

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC QUẢN LÝ VÀ CÔNG NGHỆ HẢI PHÒNG**



ISO 9001:2015

**GIÁM SÁT NHIỆT ĐỘ ĐỘ ẨM VÀ ĐIỀU KHIỂN THIẾT BỊ  
ĐIỆN QUA INTERNET DÙNG MODULE ARDUINO ESP8266**

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY  
NGÀNH ĐIỆN TỬ VIỄN THÔNG**

**HẢI PHÒNG - 2019**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC QUẢN LÝ VÀ CÔNG NGHỆ HẢI PHÒNG**

**GIÁM SÁT NHIỆT ĐỘ ĐỘ ẨM VÀ ĐIỀU KHIỂN THIẾT BỊ  
ĐIỆN QUA INTERNET DÙNG MODULE ARDUINO ESP8266**

**KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY  
NGÀNH: ĐIỆN TỬ VIỄN THÔNG**

**Sinh viên : Hoàng Công Thành  
Giảng viên hướng dẫn : TS. Đoàn Hữu Chức**

**HẢI PHÒNG - 2019**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC QUẢN LÝ VÀ CÔNG NGHỆ HẢI PHÒNG**

---

**NHIỆM VỤ ĐỀ TÀI TỐT NGHIỆP**

Sinh viên: Hoàng Công Thành - Mã SV: 1512103010

Lớp: DT1901 - Ngành: Điện Tử Viễn Thông

Tên đề tài: Giám sát nhiệt độ độ ẩm và điều khiển thiết bị điện qua internet dùng module arduino esp8266

# NHIỆM VỤ ĐỀ TÀI

1. Nội dung và các yêu cầu cần giải quyết trong nhiệm vụ đề tài tốt nghiệp (về lý luận, thực tiễn, các số liệu cần tính toán và các bản vẽ).

.....

.....

.....

.....

.....

.....

.....

.....

.....

2. Các số liệu cần thiết để thiết kế, tính toán.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

3. Địa điểm thực tập tốt nghiệp.

.....

.....

.....

## CÁN BỘ HƯỚNG DẪN ĐỀ TÀI TỐT NGHIỆP

### Người hướng dẫn thứ nhất:

Họ và tên : TS. Đoàn Hữu Chúc

Học hàm, học vị : Tiến Sĩ

Cơ quan công tác : Trường Đại Học Quản Lý và Công Nghệ Hải Phòng

Nội dung hướng dẫn : Toàn bộ đề tài

### Người hướng dẫn thứ hai:

Họ và tên:.....

Học hàm, học vị:.....

Cơ quan công tác:.....

Nội dung hướng dẫn:.....

Đề tài tốt nghiệp được giao ngày tháng năm

Yêu cầu phải hoàn thành xong trước ngày tháng năm

Đã nhận nhiệm vụ ĐTTN

*Sinh viên*

Hoàng Công Thành

Đã giao nhiệm vụ ĐTTN

*Người hướng dẫn*

TS. Đoàn Hữu Chúc

***Hải Phòng, ngày ..... tháng.....năm 2019***

**Hiệu trưởng**

**GS.TS.NGƯT *Trần Hữu Nghị***

**CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  
**Độc lập - Tự do - Hạnh phúc**

**PHIẾU NHẬN XÉT CỦA GIÁNG VIÊN HƯỚNG DẪN TỐT NGHIỆP**

Họ và tên giảng viên: .....

Đơn vị công tác: .....

Họ và tên sinh viên: ..... Chuyên ngành: .....

Đề tài tốt nghiệp: .....

Tinh thần thái độ của sinh viên trong quá trình làm đề tài tốt nghiệp

.....

.....

.....

.....

1. Đánh giá chất lượng của đề án/khóa luận (so với nội dung yêu cầu đã đề ra trong nhiệm vụ Đ.T. T.N trên các mặt lý luận, thực tiễn, tính toán số liệu...)

.....

.....

.....

.....

.....

2. Ý kiến của giảng viên hướng dẫn tốt nghiệp

Được bảo vệ  Không được bảo vệ  Điểm hướng dẫn

Hải Phòng, ngày ... tháng ... năm .....

Giảng viên hướng dẫn

(Ký và ghi rõ họ tên)

**CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**

**Độc lập - Tự do - Hạnh phúc**

**PHIẾU NHẬN XÉT CỦA GIÁO VIÊN CHĂM PHẢN BIỆN**

Họ và tên giảng viên: .....

Đơn vị công tác: .....

Họ và tên sinh viên: ..... Chuyên ngành: .....

Đề tài tốt nghiệp: .....

**1. Phân nhận xét của giáo viên chăm phản biện**

.....  
.....  
.....  
.....  
.....  
.....

**2. Những mặt còn hạn chế**

.....  
.....  
.....  
.....  
.....

**3. Ý kiến của giảng viên chăm phản biện**

Được bảo vệ  Không được bảo vệ  Điểm hướng dẫn

Hải Phòng, ngày ... tháng ... năm .....

Giảng viên chăm phản biện

(Ký và ghi rõ họ tên)

## MỤC LỤC

### LỜI CẢM ƠN

### CHƯƠNG I :TỔNG QUAN VỀ ĐỀ TÀI NGHIÊN CỨU

1.1. Đặt vấn đề .....	5
1.2. Mục tiêu.....	5
1.3. Nội dung nghiên cứu .....	5
1.4. Giới hạn.....	6
1.5. Phạm vi ứng dụng .....	6

### CHƯƠNG II: CƠ SỞ LÝ THUYẾT .....

2.1. Tổng quan về IOT .....	7
2.1.1. Giới thiệu về Internet of Things (IoT) .....	7
2.1.2. Lịch sử hình thành .....	8
2.1.3 Ứng dụng của IoT .....	8
2.2 Công nghệ wifi.....	11
2.2.1. Giới thiệu.....	11
2.2.2. Công nghệ truyền nhận dữ liệu .....	12
2.3. Giới thiệu về ESP8266 NodeMCU.....	13
2.3.1. Cấu tạo của NODEMCU ESP8266 .....	13
2.3.2. Tính năng của NODEMCU ESP8266 .....	13
2.4 Module DHT11 .....	15
2.5 Các chuẩn giao tiếp được sử dụng.....	18
2.5.1 Chuẩn One-Wire .....	18
2.5.2 Chuẩn giao tiếp UART.....	18
2.6 Phần mềm.....	20
2.6.1 Giới thiệu phần mềm lập trình . .....	20
2.6.2 Cơ sở lý thuyết về APP Blynk.....	23
2.6.3 Google Assistant .....	24
2.6.4 Công cụ IFTTT ( If This Then That) . .....	27

### CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ.....

3.1. Mô tả hoạt động của hệ thống .....	28
-----------------------------------------	----



<b>3.2 Tính toán và thiết kế hệ thống</b> .....	<b>29</b>
3.2.1. Thiết kế sơ đồ khối hệ thống.....	29
3.2.2 Tính toán và thiết kế.....	30
3.2.2.1 Thiết kế khối xử lý trung tâm... ..	30
3.2.2.2. Khối ngõ ra công suất .. ..	32
3.2.2.3. Khối cảm biến .. ..	37
3.2.2.4. Khối nguồn.....	38
3.2.2.5. Khối điều khiển.....	39
3.2.2.6. Sơ đồ nguyên lý toàn mạch.....	40
<b>3.3 Nguyên lý hoạt động</b> .....	<b>41</b>
<b>CHƯƠNG 4. THI CÔNG HỆ THỐNG</b> .....	<b>42</b>
<b>4.1 Giới thiệu</b> .....	<b>42</b>
<b>4.2 Thi công hệ thống</b> .....	<b>42</b>
4.2.1 .. Thực hiện lắp ráp và ghép nối các mạch và Module.....	42
<b>4.3 Lập trình hệ thống</b> .....	<b>43</b>
4.3.1 Lưu đồ giải thuật .....	43
4.3.2 Phần mềm lập trình cho vi điều khiển.....	45
4.3.3 Chương trình điều khiển .....	46
4.3.4 Điều khiển qua điện thoại với Blynk .....	52
4.3.5 Công cụ hỗ trợ IFTTT .....	54
<b>4.4 Hướng dẫn sử dụng, thao tác</b> .....	<b>59</b>
4.4.1. Viết tài liệu hướng dẫn sử dụng .....	59
4.4.2. Quy trình thao tác .....	59
<b>CHƯƠNG 5. KẾT QUẢ, NHẬN XÉT VÀ ĐÁNH GIÁ</b> .....	<b>61</b>
<b>5.1. Giới thiệu</b> .....	<b>61</b>
<b>5.2. Kết quả đạt được</b> .....	<b>61</b>
<b>5.3. Kết quả thực nghiệm</b> .....	<b>61</b>
5.3.1. Mô hình .....	61
5.3.2. Điều khiển và giám sát thiết bị .....	62
<b>5.4. Nhận xét và đánh giá</b> .....	<b>62</b>

5.4.1. Nhận xét .....	62
5.4.2. Đánh giá .....	63
<b>CHƯƠNG 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN .....</b>	<b>65</b>
<b>6.1. Kết luận .....</b>	<b>65</b>
6.1.1. Ưu điểm.....	65
6.1.2. Nhược điểm .....	65
<b>6.2. Hướng phát triển .....</b>	<b>66</b>
<b>TÀI LIỆU THAM KHẢO</b>	

## LỜI CẢM ƠN

Để hoàn thành tốt Đồ án tốt nghiệp, em xin gửi lời chân thành cảm ơn các thầy cô trong Khoa Điện-Điện Tử đã tạo những điều kiện tốt nhất cho chúng em hoàn thành đề tài.

Đặc biệt em xin gửi lời cảm ơn sâu sắc đến Thầy Đoàn Hữu Chức - Giảng viên Khoa Điện Điện Tử đã trực tiếp hướng dẫn và tận tình giúp đỡ tạo điều kiện để hoàn thành tốt đề tài.

Cuối cùng em xin cảm ơn đến gia đình, là chỗ dựa cũng như là nguồn động viên tinh thần mỗi khi em gặp khó khăn trong học tập cũng như trong quá trình nghiên cứu để hoàn thành đề tài tốt nghiệp. Em xin chân thành cảm ơn!

Người thực hiện đề tài

Hoàng Công Thành

# CHƯƠNG I :TỔNG QUAN VỀ ĐỀ TÀI NGHIÊN CỨU

## 1.1 Đặt vấn đề

Ngày nay, công nghệ kết nối đầu tiên cần nhắc đến hiển nhiên là Wifi – công nghệ kết nối không dây phổ biến nhất hiện nay. Cũng vì tính phổ biến của dạng kết nối này mà cái tên Wifi thường bị lạm dụng để chỉ kết nối không dây nói chung. Lí do mà kết nối Wifi được ưa chuộng như vậy đơn giản là vì khả năng hoạt động hiệu quả trong phạm vi vài chục đến vài trăm mét của các mạng WLAN. Và trong thời đại công nghiệp hóa hiện đại hóa hiện nay, việc phát minh và chế tạo ra các thiết bị thông minh có khả năng điều khiển từ xa đang và sẽ rất được quan tâm và rất hữu ích cho cuộc sống hàng ngày. Vì mục tiêu công nghệ hiện đại hóa ngày càng phát triển, em đã quyết định làm một đồ án “Giám sát nhiệt độ độ ẩm và điều khiển thiết bị điện qua internet”. Đề tài của em ngoài việc điều khiển thiết bị độc lập thì còn giám sát nhiệt độ trong ngôi nhà. Khi dự án hoàn thành chúng ta có thể điều khiển các thiết bị điện trong nhà... bằng cách tương tác qua các nút nhấn để hiển thị trạng thái hoạt động trên điện thoại và máy tính. Như vậy, dù chúng ta ở bất cứ nơi nào có internet đều có thể giám sát và điều khiển được các thiết bị đã kết nối với module điều khiển. Khi dự án thành công và được áp dụng rộng rãi thì sẽ rất tiện lợi cho cuộc sống thường ngày, giúp cho đất nước ngày càng phát triển.

## 1.2 Mục tiêu của đề tài

- Tiếp nhận tín hiệu từ các cảm biến và điều khiển các thiết bị.
- Có chức năng giám sát và điều khiển từ xa qua internet, sử dụng điện thoại hoặc máy tính.
- Điều khiển bằng giọng nói.
- Có thể thi công đồ án trên một ngôi nhà thực tế hoặc mô hình.

## 1.3 Nội dung đề tài

Việc thực hiện thiết kế mạch “Giám sát nhiệt độ độ ẩm và điều khiển thiết bị điện qua internet dùng module arduino esp8266” sẽ cần phải thực hiện các nội dung như sau :

- Nội dung 1: Nghiên cứu tài liệu về KIT NodeMCU ESP8266, giao tiếp không dây và mạng Internet.
- Nội dung 2: Nghiên cứu các mô hình điều khiển.
- Nội dung 3: Thiết kế và tính toán thiết kế mạch phần cứng cho thiết bị.
- Nội dung 4: Thi công phần cứng, thử nghiệm và hiệu chỉnh phần cứng.
- Nội dung 5: Thử nghiệm và điều chỉnh hệ thống cũng như chương trình để hệ thống được tối ưu. Đánh giá các thông số của mô hình so với thực tế.
- Nội dung 6: Viết báo cáo thực hiện.
- Nội dung 7: Bảo vệ luận văn

#### **1.4 Giới hạn**

- Kích thước mô hình
- Sử dụng KIT NodeMCU ESP8266
- Tập trung vào thiết bị điều khiển trung tâm
- Sử dụng các nền tảng đã có sẵn và các thư viện mở để phát triển sản phẩm

#### **1.5 Phạm vi ứng dụng**

Đề tài là mô hình thu nhỏ, tuy nhiên có thể được ứng dụng rộng rãi ở các môi trường khác nhau như nhà ở, nhà xưởng, nhà kính....Trong sản xuất cũng như sinh hoạt.

## CHƯƠNG II: CƠ SỞ LÝ THUYẾT

### 2.1 Tổng quan về internet of things

#### 2.1.1 Giới thiệu về Internet of Things (IoT)

Ngày nay, nhu cầu phát triển các ứng dụng liên quan đến Internet ngày càng cao. Và IoT (Internet of things) là một công nghệ quan trọng bởi chúng ta có thể tạo ra nhiều ứng dụng đa dạng phục vụ đa số mọi lĩnh vực trong đời sống từ nó.

Về cơ bản, IoT là một hệ thống mạng lưới mà trong đó tất cả các thiết bị, đối tượng được kết nối Internet thông qua thiết bị mạng (network devices) hoặc các bộ định tuyến (routers). IoT cho phép các đối tượng được điều khiển từ xa dựa trên hệ thống mạng hiện tại. Công nghệ tiên tiến này giúp giảm công sức vận hành của con người bằng cách tự động hóa việc điều khiển các thiết bị.



Các thành phần chính trong một hệ thống IoT:

- **Thiết bị:**

Mỗi thiết bị sẽ bao gồm một hoặc nhiều cảm biến để phát hiện các thông số của ứng dụng và gửi chúng đến Platform.

- **IoT – Platform:**

Nền tảng này là một phần mềm được lưu trữ trực tuyến còn được gọi là điện toán đám mây, các thiết bị được kết nối với nhau thông qua nó.

Nền tảng này thu thập dữ liệu từ thiết bị, toàn bộ dữ liệu được phân tích, xử lý, phát hiện nếu có lỗi phát sinh trong quá trình hệ thống vận hành.

- **Kết nối Internet:**

Để giao tiếp được trong IoT, kết nối Internet của các thiết bị là một điều bắt buộc. Wifi là một trong những phương thức kết nối Internet phổ biến.

- **Ứng dụng:**

Ứng dụng là giao diện để người dùng điều khiển.

### 2.1.2. Lịch sử hình thành

Khái niệm về một mạng lưới thiết bị được kết nối với nhau đã được thảo luận vào đầu năm 1982, với một máy bán hàng tự động Coke được thực hiện ở Đại học Carnegie Mellon trở thành thiết bị kết nối Internet đầu tiên trên thế giới.

Thuật ngữ “Internet of things” được sử dụng lần đầu tiên bởi Kevin Ashton vào năm 1999. Sau đó IoT trải qua nhiều giai đoạn và có bước phát triển nhảy vọt cho đến ngày nay.

### 2.1.3. Ứng dụng của IoT



- **Nhà thông minh (Smart Home)**

Đây là một trong những ứng dụng được quan tâm nhiều nhất trong những năm gần đây. Một ngôi nhà thông minh hoàn toàn có thể được giám sát và điều khiển tự động. Bạn có thể bật tắt đèn bằng một ứng dụng trên điện thoại, nếu lỡ quên tắt tivi khi ra khỏi nhà bạn hoàn toàn có thể tắt nó ở một nơi có kết nối Internet, hoặc điều hòa sẽ tự động điều chỉnh tăng hay giảm khi nhiệt độ bên

ngoài thay đổi. Và còn vô số ứng dụng khác nhằm mang lại sự tiện lợi nhất cho người dùng.

Hiện nay các chủ đầu tư xây dựng chung cư cũng đã tiếp cận với công nghệ này do nhu cầu sở hữu căn hộ thông minh của người dùng ngày càng cao.



- **Giao thông vận tải**

An toàn là điều đầu tiên khi nghĩ đến tác động của IoT đối với giao thông vận tải. Ý tưởng đưa ra là các phương tiện có khả năng liên lạc với nhau bằng cách sử dụng dữ liệu đã được phân tích để có thể giảm đáng kể các sự cố tai nạn xảy ra khi tham gia giao thông. Sử dụng cảm biến, các phương tiện như ô tô, xe buýt được cảnh báo nguy cơ tiềm ẩn trên đường, hoặc thậm chí là tình trạng ùn tắc giao thông ở một số tuyến đường.

Dịch vụ vận chuyển hàng hóa cũng được ứng dụng từ công nghệ này. Công nghệ quản lý lịch trình vận chuyển, tối ưu hóa các tuyến giao hàng, mức tiêu thụ nhiên liệu của phương tiện, giám sát tốc độ của tài xế giao hàng tuân thủ quy định an toàn nhằm mang lại những lợi ích về kinh tế và sự hài lòng của khách hàng.

- **Chăm sóc sức khỏe**

Một thiết bị có thể cảnh báo tình trạng và theo dõi sức khỏe là một trong những ứng dụng trong lĩnh vực y tế.



Miếng dán theo dõi sức khỏe cho bệnh nhân: bạn không cần đến bác sĩ, những thông số về nhịp tim, huyết áp, đều được thu thập từ xa được phân tích sau đó chuẩn đoán để đưa ra tình trạng sức khỏe hiện tại của bệnh nhân và có thể dự đoán nguy cơ mắc bệnh nhằm có biện pháp phòng ngừa kịp thời.



- **Nông nghiệp (Smart Farming)**

Mô hình nhà kính là một trong những ứng dụng điển hình của công nghệ IoT được áp dụng trong lĩnh vực nông nghiệp. Và ở nước ta đã được áp dụng rộng rãi. Bên trong hệ thống này cây trồng hoàn toàn cách ly với điều kiện thời tiết bên ngoài, việc điều khiển nhiệt độ, độ ẩm, ánh sáng đều tự động hóa. Đồng thời theo dõi được tình trạng phát triển của cây trồng, xác định thời gian thu hoạch, giảm thiểu tối đa công suất người lao động.



- **Thành phố thông minh (Smart City)**

Có thể xem đây là tập hợp của tất cả ứng dụng của IoT vào một hệ thống lớn. Một giải pháp đã và đang được nhiều quốc gia trên thế giới áp dụng ở các thành phố lớn nhằm giải quyết những vấn đề cấp bách như tình trạng kẹt xe, gia tăng dân số, ô nhiễm môi trường, ngập lụt, ...

Mọi thứ trong thành phố thông minh này được kết nối, dữ liệu sẽ được giám sát bởi một loạt các máy tính mà không cần bất kỳ sự tương tác nào của con người.



## **2.2 CÔNG NGHỆ WIFI**

### **2.2.1. Giới thiệu**

Wifi là một mạng thay thế cho mạng có dây thông thường, thường được sử dụng để kết nối các thiết bị ở chế độ không dây bằng việc sử dụng công nghệ sóng vô tuyến. Dữ liệu được truyền qua sóng vô tuyến cho phép các thiết bị truyền nhận dữ liệu ở tốc độ cao trong phạm vi của mạng Wifi. Kết nối các máy tính với nhau, với Internet và với mạng có dây.

Wifi (Wireless Fidelity) là thuật ngữ dùng chung để chỉ tiêu chuẩn IEEE802.11 cho mạng cục bộ không dây (Wireless Local Networks) hoặc WLANs.

Việc sử dụng rộng rãi và tính sẵn có của nó ở nhà và nơi công cộng như công viên, quán café, sân bay, ... đã khiến Wifi trở thành một trong những công nghệ truyền nhận dữ liệu phổ biến nhất hiện nay.

