
# Using NCD Enterprise Sensors with Ignition SCADA via MQTT

This tutorial's purpose is to jumpstart the implementation of an MQTT based sensor network and expose this data to Ignition SCADA software with the least amount of difficulty.

## Introduction:

First off, I would like to introduce the reader to the technology and terms involved with this project. MQTT is a publish/subscribe based messaging protocol and stands for message queue telemetry transport. This protocol was developed by Andy Stanford-Clark of IBM and Arlen Nipper of Cirrus Link in 1999 for the oil and gas industry. MQTT protocol requires a server, also known as a broker. This broker houses the topics, typically in JSON format, where clients can subscribe to a topic allowing it to read or write to that topic. A topic could be a lightbulb, valve, or in this project, a sensor. Overall, MQTT is light-weight and requires a small amount of network bandwidth making it a great choice for IIOT projects. With the rise of Industry 4.0, SCADA software that can funnel manufacturing and plant data from many different sources is an asset that shouldn't be overlooked. Ignition SCADA by Inductive Automation allows users to channel information from sources like OPC-UA servers, MQTT servers, PLCs, databases, RTUs, serial devices, and more. One of my favorite features of Ignition is their pricing model. There is absolutely no limit to the number of clients or tags that can be used, the only limit is the hardware that Ignition is installed on. Also, Inductive Automation has broken their software functions into modules that allows the user to purchase only the modules they need, and enabling customers to develop their own custom modules to integrate with Ignition. Ignition also has 3rd party modules from strategic partners like Sepasoft and Cirrus Link. Cirrus Link's MQTT modules will be used in this project, specifically the Distributor and Engine modules. A more economical MQTT server could be used in conjunction with a custom module, but the

Cirrus Link modules are designed to integrate into Ignition seemlessly making it a top choice for this project designed for easy setup. The Cirrus Link Distributor module is the MQTT broker, and the Engine module subscribes to the broker and exposes the topic data as tags in Ignition. Now that a basic foundation is set, lets start the setup!

## Hardware and Software Needed:

Below is a list of items needed to completely setup the sensor network.

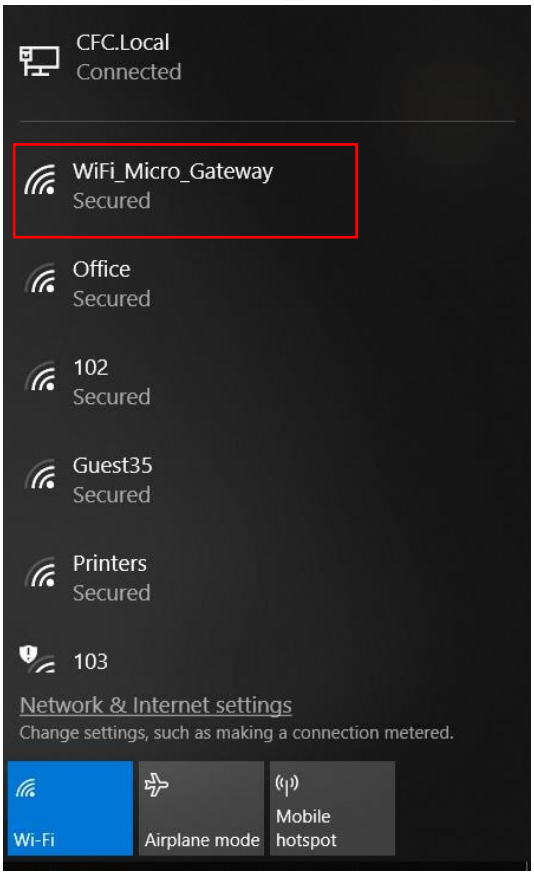
- Ignition 7.9 or higher
- Cirrus Link Distributor Module
- Cirrus Link Engine Module
- NCD MQTT Gateway
- NCD Enterprise Wireless Sensor

