

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



ĐỒ ÁN TỐT NGHIỆP

NGÀNH ĐIỆN TỰ ĐỘNG CÔNG NGHIỆP

Sinh viên : Phan Minh Phú

Giảng viên hướng dẫn : TS. Đoàn Hữu Chức

Hải Phòng – 2022

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

GIÁM SÁT VÀ ĐIỀU KHIỂN THIẾT BỊ TRONG NHÀ
BẰNG GIỌNG NÓI QUA INTERNET

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY
NGÀNH ĐIỆN TỰ ĐỘNG CÔNG NGHIỆP

Sinh viên thực hiện : Phan Minh Phú

Giảng viên hướng dẫn: TS. Đoàn Hữu Chức

Hải Phòng - 2022

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG**

NHIỆM VỤ ĐỀ TÀI TỐT NGHIỆP

Sinh viên : Phan Minh Phú

MSV : 1812102007

Lớp : DC2201

Ngành: Điện tự động công nghiệp

Tên đề tài : Giám sát và điều khiển thiết bị trong nhà băng giọng nói qua Internet

NHIỆM VỤ ĐỀ TÀI

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

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

2. Các số liệu cần thiết để tính toán.

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

3. Địa điểm thực tập tốt nghiệp.

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

CÁC CÁN BỘ HƯỚNG DẪN ĐỀ TÀI TỐT NGHIỆP

Người hướng dẫn thứ nhất:

Họ và tên : Đoàn Hữu Chúc

Học hàm, học vị : Tiến Sĩ

Cơ quan công tác : Trường Đại học Quản lý và Công nghệ Hải Phòng

Nội dung hướng dẫn : Toàn bộ đề tài

Người hướng dẫn thứ hai:

Họ và tên :

Học hàm, học vị :

Cơ quan công tác :

Nội dung hướng dẫn :

Đề tài tốt nghiệp giao ngày tháng năm 2022.

Yêu cầu phải hoàn thành xong trước ngày tháng năm 2022

Đã nhận nhiệm vụ Đ.T.T.N

Sinh viên

Đã giao nhiệm vụ Đ.T.T.N

Cán bộ hướng dẫn Đ.T.T.N

Phan Minh Phú

TS. Đoàn Hữu Chúc

Hải Phòng, ngày tháng năm 2022

TRƯỞNG KHOA

TS. ĐOÀN HỮU CHỨC

Cộng hòa xã hội chủ nghĩa Việt Nam

Độc lập - Tự do - Hạnh phúc

PHẦN NHẬN XÉT TÓM TẮT CỦA CÁN BỘ HƯỚNG DẪN

Họ và tên giảng viên: Đoàn Hữu Chức
Đơn vị công tác: Trường Đại học Quản lý và Công nghệ Hải Phòng
Họ và tên sinh viên: Phan Minh Phú
Chuyên ngành: ĐIỆN TỬ ĐỘNG CÔNG NGHIỆP
Nội dung hướng dẫn : Toàn bộ đề tài

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

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

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

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

3. Ý kiến của giảng viên hướng dẫn tốt nghiệp

Được bảo vệ Không được bảo vệ Điểm hướng dẫn

Hải Phòng, ngày.....tháng.....năm 2022

Giảng viên hướng dẫn

(ký và ghi rõ họ tên)

Cộng hòa xã hội chủ nghĩa Việt Nam

Độc Lập – Tự Do – Hạnh Phúc

**NHẬN XÉT ĐÁNH GIÁ CỦA NGƯỜI CHẤM PHẢN BIỆN
ĐỀ TÀI TỐT NGHIỆP**

Họ và tên giảng viên:

Cơ quan công tác:.....

Họ và tên sinh viên:.....

Chuyên ngành:

Đề tài tốt nghiệp:

1. Phần nhận xét, đánh giá của giảng viên chấm phản biện

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

2. Những mặt còn hạn chế

.....
.....

3. Ý kiến của giảng viên chấm phản biện

Được bảo vệ

Không được bảo vệ

Điểm hướng dẫn

Hải Phòng, ngày tháng năm 2022

Giảng viên chấm phản biện

(ký và ghi rõ họ tên)

LỜI CẢM ƠN

Sau quá trình tìm hiểu, nghiên cứu và thực hiện đồ án tốt nghiệp, em xin chân thành gửi lời cảm ơn sâu sắc nhất đến quý thầy cô, gia đình và bạn bè, những người đã hết mình giúp đỡ, đưa ra những lời khuyên, định hướng cho em thực hoàn thành đề tài này.

Em xin gửi lời cảm ơn đến thầy Đoàn Hữu Chức, giảng viên trường Đại Học Quản lý và công nghệ Hải phòng đã tận tình giúp đỡ, hướng dẫn và tạo điều kiện thuận lợi để em có thể thực hiện tốt đề tài của mình.

Và em cũng xin gửi lời tri ân đến các thầy cô công tác tại khoa Điện- Điện tử của trường Đại Học Quản lý và công nghệ Hải phòng đã tận tình dạy dỗ và chỉ bảo cho em trong suốt những năm qua, tạo cho em một cơ sở kiến thức vững vàng để thực hiện đồ án cũng như áp dụng trong cuộc sống sau này.

Gia đình luôn là nguồn động lực và là nguồn cảm hứng để em nỗ lực và phấn đấu trong học tập cũng như trong cuộc sống, Chính vì vậy, em xin gửi lời cảm ơn đến bố mẹ, những người đã giúp đỡ rất nhiều để em có được ngày hôm nay.

Và cuối cùng, nhóm Em cũng gửi lời đồng cảm ơn đến các bạn bè, người đã chia sẻ kiến thức cũng như những kinh nghiệm quý báu trong thời gian thực hiện đề tài.

Xin chân thành cảm ơn các thầy !

Người thực hiện đề tài

Phan Minh Phú

TÓM TẮT

Đề tài này hướng đến việc thiết kế, thi công mô hình hệ thống điều khiển các thiết bị điện và giám sát nhà sử dụng vi điều khiển chính là arduino mega 2560, vi điều khiển này thực hiện việc điều khiển các thiết bị bật/tắt dưới sự điều khiển của người dùng thông qua giọng nói, ứng dụng trên điện thoại hoặc công tắc trực tiếp, đồng thời đọc các module cảm biến để thu thập các thông số của môi trường xung quanh. Arduino mega 2560 kết nối với NodeMCU để truy cập vào internet nhằm nhận và gửi dữ liệu tới người dùng. Ngoài ra, mô hình còn sử dụng một arduino uno r3 để đọc dữ liệu từ bàn phím, hiển thị dữ liệu lên LCD và điều khiển động cơ servo đóng/mở cửa.

MỤC LỤC

CHƯƠNG 1 TỔNG QUAN	13
1.1 ĐẶT VẤN ĐỀ	14
1.2 MỤC TIÊU	14
1.3 NỘI DUNG THỰC HIỆN	15
1.4 GIỚI HẠN.....	15
1.5 BỐ CỤC.....	16
CHƯƠNG 2 CƠ SỞ LÝ THUYẾT.....	17
2.1 TỔNG QUAN VỀ ARDUINO	17
2.1.1. Giới thiệu	17
2.1.2. Phần cứng arduino	18
2.1.3. Arduino mega 2560	18
2.1.4. Arduino uno r3.....	21
2.1.5. Phần mềm lập trình arduino IDE.....	23
2.2 MODULE CẢM BIẾN KHÍ GA MQ2.....	25
2.3 MODULE CẢM BIẾN CHUYỂN ĐỘNG PIR HC – SR501	26
2.4 MODULE WIFI ESP8266 NODEMCU.....	27
2.5 MODULE CẢM BIẾN NHIỆT ĐỘ - ĐỘ ẨM DHT11	29
2.6 QUANG TRỞ	30
2.7 NGUỒN XUNG 12V-5A	30
2.8 BÀN PHÍM MA TRẬN 4X4.....	31
2.9 MÀN HÌNH LCD 1602	32
2.10 LÝ THUYẾT CHUẨN GIAO TIẾP UART	34
2.11 LÝ THUYẾT CHUẨN GIAO TIẾP ONE-WIRE	36
CHƯƠNG 3 TÍNH TOÁN THIẾT KẾ	39
3.1 GIỚI THIỆU	39
3.2 TÍNH TOÁN VÀ THIẾT KẾ HỆ THỐNG.....	40
3.2.1 <i>Thiết kế sơ đồ khối hệ thống</i>	40
3.2.2 <i>Tính toán và thiết kế mạch</i>	42

3.2.2.1	Thiết kế khối cảm biến.....	42
3.2.2.2	Thiết kế khối xử lí trung tâm	43
3.2.2.3	Thiết kế khối Relay điều khiển thiết bị.....	43
3.2.2.4	Khối truyền dữ liệu	44
3.2.2.5	Thiết kế khối hiển thị trên LCD.....	45
3.2.2.6	Thiết kế khối nguồn	46
3.2.2.7	Sơ đồ nguyên lý của toàn mạch	49
3.3	LẬP TRÌNH HỆ THỐNG	50
3.3.1	<i>Lưu đồ giải thuật module NodeMCU</i>	50
3.3.2	<i>Lưu đồ giải thuật module Aduino mega 2560</i>	51
CHƯƠNG 4 KẾT QUẢ THỰC HIỆN.....		53
5.1	KẾT QUẢ	53
4.1.1	<i>Mô hình hệ thống</i>	53
4.1.2	<i>Giao diện phần mềm hệ thống</i>	54
4.1.3	Hiển thị thông báo trên LCD	55
CHƯƠNG 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN		58
5.1	KẾT LUẬN.....	58
5.2	HƯỚNG PHÁT TRIỂN	59

CÁC TỪ VIẾT TẮT

Từ viết tắt	Tên tiếng anh	Chú thích
AVR	Automatic Voltage Regulator	Hệ thống tự động điều khiển điện áp đầu cực máy phát điện
ARm	Advanced RISC Machine	Là một kiến trúc dạng RISC cho các vi xử lý
AC	Alternating Current	Dòng xoay chiều
I/O	Input/Output	Đầu vào/đầu ra
IoT	Internet of Thing	
USB	Universal Serial Bus	Là một chuẩn kết nối tuần tự đa dụng trong máy tính
FTDI	Future Technology Devices International	Mô-đun mạch chuyển đổi tín hiệu USB sang UART.
TTL	Transistor-transistor logic	Là một lớp mạch kỹ thuật số được xây dựng từ các transistor lưỡng cực với một số điện trở phụ trợ
PWM	Pulse-width modulation	Điều chế độ rộng xung
DC	Direct Curent	Dòng một chiều
SRAM	Static random-access memory	Bộ nhớ truy cập ngẫu nhiên tĩnh hay Ram tĩnh
EEPROM	Electrically Erasable Programmable Read-Only Memory	Bộ nhớ chỉ cho phép đọc không mất dữ liệu khi ngừng cung cấp điện.
SPI	Serial Peripheral Interface	Là một chuẩn đồng bộ nối tiếp để truyền dữ liệu ở chế độ song công toàn phần
IDE	Integr ted Development Environment	Môi trường tích hợp dùng để viết code để phát triển ứng dụng .
WIFI	Wireless Fidelity	Hệ thống hoạt động dựa trên sóng vô tuyến không dây
UART	Universal Asynchronous Receiver/Transmitter	Bộ truyền nhận nối tiếp bất đồng bộ.
LCD	Liquid Cryst l Display	Màn hình tinh thể lỏng
IC	Integrated circuit	Là tập các mạch điện chứa các linh kiện bán dẫn và linh kiện điện tử .
PIR	Passive InfraRed	Bộ cảm biến thụ động dùng nguồn kích thích là tia hồng ngoại
ID	Identifier	Định danh nhận dạng

LỜI MỞ ĐẦU

Các quốc gia trên thế giới đang chạy đua nghiên cứu và phát triển IOT rất mạnh mẽ, kéo theo đó nhu cầu của con người trong việc ứng dụng IOT vào đời sống ngày càng tăng cao. Một trong số đó là điều khiển các thiết bị ngay trong chính ngôi nhà của mình. Sự đảm bảo về an ninh cho ngôi nhà là một nhu cầu thiết yếu. Đó cũng như một sự yên tâm đảm bảo cho mỗi cá nhân khi ra xã hội làm việc. Ai trong chúng ta cũng có lúc vội vã rời khỏi nhà mà quên tắt thiết bị điện, máy lạnh hay quên đóng cửa sổ... Lúc đó chúng ta không tránh khỏi cảm giác lo lắng về an toàn cho ngôi nhà mình.

Hệ thống nhà thông minh với các tiện ích vượt trội về an ninh, tiện ích, an toàn, tiết kiệm... Giúp thực hiện tự động các kịch bản được lập trình sẵn theo ý thích riêng của mỗi gia đình. Đồng thời chủ nhà còn có thể điều khiển, theo dõi từ xa ngôi nhà của mình. Đem đến cho gia chủ cảm giác thoải mái và hài lòng khi mọi việc được thực hiện trong tầm tay.

Là sinh viên khoa Điện của Trường Đại Học Quản Lý Và Công Nghệ Hải Phòng, với những kiến thức đã học cùng với mong muốn thiết kế một ngôi nhà tự động hóa đáp ứng nhu cầu sinh hoạt hàng ngày, em đã chọn "**Giám sát và điều khiển thiết bị trong nhà bằng giọng nói qua Internet**"

Đề án gồm các nội dung sau:

Chương 1: Tổng quan

Chương 2: Cơ sở lý thuyết

Chương 3: Tính toán và thiết kế

Chương 4: Kết quả thực hiện

Chương 5: Kết luận và hướng phát triển

CHƯƠNG 1

TỔNG QUAN

1.1 ĐẶT VẤN ĐỀ

Trong thời đại ngày nay, khi mà khoa học kỹ thuật vô cùng phát triển, cùng với sự thịnh hành của Internet, Smartphone và đặc biệt là các thiết bị cảm biến, Internet of Things (IoT) đang trở thành xu hướng mới của thế giới. Không chỉ giúp cho con người thuận tiện hơn trong các công việc hằng ngày, IoT còn có thể giúp cho các thiết bị hoạt động một cách tự động nhằm đem đến cho con người những trải nghiệm tốt nhất trong cuộc sống.

Từ thực tế đó và trên cơ sở nhu cầu ngày càng cao của con người trong việc cần có các hệ thống điều khiển, thu thập dữ liệu cũng như giám sát các thiết bị trong nhà thì em đã quyết định thực hiện đề tài " ĐIỀU KHIỂN VÀ GIÁM SÁT THIẾT BỊ ĐIỆN BẰNG GIỌNG NÓI QUA INTERNET"

Nhà thông minh được thiết kế sử dụng adruino mega 2560 cho việc điều khiển và giám sát những thiết bị trong nhà. Ngoài việc kết nối với các cảm biến và công tắc, arduino mega còn kết nối với kit NodeMCU để truyền – nhận dữ liệu, một arduino uno để đọc dữ liệu từ bàn phím. Về phía người dùng, họ có thể sử dụng giọng nói, ứng dụng trên điện thoại android để điều khiển và theo dõi ngôi nhà của mình. Việc mô phỏng các thiết bị trong nhà được thể hiện qua mô hình ngôi nhà bao gồm các thiết bị điện bên trong như đèn, quạt, cửa ra vào và các loại cảm biến khí gas, chuyển động , nhiệt độ - độ ẩm...

1.2 MỤC TIÊU

Đối với phần cứng, em sẽ tìm hiểu và nghiên cứu về cách hoạt động của kit Arduino mega 2560, NodeMCU, các loại module cảm biến:khí Gas MQ2, cảm biến nhiệt độ - độ ẩm, cảm biến chuyển động... và cách kết nối giữa các module để hoàn thành mô hình.

Về phần mềm, em sẽ sử dụng Google Assistant (trợ lý ảo do Google phát triển) để làm phương tiện điều khiển các thiết bị trong nhà bằng giọng nói. Bên cạnh đó em sẽ sử dụng ứng dụng blynk chạy trên điện thoại thông minh để giúp người vận hành điều khiển thiết bị trong nhà qua điện thoại một cách thuận tiện và dễ dàng cho người sử dụng. Phần mềm này có thể nhận được thông tin về trạng thái hoạt động của các thiết bị, dữ liệu từ cảm biến, các thông báo từ hệ thống và hiển thị một cách trực quan, nhanh chóng, chỉ cần điện thoại và các thiết bị trong nhà được kết nối với internet. Điều này giúp người sử dụng có thể dễ dàng giám sát được tình trạng của các thiết bị khi không có mặt ở nhà. Trong số các thiết bị hoạt động trong nhà, có 2 thiết bị có thể hoạt động độc lập và tự động dựa theo thông tin về môi trường xung quanh mà các cảm biến thu thập được và 7 thiết bị có thể điều khiển qua internet hoặc công tắc trực tiếp.

1.3 NỘI DUNG THỰC HIỆN

- Nghiên cứu tổng quan về kit arduino, NodeMCU ngôn ngữ lập trình C++, UART.
- Nghiên cứu về các module cảm biến, điều khiển thiết bị. Cách thu thập dữ liệu và vận hành chúng.
- Tìm hiểu về ứng dụng blynk, cơ sở dữ liệu.
- Thiết kế hệ thống điều khiển, lưu đồ giải thuật và viết chương trình điều khiển mô hình hệ thống.
- Thiết kế và xây dựng mô hình thực tế.
- Tiến hành chạy thử nghiệm và chỉnh sửa các lỗi phát sinh mô hình thực tế.
- Viết báo cáo.

