

TRƯỜNG ĐẠI HỌC ĐẠI NAM
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP
XÂY DỰNG ỨNG DỤNG TỰ ĐỘNG CẢNH BÁO BẤT THƯỜNG
CỦA MÔ HÌNH NHÀ THÔNG MINH

Sinh viên thực hiện: Trần Hoàng Nam

Giảng viên hướng dẫn: TS. Hoàng Xuân Hiến

Lớp: CNTT14 – 04

Khóa: 2020 – 2024

Hệ: Đại học chính quy

HÀ NỘI - NĂM 2024

TRƯỜNG ĐẠI HỌC ĐẠI NAM
KHOA CÔNG NGHỆ THÔNG TIN



HỌ VÀ TÊN: TRẦN HOÀNG NAM

**XÂY DỰNG ỨNG DỤNG TỰ ĐỘNG CẢNH BÁO BẤT
THƯỜNG CỦA MÔ HÌNH NHÀ THÔNG MINH**

CHUYÊN NGÀNH : CÔNG NGHỆ THÔNG TIN

MÃ SỐ: 74.80.201

GIẢNG VIÊN HƯỚNG DẪN
TS. HOÀNG XUÂN HIỂN

HÀ NỘI – NĂM 2024

LỜI CAM ĐOAN

Em xin cam đoan bài khóa luận này là công sức nghiên cứu và tìm hiểu của em dưới sự hướng dẫn của thầy TS.Hoàng Xuân Hiên. Kết quả nghiên cứu của bài luận này đảm bảo tính trung thực khách quan nhất.

Em cam kết hoàn toàn chịu trách nhiệm về tính chính xác và đáng tin cậy của kết quả nghiên cứu và tài liệu trong khóa luận, các nguồn tài liệu này đã được xác định và trích dẫn một cách rõ ràng và tuân thủ các quy tắc và quy định về bản quyền và tài liệu tham khảo.

Em xin hoàn toàn chấp nhận mọi hình phạt, hậu quả trước nhà trường nếu phát hiện bất kỳ sao chép hoặc vi phạm bản quyền nào trong quá trình thực hiện bài khóa luận này!

Hà Nội, ngày 24 tháng 04 năm 2024

Sinh viên thực hiện

Trần Hoàng Nam

LỜI CẢM ƠN

Trước hết em xin phép được bày tỏ lòng biết ơn đến các thầy cô trong **khoa Công nghệ thông tin** của trường **Đại Học Đại Nam**, bạn bè, tổ chức đã tạo điều kiện hỗ trợ, giúp đỡ em trong suốt quá trình học tập và nghiên cứu. Với vốn kiến thức được tiếp thu trong quá trình học là nền tảng, hành trang quý báu để em bước vào đời một cách vững chắc và tự tin.

Với lòng biết ơn sâu sắc nhất, em xin gửi đến thầy **Hoàng Xuân Hiên** người đã tận tình hướng dẫn, giúp đỡ hướng dẫn em hoàn thành bài báo cáo trong thời gian qua.

Do thời gian có hạn cũng như kiến thức của em còn nhiều hạn chế vì vậy em không tránh khỏi nhiều thiếu sót. Em rất mong được nhận sự góp ý của quý thầy cô trong khoa để em có thêm kinh nghiệm và hoàn thiện tốt hơn.

Em xin chân thành cảm ơn!

LỜI NÓI ĐẦU

Trong thời đại công nghệ hiện đại, ngành nhà thông minh đang phát triển mạnh mẽ, mang lại những tiện ích và an ninh tuyệt vời cho người dùng. Tuy nhiên, việc quản lý và theo dõi hệ thống nhà thông minh của em với hàng trăm thiết bị và cảm biến có thể trở nên phức tạp và đòi hỏi sự chú ý liên tục.

Với mục tiêu giải quyết vấn đề này, em đã tập trung vào việc xây dựng một ứng dụng tự động cảnh báo bất thường cho mô hình nhà thông minh của em. Mục tiêu chính của đề tài là tạo ra một hệ thống thông minh, có khả năng phát hiện và báo cáo các sự cố, hành vi bất thường hoặc nguy hiểm trong môi trường nhà thông minh của em.

Ứng dụng tự động cảnh báo bất thường này sẽ sử dụng các thuật toán máy học và trí tuệ nhân tạo để phân tích dữ liệu từ các thiết bị và cảm biến trong nhà thông minh của em. Hệ thống sẽ học và hiểu các mô hình thông thường của hoạt động trong nhà, từ đó xác định được những sự khác biệt không mong muốn hoặc nguy hiểm.

Khi phát hiện bất thường, hệ thống sẽ tự động gửi cảnh báo đến em thông qua ứng dụng di động. Điều này giúp em có thể phản ứng kịp thời và đưa ra các biện pháp cần thiết để giải quyết tình huống.

Hy vọng rằng ứng dụng tự động cảnh báo bất thường này sẽ nâng cao tính an toàn và tiện ích của nhà thông minh, mang lại sự yên tâm và thuận tiện cho em và các thành viên trong gia đình.

DANH MỤC HÌNH ẢNH

Hình 1. 1: Cấu trúc hệ thống IoT	5
Hình 1. 2: Ứng dụng của IoT	7
Hình 1. 3: Smart home	8
Hình 1. 4: Thiết bị của Google.....	10
Hình 3. 1: Arduino Uno R3	19
Hình 3. 2: CPU ESP32	21
Hình 3. 3: Sơ đồ chân Esp32.....	23
Hình 3. 4: DHT22.....	25
Hình 3. 5: sơ đồ chân của DHT22.....	26
Hình 3. 6: sơ đồ kết nối của DHT22	27
Hình 3. 7: Cách nối dây FHT22	28
Hình 3. 8: Cảm biến ánh sáng	32
Hình 3. 9: Sơ đồ chân của cảm biến ánh sáng	33
Hình 3. 10: Cảm biến khí gas.....	35
Hình 3. 11: Cách nối dây của cảm biến khí gas.....	37
Hình 3. 12: Sơ đồ chân của cảm biến chuyển động.....	39
Hình 3. 13: Cảm biến chuyển động.....	40
Hình 3. 14: Cách nối dây của cảm biến chuyển động.....	42
Hình 3. 15: Cảm biến mực nước	46
Hình 3. 16: Cách nối dây của cảm biến mực nước	47
Hình 3. 17: đèn led.....	52
Hình 3. 18: Còi.....	53
Hình 3. 19:Màn hình Lcd	54
Hình 3. 20: I2C.....	55
Hình 3. 21: Quạt.....	57

Hình 4. 1: Mô hình nhà thông minh.....	59
Hình 4. 2: Sơ đồ kết nối blynk với arduino.....	63
Hình 4. 3: đăng nhập đăng ký	64
Hình 4. 4: Tạo template.....	65
Hình 4. 5: Tạo template.....	65
Hình 4. 6: Chọn template vừa tạo	66
Hình 4. 7: Chọn new datastreams	66
Hình 4. 8: Tạo pin ảo.....	67
Hình 4. 9: Điền thông tin vô trường pin ảo.....	67
Hình 4. 10: Nhập các thông tin vô trường của virtual pin	68
Hình 4. 11: Tạo các widget	68
Hình 4. 12: Setting các widget	69
Hình 4. 13: Chọn pin ảo vừa tạo phù hợp.....	69
Hình 4. 14: Chọn các thông tin theo hướng dẫn	70
Hình 4. 15: Setting với widget đèn led	70
Hình 4. 16: Setting label và chọn pin ảo phù hợp.....	71
Hình 4. 17: Tạo thiết bị mới.....	71
Hình 4. 18: Copy các giá trị define	72
Hình 4. 19: Giao diện app blynk	72
Hình 4. 20: Chọn nút setting	73
Hình 4. 21: Chọn value display.....	73
Hình 4. 22: Ấn ô vừa tạo.....	74
Hình 4. 23: Giao diện sau khi ấn vào ô vừa tạo	74
Hình 4. 24: Chọn pin ảo phù hợp.....	75
Hình 4. 25: Giao diện sau khi chọn pin ảo.....	75
Hình 4. 26: Giao diện ở design	76

Hình 4. 27: Chọn pin ảo phù hợp.....	76
Hình 4. 28: Đổi chế độ ẩn	77
Hình 4. 29: Cài label cho switch	77
Hình 4. 30: Giao diện khi cài xong	78

MỤC LỤC BẢNG

bảng 3.1: các chân ánh xạ cho SPI.....	25
bảng 3.2: giao tiếp giữa I2C LCD với Arduino	56

MỤC LỤC

LỜI CAM ĐOAN	3
LỜI CẢM ƠN	4
LỜI NÓI ĐẦU	5
MỞ ĐẦU	1
1. Tính cấp thiết của đề tài	1
2. Lý do chọn đề tài	1
3. Mục đích nghiên cứu	2
4. Phạm vi nghiên cứu	2
5. Phương pháp nghiên cứu	2
CHƯƠNG 1: GIỚI THIỆU VỀ IOT VÀ NHÀ THÔNG MINH.....	4
1.1. Khái niệm về IoT	4
1.2. Cấu trúc hệ thống IoT	4
1.3. Ưu và nhược điểm của IoT	6
1.4. Ứng dụng của IoT	6
1.5. Nhà thông minh là gì?	8
1.6. Các mô hình nhà thông minh	9
CHƯƠNG 2: CẢNH BÁO BẤT THƯỜNG TRONG NHÀ THÔNG MINH	11
2.1. Cảnh báo về an ninh	11
2.1.1. Cảnh báo ngoài nhà	11
2.1.2. Cảnh báo trong nhà	11
2.2. Cảnh báo trong môi trường nhà	12
2.2.1. Cảnh báo cháy.....	12
2.2.2. Cảnh báo khí gas.....	13

2.2.3.	<i>Cảnh báo chất lượng không khí</i>	14
2.2.4.	<i>Cảnh báo rò rỉ nước</i>	15
2.3.	Cảnh báo hoạt động của thiết bị	16
2.3.1.	<i>Cảnh báo thiết bị đang bật hoặc đang tắt</i>	16

CHƯƠNG 3: CÁC THIẾT BỊ ARDUINO DÙNG TRONG MÔ HÌNH NHÀ THÔNG MINH

3.1.	Thiết bị vi điều khiển: vi điều khiển (Arduino Uno R3)	18
	Vi điều khiển:	18
3.2.	Thiết bị kết nối: kết nối internet bằng wifi với esp32	20
3.3.	Thiết bị cảm biến	25
3.3.1.	<i>Cảm biến nhiệt độ, độ ẩm</i>	25
3.3.2.	<i>Cảm biến ánh sáng</i>	30
3.4.	Các thiết bị cảnh báo bất thường	35
3.4.1	<i>Thiết bị cảnh báo khí gas</i>	35
3.4.2	<i>Thiết bị cảnh báo có chuyển động (chống trộm)</i>	39
3.4.3	<i>Thiết bị cảnh báo về mực nước rò rỉ</i>	45
3.5.	Thiết bị cảnh báo	51
3.5.1.	<i>Đèn Led</i>	51
3.5.2.	<i>Còi chip 5v</i>	52
3.6.	Thiết bị điện	54
3.6.1.	<i>Lcd I2c</i>	54
3.6.2.	<i>Quạt</i>	56

CHƯƠNG 4: THIẾT KẾ MÔ HÌNH,VẬN HÀNH, ỨNG DỤNG VÀ PHÁT TRIỂN

4.1.	Thiết kế mô hình	59
-------------	-------------------------------	----

4.2.	Ứng dụng vào thực tiễn	61
4.3.	Ý tưởng phát triển.....	61
4.4.	Triển khai xây dựng ứng dụng điều khiển mô hình nhà thông minh	62
4.4.1.	Ứng dụng <i>blynk</i>	62
4.4.2.	Cách sử dụng <i>blynk</i>	64
KẾT LUẬN		79
TÀI LIỆU THAM KHẢO		81

MỞ ĐẦU

1. Tính cấp thiết của đề tài

Hiện nay với sự phát triển không ngừng nghỉ, vấn đề hiện đại hóa tất cả các thiết bị, ngành nhà thông minh đang phát triển mạnh mẽ và trở thành một phần không thể thiếu trong cuộc sống hàng ngày của em và nhiều người khác.

Em nhận thấy rằng việc quản lý và giám sát một hệ thống nhà thông minh với nhiều thiết bị và cảm biến có thể trở nên phức tạp và đòi hỏi sự chú ý liên tục. Vì vậy, việc xây dựng một ứng dụng tự động cảnh báo bất thường là cần thiết để giúp em và các người dùng khác giải quyết vấn đề này.

Mục tiêu chính của đề tài là tạo ra một hệ thống thông minh và an toàn, giúp em phát hiện và báo cáo các sự cố, hành vi bất thường hoặc nguy hiểm trong môi trường nhà thông minh của em. Em sẽ sử dụng các thuật toán máy học và trí tuệ nhân tạo để phân tích dữ liệu từ các thiết bị và cảm biến trong nhà.

Tính cấp thiết của đề tài này nằm ở khả năng cung cấp an ninh và tiện ích cho ngôi nhà thông minh của em. Với ứng dụng tự động cảnh báo bất thường, em có thể yên tâm hơn về mức độ an toàn và đáng tin cậy của hệ thống. Đồng thời, đề tài này cũng đóng góp vào sự phát triển của ngành công nghệ thông minh và mang lại lợi ích rõ rệt cho em và các thành viên khác trong gia đình.

2. Lý do chọn đề tài

Đề tài xây dựng ứng dụng tự động cảnh báo bất thường của mô hình nhà thông minh là một đề tài có nhiều tiềm năng phát triển trong tương lai, nhu cầu của thị trường rộng mở, có tính ứng dụng cao trong cuộc sống của mỗi chúng ta. Em nhận thấy rằng ngành nhà thông minh đang phát triển mạnh mẽ và trở thành một phần không thể thiếu trong cuộc sống hàng ngày. Vì vậy, em cảm thấy hứng thú và muốn đóng góp vào lĩnh vực này.

Ngoài ra, đề tài này mang tính ứng dụng cao và có tiềm năng thực tiễn rất lớn. Em nhận thấy rằng việc xây dựng một ứng dụng tự động cảnh báo bất thường không chỉ mang lại an ninh và tiện ích cho ngôi nhà thông minh, mà còn đóng góp vào sự phát triển của ngành công nghệ thông minh nói chung. Điều này thúc đẩy em tiếp tục nỗ lực và hướng tới mục tiêu tạo ra một môi trường sống thông minh, an toàn hơn cho mọi người.

3. Mục đích nghiên cứu

Mục đích nghiên cứu của đề tài này là tạo ra một hệ thống thông minh và an toàn, giúp em phát hiện và báo cáo các sự cố, hành vi bất thường hoặc nguy hiểm trong môi trường nhà thông minh của em một cách tự động và kịp thời, đồng thời em cũng mong muốn rằng thành quả của nghiên cứu này có thể được ứng dụng và đóng góp trong các lĩnh vực khác, góp phần tạo nên một tương lai thông minh an toàn và bền vững.

4. Phạm vi nghiên cứu

Trong đề tài này, em sẽ tập trung vào việc điều tra và phát triển các phương pháp để phát hiện và cảnh báo các sự cố, hành vi bất thường hoặc nguy hiểm trong môi trường nhà thông minh. Cụ thể hơn là tìm hiểu về các công nghệ trong lĩnh vực nhà thông minh, xử lý tín hiệu để xây dựng một hệ thống thông minh an toàn, sử dụng các thiết bị và cảm biến để nhận diện các tình huống bất ngờ hoặc nguy hiểm.

5. Phương pháp nghiên cứu

Tiền nghiên cứu: Em sẽ tìm hiểu kỹ về các công nghệ và tiến bộ trong lĩnh vực nhà thông minh, như học máy, xử lý tín hiệu. Em sẽ nghiên cứu các nghiên cứu và công trình liên quan để hiểu rõ hơn về các phương pháp và kỹ thuật đã được áp dụng trước đó.

Thu thập dữ liệu: Em sẽ tiến hành thu thập dữ liệu từ các thiết bị và cảm biến trong môi trường nhà thông minh. Những dữ liệu này sẽ bao gồm thông tin về nhiệt độ, ánh sáng, chuyển động, và các thông số khác liên quan đến hoạt động của nhà thông minh.

Xử lý và phân tích dữ liệu: Em sẽ áp dụng các thuật toán và kỹ thuật phân tích dữ liệu để xác định các mẫu dữ liệu bất thường. Em sẽ sử dụng các phương pháp như học máy có giám sát và không giám sát để phân loại và nhận diện các hành vi không bình thường trong môi trường nhà thông minh.

Xây dựng ứng dụng: Sau khi đã phân tích và nhận diện được các sự cố và hành vi bất thường, em sẽ xây dựng một ứng dụng tự động cảnh báo. Ứng dụng này sẽ có khả năng gửi thông báo hoặc cảnh báo đến người dùng thông qua các kênh liên lạc như điện thoại di động.

Đánh giá và thử nghiệm: Em sẽ thực hiện các thử nghiệm và đánh giá hiệu suất của hệ thống bằng cách sử dụng dữ liệu thực tế và các kịch bản mô phỏng. Em sẽ đo lường độ chính xác, độ nhạy và độ phản hồi của hệ thống để đảm bảo tính hiệu quả và đáng tin cậy của ứng dụng.

Tổng quan, em sẽ tiến hành nghiên cứu bằng cách kết hợp việc tìm hiểu, thu thập dữ liệu, xử lý và phân tích dữ liệu, xây dựng ứng dụng, và đánh giá hiệu suất để đạt được mục tiêu của đề tài.

CHƯƠNG 1: GIỚI THIỆU VỀ IOT VÀ NHÀ THÔNG MINH

1.1. Khái niệm về IoT

Thuật ngữ IoT hay Internet vạn vật đề cập đến mạng lưới tập hợp các thiết bị thông minh và công nghệ tạo điều kiện thuận lợi cho hoạt động giao tiếp giữa thiết bị và đám mây cũng như giữa các thiết bị với nhau. Nhờ sự ra đời của chip máy tính giá rẻ và công nghệ viễn thông băng thông cao, ngày nay, chúng ta có hàng tỷ thiết bị được kết nối với internet. Điều này nghĩa là các thiết bị hàng ngày như bàn chải đánh răng, máy hút bụi, ô tô và máy móc có thể sử dụng cảm biến để thu thập dữ liệu và phản hồi lại người dùng một cách thông minh.

Internet vạn vật tích hợp “vạn vật” với Internet mỗi ngày. Các kỹ sư máy tính đã và đang thêm các cảm biến và bộ xử lý vào các vật dụng hàng ngày kể từ những năm 90. Tuy nhiên, tiến độ ban đầu rất chậm vì các con chip còn to và cồng kềnh. Loại chip máy tính công suất thấp gọi là thẻ tag RFID, lần đầu tiên được sử dụng để theo dõi các thiết bị đắt đỏ. Khi kích cỡ của các thiết bị điện toán dần nhỏ lại, những con chip này cũng trở nên nhỏ hơn, nhanh hơn và thông minh hơn theo thời gian.

1.2. Cấu trúc hệ thống IoT

Cấu trúc của một hệ thống IoT thường bao gồm các lớp chính sau:

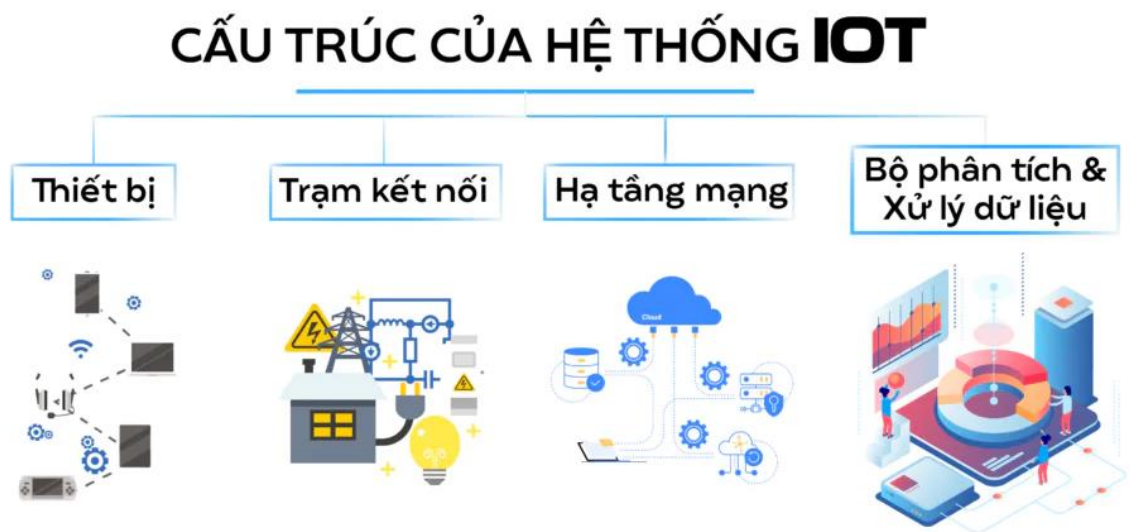
Thiết bị (Things): Lớp này chứa các thiết bị IoT, bao gồm cảm biến, actuators, và các thiết bị thông minh khác thu thập dữ liệu từ môi trường hoặc thực hiện các hành động cụ thể. Đây là nơi xuất phát của dữ liệu trong hệ thống.

Trạm kết nối (Gateways): Lớp Gateway đóng vai trò như là cầu nối giữa thiết bị và hạ tầng mạng. Nó có thể thực hiện chức năng lọc dữ liệu, xử lý cơ bản và truyền tải dữ liệu từ nhiều thiết bị về hạ tầng mạng.

Hạ tầng mạng (Network and Cloud): Lớp này bao gồm các cơ sở hạ tầng mạng, đặc biệt là môi trường đám mây (cloud) nơi dữ liệu từ thiết bị được chuyển

đến và lưu trữ. Hạ tầng mạng cũng cung cấp các dịch vụ như tích hợp, xử lý dữ liệu, và lưu trữ.

Bộ phân tích và xử lý dữ liệu (Services-creation and solution layers): Lớp này chứa các dịch vụ và giải pháp được tạo ra để xử lý và phân tích dữ liệu từ thiết bị IoT. Bao gồm các công nghệ như trí tuệ nhân tạo (AI), machine learning, và các dịch vụ tạo ra giải pháp thông minh và dự đoán.



Hình 1. 1: Cấu trúc hệ thống IoT

1.3. Ưu và nhược điểm của IoT

Ưu điểm của IoT

Tối ưu hóa kỹ thuật: Công nghệ IoT có thể thu thập dữ liệu cảm biến từ sản phẩm. Nhà sản xuất phân tích chúng để cải tiến thiết kế và làm chúng hoạt động hiệu quả hơn.

Giảm lãng phí: IoT cung cấp thông tin thời gian thực giúp đưa ra quyết định và quản lý tài nguyên hiệu quả.

Cải thiện mức độ tương tác của khách hàng: IoT cho phép bạn cải thiện trải nghiệm của khách hàng bằng cách phát hiện các vấn đề và cải thiện quy trình.

Nhược điểm của IoT

Tính bảo mật: Công nghệ IoT tạo ra một hệ sinh thái các thiết bị được kết nối. Tuy nhiên, trong quá trình này, hệ thống có thể cung cấp tính xác thực không cao mặc dù có đủ các biện pháp bảo mật.

