and dropped onto a screen, a button screen object is created. If the **Value** field is signed a single state button is created. If the **Value** field of the node is unsigned or begins with "b", a two-state button is created.

Modbus Panel Guide

4.4.2.3.1 Signed Sub-Value Node Screen Object

When a sub-node of a Value node whose **Value** field is signed is dragged and dropped on the screen, a single state button screen object is created.

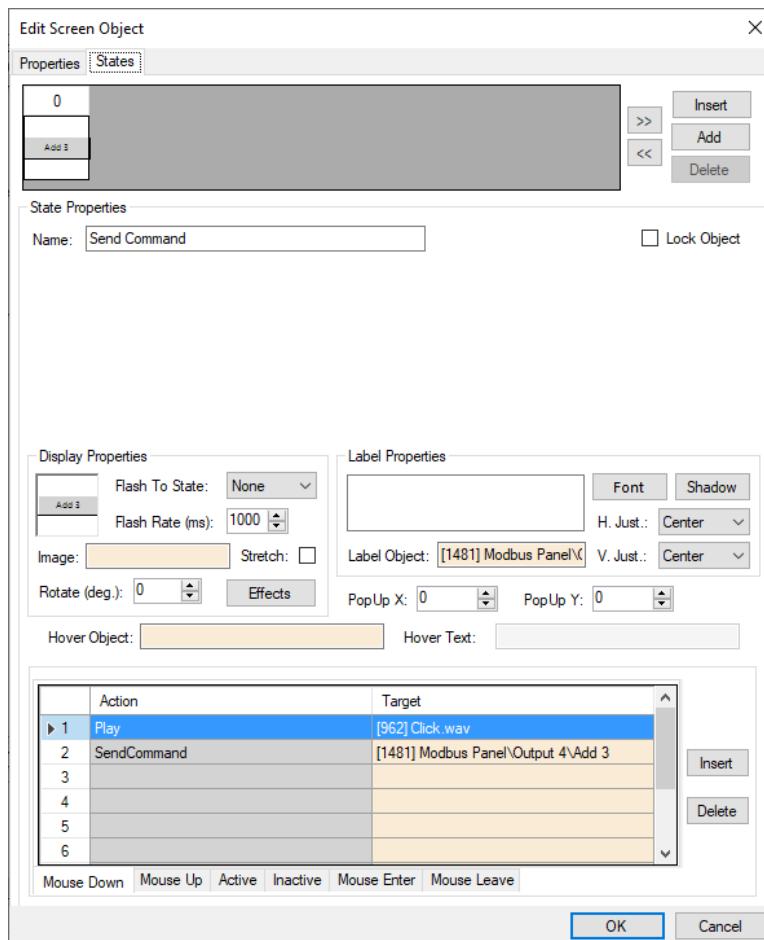


Figure 50 - Sample signed sub-value node screen object properties dialog

Notice the **Action** grid. When this button is clicked, a **SendCommand** action targeting this sub-value node is executed. If the memory polling area for the parent Value point is writeable, the value will be modified by the signed value in the sub-value node and sent to the panel. It will be incremented if the sign is positive and decremented if the sign is negative. Intelli-Site will receive the updated state of the point in the next poll.

Modbus Panel Guide

4.4.2.3.2 Bit Sub-Value Node Screen Object

When a sub-node of a Value node whose **Value** field is a bit (b0-b15) is dragged and dropped on the screen, a single state button screen object with two states is created.

4.4.2.3.2.1 State 0 – Off

State 0 is the base state. It displays when the evaluation grid on the other state does not evaluate to true. Stated more simply, when the point is off, this state displays.

