



THIẾT KẾ HỆ THỐNG ĐIỀU KHIỂN THIẾT BỊ TRONG NHÀ BẰNG GIỌNG NÓI SỬ DỤNG ESP8266

Vật liệu linh kiện điện tử (Trường Đại học Công nghiệp Hà Nội)



Scan to open on Studocu

**BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC CNKT ĐIỆN TỬ - VIỄN THÔNG

**THIẾT KẾ HỆ THỐNG ĐIỀU KHIỂN THIẾT BỊ TRONG NHÀ BẦNG
GIỌNG NÓI SỬ DỤNG ESP8266**

CBHD : Tiến Sĩ.Trần Đình Thông

Sinh viên : Chúc Minh Sơn

Mã số sinh viên : 2017602147

Hà Nội – Năm 2021

Downloaded by Nguyen Phuoc Du (21119327@student.hcmute.edu.vn)

MỤC LỤC

LỜI NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN	1
MỤC LỤC	2
DANH MỤC HÌNH ẢNH	5
LỜI CẢM ƠN	7
LỜI MỞ ĐẦU	8
CHƯƠNG 1. TỔNG QUAN LÝ THUYẾT VỀ HỆ THỐNG ĐIỀU KHIỂN THIẾT BỊ TRONG NHÀ BẰNG GIỌNG NÓI SỬ DỤNG ESP8266	10
1.1. Tình hình nghiên cứu các thiết bị điều khiển bằng giọng nói hiện nay	10
1.2. Tìm hiểu những thiết bị điều khiển thiết bị bằng giọng nói hiện nay[2]	10
1.2.1. Đèn GE	10
1.2.2. Camera Nest Cam IQ Indoor	11
1.2.3. Bộ điều chỉnh nhiệt Nest	12
1.2.4. Bộ điều chỉnh nhiệt Tado và điều khiển điều hòa thông minh	12
1.3. Tổng quan về IOT(Internet Of Things)[1]	13
1.3.1. Khái niệm	13
1.3.2. Ứng dụng	14
1.4. Tìm hiểu về ESP8266 và Kit Wifi ESP8266 NodeMCU Lua V3 CH340 [3]	16
1.4.1. Khái niệm chung về ESP8266	16
1.4.2. Phân cứng của ESP8266	17
1.4.3. Phân loại ESP8266	18
1.4.4. Kit Wifi ESP8266 NodeMCU Lua V3 CH340	20
1.5. Tìm hiểu về Google Assistant[4]	21
1.5.1. Khái niệm Google Assistant	21
1.5.2. Hoạt động Google Assistant	21
1.5.3. Server IFTTT	23

1.6. Ứng dụng Blynk theo dõi trạng thái các thiết bị[5]	23
1.6.1. Hệ sinh thái Blynk	23
1.6.2. Tính năng của Blynk.....	24
1.7. Một số linh kiện và thiết bị khác được sử dụng trong mạch[6].....	25
1.7.1. Module Relay.....	25
1.7.2. Tụ gốm	26
1.7.3. Diode.....	26
1.7.4. Quạt tản nhiệt.....	27
1.7.5. Bóng đèn Led.....	27
1.7.6. Máy bơm mini 5V.....	28
1.8. Kết luận	29
CHƯƠNG 2. THIẾT KẾ HỆ THỐNG ĐIỀU KHIỂN THIẾT BỊ TRONG NHÀ BẰNG GIỌNG NÓI SỬ DỤNG ESP8266.....	30
2.1. Thiết kế sơ đồ khối bộ điều khiển thiết bị điện, điện tử gia dụng bằng giọng nói.....	30
2.1.1. Yêu cầu	30
2.1.2. Sơ đồ khối	30
2.2. Thiết kế mạch phần cứng	31
2.2.1. Khối nguồn.....	31
2.2.2. Khối vi điều khiển.....	32
2.2.3. Khối nút nhấn.....	32
2.2.4. Khối thiết bị	33
2.3. Xây dựng phần mềm điều khiển	34
2.3.1. Phần mềm lập trình	34
2.3.2. Lưu đồ thuật toán	37
2.3.3. Triển khai phần mềm	38
2.4. Kết luận chương 2.....	48
CHƯƠNG 3. MÔ HÌNH THỰC NGHIỆM VÀ KẾT QUẢ	49
3.1. Mạch nguyên lí và mạch in hoàn thiện.....	49
3.2. Kết quả thực nghiệm.....	51

3.3. Giao diện Google Assistant khi nhận lệnh.....	52
3.4. Giao diện App Blynk	54
3.5. Đánh giá kết quả và hướng phát triển	57
3.5.1. Đánh giá kết quả	57
3.5.2. Hướng phát triển của đề tài.....	57
3.6. Kết luận chương 3	58
KẾT LUẬN	59
TÀI LIỆU THAM KHẢO	60
PHỤ LỤC	61

DANH MỤC HÌNH ẢNH

Hình 1.1.Đèn GE[2]	11
Hình 1.2. Camera Nest Cam IQ Indoor[2].....	11
Hình 1.3. Bộ điều chỉnh nhiệt Nest[2]	12
Hình 1.4. Bộ điều chỉnh nhiệt Tado[2]	13
Hình 1.5. Internet of Things	14
Hình 1.6.Theo dõi lộ trình đi của xe chở hàng	15
Hình 1.7. Theo dõi tình trạng sinh trưởng của cây trồng.....	16
Hình 1.8.Modul ESP8266-12E[3].....	18
Hình 1.9.Adafruit HUZZAH ESP8266 (ESP-12) [3]	18
Hình 1.10.NodeMCU 1.0[3]	19
Hình 1.11.Node MCU V3 CH340 – kit RF thu phát wifi ESP8266[3]	20
Hình 1.12.Sơ đồ hệ sinh thái Blynk	24
Hình 1.13.Relay[6].....	25
Hình 1.14.Tụ gốm[6].....	26
Hình 1.15.Diode[6]	26
Hình 1.16. Quạt tản nhiệt[6]	27
Hình 1.17.Đèn led usb[6]	27
Hình 1.18. Máy bơm mini DC 5V[6].....	28
Hình 2.1. Sơ đồ khối hệ thống	30
Hình 2.2.Khối mạch nguồn	32
Hình 2.3. Mạch vi điều khiển.....	32
Hình 2.4.Mạch khối nút nhấn.....	33
Hình 2.5.Mạch khối thiết bị	33
Hình 2.6. Phần mềm Arduino IDE.....	34
Hình 2.7.Giao diện của Arduino IDE	34
Hình 2.8. Các phím chức năng của Arduino IDE	35
Hình 2.9. Lưu đồ thuật toán	37
Hình 2.10. Các bước tạo tài khoản IFTTT	38

Hình 2.11. Các bước tạo câu điều khiển This	39
Hình 2.12. Lựa chọn phương thức giọng nói	40
Hình 2.13. Cài đặt lệnh bằng giọng nói	41
Hình 2.14. Phản hồi That	42
Hình 2.15. Giao diện tạo tài khoản Blynk.....	43
Hình 2.16. Nhập tài khoản gmail và mật khẩu	44
Hình 2.17. Tạo Project mới.....	45
Hình 2.18. Thông báo mã Token được gửi tới gmail	46
Hình 2.19. Lấy nút nhấn ra.....	47
Hình 2.20. Hoàn thành thiết lập	48
Hình 3.1. Mạch nguyên lí hoàn thiện.....	49
Hình 3.2. Mạch in 2D.....	50
Hình 3.3. Mạch in 3D.....	51
Hình 3.4. Mô hình hệ thống	52
Hình 3.5. Giao diện điều khiển bằng giọng nói	53
Hình 3.6. Giao diện App Blynk bật đèn.....	54
Hình 3.7. Giao diện App Blynk bật quạt.....	55
Hình 3.8. Giao diện App Blynk bật máy bơm	56

LỜI CẢM ƠN

Em xin chân thành cảm ơn quý thầy, cô trường **Đại Học Công Nghiệp Hà Nội** đã tận tình dạy dỗ chúng em trong suốt những năm qua. Trong đó phải kể đến quý thầy cô trong **Khoa Điện Tử - Viễn Thông** đã tạo điều kiện để em thực hiện đề tài thực tập tốt nghiệp này.

Đặc biệt, em xin chân thành cảm ơn giáo viên hướng dẫn thầy **Tiến Sĩ. Trần Đình Thông** đã tận tình giúp đỡ, hỗ trợ em trong quá trình thực hiện đề tài. Cung cấp cho em những kiến thức quý báu cũng như những lời khuyên cực kỳ hữu ích. Tạo động lực cho em hoàn thành tốt nhiệm vụ của mình.

Bên cạnh đó, em cũng xin cảm ơn các bạn sinh viên trong **Khoa Điện Tử - Viễn Thông** đã đóng góp ý kiến cho bản thân em cũng như các bạn trong khoa, giúp chúng em thực hiện đề tài đạt hiệu quả tốt hơn.

Với thời gian thực hiện đề tài còn ngắn, kiến thức bản thân còn hạn hẹp, điều kiện kinh phí còn hạn chế và dù đã rất cố gắng hết sức nhưng trong quá trình thực hiện đề tài không tránh khỏi những sai sót. Do đó em rất mong nhận được những ý kiến đóng góp và chỉ dẫn thêm những sai sót và khuyết điểm của đề tài thực hiện từ của quý thầy cô và các bạn. Chính từ điều đó sẽ giúp bản thân em tích lũy thêm được kinh nghiệm và giúp đề tài có thể áp dụng vào thực tế cuộc sống.

Sinh viên

Chúc Minh Sơn

LỜI MỞ ĐẦU

Lí do lựa chọn đề tài

Hiện nay với nền khoa học, công nghệ phát triển với các trợ lý ảo thông minh hỗ trợ AI như: Google Assistant (Google), Alexa (Amazon), Siri (Apple), ... Các thiết bị trong ngôi nhà thông minh ngoài được điều khiển qua các app trên điện thoại, web,... còn được điều khiển qua trợ lý ảo (giọng nói) cũng dần được phát triển. Điều này hỗ trợ rất lớn trong việc sử dụng những thiết bị ở xa trong nhà, giúp điều khiển và theo dõi chúng một cách dễ dàng. Không chỉ dừng lại ở việc điều khiển trong gia đình, các ứng dụng còn hướng tới các đối tượng người khuyết tật giúp họ làm quen và có cuộc sống tiện nghi hơn.

Công nghệ 4.0 đang là làn sóng lớn trên thế giới nói chung và Việt Nam nói riêng, việc ứng dụng trí tuệ nhân tạo giúp cải thiện đời sống là điều tất yếu. Máy móc thay thế sức lao động con người nhưng con người mới là người tạo ra máy móc. Là một sinh viên chuyên ngành Công Nghệ Kỹ Thuật Điện Tử - Viễn Thông chúng ta phải biết nắm bắt xu hướng và vận dụng những kiến thức đã được học để phát triển nó một cách có hiệu quả nhằm góp phần vào sự phát triển nền khoa học kỹ thuật thế giới nói chung và trong sự phát triển kỹ thuật điện tử, nói riêng.

Vì vậy, em đã tìm hiểu, chọn đề tài “*Thiết kế hệ thống điều khiển thiết bị trong nhà bằng giọng nói sử dụng ESP8266*” để làm đồ án tốt nghiệp của mình. Kết quả nghiên cứu từ đề tài này sẽ giúp em có nhiều kinh nghiệm để sau khi tốt nghiệp chúng em có đủ khả năng nghiên cứu chế tạo hoàn chỉnh thiết bị điều khiển hệ thống điện cho ngôi nhà thông minh đáp ứng được sử dụng yêu cầu trên thị trường với giá thành hợp lý, chất lượng đảm bảo, phù hợp với điều kiện sống tại Việt Nam.

Mục tiêu đề tài

Về phía đề tài, “*Thiết kế hệ thống điều khiển thiết bị trong nhà bằng giọng nói sử dụng ESP8266*” sử dụng bộ vi điều khiển làm trung tâm cùng trợ lý ảo Google Assistant để thực hiện điều khiển các thiết bị thông qua giọng nói.

Về phía sinh viên, qua đề tài có thể hiểu hơn về công dụng và cách sử dụng của một số linh kiện điện tử. Rèn luyện kỹ năng thiết kế phần cứng cũng như giao tiếp với ESP8266.

Nội dung nghiên cứu

Đối tượng nghiên cứu

- Tìm hiểu về ESP8266
- Tìm hiểu về hệ thống điều khiển thiết bị điện, điện tử gia dụng bằng giọng nói.
- Tìm hiểu về Google Assistant (Google Home), IOT.
- Thiết kế mạch điều khiển thiết bị bằng giọng nói qua Google Assistant (Google Home)
- Xây dựng mô hình mẫu điều khiển thiết bị qua Google Assistant (Google Home)
- Kiểm tra, đánh giá tính ứng dụng của đề tài.

Kết quả sản phẩm cần đạt được:

- Hệ thống tích hợp nhiều thiết bị được điều khiển thông qua Google Assistant.
- Ngoài ra người dùng cũng có thể theo dõi trạng thái các thiết bị qua App Blynk.

CHƯƠNG 1. TỔNG QUAN LÝ THUYẾT VỀ HỆ THỐNG ĐIỀU KHIỂN THIẾT BỊ TRONG NHÀ BẰNG GIỌNG NÓI SỬ DỤNG ESP8266

1.1. Tình hình nghiên cứu các thiết bị điều khiển bằng giọng nói hiện nay

Nhận định của giới công nghệ cho thấy, với sự phát triển mạnh mẽ của cuộc Cách mạng công nghệ 4.0 kỷ nguyên vạn vật kết nối internet (IoT), công nghệ nhận dạng giọng nói cùng các ứng dụng của nó trong đời sống sẽ trở thành xu hướng tất yếu.

Khảo sát cho thấy, điều khiển bằng giọng nói hiện nay đã được sử dụng tại nhiều quốc gia, cụ thể là ở Mỹ có tới 48% và tại Anh có 43% sử dụng thiết bị điều khiển nhà bằng giọng nói.

Tại Việt Nam hiện nay, khái niệm điều khiển các thiết bị bằng giọng nói còn mới mẻ và nhiều hạn chế, chưa được ứng dụng rộng rãi trong đời sống sinh hoạt.

1.2. Tìm hiểu những thiết bị điều khiển thiết bị bằng giọng nói hiện nay[2]

1.2.1. Đèn GE

GE được xem là tên thương hiệu bóng đèn được nhiều người biết đến, đơn giản là sản phẩm này chính là Made for Google. Loại bóng đèn này được kết nối Bluetooth thông qua một đường truyền trực tiếp tới Google Home với cách thiết lập đơn giản.

Khi sử dụng Google Assistant, bạn có thể điều khiển bóng đèn: bật, tắt cũng như điều chỉnh cường độ sáng

Thương hiệu bóng đèn này hiếm khi xuất hiện ở Anh nhưng lại phổ biến ở Mỹ.



Hình 1.1. Đèn GE[2]

1.2.2. Camera Nest Cam IQ Indoor

Camera Nest Cam IQ Indoor được Google Assistant điều khiển qua phiên bản nâng cấp của mic và bộ phận loa của nó. Thiết bị này sở hữu cảm biến 4K, tính năng nhận diện khuôn mặt, theo dõi cận cảnh HDR và nó hoạt động trong hệ sinh thái Nest.

