

TRƯỜNG ĐẠI HỌC PHENIKAA
KHOA ĐIỆN – ĐIỆN TỬ



ĐỒ ÁN TÍCH HỢP LIÊN NGÀNH

Tên đề tài: *THIẾT KẾ VÀ XÂY DỰNG MÔ HÌNH NHÀ THÔNG MINH*

Sinh viên thực hiện:

1. Vũ Thị Linh	MSSV: 21010474	Lớp: K15 -DTVT1
2. Khổng Thị Dung	MSSV: 21012374	Lớp: K15 -DTVT1
3. Lê Thị Hoàng Bắc	MSSV: 21010469	Lớp: K15 -DTVT1

Cán bộ hướng dẫn: TS. Nguyễn Thị Thanh Quỳnh

Khoa Điện – Điện tử

Hà Nội, Học kỳ I, năm học 2023-2024

MỤC LỤC

LỜI TỰA.....	8
MỞ ĐẦU	9
Lý do chọn đề tài.....	9
Đối tượng nghiên cứu	10
Mục tiêu	11
Nội dung nghiên cứu.....	12
Phương pháp nghiên cứu	13
CƠ SỞ LÝ THUYẾT	15
1.1. Tổng quan về nhà thông minh.....	15
1.1.1. Nhà thông minh là gì?.....	15
1.1.2. Nhà thông minh có những lợi ích gì?	16
1.1.3. Xu hướng phát triển	17
1.1.4. Tiềm năng	19
1.2. Tổng quan về vi điều khiển.....	19
1.2.1. Vi điều khiển là gì ?	19
1.2.2. Cấu tạo của vi điều khiển.....	21
1.3. Vi điều khiển ESP32	24
1.3.1. ESP32 là gì?.....	24
1.3.2. Wifi API ESP32.....	24
1.3.3. Board NodeMCU ESP32S.....	26
1.4. Các module / cảm biến	28
1.4.1. RFID	28
1.4.2. Cảm biến khí.....	30
1.4.3. DHT11 – Cảm biến nhiệt độ, độ ẩm.....	33

1.4.4. Cảm biến độ ẩm đất	34
1.4.5. Cảm biến hồng ngoại vật cản vật cản	36
1.5. Kết luận chương.....	39
THIẾT KẾ HỆ THỐNG.....	40
Lưu đồ thuật toán.....	40
2.1. Mô tả hệ thống	41
2.1.1. Giới thiệu	41
2.1.2. Tính năng	41
2.2. Sơ đồ hệ thống.....	42
2.2.1. Khối xử lý trung tâm ESP32.....	43
2.2.2. Khối thiết bị	46
2.2.3. Khối cảm biến.....	46
2.3. Lập trình.....	48
2.3.1. Wifi	49
2.3.2. RFID	50
2.3.3. Code main (được nạp vào ESP32).....	52
2.3.4. Giao diện webserver	58
2.3.5. PlatformIO IDE.....	59
2.4. Kết quả:	60
2.5. Tổng kết chương	66
TỔNG KẾT.....	67
3.1. Nhận xét.....	67
3.1.1. Ưu điểm	67
3.1.2. Giới hạn	67
3.2. Kết luận	67

3.3. Phương hướng phát triển.....	68
TÀI LIỆU THAM KHẢO	69

MỤC LỤC HÌNH ẢNH

Hình 0. 1: Nhà thông minh.....	9
Hình 1. 1: Giám sát an ninh.....	16
Hình 1. 2: Công nghệ 4.0 cho ngôi nhà.....	17
Hình 1. 3: Vi điều khiển LPC214X.....	20
Hình 1. 4: Cấu tạo của vi điều khiển.....	21
Hình 1. 5: GPU.....	21
Hình 1. 6: Oscillator Circuit.....	22
Hình 1. 7: Timer/Counter.....	23
Hình 1. 8: Chế độ AP.....	25
Hình 1. 9: Chế độ STA.....	25
Hình 1. 10: Board NodeMCU ESP32S.....	26
Hình 1. 11: Các chân của Board NodeMCU ESP32S.....	27
Hình 1. 12: RFID.....	28
Hình 1. 13: hoạt động của RFID.....	29
Hình 1. 14: Cảm biến khí ga MQ2.....	30
Hình 1. 15: Cấu tạo của cảm biến Khí.....	31
Hình 1. 16: Nguyên lý hoạt động của cảm biến Khí.....	32
Hình 1. 17: Cảm biến nhiệt độ - độ ẩm.....	33
Hình 1. 18: Cảm biến độ ẩm đất.....	34
Hình 1. 19: Sơ đồ mạch của cảm biến độ ẩm đất.....	35
Hình 1. 20: Cấu tạo của cảm biến độ ẩm đất.....	35
Hình 1. 21: Cảm biến hồng ngoại vật cản vật cản.....	36
Hình 1. 22: Cấu tạo của Cảm biến hồng ngoại vật cản.....	37
Hình 1. 23: Sơ đồ mạch nguyên lý của cảm biến hồng ngoại vật cản.....	38
Hình 2. 1: Lưu đồ thuật toán.....	40
Hình 2. 2: Sơ đồ hệ thống.....	42
Hình 2. 3: Mạch nguyên lý.....	43

Hình 2. 4: Sơ đồ khối nguồn	44
Hình 2. 5: Sơ đồ mạch 3D	45
Hình 2. 6: Sơ đồ mạch in.....	45
Hình 2. 7: ID và Pass được setup trên code	49
Hình 2. 8: Mobile Hotspot.....	49
Hình 2. 9: network.h.....	49
Hình 2. 10: Thiết lập mã cửa.....	50
Hình 2. 11: RFID_Check	51
Hình 2. 12: user_rfid.h	51
Hình 2. 13: Gửi dữ liệu	55
Hình 2. 14: Chế độ Auto cho cửa.....	56
Hình 2. 15: Webserver (các khu vực khác).....	57
Hình 2. 16: Giao diện webserver.....	58
Hình 2. 17: PlatformIO IDE	59
Hình 2. 18: Mô hình nhà thông minh	60
Hình 2. 19: WIFI	61
Hình 2. 20: Webserver	61
Hình 2. 21: Nhà hoạt động	62
Hình 2. 22: RFID.....	62
Hình 2. 23: Đèn (khách và ngủ)	63
Hình 2. 24: Temp – Gas	63
Hình 2. 25: Quạt phòng bếp	63
Hình 2. 26: Quạt phòng khách	64
Hình 2. 27: Soil (93).....	64
Hình 2. 28: Bơm đã ngắt	64
Hình 2. 29: Soil (33).....	65
Hình 2. 30: Bơm hoạt động.....	65
Hình 2. 31: Munaul	66

LỜI TỰA

Trước hết, chúng em xin gửi lời cảm ơn tới tất cả các thầy cô đã hỗ trợ và hướng dẫn chúng em rất nhiều trong khoảng thời gian thực hiện đồ án này, đặc biệt là TS. Nguyễn Thị Thanh Quỳnh và Khoa Điện – Điện tử đã tạo điều kiện thuận lợi nhất để chúng em có thể hoàn thành được đồ án này.

Do giới hạn thời gian thực hiện đồ án và việc thiết kế, xây dựng mô hình cần nhiều kỹ năng và kiến thức tổng hợp, đồ án của chúng em vẫn còn mắc nhiều sai sót. Với lòng ham học hỏi, chúng em rất mong nhận được sự góp ý của các thầy cô và các bạn sinh viên để chúng em có thể rút kinh nghiệm và hoàn thiện đồ án tốt hơn.

Hà Nội, ngày 05/11/2023

Sinh viên thực hiện

Vũ Thị Linh

Không Thị Dung

Lê Thị Hoàng Bắc

MỞ ĐẦU

Trong đồ án “Xây dựng và thiết kế mô hình nhà thông minh”, chúng em sẽ tập trung vào việc thiết kế và xây dựng một hệ thống nhà thông minh dựa trên các công nghệ IoT. Chúng em sử dụng PlatformIO và webserver để phát triển ứng dụng điều khiển các thiết bị trong nhà thông minh, và sử dụng ESP32 để tạo ra một hệ thống mạng không dây cho phép các thiết bị giao tiếp với nhau cũng như với người dùng qua webserver. Chúng em đã thiết kế và triển khai nhiều tính năng trong nhà thông minh, bao gồm việc bật/tắt đèn bằng cảm biến hồng ngoại vật cản, hệ thống cảnh báo rò rỉ khí ga, hệ thống tưới nước tự động, và sử dụng RFID và cảm biến nhiệt độ/độ ẩm. Tất cả các tính năng này có thể hoạt động một cách tự động hoặc được điều khiển từ xa thông qua giao diện webserver. Đồ án này không chỉ thể hiện sự kết hợp của ba ngành học là ngành Điện tử viễn thông, công nghệ thông tin và tự động hoá, mà còn đặt nền móng cho việc phát triển thêm về nhà thông minh trong tương lai. Chúng em tin rằng việc tối ưu hóa sản phẩm và áp dụng các thuật toán mới sẽ giúp chúng em cải thiện hiệu suất và tính năng của hệ thống trong thời gian tới.

Lý do chọn đề tài



Hình 0. 1: Nhà thông minh

Lý do chọn đề tài cho đồ án “Xây dựng và thiết kế mô hình nhà thông minh” là do sự phát triển nhanh chóng của công nghệ Internet of Things (IoT). IoT đã trở thành một phần không thể thiếu của cuộc sống hiện đại, giúp kết nối các thiết

bị và hệ thống với nhau thông qua internet, tạo ra những giải pháp mới cho cuộc sống hàng ngày. Với IoT, chúng ta có thể điều khiển các thiết bị trong nhà từ xa, theo dõi tình hình sức khỏe cá nhân, và thậm chí quản lý các hệ thống lớn như giao thông và năng lượng. Chúng em lựa chọn đề tài này vì chúng em nhận thấy rằng IoT có tiềm năng lớn trong việc cải thiện chất lượng cuộc sống và tạo ra một môi trường sống thông minh. Chúng em muốn khám phá và tận dụng những tiềm năng này để xây dựng một mô hình nhà thông minh, nơi mà các thiết bị và hệ thống có thể hoạt động một cách tự động và linh hoạt, đáp ứng nhu cầu của người dùng một cách nhanh chóng và hiệu quả. Ngoài ra, chúng em cũng nhận thấy rằng việc kết hợp IoT với các ngành học khác như Công nghệ thông tin, Điện tử viễn thông và Tự động hóa có thể tạo ra những giải pháp mới và sáng tạo. Chúng em tin rằng việc kết hợp kiến thức từ ba ngành học này sẽ giúp chúng em tạo ra một sản phẩm độc đáo và hiệu quả, đáp ứng nhu cầu của xã hội trong thời đại số hóa. Cuối cùng, chúng em chọn đề tài này vì chúng em muốn đóng góp vào sự phát triển của công nghệ IoT tại Việt Nam và trên thế giới. Chúng em hy vọng rằng đồ án này sẽ mở ra cơ hội cho những nghiên cứu và ứng dụng mới trong tương lai, góp phần vào sự phát triển của công nghệ và xã hội.

Đối tượng nghiên cứu

Đối tượng nghiên cứu của đề tài này là hệ thống điều khiển và giám sát nhà thông minh. Hệ thống này bao gồm các thiết bị và cảm biến sau:

- Vi điều khiển: là bộ não của hệ thống nhà thông minh. Vi điều khiển sẽ xử lý các tín hiệu từ các cảm biến và điều khiển các thiết bị trong nhà.
- Cảm biến: là thiết bị thu thập thông tin từ môi trường xung quanh. Các cảm biến thường dùng trong nhà thông minh bao gồm cảm biến nhiệt độ, cảm biến chuyển động, cảm biến khói,...
- Thiết bị: là các thiết bị điện tử trong nhà có thể được điều khiển từ xa. Các thiết bị thường dùng trong nhà thông minh bao gồm đèn, rèm cửa, điều hòa, máy bơm nước,...

Hệ thống điều khiển

- Hệ thống điều khiển của nhà thông minh có nhiệm vụ nhận tín hiệu từ các

cảm biến và điều khiển các thiết bị trong nhà. Hệ thống điều khiển này có thể hoạt động theo hai chế độ:

- Chế độ tự động: hệ thống sẽ tự động điều khiển các thiết bị trong nhà dựa trên các chương trình đã được lập trình sẵn.
- Chế độ thủ công: người dùng có thể điều khiển các thiết bị trong nhà theo ý muốn.

Hệ thống giám sát

- Hệ thống giám sát của nhà thông minh có nhiệm vụ giám sát hoạt động của hệ thống và cảnh báo cho người dùng khi có sự cố xảy ra. Hệ thống giám sát này có thể dựa trên các thông tin sau:
- Thông tin từ các cảm biến: hệ thống giám sát sẽ theo dõi các thông tin từ các cảm biến, chẳng hạn như nhiệt độ, độ ẩm, chuyển động,...
- Trạng thái của các thiết bị: hệ thống giám sát sẽ theo dõi trạng thái của các thiết bị, chẳng hạn như đèn đang bật hay tắt, cửa đang mở hay đóng,...

Mục tiêu

Mục tiêu của dự án này là thiết kế và xây dựng mô hình nhà thông minh có khả năng hoạt động tự động và thủ công, đồng thời có thể điều khiển qua webserver.

Chế độ hoạt động: mô hình nhà thông minh được thiết kế với hai chế độ hoạt động:

- Tự động: hệ thống sẽ tự động điều khiển các thiết bị trong nhà dựa trên các kịch bản được lập trình sẵn.
- Thủ công: người dùng có thể điều khiển các thiết bị trong nhà theo ý muốn.
- Điều khiển

Mô hình nhà thông minh được thiết kế để có thể điều khiển qua webserver. Người dùng có thể sử dụng trình duyệt webserver trên máy tính hoặc thiết bị di động để điều khiển các thiết bị trong nhà. Việc thiết kế mô hình nhà thông minh với hai chế độ hoạt động và khả năng điều khiển qua webserver sẽ mang lại nhiều tiện ích cho người sử dụng, giúp tiết kiệm thời gian và công sức, đồng thời tăng cường sự an toàn và tiện nghi cho cuộc sống.

Nội dung nghiên cứu

Nội dung nghiên cứu của đề tài này bao gồm việc thiết kế và xây dựng mô hình nhà thông minh có các tính năng sau:

- Hệ thống báo cháy.
- Bật tắt đèn khi có vật di chuyển.
- Cảm biến nhiệt độ.
- Tưới nước tự động.
- Cửa thông minh.

Các tính năng này sẽ được thực hiện thông qua các thiết bị và cảm biến sau:

- Vi điều khiển ESP32.
- Cảm biến khói.
- Cảm biến chuyển động hồng ngoại.
- Cảm biến nhiệt độ - độ ẩm.
- Cảm biến độ ẩm đất.
- RFID.

Mô hình nhà thông minh sẽ được thiết kế theo mô hình một căn hộ nhỏ. Các thiết bị và cảm biến sẽ được bố trí trong căn hộ theo cách hợp lý để đảm bảo tính thẩm mỹ và khả năng hoạt động của hệ thống.

Vi điều khiển ESP32 sẽ được lập trình để thực hiện các nhiệm vụ sau:

- Nhận tín hiệu từ các cảm biến.
- Điều khiển các thiết bị trong nhà.
- Giám sát hoạt động của hệ thống.

Webserver sẽ được thiết kế với giao diện thân thiện với người dùng, dễ sử dụng. Người dùng có thể sử dụng webserver để điều khiển các thiết bị trong nhà, xem trạng thái của hệ thống, cũng như thiết lập các kịch bản tự động.

Mạch cứng sẽ được thiết kế để kết nối các thiết bị và cảm biến với nhau một cách chắc chắn và an toàn.

Đề tài này chúng em sẽ xử lý các bài toán: bơm nước tự động, cảnh báo khí ga, cửa thông minh, đèn tự động, điều khiển trên webserver thông qua wifi.

Phương pháp nghiên cứu

Để nghiên cứu về nhà thông minh, cần có kiến thức về các lĩnh vực sau:

- Hệ thống điều khiển: Hệ thống điều khiển là bộ não của hệ thống nhà thông minh, do đó cần có kiến thức về hệ thống điều khiển để thiết kế và xây dựng hệ thống nhà thông minh.
- Cảm biến: Cảm biến là thiết bị thu thập thông tin từ môi trường xung quanh, do đó cần có kiến thức về cảm biến để lựa chọn và sử dụng cảm biến phù hợp.
- Thiết bị điện tử: Thiết bị điện tử là các thiết bị trong nhà có thể được điều khiển từ xa, do đó cần có kiến thức về thiết bị điện tử để lựa chọn và sử dụng thiết bị phù hợp.
- Mạng máy tính: Mạng máy tính là nền tảng để kết nối các thiết bị và hệ thống trong nhà thông minh, do đó cần có kiến thức về mạng máy tính để thiết kế và xây dựng mạng truyền thông cho hệ thống nhà thông minh.

Các công nghệ mới trong nhà thông minh: các công nghệ mới trong nhà thông minh đang phát triển nhanh chóng, bao gồm

- Internet vạn vật (IoT): IoT cho phép các thiết bị trong nhà thông minh kết nối với nhau và với Internet. Điều này tạo ra nhiều khả năng mới cho nhà thông minh, chẳng hạn như giám sát và điều khiển nhà thông minh từ xa thông qua Internet.
- Nhận dạng giọng nói: Nhận dạng giọng nói cho phép người dùng điều khiển các thiết bị trong nhà bằng giọng nói. Điều này giúp người dùng dễ dàng điều khiển nhà thông minh hơn.
- Học máy: Học máy cho phép hệ thống nhà thông minh tự động học hỏi và thích ứng với thói quen của người dùng. Điều này giúp hệ thống nhà thông minh trở nên thông minh và tiện lợi hơn.

Tổng kết chương

Phần này của đề tài nghiên cứu "Thiết kế và xây dựng mô hình nhà thông minh" đã cung cấp các kiến thức cơ bản về đề tài. Mục tiêu của đề tài là thiết kế và xây dựng mô hình nhà thông minh có khả năng hoạt động tự động và thủ công,

đồng thời có thể điều khiển qua webserver. Nhà thông minh là một lĩnh vực công nghệ đang phát triển nhanh chóng với nhiều tiềm năng ứng dụng. Nghiên cứu về nhà thông minh là một lĩnh vực thú vị và có nhiều triển vọng. Để đạt được mục tiêu của đề tài, chương 1 sẽ tìm hiểu chi tiết các lý thuyết về nhà thông minh, ESP32, các cảm biến sử dụng trong mô hình nhà thông minh. Các kiến thức này là nền tảng quan trọng để thực hiện các nghiên cứu tiếp theo trong đề tài. Chương 1 sẽ bao gồm các nội dung sau: tổng quan về nhà thông minh, ESP32 - vi điều khiển cho nhà thông minh, các cảm biến sử dụng trong nhà thông minh. Kết thúc chương 1, nhóm nghiên cứu sẽ có đầy đủ kiến thức và nền tảng cần thiết để bắt đầu thiết kế và xây dựng mô hình nhà thông minh.

CHƯƠNG I

CƠ SỞ LÝ THUYẾT

Phần đầu tiên của đề tài nghiên cứu "Thiết kế và xây dựng mô hình nhà thông minh" đã cung cấp các kiến thức cơ bản về đề tài. Để đạt được mục tiêu của đề tài, chương này sẽ tìm hiểu chi tiết các lý thuyết về nhà thông minh, ESP32, các cảm biến sử dụng trong mô hình nhà thông minh. Các kiến thức này là nền tảng quan trọng để thực hiện các nghiên cứu tiếp theo trong đề tài.

1.1. Tổng quan về nhà thông minh

Nhà thông minh là một lĩnh vực công nghệ đang phát triển nhanh chóng và có nhiều tiềm năng ứng dụng. Nhà thông minh mang lại nhiều lợi ích cho người dùng và có thể được ứng dụng trong nhiều lĩnh vực khác nhau.

1.1.1. Nhà thông minh là gì?

Nhà thông minh (Smart Home) là căn nhà được tích hợp các thiết bị, điện tử được điều khiển, giám sát, truy cập từ xa. Khác với nhà tự động (Home Automation) đơn thuần chỉ là các thiết bị hoạt động theo lịch trình cài đặt sẵn, Smart Home được xây dựng trên nền tảng khái niệm Internet vạn vật (IoT). Tại đây, các thiết bị trong nhà có thể trao đổi thông tin với nhau, điều chỉnh các chức năng theo thói quen sinh hoạt hàng ngày của người dùng.

