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 : Nguyễn Ngọc Khánh

Giảng viên hướng dẫn: TS. Đoàn Hữu Chức

HẢI PHÒNG – 2021

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

ĐIỀU KHIỂN VÀ GIÁM SÁT THIẾT BỊ ĐIỆN QUA MẠNG INTERNET

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

Sinh viên : Nguyễn Ngọc Khánh
Giảng viên hướng dẫn : TS. Đoàn Hữu Chức

HẢI PHÒNG – 2021

BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC QUẢN LÝ VÀ CÔNG NGHỆ HẢI PHÒNG

NHIỆM VỤ ĐỀ TÀI TỐT NGHIỆP

Sinh viên: Nguyễn Ngọc Khánh **Mã SV**: 1612102001

Lớp : DC2001

Ngành: Điện tự động công nghiệp

Tên đề tài: Điều khiển và giám sát thiết bị điện qua mạng Internet

NHIỆM VỤ ĐỀ TÀI

Nội dung và các yêu cầu cần giải quyết trong nhiệm vụ đề tài tốt nghiệp

……
……
……
……
……
……

Các tài liệu, số liệu cần thiết

……
……
……
……
……
……
……
……
……

Địa điểm thực tập tốt nghiệp

CÁN BỘ HƯỚNG DẪN ĐỀ TÀI TỐT NGHIỆP

Họ và tên : Đoàn Hữu Chức

Học hàm, học vị: Tiến Sỹ

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

Nội dung hướng dẫn: Điều khiển và giám sát thiết bị điện qua mạng Internet

Đề tài tốt nghiệp được giao ngày 12 tháng 04 năm 2021

Yêu cầu phải hoàn thành xong trước ngày 03 tháng 07 năm 2021

Đã nhận nhiệm vụ ĐTTN	Đã giao nhiệm vụ ĐTTN
Sinh viên	*Giảng viên hướng dẫn*
Nguyễn Ngọc Khánh	TS Đoàn Hữu Chức

Hải Phòng, ngày 04 tháng 07 năm 2021

TRƯỞNG KHOA

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN TỐT NGHIỆP

Họ và tên giảng viên: TS. Đoàn Hữu Chức

Đơn vị công tác: Trường Đại học quản lý và công nghệ Hải Phòng

Họ và tên sinh viên: Nguyễn Ngọc Khánh

Chuyên ngành: Điện Tự Động Công Nghiệp

Đề tài tốt nghiệp: Điều khiển và giám sát thiết bị điện qua mạng Internet

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

.....

.....

.....

2. Đánh giá chất lượng của khóa luận

.....

.....

.....

.....

.....

3. Ý kiến của giảng viên hướng dẫn tốt nghiệp

Được bảo vệ ☐ Không được bảo vệ ☐ Điểm hướng dẫn ☐

Hải Phòng, ngày 03 tháng 07 năm 2021

Giảng viên hướng dẫn

TS. Đoàn Hữu Chức

Cộng hòa xã hội chủ nghĩa Việt Nam

Độc lập - Tự do - Hạnh phúc

PHIẾU NHẬN XÉT CỦA GIẢNG VIÊM CHẤM PHẢN BIỆN

Họ và tên giảng viên:.....

Đơn vị công tác:.....

Họ và tên sinh viên: .....Chuyên ngành:.....

Đề tài tốt nghiệp: .....

.....

1. Phần nhận xét 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 2020

Giảng viên chấm phản biện

(ký và ghi rõ họ tên)

LỜI CẢM ƠN

Để hoàn thành tốt Đồ án tốt nghiệp, em xin gởi lời chân thành cảm ơn các thầy cô trong Khoa Điện-Điện Tử đã tạo những điều kiện tốt nhất cho chúng em hoàn thành đề tài.

Đặc biệt em xin gửi lời cảm ơn sâu sắc đến Thầy Đoàn Hữu Chức - Giảng viên Khoa Điện Điện Tử đã trực tiếp hướng dẫn và tận tình giúp đỡ tạo điều kiện để hoàn thành tốt đề tài.

Cuối cùng em xin cảm ơn đến gia đình, là chỗ dựa cũng như là nguồn động viên tinh thần mỗi khi em gặp khó khăn trong học tập cũng như trong quá trình nghiên cứu để hoàn thành đề tài tốt nghiệp. Em xin chân thành cảm ơn!

Hải Phòng, tháng 7 năm 2021

Sinh viên

Nguyễn Ngọc Khánh

Mục lục

Lời mở ầu.....9
1. Lí do chọn đề tài.....9
2. Mục tiêu nhiệm vụ nghiên cứu.....9
3. Nội dung đề tài.....9
4. Bố cục khóa luận.....10
Chương 1. Internet of things..... 11
1.1 Tổng quan về internet of things.....11
1.1.1 Giới thiệu về IOT.....11
1.1.2 Lịch sử hình thành.....12
1.1.3 Ứng dụng của IOT.....12
1.2 Các chuẩn giao tiếp được sử dụng.....15
1.2.1 Chuẩn Ethernet.....15
1.2.2 Chuẩn IP.....16
1.2.3 Chuẩn giao tiếp UART.....21
1.2.4 Công nghệ truyền nhận dữ liệu.....23
Chương 2 Tìm hiểu về Arduino..... 27
2.1 Tổng quan về arduino.....27
2.2 Arduino uno r3.....28
2.2.1 Cấu tạo.....28
2.2.2 Tính năng.....30
2.3 Arduino mega 2560.....31
2.3.1 Cấu tạo.....31
2.3.2 Tính năng.....31
2.4 Arduino ESP8266.....32
2.4.1 Cấu tạo.....32
2.4.2 Tính năng.....33
Chương 3 Ứng dụng cho điều khiển và giám sát 1 hệ thống điện 33
3.1 Giới thiệu phần mềm lập trình.....34
3.2 Relay.....37
3.3 Lựa chọn thiết bị điều khiển.....41
3.4 Chọn app đưa lên internet.....41
3.4.1 Giới thiệu về Blynk.....41
3.4.2 Cách cài đặt Blynk.....42
3.4.3 Cách hoạt động của Blynk.....42
3.5 Thiết kế xây dựng hệ thống.....46
3.5.1 Lưu đồ giải thuật.....46
3.5.2 Sơ đồ mô phỏng mạch.....48
3.5.3 Thực hiện lắp ráp mạch và ghép nối modul.....48
3.5.4 Kết luận chương.....51
3.6 Viết chương trình chạy hệ thống.....53
Kết luận
Tài liệu tham khảo.....56

Lời mở đầu

1. Lý do chọn đề tài

Ngày nay, công nghệ kết nối đầu tiên cần nhắc đến hiển nhiên là Wifi – công nghệ kết nối không dây phổ biến nhất hiện nay. Cũng vì tính phổ biến của dạng kết nối này mà cái tên Wifi thường bị lạm dụng để chỉ kết nối không dây nói chung. Lí do mà kết nối Wifi được ưa chuộng như vậy đơn giản là vì khả năng hoạt động hiệu quả trong phạm vi vài chục đến vài trăm mét của các mạng WLAN. Vì mục tiêu công nghệ hiện đại hóa ngày càng phát triển, em đã quyết định làm một đồ án "Điều khiển và giám sát thiết bị điện qua mạng internet". Đề tài của em khi hoàn thành chúng ta có thể giám sát các thiết bị điện bằng cách hiển thị trạng thái hoạt động trên điện thoại. Như vậy, dù chúng ta ở bất cứ nơi nào có internet đều có thể giám sát và điều khiển được các thiết bị đã kết nối với module điều khiển. Khi dự án thành công và được áp dụng rộng rãi thì sẽ rất tiện lợi cho cuộc sống thường ngày, giúp cho đất nước ngày càng phát triển.