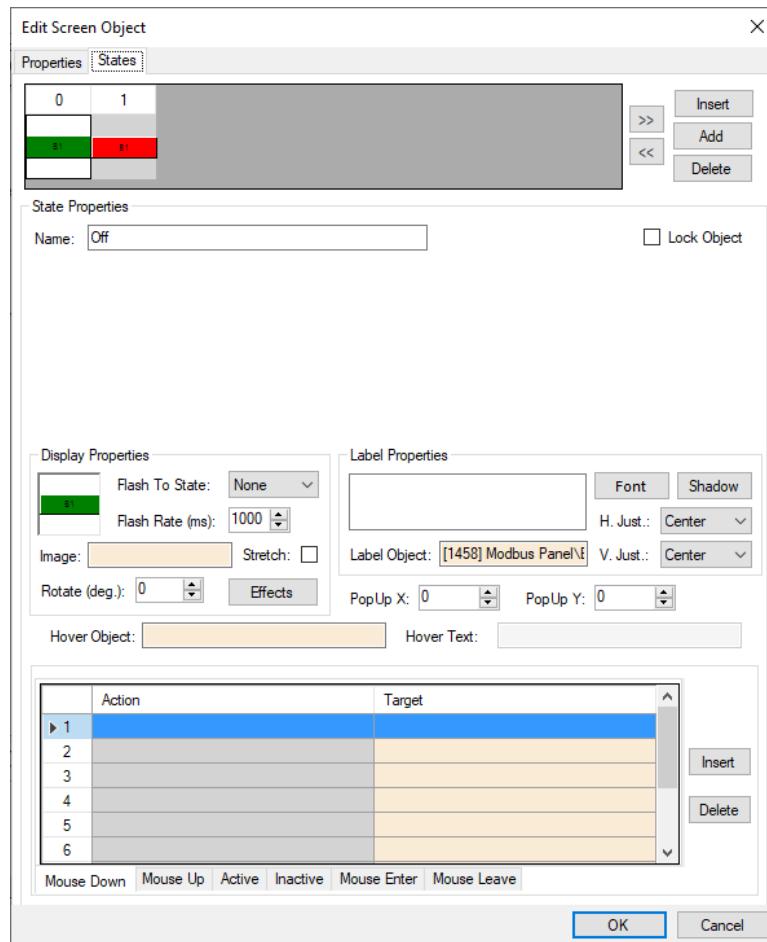


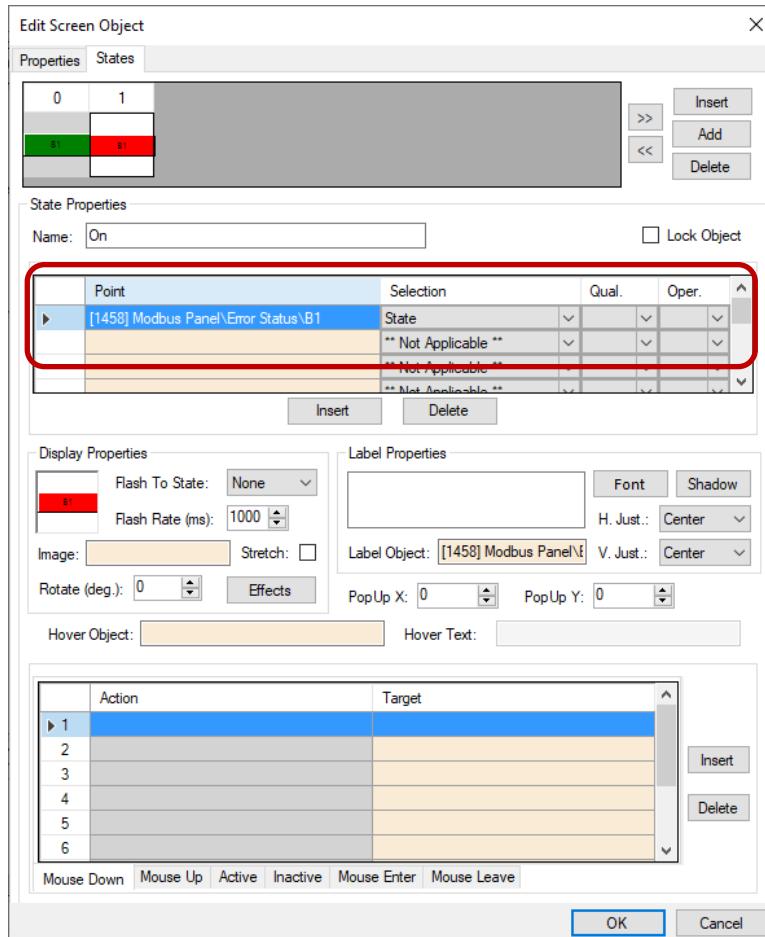
Figure 51 - Sample bit sub-value node screen object properties dialog: State 0

Notice the **Action** grid. There are no actions; therefore, clicking on this button does nothing.

Modbus Panel Guide

4.4.2.3.2.2 State 1 - On

This state displays when the point is high. See the evaluation grid. The evaluation grid is programmed to return true when the point's state evaluates as high.



**Figure 52 - Sample bit sub-value node screen object properties dialog:
State 1**

Notice the **Action** grid. There are no actions when the screen object is in this state. Clicking on this button does nothing.

