

## TECHNICAL NOTE

**Model**

Graphic HMIs

**Title**

MQTT Set-up

TN5111

P/N: 0907-5111

Rev. 01 Date: 10/04/2016



### Summary

This tech note walks through the steps necessary to configure an HMI as an edge gateway for an MQTT network. There are 4 main tasks that will be addressed in this tech note:

- An EZwarePlus project will be created that enables the HMI to act as an MQTT publisher
- HMI memory addresses will be mapped to MQTT topics
- The project will be run in the EZwarePlus HMI simulator on a PC and publish data to a public test broker
- MQTT.fx will be used to subscribe to topics and receive data from the HMI application

### What You Will Need

---

- EZwarePlus version 5.03.02.026 or later. A free upgrade can be downloaded here:  
<http://www.maplesystems.com/ezware/evaluate-hmi-software/index.htm>

A demo version can be downloaded from here:

<http://www.maplesystems.com/ezware/evaluate-hmi-software/>

- Basic knowledge using EZwarePlus
- MQTT client software. This guide will use MQTT.fx, which is available at [mqttfx.org](http://mqttfx.org).
- A windows PC

An MQTT sample project can be downloaded from:

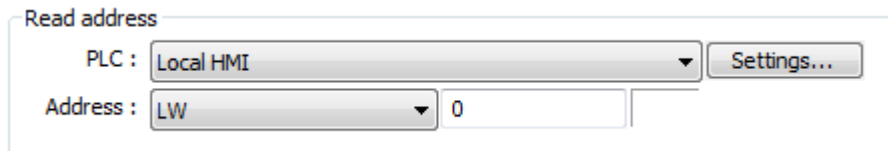
<http://www.maplesystems.com/cgi-bin/download/sample.asp>

## Step 1 - Project Set-Up

---

This step will create an HMI screen with some numeric inputs to use later:

- Start EZwarePlus and create a new project. Any model number can be selected.
- Select **Objects...Numeric** from the menu bar.
- Under **Read address** select Local HMI for the PLC
- Select LW from the address type dropdown list, and enter **0** in the address field (this is the default).



Read address

PLC : Local HMI Settings...

Address : LW 0

- Click **OK** to place this object on the screen
- Make five copies of this numeric object and configure them with the following addresses:

Address
LW-0 (already done)
LW-1
LW-2
LW-10
LW-11
LW-12

In Step 2 the addresses used above are going to be assigned to MQTT topics. You will add some labels to the screen and organize the inputs under the topic they will be assigned to so that it is easy to tell which address is assigned to which topic:

- Create five text objects (Select **Draw...Text** from the menu bar) to create the following labels:

Label
Topic: <name>/machineOne/stats
Topic: <name>/machineOne/temp
Topic: <name>/machineTwo/stats
Topic: <name>/machineTwo/temp

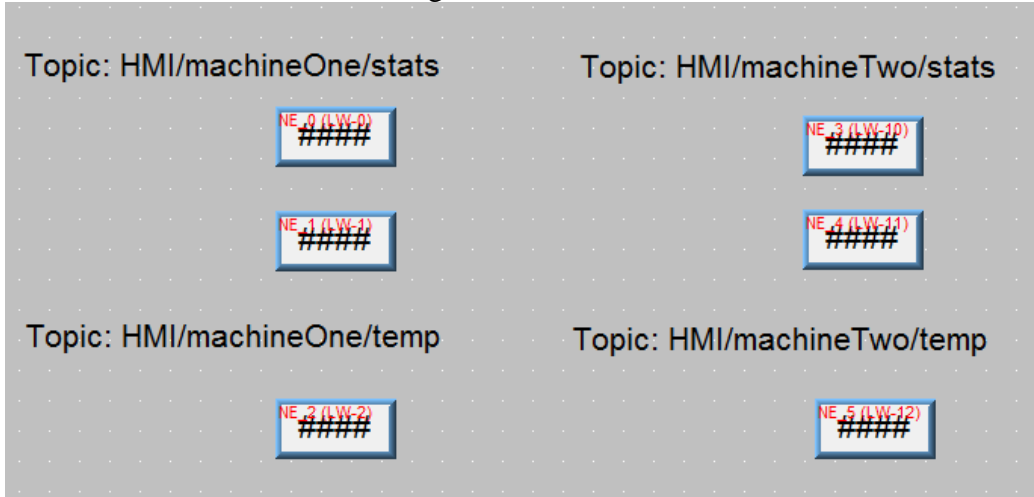
Note: replace <name> with a unique name of your own choosing to use for all topics.