Bạn có thể yêu cầu xem các cảnh quay trực tiếp trên điện thoại qua ứng dụng Nest, với câu lệnh đơn giản như "OK Google, cho tôi xem phòng ăn".



Hình 1.2. Camera Nest Cam IQ Indoor[2]

1.2.3. Bộ điều chỉnh nhiệt Nest

Bộ điều chỉnh nhiệt Nest có khả năng tương thích với Google Home, cho phép bạn kiểm soát được nhiệt độ của nước nóng, khu vực không gian mình đang ở,....



Hình 1.3. Bộ điều chỉnh nhiệt Nest[2]

1.2.4. Bộ điều chỉnh nhiệt Tado và điều khiển điều hòa thông minh

Dùng Google Assistant có thể điều khiển sản phẩm này với những câu lệnh đơn giản để tăng hoặc giảm nhiệt độ của máy lạnh. Chính vì khả năng tương thích mở rộng của bộ điều nhiệt Tado, nên sự xuất hiện của nó không thể thiếu trong những ngôi nhà hiện đại.



Hình 1.4. Bộ điều chỉnh nhiệt Tado[2]

1.3. Tổng quan về IOT(Internet Of Things)[1]

1.3.1. Khái niệm

Internet of Things (IoT) là thuật ngữ dùng để chỉ các đối tượng có thể được nhận biết cũng như sự tồn tại của chúng trong một kiến trúc mạng tính kết nối. Đây là một viễn cảnh trong đó mọi vật, mọi con vật hoặc con người được cung cấp các định danh và khả năng tự động truyền tải dữ liệu qua một mạng lưới mà không cần sự tương tác giữa con người-với-con người hoặc con người-với-máy tính. IoT tiến hoá từ sự hội tụ của các công nghệ không dây, hệ thống vi cơ điện tử (MEMS) và Internet. Cụm từ này được đưa ra bởi Kevin Ashton vào năm 1999. Ông là một nhà khoa học đã sáng lập ra Trung tâm Auto-ID ở đại học MIT.



Hình 1.5. Internet of Things

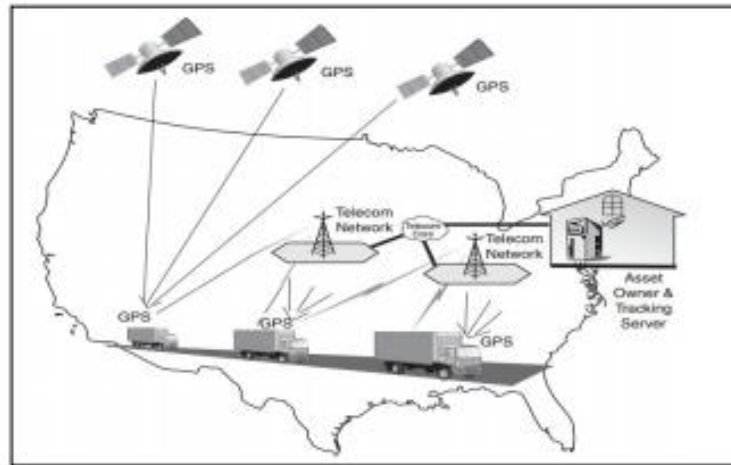
"Thing" - sự vật - trong Internet of Things, có thể là một trang trại động vật với bộ tiếp sóng chip sinh học, một chiếc xe ô tô tích hợp các cảm biến để cảnh báo lái xe khi lốp quá non, hoặc bất kỳ đồ vật nào do tự nhiên sinh ra hoặc do con người sản xuất ra mà có thể được gán với một địa chỉ IP và được cung cấp khả năng truyền tải dữ liệu qua mạng lưới. IoT phải có 2 thuộc tính: một là đó phải là một ứng dụng internet. Hai là, nó phải lấy được thông tin của vật chủ.

1.3.2. Ứng dụng

Với những hiệu quả thông minh rất thiết thực mà IoT đem đến cho con người, IoT đã và đang được tích hợp trên khắp mọi thứ, mọi nơi xung quanh thế giới mà con người đang sống. Từ chiếc vòng đeo tay, những đồ gia dụng trong nhà, những mảnh vườn đang ươm hạt giống, cho đến những sinh vật sống như động vật hay con người...đều có sử dụng giải pháp IoT. Hình 2.9: Tổng quan về ứng dụng của IoT.

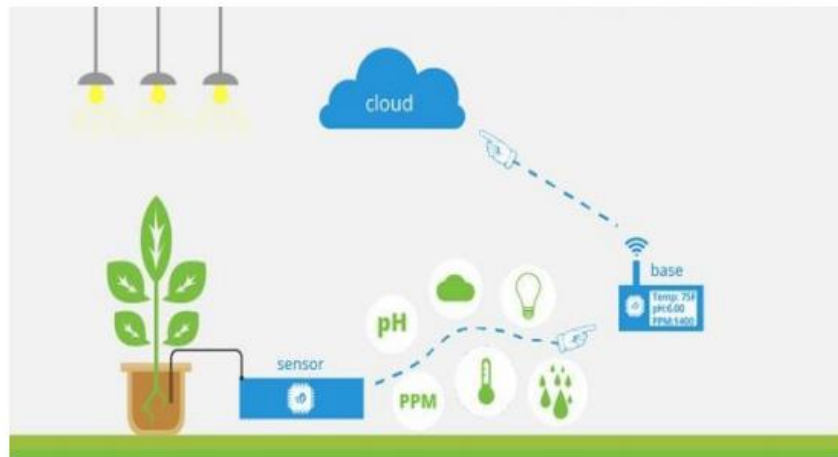
Ứng dụng trong lĩnh vực vận tải: Ứng dụng điển hình nhất trong lĩnh vực này là gắn chip lấy tọa độ GPS lên xe chở hàng, nhằm kiểm soát lộ trình, tốc độ, thời gian đi đến của các xe chở hàng. Ứng dụng này giúp quản lý tốt khâu vận chuyển, có những xử lý kịp thời khi xe đi không đúng lộ trình hoạt

động bị hỏng hóc trên những lộ trình, mà ở đó mạng di động không phủ sóng tới được kiểm soát được lượng nhiên liệu tiêu hao ứng với lộ trình đã được vạch trước...



Hình 1.6. Theo dõi lộ trình đi của xe chở hàng

Ứng dụng trong lĩnh vực sản xuất nông nghiệp: Quá trình sinh trưởng và phát triển của cây trồng trải qua nhiều giai đoạn từ hạt nảy mầm đến ra hoa kết trái. Ở mỗi giai đoạn cần có sự chăm sóc khác nhau về chất dinh dưỡng cũng như chế độ tưới tiêu phù hợp. Những yêu cầu này đòi hỏi sự bền bỉ và siêng năng của người nông dân từ ngày này sang ngày khác làm cho họ phải vất vả. Nhưng nhờ vào ứng dụng khoa học kỹ thuật, sử dụng cảm biến để lấy thông số nhiệt độ, độ ẩm, độ pH của đất trồng, cùng với bảng dữ liệu về quy trình sinh trưởng của loại cây đó, hệ thống sẽ tự động tưới tiêu bón lót cho cây trồng phù hợp với từng giai đoạn phát triển của cây trồng. Người nông dân bây giờ chỉ kiểm tra, quan sát sự vận hành của hệ thống chăm sóc cây trồng trên một màn hình máy tính có kết nối mạng.



Hình 1.7. Theo dõi tình trạng sinh trưởng của cây trồng

Ứng dụng trong nhà thông minh: Vài năm trở lại đây, khi thế giới đang dần tiến vào kỷ nguyên Internet of Things, kết nối mọi vật qua Internet, nhà thông minh trở thành một xu hướng công nghệ tất yếu là tiêu chuẩn của nhà ở hiện đại. Trong căn hộ thông minh tất cả các thiết bị từ rèm cửa, điều hoà, dàn âm thanh, hệ thống ánh sáng, hệ thống an ninh, thiết bị nhà tắm... được kết nối với nhau và hoạt động hoàn toàn tự động theo kịch bản lập trình sẵn đáp ứng đúng nhu cầu sử dụng của khách hàng. Ví dụ: vào buổi sáng, đèn tắt, rèm cửa tự động chuyển tới vị trí thích hợp để giảm bớt những tác động nào nhiệt từ đường phố và nhường không gian cho ánh sáng tự nhiên. Tối đến, hệ thống đèn bật sáng, các rèm cửa kéo lên người dùng có thể thưởng ngoạn từ trên cao bức tranh thành phố rực rỡ ánh đèn, đồng thời âm nhạc cũng nhẹ nhàng cất lên các giai điệu yêu thích của gia đình.

1.4. Tìm hiểu về ESP8266 và Kit Wifi ESP8266 NodeMCU Lua V3 CH340 [3]

1.4.1. Khái niệm chung về ESP8266

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, được sản xuất bởi bên thứ 3: AI-Thinker. 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.

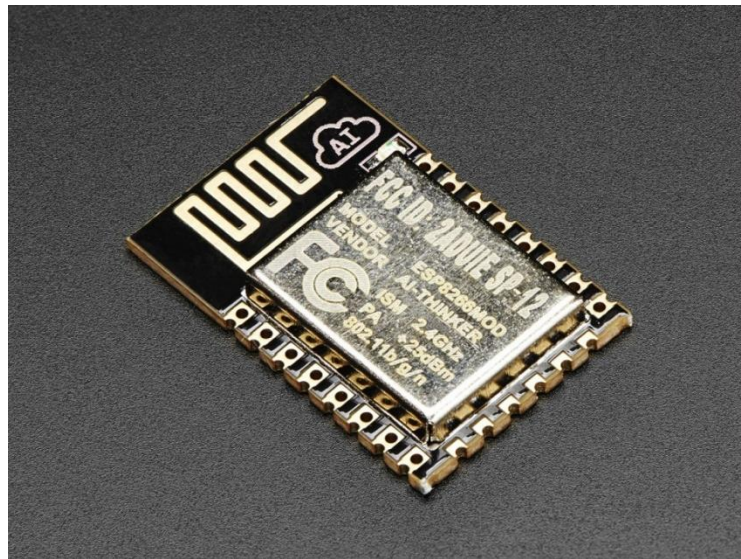
ESP8266 có một cộng đồng các nhà phát triển trên thế giới rất lớn, cung cấp nhiều Module lập trình mã nguồn mở giúp nhiều người có thể tiếp cận và xây dựng ứng dụng rất nhanh.

1.4.2. Phần cứng của ESP8266

Cấu trúc phần cứng của dòng chip sử dụng ESP8266 có thể tóm tắt như sau:

- Sử dụng 32-bit MCU core có tên là Tensilica .
- Tốc độ xung hệ thống có thể thiết lập ở 80MHz hoặc 160MHz Không tích hợp bộ nhớ Flash để lưu chương trình .
- Tích hợp 50KB RAM để lưu dữ liệu ứng dụng khi chạy .
- Có đầy đủ các ngoại vi chuẩn để giao tiếp như 17 GPIO, 1 Slave SDIO, 3 SPI, 1 I2C, 1 I2S, 2 UART, 2 PWM.
- Tích hợp các mạch RF để truyền nhận dữ liệu ở tần số 2.4GHz.

Do không hỗ trợ bộ nhớ Flash nên các board sử dụng ESP8266 phải gắn thêm chip Flash bên ngoài và thường là Flash SPI để ESP8266 có thể đọc chương trình ứng dụng với chuẩn SDIO hoặc SPI.

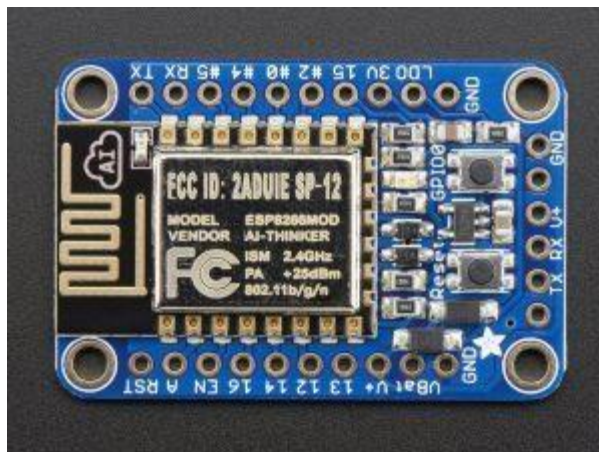


Hình 1.8. Modul ESP8266-12E[3]

1.4.3. Phân loại ESP8266

Hiện nay có rất nhiều Board ESP8266 trên thị trường. Dưới đây là một số Board thông dụng nhất trong thời điểm hiện tại

a. Adafruit HUZZAH ESP8266 (ESP-12)



Hình 1.9. Adafruit HUZZAH ESP8266 (ESP-12)[3]

Một số đặc điểm của Board:

- 1x Reset button.
- 1 x User button khi khởi động module, nhấn phím này sẽ đưa chip về bootloading mode để upload firmware.

- 1xRed LED có thể lập trình được (nhấp nháy khi có tín hiệu chẳng hạn...).
- Tích hợp chuyển đổi điện áp tín hiệu từ 5V sang 3.3V cho UART và chân Reset, lý do là esp8266 chỉ chạy ở điện áp 3.3V.
- Tích hợp IC ổn áp 3.3V, 500mA.
- 2 diot bảo vệ chống cấp ngược nguồn.
- 1 x Analog input (1.8V max).
- 9 x GPIO (3.3V logic), có thể sử dụng các giao tiếp I2C hoặc SPI.
- 2 x UART pins.
- 2 x 3-6V power inputs, reset, enable, LDO-disable, 3.3V output.

b. NodeMCU 1.0



Hình 1.10. NodeMCU 1.0[3]

Một số thông số của Board:

- Sử dụng chip USB-Serial CP2102.
- Lõi chip ESP-12E: xung nhịp 80MHz, Flash 8MB.
- Blue LED cho GPIO16.
- IC ổn áp 3.3V NCP1117.

- 2 button FLASH và RESET.

1.4.4. Kit Wifi ESP8266 NodeMCU Lua V3 CH340



Hình 1.11. Node MCU V3 CH340 – kit RF thu phát wifi ESP8266[3]

Kit RF thu phát Wifi ESP8266 NodeMCU Lua V3 CH340 là phiên bản NodeMCU sử dụng IC nạp giá rẻ CH340 từ Lolin với bộ xử lý trung tâm là module Wifi SoC ESP8266, kit có thiết kế dễ sử dụng và đặc biệt là có thể sử dụng trực tiếp trình biên dịch của Arduino để lập trình và nạp code. Điều này khiến việc sử dụng và lập trình các ứng dụng trên ESP8266 trở nên rất đơn giản.

Kit RF thu phát Wifi ESP8266 NodeMCU Lua V3 CH340 được dùng cho các ứng dụng cần kết nối, thu thập dữ liệu và điều khiển qua sóng Wifi, đặc biệt là các ứng dụng IoT.

Thông số kỹ thuật:

- IC chính: ESP8266 Wifi SoC.
- Phiên bản firmware: NodeMCU Lua.
- Chip nạp và giao tiếp UART: CH340.
- GPIO tương thích hoàn toàn với firmware Node MCU.
- Cấp nguồn: 5VDC MicroUSB hoặc Vin.
- GPIO 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: 59 x 32mm.