Ignition and the Cirrus Link modules can be downloaded for free as a trial version, you only need to fill out a contact form. The download web page is at https://inductiveautomation.com/downloads/ignition and contains both Ignition and the Cirrus Link modules. The trial version will run like a paid version but with a two hour time restriction. Once the two hours is up, just reset the trial for another two hours. Installation instructions are on the Ignition website, while Cirrus Link install documents can be found at https://docs.chariot.io/. The install process is relatively painless and should only take about 15-45 minutes to install everything depending on if you opt to install other items like a database connection. Also, note that there is a third Cirrus Link Module, Transmission, that can write Ignition tag changes to MQTT topics but is not required for this project.

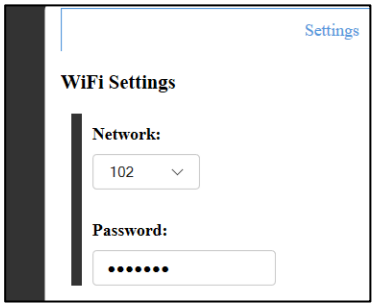
## After Ignition/Cirrus Link Install:

The next step is to setup the MQTT micro gateway purchased from NCD. A setup guide for this can be found at https://ncd.io/wifi-micro-gateway-setup-mqtt/. This is a handy setup guide, but I will also walk you through this to avoid any confusion in the process. Plug up the MQTT gateway and remove the clear cover. Press the red configure button until the LED starts to blink blue, this indicates that the gateway is in setup mode. You will now see the gateway appear in your wireless connections

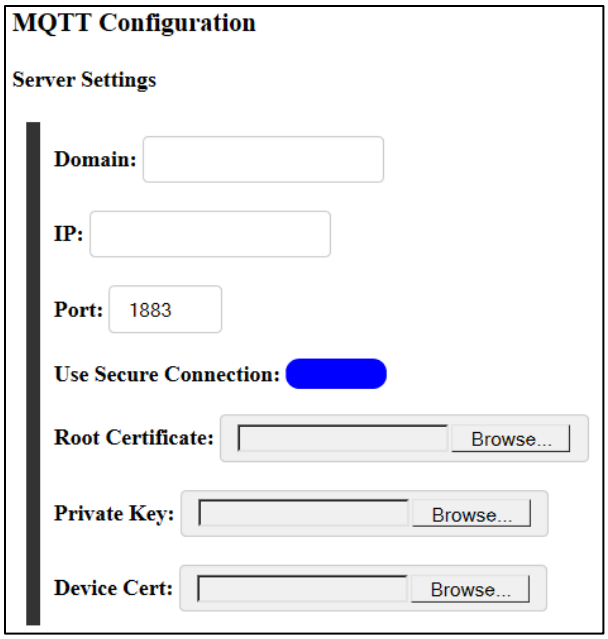
list as Wifi_Micro_Gateway. The default password for connecting is




NCDBeast. Once you connect, the gateway will open a web browser where you should navigate to **172.217.28.1** IP address. Once on the gateway configure page, the first thing to do is give the gateway access to your network via a wifi connection. Choose you wifi network from the drop down menu enter the password below. Setting up this connection will allow the gateway to send data to the MQTT broker. In my wifi