Nhà thông minh tích hợp các hệ thống cơ bản như hệ thống điều khiển ánh sáng, hệ thống an ninh, hệ thống điều khiển rèm, hệ thống kiểm soát môi trường, thiết bị cảm ứng... Việc điều khiển các hệ thống này thông qua một phần mềm cài đặt trên điện thoại hoặc máy tính có kết nối internet, bạn có thể điều khiển nó từ bất kỳ đâu hoặc điều khiển bằng giọng nói.ⁱ

Một hệ thống nhà thông minh cơ bản bao gồm các thành phần chính sau:

- **Mạng truyền thông:** Mạng truyền thông là nền tảng để kết nối các thiết bị và hệ thống trong nhà thông minh. Các loại mạng truyền thông thường được sử dụng trong nhà thông minh bao gồm mạng có dây (Ethernet, PLC,...) và mạng không dây (WiFi, Bluetooth, Zigbee,...).

- Cảm biến: Cảm biến là thiết bị thu thập thông tin từ môi trường xung quanh. Các loại cảm biến thường được sử dụng trong nhà thông minh bao gồm cảm biến nhiệt độ, cảm biến độ ẩm, cảm biến chuyển động, cảm biến khói,...
- Thiết bị: Thiết bị là các thiết bị điện tử trong nhà có thể được điều khiển từ xa. Các loại thiết bị thường được sử dụng trong nhà thông minh bao gồm đèn, rèm cửa, điều hòa, máy bơm nước,...
- Hệ thống điều khiển: Hệ thống điều khiển là bộ não của hệ thống nhà thông minh. Hệ thống này sẽ xử lý các tín hiệu từ các cảm biến và điều khiển các thiết bị trong nhà.

1.1.2. Nhà thông minh có những lợi ích gì?

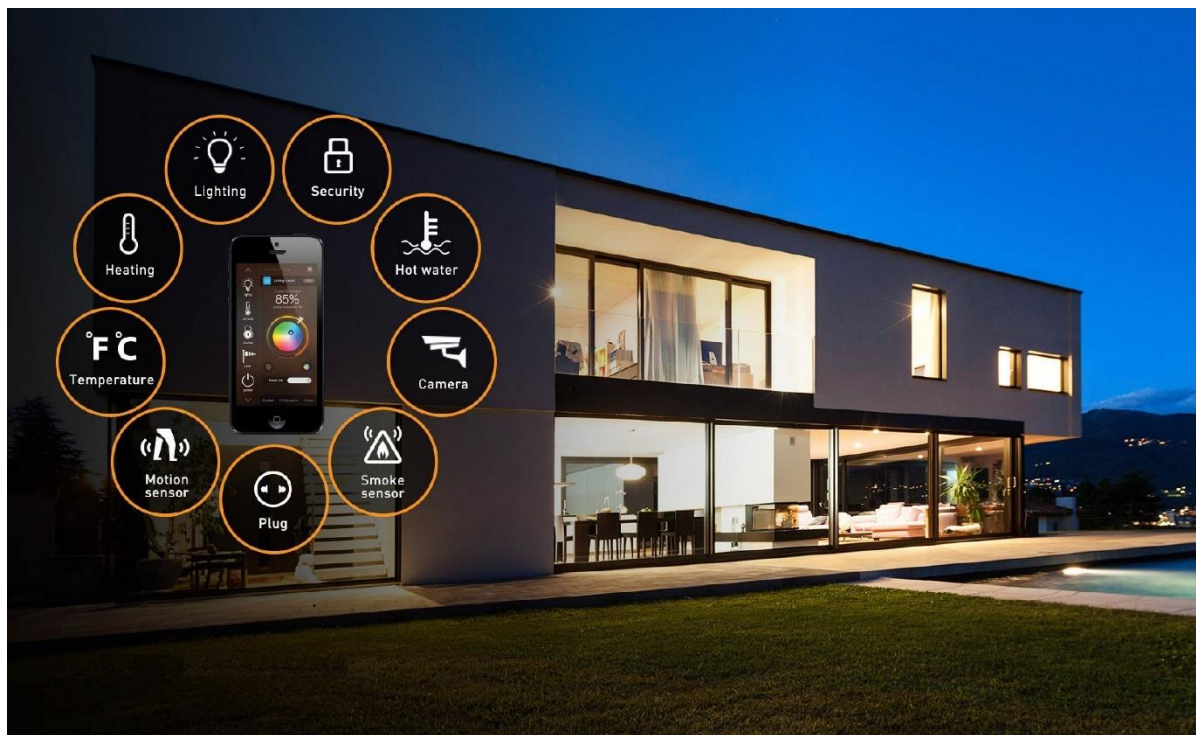


Hình 1. 1: Giám sát an ninh

Nhà thông minh mang lại nhiều lợi ích cho người dùng, bao gồm:

- Tiện nghi: Người dùng có thể điều khiển các thiết bị trong nhà từ xa, tự động hóa các tác vụ trong nhà,...
- Tiết kiệm năng lượng: Nhà thông minh có thể giúp người dùng tiết kiệm năng lượng bằng cách tự động điều chỉnh nhiệt độ, ánh sáng,...
- An toàn: Nhà thông minh có thể giúp người dùng bảo vệ nhà cửa khỏi các nguy cơ như trộm cắp, cháy nổ,...
- Giám sát: Nhà thông minh có thể giúp người dùng giám sát hoạt động của nhà ở, chẳng hạn như nhiệt độ, độ ẩm,...

1.1.3. Xu hướng phát triển



Hình 1. 2: Công nghệ 4.0 cho ngôi nhà

Nhà thông minh (smart home) là một ngôi nhà được trang bị các thiết bị điện tử và hệ thống tự động hóa, cho phép người dùng điều khiển các thiết bị trong nhà từ xa, tự động hóa các tác vụ trong nhà và giám sát hoạt động của nhà ở. Xu hướng phát triển nhà thông minh trên thế giới đang diễn ra rất mạnh mẽ. Theo thống kê của Statista, thị trường nhà thông minh toàn cầu dự kiến sẽ đạt giá trị 393,1 tỷ USD vào năm 2025.

Có nhiều yếu tố thúc đẩy sự phát triển của nhà thông minh, bao gồm:

- Sự phát triển của công nghệ: Công nghệ ngày càng phát triển, đặc biệt là sự phát triển của Internet, trí tuệ nhân tạo (AI),... đã tạo điều kiện thuận lợi cho việc phát triển nhà thông minh.
- Nhu cầu của người dùng: Người dùng ngày càng có nhu cầu cao về tiện nghi, tiết kiệm năng lượng, an toàn và giám sát cho ngôi nhà của mình. Nhà thông minh đáp ứng được những nhu cầu này.
- Chính sách của các quốc gia: Nhiều quốc gia đang có chính sách khuyến khích phát triển nhà thông minh, chẳng hạn như miễn thuế cho các thiết bị nhà thông minh.

Trên thế giới, các quốc gia dẫn đầu về thị trường nhà thông minh bao gồm:

- Trung Quốc: Trung Quốc là quốc gia có thị trường nhà thông minh lớn nhất thế giới. Theo thống kê của Statista, thị trường nhà thông minh Trung Quốc dự kiến sẽ đạt giá trị 142,4 tỷ USD vào năm 2025.
- Hoa Kỳ: Hoa Kỳ là quốc gia có thị trường nhà thông minh lớn thứ hai thế giới. Theo thống kê của Statista, thị trường nhà thông minh Hoa Kỳ dự kiến sẽ đạt giá trị 74,3 tỷ USD vào năm 2025.
- Đức: Đức là quốc gia có thị trường nhà thông minh lớn thứ ba thế giới. Theo thống kê của Statista, thị trường nhà thông minh Đức dự kiến sẽ đạt giá trị 27,3 tỷ USD vào năm 2025.

Xu hướng phát triển nhà thông minh ở Việt Nam đang diễn ra rất mạnh mẽ. Theo thống kê của Hiệp hội Thương mại điện tử Việt Nam, thị trường nhà thông minh Việt Nam dự kiến sẽ đạt giá trị 10 tỷ USD vào năm 2025.

Tại các thành phố lớn của Việt Nam, xu hướng phát triển nhà thông minh đang diễn ra rất mạnh mẽ. Các thiết bị nhà thông minh phổ biến ở các thành phố lớn bao gồm:

- Đèn thông minh: Đèn thông minh có thể được điều khiển từ xa, tự động bật/tắt theo thời gian hoặc theo cảm biến.
- Rèm cửa thông minh: Rèm cửa thông minh có thể được điều khiển từ xa, tự động đóng/mở theo thời gian hoặc theo cảm biến.
- Điều hòa thông minh: Điều hòa thông minh có thể được điều khiển từ xa, tự động điều chỉnh nhiệt độ theo thời tiết hoặc theo thói quen sinh hoạt của người dùng.
- Ổ cắm thông minh: Ổ cắm thông minh có thể được điều khiển từ xa, tự động bật/tắt các thiết bị điện trong nhà.

Tại các khu vực nông thôn của Việt Nam, xu hướng phát triển nhà thông minh đang diễn ra chậm hơn so với các thành phố lớn. Tuy nhiên, xu hướng này đang dần được phổ biến hơn. Các thiết bị nhà thông minh phổ biến ở các khu vực nông thôn bao gồm:

- Đèn thông minh: Đèn thông minh có thể giúp tiết kiệm năng lượng và bảo vệ an ninh cho ngôi nhà.

- Ổ cắm thông minh: Ổ cắm thông minh có thể giúp người dùng điều khiển các thiết bị điện trong nhà từ xa, chẳng hạn như máy bơm nước, máy tưới cây,...ⁱⁱ

1.1.4. Tiềm năng

Nhà thông minh có nhiều tiềm năng phát triển trong tương lai. Với sự phát triển của công nghệ, nhà thông minh sẽ ngày càng trở nên hiện đại và thông minh hơn.

Một số tiềm năng phát triển của nhà thông minh trong tương lai bao gồm:

- Ứng dụng trí tuệ nhân tạo (AI): AI sẽ được ứng dụng ngày càng nhiều trong nhà thông minh, giúp các hệ thống nhà thông minh trở nên thông minh hơn, có thể tự học hỏi và thích ứng với thói quen của người dùng.
- Tăng cường kết nối: Các thiết bị nhà thông minh sẽ được kết nối với nhau và với Internet một cách chặt chẽ hơn, cho phép người dùng có thể điều khiển các thiết bị trong nhà từ xa một cách dễ dàng và thuận tiện hơn.
- Giảm giá thành: Giá thành của các thiết bị nhà thông minh sẽ tiếp tục giảm, giúp các thiết bị nhà thông minh trở nên phổ biến hơn, tiếp cận được với nhiều người dùng hơn.

Trong mô hình nhà thông minh này chúng em thiết kế cho ngôi nhà những tính năng như: hệ thống báo cháy, bật tắt đèn khi có vật di chuyển, cảm biến nhiệt độ, tưới nước tự động, cửa thông minh. Do vậy, các thiết bị được sử dụng trong đồ án này gồm: cảm biến khí ga MQ-2, cảm biến hồng ngoại vật cản, cảm biến nhiệt độ/độ ẩm DHT11, cảm biến độ ẩm đất, RFID, vi điều khiển ESP32. Tóm tắt cơ sở lý thuyết về các thiết bị này sẽ được trình bày ở các phần tiếp theo.

1.2. Tổng quan về vi điều khiển

1.2.1. Vi điều khiển là gì ?

Vi điều khiển là một máy tính được tích hợp trên một chip, nó thường được sử dụng để điều khiển các thiết bị điện tử. Vi điều khiển, thực chất, là một hệ thống bao gồm một vi xử lý có hiệu suất đủ dùng và giá thành thấp (khác với các bộ vi xử lý đa năng dùng trong máy tính) kết hợp với các khối ngoại vi như bộ nhớ, các module vào/ra, các module biến đổi số sang tương tự và tương tự sang

số,...

Vi điều khiển thường được sử dụng để xây dựng các hệ thống nhúng.ⁱⁱⁱ Nó cũng được sử dụng trong các thiết bị điện, điện tử như máy giặt, lò vi sóng, điện thoại, đầu đọc DVD, thiết bị đa phương tiện hay dây chuyền sản xuất tự động,...



Hình 1. 3: Vi điều khiển LPC214X

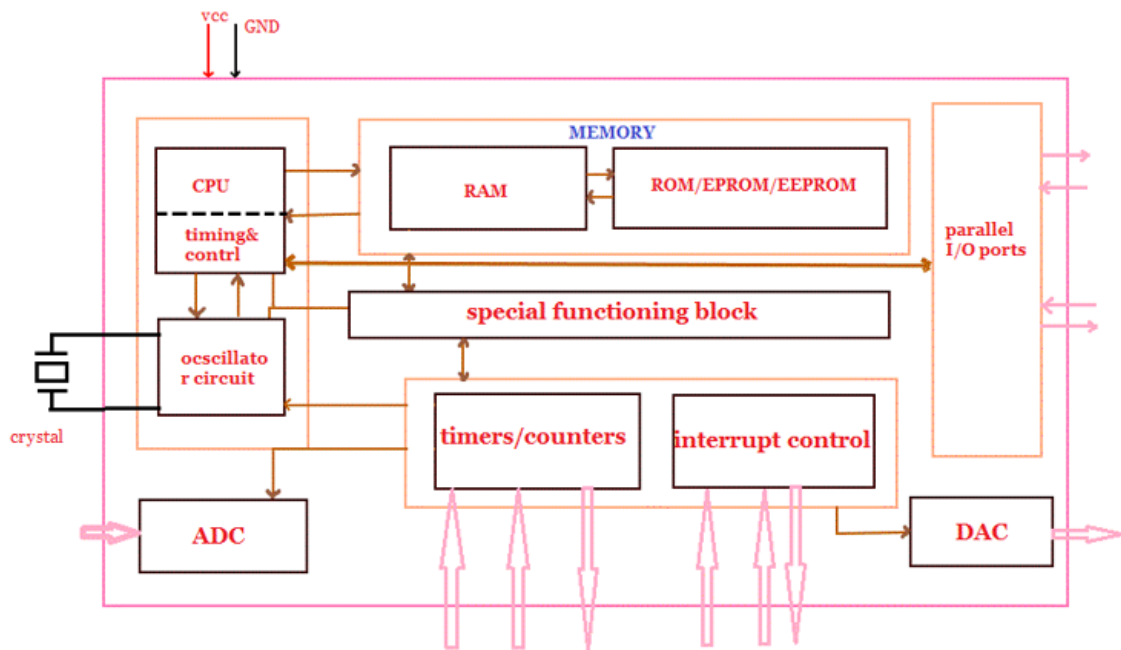
Vi điều khiển thường chứa các linh kiện sau:

- Bộ xử lý trung tâm (CPU).
- Bộ nhớ truy cập ngẫu nhiên (RAM).
- Bộ nhớ chỉ đọc (ROM).
- Cổng đầu vào / đầu ra.
- Bộ đếm thời gian và bộ đếm.
- Bộ chuyển đổi analog sang digital.
- Bộ chuyển đổi digital sang analog.
- Cổng giao tiếp nối tiếp.
- Mạch dao động.

Bên trong vi điều khiển chứa đầy đủ tất cả các tính năng cần thiết cho một hệ thống máy tính và các chức năng như một máy tính mà không cần thêm các bộ phận kỹ thuật số bên ngoài.

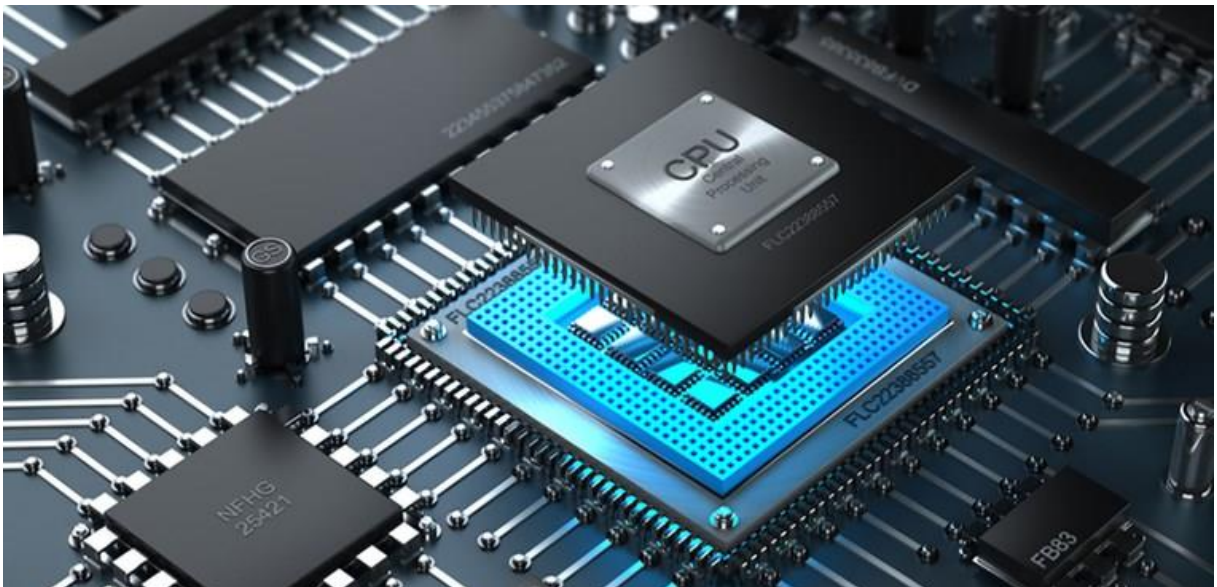
Hầu hết các chân trong chip vi điều khiển có thể được lập trình bởi người dùng. Vi điều khiển có khả năng xử lý các hàm logic. Tốc độ và hiệu suất cao. Cấu trúc on-chip ROM trong vi điều khiển giúp bảo mật firmware tốt hơn. Dễ thiết kế với chi phí thấp và kích thước nhỏ.

1.2.2. Cấu tạo của vi điều khiển



Hình 1. 4: Cấu tạo của vi điều khiển

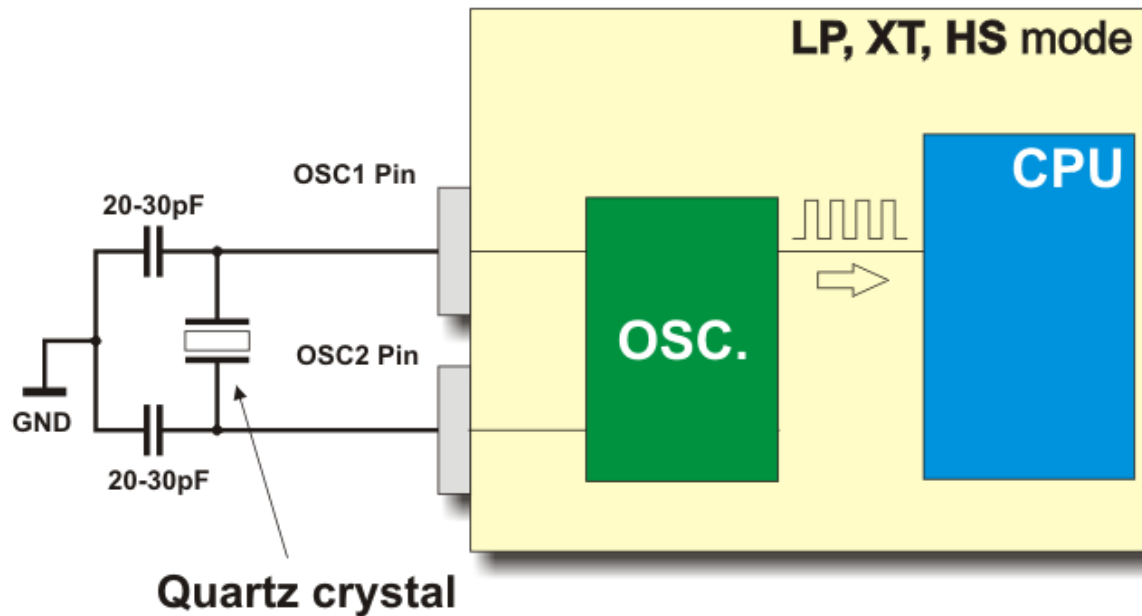
CPU hay Vi xử lý



Hình 1. 5: GPU

CPU (Center Programing Unit) hay bộ xử lý trung tâm là bộ não của vi điều khiển. CPU chịu trách nhiệm tìm nạp lệnh, giải mã và thực thi. CPU kết nối tất cả bộ phận của vi điều khiển vào một hệ thống duy nhất. Chức năng chính của CPU là tìm nạp và giải mã lệnh. Lệnh được lấy từ bộ nhớ chương trình sau đó được CPU giải mã.^{iv}

Oscillator Circuit



Hình 1. 6: Oscillator Circuit

Nếu CPU là bộ não thì Oscillator Circuit hay còn gọi là Clock được coi là trái tim của vi điều khiển. Để mọi thứ hoàn toàn có thể hoạt động giải trí, bắt buộc tất cả chúng ta phải cấp xung, trái tim hoạt động giải trí mới hoàn toàn có thể bơm máu cho hàng loạt khung hình hoạt động giải trí được .