1.5. Tìm hiểu về Google Assistant[4]

1.5.1. Khái niệm Google Assistant

Google Assistant là trợ lý ảo của Google. Đây là sự phát triển của tính năng Android có tên gọi Google Now, cung cấp cho bạn thông tin mà không cần bạn phải hỏi. Google Assistant là đối thủ cạnh tranh với Siri của Apple, Alexa của Amazon và Cortana của Microsoft.

Google Assistant ra mắt vào tháng 5 năm 2016, như một phần của ứng dụng nhắn tin thông minh Allo của Google. Ngay sau đó, Google Assistant đã chuyển sang điện thoại Pixel của Google. Tính năng này chỉ có sẵn trên dòng Pixel trong một thời gian. Nhưng giờ đây, Google Assistant có thể được tải xuống miễn phí trên bất kỳ thiết bị nào chạy Android 5 Lollipop trở lên.

Ngoài hỗ trợ Android Wear, còn có một ứng dụng Assistant cho IOS. Và dòng loa thông minh Google Home cũng sở hữu Google Assistant.

1.5.2. Hoạt động Google Assistant

Google Assistant cung cấp lệnh thoại, tìm kiếm bằng giọng nói và kích hoạt thiết bị bằng giọng nói, cho phép bạn hoàn thành một số tác vụ sau khi bạn nói từ “OK Google” hoặc “Hey Google”.

Trợ lý Google này có thể:

- Kiểm soát các thiết bị thông minh
- Truy cập thông tin từ lịch và thông tin cá nhân khác

- Tìm kiếm thông tin online như đặt chỗ nhà hàng, chỉ đường, thời tiết và tin tức
- Kiểm soát playlist âm nhạc của bạn
- Phát nội dung trên Chromecast hoặc các thiết bị tương thích khác
- Chạy bộ hẹn giờ và lời nhắc
- Lên lịch hẹn và gửi tin nhắn
- Mở ứng dụng trên điện thoại
- Đọc thông báo cho bạn
- Chơi game

Continued Conversation có nghĩa là bạn không cần phải nói “Hey Google” cho các yêu cầu tiếp theo. Thay vào đó, khi bạn bắt đầu nói chuyện với Google. Google Assistant sẽ lắng nghe và phản hồi mà không cần đến cụm từ kích hoạt. Google cũng có thể nhận dạng giọng nói của những người khác nhau, vì vậy công cụ biết ai đang nói chuyện với mình và có thể điều chỉnh phản hồi cho phù hợp. Ngoài ra, bạn cũng có thể yêu cầu nhiều việc cùng một lúc.

Khi Google Assistant biết bạn và hiểu ngữ cảnh, nó sẽ phản ứng theo cách dò hỏi hoặc lịch sự. Điều đó quan trọng vì như vậy tính năng điều khiển bằng giọng nói sẽ có nhiều quyền lực hơn để công cụ không đơn giản là chỉ phản ứng với các cụm từ hoặc lệnh cụ thể.

Các tính năng đăng ký chuyến bay (tùy thuộc vào hãng hàng không và điểm đến), cũng như đặt phòng (với một số đối tác) và có cả chế độ thông dịch viên (Interpreter Mode) trên Google Home và màn hình thông minh. Với Interpreter Mode, bạn có thể yêu cầu Google Assistant giúp bạn thực hiện một cuộc trò chuyện với nhiều ngôn ngữ khác nhau. Chỉ cần nói “Hey Google, be my Spanish interpreter” để bắt đầu Interpreter Mode và được đáp lại bằng giọng

ở thực tế và bản dịch bằng văn bản để hỗ trợ cuộc trò chuyện trên màn hình điện thoại.

Google Assistant trong các thiết bị Google Home tạo thành nền tảng của việc điều khiển các ngôi nhà thông minh. Nó tương thích với nhiều loại thiết bị, vì vậy bạn có thể điều khiển hệ thống sưởi, đèn và nhiều thiết bị khác bằng giọng nói của mình.

1.5.3. Server IFTTT

IFTTT có khả năng tự động hóa các thao tác trên mạng một cách nhanh chóng, nghe thật là mới lạ đúng không nào. Dịch vụ này đã xuất hiện được một thời gian nhưng chắc chắn với nhiều người không đi sâu tìm hiểu thì đây là một hình thức mới lần đầu nghe đến.

IFTTT là một dịch vụ web trung gian đứng giữa hai dịch vụ để thực hiện tác vụ khi có điều kiện xảy ra, IFTTT là viết tắt của If This Then That. Trong đó, If This (nếu việc này xảy ra) sẽ dẫn đến Then That (thì làm việc kia) đây được xem là nguyên lý hoạt động của câu lệnh. Tức khi có bất kỳ 1 sự thay đổi nào trên ứng dụng này, thì thông qua IFTTT ứng dụng kia cũng sẽ hoạt động và thay đổi tương tự.

1.6. Ứng dụng Blynk theo dõi trạng thái các thiết bị[5]

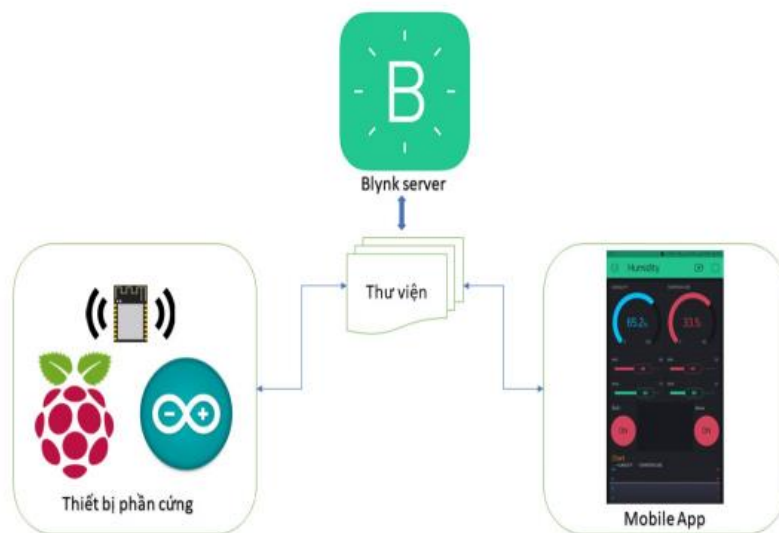
Blynk Đây là một nền tảng IoT được ưa thích bởi hơn 500.000 kỹ sư trong lĩnh vực IoT trên toàn thế giới.

1.6.1. Hệ sinh thái Blynk

Có ba thành phần chính trong nền tảng Blynk: Blynk App - cho phép tạo giao diện cho sản phẩm của bạn bằng cách kéo thả các widget khác nhau mà nhà cung cấp đã thiết kế sẵn. Blynk Server - chịu trách nhiệm xử lý dữ liệu trung tâm giữa điện thoại, máy tính bảng và phần cứng. Chúng ta có thể sử dụng Blynk Cloud của Blynk cung cấp hoặc tự tạo máy chủ Blynk riêng cho mình. Vì đây là mã nguồn mở, nên bạn có thể dễ dàng integrate vào các thiết

bị và thậm chí có thể sử dụng Raspberry Pi làm server của hệ thống. Library Blynk – support cho hầu hết tất cả các nền tảng phần cứng phổ biến - cho phép giao tiếp với máy chủ và xử lý tất cả các lệnh đến và đi.

Nguyên lý hoạt động của Blynk: mỗi khi ta nhấn một nút trong ứng dụng Blynk, yêu cầu sẽ chuyển đến server của Blynk, server sẽ kết nối đến phần cứng của chúng ta thông qua library . Tương tự thiết bị phần cứng sẽ truyền dữ liệu ngược lại đến server. Vì thế chúng ta có thể tự mình xây dựng một hệ sinh thái nhà thông minh dựa trên nền tảng của Blynk.



Hình 1.12.Sơ đồ hệ sinh thái Blynk

1.6.2. Tính năng của Blynk

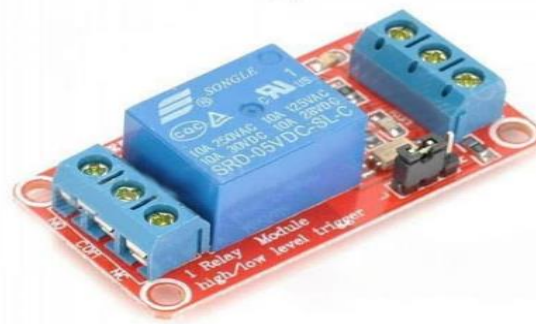
- Cung cấp API & giao diện người dùng tương tự cho tất cả các thiết bị và phần cứng được hỗ trợ
- Kết nối với server bằng cách sử dụng: Wifi, Bluetooth và BLE, Ethernet, USB (Serial), GSM, ...
- Các tiện ích trên giao diện được nhà cung cấp dễ sử dụng
- Thao tác kéo thả trực tiếp giao diện mà không cần viết mã

- Dễ dàng tích hợp và thêm chức năng mới bằng cách sử dụng các cổng kết nối ảo được tích hợp trên blynk app
- Theo dõi lịch sử dữ liệu
- Thông tin liên lạc từ thiết bị đến thiết bị bằng Widget

1.7. Một số linh kiện và thiết bị khác được sử dụng trong mạch[6]

1.7.1. Module Relay

Relay là thiết bị đóng cắt cơ bản, nó được sử dụng rất nhiều trong cuộc sống và trong các thiết bị điện tử.



Hình 1.13.Relay[6]

Cấu trúc cơ bản của relay (rơ – le) sẽ bao gồm một cuộn dây kim loại đồng hoặc nhôm được quấn quanh một lõi sắt từ. Bộ phận này có phần tĩnh được gọi là ách từ (Yoke) và phần động được gọi là phần cứng (Armature). Phần cứng sẽ được kết nối với một tiếp điểm động, cuộn dây có tác dụng hút thanh tiếp điểm lại để tạo thành trạng thái NO và NC. Mạch tiếp điểm (mạch lực) có nhiệm vụ đóng cắt các thiết bị tải với dòng điện nhỏ và được cách ly bởi cuộn hút.

Nguyên lí hoạt động: Khi có tín hiệu vào, cuộn hút sẽ hút làm các tiếp điểm thường đóng mở ra, các tiếp điểm thường mở đóng lại cấp nguồn cho thiết bị.

Thông số kỹ thuật:

- Điện áp điều khiển: 5V

- Relay tiêu thụ dòng khoảng 10A.
- Thời gian tác động: 10ms
- Thời gian nhả hãm: 5ms
- Nhiệt độ hoạt động: $-45^{\circ}\text{C} \sim 75^{\circ}\text{C}$

1.7.2. Tụ gốm

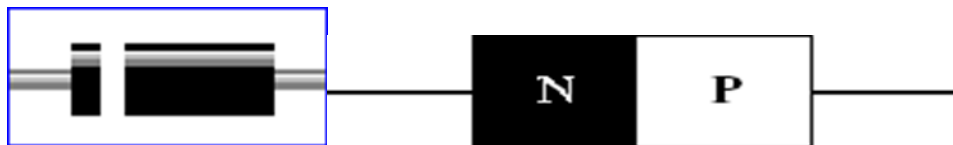
Là loại tụ không có cực và có điện dung nhỏ từ $0,47\mu\text{F}$ trở xuống, các tụ này thường được sử dụng trong các mạch điện có tần số cao hoặc mạch lọc nhiễu.



Hình 1.14. Tụ gốm[6]

1.7.3. Diode

Có 2 lớp bán dẫn P-N. Dòng điện chỉ qua từ Anot sang Katot khi áp trên hai chân phân cực thuận $V_P > V_N$ còn khi phân cực ngược $V_P < V_N$ thì không dẫn điện.



Hình 1.15. Diode[6]

1.7.4. Quạt tản nhiệt



Hình 1.16. Quạt tản nhiệt[6]

Thông số kỹ thuật:

- Điện áp: DC 5V
- Dòng điện định mức: 0.25A
- Công suất tiêu thụ: 1.25W
- Hiệu năng biến đổi: 5000+ 10% cho RPM
- Tiếng ồn: 22dBA (không gây tiếng ồn)
- Lưu lượng không khí: 5CFM

1.7.5. Bóng đèn Led



Hình 1.17. Đèn led usb[6]

Thông số:

- Công suất 1.2W.
- Điện áp 5V.
- Kích thước chiều dài 20cm.

1.7.6. Máy bơm mini 5V

Động cơ bơm chìm Mini 5VDC có kích thước rất nhỏ gọn, sử dụng điện áp 3~5VDC, vì thuộc dạng bơm chìm nên động cơ có khả năng chống nước và hoạt động khi ngâm chìm trong nước.



Hình 1.18. Máy bơm mini DC 5V[6]

Thông số kỹ thuật:

- Điện áp sử dụng: 3~5VDC.
- Dòng điện sử dụng: 100~200mA.
- Lưu lượng bơm: 1.2~1.6L / 1 phút.
- Điện áp: 3VDC – 5VDC.
- Chiều dài bơm chìm: 43.5 mm .
- Đường kính ống ra: 7-8mm .
- Chiều dài ống ra: 17mm .

- Kích thước: 34 x 43 mm.

1.8. Kết luận

Chương 1 đã giới thiệu được tình hình nghiên cứu và các sản phẩm điều khiển bằng giọng nói hiện nay. Lựa chọn, nắm được cấu trúc, chức năng Kit Wifi ESP8266 NodeMCU Lua V3 CH340. Tìm hiểu chức năng và phương thức hoạt động của Google Assistant. Tìm hiểu nguyên lý hoạt động và cách đưa giá trị lên Blynk. Lựa chọn được các linh kiện và thiết bị cần sử dụng. Đây là những kiến thức cơ sở để thiết kế và lập trình cho thiết bị điều khiển bằng giọng nói sử dụng ESP8266, sẽ được trình bày trong chương 2 và chương 3 của đồ án.