Quyền riêng tư: Sử dụng IoT tiết lộ lượng lớn dữ liệu cá nhân, rất chi tiết, mà người dùng không chủ động thực hiện điều đó.

Tính phức tạp: thiết kế hệ thống IoT khá phức tạp và bảo trì cũng không hề dễ dàng.

1.4. Ứng dụng của IoT

Với sự phát triển nhanh chóng của công nghệ, Internet of Things (IoT) đang trở thành một phần không thể thiếu trong cuộc sống hiện đại của chúng ta. Ứng dụng của IoT có thể được tìm thấy trong nhiều lĩnh vực khác nhau.

Đối với doanh nghiệp việc vận hành và tối ưu doanh nghiệp một cách hiệu quả, nhanh chóng luôn là một điều vô cùng quan trọng đối với các chủ doanh

ng nghiệp, việc ứng dụng IoT vào quá trình sản xuất, dây chuyền, kết nối và trao đổi thông tin dữ liệu trong mọi hoạt động của doanh nghiệp đã đẩy nhanh hơn tốc độ sản xuất của doanh nghiệp, đồng thời giúp các doanh nghiệp nhanh chóng tìm ra được những lỗi trong dây chuyền và kịp thời khắc phục chúng. Ngoài ra cũng nhờ có IoT, việc thực hiện cung ứng và hoàn thiện sản phẩm cũng trở nên hoàn hảo hơn.



Hình 1. 2: Ứng dụng của IoT

Đối với người tiêu dùng sự xuất hiện của IoT trong cuộc sống con người đó là kiểm soát hoạt động thông qua các thiết bị kết nối internet như camera, cảm ứng ngoài sân, cảm ứng âm thanh... giúp con người kiểm soát tốt cuộc sống và nhanh chóng có được các các giải quyết vấn đề chỉ với điện thoại di động.



Hình 1. 3: Smart home

Trong ngành marketing khi ứng dụng IoT vào lĩnh vực sẽ nhanh chóng giúp các marketer thu thập được thông tin của người dùng, tiếp cận được nhiều thông tin để từ đó có dữ liệu phân tích, tối ưu hơn hoạt động của mình cũng như đáp ứng được các nhu cầu, sở thích khác nhau của khách hàng.

1.5. Nhà thông minh là gì?

Nhà thông minh (hay còn gọi là smart home) là một hệ thống sử dụng các công nghệ và thiết bị điện tử để tự động hóa và tạo ra sự kết nối thông minh trong việc quản lý và điều khiển các thiết bị và hệ thống trong ngôi nhà. Mục tiêu của nhà thông minh là cung cấp sự tiện ích, an toàn và tiết kiệm năng lượng cho cư dân.

Các thiết bị và hệ thống trong nhà thông minh có thể được kết nối và điều khiển thông qua mạng Internet, cho phép người dùng từ xa quản lý và điều khiển các chức năng như ánh sáng, nhiệt độ, an ninh, âm thanh, hình ảnh, thiết bị gia dụng và các thiết bị khác.

Các công nghệ thông minh như học máy, trí tuệ nhân tạo (AI), Internet of Things (IoT) và các giao thức kết nối không dây như Wi-Fi và Bluetooth, được sử dụng để tạo ra một môi trường nhà thông minh thông minh và tự động.

Với nhà thông minh, người dùng có thể điều khiển các thiết bị thông qua điện thoại di động, máy tính hoặc các thiết bị điều khiển đặc biệt. Họ có thể tạo lịch trình tự động, tùy chỉnh các cài đặt và nhận thông báo cảnh báo từ hệ thống.

1.6. Các mô hình nhà thông minh

Các mô hình thiết bị nhà thông minh đang trở thành xu hướng phổ biến trong kỷ nguyên số hiện nay. Những sản phẩm như loa thông minh, camera an ninh, hệ thống điều khiển nhiệt độ, và các ứng dụng được tích hợp trong ngôi nhà đang đóng vai trò ngày càng quan trọng trong việc nâng cao chất lượng cuộc sống của người dùng.

Các thiết bị này không chỉ mang lại sự tiện lợi trong cuộc sống hàng ngày, như điều chỉnh nhiệt độ, kiểm soát an ninh, và điều khiển các thiết bị trong nhà bằng giọng nói, mà còn tiết kiệm được đáng kể chi phí năng lượng. Hơn nữa, các tính năng thông minh dựa trên công nghệ trí tuệ nhân tạo và học máy giúp các thiết bị có khả năng tự động hóa, thích ứng và lập lịch trình phù hợp với thói quen của người dùng.

Hiện nay có rất nhiều các tập đoàn lớn trong ngành công nghệ đều tập trung phát triển các sản phẩm và hệ sinh thái trong lĩnh vực này, mỗi bên đều có những điểm mạnh và định hướng riêng, ví dụ nổi bật là Google.

Mô hình thiết bị nhà thông minh của Google bao gồm các sản phẩm như Google Nest Hub - một thiết bị hiển thị thông minh với trợ lý ảo Google Assistant, có thể điều khiển các thiết bị trong nhà như đèn, máy điều hòa và camera an ninh. Sản phẩm khác là Google Nest Cam - camera an ninh thông minh, có khả năng phát hiện chuyển động và cảnh báo người dùng.



Hình 1. 4: Thiết bị của Google

Những mô hình thiết bị này đều tập trung vào việc tích hợp các chức năng thông minh, tự động hóa và kết nối, giúp người dùng quản lý và kiểm soát các thiết bị gia dụng một cách dễ dàng và hiệu quả hơn.

Tiểu kết: Chương này đã cung cấp một cái nhìn tổng quan về khái niệm Internet of Things (IoT) và ứng dụng của nó trong lĩnh vực nhà thông minh. IoT là mạng lưới các thiết bị kết nối với nhau và trao đổi dữ liệu, trong khi nhà thông minh là việc tích hợp các công nghệ IoT vào ngôi nhà để tự động hóa và tối ưu hóa các chức năng.

Các ưu điểm chính của nhà thông minh bao gồm tăng cường tiện nghi, hiệu quả năng lượng, an ninh và quản lý tài nguyên. Tuy nhiên, việc áp dụng các công nghệ IoT trong nhà ở cũng đặt ra những thách thức liên quan đến bảo mật, quyền riêng tư và độ tin cậy của hệ thống.

CHƯƠNG 2: CẢNH BÁO BẤT THƯỜNG TRONG NHÀ THÔNG MINH

2.1. Cảnh báo về an ninh

2.1.1. Cảnh báo ngoài nhà

Mô tả cảnh báo:

Rủi ro xâm nhập trái phép là mối đe dọa nghiêm trọng, khi kẻ gian cố tình vượt qua rào chắn an ninh để đột nhập vào bên trong ngôi nhà. Điều này đe dọa an toàn của gia đình. Ngoài ra, trộm cắp cũng là một rủi ro đáng lưu tâm. Kẻ gian có thể nhắm vào các tài sản có giá trị bên ngoài ngôi nhà, như xe hơi, máy móc, hay các thiết bị sân vườn

Phương pháp giải quyết:

An ninh và bảo vệ là mối quan tâm hàng đầu. Hệ thống nhà thông minh cần được trang bị các công nghệ tiên tiến như camera giám sát, cảm biến chuyển động và khóa thông minh để ngăn chặn kẻ gian xâm nhập vào ngôi nhà, khi phát hiện có kẻ lạ xâm nhập ngoài nhà lập tức đèn sẽ sáng và còi kêu, ngăn chặn các hành vi có kẻ lạ xâm nhập.

2.1.2. Cảnh báo trong nhà

Mô tả cảnh báo:

Bảo vệ dữ liệu cá nhân cũng là một mối quan ngại lớn. Người dùng cần tránh lưu trữ thông tin nhạy cảm trên các thiết bị thông minh và sử dụng mã hóa dữ liệu để bảo vệ thông tin. Việc chia sẻ dữ liệu với bên thứ ba cũng cần được cân nhắc kỹ lưỡng.

Ngoài ra, với các hành vi của kẻ gian đột nhập vào nhà và lấy cắp tài sản, có sự di chuyển bất thường của người lạ mặt, cũng là rủi ro đáng quan ngại nên cần đề cao cảnh giác.

Phương pháp giải quyết:

Sử dụng các thiết bị lưu trữ an toàn như ổ cứng di động mã hóa dữ liệu và dịch vụ lưu trữ đám mây bảo mật có thể bảo vệ thông tin cá nhân khỏi nguy cơ bị truy cập trái phép. Tiếp theo, hệ thống an ninh thông minh với camera nhận dạng khuôn mặt và chuông cửa có xác thực, sẽ theo dõi và kiểm soát quyền truy cập vào nhà một cách an toàn.

Sử dụng các thiết bị cảm biến như cảm biến chuyển động cùng với camera để phát hiện có kẻ gian đột nhập bất thường, nếu phát hiện sự bất thường trong nhà đèn và còi sẽ lập tức thông báo, đồng thời điện thoại sẽ báo cho chủ nhà biết có phát hiện bất thường.

2.2. Cảnh báo trong môi trường nhà

2.2.1. Cảnh báo cháy

Mô tả cảnh báo:

Một trong những mối nguy hàng đầu là các lỗi kỹ thuật có thể xảy ra. Các thiết bị công nghệ trong nhà thông minh, dù được thiết kế và lắp đặt cẩn thận, vẫn có khả năng gặp sự cố như chập điện, lỗi phần cứng hoặc lỗi phần mềm. Những vấn đề kỹ thuật này, nếu không được phát hiện và xử lý kịp thời, có thể trở thành ngọn lửa đầu tiên gây ra hỏa hoạn.

Ngoài ra, việc sử dụng quá nhiều thiết bị điện cùng lúc, đặc biệt là những thiết bị có công suất lớn, cũng có thể dẫn đến tình trạng quá tải điện. Điều này có thể khiến hệ thống điện bị sự cố và gây ra cháy nổ.

Một nguy cơ khác là do lỗi của con người. Việc sử dụng hoặc vận hành các thiết bị không đúng cách, chẳng hạn như để nến cháy gần các vật liệu dễ cháy, hoặc để

Các thiết bị sưởi ấm ở gần các vật dễ cháy, cũng có thể trở thành nguyên nhân gây ra hỏa hoạn.

Phương pháp giải quyết:

Sử dụng các thiết bị như cảnh báo cháy, bình chữa cháy tự động, vòi phun nước có thể được kết nối với hệ thống an ninh nhà thông minh. Khi cảm biến phát hiện cháy, chúng sẽ tự động kích hoạt, dập tắt đám cháy và báo động cho chủ nhà. Đây là giải pháp hiệu quả để ứng phó nhanh chóng với các tình huống khẩn cấp.

Bên cạnh đó, các ổ cắm thông minh có thể được lập trình để tự động ngắt điện khi phát hiện các yếu tố như quá tải, đoản mạch hoặc nhiệt độ cao. Điều này sẽ ngăn chặn nguy cơ cháy do chập điện hoặc sự cố kỹ thuật. Thông qua ứng dụng trên điện thoại, chủ nhà có thể giám sát và điều khiển dòng điện trong nhà, đảm bảo sử dụng an toàn.

Các thiết bị thông minh khác như camera giám sát, cổng truy cập, và hệ thống chiếu sáng thông minh cũng có thể được tích hợp vào hệ thống an ninh nhà thông minh. Chúng sẽ giúp người dùng kiểm soát và giám sát toàn bộ ngôi nhà từ xa, đồng thời cung cấp thông tin quan trọng để ứng phó kịp thời với các tình huống nguy hiểm.

2.2.2. Cảnh báo khí gas

Mô tả cảnh báo:

Rò rỉ khí gas là mối nguy cơ lớn nhất. Nếu có rò rỉ xảy ra, khí gas dễ cháy và nổ có thể gây ra hỏa hoạn, thậm chí là vụ nổ, gây thiệt hại nghiêm trọng về tài sản và tính mạng. Nguy cơ này càng tăng cao khi các gia đình sử dụng bình gas mini hoặc ống dẫn gas cũ, không đảm bảo an toàn.

Ngoài ra, việc sử dụng thiết bị gas không đúng cách, chẳng hạn như để bếp ga mở quá lâu hoặc đặt các vật dụng dễ cháy gần bếp, cũng có thể dẫn đến cháy nổ. Hơn nữa, việc lưu trữ và di chuyển bình gas không đúng quy trình an toàn cũng tiềm ẩn nhiều nguy hiểm.

Phương pháp giải quyết:

Sử dụng các cảm biến khí gas có thể được lắp đặt tại các vị trí chiến lược trong nhà, như gần bếp ga, khu vực lưu trữ bình gas, hay các không gian kín. Các cảm biến này sẽ liên tục theo dõi nồng độ khí gas và phát tín hiệu cảnh báo khi phát hiện rò rỉ.

Hệ thống nhà thông minh sẽ nhận và xử lý các tín hiệu từ các cảm biến này. Khi có cảnh báo về rò rỉ khí gas, hệ thống sẽ thực hiện một loạt các hành động tự động để đảm bảo an toàn. Ví dụ, nó có thể tự động ngắt nguồn cấp khí gas, bật đèn cảnh báo, và thông báo cho chủ nhà hoặc các bên liên quan như công ty cung cấp gas.

2.2.3. Cảnh báo chất lượng không khí

Mô tả cảnh báo:

Chất lượng không khí sống là vấn đề ngày càng được quan tâm trong thời đại công nghiệp hóa và đô thị hóa nhanh chóng. Nhiều rủi ro đe dọa sức khỏe của con người và môi trường sống từ ô nhiễm không khí cần được nhận diện và giải quyết.

Một trong những rủi ro chính là ô nhiễm không khí do khí thải từ các phương tiện giao thông. Khí thải từ xe cộ chứa các chất ô nhiễm như carbon monoxide, nitrogen oxides, và các hạt bụi mịn. Việc hít thở không khí ô nhiễm như vậy có thể gây ra các vấn đề về hô hấp, tim mạch, và ung thư.

Phương pháp giải quyết:

Sử dụng các thiết bị kiểm soát độ ẩm thông minh cũng rất hữu ích. Những thiết bị này có thể theo dõi và duy trì độ ẩm ở mức độ lý tưởng, ngăn ngừa sự phát triển của nấm mốc và vi khuẩn có hại. Việc kiểm soát độ ẩm chính xác là rất quan trọng đối với sức khỏe và sự thoải mái của cư dân.

Một giải pháp khác là các hệ thống thông gió thông minh. Những hệ thống này có thể tự động thay đổi tốc độ thông gió dựa trên chất lượng không khí trong nhà, đảm bảo không khí luân chuyển liên tục và loại bỏ các chất ô nhiễm. Kết hợp với các bộ lọc hiệu quả, hệ thống thông gió thông minh có thể giúp cải thiện đáng kể chất lượng không khí nội thất.

2.2.4. Cảnh báo rò rỉ nước

Mô tả cảnh báo:

Nhà rò rỉ nước cũng có thể dẫn đến sự hình thành và phát triển của nấm mốc, vi khuẩn và các loại côn trùng. Những yếu tố này có thể ảnh hưởng nghiêm trọng đến sức khỏe của cư dân, gây ra các vấn đề về hô hấp, dị ứng hoặc các bệnh truyền nhiễm. Điều này không chỉ ảnh hưởng đến chất lượng sống mà còn có thể dẫn đến chi phí y tế đáng kể.

Hơn nữa, nước rò rỉ cũng có thể làm suy yếu cấu trúc của ngôi nhà, dẫn đến nguy cơ sụp đổ hoặc các vấn đề an toàn khác. Điều này không chỉ đe dọa tài sản mà còn có thể gây nguy hiểm cho tính mạng của cư dân.

Phương pháp giải quyết:

Đề ứng phó với vấn đề rò rỉ nước nghiêm trọng, một giải pháp hiệu quả là sử dụng các thiết bị cảnh báo rò rỉ nước. Những thiết bị này không chỉ giúp phát hiện sớm các vấn đề rò rỉ mà còn cung cấp các biện pháp khắc phục nhanh chóng, qua đó giảm thiểu các rủi ro và thiệt hại đáng kể.

Các thiết bị cảnh báo rò rỉ nước thường sử dụng các cảm biến để liên tục theo dõi việc sử dụng nước trong nhà. Khi phát hiện bất thường như sự gia tăng đột ngột lượng nước tiêu thụ, thiết bị sẽ phát tín hiệu cảnh báo để thông báo cho chủ nhà. Điều này cho phép chủ nhà nhanh chóng kiểm tra và xác định vị trí rò rỉ, từ đó có thể tiến hành sửa chữa kịp thời.

2.3. Cảnh báo hoạt động của thiết bị

2.3.1. Cảnh báo thiết bị đang bật hoặc đang tắt

Mô tả cảnh báo:

Các đèn vẫn bật ngay cả khi không có ai ở nhà có thể dẫn đến lãng phí năng lượng đáng kể. Ngay cả khi các đèn được lập trình để tự động bật khi phát hiện chuyển động, điều này có thể xảy ra do các yếu tố bên ngoài như gió, động vật hoặc bóng râm. Điều này ảnh hưởng đến hóa đơn tiền điện và ảnh hưởng đến môi trường.

Ngoài ra, việc camera an ninh tự động bật nhưng lại tắt đột ngột cũng là một vấn đề đáng lo ngại. Điều này có thể xảy ra do sự cố kỹ thuật hoặc lỗi trong lập trình, khiến hệ thống giám sát bị gián đoạn. Điều này làm giảm đáng kể tính an toàn và bảo mật của ngôi nhà khi chủ nhà vắng mặt.

Hơn nữa, việc các thiết bị gia dụng như máy giặt hoặc lò nướng tự động tắt không phải lúc nào cũng an toàn. Nếu vào thời điểm đó, các thiết bị này đang hoạt động hoặc đang ở trong quá trình xử lý thì việc tự động tắt có thể gây ra hư hỏng hoặc thậm chí nguy hiểm hơn như cháy nổ.

Phương pháp giải quyết:

Cần thiết lập các chính sách và cài đặt thông minh hơn cho hệ thống nhà thông minh. Ví dụ, người dùng có thể lập trình các đèn chiếu sáng để tự động tắt sau một thời gian nhất định nếu không phát hiện chuyển động. Đối với camera an ninh, có thể cài đặt chế độ bật/tắt tự động dựa trên lịch trình hoặc sự kiện cụ thể thay vì hoạt động liên tục.

Ngoài ra, người dùng cũng nên trang bị các tính năng giám sát và kiểm soát từ xa thông qua điện thoại hoặc thiết bị di động. Điều này cho phép họ kiểm tra tình trạng các thiết bị trong nhà và thực hiện các hành động cần thiết, ví dụ như tắt đèn hoặc camera khi không cần thiết.

Tiểu kết: Chương này đã tập trung vào các cảnh báo bất thường có thể xảy ra trong một ngôi nhà thông minh và các giải pháp sử dụng các thiết bị thông minh để phát hiện và ứng phó với những tình huống này.

Chương này tìm hiểu về các loại cảnh báo bất thường như rò rỉ nước, nguy cơ cháy nổ, xâm nhập trái phép và sự cố năng lượng. Những cảnh báo này không chỉ gây nguy hiểm cho tài sản và tính mạng, mà còn có thể dẫn đến các tổn thất lớn về kinh tế và ảnh hưởng đến chất lượng sống của cư dân.

Để đối phó với những tình huống này, phải xem xét các giải pháp sử dụng các thiết bị thông minh như cảm biến rò rỉ nước, cảm biến khói và nhiệt, camera an ninh và hệ thống quản lý năng lượng thông minh. Những thiết bị này có thể liên kết với nhau thông qua mạng IoT, giúp phát hiện và đưa ra cảnh báo kịp thời, đồng thời cho phép chủ nhà điều khiển và ứng phó từ xa.

CHƯƠNG 3: CÁC THIẾT BỊ ARDUINO DÙNG TRONG MÔ HÌNH NHÀ THÔNG MINH

3.1. Thiết bị vi điều khiển: vi điều khiển (Arduino Uno R3)

Vi điều khiển:

Arduino Uno R3 sử dụng vi điều khiển Atmega328P, một trong ba vi điều khiển 8-bit AVR họ ATmega (ATmega8, ATmega168, ATmega328) vi điều khiển ATmega 328p có thể xử lý các tác vụ đơn giản như điều khiển đèn LED nhấp nháy, xử lý tín hiệu cho xe điều khiển từ xa, làm trạm đo nhiệt độ- độ ẩm và hiển thị lên màn hình LCD,...

Nguồn điện:

Arduino Uno R3 có thể được cấp nguồn 5V thông qua cổng USB, nhưng điện áp khuyến dùng là 7-12V và giới hạn là 20V nếu cấp nguồn cao hơn 20V thì có thể bị hỏng bo mạch.

Các chân nguồn của Uno R3:

+ GND: Cực âm của nguồn điện cấp cho Arduino Uno. Khi bạn dùng các thiết bị sử dụng những nguồn điện riêng biệt thì những chân này phải được nối với nhau.

+ Cấp điện áp 5V đầu ra. Dòng tối đa cho phép ở chân này là 500mA.

+ 3.3V cấp điện áp 3.3V đầu ra. Dòng tối đa cho phép ở chân này là 50mA.

+ Vin (Voltage Input): để cấp nguồn ngoài cho Arduino UNO, bạn nối cực dương của nguồn với chân này và cực âm của nguồn với chân GND.

+ IOREF: điện áp hoạt động của vi điều khiển trên Arduino UNO có thể được đo ở chân này. Và dĩ nhiên nó luôn là 5V. Mặc dù vậy bạn không được lấy nguồn 5V từ chân này để sử dụng bởi chức năng của nó không phải là cấp nguồn.

+ RESET: việc nhấn nút Reset trên board để reset vi điều khiển tương đương với việc chân RESET được nối với GND qua 1 điện trở 10KΩ.

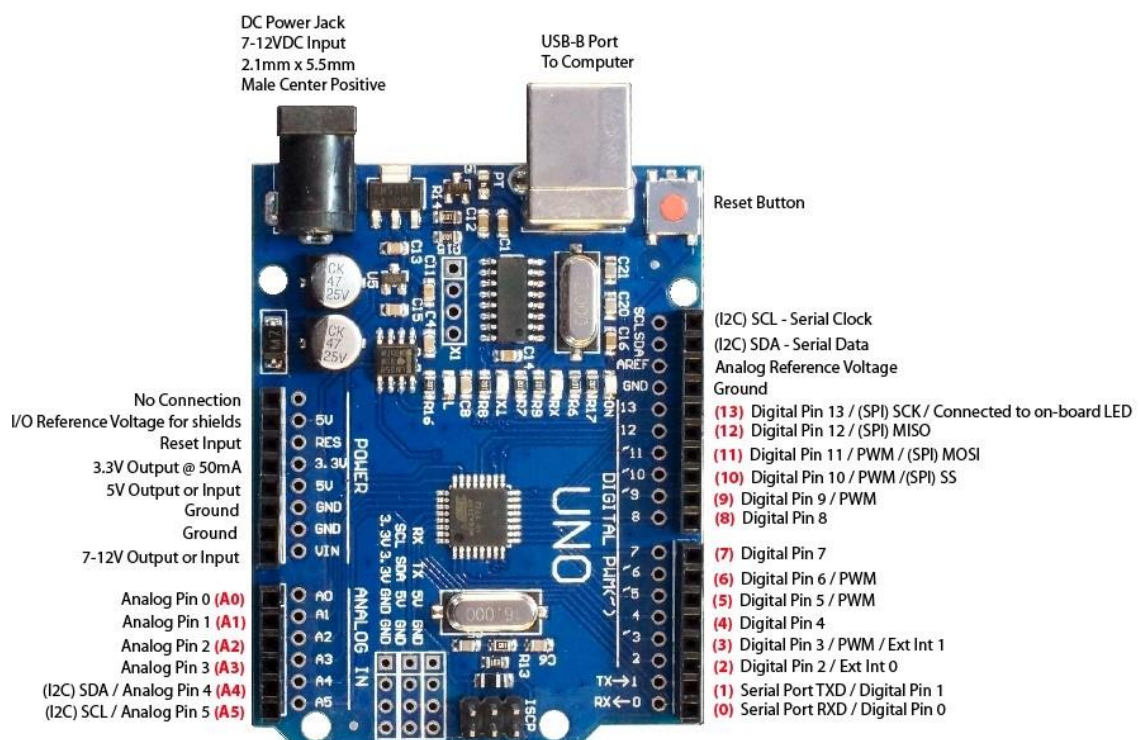
Bộ nhớ:

+ 32KB bộ nhớ Flash: những đoạn lệnh bạn lập trình sẽ được lưu trữ trong bộ nhớ Flash của vi điều khiển.