1.4 GIỚI HẠN

- Chỉ tập trung vào vấn đề điều khiển thiết bị qua app và giọng nói, chưa nhận diện giọng nói người dùng.
- Mô hình chỉ điều khiển các thiết bị trong nhà có công suất dưới 1000W, mô hình có thể điều khiển tối đa là 8 thiết bị.

- Một số vấn đề về bảo mật trong nhà như giám sát qua camera, nhận diện vân tay hay khuôn mặt chưa được phát triển.
- Hệ thống chỉ kết nối với internet qua wifi.

1.5 BỐ CỤC

- **Chương 1: Tổng quan**

Đặt vấn đề, dẫn nhập lý do chọn đề tài, trình bày mục tiêu, nội dung nghiên cứu, các giới hạn thông số và bố cục đề án.

- **Chương 2: Cơ sở lý thuyết**

Trình bày về các lý thuyết có liên quan về việc lập trình cho hệ thống, các linh kiện, phần cứng sử dụng để thực hiện đề tài.

- **Chương 3: Tính toán và thiết kế**

Trình bày về thiết kế và các tính toán xây dựng hệ thống như thiết kế sơ đồ khối hệ thống, sơ đồ nguyên lý toàn mạch, tính toán thiết kế mạch, tiến hành xây dựng hệ thống thực tế, kiểm tra và tinh chỉnh.

- **Chương 4: Kết quả thực hiện**

Trình bày về những kết quả mà em đã đạt được sau khi thực hiện đề tài, về hệ thống mà em thực hiện, đánh giá mức độ hoàn thiện, sự vận hành của hệ thống.

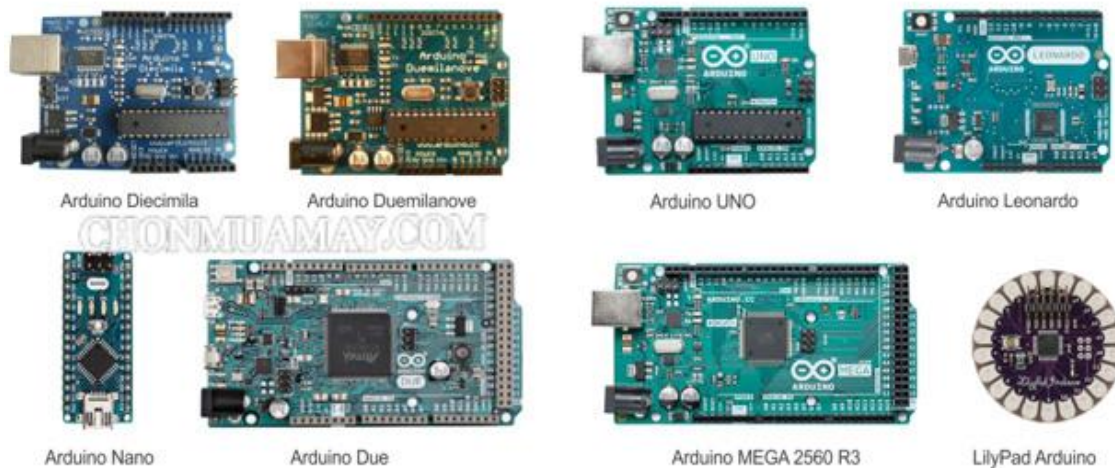
- **Chương 5: Kết luận và hướng phát triển**

Trình bày về những kết quả mà em đã đạt được sau khi thực hiện đề tài, những hạn chế, từ đó rút ra kết luận và hướng phát triển để giải quyết các vấn đề còn tồn đọng để đề án hoàn thiện hơn

CHƯƠNG 2

CƠ SỞ LÝ THUYẾT

2.1 TỔNG QUAN VỀ ARDUINO



Hình 2.1. Các loại Board Arduino

2.1.1. Giới thiệu

Arduino là một bo mạch vi xử lý được dùng để lập trình, tương tác với các thiết bị phần cứng như: cảm biến, động cơ, đèn hoặc các thiết bị khác. Đặc điểm nổi bật của arduino là môi trường phát triển ứng dụng rất dễ sử dụng, với ngôn ngữ lập trình có thể học một cách nhanh chóng ngay cả với người ít am hiểu về điện tử và lập trình cũng có thể sử dụng một cách dễ dàng. Arduino có mức giá thấp, phù hợp với nhu cầu người dùng, có tính chất nguồn mở và cộng đồng người dùng đông đảo. Với lợi thế đến từ giá thành cũng như lợi thế về cộng đồng người dùng, arduino đang ngày càng trở nên phổ biến hơn, người dùng arduino trải rộng từ học sinh phổ thông đến sinh viên đại học. Board mạch arduino được sử dụng để thực hiện nhiều ứng dụng như: điều khiển robot, điều khiển và giám sát nhiệt độ độ ẩm phòng thí nghiệm, điều khiển xe mô hình...

2.1.2. Phần cứng arduino

Phần cứng arduino bao gồm một board mạch nguồn mở được thiết kế trên nền tảng là vi xử lý AVR Atmel 8-bit, hoặc ARM Atmel 32-bit. Board arduino sẽ đưa ra hầu hết các chân I/O của vi điều khiển để sử dụng cho những mạch ngoài.

Những mẫu hiện tại thường được trang bị gồm 1 cổng giao tiếp USB, nhiều chân đầu vào analog và chân I/O kỹ thuật số tương thích với nhiều board mở rộng khác nhau. Điều này giúp người dùng dễ dàng lập trình và có thể mở rộng với các mạch khác, các module thêm vào có thể dễ dàng chuyển đổi, được gọi là shield. Một số shield kết nối với board arduino trực tiếp thông qua các chân khác nhau, ngoài ra còn một số shield được định địa chỉ thông qua serial bus I2C, người dùng có thể kết nối nhiều shield với arduino dưới dạng song song. Arduino thường sử dụng các dòng chip MegaAVR, đặc biệt là ATmega8, ATmega168, ATmega328, ATmega1280, và ATmega2560.

Theo nguyên tắc, khi sử dụng phần mềm arduino, tất cả các board được lập trình thông qua một kết nối RS-232, nhưng cách thức thực hiện lại tùy thuộc vào đời phần cứng. Các board serial arduino có chứa một mạch chuyển đổi giữa RS-232 sang TTL. Các board arduino hiện tại được lập trình thông qua cổng USB, thực hiện thông qua chip chuyển đổi USB-to-serial như là FTDI FT232.

2.1.3. Arduino mega 2560

Arduino mega 2560 sử dụng chip ATmega2560, có 54 chân digital I/O (trong đó có 15 chân điều chế độ rộng xung PWM), 16 chân đầu vào tương tự (Analog Inputs), 4 cổng UARTs (cổng nối tiếp phần cứng), một thạch anh dao động 16 MHz, kết nối USB, một jack cắm điện để cấp nguồn, một đầu ICSP và một nút reset. Arduino mega 2560 chứa tất cả mọi thứ cần thiết để hỗ trợ các vi điều khiển, người dùng chỉ đơn giản là kết nối nó với một máy tính bằng cáp USB hoặc với một bộ chuyển đổi điện AC-DC hoặc có thể sử dụng pin.

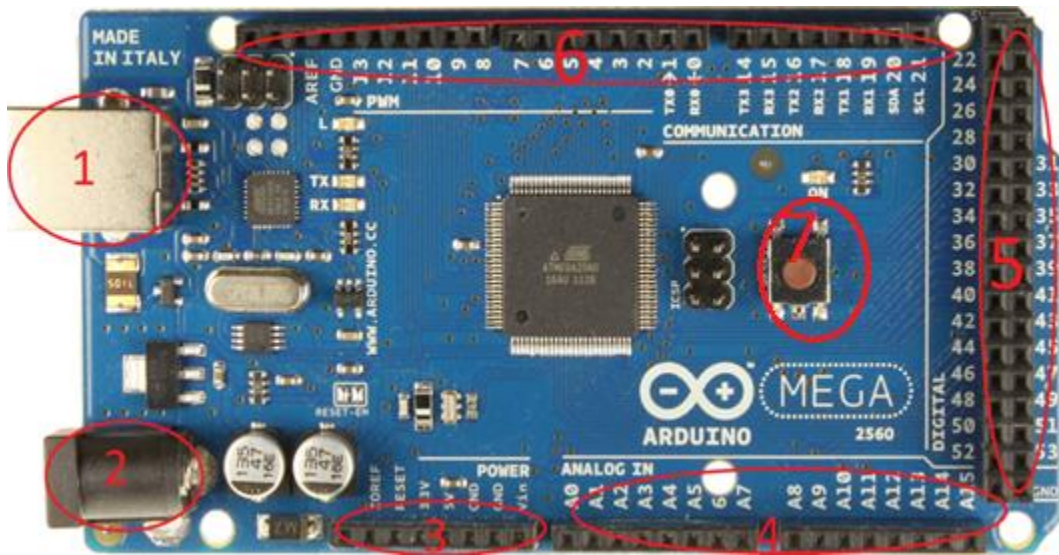
Board có khả năng tự động reset nhờ phần mềm thay vì đòi hỏi phải ấn nút reset trước khi tải lên. Phần mềm arduino sử dụng khả năng này để cho phép nạp code lên chỉ

cần nhấn vào nút Upload trong Arduino IDE. Điều này có nghĩa rằng bộ nạp khởi động có thể có một thời gian chờ ngắn hơn. Arduino mega 2560 có thể bảo vệ cổng USB của máy tính khi xảy ra hiện tượng quá dòng. Mặc dù hầu hết các máy tính đều có khả năng bảo vệ nội bộ, các cầu chì còn giúp tạo ra thêm một lớp bảo vệ. Nếu dòng cao hơn 500mA được áp dụng cho các cổng USB, cầu chì sẽ tự động phá vỡ các kết nối cho đến khi ngắt hoặc hiện tượng quá tải được khắc phục.

Thông số kỹ thuật arduino mega 2560 [1]:

- Chip vi điều khiển: ATmega2560.
- Điện áp cấp nguồn: 5V.
- Điện áp đầu vào (kiến nghị): 7-12V.
- Điện áp đầu vào (giới hạn): 6-20V.
- Số chân Digital I/O: 54 (có 15 chân điều chế độ rộng xung PWM).
- Số chân Analog (Input): 16.
- Dòng DC trên chân I/O: 40 mA.
- Dòng DC cho chân 3.3V: 50 mA.
- Flash Memory: 256KB trong đó có 8KB được sử dụng bởi bộ nạp khởi động (bootloader).
- SRAM: 8 KB.
- EEPROM: 4 KB.
- Xung nhịp: 16 MHz.
- Chiều dài: 101,52 mm.
- Chiều rộng: 53,3 mm.
- Cân nặng: 37 g.
- Sơ đồ các khối kết nối của arduino mega 2560

Trong hình 2.2 Bên dưới là hình ảnh sơ đồ các khối kết nối trên Arduino mega 2560:



Hình 2.2. Vị trí các khối kết nối trên arduino mega 2560

USB (1):

Arduino mega 2560 sử dụng cáp USB để giao tiếp với máy tính. Bằng việc sử dụng cáp USB, người dùng có thể upload chương trình cho arduino hoạt động, ngoài ra USB còn có thể dùng làm nguồn cho arduino.

Nguồn (2 và 3):

Khi không sử dụng USB làm nguồn thì người dùng có thể sử dụng nguồn ngoài thông qua jack cắm 2.1mm (cực dương ở giữa) hoặc có thể sử dụng 2 chân Vin và GND để cấp nguồn cho arduino. Bo mạch hoạt động với nguồn ngoài ở điện áp từ 5 – 20 Volt. Người dùng có thể cấp một áp lớn hơn tuy nhiên chân 5V sẽ có mức điện áp lớn hơn 5 Volt. Và nếu sử dụng nguồn lớn hơn 12 Volt thì sẽ có hiện tượng nóng và dễ dẫn đến làm hỏng bo mạch. Chân 5V và chân 3.3V (Output voltage): các chân này dùng để lấy nguồn ra từ nguồn mà chúng ta đã cung cấp cho arduino để cấp cho các thiết bị giao tiếp khác.

Lưu ý: không được cấp nguồn vào các chân này vì sẽ làm hỏng arduino.

Ngõ vào tương tự (4):

Arduino mega 2560 có 16 ngõ vào tương tự (các chân từ A0 đến A15), mỗi ngõ vào này đều có độ phân giải 10 bit (1024 giá trị). Mặc định đo từ 0 đến 5V, có thể thiết

lập giá trị bằng cách điều chỉnh chân AREF và sử dụng hàm Analog Referency() để chuyển đổi.

Ngõ vào số (5 và 6):

Mỗi một chân trong 54 chân số của board đều có thể sử dụng như một ngõ vào hoặc ngõ ra. Mỗi chân có thể cung cấp hoặc nhận được tối đa 40 mA và có một điện trở kéo lên bên trong 20-50 kOhms. Ngoài ra, một số chân có chức năng đặc biệt:

- **Serial:** để truyền và nhận dữ liệu nối tiếp. Gồm các chân: Serial 0: 0 (RX) và 1 (TX); Serial 1: 19 (RX) và 18 (TX); Serial 2: 17 (RX) và 16 (TX); Serial 3: 15 (RX) và 14 (TX).
- **External Interrupt:** Các chân này có thể được cấu hình để kích hoạt ngắt mức thấp, ngắt cạnh lên hoặc xuống. Gồm các chân: 2 (interrupt 0), 3 (interrupt 1), 18(interrupt 5), 19 (interrupt 4), 20 (interrupt 3), 21 (interrupt 2).
- **PWM:** Cung cấp ngõ ra PWM 8 bit. Gồm các chân từ chân 2 đến 13 và 44 đến 46.
- **SPI:** Các chân này hỗ trợ giao tiếp SPI bằng cách sử dụng thư viện SPI. Có các chân: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS).
- **TWI:** Hỗ trợ giao tiếp TWI bằng việc sử dụng thư viện WIRE. Có các chân: 20 (SDA) và 21 (SCL).
- **Nút Retset (7):** Đây là nút nhấn để người dùng có thể thiết lập lại vi điều khiển.

2.1.4. Arduino uno r3

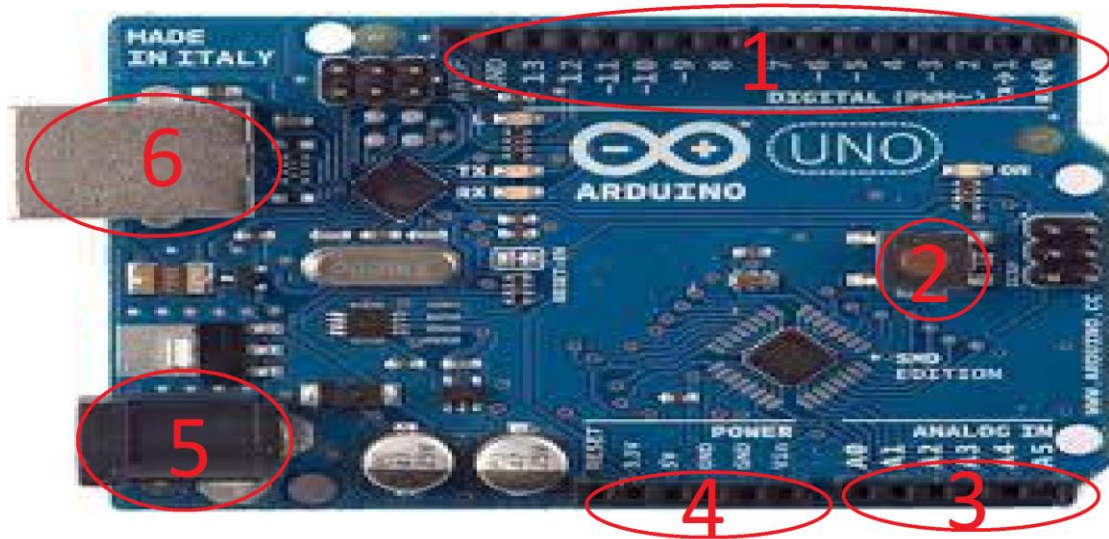
Arduino uno r3 là dòng mạch arduino phổ biến, với thiết kế tiêu chuẩn sử dụng vi điều khiển ATmega328 hoặc ATmega168 (bộ nhớ flash 16KB) đối với loại có giá thành thấp hơn. Arduino uno r3 sử dụng thạch anh có chu kỳ dao động là 16 MHz, có 14 pin (ngõ) ra/vào được đánh số từ 0 tới 13 trong đó có 6 pin PWM, ngoài ra còn có thêm 6 pin nhận tín hiệu analog được đánh kí hiệu từ A0 - A5, 6 pin này cũng có thể sử dụng được như các pin ra/vào bình thường.

Trên board arduino uno r3 còn có 1 nút reset, 1 ngõ kết nối với máy tính qua cổng USB và 1 ngõ cấp nguồn sử dụng jack 2.1mm lấy năng lượng trực tiếp từ ACDC adapter hay thông qua ắc-quy nguồn.