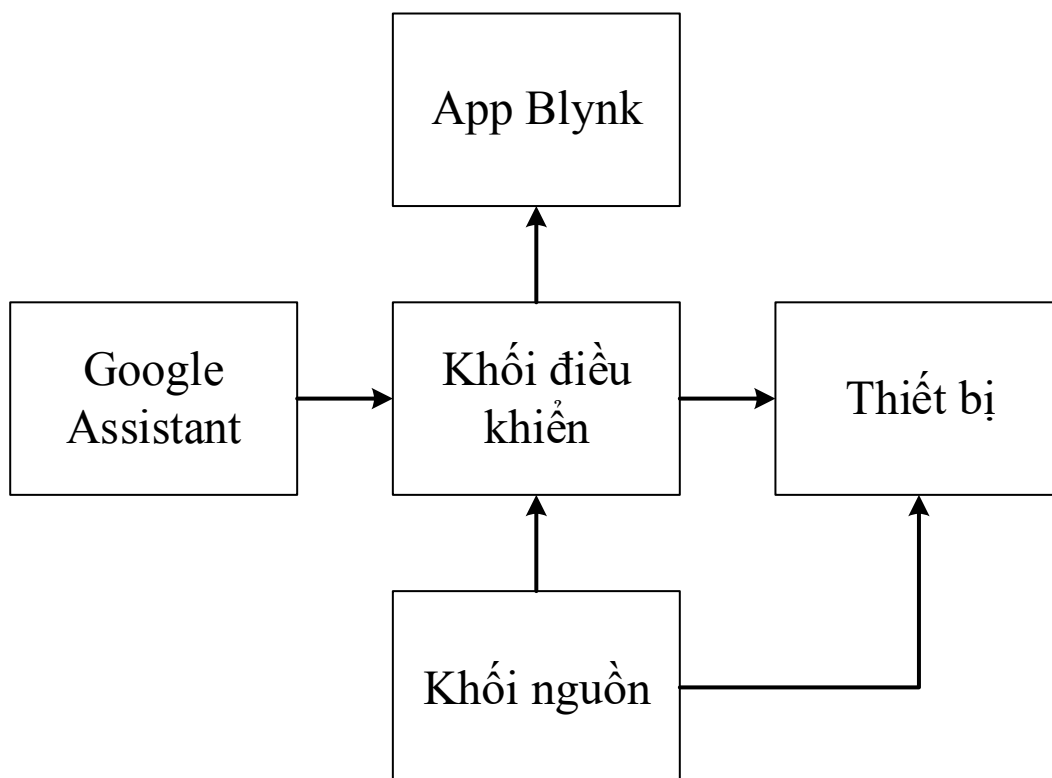
CHƯƠNG 2. THIẾT KẾ HỆ THỐNG ĐIỀU KHIỂN THIẾT BỊ TRONG NHÀ BẰNG GIỌNG NÓI SỬ DỤNG ESP8266

2.1. Thiết kế sơ đồ khối bộ điều khiển thiết bị điện, điện tử gia dụng bằng giọng nói

2.1.1. Yêu cầu

- Mô hình hệ thống hoàn thiện đảm bảo có thể điều khiển bật, tắt các thiết bị sử dụng giọng nói và theo dõi trạng thái trên ứng dụng Blynk.
- Mô hình hoạt động đảm bảo:
 - Hệ thống nhận lệnh điều khiển chính xác.
 - Thiết bị đầu ra nhận và thực hiện bật, tắt nhanh chóng.
 - Ổn định, không gặp sự cố trong quá trình hoạt động.
 - Gọn nhẹ và tiết kiệm chi phí.

2.1.2. Sơ đồ khối



Hình 2.1. Sơ đồ khối hệ thống

Chức năng của các khối:

- Khối nguồn: Cung cấp năng lượng hoạt động cho khối vi điều khiển, khối thiết bị.
- Khối vi điều khiển: ESP8266 là một thiết bị có khả năng nhận và lập trình xử lý được các tín hiệu theo yêu cầu đầu vào của hệ thống, đồng thời xuất được các tín hiệu ra theo yêu cầu.
- Khối Google Assistant: Nhận lệnh bằng giọng nói để đưa vào khối vi điều khiển xử lý.
- Khối thiết bị: Nhận lệnh đầu ra và thực hiện bật, tắt các thiết bị.
- Khối App Blynk: Là App giúp người dùng theo dõi các trạng thái của thiết bị.

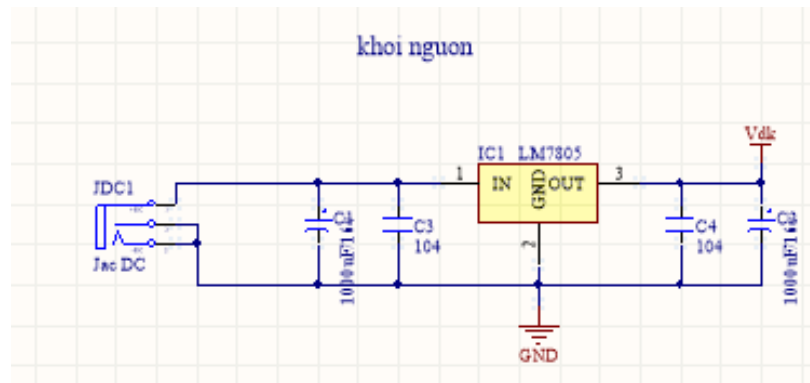
2.2. Thiết kế mạch phần cứng

2.2.1. Khối nguồn

Sử dụng nguồn 9V kết hợp LM7805 để hạ áp từ 9V xuống 5V để cấp cho vi điều khiển và các thiết bị khác.

Tụ C1 và C3 để lọc điện áp đầu vào cấp cho chân Vi của IC 7805, tụ C1 có tác dụng cung cấp điện áp tạm thời cho chân Vi khi nguồn đột ngột bị sụt áp, tụ C3 là tụ gốm nên trở kháng lớn, C3 có tác dụng ngăn nguồn đầu vào tăng áp đột ngột làm dạng sóng điện áp đầu vào có hình răng cưa.

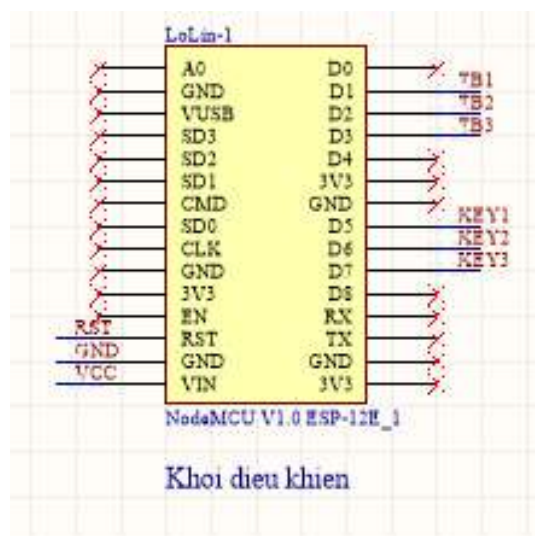
Tụ C2 và C4 để lọc điện áp cấp cho tải tiêu thụ lấy từ chân Vo của IC 7805, tụ C2 có tác dụng cung cấp điện áp tạm thời cho tải khi điện áp tải đột ngột bị sụt áp, tụ C4 trở kháng lớn, C4 có tác dụng lọc nhiễu điện áp đầu ra



Hình 2.2. Khởi mạch nguồn

2.2.2. Khối vi điều khiển

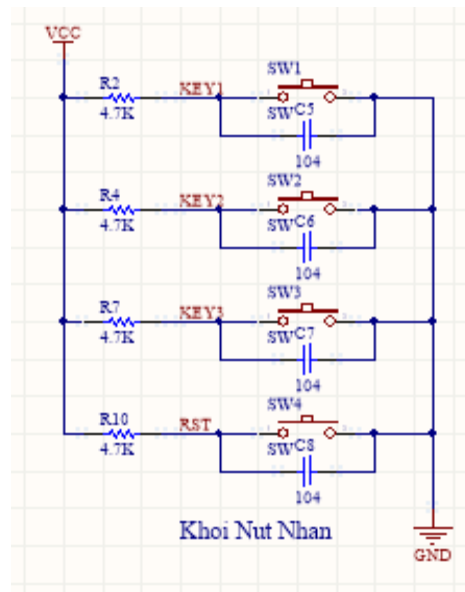
Vi điều khiển ESP điều khiển hoạt động của toàn mạch. Vi điều khiển nhận dữ liệu từ đầu vào rồi xử lý, thông báo ra điều khiển thiết bị.



Hình 2.3. Mạch vi điều khiển

2.2.3. Khối nút nhấn

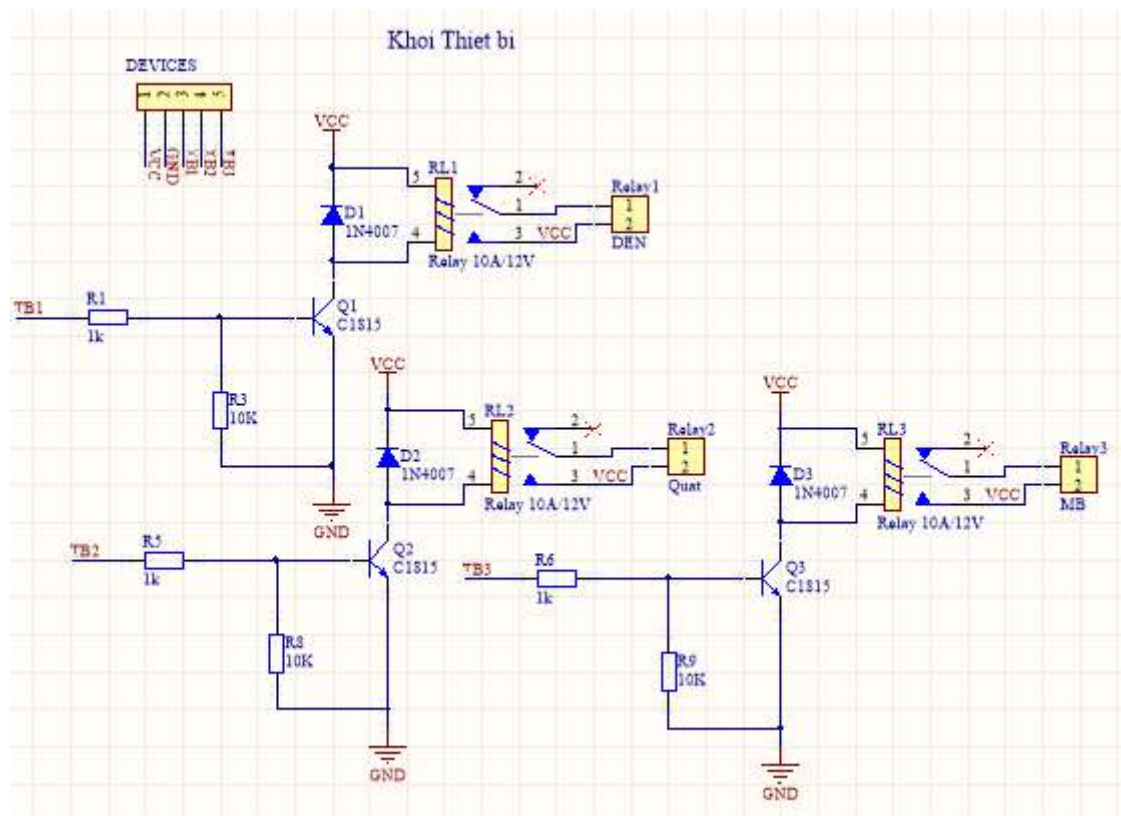
Khối bao gồm 4 nút nhấn: nút reset(SW1), nút điều khiển các thiết bị(SW2,SW3,SW4).



Hình 2.4. Mạch khởi nút nhấn

2.2.4. Khởi thiết bị

Các thiết bị gồm đèn, quạt, máy bơm được điều khiển bởi các relay 5V, khi có tín hiệu ra, relay đóng các tiếp điểm thường mở và cấp điện cho thiết bị.



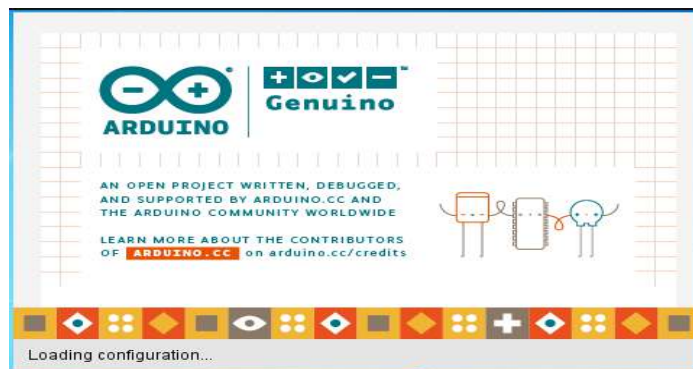
Hình 2.5. Mạch khởi thiết bị

2.3. Xây dựng phần mềm điều khiển

2.3.1. Phần mềm lập trình

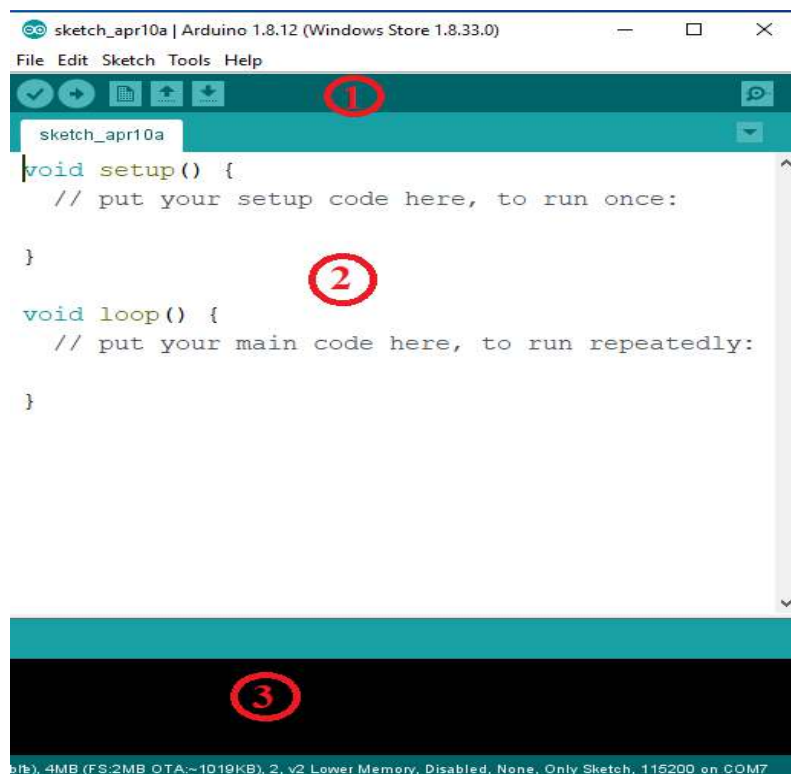
Sử dụng phần mềm Arduino IDE để lập trình cho ESP8266.

Arduino IDE là môi trường để lập trình và nạp code cho các dòng Arduino. Arduino IDE được xây dựng trên nền tảng Java nên hỗ trợ hầu hết các hệ điều hành hiện nay.









Hình 2.6. Phần mềm Arduino IDE

Giao diện của Arduino IDE được chia thành 3 vùng chính:



Hình 2.7. Giao diện của Arduino IDE

+ Vùng 1: Các phím chức năng

	Verify	Kiểm tra lỗi và biên dịch code
	Upload	Dịch và upload code vào bo mạch đã được cài đặt sẵn
	New	Tạo sketch mới
	Open	Mở một sketch có sẵn
	Save	Lưu sketch
	Serial Monitor	Mở serial monitor.