2. Mục tiêu nhiệm vụ nghiên cứu

Với đề tài "**Điều khiển và giám sát thiết bị điện qua mạng internet**" mục tiêu là:

- Có chức năng giám sát qua internet, sử dụng điện thoại
- Có thể thi công đồ án trên một ngôi nhà thực tế hoặc mô hình.

3. Nội dung đề tài

Việc thực hiện thiết kế mạch ''Điều khiển và giám sát thiết bị điện qua mạng internet'' sẽ cần phải thực hiện các nội dung như sau:

Nội dung 1: Nghiên cứu tài liệu về Arduino, KIT NodeMCU ESP8266, giao tiếp không dây và mạng Internet.

Nội dung 2: Nghiên cứu các mô hình điều khiển.

Nội dung 3: Thiết kế mạch phần cứng cho thiết bị.

Nội dung 4: Thi công phần cứng, thử nghiệm và hiệu chỉnh phần cứng.

Nội dung 5: Thử nghiệm và điều chỉnh hệ thống cũng như chương trình để hệ thống được tối ưu.

Nội dung 6: Viết báo cáo thực hiện.

Nội dung 7: Bảo vệ luận văn

- Giới hạn

+ Kích thước mô hình

+ Sử dụng KIT NodeMCU ESP8266

+ Tập trung vào thiết bị điều khiển trung tâm

+Sử dụng các nền tảng đã có sẵn và các thư viện mở để phát triển sản phẩm

- Phạm vi ứng dụng

Đề tài là mô hình thu nhỏ, tuy nhiên có thể được ứng dụng rộng rãi ở các môi trường khác nhau như nhà ở, nhà xưởng, nhà kính…Trong sản xuất cũng như sinh hoạt.

4. Bố cục khóa luận

Bố cục khóa luận gồm 4 phần không kể mở đầu và phần kết thúc:

Chương 1. Tìm hiểu về internet of things

Chương 2 Tìm hiểu về Arduino

Chương 3 Thiết kế hệ thống

Kết luận

CHƯƠNG 1 : INTERNET OF THINGS

1.1.Tổng quan về internet of things

1.1.1 Giới thiệu về Internet of Things (IoT)

Internet of things là một hệ thống mạng lưới mà trong đó tất cả các thiết bị, đối tượng được kết nối Internet thông qua thiết bị mạng (network devices) hoặc các bộ định tuyến (routers). IoT cho phép các đối tượng được điều khiển từ xa dựa trên hệ thống mạng hiện tại. Công nghệ tiên tiến này giúp giảm công sức vận hành của con người bằng cách tự động hóa việc điều khiển các thiết bị.

Hình 1.1 Internet of things sử dụng trong các phương tiện truyền thông

- **Các thành phần chính trong một hệ thống IoT:**

• **Thiết bị:**

Mỗi thiết bị sẽ bao gồm một hoặc nhiều cảm biến để phát hiện các thông số của ứng dụng và gửi chúng đến Platform.

• **IoT – Platform:**

Nền tảng này là một phần mềm được lưu trữ trực tuyến còn được gọi là điện toán đám mây, các thiết bị được kết nối với nhau thông qua nó.

Nền tảng này thu thập dữ liệu từ thiết bị, toàn bộ dữ liệu được phân tích, xử lý, phát hiện nếu có lỗi phát sinh trong quá trình hệ thống vận hành.

Kết nối Internet:

Để giao tiếp được trong IoT, kết nối Internet của các thiết bị là một điều bắt buộc. Wifi là một trong những phương thức kết nối Internet phổ biến.

- **Ứng dụng:**

Ứng dụng là giao diện để người dùng điều khiển.

1.1.2. Lịch sử hình thành

Khái niệm về một mạng lưới thiết bị được kết nối với nhau đã được thảo luận vào đầu năm 1982, với một máy bán hàng tự động Coke được thực hiện ở Đại học Carnegie Mellon trở thành thiết bị kết nối Internet đầu tiên trên thế giới.

Thuật ngữ "Internet of things" được sử dụng lần đầu tiên bởi Kevin Ashton vào năm 1999. Sau đó IoT trải qua nhiều giai đoạn và có bước phát triển nhảy vọt cho đến ngày nay.

1.1.3. Ứng dụng của IoT

- **Nhà thông minh (Smart Home)**

Đây là một trong những ứng dụng được quan tâm nhiều nhất trong những năm gần đây. Một ngôi nhà thông minh hoàn toàn có thể được giám sát và điều khiển tự động. Bạn có thể bật tắt đèn bằng một ứng dụng trên điện thoại, nếu lỡ quên tắt tivi khi ra khỏi nhà bạn hoàn toàn có thể tắt nó ở một nơi có kết nối Internet, hoặc điều hòa sẽ tự động điều chỉnh tăng hay giảm khi nhiệt độ bên ngoài thay đổi. Và còn vô số ứng dụng khác nhằm mang lại sự tiện lợi nhất cho người dùng.

Hiện nay các chủ đầu tư xây dựng chung cư cũng đã tiếp cận với công nghệ này do nhu cầu sở hữu căn hộ thông minh của người dùng ngày càng cao.

Hình 1.2 Internet of things trong nhà thông minh

- **Giao thông vận tải**

An toàn là điều đầu tiên khi nghĩ đến tác động của IoT đối với giao thông vận tải. Ý tưởng đưa ra là các phương tiện có khả năng liên lạc với nhau bằng cách sửdụng dữ liệu đã được phân tích để có thể giảm đáng kể các sự cố tai nạn xảy ra khi tham gia giao thông. Sử dụng cảm biến, các phương tiện như ô tô, xe buýt được cảnh báo nguy cơ tiềm ẩn trên đường, hoặc thậm chí là tình trạng ùn tắc giao thông ở một số tuyến đường.

Dịch vụ vận chuyển hàng hóa cũng được ứng dụng từ công nghệ này. Công nghệ quản lý lịch trình vận chuyển, tối ưu hóa các tuyến giao hàng, mức tiêu thụ nhiên liệu của phương tiện, giám sát tốc độ của tài xế giao hàng tuân thủ quy định an toàn nhằm mang lại những lợi ích về kinh tế và sự hài lòng của khách hàng.

- **Chăm sóc sức khỏe**

Một thiết bị có thể cảnh báo tình trạng và theo dõi sức khỏe là một trong những ứng dụng trong lĩnh vực y tế. Miếng dãn theo dõi sức khỏe cho bệnh nhân: bạn không cần đến bác sĩ, những thông số về nhịp tim, huyết áp, đều được thu thập từ xa được phân tích sau đó chuẩn đoán để đưa ra tình trạng sức khỏe hiện tại của bệnh nhân và có thể dự đoán nguy cơ mắc bệnh nhằm có biện pháp phòng ngừa kịp thời.