+ 2KB cho SRAM (Static Random Access Memory): là nơi để lưu giá trị các biến khi lập trình.

+ 1KB cho EEPROM (Electrically Erasable Programmable Read Only Memory): đây là nơi đọc và ghi dữ liệu lâu dài ngay cả khi mất điện.

Các chân vào/ ra:



Hình 3. 1: Arduino Uno R3

+ 2 chân Serial (0(RX) và 1(TX)): cổng TX dùng để gửi và RX dùng để nhận dữ liệu. Uno có thể giao tiếp với các thiết bị khác thông qua 2 chân này

+ Chân từ 2 đến 13: Các chân số có thể cấu hình làm đầu vào hoặc đầu ra, điều này cho phép kết nối và điều khiển các thiết bị như nút bấm, công tắc, đèn led,...

+ Chân PWM (3,5,6,9,10,11) : cho phép bạn xuất ra xung PWM với độ phân giải 8 bit bằng hàm analogWrite(), chúng ta có thể điều chỉnh được điện áp ra ở chân này từ mức 0V đến 5V thay vì chỉ cố định ở mức 0V và 5V như những chân khác.

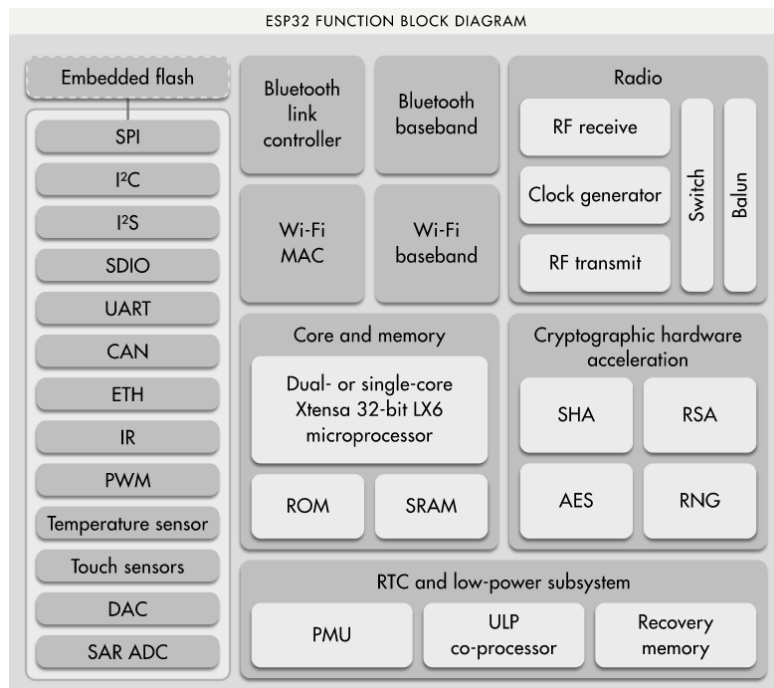
+ Chân Analog: gồm 6 chân analog từ chân A0 đến chân A5 các chân này thường được sử dụng trong các thiết bị như cảm biến ánh sáng, cảm biến độ ẩm, ... Dữ liệu analog đọc được từ các chân này có thể sử dụng trực tiếp trong các ứng dụng hoặc xử lý thêm bằng phần mềm.

+ Các chân khác: AREF chân điện áp tham chiếu analog, SCL và SDA 2 chân giao tiếp với I2C, ISCP là một header 6 chân được sử dụng để lập trình vi điều khiển ATmega328p trên Arduino Uno R3 bằng phương pháp lập trình nối tiếp.

3.2. Thiết bị kết nối: kết nối internet bằng wifi với esp32

ESP32 là một vi điều khiển chip (SoC) mạnh mẽ giúp kết nối Wifi, Bluetooth có thể lập trình nhiều ngôn ngữ khác nhau như C/C++, Python, NodeJs, Lua, Có giá thành rẻ, kế thừa từ ESP 8266, có cả hai biến thể lõi đơn và lõi kép của bộ vi xử lý 32-bit

- CPU :



Hình 3. 2: CPU 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 520 KB SRAM liền chip trong đó có 8 Kb RAM RTC tốc độ cao – 8 Kb RAM tốc độ thấp (dùng ở chế độ DeepSleep).
- + 34 GPIO có thể lập trình.
- Hỗ trợ giao tiếp không dây:
 - + 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.
- Hỗ trợ tất cả các loại giao tiếp:
 - + 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.

- Cảm biến tích hợp trên chip ESP32

+ 1 cảm biến Hall (Cảm biến từ trường)

+ 1 cảm biến đo nhiệt độ

+ Cảm biến chạm (điện dung) với 10 đầu vào khác nhau.

- Bảo mật:

+ Hỗ trợ tất cả các tính năng bảo mật chuẩn IEEE 802.11, bao gồm WPA, WPA/WPA2 và WAPI

+ Khởi động an toàn (Secure boot)

+ Mã hóa flash (Flash encryption)

+ 1024-bit OTP, lên đến 768-bit cho khách hàng

+ Tăng tốc phần cứng mật mã: AES, SHA-2, RSA, mật mã đường cong elliptic (ECC – elliptic curve cryptography), bộ tạo số ngẫu nhiên (RNG – random number generator)

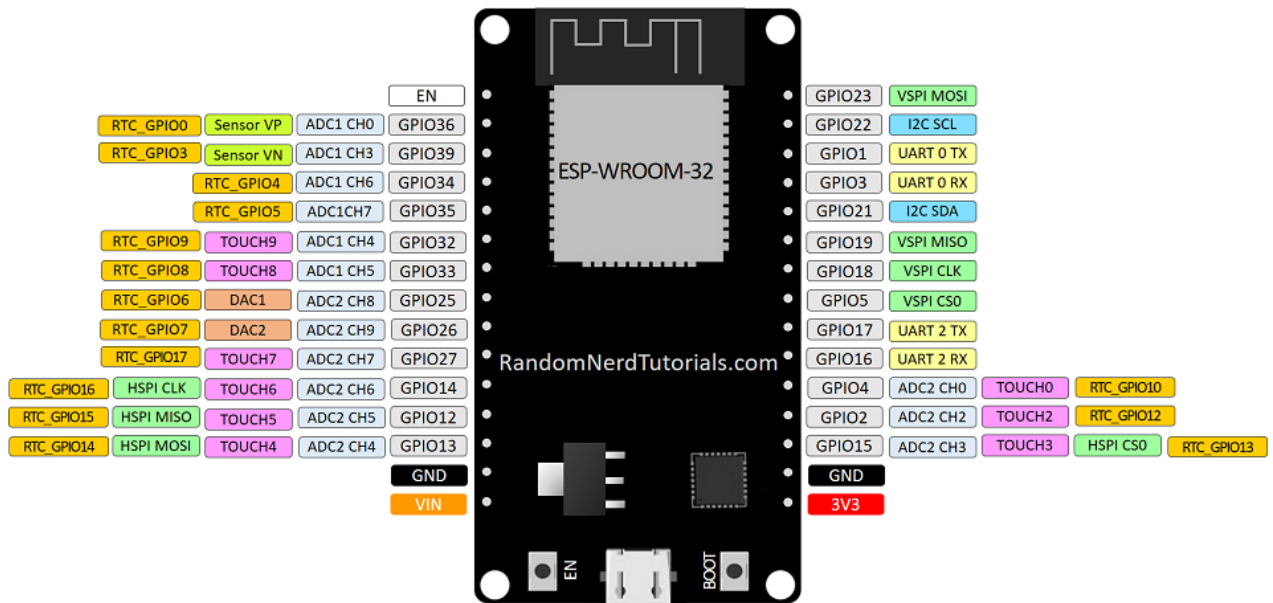
-Nguồn điện hoạt động:

+ Điện áp hoạt động: $2,2V \div 3,6V$

+ Nhiệt độ hoạt động: $-40^{\circ}C \div + 85^{\circ}C$

+ Số cổng GPIO: 36

- Ngoại vi :



Hình 3. 3: Sơ đồ chân Esp32

ESP32 có tổng cộng 48 chân với nhiều chức năng khác nhau, có các chân lộ ra nhưng cũng có các chân được ẩn đi. Ảnh trên là sơ đồ các chân của chip nhưng vị trí các chân có thể được thay đổi tùy thuộc vào nhà sản xuất.

+ Chân chỉ Input là các chân GPIO từ 34 đến 39 (GPIO34, GPIO35, GPIO36, GPIO39), các chân này không có điện trở nên chỉ có thể làm đầu vào.

+ Chân tích hợp Flash trên ESP32 : Là các chân từ GPIO6 đến GPIO11 (GPIO6, GPIO7, GPIO8, GPIO9, GPIO10, GPIO11) dùng để kết nối Flash SPI trên chip ESP-WROOM-32.

+ Chân cảm biến điện dung : có tổng cộng 10 chân, chúng có thể phát hiện sự thay đổi điện áp cảm ứng trên các chân GPIO. Chúng cũng có thể đánh thức ESP32 khỏi chế độ ngủ sâu. Các cảm biến điện dung được kết nối với các GPIO như : TOUCH0 (GPIO4), TOUCH1 (GPIO0), TOUCH2 (GPIO2), TOUCH3 (GPIO15), TOUCH4 (GPIO13), TOUCH5 (GPIO12), TOUCH6 (GPIO14), TOUCH7 (GPIO27), TOUCH8 (GPIO33), TOUCH9 (GPIO32)

+ Bộ chuyển đổi chân tương tự sang số ADC (Analog to Digital Converter): có 18 kênh đầu vào ADC 12 bit (trong khi ESP8266 chỉ có một kênh ADC là 12 bit). Các GPIO có thể được sử dụng làm ADC và các kênh tương ứng là ADC1_CH0(GPIO36), ADC1_CH1(GPIO37), ADC1_CH2(GPIO38), ADC1_CH3(GPIO39), ADC1_CH3(GPIO32), ADC1_CH5(GPIO33), ADC1_CH6(GPIO34), ADC1_CH7(GPIO35), ADC2_CH0(GPIO4), ADC2_CH1(GPIO0), ADC2_CH2(GPIO2), ADC2_CH3(GPIO5), ADC2_CH4(GPIO13), ADC2_CH5(GPIO12), ADC2_CH6(GPIO14), ADC2_CH7(GPIO27), ADC2_CH8(GPIO25), ADC2_CH9(GPIO26).

+ Bộ chuyển đổi số sang tương tự DAC (Digital to Analog Converter) : có hai kênh (DAC1 (GPIO25, DAC2 (GPIO26)) 8 bit có nghĩa là có giá trị từ 0 / 255 tương ứng với 0/3,3V.

+ Các chân thời gian thực RTC : Có tác dụng đánh thức ESP32 khi đang ở chế độ ngủ sâu. Sử dụng như một chân ngắt ngoài : RTC_GPIO0 (GPIO36), RTC_GPIO3 (GPIO39), RTC_GPIO4 (GPIO34), RTC_GPIO5 (GPIO35), RTC_GPIO6 (GPIO25), RTC_GPIO7 (GPIO26), RTC_GPIO8 (GPIO33), RTC_GPIO9 (GPIO32), RTC_GPIO10 (GPIO4), RTC_GPIO11 (GPIO0), RTC_GPIO12 (GPIO2), RTC_GPIO13 (GPIO15), RTC_GPIO14 (GPIO13), RTC_GPIO15 (GPIO12), RTC_GPIO16 (GPIO14), RTC_GPIO17 (GPIO27).

+ Chân PWM: có 16 kênh độc lập được cấu hình để tạo tín hiệu PWM với các thuộc tính khác nhau. Tất cả các chân có thể hoạt động như đầu ra đều có thể được sử dụng để làm chân PWM (GPIO từ 34 đến 39 không thể tạo PWM).

+ Chân I2C: có 2 kênh và bất kì chân nào cũng có thể làm SDA hoặc SCL. Khi sử dụng ESP32 với Arduino IDE thì chân I2C mặc định là: GPIO21(SDA) và GPIO22(SCL).

+ Chân SPI: Các chân ánh xạ cho SPI là:

SPI	MOSI	MISO	CLK	CS
VSPI	GPIO 23	GPIO 19	GPIO 18	GPIO 5
HSPI	GPIO 13	GPIO 12	GPIO 14	GPIO 15

bảng 3.1: các chân ánh xạ cho SPI

+ Chân ngắt ngoài : Tất cả các chân đều có thể sử dụng như một chân ngắt ngoài.

3.3. Thiết bị cảm biến

3.3.1. Cảm biến nhiệt độ, độ ẩm

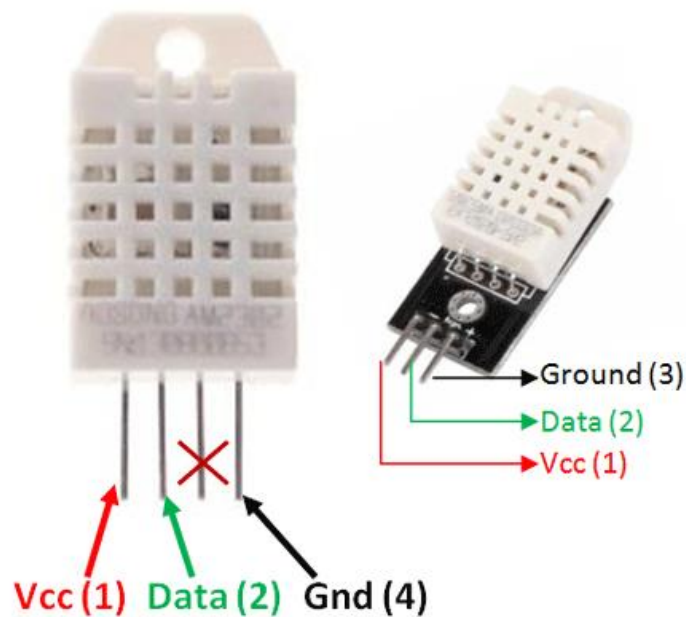
Cảm biến DHT 22 là cảm biến phổ biến trong rất nhiều dự án sản xuất điện tử và cũng được chúng tôi sử dụng trong mô hình nhà thông minh của mình.

DHT 22 được tích hợp và đo được cả 2 thông số là nhiệt độ và độ ẩm rất tiện lợi khi chúng ta không phải mua 2 cái cảm biến riêng biệt, và cảm biến này cũng có giá thành rẻ.



Hình 3. 4: DHT22

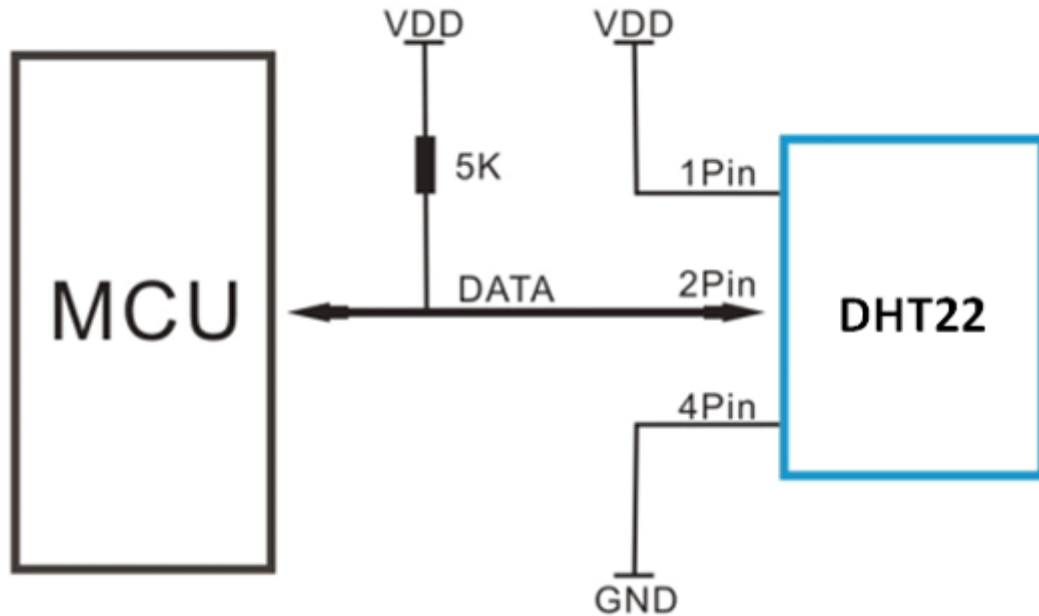
DHT 22 có độ chính xác cao hơn, tần suất thu thập dữ liệu và phạm vi đo cũng cao hơn DHT 11.



Hình 3. 5: sơ đồ chân của DHT22

- Thông số kỹ thuật của cảm biến DHT 22
- + Điện áp hoạt động: 3.5V đến 5.5V
- + Dòng hoạt động: 0,3mA (đo) 60uA (chế độ chờ)
- + Đầu ra: Dữ liệu nối tiếp
- + Phạm vi nhiệt độ: -40 ° C đến 80 ° C
- + Phạm vi độ ẩm: 0% đến 100%
- + Độ phân giải: Nhiệt độ và Độ ẩm đều là 16-bit
- + Độ chính xác: $\pm 0,5$ ° C và $\pm 1\%$

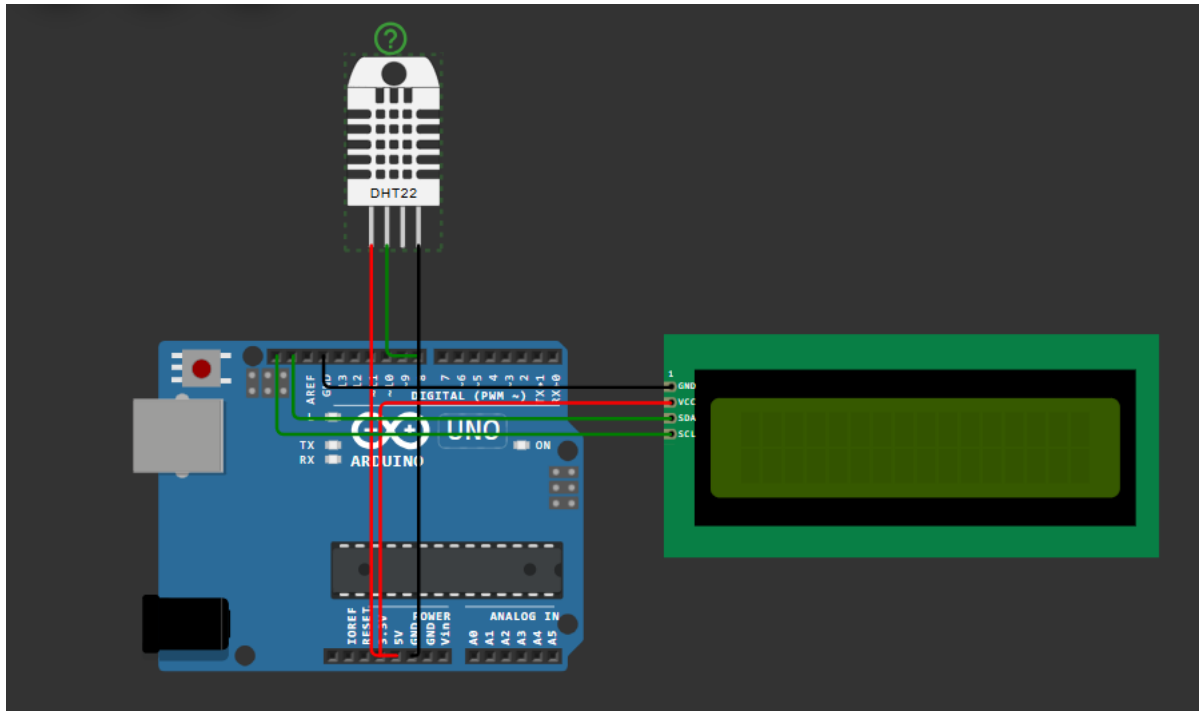
Cảm biến DHT22 đi kèm với một NTC chuyên dụng để đo nhiệt độ và bộ vi điều khiển 8 bit để xuất các giá trị nhiệt độ và độ ẩm dưới dạng dữ liệu nối tiếp, cảm biến đã được hiệu chuẩn nên dễ dàng giao tiếp với các bộ vi điều khiển khác. Cảm biến có thể đo nhiệt độ từ -40 ° C đến 80 ° C và độ ẩm từ 0% đến 100% với độ chính xác ± 1 ° C và $\pm 1\%$. Vì vậy, Cảm biến này sẽ phù hợp, chính xác khi chúng ta lựa chọn đúng môi trường để đặt cảm biến.



Hình 3. 6: sơ đồ kết nối của DHT22

- Như hình 8 thì chân dữ liệu được kết nối với chân I/O của MCU và một điện trở kéo lên 5K, chân dữ liệu này xuất ra các giá trị của nhiệt độ và độ ẩm dưới dạng dữ liệu nối tiếp.
- Các kết nối DHT 22 với các thiết bị khác trong ngôi nhà sẽ được giao tiếp thông qua Arduino Uno chạy trên môi trường Arduino IDE thông qua các thư viện đã có sẵn.

Cách nối dây:



Hình 3. 7: Cách nối dây FHT22

Trong cảm biến dht22 chân đầu tiên từ trái qua phải sẽ được nối với chân 5V của Arduino uno, chân ở giữa sẽ chân nối với chân digital, chân tiếp theo là chân no connect chân này không cần nối và chân cuối sẽ nối vào chân GND.

Đối với thiết bị LCD I2C với chân có chữ GND sẽ nối vào chân GND của Arduino tiếp theo là VCC sẽ nối vào chân 5V hoặc 3V, chân SDA sẽ nối vào chân SDA tương ứng của Arduino tại vị trí trên chân AREF, và chân cuối cùng là chân SCL sẽ nối vào chân SCL tương ứng trên Arduino tại vị trí đầu tiên tính từ trái qua phải ở cùng dãy với các chân digital.