Thông số kỹ thuật của arduino uno r3 [2]:

- Vi điều khiển: ATmega328P.
- Điện áp hoạt động: 5V.
- Điện áp vào khuyến dùng: 7-12V.
- Điện áp vào giới hạn: 6-20V.
- Chân digital I/O: 14.
- Chân PWM Ddigital I/O: 6.
- Chân analog : 6.
- Cường độ dòng điện trên mỗi chân I/O: 20 mA.
- Cường độ dòng điện trên mỗi chân 3.3V: 50 mA.
- Flash Memory: 32 KB (đối với ATmega328P).
- SRAM: 2 KB (ATmega328P)
- EEPROM: 1 KB (ATmega328P)
- Tốc độ: 16 MHz

Sơ đồ các khối kết nối của arduino uno r3:



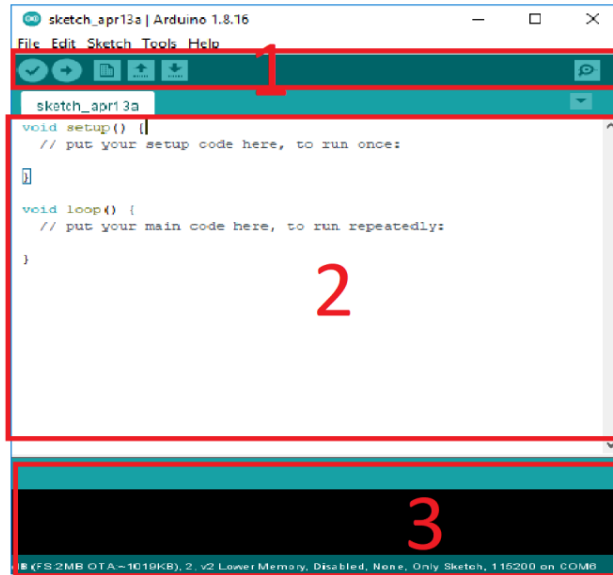
Hình 2.3 Vị trí các khối kết nối trên arduino uno r3

Các khối kết nối trên arduino uno r3 cũng có chức năng tương tự như trên arduino mega 2560 đã trình bày ở trên, bao gồm:

1. Ngõ vào số
2. Nút reset
3. Ngõ vào tương tự
4. Nguồn vào và ra
5. Nguồn vào
6. Cổng USB

2.1.5. Phần mềm lập trình arduino IDE

Giao diện phần mềm lập trình arduino IDE bao gồm 3 phần chính như hình dưới đây:



Hình 2.4 Giao diện phần mềm lập trình arduino IDE

1. Vùng lệnh: Bao gồm các nút lệnh (File, Edit, Sketch, Tools, Help). Phía dưới là các biểu tượng cho phép sử dụng nhanh các chức năng thường dùng. Chức năng lần lượt của các biểu tượng được trình bày trong hình dưới đây:

- ✓ Xác minh biên dịch mã. Sử dụng để kiểm tra mã của bạn để tìm lỗi trước khi tải lên ký họa.
- ↶ Tải lên một bản phác thảo.
- 📄 Cửa sổ trình chỉnh sửa mới sẽ mở ra một cửa sổ chỉnh sửa mã mới nằm trong vị trí của một cửa sổ hiện tại.
- 📁 Mở một tập tin.
- ⬇️ Lưu một bản phác thảo.
- 🔊 Màn hình nổi tiếp mở ra màn hình nổi tiếp, hữu ích để gỡ lỗi
- ⌵ Mũi tên xuống cung cấp cho bạn các tùy chọn như thêm một bản phác thảo cho dự án hiện tại. Nó sẽ mở ra dưới dạng tab mới trong trình chỉnh sửa mã hiện tại, điều đó rất hữu ích để tổ chức mã của bạn thành các tệp logic.

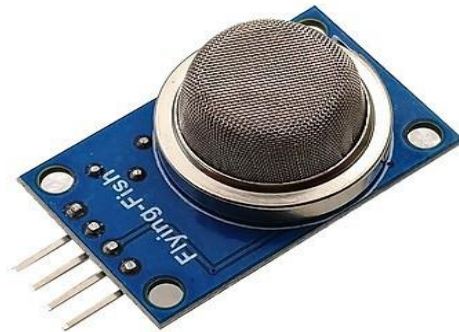
Hình 2.5 Chức năng các biểu tượng trong arduino IDE

2. Vùng viết chương trình: Là nơi để người dùng viết chương trình, phần mềm arduino IDE sử dụng ngôn ngữ C/C++ để lập trình cho arduino.
3. Vùng thông báo: Những thông báo từ IDE sẽ được hiển thị tại vùng thông báo. Ở dưới cùng bên phải hiển thị loại board arduino và cổng COM đang được sử dụng.

2.2 MODULE CẢM BIẾN KHÍ GA MQ2

Module cảm biến khí ga MQ2 là một trong những loại cảm biến được sử dụng để nhận biết nhiều loại khí như: Butan, Propane, Methane, Alcohol, khói và khí gas. MQ2 là cảm biến có độ nhạy cao, thời gian đáp ứng nhanh. Bộ phận cảm nhận của cảm biến khí MQ2 là SnO₂, có độ dẫn điện thấp trong không khí sạch. Khi phát hiện các khí dễ cháy, độ dẫn điện của cảm biến sẽ tăng khi nồng độ khí tăng.

Module MQ2 bao gồm cảm biến khí gas MQ2, một chiết áp điều chỉnh độ nhạy, LM393, đèn LED báo hiệu. Dưới đây là hình ảnh của mô đun cảm biến khí gas MQ2:



Hình 2.6 Cảm biến khí ga MQ2

Thông số kỹ thuật [3]:

- Nguồn cung cấp: 4.5V đến 5V DC
- Độ nhạy cao với khí Propane, khói và Butan
- Hỗ trợ đầu ra analog và digital
- Thời gian đáp ứng: $\leq 10s$
- Trở kháng khi nóng: $31\Omega \pm 3\Omega$
- Dòng tiêu thụ khi nóng: $\leq 180mA$
- Tích hợp LED báo động trên module

- Giao tiếp đơn giản với 4 chân VCC GND Analog và Digital. Người dùng có thể lựa chọn việc sử dụng chỉ một trong 2 chân analog và digital để lấy dữ liệu từ cảm biến.

2.3 MODULE CẢM BIẾN CHUYỂN ĐỘNG PIR HC – SR501

HC-SR501 là một module cảm chuyển động dựa trên việc thu thập tia hồng ngoại từ môi trường xung quanh, HC-SR501 có độ nhạy và độ tin cậy cao, hoạt động tiêu tốn ít năng lượng, được sử dụng rộng rãi trong các thiết bị điện cảm biến tự động khác nhau. Cảm biến PIR (Passive InfraRed) có thể phát hiện chuyển động bằng cách đo những thay đổi trong bức xạ hồng ngoại phát ra bởi các đối tượng. Khi phát hiện chuyển động cảm biến PIR sẽ xuất ra 1 xung ở mức cao, xung này được đọc bởi vi điều khiển để thực hiện chức năng mong muốn. Tín hiệu ngõ ra Digital, đọc trực tiếp trên chân vi điều khiển. Ngoài ra còn có 2 biến trở điều chỉnh độ nhạy chuyển động và thời gian giữ trạng thái kích hoạt của cảm biến.

Hoạt động của cảm biến: khi có người đi vào vùng hoạt động của cảm biến, điện áp chân ra tự động đặt lên 3.3V, sau khi người đi ra khỏi vùng hoạt động, sau khi hết khoảng thời gian trễ (có thể đặt bằng biến trở) điện áp ra trở về 0.

Hình 2.7 sau đây là hình ảnh của module cảm biến HC SR501:



Hình 2.7 Cảm biến chuyển động PIR HC – SR501

Thông số kỹ thuật [4]:

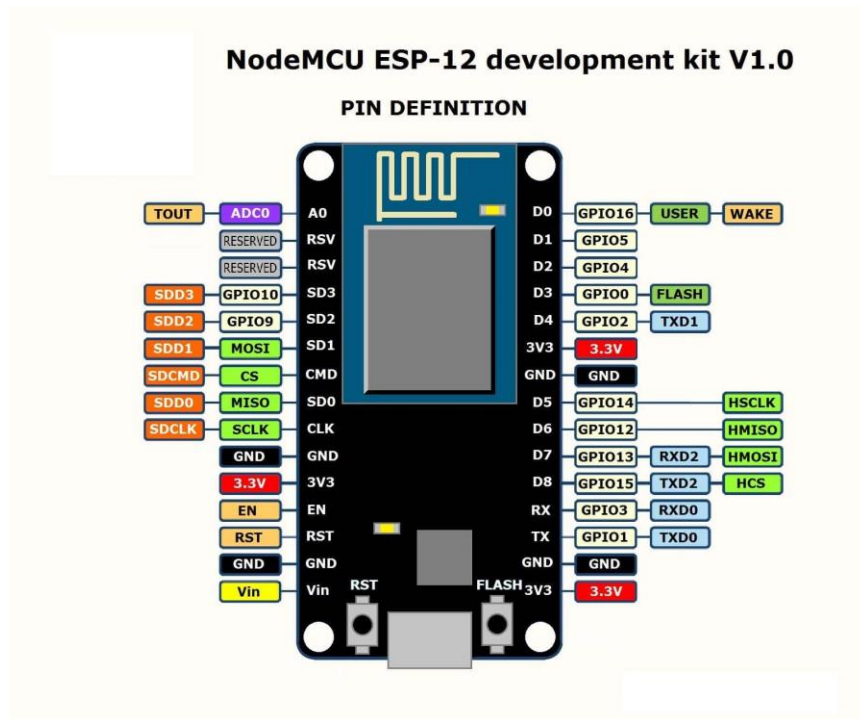
- Điện áp hoạt động: 5VDC.
- Dòng nghỉ: < 50uA.

- Nhiệt độ hoạt động: -15°C đến 70°C .
- Góc quét: 120 độ.
- Tầm quét xa: 7m.
- Thời gian trễ: điều chỉnh bằng biến trở, từ 5 giây đến 200 giây.
- Điện áp ra: 0 hoặc 3.3V.
- Chế độ kích hoạt:
 - Chế độ L: Điện áp ra tự động chuyển về 0 khi hết thời gian trễ.
 - Chế độ H (mặc định): Điện áp ra tự động giữ nguyên 3.3V cho đến khi không còn chuyển động.

2.4 MODULE WIFI ESP8266 NODEMCU

ESP8266 là vi điều khiển tích hợp WiFi (WiFi SoC) được phát triển bởi Espressif Systems. Với Vi điều khiển và WiFi tích hợp, ESP8266 cho phép lập trình viên có thể lập trình trên mô đun này để thực hiện các ứng dụng khác nhau, đặc biệt là các ứng dụng IoT.

NodeMCU phát triển dựa trên Chip WiFi ESP8266EX bên trong. Board còn tích hợp IC CP2102, giúp dễ dàng giao tiếp với máy tính thông qua Micro USB để thao tác với board. Và có sẵn nút nhấn, led báo hiệu.



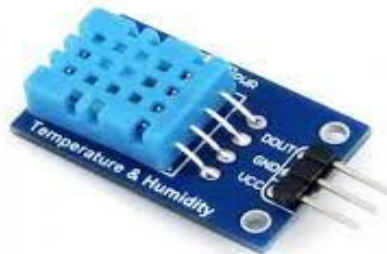
Hình 2.8 Sơ đồ chân module NodeMCU

Thông số kỹ thuật:

- Chip: ESP8266EX
- WiFi: 2.4 GHz hỗ trợ chuẩn 802.11 b/g/n
- Điện áp hoạt động: 3.3V
- Điện áp vào: 5V thông qua cổng USB
- Số chân I/O: 11 (tất cả các chân I/O đều có Interrupt/PWM/I2C/Onewire, trừ chân D0)
- Số chân Analog Input: 1 (điện áp vào tối đa 3.3V)
- Bộ nhớ Flash: 4MB
- Giao tiếp: Cáp Micro USB
- Hỗ trợ bảo mật: WPA/WPA2
- Tích hợp giao thức TCP/IP

2.5 MODULE CẢM BIẾN NHIỆT ĐỘ - ĐỘ ẨM DHT11

Module DHT11 là một cảm biến vừa có chức năng đo nhiệt độ và độ ẩm. Cảm biến DHT11 có hai phần, một cảm biến độ ẩm điện dung và một điện trở nhiệt. Dữ liệu ngõ ra của cảm biến DHT là dạng số. Dữ liệu nhiệt độ đo được trong khoảng 0 đến 50°C độ ẩm mà cảm biến đo được ở mức 20% ~ 90%.



Hình 2.9 Cảm biến nhiệt độ - độ ẩm DHT11

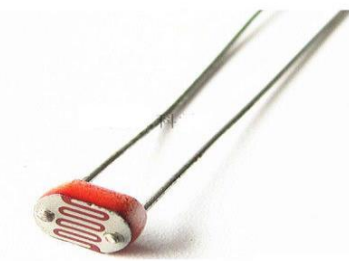
Cảm biến DHT11 có 4 chân là: VCC, DATA, NC, GND. Tuy nhiên module DHT11 thì đã được gắn sẵn điện trở và led báo nguồn, nên có 3 chân: VCC: Chân cấp nguồn 3.3 - 5.5VDC, DATA: chân dữ liệu, GND: chân nối đất, cực âm.

Thông số kỹ thuật :

- Điện áp hoạt động: 3,3V ~ 5,5 V
- Dòng điện lớn nhất sử dụng: 2.5mA (khi truyền dữ liệu).
- Nhiệt độ: 0°C ~ 50°C
- Độ ẩm: 60% rh trở xuống
- Đo ở độ ẩm từ 20-90% với sai số 5%.
- Đo tốt ở nhiệt độ 0 to 50°C sai số $\pm 2^{\circ}\text{C}$.
- Độ phân giải: 1°C
- Tần số lấy mẫu tối đa 1Hz

2.6 QUANG TRỞ

Điện trở quang hay quang trở, photoresistor, photocell, LDR, là linh kiện điện tử chế tạo bằng chất đặc biệt có điện trở thay đổi tăng hoặc giảm theo cường độ ánh sáng chiếu vào. Điện trở của quang trở càng tăng khi lượng ánh sáng nhận được càng ít và ngược lại, khi đặt ở nơi càng nhiều ánh sáng thì điện trở của quang trở càng nhỏ. Điện trở của quang trở có thể thay đổi từ vài trăm Ohm đến 1 Mega Ohm. Với những đặc điểm như vậy, quang trở là linh kiện rất thích hợp cho các mạch hoạt động dựa trên sự thay đổi của ánh sáng.



Hình 2.10 Quang trở

2.7 NGUỒN XUNG 12V-5A

Nguồn xung (hay còn gọi là nguồn tổ ong) là bộ nguồn có tác dụng biến đổi dòng điện xoay chiều (220VAC) sang nguồn điện một chiều. Mạch nguồn sử dụng nguyên lý dao động xung kết hợp với biến áp xung. Với các ưu điểm như giá thành rẻ, hiệu suất chuyển đổi cao và gọn nhẹ, ngày nay nguồn xung được sử dụng rất nhiều để cấp điện áp hoạt động cho các thiết bị điện tử, đèn LED. Tuy nhiên nguồn xung tồn tại một nhược điểm lớn là luôn tạo ra sóng hài làm ảnh hưởng xấu đến dòng điện dẫn đến điện áp ở ngõ ra không ổn định, khoảng dao động lớn.



Hình 2.11 nguồn xung

Thông số kỹ thuật:

- Điện áp ngõ vào: 220VAC.
- Điện áp ngõ ra: 12VDC.
- Dòng điện ngõ ra tối đa: 5A.

2.8 BÀN PHÍM MA TRẬN 4X4

Bàn phím ma trận là một "thiết bị nhập" chứa các nút nhấn cho phép người dùng nhập các chữ số, chữ cái hoặc ký hiệu vào bộ điều khiển. Keypad không chứa tất cả bảng mã ASCII như keyboard và vì thế keypad thường được tìm thấy trong các thiết bị chuyên dụng. Các nút nhấn trên các máy tính điện tử cầm tay là một ví dụ về keypad. Số lượng nút nhấn của một keypad thay đổi phụ thuộc vào yêu cầu ứng dụng. Gọi là keypad 4x4 vì keypad này có 16 nút nhấn được bố trí dạng ma trận 4 hàng và 4 cột. Cách bố trí ma trận hàng và cột là cách chung mà các keypad sử dụng. Cũng giống như các ma trận LED, các nút nhấn cùng hàng và cùng cột được nối với nhau, vì thế với keypad 4x4 sẽ có tổng cộng 8 ngõ ra (4 hàng và 4 cột).



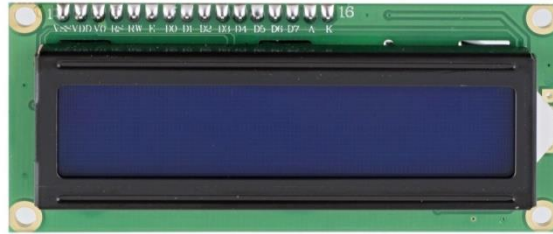
Hình 2.12 Bàn phím ma trận 4x4

Thông số kỹ thuật của Keypad 4x4:

- Module bàn phím ma trận 4x4 loại phím mềm
- Điện áp hoạt động : 3,3V-5V
- Tuổi thọ hoạt động: khoảng 2000000 lần nhấn phím
- Thời gian phản hồi: $\leq 5\text{ms}$
- Độ dài cáp: 88m.
- Nhiệt độ hoạt động: 0~70°C
- Đầu nối ra 8 chân
- Kích thước bàn phím 77 x 69mm

2.9 MÀN HÌNH LCD 1602

LCD1602 là một màn hình tinh thể lỏng dùng để hiển thị dữ liệu, màn hình có 16 cột và 2 dòng, hiển thị tối đa 32 ký tự.