Hình 1.3 Internet of things trong lĩnh vực y tế

- **Nông nghiệp (Smart Farming)**

Mô hình nhà kính là một trong những ứng dụng điển hình của công nghệ IoT được áp dụng trong lĩnh vực nông nghiệp. Và ở nước ta đã được áp dụng rộng rãi. Bên trong hệ thống này cây trồng hoàn toàn cách ly với điều kiện thời tiết bên ngoài, việc điều khiển nhiệt độ, độ ẩm, ánh sáng đều tự động hóa. Đồng thời theo

dõi được tình trạng phát triển của cây trồng, xác định thời gian thu hoạch, giảm thiểu tối đa công suất người lao động.

Hinh 1.4 Internet of things trong sản xuất nông nghiệp

- **Thành phố thông minh (Smart City)**

Có thể xem đây là tập hợp của tất cả ứng dụng của IoT vào một hệ thống lớn. Một giải pháp đã và đang được nhiều quốc gia trên thế giới áp dụng ở các thành phố lớn nhằm giải quyết những vấn đề cấp bách như tình trạng kẹt xe, gia tăng dân số, ô nhiễm môi trường, ngập lụt, ...

Mọi thứ trong thành phố thông minh này được kết nối, dữ liệu sẽ được giám sát bởi một loạt các máy tính mà không cần bất kỳ sự tương tác nào của con người.

Hình 1.5 Internet of things trong thành phố thông minh

1.2. Các chuẩn giao tiếp được sử dụng

1.2.1. Chuẩn Ethernet

Ethernet là một họ lớn và đa dạng gồm các công nghệ mạng dựa vào khung dữ liệu dành cho mạng LAN. Tên ETHERNET được xuất phát từ khái niệm Ête trong ngành vật lý học. Ethernet định nghĩa một loạt các định nghĩa nối dây và phát tín hiệu cho tầng vật lý. Hai phương tiện để truy nhập tại phần MAC (*điều khiển truy nhập môi trường truyền dẫn)* của tầng liên kết dữ liệu và một định dạng chung cho việc định địa chỉ. Vì vậy khi nhắc đến Ethernet ta sẽ liên hệ đến lớp 1 và lớp 2 trong mô hình OSI. Hai lớp này thuộc về phần cứng Ethernet , trong khi các lớp còn lại thuộc về việc xử lý của phần mềm.

Ethernet đã được chuẩn hóa thành IEEE 802.3. cấu trúc mạng hình sao, hình thức nối dây cáp xoắn (*twisted pair*) đã trở thành công nghệ mạng LAN được sử dụng rộng rãi nhất từ thập niên 1990 cho tới nay. Trong những năm gần đây, mạng wi-fi, dạng LAN không dây đã được chuẩn hóa bởi IEEE 802.11, đã được sử dụng bên cạnh hoặc thay thế Ethernet trong nhiều cấu hình mạng.

- **Cấu trúc khung tin Ethernet**

-Các chuẩn Ethernet đều hoạt động ở tầng Data Link trong mô hình 7 lớp OSI vì thế đơn vị dữ liệu mà các trạm trao đổi với nhau là các khung (frame). Cấu trúc khung Ethernet như sau:

Mở đầu 555…5H	SFD (D5H)	Địa chỉ đích	Địa chỉ nguồn	Độ dài kiểu gói	Dữ liệu	PAD	FCS

7byte	1 byte	2/6 byte	2/6 byte	2 byte	46-1500 byte	4 byte

Bảng 1.1: Cấu trúc khung MAC theo IEEE 802.3/ Ethernet

-Preamble (mở đầu): trường này đánh dấu sự xuất hiện của khung bit, nó luôn mang giá trị 10101010. Từ nhóm bit này, phía nhận có thể tạo ra xung đồng hồ 10 Mhz.

-SFD (start frame delimiter): trường này mới thực sự xác định sự bắt đầu của 1 khung. Nó luôn mang giá trị 10101011.

-Các trường Destination và Source: mang địa chỉ vật lý của các trạm nhận và gửi khung, xác định khung được gửi từ đâu và sẽ được gửi tới đâu.

-LEN: giá trị của trường nói lên độ lớn của phần dữ liệu mà khung mang theo. FCS mang CRC (cyclic redundancy checksum): phía gửi sẽ tính toán trường này trước khi truyền khung. Phía nhận tính toán lại CRC này theo cách tương tự. Nếu hai kết quả trùng nhau, khung được xem là nhận đúng, ngược lại khung coi như là lỗi và bị loại bỏ.

- **Cấu trúc địa chỉ Ethernet**

Mỗi giao tiếp mạng Ethernet được định danh duy nhất bởi 48 bit địa chỉ (6 octet). Đây là địa chỉ được ấn định khi sản xuất thiết bị, gọi là địa chỉ MAC. (Media Access Control Address). Địa chỉ MAC được biểu diễn bởi các chữ số hexa (hệ cơ số 16). Ví dụ:00:60:97:8F:4F:86 hoặc 00-60-97-8F-4F-86.Khuôn dạng địa chỉ MAC được chia làm 2 phần:

-3 octet đầu xác định hãng sản xuất, chịu sự quản lý của tổ chức IEEE.

-3 octet sau do nhà sản xuất ấn định.

-Kết hợp ta lẽ có một địa chỉ MAC duy nhất cho một giao tiếp mạng Ethernet. Địa chỉ MAC được sử dụng làm địa chỉ nguồn và địa chỉ đích trong khung Ethernet.

- **Các loại khung Ethernet**
  - **Khung unicast**

-Khung này được truyền tới một trạm xác định. Tất cả các trạm trong phân đoạn mạng trên sẽ đều nhận được khung này nhưng:

-Chỉ có trạm 2 thấy địa chỉ MAC đích của khung trùng với địa chỉ MAC của giao tiếp mạng của mình nên tiếp tục xử lý các thông tin khác trong khung.

-Các trạm khác sau khi so sánh địa chỉ sẽ bỏ qua không tiếp tục xử lý khung nữa.

  - **Khung broadcast**

Các khung broadcast có địa chỉ MAC đích là FF-FF-FF-FF-FF-FF. Khi nhận được các khung này, mặc dù không trùng với địa chỉ MAC của giao tiếp mạng của mình nhưng các trạm đều phải nhận khung và tiếp tục xử lý.

- **Khung multicast**

Trạm nguồn gửi khung tới một số trạm nhất định chứ không phải là tất cả. Địa chỉ MAC đích của khung là địa chỉ đặc biệt mà chỉ các trạm trong cùng nhóm mới chấp nhận các khung gửi tới địa chỉ này.

1.2.2. Chuẩn IP

IP được thiết kế để sử dụng trong những hệ thống liên kết bởi các mạng truyền thông máy tính chuyển mạch gói. IP truyền các gói dữ liệu đi từ nơi gửi đến nơi nhận, trong đó nơi gửi và nơi nhận là các máy được xác định bởi các địa chỉ có độ dài cố định. IP cũng cung cấp tính năng phân mảnh và đóng gói các gói tin dài nếu cần thiết để truyền tin qua những mạng có lưu lượng thấp. IP không có cơ chế đảm bảo sự tin cậy của dữ liệu truyền đi, điều khiển luồng, sắp thứ tự hay các dịch vụ phổ biến thường thấy trong các giao thức từ máy chủ đến máy chủ. IP có thể sử dụng chính các dịch vụ của các mạng máy tính hỗ trợ cho nó để cung nhiều loại dịch vụ.