Hình 2.8. Các phím chức năng của Arduino IDE

+ Vùng 2: Cửa sổ để viết chương trình

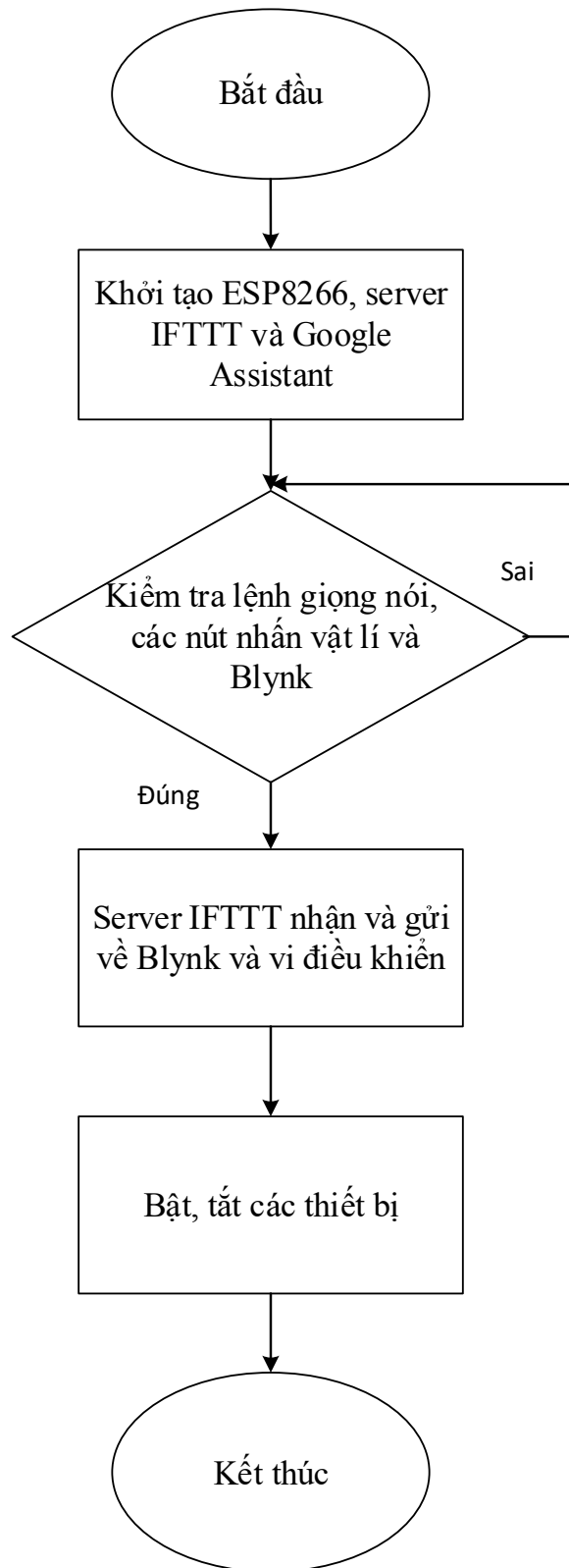
- Chương trình Code sẽ được viết tại đây. Ở đây có hai hàm quan trọng là *setup()* và *loop()*.
- Hàm *setup()* được khởi chạy một lần duy nhất. Chức năng của hàm này dùng để khởi tạo các biến, khai báo chức năng các chân, khởi tạo các thông số ban đầu
- Hàm *loop()* là nơi chương trình được chạy lặp đi lặp lại đến khi ngắt vi điều khiển.
- Các dấu “//” dùng để tạo chú thích, giúp cho việc đọc code được dễ dàng hơn
- Trong lập trình có phân biệt ký tự hoa, thường, tuyệt đối phải đánh chính xác, đồng thời cuối mỗi câu lệnh cần phải có dấu chấm phẩy (;), trừ lệnh khai báo thư viện.

+ Vùng 3: Hiện thị các thông tin liên quan đến chương trình

Là cửa sổ để hiện thị về việc build chương trình, nạp chương trình thành công xuống vi điều khiển và các cảnh báo khác liên quan đến chương trình và điều khiển của chúng ta. Lưu ý, mọi thông báo và trạng thái của cả quá trình

viết chương trình (write code), xây dựng chương trình (build code) và nạp chương trình (program code) đều được hiển thị tại đây. Cửa sổ này được gọi là cửa sổ debug.

2.3.2. Lưu đồ thuật toán



Hình 2.9. Lưu đồ thuật toán

Giải thích lưu đồ:

Bắt đầu hoạt động, hệ thống khởi động ESP8266, server IFTTT và Google Assistant. Sau đây kiểm tra lệnh điều khiển giọng nói, điều khiển bằng nút nhấn hoặc Blynk nếu có lệnh điều khiển sẽ thực hiện bật, tắt thiết bị theo yêu cầu, nếu sai quay về kiểm tra các lệnh, kết thúc chương trình.

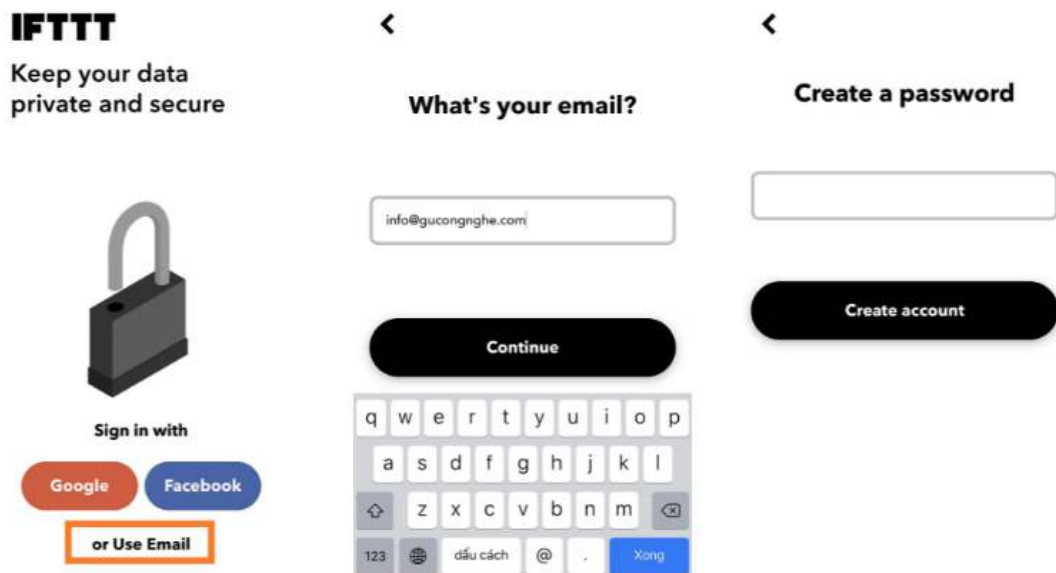
2.3.3. Triển khai phần mềm

2.3.3.1. Thiết lập IFTTT

Tạo tài khoản

Đầu tiên, chúng ta sẽ cần tạo một tài khoản để bắt đầu sử dụng IFTTT.

- Đăng ký một tài khoản mới bằng cách nhấn or Use Email (có thể dùng luôn tài khoản Google, Facebook).
- Điền địa chỉ email rồi nhấn Continue.
- Chọn mật khẩu, nhấn Create account.

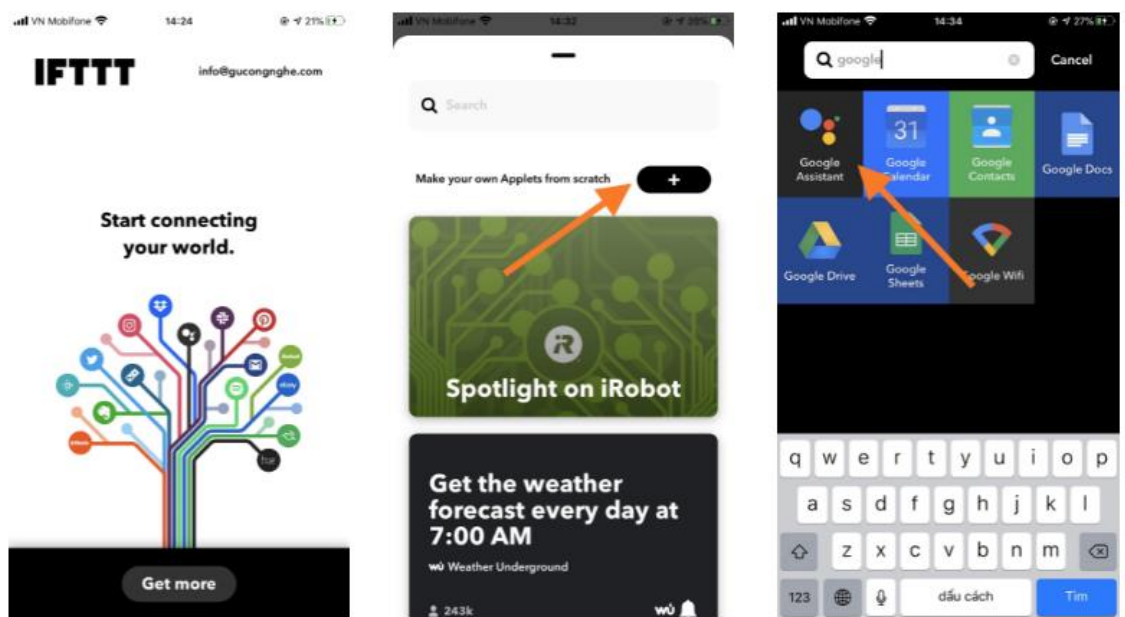


Hình 2.10. Các bước tạo tài khoản IFTTT

Tạo câu lệnh điều kiện This

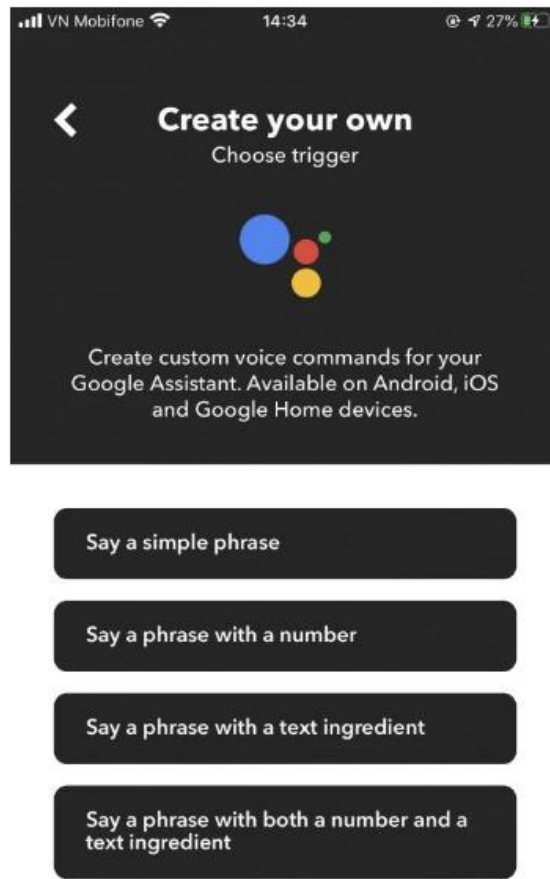
Sau khi tạo thành công tài khoản, một popup chào mừng xuất hiện có thể xem qua một chút hoặc tắt nó đi để bắt tay vào tạo applet.

- Chọn **Get more**.
- Nhấn vào box bên phải dòng chữ **Make your own Applets from scratch**.
- Nhấn + **This**.
- Tìm và nhấn chọn Google Assistant rồi cho phép IFTTT kết nối với trợ lý Google.



Hình 2.11. Các bước tạo câu điều kiện This

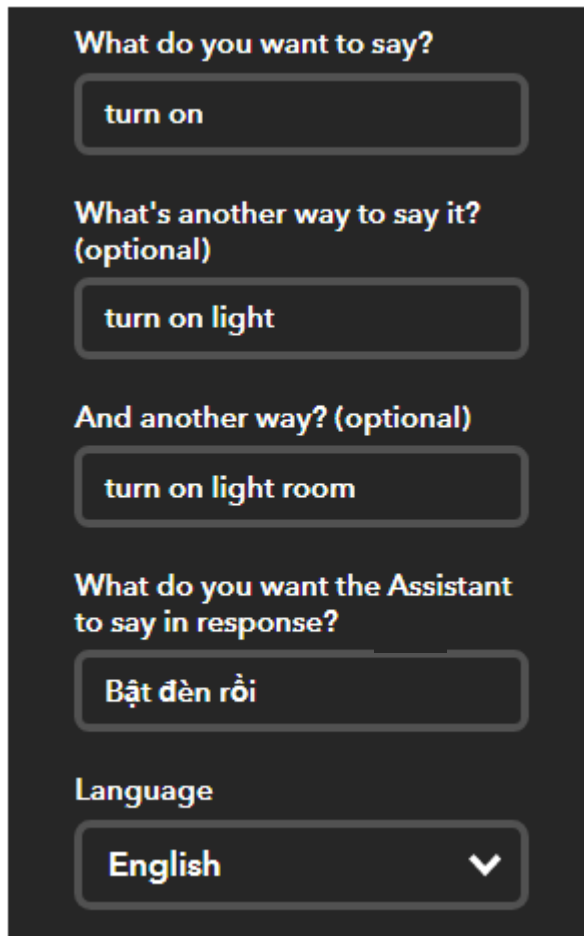
Màn hình sẽ có 4 lựa chọn như hình bên dưới, nhấn vào Say a simple phrase để thử.



Hình 2.12. Lựa chọn phương thức giọng nói

- Nhấn Add
- Đăng nhập tài khoản Google cùng với tài khoản đăng nhập trên Google Home.
- Nhấn Allow để cấp phép

Tiếp đến là phần quan trọng nhất, tùy chỉnh lệnh và phản hồi cho Google Assistant. Bạn được phép lựa chọn 3 câu lệnh chung cho 1 phản hồi của trợ lý. Vì vậy, hãy suy nghĩ và điền thông tin vào ô tương ứng. Các mục optional có thể bỏ qua nếu không muốn dùng tới. Phần ngôn ngữ, vẫn lựa chọn English.



What do you want to say?

turn on

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

turn on light

And another way? (optional)

turn on light room

What do you want the Assistant to say in response?

Bật đèn rồi

Language

English

Hình 2.13. Cài đặt lệnh bằng giọng nói

Sau khi hoàn tất, nhấn chọn Create trigger.

Lựa chọn phản hồi That



Hình 2.14. Phản hồi That

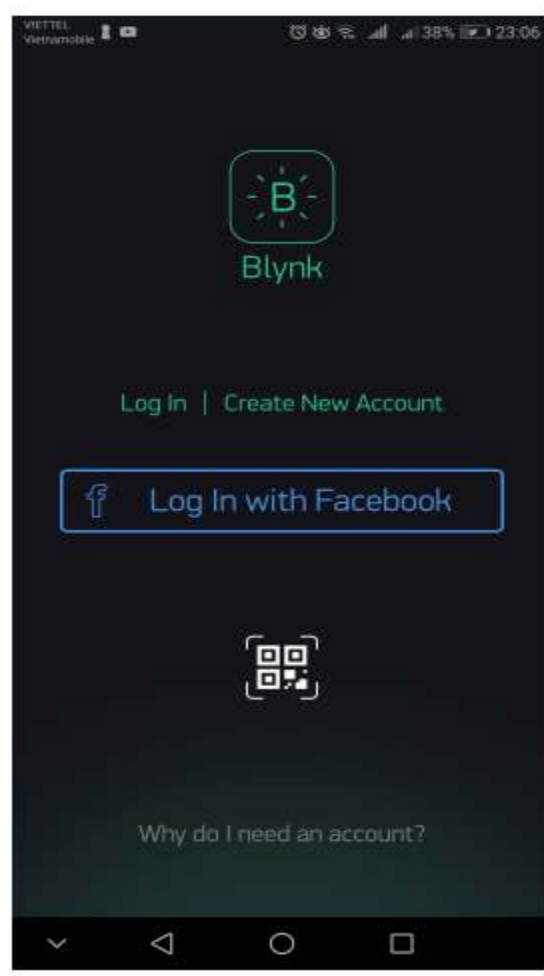
Lúc này ta sẽ bắt đầu cài đặt hành động That.

- Bấm vào dấu “+”.
- Nhập token được gửi về gmail khi đăng kí Blynk để dẫn tới server Blynk đang kết nối với ESP8266.
- Cấu hình cho lệnh từ This, Turn on là giá trị 0, Turn off là giá trị 1.
- Nhấn Finish để hoàn thành.