- Arrange the labels on the screen so that the numeric inputs are placed under the label for the topic they will be assigned to as follows:

Label	Numeric Input Address
Topic: <name>/machineOne/stats	LW-0

Label	Numeric Input Address
	LW-1
Topic: <name>/machineOne/temp	LW-2
Topic: <name>/machineTwo/stats	LW-10
	LW-11
Topic: <name>/machineTwo/temp	LW-12

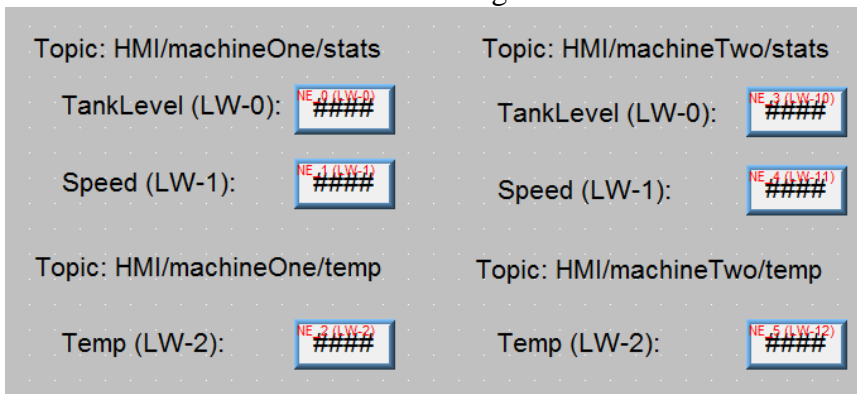
The screen should look something like this:



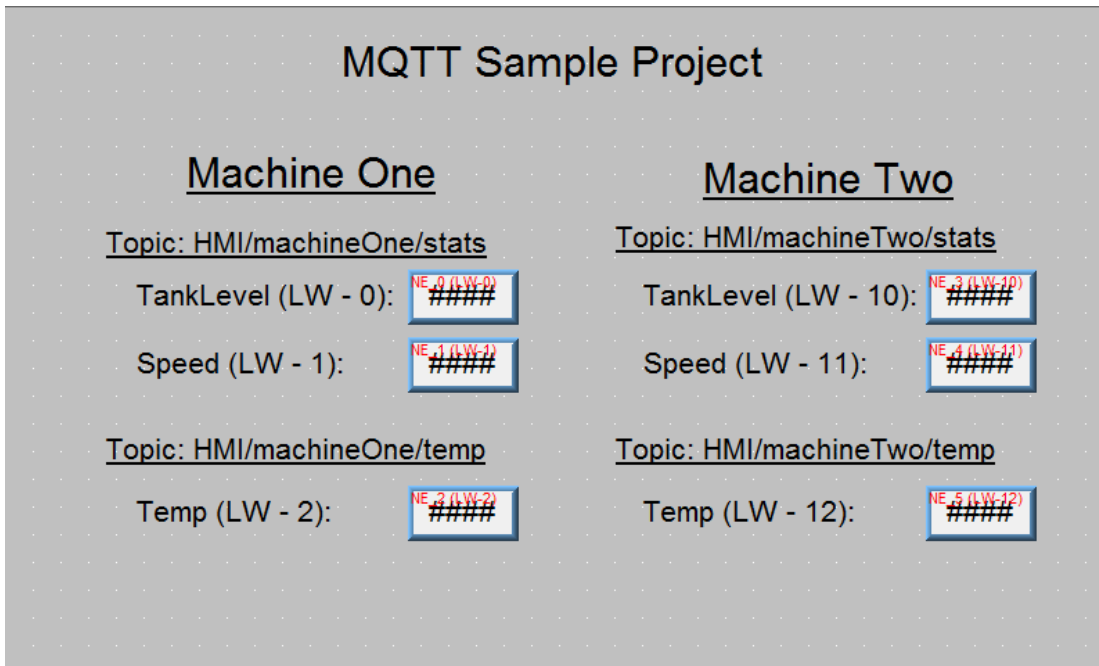
- Now add the following labels to the left of each numeric object so that it is clear what the number represents:

Numeric Input Address	Label
LW-0	TankLevel (LW – 0):
LW-1	Speed (LW – 1):
LW-2	Temp (LW – 2):
LW-10	TankLevel (LW – 10):
LW-11	Speed (LW – 11):
LW-12	Temp (LW – 12):

The screen should now look something like this:



- That completes the screen set up. The screen shot below shows some more optional labels and label formatting that can be added to better organize the screen:



## Step 2 - MQTT Object Set-Up

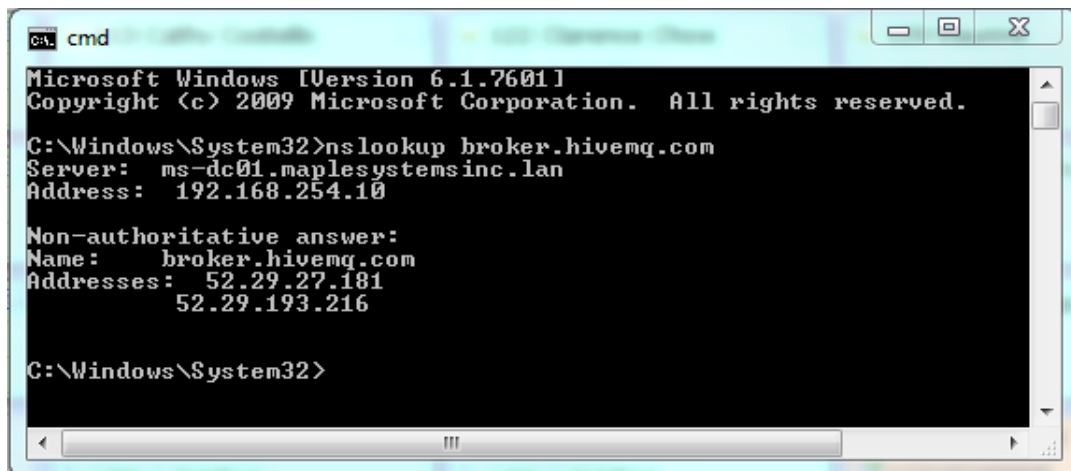
---

Next you will add an MQTT object to the project, configure the broker settings and configure the topics.

The settings below will configure the application to publish to a public broker at **broker.hivemq.com**. There are several public brokers that could be used to complete this project. Another, which is hosted by the Eclipse Mosquitto™ project, is **test.mosquitto.org**. These brokers are public and should only be used for testing purposes.

**Note:** The steps below use a specific IP address to connect to the broker. The IP address shown below will likely be out of date as you are reading this. It is necessary to look up the current IP address for the broker you are using. There are several ways to do this:

1. Use an online DNS lookup tool. Any search engine will provide several options.
2. Use the **nslookup** command built into windows:
  - In the windows start menu type **cmd** into the search field and hit enter to launch the command prompt.
  - In the command prompt, type: **nslookup broker.hivemq.com**



```
cmd
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

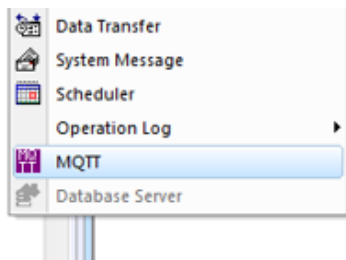
C:\Windows\System32>nslookup broker.hivemq.com
Server: ms-dc01.maplesystemsinc.lan
Address: 192.168.254.10

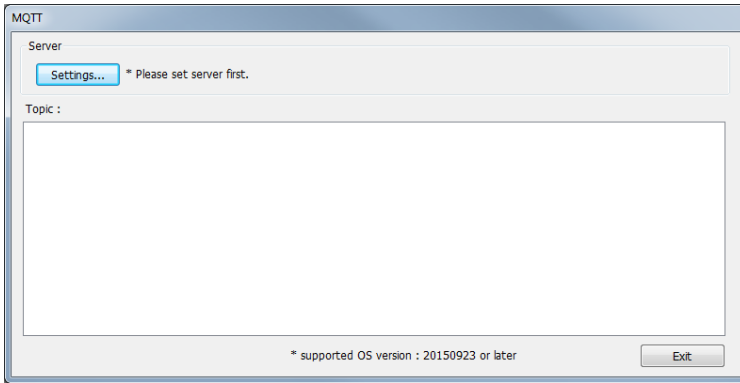
Non-authoritative answer:
Name:      broker.hivemq.com
Addresses: 52.29.27.181
           52.29.193.216

C:\Windows\System32>
```

Once you have the IP address, write it down and wherever you see an IP address in the steps below enter the current one for the broker you are using. For example, in the screen shot above the nslookup returned **52.29.27.181**, and **52.29.193.216** for **broker.hivemq.com**, either of these IP addresses could be used in the steps that follow.