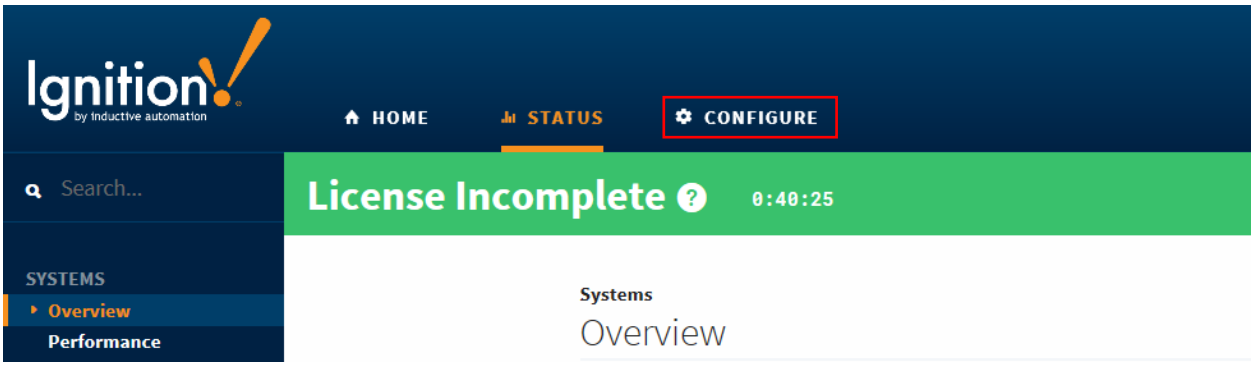


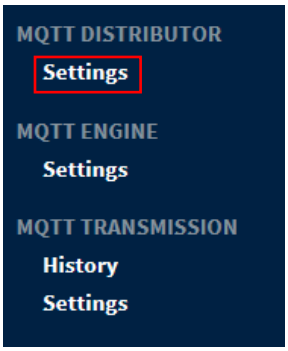

settings, you can see that I connected to the 102 network, which you can also see as an available connection on my computer's network connection list. The next step is to configure the MQTT server settings. These settings tell the gateway where the data it recieves from the sensors should be sent to. If



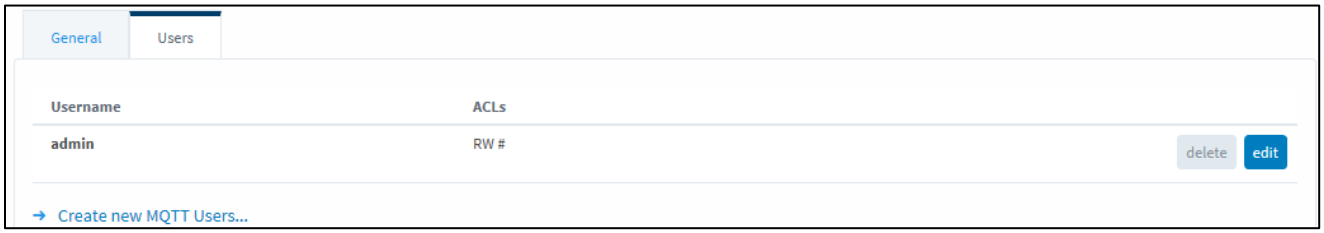

the location of the Ignition server has a domain name, you can use that, otherwise just input the IP address where you installed Ignition. The standard TCP connection port that the Cirrus Link broker listens to is 1883. However, using the port 8883 will allow you to setup TLS security to the communications. Doing this would also require filling in the secure settings fields shown to the left. Now that the MQTT connection is