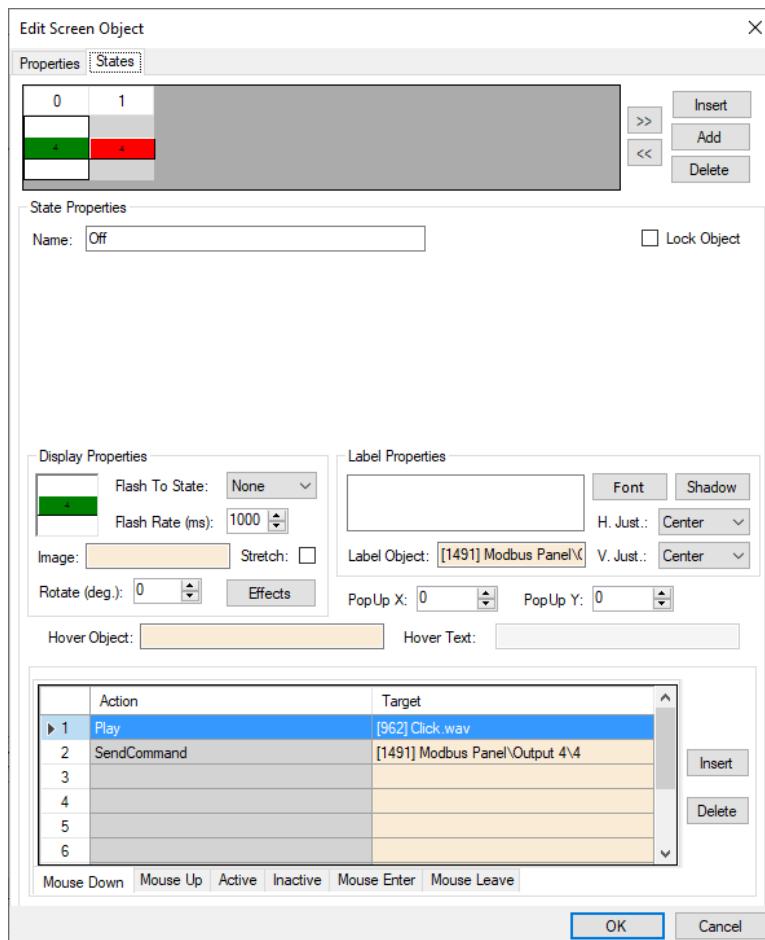
4.4.2.3.3 Unsigned Sub-Value Node Screen Object

When a sub-node of a tag node whose **Value** field is unsigned is dragged and dropped on the screen, a two-state button screen object is created.

Modbus Panel Guide

4.4.2.3.3.1 State 0 – Off

State 0 is the base state. It displays when the evaluation grid on the other state does not evaluate to true. Stated more simply, when the point is off, this state displays.



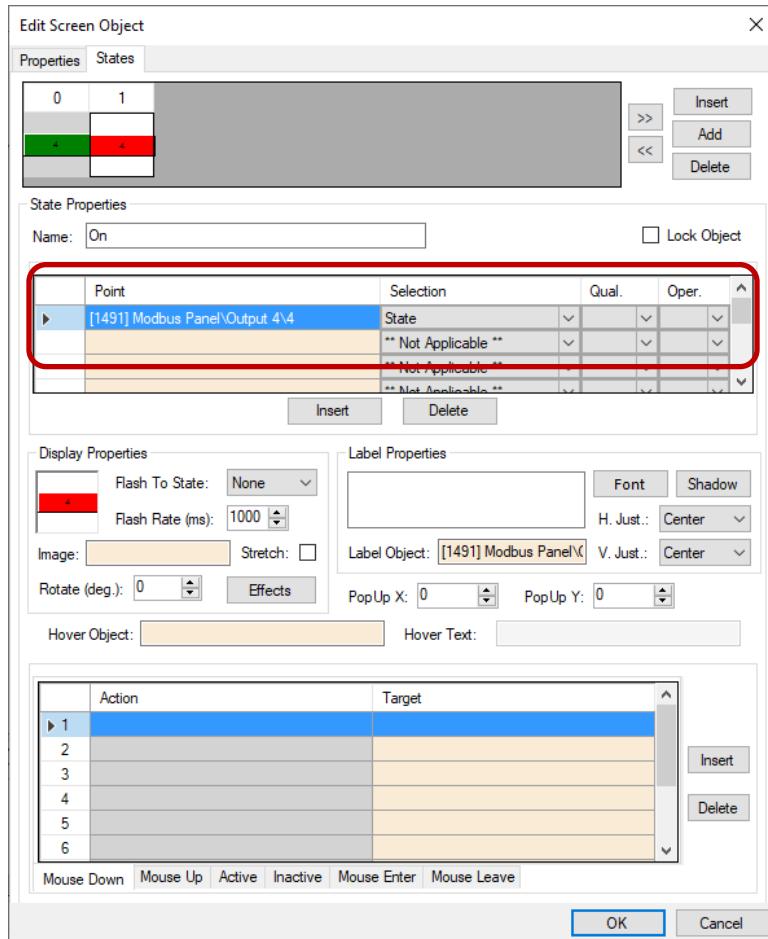
**Figure 53 - Sample unsigned sub-value node screen object properties dialog:
State 0**