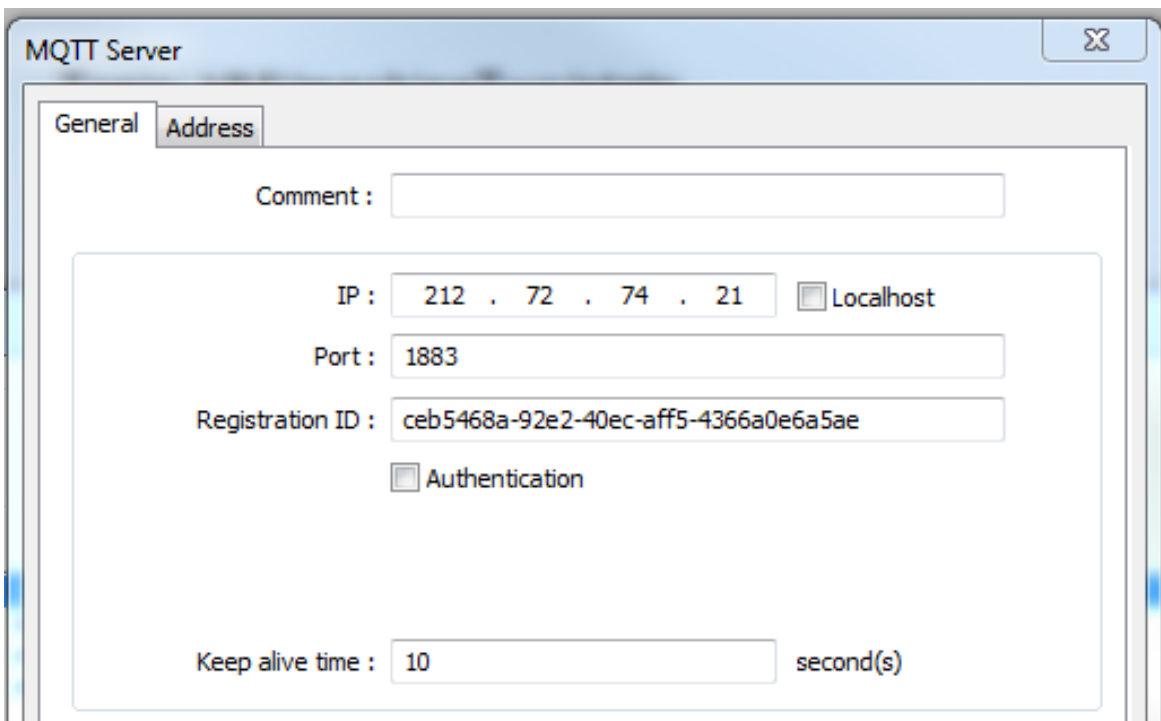
- Select **Objects...MQTT** from the menu bar





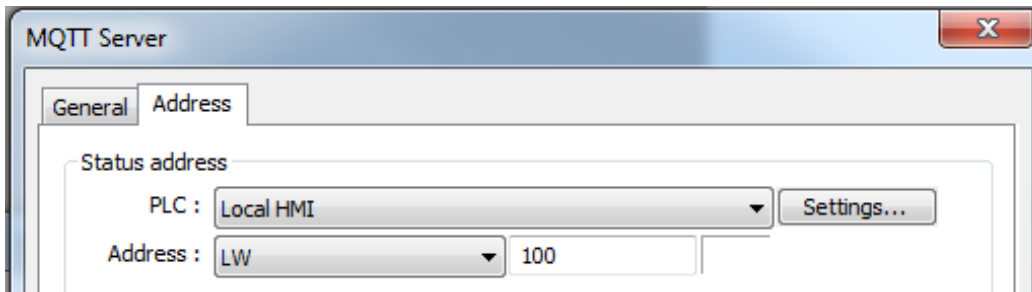
## Broker Configuration

- Click on the **Settings** button. This is where the settings for the MQTT broker are configured:



For more information on these settings refer to the MQTT chapter in the EZwarePlus programming manual.

- Uncheck the **Localhost** checkbox and enter the broker's IP address.
  - The port is set to **1883**
  - The **Authentication** box is not checked.
- Click on the **Address** tab. Change the **Status address** from **LW – 0** to **LW – 100**

