

/*

Experiment No. : 16

Statement : MQTT protocol with ESP8266 Witty Cloud Development Board and Adafruit IO.

Date of Exp. : xx/xx/xxxx

Author : Ayush Deo (A-26)

*/

#include <Adafruit_MQTT.h>

#include <Adafruit_MQTT_Client.h>

#include <Adafruit_MQTT_FONA.h>

#include <ESP8266WiFi.h> // library file for ESP8266

//#include "Adafruit_MQTT.h" // library included through
Adafruit IO Arduino

//#include "Adafruit_MQTT_Client.h" // library included through
Adafruit IO Arduino

// pinout for wittyBoard

#define led 2 // debug LED, tiny blue

#define red 15 // RGB LED red

#define green 12 // RGB LED green

#define blue 13 // RGB LED blue

#define ldr A0 // LDR

```

#define WLAN_SSID      "Jain"

#define WLAN_PASS      "jainansh1"


#define AIO_SERVER      "io.adafruit.com"

#define AIO_SERVERPORT  1883                // mqtt: 1883,
secure-mqtt: 8883

#define AIO_USERNAME    "vaishnavirathi"

#define AIO_KEY          "aio_uahg23JFNrHXorotr4KPw8mBA6nc"


WiFiClient                                client;
// declare client


Adafruit_MQTT_Client  mqtt(&client,  AIO_SERVER,  AIO_SERVERPORT,
AIO_USERNAME, AIO_KEY);                // declare MQTT client


Adafruit_MQTT_Publish  lightintensity  =  Adafruit_MQTT_Publish(
&mqtt, AIO_USERNAME "/feeds/LDR");    // declare publisher


Adafruit_MQTT_Subscribe          redbutton                                =
Adafruit_MQTT_Subscribe(&mqtt,  AIO_USERNAME  "/feeds/Red  LED");
// declare subscriber


Adafruit_MQTT_Subscribe          greenbutton                             =
Adafruit_MQTT_Subscribe(&mqtt,  AIO_USERNAME  "/feeds/Green LED");
// declare subscriber


Adafruit_MQTT_Subscribe          bluebutton                              =
Adafruit_MQTT_Subscribe(&mqtt,  AIO_USERNAME  "/feeds/Blue  LED");
// declare subscriber

```

```

void                                     MQTT_connect();
// bug fixes

void setup() {
    // put your setup code here, to run once:
    pinMode(led, OUTPUT);
    pinMode(red, OUTPUT);
    pinMode(green, OUTPUT);
    pinMode(blue, OUTPUT);
    Serial.begin(115200);
    delay(10);
    Serial.println(F("Adafruit MQTT demo"));
    // Connect to WiFi access point.
    Serial.println();
    Serial.print("Connecting to ");
    Serial.println(WLAN_SSID);

    WiFi.begin(WLAN_SSID, WLAN_PASS);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println();
    Serial.println("WiFi connected");
}

```

```

        Serial.println("IP          address:          ");
Serial.println(WiFi.localIP());

    // Setup MQTT subscription for onoff feed.

    mqtt.subscribe(&redbutton);

    mqtt.subscribe(&greenbutton);

    mqtt.subscribe(&bluebutton);

}

void loop() {

    // put your main code here, to run repeatedly:

    MQTT_connect();

Adafruit_MQTT_Subscribe *subscription;

    while ((subscription = mqtt.readSubscription(5000))) {

        if (subscription == &redbutton) {

            Serial.print(F("Got: "));

            Serial.println((char *)redbutton.lastread);

            if(strcmp((char*)redbutton.lastread, "ON"))

                digitalWrite(red, LOW);

            else

                digitalWrite(red, HIGH);

        }

        if (subscription == &greenbutton) {

            Serial.print(F("Got: "));

            Serial.println((char *)greenbutton.lastread);

            if(strcmp((char*)greenbutton.lastread, "ON"))

```

```

        digitalWrite(green, LOW);
    else
        digitalWrite(green, HIGH);
}

if (subscription == &bluebutton) {
    Serial.print(F("Got: "));
    Serial.println((char *)bluebutton.lastread);
    if(strcmp((char*)bluebutton.lastread, "ON"))
        digitalWrite(blue, LOW);
    else
        digitalWrite(blue, HIGH);
}
}

Serial.print(F("\nSending light val "));
Serial.print(analogRead(ldr));
Serial.print("...");
if (! lightintensity.publish(analogRead(ldr)))
    Serial.println(F("Failed"));
else
    Serial.println(F("OK!"));
}

// Function to connect and reconnect as necessary to the MQTT
server.

void MQTT_connect() {

```

```

int8_t ret;

// Stop if already connected.
if (mqtt.connected()) {
    return;
}

Serial.print("Connecting to MQTT... ");

uint8_t retries = 3;

while ((ret = mqtt.connect()) != 0) { // connect will return 0
for connected

    Serial.println(mqtt.connectErrorString(ret));

        Serial.println("Retrying MQTT connection in 5
seconds...");

    mqtt.disconnect();

    delay(5000); // wait 5 seconds

    retries--;

    if (retries == 0) {

        // basically die and wait for WDT to reset me

        while (1);

    }

}

Serial.println("MQTT Connected!");
}

```