### **2.2.2. Công nghệ truyền nhận dữ liệu**

#### **Các chuẩn của wifi**

Wifi là viết tắt của từ Wireless Fidelity trong tiếng Anh, được gọi chung là mạng không dây sử dụng sóng vô tuyến. Wifi là loại sóng vô tuyến tương tự như sóng điện thoại, sóng truyền hình và radio. Hầu hết các thiết bị sử dụng điện tử hiện nay như : Smartphone, Máy tính bảng, Tivi, Laptop... đều có thể kết nối được WiFi. Và Wifi là thứ gắn liền và không thể thiếu với đời sống của người dân trong hầu hết công việc cũng như giải trí hàng ngày

Chúng truyền và phát tín hiệu ở tần số 2.4 GHz hoặc 5 GHz. Tần số này cao hơn so với các tần số sử dụng cho điện thoại di động, các thiết bị cầm tay và truyền hình. Tần số cao hơn cho phép tín hiệu mang theo nhiều dữ liệu hơn

#### **Chuẩn 802.11**

IEEE 802.11 là một tập các chuẩn của tổ chức IEEE. Chuẩn IEEE 802.11 mô tả một giao tiếp “truyền qua không khí” (tiếng Anh: over-the-air) sử dụng sóng vô tuyến để truyền nhận tín hiệu giữa một thiết bị không dây và tổng đài hoặc điểm truy cập (tiếng Anh: access point), hoặc giữa 2 hay nhiều thiết bị không dây với nhau

Năm 1997, IEEE giới thiệu chuẩn mạng không dây đầu tiên và đặt tên nó là 802.11. Khi đó, tốc độ hỗ trợ tối đa của mạng này chỉ là 2 Mbps với băng tần 2.4GHz.

### **Chuẩn 802.11b**

IEEE đã mở rộng trên chuẩn 802.11 gốc vào tháng Bảy năm 1999, đó chính là chuẩn 802.11b. Chuẩn này hỗ trợ băng thông lên đến 11Mbps, tương quan với Ethernet truyền thống

802.11b sử dụng tần số vô tuyến (2.4 GHz) giống như chuẩn ban đầu 802.11. Các hãng thích sử dụng các tần số này để chi phí trong sản xuất của họ được giảm. Các thiết bị 802.11b có thể bị xuyên nhiễu từ các thiết bị điện thoại không dây (kéo dài), lò vi sóng hoặc các thiết bị khác sử dụng cùng dải tần 2.4 GHz. Mặc dù vậy, bằng cách cài đặt các thiết bị 802.11b cách xa các thiết bị như vậy có thể giảm được hiện tượng xuyên nhiễu này

- Ưu điểm của 802.11b – giá thành thấp nhất; phạm vi tín hiệu tốt và không dễ bị cản trở.
- Nhược điểm của 802.11b – tốc độ tối đa thấp nhất; các ứng dụng gia đình có thể xuyên nhiễu.

### **Chuẩn 802.11a**

Được phát triển song song cùng với chuẩn 802.11b, chuẩn 802.11a hỗ trợ tốc độ tối đa gần gấp 5 lần lên đến 54 Mbps và sử dụng băng tần 5GHz nhằm tránh bị nhiễu từ các thiết bị khác. Tuy nhiên, đây cũng là nhược điểm của chuẩn này vì phạm vi phát sẽ hẹp hơn (40-100m) và khó xuyên qua các vật cản như vách tường.

Chuẩn này thường được sử dụng trong các mạng doanh nghiệp thay vì gia đình vì giá thành của nó khá cao.

### **Chuẩn 802.11g**

Năm 2003, chuẩn Wifi thế hệ thứ 3 ra đời và mang tên 802.11g. Chuẩn này được kết hợp từ chuẩn a và b. Được hỗ trợ tốc độ 54Mbps như chuẩn a và sử dụng băng tần 2.4GHz của chuẩn b vì vậy chuẩn này có phạm vi tín hiệu

khá tốt (80- 200m) và vẫn dễ bị nhiễu từ các thiết bị điện tử khác. Ngày nay, một số hộ gia đình vẫn còn sử dụng chuẩn này .

- Ưu điểm của 802.11g – tốc độ cao; phạm vi tín hiệu tốt và ít bị che khuất.
- Nhược điểm của 802.11g – giá thành đắt hơn 802.11b; các thiết bị có thể bị xuyên nhiễu từ nhiều thiết bị khác sử dụng cùng băng tần.

### **Chuẩn 802.11n (hay 802.11 b/g/n)**

Đây là chuẩn được sử dụng phổ biến nhất hiện nay và tương đối mới. Chuẩn WiFi 802.11n được đưa ra nhằm cải thiện chuẩn 802.11g bằng cách sử dụng công nghệ MIMO (Multiple-Input Multiple-Output) tận dụng nhiều anten hơn

Chuẩn kết nối 802.11n hỗ trợ tốc độ tối đa lên đến 600 Mbps, có thể hoạt động trên cả băng tần 2,4 GHz và 5 GHz, nếu router hỗ trợ thì hai băng tần này có thể cùng phát sóng song song. Chuẩn kết nối này đã và đang dần thay thế chuẩn 802.11g với tốc độ cao, phạm vi tín hiệu rất tốt (từ 100-250m) và giá thành đang ngày càng phù hợp với túi tiền người tiêu dùng.

- Ưu điểm của 802.11n – tốc độ nhanh và phạm vi tín hiệu tốt nhất; khả năng chịu đựng tốt hơn từ việc xuyên nhiễu từ các nguồn bên ngoài.
- Nhược điểm của 802.11n – chuẩn vẫn chưa được ban bố, giá thành đắt hơn 802.11g; sử dụng nhiều tín hiệu có thể gây nhiễu với các mạng 802.11b/g ở gần.

### **Chuẩn 802.11ac (hay chuẩn 802.11 a/b/g/n/ac)**

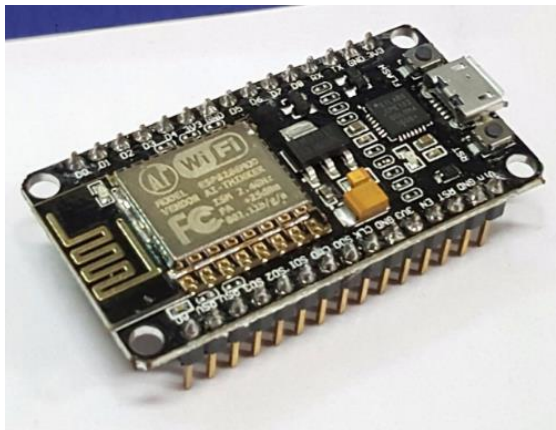
Trong khoảng một vài năm trở lại đây chúng ta được nghe nhắc nhiều đến chuẩn Wi-Fi 802.11ac, hay còn gọi là Wi-Fi thế hệ thứ năm. Nó là chuẩn mạng không dây đang ngày càng xuất hiện nhiều hơn trên các router, máy tính và tất nhiên là cả các thiết bị di động như smartphone. So với Wi-Fi 802.11n đang được dùng phổ biến hiện nay, chuẩn 802.11ac mang lại tốc độ nhanh hơn.

Là chuẩn Wifi mới nhất được IEEE giới thiệu. Chuẩn ac có hoạt động ở băng tần 5 GHz và tốc độ tối đa lên đến 1730 Mbps khi sử dụng lại công nghệ đa anten trên chuẩn 802.11n cho người dùng trải nghiệm tốc độ cao nhất.

Hiện tại, chuẩn này được sử dụng trên một số thiết bị cao cấp của các hãng điện thoại như Apple, Samsung, Sony,... Tuy nhiên, do giá thành khá cao nên các thiết bị phát tín hiệu cho chuẩn này chưa được phổ biến trên thị trường nên mặc dù các thiết bị này không hoạt động tối ưu khi sử dụng bởi sự hạn chế của các thiết bị phát.

### 2.3. Giới thiệu về ESP8266 NodeMCU

ESP8266 là dòng chip tích hợp Wi-Fi 2.4Ghz có thể lập trình được, rẻ tiền được sản xuất bởi một công ty bán dẫn Trung Quốc: Espressif Systems. Được phát hành đầu tiên vào tháng 8 năm 2014, đóng gói đưa ra thị trường dạng Module ESP-01. Có khả năng kết nối Internet qua mạng Wi-Fi một cách nhanh chóng và sử dụng rất ít linh kiện đi kèm. Với giá cả có thể nói là rất rẻ so với tính năng và khả năng ESP8266 có thể làm được. ESP8266 có một cộng đồng các nhà phát triển trên thế giới rất lớn, cung cấp nhiều Module lập trình mã mở giúp nhiều người có thể tiếp cận và xây dựng ứng dụng rất nhanh. Hiện nay tất cả các dòng chip ESP8266 trên thị trường đều mang nhãn ESP8266EX, là phiên bản nâng cấp của ESP8266, đã có hơn 14 phiên bản ESP ra đời, trong đó phổ biến nhất là ESP-12.



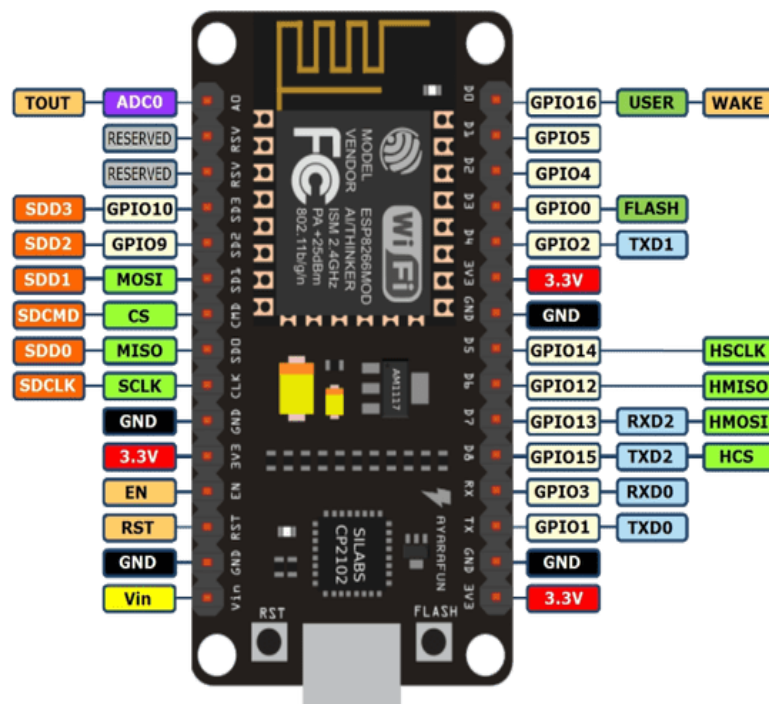
*Hình ảnh thực tế của Chip NODEMCU ESP8266*

#### 2.3.1 Cấu tạo của NODEMCU ESP8266

Module ESP8266 có các chân dùng để cấp nguồn và thực hiện kết nối. Chức năng của các chân như sau:

- + VCC: 3.3V lên đến 300Ma
- + GND: Chân Nối đất .

- + Tx: Chân Tx của giao thức UART, kết nối đến chân Rx của vi điều khiển.
- + Rx: Chân Rx của giao thức UART, kết nối đến chân Tx của vi điều khiển.
- + RST: chân reset, kéo xuống mass để reset.
- + 10 chân GPIO từ D0 – D8, có chức năng PWM, IIC, giao tiếp SPI, 1-Wire và ADC trên chân A0
- + Kết nối mạng wifi (có thể là sử dụng như điểm truy cập và/hoặc trạm máy chủ lưu trữ một, máy chủ web), kết nối internet để lấy hoặc tải lên dữ liệu.



*Hình 1.7. Hình ảnh sơ đồ chân kết nối ESP8266*

Module ESP-12 kết hợp với firmware ESP8266 trên Arduino và thiết kế phần cứng giao tiếp tiêu chuẩn đã tạo nên NodeMCU, loại Kit phát triển ESP8266 phổ biến nhất trong thời điểm hiện tại. Với cách sử dụng, kết nối dễ dàng, có thể lập trình, nạp chương trình trực tiếp trên phần mềm Arduino, đồng thời tương thích với các bộ thư viện Arduino sẵn có.

### 2.3.2. Tính năng của NODEMCU ESP8266

➤ Thông số kỹ thuật:

- IC chính: ESP8266 Wifi SoC.
- Phiên bản firmware: NodeMCU Lua
- Chip nạp và giao tiếp UART: CP2102.
- GPIO tương thích hoàn toàn với firmware Node MCU.
- Cấp nguồn: 5VDC MicroUSB hoặc Vin.
- GIPO giao tiếp mức 3.3VDC
- Tích hợp Led báo trạng thái, nút Reset, Flash.
- Tương thích hoàn toàn với trình biên dịch Arduino.
- Kích thước: 25 x 50 mm

### 2.4. Module DHT11

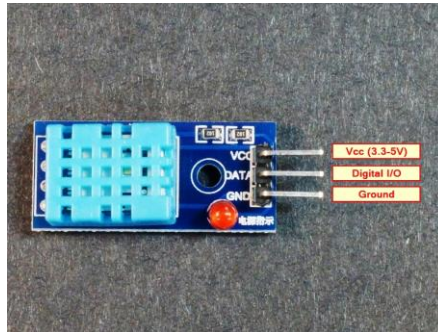
Trên thị trường hiện nay có rất nhiều loại cảm biến nhiệt độ với các đặc tính khác nhau để phù hợp với nhiều mục đích sử dụng như LM335, LM35, DHT11, Pt100,... Với đề tài này, em lựa chọn sử dụng DHT11 vì nó tích hợp đo cả nhiệt độ và độ ẩm.

Cảm biến độ ẩm và nhiệt độ DHT11 là cảm biến rất thông dụng hiện nay vì chi phí rẻ và rất dễ lấy dữ liệu thông qua giao tiếp 1 wire (giao tiếp digital 1 dây truyền dữ liệu duy nhất). Cảm biến nhiệt độ và độ ẩm DHT11 có bộ điều chỉnh nhiệt độ và độ ẩm với đầu ra tín hiệu số được hiệu chuẩn qua bộ tiền xử lý tín hiệu tích hợp trong cảm biến giúp bạn có được dữ liệu chính xác mà không phải qua bất kỳ tính toán nào. Với việc sử dụng tín hiệu kỹ thuật cao nên cảm biến luôn cho độ tin cậy cao và ổn định trong thời gian dài. Cảm biến này bao gồm một thành phần đo độ ẩm kiểu điện trở và bộ phận giảm nhiệt độ NTC, và kết nối với bộ vi điều khiển 8 bit hiệu suất cao, cung cấp chất lượng tốt, phản ứng nhanh, chống nhiễu và hiệu quả về chi phí.

Mỗi cảm biến DHT11 đều được hiệu chuẩn trong phòng thí nghiệm để có độ chính xác cao nhất. Sự kết nối hệ thống nối tiếp một dây nhanh chóng và dễ



dạng. Kích thước nhỏ, tiêu thụ điện năng thấp và truyền tín hiệu lên đến 20m, đây lựa chọn tốt nhất cho các ứng dụng khác.



Thông số kỹ thuật:

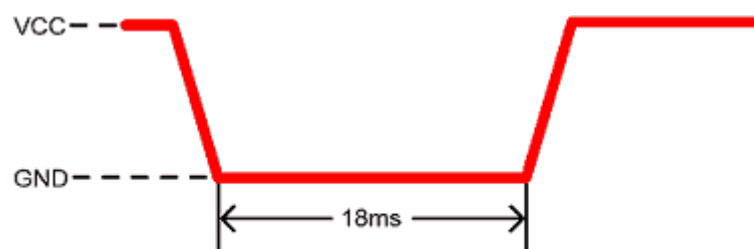
- Điện áp hoạt động: 3.3-5V
- Dải độ ẩm hoạt động: 20% - 90% RH, sai số  $\pm 5\%RH$
- Dải nhiệt độ hoạt động: 0°C - 50°C, sai số  $\pm 2^\circ C$
- Khoảng cách truyền tối đa: 20m
- Chuẩn giao tiếp: TTL, 1-wire
- Kích thước: 28x12x10mm
- Dòng tối đa: 2.5mA
- Tần số lấy mẫu tối đa: 1Hz

#### Nguyên lý hoạt động:

DHT11 chỉ sử dụng 1 dây để giao tiếp. Quá trình giao tiếp được chia làm 3 bước: đầu tiên là gửi yêu cầu đến cảm biến, kể đến cảm biến sẽ gửi xung phản hồi và

sau đó nó bắt đầu gửi dữ liệu tổng cộng 40bit đến vi điều khiển.

#### • Bắt đầu xung

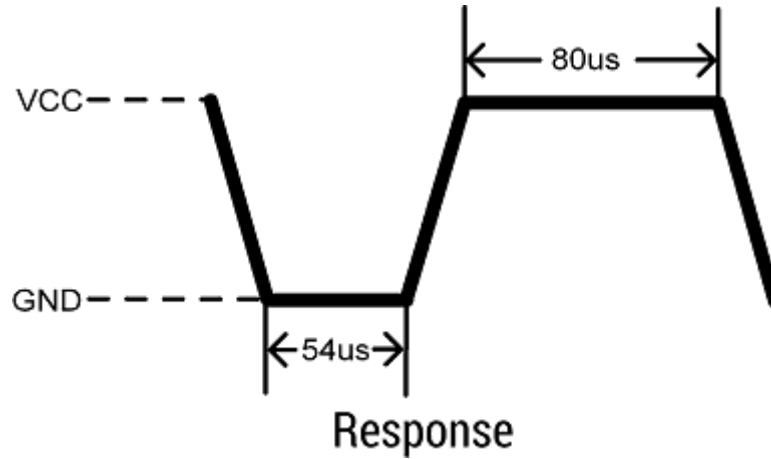


Start Pulse

Xung bắt đầu DHT11

Để bắt đầu giao tiếp với DHT11, đầu tiên ta gửi xung bắt đầu đến cảm biến. Để cung cấp xung bắt đầu, kéo chân dữ liệu xuống mức thấp trong thời gian tối thiểu 18ms và sau đó kéo lên mức cao.

- **Phản ứng**

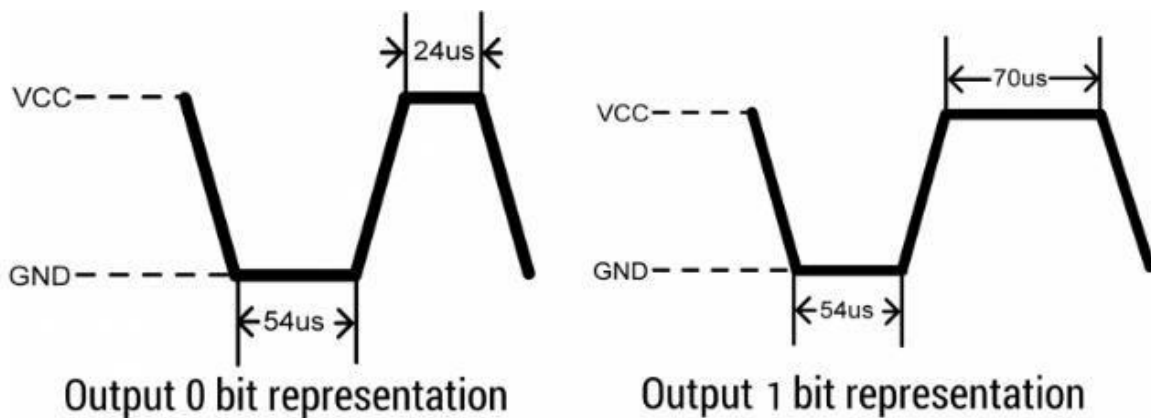


Gửi xung phản hồi của DHT11

Sau khi nhận được xung bắt đầu, cảm biến sẽ gửi xung phản hồi, để cho biết DHT11 đã nhận được xung bắt đầu.

Xung phản hồi ở mức thấp trong khoảng thời gian 54us, sau đó ở mức cao 80us.

- **Dữ liệu**



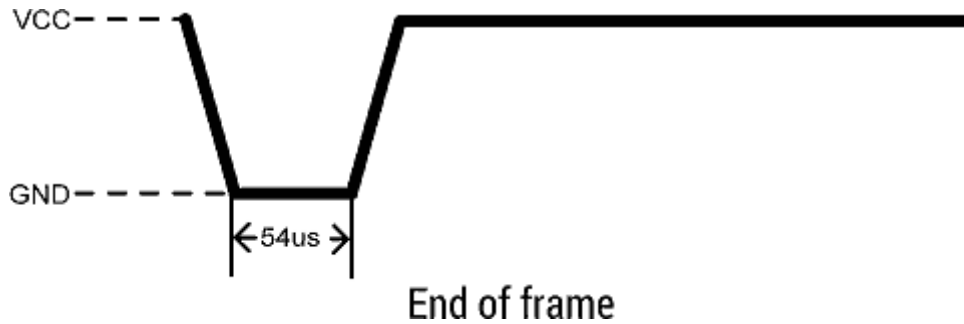
Gửi dữ liệu chứa bit 0, bit 1

Sau khi gửi xung phản hồi, DHT11 sẽ gửi dữ liệu chứa giá trị nhiệt độ và độ ẩm. Khung dữ liệu dài 40bit, được chia làm 5 phần (byte), mỗi phần 8bit.

Trong 5 phần này, hai phần đầu tiên sẽ chứa giá trị độ ẩm, 8bit đầu tiên là giá trị phần nguyên, 8bit còn lại chứa giá trị thập phân. Hai phần tiếp theo sẽ chứa giá trị nhiệt độ (°C) ở dạng số thập phân. Phần cuối cùng là 8bit để kiểm tra cho phần đo nhiệt độ và độ ẩm.