IP phiên bản 0 đến 3 (IPv0 - IPv3) là các phiên bản được sử dụng trong khoảng thời gian từ năm 1977 đến năm 1979. Tháng 09/1981, IETF phát hành IP phiên bản 4 (IPv4) tại RFC 791 thay thế cho RFC 760 ban hành tháng 01/1980 và phiên bản này được sử dụng phổ biến nhất từ đó đến nay (RFC - Request for Comments, là những tài liệu kỹ thuật và tổ chức về Internet, bao gồm những tài liệu đặc tả kỹ thuật và chính sách được tổ chức IETF phát hành). Bên cạnh đó là những đặc tả mở rộng cho IPv4 như sau:

IP phiên bản 5 (IPv5) được dùng trong Giao thức luồng Internet (Internet Stream Protocol) và chỉ là giao thức thử nghiệm, chưa bao giờ được sử dụng.

Để giải quyết bài toán cạn kiệt địa chỉ Internet, có nhiều phiên bản IP được đề xuất, từ 6 đến 9 nhưng chỉ có phiên bản 6 (v6) chính thức được công nhận và sẽ được triển khai rộng khắp. Các RFC chính về IPv6: RFC 2460: Đặc tả về giao thức IPv6 (*Internet Protocol, Version 6 (IPv6) Specification*); RFC 2461: Giao thức tìm kiếm các nút cho IPv6 (*Neighbor Discovery for IP Version 6 (IPv6));* RFC 2462: Xác định cách thức tự động cấu hình trên các giao diện trong IPv6 *(IPv6 Stateless Address Autoconfiguration*); RFC 4443: Giao thức thông điệp điều khiển Internet cho đặc tả IPv6 (*Internet Control Message Protocol (ICMPv6) for the IPv6 Specification*); RFC 2464: Truyền dẫn gói tin IPv6 động trong môi trường mạng Ethernet (*Transmission of IPv6 Packets over Ethernet Networks*); RFC 4291: Định nghĩa kiến trúc địa chỉ IPv6, (*Internet Protocol Version 6 (IPv6) Addressing Architecture*).

Trước tình hình cạn kiệt IPv4, thế hệ địa chỉ IPv6 đang được quan tâm thúc đẩy trên nhiều lĩnh vực. Số liệu thống kê thông số về IPv6 gia tăng một cách đáng kể và đều đặn trên Internet đã phản ánh mức độ tăng trưởng trong triển khai IPv6. IPv6 cũng được ghi nhận chính thức trong hoạt động Internet thông qua các sự kiện toàn cầu về IPv6 như Khai trương IPv6 toàn cầu (World IPv6 Launch 06/06/2012), Ngày IPv6 thế giới (World IPv6 Day 08/06/2011). Tại Việt Nam, Bộ

Thông tin và Truyền thông đã thành lập Ban công tác thúc đẩy IPv6 quốc gia và ban hành "Kế hoạch hành động Quốc gia về IPv6" với các định hướng, xác định các mục tiêu, lộ trình cụ thể chuyển đổi sang IPv6 của quốc gia là cơ sở để các doanh nghiệp Internet xây dựng kế hoạch chuyển đổi, ứng dụng IPv6 phù hợp với tình hình thực tế và mạng lưới của đơn vị mình. Với vai trò là đơn vị quản lý mạng DNS quốc gia và trạm trung chuyển Internet quốc gia (Vietnam National Internet eXchange - VNIX), Trung tâm Internet Việt Nam (Vietnam Internet Network Information Center - VNNIC) phối hợp các nhà cung cấp dịch vụ Internet (Internet Service Provider - ISP) thiết lập mạng IPv6 quốc gia theo lộ trình chuyển đổi IPv6, đồng thời cung cấp các dịch vụ thử nghiệm như hệ thống tên miền, web, thư điện tử, thoại trên nền IP tới người sử dụng cuối.

- **Chức năng hoạt động của IP**

IP thực hiện hai chức năng cơ bản là đánh địa chỉ và phân mảnh. Các gói tin được định tuyến từ một mô-đun internet (internet module) đến một mô-đun internet khác trong hệ thống thông qua sự biên dịch một địa chỉ internet, do đó chức năng quan trọng đầu tiên của IP là đánh địa chỉ internet. Chức năng thứ hai của IP là phân mảnh gói tin, khi truyền gói tin có kích thước lớn qua nhiều mạng, phân mảnh giúp kích thước gói tin giảm đi trước khi đến được đích, sau đó lại được đóng gói lại như gói tin ban đầu.

Các mô-đun internet sử dụng các địa chỉ lưu trong tiêu đề internet để truyền tải gói dữ liệu internet đối với điểm đích. Việc lựa chọn một đường đi để truyền tin được gọi là định tuyến. Các mô-đun internet sử dụng các trường trong tiêu đề internet để chia nhỏ và tổng hợp lại các gói tin internet khi cần thiết để truyền qua những mạng có băng thông nhỏ. Mô hình hoạt động của IP ở đây là một mô-đun internet nằm trong mỗi máy chủ tham gia vào quá trình truyền thông internet và trong mỗi cổng kết nối các mạng. Những mô-đun này chia sẻ các quy tắc chung để biên dịch các trường địa chỉ và phân mảnh, tổng hợp gói dữ liệu internet. Ngoài ra, các mô-đun (đặc biệt là trong các cổng kết nối) có các quy định để ra quyết định định tuyến và các chức năng khác. IP xử lý mỗi gói tin internet như là một thực thể độc lập và không liên quan đến bất kỳ gói internet khác. IP sử dụng bốn cơ chế quan trọng để cung cấp dịch vụ: Loại dịch vụ (Type of Service), Thời gian tồn tại (Time to Live), Tùy chọn (Options) và Kiểm tra lỗi tiêu đề (Header Checksum).

Các loại dịch vụ được sử dụng để chỉ ra chất lượng dịch vụ mong muốn. Các loại dịch vụ là một tập hợp trừu tượng hoặc tổng quát các tham số đặc trưng cho

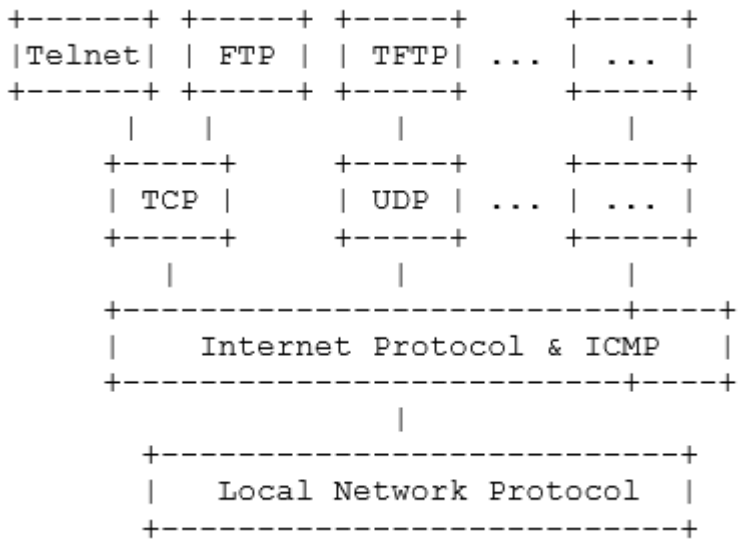
những lựa chọn dịch vụ được cung cấp trong các mạng tạo nên internet. Loại dịch vụ này chỉ được sử dụng bởi các cổng kết nối để lựa chọn các tham số truyền thực tế cho một mạng cụ thể, mạng để được sử dụng cho bước kế tiếp hoặc cho cổng kết nối tiếp theo khi định tuyến một gói tin internet, đến đích, gói tin internet sẽ bị hủy. Thời gian tồn tại có thể xem như một giới hạn thời gian tự hủy.