Notice the **Action** grid. When this button is clicked in this state, a **SendCommand** action targeting this sub-value node is executed. If the memory polling area for the parent Value point is writeable, the value of the parent Value node will be set to the value in the sub-value node. The new value for the parent Value node is sent to the panel. Intelli-Site will receive the updated state of the point in the next poll.

Modbus Panel Guide

4.4.2.3.3.2 State 1 - On

This state displays when the point is high. See the evaluation grid. The evaluation grid is programmed to return true when the point's state evaluates as high.



**Figure 54 - Sample unsigned sub-value node screen object properties dialog:
State 0**

Notice the **Action** grid. There are no actions when the screen object is in this state. Clicking on this button does nothing.

4.4.3 The Panel Control Screen

The Panel Control Screen is a popup screen that is added when the panel is added. It can be displayed in **Hardware Management View** by clicking on the panel icon.

Modbus Panel Guide

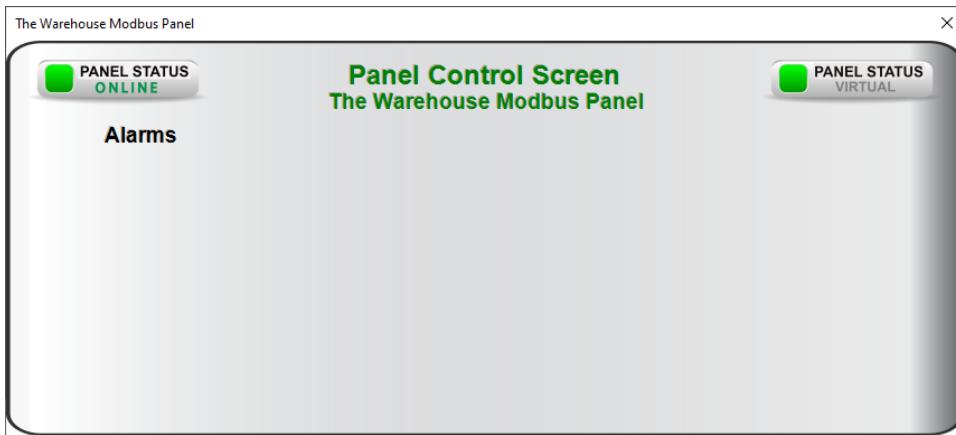


Figure 55 - Panel Control Screen

When an Modbus panel is added, there are no points. Therefore, the Panel Control Screen has no points. You must program the screen to display the points and values once the panel is programmed.

To find and modify the Panel Control Screen, you'll need to be in **Design View**. The Panel Control Screens are found by expanding **Screen Control->Popup Screens->Area Popup Screens**.



Figure 56 - Panel Control Screens in the Project Node Tree

If there is only one or two Modbus panels, finding the corresponding Panel Control Screen is relatively simple. Open the properties dialog of the Panel Control Screen and examine **Base Object** field.

