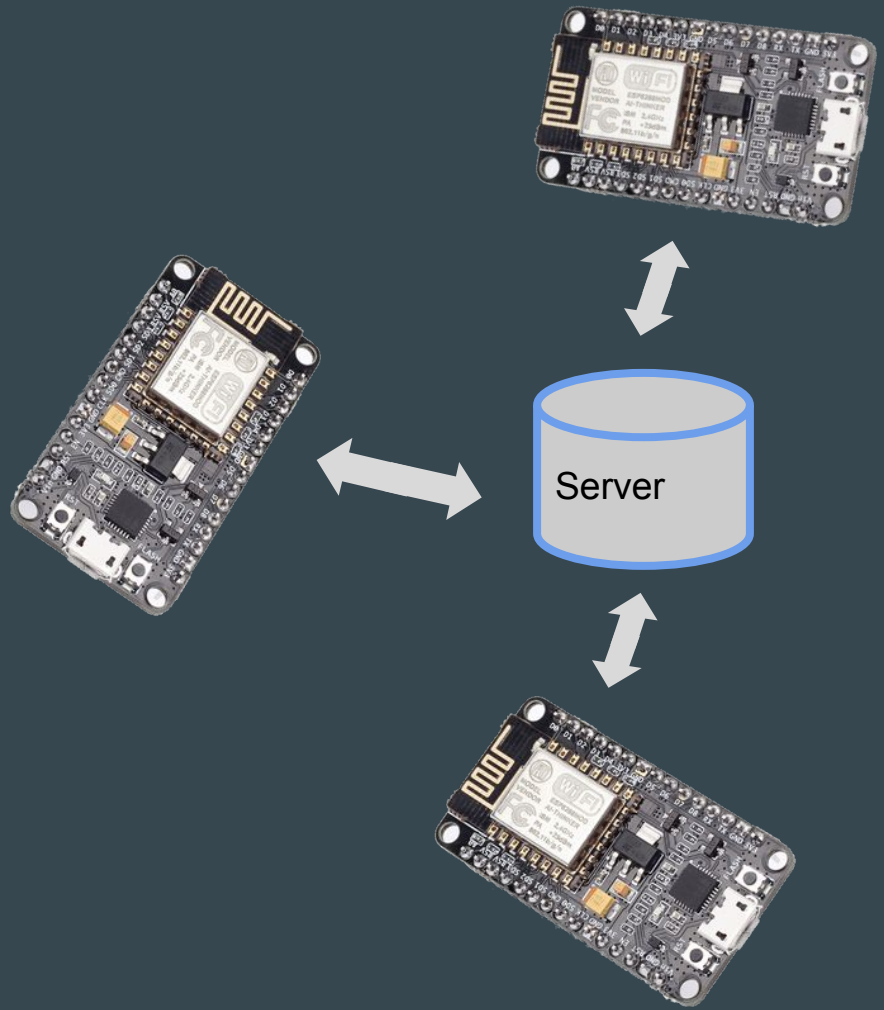
Connected Devices

By: Murtaza Ameen
School of Electronics, DAVV, Indore

Problem Statement:

Implementation of a network of Devices.

Using MQTT protocol.



What is MQTT protocol ?