Code điều khiển:

```
#include <LiquidCrystal_I2C.h>
#include <DHT.h>

// Khởi tạo LCD I2C với địa chỉ 0x27, 16 cột và 2 hàng
LiquidCrystal_I2C lcd(0x27, 16, 2);
```



```

// Khởi tạo cảm biến DHT22
#define DHTPIN 8
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);

unsigned long previousMillis = 0;
const long interval = 2000; // Đọc lại cảm biến mỗi 2 giây

void setup() {
    // Khởi động Serial
    Serial.begin(9600);

    // Khởi động LCD
    lcd.init();
    lcd.backlight();

    // Khởi động cảm biến DHT22
    dht.begin();
}

void loop() {
    unsigned long currentMillis = millis();

    if (currentMillis - previousMillis >= interval) {
        previousMillis = currentMillis;

        // Đọc nhiệt độ từ cảm biến
        float temperature = dht.readTemperature();

        // Kiểm tra xem có lỗi không
        if (isnan(temperature)) {
            Serial.println("Không thể đọc từ cảm biến DHT!");
            lcd.setCursor(0, 0);
            lcd.print("Không thể đọc");
            lcd.setCursor(0, 1);
            lcd.print("từ cảm biến");
            return;
        }

        // In nhiệt độ ra Serial Monitor
        Serial.print("Nhiệt độ: ");
        Serial.print(temperature);
        Serial.println("°C");
    }
}

```

```

// In nhiệt độ ra LCD
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Nhiệt độ:");
lcd.setCursor(0, 1);
lcd.print(temperature);
lcd.print("°C");
}
}

```

Giải thích:

Đầu tiên khai báo thư viện và biến trong đó LiquidCrystal_I2C.h: Thư viện để sử dụng LCD I2C, DHT.h: Thư viện để sử dụng cảm biến nhiệt độ DHT22

Sau đó khởi tạo LCD I2C với địa chỉ 0x27, 16 cột và 2 hàng, khởi tạo các đối tượng cho cảm biến dht22 với chân d8(DHTPIN), 2 biến previousMillis và interval: Biến để lưu thời gian trước đó và khoảng thời gian đọc cảm biến (2 giây).

Tại hàm setup, khởi động Serial để in thông tin ra Serial Monitor, khởi động LCD (lcd.init()), bật đèn nền (lcd.backlight()), khởi động cảm biến dht22(dht.begin).

Tại hàm loop, lấy thời gian hiện tại bằng millis(), kiểm tra nếu thời gian hiện tại trừ thời gian trước đó lớn hơn hoặc bằng interval (2 giây) nếu đủ thời gian, cập nhật previousMillis bằng thời gian hiện tại, tiếp theo đọc nhiệt độ từ cảm biến DHT22 bằng dht.readTemperature(), kiểm tra xem có lỗi khi đọc cảm biến không, nếu có in ra cảm biến không đọc được sau đó in ra serial và lcd.

3.3.2. Cảm biến ánh sáng

Cảm biến ánh sáng là một thiết bị quang điện chuyển đổi ánh sáng (có thể cả ánh sáng nhìn thấy và tia hồng ngoại) thành tín hiệu điện. Nó là một cảm biến thông minh có thể nhận biết được các sự biến đổi của môi trường thông qua mắt cảm biến từ đó sẽ đưa ra các điều chỉnh cho phù hợp.

Cảm biến ánh sáng hoạt động dựa trên nguyên lý của hiệu ứng quang điện. Hiệu ứng quang điện là hiện tượng một số chất đặc biệt sau khi hấp thụ ánh sáng sẽ chuyển đổi năng lượng ánh sáng thành năng lượng điện.

Hiệu ứng quang điện gồm có:

- Hiệu ứng quang điện trong: Hiện tượng quang điện trong thường diễn ra với chất bán dẫn. Khi chiếu ánh sáng vào vật liệu, năng lượng này sẽ làm thay đổi điện trở suất bên trong vật liệu gây ra suất điện động làm thay đổi tính chất điện của vật liệu.

- Hiệu ứng quang điện ngoài: Khi bề mặt của vật liệu được chiếu bởi ánh sáng, các điện tử sẽ hấp thụ năng lượng để tạo ra điện. Khi các điện tử từ bên trong vật liệu bật ra ngoài bề mặt của vật liệu sẽ tạo ra hiệu ứng quang điện ngoài.

Đặc điểm của cảm biến ánh sáng là có các tế bào phát xạ ánh sáng, các tế bào dẫn điện ánh sáng, các tế bào quang điện, thiết bị ghép nối ánh sáng.

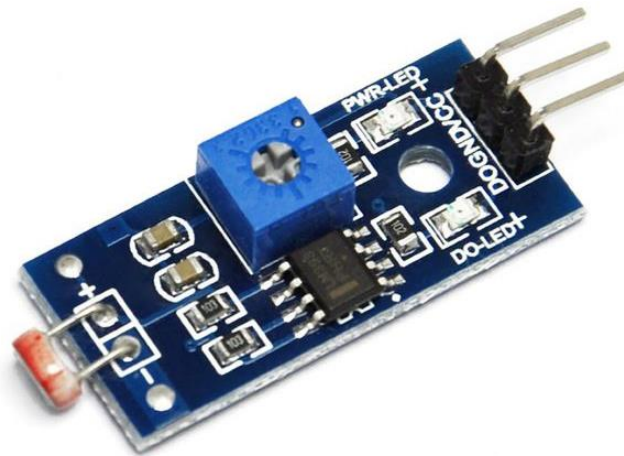
Cảm biến ánh sáng được chia làm 3 loại: Photoresistors (LDR), Photodiodes, Phototransistors.

- Cảm biến Photoresistors (LDR) được sử dụng nhiều nhất, nó là chất cảm quang hay được gọi là điện trở phụ thuộc ánh sáng (LDR). Chất cảm quang này có tác dụng kiểm tra xem đèn bật hay tắt. Và nó so sánh mức độ ánh sáng của môi trường theo tính chất tương đối trong suốt một ngày. Chất phát quang này được làm từ một vật liệu bán dẫn có điện trở cao. Chất bán dẫn này rất nhạy với ánh sáng, có thể nhìn thấy ánh sáng gần với hồng ngoại. Sự thay đổi phụ thuộc hoàn toàn vào ánh sáng môi trường. Sẽ áp dụng làm đèn sáng khi trời tối và đèn tắt khi trời sáng.

- Cảm biến Photodiodes làm thay đổi ánh sáng thành dòng điện. Nó được làm từ silicon và gecmani, thêm nhiều bộ lọc và ống kính tích hợp. Nguyên

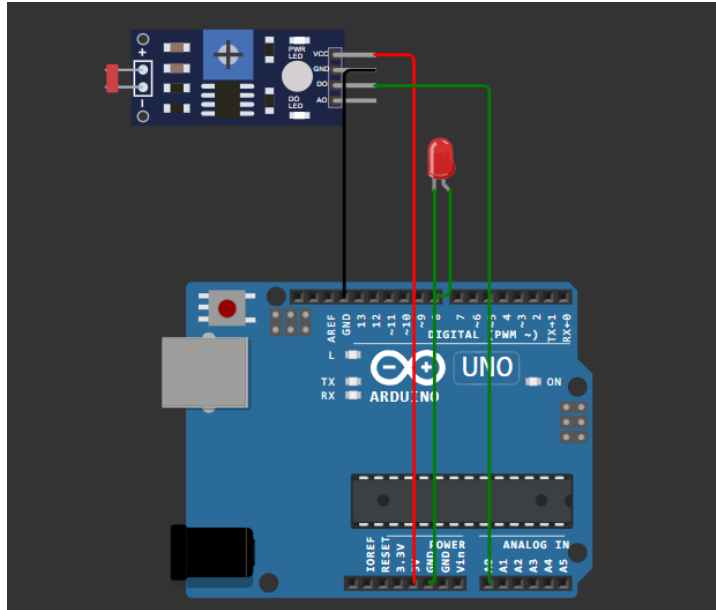
lý hoạt động của nó dựa vào hiệu ứng quang học bên trong. Khi có chùm ánh sáng chiếu vào, các electron sẽ bị nới lỏng tạo thành các lỗ cho dòng điện chạy qua. Ánh sáng càng lớn, lỗ hổng giữa các electron càng to nên dòng điện sẽ càng mạnh. Áp dụng cho điều khiển từ xa, thiết bị điện tử, thiết bị y tế, ...

- Cảm biến Phototransistors thực chất là cảm biến Photodiodes nhưng nó được khuếch đại lên nhiều lần và đồng thời độ cảm biến được tăng lên để sử dụng cho các thiết bị phù hợp.



Hình 3. 8: Cảm biến ánh sáng

Cách nối chân:



Hình 3. 9: Sơ đồ chân của cảm biến ánh sáng

Ở cảm biến ánh sáng chân đầu tiên từ trên xuống dưới là chân VCC sẽ nối với chân 5V của Arduino Uno, chân tiếp theo GND nối với chân GND của Arduino, chân DO sẽ nối với chân A0 của Arduino,

Đối với đèn led chân âm là chân ngắn sẽ nối với GND và chân dương là chân dài ở hình trên chân dương sẽ bị bẻ cong sẽ nối với chân digital.

Code điều khiển:

```
const int anhsang = A0; // Chân nối cảm biến ánh sáng với chân A0 của
Arduino
const int ledPin = 8; // Chân nối đèn LED với chân 8 của Arduino

unsigned long previousMillis = 0; // Biến lưu thời gian trước đó
const long interval = 1000; // Khoảng thời gian 1 giây (1000 milli giây)

void setup() {
  pinMode(ledPin, OUTPUT); // Đặt chân đèn LED là OUTPUT
  Serial.begin(9600); // Khởi động kết nối Serial
}

void loop() {
```

```

        unsigned long currentMillis = millis(); // Lấy thời gian hiện tại

        if (currentMillis - previousMillis >= interval) { // Kiểm tra xem đã
đủ 1 giây chưa
            previousMillis = currentMillis; // Cập nhật lại thời gian trước đó

            int cbanhsang = analogRead(anhsang); // Đọc giá trị từ cảm biến ánh
sáng

            Serial.print("Giá trị ánh sáng: ");
            Serial.println(cbanhsang); // In giá trị ánh sáng lên Serial Monitor

            // Điều khiển đèn LED dựa trên giá trị ánh sáng
            if (cbansang > 500) {
                digitalWrite(ledPin, HIGH); // Bật đèn LED
            } else {
                digitalWrite(ledPin, LOW); // Tắt đèn LED
            }
        }
    }
}

```

Giải thích:

Đầu tiên khai báo các biến `const int anhsang = A0` và `const int ledPin = 8` trong đó, A0 và 8 tương ứng với chân cảm biến và đèn led đã nối vào chân Arduino, 2 biến `previousMillis` và `interval`: Biến để lưu thời gian trước đó và khoảng thời gian đọc cảm biến.

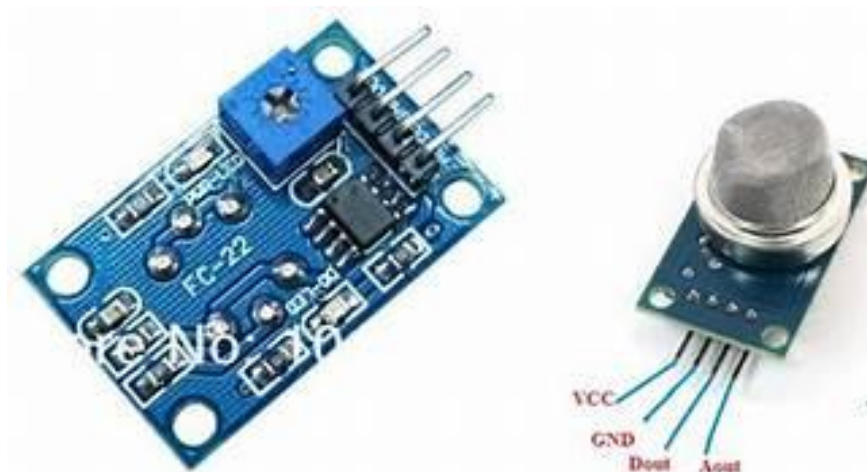
Tại hàm `setup`, khai báo `pinMode(ledPin, OUTPUT)` đặt chân đèn led ở chế độ Output(đầu ra), khởi động serial.

Tại hàm `loop`, lấy thời gian hiện tại bằng `millis()`, kiểm tra nếu thời gian hiện tại trừ thời gian trước đó lớn hơn hoặc bằng `interval` (1 giây) nếu đủ thời gian, cập nhật `previousMillis` bằng thời gian hiện tại, `int cbanhsang = analogRead(anhsang)`: Đọc giá trị từ cảm biến ánh sáng và lưu vào biến `cbansang`, sau đó in giá trị ánh sáng trên serial, kiểm tra giá trị ánh sáng nếu `cbansang > 500`: Bật đèn led bằng `digitalWrite(ledPin, HIGH)`, và nếu ngược lại đèn led sẽ tắt.

3.4. Các thiết bị cảnh báo bất thường

3.4.1 Thiết bị cảnh báo khí gas

Cảm biến khí gas là một thiết bị quan trọng được sử dụng để đo lường và giám sát khí trong môi trường dựa trên các nguyên tắc cảm biến khác nhau và sử dụng các thiết bị điện tử. Nó cho phép chúng ta xác định các thành phần và nồng độ của không khí giúp ta nắm được chất lượng không khí để dễ dàng kiểm soát.



Hình 3. 10: Cảm biến khí gas

Cảm biến này hoạt động phù thuộc vào các loại khí tương tác với vật liệu của cảm biến. Nó có thể sử dụng nhiều phương pháp khác nhau để xác định các loại khí có mặt trong không khí như : Phương pháp hấp thụ, phương pháp quang phổ, phương pháp dẫn điện.

Thế giới hiện nay có nhiều loại cảm biến khí gas khác nhau và mỗi loại sẽ dùng cho những mục đích khác nhau như :

- Cảm biến khí gas bán dẫn : sử dụng các vật liệu bán dẫn để tương tác với khí và sự thay đổi của điện trở sẽ được chuyển thành tín hiệu điện và đưa ra cảnh báo.

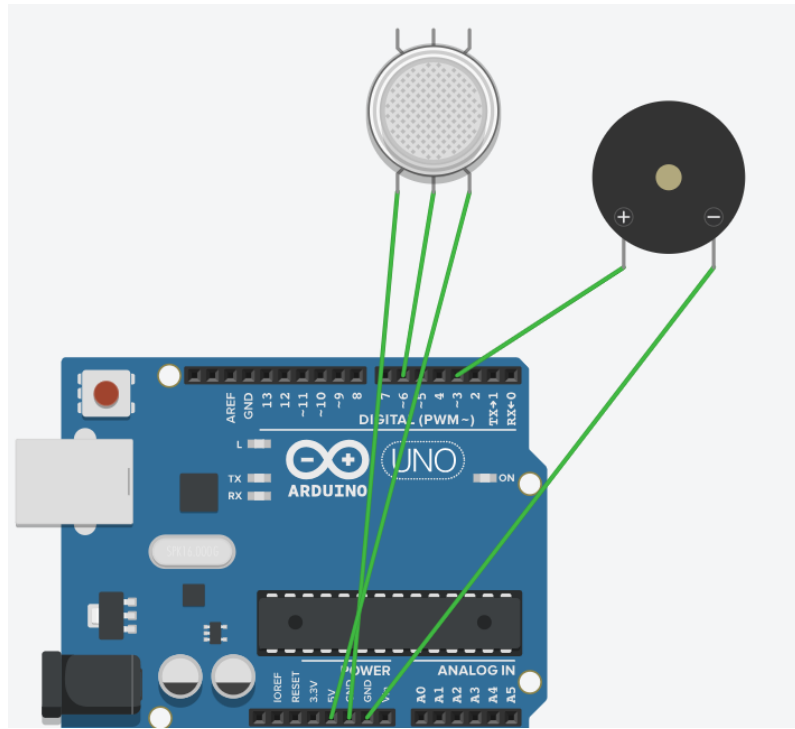
- Cảm biến khí gas điện hóa : phát hiện khí bằng cách sử dụng phản ứng điện hóa giữa khí và điện cực
- Cảm biến rò rỉ khí gas NDIR : hoạt động dựa trên nguyên tắc đo sự hấp thụ của khí gas đối với ánh sáng hồng ngoại để xác định nồng độ khí trong không khí.
- Cảm biến khí gas xúc tác : sử dụng để phát hiện và đo nồng độ khí gas trong môi trường bằng cách sử dụng phản ứng hóa học giữa khí gas và chất xúc tác để tạo ra tín hiệu điện.
- Cảm biến khí gas quang hóa : sử dụng loại cảm biến này để phát hiện và đo nồng độ các loại khí gas trong môi trường.
- Cảm biến khí gas từ tính : sử dụng từ tính để đo lường hàm lượng khí trong không khí.
- Cảm biến khí gas dẫn nhiệt : sử dụng nguyên lý dẫn nhiệt để đo lường hàm lượng khí trong không khí.

Cảm biến khí gas được sử dụng cho nhiều mô hình phổ biến như : Phát hiện ra rò rỉ khí gas ở trong nhà, Giám sát môi trường làm việc trong ngành công nghiệp, đo lường chất lượng không khí trong khu vực công cộng, ứng dụng vào y tế để đo nồng độ cồn, kiểm tra chất lượng không khí trong phòng mổ, ứng dụng phòng cháy chữa cháy để không rơi vào các thảm họa nghiêm trọng.

Cách nối dây:

Trong cảm biến khí gas sẽ có 3 hoặc 4 đầu tùy từng loại cảm biến, đầu đầu tiên sẽ nối với chân GND, đầu giữa sẽ là chân số và đầu cuối sẽ là chân von.

Đối với nối còi thì đơn giản hơn đầu âm sẽ nối với chân GND và đầu dương sẽ nối với chân số 3.



Hình 3. 11: Cách nối dây của cảm biến khí gas

Code điều khiển:

```
// Khai báo chân cảm biến khí gas và con
int gas_pin = 6;
int buzzer_pin = 3;

// Khai báo biến lưu giá trị đọc được từ cảm biến
int gia_tri_gas;

// Ngưỡng cảnh báo (có thể điều chỉnh)
int ngưỡng = 400;

// Biến lưu thời gian gần đây nhất
unsigned long thời_gian_gần_dây;

// Thời gian kiểm tra (1 giây)
const int thời_gian_kiểm_tra = 1000;

void setup() {
    // Cấu hình chân cảm biến và con
    pinMode(gas_pin, INPUT);
    pinMode(buzzer_pin, OUTPUT);
}
```

```

// Khoi dong Serial Monitor
Serial.begin(9600);

// Lay thoi gian gan day nhat
thoi_gian_gan_day = millis();
}

void loop() {
    // Kiem tra thoi gian da qua
    if (millis() - thoi_gian_gan_day >= thoi_gian_kiem_tra) {
        // Cap nhat thoi gian gan day nhat
        thoi_gian_gan_day = millis();

        // Doc gia tri tu cam bien khi gas
        gia_tri_gas = analogRead(gas_pin);

        // In gia tri doc duoc ra Serial Monitor
        Serial.print("Gia tri gas: ");
        Serial.println(gia_tri_gas);

        // Kiem tra neu gia tri vuot nguong
        if (gia_tri_gas > nguong) {
            // Bat con
            digitalWrite(buzzer_pin, HIGH);
            Serial.println("Phat hien rac roi khi gas! Con dang bat.");
        } else {
            // Tat con
            digitalWrite(buzzer_pin, LOW);
            Serial.println("Muc khi gas binh thuong.");
        }
    }
}

```

Giải thích:

Đầu tiên khai báo các biến gồm gas, còi, giá trị gas để lưu giá trị từ cảm biến và thời gian gần đây để lưu giá trị thời gian, thời gian kiểm tra để kiểm tra thời gian, ngưỡng để biết mức độ nào thì cảnh báo.

Trong hàm void loop tạo if với millis để cập nhật lại thời gian sau 1 khoảng thời gian cảm biến chạy, sau đó gán gia_tri_gas từ bằng với dữ liệu đọc được từ gas_pin sau đó check điều kiện nếu gia_tri_gas vượt mức cho phép ở đây là vượt qua giá trị của ngưỡng thì còi kêu nếu không thì tắt.

3.4.2 Thiết bị cảnh báo có chuyển động (chống trộm)

Cảm biến HC-SR501 là một cảm biến có thể phát hiện chuyển động của con người (hoặc động vật). Nó được sử dụng rộng rãi để phát hiện sự hiện diện của con người trong nhiều ứng dụng (tự động bật tắt đèn, đóng mở cửa, kích hoạt hoặc tắt thang cuốn, phát hiện trộm).

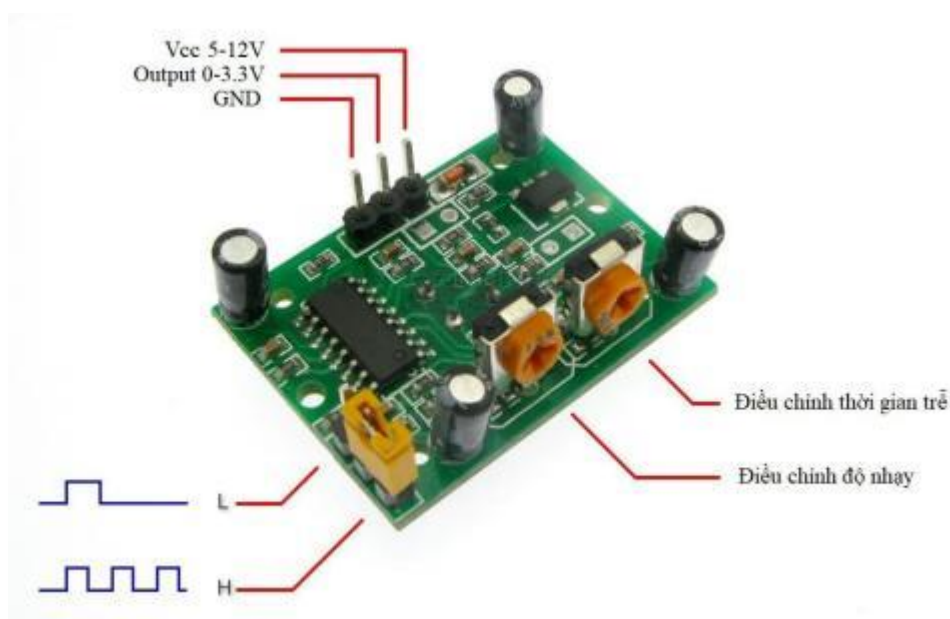
Cấu tạo

Cảm biến chuyển động HC-SR501 có 3 chân:

Chân GND: Chân nối đất

Chân Vcc: Chân nguồn 5V

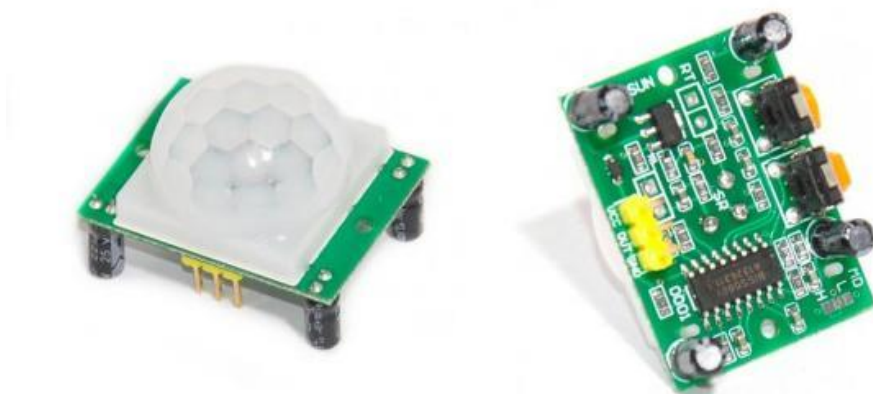
Chân Output: Chân ngõ ra, chân này ở mức thấp khi không phát hiện chuyển động và ở mức cao khi phát hiện chuyển động. Chân này được kết nối với chân ngõ vào của Arduino.



