***TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI***

------o0o------

****

Báo cáo môn Hệ Nhúng

**Đề tài: Lập trình nhúng cho ESP8266 qua Internet bằng giao thức MQTT**

**Demo: Local Area Network**

***Giảng viên hướng dẫn:* TS . Ngô Lam Trung**

|  |  |  |
| --- | --- | --- |
| ***Nhóm Sinh viên :*** | **Họ và tên** | **MSSV** |

**Trịnh Hoàng Anh 2017**

**Phạm Thành Phong 20173297**

**Hoàng Thị Hiền 20173102**

*Hà Nội, tháng 6 năm 2020*

1. **Các thiết bị sử dụng:**
   1. **ESP8266 là gì?**

ESP là một mạch vi điều khiển có thể giúp chúng ta điều khiển các thiết bị điện tử.

Điều đặc biệt của nó, đó là sự kết hợp của module Wifi tích hợp sẵn bên trong con vi điều khiển chính.

* 1. **DHT11 là gì ?**

DHT 11 là cảm biến nhiệt độ , độ ẩm rất thông dụng hiện nay vì có chi phí rẻ và rất dễ lấy dữ liệu thông qua giao tiếp 1-wire (giao tiếp digital 1-wire truyền dữ liệu duy nhất). Cảm biến được tích hợp bộ tiền xử lí tín hiệu giúp dữ liệu nhận về được chính xác mà không cần phải qua bất kì tính toán nào.

* Đặc điểm:

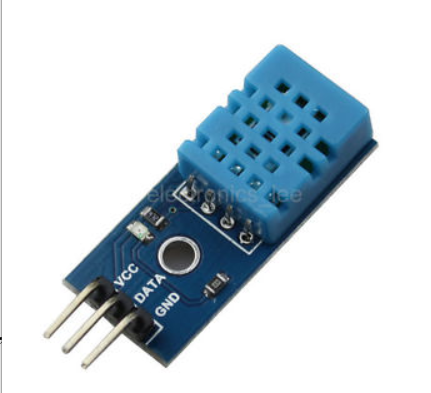
Điện áp hoạt động: 3V-5V(DC)

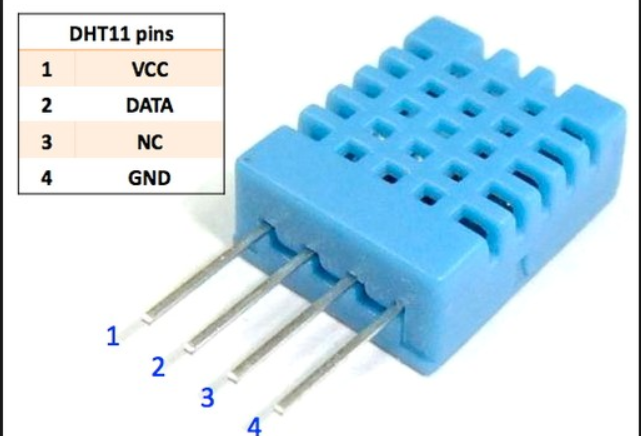
Dải độ ẩm hoạt động: 20%-90% RH

Tần số lấy mỗ tối đa: 1 Hz

Khoảng cách truyền tối đa: 20m

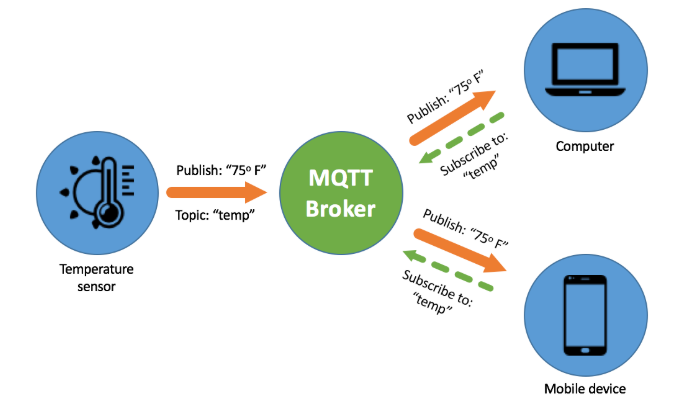
* Sơ đồ chân cảm biến DHTT11 gồm 2 chân cấp nguồn, và 1 chân tín hiệu. Hiện nay, thông dụng dụng ngoài thị trường có hai loại đóng gói cho DHT11: 3 chân và 4 chân





**1.3 Mosquitto**

Mosquitto là một MQTT Broker mã nguồn mở cho phép thiết bị truyền nhận dữ liệu theo giao thức MQTT versions 3.1 và 3.1.1 – Một giao thức nhanh, nhẹ theo mô hình publish/subscribe được sử dụng rất nhiều trong lĩnh vực Internet of Things. Bài viết này cung cấp các hướng dẫn để các bạn có thể cài đặt được Mosquitto Broker trên máy tính Windows.