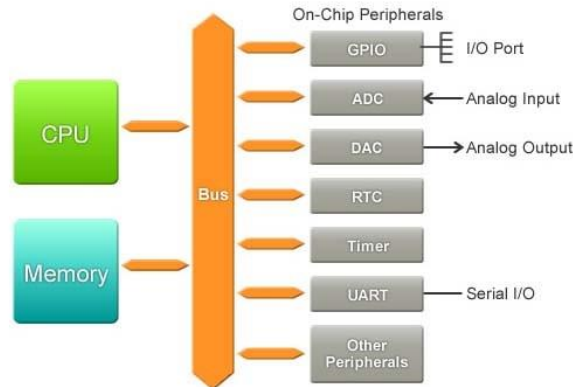
Chúng ta thường nghe quảng cáo dòng vi xử lý có vận tốc bao nhiêu Ghz gì đó, chính là vận tốc Clock mà vi giải quyết và xử lý đó hoàn toàn có thể phân phối được, vận tốc xung càng cao thì vận tốc giải quyết và xử lý của CPU cũng tăng lên. Đương nhiên mọi thứ đều có số lượng giới hạn của nó .^v

Memory – Bộ nhớ^{vi}

Bộ nhớ hoàn toàn có thể coi là một phần không hề thiếu, chúng là nơi tàng trữ chương trình nạp lên hoặc dùng làm nơi chứa những thông tin tức thời mà CPU cần dùng tới. Có 2 kiểu bộ nhớ cơ bản :

- RAM (Random access memory) là bộ nhớ lưu các dữ liệu mà CPU cần dùng để tính toán, đưa ra quyết định, chúng sẽ bị xóa khi mất điện.
- ROM/EPROM/EEPROM hoặc Flash: là bộ nhớ lưu trữ chương trình hay trí khôn của vi điều khiển, chúng được ghi khi chúng ta nạp chương trình vào vi điều khiển, không bị mất khi tắt điện hoặc reset.

Timer/Counter



Hình 1. 7: Timer/Counter

Một vi điều khiển hoàn toàn có thể có nhiều bộ đếm thời hạn và bộ đếm. Bộ đếm thời hạn và bộ đếm có công dụng đếm thời hạn tạo ra những sự kiện để vi điều khiển hoạt động giải trí đúng thời gian .

Các ngoại vi của vi điều khiển

I/O Ports – Input/output: Có thể coi I / O Port là tay chân của vi điều khiển, chúng giúp cho vi điều khiển tương tác với những thành phần khác ngoài môi trường tự nhiên. Cổng đầu vào / đầu ra được sử dụng đa phần điều khiển hoặc tiếp xúc những thiết bị như màn hình hiển thị LCD, đèn LED, máy in, ... cho vi điều khiển.

Các chuẩn giao tiếp: Giống như miệng và tai vậy. Vi điều khiển sẽ sử dụng những chuẩn tiếp xúc khác nhau để liên lạc với nhau hoặc liên lạc với những phân tử khác trên mạch. Có thể kể đến như I2C, SPI, UART, USB,

Bộ chuyển đổi analog sang digital (ADC): Bộ chuyển đổi ADC được sử dụng để quy đổi tín hiệu analog sang dạng digital. Tín hiệu nguồn vào trong bộ chuyển đổi này phải ở dạng analog (ví dụ : đầu ra cảm ứng) và đầu ra từ thiết bị này ở dạng digital. Đầu ra digital hoàn toàn có thể được sử dụng cho những ứng dụng kỹ thuật số (ví dụ : những thiết bị đo lường và thống kê) .

Bộ chuyển đổi Digital sang Analog (DAC): Hoạt động của DAC là đảo ngược của ADC. DAC quy đổi tín hiệu digital thành định dạng analog. Nó thường được sử dụng để điều khiển những thiết bị analog như động cơ DC, những ổ đĩa ...

Interrupt control hay quản lý sự kiện

Ngoài việc thực thi chương trình, vi điều khiển còn phải tương tác với những tác nhân bên trong và bên ngoài. Các tác nhân này sẽ tạo ra những sự kiện gọi là Ngắt, để quản trị nó cần có một khối quản trị ngắt (Interrupt control).

Special functioning block

Một số vi điều khiển chỉ được sử dụng cho 1 số ít ứng dụng đặc biệt quan trọng (ví dụ : mạng lưới hệ thống khoảng trống và rô bốt) những bộ điều khiển này có chứa những công bố trợ để thực thi những hoạt động giải trí đặc biệt quan trọng đó. Đây được coi là khối tính năng đặc biệt quan trọng.

1.3. Vi điều khiển ESP32

1.3.1. ESP32 là gì?

ESP32 là một board mạch vi xử lý, nhằm xây dựng các ứng dụng tương tác với nhau hoặc với môi trường được thuận lợi hơn. ESP32 đã được tích hợp ăng-ten và balun RF, bộ khuếch đại công suất, bộ khuếch đại tiếng ồn thấp, bộ lọc và mô-đun quản lý điện năng. Toàn bộ giải pháp chiếm ít diện tích bảng mạch in. Bảng mạch này được sử dụng với các chip Wifi và bluetooth chế độ kép 2,4 GHz của công nghệ năng lượng thấp TSMC 40nm, năng lượng và tính chất RF tốt nhất, an toàn, đáng tin cậy và có thể mở rộng cho nhiều ứng dụng. Tỷ lệ hiệu suất-giá cao Khối lượng nhỏ, dễ dàng kết hợp với các sản phẩm khác Chức năng mạnh với giao thức LWIP hỗ trợ, Freertos Hỗ trợ 3 chế độ: AP, STA và AP + STA Hỗ trợ chương trình dễ dàng phát triển.

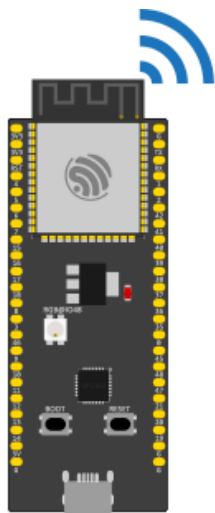
1.3.2. Wifi API ESP32^{vii}

API WiFi cung cấp hỗ trợ cho trình điều khiển giao thức 802.11b/g/n. API này bao gồm:

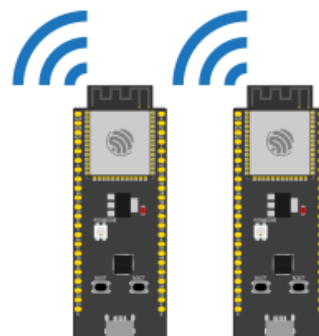
- Chế độ trạm (chế độ STA hoặc chế độ máy khách WiFi). ESP32 kết nối với điểm truy cập.
- Chế độ AP (còn gọi là chế độ Soft-AP hoặc chế độ Điểm truy cập). Các thiết bị kết nối với ESP32.
- Chế độ bảo mật (WPA2, WPA3, v.v.).
- Quét các điểm truy cập.

Làm việc bằng AP

Ở chế độ này, ESP32 được định cấu hình làm Điểm truy cập (AP) và nó có khả năng nhận các kết nối đến từ các thiết bị (trạm) khác bằng cách cung cấp mạng Wifi. Ở chế độ này có thể được sử dụng để phục vụ máy chủ HTTP hoặc HTTPS bên trong ESP32.



ESP32 as Wi-Fi Access Point (AP)



ESP32' as Wi-Fi Station (STA)

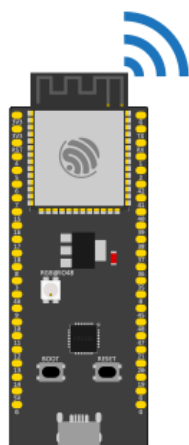
Hình 1. 8: Chế độ AP

Làm việc bằng STA

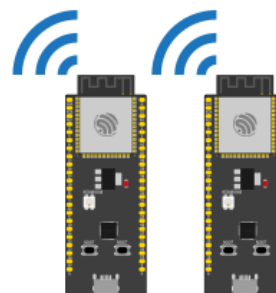
Chế độ STA được sử dụng để kết nối ESP32 với mạng Wifi do Điểm truy cập cung cấp. Đây là chế độ được sử dụng nếu bạn muốn kết nối dự án của mình với Internet.



Wi-Fi Access Point (AP)



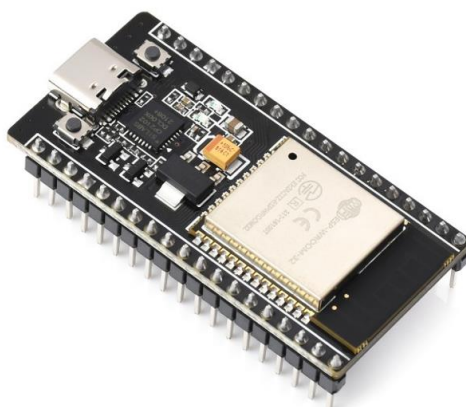
ESP32 as Wi-Fi Station (STA)



ESP32' as Wi-Fi Station (STA)

Hình 1. 9: Chế độ STA

1.3.3. Board NodeMCU ESP32S



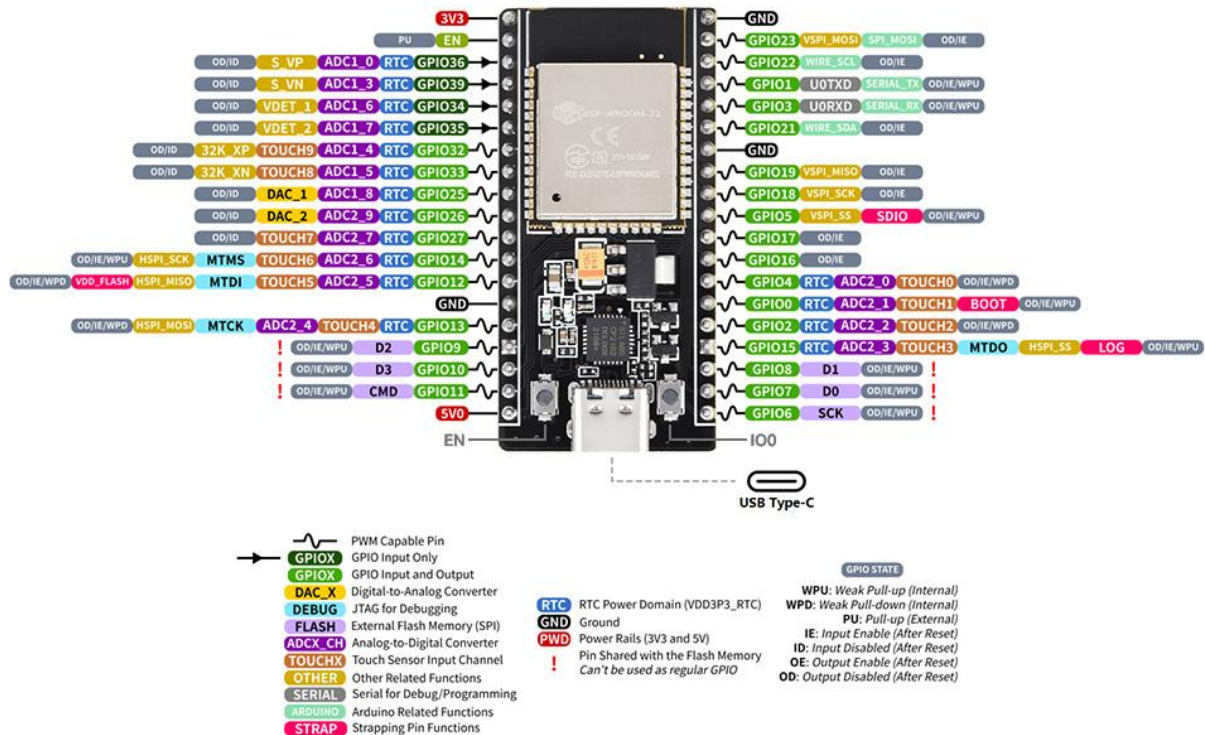
Hình 1. 10: Board NodeMCU ESP32S

NodeMCU ESP32S là board KIT phát triển bởi DOIT, có module điều khiển trung tâm là ESP-WROOM-32. Board dựa trên nền tảng ESP32, hỗ trợ các kết nối Wifi, Bluetooth và các chế độ hoạt động sử dụng năng lượng thấp. Cốt lõi của mô-đun này là chip ESP32-D0WDQ6. Chip nhúng được thiết kế để có thể mở rộng và thích ứng. Có hai lõi CPU có thể được điều khiển riêng và tần số xung nhịp CPU có thể điều chỉnh từ 80 MHz đến 240 MHz. Người dùng cũng có thể tắt nguồn CPU và sử dụng bộ đồng xử lý công suất thấp để liên tục theo dõi các thiết bị ngoại vi để biết các thay đổi hoặc vượt qua ngưỡng. ESP32 tích hợp một bộ thiết bị ngoại vi phong phú, từ cảm ứng điện dung, cảm biến Hall, giao diện thẻ SD, Ethernet, SPI tốc độ cao, UART, I2S và I2C.

Thông số cơ bản NodeMCU ESP32S^{viii}

- Vi điều khiển: ESP32-D0WDQ6, Dual-core Xtensa® 32-bit LX6 microprocessor, operating at 160 or 240 MHz and performing at up to 600 DMIPS.
- Bộ nhớ: 520KBytes SRAM, 448KBytes ROM.
- Kết nối không dây: Wifi: 802.11 b/g/n/e/i, Bluetooth BR/EDR & BLE.
- Giao tiếp: UART: 3, SPI: 4, I2C: 2, I2S: 2, SDIO: 1, GPIO: 30, ADC: 16-bit SAR ADC, DAC: 2 × 8-bit, Touch sensor: 10, Temperature sensor, RTC
- Cấp nguồn: 2.3 - 3.6V.
- Kích thước: 18 x 25.5 x 3.1mm.

Các chân của NodeMCU ESP32S



Hình 1. 11: Các chân của Board NodeMCU ESP32S

NodeMCU ESP32S có 30 chân GPIO, được chia thành 2 loại:

Chân GPIO tiêu chuẩn: Các chân này có thể được sử dụng cho nhiều mục đích khác nhau, chẳng hạn như kết nối với các thiết bị ngoại vi, điều khiển các thiết bị điện,...

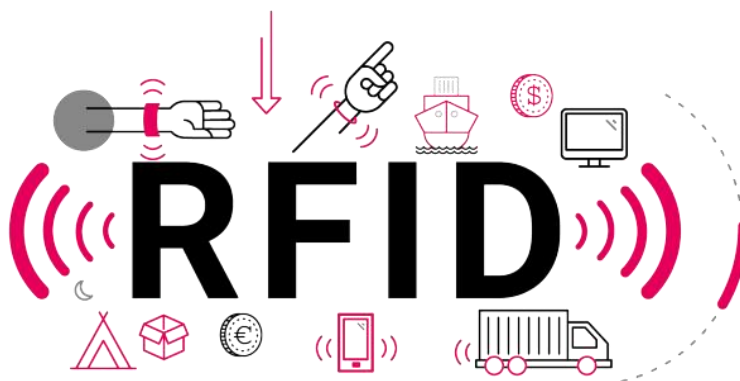
Chân GPIO chuyên dụng: Các chân này được thiết kế cho các mục đích cụ thể, chẳng hạn như:

- Chân GPIO 34 và 35: Chân này được sử dụng cho kết nối SPI.
- Chân GPIO 21 và 22: Chân này được sử dụng cho kết nối I2C.
- Chân GPIO 23: Chân này được sử dụng cho kết nối I2S.
- Chân GPIO 26: Chân này được sử dụng cho kết nối SDIO.
- Chân GPIO 27: Chân này được sử dụng cho kết nối ADC.
- Chân GPIO 28: Chân này được sử dụng cho kết nối DAC.
- Chân GPIO 32: Chân này được sử dụng cho kết nối cảm biến chạm.
- Chân GPIO 36: Chân này được sử dụng cho kết nối cảm biến nhiệt độ.
- Chân GPIO 37: Chân này được sử dụng cho kết nối RTC.

1.4. Các module / cảm biến

1.4.1. RFID

RFID là gì?



Hình 1. 12: RFID

RFID (Là tên viết tắt của cụm từ tiếng anh: Radio Frequency Identification) là công nghệ định vị và nhận dạng các đối tượng bằng cách sử dụng sóng vô tuyến. Công nghệ này sử dụng các thẻ RFID chứa thông tin được lưu trữ trên một chip và gửi đến đầu đọc thông qua sóng vô tuyến.

Đây là ứng dụng có thể được sử dụng để theo dõi sản phẩm trong chuỗi cung ứng, theo dõi hành lý trên các chuyến bay, quản lý kho bãi, kiểm soát truy cập và nhiều ứng dụng khác. Công nghệ này cũng được sử dụng trong các bài kiểm tra an ninh, ví dụ như thẻ thông minh trong hệ thống kiểm soát ra vào tại sân bay.

Đây là một công nghệ phát triển nhanh chóng và đang được áp dụng rộng rãi trong nhiều lĩnh vực khác nhau. Công nghệ này giúp tăng cường sự chính xác, nâng cao hiệu quả vận hành và đảm bảo an toàn trong quản lý và kiểm soát các đối tượng.

Đặc điểm của RFID

Tự động hóa: RFID có thể tự động xác định và theo dõi các đối tượng, giúp giảm bớt sự can thiệp của con người.

Không tiếp xúc: RFID không yêu cầu tiếp xúc vật lý giữa thẻ và đầu đọc, giúp tăng độ bền và an toàn.

Tầm xa: RFID có thể hoạt động ở khoảng cách xa, giúp tăng năng suất và hiệu quả.

Dữ liệu nhiều: RFID có thể chứa nhiều dữ liệu, giúp tăng tính linh hoạt của hệ thống.

Nguyên lý hoạt động của RFID



Hình 1. 13: hoạt động của RFID

Công nghệ Radio Frequency Identification hoạt động dựa trên việc sử dụng sóng vô tuyến để truyền tải dữ liệu giữa thẻ RFID và đầu đọc RFID. Các thành phần chính của hệ thống này bao gồm:

- Thẻ RFID: Thẻ RFID chứa thông tin được lưu trữ trên một chip và được phát tán qua một anten.
- Đầu đọc RFID: là thiết bị đọc và ghi dữ liệu từ các thẻ RFID thông qua sóng vô tuyến.
- Anten: Anten được sử dụng để truyền tải dữ liệu giữa thẻ và đầu đọc thông qua sóng vô tuyến.

Khi thẻ RFID được đưa vào vùng phủ sóng của đầu đọc RFID, anten của thẻ sẽ bắt đầu phát tín hiệu sóng vô tuyến. Đầu đọc RFID sẽ bắt đầu thu sóng vô tuyến này và dùng thông tin trong sóng vô tuyến để đọc dữ liệu từ thẻ. Sau khi đọc xong, đầu đọc có thể ghi dữ liệu mới lên thẻ nếu cần thiết.

Sóng vô tuyến được sử dụng trong công nghệ này có thể hoạt động ở nhiều tần số khác nhau tùy thuộc vào ứng dụng cụ thể. Thông thường, các hệ thống này sử dụng các tần số từ 125kHz đến 5.8GHz. Các loại thẻ RFID có thể là pasive (không có nguồn điện riêng) hoặc active (có nguồn điện riêng), tuy nhiên cả hai loại đều có thể được đọc và ghi dữ liệu từ xa thông qua sóng vô tuyến.

Tóm lại, công nghệ này hoạt động bằng cách sử dụng sóng vô tuyến để truyền tải dữ liệu giữa thẻ và đầu đọc. Việc đọc và ghi dữ liệu được thực hiện thông qua các

anten và tần số sóng vô tuyến tương ứng.^{ix}

1.4.2. Cảm biến khí^x

Cảm biến khí là gì?

