|  |  |
| --- | --- |
| TRƯỜNG ĐẠI HỌC …………………….  KHOA............................  NGÀNH/CHUYÊN NGÀNH: ..........................  **ĐỒ ÁN TỐT NGHIỆP**  **NGHIÊN CỨU, THIẾT KẾ, CHẾ TẠO MÔ HÌNH NGÔI NHÀ THÔNG MINH**  ***Giảng viên hướng dẫn*: 1. *TS.GVC.* Trần Văn A**  **2. *Ths.GV.* Nguyễn Thị B**  ***Sinh viên thực hiện* : Nguyễn Văn A**  ***Lớp* : ………………..**  Nghệ An - 20 .….. | |
| (TÊN SINH VIÊN):............ĐATN ngành...........Năm .......(Cỡ chữ 18) | | TRƯỜNG ĐẠI HỌC ……………………..  KHOA............................  NGÀNH/CHUYÊN NGÀNH: ..........................  **ĐỒ ÁN TỐT NGHIỆP**  **NGHIÊN CỨU, THIẾT KẾ, CHẾ TẠO MÔ HÌNH NGÔI NHÀ THÔNG MINH**  *Giảng viên hướng dẫn*: 1. *TS.GVC.* Trần Văn A  2. *Ths.GV.* Nguyễn Thị B  *Sinh viên thực hiện* : Nguyễn Văn A  *Lớp* : ………………..  Nghệ An - 20 .….. | |

**Lời mở đầu**

Ngày nay trên thế giới với sự bùng nổ của các ngành công nghệ thông tin, điện tử v.v. dã làm cho đời sống của con người ngày càng hoàn thiện. Các thiết bị tự động hóa đã ngày càng xâm lấn vào trong sản xuất và thậm chí là vào cuộc sống sinh hoạt hằng ngày của mỗi con người. Do đó việc sở hữu một ngôi nhà thông minh và những thiết bị được điều khiển tự động không còn là mơ ước của con người nữa mà nó đã trở thành hiện thực hóa . Với những kiến thức đã học cùng với mong muốn thiết kế một ngôi nhà thông minh nên em đã quyết định chọn đề tài này để thực hiện.

**Tóm tắt nội dung**

Đồ án này thực hiện ý tưởng thiết kế và chế tạo mô hình nhà thông minh sử dụng ESP32. Bộ điều khiển thiết kế có khả năng bật tắt các thiết bị một cách linh hoạt bằng các thiết bị di động có kết nối mạng wifi hoặc mạng viễn thông di động. Các trạng thái của thiết bị cũng có thể được cập nhật tức thời lên các thiết bị này nhằm kiểm soát sự hoạt động của ngôi nhà. Ngoài ra, thông qua các cảm biến được bố trí trong ngôi nhà, bộ điều khiển có khả năng tự động điều chỉnh các thiết bị phù hợp với nhu cầu sử dụng. Mô hình thiết kế là một giải pháp hoàn thiện cho nhà thông minh với khả năng điều khiển mềm dẻo và giá thành thấp so với các sản phẩm khác trên thị trường.

*Nghệ An, ngày ….. tháng ….. năm 20…*

**Sinh viên thực hiện**

*(Ký và ghi rõ họ tên)*

**Nhận xét của giảng viên hướng dẫn**

……………………………………………………………………………………………

……………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

*Nghệ An, ngày ….. tháng ….. năm 20…*

**Giảng viên hướng dẫn**

*(Ký và ghi rõ họ tên)*

**MỤC LỤC**

[CHƯƠNG 1. GIỚI THIỆU CHUNG 1](#_Toc185379176)

[1.1 Giới thiệu về IoT 1](#_Toc185379177)

[1.1.1 Một số khái niệm về IoT 1](#_Toc185379178)

[1.1.2 Cơ sở kỹ thuật của IoT 1](#_Toc185379179)

[1.1.3 Xu hướng phát triển của thế giới với IoT 1](#_Toc185379180)

[1.1.4 Ứng dụng của IoT 2](#_Toc185379181)

[1.1.5 Thách thức trong việc nghiên cứu và triển khai IoT 3](#_Toc185379182)

[1.2 Tổng quan về nhà thông minh 3](#_Toc185379183)

[1.2.1 Giới thiệu mô hình nhà thông minh 3](#_Toc185379184)

[1.2.2 Nguyên tắc hoạt động của nhà thông minh 5](#_Toc185379185)

[1.2.3 Tiêu chuẩn của nhà thông minh 5](#_Toc185379186)

[1.2.4 Những xu hướng phát triển của nhà thông minh ở Việt Nam 6](#_Toc185379187)

[1.2.5 Một số hãng sản xuất sản phẩm nhà thông minh hiện nay 7](#_Toc185379188)

[1.3 Chọn đề tài 8](#_Toc185379189)

[CHƯƠNG 2. THIẾT KẾ PHẦN CỨNG 9](#_Toc185379190)

[2.1 Tổng quan về phần cứng 9](#_Toc185379191)

[2.1.1 Sơ đồ khối 9](#_Toc185379192)

[2.1.2 Khối vi điều khiển 10](#_Toc185379193)

[2.1.3 Khối cảm biến 14](#_Toc185379194)

[2.1.4 Khối chấp hành 19](#_Toc185379195)

[2.1.5 Khối hiển thị 24](#_Toc185379196)

[2.1.6 Các linh kiện khác 25](#_Toc185379197)

[2.2 Thiết kế mạch in 30](#_Toc185379198)

[CHƯƠNG 3. THIẾT KẾ PHẦN MỀM 35](#_Toc185379199)

[3.1 Nguyên lý hoạt động 35](#_Toc185379200)

[3.2 Phần mềm lập trình Arduino IDE 35](#_Toc185379201)

[3.3 Lưu đồ thuật toán của Blynk và ESP32 36](#_Toc185379202)

[3.4 Lưu đồ thuật toán của vi điều khiển 38](#_Toc185379203)

[3.5 Thuật toán xử lý ảnh 39](#_Toc185379204)

[3.6 Blynk IoT 41](#_Toc185379205)

[3.6.1 Giới thiệu về App Blynk 41](#_Toc185379206)

[3.6.2 Cách hoạt động của Blynk 42](#_Toc185379207)

[3.6.3 Thiết kế giao diện Blynk 44](#_Toc185379208)

[CHƯƠNG 4. KẾT QUẢ ĐẠT ĐƯỢC VÀ ĐÁNH GIÁ 48](#_Toc185379209)

[4.1 Kết quả đạt được 48](#_Toc185379210)

[4.1.1 Phần cơ khí 48](#_Toc185379211)

[4.1.2 Phần mềm giám sát điều khiển Blynk IoT 48](#_Toc185379212)

[4.2 Thử nghiệm và đánh giá 51](#_Toc185379213)

[4.2.1 Thử nghiệm 51](#_Toc185379214)

[4.2.2 Đánh giá 51](#_Toc185379215)

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 52](#_Toc185379216)

[TÀI LIỆU THAM KHẢO 47](#_Toc185379217)

**DANH MỤC HÌNH VẼ**

[Hình 1.1 Điều khiển hệ thống đèn bằng smartphone 4](#_Toc185379420)

[Hình 1.2 Quan sát ngôi nhà qua điện thoại 4](#_Toc185379421)

[Hình 1.3 Hiệu chỉnh nhiệt độ điều hòa từ xa 5](#_Toc185379422)

[Hình 2.1 Sơ đồ khối nhà thông minh 9](#_Toc185379423)

[Hình 2.2 Hình ảnh thực tế của IC ESP32 WROOM 11](#_Toc185379424)

[Hình 2.3 Sơ đồ khối của ESP32 WROOM 12](#_Toc185379425)

[Hình 2.4 Module ESP - WROOM – 32 12](#_Toc185379426)

[Hình 2.5 Sơ đồ kết nối chân của ESP32 với các linh kiện trong mạch 14](#_Toc185379427)

[Hình 2.6 Sơ đồ kết nối cảm biến MQ2 với vi điều khiển 15](#_Toc185379428)

[Hình 2.7 Cảm biến nhiệt độ, độ ẩm DHT11 15](#_Toc185379429)

[Hình 2.8 Sơ đồ kết nối DHT11 với vi điều khiển 16](#_Toc185379430)

[Hình 2.9 Cảm biến vân tay AS608 17](#_Toc185379431)

[Hình 2.10 Sơ đồ kết nối AS608 với vi điều khiển 18](#_Toc185379432)

[Hình 2.11 Bàn phím ma trận 4x4 18](#_Toc185379433)

[Hình 2.12 Động cơ servo SGG90 19](#_Toc185379434)

[Hình 2.13 Độ rộng xung 19](#_Toc185379435)

[Hình 2.14 Sơ đồ kết nối servo SG90 với vi điều khiển 20](#_Toc185379436)

[Hình 2.15 Quạt 12V 20](#_Toc185379437)

[Hình 2.16 Sơ đồ nguyên lý của động cơ điện một chiều độc lập 21](#_Toc185379438)

[Hình 2.17 Sơ đồ nguyên lý động cơ một chiều khi mắc thêm điện trở phụ 21](#_Toc185379439)

[Hình 2.18 Đặc tính cơ của động cơ khi thay đổi điện trở phần ứng 22](#_Toc185379440)

[Hình 2.19 Đặc tính cơ của động cơ khi giảm từ thông 23](#_Toc185379441)

[Hình 2.20 Đặc tính cơ của động cơ khi giảm điện áp phần ứng 23](#_Toc185379442)

[Hình 2.21 Led 5mm 24](#_Toc185379443)

[Hình 2.22 LCD 1602 24](#_Toc185379444)

[Hình 2.23 Sơ đồ kết nối LCD I2C với vi điều khiển 25](#_Toc185379445)

[Hình 2.24 Mạch mở rộng PCF8574 25](#_Toc185379446)

[Hình 2.25 Kết nối ma trận 4x4 với PCF8574 26](#_Toc185379447)

[Hình 2.26 Sơ đồ kết nối PCF8574 với vi điều khiển 27](#_Toc185379448)

[Hình 2.27 Mạch hạ áp LM2596 27](#_Toc185379449)

[Hình 2.28 Nguồn adapter 12V 5A 28](#_Toc185379450)

[Hình 2.29 Giao tiếp UART giữa Laptop và Esp32 30](#_Toc185379451)

[Hình 2.30 Phần mềm Altium Designer 30](#_Toc185379452)

[Hình 2.31 Sơ đồ kết nối chi tiết 32](#_Toc185379453)

[Hình 2.32 Hình ảnh 2D của mạch in 33](#_Toc185379454)

[Hình 2.33 Hình ảnh 3D mặt trước của mạch in 33](#_Toc185379455)

[Hình 2.34 Hình ảnh 3D mặt sau của mạch in 34](#_Toc185379456)

[Hình 3.1 Giao diện Arduino IDE 36](#_Toc185379457)

[Hình 3.2 Lưu đồ thuật toán của ESP32 và Blynk 37](#_Toc185379458)

[Hình 3.3 Lưu đồ thuật toán của vi điều khiển 38](#_Toc185379459)

[Hình 3.4 Xử lý ảnh nhận diện khuôn mặt 39](#_Toc185379460)

[Hình 3.5 Blynk Iot 42](#_Toc185379461)

[Hình 3.6 Đăng nhập Blynk IoT 44](#_Toc185379462)

[Hình 3.7 Tạo Template 44](#_Toc185379463)

[Hình 3.8 Tạo Device 45](#_Toc185379464)

[Hình 3.9 Tạo Datastreams 45](#_Toc185379465)

[Hình 3.10 Tạo giao diện Blynk IoT 46](#_Toc185379466)

[Hình 3.11 Giao diện Blynk được thiết kế trên Web 46](#_Toc185379467)

[Hình 3.12 Dowload Blynk Iot cho Android 47](#_Toc185379468)

[Hình 3.13 Giao diện Blynk được thiết kế trên App 47](#_Toc185379469)

[Hình 4.1 Hình ảnh thực tế của mạch bên trong 48](#_Toc185379470)

[Hình 4.2 Hình ảnh thực tế của phần mô hình 48](#_Toc185379471)

[Hình 4.3 Giám sát ngôi nhà thông minh qua Blynk trên SmartPhone 48](#_Toc185379472)

[Hình 4.4 Thiết lập chức năng cảnh báo khí gas 49](#_Toc185379473)

[Hình 4.5 Thiết lập chức năng tự động bật tắt quạt 51](#_Toc185379474)

**DANH MỤC BẢNG BIỂU**

[Bảng 2.1 So sánh các loại module Wifi phổ biến trên thị trường 10](#_Toc185379269)

[Bảng 2.2 Mô tả chân của cảm biến MQ2 15](#_Toc185379270)

[Bảng 2.3 Mô tả các chân của DHT11 16](#_Toc185379271)

[Bảng 2.4 Mô tả các chân của AS608 17](#_Toc185379272)

[Bảng 2.5 Mô tả các chân của servo SG90 20](#_Toc185379273)

[Bảng 2.6 Mô tả các chân của I2C kèm I2C 25](#_Toc185379274)

[Bảng 2.7 Mô tả các chân của PCF8574 26](#_Toc185379275)

# GIỚI THIỆU CHUNG

## Giới thiệu về IoT

### Một số khái niệm về IoT

Mạng lưới vạn vật kết nối Internet hoặc là mạng lưới thiết bị kết nối internet viết tắt là IoT là một kịch bản của thế giới, khi mà mỗi đồ vật, con người được cung cấp một định danh riêng của mình, và tất cả có khả năng truyền tải, trao đổi thông tin, dữ liệu qua một mạng duy nhất mà không cần sự tương tác trực tiếp giữa người với người, hay người với máy tính. IoT đã phát triển từ sự hội tụ của công nghệ không dây, công nghệ vi cơ điện tử và internet. Nó đơn giản là một tập hợp các thiết bị có khả năng kết nối với nhau, với internet và với thế giới bên ngoài để thực hiện một công việc nào đó.

Hiểu một cách đơn giản IoT là tất cả các thiết bị có thể kết nối với nhau. Việc kết nối có thể thực hiện qua wifi, kết nối băng rộng (3G, 4G), Bluetooth, ZigBee, hồng ngoại… Các thiết bị có thể là đện thoại thông minh, máy máy tính bảng, điều hòa, bóng đèn, máy giặt và nhiều thiết bị khác. Cisco nhà cung cấp giải pháp thiết bị hàng đầu hiện nay dự báo: Đến năm 2020, sẽ có khoảng 50 tỷ đồ vật kết nối internet, thậm chí con số này còn gia tăng nhiều hơn nữa. IoT sẽ là mạng khổng lồ kết nối tất cả mọi thứ bao gồm cả con người và sẽ tồn tại các mối quan hệ giữa người và người, người và thiết bị, thiết bị và thiết bị.

### Cơ sở kỹ thuật của IoT

Điểm quan trọng của IoT đó là các đối tượng phải có thể được nhận biết và định dạng (identifiable). Nếu mọi thứ được "đánh dấu" để phân biệt bản thân đối tượng đó với những thứ xung quanh thì chúng ta có thể hoàn toàn quản lí được nó thông qua máy tính. Việc đánh dấu (tagging) có thể được thực hiện thông qua nhiều công nghệ, chẳng hạn như RFID, NFC, mã vạch, mã QR, watermark kĩ thuật số... Việc kết nối thì có thể thực hiện qua wifi, mạng viễn thông băng rộng (3G, 4G), Bluetooth, ZigBee, hồng ngoại...

Ngoài những kĩ thuật nói trên, nếu nhìn từ thế giới web, chúng ta có thể sử dụng các địa chỉ độc nhất để xác định từng vật, chẳng hạn như địa chỉ IP. Mỗi thiết bị sẽ có một IP riêng biệt không nhầm lẫn. Sự xuất hiện của IPv6 với không gian địa chỉ cực kì rộng lớn sẽ giúp mọi thứ có thể dễ dàng kết nối vào Internet cũng như kết nối với nhau.

### Xu hướng phát triển của thế giới với IoT

Mặc dù đã có từ lâu nhưng kỷ nguyên Internet of Things chỉ thực sự được chú ý và bùng nổ trong những năm gần đây, sau sự phát triển của smartphone, tablet và những kết nối không dây,… Và ngay sau khi nhận được sự chú ý của cộng đồng, IoT đã cho thấy tiềm năng của mình với những số liệu đáng kinh ngạc.

Là “một trong những phát kiến quan trọng và quyền lực nhất của loài người”, Cisco IBSG, nhà cung cấp giải pháp và thiết bị mạng hàng đầu hiện nay dự báo: Đến năm 2020, sẽ có khoảng 50 tỷ đồ vật kết nối vào Internet, bao gồm hàng tỷ thiết bị di động, tivi, máy giặt, … Để thấy được sự phát triển của lĩnh vực này, họ cũng đưa ra số liệu vào năm 1984, khi mà Cisco mới thành lập mới chỉ có khoảng 1.000 thiết bị được kết nối mạng toàn cầu, đến năm 2010, con số này đã lên mức 10 tỷ.

Intel, đơn vị mới tham gia vào thị trường sản xuất chip cho các thiết bị thông minh phục vụ IoT cũng đã thu về hơn 2 tỷ USD trong năm 2014 từ lĩnh vực này, tăng trưởng 19% so với năm 2013. Những con số khẳng định IoT là xu hướng của tương lai. Internet of Things đến năm 2020 dự kiến sẽ đạt đến: 4 tỷ người kết nối với nhau, 4 ngàn tỷ USD doanh thu, hơn 25 triệu ứng dụng,hơn 25 tỷ hệ thống nhúng thông minh và 50 ngàn tỷ Gigabytes dữ liệu

Tác động của IoT rất đa dạng, trên các lĩnh vực: quản lý hạ tầng, y tế, xây dựng và tự động hóa, giao thông….

Cụ thể với lĩnh vực sản xuất - chế tạo, hiện theo thống kê của PwC, đã có 35% nhà sản xuất sử dụng cảm biến thông minh, 10% dự kiến sẽ sử dụng và 8% có kế hoạch sử dụng các thiết bị thông minh này trong 3 năm tới.

Trong lĩnh vực dầu khí, khai thác mỏ, dự kiến sẽ có 5,4 triệu thiết bị IoT được triển khai tại các cơ sở khai thác tới năm 2020. Chủ yếu sẽ là các bộ cảm biến kết nối Internet giúp cung cấp thông tin về môi trường. Dầu khí là một trong những ngành công nghiệp chủ chốt ứng dụng IoT trên diện rộng tới năm 2020. Trong khi đó, xe hơi kết nối đang là xu hướng nổi bật của thiết bị IoT hiện nay. Dự tính tới năm 2020, sẽ có hơn 220 triệu xe kết nối lưu thông trên đường.

Về bảo hiểm, 74% lãnh đạo trong ngành bảo hiểm tin rằng IoT sẽ thay đổi cơ bản chính sách bảo hiểm trong 5 năm tới, 74% có kế hoạch đầu tư phát triển và thực hiện các chiến lược về IoT - theo một nghiên cứu của SMA Research.

Còn với quốc phòng, chi tiêu cho các thiết bị bay không người lái dự kiến sẽ đạt 8,7 tỉ USD vào năm 2020. Ngoài ra, theo dự báo của Frost & Sullivan, sẽ có khoảng 126.000 robot quân sự sẽ được triển khai vào năm 2020.

Lĩnh vực nông nghiệp cũng không nằm ngoài vòng xoáy IoT. Dự kiến sẽ có 75 triệu thiết bị IoT được triển khai trong lĩnh vực này vào năm 2020, với tỉ lệ tăng hàng năm đạt 20%. Chủ yếu đó sẽ là những bộ cảm biến đặt trong lòng đất để theo dõi độ axit, nhiệt độ và các thông số giúp canh tác vụ mùa hiệu quả hơn.

Vì thế, Internet of Thing đang là chìa khóa của thành công trong tương laicents. Bên cạnh đó, công nghệ không dây đáp ứng đa tiêu chuẩn đang giúp giảm giá thành các mặt hàng thiết bị kết nối không dây; và những giao thức Internet mới đã giúp hiện thực hóa việc kết nối hàng tỷ thiết bị vào mạng lưới Internet.