Tùy chọn cung cấp các chức năng kiểm soát cần thiết hoặc hữu ích trong một số tình huống, nhưng không cần thiết cho việc truyền thông tin phổ biến nhất. Các tùy chọn bao gồm quy định về thời gian lưu của hệ thống, an ninh và đặc biệt là định tuyến.

Kiểm tra lỗi tiêu đề xác minh các thông tin sử dụng để xử lý gói tin internet đã được truyền đi chính xác, dữ liệu có thể bị lỗi hay không. Nếu kiểm tra lỗi phần tiêu đề không thành công, gói tin internet bị loại bỏ ngay lập tức. IP không cung cấp tính năng truyền thông đáng tin cậy cơ sở, không có sự xác nhận giữa các điểm truyền nhận, không có kiểm soát lỗi dữ liệu, chỉ có một kiểm tra tiêu đề, không truyền lại và không kiểm soát luồng dữ liệu. Lỗi được phát hiện có thể được báo cáo thông qua Giao thức thông điệp điều khiển Internet (Internet Control Message Protocol - ICMP) được cài đặt trong mô-đun IP.

- **Quan hệ của IP với giao thức khác**

Sơ đồ dưới đây minh họa vị trí của IP trong hệ thống phân cấp giao thức:



Hình 1.6 Quan hệ giữa các giao thức
(Nguồn: *Internet Protocol, Darpa Internet Program, Protocol Specification*)

Các giao diện IP nằm dưới các giao thức lớp ứng dụng (truyền thông điệp từ máy chủ đến máy chủ), ví dụ Giao thức điều khiển truyền tin (Transmission Control Protocol - TCP) và nằm trên giao thức mạng cục bộ (Local Network Protocool). Trong trường hợp này " mạng cục bộ "có thể là một mạng nhỏ trong mạng rộng lớn như ARPANET. IP và TCP là hai giao thức quan trọng của bộ giao thức Internet (gọi là bộ giao thức hay mô hình TCP/IP).

- **Địa chỉ IPv4**

Mỗi địa chỉ IP được chia thành 2 phần: Phần địa chỉ mạng (Net ID) và Phần địa chỉ máy (Host ID), trong đó:

Net ID dùng để nhận dạng những hệ thống trong cùng một khu vực vật lý. Mọi hệ thống trong cùng một khu vực vật lý phải có cùng địa chỉ mạng và địa chỉ mạng phải là duy nhất trong số các mạng hiện có.

Host ID dùng để nhận dạng một máy có thể là máy khách, máy chủ, bộ định tuyến và địa chỉ máy cũng phải là duy nhất trong một mạng.

Sự kết hợp giữa Net ID và Host ID cho phép nhận dạng duy nhất mỗi máy vi tính riêng biệt.

Mỗi địa chỉ IPv4 có độ dài 32 bít được chia thành 4 vùng (mỗi vùng 1 byte), có thể biểu thị dưới dạng thập phân, bát phân, thập lục phân hay nhị phân. Cách viết phổ biến nhất là dùng ký pháp thập phân có dấu chấm (dotted decimal notation) để tách các vùng. Số địa chỉ IPv4 tối đa có thể sử dụng là 4,3 x 10^9 địa chỉ. Tuy nhiên, do một số địa chỉ được sử dụng cho các mục đích khác như: cấp cho mạng cá nhân (xấp xỉ 18 triệu địa chỉ) hoặc sử dụng làm địa chỉ quảng bá (xấp xỉ 16 triệu), nên số lượng địa chỉ thực tế có thể sử dụng cho mạng Internet công cộng bị giảm xuống.

- Có 5 lớp địa chỉ IPv4 là: Lớp A, Lớp B, Lớp C, Lớp D, Lớp E, trong đó các byte đầu tiên (từ một đến ba) xác định Net ID để chia các lớp mạng A, B, C:

+ Lớp A được gán cho các mạng có kích thước rất lớn, bít cao nhất của byte này được gán bằng 0, ba byte còn lại xác định Host ID. Do đó lớp A có thể cấp cho 126 mạng với 16.777.214 máy trên mỗi mạng.

+ Lớp B được gán cho các mạng có kích thước vừa và lớn, hai byte đầu tiên xác định Net ID, hai bít cao nhất của byte đầu tiên được gán bằng 10, hai byte còn lại xác định Host ID. Do đó lớp B có thể cấp cho 16.384 mạng với 65.534 máy trên mỗi mạng.

+ Lớp C được gán cho các mạng có kích thước nhỏ, ba byte đầu tiên xác định NET ID, ba bít cao nhất của byte đầu tiên được gán bằng 110, byte cuối cùng xác định Host ID. Do đó lớp C có thể cấp cho 2.097.152 mạng với 254 máy trên mỗi mạng.

+ Lớp D: Các địa chỉ lớp D sử dụng cho dạng multicast. Một nhóm multicast có thể chứa một hoặc nhiều máy. Trong lớp này bốn bít cao nhất của byte đầu tiên luôn được gán bằng 1110, các bít còn lại xác định nhóm multicast. Địa chỉ lớp D

không được chia thành Net ID và Host ID. Các gói tin dạng multicast được truyền tới một nhóm máy cụ thể và chỉ có các máy đăng ký vào nhóm này mới nhận được gói tin.

+ Lớp E: Các địa chỉ lớp E không được thiết kế cho mục đích sử dụng chung. Lớp E được dự phòng cho các ứng dụng tương lai. Các bít cao nhất của byte đầu tiên luôn được gán bằng 1111.

- IPv6 được thiết kế để thay thế cho IPv4 với các mục tiêu quan trọng như sau:

+ Mở rộng khả năng đánh địa chỉ: IPv6 tăng kích thước địa chỉ IP từ 32 bít lên 128 bít, nghĩa là có 3,4 x 10^{38} địa chỉ, hỗ trợ nhiều mức phân cấp địa chỉ, số lượng nút có thể đánh địa chỉ nhiều hơn và tự động cấu hình đơn giản hơn, gia tăng khả năng multicast.

+ Đơn giản hóa định dạng trường tiêu đề trong cấu trúc địa chỉ IP để giảm chi phí xử lý gói tin và chi phí băng thông.

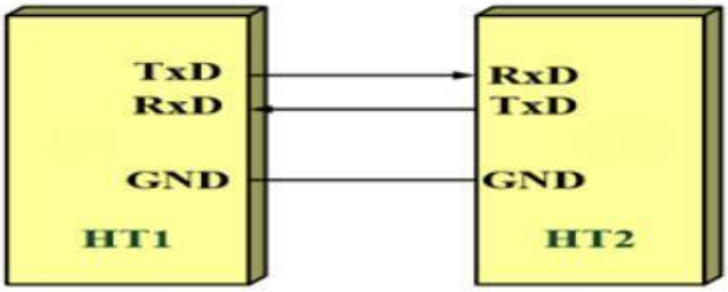
+ Cải thiện hỗ trợ mở rộng và tùy chọn trong trường tiêu đề giúp chuyển tin hiệu quả, giảm các hạn chế và mở rộng khả năng linh hoạt của độ dài trường tùy chọn.