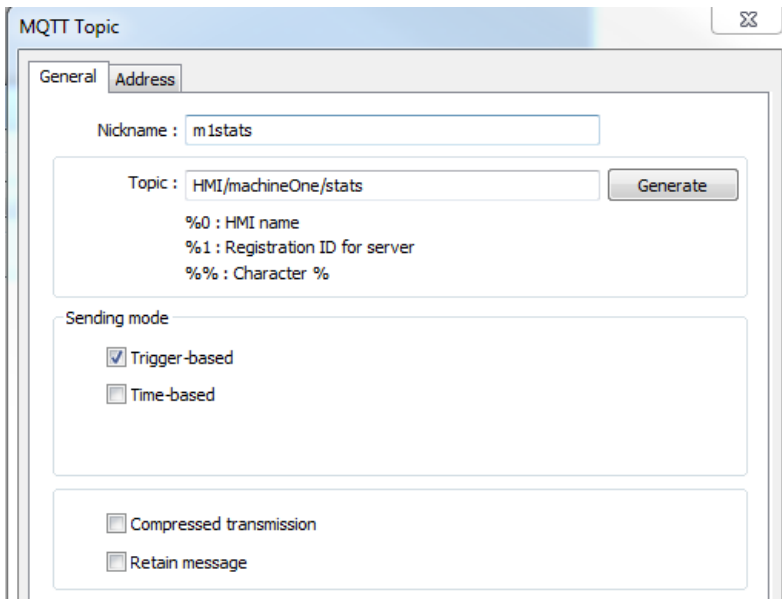


- Click **OK** to accept the settings and return to the MQTT dialog

## MQTT Topic Configuration

The next task is to create topics and assign HMI memory addresses to these topics. The process is the same for PLC addresses or tags.

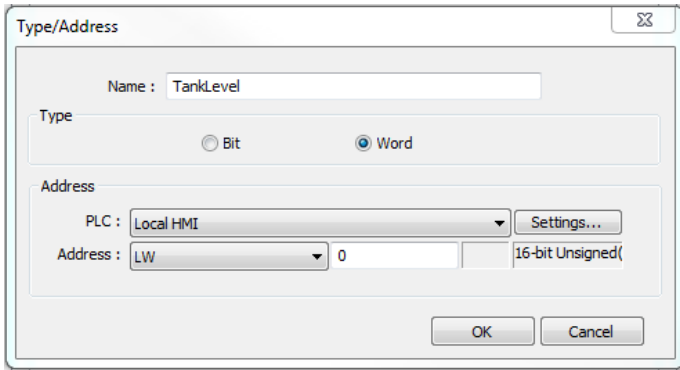
- Click on **New...** from the MQTT dialog to create a new MQTT topic:



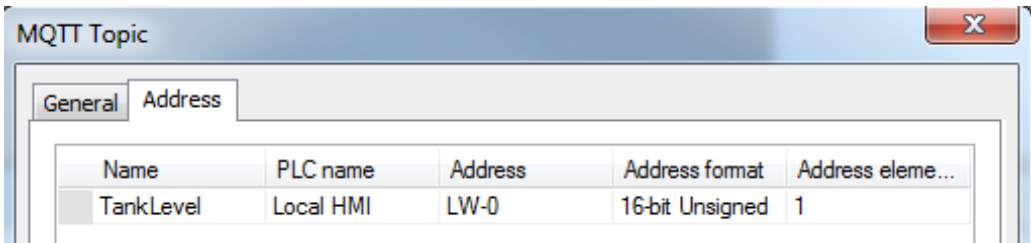
- Enter: **m1stats** for the **Nickname** of this topic
- Enter: **<name>/machineOne/stats** under **Topic**. Replace **<name>** with a unique identifier of your own choosing.
- Leave the **Trigger-based** checkbox checked under sending mode

Now you will assign addresses to the topic:

- Click on the **Address** tab of the **MQTT Topic** window
- Click on **New** to assign a new address
- Enter: **TankLevel** for the **Name**
- Set the Type to **Word**
- Select **LocalHMI**, **LW** and enter **0** for the address fields.



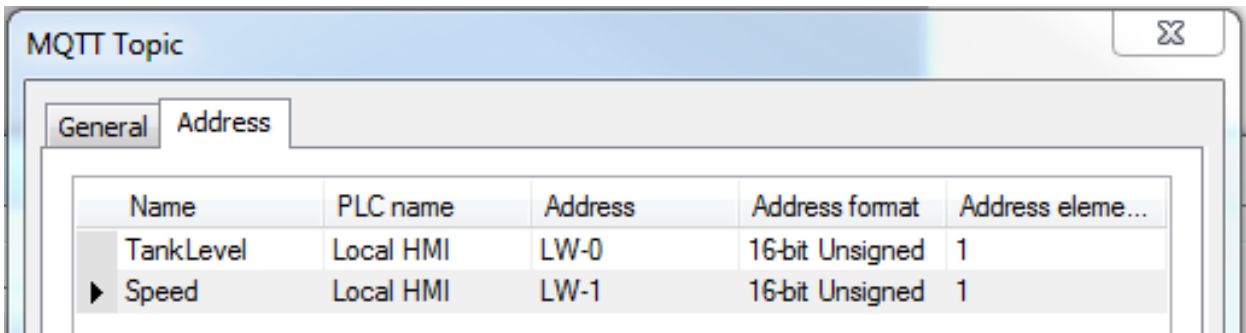
- Click **OK** to add the address to the topic.