Cảm biến khí MQ2 là một trong những cảm biến được sử dụng rộng rãi nhất trong các dòng cảm biến MQ. Nó là thiết bị phát hiện sự hiện diện hoặc nồng độ của khí trong khí quyển. Dựa trên nồng độ của khí, cảm biến tạo ra sự chênh lệch điện thế tương ứng bằng cách thay đổi điện trở của vật liệu bên trong cảm biến, có thể đo được bằng điện áp đầu ra. Dựa trên giá trị điện áp này, loại và nồng độ của khí có thể được ước tính.



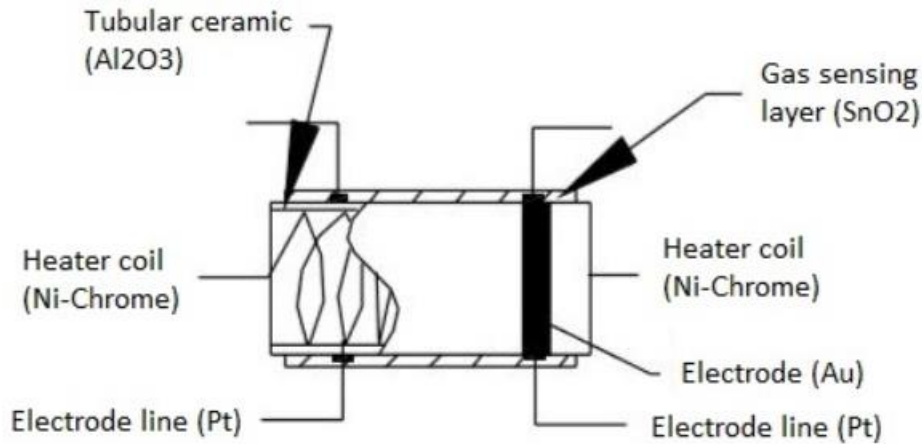
Hình 1. 14: Cảm biến khí ga MQ2

Loại khí mà cảm biến có thể phát hiện tùy thuộc vào vật liệu cảm biến có bên trong cảm biến. Thông thường, các cảm biến này có sẵn dưới dạng mô-đun với bộ so sánh như trình bày ở trên. Những bộ so sánh này có thể được đặt cho một giá trị ngưỡng cụ thể của nồng độ khí. Khi nồng độ khí vượt quá ngưỡng này, chân kỹ thuật số sẽ ở mức cao. Chân analog có thể được sử dụng để đo nồng độ của khí. Cảm biến MQ-2 có khả năng phát hiện nhiều loại khí khác nhau như khí kháng, khí nhiên độc (như CO - Carbon Monoxide) và khí hợp chất dễ cháy (như LPG - Liquefied Petroleum Gas). Cảm biến có thể nhận biết khí kháng dựa trên sự thay đổi trong kháng độ của nó khi tiếp xúc với các khí kháng cụ thể.

Cấu tạo

Lớp cảm biến khí: Đây là thành phần chính trong cảm biến có thể được sử dụng để cảm nhận sự thay đổi nồng độ của khí và tạo ra sự thay đổi điện trở. Lớp cảm biến khí về cơ bản là một điện trở hóa học thay đổi giá trị điện trở của nó dựa trên.

Nồng độ của khí cụ thể trong môi trường. Ở đây, phân tử cảm biến được tạo thành từ Thiếc Dioxide (SnO_2), nói chung, có lượng electron dư thừa (phần tử cho). Vì vậy, bất cứ khi nào phát hiện ra khí độc, điện trở của nguyên tố đó sẽ thay đổi và dòng điện chạy qua nó sẽ thay đổi, điều này thể hiện sự thay đổi nồng độ của khí.



Hình 1. 15: Cấu tạo của cảm biến Khí

Cuộn dây sưởi: Mục đích của cuộn dây gia nhiệt là đốt cháy phân tử cảm biến để độ nhạy và hiệu quả của phân tử cảm biến tăng lên. Nó được làm bằng Niken-Chromium có điểm nóng chảy cao để có thể giữ nóng mà không bị tan chảy.

Dòng điện cực: Vì phân tử cảm biến tạo ra dòng điện rất nhỏ khi phát hiện thấy khí, điều quan trọng hơn là phải duy trì hiệu quả mang các dòng điện nhỏ đó. Vì vậy, dây bạch kim phát huy tác dụng giúp di chuyển các electron một cách hiệu quả.

Điện cực: Là điểm nối nơi đầu ra của lớp cảm biến được kết nối với đường dây Điện cực. Để dòng điện đầu ra có thể chảy đến thiết bị đầu cuối cần thiết. Điện cực ở đây được làm bằng Vàng (Au – Aurum) có tính dẫn điện rất tốt.

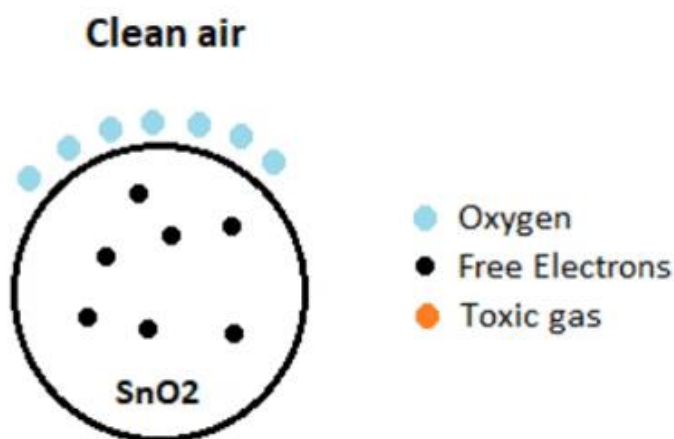
Gốm hình ống: Ở giữa cuộn dây gia nhiệt và lớp cảm biến khí tồn tại gốm ống được làm từ nhôm oxit (Al_2O_3). Vì nó có điểm nóng chảy cao nên nó giúp duy trì hiện tượng đốt nóng (làm nóng trước) của lớp cảm biến, mang lại độ nhạy cao cho lớp cảm biến để có được dòng điện đầu ra hiệu quả.

Lưới thép bên trên cảm biến: Để bảo vệ các phân tử cảm biến và thiết lập, một lưới kim loại được sử dụng phía trên nó, cũng được sử dụng để tránh/giữ các hạt bụi xâm nhập vào lưới và ngăn ngừa làm hỏng lớp cảm biến khí khỏi các hạt ăn mòn.

Nguyên lý hoạt động

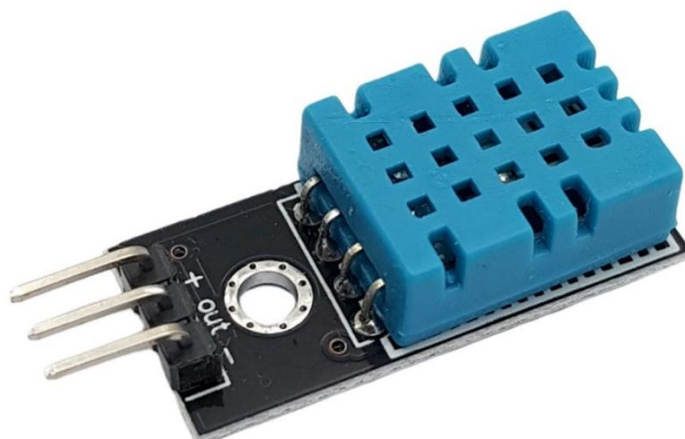
Khả năng phát hiện khí của cảm biến khí phụ thuộc vào điện trở hóa học để dẫn dòng điện. Chất hóa học được sử dụng phổ biến nhất là Tin Dioxide (SnO_2) là chất bán dẫn loại n có các electron tự do (còn gọi là chất cho). Thông thường bầu không khí sẽ chứa nhiều oxy hơn các loại khí dễ cháy. Các hạt oxy thu hút các electron tự do có trong SnO_2 đẩy chúng lên bề mặt SnO_2 . Vì không có electron tự do nên dòng điện đầu ra sẽ bằng không. Ảnh gif bên dưới cho thấy các phân tử oxy (màu xanh lam) thu hút các electron tự do (màu đen) bên trong SnO_2 và ngăn không cho nó có các electron tự do để dẫn dòng điện.

Khi cảm biến được đặt trong môi trường khí độc hoặc dễ cháy, khí khử này (màu cam) sẽ phản ứng với các hạt oxy bị hấp phụ và phá vỡ liên kết hóa học giữa oxy và các electron tự do đó giải phóng các electron tự do. Khi các electron tự do trở lại vị trí ban đầu, giờ đây chúng có thể dẫn dòng điện, sự dẫn điện này sẽ tỷ lệ thuận với lượng electron tự do có sẵn trong SnO_2 , nếu khí có độc tính cao thì sẽ có nhiều electron tự do hơn.



Hình 1. 16: Nguyên lý hoạt động của cảm biến Khí

1.4.3. DHT11 – Cảm biến nhiệt độ, độ ẩm



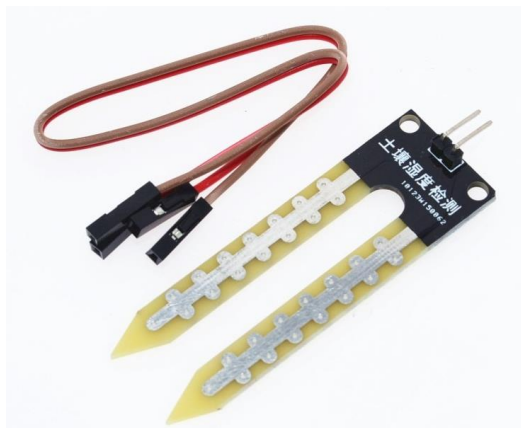
Hình 1. 17: Cảm biến nhiệt độ - độ ẩm

Cảm biến nhiệt độ và độ ẩm DHT11 có tổ hợp cảm biến nhiệt độ và độ ẩm với đầu ra tín hiệu số được hiệu chỉnh. Bằng cách sử dụng kỹ thuật thu tín hiệu số độc quyền và công nghệ cảm biến nhiệt độ & độ ẩm, nó đảm bảo độ tin cậy cao và độ ổn định lâu dài tuyệt vời. Cảm biến này bao gồm thành phần đo độ ẩm loại điện trở và thành phần đo nhiệt độ NTC, đồng thời kết nối với bộ vi điều khiển 8 bit hiệu suất cao, mang lại chất lượng tuyệt vời, phản hồi nhanh, khả năng chống nhiễu và tiết kiệm chi phí.

Thông số kỹ thuật của DHT11:

- Điện áp hoạt động: 3,5V đến 5,5V.
- Dòng điện hoạt động: 0,3mA (đo) 60uA (chế độ chờ).
- Đầu ra: Dữ liệu nối tiếp.
- Phạm vi nhiệt độ: 0°C đến 50°C.
- Phạm vi độ ẩm: 20% đến 90%.
- Độ phân giải: Nhiệt độ và độ ẩm đều là 16-bit.
- Độ chính xác: $\pm 1^{\circ}\text{C}$ và $\pm 1\%$.

1.4.4. Cảm biến độ ẩm đất

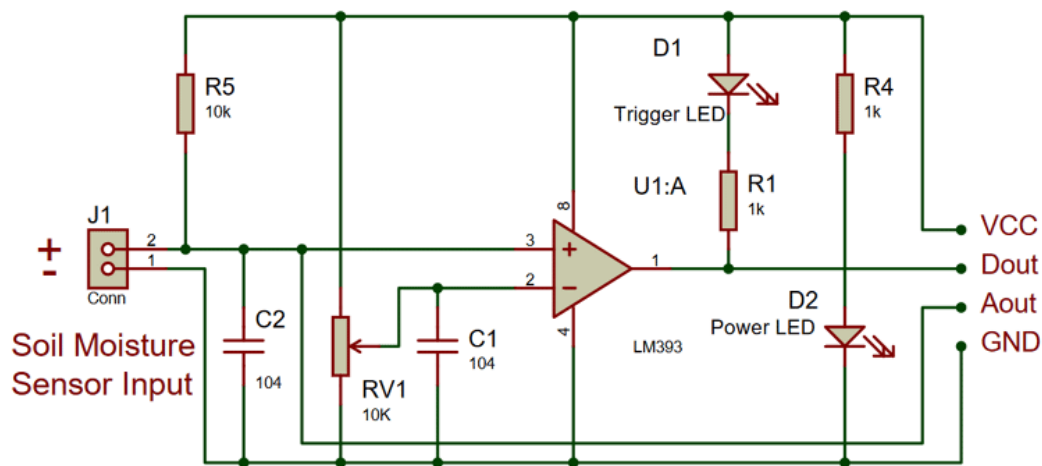


Hình 1. 18: Cảm biến độ ẩm đất

Module cảm biến độ ẩm đất này được sử dụng để phát hiện độ ẩm của đất. Nó đo hàm lượng thể tích của nước bên trong đất và cung cấp cho chúng ta độ ẩm đầu ra, khi đất thiếu nước thì đầu ra của module ở mức cao, nếu không thì đầu ra ở mức thấp. Mô-đun này có cả đầu ra kỹ thuật số và analog và một chiết áp để điều chỉnh các mức. Bằng cách sử dụng cảm biến này, người ta có thể tự động tưới nước cho cây hoa hoặc bất kỳ loại cây nào khác cần kỹ thuật tưới nước tự động.

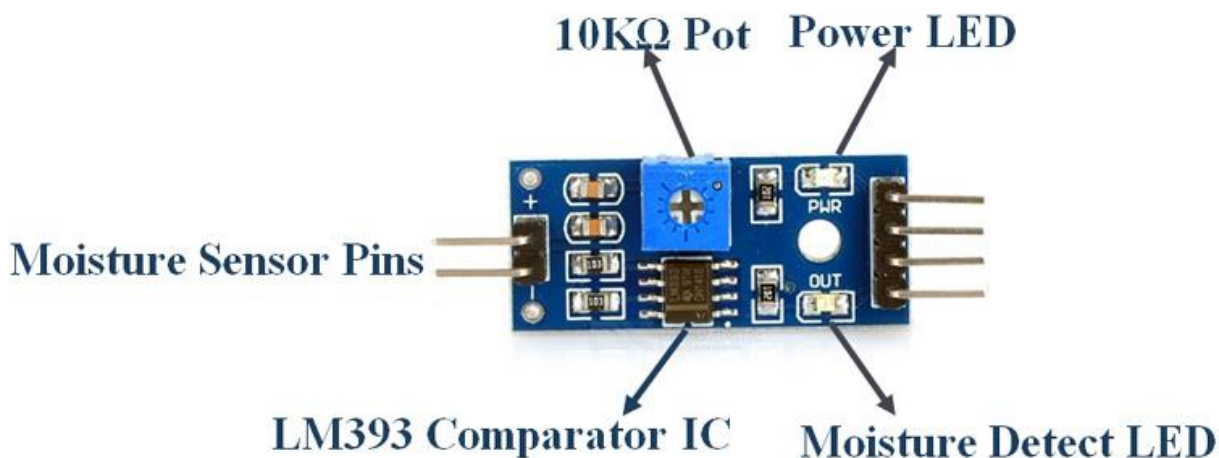
Thông số kỹ thuật và tính năng của Module cảm biến độ ẩm đất:

- Điện áp hoạt động: 3,3V đến 5V DC.
- Dòng điện hoạt động: 15mA.
- Đầu ra kỹ thuật số - 0V đến 5V, Mức kích hoạt có thể điều chỉnh từ giá trị đặt trước.
- Đầu ra Analog - 0V đến 5V dựa trên bức xạ hồng ngoại từ ngọn lửa rơi vào cảm biến.
- Đèn LED cho biết đầu ra và nguồn điện.
- Kích thước PCB: 3,2cm x 1,4cm.
- Thiết kế dựa trên LM393.
- Dễ dàng sử dụng với Bộ vi điều khiển hoặc thậm chí với IC Kỹ thuật số/Analog thông thường.
- Nhỏ, rẻ và dễ dàng có sẵn.



Hình 1. 19: Sơ đồ mạch của cảm biến độ ẩm đất

Mô-đun cảm biến độ ẩm bao gồm: cảm biến độ ẩm, điện trở, tụ điện, thiết áp, IC so sánh LM393, đèn LED nguồn và trạng thái trong một mạch tích hợp.

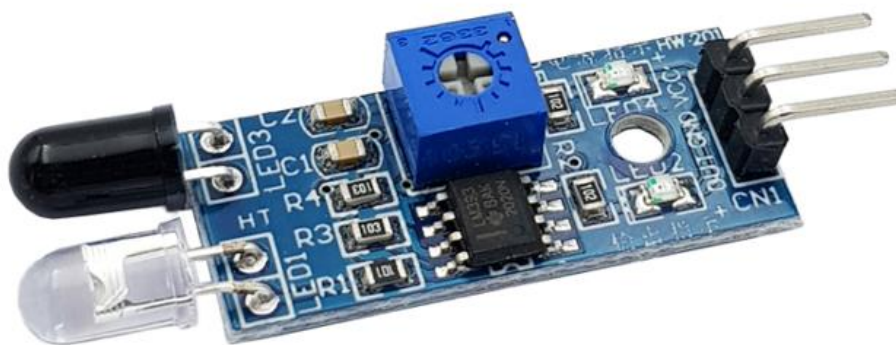


Hình 1. 20: Cấu tạo của cảm biến độ ẩm đất

IC LM393: IC so sánh LM393 được sử dụng làm bộ so sánh điện áp trong mô-đun cảm biến độ ẩm này. Chân 2 của LM393 được kết nối với Preset (Nồi 10KΩ) trong khi chân 3 được kết nối với chân cảm biến Độ ẩm. IC so sánh sẽ so sánh điện áp ngưỡng được đặt bằng cách sử dụng giá trị đặt trước (pin2) và chân cảm biến (pin3).

Cảm biến độ ẩm: cảm biến độ ẩm bao gồm hai đầu dò được sử dụng để phát hiện độ ẩm của đất . Các đầu dò cảm biến độ ẩm được phủ một lớp vàng ngậm để bảo vệ Niken khỏi quá trình oxy hóa. Hai đầu dò này được sử dụng để truyền dòng điện qua đất và sau đó cảm biến sẽ đọc điện trở để lấy giá trị độ ẩm.

1.4.5. Cảm biến hồng ngoại vật cản vật cản



Hình 1. 21: Cảm biến hồng ngoại vật cản vật cản

Cảm biến hồng ngoại vật cản là gì?

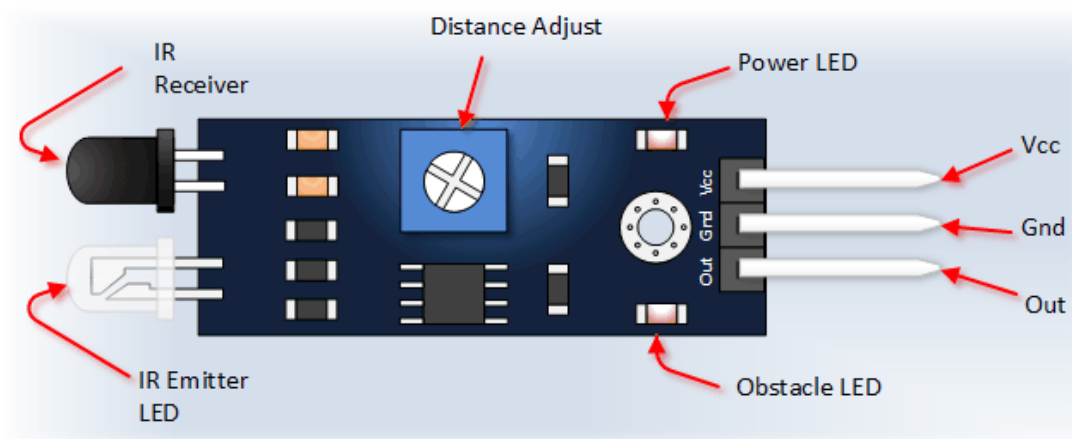
Cảm biến vật cản hồng ngoại sử dụng một cặp truyền và nhận tia hồng ngoại. Tia hồng ngoại phát một tần số nhất định khi gặp vật cản sẽ phản xạ vào đèn thu hồng ngoại, sau khi qua IC so sánh đèn màu xanh sẽ sáng lên, đồng thời cho tín hiệu số đầu ra.