Hình 2.13 màn hình LCD 1602

Các chân kết nối của màn hình LCD được thể hiện qua bảng sau:

Thứ tự	Tên kí hiệu	I/O	Mô tả
1	Vss	power	GND
2	Vdd	Power	+5V
3	V ₀	Analog	Điều khiển ánh sáng nền
4	RS	Input	Register Select
5	R/W	Input	Read/Write
6	E	Input	Enable(Storage)
7	D0	I/O	Data LSB
8	D1	I/O	Data
9	D2	I/O	Data
10	D3	I/O	Data
11	D4	I/O	Data
12	D5	I/O	Data
13	D6	I/O	Data
14	D7	I/O	Data
15	A	I	Nguồn dương 5V
16	K	I	GND

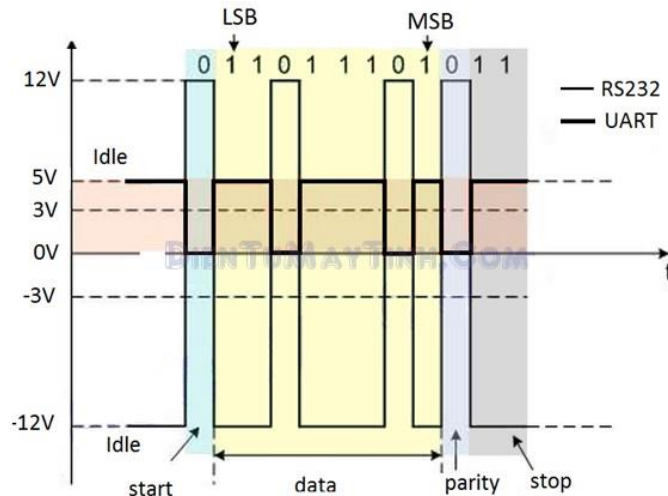
Bảng 2.1 Các chân của LCD

- Chân cấp nguồn: VSS nối mass (0V), VDD (nối nguồn +5V), V₀ (điều chỉnh độ trong phần thường nối với biến trở).

- RS: Chân chọn thanh ghi (Register select). LCD sẽ dựa trên mức logic của chân RS để chọn thanh ghi.:
 - o Logic “0”: Bus DB0-DB7 sẽ nối với thanh ghi lệnh IR của LCD (ở chế độ “ghi” - write) hoặc nối với bộ đếm địa chỉ của LCD (ở chế độ “đọc” - read).
 - o Logic “1”: Bus DB0-DB7 sẽ nối với thanh ghi dữ liệu DR bên trong LCD.
- R/W: Chân chọn chế độ đọc/ghi (Read/Write). Nối chân R/W với logic “0” để LCD hoạt động ở chế độ ghi, hoặc nối với logic “1” để LCD ở chế độ đọc.
- E: Chân cho phép chốt xung kí tự (Enable). Sau khi các tín hiệu được đặt lên bus DB0-DB7, các lệnh chỉ được chấp nhận khi có 1 xung cho phép của chân E. Ở chế độ ghi: Dữ liệu ở bus sẽ được LCD chuyển vào (chấp nhận) thanh ghi bên trong nó khi phát hiện một xung (high-to-low transition) của tín hiệu chân E. Ở chế độ đọc: Dữ liệu sẽ được LCD xuất ra DB0-DB7 khi phát hiện cạnh lên (low-to-high transition) ở chân E và được LCD giữ ở bus đến khi nào chân E xuống mức thấp.
- D0-D7: Chân dữ liệu dùng để trao đổi dữ liệu giữa thiết bị điều khiển và LCD.
- A, K: Chân điều khiển đèn nền.

2.10 LÝ THUYẾT CHUẨN GIAO TIẾP UART

UART: Viết tắt là Universal Synchronous & Asynchronous serial Receiver and Transmitter nghĩa là bộ truyền nhận nối tiếp đồng bộ và không đồng bộ. UART không hẳn là một chuẩn truyền nhận, nó chỉ được coi như là một phần cứng. Để tạo thành một chuẩn giao tiếp cần sử dụng với bộ chuyển đổi mức điện áp để tạo thành một chuẩn giao tiếp, như RS232 là chuẩn +12, -12 còn UART thường thì dùng mức logic 5V – 0V.



Hình 2.14 Tín hiệu tương đương của UART và RS232.

Baud rate (tốc độ Baud): Khi truyền không đồng bộ đến hai module, để truyền nhận dữ liệu được chính xác thì cần quy định một khoảng thời gian cho 1 bit truyền nhận, nghĩa là trước khi truyền thì tốc độ phải được cài đặt đầu tiên. Theo định nghĩa thì tốc độ baud là số bit truyền trong một giây.

Frame (khung truyền): Do kiểu truyền thông nối tiếp này rất dễ mất dữ liệu nên ngoài tốc độ, khung truyền cũng được cài đặt từ ban đầu để tránh bớt sự mất mát dữ liệu này. Khung truyền quy định số bit trong mỗi lần truyền, các bit báo như start, stop, các bit kiểm tra như parity, và số bit trong một data.

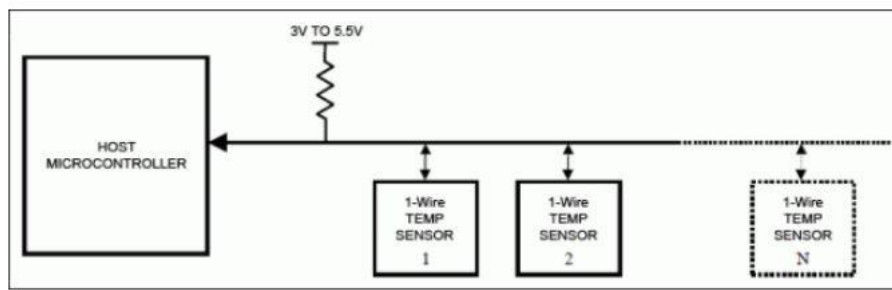
Bit Start: Là bit bắt đầu trong khung truyền Bit này nhằm mục đích báo cho thiết bị nhận biết quá trình truyền bắt đầu, trên AVR bit Start có trạng thái là 0.

Data: Dữ liệu cần truyền Data không nhất thiết phải 8 bit. có thể là 5, 6, 7, 8, 9. Trong UART bit LSB được truyền đi trước, bit MSB được truyền đi sau.

Parity bit: Là bit kiểm tra dữ liệu đúng không. Có 2 loại parity: chẵn (even parity), lẻ (old parity). Parity chẵn là bit parity thêm vào để số số 1 trong data + parity = chẵn.

2.11 LÝ THUYẾT CHUẨN GIAO TIẾP ONE-WIRE

Chuẩn giao tiếp một dây (one-wire) do hãng Dallas giới thiệu. Trong chuẩn giao tiếp này chỉ cần một dây để truyền tín hiệu và làm nguồn nuôi (nếu không tính dây mass). Đây là chuẩn giao tiếp không đồng bộ và bán song công(half-duplex). Trong giao tiếp này tuân theo mô hình liên hệ chủ tớ một cách chặt chẽ. Trên một bus có thể gắn một hoặc nhiều thiết bị slave. Nhưng chỉ có một master có thể kết nối đến bus này. Bus dữ liệu khi ở trạng thái rảnh (khi không có dữ liệu trên đường truyền) phải ở mức cao do vậy bus dữ liệu phải được kéo lên nguồn thông qua một điện trở. Giá trị điện trở này có thể tham khảo trong datasheet của thiết bị / các thiết bị slave.



Hình 2.15 Sơ đồ khối kết nối của One-Wire.

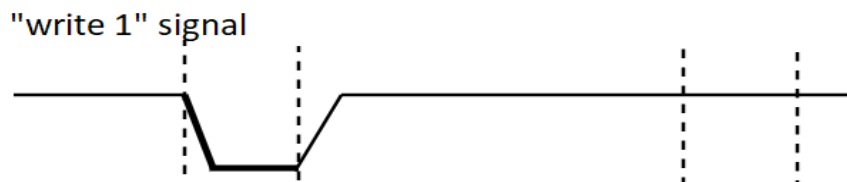
Các thiết bị tớ (slave) kết nối với cùng một bus được phân biệt với nhau nhờ 64 bit địa chỉ duy nhất (64-bit serial number). Tám byte (64 bit) này và được chia làm ba phần chính:

Bắt đầu với LSB, đây là byte đầu tiên là mã họ thiết bị có độ lớn 8 bit (8-bit family codes) xác định kiểu thiết bị, 6 byte tiếp theo lưu trữ địa chỉ riêng của thiết bị. Byte cuối cùng (MSB) là byte kiểm tra tính toàn vẹn dữ liệu cyclic redundancy check (CRC) có giá trị tương ứng với giá trị của 7 byte đầu tiên. Nhờ byte CRC giúp cho master xác định có địa chỉ được đọc có bị lỗi hay không. Với 2²⁴ địa chỉ khác nhau tạo ra một số lượng rất lớn các địa chỉ. Do vậy vấn đề về địa chỉ không phải là vấn đề chính trong chuẩn giao tiếp này.

Cách thức hoạt động :

Bốn thao tác hoạt động cơ bản của bus 1 wire là Reset/Presence, gửi bit 1, gửi bit 0, và đọc bit. Thao tác byte như gửi byte và đọc byte dựa trên thao tác từng bit.

- Gửi bit 1 ("Write 1" signal): thiết bị master kéo bus xuống mức thấp trong khoảng 1 đến 15 μ s. Sau đó nhả bus (releases the bus) cho đến hết phần còn lại của khe thời gian.



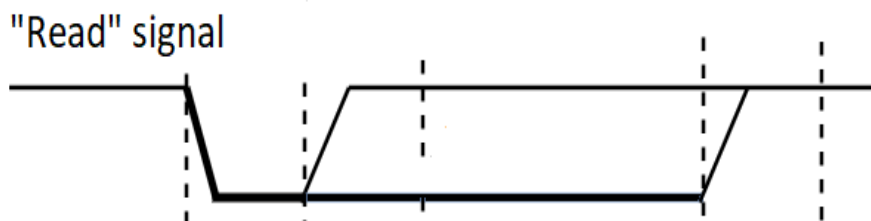
Hình 2.16 Dạng sóng gửi bit 1 của chuẩn One-Wire

- Gửi bit 0 ("Write 0" signal): kéo bus xuống mức thấp trong ít nhất 60 μ s, với chiều dài tối đa là 120 μ s.



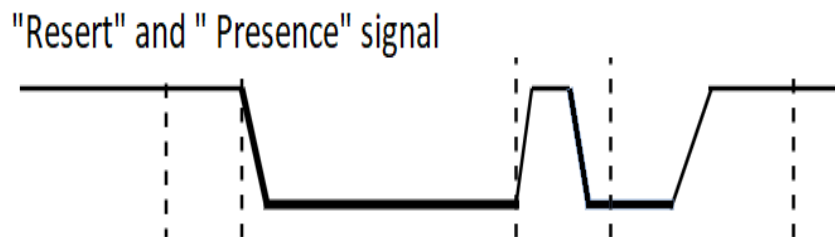
Hình 2.17 Dạng sóng gửi bit 0 của chuẩn One-Wire

- Đọc bit: Thiết bị master kéo bus xuống mức thấp từ 0 -15 μ s. Khi đó thiết bị tớ khi đó sẽ giữ bus ở mức thấp nếu muốn gửi bit 0, nếu muốn gửi bit 1 đơn giản là nhả bus. Bus nên lấy mẫu 15 μ s sau khi bus kéo xuống mức thấp



Hình 2.18 Dạng sóng đọc bit của chuẩn One-Wire

- "Reset/Presence": Tín hiệu Reset và Presence(Báo hiện diện) được trình bày như hình bên dưới. Thiết bị master kéo bus xuống thấp ít nhất 8 khe thời gian (tức là $480 \mu\text{s}$) và sau đó nhả bus. Khoảng thời gian bus ở mức thấp đó gọi là tín hiệu reset. Nếu có thiết bị slave gắn trên bus nó sẽ trả lời bằng tín hiệu Presence tức là thiết bị đó sẽ kéo bus xuống mức thấp trong khoảng thời gian $60\mu\text{s}$. Nếu không có tín hiệu Presence, thiết bị master sẽ hiểu rằng không có thiết bị slave nào trên bus, và các giao tiếp tiếp theo sẽ không thể diễn ra.



Hình 2.19 Dạng sóng Reset/Presert của chuẩn One-Wire

CHƯƠNG 3

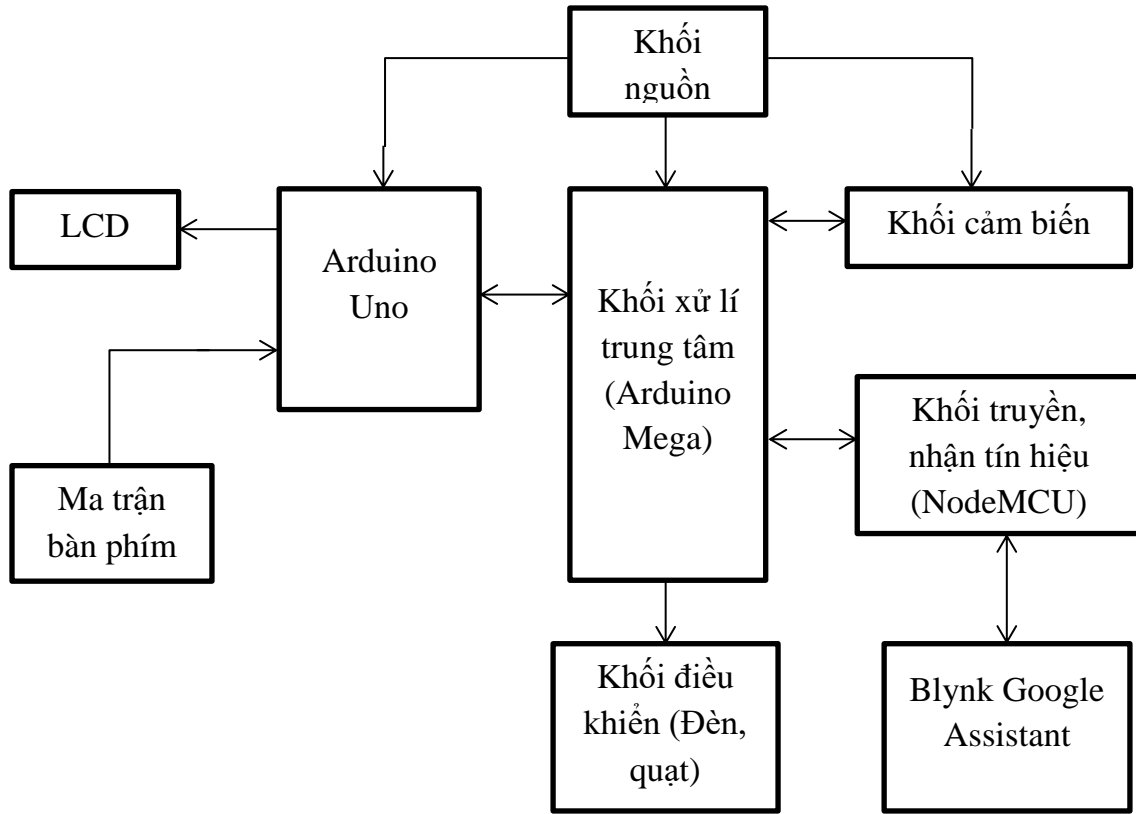
TÍNH TOÁN THIẾT KẾ

3.1 GIỚI THIỆU

Mô hình bao gồm cảm biến nhiệt độ, độ ẩm, nồng độ khí ga, cảm biến chuyển động, cảm biến ánh sáng, các bộ vi xử lý(arduino uno, arduino mega), các module truyền nhận tín hiệu không dây như NodeMCU để điều khiển thiết bị trong nhà, giám sát các thông số của cảm biến cũng như trạng thái của thiết bị thông qua các thiết bị ngoại vi như điện thoại và máy tính mà trực tiếp là qua app Blynk. Ngoài ra em thực hiện sử dụng Google Assistant để điều khiển bật tắt từ các bóng đèn và quạt trong căn nhà bằng giọng nói thông qua module 8 relay 5 VCD. Các thành phần của hệ thống được kết nối, giao tiếp với nhau thông qua các chuẩn truyền thông UART, One-Wire, Wifi tạo lên một hệ thống chặt chẽ và đáp ứng kịp thời, chính xác.

3.2 TÍNH TOÁN VÀ THIẾT KẾ HỆ THỐNG

3.2.1 Thiết kế sơ đồ khối hệ thống



Hình 3.1 Sơ đồ khối của hệ thống

Khối cảm biến

- Cảm biến DHT11 có chức năng thu thập thông số nhiệt độ, độ ẩm trong vùng kiểm soát và gửi về cho module arduino mega.
- Cảm biến MQ2 thu thập thông số nồng độ khí gas trong vùng kiểm soát và gửi về cho module arduino mega.
- Cảm biến PIR cũng thu thập thông số về chuyển động của các đối tượng trong vùng kiểm soát và gửi về cho module arduino mega.
- Quang trở đóng vai trò là một cảm biến ánh sáng các tác dụng thu thập dữ liệu về cường độ ánh sáng của môi trường và gửi về module arduino mega.

Khối xử lý trung tâm

- Module xử lý chính arduino mega sẽ gửi chuỗi gồm các thông số của các cảm biến qua module NodeMCu bằng giao thức truyền UART để cập nhật thông tin lên app điện thoại Blynk.
- Thực hiện đồng thời với quá trình cập nhật thông tin lên Blynk, NodeMCU sẽ cập nhật các tác vụ điều khiển Blynk qua Wifi và gửi lại cho module xử lý chính arduino mega qua giao thức truyền UART để thực hiện tác vụ điều khiển.
- Dựa vào tín hiệu nhận được từ arduino uno, module xử lý chính arduino mega sẽ xử lý và đáp ứng kịp thời các yêu cầu từ người dùng.

