

```
1
2
3 #include <PubSubClient.h>
4 #include <ESP8266WiFi.h>
5 #include <ArduinoOTA.h>
6
7 void callback(char* topic, byte* payload, unsigned int length);
8
9 //EDIT THESE LINES TO MATCH YOUR SETUP
10#define MQTT_SERVER "192.168.1.60" //you MQTT IP Address
11const char* ssid = "NETHEAD";
12const char* password = "wifi@office";
13
14
15
16const int switchPin1 = D1;
17const int switchPin2 = D2;
18const int switchPin3 = D3;
19const int switchPin4 = D5;
20
21
22
23char const* switchTopic1 = "/house/switch1/";
24char const* switchTopic2 = "/house/switch2/";
25char const* switchTopic3 = "/house/switch3/";
26char const* switchTopic4 = "/house/switch4/";
27
28
29 WiFiClient wifiClient;
30 PubSubClient client(MQTT_SERVER, 1883, callback, wifiClient);
31
32 void setup() {
33     //initialize the switch as an output and set to LOW (off)
34     pinMode(switchPin1, OUTPUT); // Relay Switch 1
35     digitalWrite(switchPin1, LOW);
36
37     pinMode(switchPin2, OUTPUT); // Relay Switch 2
38     digitalWrite(switchPin2, LOW);
39
40     pinMode(switchPin3, OUTPUT); // Relay Switch 3
41     digitalWrite(switchPin3, LOW);
42
43     pinMode(switchPin4, OUTPUT); // Relay Switch 4
44     digitalWrite(switchPin4, LOW);
45
46     ArduinoOTA.setHostname("My Arduino WEMO"); // A name given to your ESP8266 module
47     when discovering it as a port in ARDUINO IDE
48     ArduinoOTA.begin(); // OTA initialization
49
50     //start the serial line for debugging
51     Serial.begin(115200);
52     delay(100);
53
54     //start wifi subsystem
55     WiFi.begin(ssid, password);
56     //attempt to connect to the WIFI network and then connect to the MQTT server
57     reconnect();
58
59     //wait a bit before starting the main loop
60     delay(2000);
61 }
62
63 void loop(){
64
65     //reconnect if connection is lost
66     if (!client.connected() && WiFi.status() == 3) {reconnect();}
67
68     //maintain MQTT connection
```

```
69     client.loop();
70
71 //MUST delay to allow ESP8266 WIFI functions to run
72 delay(10);
73 ArduinoOTA.handle();
74 }
75
76 void callback(char* topic, byte* payload, unsigned int length) {
77
78     //convert topic to string to make it easier to work with
79     String topicStr = topic;
80
81
82     //Print out some debugging info
83     Serial.println("Callback update.");
84     Serial.print("Topic: ");
85     Serial.println(topicStr);
86
87     if (topicStr == "/house/switch1/")
88     {
89
90         //turn the switch on if the payload is '1' and publish to the MQTT server a
91         //confirmation message
92         if(payload[0] == '1'){
93             digitalWrite(switchPin1, HIGH);
94             client.publish("/house/switchConfirm1/", "1");
95         }
96
97         //turn the switch off if the payload is '0' and publish to the MQTT server a
98         //confirmation message
99         else if (payload[0] == '0'){
100             digitalWrite(switchPin1, LOW);
101             client.publish("/house/switchConfirm1/", "0");
102         }
103
104     else if (topicStr == "/house/switch2/")
105     {
106         //turn the switch on if the payload is '1' and publish to the MQTT server a
107         //confirmation message
108         if(payload[0] == '1'){
109             digitalWrite(switchPin2, HIGH);
110             client.publish("/house/switchConfirm2/", "1");
111         }
112
113         //turn the switch off if the payload is '0' and publish to the MQTT server a
114         //confirmation message
115         else if (payload[0] == '0'){
116             digitalWrite(switchPin2, LOW);
117             client.publish("/house/switchConfirm2/", "0");
118         }
119
120     else if (topicStr == "/house/switch3/")
121     {
122         //turn the switch on if the payload is '1' and publish to the MQTT server a
123         //confirmation message
124         if(payload[0] == '1'){
125             digitalWrite(switchPin3, HIGH);
126             client.publish("/house/switchConfirm3/", "1");
127         }
128
129         //turn the switch off if the payload is '0' and publish to the MQTT server a
130         //confirmation message
131         else if (payload[0] == '0'){
132             digitalWrite(switchPin3, LOW);
133             client.publish("/house/switchConfirm3/", "0");
134         }
135     }
```

```

132     else if (topicStr == "/house/switch4/")
133     {
134         //turn the switch on if the payload is '1' and publish to the MQTT server a
135         //confirmation message
136         if(payload[0] == '1'){
137             digitalWrite(switchPin4, HIGH);
138             client.publish("/house/switchConfirm4/", "1");
139         }
140
141         //turn the switch off if the payload is '0' and publish to the MQTT server a
142         //confirmation message
143         else if (payload[0] == '0'){
144             digitalWrite(switchPin4, LOW);
145             client.publish("/house/switchConfirm4/", "0");
146         }
147     }
148
149 void reconnect() {
150
151     //attempt to connect to the wifi if connection is lost
152     if(WiFi.status() != WL_CONNECTED){
153         //debug printing
154         Serial.print("Connecting to ");
155         Serial.println(ssid);
156
157         //loop while we wait for connection
158         while (WiFi.status() != WL_CONNECTED) {
159             delay(500);
160             Serial.print(".");
161         }
162
163         //print out some more debug once connected
164         Serial.println("");
165         Serial.println("WiFi connected");
166         Serial.println("IP address: ");
167         Serial.println(WiFi.localIP());
168     }
169
170     //make sure we are connected to WIFI before attempting to reconnect to MQTT
171     if(WiFi.status() == WL_CONNECTED){
172         // Loop until we're reconnected to the MQTT server
173         while (!client.connected()) {
174             Serial.print("Attempting MQTT connection...");
175
176             // Generate client name based on MAC address and last 8 bits of microsecond counter
177             String clientName;
178             clientName += "openhab";
179             uint8_t mac[6];
180             WiFi.macAddress(mac);
181             clientName += macToStr(mac);
182
183             //if connected, subscribe to the topic(s) we want to be notified about
184             //EJ: Delete "mqtt_username", and "mqtt_password" here if you are not using any
185             if (client.connect((char*) clientName.c_str(),"openhabian", "openhabian")) {
186                 //EJ: Update accordingly with your MQTT account
187                 Serial.print("\tMQTT Connected");
188                 client.subscribe(switchTopic1);
189                 client.subscribe(switchTopic2);
190                 client.subscribe(switchTopic3);
191                 client.subscribe(switchTopic4);
192                 //EJ: Do not forget to replicate the above line if you will have more than the
193                 //above number of relay switches
194
195                 //otherwise print failed for debugging
196                 else{Serial.println("\tFailed."); abort();}
197             }

```

```
197     }
198 }
199
200 //generate unique name from MAC addr
201 String macToStr(const uint8_t* mac) {
202
203     String result;
204
205     for (int i = 0; i < 6; ++i) {
206         result += String(mac[i], 16);
207
208         if (i < 5){
209             result += ':';
210         }
211     }
212
213     return result;
214 }
```