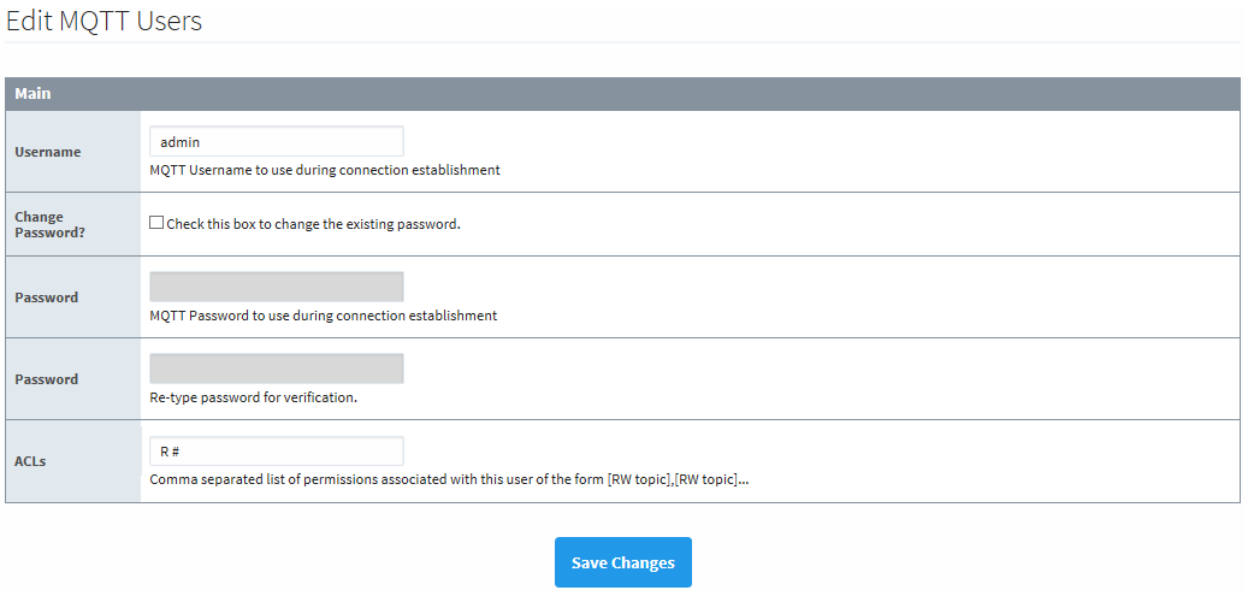
configured, we need to setup a user name and password for the NCD gateway to use to connect to the Cirrus Link broker. This is created in the Cirrus Link Distributor module. From the Ignition gateway home page, located at youripaddresshere:8088, navigate to the configure page.



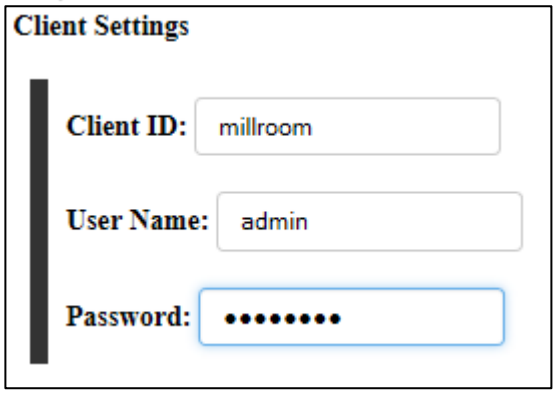




Once at the configure page, scroll down until you see the MQTT modules on the left hand side of the webpage. Click on settings under the Distributor module. The general settings displays all the ports and their settings. Click on the users tab at the at top. Here you will see all the users that currently exist, you will be able to edit or create a new user.