Khối truyền nhận dữ liệu:

- Bao gồm các thiết bị ngoại vi như điện thoại và máy tính chính xác hơn là ứng dụng Blynk để hiển thị các giá trị của cảm biến cũng như điều khiển thiết bị qua các ứng dụng trên thông qua Wifi.
- Các dữ liệu từ thiết bị ngoại vi được nhận từ NodeMCU qua Wifi để giám sát và đồng thời gửi dữ liệu cho NodeMCU để điều khiển.

Khối điều khiển: Em thực hiện sử dụng các relay để tắt mở các thiết bị trong nhà như đèn, quạt.

Khối nguồn: Em thực hiện sử dụng nguồn xung 12V 5A để cấp nguồn cho các thiết bị sử dụng IC LM256 để tạo nguồn 5V cấp cho các module , cảm biến và cảm biến trong toàn bộ hệ thống với nguồn là 5V.

Hệ thống giám sát này có thể gắn trực tiếp vào trong nhà, căn hộ, những nơi cần thiết, Người dùng sẽ nắm bắt được tình hình các thông số nhiệt độ, độ ẩm và nồng độ khí gas, chuyển động bất cứ lúc nào kết nối Wifi cùng với một chiếc điện thoại smartpone, có thể bật tắt thiết bị theo nhu cầu của người dùng .

3.2.2 Tính toán và thiết kế mạch

3.2.2.1 Thiết kế khối cảm biến

Khối cảm biến gồm các module cảm biến khí gas MQ2, module cảm biến chuyển động PIR, module cảm biến DHT11 và quang trở, tất cả đều được kết nối với khối điều khiển trung tâm là arduino mega để đọc dữ liệu.

Cảm biến nhiệt độ DHT11 được lựa chọn để sử dụng trong thiết bị này vì giá thành rẻ, độ chính xác tương đối thích hợp cho ứng dụng nhiệt độ trong phòng. Giao tiếp với điều khiển thông qua chuẩn one-wire, đặc biệt có bộ tiền xử lý tín hiệu của cảm biến. do đó dễ dàng đọc dữ liệu cảm biến. Dữ liệu ngõ ra của cảm biến DHT11 là dạng số dễ dàng cho việc lập trình. Nhiệt độ có thể đo được trong khoảng 0 đến 50°C, sai số $\pm 2^\circ\text{C}$. Độ ẩm trong khoảng 20 đến 90% sai số 5%. Tiêu tốn năng lượng rất thấp, điện áp hoạt động là 3-5V, dòng 2.5mA phù hợp với nguồn cấp cho toàn mạch.

MQ2 là một trong những loại cảm biến được sử dụng để nhận biết được nhiều loại khí như: butan, Propane, Methane, Alcohol, khói và khí ga. Được thiết kế với độ nhạy cao, thời gian đáp ứng nhanh, nguồn cung cấp: 4,5V đến 5V DC phù hợp với thiết kế mô hình, tuổi thọ cao, giá thành thấp, thời gian đáp ứng: $\leq 10\text{s}$, dòng tiêu thụ khí nóng là 100mA.

Để bật tắt các thiết bị một cách tự động như đèn nhà tắm, đèn ngoài trời tự động em thực hiện sử dụng thêm cảm biến chuyển động và quang trở vì các tính năng của chúng.

PIR (Passive Infrared) được em thực hiện lựa chọn vì có thể phát hiện chuyển động bằng cách đo những thay đổi trong bức xạ hồng ngoại phát ra bởi các đối tượng, kích thước nhỏ gọn phù hợp với mô hình. Khi phát hiện chuyển động thì các cảm biến PIR sẽ xuất ra 1 xung ở mức cao, xung này được đọc bởi vì điều khiển để thực hiện chức năng mong muốn. Ngoài ra trên module PIR còn có 2 biến trở điều chỉnh độ nhạy chuyển động và thời gian giữ trạng thái kích hoạt của cảm biến. Điện áp hoạt động của cảm biến là 5V, góc quét: 120 độ, tầm xa: 7m, dòng tiêu thụ: 50uA.

Em thực hiện chọn quang trở 5mm vì giá thành rẻ, kích cỡ nhỏ năng lượng và điện áp hoạt động của quang trở 5mm rất phù hợp với mô hình của đề tài.

Dòng tiêu thụ của các cảm biến là $50\mu A + 100mA + 2.5mA = 102.5mA$

3.2.2.2 Thiết kế khối xử lý trung tâm

Ý tưởng thiết kế của em là kết hợp nhiều module cảm biến lại với nhau, do vậy sẽ có nhiều chân kết nối, sử dụng giao thức UART, ngoài ra việc lập trình với arduino tương đối dễ nên arduino mega 2560 là rất phù hợp. Arduino mega 2560 sử dụng chip ATmega2560. Arduino mega có 54 chân digital I/O 16 chân đầu vào tương đương (Analog Inputs), 4 UARTs (cổng nối tiếp phân cứng), một thạch anh dao động 16MHz, kết nối USB, một jack cắm điện, một đầu ICSP và một nút reset. Arduino mega chứa tất cả mọi thứ cần thiết để tạo thành khối xử lý trung tâm với đầy đủ các cổng kết nối.

Trong quá trình kết nối các module và lập trình cho hệ thống: Bộ nhớ sử dụng hết 2958 bytes vào khoảng 36% của bộ nhớ. Tổng số chân I/O sử dụng là 23 chân và dòng tối đa trên mọi chân I/O là $I_{IC} = 40mA$, nên dòng tiêu thụ của arduino là:

$$I_{Ar} = I_{IC} \times N = 40 \times 23 = 920 \text{ mA}$$

3.2.2.3 Thiết kế khối Relay điều khiển thiết bị

Với việc điều khiển thiết bị trong nhà như đèn quạt sử dụng dòng điện có điện áp cao hơn so với điện áp 5V của khối xử lý, em thực hiện sử dụng relay 5V như một công tắc là rất hợp lý vì relay chịu được hiệu điện thế lên đến 250VAC. Ngoài ra vì nguồn cấp cho mạch là 5V nên em thực hiện đã lựa chọn sử dụng Relay 5V để phù hợp với điện áp nguồn.

Em chọn module 8-Relay 5V vì đề tài điều khiển tổng cộng 8 thiết bị và ngoài ra còn việc module còn được thiết kế chắc chắn, khả năng cách điện tốt, kết nối với vi điều khiển đơn giản với các header, chịu được hiệu điện thế lên đến 250VAC 10A. Trên module đã có sẵn mạch kích relay sử dụng transistor và opto cách ly giúp cách ly hoàn toàn mạch điều khiển với relay, bảo đảm vi điều khiển hoạt động ổn định.



Hình 3.2 Module 8-Relay 5V

3.2.2.4 Khối truyền dữ liệu

Trong khối truyền dữ liệu em thực hiện sử dụng NodeMCU vì có điện áp hoạt động 3.3V điện áp vào 5V thông qua cổng chân Vin nên rất thích hợp với điện áp cấp cho mạch. NodeMCU được tích hợp Wifi 2.4GHz để kết nối Internet không dây. Ngoài ra NodeMCU dễ dàng giao tiếp với Blynk vì NodeMCU đã được hỗ trợ giao tiếp với Blynk.



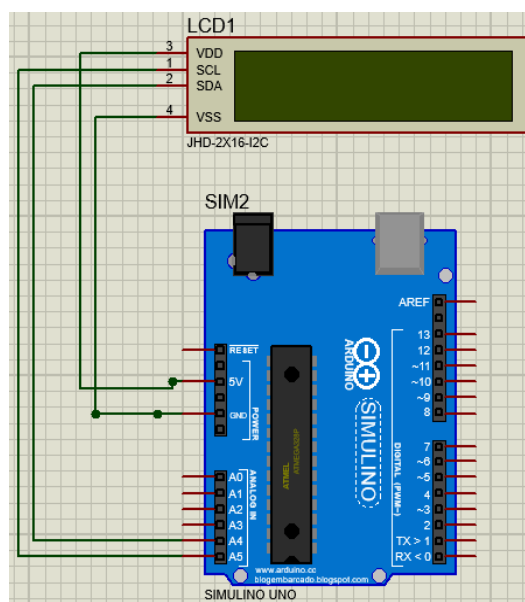
Hình 3.3 Giao diện thiết kế trên Blynk

NodeMCU sẽ truyền dữ liệu của cảm biến tới Blynk để hiển thị đồng thời nhận dữ liệu điều khiển từ Blynk để truyền cho arduino mega nhằm điều khiển các thiết bị trong nhà như đèn hay quạt tương ứng.

Ngoài ra em thực hiện điều khiển thiết bị bằng giọng nói với Google Assistant vì

được hỗ trợ giao tiếp với Blynk thông qua 2 ứng dụng trung gian là IFTTT: một dịch vụ dựa trên web miễn phí để tạo ra chuỗi các câu lệnh, các điều kiện đơn giản để thuận tiện cho việc điều khiển và WEBHOOK, ứng dụng cho phép cung cấp dữ liệu cho một ứng dụng khác trong thời gian thực.

3.2.2.5 Thiết kế khối hiển thị trên LCD



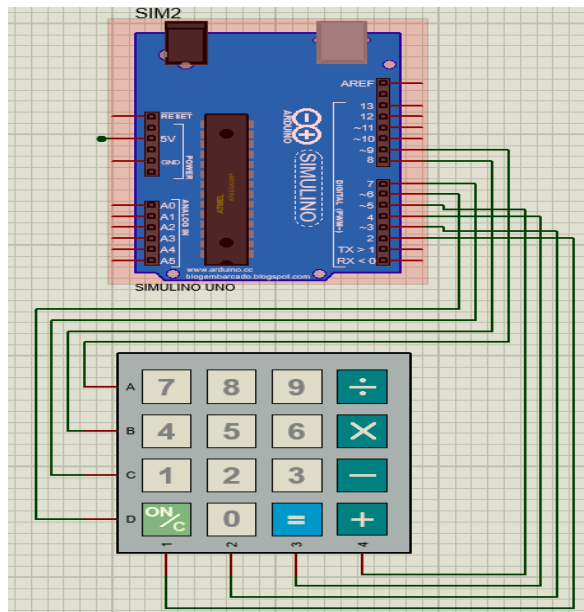
Hình 3.4 Sơ đồ nối chân của LCD

Em sử dụng mạch chuyển giao tiếp I2C cho LCD nhằm tiết kiệm chân I/O cho arduino uno

Em đã sử dụng bàn phím ma trận 4x4 để phục vụ cho việc nhập mật khẩu. Em chọn loại bàn phím như hình dưới đây vì có đủ số nút nhấn cần thiết cho đề tài, đồng thời nút nhấn rất chắc chắn, đáp ứng tức thời và không bị kẹt phím.

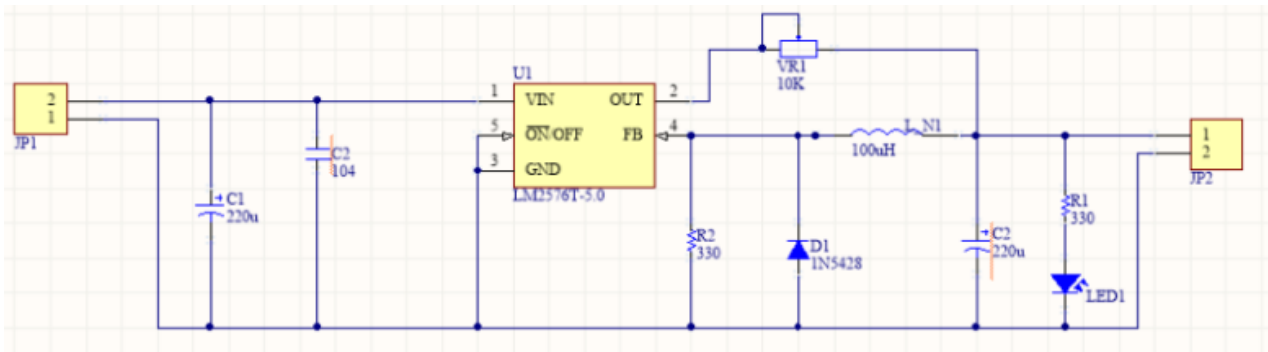


Hình 3.5 Bàn phím ma trận 4x4



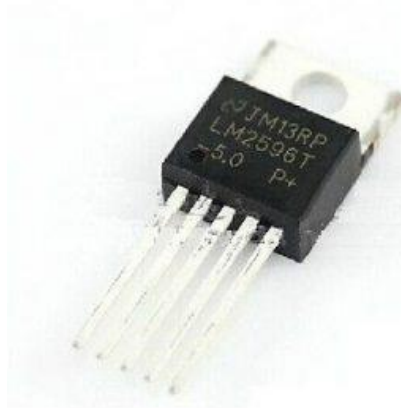
Hình 3.6 Sơ đồ kết nối bàn phím

3.2.2.6 Thiết kế khối nguồn



Hình 3.7 Sơ đồ nguyên lý mạch nguồn

Để cấp nguồn 5V cho đề án em sử dụng IC LM2596, đây là một IC ổn áp dạng xung DC - DC. Điện áp đầu vào trong dải từ 4,5V - 40V. Điện áp đầu ra là 5V dòng điện áp đầu ra tối đa đạt 3A, IC LM2596 có hiệu suất cao nhờ cơ chế băm xung ở tần số lên tới 150KHz. Trong quá trình hoạt động LM2596 luôn được đặt trong các chế độ bảo vệ quá nhiệt và quá dòng.



Hình 3.8 IC LM2596

Thông số cơ bản

- Điện áp đầu ra cố định: 5V
- Phạm vi điện áp đầu vào rộng, lên đến 40V
- Kích thước PCB: 2,5 x 5,0 cm

- Dòng điện ra tối đa 3.0A
- Bộ dao động nội tần số cố định 150KHz
- Hiệu suất chuyển động lên đến 92%

Bảng 3.2 Chức năng các chân IC LM2596

Chân	Tên	Chức năng
1	Vin	Đây là đầu vào điện áp dương cho IC. Chân này phải được nối với một tụ điện có giá trị thích hợp để giảm thiểu quá độ điện áp và để cung cấp dòng chuyển mạch cần thiết bởi bộ điều chỉnh.
2	GND	Chân nối đất
3	Output	Công tắc bên trong. Điện áp ở chân này chuyển đổi giữa (+ VIN - VSAT0 và khoảng -0,5V, với chu kỳ làm việc xấp xỉ VOUT/VIN
4	Feedback	Cảm biến điện áp đã điều chỉnh ở ngõ ra để hoàn tất vòng phản hồi
5	ON/OFF	Cho phép tắt mạch điều chỉnh chuyển mạch bằng cách sử dụng tín hiệu mức logic, do đó giảm dòng tổng cung cấp đầu vào xuống khoảng 80. Kéo chân này xuống dưới mức điện áp ngưỡng khoảng 1,3V, bật bộ điều chỉnh và kéo chân này lên trên 1,3V (tối đa 25V) để tắt bộ điều chỉnh. Nếu không caanf tính năng tắt máy này, chân ON/OFF có thể được nối đất hoặc nó có thể được bỏ trống; trong cả hai trường hợp, bộ điều chỉnh sẽ ở trong trạng thái ON

Nguồn cấp cho Arduino

Board Arduino Uno sử dụng hết 16 chân (N). Dòng DC trên mỗi chân I/O (I_{IC}) là 30mA.

$$I_{Ar_1} = I_{IC_1} \times N = 30 \times 16 = 480 \text{ mA}$$

Board Arduino Mega sử dụng hết 23 chân (N). Dòng tiêu thụ tối đa trên mỗi chân I/O (I_{IC}) là 40mA.

$$I_{Ar_2} = I_{IC_1} \times N = 40 \times 23 = 920 \text{ mA}$$

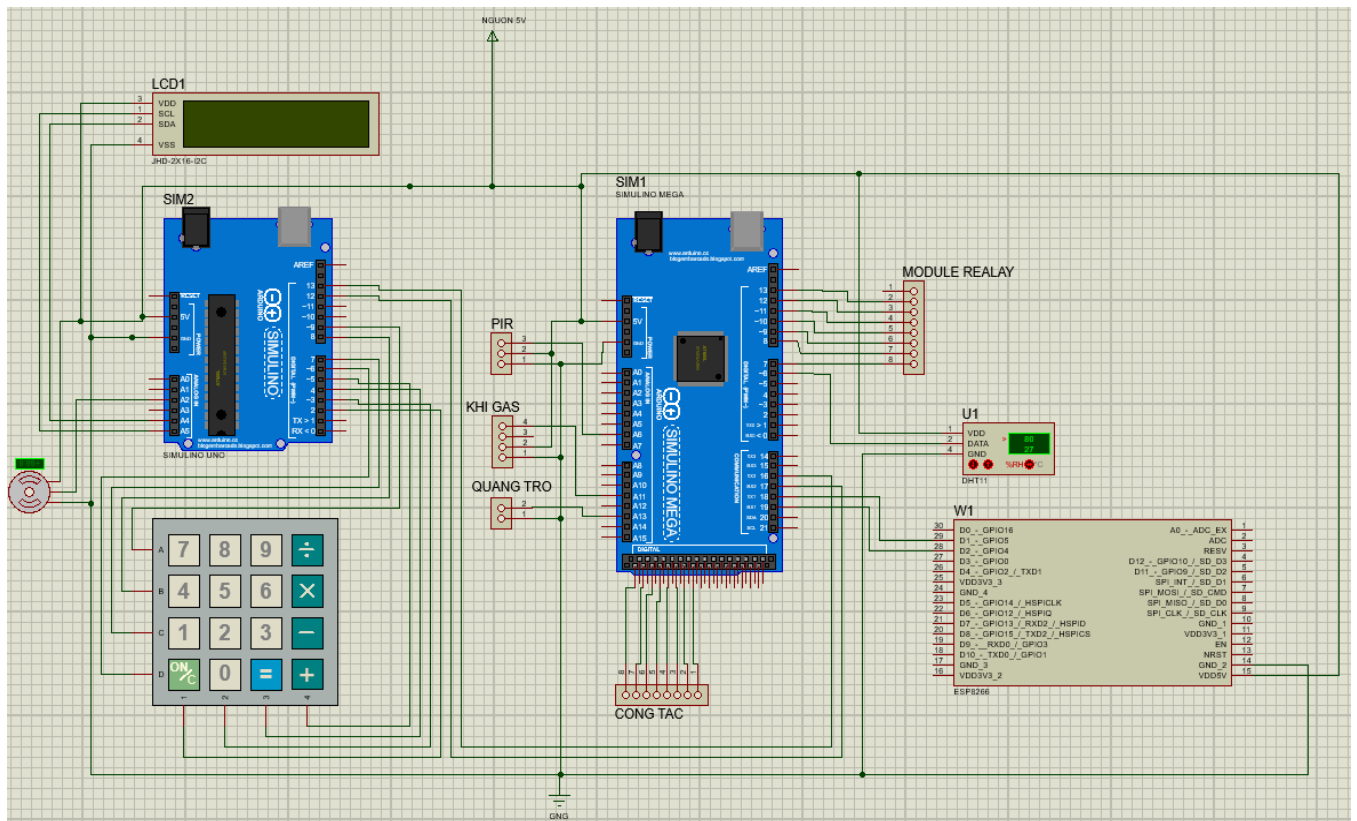
Dòng tiêu thụ của mỗi relay (I_{relay}) là 80 mA. Dòng tiêu thụ của các cảm biến không quá 100mA.