Hình 3. 12: Sơ đồ chân của cảm biến chuyển động

Trên đỉnh của board cảm biến, có một loại ống kính đặc biệt gọi là lăng kính Fresnel che phủ cảm biến nhiệt điện (pyroelectric sensor). Nhiệm vụ của

lăng kính Fresnel là tập trung tất cả các bức xạ hồng ngoại vào cảm biến nhiệt điện.



Hình 3. 13: Cảm biến chuyển động

Nếu nhìn vào mặt sau của board cảm biến PIR, bạn sẽ thấy toàn bộ mạch được đặt ở đây. Bộ não của mô-đun cảm biến PIR là IC phát hiện chuyển động [BISS0001](#). Gần IC này, chúng ta có hai chiết áp, một để điều chỉnh độ nhạy của cảm biến để giới hạn khoảng cách bắt xa gần và cái còn lại để điều chỉnh thời gian tín hiệu đầu ra duy trì ở mức cao khi phát hiện đối tượng. Thời gian này có thể được điều chỉnh từ 0,3 giây lên đến 5 phút.

Thông số kỹ thuật

Điện áp hoạt động: 5V ~ 12V DC (khuyên dùng: 5V)

Dòng điện tiêu thụ: 65mA

Điện áp đầu ra: mức cao 3,3V, mức thấp 0V

Thời gian trễ: Điều chỉnh (0,3 giây ~ 18 giây)

Phạm vi cảm ứng: góc quét < 100° và xa 7m

Nhiệt độ hoạt động: -20°C ~ +80°C

Kích thước mạch: 32 * 24mm, lỗ vít 28mm, đường kính vít 2 mm, nắp cảm ứng (đường kính): 23mm

Chế độ hoạt động:

+ (H) Lặp lại: Khi phát hiện một chuyển động ngõ ra sẽ ở mức cao (3.3V), sau một khoảng thời gian (được chỉnh bằng biến trở) ngõ ra sẽ xuống mức thấp (0V). Ở chế độ này, khi ngõ ra lên mức cao, dù người đó đã đi khỏi hay chưa thì sau một thời gian mới xuống lại mức thấp.

+ (L) Không lặp lại: Khi phát hiện một chuyển động ngõ ra sẽ ở mức cao (3.3V) và giữ nguyên trạng thái. Khi người đó đi ra khỏi phạm vi hoạt động của cảm biến thì sau một khoảng thời gian (được chỉnh bằng biến trở) ngõ ra xuống mức thấp (0V).

Hoạt động của cảm biến PIR

Cảm biến HC-SR501 phát hiện chuyển động dựa trên sự thay đổi bức xạ hồng ngoại từ vật thể chuyển động. Để được cảm biến HC-SR501 phát hiện, đối tượng phải đáp ứng hai điều kiện:

Đang chuyển động hoặc đang lay động

Đang phát ra tia hồng ngoại

Nếu một vật thể đang di chuyển nhưng KHÔNG phát ra tia hồng ngoại (ví dụ: robot hoặc xe đồ chơi), nó sẽ không được cảm biến phát hiện.

Nếu một đối tượng đang phát ra tia hồng ngoại nhưng KHÔNG di chuyển (ví dụ: một người đứng yên mà không di chuyển), nó sẽ không được cảm biến phát hiện.

Con người và động vật phát ra tia hồng ngoại một cách tự nhiên. Do đó, cảm biến có thể phát hiện chuyển động của con người và động vật.

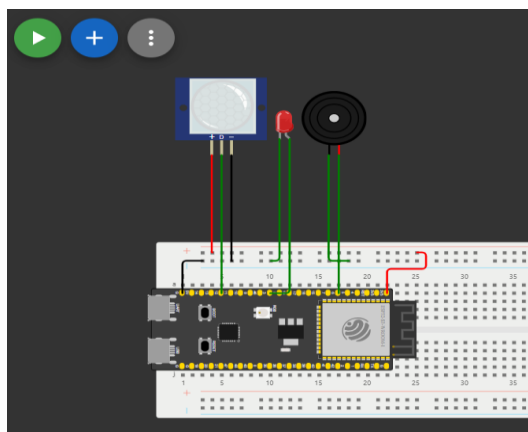
Chúng ta dựa vào trạng thái chân OUTPUT của cảm biến để xác định cảm biến có phát hiện đối tượng hay không.

Khi KHÔNG có người (hoặc động vật) di chuyển trong phạm vi được phát hiện của cảm biến, chân OUTPUT của cảm biến ở mức THẤP.

Khi có con người (hoặc động vật) di chuyển vào phạm vi được phát hiện của cảm biến, chân OUTPUT của cảm biến thay đổi từ THẤP lên CAO (phát hiện chuyển động).

Khi có con người (hoặc động vật) đi ra khỏi phạm vi được phát hiện của cảm biến, chân OUTPUT của cảm biến thay đổi từ CAO sang THẤP (không phát hiện chuyển động).

Cách nối dây:



Hình 3. 14: Cách nối dây của cảm biến chuyển động

Trong cảm biến chuyển động chân đầu tiên có dấu cộng sẽ được nối với phần 5V, chân ở giữa sẽ là chân số nối với các chân có số trên ESP32 và chân cuối là chân có dấu âm sẽ luôn nối vào chân GND.

Đối với chân của đèn led chân ngắn hơn sẽ là chân âm nối với GND ở ESP32 và chân dài hơn sẽ là chân dương nối với chân số. Và tương tự ở còi cũng như vậy.

Tại ESP32 vì có quá nhiều thiết bị cùng cần chân GND chính vì thế việc nối 1 chân từ GND của ESP32 sang một mạch âm sẽ giải quyết được vấn đề vì các mạch âm có dấu trừ trên board sẽ thông với nhau trên một hàng ngang đó, tương tự như thế mạch dương có dấu cộng cũng như vậy.

Code điều khiển:

```

#define BLYNK_TEMPLATE_ID "TMPL6wiYcI_JM"
#define BLYNK_TEMPLATE_NAME "Pirr"
#define BLYNK_AUTH_TOKEN "YRW8bWZ5j_1cW_RoJDENm-8inJeRAsE0"

/* Comment this out to disable prints and save space */
#define BLYNK_PRINT Serial

#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>

// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "NAM";
char pass[] = "12345678";

int denled = 12; // Đèn LED
int coi = 6; // Còi báo động
int motionsensor = 34; // Cảm biến chuyển động
int pinTrangThaiHienTai = LOW; // Trạng thái hiện tại của chân
int pinTrangThaiTruoc = LOW; // Trạng thái trước đó của chân
int blynkTrangThai = 0; // 0 = Cảm biến chuyển động tắt, 1 = Cảm biến
chuyển động bật

BlynkTimer timer;

// Hàm này được gọi mỗi khi thiết bị được kết nối tới Blynk.Cloud
BLYNK_CONNECTED()
{
  Blynk.syncVirtual(V0); // sẽ gây ra BLYNK_WRITE(V0) được thực thi
}

BLYNK_WRITE(V0)
{
  {
    int gia_tri = param.asInt(); // gán giá trị
    if (gia_tri == 1) {
      blynkTrangThai = 1;
      Serial.println("Cảm biến chuyển động đã bật");
    }
    else {
      blynkTrangThai = 0;
      Serial.println("Cảm biến chuyển động đã tắt");
    }
  }
}

```

```

void setup()
{
  // Debug console
  Serial.begin(115200); // khởi động serial

  Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);
  // Bạn cũng có thể chỉ định máy chủ:
  //Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass, "blynk.cloud", 80);
  //Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass, IPAddress(192,168,1,100),
8080);

  pinMode(motionsensor, INPUT); // Thiết lập chân Arduino là chế độ đầu
vào để đọc giá trị từ chân đầu ra của cảm biến
  pinMode(denled, OUTPUT); // Khởi tạo chân số 16 GPIO là một chân đầu ra
  pinMode(coi, OUTPUT);
  ledcSetup(0, 1000, 8);
  ledcAttachPin(coi, 0);
}

void loop()
{
  Blynk.run();
  timer.run();
  pinTrangThaiTruoc = pinTrangThaiHienTai;
  pinTrangThaiHienTai = digitalRead(motionsensor); // đọc trạng thái mới

  if (blynkTrangThai == 1) { // Cảm biến chuyển động được bật
    if (pinTrangThaiTruoc == LOW && pinTrangThaiHienTai == HIGH) { //
Chuyển động đã dừng
      Serial.println("phát hiện chuyển động ");
      Blynk.virtualWrite(V1, " phát hiện chuyển động ");
      digitalWrite(denled, HIGH); // tắt relay
      ledcWriteNote(0, NOTE_C, 4);
    }
    else if (pinTrangThaiTruoc == HIGH && pinTrangThaiHienTai == LOW)
{ // Trạng thái chân thay đổi: HIGH -> LOW
      Serial.println("Chuyển động đã dừng");
      Blynk.virtualWrite(V1, "Chuyển động đã dừng");
      digitalWrite(denled, LOW);
      ledcWrite(0, 0);
    }
  }
  else { // Cảm biến chuyển động đã tắt
    digitalWrite(denled, LOW); // đảm bảo rằng relay đã tắt
    ledcWrite(0, 0);
  }
}

```



```
}  
  
pinTrangThaiTruoc = pinTrangThaiHienTai; // cập nhật trạng thái trước đó  
}
```

Giải thích:

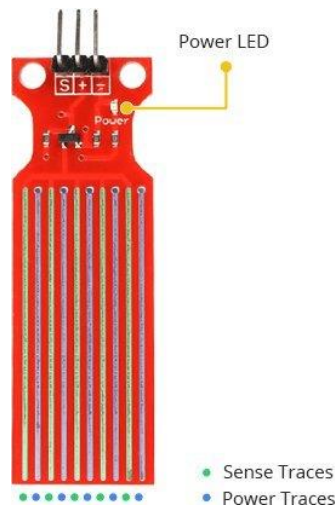
Đầu tiên khai báo các define của Blynk để nhận biết kết nối với app Blynk , sau đó thiết lập wifi tự tạo và tạo các biến ứng với các chân , ngoài ra ở đây tạo ra 2 biến trangthaihientai và trangthaitruoc để nhận biết các trạng thái của cảm biến và blynktrangthai để nhận biết trạng thái của blynk.

Sau đó check trạng thái của blynk tại pin ảo là 0 nếu trạng thái blynk là 1 thì cảm biến chuyển động được bật còn nếu không thì cảm biến chuyển động sẽ tắt, sau đó khai báo các pin, còi blynk trong void setup,

Tại void loop gán giá trị trangthaitruoc bằng trangthai hiện tại và đọc giá trị của trạng thái hiện tại nếu blynk bằng 1 tức là bật cảm biến và trạng thái trước thấp còn trạng thái hiện tại cao thì bật đèn và còi ngược lại nếu không thì tắt và cập nhật lại giá trị trạng thái.

3.4.3 Thiết bị cảnh báo về mực nước rò rỉ

Cảm biến mực nước chất lỏng là loại cảm biến dùng để đo mức nước hoặc chất lỏng trong bồn, hồ chứa, bể bơi hoặc trong các hệ thống khác. Cảm biến này hoạt động dựa trên nguyên lý đổi kháng (resistive sensing).



Hình 3. 15: Cảm biến mực nước

Cảm biến mực nước thường bao gồm các thanh kim loại song song, được đặt trong chất lỏng cần đo mức nước. Khi mức nước tăng lên và chạm vào thanh kim loại, dòng điện sẽ chạy qua một đường dẫn điện trở giữa hai thanh kim loại. Điện trở này thay đổi tùy thuộc vào mức độ tiếp xúc của chất lỏng với thanh kim loại. Cuối cùng, giá trị điện trở này sẽ được chuyển đổi sang tín hiệu điện tử, thông qua bộ chuyển đổi tương tự (ADC), để đo được mức nước.

Tùy thuộc vào mức độ tiếp xúc của chất lỏng, giá trị điện trở sẽ thay đổi, giúp cho việc đo mức nước trở nên dễ dàng và chính xác.

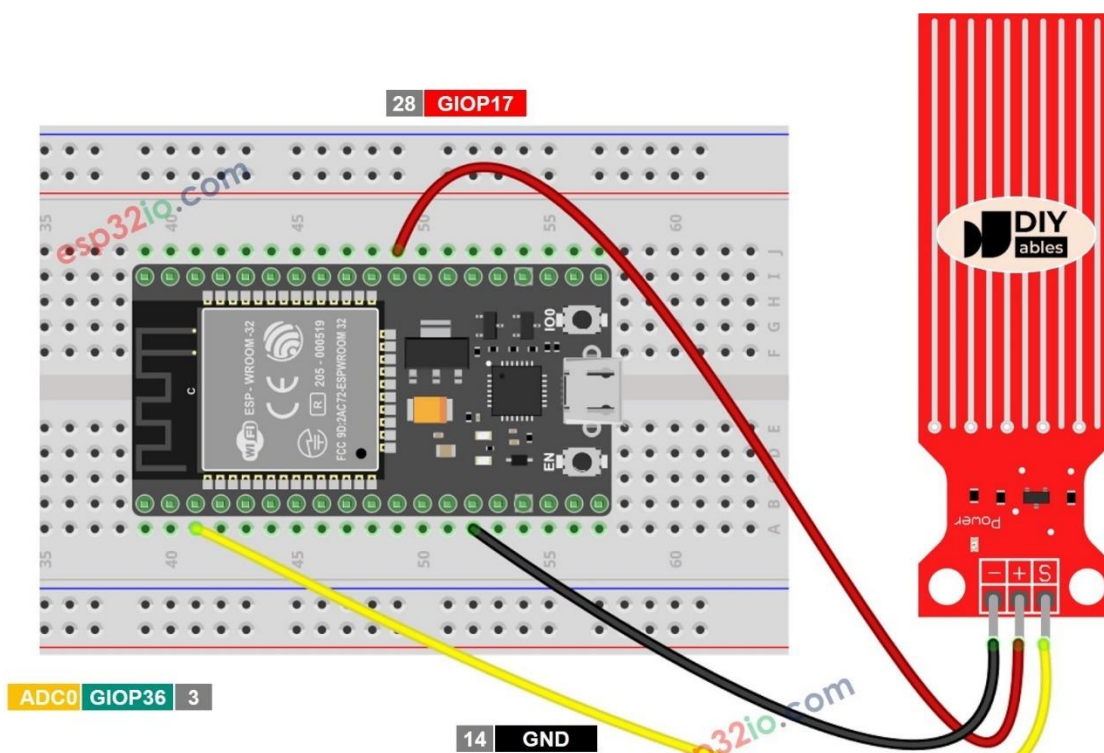
Cấu tạo của cảm biến mực nước thường bao gồm các thành phần chính như cảm biến áp lực, vỏ bảo vệ, mạch điện tử và phụ kiện lắp đặt. Cảm biến áp lực là bộ phận then chốt, dùng để đo áp lực thủy tĩnh tác động lên thiết bị khi nó được nhúng vào nước. Các công nghệ cảm biến phổ biến bao gồm cảm biến silicon và cảm biến dung nạp.

Vỏ bảo vệ giúp cách ly cảm biến khỏi ảnh hưởng của môi trường bên ngoài như nước, bụi hoặc nhiệt độ. Mạch điện tử xử lý tín hiệu từ cảm biến áp lực, chuyển đổi sang tín hiệu điện hoặc số, và truyền đến bộ xử lý hoặc thiết bị hiển

thị. Các phụ kiện lắp đặt như cáp, đầu nối, giá đỡ giúp người dùng dễ dàng lắp đặt và kết nối cảm biến.

Khi cảm biến được nhúng vào nước, áp lực thủy tĩnh của cột nước sẽ tác động lên cảm biến áp lực. Cảm biến áp lực sẽ chuyển đổi áp lực này thành tín hiệu điện tương ứng, thường là điện áp hoặc dòng điện. Mạch điện tử sẽ xử lý tín hiệu này, chuyển đổi sang tín hiệu số hoặc tương tự, và truyền đến bộ xử lý hoặc hiển thị. Từ tín hiệu này, hệ thống có thể tính toán và hiển thị mực nước chính xác.

Cách nối dây:



Hình 3. 16: Cách nối dây của cảm biến mực nước

Chân có chữ S sẽ nối vào chân analog là các chân có chữ từ A0 đến A5 sau đó chân có dấu cộng sẽ nối vào chân 5V hoặc 3.3V của esp32 và chân cuối là chân có dấu trừ sẽ nối vào chân GND của esp32.

Code điều khiển:

```
#define BLYNK_TEMPLATE_ID "TMPL6wiYcI_JM"
```

```

#define BLYNK_TEMPLATE_NAME "Pirr"
#define BLYNK_AUTH_TOKEN "YRW8bWZ5j_1cW_RoJDENm-8inJeRASe0"

/* Comment this out to disable prints and save space */
#define BLYNK_PRINT Serial

#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>

// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "NAM";
char pass[] = "12345678";

// Khai báo chân kết nối
const int analogInPin = 32; // Chân cảm biến
int sensorValue = 0;
int coinuoc = 12; // Chân còi
int dennuoc = 14; // Chân LED
int blynkStatus = 0;
// Thiết lập ngưỡng cảm biến mực nước
int lowThreshold = 2000;
int highThreshold = 4000;

// Biến theo dõi thời gian
unsigned long previousMillis = 0;
const long interval = 100; // 100ms

BLYNK_CONNECTED()
{
  Blynk.syncVirtual(V3); // will cause BLYNK_WRITE(V0) to be executed
}

BLYNK_WRITE(V3)
{
  int value = param.asInt(); // assigning the value
  if (value == 1) {
    blynkStatus = 1;
    Serial.println("Motion sensor turned on");
  }
  else {
    blynkStatus = 0;
    Serial.println("Motion sensor turned off");
  }
}

```

```

}

void setup() {
    Serial.begin(115200); // initialize serial

    Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);
    // You can also specify server:
    //Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass, "blynk.cloud", 80);
    //Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass,
IPAddress(192,168,1,100), 8080);

    pinMode(analogInPin, INPUT); // set arduino pin to input mode to
read value from OUTPUT pin of sensor
    pinMode(dennuoc, OUTPUT); // Initialise digital pin GPIO 16 as an
output pin
    pinMode(coinuoc, OUTPUT);
    ledcSetup(0, 1000, 8);
    ledcAttachPin(coinuoc, 0);
}

void loop() {
    Blynk.run(); // Chạy Blynk

    unsigned long currentMillis = millis();
    if (currentMillis - previousMillis >= interval) {
        previousMillis = currentMillis;

        sensorValue = analogRead(analogInPin);

        Serial.print("sensor = ");
        Serial.println(sensorValue);

        if (blynkStatus == 1) {
            if (sensorValue >= lowThreshold && sensorValue <=
highThreshold) {
                Serial.println("Motion stopped!");
                Blynk.virtualWrite(V2, "Motion Stopped");
                digitalWrite(dennuoc, HIGH); // turn off the relay
                ledcWriteNote(0, NOTE_C, 4);
            }
            else if (sensorValue >= highThreshold) {
                Blynk.virtualWrite(V2, "Motion Detected");
                digitalWrite(dennuoc, HIGH); // turn on the relay
                ledcWriteNote(0, NOTE_C, 6);
            }
        }
    }
}

```

```

        else {
            digitalWrite(dennuoc, HIGH); // ensure relay is turned
off
            ledcWrite(0, 0);
        }
    }
    else { // Motion sensor is turned off
        digitalWrite(dennuoc, LOW); // ensure relay is turned off
        ledcWrite(0, 0);
    }
}
}
}

```

Giải thích:

Đoạn mã Arduino được cung cấp sử dụng nền tảng IoT Blynk để tạo ra một hệ thống giám sát và cảnh báo mức nước. Mã bắt đầu bằng cách định nghĩa một số hằng số, bao gồm ID mẫu Blynk, tên và mã xác thực, cũng như thiết lập thông tin đăng nhập WiFi. Sau đó, nó khai báo các chân được sử dụng cho cảm biến mức nước, loa báo động và đèn LED.

Hàm setup khởi tạo giao tiếp serial, kết nối với máy chủ Blynk và đặt chế độ đầu vào/đầu ra cho các chân khác nhau. Hàm loop chạy liên tục, kiểm tra kết nối Blynk và đọc giá trị cảm biến mức nước mỗi 100 mili giây.

Khi cảm biến chuyển động được bật (thông qua ứng dụng Blynk), mã sẽ kiểm tra giá trị cảm biến so với ngưỡng mức nước thấp và cao. Nếu giá trị nằm trong phạm vi chấp nhận được, nó sẽ gửi thông báo "Chuyển động ngừng" đến ứng dụng Blynk, tắt đèn LED và vô hiệu hóa loa báo động. Nếu giá trị vượt quá ngưỡng cao, nó sẽ gửi thông báo "Chuyển động được phát hiện", bật đèn LED và kích hoạt loa báo động. Nếu giá trị cảm biến dưới ngưỡng thấp, nó sẽ chỉ tắt đèn LED và loa báo động. Khi cảm biến chuyển động bị tắt, đèn LED và loa báo động sẽ tắt bất kể giá trị cảm biến.

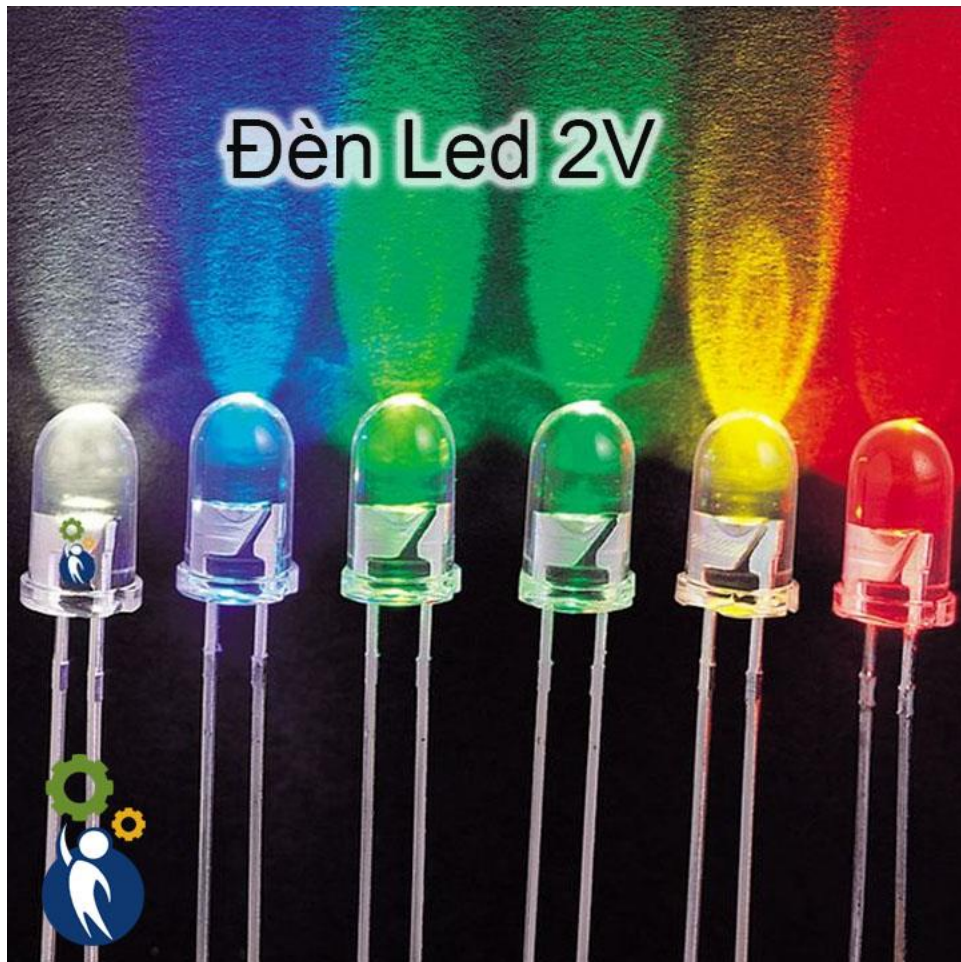
3.5. Thiết bị cảnh báo

3.5.1. Đèn Led

Để xây dựng và thiết kế một mô hình nhà thông minh hoàn chỉnh, chúng ta không thể thiếu thiết bị đèn led trong ngôi nhà của mình được. Đèn có rất nhiều ứng dụng như chúng ta có thể sử dụng nó để chiếu sáng hay cảnh báo. Như trong đề tài nhà thông minh này chúng tôi sử dụng đèn led 2V để có thể tiết kiệm điện tích và làm sản phẩm để demo nên cần nhỏ gọn nhất có thể.

