



MODBUS TCP SLAVE SERVICE

Document created with reference to Software version 19



INNER RANGE recommends that all Inner Range systems be installed & maintained by FACTORY CERTIFIED TECHNICIANS.

For a list of Accredited Dealers in your area refer to the Inner Range Website.

<http://www.innerrange.com>

Overview:

The Integriti Modbus TCP Slave service allows you to read and modify certain states of Doors, Inputs, Auxiliaries, and Areas in the Integriti system by sending messages from an external “Master” device that supports the Modbus protocol.

The service supports mapping Integriti state and control points to Modbus Discrete Inputs and Coils (1-bit data types) respectively; Modbus Input Registers and Holding Registers (16-bit data types) are not currently supported or implemented.

Licensing Requirements:

This service requires the “Modbus TCP Slave Service” license for Integriti. Licenses can be viewed and updated from the License Manager dialog in the Administration tab of Integriti.

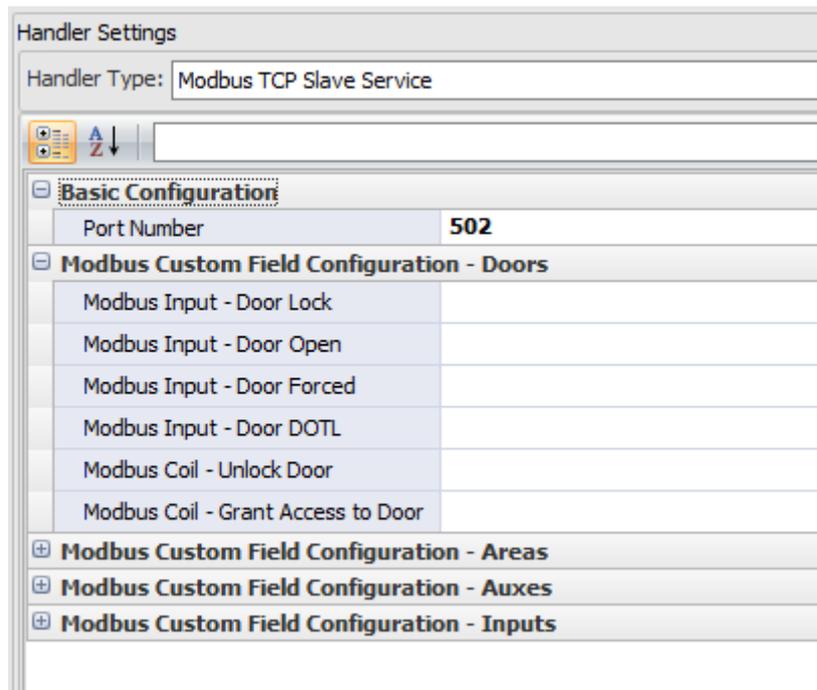


Setup:

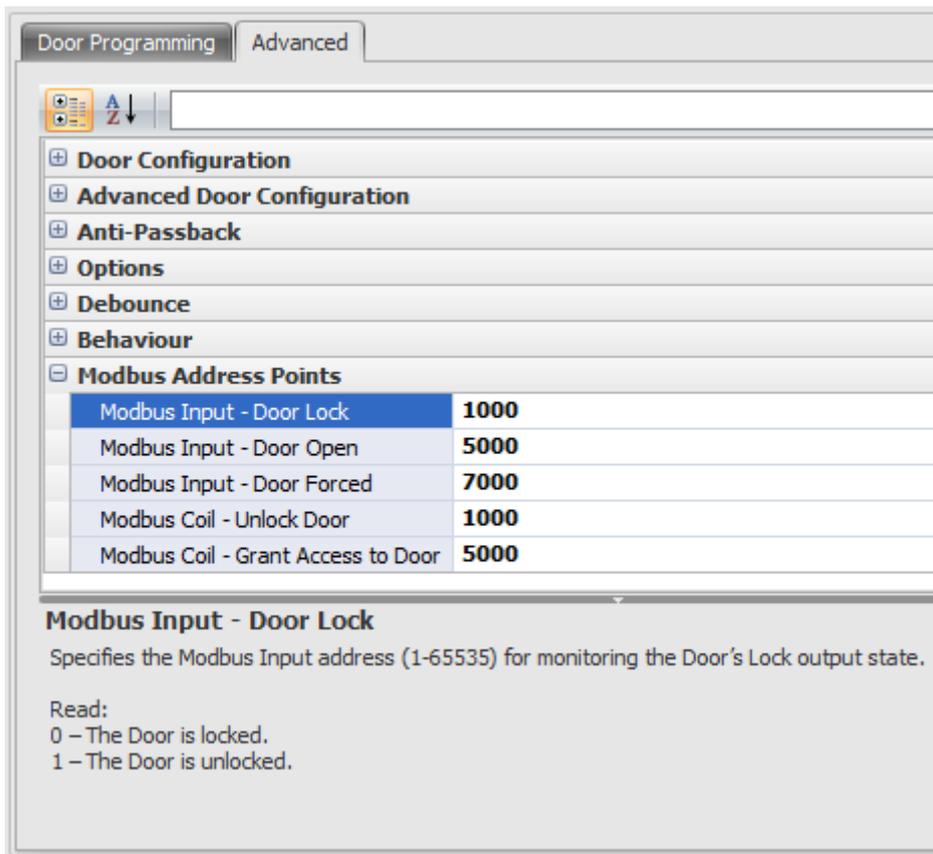
Setting up the Comms Handler:



1. Open the **Communication Handlers** panel from the **Administration** tab of Integriti.
2. Click the **+ Add New** button to create a new Modbus TCP Slave Service communications handler.



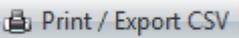
3. Give the handler a name and change the Port Number if required.
4. Click the “Create” button on each category of Modbus point that you want to support, for example, Create the **Modbus Input – Door Lock** and **Modbus Input – Door Open** fields if you want to be able to map the Locked and Open states of Doors to Modbus Input addresses. This creates custom fields for the entity type, allowing you to specify Modbus address numbers in the entity’s settings in order to set up the mapping. The custom fields should be present when you edit a Door entity in Integriti, similar to the sample image below.



Once the communication handler is configured and the custom fields have been created, mapping data can be imported for the entities you want to monitor and/or control.

Importing data (via CSV Import):

Mapping data can be entered manually by editing entity custom field values in the Modbus Address Points category; however this can be slow and inconvenient when configuring Modbus points for hundreds or thousands of entities. To speed up this process, Modbus mapping data can be mass-imported by using the CSV Data Import feature in Integrati.

Use the  **Print / Export CSV** button on the item page (e.g. Doors) to export the entities you want to configure Modbus addresses for.

Load the CSV file into an external spreadsheet editing program (e.g. Microsoft Excel), and add the relevant columns and address data for the entities. These columns will be mapped to the custom Modbus fields when you import the CSV.

	A	B	C	D	E	F
1	Type	Controller	Site	ID	Name	Modbus Input - Door Lock
2	Door	222	Default Site	D3	Modbus Door 1000	9000
3	Door	222	Default Site	D4	Modbus Door 1001	9001
4	Door	222	Default Site	D5	Modbus Door 1002	9002
5	Door	222	Default Site	D6	Modbus Door 1003	9003
6	Door	222	Default Site	D7	Modbus Door 1004	9004



Open the **Import Data** window from the **Administration** tab, and choose to open the CSV that you prepared earlier. Select the 'Define settings as you go' option from the dialog and click Next. On the CSV Configuration screen, select the options that apply to your CSV file (the defaults should work fine in most cases, with the 'Has Column Headers' checkbox ticked if applicable), and click the Test Import button to ensure the data preview looks correct.