Hiện trên thị trường đang có ngày càng nhiều thiết bị di động giá rẻ, sự cải thiện về điều kiện kinh tế của nhóm khách hàng Châu Á đã dẫn đến sự gia tăng theo cấp số nhân về sử dụng thiết bị di động ở khu vực này.

### Ứng dụng của IoT

IoT có rất nhiều ứng dụng trong nhiều lĩnh vực khác nhau như tự động hóa ngôi nhà, mua sắm thông minh, quản lí các thiết bị cá nhân, đồng hồ đo thông minh, phản hồi trong các tình huống khẩn cấp, quản lí môi trường. Hiện nay nhiều hãng, công ty, tổ chức trên thế giới đang nghiên cứu các nền tảng giúp xây dựng nhanh ứng dụng dành cho IoT. Đại học British Columbia ở Canada hiện đang tập trung vào một bộ toolkit cho phép phát triển phần mềm IoT chỉ bằng các công nghệ/tiêu chuẩn Web cũng như giao thức phổ biến. Công ty như ioBridge thì cung cấp giải pháp kết nối và điều khiển hầu như bất kì thiết bị nào có khả năng kết nối Internet, kể cả đèn bàn, quạt máy...

### Thách thức trong việc nghiên cứu và triển khai IoT

IoT vẫn còn một số trở ngại như chưa có một ngôn ngữ chung. Ở mức cơ bản nhất, Internet là một mạng dùng để nối thiết bị này với thiết bị khác. Nếu chỉ riêng có kết nối không thôi thì không có gì đảm bảo rằng các thiết bị biết cách nói chuyện nói nhau. Để các thiết bị có thể giao tiếp với nhau, chúng sẽ cần một hoặc nhiều giao thức (protocols), có thể xem là một thứ ngôn ngữ chuyên biệt để giải quyết một tác vụ nào đó. Chắc chắn bạn đã ít nhiều sử dụng một trong những giao thức phổ biến nhất thế giới, đó là HyperText Transfer Protocol (HTTP) để tải web. Ngoài ra chúng ta còn có SMTP, POP, IMAP dành cho email, FTP dùng để trao đổi file... Những giao thức như thế này hoạt động ổn bởi các máy chủ web, mail và FTP thường không phải nói với nhau nhiều, khi cần, một phần mềm phiên dịch đơn giản sẽ đứng ra làm trung gian để hai bên hiểu nhau. Còn với các thiết bị IoT, chúng phải đảm đương rất nhiều thứ, phải nói chuyện với nhiều loại máy móc thiết bị khác nhau. Đáng tiếc rằng hiện người ta chưa có nhiều sự đồng thuận về các giao thức để IoT trao đổi dữ liệu.

## Tổng quan về nhà thông minh

### Giới thiệu mô hình nhà thông minh

Nhà thông minh (tiếng Anh: home automation, domotics, smart home hoặc Intellihome) là kiểu nhà được lắp đặt các thiết bị điện, điện tử có thể được điều khiển hoặc tự động hoá hoặc bán tự động, thay thế con người trong thực hiện một hoặc một số thao tác quản lý, điều khiển. Hệ thống điện tử này giao tiếp với người dùng thông qua bảng điện tử đặt trong nhà, ứng dụng trên điện thoại di động, máy tính bảng hoặc một giao diện web.

Trong nhà thông minh, đồ dùng trong nhà từ phòng ngủ, phòng khách đều gắn các bộ điều khiển điện tử có thể kết nối với Internet và điện thoại di động, cho phép chủ nhân điều khiển vật dụng từ xa hoặc lập trình cho thiết bị ở nhà hoạt động theo lịch.. Các thiết bị này có thể tự đưa ra cách xử lý tình huống được lập trình trước, hoặc là được điều khiển và giám sát từ xa. Sau đây là các lợi ích đem đến cho bạn từ một hệ thống tự động hóa nhà thông minh

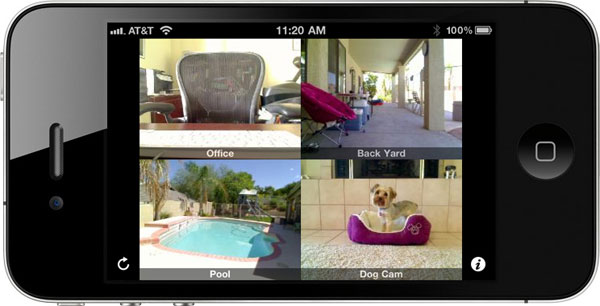
- Lợi ích 1 Tăng thêm sự an toàn qua việc điều khiển chiếu sáng và thiết bị điện (Appliance and Lighting Control)



Hình 1.1 Điều khiển hệ thống đèn bằng smartphone

Một lợi ích nữa của home automation đó là tăng thêm sự an toàn cho gia đình và ngôi nhà. Bạn có thể kiểm soát các thiết bị điện dù nhỏ và cả việc chiếu sáng, bằng một cái chạm nhẹ đầu ngón tay vào thiết bị công nghệ yêu thích của bạn. Không chỉ tiết kiệm tiền điện cho bạn thông qua việc tự động tắt khi không có người, lighting control còn có thể tự động bắt tắt đèn theo chu kỳ để đánh lừa kẻ xấu tưởng bạn vẫn có nhà. Điều này làm tăng thêm độ an toàn và an ninh cho nhà bạn.

- Lợi ích 2 Gia tăng quan sát thông qua camera an ninh



Hình 1.2 Quan sát ngôi nhà qua điện thoại

Chúng ta không thể có mặt ở mọi nơi cùng lúc. Điều này khiến chúng ta thường bỏ lỡ nhiều việc đang diễn ra, có lẽ với ngay cả trong nhà hay sân vườn mình. Với hệ thống tự động hóa nhà thông minh, có thể dễ dàng quan sát thấy việc đang diễn ra.. Các camera an ninh sẽ gia tăng độ an toàn cho gia đình bằng cách ghi nhận lại các hình ảnh khi nó phát hiện có chuyển động hoặc tự động ghi hình tại một thời điểm nhất định nào đó trong ngày.

- Lợi ích 3 Gia tăng tiện nghi thông qua việc hiệu chỉnh nhiệt độ điều hòa



Hình 1.3 Hiệu chỉnh nhiệt độ điều hòa từ xa

Thường thì chúng ta rời nhà sớm đi làm vào buổi sáng và quên chỉnh lại nhiệt độ của bộ điều chỉnh nhiệt (thermostat). Kết quả là khi về nhà chúng ta sẽ thấy hoặc là quá nóng hoặc quá lạnh. Với hệ thống nhà thông minh, có thể chỉ cần đơn giản điều chỉnh nhiệt độ sẵn từ xa một vài giờ trước khi về đến nhà. Điều này còn giúp tiết kiệm tiền, tiết kiệm năng lượng.

- Lợi ích 4 Tiết kiệm thời gian

. Dễ dàng tiết kiệm được các khoảng thời gian quý báu và dành hiệu năng tốt hơn cho công việc.

- Lợi ích 5 Tiết kiệm tiền và gia tăng tiện nghi

Như vừa đề cập trên, hệ thống home automation giúp bạn tiết kiệm tiền.

### Nguyên tắc hoạt động của nhà thông minh

Nhà thông minh được hiểu là hệ thống các thiết bị điện tử gia dụng được kết hợp với nhau thành mạng thiết bị và hoạt động theo kịch bản tùy biến nhằm tạo ra môi trường sống tiện nghi, an toàn, tiết kiệm năng lượng. Một hệ thống nhà thông minh cơ bản bao gồm một máy tính điều khiển trung tâm, được gọi là máy chủ (Home Server), có nhiệm vụ kết nối các thiết bị với nhau và điều khiển toàn bộ hệ thống nhà.

Các thiết bị gia dụng đầu cuối là những vật dụng điện tử trong nhà như các thiết bị an ninh, hệ thống cửa, điều hòa, rèm mành, hệ thống đèn, quạt thông gió, ti vi, bếp gas… Các thiết bị này được kết nối với nhau trong hệ thống mạng thiết bị bằng công nghệ truyền dữ liệu, qua đường điện (Power line communication – PLC) hoặc không dây (Zigbee) và được kết nối trực tiếp đến Home Server.

Cuối cùng là hệ thống các phần mềm điều khiển ngôi nhà cài đặt trên Home Server, trên các thiết bị điều khiển và các thiết bị điện tử gia dụng đầu cuối. Chủ nhân của Hệ thống nhà thông minh có thể kiểm soát, điều khiển ngôi nhà cũng như các thiết bị trong nhà bằng nhiều phương tiện như: điện thoại di động, tablet, laptop… ở bất kì đâu và bất kỳ lúc nào.

### Tiêu chuẩn của nhà thông minh

Với sự phát triển không ngừng của khoa học hiện đại, con người đã ngày càng nâng cao đời sống của mình hơn và luôn mơ ước tới một cuộc sống hiện đại và tiện nghi nhất. Chính từ những nhu cầu đó, con người đã có rất nhiều sáng tạo phục vụ cho cuộc sống. Và ý tưởng “ngôi nhà thông minh” cũng xuất phát từ nhu cầu thực tiển như vậy. Những công nghệ phục vụ cho ngôi nhà mơ ước đã có từ rất lâu nhưng gần đây mới được đưa ra công bố rộng rãi. Có rất nhiều công ty đã đưa ra giải pháp cho ngôi nhà thông minh, nhưng nhìn chung tất cả đều hướng đến các tiêu chuẩn sau đây:

Tự động hóa hoạt động của ngôi nhà. Các thiết bị cảm biến, giám sát sẽ thu thập thông tin của ngôi nhà như nhiệt độ, độ ẩm, lượng mưa... Các thông tin thu thập được sẽ được phân tích ở bộ xử lý trung tâm. Từ kết quả phân tích được sẽ đưa ra các điều khiển hoạt động của các thiết bị cho phù hợp nhất. Ngoài ra các tính năng như bật đèn, đóng mở rèm… cũng có thể được cài đặt hoạt động tự động theo ý muốn của người sử dụng.

Đảm bảo an ninh, an toàn cho ngôi nhà. Hiện nay hệ thống giám sát an ninh như camera, dấu vân tay hoặc nhận dạng qua hình ảnh đã và đang được sử dụng rất phổ biến. Hầu như nhà nào cũng được trang bị đầy đủ, vì vậy các hệ thống nhà hiện nay đang được đánh giá có chất lượng an ninh, an toàn rất tốt.

Đem lại sự thoải mái cho người sử dụng. Đây là tiêu chí đánh giá quan trọng nhất cho mỗi ngôi nhà. Mỗi ngôi nhà sẽ được thiết kế sao cho đem lại sự thoải mái nhất cho người sử dụng. Qua đó nâng cao chất lượng cuộc sống cho mỗi người sử dụng.

Cung cấp các dịch vụ giải trí chất lượng cao. Ngày nay nhu cầu giải trí của người sử dụng ngày càng cao. Để đáp ứng được nhu cầu này cho người sử dụng, mỗi ngôi nhà đều được trang bị các hệ thống giải trí có chất lượng tốt nhất.

Cung cấp khả năng giám sát, điều khiển từ xa. Mỗi ngôi nhà thông minh đều được trang bị các hệ thống giám sát, các hệ thống này đều được kết nối tới thiết bị của người sử dụng nên ngôi nhà luôn được giám sát rất tốt. Các thiết bị đều được kết nối tới bộ quản lý trung tâm vì thế người sử dụng có thể điều khiển bất kỳ thiết bị nào ở mọi nơi vào mọi thời điểm khi mà các thiết bị đã được kết nối qua internet.

Tăng hiệu suất các hệ thống, giảm điện năng tiêu thụ. Với các hệ thống điều khiển đơn giản nhưng hiệu quả thì năng lượng luôn được tiết kiệm một cách tối ưu.

### Những xu hướng phát triển của nhà thông minh ở Việt Nam

Vài năm trở lại đây, khi thế giới đang dần tiến vào kỷ nguyên Internet of Things (IoTs), 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. Việt Nam cũng không nằm ngoài xu hướng này. Nhà thông minh Việt Nam là một khái niệm không còn xa lạ với nhiều người. Không những thế, thị trường nhà thông minh Việt Nam phát triển mạnh chủ yếu tại những thành phố lớn như Hải Phòng, Hà Nội, Đà Nẵng, tp Hồ Chí Minh. Bởi tại những thành phố lớn, việc tiếp cận các khái niệm và công nghệ mới dễ dàng hơn.

- Xu hướng nhà thông minh trên thế giới:

Đối với các công ty lớn về công nghệ thì cuộc cánh mạng công nghệ 4.0 và công nghệ IoT được xem là cơ hội tỉ USD trên thị trường đầy tiềm năng này. Theo một thống kê của công ty nghiên cứu thị trường Statista thì vào năm 2020 giá trị thị trường của Smarthome -nhà thông minh dự báo đạt tới 43 tỉ USD. Con số này tăng gấp 3 lần so với năm 2014. Xu hướng nhà thông minh được dự báo như một trong những ứng dụng công nghệ một cách toàn diện nhất vào cuộc sống, là cả một căn nhà chứ không chỉ là một thiết bị thông minh.

- Xu hướng nhà thông minh tại Việt Nam:

Với những tiềm năng phát triển đó, có nhiều nhà phát triển và xâm nhập thị trường nhà thông minh Việt Nam như Lumi, Bkav,… hay tới các nhà đầu tư nước ngoài khác. Tuy chỉ mới phát triển từ 3 -5 năm nay, nhưng nhiều đơn vị trong nước đã nắm được thị phần phân phối nhà thông minh tại Việt Nam khá lớn như Lumi, Bkav. Các doanh nghiệp ở Việt Nam phần lớn cung cấp các giải pháp nhà thông minh thiên về giải pháp an ninh, an toàn, điều khiển thiết bị thông qua smartphone, điều khiển qua loa thông minh,..

### Một số hãng sản xuất sản phẩm nhà thông minh hiện nay

Tại Việt Nam hiện nay, nhu cầu lắp đặt nhà thông minh chưa nhiều vì đây là một lĩnh vực còn khá mới. Nhà thông minh được lắp đặt chủ yếu tại các tòa nhà mới xây, khu chung cư cao cấp, và một số ít biệt thự, khách sạn sang trọng.

Hiện tại, thị trường nhà thông minh ở Việt Nam chia thành 2 phân khúc là trung cấp và cao cấp. Với phân khúc cao cấp khách hàng phải bỏ từ vài trăm triệu tới vài tỉ động cho trọn bộ giải pháp nhà thông minh. Các đơn vị thực hiện thường là đối tác ủy quyền của những ông lớn về công nghệ trên thế giới như Mỹ, Pháp, Đức,…

Còn đối với nhà thông minh Việt Nam, khách hàng chỉ mất chi phí từ 30 -50 triệu là có thể hoàn thiện một căn hộ chung cư thông minh. Và với chi phí từ 70 -100 triệu động cho một căn biệt thự đầy đủ giải pháp. Không những thế, việc thi công chỉ mất từ 2 -3 ngày mà không đập phá hay phải đi dây lại hệ thống điện.

Một số tập đoàn sản xuất nhà thông minh nổi tiếng

a. Tập đoàn Scheneide – Pháp

Scheneider là một tập đoàn lớn trên thế giới cung cấp thiết bị điện. Ngày 18/ 4/2017 tập đoàn Scheneider đã khánh thành nhà máy chuyên sản xuất thiết bị điện cho giải pháp nhà thông minh tại khu công nghệ cao Sài Gòn, Quận 9, TPHCM. Có thể xem đây là một trong những nhà máy có quy mô lớn nhất của tập đoàn Scheneider trong chuỗi cung ứng toàn cầu gồm hơn 200 trung tâm sản xuất. Các thiết bị điện thông minh của Scheneider đều có những ưu điểm vượt trội về chất lượng cũng như độ bền của sản phẩm. Tuy nhiên mẫu mã thiết kế của sản phẩm chưa thật sự sắc xảo, tinh tế. So với giá thành mà khách hàng phải bỏ ra rất cao để sở hữu một thiết bị điện của Scheneider.

b. Nhà thông minh Smart 4G – Mỹ

Smart 4g tập đoàn chuyên về tư vấn và cung cấp thiết bị nhà thông minh, được thành lập từ năm 1986 có trụ sở chính tại Mỹ. Giải pháp nhà thông minh của Smart 4G cũng tương tự như các hãng nhà thông minh hiện nay chưa có gì nổi trội và khác biệt lớn. Thiết kế thiết bị điện thông minh của hãng được đánh giá là chưa được đẹp, thiết kế của sản phẩm vẫn còn 1 phần nào đó hơi thô chưa được sang trọng. Giá thành sản phẩm khá cao so với mức thu nhập của người Việt Nam.

c. Công ty nhà thông minh TIS Smarthome – Mỹ

TIS SMART HOME là thương hiệu đến từ Mỹ với kinh nghiệm hơn 15 năm trong lĩnh vực nhà tự động (home automation) và nhà thông minh (smart home). Văn phòng và nhà máy của TIS được đặt tại Texas. Sau đó, nhằm muốn giảm chi phí sản xuất thiết bị nên TIS đã dời xưởng sản xuất về Hong Kong – Trung Quốc, điều này giúp thiết bị thông minh của TIS có giá thành khá tốt so với các hãng khác. về thiết kế sản phẩm của TIS , thiết kế mặt kính hiện đại và sang trọng. Sản phẩm đạt chất lượng tiêu chuẩn châu Âu. Công ty đã có những dự án công trình lớn như Vinhome Tân Cảng, khu căn hộ cao cấp Horizon, khu dân cư Fezzila, biệt thự biển, bãi giữ xe quy mô lớn ở các trung tâm thương mại. Hiện nay, TIS cũng đang đẩy mạnh vào phân khúc các hộ gia đình trung lưu muốn sở hữu nhà thông minh... vv

## Chọn đề tài

Từ những ưu điểm vừa đưa ra, em lựa chọn đề tài: “*Nghiên cứu, thiết kế, chế tạo mô hình nhà thông minh*” làm nội dung cho đồ án tốt nghiệp.

Với đề tài này, mục tiêu được đặt ra như sau:

* Sử dụng bàn phím số và cảm biến vân tay để mở cửa tự động.
* Sử dụng xử lý ảnh nhận diện đúng người để mở cửa.
* Sử dụng cảm biến gas để ngăn chặn khả năng rò rỉ gas.
* Giám sát nhiệt độ, độ ẩm phòng ngủ, phòng khách.
* Hẹn giờ bật tắt điều hoà

Ngoài ra, từ những tiêu chí về phần cứng, em đã khảo sát và thực hiện chọn một phần mềm ổn định cho dự án của mình. Hiện tại em lựa chọn Blynk IoT. Em đang hiện thị dữ liệu lên trên app Blynk để có thể theo dõi một cách trực quan nhất. Sau đây, em xin phép đến với chương 2 Thiết kế phần cứng để có thể làm rõ chi tiết hơn về mặt công nghệ trong dự án của mình.

# THIẾT KẾ PHẦN CỨNG

## Tổng quan về phần cứng

### Sơ đồ khối

A diagram of a network

Description automatically generated

Hình 2.1 Sơ đồ khối nhà thông minh

Chức năng và nhiệm vụ các khối trong sơ đồ hình 2.1 như sau:

* Khối nguồn là khối cung cấp nguồn cho bộ điều khiển hệ thống và các khối mạch khác hoạt động. Yêu cầu đối với khối nguồn là tính ổn định và giá trị điện áp cung cấp phải phù hợp với các khối chức năng khác của bộ điều khiển.
* SmartPhone cung cấp giao diện để giám sát nhiệt độ độ ẩm, điều khiển đèn, hẹn giờ,…
* Laptop nhận diện đúng người đã được traning thì gửi dữ liệu qua cổng COM đến Esp32 mở cửa
* Khối vi điều khiển sử dụng ESP32 là khối xử lý trung tâm của hệ thống, thực hiện việc tiếp nhận, xử lý thông tin và giao tiếp với các thiết bị bên ngoài. Các nhiệm vụ chính như nhận giá trị nhiệt độ, độ ẩm, khí gas, truyền dữ liệu tới người sử dụng và nhận lệnh điều khiển trực tiếp từ ngưới sử dụng, xử lý dữ liệu.
* Khối cảm biến gồm cảm biến khí Gas theo dõi khí Gas, nếu vượt ngưỡng sẽ gửi tín hiệu về ESP32 để bật quạt. Cảm biến nhiệt độ độ ẩm ở đây sử dụng cảm biến DHT11 là module tích hợp việc đo giá trị nhiệt độ và độ ẩm kết quả đo được được chuyển về bộ xử lý trung tâm. Cảm biến vân tay và KeyPad để kiểm tra xem mật khẩu hoặc vân tay đã nhập có hợp lệ hay không để mở cửa.
* Khối chấp hành là các thiết bị cần điều khiển, trong nhà kính đó là các thiết bị như: quạt hút gió, led chiếu sáng,servo …
* Khối hiển thị sử dụng LCD 1602 để hiện thị xem việc dùng vân tay hay nhập mật khẩu mở cửa có đúng hay không.