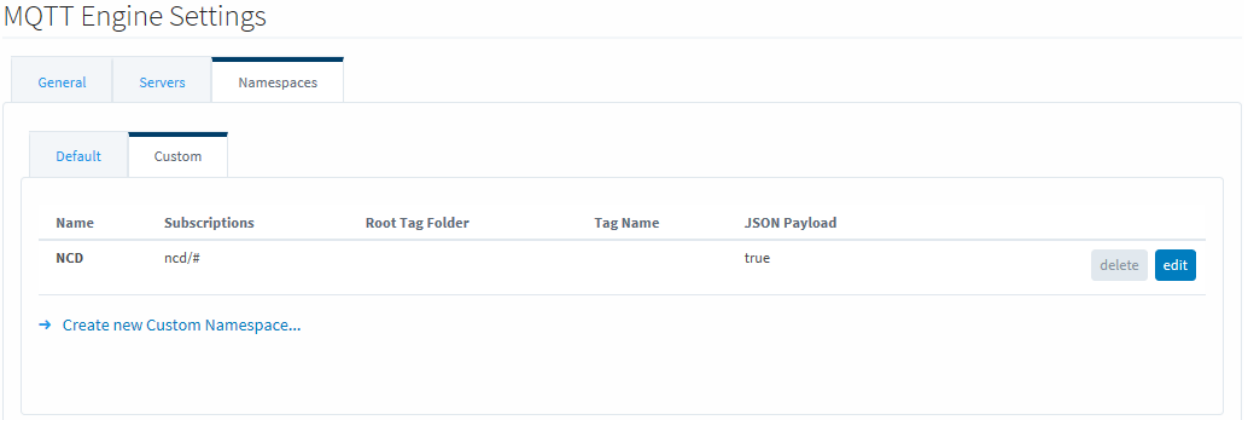



Once you select edit, or create new, you will be able to assign the user name, password, and access. The ACLs field establishes the read/write access to the user. "R" gives read access and "W" gives write access. This can be set on a topic by topic basis, meaning some users could have access to write to certain topics, but only read others. This field holds a comma seperated list in the format of R[topic], W[topic], or RW[topic]. For this project the sensors are the only topics we have, and do not have write capabilities, so the ACL in this case would be R #. The "#" is a wildcard character and applies the settings to all available topics. Once these changes are saved, you can now put the credentials into the NCD gateway client settings. The client ID can be left blank or used to give a string-type name to the NCD gateway.

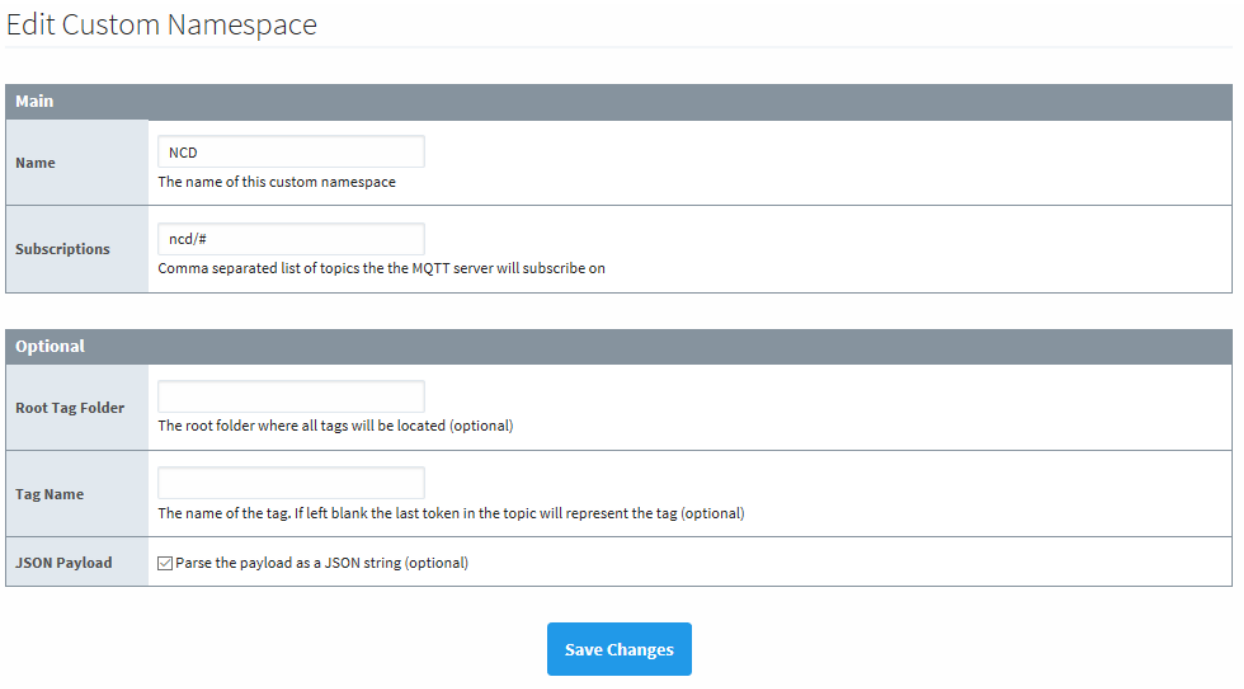



After you have your user setup, navigate back to the congifure page and click on settings under the Engine module. Since the NCD gateway does not use a default namespace like the Sparkplug specification, we will create