Modbus Panel Guide

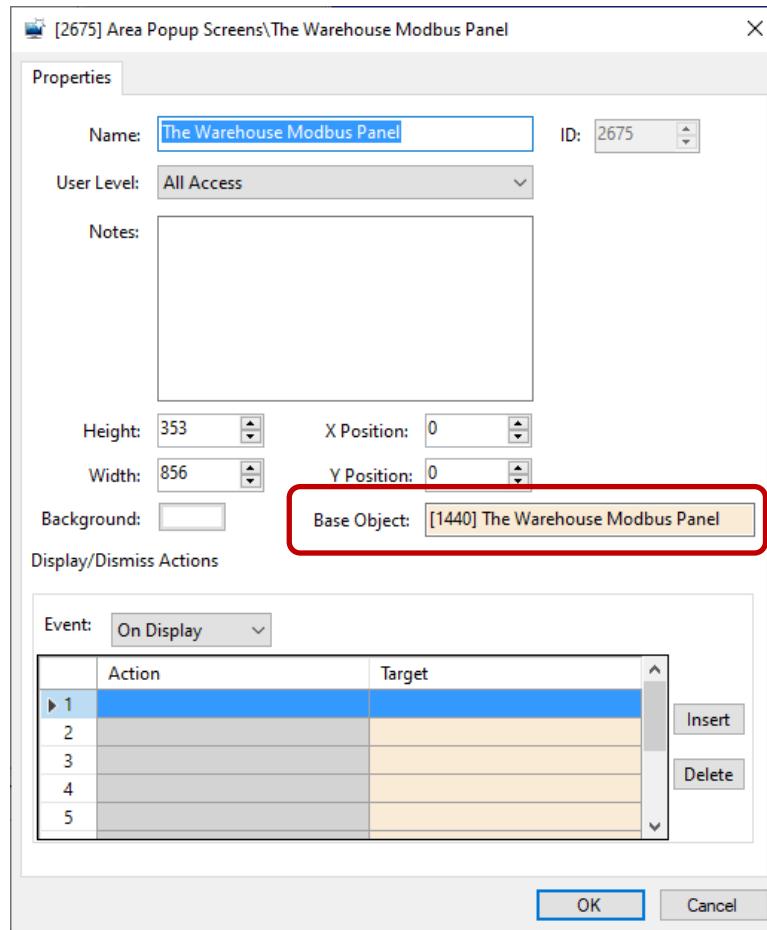


Figure 57 - Panel Control Screen properties dialog

Another way to locate the corresponding Panel Control Screen is to use *Find...->All objects referencing this node* on the panel node itself.

Modbus Panel Guide

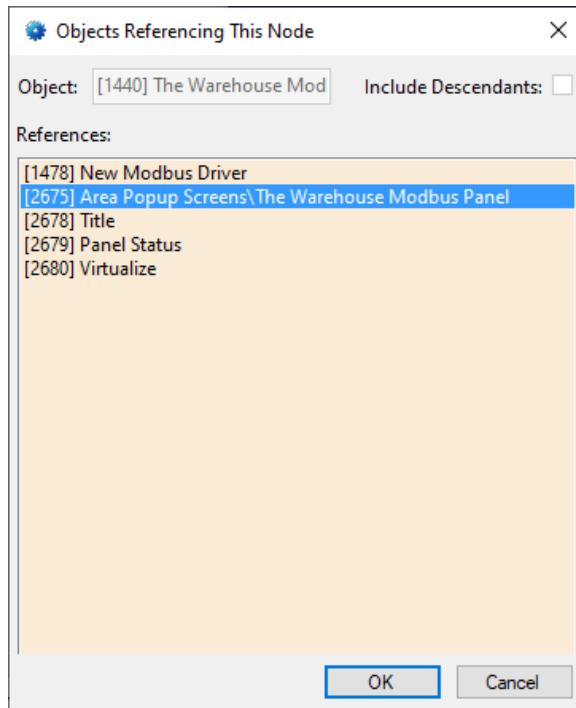
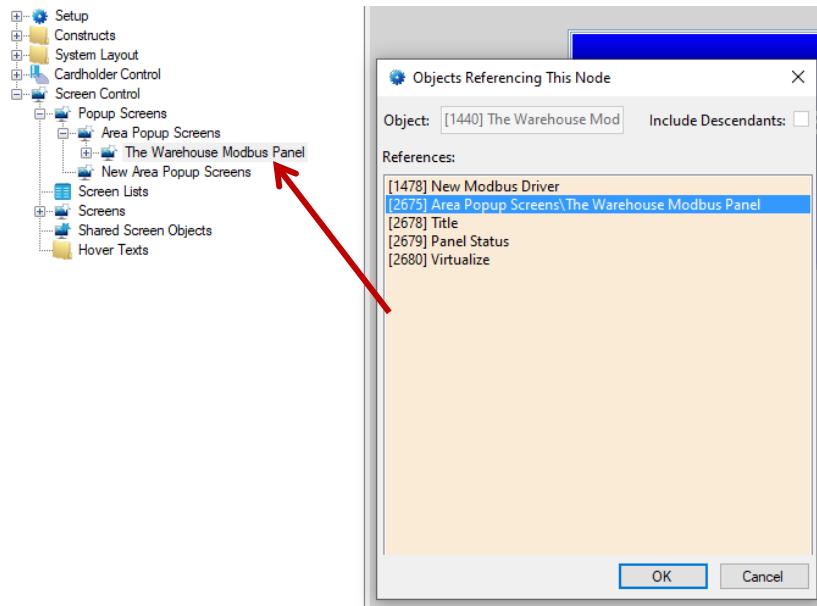


Figure 58 - Find...->All objects referencing this node dialog

Right-click on the Panel Control Screen in the **References** and select *Go To/Find In Tree*. The software will expand the Tree and select the Panel Control Screen node.