- This topic will have two addresses. Click new again and enter the following values to configure the second address:

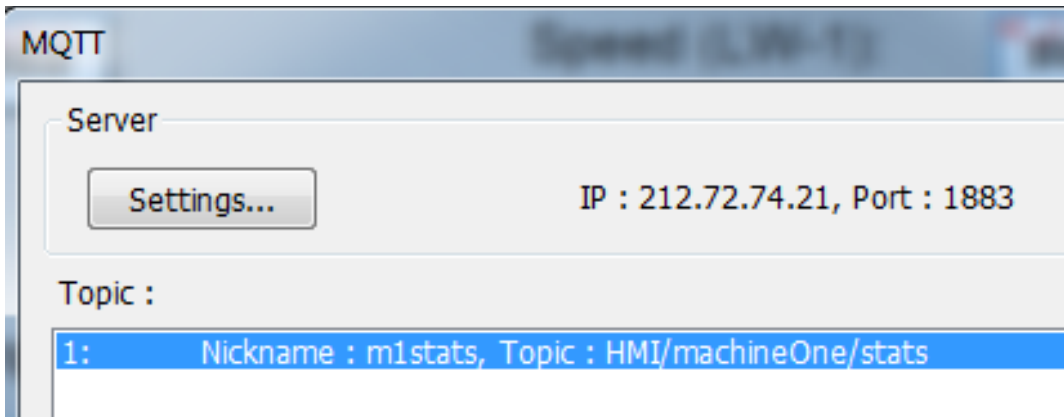
Name: **Speed**  
 Type: **Word**  
 Address: **LocalHMI, LW-1**

- The address tab for this topic should now look like:



- Click **OK** and the topic should appear in the **Topic:** list.

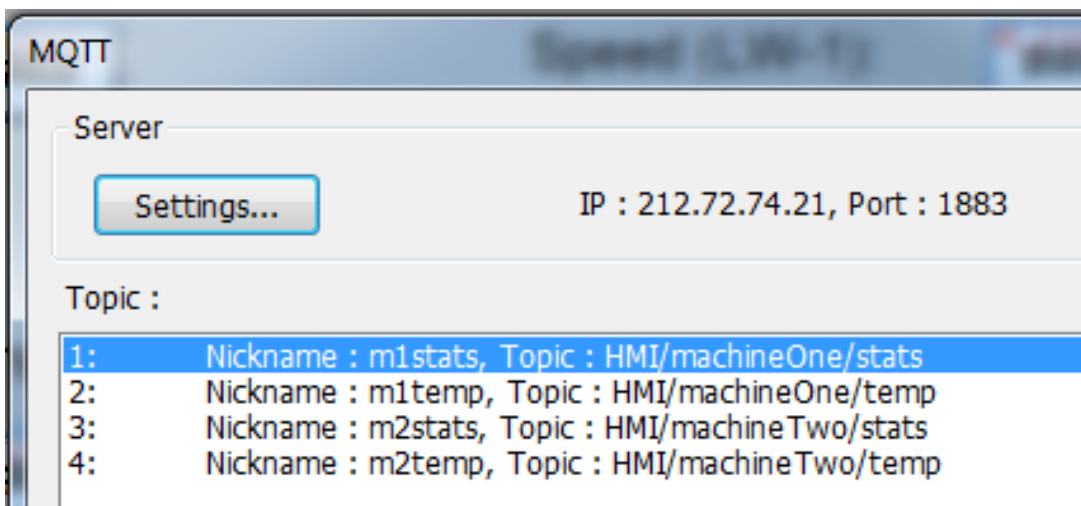




- Repeat this process to create the topics and addresses shown below (**Note:** Replace <name> with a unique name of your own choosing. The first topic shown was created in the preceding steps.):

Nickname	Topic	Address Name	Address
m1stats	<name>/machineOne/stats	TankLevel	LW – 0
		Speed	LW – 1
m1temp	<name>/machineOne/temp	Temp	LW – 2
m2stats	<name>/machineTwo/stats	TankLevel	LW – 10
		Speed	LW – 11
m2temp	<name>/machineTwo/temp	Temp	LW – 12

- The Topic list should now contain four topics:



- When everything looks correct click **Exit** to return to the development screen.
- Select **file > Save** from the menu bar then **Tools > Compile** to build an executable version of the project that can be run in the simulator.