**1.4 XAMPP**

***XAMPP là chương trình tạo web server được ứng dụng trên các hệ điều hành Linux, MacOS, Windows, Cross-platform, Solaris***.

**1.5 Máy bơm nước**

**2. Cơ sở lý thuyết**

**2.1. Smartconfig:**

**2.1.1. Khái niệm:**

Smartconfig là một khái niệm được nhắc đến khi muốn cấu hình thông tin cho thiết bị Wifi kết nối nhanh chóng đến Internet nhất từ người dùng bằng chính điện thoại của họ.

Chúng ta gửi thông tin mạng wifi (bao gồm tên wifi và password wifi) cho ESP thông qua Smartphone thay cho bằng cách thông thường là phải khai báo thông tin này trong chương trình và nạp firmware xuống.

**2.2.2 .Lợi ích của Smartconfig :**

* Dễ Dàng cấu hình wifi cho ESP8266 thông qua smartphone.
* Không cần phải nạp lại code để cấu hình.
* Có thể dùng smartconfig để cấu hình nhiều thiết bị một lúc.

.

**2.2.MQTT protocol:**

MQTT (Message Queue Telemetry Transport) là một giao thức mã nguồn mở để truyền các message giữa nhiều client (Publisher và Subscriber) thông qua một Broker trung gian, được thiết kế để đơn giản và dễ dàng triển khai.

Kiến trúc MQTT dựa trên Broker trung gian và sử dụng kết nối TCP ong-lived từ các Client đến Broker.

MQTT hỗ trợ tổ chức hệ thốn Topics có tính phân cấp , như một hệ thống tập tin, cung cấp nhiều lựa chọn điều khiển và QoS ( Quality of Service).

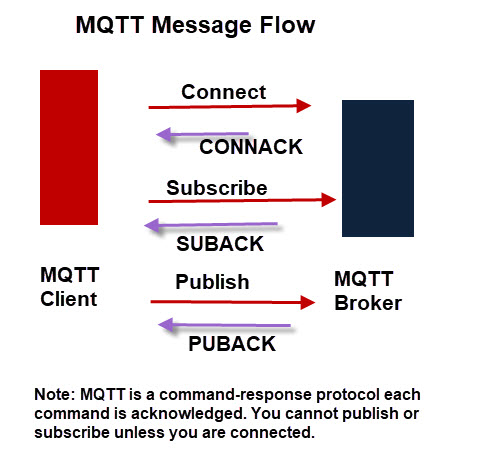
MQTT là một giao thức khá nhẹ nên có thể sử dụng cho truyền thông hai chiều thông qua các mạng có độ trễ cao và độ tin cậy thấp , nó cũng tương thích với các thiết bị tiêu thụ điện năng thấp.

# Important Points to Note

* Clients **do not have addresses** like in email systems, and messages are not sent to clients.
* Messages are **published to a broker on a topic**.
* The job of an MQTT broker is to**filter messages** based on topic, and then **distribute them to subscribers**.
* A client can receive these messages by subscribing to that topic on the same broker
* There is **no direct connection** between a publisher and subscriber.
* **All clients** can publish (broadcast) and subscribe (receive).
* MQTT brokers do not normally store messages.

# MQTT Client-Broker Connections

* MQTT uses TCP/IP to connect to the broker.
* TCP is a **connection orientated** protocol with error correction and guarantees that packets are received in order.
* You can consider a TCP/IP connection to be**similar to a telephone connection**.
* Once a telephone connection is established you can talk over it until **one party hangs up**.
* Most MQTT clients will connect to the broker and remain connected even if they aren’t sending data.
* Connections are acknowledged by the broker using a **Connection acknowledgement message**.
* *You cannot publish or subscribe unless you are connected.*



MQTT clients publish a **keepalive message** at regular intervals (usually 60 seconds) which tells the broker that the client is still connected.

# Websockets

**Websockets** allows you to receive MQTT data directly into a web browser.

This is important as the web browser may become the DE-facto interface for displaying MQTT data.

MQTT websocket support for web browsers is provided by the **Javascript MQTT Client**.

# The Client Name or Client ID

All clients are required to have a **client name** or ID.

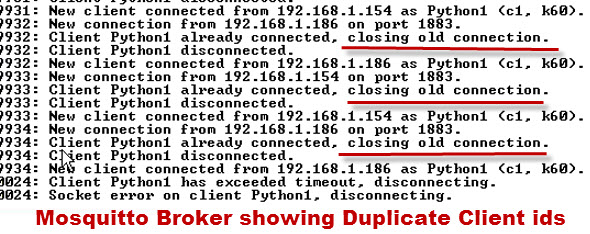
The client name is used by the MQTT broker to track subscriptions etc.