Với sơ đồ khối hệ thống ở trên bài toán đặt ra là nghiên cứu chế tạo ngôi nhà thông minh thực hiện được các chức năng như điều khiển bật tắt thiết bị từ xa thông qua mạng wifi, điều khiển bật tắt thiết bị bằng công tắc, giám sát nhiệt độ độ ẩm,…..

### Khối vi điều khiển

1. Lựa chọn vi điều khiển

Bởi vì yêu cầu của hệ thống là kết nối với ứng dụng trên điện thoại vậy nên các giao thức truyền thông không dây như Bluetooth, Zigbee, LoRa,… không phù hợp với hệ thống. Các giao thức truyền thông không dây này thường phù hợp với các ứng dụng M2M (Machine to Machine – máy và máy) vậy nên nếu sử dụng các giao thức này thì em cần phải thiết kế thêm một bộ gateway để nhận và xử lý dữ liệu. Đồng thời các giao thức này có khoảng cách truyền ngắn (Bluetooth khoảng 20m, Zigbee lên đến 75m, LoRa có thể lên đến hàng km). Thay vào đó, em sử dụng các giao thức truyền thông có khả năng kết nối internet trực tiếp cho hệ thống như WiFi, mạng di động (3G, 4G, thậm chí là 5G). Hơn nữa vì hệ thống nằm hoàn toàn trong ngôi nhà của chúng ta, chi phí đầu tư cho mạng cáp quang/WiFi thấp hơn nhiều so với mạng di động. Đồng thời sử dụng mạng cáp quang/WiFi đem lại sử ổn định tốt hơn. Vậy nên em lựa chọn WiFi để kết nối internet cho hệ thống. Hiện nay, có rất nhiều thiết bị có thể kết nối đến mạng WiFi, phổ biến và nổi bật nhất là họ IC ESP.

Bảng 2.1 So sánh các loại module Wifi phổ biến trên thị trường

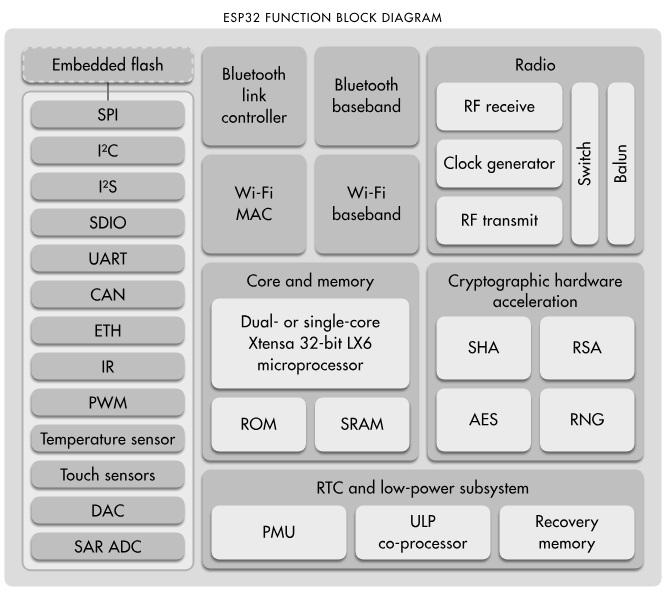
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tên IC** | **Loại mạng**  **hỗ trợ** | **Chuẩn**  **WiFi** | **Dải tần**  **(MHz)** | **Giá**  **thành** | **Đánh giá** |
| ESP8266  ESP-12F | WiFi | 802.11  b/g/n | 2412 -  2484 | $2 | Giá rẻ phù hợp triển khai số lượng lớn, hỗ trợ ít mạng |
| ESP8266  ESP-01 | WiFi | 802.11  b/g/n | 2400 -  2483.5 | $2 | Giá rẻ phù  hợp triển  khai số lượng  lớn, hỗ trợ ít  mạng |
| ESP32  WROOM | WiFi/ Bluetooth | 802.11  b/g/n | 2412 –  2484 | $3.5 | Giá rẻ, hỗ trợ  thêm  Bluetooth |
| ESP8266  ESP-07 | WiFi | 802.11  b/g/n | 2412 –  2484 | $2.5 | Giá rẻ phù  hợp triển  khai số lượng  lớn, hỗ trợ ít  mạng |

Từ bảng so sánh, sau khi xem xét các yêu cầu của hệ thống, em xin lựa chọn sử dụng IC ESP32 WROOM bởi vì:

* Hỗ trợ mạng WiFi phù hợp cho các ứng dụng IoT cũng như đồ án này,
* Hỗ trợ các chuẩn WiFi phổ biến hiện nay,
* Ngoài ra còn hỗ trợ chuẩn Bluetooth v4.2 và BLE (Bluetooth Low Energy) giúp hệ thống có tính linh hoạt, có thể mở rộng khi cần thiết.
* Việc tích hợp thành IC này cũng khá thuận tiện khi chúng hỗ trợ sẵn chuẩn giao tiếp nối tiếp UART để trao đổi dữ liệu,
* Kích thước nhỏ gọn dễ tích hợp, giá thành tương đối rẻ.



Hình 2.2 Hình ảnh thực tế của IC ESP32 WROOM

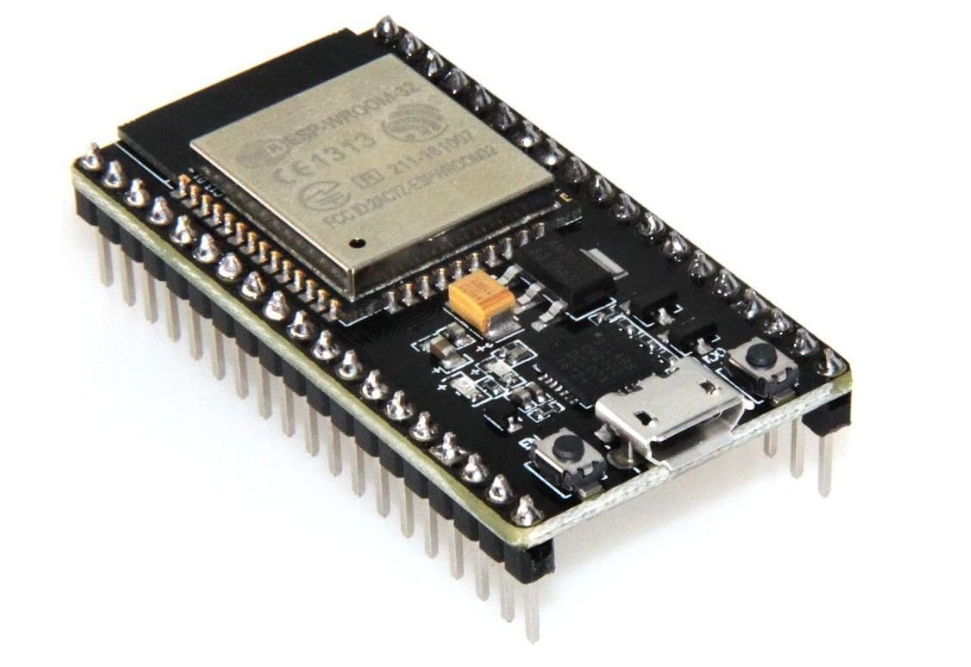


Hình 2.3 Sơ đồ khối của ESP32 WROOM

Vai trò của từng khối:

* Khối radio: có vai trò bức xạ và thu nhận sóng điện từ; trộn dao động điện từ cao tần với tín hiệu cần truyền tải hoặc tách tín hiệu ra khỏi dao động điện từ; khuếch đại dao động điện từ đã được biến điệu đối với trường hợp phát, khuếch đại tín hiệu điện từ nhận được đối với trường hợp nhận.
* Khối Core and Memory: điều khiển hệ thống, giao tiếp với các thiết bị thông qua các ngoại vi.
* Khối Embedded Flash: lưu trữ chương trình của khối Core and memory.
* Ngoài ra còn một số khối ngoại vi phục vụ các yêu cầu của hệ thống.: UART, PWM, GPIO, RTC,…

1. Đặc điểm của ESP32



*Hình 2.4 Module ESP - WROOM – 32*

ESP32 là một hệ thống vi điều khiển trên chip (SoC) giá rẻ của Espressif Systems, nhà phát triển của ESP8266 SoC. Nó là sự kế thừa của SoC ESP8266 và có cả hai biến thể lõi đơn và lõi kép của bộ vi xử lý 32-bit Xtensa LX6 của Tensilica với Wi-Fi và Bluetooth tích hợp.

Điểm tốt về ESP32, giống như ESP8266 là các thành phần RF tích hợp của nó như bộ khuếch đại công suất, bộ khuếch đại nhận tiếng ồn thấp, công tắc ăng-ten, bộ lọc và Balun RF. Điều này làm cho việc thiết kế phần cứng xung quanh ESP32 rất dễ dàng vì bạn cần rất ít thành phần bên ngoài.

Một điều quan trọng khác cần biết về ESP32 là nó được sản xuất bằng công nghệ 40 nm công suất cực thấp của TSMC. Vì vậy, việc thiết kế các ứng dụng hoạt động bằng pin như thiết bị đeo, thiết bị âm thanh, đồng hồ thông minh, ..., sử dụng ESP32 sẽ rất dễ dàng.

Thông số ESP32:

- Bộ vi xử lý LX6 32-bit lõi đơn hoặc lõi kép với xung nhịp lên đến 240 MHz.

- 520 KB SRAM, 448 KB ROM và 16 KB SRAM RTC.

- Hỗ trợ kết nối Wi-Fi 802.11 b / g / n với tốc độ lên đến 150 Mbps.

- Hỗ trợ cho cả thông số kỹ thuật Bluetooth v4.2 và BLE cổ điển.

- 30 GPIO có thể lập trình.

- 18 kênh SAR ADC 12 bit và 2 kênh DAC 8 bit.

- Kết nối nối tiếp bao gồm 4 x SPI, 2 x I2C, 2 x I2S, 3 x UART.

- Ethernet MAC cho giao tiếp mạng LAN vật lý (yêu cầu PHY bên ngoài).

- 1 bộ điều khiển host cho SD / SDIO / MMC và 1 bộ điều khiển slave cho SDIO/SPI.

- Động cơ PWM và 16 kênh LED PWM.

- Khởi động an toàn và mã hóa Flash.

- Tăng tốc phần cứng mật mã cho AES, Hash (SHA-2), RSA, ECC và RNG.

Điểm mạnh của ESP32:

ESP32 là sự nâng cấp hoàn hảo của ESP8266, với ESP8266 phù hợp với các dự án nhỏ và tiết kiệm chi phí. ESP32 lại phù hợp với các dự án phức tạp hơn, tốc độ xử lý cao hơn và tích hợp nhiều ngoại vi mạnh mẽ hơn.

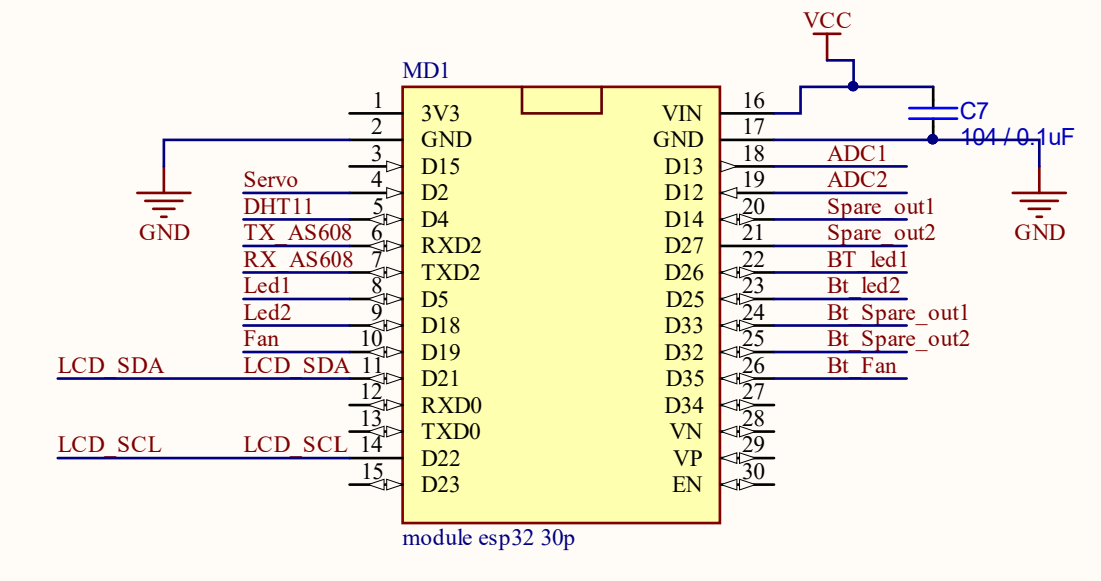
ESP8266 là 17 chân GPIO, ADC độ phân giải 10 bit, 8 kênh PWM mềm trong khi đó ESP 32 có tới 30/38 chân GPIO, 18 kênh ADC độ phân giải 12-bit, 16 kênh PWM mềm, Touch Sensor, Hall Effect Sensor, Ethernet MAC Interface, Cảm biến nhiệt độ được tích hợp sẵn.

Về bộ nhớ ESP32 có thêm 4MB External Flash và 520KB SRAM (static random access memory) trong đó 8 KB RAM RTC tốc độ cao – 8 KB RAM RTC tốc độ thấp (dùng ở chế độ DeepSleep).

ESP32 hỗ trợ Bluetooth 4.2 và BLE (Bluetooth Low Energy). Việc hỗ trợ cả bluetooth khiến ESP32 có thể tương tác với các thiết bị như là bàn phím, chuột, điện thoại khi mà không có wifi.

Ultra Low Power giải quyết vấn đề năng lượng cho ESP bởi vì sử dụng Wi-Fi sẽ rất ngốn điện đặc biệt khi chúng ta sử dụng pin phải tính toán rất kĩ.

Ngoài ra ESP32 đang được rất nhiều các công ty trong và ngoài nước ưa chuộng.



*Hình 2.5 Sơ đồ kết nối chân của ESP32 với các linh kiện trong mạch*

### Khối cảm biến

1. Cảm biến khí gas MQ2

Cảm biến khí gas sử dụng phần tử SnO2 có độ dẫn điện thấp hơn trong không khí sạch, khi khí dễ cháy tồn tại, cảm biến có độ dẫn điện cao hơn, nồng độ chất dễ cháy càng cao thì độ dẫn điện của SnO2 sẽ càng cao và được tương ứng chuyển đổi thành mức tín hiệu điện.

**Cảm biến khí gas MQ-2** là cảm biến khí có độ nhạy cao với LPG, Propane và Hydrogen, mê-tan (CH4) và hơi dễ bắt lửa khác, với chi phí thấp và phù hợp cho các ứng dụng khác nhau.

Cảm biến xuất ra cả hai dạng tín hiệu là Analog và Digital, tín hiệu Digital có thể điều chỉnh mức báo bằng biến trở.

**Thông số kỹ thuật**

* Nguồn hoạt động: 5V
* Loại dữ liệu: Analog
* Phạm vi phát hiện rộng
* Tốc độ phản hồi nhanh và độ nhạy cao
* Mạch đơn giản
* Ổn định khi sử dụng trong thời gian dài

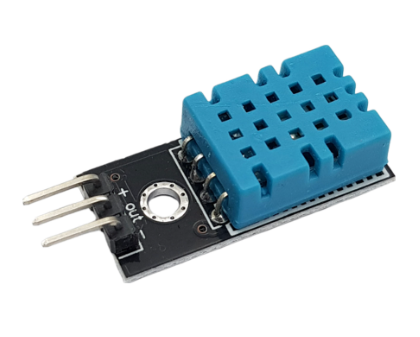
Bảng 2.2 Mô tả chân của cảm biến MQ2

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên chân** | **Mô tả** |
| 1 | Vcc | Đầu vào 5V |
| 2 | GND | Đầu vào chân đất |
| 3 | A0 | Chân tín hiệu analog |



Hình 2.6 Sơ đồ kết nối cảm biến MQ2 với vi điều khiển

1. Cảm biến nhiệt độ, độ ẩm DHT11



*Hình 2.7 Cảm biến nhiệt độ, độ ẩm DHT11*

Cảm biến DHT11 bao gồm một phần tử cảm biến độ ẩm điện dung và một điện trở nhiệt để cảm nhận nhiệt độ. Tụ điện cảm biến độ ẩm có hai điện cực với chất nền giữ ẩm làm chất điện môi giữa chúng. Thay đổi giá trị điện dung xảy ra với sự thay đổi của các mức độ ẩm. IC đo, xử lý các giá trị điện trở đã thay đổi này và chuyển chúng thành dạng kỹ thuật số.

Để đo nhiệt độ, cảm biến này sử dụng một nhiệt điện trở có hệ số nhiệt độ âm, làm giảm giá trị điện trở của nó khi nhiệt độ tăng. Để có được giá trị điện trở lớn hơn ngay cả đối với sự thay đổi nhỏ nhất của nhiệt độ, cảm biến này thường được làm bằng gốm bán dẫn hoặc polymer.

Cảm Biến Nhiệt Độ Và Độ Ẩm 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). 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. So với cảm biến đời mới hơn là DHT22 thì DHT11 cho khoảng đo và độ chính xác kém hơn rất nhiều.

**Thông số kỹ thuật DHT11**

 - Điện áp hoạt động: 3V - 5V DC

 - Dòng điện tiêu thụ: 2.5mA

 - Phạm vi cảm biến độ ẩm: 20% - 90% RH, sai số ±5%RH

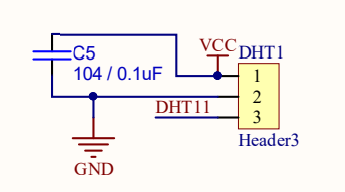
 - Phạm vi cảm biến nhiệt độ: 0°C ~ 50°C, sai số ±2°C

 - Tần số lấy mẫu tối đa: 1Hz (1 giây 1 lần)

 - Kích thước: 23 \* 12 \* 5 mm

Bảng 2.3 Mô tả các chân của DHT11

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên chân** | **Mô tả** |
| 1 | Vcc | Đầu vào nguồn cấp |
| 2 | DATA | Đầu tín hiệu |
| 3 | NC |  |
| 4 | GND | Chân đất |



Hình 2.8 Sơ đồ kết nối DHT11 với vi điều khiển

1. Cảm biến vân tay AS608



Hình 2.9 Cảm biến vân tay AS608

Cảm biến nhận dạng vân tay AS608 Fingerprint Sensor sử dụng giao tiếp UART TTL hoặc USB để giao tiếp với Vi điều khiển hoặc kết nối trực tiếp với máy tính ([thông qua mạch chuyển USB-UART](http://hshop.vn/products/mach-chuyen-usb-uart-cp2102) hoặc giao tiếp USB).

