MQTT Monitor Instructions

Description

The MQTT Monitor is designed to provide a wireless handheld tool to display MQTT activity, and to publish test messages.

- Several hours run time, depending on the size of LiIon or LiPo battery installed.
- Charge through a standard micro-usb plug.
- Sends turnout and sensor messages.
- Clears the retained messages on the MQTT broker.
- Allows the MQTT broker IP address to be changed and saved in EEPROM.
- A red/green LED indicates battery charging and charge complete.
- ESP8266-based Wemos D1 Mini providing WiFi connection to the MQTT broker.

Note: In this document, the words server and broker mean the same thing. It refers to the MQTT broker/server, such as "mosquitto" (spelled with two t's) that handles all the MQTT messages for your system.

Assembly

Surface Mount Parts

The surface mount parts need to be soldered onto the board before the through-hole parts since the top surface needs to be laid flat on the "cookie sheet". My cookie sheet is a piece of 1/16" thick aluminum big enough to cover the hot plate, with a wooden handle attached. My hot plate is actually a heater section from a semiconductor wafer oven with a digital controller.

Set the hot plate to 250^{o}C. Having the hot plate too hot will cause some of the parts to dance around when the flux boils. Check the profile included with the solder paste for details.

The surface mount parts chosen for this project are large enough to allow for manual solder paste application and part placement, even if you're not an expert.

Apply solder paste to all the pads, then place each part.

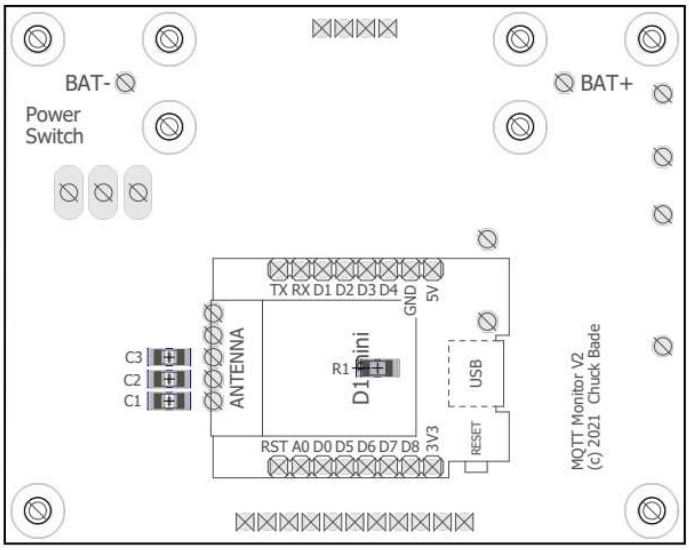


Figure 1: Surface Mount Parts Placement

Table 1: Surface Mount Parts Needed

Qty	Part	Value	DigiKey #	Device
3	C1,2,3	.1uF	399-C1206C104K5RAC7800CT-ND	C-USC1206
1	R1	182K 1%	311-182KFRCT-ND	R-US_M1206

Place the board on the cookie sheet then place the cookie sheet on the hot plate. Watch the board carefully so you can nudge parts back into place if they move. Carefully remove and let cool when everything flows nicely.

Through-Hole Parts

Prepare the D1 Mini first. Hopefully you bought your minis with the headers included. Solder the male headers onto the D1 Mini, tacking only one pin at first to make sure the header is perpendicular to the board, then solder the rest of the pins.

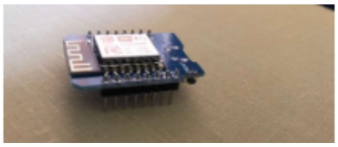


Figure 2: Mini with male headers installed

To ensure good alignment of the short female headers on the circuit board, install them onto the pins of the Mini, then insert the headers and Mini into the BACK of the MQTT monitor board.

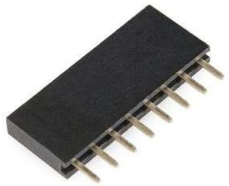


Figure 3: Short female header.

Solder the remaining pins of the headers then remove the Mini.

Using the same method, install a 4 pin male header on the LCD and the female headers on the front of the main board. Unplug the LCD for now since it would otherwise be in the way of the installation of the rotary encoder and the battery charger.