Client names**must also be unique.**

If you attempt to connect to an MQTT broker with the same name as an existing client then the**existing client connection is dropped.**

Because most MQTT clients will **attempt to reconnect** following a disconnect this can result in a loop of**disconnect and connect.**

The screen shot below show what happens when I try and connect a client to the broker using the same client id as an existing connected client.



# Last Will Messages

The idea of the **last will message** is to notify a subscriber that the publisher is unavailable due to network outage.

The last will message is set by the publishing client, and is set on a per topic basis which means that each topic can have its own last will message.

This means that each topic cab have its own last will message associated with it.

The message is stored on the broker and sent to any subscribing client (to that topic) if the connection to the publisher fails.

# Last Will and Testament

The**last will and testament** message is used to notify subscribers of an unexpected shut down of the publisher.

Each topic can have a**last will and testament** message stored on the broker.

# Keep Alives

MQTT uses a TCP/IP connection which is normally left open by the client so that is can send and receive data at any time.

If no data flows over an open connection for a certain **time period** then the client will **generate a PINGREQ** and expect to receive a **PINGRESP** from the broker.

# 2.3 **Eclipse Paho JavaScript Client**

The Paho JavaScript Client is an MQTT browser-based client library written in Javascript that uses WebSockets to connect to an MQTT Broker.

<https://www.eclipse.org/paho/clients/js/>

# 2.4. **PubSubClient library for Arduino**

This library allows you to send and receive MQTT messages. It supports the latest MQTT 3.1.1 protocol and can be configured to use the older MQTT 3.1 if needed. It supports all Arduino Ethernet Client compatible hardware, including the Intel Galileo/Edison, ESP8266 and TI CC3000.

<https://github.com/knolleary/pubsubclient>

### 3. Thiết kế phần cứng

### Sơ đồ máy bơm

### 

### Sơ đồ hoàn chỉnh

### 4. Thiết kế phần mềm

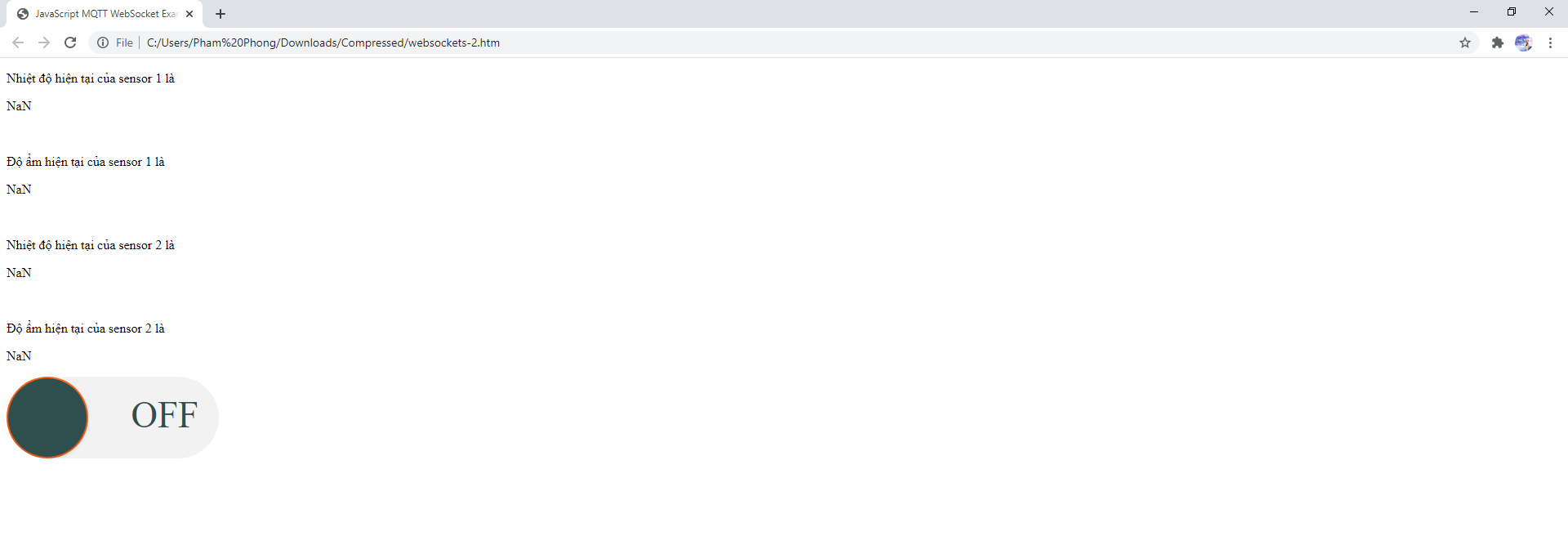
# 4.1 Mosquitto broker

### 

### Thiết lập 2 socket kết nối với website và kết nối với ESP8266: cổng 1883 của wifi adapter nhận dữ liệu của ESP8266 và cổng 1884 của wifi adapter kết nối với website