Cảm biến nhận dạng vân tay AS608 Fingerprint Sensor được tích hợp nhân xử lý nhận dạng vân tay phía trong, tự động gán vân tay với 1 chuỗi data và truyền qua giao tiếp UART ra ngoài nên hoàn toàn không cần các thao tác xử lý hình ảnh, đơn giản chỉ là phát lệnh đọc/ghi và so sánh chuỗi UART nên rất dễ sử dụng và lập trình.

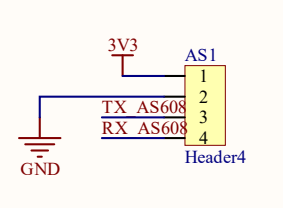
[**Cảm biến nhận dạng vân tay**](https://nshopvn.com/product/cam-bien-nhan-dang-van-tay/) có khả năng lưu nhiều vân tay cho 1 ID (1 người), thích hợp cho các ứng dụng bảo mật, khóa cửa, sinh trắc học,…

**Thông số kỹ thuật**

* Điệp áp hoạt động: 3.6 đến 6.0VDC
* Dòng điện tiêu thụ: <120mA
* Phương thức giao tiếp: UART
* Mức độ an toàn: 5
* Tỉ lệ chấp nhận sai (FAR): <0.001% (mức bảo mật 3)
* Tỉ lệ từ chối sai (FRR): <1.0% (mức bảo mật 3)
* Có thể lưu trữ được 127 dấu vân tay khác nhau

Bảng 2.4 Mô tả các chân của AS608

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên chân** | **Mô tả** |
| 1 | Vcc | Đầu vào nguồn cấp |
| 2 | TX | Chân cấp tín hiệu từ cảm biến đến vi điều khiển |
| 3 | RX | Chân nhận tín hiệu từ vi điều khiển |
| 4 | GND | Chân đất |



Hình 2.10 Sơ đồ kết nối AS608 với vi điều khiển

1. Bàn phím ma trận 4x4



*Hình 2.11 Bàn phím ma trận 4x4*

[Bàn phím ma trận mềm 4×4](https://nshopvn.com/product/ban-phim-ma-tran-mem-4x4/) nút được thiết kế với giao diện đơn giản giúp dễ dàng giao tiếp với bất kì vi điều khiển nào. Mặt sau dính thuận tiện để gắn bàn phím trong nhiều ứng dụng dự án.

Bàn phím 4×4 có tổng cộng 16 nút ở dạng Ma trận.

Đây là một bàn phím màng không có bộ phận chuyển động. Nó có một lớp phủ giống mô tả một bàn phím điện thoại với bốn nút chức năng bổ sung. Một đầu nối nổi 8 chân được cung cấp để kết nối nó với các mạch điều khiển mirocontler của bạn.

**Thông số bàn phím ma trận mềm 4×4**

* Module bàn phím ma trận 4×4 loại phím mềm.
* Độ dài cáp: 88mm.
* Nhiệt độ hoạt động 0 ~ 70oC.
* Đầu nối ra 8 chân.
* Kích thước bàn phím 77 x 69mm.​​​

### Khối chấp hành

1. Động cơ servo SG90



Hình 2.12 Động cơ servo SGG90

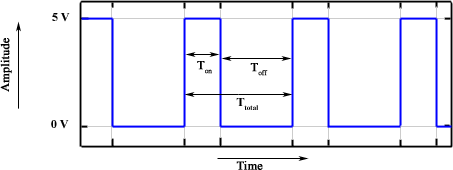
[Động cơ servo SG90](https://nshopvn.com/product/dong-co-servo-sg90-tower-pro/) có kích thước nhỏ, là loại được sử dụng nhiều nhất để làm các mô hình nhỏ hoặc các cơ cấu kéo không cần đến lực nặng.

Động cơ servo SG90 180 độ có tốc độ phản ứng nhanh, các bánh răng được làm bằng nhựa nên cần lưu ý khi nâng tải nặng vì có thể làm hư bánh răng, động cơ RC Servo 9G có tích hợp sẵn Driver điều khiển động cơ bên trong nên có thể dễ dàng điều khiển góc quay bằng phương pháp điều độ rộng xung PWM.

Pulse Width Modulation (PWM) là một trong những kỹ thuật thường được được sử dụng trong các hệ thống điều khiển ngày nay. PWM được sử dung rộng rãi trong các ứng dụng điều khiển như: điều khiển tốc độ, điều khiển nguồn điện, đo lường và truyền thông.

Điều chế độ rộng xung PWM là tạo ra một xung vuông có chu kỳ có thể thay

đổi được từ đó làm cho điện áp ra thay đổi như là một giá trị trung bình của dạng sóng Có thể biểu diễn nguyên tắc cơ bản của phương pháp điều chế độ rộng xung PWM thông qua các công thức sau:



Hình 2.13 Độ rộng xung

Xét một sóng vuông như hình trên:

Ton là khoảng thời gian đầu ra ở mức cao, Toff là khoảng thời gian đầu ra ở mức thấp, Ttotal là khoảng thời gian đạt được dạng sóng như vậy (chu kỳ).

Ttotal = Ton + Toff

Nhiệm vụ của một chu kỳ sóng vuông được định nghĩa là:

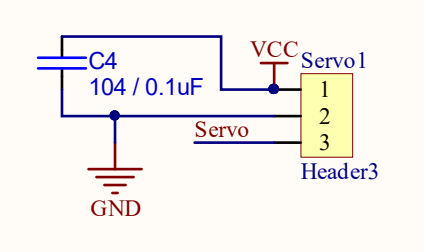
D = Ton / (Ton + Toff) = Ton / Ttotal

Điện áp đầu ra được tính như sau:

Vout = D.Vin = Vin.Ton / Ttotal

Bảng 2.5 Mô tả các chân của servo SG90

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên chân** | **Mô tả** |
| 1 | Vcc | Đầu vào nguồn cấp |
| 3 | PWM | Cấp xung để quay theo ý muốn |
| 4 | GND | Chân đất |



Hình 2.14 Sơ đồ kết nối servo SG90 với vi điều khiển

1. Quạt



Hình 2.15 Quạt 12V

Thông số kỹ thuật:

* Điện áp làm việc: 12VDC
* Dòng điện định mức: 0.2A
* Kích thước: 8 x 8 x 2.4cm
* Trọng lượng: 62g

Các phương pháp điều khiển tốc độ động cơ điện một chiều

Diagram, schematic

Description automatically generated

Hình 2.16 Sơ đồ nguyên lý của động cơ điện một chiều độc lập

Ứng với sơ đồ nguyên lý của động cơ một chiều như trên, ta có phương trình

đặc tính cơ như sau:

|  |  |  |
| --- | --- | --- |
|  |  | PT 2.1 |
|  | Trong đó: Uu, Ru là điện áp (V) và điện trở (Ω) phần ứng,  Rf là điện trở phụ (Ω),  KФ là từ thông (Wb),  ω là tốc độ góc (rad/s),  M là momen (N.m). | |

Từ phương trình đặc tính cơ (PT 2.1) của động cơ điện một chiều, ta thấy tốc

độ động cơ phụ thuộc vào các tham số Uu, Rf, KФ. Khi ta giữ nguyên momen

tải và thay đổi giá trị của một trong 3 tham số Uu, Rf, KФ thì ta sẽ được một

đường đặc tính cơ mới ứng với một tốc độ mới.*Điều chỉnh tốc độ bằng cách thay đổi điện trở mạch phần ứng*

Muốn thay đổi điện trở mạch phần ứng, ta nối thêm điện trở phụ Rf.

Diagram, schematic

Description automatically generated

Hình 2.17 Sơ đồ nguyên lý động cơ một chiều khi mắc thêm điện trở phụ

Nếu ta giữ nguyên điện áp phần ứng Uu = Uđm = const, và từ thông KФ = KФđm = const khi thay đổi điện trở phần ứng ta sẽ được:

* Tốc độ không tải: ,
* Độ cứng đặc tính cơ: .

Diagram

Description automatically generated

Hình 2.18 Đặc tính cơ của động cơ khi thay đổi điện trở phần ứng

Ta thấy khi Rf  càng lớn (β càng nhỏ) đặc tính cơ càng dốc. Do vậy phương

pháp này chỉ cho phép giảm tốc độ bằng cách tăng điện trở phần ứng. Trong

thực tế, khi thêm điện trở phụ vào mạch phần ứng sẽ gây ra một tổn hao công

suất rất lớn và không thể điều chỉnh trơn tốc độ nên phải điều chỉnh theo từng

cấp điện trở. Chính vì thế, phương pháp này không phổ biến như hai phương

pháp thay đổi điện áp phần ứng và từ thông kích từ.

* *Điều chỉnh tốc độ bằng cách thay đổi từ thông kích từ*

Giả thiết ta giữ nguyên điện áp phần ứng Uu = Udm = const, điện trở phần ứng

Ru = const, và thay đổi từ thông kích từ. Ta được:

* Tốc độ không tải: ,
* Độ cứng đặc tính cơ: .

Diagram

Description automatically generated

Hình 2.19 Đặc tính cơ của động cơ khi giảm từ thông

Đặc điểm của phương pháp này:

* Do cấu trúc của máy điện, nên thực tế chỉ sử dụng phương pháp này để điều chỉnh giảm từ thông. Khi giảm từ thông thì ω0x tăng dần (ω0 < ω01 < ω02 < …), độ cứng đặc tính cơ β giảm. Nên phương pháp này dùng để tăng tốc độ.
* Do việc điều chỉnh được thực hiện ở mạch kích từ, mạch này có dòng điện nhỏ hơn rất nhiều so với mạch lực, nên công suất tổn hao ít.
* Phương pháp này chịu ảnh của hiện tượng từ dư và các nhiễu làm ảnh hưởng xấu đến chất lượng của các hệ truyền động đảo chiều bằng kích từ.
* *Điều chỉnh tốc độ bằng cách thay đổi điện áp phần ứng*

Nếu ta giữ KФ = KФđm = const, Ru = const và thay đổi điện áp phần ứng theo hướng giảm so với Uđm thì ta được:

* Tốc độ không tải: ,
* Độ cứng đặc tính cơ: .

A picture containing chart

Description automatically generated

Hình 2.20 Đặc tính cơ của động cơ khi giảm điện áp phần ứng

Như vậy khi thay đổi điện áp đặt vào phần ứng động cơ, ta được một họ đặc tính cơ song song với đặc tính cơ tự nhiên và có độ cứng đặc tính cơ là không đổi.

Khi giảm điện áp phần ứng đặt vào động cơ thì dòng điện ngắn mạch sẽ giảm (), momen ngắn mạch của động cơ () cũng giảm. Và do đó tốc độ động cơ cũng giảm ứng với một phụ tải nhất định.

Phương pháp điều chỉnh tốc độ bằng cách thay đổi điện áp chỉ cho phép điều chỉnh dưới tốc độ định mức.

* *Kết luận*

Từ việc phân tích các phương pháp điều chỉnh tốc độ của động cơ điện một chiều, em thấy phương pháp điều chỉnh tốc độ bằng cách thay đổi điện áp phần ứng có rất nhiều ưu điểm (nổi bật nhất là độ cứng đặc tính cơ không thay đổi) phù hợp với động cơ công suất nhỏ, điều chỉnh tốc độ ở vùng dưới tốc độ định mức, momen tải không đổi trong toàn dải điều chỉnh.

1. Led



Hình 2.21 Led 5mm

### Khối hiển thị



Hình 2.22 LCD 1602

Màn hình text [LCD 1602 kèm module I2C](https://nshopvn.com/product/lcd-1602-kem-module-i2c/?variant=117000) sử dụng driver HD44780, có khả năng hiển thị 2 dòng với mỗi dòng 16 ký tự, màn hình có độ bền cao, rất phổ biến, nhiều code mẫu và dễ sử dụng thích hợp cho những người mới học và làm dự án.

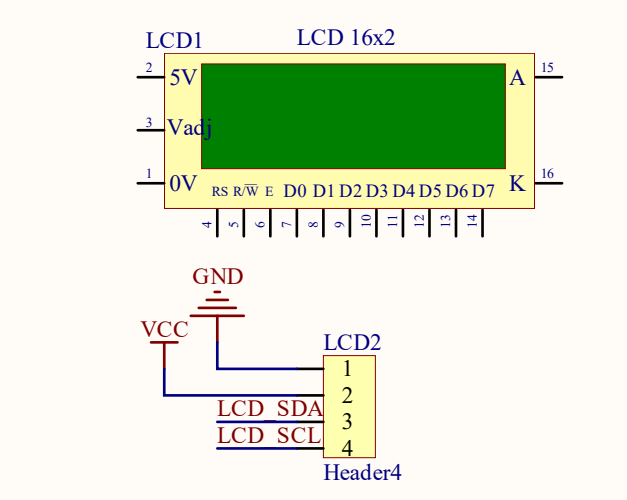
Màn hình LCD được hàn sẵn module giao tiếp I2C giúp việc giao tiếp được dễ dàng và nhanh chóng hơn rất nhiều, người dùng không phải tốn công hàn i2c, mà giá thành lại rẻ hơn mua từng món.

**Thông số kỹ thuật của LCD 1602 kèm module I2C**

* Điện áp hoạt động là 5 V.
* Địa chỉ I2C: 0x27 (có thể là 0X3F thay đổi theo đơn hàng của nhà sản xuất)
* Màu: Xanh lá hoặc Xanh dương (tùy chọn)
* Kích thước lỗ bắt ốc: 74mm x 30mm
* Kích thước của mạch: 80mm x 36mm x 19m
* Trọng lượng 38g

Bảng 2.6 Mô tả các chân của I2C kèm I2C

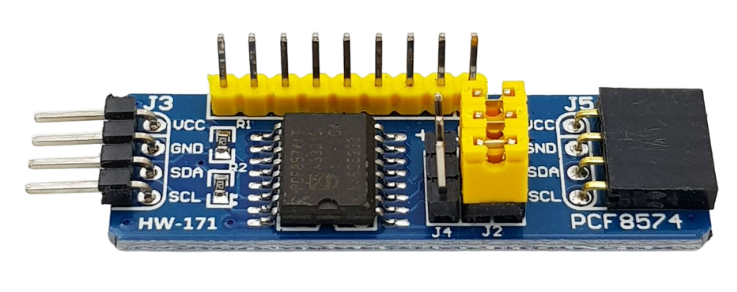
|  |  |  |
| --- | --- | --- |
| **STT** | **Tên chân** | **Mô tả** |
| 1 | Vcc | Đầu vào nguồn cấp |
| 2 | SDA | Đường truyền cho master và slave để gửi và nhận dữ liệu |
| 3 | SCL | Đường mang tín hiệu xung nhịp |
| 4 | GND | Chân đất |



Hình 2.23 Sơ đồ kết nối LCD I2C với vi điều khiển

### Các linh kiện khác

1. Mạch mở rộng I2C WaveShare PCFF8574



Hình 2.24 Mạch mở rộng PCF8574

[Mạch mở rộng I2C WaveShare PCF8574](https://nshopvn.com/product/mach-mo-rong-i2c-waveshare-pcf8574/). Khi giao tiếp các sản phẩm cần nhiều chân tín hiệu như bàn phím, led,… thì sẽ cần nhiều dây nhiều chân kết nối gây khó khăn trong quá trình làm việc. [Board mở rộng I2C WaveShare](https://youtu.be/MlvgxQLUtS8)PCF8574 sẽ giải quyết được sẽ giải quyết được vấn đề trên thông qua chuẩn giao tiếp I2C.

Chỉ cần 2 chân SCL và SDA có thể điều khiển được 8 chân tín hiệu.

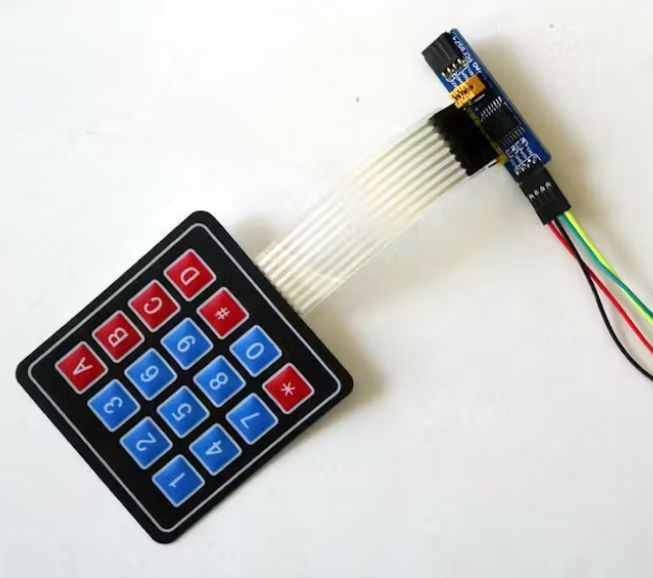
Ngoài ra có thể kết nối song song lên đến 8 board, lên đến 64 chân tín hiệu

**Thông số kỹ thuật**

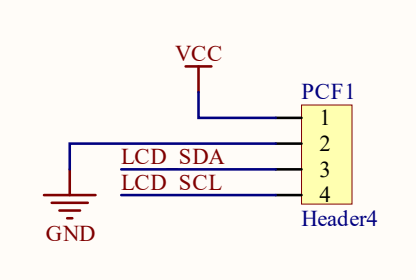
* Điện áp hoạt động: 3.6 – 5V
* Đầu vào: 2 chân I2C SCL & SDA
* Đầu ra: 8 chân I/O
* Kết nối song song 8 board liên tiếp: 64 chân tín hiệu I/O
* Kích thước: 48x16x15mm
* Trọng lượng: 5g

Bảng 2.7 Mô tả các chân của PCF8574

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên chân** | **Mô tả** |
| 1 | Vcc | Đầu vào nguồn cấp |
| 2 | SDA | Đường truyền cho master và slave để gửi và nhận dữ liệu |
| 3 | SCL | Đường mang tín hiệu xung nhịp |
| 4 | GND | Chân đất |



Hình 2.25 Kết nối ma trận 4x4 với PCF8574



Hình 2.26 Sơ đồ kết nối PCF8574 với vi điều khiển

1. Mạch hạ áp LM2596



*Hình 2.27 Mạch hạ áp LM2596*

[**Mạch giảm áp DC LM2596 3A**](https://nshopvn.com/product/mach-giam-ap-dc-lm2596-3a/) nhỏ gọn có khả năng giảm áp từ 30V xuống 1.5V mà vẫn đạt hiệu suất cao (92%) . Thích hợp cho các ứng dụng chia nguồn, hạ áp, cấp cho các thiết bị như camera, motor, robot,…

**Thông số kỹ thuật**

* Điện áp đầu vào: Từ 3V đến 30V.
* Điện áp đầu ra: Điều chỉnh được trong khoảng 1.5V đến 30V.
* Dòng đáp ứng tối đa là 3A.
* Hiệu suất: 92%
* Công suất: 15W
* Kích thước: 45 (dài) \* 20 (rộng) \* 14 (cao) mm

1. Adapter



Hình 2.28 Nguồn adapter 12V 5A

– Điện áp đầu vào: AC100-240V 50 / 60HZ  
– Đầu cắm AC: chuẩn Hoa Kỳ  
– Điện áp đầu ra:  DC12V  
– Dòng đầu ra tối đa: 5A  
– Tổng chiều dài nguồn ~ 1m5  
– Jack cắm DC : 5.5\*2.5mm (tương thích 5.5 \* 2.1mm)

1. Laptop

Giao tiếp UART (Universal Asynchronous Receiver/Transmitter) là một phương thức truyền dữ liệu nối tiếp (serial communication) phổ biến và cơ bản được sử dụng trong nhiều hệ thống vi điều khiển và các thiết bị điện tử. UART không yêu cầu đồng bộ hóa thời gian giữa hai thiết bị gửi và nhận, và truyền tải dữ liệu qua một kênh duy nhất, tiết kiệm băng thông và dễ dàng triển khai.

**Các thành phần chính của UART:**

UART gồm hai thành phần chính:

1. Transmitter (TX): Phần truyền tín hiệu, thường được kết nối với chân TX trên vi điều khiển hoặc thiết bị.
2. Receiver (RX): Phần nhận tín hiệu, kết nối với chân RX trên vi điều khiển hoặc thiết bị.