+ Khả năng đánh dấu luồng: Cho phép đánh dấu các gói tin thuộc các luồng mà nơi gửi yêu cầu xử lý.

+ Khả năng chứng thực và riêng tư: Đây là sự mở rộng để hỗ trợ chứng thực, toàn vẹn dữ liệu, bí mật dữ liệu (tùy chọn) trong IPv6 so với IPv4.

1.2.3. Chuẩn giao tiếp UART

UART là viết tắt của Universal Asynchronous Receiver – Transmitter có nghĩa là truyền dữ liệu nối tiếp bất đồng bộ. Truyền dữ liệu nối tiếp bất đồng bộ có 1 đường phát dữ liệu và 1 đường nhận dữ liệu, không có tín hiệu xung clock nên gọi là bất đồng bộ. Để truyền được dữ liệu thì cả bên phát và bên nhận phải tự tạo xung clock có cùng tần số và thường được gọi là tốc độ baud, 9600 baud...



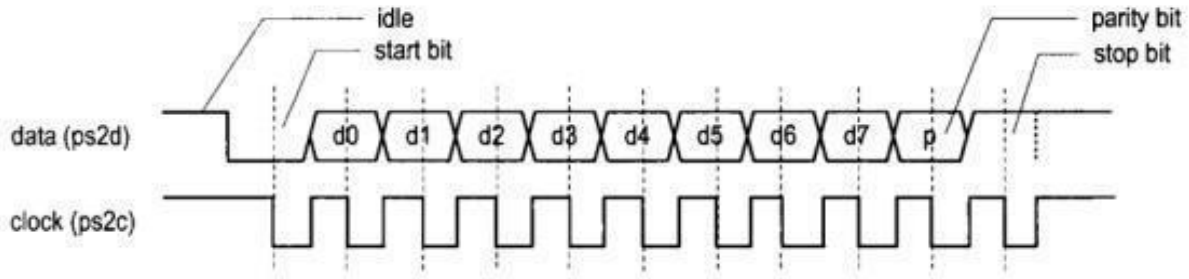
Hình 1.7 Hình ảnh các chân giao tiếp của UART

Giao tiếp UART chế độ bất đồng bộ sử dụng một dây kết nối cho mỗi chiều truyền dữ liệu do đó để quá trình truyền nhận dữ liệu thành công thì việc tuân thủ các tiêu chuẩn truyền là hết sức quan trọng. Sau đây là các khái niệm quan trọng trong chếđộ truyền thông này

Baud rate (tốc độ Baud): Để việc truyền và nhận bất đồng bộ xảy ra thành công thì các thiết bị tham gia phải thống nhất với nhau về khoảng thời gian dành cho 1 bit truyền, hay nói cách khác tốc độ truyền phải được cài đặt như nhau trước khi truyền nhận, tốc độ này gọi là tốc độ Baud. Tốc độ Baud là số bit truyền trong một giây. Ví dụ, nếu tốc độ Baud được đặt là 9600 bit/giây thì thời gian dành cho một bit truyền là 1/9600~104.167us.

Frame (khung truyền): Do truyền thông nối tiếp mà nhất là nối tiếp bất đồng bộ rất dễ mất hoặc sai lệch dữ liệu, quá trình truyền thông theo kiểu này phải tuân theo một số quy cách nhất định. Bên cạnh tốc độ Baud, khung truyền là một yếu tố quan trọng tạo nên sự thành công khi truyền và nhận. Khung truyền bao gồm các quy định về số bit trong mỗi lần truyền, các bit báo hiệu như bit Start và bit Stop, các bit kiểm tra như Parity, ngoài ra số lượng các bit dữ liệu trong mỗi lần truyền cũng được quy định bởi khung truyền

Để bắt đầu cho việc truyền dữ liệu bằng UART, một START bit được gửi đi,



sau đó là các bit dữ liệu và kết thúc quá trình truyền là STOP bit.

Hình 1.8 Sơ đồ truyền dữ liệu bằng UART

Start bit: Là bit đầu tiên được truyền trong một khung truyền, bit này có chức năng báo cho thiết bị nhận biết rằng có một gói dữ liệu sắp được truyền tới. Start bit là bit bắt buộc phải có trong khung truyền.

Data: Data hay dữ liệu cần truyền là thông tin chính mà chúng ta cần gửi và nhận. Dữ liệu cần truyền không nhất thiết phải là gói 8 bit, có thể quy định số lượng bit của dữ liệu là 5, 6, 7, 8 hoặc 9. Trong truyền thông nối tiếp USART, bit có ảnh hưởng nhỏ nhất của dữ liệu sẽ được truyền trước và cuối cùng là bit có ảnh hưởng lớn nhất.

Parity bit: Là bit dùng kiểm tra dữ liệu truyền đúng không (một cách tương đối). Có 2 loại parity là parity chẵn và parity lẻ. Parity chẵn nghĩa là số lượng bit 1 trong dữ liệu bao gồm bit parity luôn là số chẵn. Ngược lại, tổng số lượng các bit 1 trong parity lẻ luôn là lẻ. Ví dụ, nếu dữ liệu của bạn là 10111011 nhị phân, có tất cả 6 bit 1 trong dữ liệu này, nếu parity chẵn được dùng, bit parity sẽ mang giá trị 0 để đảm bảo tổng các bit 1 là số chẵn (6 bit 1). Nếu parity lẻ được yêu cầu thì giá trị của parity bit là 1. Parity bit không phải là bit bắt buộc và vì thế chúng ta có thể loại bit này khỏi khung truyền.

Stop bits: Là một hoặc các bit báo cho thiết bị nhận rằng một gói dữ liệu đã được gửi xong. Sau khi nhận được stop bits, thiết bị nhận sẽ tiến hành kiểm tra khung truyền để đảm bảo tính chính xác của dữ liệu. Stop bits là các bit bắt buộc xuất hiện trong khung truyền. Khung truyền phổ biến nhất là (Start bit + 8 bit dữ liệu + stop bit).

1.2.4. Công nghệ Wifi

- **Các chuẩn của wifi**

Wifi là viết tắt của từ Wireless Fidelity trong tiếng Anh, được gọi chung là mạng không dây sử dụng sóng vô tuyến. Wifi là loại sóng vô tuyến tương tự như sóng điện thoại, sóng truyền hình và radio. Hầu hết các thiết bị sử dụng điện tử hiện nay như : Smartphone, Máy tính bảng, Tivi, Laptop… đều có thể kết nối được WiFi. Và Wifi là thứ gắn liền và không thể thiếu với đời sống của người dân trong hầu hết công việc cũng như giải trí hàng ngày. Chúng truyền và phát tín hiệu ở tần số 2.4 GHz hoặc 5 GHz. Tần số này cao hơn so với các tần số sử dụng cho điện thoại di động, các thiết bị cầm tay và truyền hình. Tần số cao hơn cho phép tín hiệu mang theo nhiều dữ liệu hơn

- Chuẩn 802.11

IEEE 802.11 là một tập các chuẩn của tổ chức IEEE. Chuẩn IEEE 802.11 mô tả một giao tiếp “truyền qua không khí” (tiếng Anh: over-the-air) sử dụng sóng vô tuyến để truyền nhận tín hiệu giữa một thiết bị không dây và tổng đài hoặc điểm truy cập (tiếng Anh: access point), hoặc giữa 2 hay nhiều thiết bị không dây với nhau. Năm 1997, IEEE giới thiệu chuẩn mạng không dây đầu tiên và đặt tên nó là