Type	Controller	Site	ID	Name	Modbus Inp...
> Door	222	Default Site	D3	Modbus Do...	9000
Door	222	Default Site	D4	Modbus Do...	9001
Door	222	Default Site	D5	Modbus Do...	9002
Door	222	Default Site	D6	Modbus Do...	9003
Door	222	Default Site	D7	Modbus Do...	9004

Click the Next button to continue to the Configure Field Mappings page. Set up the column mappings to identify the entities by Controller and ID, and to import the new data into Modbus custom fields.

Key	Property Name	Import File Column Name	Transformation
>	Controller	Controller	CSVTransformNameLookup
	ID	ID	
	<input type="checkbox"/> Modbus Input - Door Lock	Modbus Input - Door Lock	

Click the Next button and set the Update Mode to 'Update or Ignore' to only import new data into existing entities.

Click Next again to begin the import process. If the import was successful, you should see that the Modbus custom field values have been inserted into the entities' properties. Repeat the import process for any other entity types you want to import Modbus address data for (e.g. Inputs, Areas, Auxiliaries).

To finish setting up, the service needs to be restarted to cause it to load the new data.



From the **Communication Handlers** panel, disable and restart the Modbus service to cause it to load the new mapping data. The service must be restarted whenever new Modbus mapping addresses are configured or imported in order for the service to reload the mapping data.

The Modbus service should now be ready to connect to with an external Modbus master.

Troubleshooting:

Modbus master fails to connect to the slave service

- Make sure that the Modbus slave service is active (its status should appear as “Running” in Integriti’s Communications Handler window).
- Make sure both devices (the slave/Integriti server and the master) are connected to the local network and are able to communicate with each other, and that any TCP ports being used are allowed through firewalls if necessary.

Modbus inputs or coils do not appear to change value when expected

- There may have been changes to entities’ Modbus address mappings after the service was started, stop the service and start it again after any mapping changes to ensure the service is using the latest data.
- Check that the custom field are present and set up correctly in the communication handler settings.
- There may also be duplicate point mappings on certain Modbus addresses (e.g. if you accidentally assign 2 or more Door Locks to the same coil address), check the Integriti Log Viewer for warnings about duplicate point mappings and resolve any if found.

Appendix – Modbus Mapping Points

Supported Modbus mapping points:

The available mapping point types fall under the categories of:

- Modbus Discrete Inputs (1-bit, read-only values) which are only used for reading state information. Valid Input address numbers are 1 – 65535.
- Modbus Coils (1-bit, read/writable values) which are generally used to execute actions (through writing the bit) as well as to read state information. Valid Coil address numbers are 1 – 65535.

All points in the Integriti Modbus Slave Service are mapped to Slave address 1, so make sure that all messages sent from the Master device have the value “1” in the message header’s Slave ID field.

NOTE: When assigning Modbus addresses, be sure to avoid conflicts/duplicate addresses in either of the two address spaces. Any field values marked as Inputs must have a value that is unique from all other assigned Input fields, the same applies for Coils. For instance: you cannot assign Door 1’s Door Lock state to Modbus Input 44, and also assign Door 2’s Door Forced state to Modbus Input 44. Integriti Modbus service does not accept 0 as a valid Modbus address for Inputs or Coils.

Doors:

- Modbus Inputs:
 - Door Lock
 - Door Open
 - Door Forced
 - Door DOTL

INTEGRITI MODBUS TCP SLAVE SERVICE

- Modbus Coils:
 - Unlock Door
 - Grant Access to Door

Areas:

- Modbus Inputs:
 - Area Armed (Input)
- Modbus Coils:
 - Arm Area (Coil)

Inputs:

- Modbus Inputs:
 - Input Alarm (Input)
 - Input Isolated (Input)
- Modbus Coils:
 - Isolate Input (Coil)

Auxiliaries:

- Modbus Inputs:
 - Aux On
- Modbus Coils:
 - Turn on Aux