Giao tiếp giữa các thiết bị qua UART chỉ yêu cầu hai dây chính: một dây truyền (TX) và một dây nhận (RX). Tuy nhiên, để đảm bảo tín hiệu ổn định, có thể cần một số dây phụ như Ground (GND).

**Cấu trúc truyền dữ liệu trong UART:**

Khi dữ liệu được truyền đi qua UART, nó sẽ được chia thành các khối dữ liệu nhỏ gọi là bít (bit). Một khung dữ liệu (data frame) trong UART thường bao gồm các thành phần sau:

1. Start Bit: Bắt đầu một khung dữ liệu. Mỗi lần truyền dữ liệu, sẽ có một start bit (thường là 0).
2. Data Bits: Các bit dữ liệu thực tế được truyền. Số lượng bit dữ liệu thường là 5, 6, 7 hoặc 8 bit, tùy theo cấu hình.
3. Parity Bit: Một bit kiểm tra lỗi tùy chọn, có thể là None (không sử dụng), Even (kiểm tra lỗi chẵn) hoặc Odd (kiểm tra lỗi lẻ).
4. Stop Bit: Kết thúc một khung dữ liệu. Thường là 1 bit hoặc 2 bit, giúp nhận biết khi khung dữ liệu kết thúc.

**Cách thức hoạt động:**

* Gửi dữ liệu: Dữ liệu được truyền từ bộ phát (transmitter) qua dây TX, rồi tới bộ thu (receiver) qua dây RX. Quá trình truyền diễn ra một cách tuần tự (một bit tại một thời điểm).
* Nhận dữ liệu: Thiết bị nhận (receiver) đọc từng bit dữ liệu, tái tạo lại byte hoặc khối dữ liệu ban đầu từ các bit này.

**Đặc điểm của UART:**

* Asynchronous: Giao tiếp UART không yêu cầu tín hiệu đồng bộ (clock signal) giữa thiết bị gửi và nhận, tức là thiết bị nhận phải biết tốc độ (baud rate) mà thiết bị gửi sử dụng để có thể giải mã đúng dữ liệu.
* Tốc độ Baud Rate: Baud rate xác định tốc độ truyền dữ liệu, được đo bằng số bit trên giây (bps). Cả hai thiết bị gửi và nhận phải sử dụng cùng một baud rate để đảm bảo đồng bộ.
* Một chiều hoặc hai chiều: Mặc dù UART sử dụng hai dây (TX và RX), nó có thể hoạt động theo chế độ một chiều (half-duplex) hoặc hai chiều (full-duplex), tùy thuộc vào thiết kế của hệ thống.

**Các tín hiệu trong giao tiếp UART:**

* TX (Transmit): Chân truyền dữ liệu từ thiết bị gửi đến thiết bị nhận.
* RX (Receive): Chân nhận dữ liệu từ thiết bị gửi tới thiết bị nhận.
* GND (Ground): Dây nối đất giữa các thiết bị.

**Các ứng dụng phổ biến của UART:**

1. Giao tiếp với các vi điều khiển: UART thường được sử dụng để giao tiếp giữa vi điều khiển (như Arduino, ESP32) và các thiết bị ngoại vi.
2. Giao tiếp với máy tính: Các thiết bị USB-to-UART (chẳng hạn như USB-to-Serial converter) cho phép máy tính giao tiếp với các vi điều khiển qua cổng COM.
3. Debugging: UART thường được sử dụng để in thông tin debug, đặc biệt trong các hệ thống nhúng, giúp lập trình viên theo dõi và kiểm tra trạng thái hệ thống.
4. Giao tiếp với cảm biến, mô-đun ngoại vi: UART được sử dụng để giao tiếp với các mô-đun như GPS, Bluetooth, Wi-Fi, hay các cảm biến có giao diện nối tiếp.

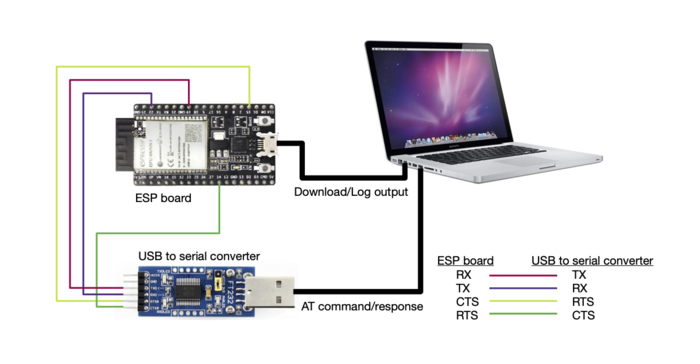
**Ưu và nhược điểm của UART:**

**Ưu điểm:**

* Đơn giản: Cấu trúc giao tiếp UART khá đơn giản và dễ triển khai.
* Tiết kiệm băng thông: Giao tiếp UART chỉ yêu cầu hai dây chính, giúp giảm thiểu số lượng dây nối.
* Không cần đồng bộ hóa: Không yêu cầu tín hiệu đồng hồ ngoài, giúp giảm chi phí và phức tạp phần cứng.

**Nhược điểm:**

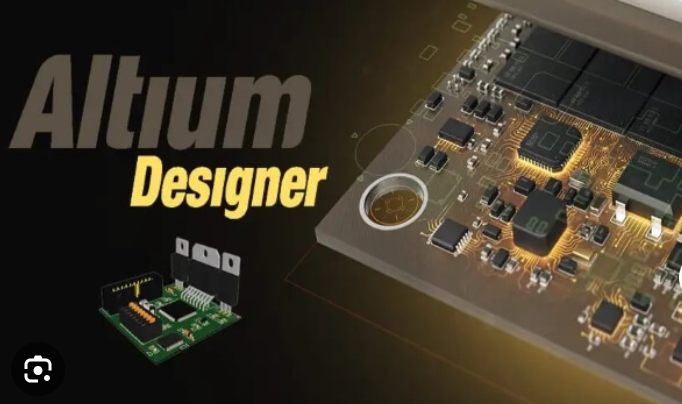
* Khoảng cách truyền ngắn: Giao tiếp UART thường chỉ hiệu quả trong khoảng cách ngắn (vài mét).
* Không đa điểm: UART không thể dễ dàng kết nối nhiều thiết bị với nhau (khác với I2C hoặc SPI).
* Tốc độ truyền có hạn: Tốc độ truyền dữ liệu của UART phụ thuộc vào baud rate, và tốc độ truyền có thể bị giới hạn khi sử dụng khoảng cách dài hoặc truyền tải nhiều dữ liệu.



Hình 2.29 Giao tiếp UART giữa Laptop và Esp32

## Thiết kế mạch in

Mạch in được em thiết kế trên phần mềm Altium Designer



Hình 2.30 Phần mềm Altium Designer

Altium Designer trước kia có tên gọi quen thuộc là Protel DXP, là một trong những công cụ vẽ mạch điện tử mạnh nhất hiện nay. Được phát triển bởi hãng Altium Limited. Altium designer là một phần mềm chuyên nghành được sử dụng trong thiết kế mạch điện tử. Nó là một phần mềm mạnh với nhiều tính năng thú vị, tuy nhiên phần mềm này còn được ít người biết đến so với các phần mềm thiết kế mạch khác như orcad hay proteus.

Altium Designer có một số đặc trưng sau:

- Giao diện thiết kế, quản lý và chỉnh sửa thân thiện, dễ dàng biên dịch, quản lý file, quản lý phiên bản cho các tài liệu thiết kế.

- Hỗ trợ mạnh mẽ cho việc thiết kế tự động, đi dây tự động theo thuật toán tối ưu, phân tích lắp ráp linh kiện. Hỗ trợ việc tìm các giải pháp thiết kế hoặc chỉnh sửa mạch, linh kiện, netlist có sẵn từ trước theo các tham số mới.

- Mở, xem và in các file thiết kế mạch dễ dàng với đầy đủ các thông tin linh kiện, netlist, dữ liệu bản vẽ, kích thước, số lượng…

- Hệ thống các thư viện linh kiện phong phú, chi tiết và hoàn chỉnh bao gồm tất cả các linh kiện nhúng, số, tương tự…

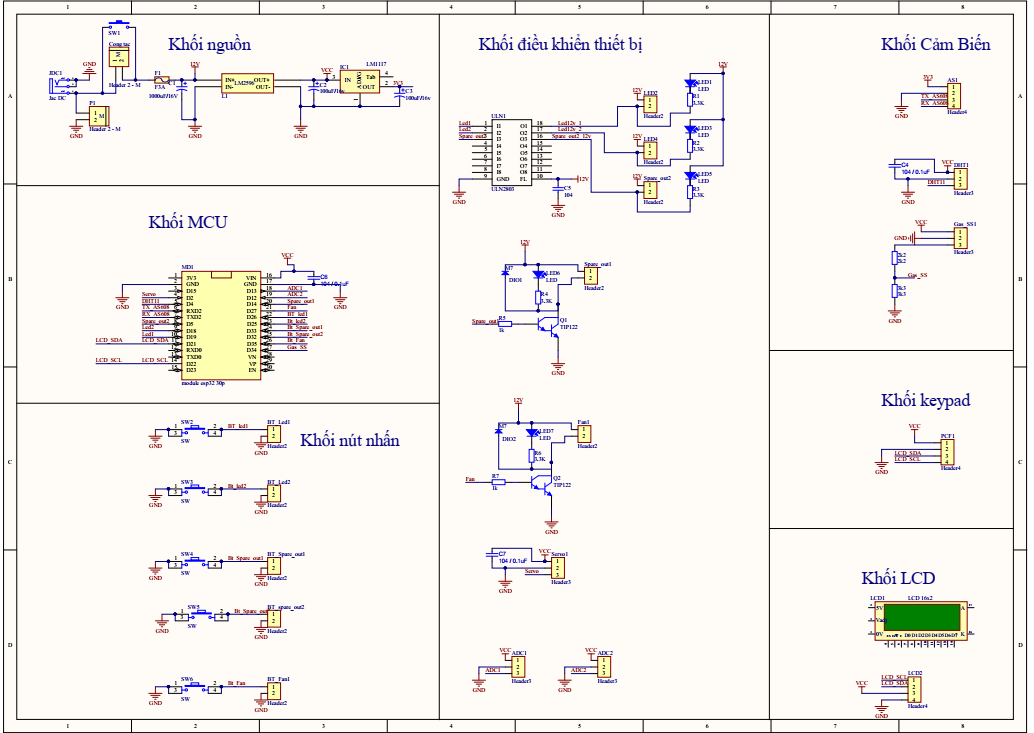
 - Đặt và sửa đối tượng trên các lớp cơ khí, định nghĩa các luật thiết kế, tùy chỉnh các lớp mạch in, chuyển từ schematic sang PCB, đặt vị trí linh kiện trên PCB.

- Mô phỏng mạch PCB 3D, đem lại hình ảnh mạch điện trung thực trong không gian 3 chiều, hỗ trợ MCAD-ECAD, liên kết trực tiếp với mô hình STEP, kiểm tra khoảng cách cách điện, cấu hình cho cả 2D và 3D

- Hỗ trợ thiết kế PCB sang FPGA và ngược lại.

       Từ đó, chúng ta thấy Altium designer có nhiều điểm mạnh so với các phần mềm  khác như đặt luật thiết kế, quản lý đề tài mô phỏng dễ dàng, giao diện thân thiện,…

Sau khi thiết kế xong các khối liên quan đến vi điều khiển, em đưa ra thiết kế mạch in chi tiết cho phần cứng như sau

****

Hình 2.31 Sơ đồ kết nối chi tiết

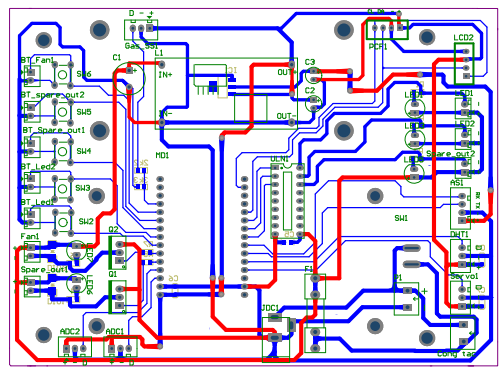
Dòng điện tiêu thụ của khối nguồn 5V là khoảng 0.4A nên em sử dụng IC nguồn xung LM2576 – 5V tần số 52kHz với khả năng chịu dòng lên đến 3A để tăng hiệu suất và giảm nhiệt lượng tiêu hao. IC LM2576 – 5V hoạt động dựa trên nguyên lý mạch biến đổi giảm áp một chiều (DC/DC Buck Converter). IC LM2576 – 5V có thể nhận điện áp đầu vào từ 7V đến 40V và có thể điều chỉnh hạ áp xuống mức điện áp 5V.

Mạch in được em thiết kế trên phần mềm Altium với một số luật như sau:

Khoảng cách giữa các đường dây tối thiểu là 10mil ≈ 0.254mm (1mil ≈ 0.0254mm).

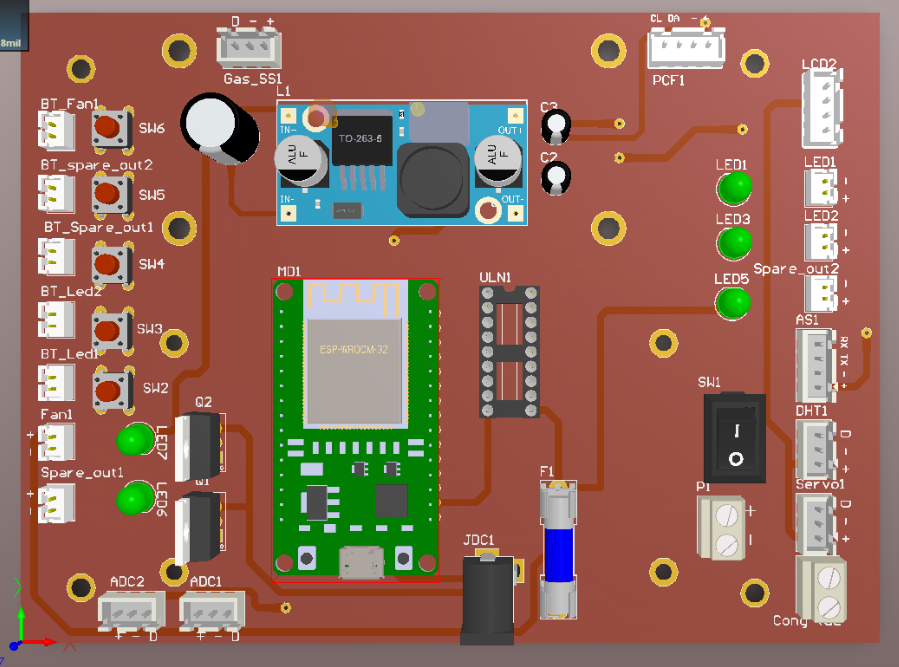
Độ rộng đường dây tối thiểu là 10mil (dành cho các đường dây tín hiệu có công suất thấp). Đối với các đường cấp nguồn (5V, 3.3V) độ rộng của dây là 30 mil. Còn đối với đường nguồn có công suất lớn em sẽ phủ đồng để đảm bảm sự ổn định cũng như tản nhiệt tốt cho đường nguồn.

Tại những vị trí phủ đất, em thêm các lỗ via để nối đất ở hai mặt.

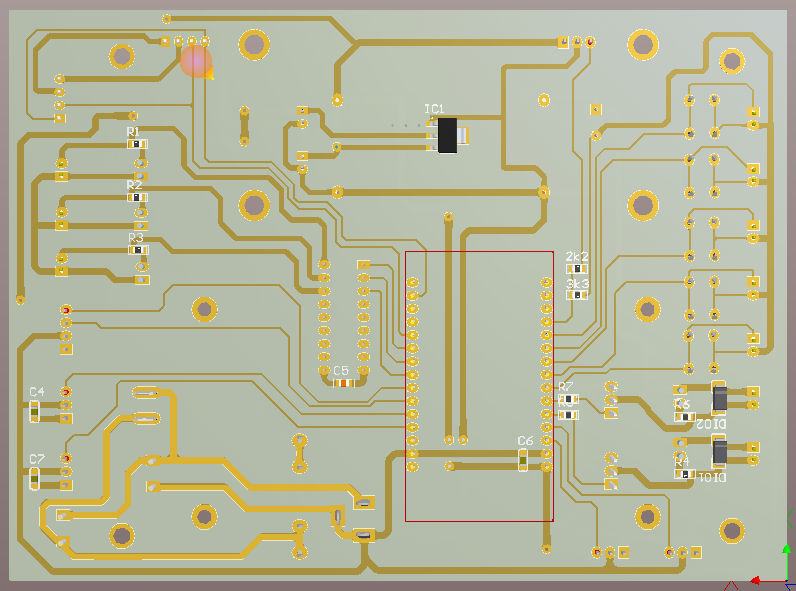
**

Hình 2.32 Hình ảnh 2D của mạch in

Sau khi thiết kế xong, em được một mạch in như *Hình* 2.33 Hình ảnh 3D mặt trước của mạch in và *Hình 2.34 Hình ảnh 3D mặt sau của mạch in*.

**

Hình 2.33 Hình ảnh 3D mặt trước của mạch in

**

*Hình 2.34 Hình ảnh 3D mặt sau của mạch in*

# THIẾT KẾ PHẦN MỀM

## Nguyên lý hoạt động

Qua các chương trên, chúng ta cũng đã có một cái nhìn tổng quan về nhà thông minh sử dụng các thiết bị cảm biến để có thể điều khiển các thiết bị đầu ra từ xa. Sau đây, em xin phép trình bày về nguyên lý hoạt động từng chức năng trong ngôi nhà thông minh

* Mở cửa bằng pass số, vân tay hoặc xử lý ảnh: Khi người dùng nhập pass số hoặc đặt tay lên cảm biến vân tay, hệ thống sẽ kiểm tra thông tin và mở khóa cửa tự động bằng cách sử dụng servo để mô phỏng đóng mở cửa điện tử, ngoài ra sử dụng xử lý ảnh để nhận diện đúng người đã traning để tự động mở cửa
* Nguồn dự phòng: Hệ thống được trang bị nguồn dự phòng để đảm bảo rằng trong trường hợp mất điện, các chức năng cơ bản như mở khóa cửa, giám sát nhiệt độ, và các tính năng khác vẫn có thể hoạt động.
* Sử dụng servo mô phỏng đóng mở cửa điện tử: Servo được sử dụng để điều khiển cơ cấu cửa điện tử, mở và đóng cửa dựa trên lệnh từ hệ thống điều khiển.
* Sử dụng app Blynk để giám sát nhiệt độ, độ ẩm, phòng ngủ, phòng khách: Hệ thống sử dụng cảm biến để đo đạc nhiệt độ và độ ẩm trong các phòng. Thông tin này được truyền đến ứng dụng trên điện thoại người dùng để họ có thể giám sát và điều chỉnh môi trường sống theo ý muốn.
* Hẹn giờ bật tắt điều hòa và quạt: Người dùng có thể lên lịch trên ứng dụng điện thoại để tự động bật hoặc tắt thiết bị điều hòa hoặc quạt tại các thời điểm cụ thể.
* Bật tắt đèn qua ứng dụng: Người dùng có khả năng điều khiển bật tắt đèn trong nhà thông minh thông qua ứng dụng trên điện thoại.
* Giám sát nồng độ khí gas và xử lý sự cố: Hệ thống sử dụng cảm biến để giám sát nồng độ khí gas. Nếu nồng độ vượt ngưỡng an toàn, hệ thống sẽ thông báo cảnh báo còi và gửi thông báo lên ứng dụng điện thoại. Đồng thời, có thể kích hoạt quạt hút để loại bỏ khí độc hại ra khỏi nhà thông minh.

## Phần mềm lập trình Arduino IDE

Arduino IDE là một phần mềm mã nguồn mở chủ yếu được sử dụng để viết và biên dịch mã vào module Arduino.

Đây là một phần mềm Arduino chính thức, giúp cho việc biên dịch mã trở nên dễ dàng mà ngay cả một người bình thường không có kiến thức kỹ thuật cũng có thể làm được.