2.3.3.2. Tạo tài khoản và cài đặt App Blynk

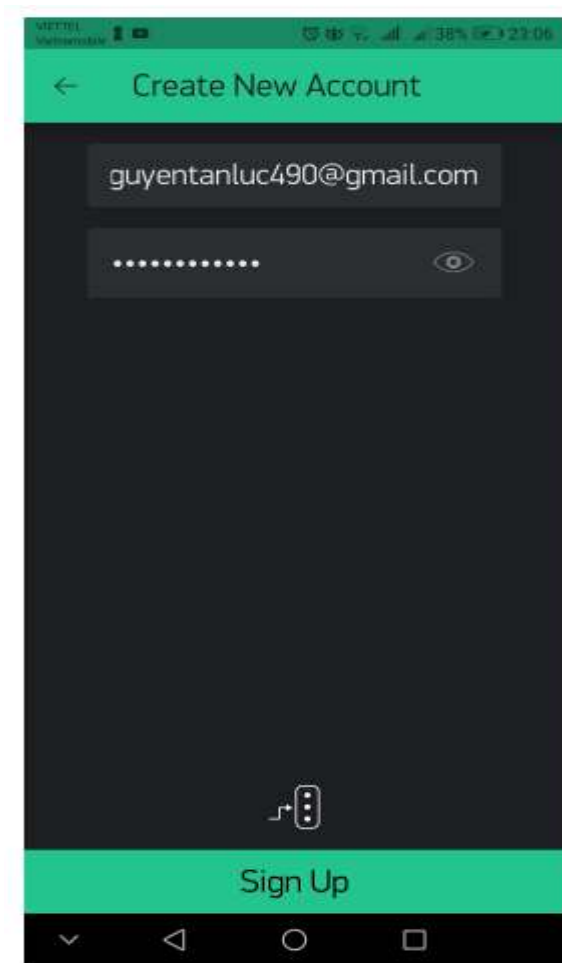
Tạo tài khoản Blynk

Để có thể sử dụng App Blynk chúng ta cần phải đăng ký một tài khoản mới. Ở đây có 2 cách đăng ký bằng Facebook hoặc tạo một tài khoản mới (Create New Account). Ở đây em thực hiện đăng ký tài khoản mới.



Hình 2.15. Giao diện tạo tài khoản Blynk

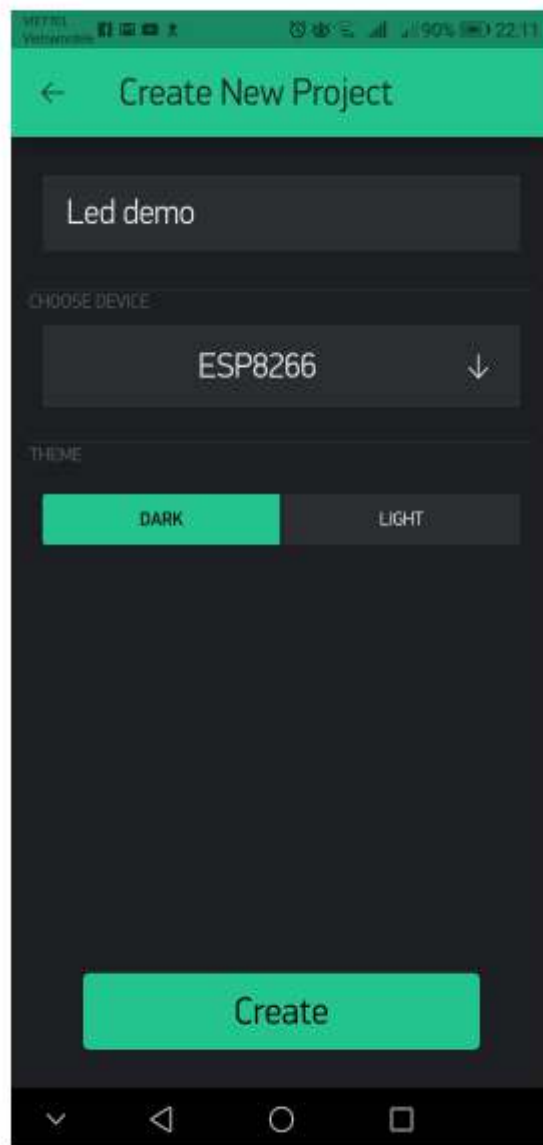
Nhập thông tin tài khoản Gmail sau đây nhấn Sign Up để tiến hành đăng ký.



Hình 2.16. Nhập tài khoản gmail và mật khẩu

Sau khi đăng ký hoàn thành, chúng ta sẽ tiến hành tạo một Project mới.

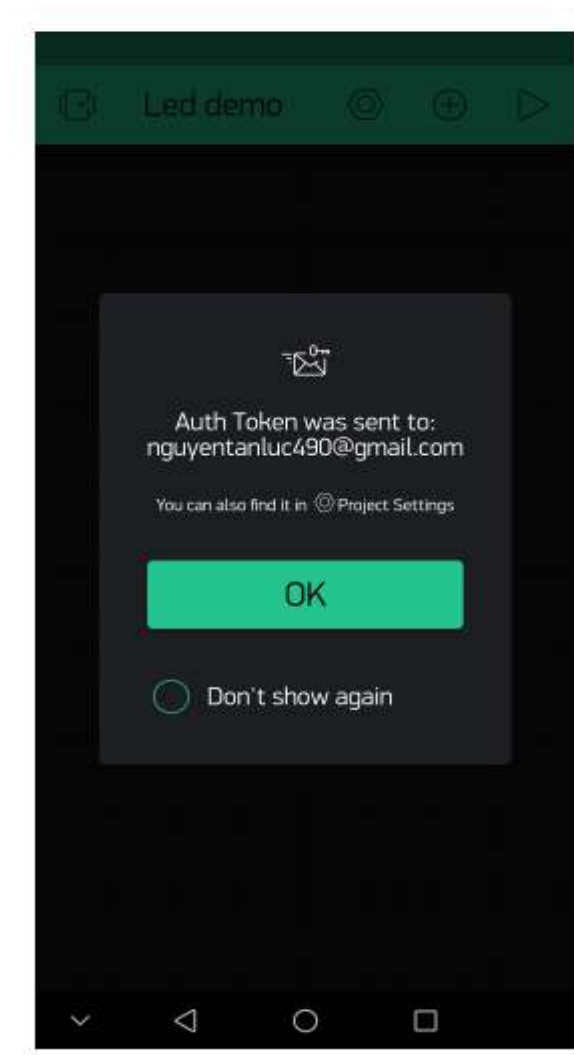
- Click vào dấu cộng (+) trên thanh công cụ để tiến hành tạo Project mới.
- Click vào Create để hoàn tất việc tạo mới.



Hình 2.17. Tạo Project mới

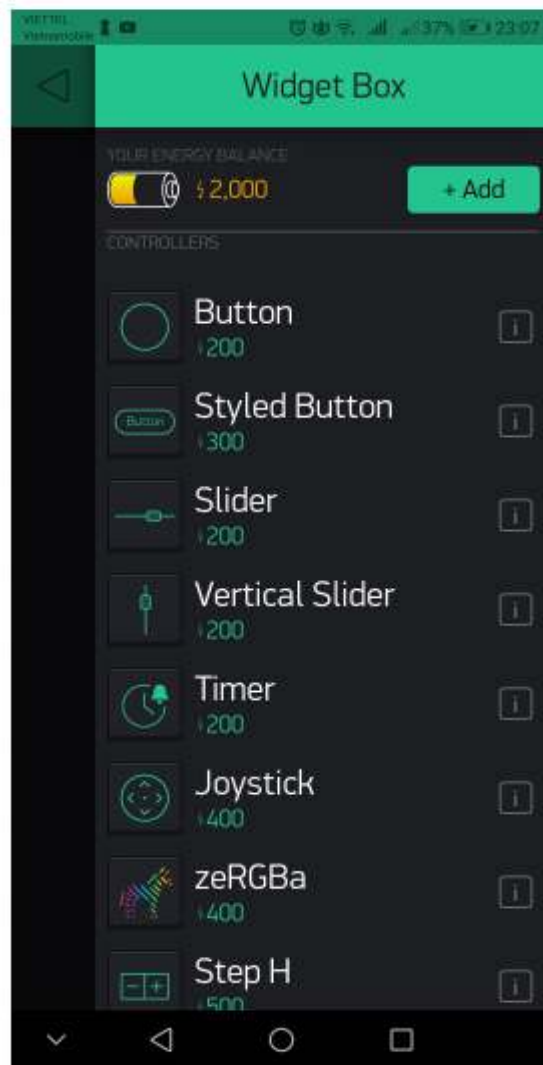
Blynk sẽ cấp một mã Token, mã này sẽ gửi trực tiếp vào Gmail đã đăng ký. Mã Token này dùng để chèn vào code Example của Blynk.

Thiết lập các phím điều khiển



Hình 2.18. Thông báo mã Token được gửi tới gmail

Để Bật/Tắt thiết bị chúng ta cần có một nút bấm để điều khiển. Vào Widget Box (+) > Click vào Button để lấy nút nhấn ra.

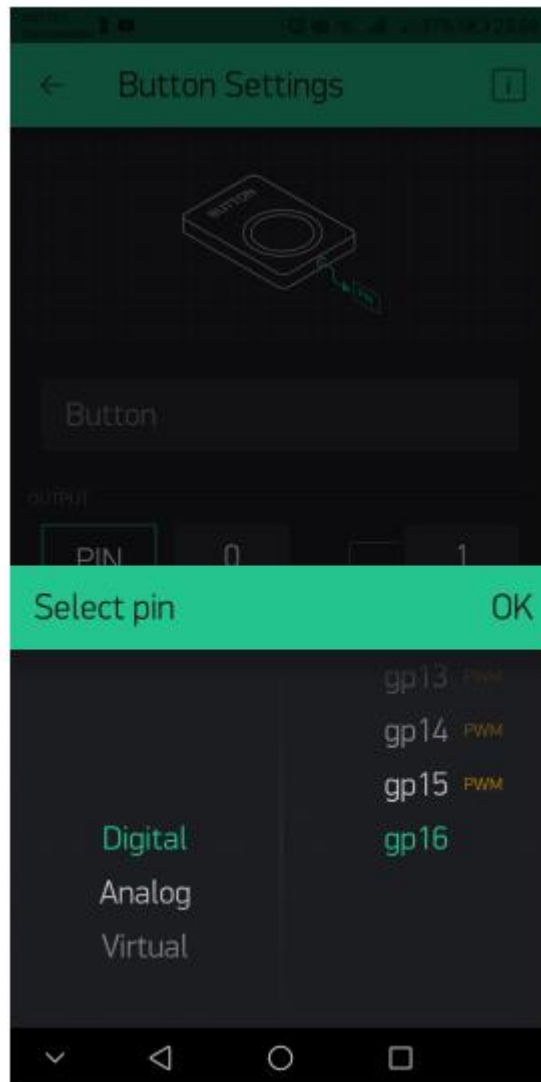


Hình 2.19. Lấy nút nhấn ra

Click vào Button để tiến hành cài đặt thông số.

- Ở mục Button: Đặt tên cho nút nhấn.
- OUTPUT: Cấu hình cho PIN cần kết nối, ở đây mình chọn Digital > gp16 (chân này có LED được tích hợp trên board mạch sẵn).
- MODE: có 2 chế độ PUSH (nhấn thả) và SWITCH (nhấn giữ).
- ON/OFF LABELS: Thay đổi chế độ hiển thị cho nút nhấn (Vd: Các bạn không thích để ON/OFF có thể thay bằng BẬT/TẮT hoặc một cái tên nào mà bạn muốn).

- DESIGN: Ở phần này các bạn có thể điều chỉnh màu sắc của nút nhấn.