Tổng dòng tiêu thụ trên là:

$$I = I_{Ar_1} + I_{Ar_2} + I_{sennsor} + I_{relay} = 920 + 6 \times 80 + 100 + 480 = 1.98 \text{ A}$$

Theo tính toán thì sẽ cấp dòng tối thiểu là 1.98A. Vì vậy, em sử dụng IC LM2596 từ nguồn xung (hay nguồn tổ ong) 12V 5A để có ngõ ra là 5V 3A.

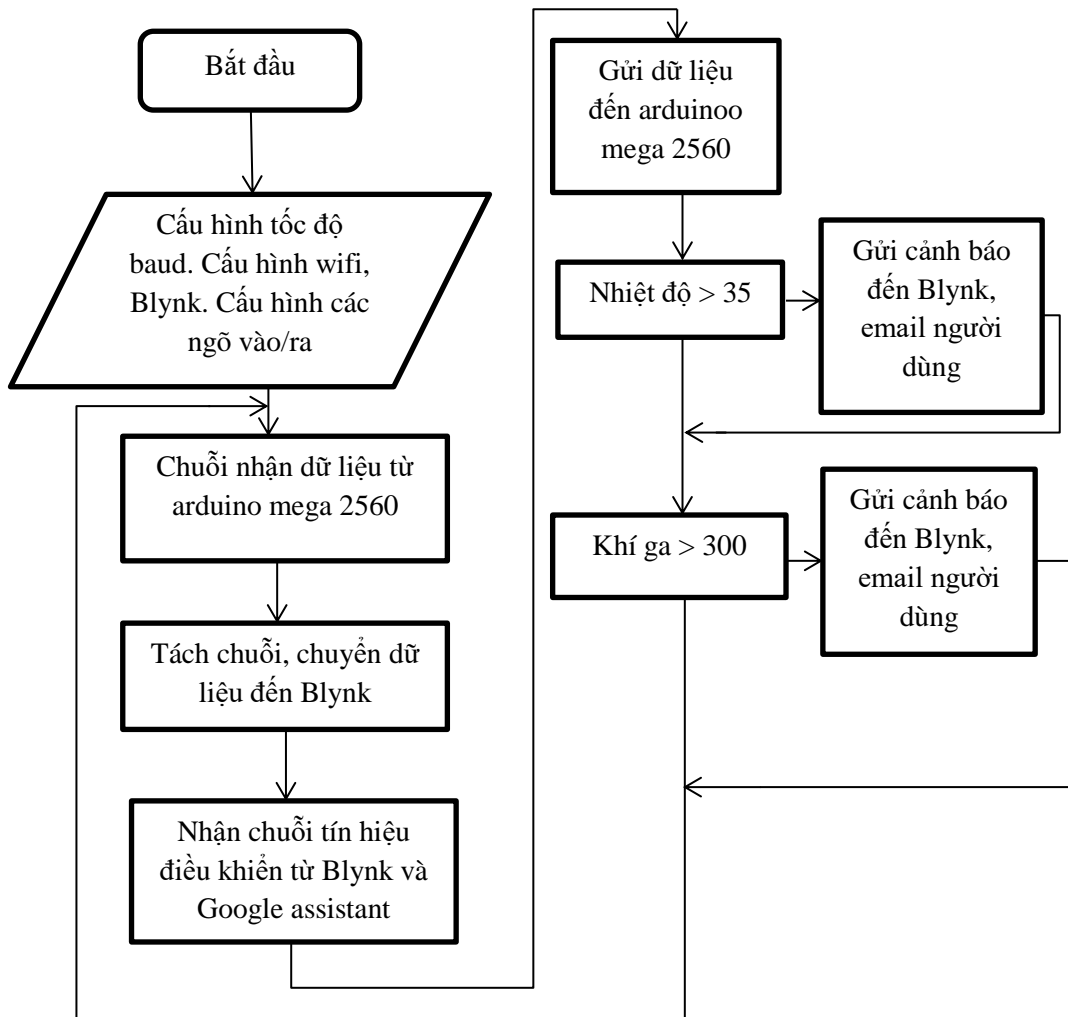
3.2.2.7 Sơ đồ nguyên lý của toàn mạch



Hình 3.9 Sơ đồ nguyên lý toàn mạch

3.3 LẬP TRÌNH HỆ THỐNG

3.3.1 Lưu đồ giải thuật module NodeMCU



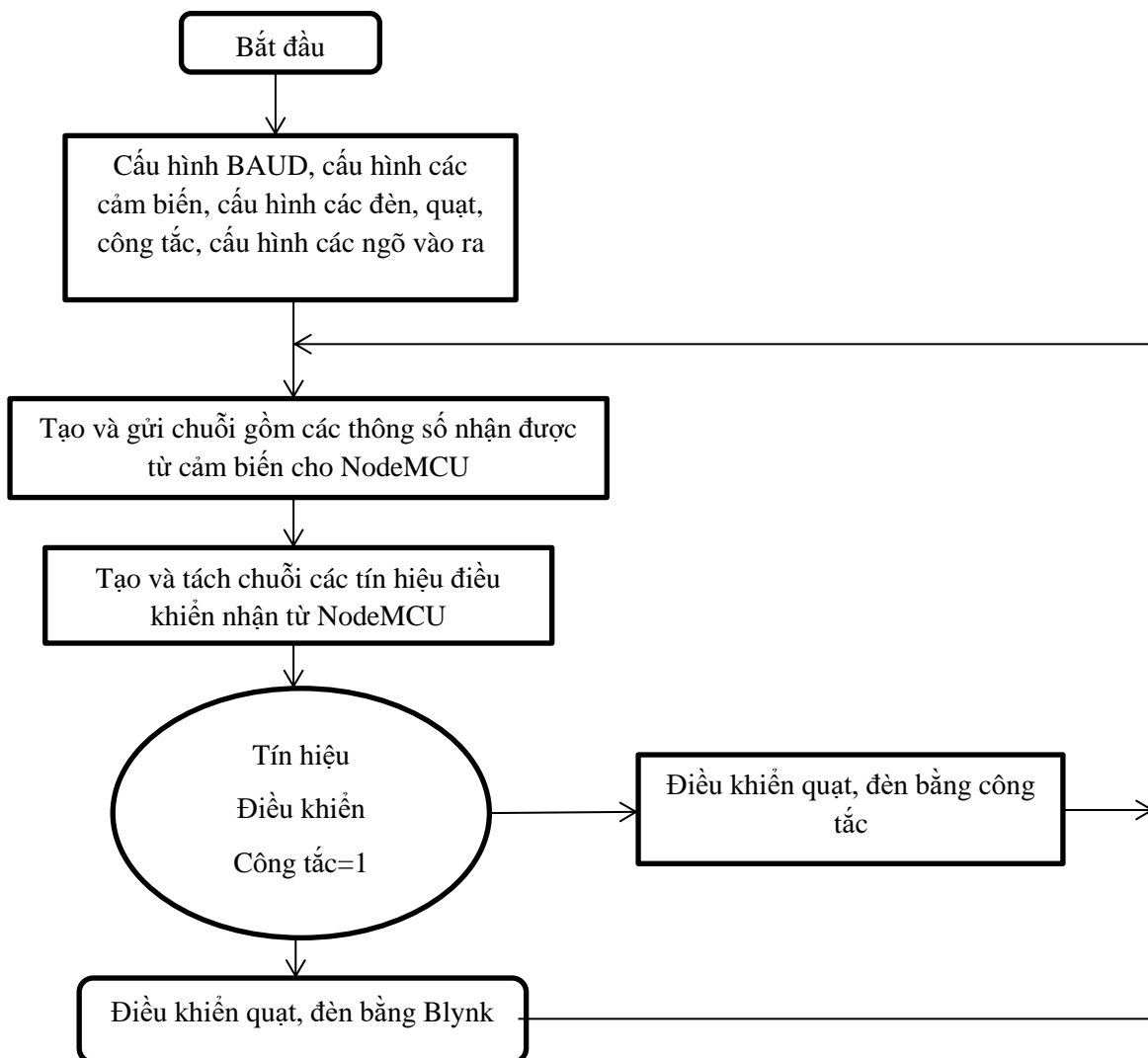
Hình 3.11 Lưu đồ giải thuật của module NodeMCU

Giải thích lưu đồ:

NodeMCU trước tiên sẽ được cấu hình Wifi, các ngõ vào in-out sau đó sẽ nhận chuỗi dữ liệu được gửi từ arduino mega qua giao thức truyền UART. Tiếp theo, NodeMCU thực hiện tách chuỗi để lấy các giá trị cảm biến và trạng thái hoạt động của

các thiết bị, sau đó đẩy các thông số này lên ứng dụng blynk qua wifi để giám sát theo dõi. Đồng thời NodeMCU cũng nhận tín hiệu điều khiển từ Blynk và Google Assistant, gộp các tín hiệu này thành một chuỗi để gửi cho arduino mega thực hiện. Các bước nhận và truyền dữ liệu của NodeMCU đều được thực hiện liên tục thông qua Wifi và AURT. Nếu bất kì thông số nào trong nhiệt độ và khí gas vượt quá mức cho phép thì gửi cảnh báo đến điện thoại và qua email của người dùng. Các bước trên lặp lại liên tục.

3.3.2 Lưu đồ giải thuật module Aduino mega 2560



Hình 3.12 Lưu đồ giải thuật của module Arduino Mega

Giải thích lưu đồ

Đối với arduino mega, sau khi cấu hình tốc độ baud, cấu hình ngõ vào ra và cảm biến thì arduino sẽ đọc các giá trị cảm biến, tạo 1 chuỗi gồm các thông số của các cảm biến, trạng thái các thiết bị để gửi cho NodeMCU thông qua AURT và đồng thời nhận chuỗi tín hiệu điều khiển cho NodeMCU truyền đến, tách chuỗi để lấy các giá trị điều khiển và dùng các giá trị đó để điều khiển các thiết bị tương ứng. Arduino uno sẽ truyền dữ liệu và cảnh báo qua cho arduino mega thông qua AURT và arduino mega sẽ truyền lên cho NodeMCU cũng thông qua AURT để thực hiện việc giám sát và điều khiển từ xa. Các bước trên sẽ được lặp lại liên tục để đảm bảo hệ thống luôn điều khiển đúng thiết bị cập nhật chính xác các giá trị thu được từ các cảm biến.

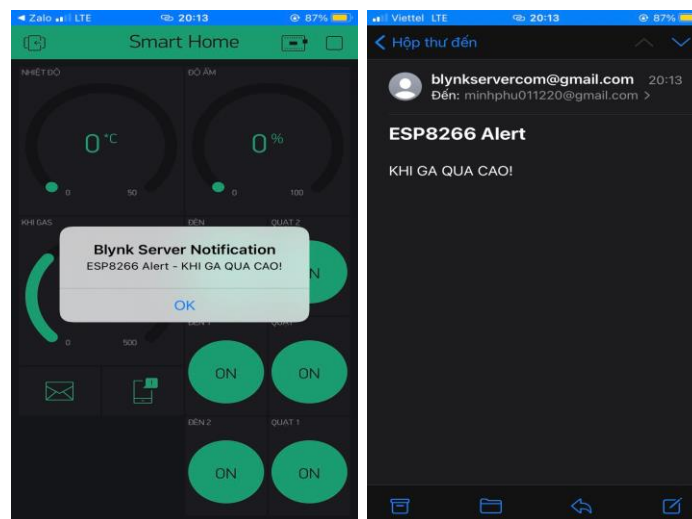
CHƯƠNG 4

KẾT QUẢ THỰC HIỆN

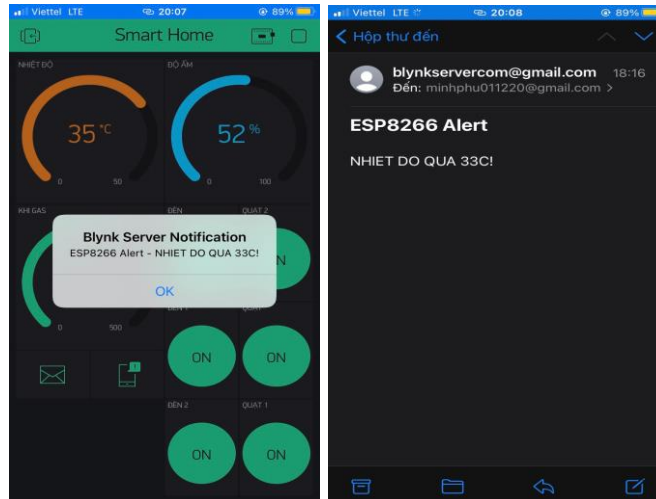
5.1 KẾT QUẢ

Sau quãng thời gian nghiên cứu, tìm hiểu và tổng hợp lại các kiến thức đã học trong suốt 4 năm học, em cũng đã hoàn thành được đề án tốt nghiệp với đề tài " ĐIỀU KHIỂN VÀ GIÁM SÁT THIẾT BỊ ĐIỆN BẰNG GIỌNG NÓI QUA INTERNET ".Em đã tìm hiểu và nghiên cứu về hoạt động của kit arduino mega 2560, NodeMCU, các loại module cảm biến: khí gas MQ2, cảm biến nhiệt độ-độ ẩm, cảm biến chuyển động... và cách kết nối giữa các module để hoàn thành một mô hình hoàn thiện với các chức năng như: điều khiển bật tắt thiết bị từ xa bằng giọng nói với Google Assistant và bằng điện thoại thông minh thông qua Wifi, giám sát từ xa các thông số về nhiệt độ, khí ga, độ ẩm... cũng như thông báo về điện thoại khi có sự cố. Để giúp căn nhà có tính bảo mật cao hơn, em đã cài cảnh báo về mail và điện thoại khi có người lạ đột nhập hay nhập mật khẩu sai nhiều lần.

4.1.1 Mô hình hệ thống



Hình 4.1 Cảnh báo khí gas quá cao



Hình 4.2 Cảnh báo nhiệt độ quá cao

4.1.2 *Giao diện phần mềm hệ thống*

Em đã sử dụng app Blynk làm giao diện điều khiển của mô hình. Sau khi hoàn thiện thì hệ thống có giao diện như hình dưới.



Hình 4.3 Giao diện Blynk

Giao diện Blynk của hệ thống bao gồm 3 khối:

- Khối điều khiển: bao gồm 6 nút để điều khiển bật tắt các thiết bị như đèn quạt trong ngôi nhà.

- **Khối hiển thị:** Bao gồm 3 biểu đồ biểu thị các thông số về nhiệt độ, độ ẩm, khí gas của ngôi nhà để giúp người dùng dễ dàng theo dõi ngôi nhà từ xa.

Ngoài điều khiển bằng điện thoại đề tài của em còn điều khiển các thiết bị trong ngôi nhà bằng giọng nói với Google Assistant như hình sau:



Hình 4.4 Giao diện điều khiển giọng nói với Google Assistant

4.1.3 Hiển thị thông báo trên LCD

Khi khởi động hệ thống, LCD hiển thị giao diện nhập mật khẩu trên màn hình LCD. Để thực hiện các thao tác nhập mật khẩu chọn phím A từ bàn phím, người dùng thực hiện nhập mật khẩu 4 số từ bàn phím:



Hình 4.5 Hệ thống yêu cầu nhập mật khẩu



Hình 4.6 Nhập mật khẩu



Hình 4.7 Các tùy chọn sau khi nhập mật khẩu thành công

Sau khi quản trị viên nhập mật khẩu chính xác, màn hình thông báo quản trị viên biết rằng đã nhập mật khẩu thành công và hiện menu quản trị

Tùy vào nhu cầu mà người quản trị viên có thể chọn 3 để mở cửa hay muốn quay lại menu lựa chọn thì ấn D.