Nó có các phiên bản cho các hệ điều hành như MAC, Windows, Linux và chạy trên nền tảng Java đi kèm với các chức năng và lệnh có sẵn đóng vai trò quan trọng để gỡ lỗi, chỉnh sửa và biên dịch mã trong môi trường.

Có rất nhiều các module Arduino như Arduino Uno, Arduino Mega, Arduino Leonardo, Arduino Micro và nhiều module khác. Mỗi module chứa một bộ vi điều khiển trên bo mạch được lập trình và chấp nhận thông tin dưới dạng mã.

Mã chính, còn được gọi là sketch, được tạo trên nền tảng IDE sẽ tạo ra một file Hex, sau đó được chuyển và tải lên trong bộ điều khiển trên bo.

Môi trường IDE chủ yếu chứa hai phần cơ bản: Trình chỉnh sửa và Trình biên dịch, phần đầu sử dụng để viết mã được yêu cầu và phần sau được sử dụng để biên dịch và tải mã lên module Arduino.Môi trường này hỗ trợ cả ngôn ngữ C và C ++.

Khi người dùng viết mã và biên dịch, IDE sẽ tạo file Hex cho mã. File Hex là các file thập phân Hexa được Arduino hiểu và sau đó được gửi đến bo mạch bằng cáp USB. Mỗi bo Arduino đều được tích hợp một bộ vi điều khiển, bộ vi điều khiển sẽ nhận file hex và chạy theo mã được viết.



*Hình 3.1 Giao diện Arduino IDE*

Arduino IDE bao gồm các phần khác nhau:

1. Window bar

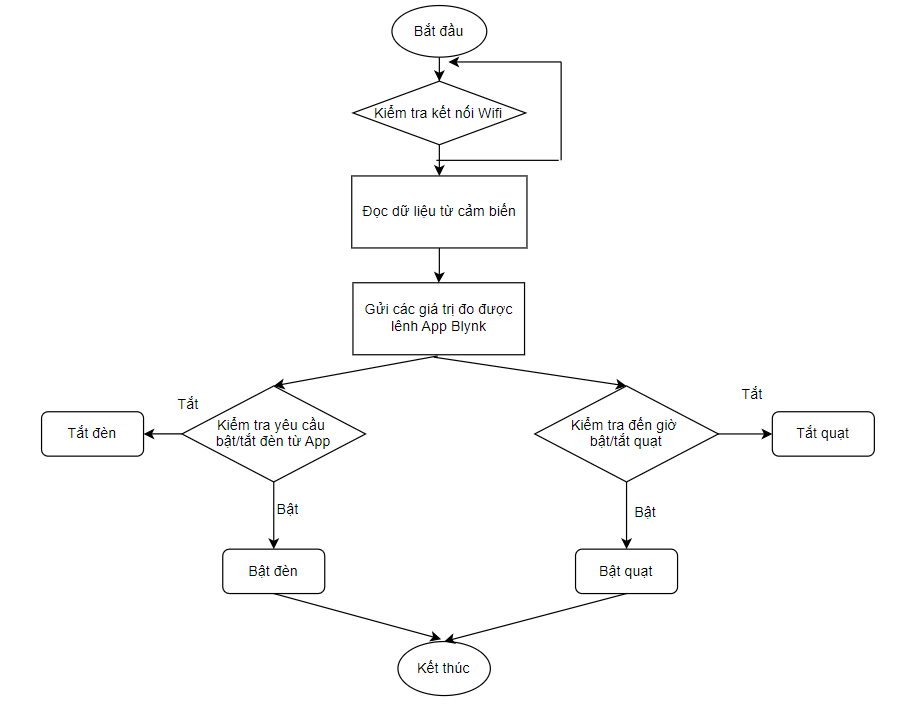
2. Menu bar

3. Phím tắt

4. Text Editor

5. Output Panel

## Lưu đồ thuật toán của Blynk và ESP32



Hình 3.2 Lưu đồ thuật toán của ESP32 và Blynk

Dưới đây là giải thích chi tiết về lưu đồ thuật toán giữa Blynk và ESP32

1. Bắt đầu - Đây là điểm khởi đầu của quy trình. Hệ thống bắt đầu hoạt động từ đây.

2. Kiểm tra kết nối WiFi

- Bước đầu tiên sau khi khởi động là kiểm tra xem thiết bị có kết nối WiFi không.

- Nếu kết nối thành công, hệ thống sẽ tiếp tục đến bước tiếp theo.

- Nếu không kết nối được, hệ thống sẽ đợi cho đến khi có kết nối.

3. Đọc dữ liệu từ cảm biến

- Sau khi đã kết nối với WiFi, hệ thống sẽ tiến hành đọc dữ liệu từ các cảm biến.

- Các cảm biến này bao gồm cảm biến nhiệt độ, độ ẩm, cảm biến khí gas,...

4. Gửi các giá trị đo được lên App Blynk

- Dữ liệu thu thập được từ các cảm biến sẽ được gửi lên ứng dụng Blynk.

- Ứng dụng Blynk cho phép người dùng giám sát và điều khiển các thiết bị từ xa.

5. Kiểm tra yêu cầu bật/tắt đèn từ App

- Hệ thống kiểm tra xem có yêu cầu bật hoặc tắt đèn từ ứng dụng Blynk không.

- Nếu có yêu cầu bật đèn, hệ thống sẽ thực hiện việc bật đèn.

- Nếu có yêu cầu tắt đèn, hệ thống sẽ thực hiện việc tắt đèn.

- Nếu không có yêu cầu gì, hệ thống sẽ tiếp tục đến bước tiếp theo.

6. Kiểm tra đến giờ bật/tắt quạt

- Hệ thống kiểm tra xem có lịch hẹn giờ bật hoặc tắt quạt từ ứng dụng Blynk không.

- Nếu đến giờ bật quạt, hệ thống sẽ thực hiện việc bật quạt.

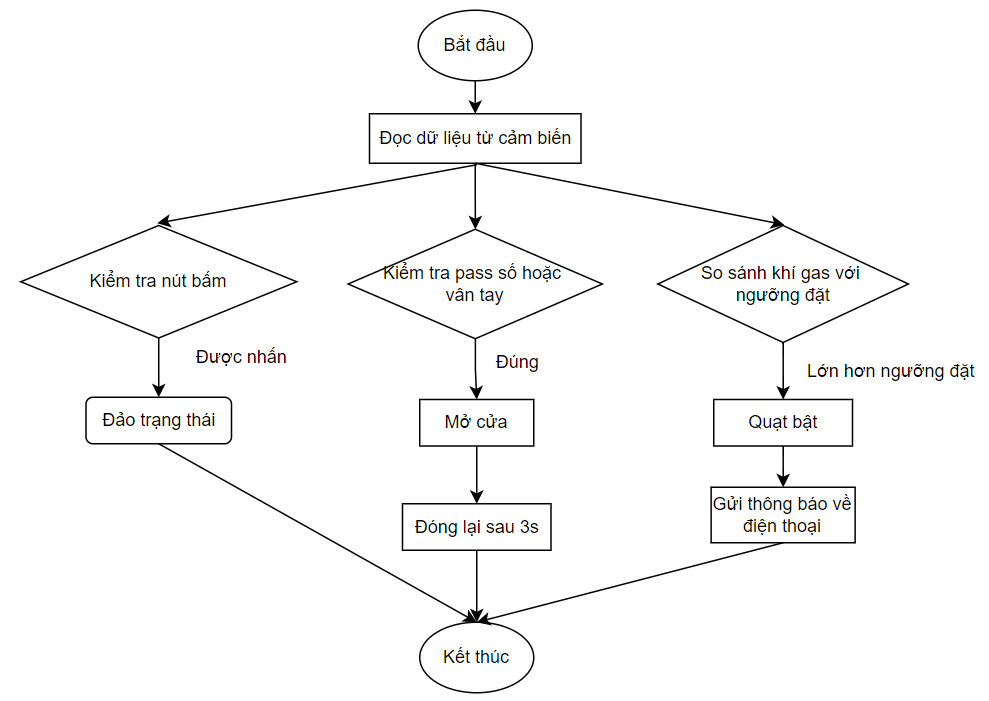
- Nếu đến giờ tắt quạt, hệ thống sẽ thực hiện việc tắt quạt.

- Nếu không có yêu cầu gì, hệ thống sẽ tiếp tục đến bước tiếp theo.

7. Kết thúc - Đây là điểm kết thúc của quy trình. Sau khi hoàn tất các kiểm tra và thực hiện các yêu cầu từ ứng dụng Blynk, hệ thống sẽ dừng lại.

Tổng Kết Lưu đồ này mô tả cách hệ thống ESP32 kết nối với WiFi, đọc dữ liệu từ cảm biến, gửi dữ liệu lên ứng dụng Blynk, và thực hiện các yêu cầu bật/tắt đèn, quạt từ người dùng. Mục tiêu chính là tạo ra một hệ thống nhà thông minh có khả năng giám sát và điều khiển từ xa, đảm bảo tiện nghi và an toàn cho người sử dụng.

## Lưu đồ thuật toán của vi điều khiển



Hình 3.3 Lưu đồ thuật toán của vi điều khiển

1. Bắt đầu

Đây là điểm khởi đầu của quy trình. Hệ thống bắt đầu hoạt động từ đây.

2. Đọc dữ liệu từ cảm biến

Sau khi khởi động, hệ thống tiến hành đọc dữ liệu từ các cảm biến. Các cảm biến này có thể bao gồm cảm biến vân tay, cảm biến khí gas, cảm biến nhiệt độ, độ ẩm, v.v.

3. Kiểm tra nút bấm

Hệ thống kiểm tra xem nút bấm có được nhấn hay không.

Nếu nút bấm được nhấn: Hệ thống sẽ thực hiện hành động đảo trạng thái, tức là thay đổi trạng thái hiện tại (ví dụ từ bật sang tắt hoặc ngược lại).

4. Kiểm tra pass số hoặc vân tay

Hệ thống kiểm tra thông tin pass số hoặc vân tay của người dùng.

Nếu thông tin đúng: Hệ thống sẽ mở cửa. Sau khi mở cửa, hệ thống sẽ đợi 3 giây rồi đóng cửa lại.

5. So sánh khí gas với ngưỡng đặt

Hệ thống kiểm tra nồng độ khí gas trong nhà và so sánh với ngưỡng an toàn đã đặt trước.

Nếu nồng độ khí gas lớn hơn ngưỡng đặt: Hệ thống sẽ bật quạt để giảm nồng độ khí gas trong nhà, đồng thời gửi thông báo về điện thoại người dùng để cảnh báo về nguy cơ tiềm ẩn.

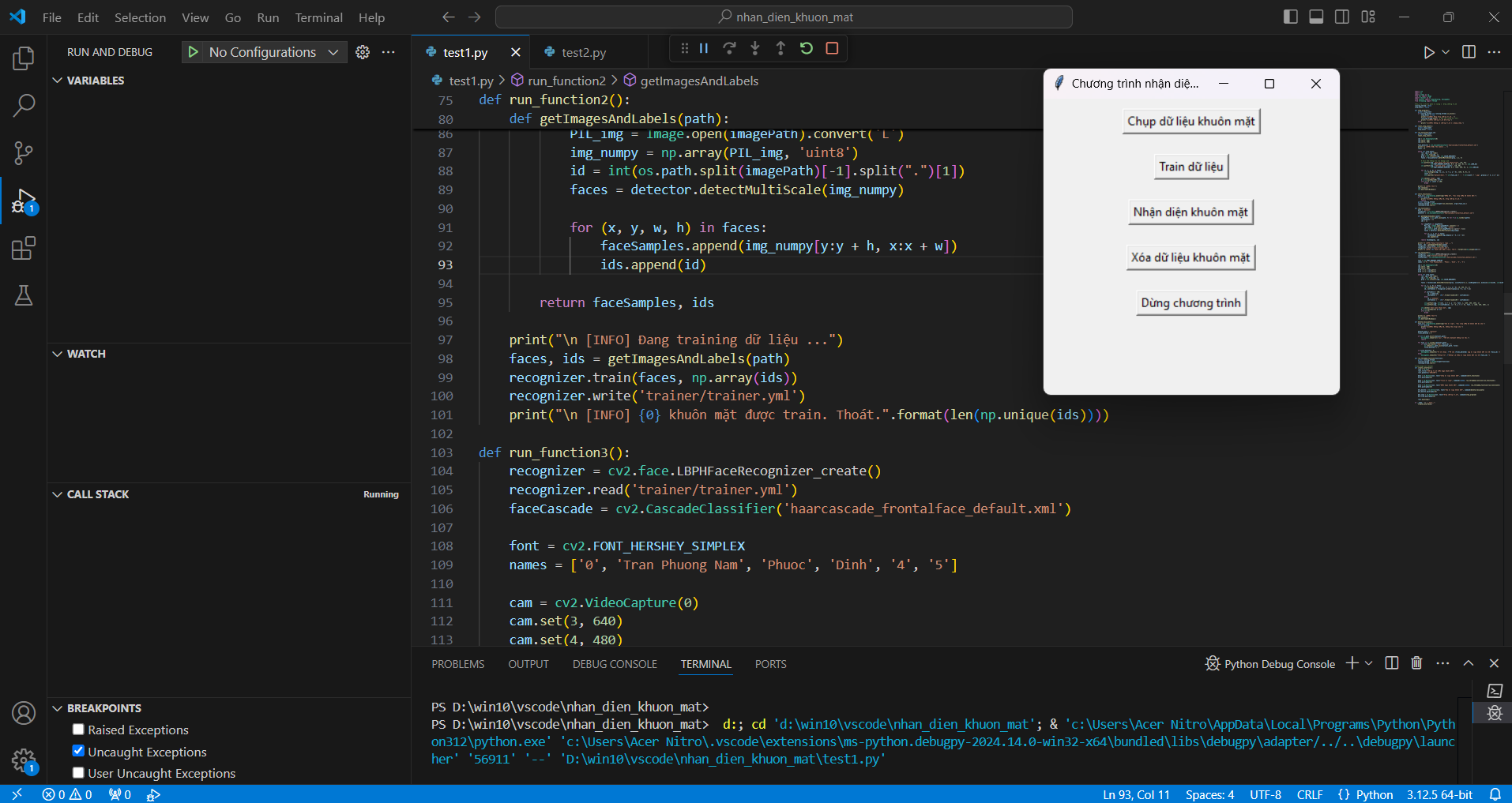
6. Kết thúc

Đây là điểm kết thúc của quy trình. Sau khi hoàn tất các kiểm tra và thực hiện các hành động cần thiết, hệ thống sẽ dừng lại.

Tổng Kết

Lưu đồ này mô tả cách hệ thống vi điều khiển trong nhà thông minh hoạt động, bao gồm kiểm tra nút bấm, kiểm tra pass số hoặc vân tay để mở cửa, và giám sát nồng độ khí gas. Mục tiêu là đảm bảo an toàn và tiện nghi cho người sử dụng thông qua các biện pháp tự động và thông báo kịp thời.

## Thuật toán xử lý ảnh



Hình 3.4 Xử lý ảnh nhận diện khuôn mặt

**Thuật toán sử dụng cho xử lý ảnh**

1. Chụp dữ liệu khuôn mặt (Run Function 1)

Thuật toán phát hiện khuôn mặt:

* Đoạn mã sử dụng cv2.CascadeClassifier với bộ phân loại Haar để phát hiện khuôn mặt trong hình ảnh.
* Sau khi khởi tạo camera, mã lặp qua các khung hình từ camera, chuyển đổi chúng thành ảnh xám (grayscale) và sử dụng detectMultiScale() để phát hiện các khuôn mặt.
* Nếu phát hiện được khuôn mặt, mã sẽ cắt các vùng mặt và lưu chúng vào thư mục dataset với tên theo định dạng User.face\_id.count.jpg.
* Việc này được lặp lại cho đến khi có 100 ảnh được chụp hoặc nhấn phím ESC (mã 27).

Điểm cần chú ý:

* Phương pháp này sử dụng thuật toán Haar Cascade khá cơ bản và có thể không đạt được độ chính xác cao trong môi trường ánh sáng phức tạp. Tuy nhiên, nó đủ để thực hiện nhận diện khuôn mặt trong các điều kiện thông thường.

2. Huấn luyện dữ liệu khuôn mặt (Run Function 2)

Thuật toán huấn luyện:

* Sau khi thu thập đủ dữ liệu khuôn mặt, đoạn mã sẽ sử dụng cv2.face.LBPHFaceRecognizer\_create() để huấn luyện mô hình nhận diện khuôn mặt.
* getImagesAndLabels() giúp lấy các ảnh đã được lưu trong thư mục dataset, trích xuất các đặc trưng khuôn mặt từ mỗi ảnh và gán nhãn cho mỗi ảnh (ID của người).
* Mô hình LBPH (Local Binary Patterns Histograms) là một trong những phương pháp nhận diện khuôn mặt phổ biến, đặc biệt là khi dữ liệu huấn luyện không quá phức tạp và môi trường không quá thay đổi.

Điểm cần chú ý:

* LBPHFaceRecognizer phù hợp với những ứng dụng nhận diện khuôn mặt trong môi trường ánh sáng ổn định và có ít sự thay đổi về góc nhìn hoặc các yếu tố khác. Nếu môi trường thay đổi nhiều, bạn có thể phải xem xét các mô hình phức tạp hơn như CNN hoặc DNN.

3. Nhận diện khuôn mặt trong thời gian thực (Run Function 3)

Thuật toán nhận diện khuôn mặt:

* Trong phần này, mã tiếp tục sử dụng cv2.CascadeClassifier để phát hiện khuôn mặt trong thời gian thực.
* Sau khi phát hiện khuôn mặt, hệ thống sử dụng mô hình đã huấn luyện (LBPH) để dự đoán ID của khuôn mặt và độ tin cậy (confidence) của dự đoán.
* Nếu độ tin cậy cao (confidence < 100), hệ thống gửi tín hiệu qua cổng serial để mở cửa.
* Nếu độ tin cậy thấp, hệ thống sẽ báo "unknown".

**Các chức năng của giao diện xử lý ảnh**

Khởi tạo giao diện GUI: Đây là bước đầu tiên khi chương trình khởi động. Giao diện cung cấp các nút để người dùng chọn các chức năng khác nhau.

Chụp dữ liệu khuôn mặt:

* Chương trình sẽ yêu cầu người dùng nhập một ID cho khuôn mặt, mở camera và bắt đầu quay video.
* Nó sẽ liên tục phát hiện khuôn mặt và lưu ảnh khuôn mặt vào thư mục dataset với tên "User.ID.number.jpg", nơi ID là ID người dùng nhập và number là số thứ tự ảnh được chụp.

Train dữ liệu:

* Sau khi chụp đủ ảnh, người dùng có thể chọn "Train dữ liệu", chương trình sẽ sử dụng các ảnh đã chụp để huấn luyện mô hình nhận diện khuôn mặt. Mô hình được lưu vào file trainer/trainer.yml.

Nhận diện khuôn mặt:

* Chương trình tiếp tục mở camera để nhận diện khuôn mặt. Nếu khuôn mặt được nhận diện và có độ tin cậy cao (confidence < 100), chương trình sẽ gửi tín hiệu "ok" qua cổng COM cho ESP32.

Xóa dữ liệu khuôn mặt:

* Người dùng có thể yêu cầu xóa dữ liệu khuôn mặt bằng cách nhập ID và tìm các tệp tin dữ liệu khuôn mặt trong thư mục dataset có ID tương ứng để xóa.

Dừng chương trình:

* Khi người dùng muốn dừng chương trình, tất cả các luồng đang chạy sẽ bị dừng và các tài nguyên sẽ được giải phóng.

## Blynk IoT

### Giới thiệu về App Blynk

Blynk là một nền tảng IoT cung cấp các công cụ để kết nối, quản lý và điều khiển các thiết bị IoT từ xa thông qua mạng Internet. Điểm nổi bật của Blynk là sự dễ dàng và nhanh chóng trong việc tạo và quản lý các ứng dụng IoT, phù hợp cho cả những người mới bắt đầu và những nhà phát triển chuyên nghiệp.