Modbus Panel Guide



The highlighting is grey because it does not have focus. Cancel out of the **Objects Referencing This Node** dialog and the highlighting will be blue. Double-click on the Panel Control Screen node to open the screen for programming.

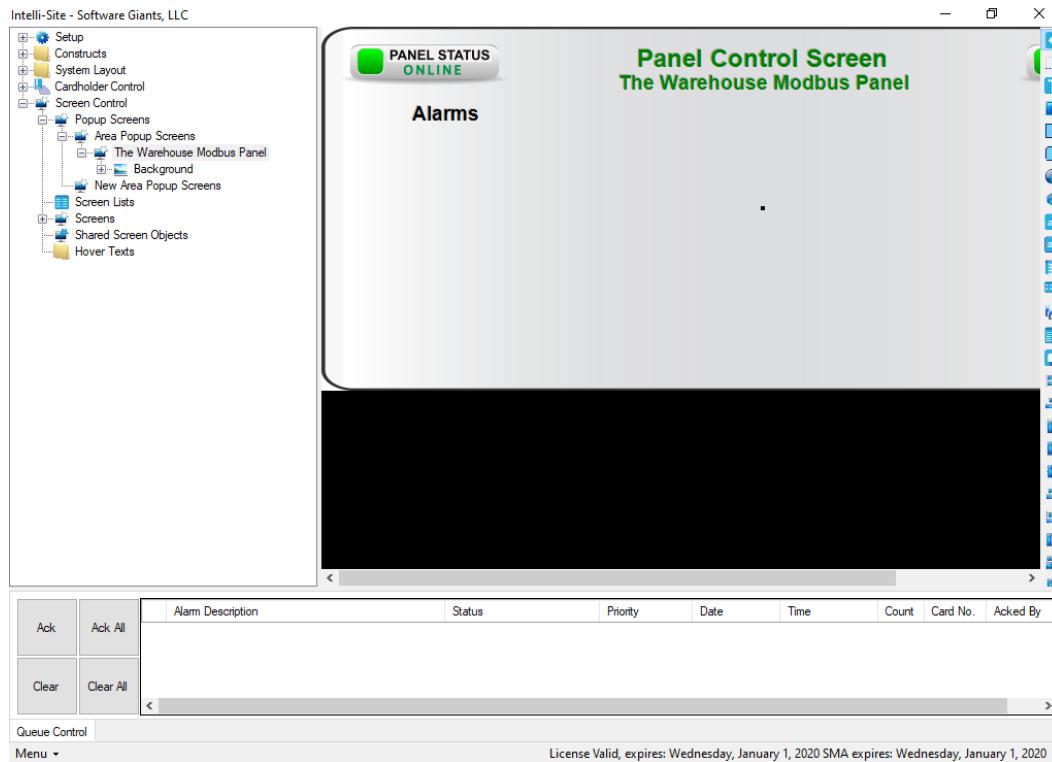


Figure 59 - Panel Control Screen in Design View

Modbus Panel Guide

Add the points that are important to watch in **Hardware Management View**.

4.4.4 Transaction Types Programming

Section [Transaction Types Properties](#) explains the various fields used to define the actions Intelli-Site must take to update the appropriate nodes on a status update by a Modbus panel. These transactions and actions differ from installation to installation and what the panel is being used for. Let's now consider a Modbus panel as an intercom system.

In this intercom system there are two (2) master stations and 193 substations numbered 101-293. These were added using the **Quick Config** dialog.