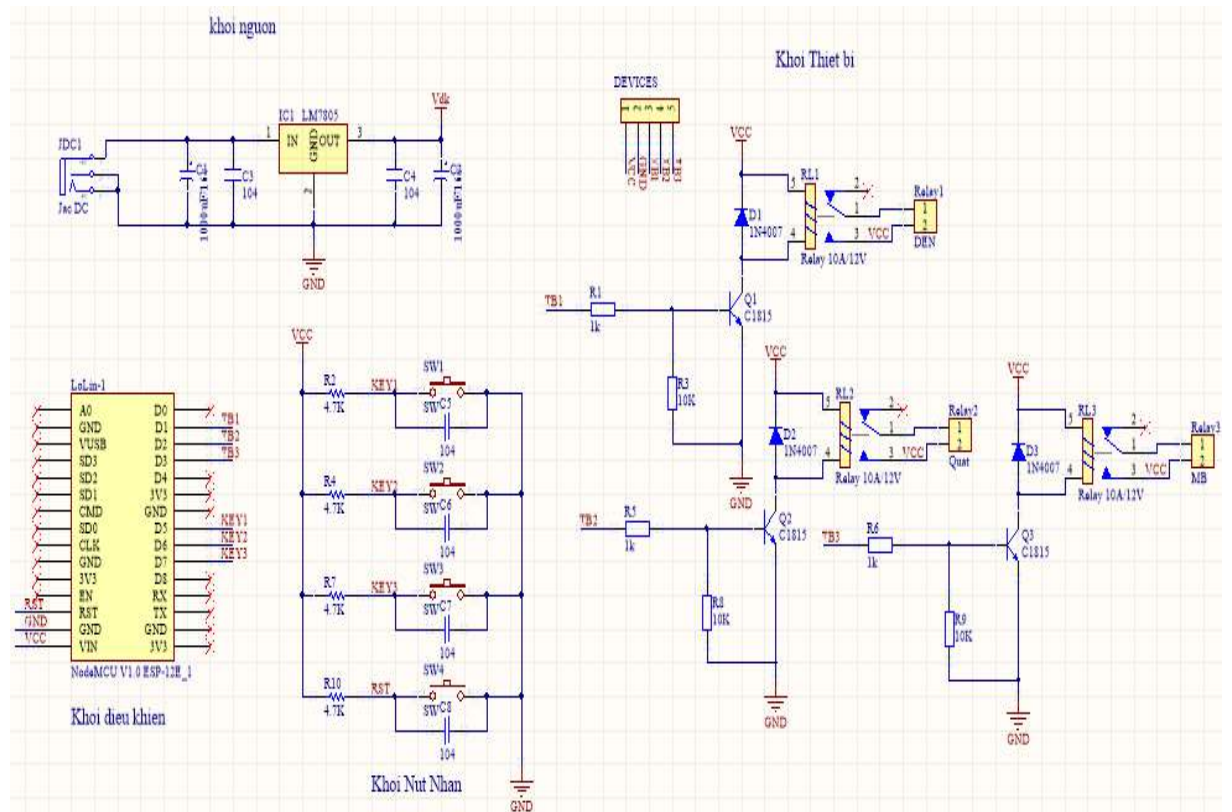
Hình 2.20. Hoàn thành thiết lập

2.4. Kết luận chương 2

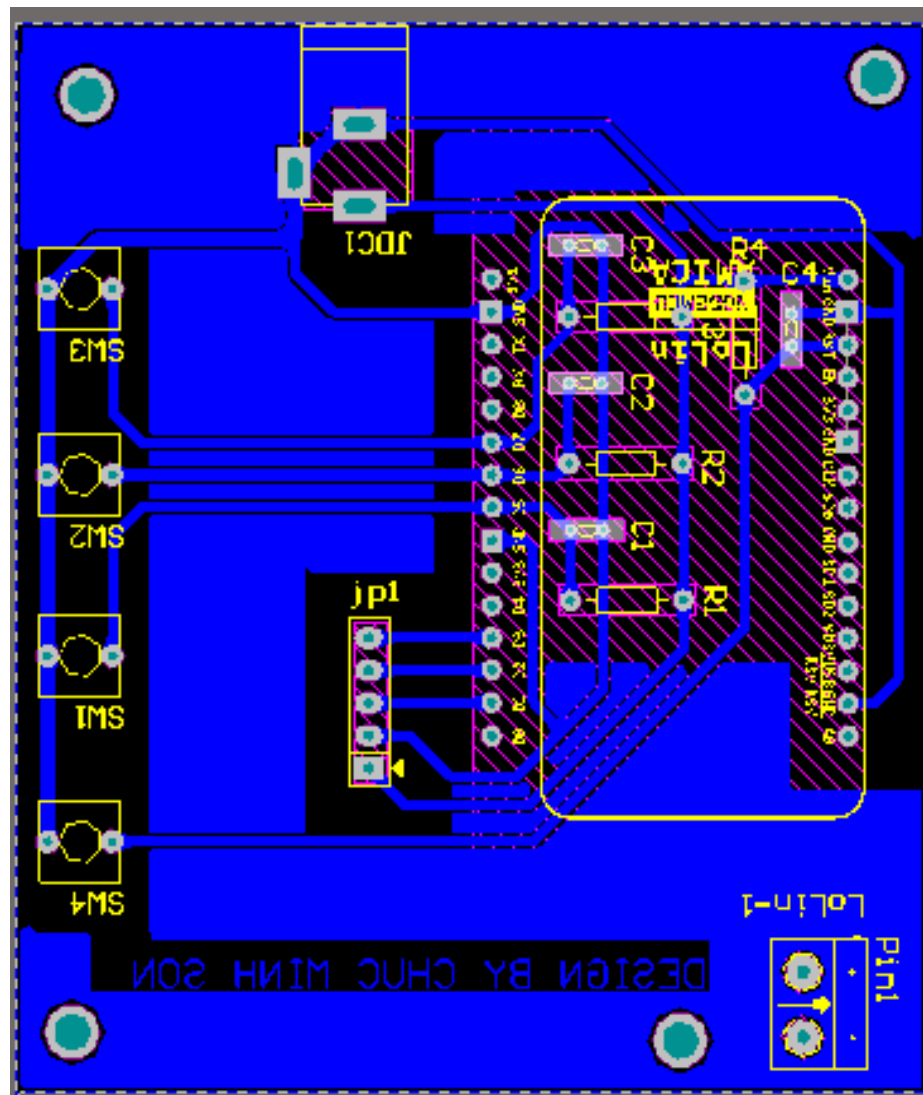
Chương 2 thể hiện toàn bộ quá trình thiết kế mô hình hệ thống điều khiển thiết bị trong nhà bằng giọng nói sử dụng ESP8266. Thực hiện việc thiết kế sơ đồ nguyên lý, vẽ mạch in, sắp xếp linh kiện cũng như viết chương trình điều khiển. Biết xác định lỗi và sửa lỗi cho sản phẩm, để hoàn thành mô hình sẽ được trình bày chi tiết trong chương 3.

CHƯƠNG 3. MÔ HÌNH THỰC NGHIỆM VÀ KẾT QUẢ

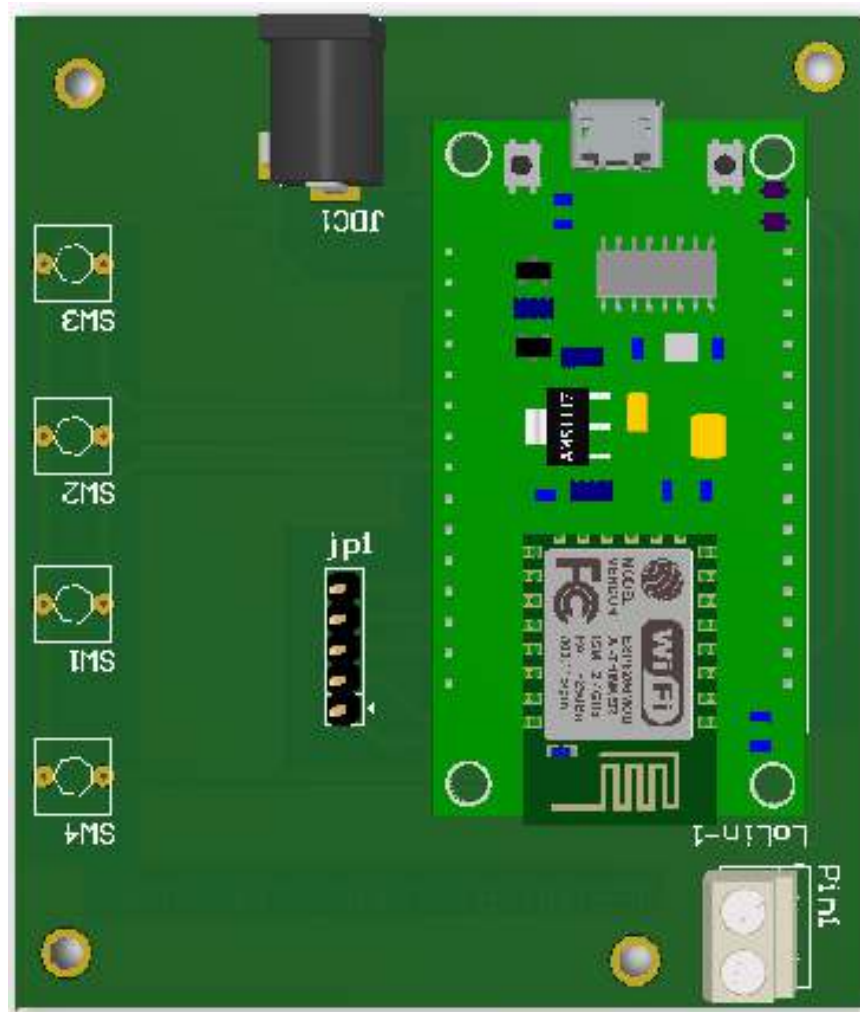
3.1. Mạch nguyên lí và mạch in hoàn thiện



Hình 3.1. Mạch nguyên lí hoàn thiện



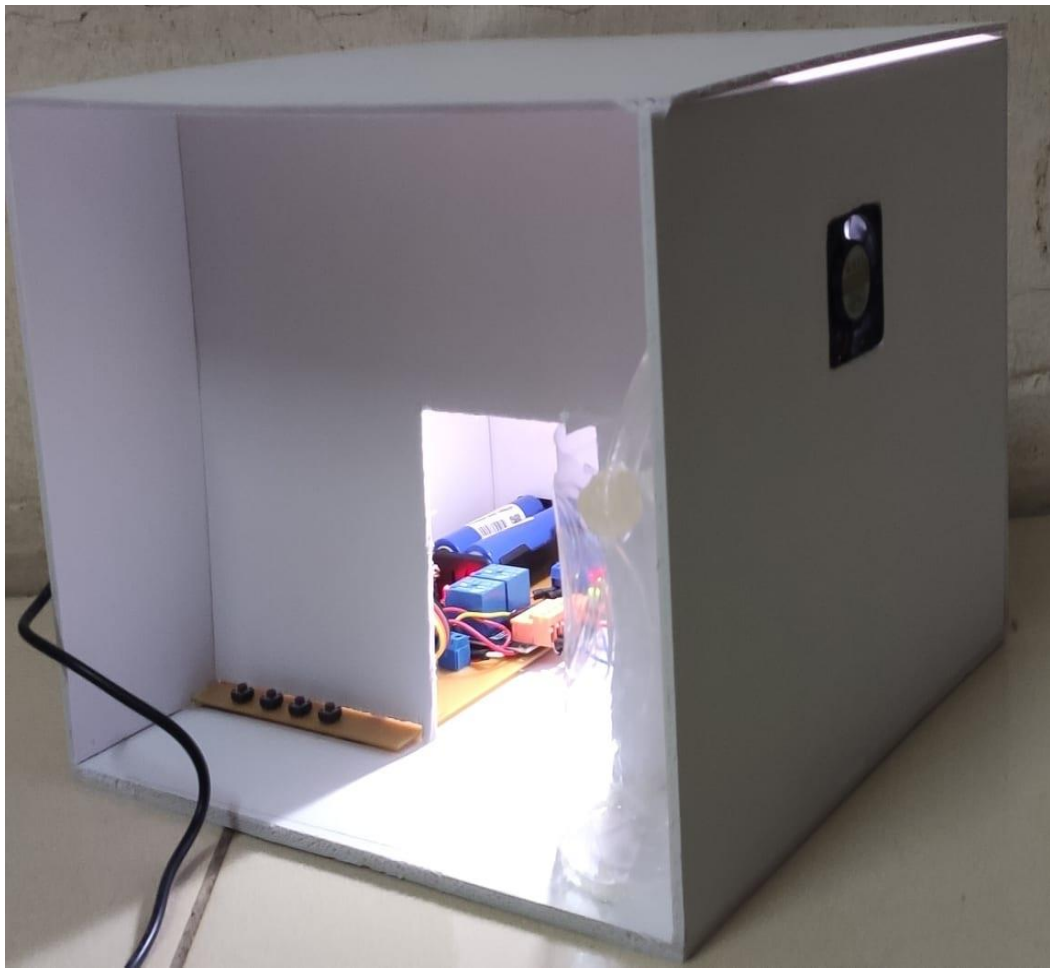
Hình 3.2. Mạch in 2D



Hình 3.3. Mạch in 3D

3.2. Kết quả thực nghiệm

Với đề tài nghiên cứu này, thực nghiệm trên mô hình dựa trên việc sử dụng các thiết bị 5V bao gồm đèn, quạt tản nhiệt và máy bơm 5V. Tất cả được lắp đặt trên cùng một mô hình và được điều khiển bật, tắt bởi giọng nói. Ngoài ra, người dùng cũng có thể điều khiển và theo dõi trạng thái hoạt động của các thiết bị thông qua giao diện App Blynk, việc sử dụng thông qua App giúp việc điều khiển từ xa trở nên dễ dàng và thuận tiện.



Hình 3.4. Mô hình hệ thống

3.3. Giao diện Google Assistant khi nhận lệnh

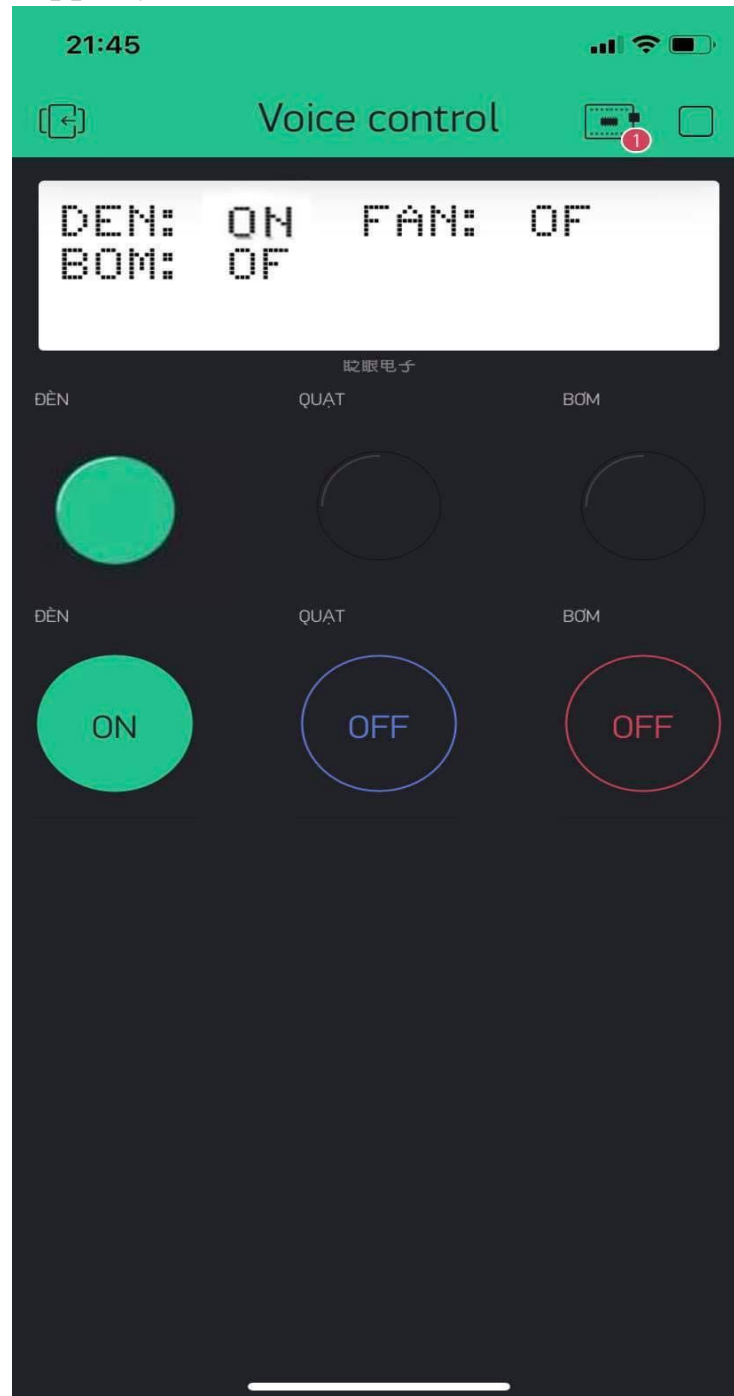
Sau khi khởi động ứng dụng, người dùng sẽ thực hiện nói và hệ thống sẽ trả lời theo các bước:

- Người dùng sẽ nói “OK Google” để bắt đầu ra lệnh.
- Google phản hồi “Xin chào, tôi là trợ lý google của bạn, tôi có thể tìm thấy những gì bạn cần và hoàn thành công việc. Nếu bạn cần trợ giúp chỉ cần hỏi :Bạn có thể làm gì. Vậy tôi có thể giúp gì cho bạn?”
- Người dùng sẽ đưa ra lệnh “Turn on the light”.
- Google sẽ phản hồi là đã bật.