Sau khi nhận được dữ liệu, chân DHT11 sẽ ở chế độ tiêu thụ điện năng thấp cho đến khi có xung bắt đầu tiếp theo.

- **Kết thúc**



Kết thúc đọc giá trị của DHT11

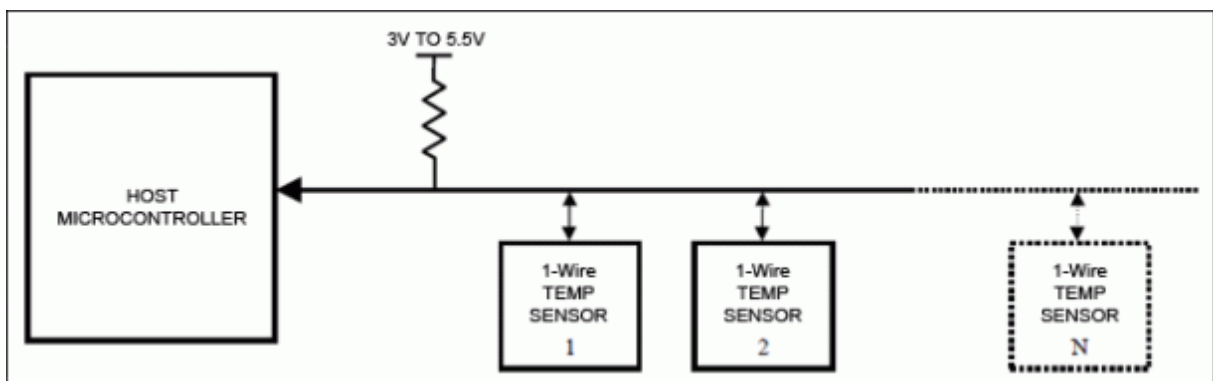
Sau khi gửi dữ liệu 40bit, DHT11 sẽ ở mức thấp 54us rồi lên mức cao và sau đó nó chuyển sang chế độ ngủ.

## 2.5 CÁC CHUẨN GIAO TIẾP ĐƯỢC SỬ DỤNG

### 2.5.1 Chuẩn One-Wire

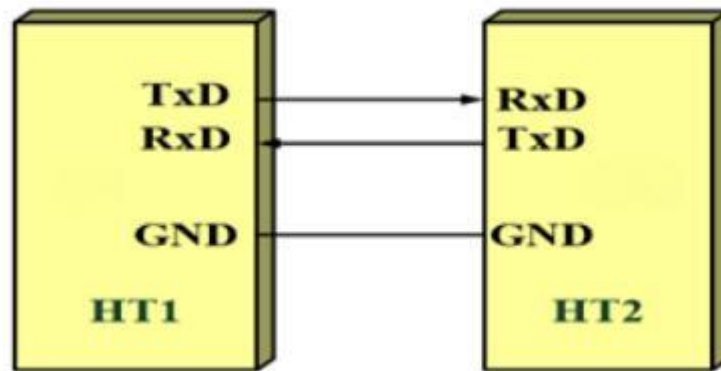
Chuẩn giao tiếp 1 dây (one-wire) được thiết kế bởi Dallas Semiconductor và đã được Maxim mua lại năm 2001. Maxim là một hãng sản xuất chip lớn. One-Wire dùng một dây để truyền nhận nên có tốc độ thấp. Chủ yếu sử dụng cho việc thu thập dữ liệu, truyền nhận dữ liệu thời tiết, nhiệt độ, công việc không yêu cầu tốc độ cao.

Là chuẩn giao tiếp không đồng bộ và bán song công (half-duplex). Giao tiếp tuân theo mối quan hệ chủ tớ một cách chắc chắn. Trên cùng một bus thì chúng ta có thể gắn 1 hoặc nhiều thiết bị slave nhưng chỉ có một master có thể kết nối được với bus này. Khi không có dữ liệu trên đường truyền thì bus dữ liệu được xem là ở trạng thái rảnh.



## 2.5.2 Chuẩn giao tiếp UART

UART là viết tắt của Universal Asynchronous Receiver – Transmitter có nghĩa là truyền dữ liệu nối tiếp bất đồng bộ. Truyền dữ liệu nối tiếp bất đồng bộ có 1 đường phát dữ liệu và 1 đường nhận dữ liệu, không có tín hiệu xung clock nên gọi là bất đồng bộ. Để truyền được dữ liệu thì cả bên phát và bên nhận phải tự tạo xung clock có cùng tần số và thường được gọi là tốc độ baud, ví dụ như 2400 baud, 4800 baud, 9600 baud...



Hệ thống truyền dữ liệu bất đồng bộ.

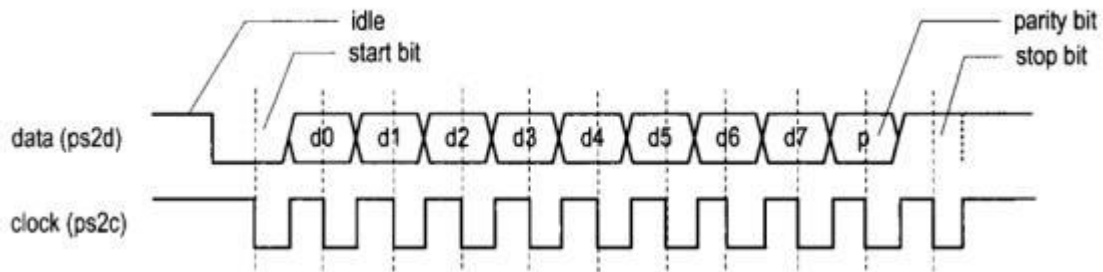
Giao tiếp UART chế độ bất đồng bộ sử dụng một dây kết nối cho mỗi chiều truyền dữ liệu do đó để quá trình truyền nhận dữ liệu thành công thì việc tuân thủ các tiêu chuẩn truyền là hết sức quan trọng. Sau đây là các khái niệm quan trọng trong chế độ truyền thông này

**Baud rate (tốc độ Baud):** Để việc truyền và nhận bất đồng bộ xảy ra thành công thì các thiết bị tham gia phải thống nhất với nhau về khoảng thời gian dành cho 1 bit truyền, hay nói cách khác tốc độ truyền phải được cài đặt như nhau trước khi truyền nhận, tốc độ này gọi là tốc độ Baud. Tốc độ Baud là số bit truyền trong một giây. Ví dụ, nếu tốc độ Baud được đặt là 9600 bit/giây thì thời gian dành cho một bit truyền là  $1/9600 \sim 104.167\mu s$ .

**Frame (khung truyền):** Do truyền thông nối tiếp mà nhất là nối tiếp bất đồng bộ rất dễ mất hoặc sai lệch dữ liệu, quá trình truyền thông theo kiểu này phải tuân theo một số quy cách nhất định. Bên cạnh tốc độ Baud, khung truyền là một yếu tố quan trọng tạo nên sự thành công khi truyền và nhận. Khung truyền bao gồm các quy định về số bit trong mỗi lần truyền, các bit báo hiệu như bit Start và bit

Stop, các bit kiểm tra như Parity, ngoài ra số lượng các bit dữ liệu trong mỗi lần truyền cũng được quy định bởi khung truyền

Để bắt đầu cho việc truyền dữ liệu bằng UART, một START bit được gửi đi, sau đó là các bit dữ liệu và kết thúc quá trình truyền là STOP bit



Start bit: Là bit đầu tiên được truyền trong một khung truyền, bit này có chức năng báo cho thiết bị nhận biết rằng có một gói dữ liệu sắp được truyền tới. Start bit là bit bắt buộc phải có trong khung truyền.

Data: Data hay dữ liệu cần truyền là thông tin chính mà chúng ta cần gửi và nhận. Dữ liệu cần truyền không nhất thiết phải là gói 8 bit, có thể quy định số lượng bit của dữ liệu là 5, 6, 7, 8 hoặc 9. Trong truyền thông nối tiếp USART, bit có ảnh hưởng nhỏ nhất của dữ liệu sẽ được truyền trước và cuối cùng là bit có ảnh hưởng lớn nhất.

Parity bit: Là bit dùng kiểm tra dữ liệu truyền đúng không (một cách tương đối). Có 2 loại parity là parity chẵn và parity lẻ. Parity chẵn nghĩa là số lượng bit 1 trong dữ liệu bao gồm bit parity luôn là số chẵn. Ngược lại, tổng số lượng các bit 1 trong parity lẻ luôn là lẻ. Ví dụ, nếu dữ liệu của bạn là 10111011 nhị phân, có tất cả 6 bit 1 trong dữ liệu này, nếu parity chẵn được dùng, bit parity sẽ mang giá trị 0 để đảm bảo tổng các bit 1 là số chẵn (6 bit 1). Nếu parity lẻ được yêu cầu thì giá trị của parity bit là 1. Parity bit không phải là bit bắt buộc và vì thế chúng ta có thể loại bit này khỏi khung truyền.

Stop bits: Là một hoặc các bit báo cho thiết bị nhận rằng một gói dữ liệu đã được gửi xong. Sau khi nhận được stop bits, thiết bị nhận sẽ tiến hành kiểm tra khung truyền để đảm bảo tính chính xác của dữ liệu. Stop bits là các bit bắt buộc xuất hiện trong khung truyền.

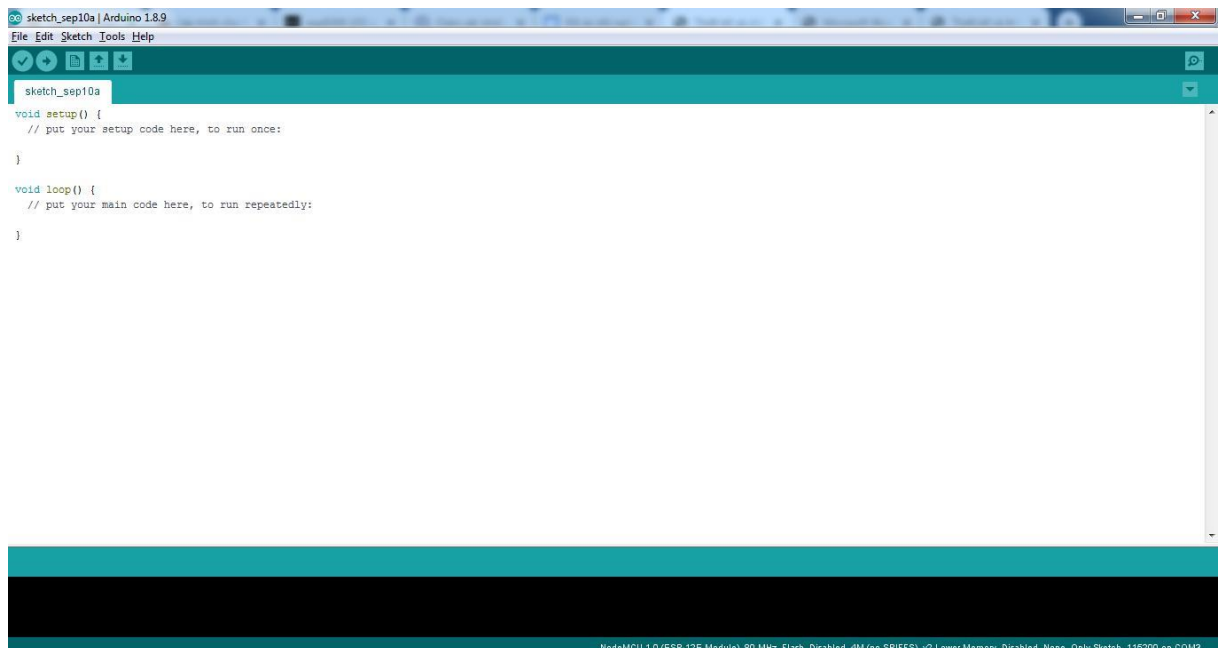
Khung truyền phổ biến nhất là (Start bit + 8 bit dữ liệu + stop bit).

## 2.6 PHẦN MỀM

### 2.6.1 Giới thiệu phần mềm lập trình

Arduino IDE [15] là môi trường phát triển tích hợp mã nguồn mở, cho phép người dùng dễ dàng viết code và tải nó lên bo mạch. Môi trường phát triển được viết bằng Java dựa trên ngôn ngữ lập trình xử lý và phần mềm mã nguồn mở khác. Phần mềm này có thể được sử dụng với bất kỳ bo mạch Arduino nào.

Arduino IDE là một môi trường phát triển tích hợp đa nền tảng, làm việc cùng với một bộ điều khiển Arduino để viết, biên dịch và tải code lên bo mạch. Phần mềm này cung cấp sự hỗ trợ cho một loạt các bo mạch Arduino như Arduino Uno, Nano, Mega, Pro hay Pro Mini, .... Ngôn ngữ tổng quát cho Arduino C và C++, do đó phần mềm phù hợp cho những lập trình viên đã quen thuộc với cả 2 ngôn ngữ này. Các tính năng như làm nổi bật cú pháp, thực đầu dòng tự động, ... làm cho nó trở thành một sự thay thế hiện đại cho các IDE khác. Arduino IDE có thư viện code mẫu quá phong phú, viết chương trình trên Arduino IDE khá dễ dàng cộng thêm OpenSource viết riêng cho Arduino thì ngày càng nhiều.



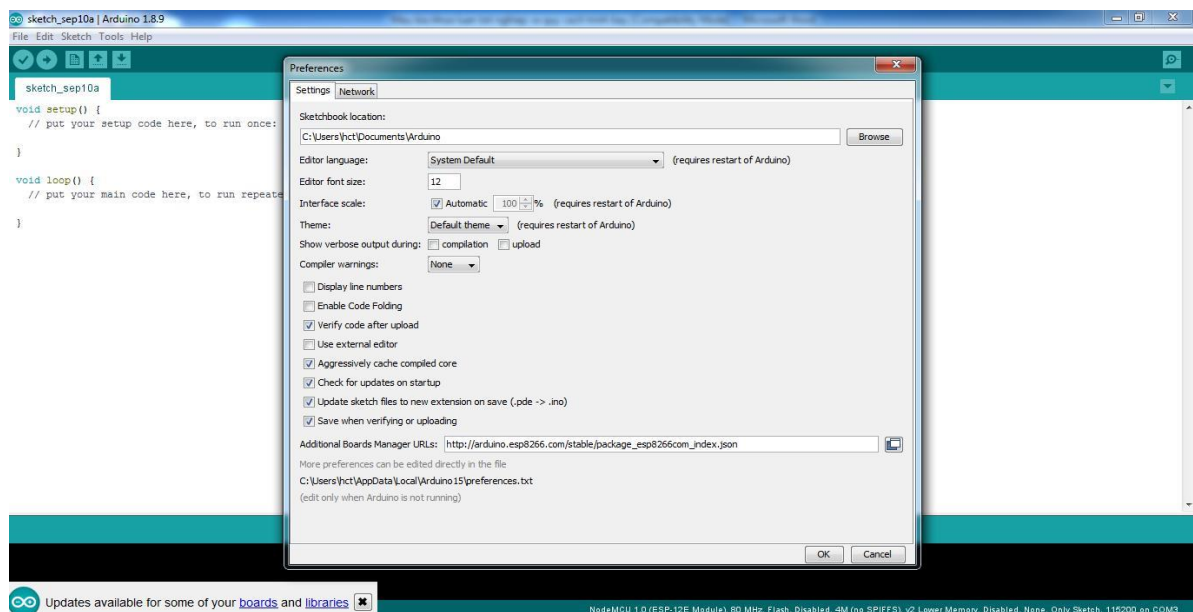
Giao diện của phần mềm Arduino.

Đây là công cụ hỗ trợ viết code và nạp code cho các bo mạch Arduino cũng như các mạch NodeMCU. Truy cập vào trang chủ Arduino <http://arduino.cc> để download phần mềm và cài đặt. Phần mềm được hỗ trợ miễn phí cho người dùng, với bản cập nhật mới nhất là Arduino 1.8.5.

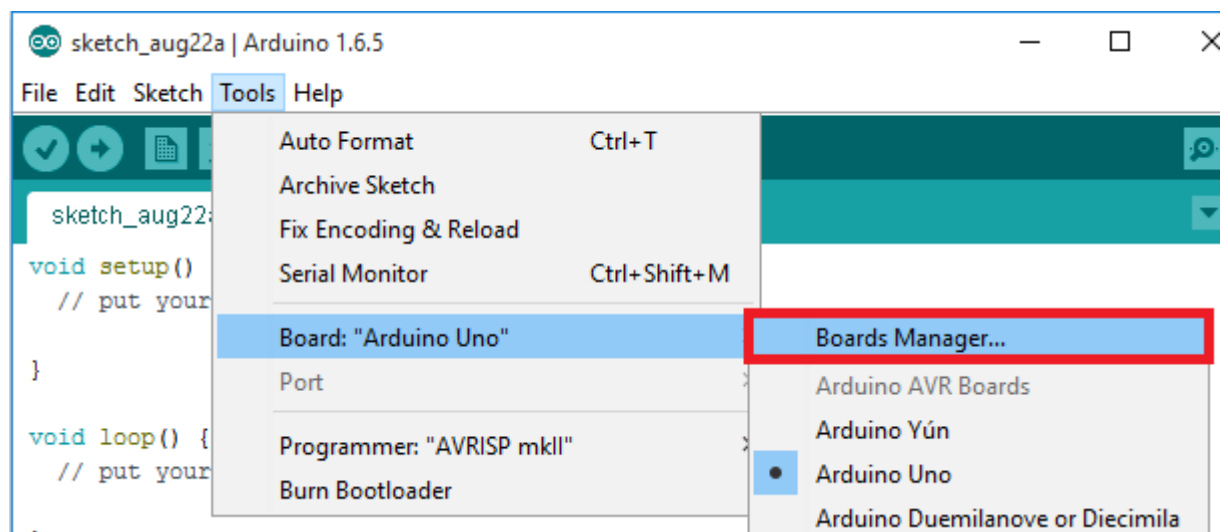
Sau khi tải về thì cần thêm thư viện để có thể sử dụng với module NodeMCU ESP8266. Vào File → Preferences, vào textbox Additional Board Manage URL thêm đường link sau vào:

[http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json)

Sau đó click OK để chấp nhận.

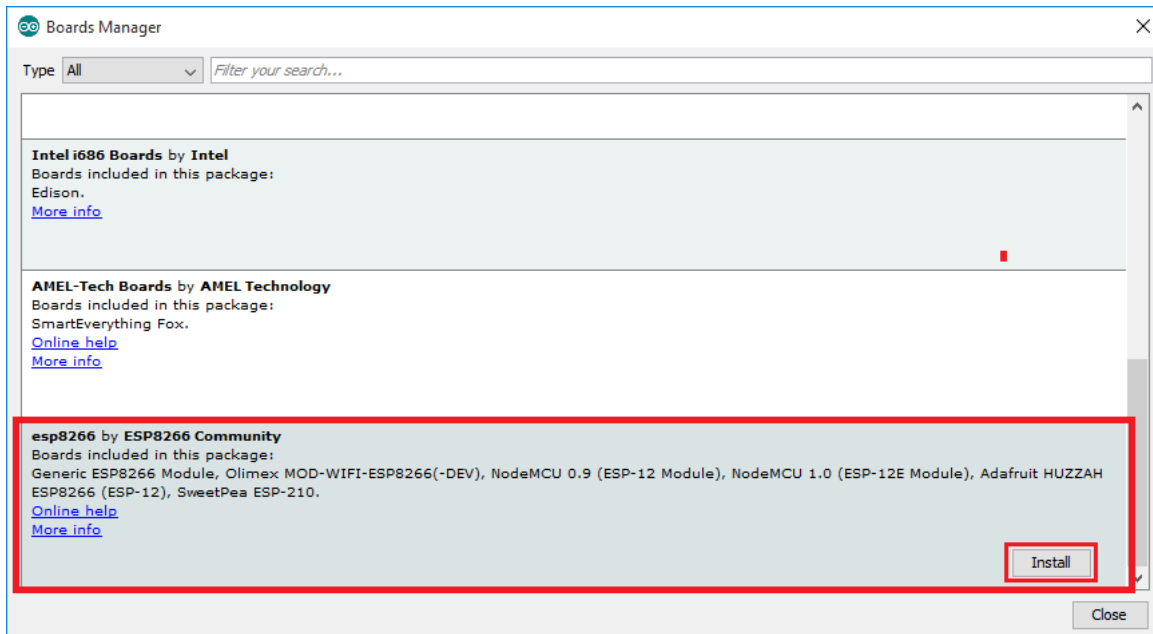


Tiếp theo vào Tool→Board→Boards Manager





Đợi một lát để chương trình tìm kiếm. Ta kéo xuống và click vào ESP8266 by ESP8266 Community, click vào Install. Chờ phần mềm tự động download và cài đặt.



Kết nối module USB-to-UART vào máy tính.

Vào Tool→Board→Generic ESP8266 Module, chọn cổng COM tương ứng với module USB-to-UART tương ứng. Tương tự ta nhập thư viện cho cảm biến đo nhiệt độ DHT11.