Điện áp định mức là mức nguồn điện đảm bảo cho đèn được hoạt động một cách bình thường. Điện áp định mức ở mỗi thiết bị đèn LED là không giống nhau. Sau khi dòng điện được cung cấp cho chip LED đúng theo yêu cầu của nhà sản xuất, hiệu quả chiếu sáng của đèn LED sẽ đạt mức tốt nhất. Khi dòng điện của đèn LED vượt quá hoặc thấp hơn điện áp định mức sẽ khiến đèn không hoạt động hoặc nhanh bị hỏng. Điện áp cung cấp cho đèn không đạt điện áp định mức sẽ ảnh hưởng tới cường độ ánh sáng của đèn. Vì vậy, chúng ta cần để ý các thông số kỹ thuật mà nhà sản xuất đã ghi rõ giúp ta chọn loại led phù hợp với nhu cầu của chúng ta tránh bị hư hỏng, lãng phí.

Mức điện áp dùng cho led có thể bằng hoặc lớn hơn điện áp định mức. Nếu điện áp bằng điện áp định mức thì led sẽ được hoạt động ở trạng thái tốt nhất, ngược lại nếu cao hơn điện áp định mức thì tuổi thọ, chất lượng của led sẽ bị giảm nhanh chóng.



Hình 3. 17: đèn led

LED cũng bị ảnh hưởng bởi nhiều điện áp khác nhau : Cường độ dòng điện của LED, nguồn điện cấp cho đèn LED, thông số LED đơn, công suất của chip LED, thông số LED đỏ, cực của đèn LED, điện áp đèn LED sẽ bị ảnh hưởng bởi các yếu tố ngoài như : thời tiết, nhiệt độ, độ ẩm, ...

Chính vì vậy chúng ta phải sử dụng loại đèn LED phù hợp với nhu cầu cần sử dụng để tránh tốn công, lãng phí.

3.5.2. Còi chip 5v

Là linh kiện được sử dụng trong các mạch điện tử nhằm tạo ra âm thanh để làm gì đó. Như trong mô hình nhà thông minh của chúng tôi, chúng tôi sử dụng còi để tạo ra âm thanh cảnh báo cháy khi cảm biến khí gas đo được có một lượng khí có thể khiến

cho căn nhà bốc cháy để chủ căn nhà có thể phát hiện, thoát thân kịp thời tránh thiệt hại về tính mạng và tài sản nhất có thể. Còi có kích thước nhỏ, nhẹ giúp dễ dàng lắp đặt, triển khai sử dụng. Cũng như các thiết bị khác, tùy vào từng trường hợp và mục đích sử dụng để chúng ta lựa chọn loại còi phù hợp. Đối với mô hình nhà thông minh của chúng tôi ưu tiên sự nhỏ gọn, không cần loại còi có công suất lớn nên còi chip 5V là lựa chọn phù hợp đối với chúng tôi.

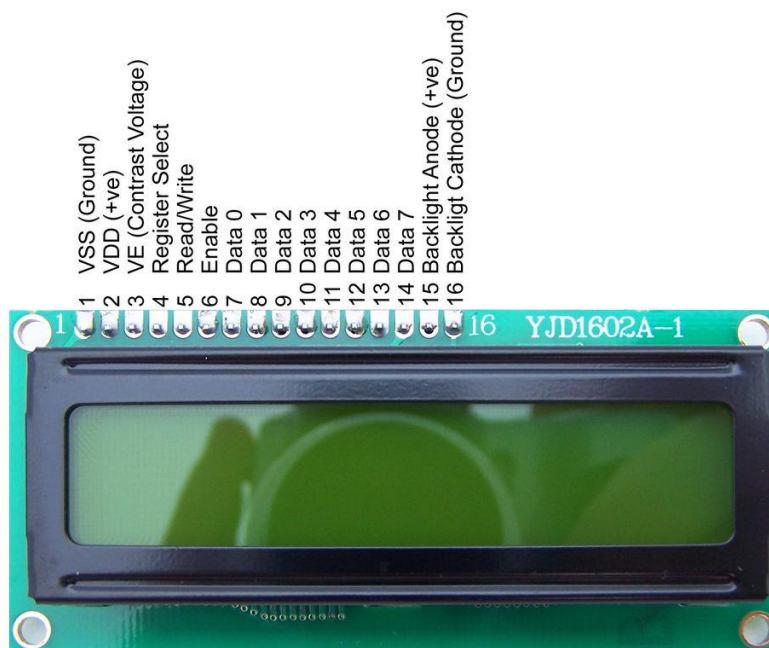
- Thông số kỹ thuật :
 - + Điện áp đầu vào : 3.5 -5VDC
 - + Dòng điện tiêu thụ $I_L < 25\text{mA}$
 - + Tần số âm thanh : 2300Hz +500Hz (-500Hz)
 - + Âm thanh đầu ra : Tít tít
 - + Biên độ âm thanh : $>80\text{ dB}$
 - + Hoạt động trong môi trường có nhiệt độ : -20 độ C đến 70 độ C
 - + Kích thước : Đường kính 12mm, cao 9,7mm
 - + Màu sắc : Đen
 - + 2 cực : Cực âm (chân ngắn) và cực dương (chân dài)



Hình 3. 18: Còi

3.6. Thiết bị điện

3.6.1. Lcd I2c



Hình 3. 19: Màn hình Lcd

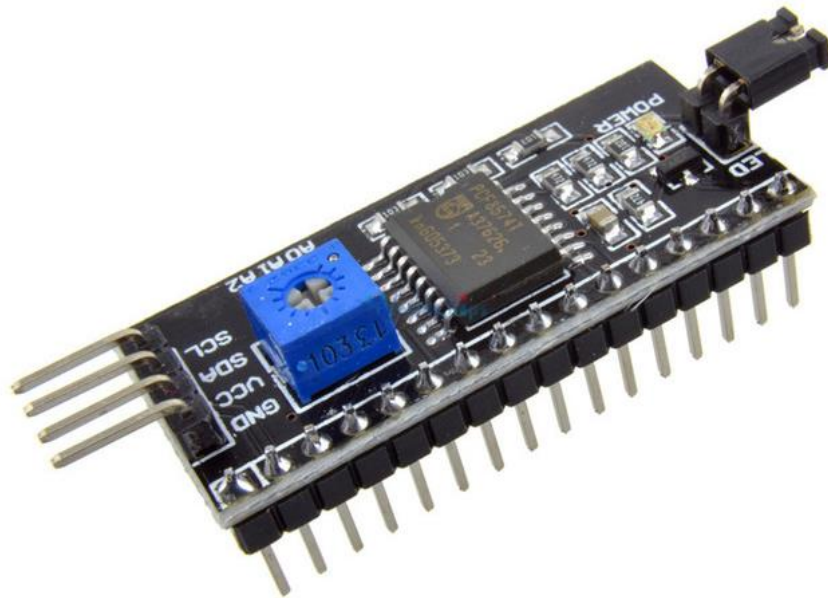
Chúng ta sử dụng màn hình LCD này để hiển thị các thông tin cần thiết ra màn hình, nó giúp cho những người không biết về code, về những thứ không thể hiện rõ ràng trên các thiết bị khác nhưng có thể hiện trên màn hình LCD. Ví dụ, nếu bạn muốn biết nhiệt độ phòng hiện tại của bạn là bao nhiêu, bạn không thể nhìn vào cảm biến để có thể thấy được mà phải nhìn vào màn hình LCD để biết, ...

Thông số kỹ thuật của LCD :

- + LCD 16×2 có 16 chân trong đó 8 chân dữ liệu (D0 – D7) và 3 chân điều khiển (RS, RW, EN).
- + 5 chân còn lại dùng để cấp nguồn và đèn nền cho LCD 16×2.
- + Các chân điều khiển giúp ta dễ dàng cấu hình LCD ở chế độ lệnh hoặc chế độ dữ liệu.

+ Chúng còn giúp ta cấu hình ở chế độ đọc hoặc ghi
LCD 16×2 có thể sử dụng ở chế độ 4 bit hoặc 8 bit tùy theo ứng dụng ta đang làm.

Module I2C Arduino



Hình 3. 20: I2C

Module I2C ra đời giải quyết cho vấn đề khó khăn trong việc kết nối và chiếm dụng nhiều chân trên vi điều khiển. Thay vì phải mất 6 chân vi điều khiển để kết nối với LCD 16×2 (RS, EN, D7, D6, D5 và D4) thì module IC2 bạn chỉ cần tốn 2 chân (SCL, SDA) để kết nối.

- Thông số kỹ thuật của I2C:
 - + Điện áp hoạt động: 2.5-6V DC.
 - + Hỗ trợ màn hình: LCD1602,1604,2004 (driver HD44780).
 - + Giao tiếp: I2C.
 - + Địa chỉ mặc định: 0X27 (có thể điều chỉnh bằng ngắn mạch chân A0/A1/A2).
 - + Tích hợp Jump chót để cung cấp đèn cho LCD hoặc ngắt.

+ Tích hợp biến trở xoay điều chỉnh độ tương phản cho LCD.

- Các lỗi thường gặp khi sử dụng I2C LCD : Hiển thị một dãy ô vuông, màn hình chỉ in ra một ký tự đầu, màn hình nhấp nháy. Các lỗi này xảy ra chủ yếu là do sai địa chỉ bus, để sửa lỗi thì thay địa chỉ mặc định là « 0x27 » thành « 0x3F ».

- Giao tiếp I2C LCD Arduino :

-

Module I2C LCD 16×2	Arduino UNO
GND	GND
VCC	5V
SDA	A4/SDA
SCL	A5/SCL

bảng 3.2: giao tiếp giữa I2C LCD với Arduino

3.6.2. Quạt

Quạt là một trong những thành phần phổ biến và quan trọng trong các dự án sử dụng hệ thống vi điều khiển như Arduino. Chúng có nhiều ứng dụng thiết thực, từ việc làm mát các linh kiện điện tử nhạy cảm đến việc tạo ra luồng không khí để điều chỉnh nhiệt độ môi trường.

Khi làm việc với quạt trong các dự án Arduino, cách thức kết nối và điều khiển chúng là điều cần quan tâm hàng đầu. Thông thường, quạt được kết nối với các chân PWM (Pulse Width Modulation) của Arduino để có thể điều chỉnh tốc độ quạt một cách linh hoạt. Bằng cách sử dụng các hàm như `analogWrite()`, bạn có thể thay đổi chu kỳ điều khiển (duty cycle) của tín hiệu PWM, từ đó kiểm soát được tốc độ quay của quạt. Một số loại quạt còn tích hợp các cảm biến tốc độ,

thường là cảm biến hiệu ứng Hall. Trong trường hợp này, bạn có thể sử dụng các chân digital I/O của Arduino để đọc tín hiệu từ cảm biến và theo dõi tốc độ quạt. Điều này cho phép bạn thực hiện các ứng dụng nâng cao hơn, chẳng hạn như điều khiển nhiệt độ thông minh bằng cách tăng hoặc giảm tốc độ quạt.

Ngoài cách kết nối và điều khiển cơ bản, Arduino cũng cung cấp các thư viện hữu ích để làm việc với quạt. Chẳng hạn, thư viện Servo.h có thể được sử dụng để điều khiển tốc độ quạt, trong khi các thư viện bên thứ ba như FanController.h cung cấp các tính năng nâng cao hơn, như đọc tốc độ quạt, đặt ngưỡng nhiệt độ, và điều khiển nhiều quạt cùng lúc.

Bằng cách kết hợp các kỹ thuật về kết nối, điều khiển tốc độ, và phát hiện tốc độ quạt, cùng với các thư viện hỗ trợ, bạn có thể tích hợp quạt vào các dự án Arduino một cách linh hoạt và hiệu quả. Quạt không chỉ là một thành phần quan trọng để làm mát các linh kiện mà còn mang lại nhiều ứng dụng thú vị khác, như tạo hiệu ứng gió, điều chỉnh nhiệt độ môi trường, và nhiều ứng dụng sáng tạo khác.

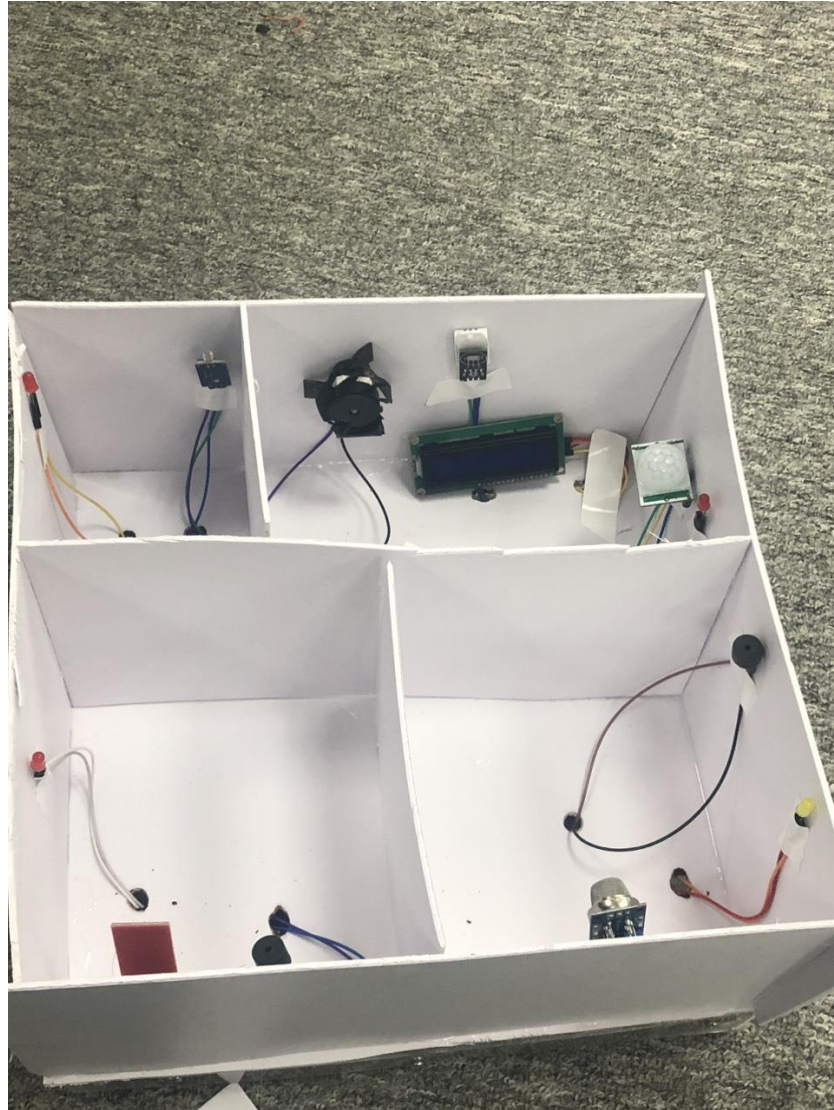


Hình 3. 21: Quạt

Tiểu kết: Trong chương này tìm hiểu về tổng quan của các thiết bị có trong mô hình nhà thông minh, có trong đề tài của em qua đó chương này cung cấp các khái niệm giới thiệu về các chân và cách nối và code điều khiển của các thiết bị.

CHƯƠNG 4: THIẾT KẾ MÔ HÌNH, VẬN HÀNH, ỨNG DỤNG VÀ PHÁT TRIỂN

4.1. Thiết kế mô hình



Hình 4. 1: Mô hình nhà thông minh

Mô hình nhà thông minh sẽ bao gồm 1 phòng khách, 1 phòng bếp, 1 phòng ngủ, 1 phòng tắm. Các phòng sẽ được lắp thiết bị thông minh để tạo nên một hệ sinh thái IoT

đa dạng, gần gũi giúp con người sắp xếp, tiết kiệm không gian, thời gian một cách hợp lý và có khoa học. Phòng khách sẽ tích hợp các thiết bị như là:

Cảm biến nhiệt độ (dùng để đo nhiệt độ trong phòng đây là tiền đề để sau này phát triển thêm để có thể điều hòa nhiệt độ trong phòng nếu trời nóng thì bật điều hòa mát lên, lạnh thì tăng nhiệt độ điều hòa lên).

Cảm biến chuyển động ở phòng khách nhằm để phát hiện chuyển động lạ khi chủ nhà không có nhà hoặc đang ngủ, biết được có người xâm nhập và có thể dùng để chống trộm nếu phát hiện có gì đó lạ thì sẽ còi sẽ kêu và thông báo đồng thời đèn sẽ sáng cho chủ nhà biết.

LCD sẽ hiển thị nhiệt độ, độ ẩm mà cảm biến nhiệt độ đã đo được.

Quạt ở phòng khách không chỉ giúp giảm nhiệt độ cho các thiết bị mà còn là 1 phần không thể thiếu trong các ngôi nhà.

Cảm biến khí Gas đặt ở phòng bếp để đo lường chất lượng không khí (nếu khí gas ở trong phòng bếp vượt quá ngưỡng thì cảm biến sẽ đo được và tự động kêu để cảnh báo cho người dùng để tránh khỏi thiệt hại về người và tài sản).

Bên cạnh đó, phòng ngủ sẽ được lắp các thiết bị như cảm biến ánh sáng khi trời tối đèn sẽ bật khi trời sáng đèn sẽ tắt đây là giải pháp thay thế chiếc đèn ngủ ngoài ra có thể tự bật tắt thiết bị qua app.

Tại phòng tắm sẽ có cảm biến mực nước nếu phát hiện mực nước cao quá mức cho phép có thể do chủ nhà quên tắt vòi nước hoặc vòi nước bị rò rỉ thì lập tức sẽ thông báo còi và đèn ngoài ra có thể tắt tiếng còi và đèn qua app.

Tại phòng ngủ sẽ có cảm biến ánh sáng nó là phần hoàn toàn có thể thay thế cho một chiếc đèn ngủ khi trời tối cảm biến sẽ bật đèn khi trời sáng cảm biến sẽ tắt.

Tại phòng bếp sẽ có cảnh báo khí gas do lắp đặt cảm biến khí gas nếu thấy khí gas quá mức cho phép lập tức còi sẽ báo để chủ nhà biết.

Với 2 thiết bị led và còi sẽ là các thiết bị phát ra tín hiệu để chủ nhà biết được nhà mình đang có vấn đề gì bất thường ngoài ra đèn led cũng để chiếu sáng cho căn nhà.

4.2. Ứng dụng vào thực tiễn

Đề tài nhà thông minh là một chủ đề tiềm năng và có rất nhiều cơ hội để phát triển trong tương lai. Nó mang lại nhiều lợi ích và tiện ích cho người sử dụng chúng. Nhà thông minh có thể giúp người dùng quản lý tiêu thụ năng lượng của ngôi nhà thông qua việc tự động điều chỉnh ánh sáng, nhiệt độ, thiết bị gia dụng. Những điều này giúp giảm chi phí năng lượng và bảo vệ môi trường. Bên cạnh đó hệ thống nhà thông minh có thể cung cấp tính năng an ninh thông minh như hệ thống giám sát video, cảnh báo trộm, và khóa cửa tự động sau mỗi lần mở cửa. Điều này giúp người dùng cảm thấy an toàn, đồng thời nếu có vấn đề xảy ra với ngôi nhà thì có thể dễ dàng điều tra do có hệ thống an ninh chặt chẽ. Mô hình nhà thông minh tạo ra một môi trường sống thoải mái, với đầy đủ tiện nghi do tất cả các thiết bị có thể tự động hóa và điều chỉnh hoạt động theo môi trường chúng ta sống. Mô hình giúp người dùng quản lý thời gian bằng cách nhắc nhở, lập lịch và thông báo tự động để chúng ta có thể thực hiện theo.

Tóm lại, còn rất nhiều lĩnh vực mà nhà thông minh có thể ứng dụng vào giúp nâng cao hiệu quả, đem lại lợi ích to lớn như: Y tế và sức khỏe, giải trí và giáo dục.

4.3. Ý tưởng phát triển

Môi trường trong tòa nhà đóng vai trò quan trọng đối với sức khỏe, thoải mái và hiệu suất làm việc của nhân viên hoặc người dân. Chính vì vậy, sự quản lý thông minh của môi trường trở thành một ưu tiên. Bài báo cáo này giới thiệu ý tưởng về việc ứng dụng cảm biến nhiệt độ, độ ẩm, tiếng ồn, khí ga, và ánh sáng

để cải thiện chất lượng môi trường cũng như điều khiển các thiết bị từ xa giúp bảo vệ tòa nhà của mọi người.

Chính vì vậy chúng ta cần phát triển mô hình này và phổ biến rộng rãi cho nhiều người sử dụng, giúp cải thiện chất lượng cuộc sống. Trong tương lai, tích hợp thêm trí tuệ nhân tạo (AI) vào ngôi nhà thông minh để phân tích dữ liệu, có thể tự động đưa ra các quyết định thay cho con người dựa trên thói quen để tối ưu mô hình về mặt không gian, thời gian, mô hình thực tế. Tạo ra nhiều kiểu tương tác hơn giữa người với ngôi nhà thông minh như: cử chỉ, hành động, và đặc biệt là bằng giọng nói. Chúng ta có thể xây dựng mô hình nhà thông minh cho những người bị khiếm khuyết, vốn dĩ họ đã thiệt thòi hơn chúng ta rất nhiều nên họ cũng nên được quan tâm và có thể bắt kịp với xu hướng của xã hội. Điều đó cũng có nghĩa là chúng ta phải có chính sách bảo mật quyền riêng tư, trách để lộ thông tin người dùng ra bên ngoài, bởi các thiết bị trong ngôi nhà đều được kết nối với Internet nên dễ bị tin tặc tấn công. Chúng ta phải phấn đấu để đất nước, con người Việt Nam không bị tụt lùi lại phía sau, bởi công nghệ vốn đa dạng và phát triển rất nhanh.