802.11. Khi đó, tốc độ hỗ trợ tối đa của mạng này chỉ là 2 Mbps với bang tầng 2.4GHz.

- Chuẩn 802.11b

IEEE đã mở rộng trên chuẩn 802.11 gốc vào tháng Bảy năm 1999, đó chính là chuẩn 802.11b. Chuẩn này hỗ trợ băng thông lên đến 11Mbps, tương quan với Ethernet truyền thống. 802.11b sử dụng tần số vô tuyến (2.4 GHz) giống như chuẩn ban đầu 802.11. Các hãng thích sử dụng các tần số này để chi phí trong sản xuất của họ được giảm. Các thiết bị 802.11b có thể bị xuyên nhiễu từ các thiết bị điện thoại không dây (kéo dài), lò vi sóng hoặc các thiết bị khác sử dụng cùng dải tần 2.4 GHz. Mặc dù vậy, bằng cách cài đặt các thiết bị 802.11b cách xa các thiết bị như vậy có thể giảm được hiện tượng xuyên nhiễu này

+ Ưu điểm của 802.11b – giá thành thấp nhất; phạm vi tín hiệu tốt và không dễ bị cản trở.

+ Nhược điểm của 802.11b – tốc độ tối đa thấp nhất; các ứng dụng gia đình có thể xuyên nhiễu.

- Chuẩn 802.11a

Được phát triển song song cùng với chuẩn 802.11b, chuẩn 802.11a hỗ trợ tốc độ tối đa gần gấp 5 lần lên đến 54 Mpbs và sử dụng bằng tầng 5Ghz nhằm tránh bị nhiễu từ các thiết bị khác. Tuy nhiên, đây cũng là nhược điểm của chuẩn này vì phạm vi phát sẽ hẹp hơn (40-100m) và khó xuyên qua các vật cản như vách tường. Chuẩn này thường được sử dụng trong các mạng doanh nghiệp thay vì gia đình vì giá thành của nó khá cao.

- Chuẩn 802.11g

Năm 2003, chuẩn Wifi thế hệ thứ 3 ra đời và mang tên 802.11g. Chuẩn này được kết hợp từ chuẩn a và b. Được hỗ trợ tốc độ 54Mpbs như chuẩn a và sử dụng băng tầng 2.4GHz của chuẩn b vì vậy chuẩn này có phạm vi tín hiệu khá tốt (80-200m) và vẫn dễ bị nhiễu từ các thiết bị điện tử khác. Ngày nay, một số hộ gia đình vẫn còn sử dụng chuẩn này .

+ Ưu điểm của 802.11g – tốc độ cao; phạm vi tín hiệu tốt và ít bị che khuất.

+ Nhược điểm của 802.11g – giá thành đắt hơn 802.11b; các thiết bị có thể bị xuyên nhiễu từ nhiều thiết bị khác sử dụng cùng băng tần.

- Chuẩn 802.11n (hay 802.11 b/g/n)

Đây là chuẩn được sử dụng phổ biến nhất hiện nay và tương đối mới. Chuẩn WiFi 802.11n được đưa ra nhằm cải thiện chuẩn 802.11g bằng cách sử dụng công nghệ MIMO (Multiple-Input Multiple-Output) tận dụng nhiều anten hơn. Chuẩn kết nối 802.11n hỗ trợ tốc độ tối đa lên đến 600 Mpbs, có thể hoạt động trên cả băng tần 2,4 GHz và 5 GHz, nếu router hỗ trợ thì hai băng tần này có thể cùng phát sóng song song. Chuẩn kết nối này đã và đang dần thay thế chuẩn 802.11g với tốc độ cao, phạm vi tín hiệu rất tốt (từ 100-250m) và giá thành đang ngày càng phù hợp với túi tiền người tiêu dùng.

+ Ưu điểm của 802.11n – tốc độ nhanh và phạm vi tín hiệu tốt nhất; khả năng chịu đựng tốt hơn từ việc xuyên nhiễu từ các nguồn bên ngoài.

+ Nhược điểm của 802.11n – chuẩn vẫn chưa được ban bố, giá thành đắt hơn 802.11g; sử dụng nhiều tín hiệu có thể gây nhiễu với các mạng 802.11b/g ở gần.

- Chuẩn 802.11ac (hay chuẩn 802.11 a/b/g/n/ac)

Trong khoảng một vài năm trở lại đây chúng ta được nghe nhắc nhiều đến chuẩn Wi-Fi 802.11ac, hay còn gọi là Wi-Fi thế hệ thứ năm. Nó là chuẩn mạng không dây đang ngày càng xuất hiện nhiều hơn trên các router, máy tính và tất nhiên là cả các thiết bị di động như smartphone. So với Wi-Fi 802.11n đang được dùng phổ biến hiện nay, chuẩn 802.11ac mang lại tốc độ nhanh hơn. Là chuẩn Wifi mới nhất được IEEE giới thiệu. Chuẩn ac có hoạt động ở băng tầng 5 GHz và tốc độ tối đa lên đến 1730 Mpbs khi sử dụng lại công nghệ đa anten trên chuẩn 802.11n cho người dùng trải nghiệm tốc độ cao nhất. Hiện tại, chuẩn này được sử dụng trên một số thiết bị cao cấp của các hang điện thoại như Apple, Samsung, Sony,… Tuy nhiên, do giá thành khá cao nên các thiết bị phát tín hiệu cho chuẩn này chưa được phổ biến trên thị trường nên mặc dù các thiết bị này không hoạt động tối ưu khi sử dụng bởi sự hạn chế của các thiết bị phát.

CHƯƠNG 2 :ARDUINO

2.1. Tổng quan Arduino

- Arduino Uno là một board mạch vi điều khiển được phát triển bởi Arduino.cc, một nền tảng điện tử mã nguồn mở chủ yếu dựa trên vi điều khiển AVR Atmega328P. Với Arduino chúng ta có thể xây dựng các ứng dụng điện tử tương tác với nhau thông qua phần mềm và phần cứng hỗ trợ.

- Khi arduino chưa ra đời, để làm được một dự án điện tử nhỏ liên quan đến lập trình, biên dịch, chúng ta cần đến sự hỗ trợ của các thiết bị biên dịch khác để hỗ trợ. Ví dụ như, dùng Vi điều khiển PIC hoặc IC vi điều khiển họ 8051..., chúng ta phải thiết kế chân nạp onboard, hoặc mua các thiết bị hỗ trợ nạp và biên dịch như mạch nạp 8051, mạch nạp PIC...

- Hiện nay Arduino được biết đến ở Việt Nam rất rộng rãi. Từ học sinh trung học, đến sinh viên và người đi làm. Những dự án nhỏ và lớn được thực hiện một cách rất nhanh, các mã nguồn mở được chia sẻ nhiều trên diễn dàn trong nước và nước ngoài. Giúp ích rất nhiều cho những bạn theo đam mê nghiên cứu chế tạo những sản phẩm có ích cho xã hội.

- Trong những năm qua, Arduino là bộ não cho hàng ngàn dự án điện tử lớn nhỏ, từ những sản phẩm ra đời ứng dụng đơn giản trong cuộc sống đến những dự án khoa học phức tạp.