If your rotary encoder has a right angle header installed, carefully remove it and install a straight 5 pin header. Solder it onto the main board as shown.

Install a SPST or SPDT toggle switch in the holes provided. If the switch only has two terminals, insert them into the holes closest to the edge of the board.

Cut 7 single pin male headers to support the battery charger board and solder the battery charger onto the main board. Solder a pin and check the alignment of the charger before soldering the other pins.

Figure 4: Through-Hole Parts Placement

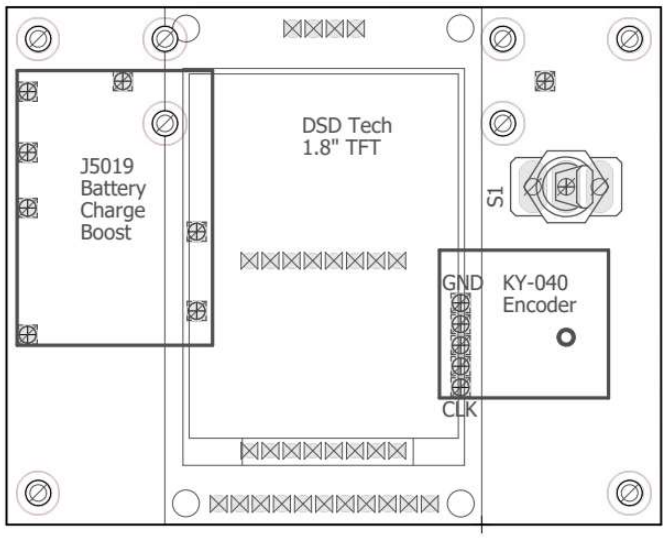


Table 2: Through-Hole Parts Needed

Qty	Part	Value	Part Number	Device
1	na	Charge/Boost	ALAMSCN TP4056 (Amazon)	na
1	na	Rotary Encoder	(Amazon or Ebay)	KY-040
1	S1	SPST Toggle Switch	M2011S3A1W03	na
1	U$1	WEMOS-D1-MINI	(Amazon or Ebay)	See Instructions

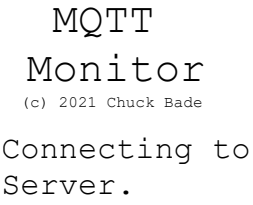
Testing

1. Make sure the toggle switch is in the OFF position (toward the edge of the board).
2. Solder a suitable battery to the BAT+ and BAT- terminals.
3. Plug a USB charger into the battery charger. The red LED will turn on until the battery is charged, then the green LED will turn on.
4. Plug in the LCD.
5. Program a mini. See the Programming section below.

Operation

Main Menu

Turn on the toggle switch. The LCD should display solid white for a second, then go to the title screen where it will say “Connecting to server.”



If it fails to connect to the server/broker, after a while it will say “MQTT server not found” and then go to the MQTT Server screen. If the server is found and a connection is made, MQTT Monitor will clear the screen then display the retained messages on the server. If there are no retained messages, the screen will be blank until a message arrives.

To go to the main menu, press down on the knob.

NOTE: You don’t have to wait for all the messages to be displayed. You can press the knob at any time to interrupt the scrolling messages.

You will see the main menu screen with the menu highlighted. In MQTT Monitor, whenever you see a highlighted item, that's the item that can be changed by rotating the knob and selected by pressing the knob.

```
  MAIN MENU:
Menu=MESSAGE
```

To get to the CLEAR function, turn the knob one click. To get to the SERVER function turn the knob two clicks. To enter the desired function, press the knob while that function is displayed.

NOTE: Sometimes the knob selection will "bounce", causing the program to skip over a function. If this happens, you can just hit the knob a few more times to cycle through back to the desired function. Most functions have a confirmation option to allow you to cycle through without actually taking the action.