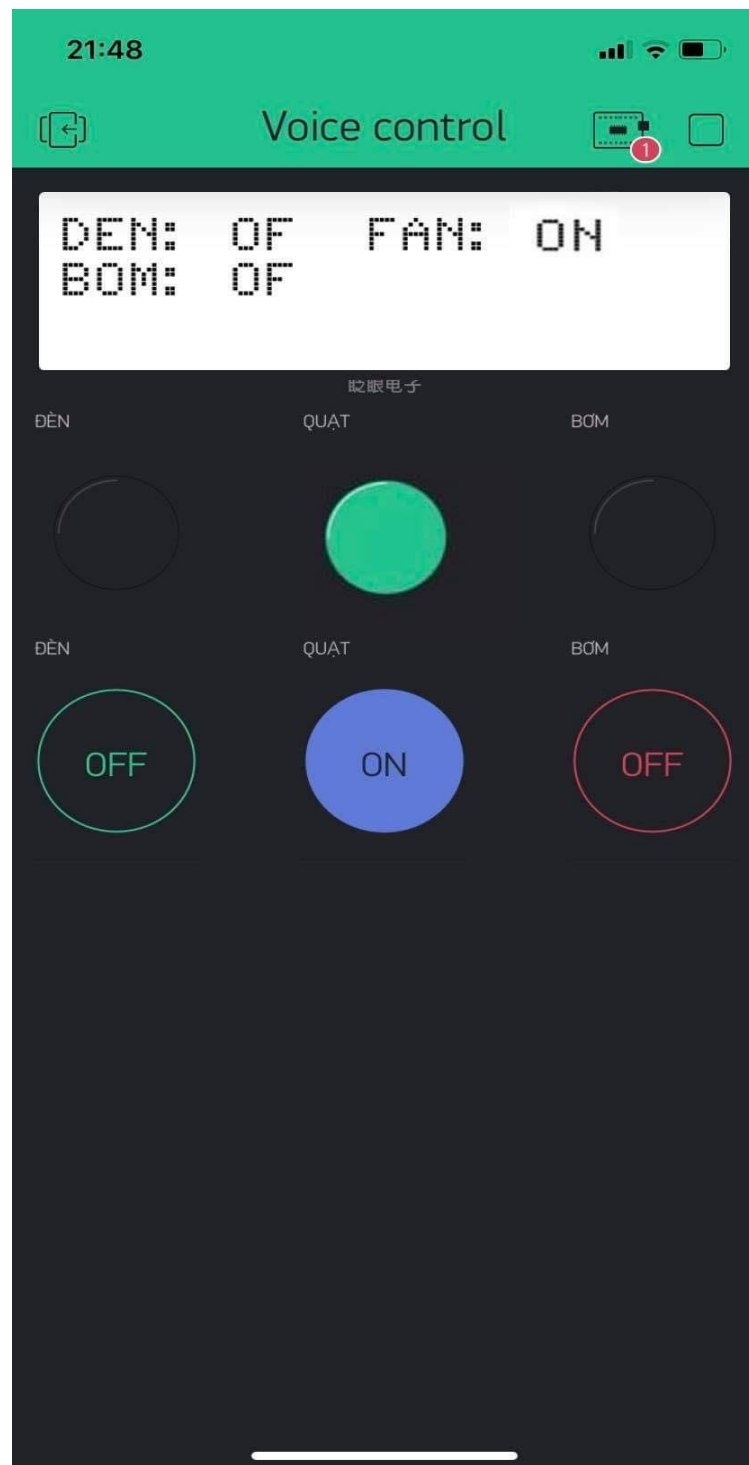


Hình 3.5. Giao diện điều khiển bằng giọng nói

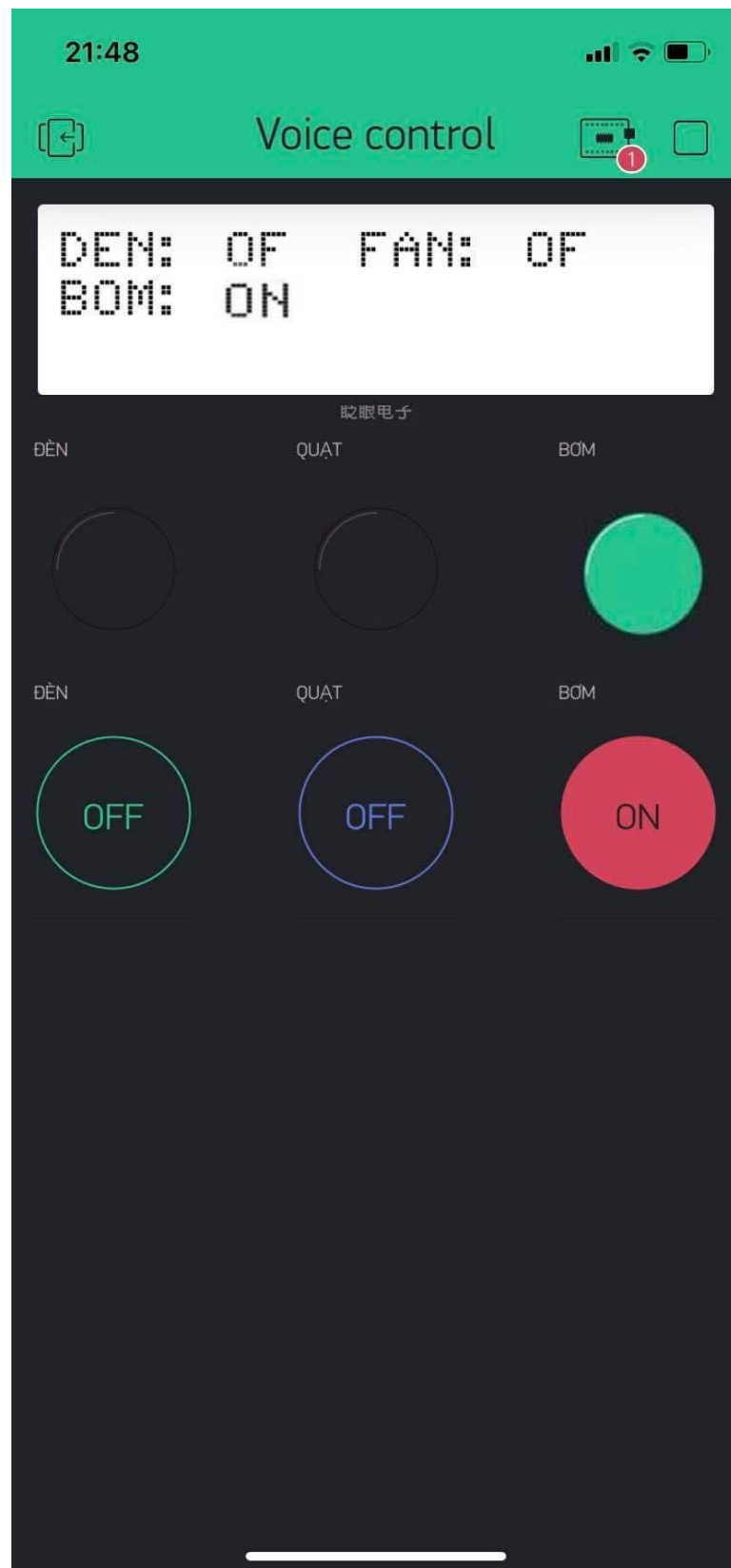
3.4. Giao diện App Blynk



Hình 3.6. Giao diện App Blynk bật đèn



Hình 3.7. Giao diện App Blynk bật quạt



Hình 3.8. Giao diện App Blynk bật máy bơm

Chức năng các icon trên app:

- Các LED màn hình : hiển thị trạng thái bật tắt các thiết bị
- Bốn nút nhấn : điều khiển các thiết bị

3.5. Đánh giá kết quả và hướng phát triển

3.5.1. Đánh giá kết quả

Ưu điểm:

Hoàn thành mô hình hệ thống với các yêu cầu đề ra, có thể điều khiển hoạt động của đèn, quạt, máy bơm bằng giọng nói thông qua Google Assistant. Việc nhận lệnh và thực hiện một cách chính xác với yêu cầu điều khiển. Ngoài ra, việc điều khiển không chỉ dừng lại ở một khoảng cách nhất định mà có thể điều khiển từ xa, không phụ thuộc vào khoảng cách thông qua app Blynk. Người dùng có thể điều khiển các thiết bị bằng app dù không có mặt ở nhà với mức chi phí lắp đặt tương đối phù hợp.

Nhược điểm:

Do thực hiện trong thời gian ngắn và chỉ dừng lại ở mức mô hình nên sản phẩm còn nhiều thiếu sót như tính thẩm mỹ chưa cao, còn lộ nhiều đường dây dẫn, cắt dán,...

Việc điều khiển bao gồm điều khiển bật tắt và điều khiển chế độ của thiết bị. Tuy nhiên, với mô hình này em mới chỉ dừng lại ở việc điều khiển bật, tắt thiết bị cũng chỉ sử dụng các thiết bị 5V chứ không phải 220V trong thực tế do vẫn là mô hình.

Tốc độ thực hiện lệnh trong một số trường hợp còn chậm do tính ổn định của vi điều khiển chưa cao.

3.5.2. Hướng phát triển của đề tài

Nhận thấy còn nhiều nhược điểm và một số tính năng của sản phẩm chưa đạt được như mong muốn nhưng do thời gian có hạn nên em đã đưa ra hướng phát triển của đề tài trong tương lai:

- Nâng cao tính thẩm mỹ và tự động của hệ thống.
- Sử dụng Tiếng Việt để điều khiển thay vì Tiếng Anh.
- Nâng cao và cải thiện đường truyền, kết nối của App điều khiển.
- Áp dụng đề tài ra ngoài thực tế , trong một ngôi nhà.

3.6. Kết luận chương 3

Chương 3 là chương tổng kết quá trình bằng kết quả sản phẩm đạt được. Hệ thống tích hợp nhiều thiết bị được điều khiển thông qua Google Assistant. Ngoài ra người dùng cũng có thể theo dõi trạng thái các thiết bị qua App Blynk. Thể hiện được các chức năng của hệ thống cũng như đánh giá được các ưu điểm và nhược điểm cần khắc phục từ đó nêu lên hướng phát triển của đề tài.

KẾT LUẬN

Sau thời gian làm đồ án với sự hướng dẫn tận tình giúp đỡ của thầy **Tiến Sĩ. Trần Đình Thông** và các thầy, cô trong khoa Điện tử - Viễn Thông của trường Đại học Công Nghiệp Hà Nội em đã hoàn thành tốt đề tài “Thiết kế hệ thống điều khiển thiết bị trong nhà bằng giọng nói sử dụng ESP8266”.

Trong quá trình thực hiện đề tài, em đã học tập được nguyên tắc xây dựng và giải quyết bài toán một bài toán ứng dụng cụ thể. Tìm hiểu được các nguyên tắc hoạt động của các thành phần trong hệ thống, cụ thể là tìm hiểu về ESP8266, Google Assistant và ứng dụng Blynk.

Trong thời gian tới, bên cạnh khắc phục những tồn tại của đề tài, em hi vọng sẽ có thể ứng dụng mô hình này tích hợp vào trong những smart home thực sự chứ ko phải trên mô hình nữa.

Dù đã rất cố gắng nhưng do đây là thiết kế lần đầu nên không thể tránh khỏi sai sót. Một lần nữa, em xin chân thành cảm ơn sự giúp đỡ nhiệt tình của thầy **Tiến Sĩ. Trần Đình Thông**, các thầy cô bộ môn cùng các bạn trong suốt thời gian qua. Em hy vọng sẽ nhận được nhiều ý kiến, góp ý từ thầy cô và các bạn để bản thiết kế này, cùng các bản thiết kế khác ngày càng hoàn thiện hơn.

TÀI LIỆU THAM KHẢO

Tài liệu Tiếng Việt

[I] **Phạm Minh Tuấn:** *Arduino và IOT(Internet Of Things) cho người mới bắt đầu.*

[II]. Giáo trình Arduino cho người mới bắt đầu, Cộng đồng Arduino Việt Nam

Tài liệu Tiếng Anh

<https://www.alldatasheet.com/datasheetpdf/pdf/297157/PCA/EP8266.html>

Trang Web và hình ảnh tham khảo

[1] <http://thuvienso.bvu.edu.vn/bitstream/TVDHBRVT/19773/1/Mai-Thanh-Tuan.pdf>

[2] <https://www.dienmayxanh.com/kinh-nghiem-hay/top-nhung-thiet-bi-ho-tro-google-assistant-giup-ca-1228866>

[3] <https://esp8266.vn/>

[4] <https://www.toponseek.com/blogs/cong-cu-google-assistant-la-gi-cach-google-assistant-hoat-dong/>

[5] <https://mechasolution.vn/Blog/blynk-la-gi>

[6] <https://banlinhkien.com/>

PHỤ LỤC

```

#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>
#include <BlynkSimpleEsp8266.h>

//=====

char auth[] = "x4i-VGAnf7SSRh4oD8dbfZuCAfkEYKts";
char *ssid = "Ok";
char *pass = "22222222";

//=====

WidgetLCD lc(V0);
int Relay1State = HIGH;
int Relay2State = HIGH;
int Relay3State = HIGH;
WidgetLED led1(V4);WidgetLED led2(V5);WidgetLED led3(V6);

//=====

void setup() {
  Blynk.virtualWrite(V1, 1);
  Blynk.virtualWrite(V2, 1);
  Blynk.virtualWrite(V3, 1);
  Serial.begin(9600);//
  WiFi.mode(WIFI_STA);WiFi.setAutoConnect
(true);WiFi.setAutoReconnect (true);
  WiFi.begin(ssid, pass);Blynk.config(auth);
  while (((WiFi.status() == WL_CONNECTED))) {
    Serial.print(".");
    delay(500);
  }
}

```

```
pinMode(D1,OUTPUT);pinMode(D2,OUTPUT);pinMode(D3,OUTPUT);
pinMode(D5, INPUT);pinMode(D6, INPUT);pinMode(D7, INPUT);
digitalWrite(D1,LOW);digitalWrite(D2,HIGH);digitalWrite(D3,HIGH);
```

```
attachInterrupt(D5, checkPhysicalButton, FALLING);
attachInterrupt(D6, checkPhysicalButton, FALLING);
attachInterrupt(D7, checkPhysicalButton, FALLING);
}
//=====================================================
void loop() {
  display();
  Blynk.run();
}
//=====================================================
ICACHE_RAM_ATTR void checkPhysicalButton()
{
  if (digitalRead(D5) == LOW) {
    while(digitalRead(D5) == LOW);
    Relay1State = !Relay1State;
    digitalWrite(D1, Relay1State);
    Blynk.virtualWrite(V1, Relay1State);
  }
  if (digitalRead(D6) == LOW) {
    while(digitalRead(D6) == LOW);
    Relay2State = !Relay2State;
    digitalWrite(D2, Relay2State);
    Blynk.virtualWrite(V2, Relay2State);
  }
}
```



```

if (digitalRead(D7) == LOW) {
    while(digitalRead(D7) == LOW);
    Relay3State = !Relay3State;
    digitalWrite(D3, Relay3State);
    Blynk.virtualWrite(V3, Relay3State);
}
}

//=====================================================

BLYNK_CONNECTED() {
    Blynk.syncAll();
}

//=====================================================

BLYNK_WRITE(V1) {
    Relay1State = param.asInt();
    digitalWrite(D1, Relay1State);
}

BLYNK_WRITE(V2) {
    Relay2State = param.asInt();
    digitalWrite(D2, Relay2State);
}

BLYNK_WRITE(V3) {
    Relay3State = param.asInt();
    digitalWrite(D3, Relay3State);
}

//=====================================================

void display() {
    if(Relay1State == HIGH){
        lc.print(0,0,"DEN: ON ");led1.on();}
    else{lc.print(0,0,"DEN: OF ");led1.off();}
}

```

```
if(Relay2State == LOW){  
    lc.print(9,0,"FAN: ON"); led2.on();}  
else{lc.print(9,0,"FAN: OF"); led2.off();}  
  
if(Relay3State == LOW){  
    lc.print(0,1,"BOM: ON ");led3.on();}  
else{lc.print(0,1,"BOM: OF ");led3.off();}  
}
```