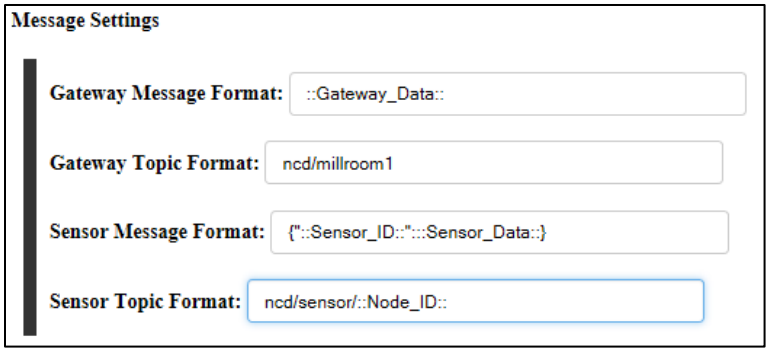
a custom namespace that will handle a JSON payload. You can see I



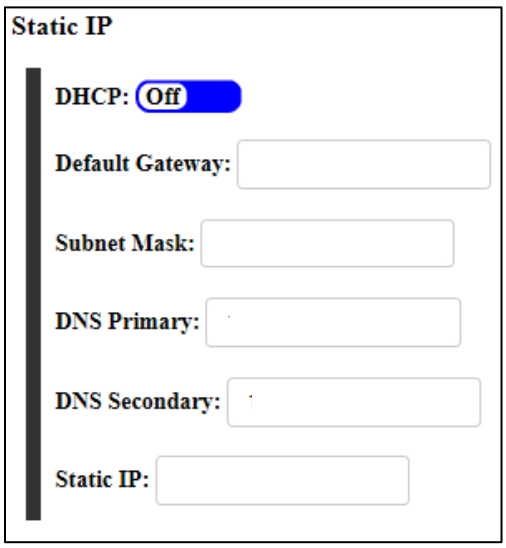

have already created the custom namespace above. You will click on "create new Custom Namespace." I named the new namespace "NCD" and the subscription is ncd/#, which can be changed to be whatever you want to assign to it. Again we are using the "#" wildcard character, this will allow the broker to subscribe to all topics that start with "ncd/". Make sure that you select the JSON payload check box at the bottom or you will not see your data. An example of this configuration is shown below.




Save the settings and navigate back to the NCD gateway configuration page. Here we will input our namepsace/messaging conventions.




The first two input boxes configures the formatting of the NCD gateway itself. Tokens are used here to setup the format that you wish to use, you can think of the tokens as variables that are just placeholders for the sensor data. For the Gateway Message Format, use "::Gateway_Data::". For the Gateway Topic Format I used "ncd/millroom1". This can be customized, but notice that the custom namespace we created is subscribing to topics that start with "ncd/" so make sure that this matches what you put in the Engine module namespace. For Sensor Message Format use {"::Sensor_ID::":::Sensor_Data::}, and for Sensor Topic Format use "ncd/sensor/::Node_ID::" The Sensor Topic Format can be cusomized, but I recommend using this format. Of course the "ncd/" portion can easily be changed as long as it matches what you put in the Gateway Topic Format and the custom namespace. **Note that whenever you save the NCD configuration settings, it will reboot the gateway into run mode afterwards, and you will have to re-enter config mode to change any settings.** The next step is to configure the Static IP under the Advanced Tab in the NCD gateway config page. This part is probably the trickiest portion of the setup. The Default Gateway is the main router IP address of your network. A subnet mask separates the IP address into the network and host addresses (<network><host>). Subnetting further divides the host part of an IP address into a subnet and host address. The DNS portion is basically the phonebook of the internet and translates domain names to IP addresses. The Static IP is what you want the NCD gateway's IP address to be. Just