4.4. Triển khai xây dựng ứng dụng điều khiển mô hình nhà thông minh

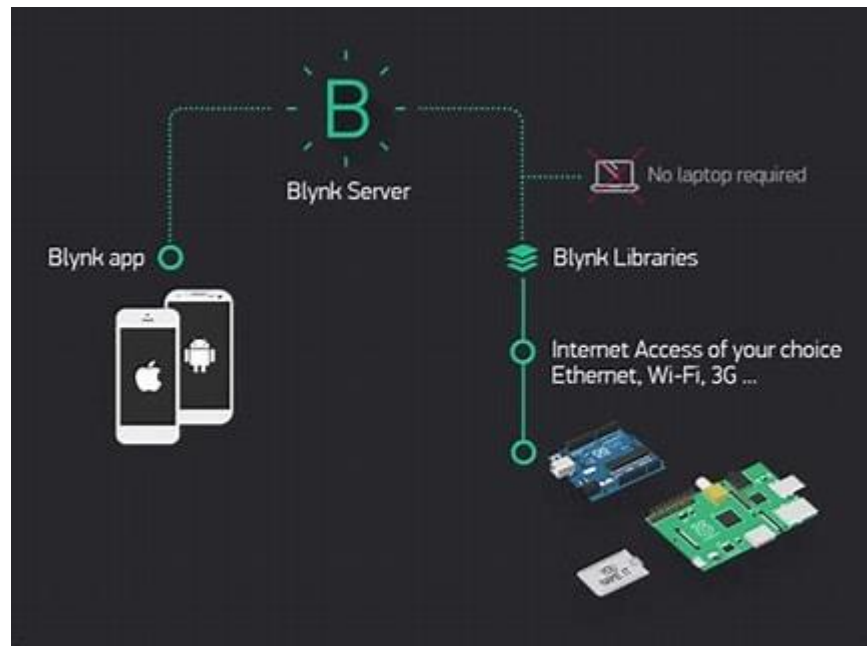
4.4.1. Ứng dụng blynk

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 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.



Hình 4. 2: Sơ đồ kết nối blynk với arduino

Blynk có thể được sử dụng để kết nối và quản lý đa dạng các thiết bị IoT, bao gồm:

Arduino: Blynk hỗ trợ kết nối và điều khiển các board Arduino thông qua mạng Internet. Điều này cho phép bạn tạo ra các ứng dụng IoT đa dạng sử dụng Arduino.

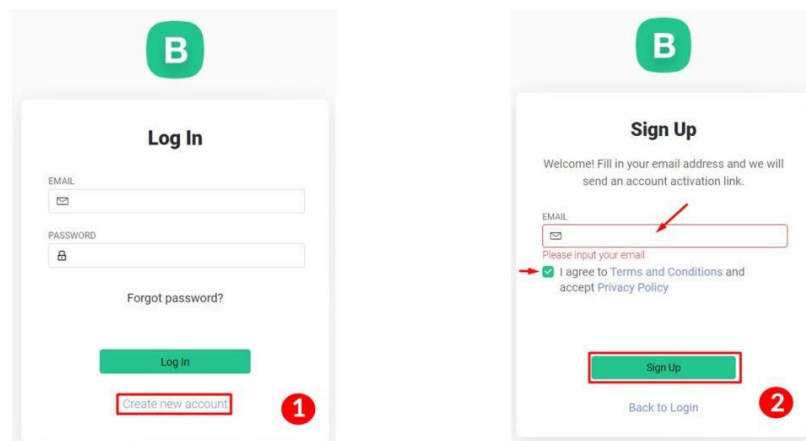
ESP8266 và ESP32: Blynk hỗ trợ việc kết nối và điều khiển các module ESP8266 và ESP32, cung cấp các phương thức dễ dàng để tương tác với chúng từ điện thoại thông minh.

Các thiết bị khác: Blynk cung cấp các thư viện và hướng dẫn để bạn có thể tùy chỉnh kết nối và quản lý các thiết bị IoT khác như NodeMCU, Particle, Intel Edison.

Với Blynk, bạn có thể tạo ra các ứng dụng IoT linh hoạt và tùy chỉnh theo nhu cầu của mình, kết nối và điều khiển các thiết bị từ xa thông qua mạng Internet.

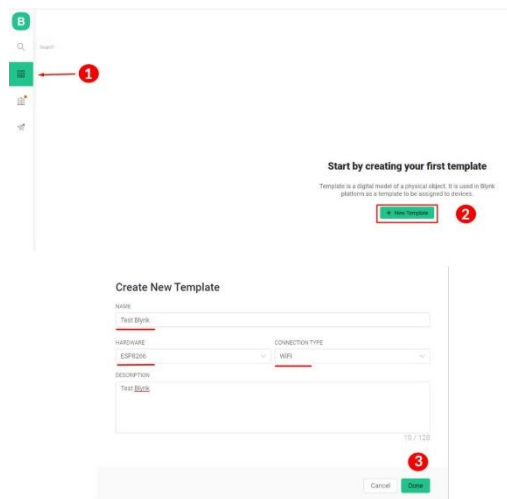
4.4.2. Cách sử dụng blynk

Đầu tiên hãy vào trang chủ của blynk và tạo tài khoản



Hình 4. 3: đăng nhập đăng ký

Sau khi đã có tài khoản hãy click chọn new template và chọn các thông số tùy thuộc vào các thiết bị



Hình 4. 4: Tạo template

Tạo Template mới với tên tự đặt tùy chỉnh ngoài ra ở trường hardware chọn thiết bị phù hợp với bài toàn của mình. Ở đây thì em chọn ESP32.

Create New Template

NAME
Name
Field is required

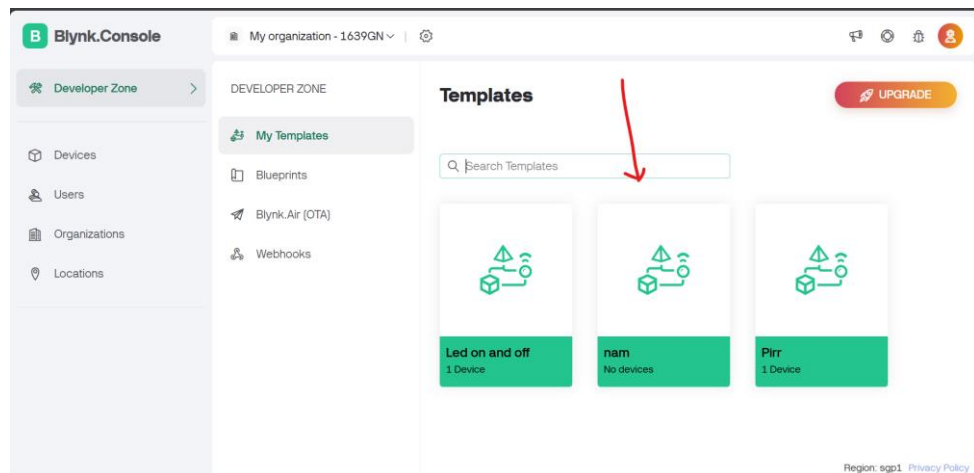
HARDWARE CONNECTION TYPE
ESP32 WIFI

DESCRIPTION
Description
0 / 128

Cancel Done

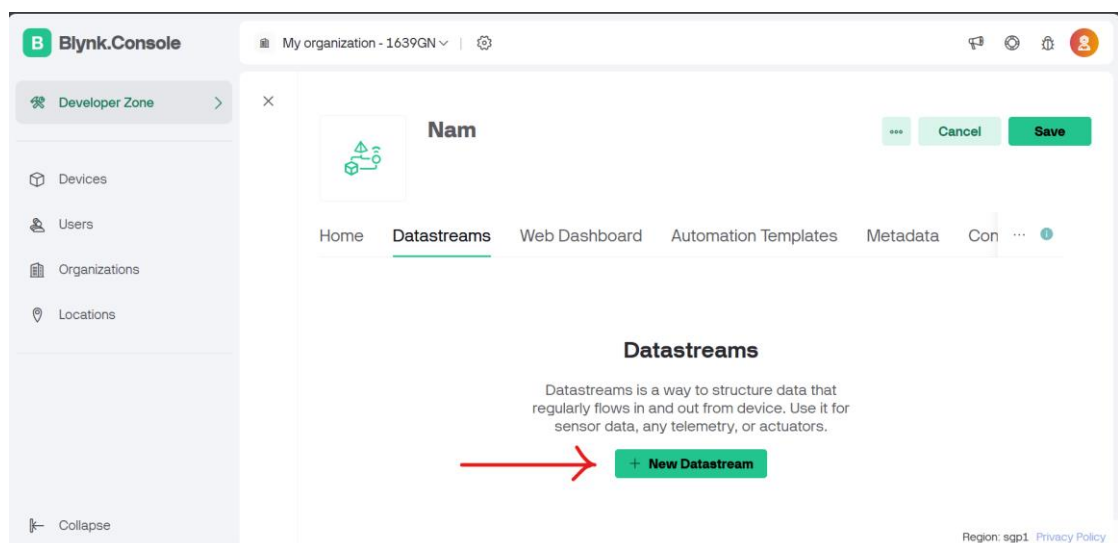
Hình 4. 5: Tạo template

Sau đó click vào Template vừa tạo xong ở đây em vừa tạo Template tên nam nên click vào Template đó.



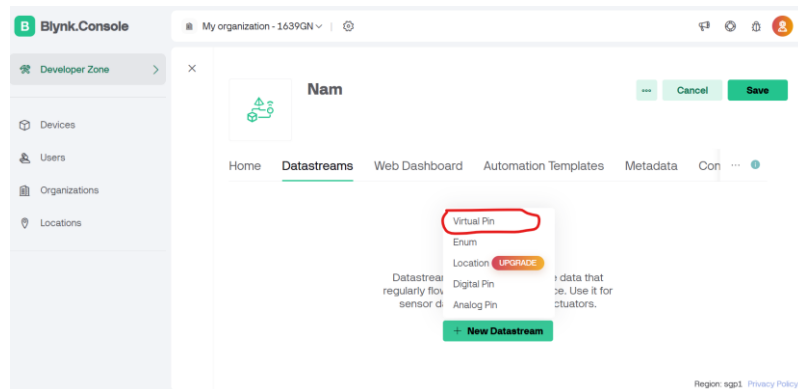
Hình 4. 6: Chọn template vừa tạo

Sau khi màn hình đã hiện như ảnh dưới đây click vào ô Datastreams và tạo new Datastreams



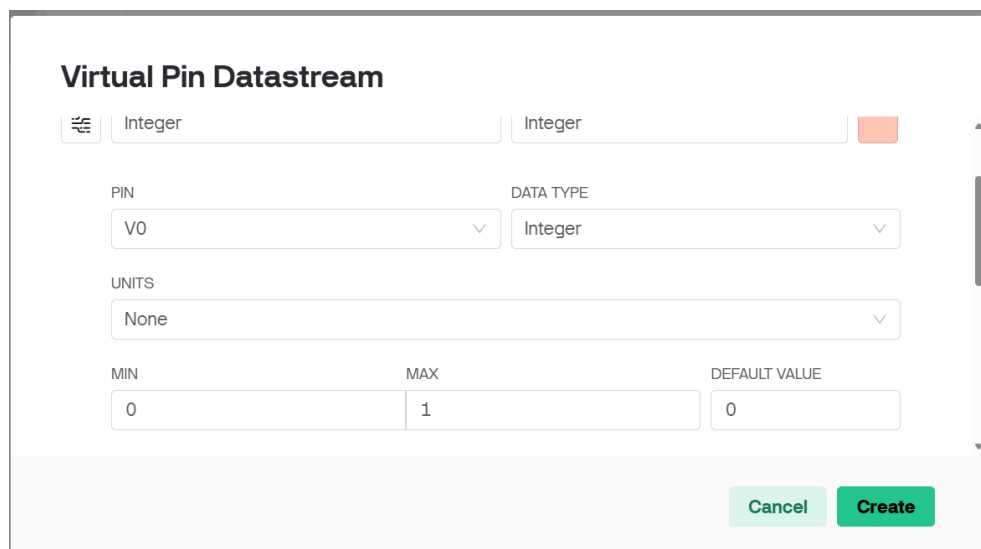
Hình 4. 7: Chọn new datastreams

Sau đó tại new Datastream chọn Virtual Pin để tạo đầu Pin ảo



Hình 4. 8: Tạo pin ảo

Sau đó thiết lập như các thông số ở dưới đây và tạo create.

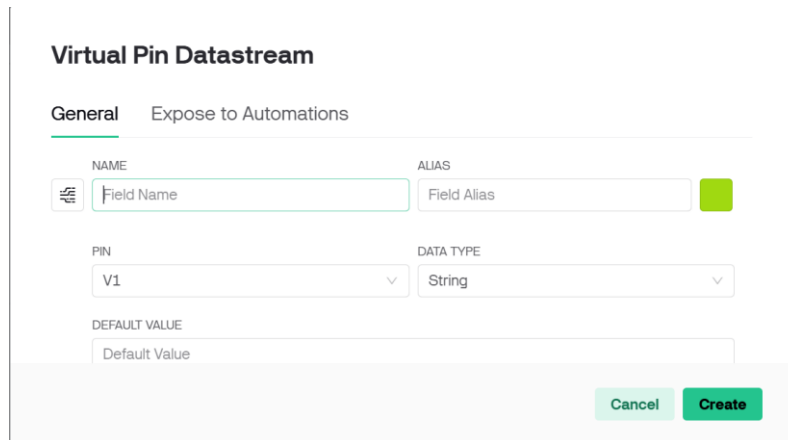


Hình 4. 9: Điền thông tin vô trường pin ảo

Trường đầu tiên sẽ là name với tên tự đặt, tiếp theo là alias sẽ tự đặt theo tên của mình ngoài ra có thể đổi màu nhờ nhấn nút bên cạnh, Pin sẽ là Pin ảo tự tạo là V0 có thể điều chỉnh tùy thích, DATA TYPE sẽ là Integer, UNIT lưu trữ

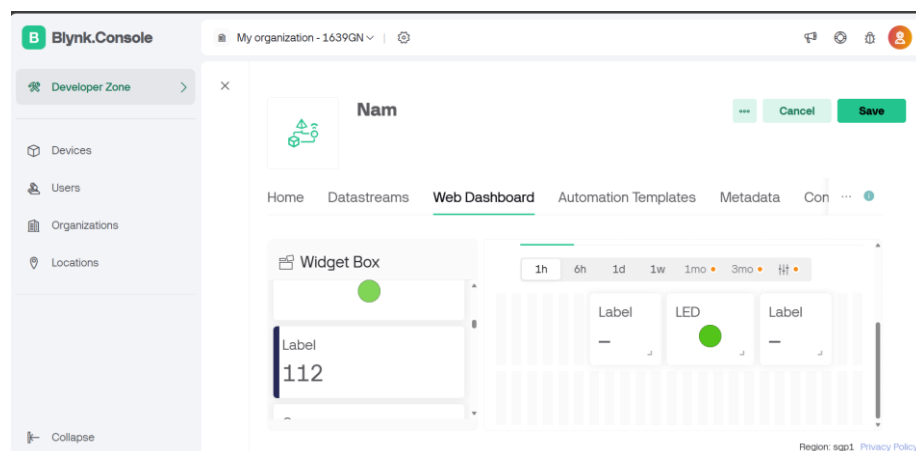
quản lý thông tin về thước đo chọn None và sau đó set giá trị min là 0 max là 1 với giá trị mặc định là 0 và ấn Create.

Tiếp tục tạo Datastream khác với các giá trị sau



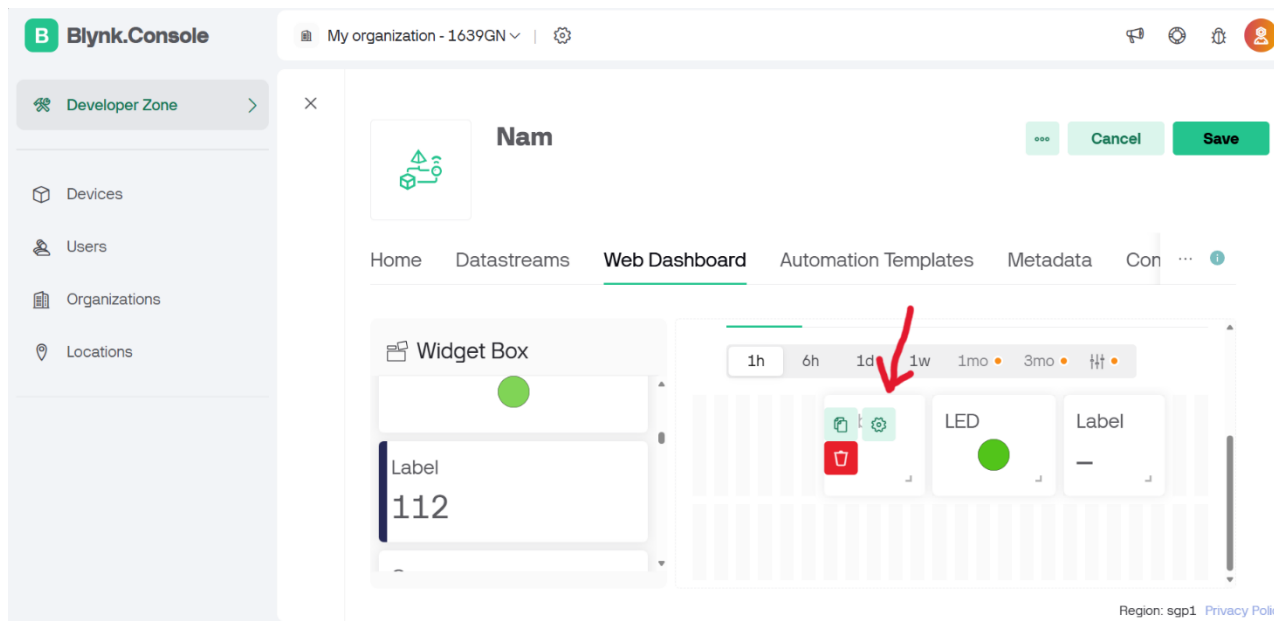
Hình 4. 10: Nhập các thông tin vô trường của virtual pin

Sau khi tạo xong ở Datastreams chọn Web dashboard để thiết lập dashboard ở đó chọn các widget box ở dưới và cho vào khung ở dưới các số giờ như hình sau