### 2.6.2 Cơ sở lý thuyết về APP Blynk

Blynk là một nền tảng có ứng dụng iOS, Android cho phép điều khiển Arduino, Raspberry Pi, ESP8266. Ta có thể xây dựng ứng dụng điều khiển bằng cách kéo, thả các Widget.

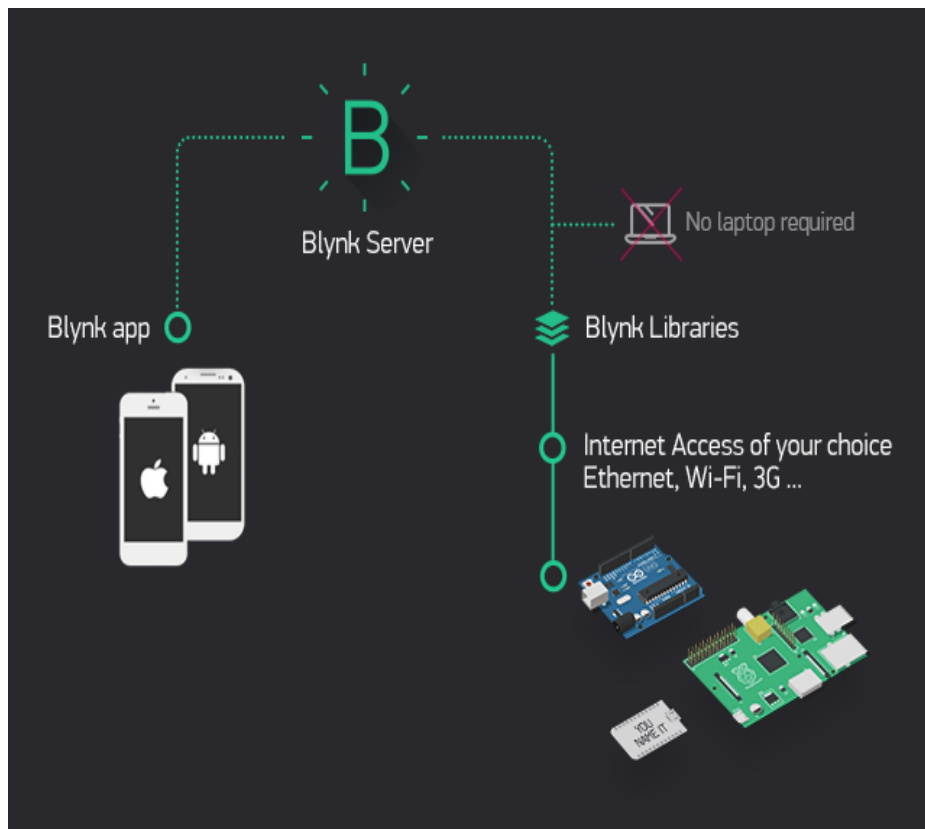
#### Cách hoạt động

Blynk được thiết kế cho IoT, nó có thể điều khiển phần cứng từ xa, hiển thị dữ liệu cảm biến, lưu trữ dữ liệu ...Blynk gồm 3 phần:

**Blynk App:** cho phép tạo các giao diện từ Widget có sẵn

**Blynk Server:** truyền tải thông tin giữa Smarthome và thiết bị. Blynk Server có thể là 1 đám mây của Blynk hoặc có thể cài đặt trên máy cá nhân. Có thể cài đặt trên Raspberry Pi.

**Blynk Libraries:** thư viện cung cấp kết nối phần cứng đến server, xử lý các lệnh đến và đi.



## Để bắt đầu Blynk cần cái gì?

### 1. Thiết bị

Thiết bị dùng các bộ kit phát triển như Arduino, Raspberry, ESP8266.

### 2. Smartphone

Ứng dụng Blynk có thể hoạt động trên iOS và Android. Có thể download từ trên điện thoại.

Thư viện Blynk cho Arduino: <https://github.com/blynkkk/blynk-library/releases/latest>

Trong Arduino IDE có thể thêm thư viện này từ Manager Libraries bằng cách tìm kiếm từ khóa Blynk và cài đặt.

### 2.6.3 Google Assistant

#### a. Khái niệm

Google Assistant là một trợ lý ảo thông minh tương tự như Siri của Apple, Bixby của Samsung hay Cortana của Microsoft trên Windows. Google Assistant hỗ trợ nhiều tính năng thông minh như tìm kiếm thông tin hay thực hiện yêu cầu mà người dùng đưa ra như mở danh bạ, gọi điện cho người có tên trong danh bạ, đọc tin nhắn, mở nhạc...



# Google Assistant

Người dùng chủ yếu có thể tương tác với Google Assistant qua giọng nói tự nhiên, hoặc có thể nhập qua bàn phím. Các chức năng cơ bản của nó cũng tương tự như Google Now, như tìm kiếm trên Internet, đặt sự kiện trên lịch và báo thức, điều chỉnh cài đặt phần cứng trên thiết bị người dùng và hiển thị thông tin từ tài khoản Google của người dùng.

## **b. Lịch sử phát triển**

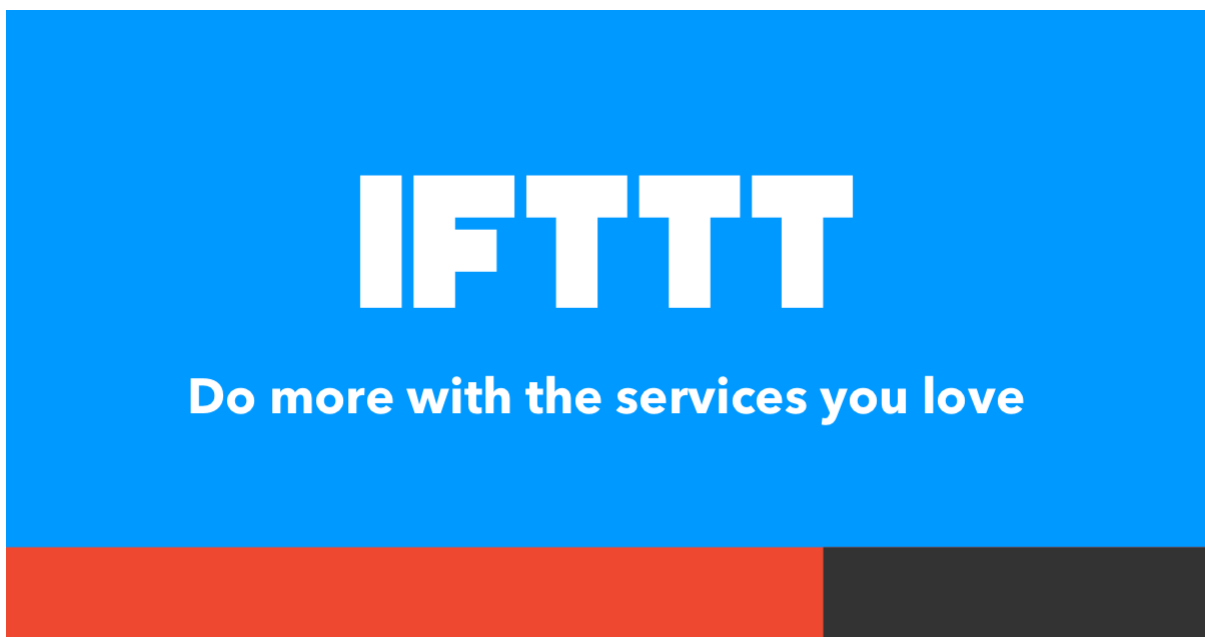
Google Assistant là một trợ lý cá nhân ảo được phát triển bởi Google và được giới thiệu tại hội nghị nhà phát triển của hãng vào tháng 5 năm 2016. Google Assistant ban đầu được đưa vào ứng dụng nhắn tin Google Allo, và loa thông minh Google Home. Sau một thời gian chỉ có mặt trên hai chiếc điện thoại thông minh Pixel và Pixel XL của hãng, Google bắt đầu triển khai Assistant trên các thiết bị Android khác vào tháng 2 năm 2017, bao gồm cả các điện thoại thông minh bên thứ ba và các thiết bị Android Wear, và được phát hành dưới dạng ứng dụng riêng biệt trên iOS vào tháng 5. Cùng với sự ra mắt một bộ phát triển phần mềm vào tháng 4 năm 2017, Assistant đã và đang được tiếp tục mở rộng hỗ trợ cho một lượng lớn thiết bị, bao gồm cả xe hơi và các thiết bị nhà thông minh. Các chức năng của Assistant cũng có thể được bổ sung bởi các nhà phát triển bên thứ ba.

## **c. Ưu điểm và nhược điểm**

- Ưu điểm

- Khả năng nhận dạng Tiếng Việt trên Google Assistant rất tốt.
- Người dùng thao tác mà không cần chạm vào điện thoại như yêu cầu gọi điện hoặc nhắn tin cho một ai đó, hẹn giờ, đặt nhắc nhở, lên lịch hẹn...
- Trợ lý của Google có thể làm tính, chuyển đổi đại lượng, tính tỷ giá... trả lời các câu hỏi liên quan đến thể thao, du lịch, tìm nhà hàng, trạm xăng...
- Google Assistant cho phép người dùng điều khiển các thiết bị thông minh trong nhà bằng Tiếng Việt.
- Nhược điểm
  - Nhiều câu hỏi trợ lý ảo không đưa ra câu trả lời trực tiếp mà chỉ gợi ý các kết quả từ Google Search.
  - Mặc dù đã hỗ trợ Tiếng Việt, nhưng với các câu hỏi cho nhà thông minh, Assistant lại trả lời bằng Tiếng Anh.

#### 2.6.4 Công cụ IFTTT ( If This Then That)



Đây là một dịch vụ web trung gian. Nó đứng giữa hai dịch vụ để thực hiện một tác vụ khi một điều kiện nào đó xảy ra, bởi vậy mới có cái tên If this (nếu điều này xảy ra) then that (thì làm việc kia). Toàn bộ hoạt động của IFTTT dựa hết vào nguyên lý này, và cứ mỗi một lệnh IFTTT thì được gọi là một "công thức", hay recipe.

## CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

### a. Mô tả hoạt động của hệ thống

Trong đề tài này, em thiết kế các relay có thể điều chỉnh được bằng giọng nói thông qua Google assistant hoặc thao tác trên app Android và nút ấn cơ.

Ngoài ra còn thiết kế mạch giám sát nhiệt độ, độ ẩm để người dùng có thể biết được nhiệt độ, độ ẩm tại vị trí đặt hiện tại thông qua app trên điện thoại.

Hệ thống hoạt động dựa trên sự kết hợp của Module Nodemcu Esp8266 và app android trên smartphone. App android ngoài chức năng lưu trữ dữ liệu từ mạch đo gửi lên còn có chức năng hiển thị giao diện điều khiển thiết bị, dữ liệu về nhiệt độ, độ ẩm ra giao diện người dùng.

Khi nhận được tín hiệu nhiệt độ báo về từ cảm biến thì bộ vi xử lý của hệ thống module Nodemcu Wifi ESP8266 sẽ xử lý tín hiệu và sau đó truyền lên sever blynk thông qua môi trường ko dây wifi.

Khi module wifi ESP8266 gửi tín hiệu về app android ta có thể truy cập vào hệ thống để điều khiển thiết bị trong gia đình và giám sát nhiệt độ, độ ẩm từ xa có mạng internet.

### Bộ xử lý trung tâm

- Điện áp 5VDC
- Giao tiếp ứng dụng Android
- Giao tiếp với Server bằng Wifi
- Ngõ ra nối các thiết bị điện 220VAC
- Thiết kế nhỏ gọn, đảm bảo tính an toàn

### Ứng dụng Android

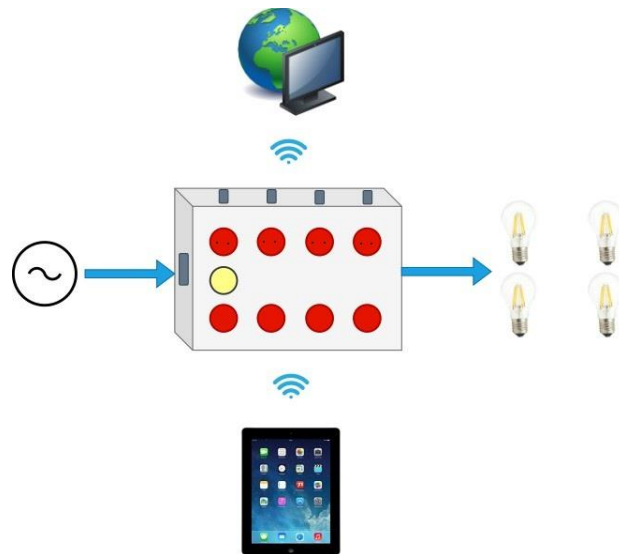
- Giám sát nhiệt độ liên tục tại các thời điểm trong ngày.
- Giám sát độ ẩm trong phòng đặt thiết bị
- Điều khiển được nhiều thiết bị cùng một lúc.
- Giao diện trực quan, thân thiện người dùng.

### Nút điều khiển

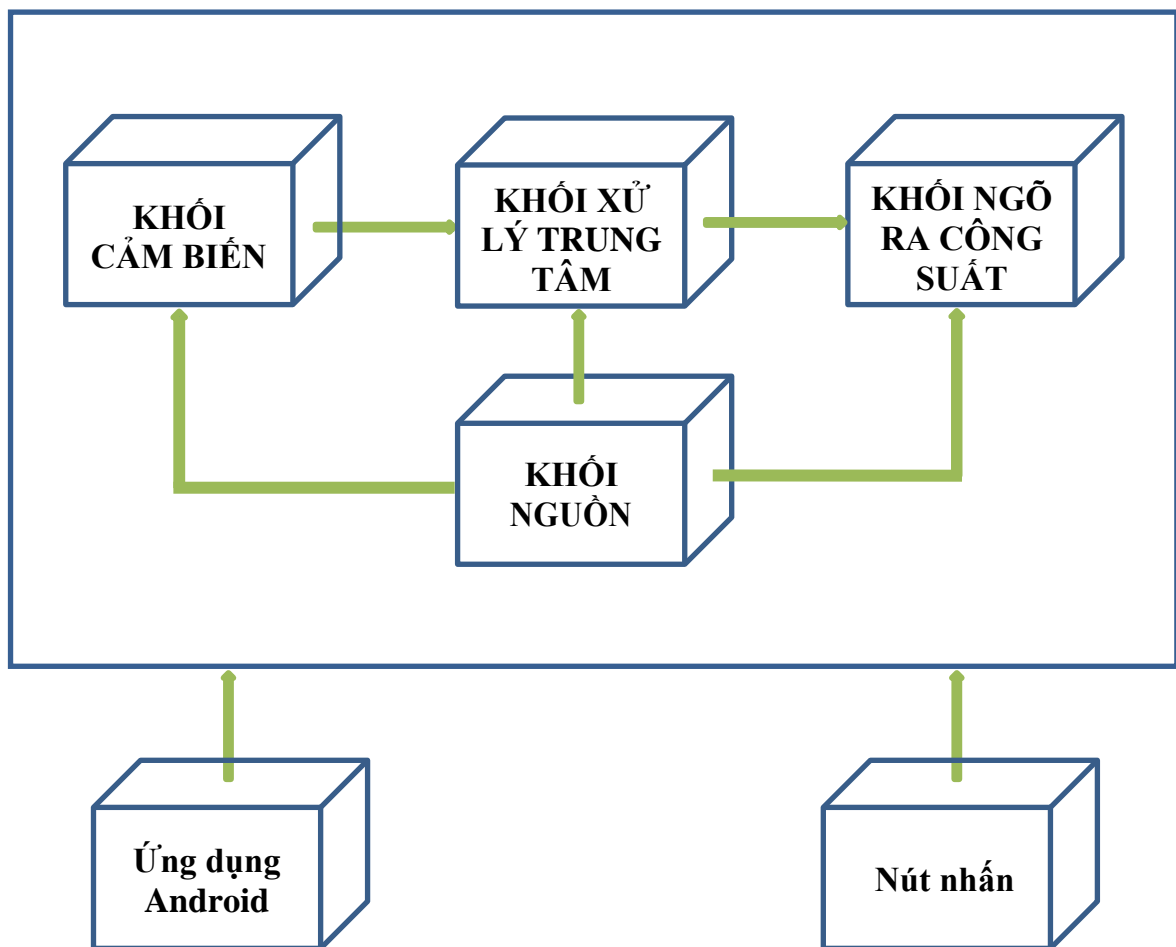
- Nút có điều khiển trực tiếp các thiết bị
- Phản hồi nhanh, không trễ

## 3.2 TÍNH TOÁN VÀ THIẾT KẾ

### 3.2.1. Thiết kế sơ đồ khối hệ thống



Sơ đồ mô hình hệ thống



Sơ đồ khối hệ thống

## ❖ Chức năng từng khối

### • Khối xử lý trung tâm

Trung tâm điều khiển hoạt động của toàn bộ hệ thống. Nhận tín hiệu từ ứng dụng Android hoặc nút nhấn, xử lý sau đó chuyển tín hiệu điều khiển đến khối công suất thực thi, tiếp theo dữ liệu được gửi lên khối Server. Khối cảm biến nhiệt độ sử dụng cảm biến DHT11 có khả năng đo cả 2 thông số nhiệt độ và độ ẩm.

### • Khối nguồn:

Cấp nguồn cho toàn mạch, sử dụng nguồn 5VDC cấp cho khối xử lý trung tâm, mạch Relay, cảm biến và nguồn 220VAC cho các thiết bị điện.

### • Khối ngõ ra công suất:

Đóng ngắt các tiếp điểm Relay theo sự điều khiển của ngõ ra vi điều khiển, từ đó điều khiển các thiết bị điện (220VAC). Đồng thời cách ly giữa mạch công suất và mạch điều khiển.

### • Khối cảm biến:

Có chức năng giám sát nhiệt độ và độ ẩm của môi trường để đảm bảo tính an toàn cho hệ thống.

### • Ứng dụng Android:

Xử lý và gửi tín hiệu điều khiển đến vi điều khiển, điều khiển trực tiếp trên thiết bị Android.

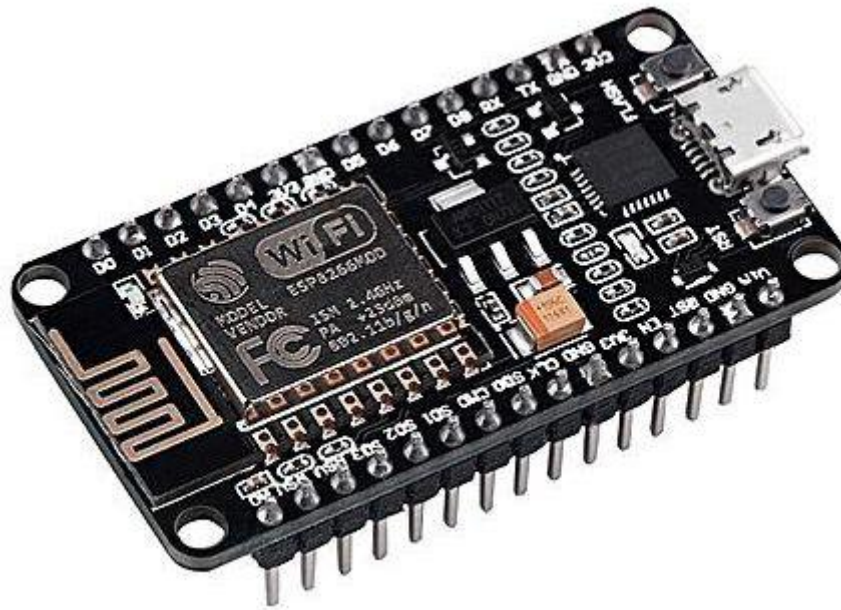
### • Nút nhấn:

Gửi tín hiệu đến vi điều khiển để điều khiển ngõ ra của vi điều khiển. Từ đó điều khiển trạng thái tắt bật của relay

## 3.2.2. Tính toán và thiết kế

### 3.2.2.1. Khối xử lý trung tâm

Khối điều khiển sử dụng board ESP8266 đáp ứng được yêu cầu đặt ra và có khả năng mở rộng cho nhiều ứng dụng.



Ảnh thực tế board ESP8266

Board được cấp nguồn 5VDC. Chi tiết kết nối giữa board mạch và các khối khác được thể hiện trong hình bên dưới:

		U1			
3V3	1	3V3	Vin	30	VCC
GND	2	GND	GND	29	GND
	3			28	
BUT4	4	TX	RST	27	
RL4	5	RX	EN	26	
RL3	6	D8	3V3	25	GND
RL2	7	D7	GND	24	
RL1	8	D6	CLK	23	
	9	D5	SDO	22	
	10	GND	CMD	21	
BUT3	11	3V3	SD1	20	
	12	D4	SD2	19	SEN1
BUT2	13	D3	SD3	18	
BUT1	14	D2	RSV	17	
	15	D1	RSV	16	
		D0	A0		

NodeMCU 1.0

Sơ đồ nguyên lý board điều khiển ESP8266



- Chân nguồn Vin số 30 và GND số 29 được nối với mạch nguồn cung cấp 5VDC.
- Các chân GPIO số 5, 6, 7, 8 lần lượt nối với ngõ vào của các Relay tương ứng.
- Các chân D1, D2, D4, RX, tức GPIO 2, 3, 4, 5 được nối với các nút nhấn
- Chân SD3 tức GPIO số 10 nối với cảm biến DHT11.