CHƯƠNG 5

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 KẾT LUẬN

Sau quá trình thực hiện và hoàn tất đề tài, em đã tích lũy được thêm nhiều kiến thức và kinh nghiệm mới:

- Sử dụng được arduino, NodeMCU, ma trận phím, LCD, các loại cảm biến như: cảm biến chuyển động, cảm biến khí gas MQ2, cảm biến DHT11...
- Tìm hiểu và tiến hành kết nối arduino, NodeMCU với các module và cảm biến bằng các chuẩn kết nối như UART, I2C, one-wire.
- Biết cách thiết kế mạch điện bằng Proteus và thi công mạch điện thực tế.
- Tạo được các dự án trên Adafruit.
- Sử dụng được Google Assistant và ứng dụng Blynk để điều khiển thiết bị qua giọng nói và ứng dụng trên điện thoại.
- Nâng cao kỹ năng lập trình trên phần mềm Arduino IDE.

Sau quá trình nghiên cứu, thực hiện đề tài “ĐIỀU KHIỂN VÀ GIÁM SÁT THIẾT BỊ ĐIỆN BẰNG GIỌNG NÓI QUA INTERNET”, hệ thống đã đáp ứng đầy đủ các tính năng, nội dung và mục tiêu ban đầu đã đề ra.

- Mô hình hoạt động ổn định, có thể làm việc liên tục và đáp ứng được các yêu cầu đã đề ra. Người dùng thao tác một cách đơn giản, dễ sử dụng. Hệ thống đảm bảo an toàn và bảo mật cho người dùng.
- Các thiết bị trong nhà có thể điều khiển được thông qua giọng nói, ứng dụng. Các cảm biến điều hoạt động tốt, gửi dữ liệu đến điện thoại liên tục, trạng thái thiết bị được cập nhật liên tục trong thời gian hệ thống hoạt động.
- Hệ thống tiến hành cảnh báo cho người dùng bằng cách gửi mail và thông báo qua điện thoại.

Em đã cố gắng thực hiện nhưng vẫn còn tồn đọng một số hạn chế về mặt kiến thức cũng như thời gian thực hiện nên đề tài khó tránh khỏi sai sót và hạn chế:

- Một số vấn đề chưa được xử lý như khi nhà đột ngột mất điện hay là khi đã cấp điện cho thiết bị nhưng thiết bị bị hỏng hoặc trục trặc thì trạng thái thiết bị không được cập nhật chính xác.
- Chưa đáp ứng được điều khiển trong thời gian thực.

5.2 HƯỚNG PHÁT TRIỂN

- Thêm cảm biến vân tay để dễ dàng hơn khi mở cửa.
- Thêm nguồn dự phòng để giúp hệ thống có thể hoạt động khi mất điện.
- Tối ưu hóa giải thuật nhằm giảm thời gian trễ khi điều khiển.
- Thêm chức năng nhận diện giọng nói để giúp cho mô hình có tính bảo mật cao hơn.
- Thêm cảm biến dòng điện để giúp người sử dụng có thể phát hiện được lỗi khi đã đóng relay nhưng thiết bị trong nhà không hoạt động.

PHỤ LỤC

1. MÃ NGUỒN CHƯƠNG TRÌNH ARDUINO MEGA 2560

```
#include "DHT.h"

const int DHTPIN = 5;

const int DHTTYPE = DHT11;

#define den 7

#define quat 8

#define den1 9

#define quat1 10

#define den2 11

#define quat2 12

#define denqt 6

#define denpir 13

bool en1, en2, en3, en4, en5, en6;

int p1, p2, p3, p4, p5, p6;

String r1, r2, r3, r4, r5, r6;

#define sw1 53

#define sw2 51

#define sw3 49

#define sw4 47

#define sw5 45

#define sw6 43

String q1, q2, q3, q4, q5, q6;
```

```

DHT dht(DHTPIN, DHTTYPE);

String str, mocua1;

void setup()
{
  Serial.begin(57600) ; Serial1.begin(57600); Serial2.begin(57600);
  dht.begin();

  pinMode(den, OUTPUT);
  pinMode(quat, OUTPUT);
  digitalWrite(den, LOW);
  digitalWrite(quat, LOW);
  pinMode(den1, OUTPUT);
  pinMode(quat1, OUTPUT);

  digitalWrite(den1, LOW);
  digitalWrite(quat1, LOW);
  pinMode(den2, OUTPUT);
  pinMode(quat2, OUTPUT);
  digitalWrite(den2, LOW);
  digitalWrite(quat2, LOW);
}

String nhandulieu2()
{
  String buffer1;

```

```

while (Serial2.available() > 0) {
    char c = Serial2.read();
    buffer1.concat(c); delay(3);
}
return buffer1;
}

String nhandulieu1()
{
    String buffer1;
    while (Serial1.available() > 0)
    {
        char c = Serial1.read();
        buffer1.concat(c); delay(3);
    }
    return buffer1;
}

unsigned long time1 = 0;

void loop() {
    int doam = dht.readHumidity();
    int nhietdo = dht.readTemperature();
    int khiga = analogRead(A11);
    int quang_tro = analogRead(A13);
    int pir = analogRead(A6);

```

```

int chophepmocua, cong tac, vantay;
if (isnan(doam) || isnan(nhietdo)) {
    Serial.println("Failed to read from DHT sensor!"); return;
}
String canhbao = nhandulieu2();
Serial.println("Van tay :");
vantay = canhbao.toInt();
Serial.println(vantay);
if ( (unsigned long) (millis() - time1) > 5000 ) {
    if (nhietdo >= 0 and nhietdo < 60) {
        str = String('T') + String(nhietdo) + String('H') + String(doam) + String('G') +
            String(khiga) + String('E') + String(pir) + String('P') + String(quang_tro) +
String('Q') +
            String(vantay) + String('V') + String(p1) + String('D') + String(p2) + String('O') +
String(p3
                ) +
            String('R') + String(p4) + String('T') + String(p5) + String('Y') + String(p6) +
String('U');
        Serial1.println(str);
    }
    time1 = millis();
}
String chuoi = nhandulieu1();
Serial.println("Da Nhan Chuoi");
Serial.println(chuoi); // D0Q1

```

```
String chophep = chuoi.substring(15, 16);
congtac = chophep.toInt();
p1 = digitalRead(den); //Dieu khien thiet bi
p2 = digitalRead(quat);
p3 = digitalRead(den1);
p4 = digitalRead(quat1);
p5 = digitalRead(den2);
p6 = digitalRead(quat2);
if (congtac == 1) {
    if (digitalRead(sw1) == "1") {
        digitalWrite(den, digitalRead(sw1));
    }
    else {
        digitalWrite(den, digitalRead(sw1));
    }
    if (digitalRead(sw2) == "1") {
        digitalWrite(quat, digitalRead(sw2));
    }
    else {
        digitalWrite(quat, digitalRead(sw2));
    }
    if (digitalRead(sw3) == "1") {
        digitalWrite(den, digitalRead(sw3));
    }
}
```



```
}  
else {  
    digitalWrite(den, digitalRead(sw3));  
}  
if (digitalRead(sw4) == "1") {  
    digitalWrite(quat1, digitalRead(sw4));  
}  
else {  
    digitalWrite(quat1, digitalRead(sw4));  
}  
  
if (digitalRead(sw3) == "1") {  
    digitalWrite(den2, digitalRead(sw5));  
}  
else {  
    digitalWrite(den2, digitalRead(sw5));  
}  
if (digitalRead(sw4) == "1") {  
    digitalWrite(quat2, digitalRead(sw6));  
}  
else {  
    digitalWrite(quat2, digitalRead(sw6));  
}
```

```

}
if (congtac == 0) {
    if (r1 != chuoi.substring(1, 2)) {
        if (chuoi.substring(1, 2) == "1") {
            digitalWrite(den, HIGH);
            Serial.println("Den chay");
            r1 = "1";
        }
        else if (chuoi.substring(1, 2) == "0") {
            digitalWrite(den, LOW);
            Serial.println("Den tat");
            r1 = "0";
        }
    }
}

if (r2 != chuoi.substring(3, 4)) { // Dieu khien quat
    if (chuoi.substring(3, 4) == "1") {
        digitalWrite(quat, HIGH);
        Serial.println("quat chay");
        r2 = "1";
    }
    else if (chuoi.substring(3, 4) == "0") {
        digitalWrite(quat, LOW);
        Serial.println("quat tat");
    }
}

```

```

    r2 = "0";
}
}
if (r3 != chuoi.substring(5, 6)) { // Dieu khien Den1
    if (chuoi.substring(5, 6) == "1") {
        digitalWrite(den1, HIGH);
        Serial.println("Den1 sang");
        r3 = "1";
    }
    else if (chuoi.substring(5, 6) == "0") {
        digitalWrite(den1, LOW);
        Serial.println("Den1 tat");
        r3 = "0";
    }
}
if (r4 != chuoi.substring(7, 8)) { // Dieu khien quat1
    if (chuoi.substring(7, 8) == "1") {
        digitalWrite(quat1, HIGH);
        Serial.println("quat1 chay");
        r4 = "1";
    }
    else if (chuoi.substring(7, 8) == "0") {
        digitalWrite(quat1, LOW);

```

```

Serial.println("quat1 tat");
r4 = "0";
}
}
if (r5 != chuo.substr(9, 10)) { // Dieu khien Den
    if (chuo.substr(9, 10) == "1") {
        digitalWrite(den2, HIGH);
        Serial.println("Den2 sang");
        r5 = "1";
    }
    else if (chuo.substr(9, 10) == "0") {
        digitalWrite(den2, LOW);
        Serial.println("Den2 tat");
        r5 = "0";
    }
}
if (r6 != chuo.substr(11, 12)) { // Dieu khien quat
    if (chuo.substr(11, 12) == "1") {
        digitalWrite(quat2, HIGH);
        Serial.println("quat2 chay");
        r6 = "1";
    }
    else if (chuo.substr(11, 12) == "0") {

```

```

    digitalWrite(quat2, LOW);
    Serial.println("quat2 tat");
    r6 = "0";
}
}
}
if (quang_tro < 500) { //điều khiển đèn quang trở
    digitalWrite(denqt, LOW);
    Serial.println("den quang tro tat");
}
else {
    digitalWrite(denqt, HIGH);
    Serial.println("den quang tro");
}
if ( pir > 500) { //điều khiển pir
    digitalWrite(denpir, LOW);
    Serial.println("den pir tat");
}
else {
    digitalWrite(denpir, HIGH);
    Serial.println("den pir sang");
} //chuyen dong
Serial.print("chuyen dong: ");

```

```
Serial.println(pir);  
Serial.print("quang tro : ");  
Serial.println(quang_tro);  
Serial.print("Nhiet do: ");  
Serial.println(nhietdo);  
Serial.print("Do am: ");  
Serial.println(doam);  
Serial.print("KHI GAS: ");  
Serial.println(khiga);  
Serial.print("canh bao van tay: ");  
Serial.println(vantay);  
Serial.println();  
}
```

2. MÃ NGUỒN CHƯƠNG TRÌNH NODEMCU

```
#define BLYNK_PRINT Serial  
  
#include <ESP8266WiFi.h>  
  
#include <SoftwareSerial.h>  
  
#include <BlynkSimpleEsp8266.h>  
  
#include <SimpleTimer.h>  
  
SoftwareSerial mySerial(D1,D2); // RX, TX  
  
int pinValue1;  
  
int pinValue2;  
  
int pinValue3;  
  
int pinValue4;
```

```

int pinValue5;

int pinValue6;

int pinValue7;

int pinValue8;

int staden,staquat,staden1,staquat1,staden2,staquat2;

SimpleTimer timer;

char auth[] ="4xH4c3CBRSMjc1GV1RWR9c5yJTa3gSxq";

#define WIFI_SSID "abc"//Ten Wifi truy cap

#define WIFI_PASSWORD "12345678" // Mat Khau Wifi

void myTimerEvent(){

// You can send any value at any time.

// Please don't send more that 10 values per second.

Blynk.virtualWrite(V1, millis() / 1000);

}

void setup() {

Serial.begin(57600);

mySerial.begin(57600);

Blynk.begin(auth,WIFI_SSID,WIFI_PASSWORD,"blynk-server.com",8080); // ket noi
blynk

Serial.print("connected to Blynk");

timer.setInterval(1000L,sensorvalue1); timer.setInterval(1000L,sensorvalue2);

timer.setInterval(1000L,sensorvalue3); timer.setInterval(1000L,sensorvalue4);

timer.setInterval(1000L,sensorvalue5); timer.setInterval(1000L,sensorvalue6);

timer.setInterval(1000L,sensorvalue7); timer.setInterval(1000L,sensorvalue8);

```

```

timer.setInterval(1000L,sensorvalue9);
WiFi.begin(WIFI_SSID, WIFI_PASSWORD); Serial.print("connecting");
while (WiFi.status() != WL_CONNECTED){
Serial.print("."); delay(500); }
Serial.print("connected: ");
Serial.println(WiFi.localIP());
}
String str,nhietdo,doam,khiga,chuoi,chuoitam,pir,quang_tro,vantay;
int chieudai;
String quat,den,quat1,den1,quat2,den2;
String nhandulieu(){
String buffer1;
while (mySerial.available(>0){
char c = mySerial.read();
buffer1.concat(c);
delay(3);}
return buffer1; }
void xulychuoi()
{
int moc_h, moc_g, moc_e, moc_q, moc_p, moc_v, moc_d, moc_o, moc_r, moc_t,
moc_y, moc_u;
chuoitam.remove(0,1); //xoa chu T dau tien // 74H33G1027E
for (int i = 0; i < chuoitam.length(); i++) { // Tim vi tri chu H
if (chuoitam.charAt(i) == 'H' and i > 0 ) {

```



```

moc_h = i;
break;}
}
nhietdo=chuoitam; // 74H33G1027E // Moc H = 2
nhietdo.remove(moc_h); // 74
Serial.println("Nhiet do = "+nhietdo);
chuoitam.remove(0, moc_h + 1); //33G1027E
Serial.println("Chuoi hien tai : "+chuoitam);
for (int i = 0; i < chuoitam.length(); i++) { // Tim vi tri chu G
if (chuoitam.charAt(i) == 'G' and i > 0 ) {
moc_g = i; break;}
}
doam = chuoitam;
doam.remove(moc_g);
Serial.println("Do Am = " + doam);
chuoitam.remove(0, moc_g + 1); //1027E
Serial.println("Chuoi hien tai : "+chuoitam);
for (int i = 0; i < chuoitam.length(); i++) { // Tim vi tri chu E
if (chuoitam.charAt(i) == 'E' and i > 0 ) {
moc_e = i; break;}
}
khiga = chuoitam;
khiga.remove(moc_e);

```

```

Serial.println("Khi ga = " + khiga);
chuoitam.remove(0, moc_e + 1); //1027E
Serial.println("Chuoì hien tai : "+chuoitam);
for (int i = 0; i < chuoitam.length(); i++) { // Tim vi tri chu P
if (chuoitam.charAt(i) == 'P' and i > 0 ) {
moc_p = i; break;}
}
pir = chuoitam;
pir.remove(moc_p);
Serial.println("PIR = " + pir);
chuoitam.remove(0, moc_p + 1); //1027E
Serial.println("Chuoì hien tai : "+chuoitam);
for (int i = 0; i < chuoitam.length(); i++) { // Tim vi tri chu Q
if (chuoitam.charAt(i) == 'Q' and i > 0 ) {
moc_q = i; break;}
}
quang_tro = chuoitam;
quang_tro.remove(moc_q);
Serial.println("Quang tro = " + quang_tro);
chuoitam.remove(0, moc_q + 1); //1027E
Serial.println("Chuoì hien tai : "+chuoitam);
for (int i = 0; i < chuoitam.length(); i++) { // Tim vi tri chu v
if (chuoitam.charAt(i) == 'V' and i > 0 ) {

```

```

moc_v = i; break;}
}
vantay = chuoitam;
vantay.remove(moc_v);
Serial.println("canh bao van tay = " + vantay);
chuoitam.remove(0, moc_v + 1); //1027E
Serial.println("Chuoi hien tai : "+chuoitam);
for (int i = 0; i < chuoitam.length(); i++) { // Tim vi tri chu d
if (chuoitam.charAt(i) == 'D' and i > 0 ) {
moc_d = i; break;}
}
den = chuoitam;
den.remove(moc_d);
Serial.println("den = " + den);
chuoitam.remove(0, moc_d + 1); //1027E
Serial.println("Chuoi hien tai : "+chuoitam);
for (int i = 0; i < chuoitam.length(); i++) { // Tim vi tri chu v
if (chuoitam.charAt(i) == 'O' and i > 0 ) {
moc_o = i; break;}
}
quat = chuoitam;
quat.remove(moc_o);
Serial.println("quat = " + quat);

```

```

chuoitam.remove(0, moc_o + 1); //1027E
Serial.println("Chuoi hien tai : "+chuoitam);
for (int i = 0; i < chuoitam.length(); i++) { // Tim vi tri chu v
if (chuoitam.charAt(i) == 'R' and i > 0 ) {
moc_r = i; break;}
}
den1 = chuoitam;
den1.remove(moc_r);
Serial.println("den1 = " + den1);
chuoitam.remove(0, moc_r + 1); //1027E
Serial.println("Chuoi hien tai : "+chuoitam);
for (int i = 0; i < chuoitam.length(); i++) { // Tim vi tri chu v
if (chuoitam.charAt(i) == 'T' and i > 0 ) {
moc_t = i; break;}
}
quat1 = chuoitam;
quat1.remove(moc_t);
Serial.println("quat1 = " + quat1);
chuoitam.remove(0, moc_t + 1); //1027E
Serial.println("Chuoi hien tai : "+chuoitam);
for (int i = 0; i < chuoitam.length(); i++) { // Tim vi tri chu v
if (chuoitam.charAt(i) == 'Y' and i > 0 ) {
moc_y = i; break;}
}