Chúng ta có thể thấy rằng mắt phát hồng ngoại truyền trong một phạm vi giới hạn và nó đi đến một khoảng cách nhất định vì không có đối tượng phản xạ lại nên mắt thu hồng ngoại không thể phát hiện vật cản, ngược lại khi gặp vật cản ánh sáng hồng ngoại được phản chiếu lại khi đó mắt thu sẽ phát hiện được vật cản và xuất tín hiệu ra.

Ngoài ra độ nhạy của cảm biến còn phụ thuộc vào về mặt vật cản gặp phải, ví dụ như các vật có xu hướng màu trắng có khả năng phản xạ ánh sáng hồng ngoại mạnh hơn các vật cản có màu tối, các vật cản màu tối hấp thụ ánh sáng hồng ngoại nhiều hơn và phản xạ lại ít hơn số lượng không đủ để mắt thu có thể phát hiện.

Cấu tạo

Mô-đun Cảm biến hồng ngoại vật cản bao gồm chủ yếu là Bộ phát và thu hồng ngoại, Op-amp, Điện trở biến đổi (Nồi tông đơ), đèn LED đầu ra cùng với một số điện trở.



Hình 1. 22: Cấu tạo của Cảm biến hồng ngoại vật cản

Máy phát LED hồng ngoại: IR LED phát ra ánh sáng, trong dải tần số Hồng ngoại. Chúng ta không thể nhìn thấy ánh sáng hồng ngoại vì bước sóng của nó (700nm – 1mm) cao hơn nhiều so với phạm vi ánh sáng khả kiến. Đèn LED hồng ngoại có góc phát sáng khoảng. 20-60 độ và phạm vi khoảng. vài cm đến vài feet, tùy thuộc vào loại thiết bị phát hồng ngoại và nhà sản xuất. Một số máy phát có phạm vi tính bằng km. IR LED có màu trắng hoặc trong suốt nên có thể phát ra lượng ánh sáng tối đa.

Bộ thu diode quang: Photodiode hoạt động như bộ thu IR khi nó dẫn điện khi ánh sáng chiếu vào nó. Photodiode là một chất bán dẫn có điểm nối PN, hoạt động ở chế độ Phân cực ngược, có nghĩa là nó bắt đầu dẫn dòng điện theo hướng ngược lại khi Ánh sáng chiếu vào nó và lượng dòng điện tỷ lệ thuận với lượng Ánh sáng. Thuộc tính này làm cho nó hữu ích cho việc phát hiện IR. Photodiode trông giống như một đèn LED, với lớp phủ màu đen ở mặt ngoài, Màu đen hấp thụ lượng ánh sáng cao nhất.

Bộ khuếch đại LM358: LM358 là Bộ khuếch đại hoạt động (Op-Amp) được sử dụng làm bộ so sánh điện áp trong Cảm biến hồng ngoại vật cản. bộ so sánh sẽ so sánh điện áp ngưỡng được đặt bằng cách sử dụng giá trị đặt trước (pin2) và điện áp điện trở nối tiếp của photodiode (pin3).

Sự áp điện trở nối tiếp của Photodiode > Điện áp ngưỡng = Đầu ra Opamp ở mức Cao.

Giảm điện áp nối tiếp của photodiode < Ngưỡng điện áp = Đầu ra Opamp ở mức Thấp.

Khi đầu ra của Opamp ở *mức cao* , đèn LED ở đầu ra của Opamp sẽ *bật* (Biểu thị việc phát hiện Đối tượng).

Biến trở: Biến trở được sử dụng ở đây là một giá trị đặt trước. Nó được sử dụng để hiệu chỉnh phạm vi khoảng cách mà đối tượng cần được phát hiện.

Thông số kỹ thuật.

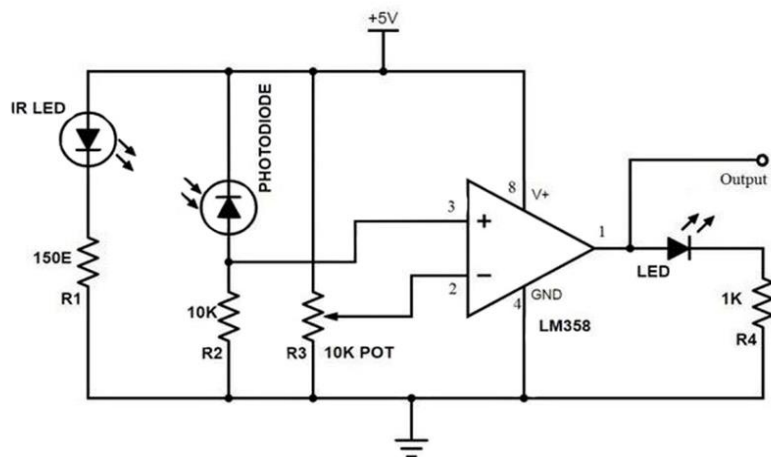
Điện áp cung cấp cho mạch: 3.3V-5V.

Bộ so sánh dùng Op Amp LM358 hoặc LM393.

Đầu ra kỹ thuật số mức (1 và 0).

Nguyên lý hoạt động

Led phát hồng ngoại (IR LED) luôn phát ra sóng ánh sáng có bước sóng hồng ngoại mắt người không thể nhìn thấy ánh sáng này, vì vậy người ta sử dụng led thu hồng ngoại, led thu hồng ngoại bình thường nó có nội trở rất lớn (vài trăm K Ω), khi led thu được tia hồng ngoại chiếu vào đủ lớn thì nội trở của nó giảm xuống (cỡ vài chục Ω).



Hình 1. 23: Sơ đồ mạch nguyên lý của cảm biến hồng ngoại vật cản

Khi gặp vật cản, những chùm tia hồng ngoại gặp vật cản và phản xạ lại led thu làm led thu thay đổi giá trị điện trở. Ở đây chúng ta thấy cầu chia áp ở điện trở R2 và mắt thu hồng ngoại, sự thay đổi điện trở của mắt thu hồng ngoại dẫn đến điện áp đầu vào chân 3 Op-Amp cũng thay đổi.

Khi khoảng cách càng gần, sự thay đổi càng lớn.

Khi đó, điện áp đầu vào chân 3 Op-Amp được so sánh với giá trị điện áp không đổi ghim trên biến trở R3, nếu điện áp chân 3 Op-Amp lớn hơn điện áp chân 2

Op-Amp thì Op-Amp xuất mức 1 (bằng VCC) . Ngược lại nếu điện áp chân 3 Op-Amp nhỏ hơn điện áp chân 2 Op-Amp thì Op-Amp xuất mức 0 (bằng GND) Điện trở như R1 (150 Ω), R2 (10K Ω), R4 (1 K Ω) giúp các led hoạt động mà không bị cháy.

Biến trở R3 dùng để chỉnh độ nhạy của biến trở.

1.5. Kết luận chương

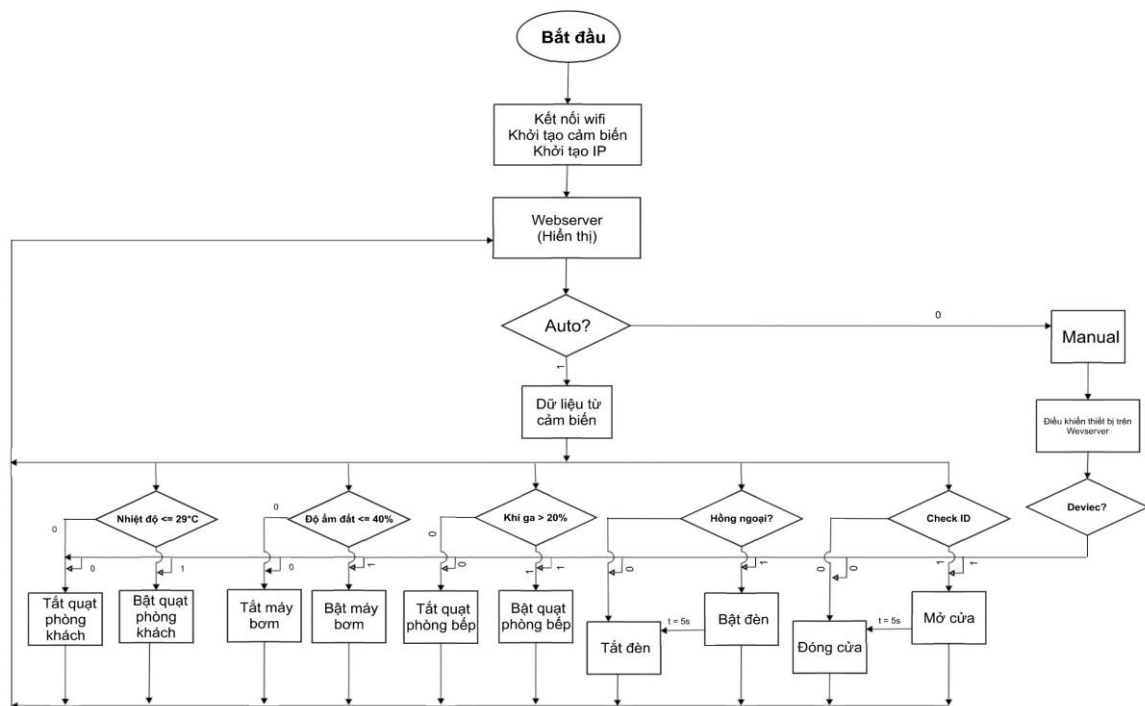
Trong chương này, chúng em đã tìm hiểu chi tiết cấu tạo cũng như nguyên lý hoạt động của các module và Vi điều khiển ESP32 được sử dụng để thiết kế mô hình “ Nhà thông minh”. Chương này bắt đầu bằng việc định nghĩa và giải thích khái niệm "nhà thông minh", từ đó nêu rõ những lợi ích mà nó mang lại cho cuộc sống của con người. Ngoài ra, chương cũng đi sâu vào việc trình bày các chức năng và tính năng quan trọng của nhà thông minh, cũng như các thiết bị cần có để xây dựng một hệ thống nhà thông minh hoàn chỉnh. Tiếp theo là giới thiệu khái niệm "vi điều khiển" và trình bày cấu tạo cơ bản của vi điều khiển. Đây là yếu tố quan trọng để hiểu cách hệ thống nhà thông minh hoạt động và làm thế nào để tương tác với nó. Tiếp tục bằng việc giới thiệu một loại vi điều khiển cụ thể, đó là ESP32, với việc cung cấp thông tin về nó và một số board ESP32 phổ biến. Thông qua việc tìm hiểu về ESP32, người đọc có cơ hội hiểu rõ hơn về công nghệ vi điều khiển được sử dụng trong hệ thống nhà thông minh. Cuối cùng, chương này giới thiệu các module và cảm biến quan trọng mà hệ thống nhà thông minh có thể sử dụng, bao gồm cảm biến RFID, cảm biến khí, cảm biến nhiệt độ và độ ẩm DHT11, cảm biến độ ẩm đất và Cảm biến hồng ngoại vật cản. Những thông tin này sẽ cung cấp nền tảng quan trọng cho việc xây dựng các ứng dụng cụ thể trong tương lai. Chương 2 đã giúp định hình cơ sở lý thuyết và kiến thức căn bản về nhà thông minh và vi điều khiển, chuẩn bị cho các phần tiếp theo của nghiên cứu và phát triển hệ thống nhà thông minh.

CHƯƠNG II

THIẾT KẾ HỆ THỐNG

Chương trước đã giới thiệu về các cơ sở lý thuyết của một hệ thống nhà thông minh và các linh kiện cần thiết cho việc thiết kế và xây dựng mô hình nhà thông minh được chúng em thực hiện trong đề tài này. Trong chương này, chúng ta sẽ đi sâu vào việc thiết kế các thành phần của hệ thống nhà thông minh. Đầu tiên chúng ta sẽ đi qua lưu đồ thuật toán.

Lưu đồ thuật toán



Hình 2. 1: Lưu đồ thuật toán

Hệ thống nhà thông minh được vận hành như sau:

Khi hệ thống nhà thông minh bắt đầu hoạt động, hệ thống sẽ kết nối wifi, khởi tạo các cảm biến (cảm biến nhiệt độ - độ ẩm, cảm biến độ ẩm đất, cảm biến khí ga, cảm biến chuyển động hồng ngoại, cảm biến chạm (RFID)) và khởi tạo webserver (dữ liệu của các cảm biến cũng sẽ hiển thị ở trên webserver). Chế độ hoạt động mặc định của hệ thống là Auto, hệ thống sẽ kiểm tra xem đang ở chế độ hoạt động nào, Auto hay Manual.

Ở chế độ Auto: các dữ liệu sẽ được truyền từ các cảm biến đến hệ thống điều

hiển để phân tích và kiểm tra dữ liệu sau đó đưa ra điều khiển tương thích với các thiết bị trong nhà:

- Nhiệt độ trên 29°C thì quạt phòng khách sẽ được bật.
- Độ ẩm đất dưới 40% thì máy bơm sẽ hoạt động.
- Khí ga trên 20% thì sẽ bật quạt phòng bếp sẽ được bật.
- Khi có vật cản qua Cảm biến hồng ngoại vật cản được lắp ở phòng khách và phòng ngủ thì đèn tương ứng trong hai phòng sẽ được bật lên.

Ở chế độ Manual: các thiết bị trong nhà sẽ được điều khiển trực tiếp thông qua webserver.

2.1. Mô tả hệ thống

2.1.1. Giới thiệu

Hệ thống nhà thông minh hoạt động theo hai chức năng là Auto hoặc Manual. Người dùng có thể chọn một trong hai chức năng rồi bắt đầu sử dụng. Khi bắt đầu, tín hiệu từ các cảm biến được chuyển đổi thành tín hiệu điều khiển thông qua khối code và được xử lý bởi khối xử lý để điều khiển các thiết bị trong nhà theo ý của người dùng. Nhà thông minh hiện tại đang ngày càng phổ biến hơn và các tính năng của nó luôn được cập nhật, nâng cấp và đổi mới. Nhà thông minh cũng đem lại rất nhiều lợi ích cho người dùng.

2.1.2. Tính năng

Các tính năng của nhà thông minh:

Điều khiển thông qua webserver: Để tạo sự tiện lợi cũng như giám sát được ngôi nhà của người dùng, thì mô hình này có liên kết với webserver để có thể theo dõi cũng như điều khiển ngôi nhà.

Bật tắt đèn tự động: Bật – tắt đèn tự động đã là tính năng gần như bắt buộc trong mỗi thiết kế của nhà thông minh. Hệ thống có thể bật – tắt đèn theo ý của người dùng bằng cách sử dụng cảm biến vật cản hồng ngoại hoặc thao tác trên webserver đi kèm.

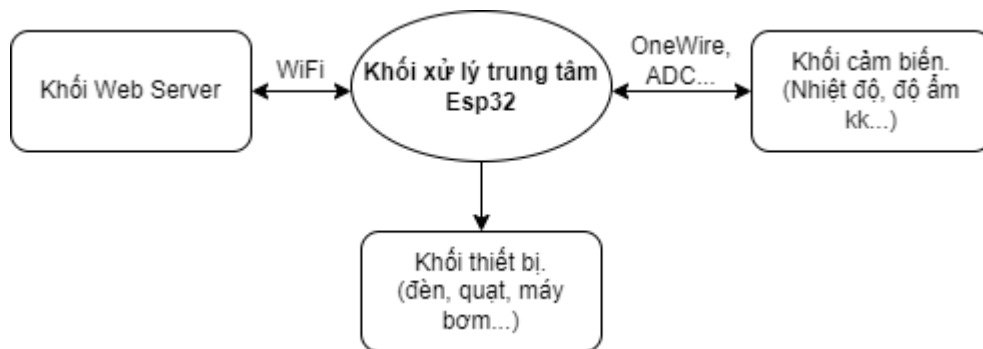
Tưới nước tự động: Với tính năng này người dùng có thể theo dõi độ ẩm của đất cũng như tưới nước cho cây khi không có mặt ở đó. Ở tính năng này người dùng có thể bật chế độ tưới nước tự động khi độ ẩm dưới 40% hoặc bật tắt máy bơm

thông qua webserver.

Cửa thông minh: Để đảm bảo an toàn cho ngôi nhà, cửa của mô hình Nhà thông minh đã được trang bị công nghệ RFID. Người dùng chỉ có thể vào nhà bằng thẻ từ đã được định danh hoặc bằng webserver.

Hệ thống cảnh báo rò rỉ khí ga: Khi trong nhà có nồng độ khí ga lớn hơn 20%, quạt phòng bếp sẽ được bật để thổi khí ga bay ra ngoài nhằm đảm bảo an toàn cho người nhà và tài sản.

2.2. Sơ đồ hệ thống



Hình 2. 2: Sơ đồ hệ thống

Hệ thống bao gồm 4 khối chính:

Khối xử lý trung tâm ESP32: có chức năng thu thập dữ liệu từ các cảm biến để gửi dữ liệu lên webserver và điều khiển các thiết bị.

Khối webserver: nhận dữ liệu từ ESP32 và gửi tín hiệu điều khiển cho ESP32 để điều khiển thiết bị.

Khối cảm biến: bao gồm các cảm biến:

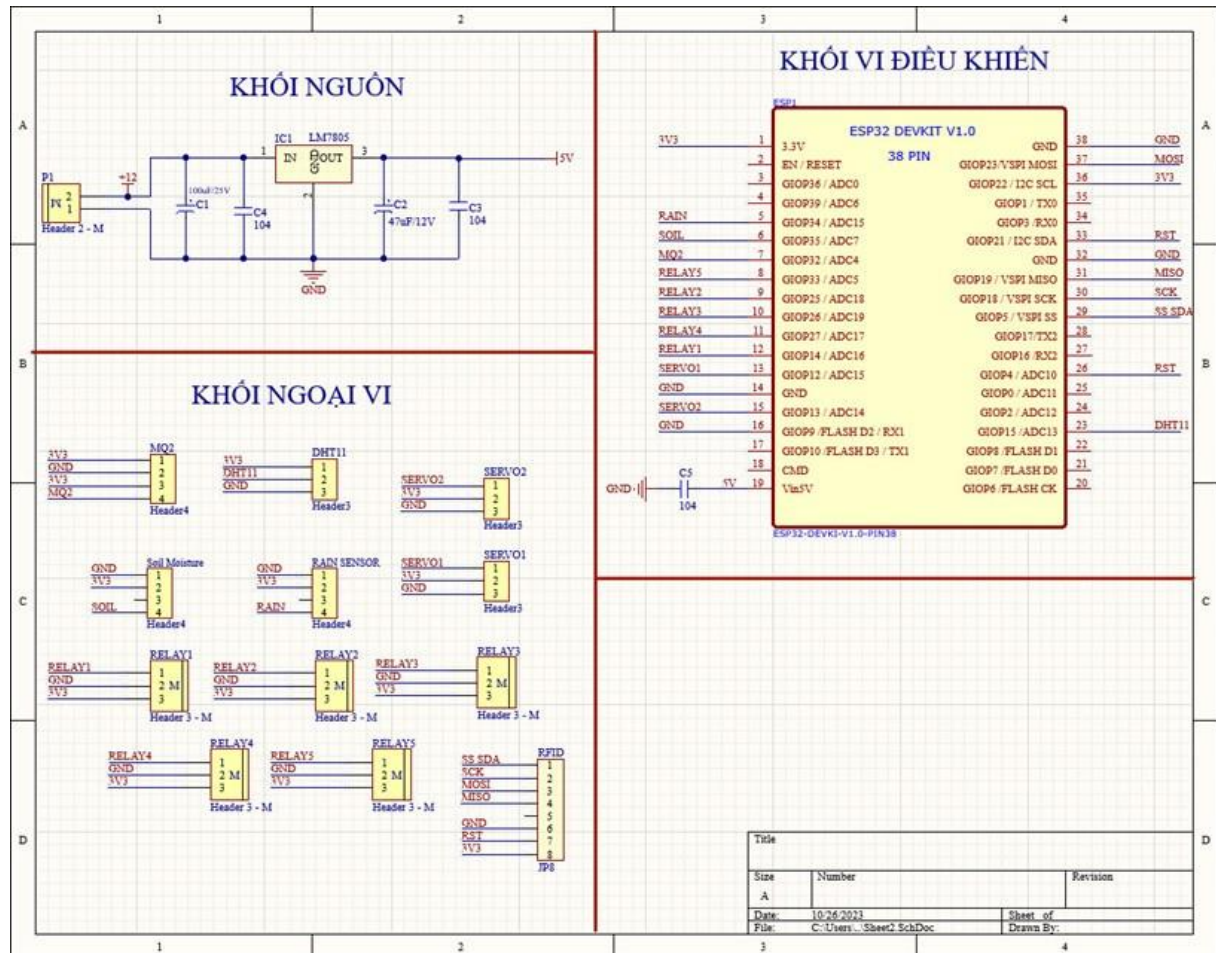
- Cảm biến nhiệt độ, độ ẩm.
- Cảm biến khí ga.
- Cảm biến độ ẩm đất.
- Cảm biến hồng ngoại vật cản.
- Cảm biến RFID.