### 3.2.2.2. Khối ngõ ra công suất

Các GPIO của board ESP8266 nhận tín hiệu sẽ điều khiển kích hoạt các Relay hoạt động, từ đó điều khiển các thiết bị điện.

Relay là một loại thiết bị điện tự động mà tín hiệu đầu ra thay đổi nhảy cấp khi tín hiệu đầu vào đạt những giá trị xác định. Relay là thiết bị điện dùng để đóng cắt mạch điện điều khiển, bảo vệ và điều khiển sự làm việc của mạch điện động lực.

Chọn Relay 5V, chỉ cần cung cấp nguồn 5VDC và dòng khoảng 80mA cho Relay là các tiếp điểm có thể đóng ngắt khi được kích. Bên cạnh đó, dòng điện tối đa mà Relay có thể chịu được là 10A, nên đảm bảo dòng của các thiết bị điện khi chạy qua các tiếp điểm của Relay sẽ an toàn.

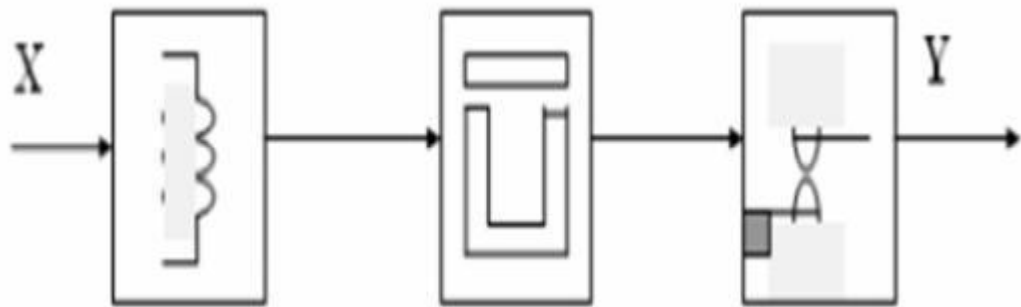


Relay 5V-10A

Các bộ phận chính của relay:

- Cơ cấu tiếp thu (khối tiếp thu): Có nhiệm vụ tiếp nhận những tín hiệu đầu vào và biến đổi nó thành đại lượng cần thiết cung cấp tín hiệu phù hợp cho khối trung gian.

- Cơ cấu trung gian (khôi trung gian): Làm nhiệm vụ tiếp nhận những tín hiệu đưa đến từ khối tiếp thu và biến đổi nó thành đại lượng cần thiết cho role tác động.
- Cơ cấu chấp hành (khôi chấp hành): Làm nhiệm vụ phát tín hiệu cho mạch điều khiển.



Sơ đồ khối của relay điện từ

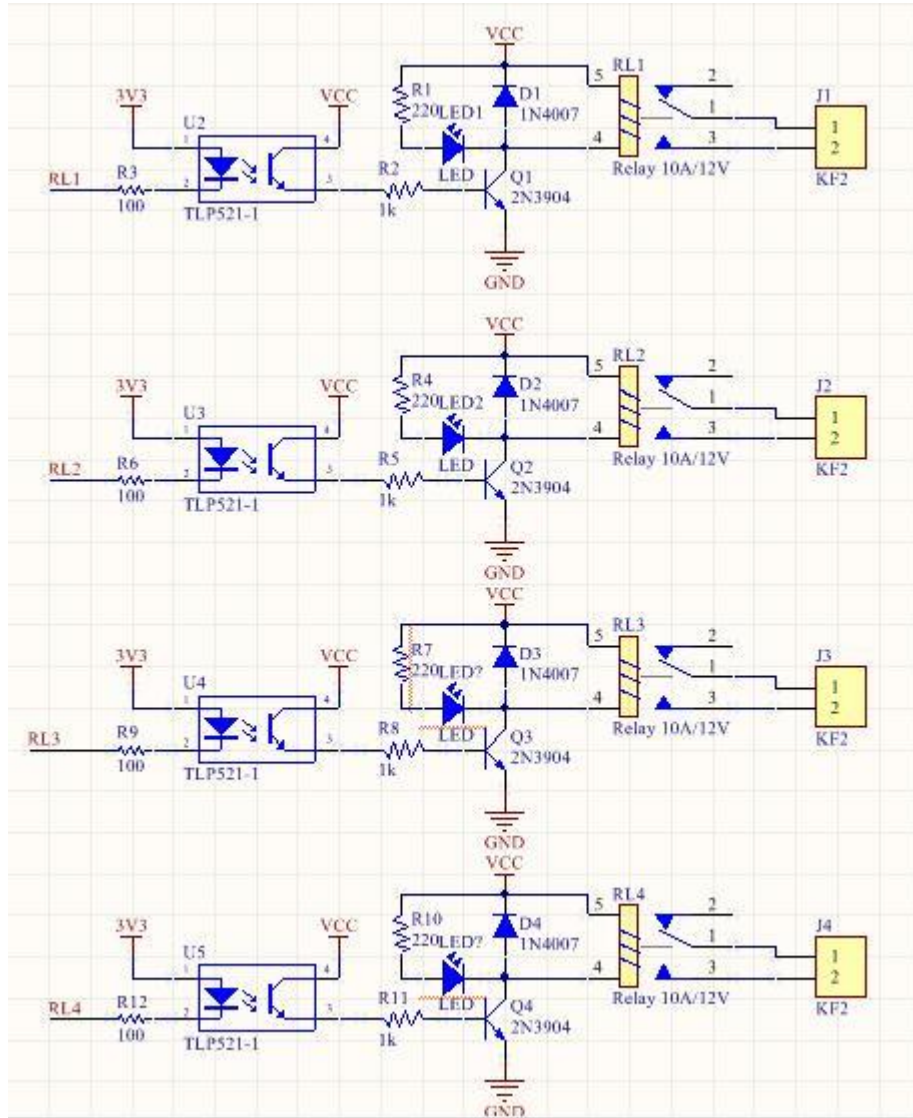
Relay có 3 tiếp điểm đóng ngắt NO (thường mở), NC (thường đóng) và chân COM, ở trạng thái bình thường khi chưa được kích chân COM sẽ nối với NC, khi kích chân COM chuyển sang nối với NO, NC mất kết nối. Đồng thời có 2 chân nguồn DC để cấp nguồn cho Relay hoạt động.

❖ Thông số kỹ thuật

Model	SRD-05VDC
Điện áp kích hoạt	5VDC
Dòng tiêu thụ	80mA
Dòng AC tối đa	10A-250/125VAC
Dòng DC tối đa	10A-30/28VDC
Số chân	5
Nhiệt độ làm việc	-25°C ~ +70°C

Thông số kỹ thuật Relay

## Thiết kế



Sơ đồ nguyên lý khối công suất ngõ ra



Relay 4 Kênh 5V gồm 4 rơ le hoạt động tại điện áp 5VDC, chịu được hiệu điện thế lên đến 250VAC 10A. Relay 4 kênh 5V được thiết kế chắc chắn, khả năng cách điện tốt. Trên module đã có sẵn mạch kích relay sử dụng transistor và IC cách ly quang giúp cách ly hoàn toàn mạch điều khiển (vi điều khiển) với rơ

le bảo đảm vi điều khiển hoạt động ổn định. Có sẵn header rất tiện dụng khi kết nối với vi điều khiển.

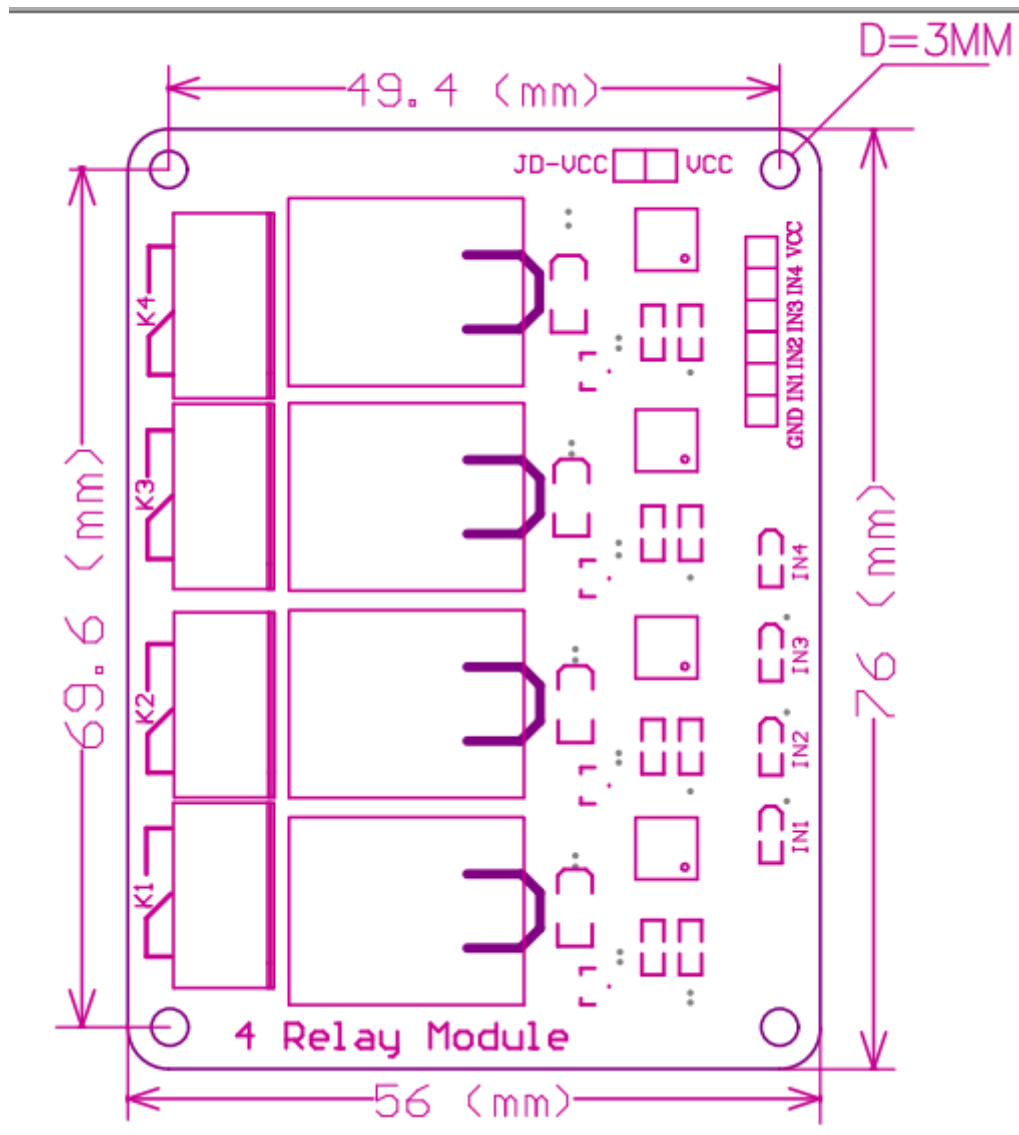
Relay 4 kênh sử dụng chân kích mức Thấp (0V), khi có tín hiệu 0V vào chân IN thì relay sẽ nhảy qua thường Hở của Relay. Ứng dụng với relay module khá nhiều bao gồm cả điện DC hay AC

Module được kết nối với các board điều khiển bằng 6 chân header như sau:

- VCC cung cấp nguồn cho các opto.
- GND kết nối với GND của board điều khiển.
- IN1, IN2, IN3, IN4 dùng để điều khiển relay 1, relay 2, relay 3, relay4 tích cực mức thấp.

Ngoài ra còn một 3 chân header được dùng để cấp nguồn cho relay, header này sẽ có một jumper dùng để kết nối chân VCC với chân RY\_VCC mục đích dùng chung nguồn VCC (5V) từ header 4 chân cho relay, thông thường jumper được nối lại với nhau. Nếu như muốn cách ly tín hiệu điều khiển với nguồn cấp cho relay thì có thể bỏ jumper này ra và cấp nguồn riêng 5V cho chân RY\_VCC.

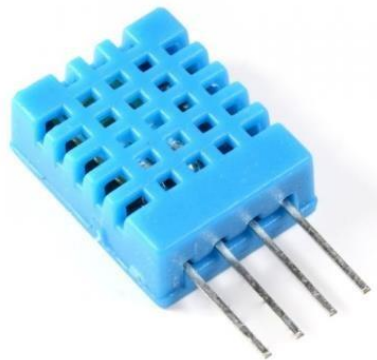
**Thông số kỹ thuật**



- Đóng ngắt được dòng điện cao: AC250V 10A, DC30V 10A
- 4 led báo trạng thái relay
- Điện áp điều khiển: 5V
- Mạch cách ly bằng opto
- Kích thước: 76x56 mm

### 3.2.2.3. Khôi cảm biến

Đề tài này có giám sát nhiệt độ độ ẩm, hiển thị lên internet. Với yêu cầu đề ra, em quyết định chọn cảm biến nhiệt độ độ ẩm DHT11. DHT11 là cảm biến ngõ ra số, mức điện áp hoạt động từ 3-5VDC, dòng cung cấp 0.5mA - 2.5mA phù hợp với dòng và áp ra của bộ xử lý trung tâm để module hoạt động bình thường



Cảm biến DHT11

### ❖ Thông số kỹ thuật

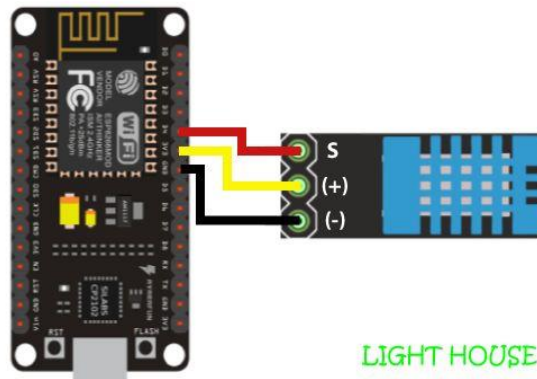
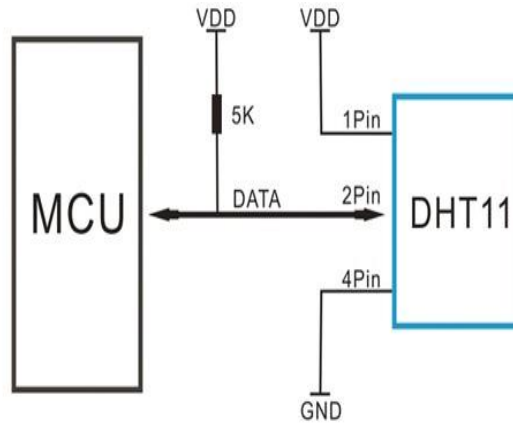
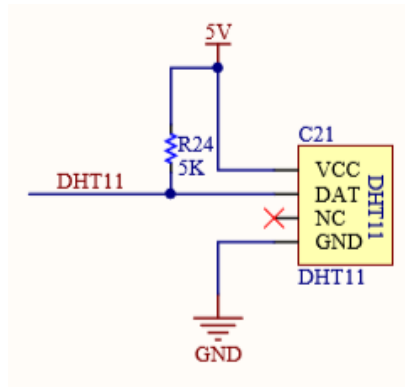
Điện áp hoạt động	3 - 5VDC
Dòng sử dụng	Tối đa 2.5mA
Đo nhiệt độ	0 - 50°C, sai số $\pm 2^\circ\text{C}$
Đo độ ẩm	20 - 80%, sai số $\pm 5\%$
Tốc độ lấy mẫu	1Hz (1 giây 1 lần)

Thông số kỹ thuật module cảm biến DHT11

### ❖ Thiết kế

Cảm biến DHT11 gồm 4 chân được kết nối như sau:

- Chân VCC được nối với nguồn 5VDC.
- Chân GND nối với chân GND của nguồn.
- Chân DATA nối với chân GPIO của vi điều khiển (ESP8266) qua một điện trở kéo lên nguồn



Sơ đồ kết nối cảm biến DHT11 với ESP8266

### 3.2.2.4. Nguồn

Nguồn chính sử dụng trong mạch là nguồn 5VDC. Nguồn này được lấy từ nguồn 220VAC qua module hạ áp AC-DC về 5V để cấp cho các module: board ESP8266, cảm biến DHT11, Relay.



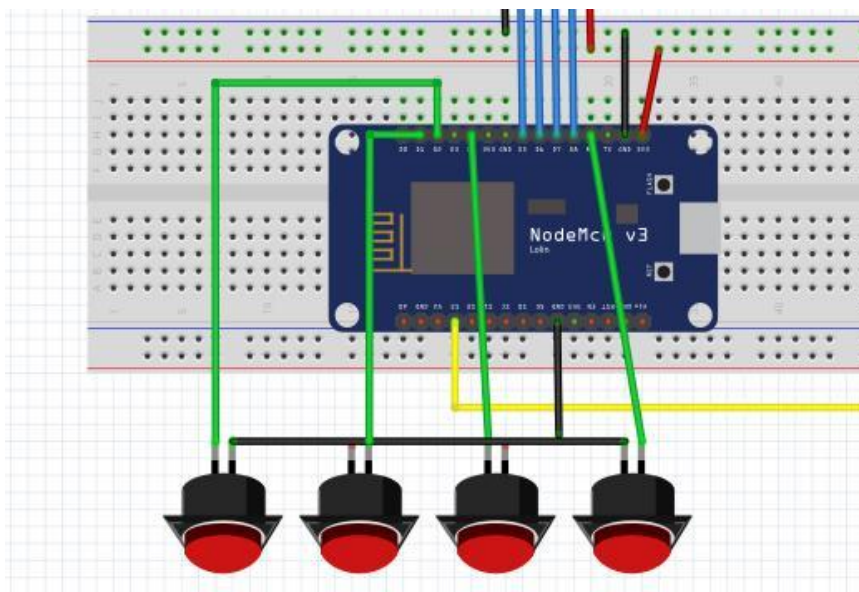
Module nguồn AC-DC

STT	Tên linh kiện	Số lượng	Dòng tiêu thụ (mA)	Tổng dòng điện (A)
1	ESP8266	1	150	0.15
2	DHT11	1	2.5	0.0025
3	Relay	4	80	0.32
4	Opto TLP521-1	4	70	0.28

Dòng điện của các linh kiện sử dụng trong mạch điều khiển

Từ bảng trên tổng dòng tiêu thụ cho toàn bộ mạch điều khiển là 0.7525A vì vậy ta sử dụng module nguồn AC-DC 5V-1A là hoàn toàn đủ để đáp ứng cho toàn mạch điều khiển.