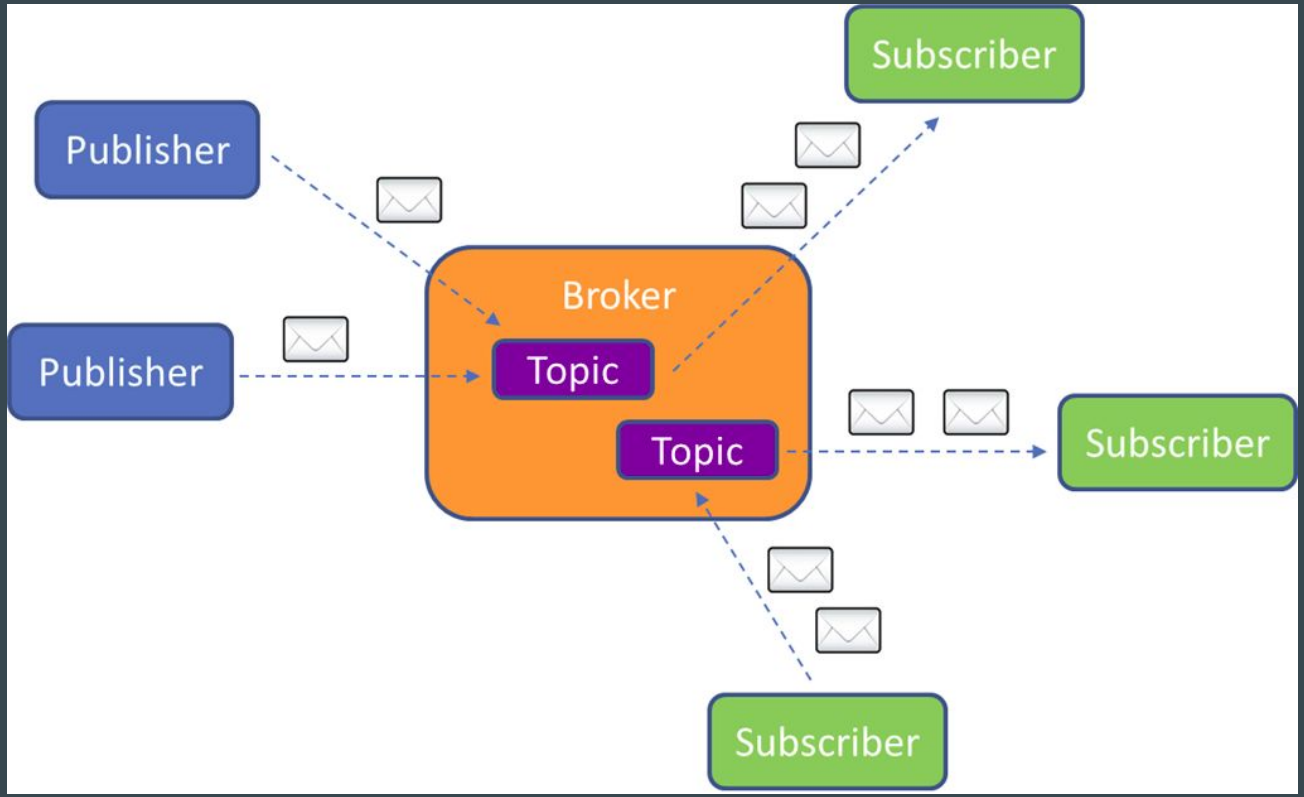
MQTT stands for MQ Telemetry Transport.

It is a publish/subscribe, extremely simple and lightweight messaging protocol.

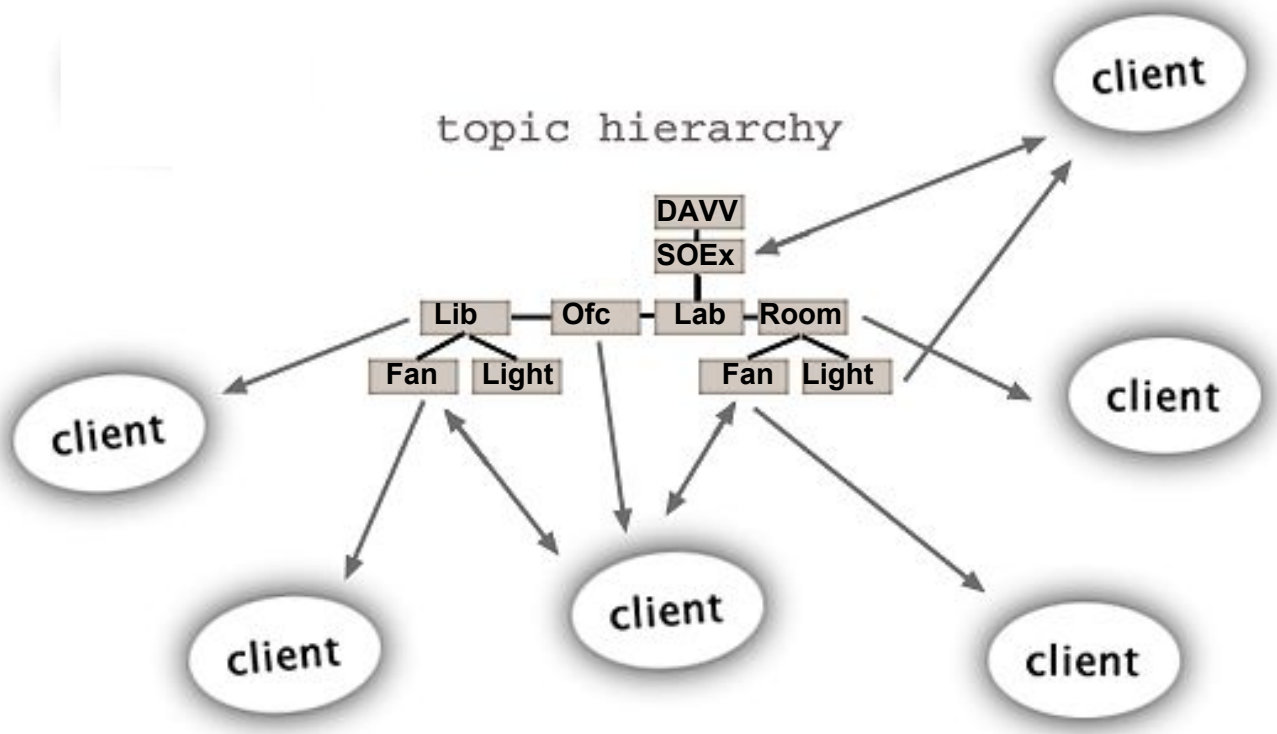
The design principles are to minimise network bandwidth and device resource requirements whilst also attempting to ensure reliability and some degree of assurance of delivery.