### 

# 4.2 Website



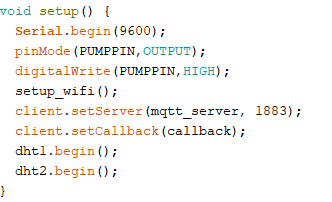
Link code: <https://github.com/phamphong9981/Esp8266OverTheAirByMosquitto/blob/master/Esp8266OverTheAirByMosquitto.htm>

# 4.3 Code Arduino

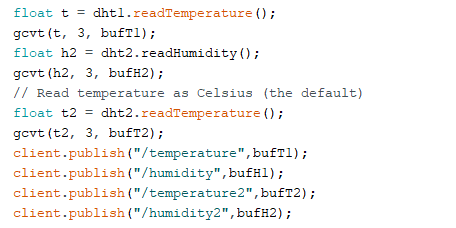
Đoạn code thiết lập smart config:



Đoạn code khởi tạo:

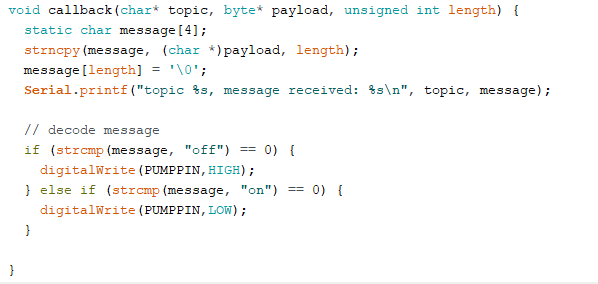


Publish dữ liệu đến Mosquitto:



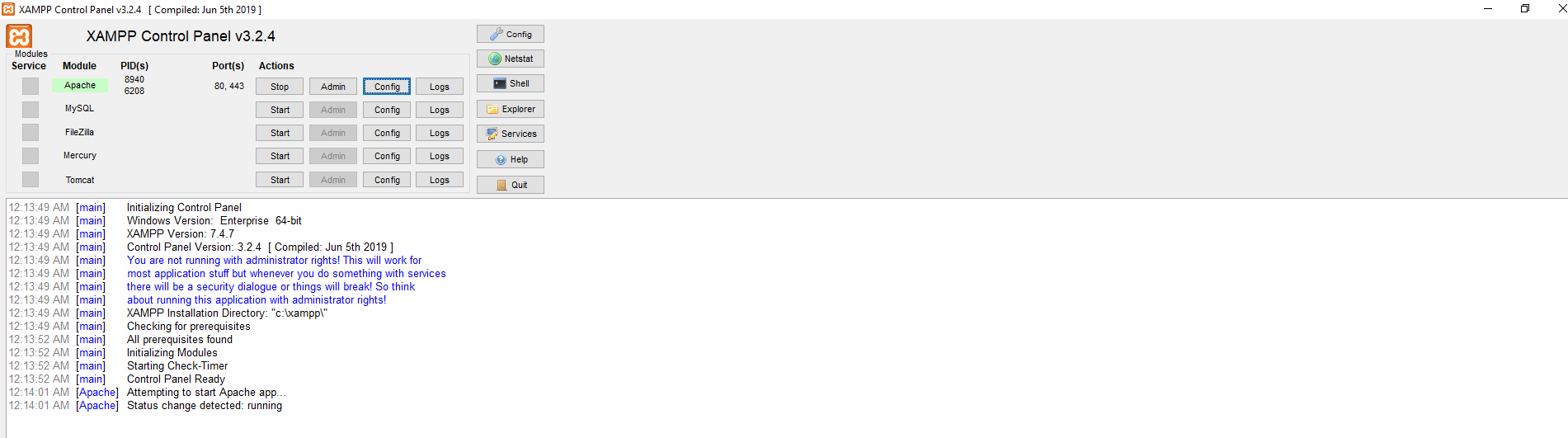
Subcribe topic “button”:





Link code: <https://github.com/phamphong9981/Esp8266OverTheAirByMosquitto/blob/master/Prj2.ino>

# 4.4 Sử dụng apache để mở port cho website ở cổng 80 của wifi adapter



### 5. Kết quả

### Trên cùng một LAN, một thiết bị cùng mạng có thể truy nhập vào website để quản lý sự kiện

### 

### 6. Hạn chế

### Do đây là bản demo nên sẽ có nhiều thiếu sót và chưa phù hợp với thực tế, ví dụ như việc lưu trữ đăng nhập wifi trên ESP8266 trong trường hợp bị ngắt điện, chưa public ra internet, chưa lưu dữ liệu trạng thái của máy bơm truyền lại cho server,…