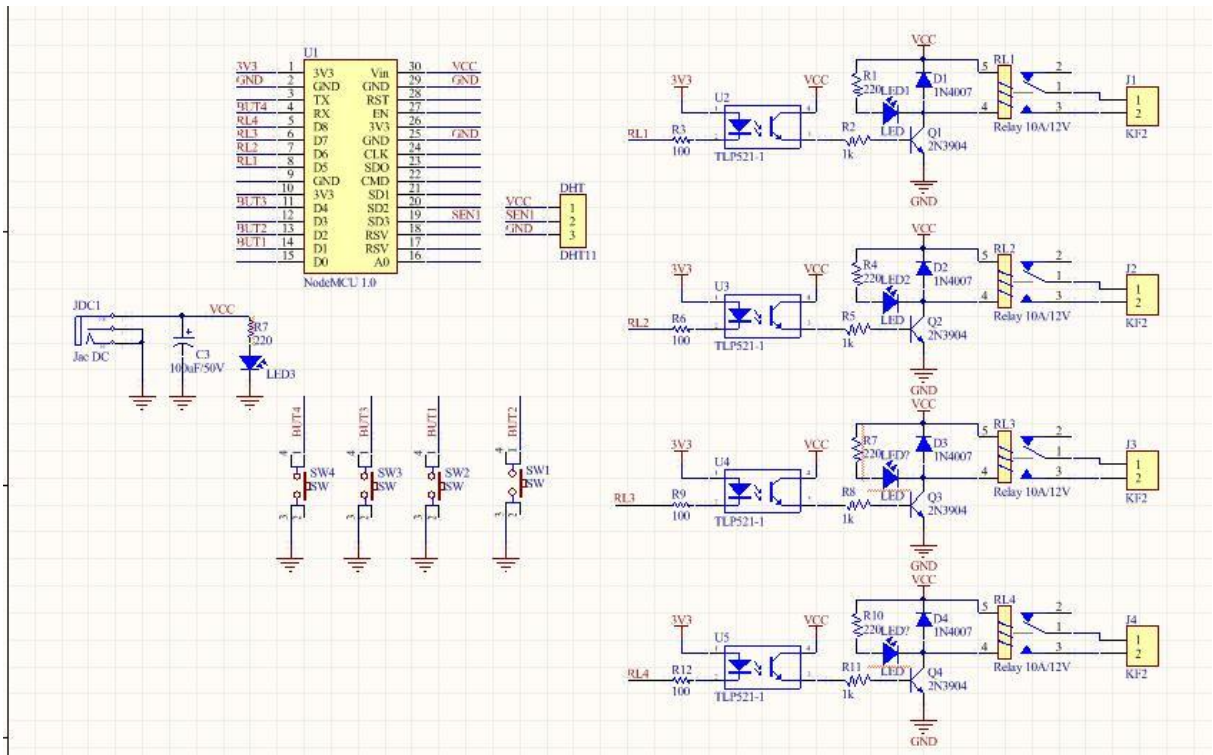
### 3.2.2.5. Khối điều khiển



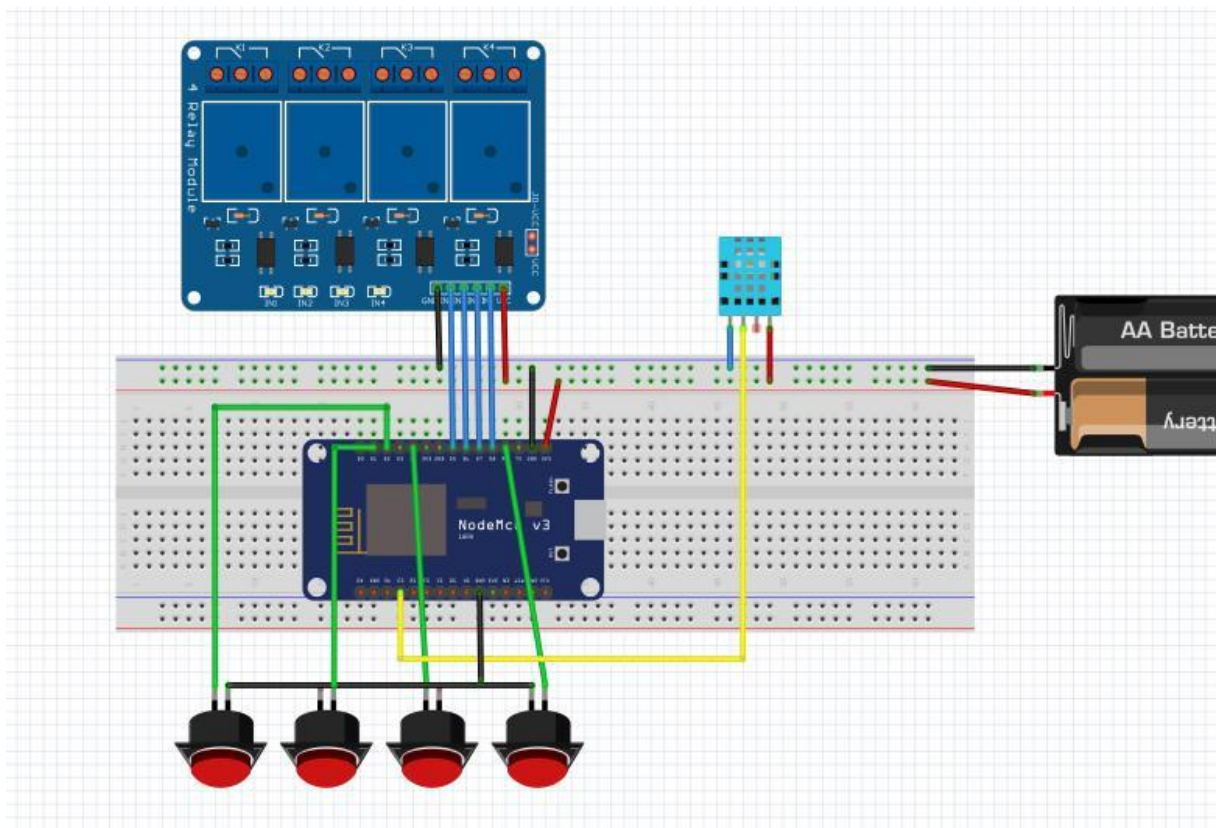
Sử dụng 4 nút nhấn nhỏ để điều khiển trực tiếp thiết bị qua cổng I/O



### 3.2.2.6. Sơ đồ nguyên lý toàn mạch



Sơ đồ nguyên lý toàn mạch



Mô phỏng trên phần mềm fritzing

### 3.2.2 Nguyên lý hoạt động

Để hệ thống hoạt động được trước tiên các thiết bị phải kết nối được với sóng Wifi. Riêng Google assistant, phần mềm điện thoại là các ứng dụng trên điện thoại nên yêu cầu điện thoại kết nối Wifi hoặc 3G.

Sau khi các thiết bị đã kết nối thành công với Wifi, việc điều khiển thiết bị sẽ thông qua điện thoại bằng 2 cách.

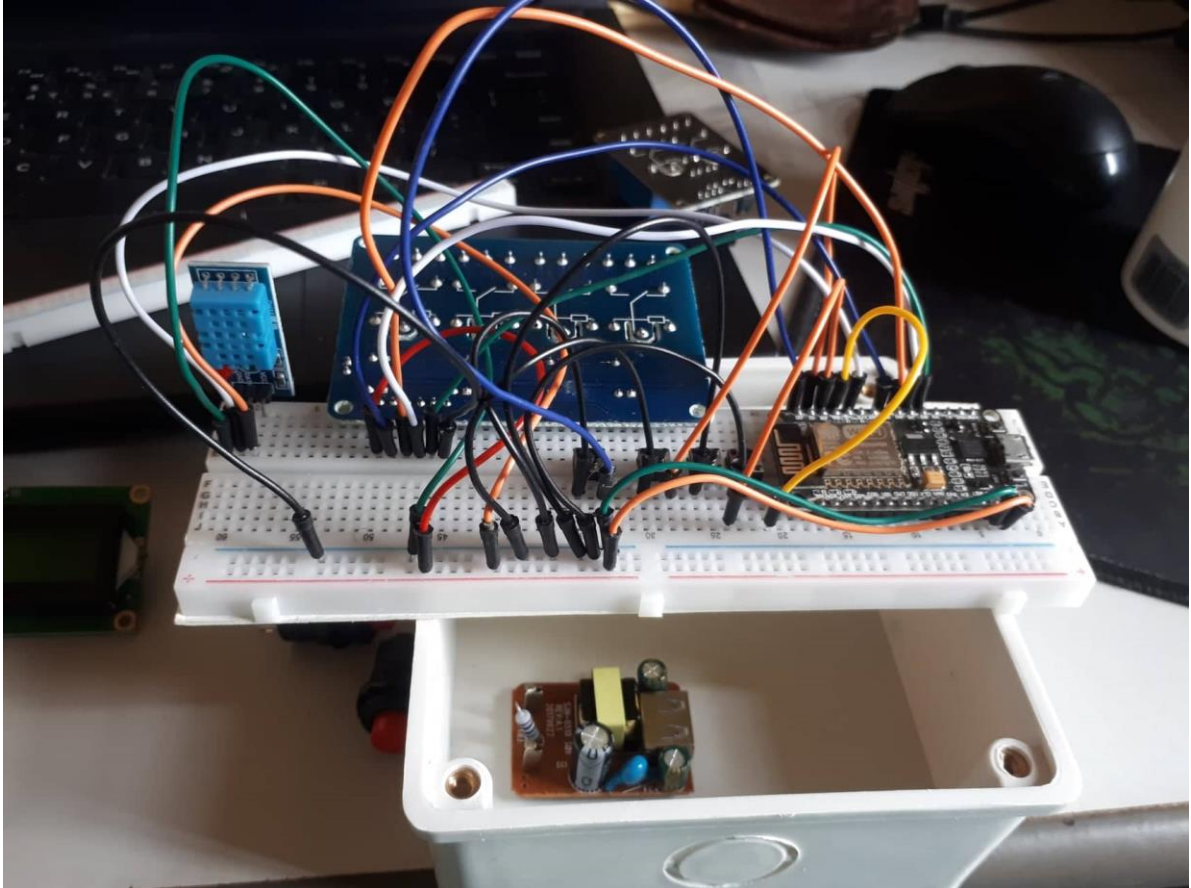
- Cách 1: Ta sẽ sử dụng trợ lý ảo Google (Google Assistant) để ra lệnh điều khiển thiết bị bằng giọng nói. Dữ liệu đó sẽ được esp8266 nhận xử lý để điều khiển các thiết bị điện sau đó đưa dữ liệu đó lên sever, riêng mạch giám sát nhiệt độ, độ ẩm ta chỉ hiển thị dữ liệu nhận được.
- Cách 2: Ta sẽ sử dụng đã App Android để điều khiển bằng cách gửi dữ liệu thông qua sever blynk, esp8266 sẽ nhận dữ liệu đó và điều khiển thiết bị điện.

Ngoài tính năng điều khiển thì phần mềm còn có chức năng hiển thị các trạng thái của thiết bị điện, giá trị nhiệt độ, độ ẩm, thông qua việc đọc dữ liệu trên sever do esp8266 gửi lên.

## Chương 4. THI CÔNG HỆ THỐNG

### 4.1 GIỚI THIỆU

Sau khi thực hiện xong quá trình tính toán các thiết bị để sử dụng trong mô hình em đã tiến hành việc xây dựng và thi công mô hình hệ thống.



Mô hình trên Breadboard

### 4.2 THI CÔNG HỆ THỐNG

#### 4.2.1 Thực hiện lắp ráp và ghép nối các mạch và Module

B1: Hàn module esp8266 vào mạch in sẵn

B2: Kết nối module esp8266 và module relay

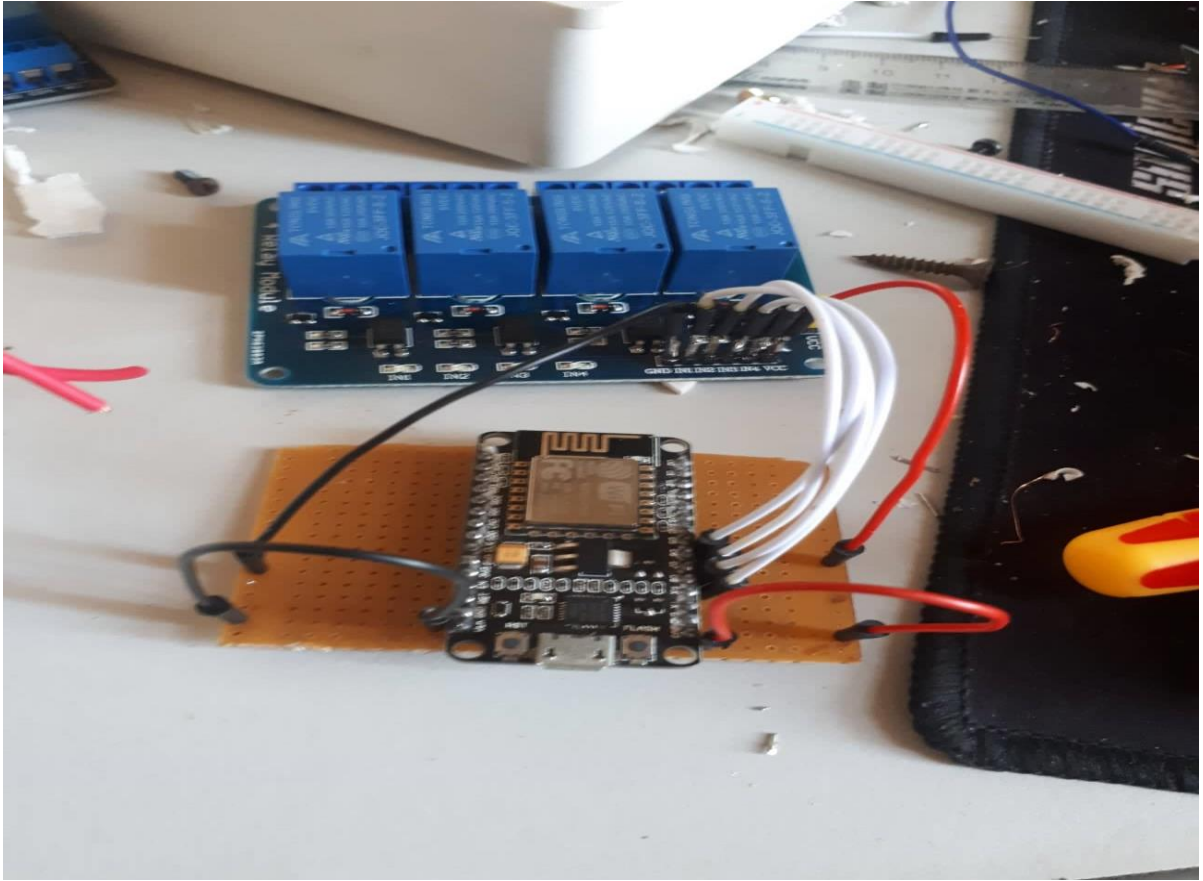
B3: Hàn module DHT11 vào mạch và kết nối với esp8266

B4: Cấp nguồn cho mạch từ module 5v1a

B5: Lắp nút nhấn và ổ cắm lên mặt hộp

B6: Đo kiểm tra từng chân của các thiết bị đã kết nối hết chưa

B7: Cuối cùng nạp chương trình và test lại chương trình có đạt như yêu cầu ban đầu không.

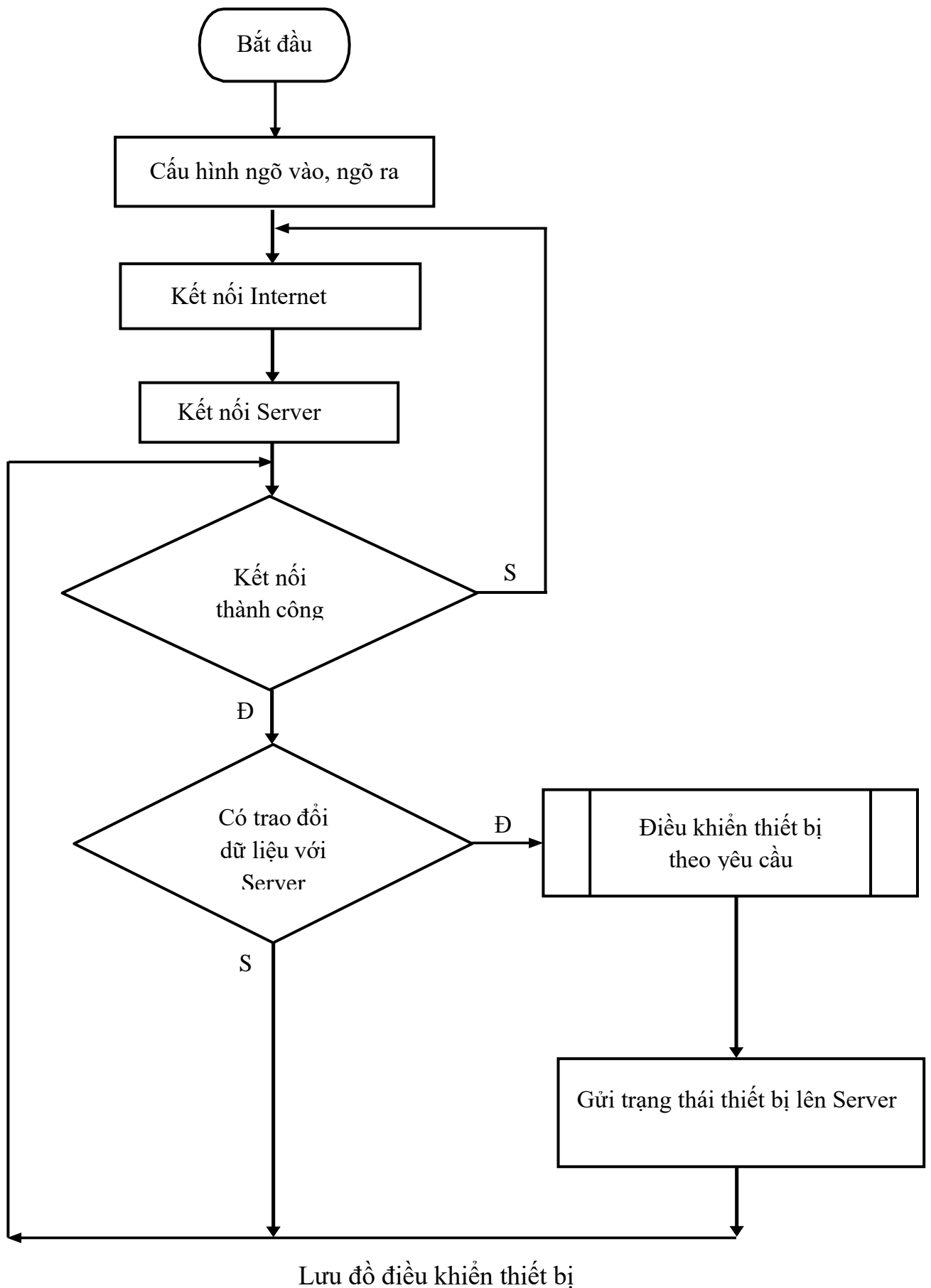


## 4.3 LẬP TRÌNH HỆ THỐNG

### 4.3.1 Lưu đồ giải thuật

Dựa vào lưu đồ ta thấy hoạt động của hệ thống hoạt động rõ ràng. Khi bắt đầu quá trình hoạt động thì sẽ thực hiện việc khởi tạo hệ thống. Kiểm tra hệ thống có được thiết lập hay chưa.

Hệ thống sẽ thực hiện việc kiểm tra xem có nhận được tín hiệu điều khiển hay chưa. Nếu có nhận được tín hiệu thì bắt đầu quá trình xử lý và đưa ra đề điều khiển thiết bị được kết nối.



### • **Giải thích:**

Mạch điều khiển thực hiện được trong 3 trường hợp:

- Khi người dùng tác động vào giao diện trên ứng dụng Android, thì thiết bị tương ứng được bật tắt, đồng thời sẽ gửi trạng thái của thiết bị vừa được tác động lên Server.
- Khi người dùng tác động vào nút nhấn, thì ESP sẽ nhận tín hiệu từ nút nhấn, thiết bị tương ứng được bật tắt, đồng thời sẽ gửi trạng thái của thiết bị vừa được tác động qua Server.
- Khi người dùng nói lệnh điều khiển qua google assistant. Lệnh sẽ được gửi tới ifttt, từ ifttt gửi về server, từ server về vi xử lý và điều khiển thiết bị

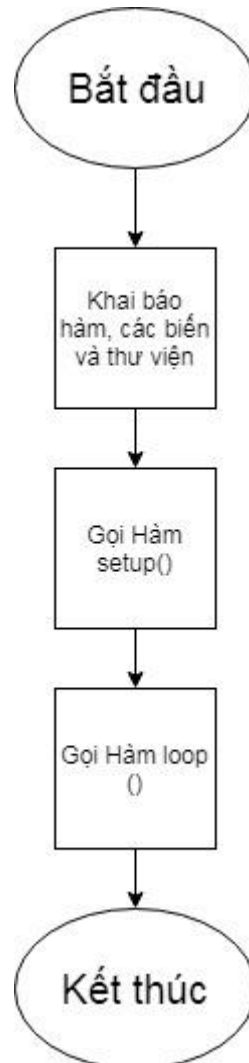
ESP8266 sẽ tiến hành kết nối Internet (Wifi), và thiết lập kết nối với Server. Mọi khi kết nối thành công. Nếu có trao đổi dữ liệu với Server (người dùng tác động vào giao diện ứng dụng Android hoặc có tín hiệu từ Server hoặc nút nhấn gửi xuống), thì thiết bị sẽ được điều khiển theo yêu cầu người dùng. Ở bất kỳ trường hợp điều khiển nào thì trạng thái điều khiển của thiết bị cũng đều được đồng bộ trên điện thoại.

#### **4.3.2 Phần mềm lập trình cho vi điều khiển**

Arduino IDE là môi trường phát triển tích hợp với mã nguồn mở của Arduino. Đây là một ứng dụng đa nền tảng và được viết trên nền tảng bằng ngôn ngữ Java, và từ IDE này chương trình sẽ được sử dụng cho ngôn ngữ lập trình nguồn mở khác. Chương trình được thiết kế nhằm giúp cho những người dùng mới có thể làm quen dễ dàng với lĩnh vực phát triển phần mềm. Nó bao gồm đầy đủ các phần như các phần mềm lập trình khác nhưng với mức độ dễ sử dụng hơn như: đánh dấu cú pháp, tự động canh lề, biên dịch và nạp chương trình lên board. Chương trình của Arduino được gọi là Sketch.

Các chương trình khi lập trình trên phần mềm được viết bằng ngôn ngữ C hoặc C++. Trên Arduino IDE người dùng chỉ cần định nghĩa 2 hàm để tạo ra được một chương trình hoàn chỉnh có thể chạy được gồm:

- Setup(): hàm này chạy mỗi khi khởi động chương trình, dùng để thiết đặt các thông số cài đặt từ đầu.
- Loop(): hàm này được hiểu là vòng lặp cho đến khi không sử dụng nữa hay ngắt nguồn board điều khiển.