```

```

}
den2 = chuoitam;
den2.remove(moc_y);
Serial.println("den2 = " + den2);
chuoitam.remove(0, moc_y + 1); //1027E
Serial.println("Chuoi hien tai : "+chuoitam);
for (int i = 0; i < chuoitam.length(); i++) { // Tim vi tri chu v
if (chuoitam.charAt(i) == 'U' and i > 0 ) {
moc_u = i; break;}
}
quat2 = chuoitam;
quat2.remove(moc_u);
Serial.println("quat2 = " + quat2);
chuoitam.remove(0, moc_u + 1); //1027E
Serial.println("Chuoi hien tai : "+chuoitam);
}
void loop()
{
if (Serial.available() == 0 ){ //blynk{
Blynk.run();
timer.run();} // Initiates BlynkTimer
String chuoitai = nhandulieu();
Serial.println("Da Nhan Chuoi");

```

```

chieudai = chuoi.length();
Serial.println("chieu dai :" + chieudai);
if(chieudai > 0) {
  chuoitam = chuoi;
  Serial.println("chuoi tam : " + chuoitam);
  xulychuoi(); }

int quat = pinValue2; int den = pinValue1;
int quat1 = pinValue3; int den1 = pinValue4;
int quat2 = pinValue5; int den2 = pinValue6;

str=String('V')+den +String('D')+ quat +String('O')+ den1 +String('R')+ quat1 +
String('T')+ den2 +String('Y')+ quat2+String('U')+pinValue7+String('Z')+pinValue8;

mySerial.println(str);

Serial.println("Da nhan chuoi tu Blynk: " + str);
}

void sensorvalue1()
{
  Blynk.virtualWrite(V2, nhietdo);
  if(pir.toInt() > 1021){
    Blynk.email("minhphu011220@gmail.com", "ESP8266 Alert", "CO CHUYEN
DONG!");
    Blynk.notify("ESP8266 Alert - CO CHUYEN DONG!");}
  if(khiga.toInt() > 300){
    Blynk.email("minhphu011220@gmail.com", "ESP8266 Alert", "KHI GA QUA CAO!");
    Blynk.notify("ESP8266 Alert - KHI GA QUA CAO!");}
}

```

```

if(nhietdo.toInt() > 33){

  Blynk.email("minhphu011220@gmail.com", "ESP8266 Alert", "NHIET DO QUA
33C!");

  Blynk.notify("ESP8266 Alert - NHIET DO QUA 33C!");}

if(vantay.toInt() == 3){

}

}

void sensorvalue2(){ Blynk.virtualWrite(V3, doam);}
void sensorvalue3(){ Blynk.virtualWrite(V4, khiga);}
void sensorvalue4(){ Blynk.virtualWrite(V14, den);}
void sensorvalue5(){ Blynk.virtualWrite(V15, quat);}
void sensorvalue6(){ Blynk.virtualWrite(V16, den1);}
void sensorvalue7(){ Blynk.virtualWrite(V17, quat1);}
void sensorvalue8(){ Blynk.virtualWrite(V18,den2);}
void sensorvalue9(){ Blynk.virtualWrite(V19,quat2);}

BLYNK_WRITE(V10){ pinValue1 = param.asInt(); Serial.print(pinValue1);}
BLYNK_WRITE(V11){ pinValue2 = param.asInt(); Serial.print(pinValue2);}
BLYNK_WRITE(V9){ pinValue3 = param.asInt(); Serial.print(pinValue3);}
BLYNK_WRITE(V8){ pinValue4 = param.asInt(); Serial.print(pinValue4);}
BLYNK_WRITE(V7){ pinValue5 = param.asInt(); Serial.print(pinValue5);}
BLYNK_WRITE(V12){ pinValue6 = param.asInt(); Serial.print(pinValue6);}
BLYNK_WRITE(V13){ pinValue7 = param.asInt(); Serial.print(pinValue7);}
BLYNK_WRITE(V20){ pinValue8 = param.asInt(); Serial.print(pinValue8);}

```

3. MÃ NGUỒN CHƯƠNG TRÌNH ARDUINO UNO R3

```
#include <Keypad.h>

#include <LiquidCrystal_I2C.h>

#include <Adafruit_Fingerprint.h>

#include <Wire.h>

#include <Servo.h>

#define SERVO_PIN A2

Servo gServo;

const byte ROWS = 4; //four rows

const byte COLS = 4; //three columns

char keys[ROWS][COLS] =

{

  {'1', '2', '3', 'A'},

  {'4', '5', '6', 'B'},

  {'7', '8', '9', 'C'},

  {'*', '0', '#', 'D'}

};

unsigned long time1 = 0;

byte rowPins[ROWS] = {9, 8, 7, 6};

byte colPins[COLS] = {5, 4, 3, 2};

Keypad mykeypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

LiquidCrystal_I2C lcd(0x27, 16, 2);

const int buttonPin = A3; //

int buttonState =0;
```



```

char class_x = '0';
char class_y = '3';
byte index = 0;
char matkhau[10] = { };
String password_admin = "1111"; //Đặt mật khẩu
int id=0;
int canhbao,chophep,bep;
int id_del=0;
byte index_id = 0;
char id_buff[] = { };
uint8_t getFINGERPRINTEnroll();
SoftwareSerial mySerial(10, 11); //serial vân tay
SoftwareSerial mySerial1(12, 13); // RX, TX //nối với ATMEGA2560
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);
uint8_t getFINGERPRINTEnroll();
bool check_dangky = false;
bool check_delete = false;
boolean out_FINGERPRINT = false;
char statu[4] = {'0', '0', '0', '0'};
void setup()
{
  lcd.init(); //Khởi động LCD nếu lỗi dòng này thì chuyển lcd.int(); thành lcd.begin();
  lcd.backlight (); lcd.display ();

```

```

Serial.begin (57600);
mySerial1.begin (57600);
pinMode (buttonPin, INPUT);
finger.begin (57600); //Khoi tao van tay
gServo.attach (SERVO_PIN);
gServo.write (0);
delay (1000);
}
String nhandulieu()
{
String buffer1;
while (mySerial1.available () > 0)
{
char c = mySerial1.read ();
buffer1.concat (c);
delay (3);
}
return buffer1;
}
void loop()
{
char key = mykeypad.getKey ();
switch (class_x)

```

```

{
case '0':
{
hienthi_lcd_clear (0, 0, "NHAN PHIM A DE");
hienthi_lcd (1, 0, "NHAP MAT KHAU ");
mySerial1.println (canhbao); Serial.println ("canhbaocvantay :");
Serial.println (canhbao); delay (500);
while (class_x == '0')
{

char key = mykeypad.getKey ();
if (key == '1')
{
class_x = '1';
mySerial1.println (canhbao); Serial.println ("canhbaocvantay :");
Serial.println (canhbao); break;
}
if(key == 'A')
{
class_x = '2'; class_y = '3';
mySerial1.println (canhbao); Serial.println ("canhbaocvantay :");
Serial.println (canhbao);
hienthi_lcd_clear (0, 0, "MOI SEP NHAP MK:");

```

```

lcd.setCursor (0, 1); break;
}
}
break;
}
case '1':
{
hienthi_lcd_clear (0, 0, "DANG QUET...");
while (!out_FINGERPRINT)
{
if (canhbao == 4)
{
getFINGERPRINTEnroll1 (); canhbao = 0; mySerial1.println (canhbao);
Serial.println ("canhbaocvantay :");
Serial.println (canhbao);
}
else
{
getFINGERPRINTIDez (); mySerial1.println (canhbao); Serial.println
("canhbaocvantay :");
Serial.println (canhbao); delay (50);
}
}
}

```

```
out_FINGERPRINT = false; break;
}
case '2':
{
switch (class_y)
{
case '3':
{
mySerial1.println (canhbao); Serial.println ("canhbaocvantay :");
Serial.println (canhbao);
if (key)
{
if (key == '#')
{
matkhau[index] = '\0';
if (String (matkhau) == password_admin)
{
hienthi_lcd_clear (0, 0, "MOI SEP VAO NHA ! ");
index = 0; delay (1000);
class_y = '0';
}
else
{
```

```

hienthi_lcd_clear (0, 0, "HONG DUNG BE OI:");
lcd.setCursor (0, 1); index = 0;
}
}
else if (key == 'D')
{
class_x = '0';
index = 0;
}
else if (index < 4)
{
matkhau[index] = key;
index++; lcd.print (" * ");
}
}
break;
}
case '0':
{
hienthi_lcd_clear (0, 0, "1.THEM"); hienthi_lcd (0, 8, "2.XOA");
hienthi_lcd (1, 0, "3.MOCUA"); hienthi_lcd (1, 8, "D.TROLAI"); delay
(500);
while (class_y == '0')

```

```

{
char key = mykeypad.getKey ();
if (key == '1')
{
class_y = '1'; hienthi_lcd_clear (0, 0, "Nhap ID Can Them:");
lcd.setCursor (0, 1);
}
if (key == '2')
{
class_y = '2'; hienthi_lcd_clear (0, 0, "Nhap ID Can Xoa:");
lcd.setCursor (0, 1);
}
if (key == 'D'){class_y = 'D'; }
if (key == '3'){class_y = '4'; }
}
break;
}
case '1':
{
id = 0;
char id_buff[] = {};
index_id = 0;
while (!check_dangky)

```

```

{
char key = mykeypad.getKey ();
if (key)
{
if (key == '#')
{
String str_id = String(id_buff);
id = str_id.toInt();
check_dangky = 1;
}
else if (key == 'D')
{
class_y = '0'; id = 0;
check_dangky = 1;
}
else
{
id_buff[index] = key;
index_id++;
check_dangky = 0;
lcd.print (key);
}
}
}

```



```

}
if (id != 0) getFINGERPRINTEnroll () ;
delay (1000);
check_dangky = false;
class_y = '0'; class_x = '2'; break;
}
case '2': //xoa van tay
{
id = 0; char id_buff[] = {};
index_id = 0;
while (!check_delete)
{
char key = mykeypad.getKey ();
if (key)
{
if (key == '#')
{
String str_id = String(id_buff);
id = str_id.toInt ();
check_delete = 1;
}
else if (key == 'D')
{

```

```

class_y = '0';
check_delete = 1; id = 0;
}
else
{
id_buff[index_id] = key;
index_id++; check_delete = 0;
lcd.print (key);
}
}
}
id_del = id;
if (id != 0) deleteFINGERPRINT (id);
id = 0; check_delete = false;
class_y = '0'; break;
}
case 'D':
{
class_y = '0'; class_x = '0'; break;
}
case '4':
{
hienthi_lcd_clear (0, 5, "MOI VAO") ;

```

```
gServo.write (0); delay (1000);  
gServo.write (90); delay (1000);  
gServo.write ( - 90); delay (1000);
```

```
class_y = '0'; class_x = '0'; break;  
}  
}  
break;  
}  
}  
}
```

```
//-----FINDGER-----
```

```
uint8_t getFINGERPRINTEnroll1()  
{  
int p = - 1;  
while (p != FINGERPRINT_OK)  
{  
p = finger.getImage ();  
}  
p = finger.image2Tz (); return p;  
id = 9; p = finger.storeModel (id);  
if (p == FINGERPRINT_OK)  
{
```

```

//Stored;

hienthi_lcd_clear (0, 2, "LUU VAN TAY"); hienthi_lcd (1, 2, " THANH CONG");

mySerial1.println (canhbao); Serial.println ("canhbaocvantay :");

Serial.println (canhbao); delay (1000);

class_x = '0'; out_FINGERPRINT = true; // check de out

delay (1); return - 1;

}

}

//-----Enroll-----

uint8_t getFINGERPRINTEnroll()

{

int p = -1;

hienthi_lcd_clear (0, 0, "THEM VAN TAY"); hienthi_lcd (1, 3, " VOI ID:");

lcd.print (id);

while (p != FINGERPRINT_OK){p = finger.getImage (); }

p = finger.image2Tz (1); return p;

hienthi_lcd_clear (0, 0, "Remove finger"); delay (2000);

p = 0;

while (p != FINGERPRINT_NOFINGER) {p = finger.getImage (); }

p = -1; hienthi_lcd_clear (0, 0, "Same finger");

while (p != FINGERPRINT_OK)

{

p = finger.getImage ();

```

```

break;
}
p = finger.image2Tz (2);
return p;
p = finger.createModel ();
if (p == FINGERPRINT_OK) {}
else
{
return p;
}
p = finger.storeModel(id);
if (p == FINGERPRINT_OK) {hienthi_lcd_clear(0, 2, "THEM VAN TAY");
hienthi_lcd (1, 2, " THANH CONG"); }
else if (p == FINGERPRINT_PACKETRECIEVEERR) {}
else{return p; }}
//-----FINGER - PRint-----
uint8_t getFINGERPRINTID ()
{
uint8_t p = finger.getImage ();
switch (p)
{
case FINGERPRINT_OK:
break;

```

```

DEFAULT;

return p;

}

p = finger.image2Tz ();

switch (p)

{

case FINGERPRINT_OK:

break;

DEFAULT;

return p;

}

p = finger.fingerFastSearch ();

if (p == FINGERPRINT_OK) {}

else

{

return p;

}

}

int getFINGERPRINTIDez ()

{

uint8_t p = finger.getImage (); char dataString[50] = {0};

if (p != FINGERPRINT_OK)

{

```

```

return - 1; canhbao++; mySerial1.println (canhbao);
Serial.println ("canhbaocvantay :"); Serial.println (canhbao);
}

p = finger.image2Tz ();
if (p != FINGERPRINT_OK)
{
return - 1; canhbao++; mySerial1.println (canhbao);
Serial.println ("canhbaocvantay :"); Serial.println (canhbao);
}

p = finger.fingerFastSearch ();
if (p != FINGERPRINT_OK)
{
hienthi_lcd_clear (0, 0, "van tay khong");
hienthi_lcd (1, 3, " trung khop"); delay (1000);
canhbao++; Serial.println ("canhbaovantay:"); Serial.println (canhbao);
mySerial1.println (canhbao); Serial.println ("canhbaocvantay :");
Serial.println (canhbao); delay (20);
class_x = '0'; out_FINGERPRINT = true; // check de out
delay (1);
return - 1;
}

if (finger.fingerID != 9)
{

```

```

hienthi_lcd_clear (0, 0, "Xin chao ID :");
lcd.print (finger.fingerID); delay (1000);
hienthi_lcd (1, 2, " MOI VAO "); delay (1000);
gServo.write (0); delay (1000); //mo cua
gServo.write (90); delay (1000);
gServo.write ( - 90); delay (1000);
hienthi_lcd_clear (0, 0, "Dang tai du lieu...");
delay (1);
class_x = '0'; // Quay lai menu
out_FINGERPRINT = true; // check de out
delay(1);
return finger.fingerID;
}
else
{
hienthi_lcd_clear (0, 0, "CANH BAO"); delay (5000); String chophep1 = nhandulieu ();
Serial.println ("cho phep :");
Serial.println (chophep1); class_x = '0'; class_y = '3'; out_FINGERPRINT = false;
delay (1);
}
}
//-----DeleteFinger-----
uint8_t deleteFINGERPRINT (uint8_t id)

```



```

{
uint8_t p = -1;
char dataString[50] = {0};
p = finger.deleteModel(id);
if (p == FINGERPRINT_OK) { hienthi_lcd_clear(0, 4, "DA XOA !");
delay (1000); }
else if (p == FINGERPRINT_PACKETRECEIVEERR)
{
hienthi_lcd_clear (0, 0, "communication error"); delay (2000);
return p;
}
else if (p == FINGERPRINT_BADLOCATION) {return p; }
else if (p == FINGERPRINT_FLASHERR) {return p; }
else{return p; }
}
//-----LCD-----
void hienthi_lcd (int hang, int cot, char * content)
{
lcd.setCursor (cot, hang);
lcd.print (content);
}
void hienthi_lcd_clear (int hang, int cot, char * content)
{

```

```
lcd.clear ();  
lcd.setCursor (cot, hang);  
lcd.print (content);  
}
```

TÀI LIỆU THAM KHẢO

- [1] Arduino mega 2560:
<https://www.robotshop.com/media/files/pdf/arduinomega2560datasheet.pdf>
- [2] Arduino uno r3:
https://www.fecegypt.com/uploads/dataSheet/1522237550_arduino%20uno%20r3.pdf
- [3] Module cảm biến khí ga MQ2: <https://www.pololu.com/file/0J309/MQ2.pdf>
- [4] Cảm biến chuyển động PIR HC – SR501:
<https://www.mpja.com/download/31227sc.pdf>
- [5] Cảm biến nhiệt độ - độ ẩm DHT11:
<https://image.dfrobot.com/image/data/KIT0003/DHT11%20datasheet.pdf>
- [6] Bàn phím ma trận 4x4: <https://thegioiic.com/products/ban-phim-4x4-16-nut>
- [7] Nguyễn Đình Phú, “*vi điều khiển PIC 16f887*”, trường ĐH Sư Phạm Kỹ Thuật TP.HCM, 08/2016.
- [8] Giao tiếp UART: <http://www.hocavr.com/2018/06/bai-5-giao-tiep-uart.html>
- [9] Nguyễn Văn Phúc, “Giám sát và điều khiển ngôi nhà thông minh bằng giọng nói “, Trường đại học sư phạm kỹ thuật Tp. Hồ Chí Minh khoa điện điện tử.