make sure you assign one to it that it is not already in use. If you are unfamiliar with these settings it would be best to get the input from your IT department, or someone you know that can help you set this up. Now is a good time to fire up one of your NCD wireless enterprise sensors and make sure it connects to the NCD wireless gateway. The sensors are shipped with external power enabled so you can use a barrel plug ac adaptor or wire it with the screw terminals. I like using the batteries that are preinstalled which makes installing the sensors much easier. To switch to

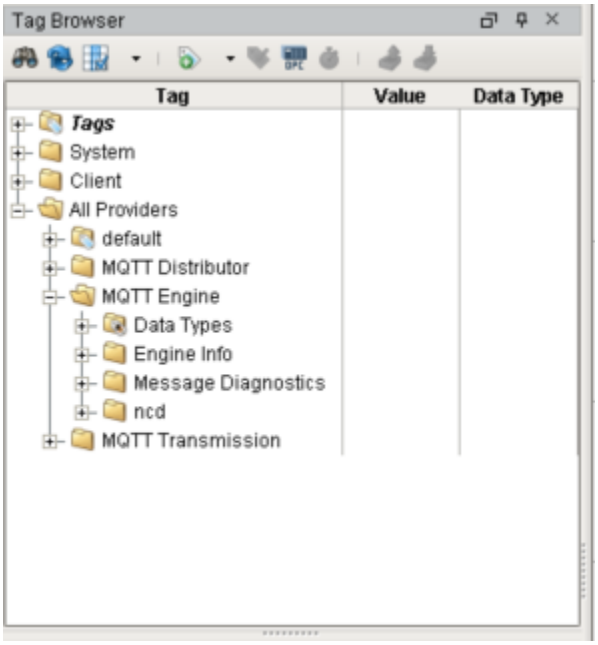

battery mode, open the sensor case and shift the power pin jumper over one. The red rectangle shows the power jumper mentioned and is shown in the battery powered position. Once this is in place, the sensor will automatically connect to the wireless gateway. Depending on the factory settings, it may take 10 minutes or so for the sensor to send out the first data packet. Once it does you will be able to see it in the NCD gateway configuration webpage under the Devices tab.

| Sensor MAC | Sensor Type | Battery Level | Signal | Transmissions | Telemetry | Settings |
|---|---|---|---|---|---|---|
| 00:13:A2:00:41:84:57:9B | AMS Pressure & Temperature | 99% | 100% | 170 | View | Configure |
| 00:13:A2:00:41:84:57:6A | AMS Pressure & Temperature | 99% | 100% | 133 | View | Configure |