Lưu đồ một chương trình trong Arduino IDE

### 4.3.3 Chương trình điều khiển

```

#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include "DHT.h"
#define DHTPIN 10 // Chân DATA nối với chân io10
#define DHTTYPE DHT11 // DHT 11
BlynkTimer timer;
  
```

```

DHT dht(DHTPIN, DHTTYPE);

char auth[] = "d092297fdb3149c6a12073e30bbfd0e3";
char ssid[] = "Thanh Dat 2";
char pass[] = "thanhdatt";
int virtualPin;
int flag=0, flag1=0, flag2=0, flag3=0;
//Gửi dữ liệu trạng thái Button từ Blynk về Nodemcu
BLYNK_WRITE(V0)
{virtualPin = param.asInt(); // gán giá trị cho biến V0
  flag = 1;}
BLYNK_WRITE(V1)
{virtualPin = param.asInt(); // gán giá trị cho biến V1
  flag1 = 1 ;}
BLYNK_WRITE(V2)
{virtualPin = param.asInt(); // gán giá trị cho biến V2
  flag2 = 1 ;}
BLYNK_WRITE(V3)
{virtualPin = param.asInt(); // gán giá trị cho biến V3
  flag3 = 1 ;}
//Đồng bộ lại trạng thái khi mất nguồn
BLYNK_CONNECTED(){
  Blynk.syncVirtual(V0);
  Blynk.syncVirtual(V1);
  Blynk.syncVirtual(V2);
  Blynk.syncVirtual(V3);
}
void setup()
{
  Serial.begin(115200); // Khởi tạo cổng serial baud 115200

```



```
Blynk.begin(auth, ssid, pass);  
Serial.println("DHTxx test!");  
dht.begin();// Bắt đầu đọc dữ liệu  
  
pinMode(2, INPUT_PULLUP);  
pinMode(3, INPUT_PULLUP);  
pinMode(4, INPUT_PULLUP);  
pinMode(5, INPUT_PULLUP);  
pinMode(12, OUTPUT);  
pinMode(13, OUTPUT);  
pinMode(14, OUTPUT);  
pinMode(15, OUTPUT);  
  
timer.setInterval(1000L, sendSensor);// 1s đọc cảm biến DHT 1 lần  
}  
void sendSensor()  
{  
float h = dht.readHumidity();//Gan gia tri do am vao bien h  
float t = dht.readTemperature();//Gan gia tri nhiet do vao bien t  
// gan vao bien tren blink  
Blynk.virtualWrite(V10, h);  
Blynk.virtualWrite(V11, t);  
// IN thông tin ra màn hình  
Serial.print("Do am: ");  
Serial.print(h);  
Serial.print(" %\t");  
Serial.print("Nhiet do: ");  
Serial.print(t);  
Serial.println(" *C ");  
}
```



```

if(virtualPin == 0){
  digitalWrite(12, LOW);
  }
  else digitalWrite(12, HIGH);
}

if(digitalRead(5) == LOW){delay(200);
  while(digitalRead(5) == LOW);
  int ledStatus = digitalRead(12);
  if(ledStatus == 0){
    digitalWrite(12, HIGH);
    Blynk.virtualWrite(V1,1); //Ghi dữ liệu từ Nodemcu lên Blynk
    //led1.off();
    }
    else {
      digitalWrite(12, LOW);
      Blynk.virtualWrite(V1,0); ////Ghi dữ liệu từ Nodemcu lên Blynk
      // led1.on();
      }
    }
flag1 = 0 ;
////////////////////////////////////
if(flag2 == 1){
if(virtualPin == 0){
  digitalWrite(13, LOW);
  }
  else digitalWrite(13, HIGH);
}

if(digitalRead(3) == LOW){delay(200);

```

```

while(digitalRead(3) == LOW);
int ledStatus = digitalRead(13);
if(ledStatus == 0){
    digitalWrite(13, HIGH);
    Blynk.virtualWrite(V2,1); //Ghi dữ liệu từ Nodemcu lên Blynk
    //led1.off();
}
else {
    digitalWrite(13, LOW);
    Blynk.virtualWrite(V2,0); ///Ghi dữ liệu từ Nodemcu lên Blynk
    // led1.on();
}
}
flag2 = 0 ;
////////////////////////////////////
if(flag3 == 1){
if(virtualPin == 0){
    digitalWrite(15, LOW);
}
else digitalWrite(15, HIGH);
}

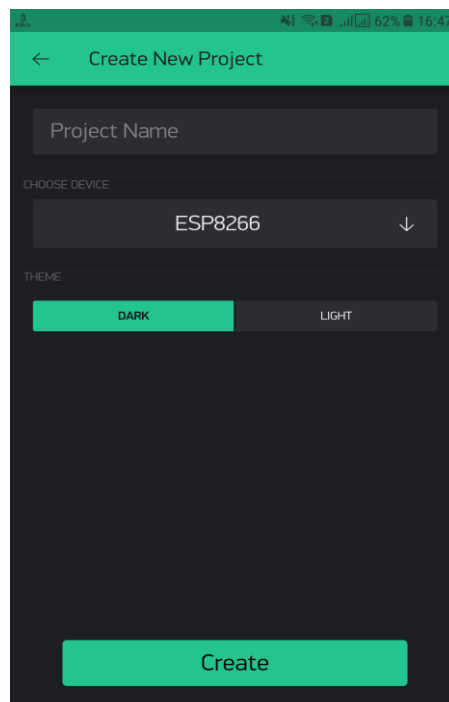
if(digitalRead(2) == LOW){delay(200);
    while(digitalRead(2) == LOW);
    int ledStatus = digitalRead(15);
    if(ledStatus == 0){
        digitalWrite(15, HIGH);
        Blynk.virtualWrite(V3,1); //Ghi dữ liệu từ Nodemcu lên Blynk
        //led1.off();
    }
}

```

```
else {  
  digitalWrite(15, LOW);  
  Blynk.virtualWrite(V3,0); ////Ghi dữ liệu từ Nodemcu lên Blynk  
  // led1.on();  
}  
}  
flag3 = 0 ;  
timer.run();  
}
```

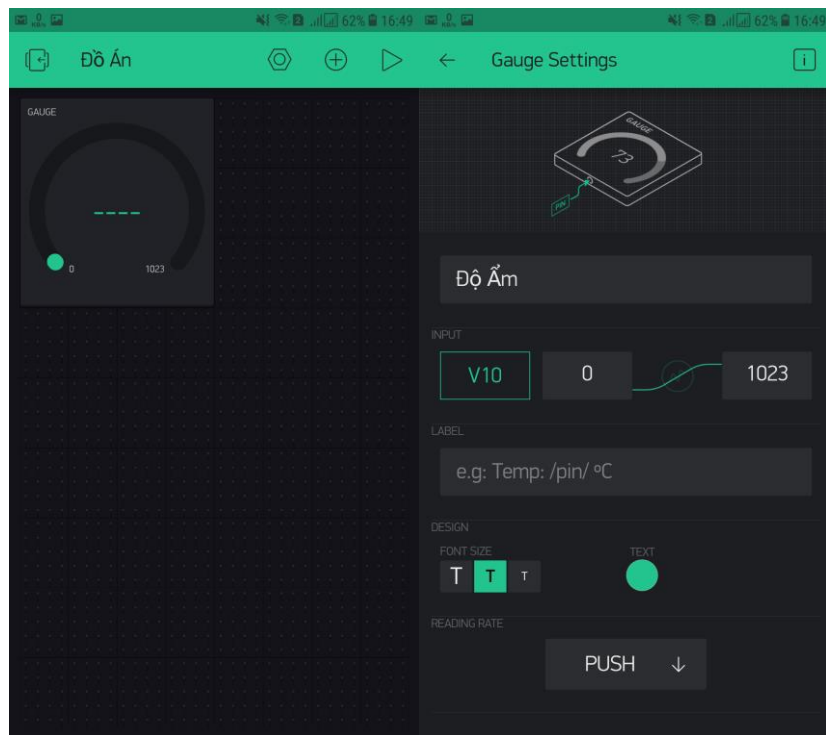
#### 4.3.4 Điều khiển qua điện thoại với Blynk

Trước tiên sẽ là tạo giao diện thông qua app Blynk trên smartphone có các chức năng sau:



- Đọc và hiển thị toàn bộ dữ liệu từ cảm biến và trạng thái của các relay
- Điều khiển từ xa 4 relay
- Hiển thị thông báo hệ thống offline hoặc online

Ta sẽ tạo ra 2 tab để hiển thị thông tin trên:

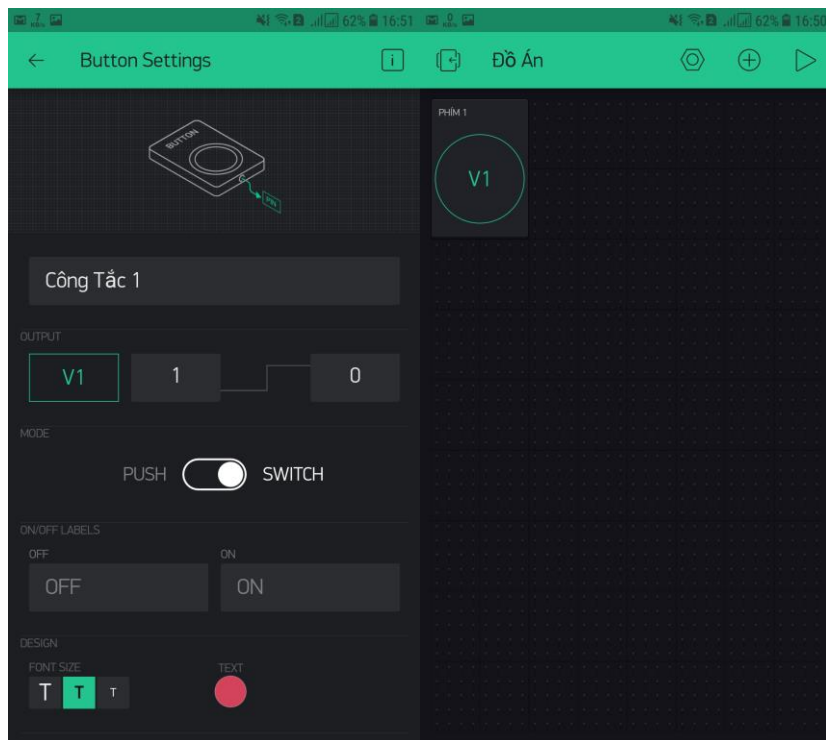


#### Cảm biến:

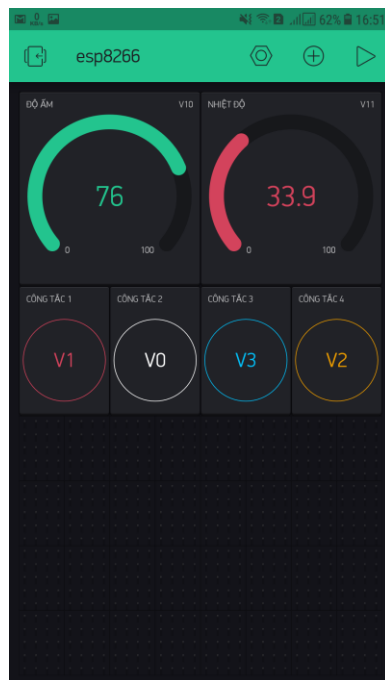
- Nhiệt độ, input là V10 value 0 – 50, frequency: 5s
- Độ ẩm, input là V11 value 0 – 100, frequency:5s

#### Điều khiển:

- Nút Công tắc 1, output: V1 1 – 0, mode:push, label: on, off
- Nút Công tắc 2, output: V2 1 – 0, mode:push, label: on, off
- Nút Công tắc 3, output: V3 1 – 0, mode:push, label: on, off
- Nút Công tắc 4, output: V4 1 – 0, mode:push, label: on, off

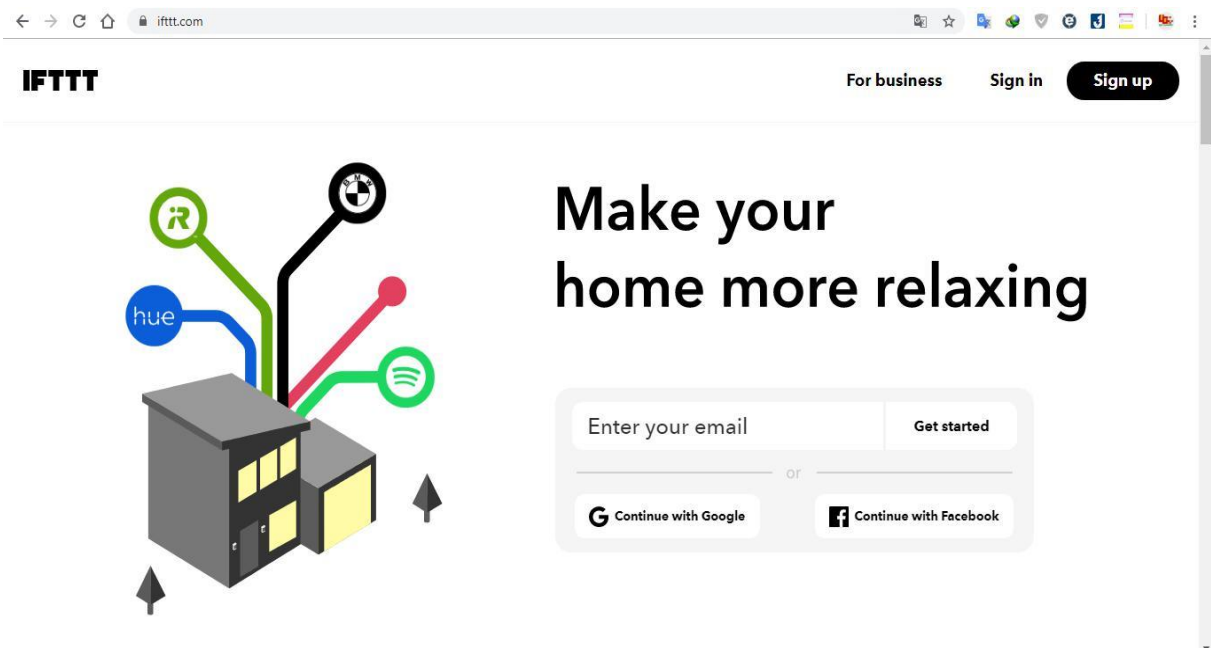


Sau khi hoàn tất ta được:

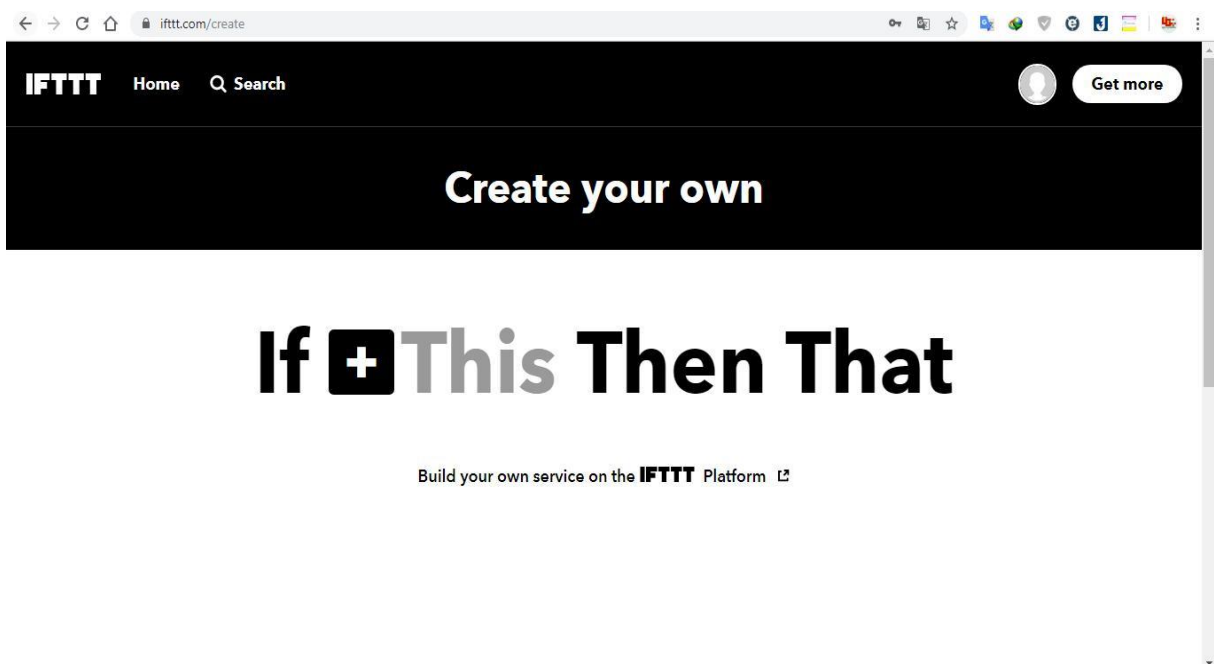


#### 4.3.5 Công cụ hỗ trợ IFTTT

Để điều khiển Relay thông qua giọng nói trên Google Assistant thì phải cần công cụ IFTTT.

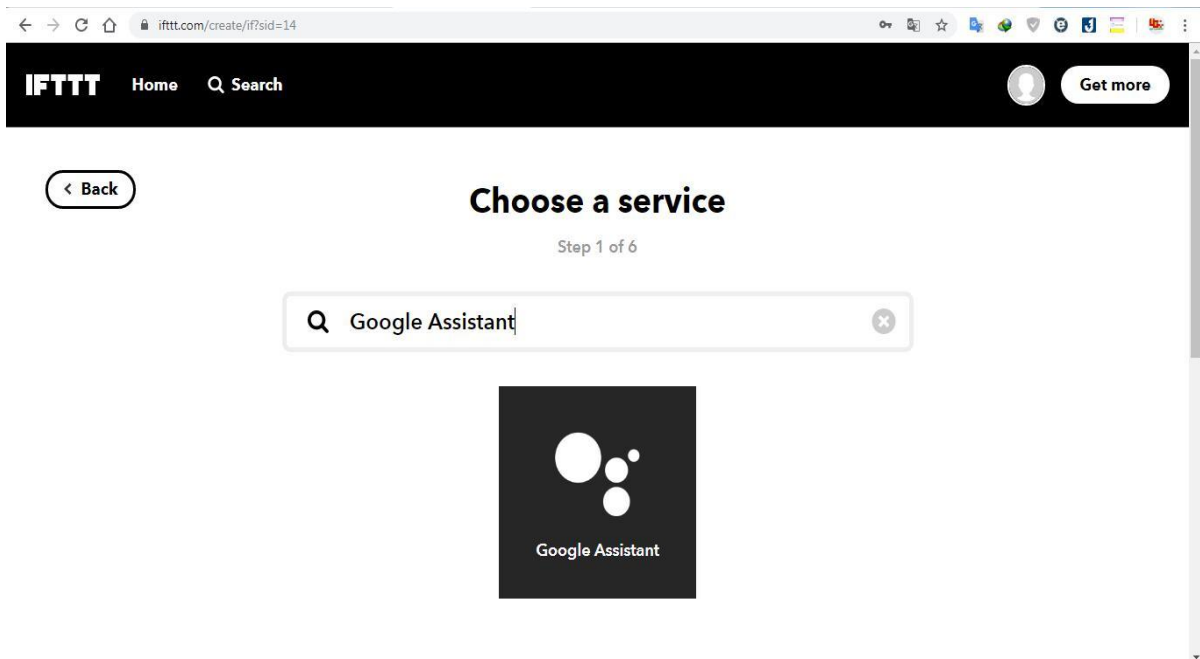


- Bước đầu tiên là đăng nhập vào Web IFTTT để đăng ký tài khoản ở đây ta đăng ký bằng tài khoản Google

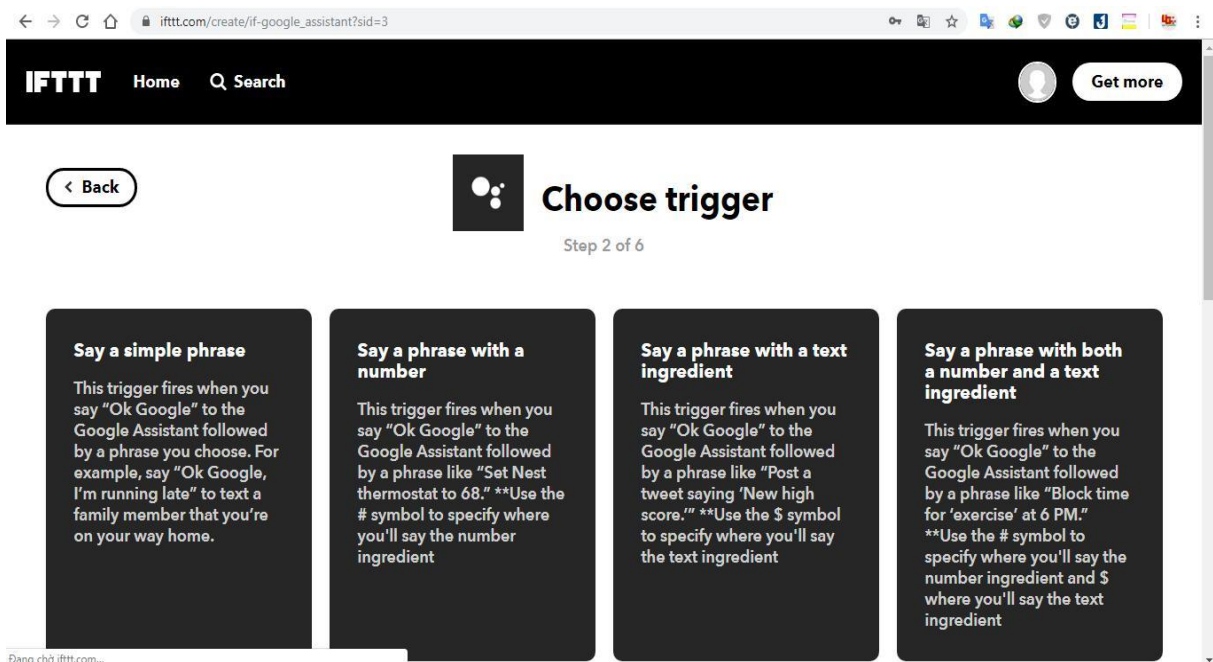


- Sau khi đăng nhập thành công ta Click vào "My Applets" và chọn "New Applet" để tạo một dự án mới.
- Bấm vào + this





Tìm kiếm đến Google Assistant và bấm kết nối.