You have completed the HMI project set up. Now you can run the project in the simulator and start receiving data with an MQTT subscriber application.

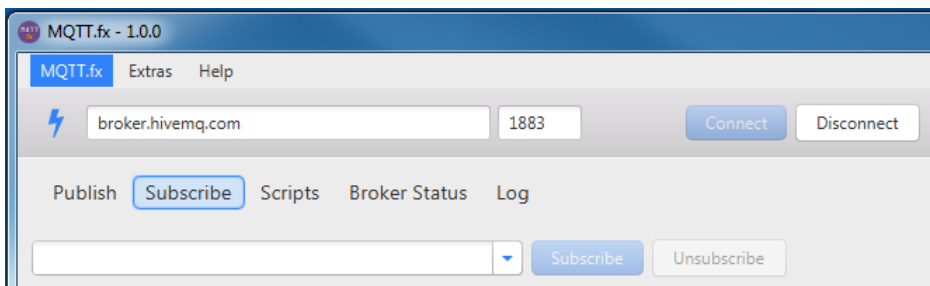
## Step 3 - Testing

This step will use MQTT.fx for the subscriber, if you don't have an MQTT client download and install MQTT.fx before proceeding.

- Start the EZwarePlus simulator by selecting **Tools...On-line Simulation** from the menu bar. This will start the publisher on the PC. Leave this simulator running.
- Start MQTT.fx. Toggle the connection mode icon so that the lightning bolt is shown. This will allow an address to be entered directly in the address field instead of creating a configuration profile:

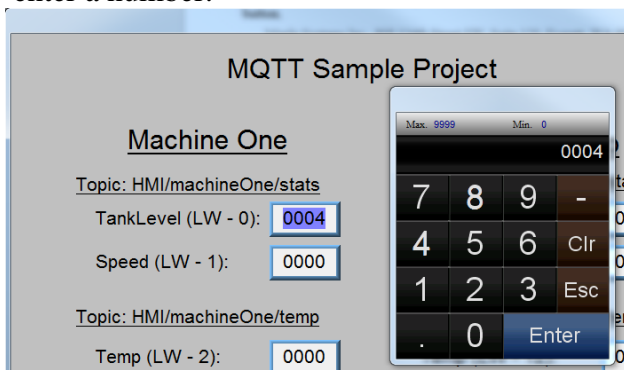


- Enter **broker.hivemq.com** as the **Broker Address**. Use **1883** for the port number and click connect.
- Next click on the **Subscribe** option, enter **<name>/machineOne/stats** for the topic and click the **Subscribe** button.



Since the **Sending mode** in the HMI application is set to **Trigger-based** you will not receive any message until one of the values changes on the simulator.

- Switch to the simulator window, click on the **TankLevel** numeric object and use the keypad to enter a number.



- After clicking enter on the keypad a message should appear in the MQTT.fx window:

```
HMI/machineOne/stats
▼ {...
{
  "d": {
    "Speed": [ 0 ],
    "TankLevel": [ 4 ]
  },
  "ts": "2016-04-14T13:10:33.629078"
}
```

The packet payload delivered to the broker is in Java Script Object Notation (JSON).

The data in the “**d**” object are the data points added to the topic in EZware listed alphabetically, based on the names given in the address tab of the MQTT configuration window.

The data in the “**ts**” object is a combined date and time timestamp based on the HMI local time when the message was sent.

Congratulations, the HMI project is configured as an MQTT edge gateway.

Once your data is in the broker, you can subscribe to it from several software packages, or your own custom application. This data can be delivered to your entire team without overloading the control network with communications requests.

**Trouble Shooting Note:** If you don’t get any messages from the HMI project try:

1. Change one digit in the **Registration ID** in the MQTT object settings in EZwarePlus. Recompile the project and relaunch the simulator.
2. Double check that the MQTT server is active and that the current IP address for the server was entered in the MQTT object. See the note about looking up the current IP address at the beginning of Step 2 in this document.

## Additional Information:

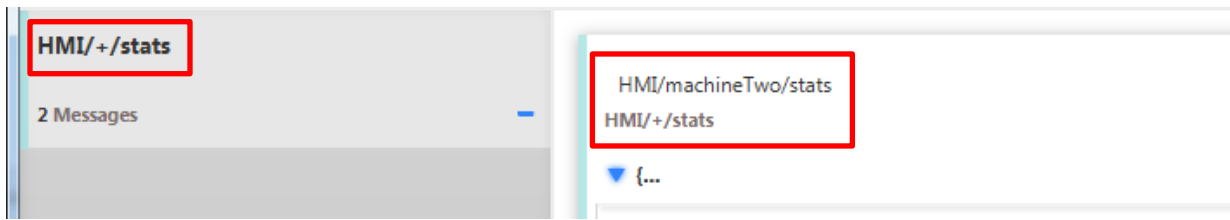
The following sections will give you more information on what can be done with the MQTT object and how to use it in your IIoT solution.