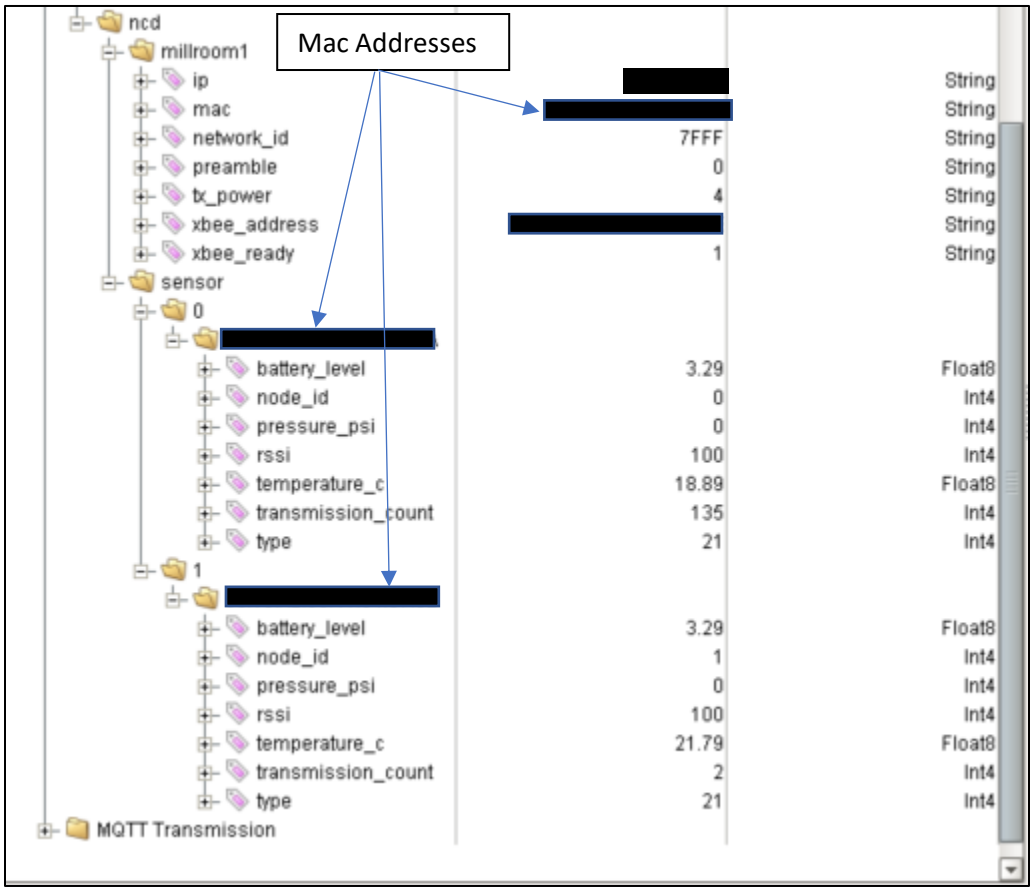
Currently, NCD sensors must be configured with a Zigmo Router from NCD and is purchased seperately. The configuration sets how often the sensor sends out data, the node id, etc. NCD is currently working on getting the sensors configurable through the gateway configuration page, which will be a huge plus once that is complete. After you have verified that your sensors are sending data to the NCD gateway, save the settings.

## Seeing the Data:

After your NCD gateway settings are saved, it will reboot in run mode. Once it succesfully connects to the Cirrus Link broker, the LED will turn green. If it doesn't, cycle the power and boot again. If the LED turns red after booting there is a connection error. Double check all of the ip addresses, user name, and password ensuring all are correct. If you still cannot connect, verify that you can ping the ip address of your Ignition server and port 1883 is not blocked by any firewalls. If all goes accordingly, congradulations, your MQTT sensor network is now functional! Now, lets see where the sensor data is inside Ignition. The Cirrus Link Engine module automatically creates the tags and folder structure inside Ignition once topics are being posted to from the NCD side. Open up an Ignition designer window, and in the tag browser pane you will see all the parent folders. Click on "All Providers" then "MQTT ENGINE" and you will see the ncd folder which has our sensor data inside. The name "ncd" came from the gateway and sensor message topic format we configured earlier. Once you drill down into the ncd folder you will see the NCD MQTT gateway folder, and the sensor folder. This folder structure, again, came from the



message formatting we did with the tokens. “Millroom1” is the NCD gateway,




The “0” and “1” folders are the node ID of both sensors I connected and contains the mac address of each sensor inside. Inside these folders is the data from the sensors: Battery level, node_id, transmission count, sensor type, and other data specific to that type of sensor. The ones I’m using are pressure sensors that will be monitoring the vacuum pumps of CNC machines. From here the data can have alarms on them, stored historically in a database, and displayed on machine HMIs or manager dashboards.

Written By: Daniel Hayes