Blynk cung cấp ứng dụng di động và API cho phép người dùng kết nối và điều khiển các thiết bị IoT bằng cách sử dụng các cảm biến và các tín hiệu đầu vào. Giao diện người dùng được tùy chỉnh linh hoạt để điều khiển thiết bị IoT theo cách tùy chỉnh và tạo ra các hành động và tương tác phức tạp thông qua mã code.

App Blynk hỗ trợ các nền tảng phần cứng, bao gồm Arduino, Raspberry Pi, ESP8266, ESP32 và nhiều nền tảng khác nữa. Điều này cho phép người dùng dễ dàng tích hợp các thiết bị IoT vào các dự án của mình mà không cần có nhiều kinh nghiệm lập trình.

****

Hình 3.5 Blynk Iot

### Cách hoạt động của Blynk

Blynk có thể điều khiển các I/O analog hoặc digital trên hardware trực tiếp và giữa Blynk App và hardware có thể trao đổi dữ liệu qua Virtual Pin. Nhưng như vậy là chưa đủ. Blynk còn thiết kế thêm Virtual pin (chân I/O ảo). Virtual Pin được sử dụng để gửi dữ liệu từ bộ vi điều khiển đến Blynk App và ngược lại. Điều này mang lại lợi ích là tất cả những thứ kết nối đến vi điều khiển có thể làm việc với Blynk. Bạn có thể gửi dữ liệu từ Blynk App đến Virtual Pin, ở vi điều khiển nhận dữ liệu Virtual Pin xử lý dữ liệu và phản hồi lại Smarthome.

Cần phân biệt giữa Virtual Pin và GPIO của vi điều khiển. Vi điều khiển có thể gửi dữ liệu đến App bằng cách sử dụng Blynk.virtualWrite(pin,value) và nhận dữ liệu từ App bằng cách sử dụng BLYNK\_WRITE(vPIN).

- Gửi dữ liệu từ App đến hardware

Bạn có thể gửi dữ liệu từ Widget trên App đến hardware qua Virtual Pin. Ví dụ, Button Widget trên App gửi tín hiệu 0 hoặc 1 đến Hardware bằng cách gửi qua Virtual V1.

Trên hardware để nhận dữ liệu từ Widget có thể thực hiện:

BLYNK-WRITE(V1)

{

Int pinData=param.asInt();

}

- Gửi mảng dữ liệu

Một số Widget nhiều hơn một output. Các output từ Widget có thể được gửi đến Hardware ở dạng mảng. Hardware có thể lấy dữ liệu như sau:

BLYNK\_WRITE(V1)

{

Int x=param[0].asInt();

Int y=param[1].asInt();

Int z=param[N].asInt();

}

- Lấy dữ liệu từ Hardware

Có hai cách App lấy dữ liệu từ Hardware qua virtual pin

Widget yêu cầu hardware:

Trên Hardware sử dụng BLYNK\_READ() để gửi dữ liệu đến Widget khi widget yêu cầu:

BLYNK\_READ(V0)

{

Blynk.virtualWrite(5,millis()/1000);

}

Hardware gửi dữ liệu đến Widget:

Hardware có thể gửi dữ liệu đến Widget. Dữ liệu từ hardware gửi lên Blynk Server sẽ được lưu trữ trên server. Lưu ý, nên gửi dữ liệu theo thời gian định trước và tránh hiện tượng Flood Error (hardware gửi quá nhiều lần, làm cho hardware bị mất kết nối), nên sử dụng BlynkTimer.

- Đồng bộ đối với Hardware

Nếu hardware mất kết nối đến internet hoặc reset, hardware có thể lấy tất cả các giá trị Widget trên App:

BLYNK\_CONNECTED() {

Blynk.syncAll();

}

BLYNK\_WRITE(V0){

….

}

Hardware cũng có thể cập nhật các giá trị Virtual Pin bằng cách sử dụng Blynk.syncVirtual(V0) hoặc nhiều Virtual Pin bằngBlynk.syncVirtual(V0, V1, V2…)

Hardware có thể lưu dữ liệu trên server mà không cần có Widget bằng cách gọi Blynk.virtualWrite(V0, value)

- Đối với App

Nếu muốn Widget đồng bộ với Hardware ngay cả khi App offline sử dụng Blynk.virtualWrite. Ví dụ như bạn có một Led Widget kết nối đến Virtual Pin V1, và có 1 nút vật lý kết nối đến Hardware. Khi nhấn nút trên hardware, bạn muốn cập nhật trạng thái đến Led Widget. Để làm được điều đó có thể gửi Blynk.virtualWrite(V1, 255) khi nút vật lý trên hardware được nhấn.

- Điều khiển nhiều thiết bị

Blynk App hỗ trợ nhiều thiết bị. Điều đó có nghĩa rằng có thể gán bất kỳ Widget đến một thiết bị với auth tocken riêng. Ví dụ button V1 có thể điều khiển thiết bị A, và một button V1 khác có thể điều khiển thiết bị B. Để sử dụng nhiều hơn một thiết bị trên project, trong phần Project Setting, kích vào Devices để chọn thêm thiết bị.

- Trạng thái online của thiết bị

Blynk App hỗ trợ trạng thái online của nhiều thiết bị.

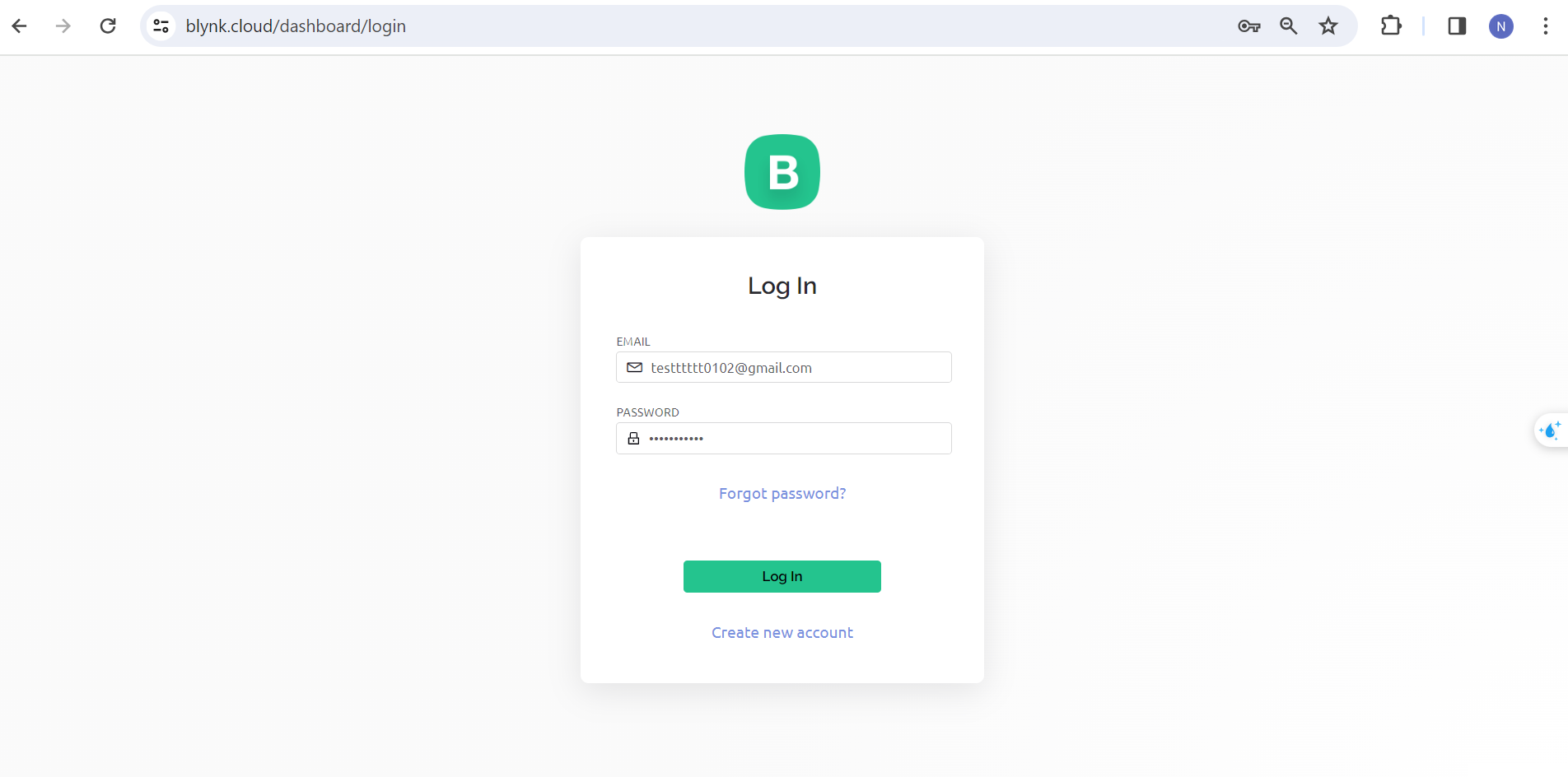
Blynk sử dụng HEARTBEAT. Cách tiếp cận này được thực hiện bằng cách hardware định kỳ gửi lệnh ping (mặc định là 10s). Trong trường hợp, Server không nhận được lệnh ping nào từ hardware trong 10s và thêm 5s, server sẽ cho rằng hardware mất kết nối đến Server.

- Hạn chế và khuyến nghị:

Không thực hiện Blynk.virtualWrite và Blynk.\* trong void loop()- điều này có thể gây ra hardware mất kết nối. Sử dụng các hàm có Timer như BlynkTimer. Tránh sử dụng delay() – nó có thể gây ra mất kết nối. Không gửi quá 100 giá trị/s – dẫn đến Flood Error. Khi sử dụng ESP32 lưu ý không gửi quá nhiều lệnh Blynk.virtualWrite do sự hạn chế của thiết bị này là số lượng request có thể xử lý.