Khối thiết bị: bao gồm các thiết bị như đèn, quạt, máy bơm.

2.2.1. Khối xử lý trung tâm ESP32

Đây là khối có chức năng thu thập dữ liệu từ các cảm biến để gửi dữ liệu lên webserver và điều khiển các thiết bị được lắp đặt trong ngôi nhà. Được kết nối với khối cảm biến, khối Web Server và khối thiết bị. Nhận dữ liệu từ cảm biến sau đó gửi dữ liệu lên webserver thông qua khối webserver và trả lại kết quả ở khối thiết bị tín hiệu hiển thị trên đèn, quạt, máy bơm, ...

Mạch nguyên lý

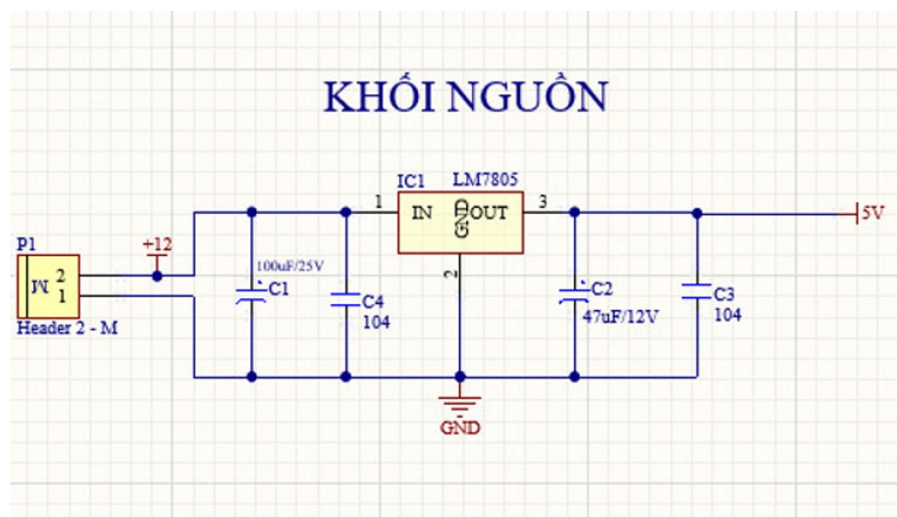


Hình 2. 3: Mạch nguyên lý

Khi có nguồn P1 cấp vào hai chân đầu vào của khối nguồn thì chúng đi qua IC7805 thì điện áp sẽ ổn định ở mức 5V sinh ra một khoảng 4,9V các tụ dùng để san bằng điện áp nhấp nhô (răng cưa) để có dòng ra ổn định sau đó nguồn sẽ được tách nhỏ cấp cho ESP, cảm biến, relay. Sensor đóng vai trò thu đo nhận tín hiệu từ môi trường bên ngoài ra chân digital về ESP32 để thu tín hiệu từ đó so sánh với mức mình yêu cầu khi vượt quá hạn mức đó sẽ kích hoạt một tín hiệu đẩy sang digital khác để kích hoạt động cho relay (âm). Khi đó, relay đóng vai trò như

một công tắc điện để kích mở các thiết bị ngoại vi.

Khối nguồn: là khối cung cấp nguồn điện cho toàn bộ hệ thống. Nó bao gồm một bộ điều chỉnh áp suất và bộ biến áp để cung cấp điện áp ổn định cho hệ thống. Chúng ta sẽ cấp điện áp đầu vào qua J2 (tương ứng theo các chân âm dương) và điện áp 5V ở ngõ ra sẽ được lấy qua chân J1. Tụ C1 và C2 để lọc điện áp đầu vào cấp cho chân Vi của IC 7805, tụ C1 có các dụng cung cấp điện áp tạm thời cho chân Vi khi nguồn đột ngột bị sụt áp, tụ C2 là tụ gốm nên trở kháng lớn, C2 có tác dụng ngăn nguồn đầu vào tăng áp đột ngột làm dạng sóng điện áp đầu vào có hình răng cưa. Tụ C3 và C4 để lọc điện áp cấp cho tải tiêu thụ lấy từ chân Vo của IC 7805, tụ C3 có các 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 (nhiều là các điện áp không mong muốn làm cho dạng sóng điện áp ngõ ra có hình răng cưa).



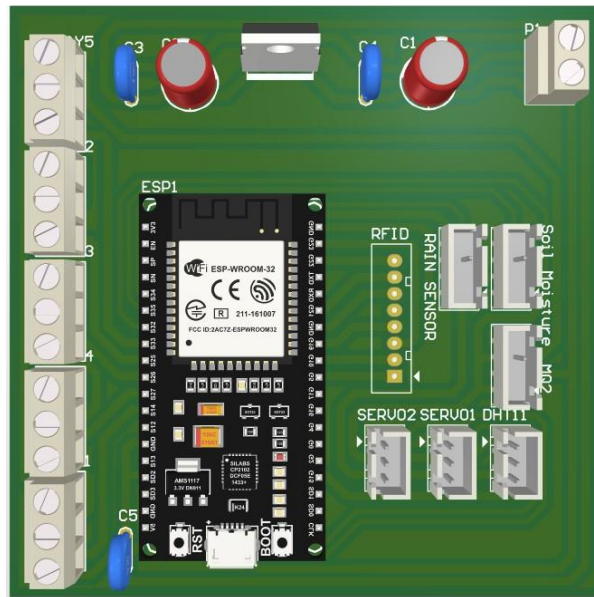
Hình 2. 4: Sơ đồ khối nguồn

Từ hình trên có thể thấy được, mạch sử dụng DC Power để cấp nguồn. DC Power có một đầu nối với chân 1 của LM 7805 và một đầu nối với chân 2 của 7805. LM7805 có tác dụng điều chỉnh điện áp dương đầu ra là 5V. Chân 3 của 7805 nối với 2 tụ hóa 470uF sau đó nối đất.

Thiết kế mạch cứng

Sơ đồ mạch: bước đầu tiên trong thiết kế mạch in, trong đó các thành phần và linh kiện được sắp xếp trong một mạch điện tử. Sơ đồ mạch cần phải đảm bảo rằng các linh kiện được kết nối đúng cách và tương tác với nhau một cách chính xác.

Thiết kế bố trí: Thiết kế bố trí là quá trình sắp xếp các linh kiện và đường dẫn trên một mặt phẳng. Việc sắp xếp các linh kiện phải đảm bảo không xảy ra xung đột hoặc xung đột giữa các linh kiện và đường dẫn.

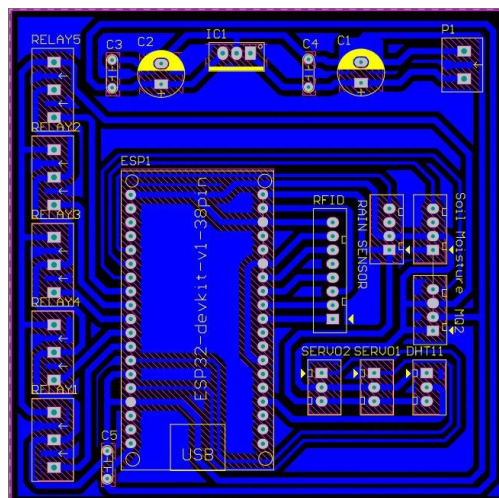


Hình 2. 5: Sơ đồ mạch 3D

Thiết kế đường dẫn: Đường dẫn được thiết kế để kết nối các linh kiện và thành phần khác nhau với nhau trên mạch in. Đường dẫn cần phải đảm bảo rằng các tín hiệu điện tử được truyền tải một cách đáng tin cậy, không bị nhiễu hoặc suy giảm.

Thiết kế lớp: Trong một mạch in có thể có nhiều lớp để kết nối các linh kiện và đường dẫn. Việc thiết kế lớp cần phải đảm bảo rằng mỗi lớp được kết nối với nhau một cách chính xác và không xảy ra xung đột.

Kiểm tra và xác thực: Sau khi thiết kế mạch in hoàn thành, cần phải kiểm tra và xác thực để đảm bảo rằng mạch in hoạt động chính xác.



Hình 2. 6: Sơ đồ mạch in

2.2.2. Khối thiết bị

Khối này có công việc là chấp hành theo những tín hiệu đã được khối xử lý đưa ra: hoạt động quạt, máy bơm, đèn và cửa ra vào.

Trong mạch, relay được sử dụng như một công tắc dùng để bật/tắt thiết bị. Relay là một công tắc (khóa K). Nhưng khác với công tắc ở một chỗ cơ bản là relay được kích hoạt bằng điện thay vì dùng tay người. Chính vì lẽ đó, relay được dùng làm công tắc điện tử. Nguyên lý hoạt động của relay có thể được diễn giải đơn giản như sau: khi dòng điện (thường có công suất nhỏ hơn nguồn điện thứ hai) chạy qua mạch điện thứ nhất sẽ kích hoạt nam châm điện, tạo ra từ trường, tín hiệu. Từ trường này sẽ hút một tiếp điểm (tương tự như khóa K) để kích hoạt mạch điện thứ hai, từ đó cho phép thiết bị kết nối với nguồn điện thứ hai mà không ảnh hưởng đến nguồn điện ban đầu.

2.2.3. Khối cảm biến

Đây là khối sẽ lấy dữ liệu từ môi trường bên ngoài thông qua các con cảm biến để truyền lên khối xử lý trung tâm. Ở các module khác nhau sẽ sử dụng những giao thức khác nhau để có thể xử lý và truyền tín hiệu lên khối vi điều khiển một cách gần đúng nhất với các dữ liệu cảm biến thu được.

Cảm biến nhiệt độ – độ ẩm

Để đo độ ẩm, sử dụng thành phần cảm biến độ ẩm có hai điện cực với chất giữ ẩm giữa chúng. Vì vậy, khi độ ẩm thay đổi, độ dẫn của chất nền thay đổi hoặc điện trở giữa các điện cực này thay đổi. Sự thay đổi điện trở này được đo và xử lý bởi IC khiến cho vi điều khiển luôn sẵn sàng để đọc.

Chân kết nối với vi điều khiển:

- Chân 1 của DHT11 nối với chân 1 của ESP.
- Chân 2 nối với chân 23 của ESP32.
- Chân 3 DHT11 nối với chân 38 ESP32.

Hoạt động: Để có thể giao tiếp với DHT11, đầu tiên cần gửi tín hiệu muốn đo (start) tới DHT11, sau đó DHT11 xác nhận lại. Khi đã giao tiếp được với DHT11, cảm biến sẽ gửi lại 5 byte dữ liệu và nhiệt độ đo được. Tùy theo nhiệt độ đo được mà khối xử lý sẽ đưa ra lệnh cho quạt.

Cảm biến khí ga

Ở chân DO: khi cảm biến tiếp xúc với khí có nồng độ đạt ngưỡng, chân DO sẽ có mức logic cao (5V), khi không phát hiện thấy khí, chân đầu ra digital có điện áp 0V (Led báo sẽ bật khi phát hiện nồng độ SNO₂ và sẽ tắt khi không phát hiện).

Ở chân AC: Giá trị sẽ được đọc bằng vi điều khiển và được chuyển đổi qua ADC (giá trị này sẽ tỷ lệ thuận với nồng độ khí mà cảm biến phát hiện).

Dữ liệu sẽ được chuyển đổi thành ... với “map(value /10, 0, 4095, 0, 35)”.

Chân kết nối với vi điều khiển:

- Chân 1, 3 của MQ2 nối với chân 1 của ESP.
- Chân 2 nối với chân 7 của ESP32.
- Chân 4 nối với chân 14 ESP32.

Cảm biến độ ẩm đất

Trạng thái đầu ra mức thấp (0V), khi đất thiếu nước đầu ra sẽ là mức cao (5V), độ nhạy cao chúng ta có thể điều chỉnh được bằng biến trở. Phần đầu đo được cắm vào đất để phát hiện độ ẩm của đất. Khi module cảm biến độ ẩm phát hiện, khi đó sẽ có sự thay đổi điện áp ngay tại đầu vào của IC LM393. IC này nhận biết có sự thay đổi nó sẽ đưa ra một tín hiệu 0V để báo hiệu, khi độ ẩm của đất đạt ngưỡng thiết lập, đầu ra DO sẽ chuyển trạng thái từ mức thấp lên mức cao (5V).

Đầu ra DO (cho ra tín hiệu ở mức cao 1 hoặc mức thấp 0) được kết nối với vi điều khiển ESP32 ở chân 6 để phát hiện tín hiệu cao hay thấp (led xanh là led báo mức độ ẩm của chân D0).

Giá trị Analog từ chân A0 của module được xử lý bằng bộ chuyển đổi ADC để có giá trị độ ẩm hiển thị lên giao diện webserver.

Điện áp đầu vào: 3.3V.

Khi giá trị của độ ẩm đất đã được truyền từ module lên, khối xử lý thì khối xử lý bắt đầu kiểm tra giá trị. Nếu giá trị lớn hơn 40% thì không kích hoạt máy bơm, giá trị nhỏ hơn 40% thì kích hoạt máy bơm.

Chân kết nối với vi điều khiển:

- Chân 1 của SOIL nối với chân 16 của ESP.
- Chân 2 nối với chân 1 của ESP32.

- Chân 4 nối với chân 6 ESP32.

Cảm biến hồng ngoại vật cản vật cản

Module này bao gồm hai thành phần chính là đèn phát hồng ngoại và mắt thu hồng ngoại. Đèn phát hồng ngoại sẽ phát ra chùm tia hồng ngoại. Mắt thu hồng ngoại sẽ thu lại chùm tia hồng ngoại phản xạ từ vật cản.

Khi không có vật cản ở gần, chùm tia hồng ngoại phát ra từ đèn phát hồng ngoại sẽ đi thẳng đến mắt thu hồng ngoại. Mắt thu hồng ngoại sẽ không nhận thấy có vật cản.

Khi có vật cản ở gần, chùm tia hồng ngoại phát ra từ đèn phát hồng ngoại sẽ bị vật cản phản xạ lại. Mắt thu hồng ngoại sẽ nhận thấy có vật cản và phát tín hiệu báo động.

Khi có tín hiệu được truyền lên khối xử lý, khối xử lý sẽ kiểm tra và đưa ra lệnh tương ứng cho đèn ở khối thiết bị.

Chân kết nối với vi điều khiển:

- Chân 1 của IR1 nối với chân 22 của ESP.
- Chân 1 của IR2 nối với chân 17 của ESP.
- Chân 2 nối với chân 14 của ESP32.
- Chân 3 nối với chân 1 ESP32.

RFID

RC522 giao tiếp với ESP32 bằng giao thức SPI. Hoạt động dựa trên việc sử dụng sóng vô tuyến để truyền tải dữ liệu giữa thẻ RFID và đầu đọc RC522. Khi đưa thẻ vào gần đầu đọc, đầu đọc sẽ đọc và ghi dữ liệu từ thẻ thông qua sóng vô tuyến và sử dụng giao thức SPI để giao tiếp truyền dữ liệu lên khối xử lý để bắt đầu kiểm tra sau đó sẽ đưa ra các lệnh tương ứng.

Mã của thẻ: {131, 174, 143, 21}.

2.3. Lập trình

Tiếp theo là phần lập trình cho hệ thống nhà thông minh. Vì báo cáo không thể đưa được hết code vào nên chúng em xin phép được đính kèm link Github để có thể truy cập vào và xem toàn bộ code.

Link Github: https://github.com/bonn21/Do_an_lien_nganh_2023.git

2.3.1. Wifi

Ở phần này, chúng em thiết lập và quản lý kết nối Wifi, truy xuất địa chỉ IP. Thiết lập wifi sẽ kết nối. Khi hệ thống bắt đầu hoạt động, hệ thống sẽ kết nối với wifi đã được setup trước, sau khi kết nối thành công người dùng sẽ có 1 địa chỉ IP. Truy cập vào địa chỉ IP này sẽ dẫn người dùng đến với giao diện webserver của hệ thống.

Thiết lập wifi: ID và mật khẩu

```
6  const char *WIFI_SSID = "lthbac21";
7  const char *WIFI_PASSWORD = "21112003";
```

Hình 2. 7: ID và Pass được setup trên code

Network properties	
Name:	lthbac21
Password:	21112003
Band:	2.4 GHz

Hình 2. 8: Mobile Hotspot

Hai hàm xử lý thiết lập Wifi và thiết lập kết nối.

Wifi_Setup(): Thực hiện cấu hình Wifi ban đầu và cố gắng kết nối với SSID và mật khẩu được chỉ định.

WIFI_Connect(): Thử kết nối rõ ràng với mạng Wifi, đặt chế độ thành STA, ngắt kết nối khỏi các mạng hiện có và triển khai cơ chế thời gian chờ cho các lần thử kết nối. Khi kết nối thành công, nó cập nhật trạng thái mạng và in địa chỉ IP. Nếu không, nó chỉ ra một lỗi kết nối.

```
5  #ifndef _NETWORK_H_
6  #define _NETWORK_H_
7
8  void Wifi_Setup();
9  void WIFI_Connect();
10 #endif
```

Hình 2. 9: network.h

Đây là file được viết lại sang dạng thư viện để được đưa vào code chính để thiết lập và kết nối wifi cho hệ thống.

2.3.2. RFID

Hệ thống cửa thông minh của mô hình chỉ có 1 mã thẻ duy nhất có thể mở được cửa. Phần này chúng em sẽ thiết lập cho RFID.

user_rfid.cpp

Trong file này chúng em sẽ thiết lập mã thẻ và kiểm tra khi có thẻ được quét vào cảm biến.

```
7  
8   byte nuidPICC[4];  
9   const int ID_DEC[4] = {131, 174, 143, 21}; // mã  
10  const int ID_HEX[4] = {0x83, 0xAE, 0x8F, 0x15};  
11
```

Hình 2. 10: Thiết lập mã cửa

Mã của cửa nhà là : 131, 174, 143, 21.

Hàm RFID_Check() được sử dụng để kiểm tra xem có thẻ RFID được ủy quyền nào hiện diện hay không. Hàm này hoạt động bằng cách thực hiện các bước sau:

- Xóa bộ đệm khóa: Đây là bước cần thiết để đảm bảo rằng bất kỳ khóa nào còn sót lại từ lần đọc thẻ trước đó sẽ bị xóa.
- Kiểm tra xem có thẻ RFID mới nào hiện diện hay không: Nếu không có thẻ mới, hàm sẽ trả về.
- Đọc số sê-ri của thẻ RFID: Số sê-ri này là một mã duy nhất được gán cho mỗi thẻ RFID.
- So sánh số sê-ri thẻ với số sê-ri được ủy quyền: Nếu khớp, hàm sẽ mở cửa. Nếu không, hàm sẽ in thông báo "Faild!".

```

23 void RFID_Check()
24 {
25     for (byte i = 0; i < 6; i++)
26     {
27         key.keyByte[i] = 0xFF;
28     }
29     if (!rfid.PICC_IsNewCardPresent())
30         return;
31     if (!rfid.PICC_ReadCardSerial())
32         return;
33     for (byte i = 0; i < 4; i++)
34     {
35         nuidPICC[i] = rfid.uid.uidByte[i];
36     }
37     if (nuidPICC[0] == ID_HEX[0] && nuidPICC[1] == ID_HEX[1] && nuidPICC[2] == ID_HEX[2] && nuidPICC[3] == ID_HEX[3])
38     {
39         flag_open_door = 1;
40         Serial.println("Open the door.");
41     }
42     else
43     {
44         // count_wrong++;
45         flag_open_door = 0;
46         Serial.println("Failed!");
47     }
48     rfid.PICC_HaltA();
49     rfid.PCD_StopCrypto1();
50 }
51

```

Hình 2. 11: RFID_Check

user_rfid.h (thư viện)

```

3
4 #ifndef _USER_RFID_H_
5 #define _USER_RFID_H_
6
7 #include <stdio.h>
8 #include <stdlib.h>
9 #include <Arduino.h>
10
11 void printDec(byte *buffer, byte bufferSize);
12 void printHex(byte *buffer, byte bufferSize);
13
14 extern int flag_open_door;
15
16 void RFID_Init();
17 void RFID_Check();
18
19
20 #endif

```

Hình 2. 12: user_rfid.h

Hàm printDec() và printHex() được sử dụng để hiển thị giá trị của một mảng byte dưới dạng thập phân hoặc thập lục phân. Chúng được định nghĩa trong tệp tiêu đề để có thể được sử dụng trong các hàm khác trong chương trình.