Message Function

When you enter the message function you will initially see the message setup screen with the message type highlighted.

```
MESSAGE SETUP
Type=sensor
Addr=0
Mesg=ACTIVE
```

If you change the message type, the available messages change also.

```
MESSAGE SETUP
Type=turnout
Addr=0
Mesg=THROWN
```

Press the knob to select the type. Now the address needs to be selected.

```
MESSAGE SETUP
Type=turnout
Addr=0 ±50
Mesg=THROWN
```

The ±50 label indicates that each click of knob rotation changes the address by 50. Select the nearest multiple of 50 for the address you want to send then press the knob.

```
MESSAGE SETUP
Type=turnout
Addr=150
Mesg=THROWN
```

Notice the ±50 label went away. This indicates that you can now change the address by one. Select the desired address.

```
MESSAGE SETUP
Type=turnout
Addr=153
Mesg=THROWN
```

For turnouts you can select THROWN, CLOSED, or none. For sensors you can select ACTIVE, INACTIVE, or none. If none is selected, an empty message will be sent. This allows retained messages to be deleted for that address only.

```
MESSAGE SETUP
Type=turnout
Addr=153
Mesg=CLOSED
Send=NO
```

Once the message is selected, the send option is displayed so you can confirm whether or not to send the message. Select NO or YES. If you select yes, the message will be sent and the new message will be displayed at the bottom of the message list.

```
...
T153 CLOSED
```

Clear Function

When you enter the clear function you will initially see the number of retained messages and a description of the function. The choice here is CANCEL to return to the message list or START to clear all messages.

The value for free memory is for the ESP8266 device running the program. It is primarily a troubleshooting aid for program development.

```
Retained
Messages=49
This function provides a
way to clear retained
messages that can cause
performance issues for the
clients on the network.
This deletes all messages.
Free memory: 45760
Clear=CANCEL
```

If START is selected, all messages will be clear which should only take a few seconds.

```
Clearing...
Done.
```

After the messages are cleared, the program will return to the message list, but since all the retained messages were cleared, the screen will be blank until a new message arrives at the server.

Server Function

When you enter the server function, the program will display the current server IP address. This program assumes the first two numbers (octets) in the IP address will be 192 and 168. If your application required an IP that starts with different octets than 192.168, then you will need to change the sketch to allow that.

```
MQTT Server:
This function provides a
way to change the address
of the MQTT server.
IP=192.168.1.13
IP High=1
IP Low=13
```

After selecting the high and low numbers, you will need to confirm that you want to save the new values.

```
MQTT Server:
This function provides a
way to change the address
of the MQTT server.
IP=192.168.1.13
IP High=1
IP Low=13
Save=YES
```

If the YES option is chosen, the values will be saved in EEPROM.

```
Saving...
Done.
```

If the MQTT server cannot be found at the newly selected IP address then an error will be displayed for a second and the program will return to the server function.

```
MQTT Server
   not found.
```

If this message is displayed, the user will need to find and enter the correct numbers.

Programming

Setting up Arduino IDE

1. Install Arduino IDE.

2. In Arduino IDE, go to File->Preferences and enter http://arduino.esp8266.com/stable/package_esp8266com_index.json into the “Additional Boards Manager URLs” field.
3. Go to Tools->Board->Boards Manager…, search for ESP8266 and press the install button for the “ESP8266 by ESP8266 Community“.

You will probably need the following libraries:

- PubSubClient by Nick O'Leary (If you have other MQTT sketches, you have this already)
- Adafruit GFX Library
- Adafruit ST7735 and ST7789 Library

D1 Mini Configuration

1. Go to https://github.com/chuckbade/Wifi_MQTT_Modules
2. Download the latest ino file from the MQTT Monitor directory.
3. Open it in Arduino IDE.
4. If you don’t have a password file, create one in Documents/Arduino/libraries/Personal/SSIDPASSWD.h
5. In this file, place the following definitions:

   ```
   #define MYSSID "myssid"
   #define PASSWD "mypasswd"
   #define MQTTIP "192.168.1.13"
   ```

6. You will need to set up an MQTT broker somewhere on your network and have it start automatically. (See my document **MQTT Instructions).** The IP address should be listed in the file above for the sake of the other sketches. The MQTT Monitor uses its own configuration for this IP address.
7. Change the **JRMISensorNumber** to something that is not used by any sensor on your layout. This value is used as an ID for connecting to the server.
8. Connect the Mini to the USB cable and compile the sketch.
9. After it says it is resetting the device, disconnect the Mini and install it on the MQTT monitor board. **Be very careful installing the Mini on the board, making sure it is properly oriented and aligned with the correct pins. Mini’s will self-destruct if off by just one pin.**