## Using Wildcards

Subscribers can use wildcards to get data from multiple topics at once.

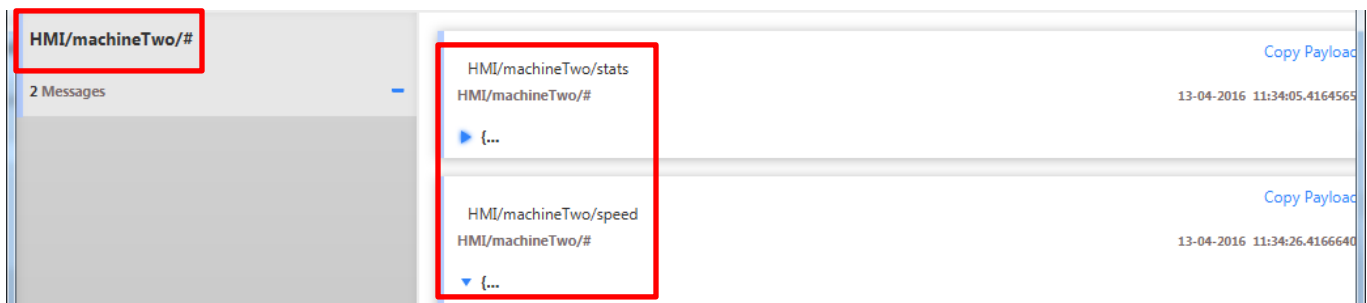
+ is the single level wildcard. As you saw above, we created the <name>/**machineOne/stats** and <name>/**machineTwo/stats** topics. If you subscribe to the topic <name>/+/stats, the topics of <name>/**machineOne/stats** and <name>/**machineTwo/stats** are delivered to the subscriber. Any time

the variables in either of the **stats** topics changes a message will be received in MQTT.fx.



In this example the two topics are on the same physical device but they do not have to be. The client would be subscribed to any device with a **stats** topic that was publishing to this broker.

The pound, hashtag or number sign (#) is the multi-level wildcard. Similar to the single level wildcard, you can subscribe to multiple topics at once. If subscribed to `<name>/machineOne/#`, you will receive ALL topics that start with `<name>/machineOne/`. This will include `<name>/machineOne/stats`, `<name>/machineOne/temp`, etc.



Subscribing to the topic `<name>/#` will subscribe to ALL of the topics on the HMI.

For more information on wildcards and MQTT best practices, go to: <http://www.hivemq.com/blog/mqtt-essentials-part-5-mqtt-topics-best-practices>

## Other Broker Options

### Internal Broker

Selecting the local host option in the server settings dialog of the MQTT object will cause the HMI to use an internal broker. Other HMIs can publish through this HMI's broker using the IP address assigned to the HMI in the system settings dialog.

### Third Party Broker

There are several organizations that provide MQTT broker software. Generally these can be run on any PC and are pretty straightforward to configure. Once the server is up and running the HMI should be configured with the IP address of the computer where the broker is running.

## **Mosquitto**

Mosquitto ([www.mosquitto.org](http://www.mosquitto.org)) is an open source MQTT broker software that can be installed and hosted on several platforms. Mosquitto can be installed on a PC, a laptop, or even on a device like a Raspberry Pi.

## **HiveMQ**

HiveMQ ([www.hivemq.com](http://www.hivemq.com)) is a paid MQTT broker that offers additional features over Mosquitto.

## **Cloud Broker**

Several companies offer MQTT brokers hosted on their cloud services. Instead of creating and maintaining a server and software to distribute your data, the data can be transmitted to an offsite broker.

## **Online Testing Broker**

[test.mosquitto.org](http://test.mosquitto.org), and [broker.hivemq.com](http://broker.hivemq.com) are online test brokers that are open to the public. However, these broker is not private, so your data can be exposed. With that being said, it's a great place to start transmitting data from your HMI and subscribing with a testing client.

## **Custom Client Applications**

The JSON format used to publish messages from the HMI can easily be incorporated into website application, smart phone applications and much more.

A sample Android application can be found here:

<http://www.eclipse.org/paho/clients/android/sample/>

Information on using JSON objects in many programming languages can be found here:

<http://json.org/>