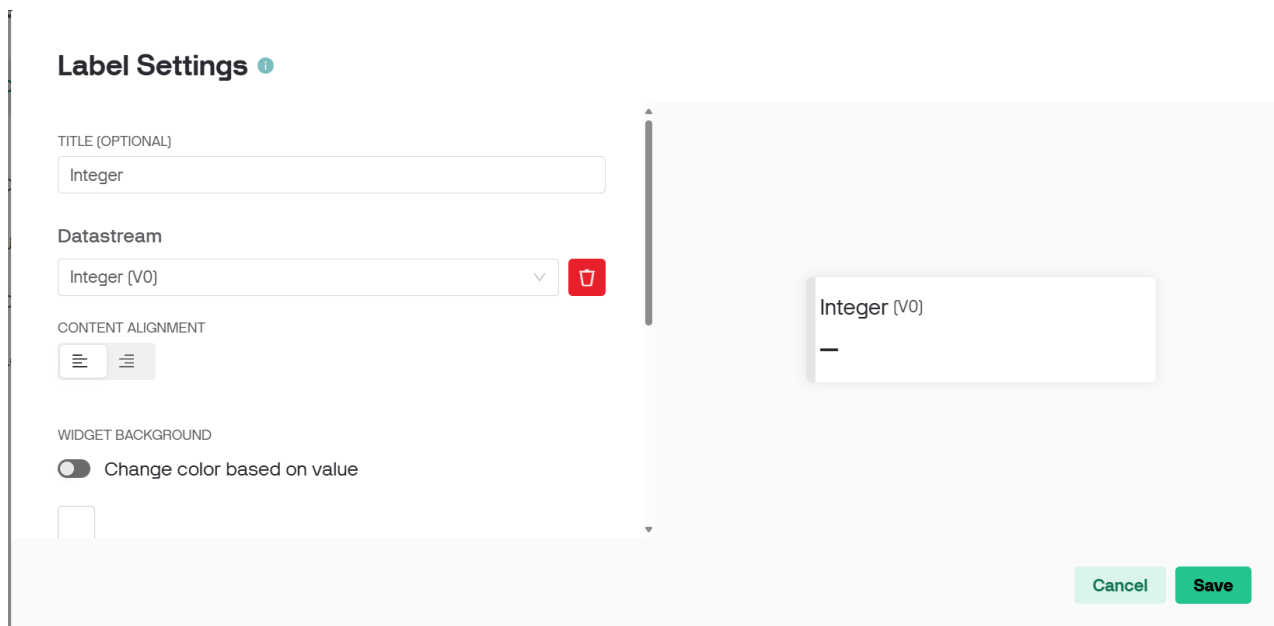
Hình 4. 11: Tạo các widget

Tiếp theo đó ấn nút cài đặt như hình dưới để thiết lập cho các widget vừa tạo

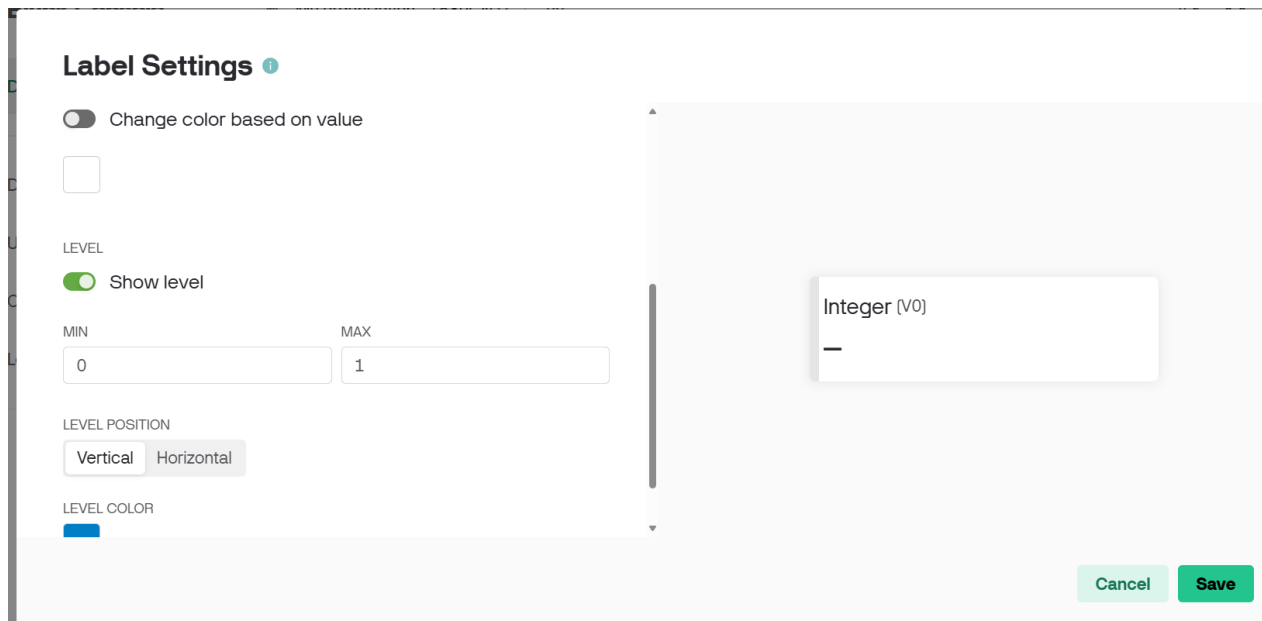


Hình 4. 12: Setting các widget

Tiếp đó chọn Datastream có Pin ảo là V0 và thiết lập như các hình dưới

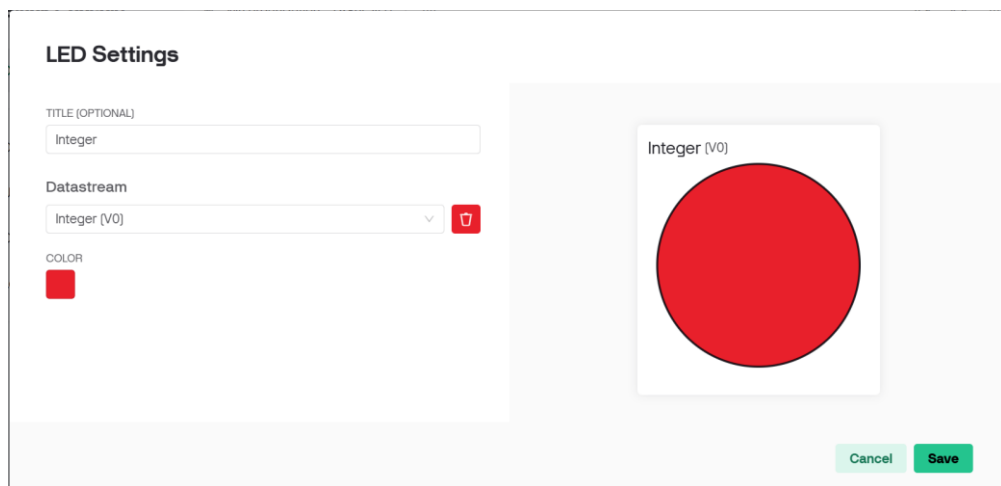


Hình 4. 13: Chọn pin ảo vừa tạo phù hợp



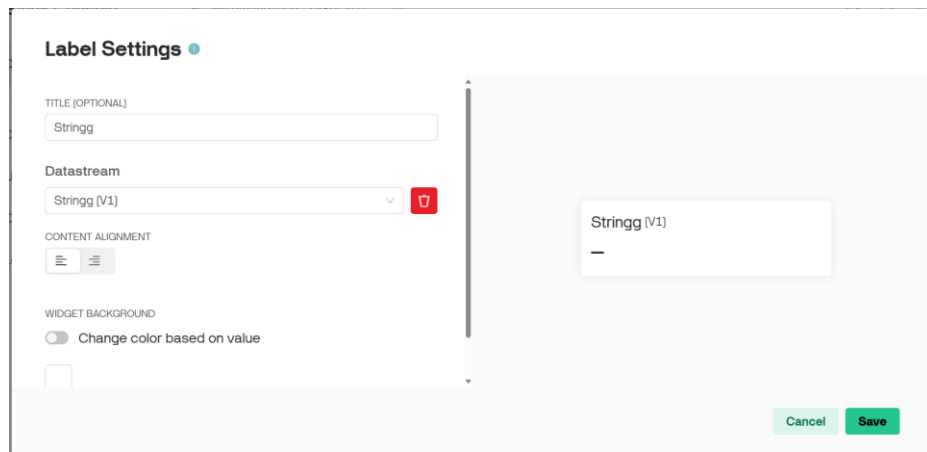
Hình 4. 14: Chọn các thông tin theo hướng dẫn

Tương tự ở đèn led cũng vậy, tuy nhiên nếu muốn làm thiết bị khác hoặc không thích dùng led có thể đổi sang switch qua đó thể hiện được bật tắt nhiều thiết bị cùng 1 lúc hoặc làm nhiều thiết bị khác nhau nhưng chỉ dùng switch đều được.



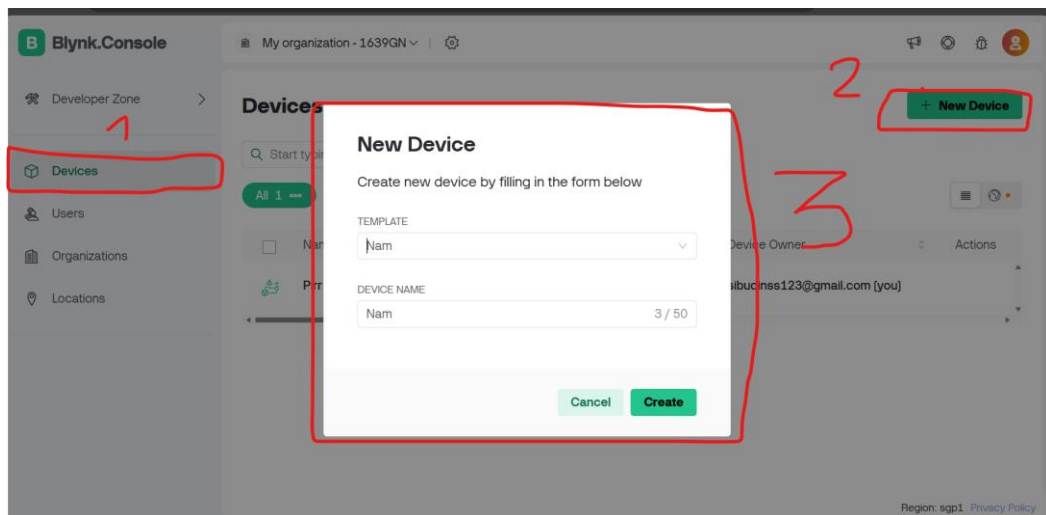
Hình 4. 15: Setting với widget đèn led

Tiếp nữa ở label khác chọn String như hình sau



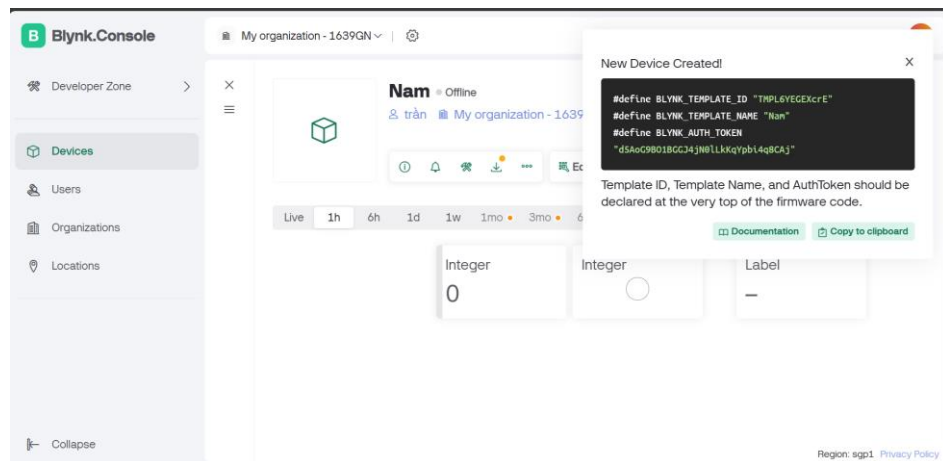
Hình 4. 16: Setting label và chọn pin ảo phù hợp

Sau khi đã tạo xong hoàn tất thì vô phần device và sau đó ấn new device chọn template vừa mới tạo và create.



Hình 4. 17: Tạo thiết bị mới

Sau đó màn hình sẽ hiện ra ô New Device Create chứa bao gồm id tên và token hãy copy ô đó và để dán vào code nhằm cho App Blynk nhận diện được thiết bị của mình.

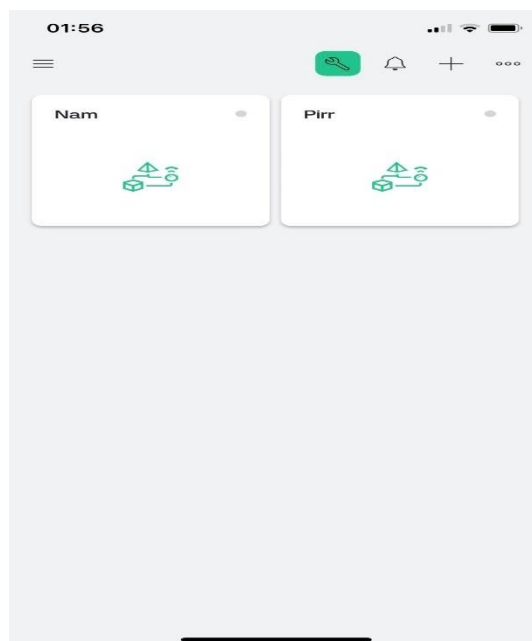


Hình 4. 18: Copy các giá trị define

Code sẽ được dán như hình dưới đây

```
canhbaochongtrom.ino
1  #define BLYNK_TEMPLATE_ID "TMPL6wiYcI_JM"
2  #define BLYNK_TEMPLATE_NAME "Pirr"
3  #define BLYNK_AUTH_TOKEN "YRW8bwZ5j_1cW_RoJDENm-8inJeRASe0"
4
5  /* Comment this out to disable prints and save space */
6  #define BLYNK_PRINT Serial
```

Sau đó mở app Blynk trên điện thoại chọn Device vừa tạo



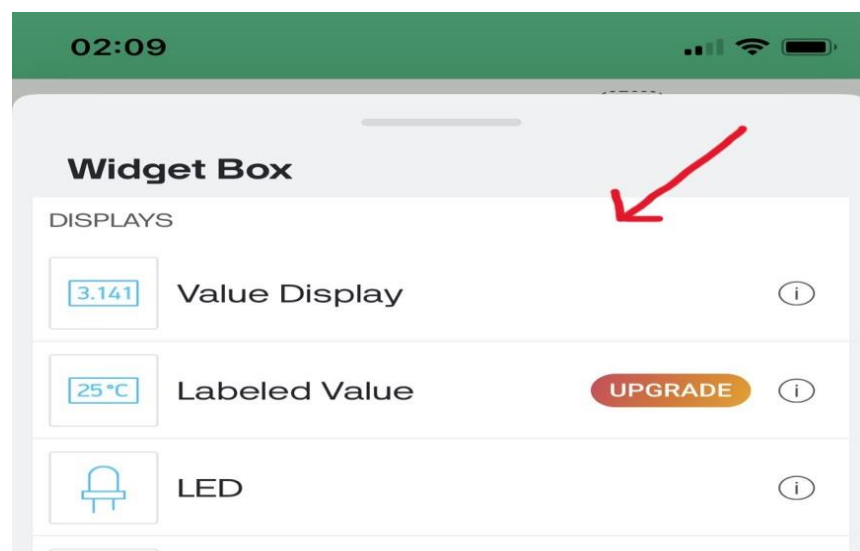
Hình 4. 19: Giao diện app blynk

Sau đó ấn vào nút cài đặt như hình dưới



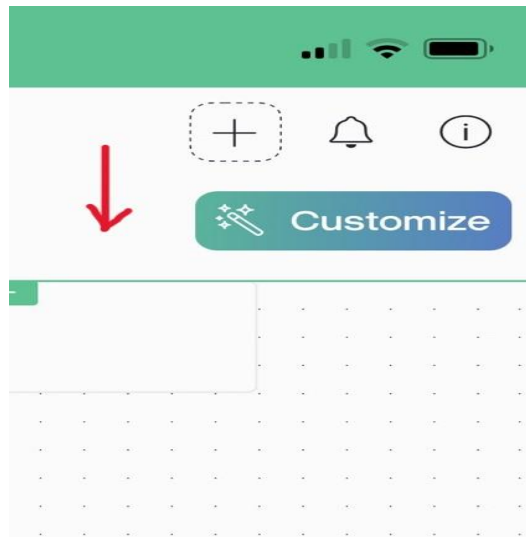
Hình 4. 20: Chọn nút setting

Và app sẽ hiện các ô để chọn, tiếp nữa kéo xuống tìm ô Value Display và chọn nó



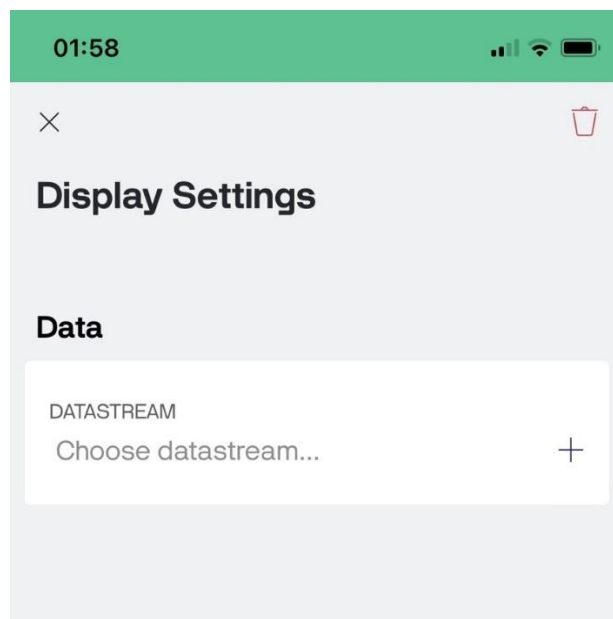
Hình 4. 21: Chọn value display

Sau khi làm như này xong tiếp theo chọn ô vừa tạo ở hình dưới



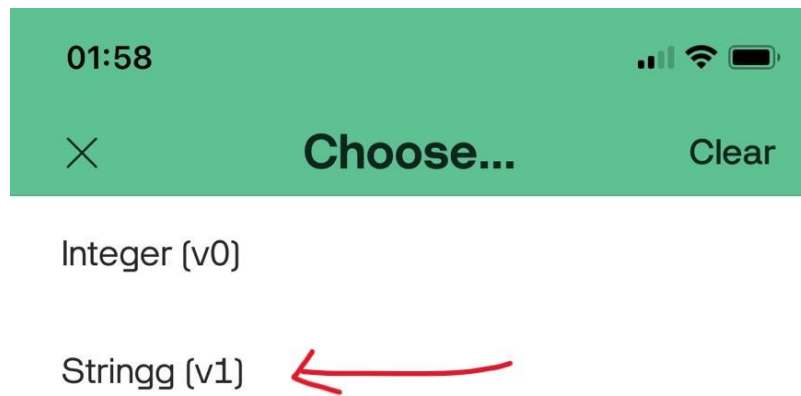
Hình 4. 22: Ấn ô vừa tạo

Và hình nền điện thoại sẽ hiện ra bảng sau và chọn trường ở dưới chữ Data là DATASTREAM



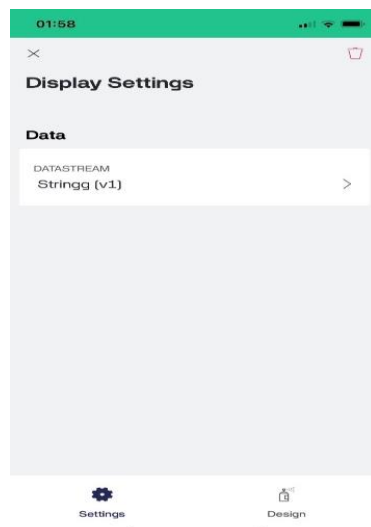
Hình 4. 23: Giao diện sau khi ấn vào ô vừa tạo

Sau khi ấn giao diện sẽ hiện như sau và ta ấn vào hình String đã thiết lập

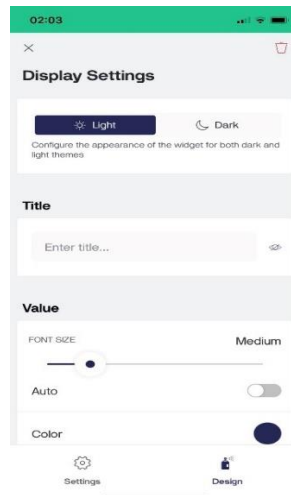


Hình 4. 24: Chọn pin ảo phù hợp

Sau đó sẽ hiện giao diện ở khu vực setting như hình đầu tiên thì sẽ chọn design nút ở dưới cùng góc bên phải có hình lọ sơn, tại đây setting tùy chỉnh theo mình muốn ngoài ra ở Title có thể đặt tên tùy chỉnh.

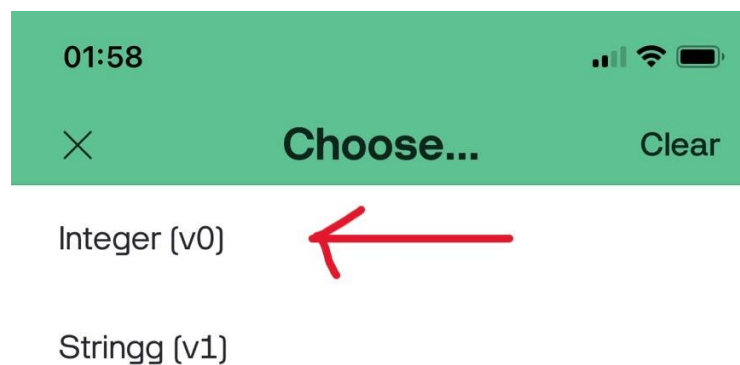


Hình 4. 25: Giao diện sau khi chọn pin ảo



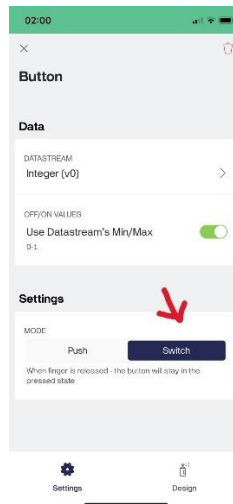
Hình 4. 26: Giao diện ở design

Ở switch cũng tương tự như vậy cho đến lúc chọn Datastream thay vì chọn Stringg thì chọn Integer.



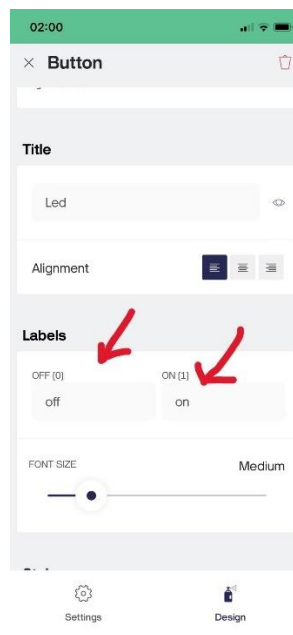
Hình 4. 27: Chọn pin ảo phù hợp

Tuy nhiên đối với switch sẽ có đôi chút khác với label sau khi đã chọn xong mục Integer ở switch sẽ hiện như hình sau, và ở đoạn setting chọn switch thay vì push.



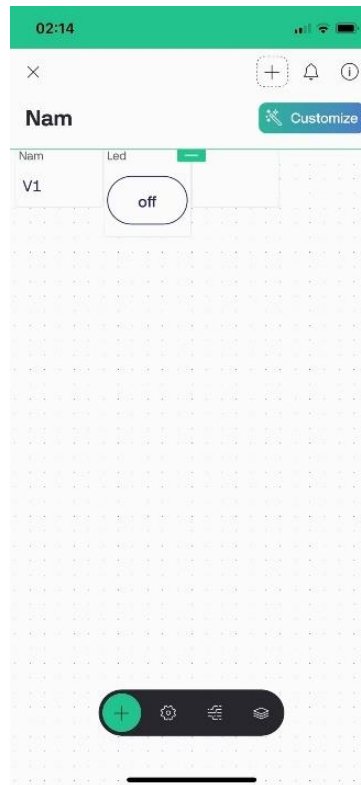
Hình 4. 28: Đổi chế độ ấn

Tiếp theo đó ấn vào design ở góc phải có hình lọ sơn như ở Label và setting tùy chỉnh để nguyên cũng được ở đây dưới chỗ label có 2 nhãn là on off thì đặt là on off hoặc bật tắt cho đúng với nút bật tắt cho đèn led hoặc có thể đặt khác tùy người set không nhất thiết phải để on và off.



Hình 4. 29: Cài label cho switch

Và sau đó thoát ra nếu đã hiển thị giao diện có 2 ô label và switch như hình dưới đây thì là đã tạo thành công.



Hình 4. 30: Giao diện khi cài xong

Tiểu kết: Trong chương này cùng tìm hiểu về mô hình của em, ứng dụng mô hình đó vào trong thực tiễn và ý tưởng để phát triển tiếp mô hình qua đó tìm hiểu về ứng dụng blynk để điều khiển các thiết bị trong ngôi nhà.

KẾT LUẬN

Việc xây dựng ứng dụng tự động cảnh báo bất thường trong mô hình nhà thông minh mang lại nhiều lợi ích đáng kể cho người dùng. Hệ thống này giúp phát hiện và cảnh báo kịp thời về các tình huống nguy hiểm, sự cố hoặc hành vi không bình thường trong ngôi nhà, tạo ra một môi trường sống an toàn, tiện nghi và thông minh hơn.

Bằng cách tích hợp các cảm biến và thiết bị thông minh, hệ thống nhà thông minh có khả năng theo dõi và phát hiện các tín hiệu bất thường như xâm nhập trái phép, cháy nổ, rò rỉ nước, hay tình huống y tế khẩn cấp. Khi phát hiện xảy ra, hệ thống tự động gửi cảnh báo cho người dùng, giúp họ có thể phản ứng kịp thời và đưa ra các biện pháp phòng ngừa, khắc phục hoặc yêu cầu sự hỗ trợ cần thiết.

Ứng dụng tự động cảnh báo bất thường trong mô hình nhà thông minh đóng vai trò quan trọng trong việc tăng cường an ninh và an toàn cho người dùng. Nó giúp ngăn chặn tình huống nguy hiểm, bảo vệ tài sản và tính mạng của người dùng. Đồng thời, hệ thống này cũng giúp tiết kiệm thời gian, công sức và nguồn lực của người dùng bằng cách tự động giám sát và phát hiện các tình huống nguy hiểm, thay vì phải theo dõi liên tục.

Việc tích hợp trí tuệ nhân tạo, học máy và Internet of Things (IoT) trong ứng dụng tự động cảnh báo bất thường giúp nâng cao khả năng phát hiện và cảnh báo của hệ thống. Hệ thống có khả năng học từ dữ liệu và thích nghi với các mô hình sử dụng của người dùng, tăng tính hiệu quả và chính xác.

Tổng quan, việc xây dựng ứng dụng tự động cảnh báo bất thường trong mô hình nhà thông minh là một đóng góp quan trọng để tạo ra một môi trường sống an toàn, tiện nghi và thông minh. Nó không chỉ đảm bảo an ninh và an

toàn cho người dùng, mà còn mang lại sự tiện lợi và hòa nhập với cuộc sống hiện đại.

TÀI LIỆU THAM KHẢO

- [1] 19. Project 4: Hệ thống chống trộm và báo động — OhStem Education Wiki 2.0 documentation
- [2] Internet of Things (IoT) là gì? Ứng dụng của IoT và Ưu điểm, nhược điểm (funix.edu.vn)
- [3] How to set up the new Blynk app with an ESP32 board | ESP32 projects (youtube.com)
- [4] Step-by-step video guide to making your first project on Blynk
- [5] Arduino Remote Control Tutorial (build-electronic-circuits.com)
- [6] Arduino Tutorial (tutorialspoint.com)
- [7] Getting Started with the ESP32 Development Board | Random Nerd Tutorials