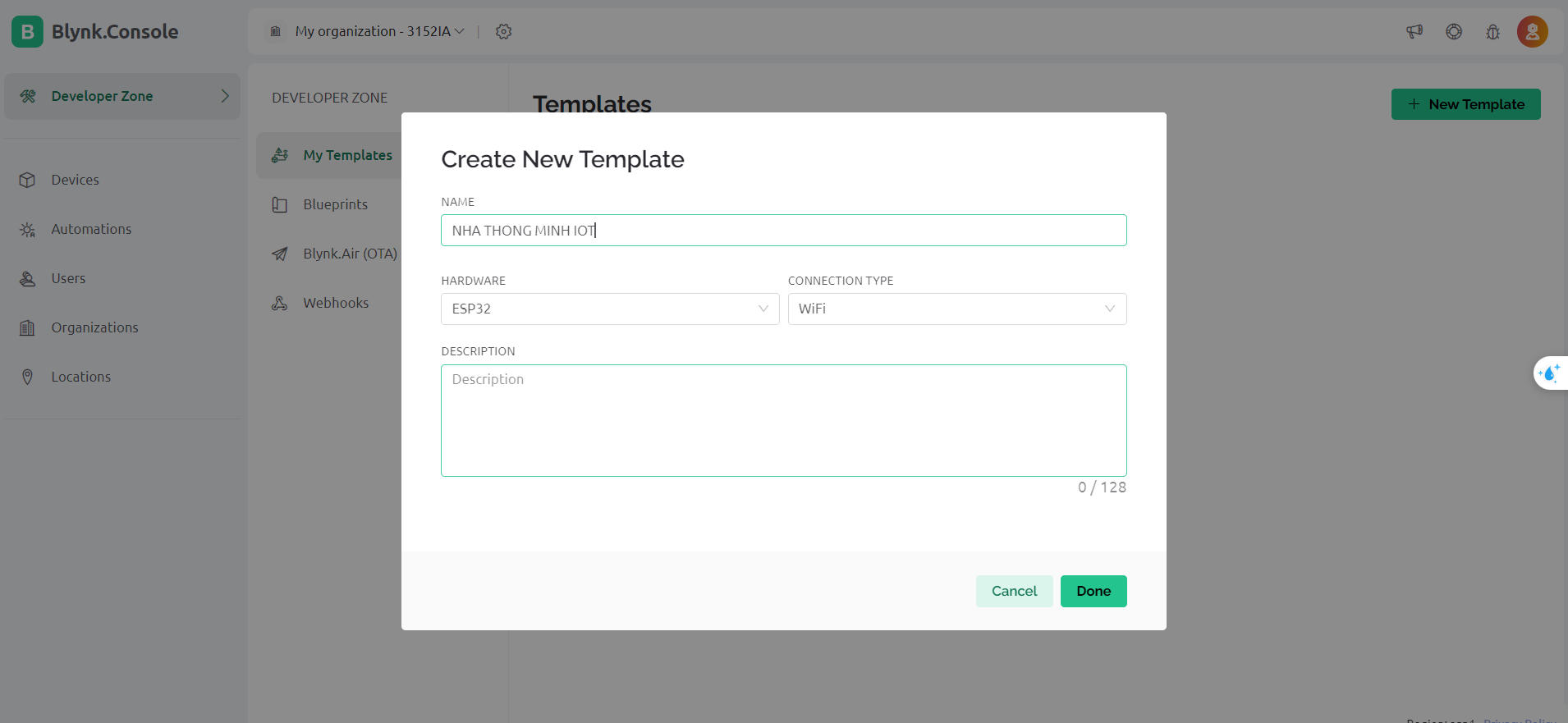
### Thiết kế giao diện Blynk

Đầu tiên đăng nhập vào: <https://blynk.cloud/dashboard/login>



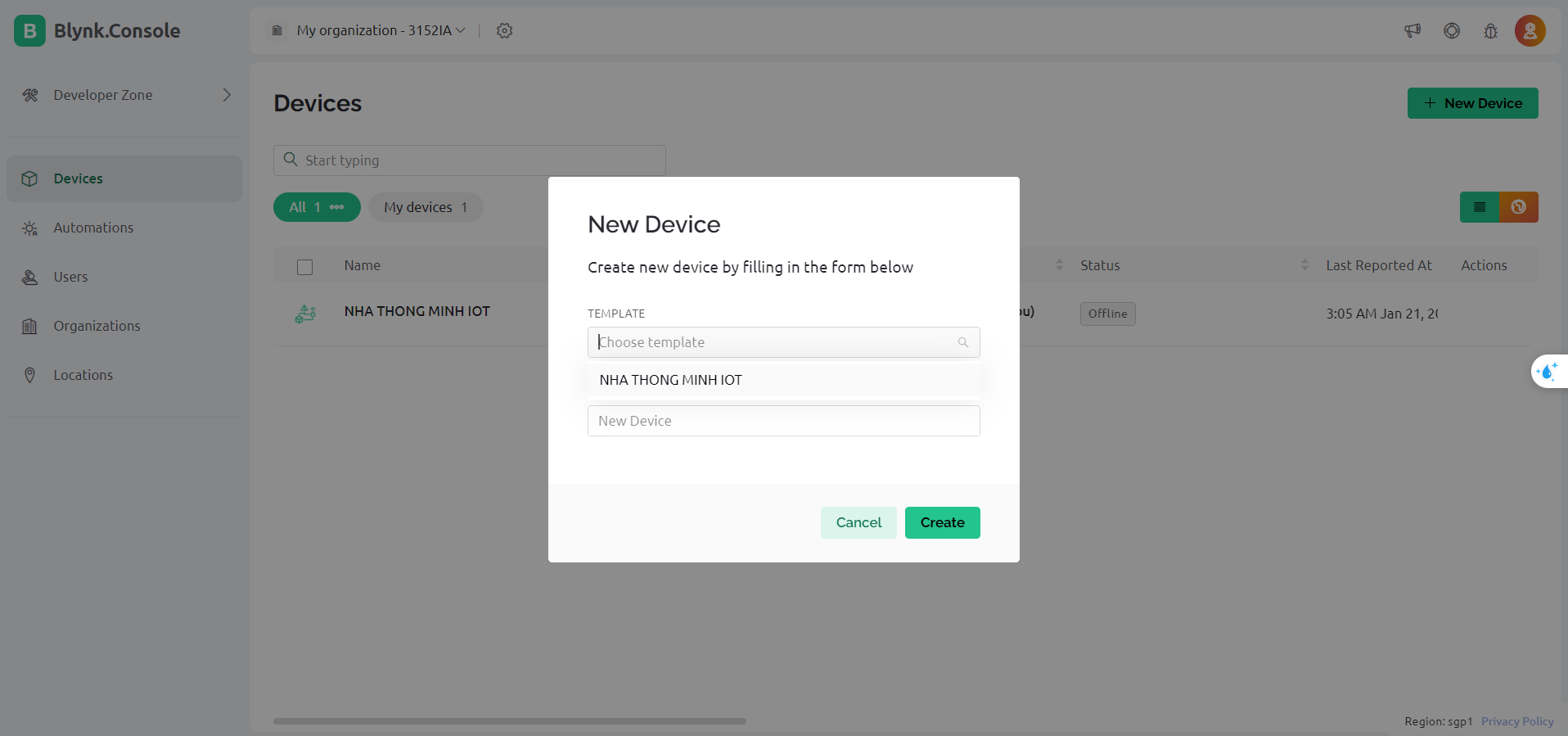
Hình 3.6 Đăng nhập Blynk IoT

Tiếp theo tạo một Template mới



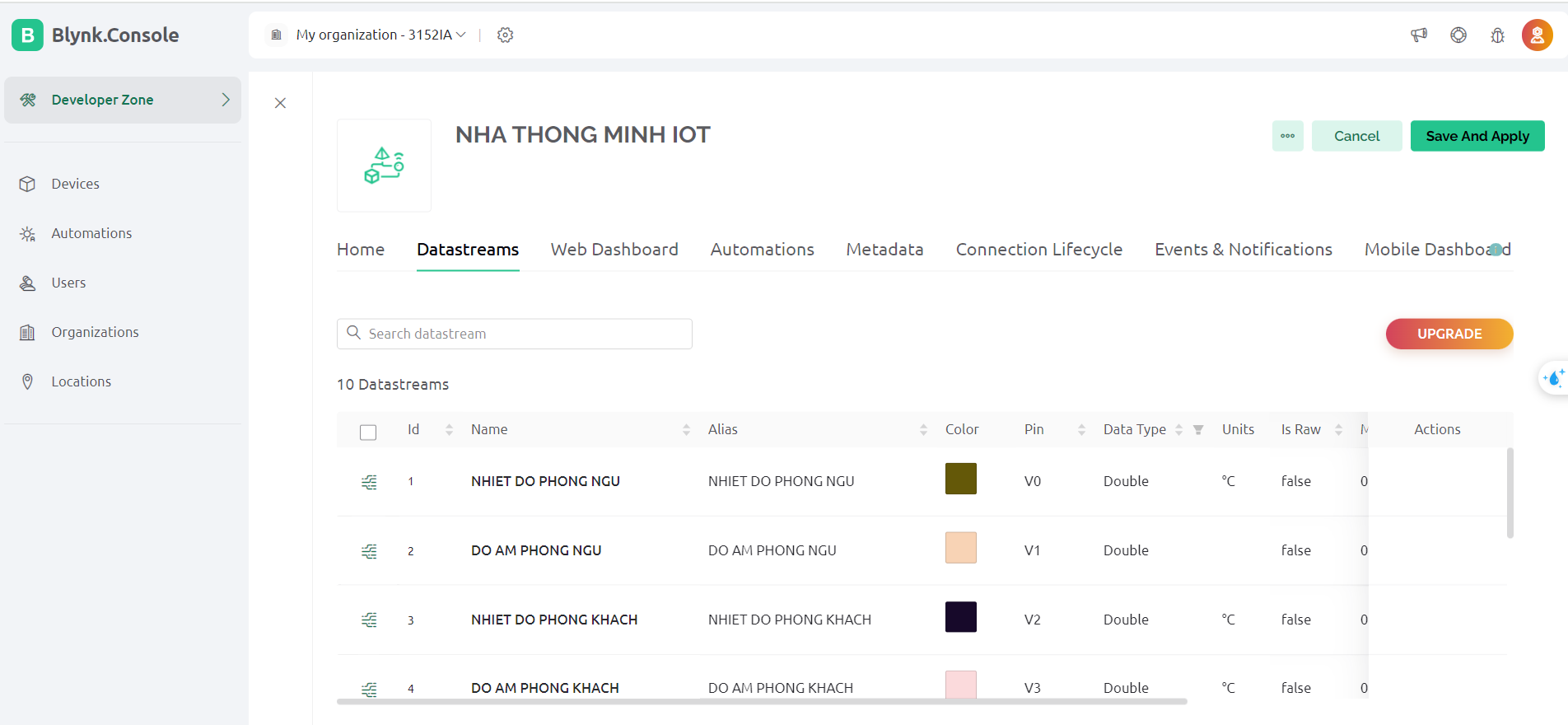
Hình 3.7 Tạo Template

Tạo Device mới



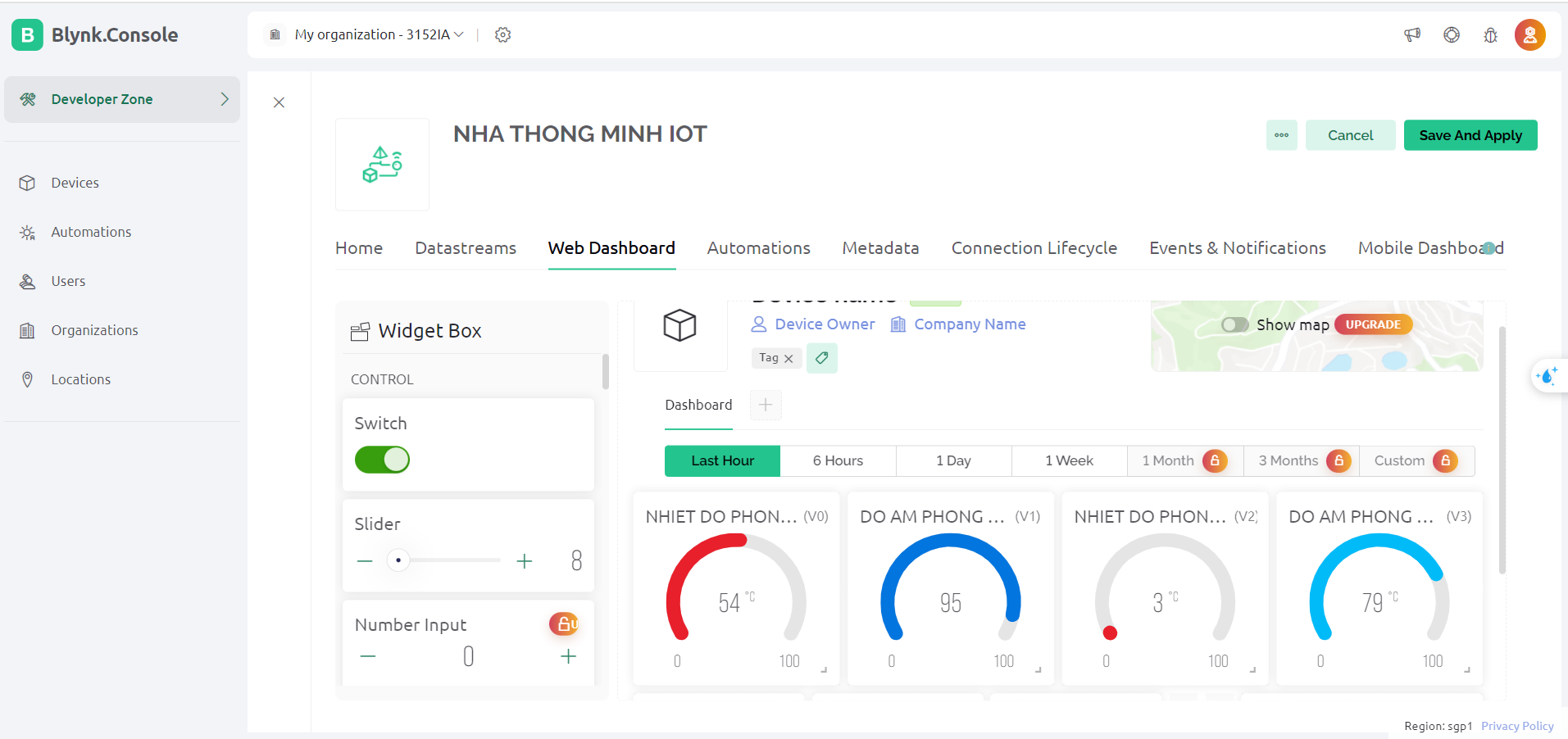
Hình 3.8 Tạo Device

Tạo các Datastreams

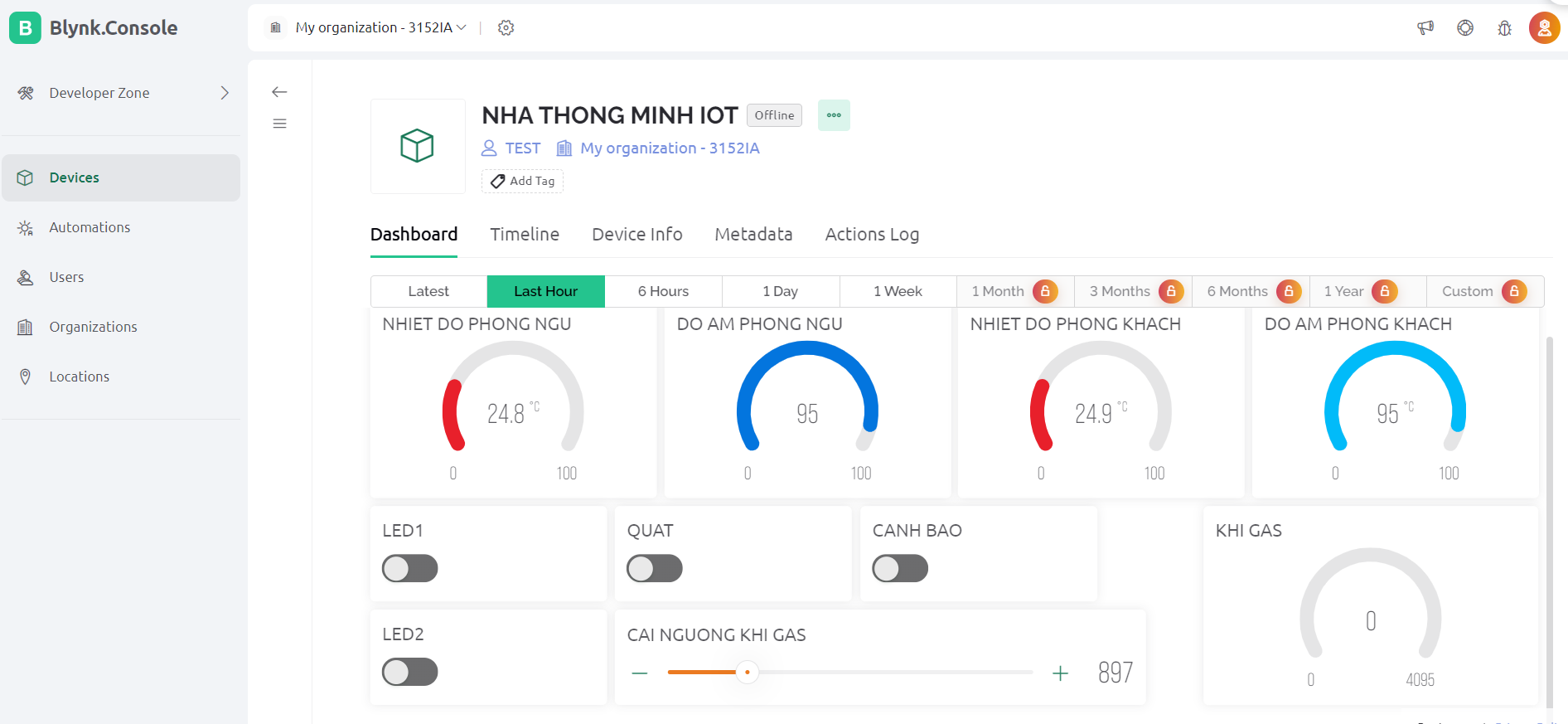


Hình 3.9 Tạo Datastreams

Truy cập vào Web Dashboard để lấy các Widget cho giao diện

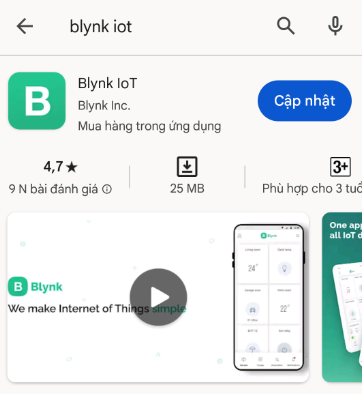


Hình 3.10 Tạo giao diện Blynk IoT



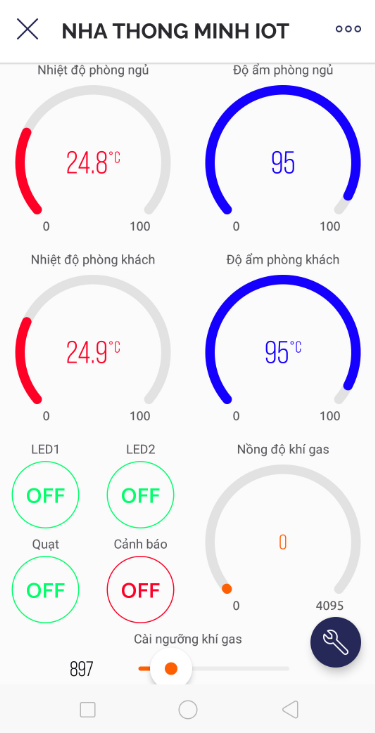
Hình 3.11 Giao diện Blynk được thiết kế trên Web

Nếu sử dụng SmartPhone hệ đều hành Android, ta truy cập CH Play, vào mục tìm kiếm gõ từ khoá “Blynk Iot” và dowload.



Hình 3.12 Dowload Blynk Iot cho Android

Tương tự ta xây dựng được giao diện Blynk IoT trên điện thoại như sau:



Hình 3.13 Giao diện Blynk được thiết kế trên App

# KẾT QUẢ ĐẠT ĐƯỢC VÀ ĐÁNH GIÁ

## Kết quả đạt được

Qua thời gian nghiên cứu, tìm tòi, học hỏi em đã thiết kế một mô hình ngôi nhà thông minh trực quan nhất để có thể ứng dụng vào thực tế với những tính năng như:

* Mở cửa tự động thông qua cảm biến vân tay hoặc pass số.
* Giám sát nhiệt độ, độ ẩm phòng ngủ phòng khách từ xa.
* Hẹn giờ bật tắt quạt.
* Giám sát nồng độ khí gas và đưa ra cảnh báo khi vượt ngưỡng.

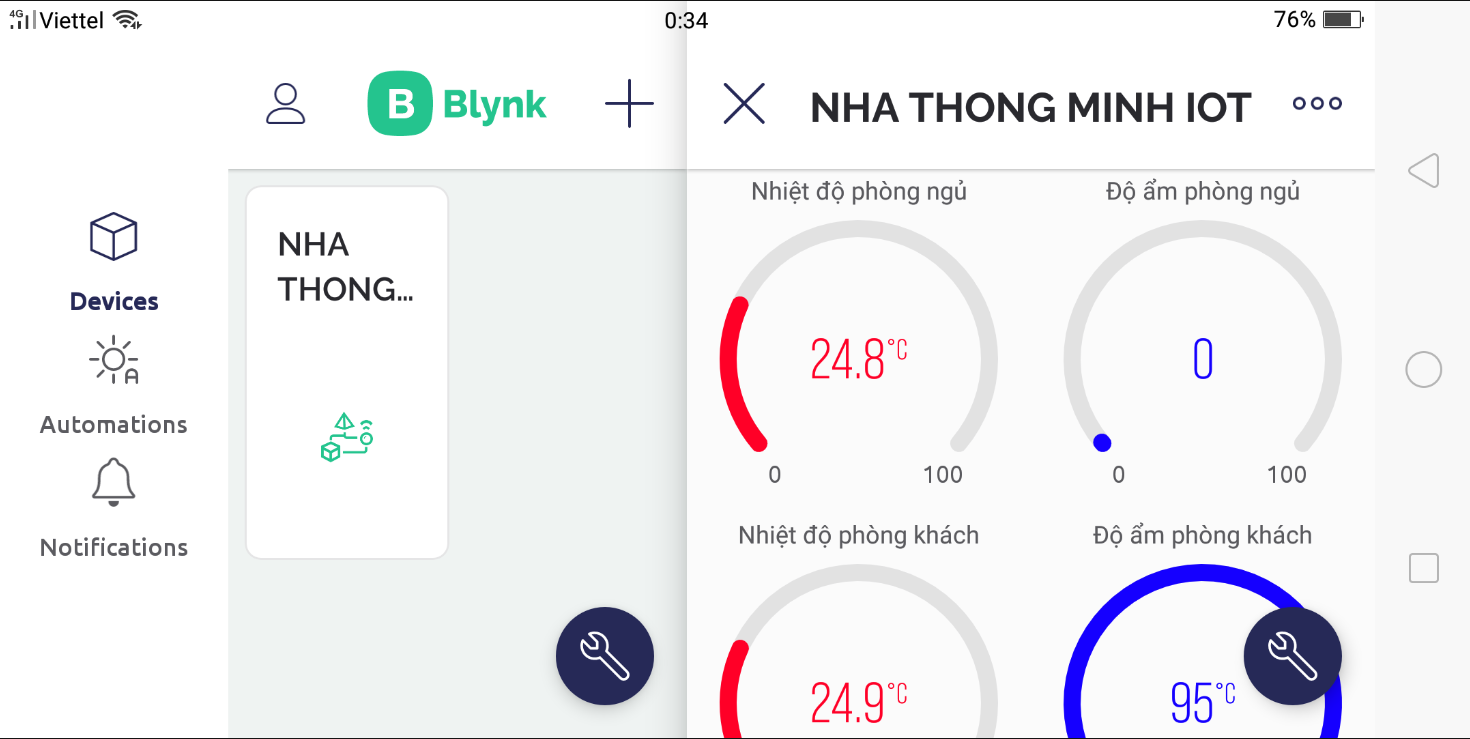
### Phần cơ khí

Hình 4.1 Hình ảnh thực tế của mạch bên trong

Hình 4.2 Hình ảnh thực tế của phần mô hình

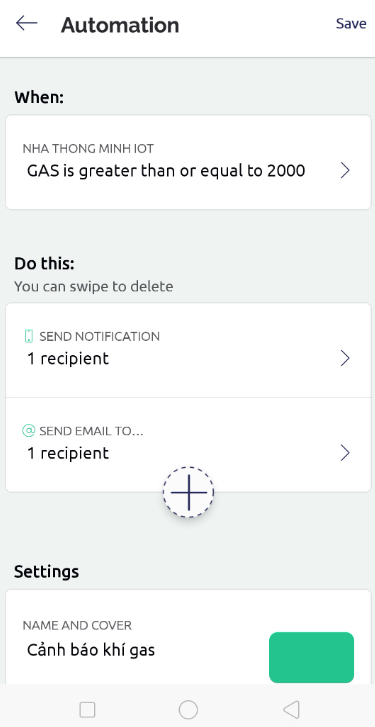
### Phần mềm giám sát điều khiển Blynk IoT

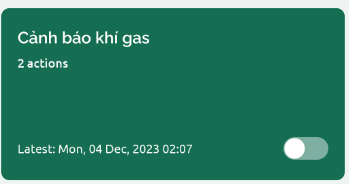
Các dữ liệu về các cảm biến nhiệt độ độ ẩm, cảm biến gas, … đều được hiện thị trên app Blynk IOT, nó giúp chúng ta theo dõi dễ dàng và cũng là điểm bắt đầu của một mô hình nhà thông minh lớn hơn.



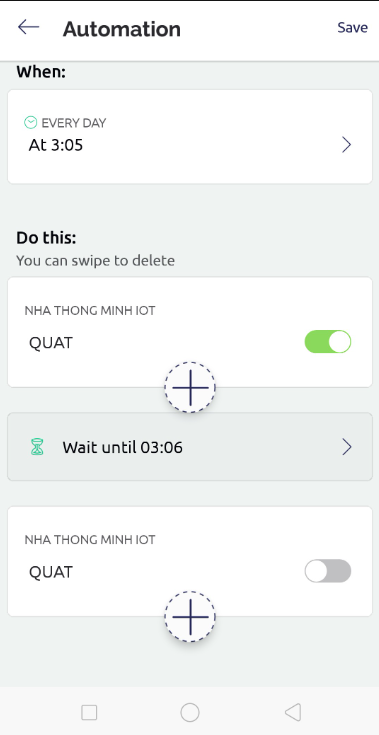
Hình 4.3 Giám sát ngôi nhà thông minh qua Blynk trên SmartPhone

Ngoài ra chúng ta thiết lập thêm một số chức năng tự động như: tự động cảnh báo đến người dùng ngay trên điện thoại khi khí gas vượt ngưỡng, tự động bật tắt quạt theo thời gian cài đặt…



**

Hình 4.4 Thiết lập chức năng cảnh báo khí gas



**

Hình 4.5 Thiết lập chức năng tự động bật tắt quạt

## Thử nghiệm và đánh giá

### Thử nghiệm

//chèn một số hình ảnh ví dụ test xử lý ảnh nhận diện người, điều khiển bật tắt qua app

### Đánh giá

* Mạch hoạt động ổn định và chính xác
* Xử lý ảnh và nhận diện đúng người để mở cửa
* Nhận tín hiệu từ điện thoại còn phụ thuộc vào tốc độ mạng

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Với đề tài “Nghiên cứu, thiết kế, chế tạo mô hình ngôi nhà thông minh”, em đã thực hiện được những công việc sau:

- Đã tìm hiểu được nguyên lý, cơ chế hoạt động của một mô hình nhà thông minh.

- Tìm hiểu các phương pháp điều khiển tốc độ và đọc dữ liệu cảm biến.

- Tìm hiểu về xử lý ảnh

- Thiết kế và thực hiện thành công phần cứng của nhà thông minh sử dụng truyền thông không dây. Hệ thống hoạt động dựa trên vi điều khiển ESP32 xử lý chương trình.

- Lập trình được phần mềm cho vi điều khiển với các chức năng đọc số liệu từ cảm biến, điều khiển các động cơ, đồng thời gửi dữ liệu được lên mạng..

- Sử dụng Blynk IoT để giám sát, và điều khiển các thiết bị trong ngôi nhà thông minh

Bên cạnh những công việc đã đạt được, khi thực hiện đồ án này, em cũng gặp phải một số khó khăn nhất định:

- Thiết kế phần mô hình: do không được học chuyên sâu về phần này nên khi thiết kế gặp rất nhiều trở ngại.

- Lập trình vi điều khiển: Chưa tối ưu được chương trình

Bước đầu cho thấy mô hình hoạt động tương đối ổn định, thực hiện đúng các chức năng theo yêu cầu đề ra. Nếu điều kiện cho phép, trong thời gian tới, em mong muốn được phát triển hệ thống thêm các phần sau:

- Tự xây dựng một ứng dụng riêng để thêm được nhiều chức năng hơn.

- Thiết kế thêm nhiều cảm biến và thiết bị để có nhiều tình huống xảy ra hơn.

# TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | N. V. L. N. T. H. Bùi Quốc Khánh, Truyền Động Điện, Hà Nội: Nhà xuất bản khoa học và Kỹ thuật, 2005. |
| [2] | HHChanh, "Hội quán điện tử," 05 04 2017. [Online]. Available: https://hqdt.vn/baiviet/nguyen-tac-dieu-che-pwm-pulse-width-modulation.html. [Accessed 20 01 2024]. |
| [3] | "slideshare," [Online]. Available: https://fr.slideshare.net/trongthuy3/luan-van-che-tao-mo-hinh-nha-thong-minh-su-dung-arduino-hay. [Accessed 20 01 2024]. |