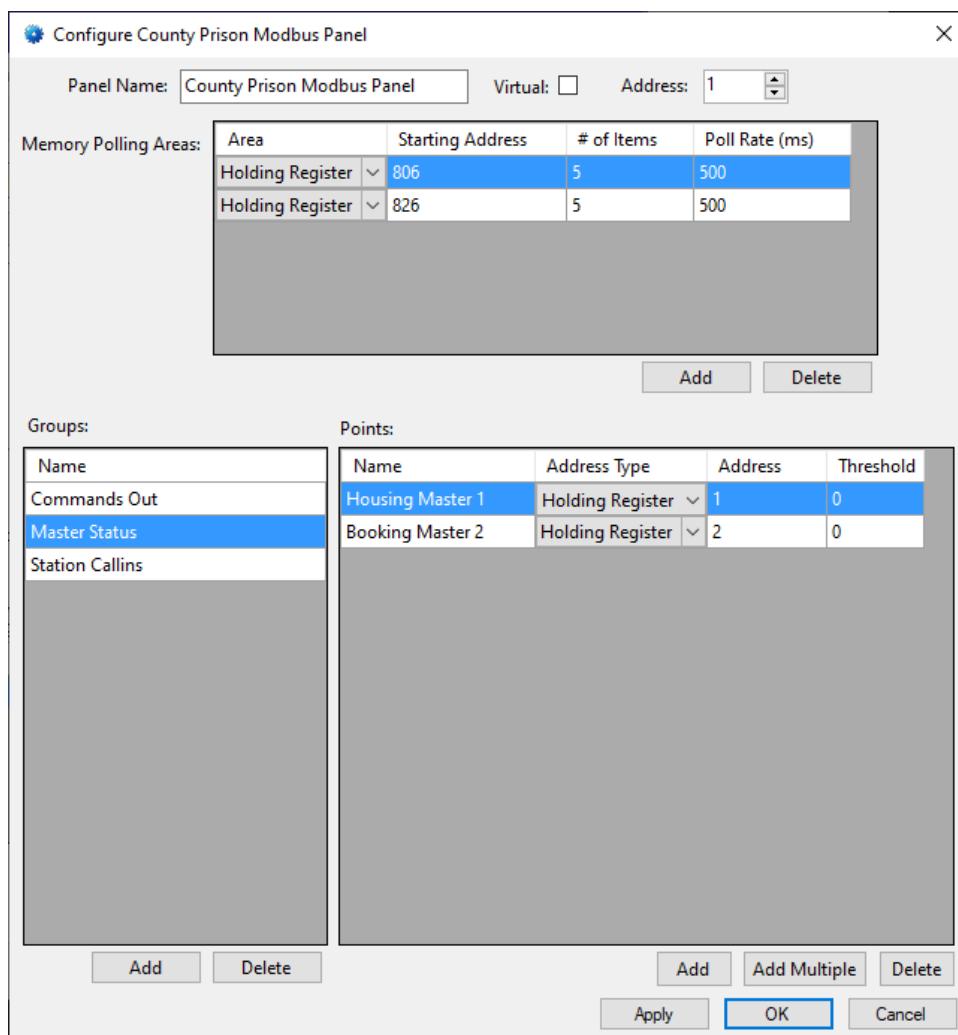


Figure 60 - Example intercom Quick Config dialog

Modbus Panel Guide

Additional nodes were added to the master stations in  **Design View**. These nodes are used to show the connections state of the master station and to which station or paging zone the master is connected.

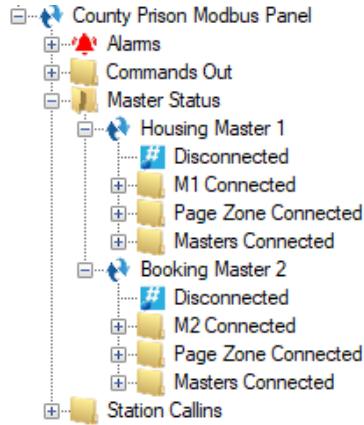


Figure 61 - Master Station sub-nodes

Status updates for each master station come in as a group of registers that is polled. There are five (5) types of transactions:

- Call Request (Transaction Type = 1)
- Call Cancel (Transaction Type = 2)
- Connect Call (Transaction Type = 7)
- Connect Page (Transaction Type = 9)
- End Call (Transaction Type = 10)

Each transaction takes a group of five (5) registers in a Holding Register polling area. There is a different polling area for each master station. The transaction registers for Master Station 1 are Holding Registers 806-810. Master Station 2 uses Holding Registers 826-836.

Register	Transaction Type
806	Master
807	Substation or Page Zone
808	Unused
809	Unused
810	Unused

Figure 62 - Transaction Register Format

Now we can begin defining the Transaction Types and the actions the Intelli-Site will take.

Modbus Panel Guide

Select the **Memory Polling Area** for the first master station. Add a row to the **Transaction Types** table for each of the possible transaction types. Set the **Name**, the **Register** and the **Value** appropriately.