Biến toàn cục “flag_open_door” được sử dụng để chỉ định liệu cửa có nên được mở hay không. Giá trị của biến này được đặt bởi hàm RFID_Check().

Hàm RFID_Init(): được sử dụng để khởi tạo trình đọc RFID. Nó thiết lập giao

diện SPI và khởi tạo trình đọc.

Hàm RFID_Check(): được sử dụng để kiểm tra xem có thẻ RFID được ủy quyền nào hiện diện hay không. Nếu có, hàm sẽ mở cửa.

2.3.3. Code main (được nạp vào ESP32)

Phần code chính của hệ thống được viết bằng Arduino C++ và sử dụng thư viện ESPAsyncWebserver để xử lý các yêu cầu webserver.

Khởi tạo:

Khai báo các thư viện sử dụng cho ứng dụng, bao gồm: Arduino, Wire (thư viện I2C), ArduinoJson (thư viện Json dùng để trao đổi dữ liệu), ESP32Servo (cho phép điều khiển động cơ bằng ESP32), DHT (thư viện của cảm biến nhiệt độ - độ ẩm), user_rfid (RFID), network(Wifi), ESPAsyncWebserver(thư viện quản lý máy chủ webserver không đồng bộ trên ESP32), index_html (chứa mã HTML cho trang webserver chính của máy chủ webserver) và SPIFFS (cho phép truy cập hệ thống SPIFFS - lưu trữ dữ liệu trên ESP32).

Xác định các chân cho: các relay (từ 1 đến 6) và trạng thái hoạt động của relay (Low - High). Các chân cảm biến khí ga (MQ2), cảm biến độ ẩm đất (SOIL), Cảm biến hồng ngoại vật cản (IR1 và IR2). Định nghĩa cho động cơ (SERVO) đóng (CLOSE) và mở (OPEN). Định nghĩa loại cảm biến nhiệt độ - độ ẩm(DHTTYPE) là (DHT11_DATA), lưu trữ dữ liệu (sensor) và dữ liệu sự kiện (event) của cảm biến.

AsyncWebServer(80);: tạo đối tượng máy chủ webserver không đồng bộ đặt trên cổng 80.

Nó khởi tạo cảm biến DHT11 để đo nhiệt độ và độ ẩm với “_dht_temp” để đọc giá trị và “_dht_humi” để lưu giá trị. “_soil” dùng để lưu giá trị đọc độ ẩm đất.

Sử dụng Timer0 để cấu hình bộ đếm thời gian, đọc cảm biến và xử lý ngắt.

Nó khởi tạo mô-đun RFID để xác thực người dùng.

Khởi tạo máy chủ webserver

Thư viện “index_html.h” là giao diện webserver chính.

Các biến “temperature”, “soil” và “gas” cung cấp các giá trị hiện tại của nhiệt

độ, độ ẩm đất và nồng độ khí tương ứng.

“mode_auto” dùng để đặt hệ thống sang chế độ tự động, nơi các thiết bị được điều khiển tự động dựa trên các giá trị cảm biến.

Cấu trúc “Dev” để lưu trữ trạng thái của các thiết bị khác nhau. Mỗi biến đại diện cho trạng thái của một thiết bị cụ thể (0 là tắt, 1 là bật) như cửa ban công, đèn phòng khách, quạt phòng khách, quạt bếp, đèn phòng ngủ và bơm vườn, cũng được xác định.

Đọc cảm biến và điều khiển thiết bị

Hàm “MQ2_Read()” đọc giá trị tương tự từ cảm biến khí MQ2 và ánh xạ nó sang mức nồng độ khí tương ứng.

Các hàm “readDHTTemperature()” và “readDHTHumidity()” đọc các giá trị nhiệt độ và độ ẩm từ cảm biến DHT11 và trả về chúng dưới dạng chuỗi.

Hàm “Soil_Read()” đọc giá trị tương tự từ cảm biến độ ẩm đất và ánh xạ nó sang phần trăm độ ẩm đất tương ứng.

Hàm Setup: thực hiện các khởi tạo các hàm cho phần cứng và phần mềm, kiểm tra tệp SPIFFS đã gắn đúng chưa, kết nối Wifi đảm bảo hệ thống đã sẵn sàng hoạt động.

Hàm loop: luôn thực thi liên tục trong chương trình, xử lý các tác vụ chính của hệ thống tự động.

Khởi động các cờ trạng thái: Khởi động các cờ liên quan đến việc kiểm tra thẻ RFID, kiểm tra điều khiển IR, đọc cảm biến và mở cửa.

Đọc cảm biến: Kiểm tra nếu cờ “flag_read_sensor” có giá trị chia hết cho 20, nếu có thì thực hiện hàm Read_Sensor() để đọc các giá trị cảm biến (nhiệt độ, độ ẩm đất, nồng độ khí).

Kiểm tra thẻ RFID: Kiểm tra nếu cờ “flag_check_rf” có giá trị 1, nếu có thì thực hiện hàm RFID_Check() để kiểm tra thẻ RFID và thực hiện các hành động tương ứng.

Điều khiển mở cửa: Kiểm tra nếu cờ “flag_open_door” có giá trị 1, nếu có thì đặt cờ “mode_select_door” bằng 1 để chuyển sang chế độ điều khiển mở cửa.

Điều khiển servo động cơ: Nếu số lần đếm cnt chia hết cho 50 thì đặt servo

động cơ về vị trí đóng cửa, nếu không thì đặt servo động cơ về vị trí mở cửa.

Điều khiển tự động: Kiểm tra nếu cờ “mode_auto” có giá trị 1, nếu có thì thực hiện các hành động điều khiển tự động.

Điều khiển quạt bếp: Kiểm tra giá trị cảm biến khí MQ2, nếu vượt quá ngưỡng 20 thì bật quạt bếp trong 50 lần đếm tiếp theo, nếu không thì tắt quạt bếp.

Điều khiển quạt phòng khách: Kiểm tra giá trị nhiệt độ DHT11, nếu vượt quá 29°C thì bật quạt phòng khách trong 50 lần đếm tiếp theo, nếu không thì tắt quạt phòng khách.

Điều khiển bơm vườn: Kiểm tra độ ẩm đất, nếu thấp hơn 40% thì bật bơm vườn trong 50 lần đếm tiếp theo, nếu không thì tắt bơm vườn.

Tắt các thiết bị trong chế độ thủ công: Nếu không ở chế độ tự động, tắt tất cả các thiết bị được điều khiển.

Cập nhật trạng thái thiết bị: Gọi hàm Operate_System() để cập nhật trạng thái của các thiết bị dựa trên các cờ trạng thái và lệnh điều khiển.

Trễ 20ms: Trễ 20ms để đảm bảo thực thi các tác vụ một cách mượt mà và không quá tải hệ thống.

Hàm Interrupt_Init thiết lập Timer0 để tạo ngắt ở tần số 8000 Hz, tương ứng với một ngắt mỗi 1000 lần đếm đồng hồ. Hàm Blink_Led() được gán làm trình xử lý ngắt, sẽ được gọi bất cứ khi nào xảy ra ngắt.

Hàm Gpio_Init chịu trách nhiệm khởi tạo các chân GPIO (General Purpose Input/Output) được sử dụng trong hệ thống.

Hàm Dht11_Init khởi tạo cảm biến nhiệt độ và độ ẩm DHT11 và chuẩn bị nó để đọc các giá trị cảm biến. Cảm biến nhiệt độ và độ ẩm được lưu trữ trong biến “sensor”, cho phép truy cập các giá trị nhiệt độ và độ ẩm bằng cách sử dụng các hàm dht.readTemperature() và dht.readHumidity().

Hàm MQ2_Read đọc giá trị tương tự từ cảm biến khí MQ2 và chuyển đổi nó thành mức nồng độ từ 0 đến 35. Giá trị tương tự trung bình (từ 0 đến 1023) được chuyển đổi thành mức nồng độ (từ 0 đến 35) bằng hàm map() - một hàm chuyển đổi các giá trị giữa các thang đo khác nhau.

Hàm readDHTTemperature đọc giá trị nhiệt độ từ cảm biến DHT11 và trả về

một đại diện chuỗi của nhiệt độ.

Hàm readDHTHumidity đọc giá trị độ ẩm từ cảm biến DHT11 và trả về một đại diện chuỗi của độ ẩm.

Hàm Soil_Read đọc giá trị tương tự từ cảm biến độ ẩm đất và chuyển đổi nó thành phần trăm độ ẩm đất, hiển thị lên % độ ẩm đất.

Hàm Webserver_Control thiết lập các điểm cuối webserver để điều khiển các thiết bị nhà thông minh bằng các yêu cầu HTTP. Nó xử lý các yêu cầu khác nhau cho các hành động khác nhau, chẳng hạn như đọc giá trị cảm biến, kích hoạt chế độ tự động hoặc điều khiển cửa ban công.

Gửi dữ liệu lên webserver

```
370 // đường dẫn khi mở web
371 server.on("/", HTTP_GET, [](AsyncWebServerRequest *request)
372     {
373         // flag_manual = 0; // mặc định quạt chạy ở chế độ auto
374         request->send_P(200, "text/html", index_html, processor); });
375
376 server.on("/temperature", HTTP_GET, [](AsyncWebServerRequest *request)
377     { request->send_P(200, "text/plain", readDHTTemperature().c_str()); });
378
379 // lấy thông tin độ ẩm
380 server.on("/soil", HTTP_GET, [](AsyncWebServerRequest *request)
381     { request->send_P(200, "text/plain", Soil_Read().c_str()); });
382
383 server.on("/gas", HTTP_GET, [](AsyncWebServerRequest *request)
384     { request->send_P(200, "text/plain", MQ2_Read().c_str()); });
385 // Mode auto (cần)
```

Hình 2. 13: Gửi dữ liệu

Phần này thiết lập trình xử lý đường dẫn cho đường dẫn gốc "/". Khi có yêu cầu GET được gửi đến đường dẫn gốc, hàm xử lý sẽ được gọi. Hàm xử lý gửi tệp index_html dưới dạng phản hồi HTML. Đối số processor được sử dụng để xử lý bất kỳ biến nào được sử dụng trong tệp HTML. Tiếp theo, thiết lập trình xử lý đường dẫn cho các đường dẫn "/temperature", "/soil", "/gas". Khi có yêu cầu GET được gửi đến 1 trong 3 đường dẫn hàm xử lý sẽ được gọi.

"/temperature": hàm xử lý đọc nhiệt độ từ cảm biến DHT và gửi nhiệt độ dưới dạng phản hồi văn bản đơn giản.

"/soil", hàm xử lý đọc độ ẩm đất từ cảm biến và gửi độ ẩm đất dưới dạng phản hồi văn bản đơn giản.

"/gas", hàm xử lý đọc nồng độ khí từ cảm biến MQ2 và gửi nồng độ khí dưới dạng phản hồi văn bản đơn giản.

Mode auto

“auto” : đặt cờ “mode_auto” là 1, hệ thống đang ở chế độ tự động. Chế độ tự động có nghĩa là cửa ra vào sẽ được mở khi nhận định đúng mã thẻ RFID. Trường hợp này cũng gửi phản hồi HTML với nội dung của biến index_html được xử lý bởi hàm processor. Hàm processor có thể được sử dụng để thêm nội dung tùy chỉnh vào phản hồi HTML.

“manual/balcony/open_door”: đặt cờ “deviec.balcony_door” là 1, cho biết cửa ra vào nên được mở. Nó cũng đặt “mode_select_door” và “mode_auto” thành 0, đảm bảo rằng hệ thống đang ở chế độ thủ công để điều khiển cửa ra vào. Chế độ thủ công có nghĩa là người dùng có thể mở hoặc đóng cửa sổ ban công theo cách thủ công bằng cách sử dụng các nút trên giao diện webserver.

```
386 server.on("/auto", HTTP_GET, [](AsyncWebServerRequest *request)
387     {
388         mode_auto = 1;
389         request->send_P(200, "text/html", index_html, processor); });
390
391 server.on("/manual/balcony/open_door", HTTP_GET, [](AsyncWebServerRequest *request)
392     {
393         deviec.balcony_door = 1;
394         mode_select_door = 0;
395         mode_auto = 0;
396         request->send_P(200, "text/html", index_html, processor); });
397 server.on("/manual/balcony/cloes_door", HTTP_GET, [](AsyncWebServerRequest *request)
398     {
399         deviec.balcony_door = 0;
400         mode_select_door = 0;
401         mode_auto = 0;
402         request->send_P(200, "text/html", index_html, processor); });
403
```

Hình 2. 14: Chế độ Auto cho cửa

“manual/balcony/cloes_door”: đặt cờ “deviec.balcony_door” là 0, cho biết cửa ra vào nên được đóng. Nó cũng đặt “mode_select_door” và “mode_auto” thành 0, đảm bảo rằng hệ thống đang ở chế độ thủ công để điều khiển cửa ra vào. Trường hợp này hoạt động giống như trường hợp “manual/balcony/open_door”, chỉ khác là nó đóng cửa ra vào thay vì mở nó.

Đó là giải thích ở phần cửa ra vào, đối với các thiết bị khác cũng được thiết lập tương tự như vậy cho từng khu vực trong nhà.


```

403 // Khu vực phòng khách - bếp
404 > server.on("/manual/livingroom/ledon", HTTP_GET, [(AsyncWebServerRequest *request) ...
409   request->send_P(200, "text/html", index_html, processor); });
410 > server.on("/manual/livingroom/ledoff", HTTP_GET, [(AsyncWebServerRequest *request) ...
414   request->send_P(200, "text/html", index_html, processor); });
415 //-----
416 > server.on("/manual/livingroom/fanon", HTTP_GET, [(AsyncWebServerRequest *request) ...
420   request->send_P(200, "text/html", index_html, processor); });
421 > server.on("/manual/livingroom/fanoff", HTTP_GET, [(AsyncWebServerRequest *request) ...
425   request->send_P(200, "text/html", index_html, processor); });
426 //-----
427 > server.on("/manual/kitchen/fanon", HTTP_GET, [(AsyncWebServerRequest *request) ...
431   request->send_P(200, "text/html", index_html, processor); });
432 > server.on("/manual/kitchen/fanoff", HTTP_GET, [(AsyncWebServerRequest *request) ...
436   request->send_P(200, "text/html", index_html, processor); });
437
438 // khu vực phòng ngủ
439 > server.on("/manual/bedroom/ledon", HTTP_GET, [(AsyncWebServerRequest *request) ...
444   request->send_P(200, "text/html", index_html, processor); });
445 > server.on("/manual/bedroom/ledoff", HTTP_GET, [(AsyncWebServerRequest *request) ...
450   request->send_P(200, "text/html", index_html, processor); });
451
452 // khu vực chậu cây
453 > server.on("/manual/garden/pumpon", HTTP_GET, [(AsyncWebServerRequest *request) ...
457   request->send_P(200, "text/html", index_html, processor); });
458 > server.on("/manual/garden/pumpoff", HTTP_GET, [(AsyncWebServerRequest *request) ...
462   request->send_P(200, "text/html", index_html, processor); });
463 server.begin();
464 }
465

```

Hình 2. 15: Webserver (các khu vực khác)

Hàm *Operate_System* có nhiệm vụ điều khiển trạng thái của các thiết bị trong nhà ở chế độ thủ công.

Đầu tiên, kiểm tra trạng thái của cửa ra vào và biến “mode_select_door”. Nếu cửa ra vào đang mở và biến “mode_select_door” bằng 0, thì chức năng này sẽ mở cửa ra vào bằng động cơ servo myservo1. Ngược lại, nếu cửa ra vào đang đóng và biến “mode_select_door” bằng 0, thì chức năng này sẽ đóng cửa ra vào và đặt biến “mode_select_door” thành 1.

Sau đó kiểm tra trạng thái của đèn LED phòng khách, quạt phòng khách và quạt bếp. Nếu bất kỳ thiết bị nào trong số này được bật và biến “mode_auto” bằng 0, thì chức năng này sẽ bật relay tương ứng bằng hàm digitalWrite(). Ngược lại, nếu thiết bị đang tắt và biến “mode_auto” bằng 0, thì chức năng này sẽ tắt relay tương ứng.

Tiếp tục, kiểm tra trạng thái của đèn LED phòng ngủ. Nếu đèn LED phòng ngủ đang bật và biến “mode_auto” bằng 0, thì chức năng này sẽ bật relay tương ứng bằng hàm digitalWrite(). Ngược lại, nếu đèn LED đang tắt và biến “mode_auto” bằng 0, thì chức năng này sẽ tắt relay tương ứng.

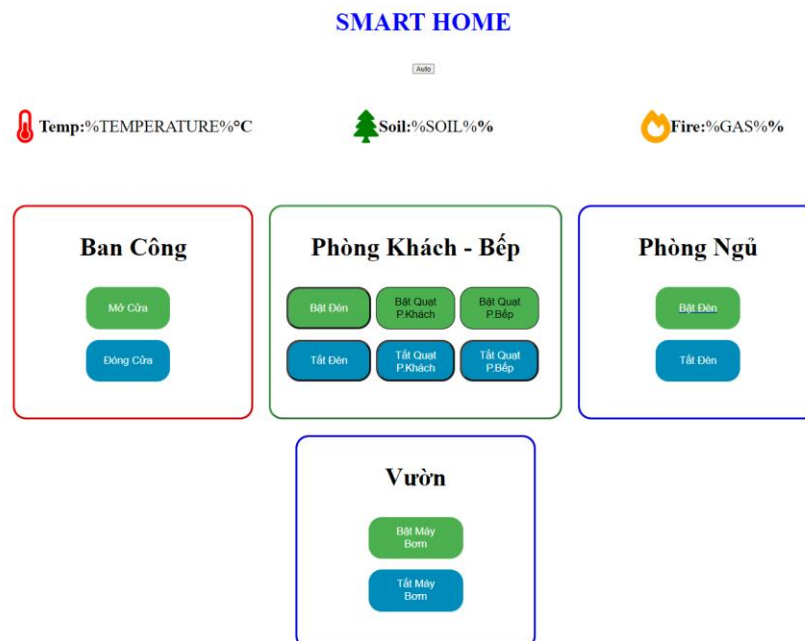
Cuối cùng, chức năng kiểm tra trạng thái của máy bơm ở cây. Nếu máy bơm ở cây được bật và biến “mode_auto” bằng 0 thì hàm này sẽ bật relay tương ứng bằng hàm digitalWrite(). Ngược lại, nếu máy bơm bị tắt và biến “mode_auto” bằng 0 thì chức năng sẽ tắt relay tương ứng.

Vậy là chúng em đã phân tích tổng kết xong, phần code quan trọng nhất của mô hình.

2.3.4. Giao diện webserver

Đây chính là giao diện webserver sau khi dán địa chỉ IP được cấp sau khi hệ thống kết nối với wifi thành công. Giao diện này sử dụng HTML5, CSS và JavaScript để lập trình. Nó hiển thị nhiệt độ, độ ẩm đất và nồng độ khí trên trang webserver. Nó cũng cho phép người dùng điều khiển cửa ra vào, đèn LED phòng khách, quạt phòng khách, quạt bếp, đèn LED phòng ngủ và máy bơm.

Các nút được sử dụng để điều khiển các thiết bị. Khi người dùng nhấp vào một nút, mã JavaScript sẽ gửi yêu cầu đến hệ thống để bật hoặc tắt thiết bị.



Hình 2. 16: Giao diện webserver

Các hiển thị nhiệt độ, độ ẩm đất và nồng độ khí được cập nhật bằng cách truy vấn các cảm biến tương ứng.