These principles also turn out to make the protocol ideal of the emerging “machine-to-machine” (M2M) or “Internet of Things” world of connected devices, and for mobile applications where bandwidth and battery power are at a premium.

How does MQTT works ?



Publish/Subscribe



Mosquitto Broker/Server

Eclipse Mosquitto is an open source message broker that implements the MQTT protocol versions 3.1 and 3.1.1. Mosquitto is lightweight and is suitable for use on all devices from low power single board computers to full servers.

The Mosquitto project also provides a C library for implementing MQTT clients, and the very popular mosquitto_pub and mosquitto_sub command line MQTT clients.

Install Mosquitto...

sudo apt-get install mosquitto

mosquitto_sub -h test.mosquitto.org -t “soex”

Mosquitto_pub -h test.mosquitto.org -t “soex” -m “hello”

Implementation of MQTT client on NodeMCU

https://github.com/knolleary/pubsubclient

Add pubsubclient library in Arduino IDE.

```
#include <ESP8266WiFi.h>
#include <PubSubClient.h>

// Update these with values suitable for your network.

const char* ssid = "RobuByte";
const char* password = "esp@8265";
const char* mqtt_server = "test.mosquitto.org";

WiFiClient espClient;
PubSubClient client(espClient);

void setup_wifi() {
  delay(10);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
  }
}
```

```
void callback(char* topic, byte* payload, unsigned int length) {
  String s="";
  for (int i = 0; i < length; i++) {
        s += ((char)payload[i]);
  }
  if(topic.equals("room")){
        if (s.equals("on"))
        turnOn();
        else
        turnOff();
        }}
void reconnect() {
  // Loop until we're reconnected
  while (!client.connected()) {
        String clientId = "Murtaza";
        if (client.connect(clientId.c_str())) {
        client.subscribe("room");
        } else {
        delay(5000);
        }  }}
```

```
void setup() {
  pinMode(D4, OUTPUT);      // Initialize the BUILTIN_LED pin as an output

  setup_wifi();
  client.setServer(mqtt_server, 1883);
  client.setCallback(callback);
}

void loop() {
  if (!client.connected()) {
        reconnect();
  }
  client.loop();
}
void turnOn(){
  digitalWrite(D4,HIGH);
  client.publish(“status/light”,”On”);
  }
void turnOff(){
  digitalWrite(D4,LOW);
  client.publish(“status/light”,”On”);
  }
```

Local MQTT Mosquitto broker

Find WLAN ip of your PC.

Run ‘mosquitto’ or ‘sudo systemctl restart mosquitto’ on terminal

Change server name in nodeMCU code.

```
const char* mqtt_server = "test.mosquitto.org"
to
const char* mqtt_server = "192.168.43.253"//write ip of your pc
```

Run the code again without internet.

Conclusion

MQTT is an IoT protocol.

Broker is nothing but a kind of server.

Mosquitto is most common open source Broker for MQTT.