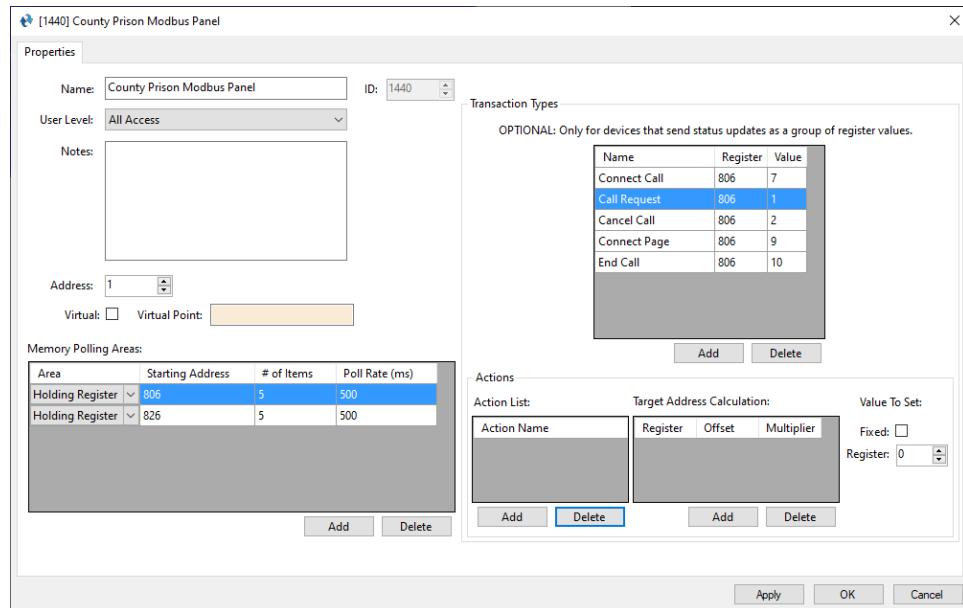


Figure 63 - Modbus panel properties dialog: Transaction Types

When a person pushes the call button on Station 101, the Holding Registers for the status update are filled in with a "Call Request".

Register 806	1
807	Unused
808	101
809	Unused
810	Unused

Intelli-Site must set the Station CallIn node to 1, a fixed number.

Modbus Panel Guide

The screenshot shows the Modbus Panel Guide interface. At the top, it says "Transaction Types" with a note: "OPTIONAL: Only for devices that send status updates as a group of register values." Below this is a table:

Name	Register	Value
Connect Call	806	7
Call Request	806	1
Cancel Call	806	2
Connect Page	806	9
End Call	806	10

Below the table are "Add" and "Delete" buttons. Under "Actions", there's an "Action List" table and a "Target Address Calculation" table:

Action Name	Register	Offset	Multiplier
Turn On Station CallIn	808	0	1

Under "Value To Set":

Fixed:
Value:

Buttons for "Add" and "Delete" actions are also present.

This action takes the value at register 808 which is 101 for our example, multiples this value by 1 and adds the offset. 101 is the register address for **101 CallIn** under the **Station CallIns** node in the Tree. Since **Fixed** is checked, the number in **Value** is assigned to **101 CallIn**.

Next, let's program the "Connect Call" status update transaction type actions. The Holding Registers for a "Connect Call" status update for master station 1 and substation 101 are below.

Register 806	7
807	1
808	101
809	Unused
810	Unused

There are two actions Intelli-Site must process to update the appropriate nodes. The first is "Connect Master to Station" and the second is "Cancel Call" since the call has been answered.

Modbus Panel Guide

The “Connect Master to Station” action is set the master node to the value in register 808.

Actions			
Action List:	Target Address Calculation:		Value To Set:
Action Name	Register	Offset	Multiplier
Connect Master to Station	807	0	1
Cancel Call			
<input type="button" value="Add"/>	<input type="button" value="Add"/>	<input type="button" value="Delete"/>	<input type="button" value="Delete"/>

The “Cancel Call” action is to clear the CallIn point.

Actions			
Action List:	Target Address Calculation:		Value To Set:
Action Name	Register	Offset	Multiplier
Connect Master to Station	808	0	1
Cancel Call			
<input type="button" value="Add"/>	<input type="button" value="Add"/>	<input type="button" value="Delete"/>	<input type="button" value="Delete"/>

Modbus Panel Guide

Revision History

2019-08-23 Creation Date

2019-11-08 pg 9 - References to Omron removed
pg 15 – Address field must be set
pg 16 - Maximum size of a poll is 125 Holding Registers