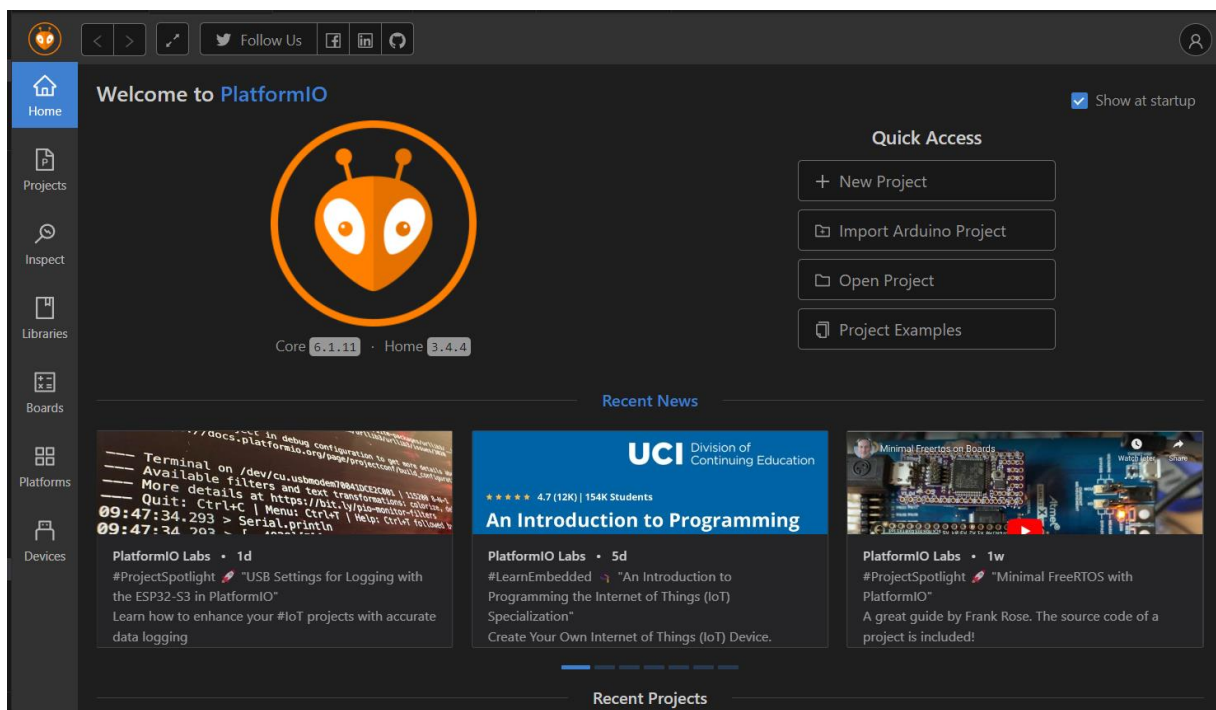
Để có thể truy cập vào trang webserver ngoài địa chỉ IP còn phải đưa giao diện webserver vào phần code chính để có thể nhận dữ liệu hay điều khiển hệ

thống nhà được. Tạo một mảng chuỗi ký tự “index_html[] PROGMEM” và dán toàn bộ file “test.html” vào chuỗi đó. Sau đó lưu thành 1 thư viện và gọi lại ở trong hàm code chính trước khi nạp code chính vào ESP32.

2.3.5. PlatformIO IDE

PlatformIO IDE là một môi trường phát triển tích hợp (IDE) mã nguồn mở cho các dự án phần cứng nhúng. IDE này cung cấp hỗ trợ cho nhiều nền tảng phần cứng nhúng khác nhau, bao gồm Arduino, ESP8266, ESP32, STM32, Raspberry Pi, NXP LPC, và nhiều nền tảng khác.

Với đề tài lần này, chúng em sử dụng PlatformIO để xây dựng lên hệ thống nhà thông minh. Thực chất Arduino IDE hoạt động tốt cho các ứng dụng nhỏ. Tuy nhiên, đối với các dự án nâng cao với hơn 200 dòng code, nhiều file và các tính năng nâng cao khác như tự động hoàn thành và kiểm tra lỗi, VS Code với phần mở rộng PlatformIO IDE là giải pháp thay thế tốt nhất.



Hình 2. 17: PlatformIO IDE

Vì PlatformIO là một phần mở rộng của VS Code nên nó có những thứ tốt hơn Arduino IDE:

- Tự động phát hiện cổng COM của board mạch.
- IntelliSense: Tự động hoàn thành code khi chúng ta đang gõ. VS Code sẽ

cố gắng đoán những gì bạn muốn viết, hiển thị các khả năng khác nhau và cung cấp thông tin chi tiết về các tham số của hàm đang được gọi.

- Tự động phát hiện và gạch chân các lỗi có trong mã trước khi biên dịch.
- Hỗ trợ cho nhiều nền tảng phần cứng nhúng.
- Hỗ trợ cho nhiều ngôn ngữ lập trình, bao gồm C, C++, Python, và Rust.
- Trình quản lý thư viện tích hợp.
- Trình quản lý dự án tích hợp.
- Trình gỡ lỗi tích hợp.
- Trình biên dịch và liên kết tích hợp.

PlatformIO IDE này cung cấp một môi trường phát triển tích hợp toàn diện, giúp bạn dễ dàng phát triển các ứng dụng phần cứng nhúng.

2.4. Kết quả:

Mô hình nhà thông minh sau khi hoàn thành:



Hình 2. 18: Mô hình nhà thông minh

Video hoạt động của mô hình:

<https://drive.google.com/file/d/1DJXe8Xr81OeWJSgbqBDgU0CkCjuqUXBl/view?usp=sharing>

Kết nối wifi

Properties

Network properties

Edit

Name:

lthbac21

Password:

21112003

Band:

2.4 GHz

Devices connected: 1 of 8

Device name	IP address	Physical address (MAC)
esp32-AF9AB0	192.168.137.151	48:e7:29:af:9a:b0

Hình 2. 19: WIFI

Tuy cập webserver

SMART HOME

Auto

Temp:30°C

Soil:75%

Fire:18%

Ban Công

Mở Cửa

Đóng Cửa

Phòng Khách - Bếp

Bật Đèn

Tắt Đèn

Bật Quạt P.Khách

Tắt Quạt P.Khách

Bật Quạt P.Bếp

Tắt Quạt P.Bếp

Phòng Ngủ

Bật Đèn

Tắt Đèn

Vườn

Bật Máy Bơm

Tắt Máy Bơm

Hình 2. 20: Webserver

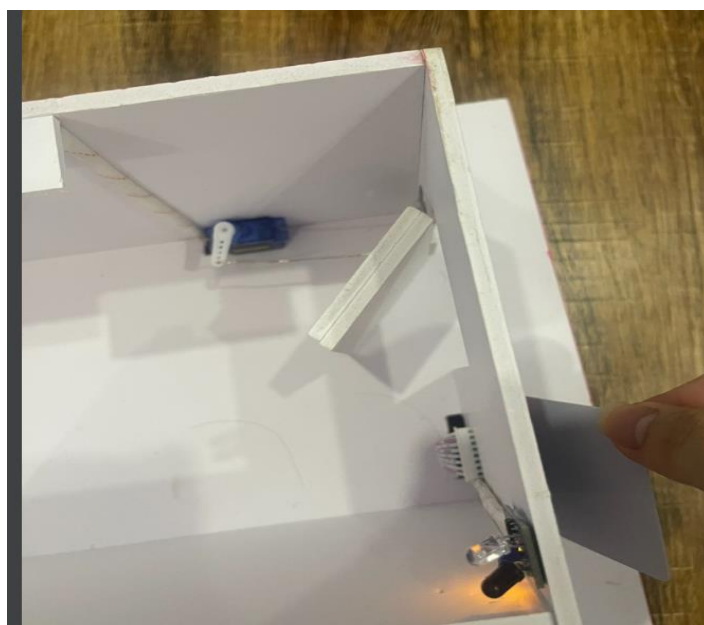
Nhà hoạt động



Hình 2. 21: Nhà hoạt động

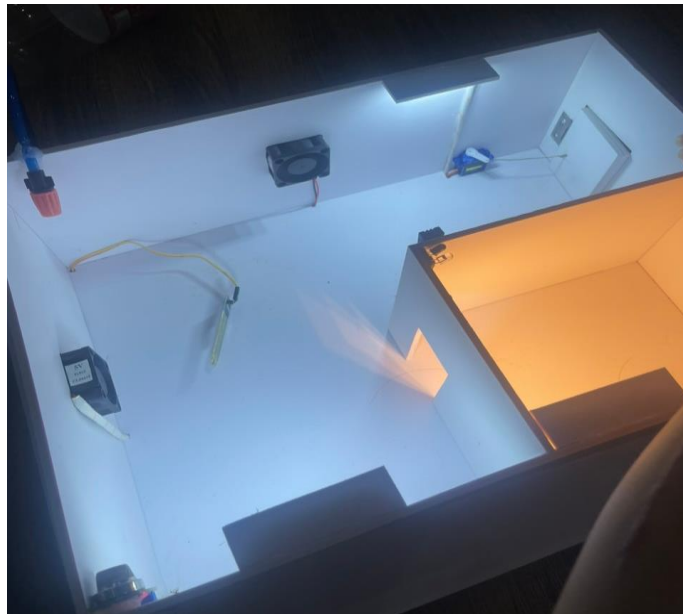
Auto

RFID

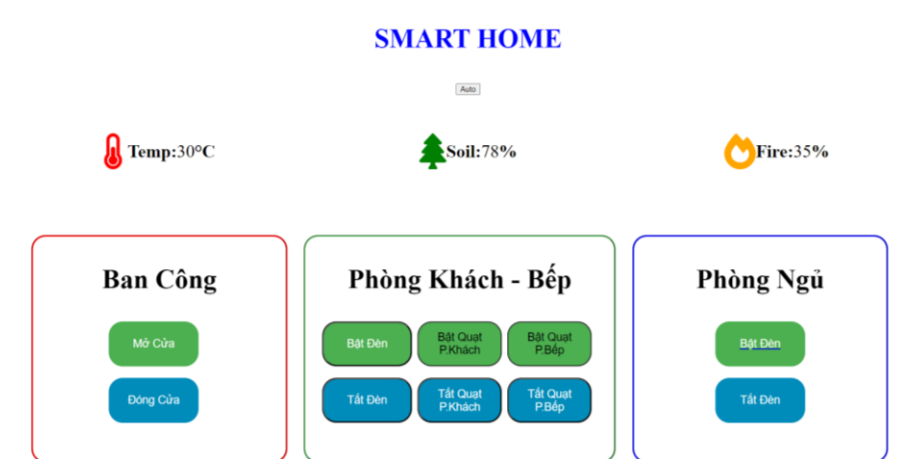


Hình 2. 22: RFID

Đèn bật khi có vật cản



Hình 2. 23: Đèn (khách và ngủ)



Hình 2. 24: Temp – Gas

Quạt bếp quay khi nồng độ khí gas là 35%



Hình 2. 25: Quạt phòng bếp

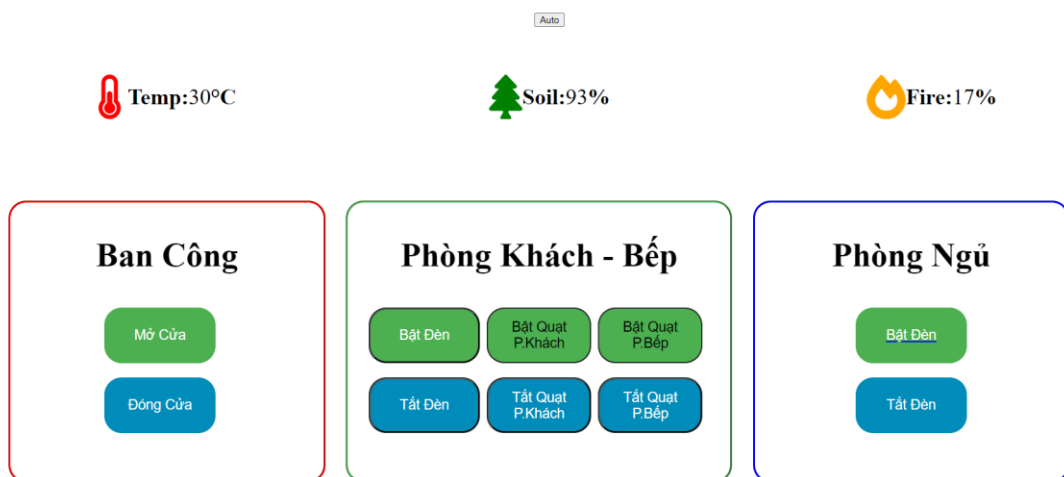
Quạt phòng khách chạy khi nhiệt độ là 30°C



Hình 2. 26: Quạt phòng khách

Máy bơm không hoạt động với độ ẩm là 93%

SMART HOME

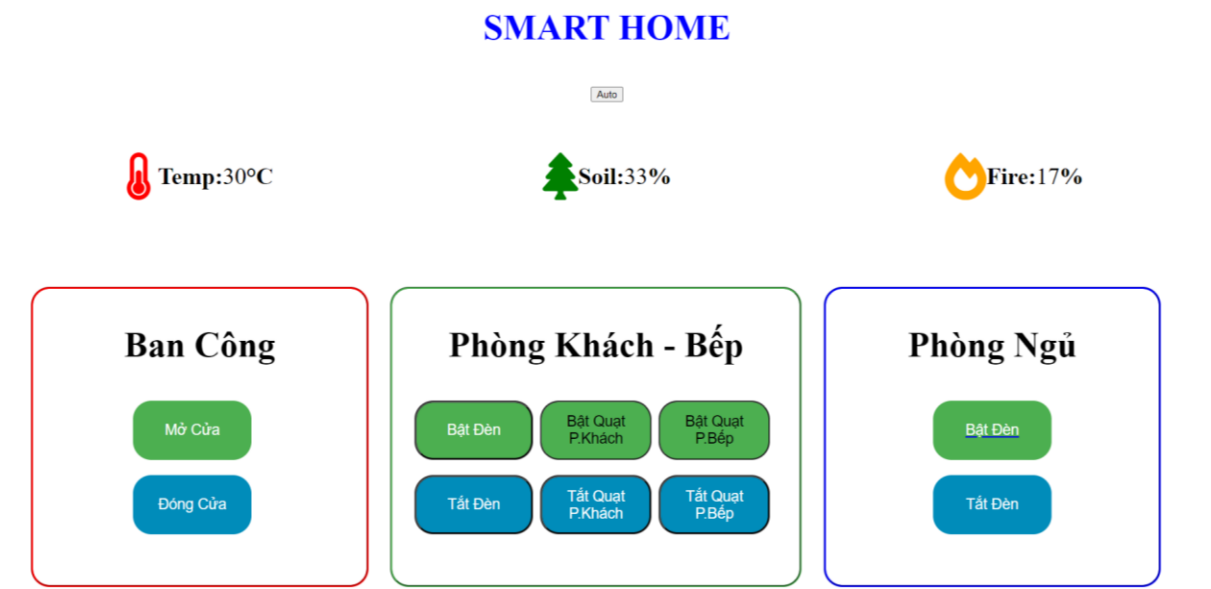


Hình 2. 27: Soil (93)



Hình 2. 28: Bơm đã ngắt

Máy bơm đang hoạt động với độ ẩm là 33%



Hình 2. 29: Soil (33)



Hình 2. 30: Bơm hoạt động

Điều khiển qua webserver

Video tổng quan xung quanh nhà khi ở chế độ này và được bật hết các tính năng:

<https://drive.google.com/file/d/1NSlHY7iaWGF6cyPuf9yBB7GWJRCkkDpi/view?usp=sharing>

Bật hết các chức năng trong nhà (trừ RFID)



Hình 2. 31: Munaul

2.5. Tổng kết chương

Kết quả là hệ thống đã hoạt động ổn định, các dữ liệu lên thành công. Phần này chúng em đã đi qua mai mục chính là: thiết kế hệ thống và lập trình. Ở phần thiết kế hệ thống đã nêu ra được phần mềm và phần cứng được thiết kế như thế nào và từng bước thực hiện ra sao. Về phần lập trình, PlatformIO là một IDE hỗ trợ rất tốt trong việc lập trình lên hệ thống này. Chúng em có thể thiết lập wifi riêng và liên kết với webserver thành công để dữ liệu từ các module cảm biến có thể hiển thị trên đó. Phần cuối cùng của báo cáo này sẽ là phần tổng kết.

TỔNG KẾT

3.1. Nhận xét

3.1.1. Ưu điểm

Đã thiết kế và xây dựng được mô hình nhà thông minh theo mục tiêu ban đầu:

- Tính ổn định: Mạch chỉ cần được kết nối với wifi ổn định có thể hoạt động trong thời gian dài.
- Giá thành hợp lý: Mô hình có giá thành không quá cao, dễ dàng sử dụng và bảo trì.
- Đây là một đề án cần kiến thức liên ngành và xây dựng mô hình thực, chúng em đã củng cố và bổ sung thêm được nhiều kỹ năng cũng như kiến thức thực sự hữu ích.

3.1.2. Giới hạn

- Khó khăn ban đầu khi lựa chọn các chức năng cho mô hình nhà thông minh do hạn chế về số lượng chân I/O của Vi điều khiển cũng như phù hợp với khả năng tài chính.
- Dành nhiều thời gian cho việc tự học và tìm hiểu các kiến thức mới về ESP32, kết nối hệ thống với app/webserver, Visual Studio Code...

3.2. Kết luận

Đề án "Thiết kế và xây dựng mô hình nhà thông minh" sử dụng PlatformIO và vi điều khiển ESP32 đã thành công trong việc xây dựng một mô hình nhà thông minh cơ bản, có khả năng điều khiển các thiết bị điện trong nhà từ xa thông qua webserver và kết nối với wifi. Mô hình này đã thể hiện được những ưu điểm của IoT trong việc ứng dụng vào cuộc sống, giúp cho việc quản lý, kiểm soát trong nhà trở nên dễ dàng và tiện lợi hơn. Đề án này cũng đã góp phần nâng cao kiến thức và kỹ năng của chúng em, giúp chúng em hiểu rõ hơn về IoT và cách ứng dụng nó vào thực tế. Ngoài ra, đây cũng có thể trở thành một đề tài hấp dẫn cho các bạn sinh viên hoặc những người đam mê công nghệ, có thể nghiên cứu và phát triển để tạo ra những sản phẩm ứng dụng thực tế và giúp cho cuộc sống của mọi người trở nên thông minh hơn.

3.3. Phương hướng phát triển

Một số hướng phát triển của mô hình:

- Thêm điều khiển ở trên app, giám sát an ninh.
- Thêm tính năng thay đổi nhiệt độ nhiệt độ trong phòng.

TÀI LIỆU THAM KHẢO

Tài liệu về ESP32:

Có tất cả về ESP32: <https://github.com/espressif>

Thông số chính xác của ESP: <https://products.espressif.com/#/>

Tìm hiểu về ESP32 : <https://deviot.vn/tutorials/esp32.66047996/tong-quan-ve-esp32.18482631> (Tiếng việt)

Lập trình RFID:

https://github.com/AdaSalvadorAvalos/practica6_adasalvadoravalos.git

Lập trình: <https://esp32.vn/idf/index.html> (Tiếng việt)

Datasheet của linh kiện: <https://datasheetspdf.com>

Thông tin về các module: <https://components101.com>

Tìm hiểu về nhà thông minh: <https://www.researchgate.net/profile/Jian-Mao-3> ,

<https://www.infocomm-journal.com/jcin/EN/abstract/abstract168837.shtml>

ⁱ<https://cafeland.vn/kien-thuc/nha-thong-minh-la-gi-kien-thuc-tong-quan-ve-smart-home-107299.html>

ⁱⁱ Các số liệu được lấy từ bài viết "Xu hướng phát triển nhà thông minh" trên website của Hiệp hội Thương mại điện tử Việt Nam (VECOM)

ⁱⁱⁱ <https://dientutuonglai.com/tim-hieu-vi-dieu-khien.html>

^{iv} <https://dientutuonglai.com/tim-hieu-vi-dieu-khien.html>

^v <https://dvn.com.vn/lap-trinh-vi-dieu-khien-1403978719/>

^{vi} <https://khuenguyencreator.com/tong-quan-ve-vi-dieu-khien/>

^{vii} <https://github.com/espressif/arduino-esp32/blob/master/docs/source/api/wifi.rst>

^{viii} https://docs.ai-thinker.com/_media/esp32/docs/nodemcu-32s_product_specification.pdf

^{ix} <https://vancongnghiepvnn.net/rfid-la-gi.html>

^x <https://components101.com/articles/introduction-to-gas-sensors-types-working-and-applications>