Chọn thẻ "Say a simple phrase."

← → ↻ ifttt.com/create/ff-say-a-simple-phrase?sid=4

Step 2 of 6

What do you want to say?

What's another way to say it? (optional)

And another way? (optional)

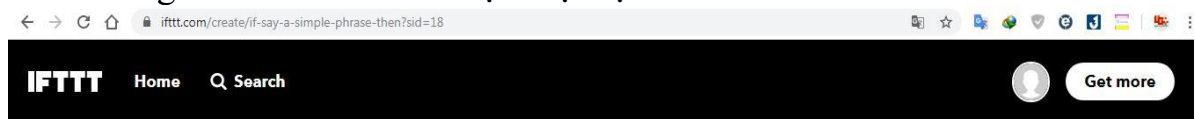
What do you want the Assistant to say in response?

Language  
English ▾

What do you want to say?: ta đặt tên câu lệnh cần nói ở đây là " turn on the device 1".

Ngoài ra ở 2 mục tiếp theo ta có thể đặt thêm những câu lệnh khác để có thể bật/tắt thiết bị của mình.

What do you want the Assistant to say in the response?: Ở phần này là câu phản hồi từ Google Assistant khi ta thực hiện lệnh vừa nói.



< Back

If  Then  That

<https://ifttt.com/create/ff-say-a-simple-phrase-then?sid=18#>

Tiếp theo ta chọn + that

< Back

## Choose action service

Step 3 of 6

Webhooks



Và chọn Webhooks

### Make a web request

This action will make a web request to a publicly accessible URL. NOTE: Requests may be rate limited.

**URL**

`http://188.166.206.43/90e6fc3ed332435587af6799fb58de88 / update / D16`

Surround any text with "<<>>" to escape the content **Add ingredient**

**Method**

PUT

The method of the request e.g. GET, POST, DELETE

**Content Type**

application/json

Optional

**Body**

["1"]

Ta điền đầy đủ các thông tin trên trường hiển thị

**http://188.166.206.43:** Đây là URL của Blynk Server.

**d092297fdb3149c6a12073e30bbfd0e3:** Mã Token được cấp khi chúng ta tạo một dự án mới trên App Blynk.

**D16:** chân để điều khiển tín hiệu

Sau khi điền đầy đủ thông tin rồi ấn Create action là đã hoàn thành 1 câu lệnh. Các câu lệnh khác làm tương tự như vậy.

## **4.4 Hướng dẫn sử dụng, thao tác**

### **4.4.1. Viết tài liệu hướng dẫn sử dụng**

- Bước 1: Tiến hành cấp nguồn 220VAC cho toàn bộ hệ thống bao gồm nguồn 220V cho ngõ ra mạch công suất, và nguồn 220V cho hệ thống để tạo nguồn DC nuôi mạch điều khiển.
- Bước 2: Chờ đến khi ESP8266 kết nối Wifi, mở ứng dụng Android. Khi blynk đã được kết nối, tiến hành điều khiển, giám sát trực tiếp các thiết bị sử dụng trên giao diện.

### **4.4.2. Quy trình thao tác**

**Cấp nguồn → Kiểm tra kết nối wifi → Đợi kết nối cloud blynk → Điều khiển thiết bị**

Sau khi tiến hành cấp nguồn, hệ thống sẽ hoạt động như sau:

- **Điều khiển trực tiếp bằng điện thoại**

Người sử dụng mở ứng dụng Android, nhấn nút điều khiển trên màn hình, tín hiệu điều khiển được chuyển đến khối xử lý trung tâm, khối xử lý trung tâm gửi tín hiệu điều khiển đến mạch công suất, các thiết bị điện sẽ được bật tắt tương ứng với thao tác người dùng. Sau đó trạng thái của thiết bị sẽ được gửi lên Server thông qua Internet. Server này chính là Cloud Blynk, nơi tiến hành xử lý dữ liệu của từng thiết bị.

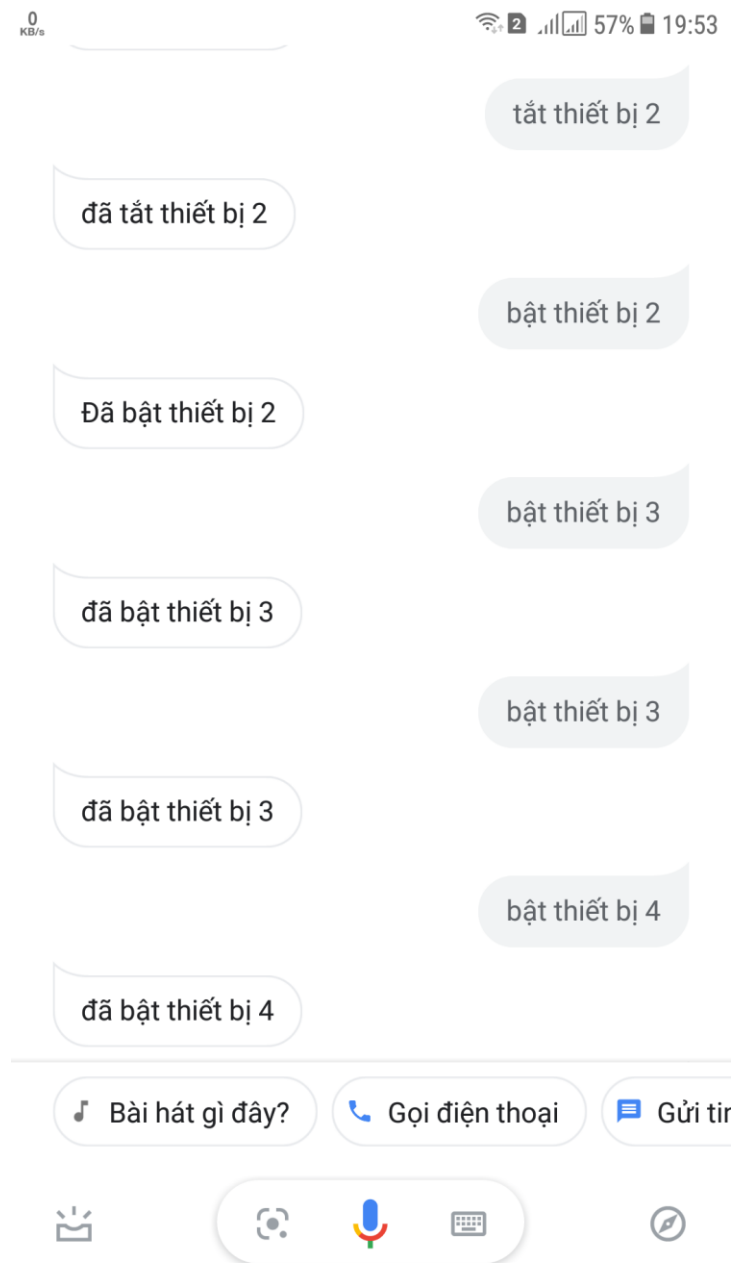
- **Điều khiển thông qua nút nhấn**

Ta trực tiếp nhấn nút điều khiển trên bộ điều khiển, nút nhấn sẽ gửi xung tín hiệu kích hoạt mức thấp về khối xử lý trung tâm. Bộ xử lý trung tâm nhận được

tín hiệu và tác động đến khối công suất để tiến hành bật tắt thiết bị. Đồng thời Server cũng cập nhật trạng thái của thiết bị vừa được điều khiển.

- **Điều khiển thông qua giọng nói**

Đối với trợ lý ảo Google Assistant ta sẽ dùng lệnh “ Ok Google” để kích hoạt ứng dụng và ra các câu lệnh đã được cài đặt sẵn. Lưu ý mỗi lần thực hiện ra lệnh chúng ta phải sử dụng từ khóa “ Ok Google”



Bất kể khi điều khiển tại ứng dụng Android, nút nhấn hay google assistant (một trong ba cái trên), thì trạng thái của thiết bị được điều khiển đều cập nhật ngay lập tức ở hai cái còn lại.

## Chương 5. KẾT QUẢ, NHẬN XÉT VÀ ĐÁNH GIÁ

### 5.1. GIỚI THIỆU

Sau thời gian thực hiện đề tài, qua quá trình nghiên cứu và tìm hiểu em đã học hỏi được nhiều kiến thức mới, công nghệ mới. Đồng thời áp dụng được nhiều lý thuyết vào dự án thực tế.

### 5.2. KẾT QUẢ ĐẠT ĐƯỢC

Board ESP8266 là mạch điều khiển chính trong hệ thống. Em đã biết cách ứng dụng chức năng tích hợp Wifi trong board để phát triển hệ thống điều khiển thiết bị từ xa. Từ đó nắm được bản chất điều khiển, để có thể mở rộng cho các ứng dụng IoT sau này, điều khiển được một hệ thống lớn.

Thông qua đề tài em đã biết cách điều khiển và giám sát thiết bị qua Internet (Wifi). Đồng thời thiết kế được mô hình, ứng dụng Android cho hệ thống điều khiển.

Đề tài có sử dụng cảm biến đo nhiệt độ, độ ẩm, qua đó em biết được cấu tạo, nguyên lý hoạt động cũng như cách sử dụng cảm biến. Cảm biến này rất phổ biến, thích hợp cho việc giám sát nhiệt độ, độ ẩm trên các hệ thống nhỏ.

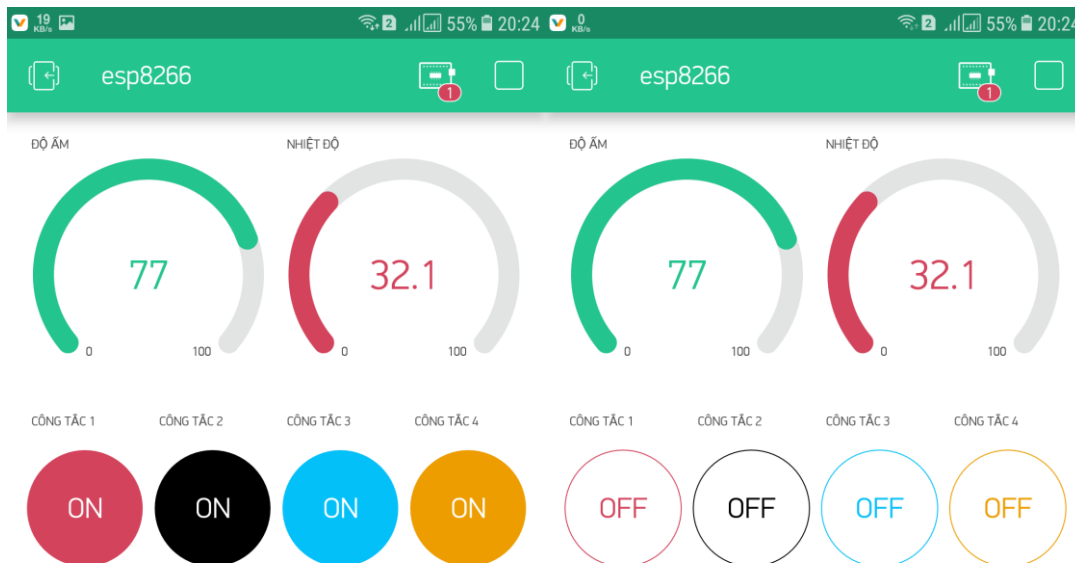
### 5.3. KẾT QUẢ THỰC NGHIỆM

#### 5.3.1. Mô hình

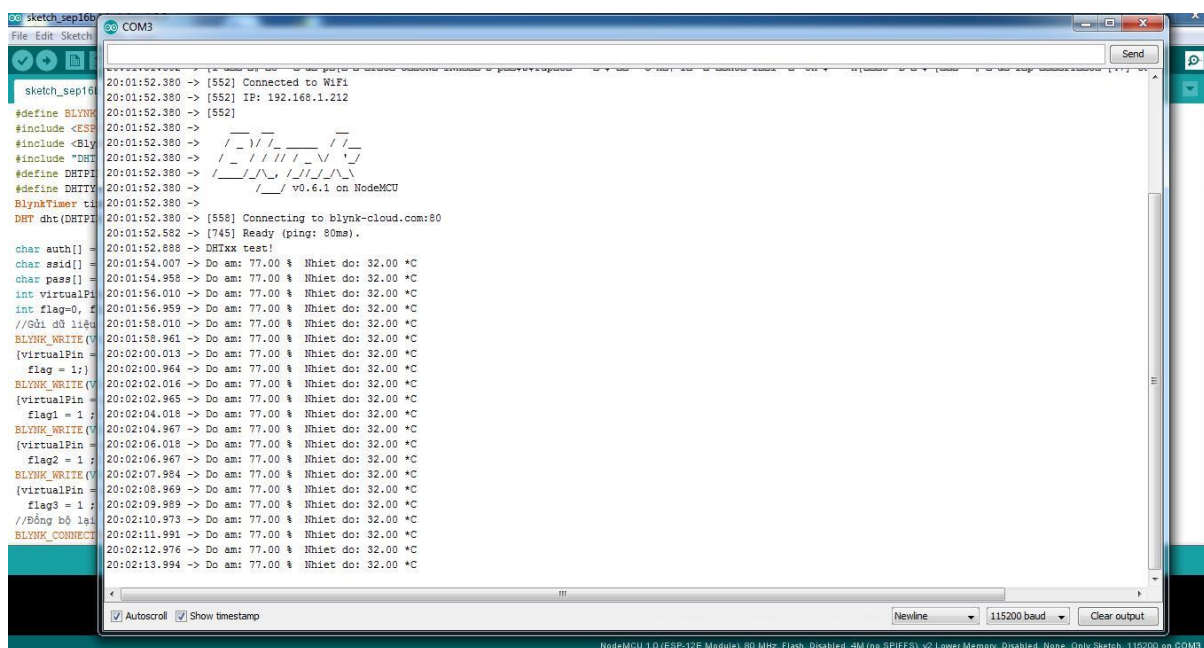


Mô hình hệ thống

### 5.3.2. Điều khiển và giám sát thiết bị,



Giao diện màn hình điều khiển khi tất cả thiết bị bật và tắt



Chạy chương trình trên phần mềm

## 5.4. NHẬN XÉT VÀ ĐÁNH GIÁ

### 5.4.1. Nhận xét

Sau khi thực hiện, em đã hoàn thành hệ thống đáp ứng cơ bản những yêu cầu ban đầu đặt ra, dưới đây là một số nhận xét:

- **Ưu điểm**
  - Hệ thống hoạt động ổn định qua nhiều lần thử nghiệm.

- Đồng bộ trạng thái điều khiển từ mô hình hệ thống, ứng dụng Android và Server.
- Giám sát được nhiệt độ và độ ẩm của môi trường xung quanh.
- Tốc độ điều khiển bằng tay tương đối nhanh.
- Giao diện điều khiển trực quan, đẹp mắt.
- Dễ dàng sử dụng, lắp đặt và bảo dưỡng.
- **Nhược điểm**
  - Hệ thống phụ thuộc vào tốc độ mạng Wifi, và sự ổn định của Server Cloud Blynk.
  - Chưa tích hợp nhiều tính năng thành một hệ thống IoT hoàn chỉnh.

#### 5.4.2. Đánh giá

Trong quá trình vận hành hệ thống, em đã ghi nhận lại kết quả được tổng hợp.

Công việc	Số lần thao tác	Số lần thành công	Thời gian đáp ứng	Đánh giá
Điều khiển thiết bị qua ứng dụng	50	48	1 – 2 giây	Đạt
Điều khiển thiết bị qua google assistant	50	48	3 – 4 giây	Đạt
Điều khiển bằng nút nhấn	50	50	1 giây	Đạt
Giám sát cảm biến	Ổn định	Ổn định	2 giây	Đạt
<b>Đánh giá chung</b>				Đạt

#### Số liệu thực nghiệm

Qua những số liệu được thống kê ở bảng trên, em đánh giá hệ thống về cơ bản đã đáp ứng được mục tiêu đặt ra. Hệ thống hoạt động ổn định sau nhiều lần chạy, kiểm tra thử trong nhiều trường hợp. Mô hình nhỏ gọn, thẩm mỹ, nhưng vẫn đảm bảo tính an toàn cao, dễ dàng lắp đặt và sử dụng. Nhưng để đưa hệ thống này áp dụng vào thực tế thì em cần phải hoàn thiện một số phần như sau: tăng tốc độ điều khiển cũng như phản hồi, tối ưu hóa mô hình, thêm một số chức năng như: giám sát nơi điều khiển, cảnh báo chống trộm, báo cháy, ...



# CHƯƠNG 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## 6.1. KẾT LUẬN

Sau thời gian nghiên cứu, thi công thì đồ án tốt nghiệp của em với đề tài **“giám sát nhiệt độ độ ẩm và điều khiển thiết bị điện qua internet dùng module arduino esp8266”** đã hoàn thiện, đáp ứng được những yêu cầu ban đầu đặt ra.

### 6.1.1. Ưu điểm

- Mạch điều khiển nhỏ gọn, hoạt động khá ổn định, thời gian đáp ứng khá nhanh.
- Giao diện điều khiển và giám sát dễ sử dụng, thân thiện người dùng.
- Mô hình hệ thống có độ chính xác, tính an toàn và dễ dàng thao tác với người dùng.
- Phù hợp cho các hệ thống điện trong phòng học, hộ gia đình.

Nhìn chung, mô hình đã hoạt động tương đối ổn định, có thể làm việc liên tục, đạt 100% yêu cầu đề ra ban đầu. Bên cạnh đó hệ thống mở rộng thêm chức năng điều khiển bằng giọng nói. Người dùng thao tác một cách đơn giản, dễ sử dụng.

### 6.1.2. Khuyết điểm

Tuy nhiên, do sự hạn chế về kiến thức và thời gian thực hiện, nguồn tài liệu tham khảo chủ yếu thông qua internet nên đề tài không tránh khỏi sai sót và còn một số hạn chế:

- Hạn chế lớn nhất là tác động điều khiển còn chậm do giao thức hoạt động chính sử dụng dịch vụ Cloud.
- Hệ thống phụ thuộc vào nguồn điện 220VAC, và tốc độ truy cập mạng Internet.
- Hoạt động chủ yếu tại môi trường có phủ sóng wifi.
- Hộp mô hình còn mang tính tượng trưng.
- Kích thước sản phẩm còn thô, thiếu tính thẩm mỹ.
- Số lượng thiết bị còn hạn chế
- Giới hạn về thời gian, kiến thức nên hệ thống chưa được tối ưu.

Nhìn chung hệ thống điều khiển về cơ bản hoạt động tốt.

## **6.2. HƯỚNG PHÁT TRIỂN**

Hệ thống hiện tại đã đáp ứng được việc điều khiển các thiết bị. Trong quá trình thực hiện, em thấy rằng đề tài này rất phổ biến, có tính ứng dụng rất cao trong nhiều dự án thực tế. Vì vậy em đưa ra một số đề xuất nhằm cải tiến và nâng cấp hệ thống:

- Mở rộng số lượng cũng như công suất thiết bị điều khiển.
- Giám sát nơi điều khiển bằng camera, cảnh báo chống trộm, báo cháy.
- Điều chỉnh độ sáng đèn, tốc độ quạt, nhiệt độ điều hòa, ...
- Thiết lập hệ thống điều khiển thiết bị tự động nhằm tối ưu hóa việc sử dụng và tiết kiệm điện năng.
- Ứng dụng đề tài vào hệ thống thực tế.

# TÀI LIỆU THAM KHẢO

## Sách tham khảo

1. Giáo Trình Kỹ Thuật Lập Trình C Căn Bản Và Nâng Cao – Phạm Văn Ất
2. Nguyễn Đình Phú, Giáo trình Vi điều khiển PIC, Nhà xuất bản ĐH Quốc gia Tp. Hồ Chí Minh, 8/2016.
3. Nguyễn Đình Phú – Nguyễn Trường Duy, Giáo trình Kỹ thuật số, Nhà xuất bản ĐH Quốc gia Tp. Hồ Chí Minh.

## Các Webside tham khảo

1. [https://cdn-shop.adafruit.com/product-files/2471/0AESP8266\\_Datasheet\\_EN\\_v4.3.pdf](https://cdn-shop.adafruit.com/product-files/2471/0AESP8266_Datasheet_EN_v4.3.pdf) ESP8266
2. <http://www.micropik.com/PDF/dht11.pdf> Cảm biến DHT11
3. <https://tailieu.vn/>
4. <https://www.youtube.com/>
5. <https://esp8266.vn/>