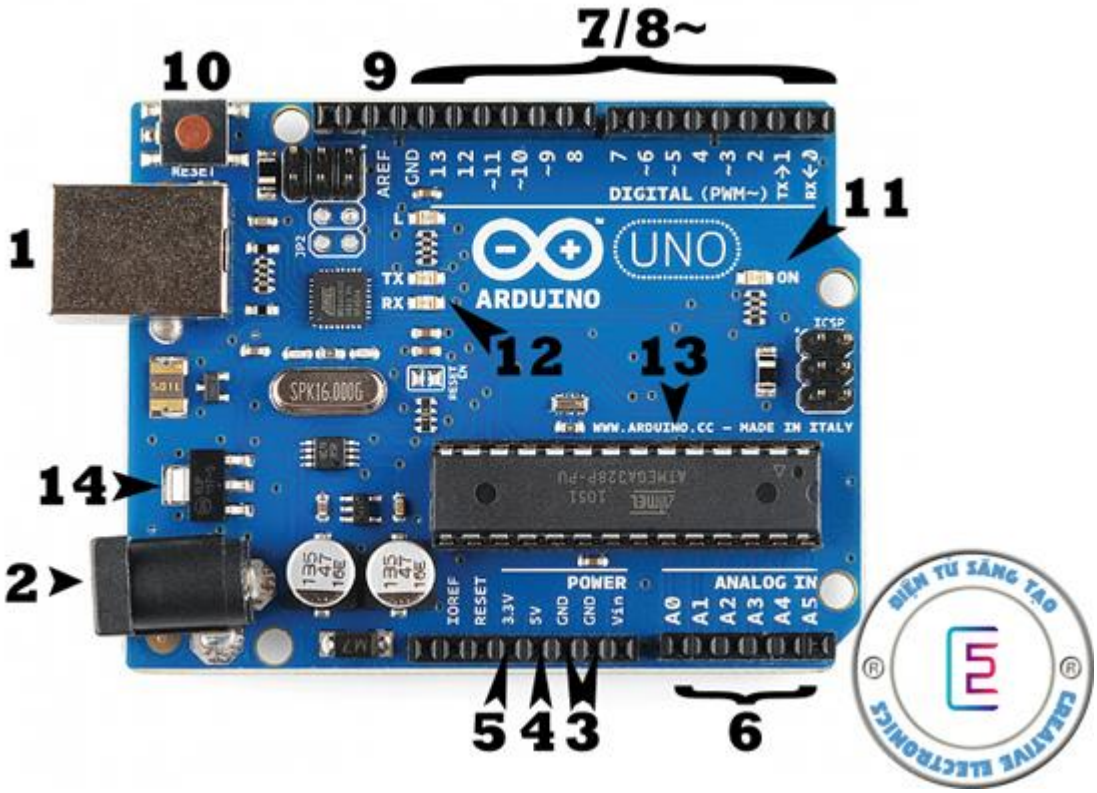
Cứ như vậy, thư viện mã nguồn mở ngày một tăng lên, giúp ích cho rất nhiều người mới biết đến Arduino cũng như những chuyên viên lập trình nhúng và chuyên gia cùng tham khảo và xây dựng tiếp nối....

- Bạn muốn thiết kế điều khiển thiết bị thông qua cảm biến ánh sáng, Đo nồng độ hóa chất, khí ga và xử lý thông qua cảm biến nồng độ và cảm biến khí, Bạn muốn làm 1 con robot mini, Bạn muốn quản lý tắt mở thiết bị điện trong nhà, bạn muốn điều khiển motor, nhận dạng ID, Khó hơn xíu là bạn muốn làm một máy CNC hoặc máy in 3D mini, máy bay không người lái (Flycam) một hệ thống thu thập dữ liệu thông qua GSM, xử lý ảnh,điều khiển vạn vật thông qua internet giao tiếp với điện thoại thông minh...

- Để làm được điều đó, từ đơn giản đến phức tạp bạn cần sử dụng ngôn ngữ lập trình Arduino dựa trên sơ đồ, hệ thống của bạn thiết kế, thông qua phần mềm Arduino IDE, để thực hiện những yêu cầu đó đưa về bộ phận xử lý trung tâm (Arduino).

2.2. Arduino Uno R3

2.2.1. Cấu tạo



Hình 2.1 Vị trí các chân của Arduino uno

❖ **Nguồn (USB / Barrel Jack)**

Mỗi mạch Arduino đều có cổng kết nối với nguồn điện. Cụ thể trên đây mà mạch Arduino UNO có thể được lấy nguồn từ dây cáp USB từ máy tính của bạn, hoặc một số nguồn DC khác có Jack DC. Trong hình trên nguồn kết nối qua cổng USB được dán nhãn (1) và Jack DC được dán nhãn (2).

Chân kết nối USB cũng là chân để bạn có thể tải code lên bo mạch Arduino.

Lưu ý: Tuyệt đối không được sử dụng nguồn lớn hơn 20V vì với nguồn điện áp này sẽ có thể phá hủy mạch Arduino của bạn. Điện áp được các nhà sản xuất đề nghị cho hầu hết các bo mạch Arduino là từ 6 – 12V.

❖ **Các chân (5V, 3.3V, GND, Digital, Analog, PWM, ISF)**

Các chân trên là nguồn ra mà bạn có thể kết nối dây đầu ra với các tải hoặc một số mạch kết nối bên ngoài. Với các loại Arduino sẽ có thể một số loại chân khác nhau. Ở mỗi chân đều được in các nhãn và ký tự để người sử dụng có thể phân biệt được.

- **GND (3)** : Viết tắt của ‘Ground’ là mass. Có một số chân GND trên Arduino, bất kỳ các chân GND trong số đó có thể được sử dụng để nối mass mạch của bạn.
- **5V (4) & 3.3V (5)** : Chân 5V cung cấp năng lượng 5 volt và chân 3,3V cung cấp 3,3 volt. Hầu hết các thành phần đơn giản được sử dụng với Arduino đều hoạt động bình thường ở mức 5 hoặc 3,3 volt.

- **Analog (6)** : Các chân được dán nhãn 'Analog In' (A0 đến A5 trên UNO) là các chân Analog In. Các chân này có thể đọc tín hiệu từ các cảm biến tương tự (như cảm biến nhiệt độ) và chuyển đổi nó thành một giá trị Digital mà chúng ta có thể đọc được.
- **Digital (7)**: Các chân Digital được dán nhãn từ 0 – 13 trên Arduino UNO, các chân này có thể được sử dụng cho cả đầu vào digital nếu như là các nút nhấn và đầu ra digital nếu như cấp nguồn cho LED.
- **PWM (8)**: Bạn có thể nhìn thấy những dấu (~) nằm ở bên cạnh các chân 3, 5, 6, 9, 10 và 11 trên mạch. Các chân này đều có chức năng hoạt động như các chân Digital thông thường, nhưng cũng có thể sử dụng để điều chế độ rộng xung PWM. Bạn có thể hình dung các chân này có thể được sử dụng mô phỏng đầu ra tín hiệu Analog.
- **ISF (9):** Được viết tắt của cụm từ **Analog Reference,** hầu hết chân này thường không được sử dụng. Đôi khi nó được sử dụng để có thể đặt điện áp tham chiếu trong khoảng từ 0 – 5V làm giới hạn cho các chân đầu vào Analog.

❖ **Nút Reset (Reset Button)**

Nút reset (10) có nhiệm vụ khởi động lại bất kỳ đoạn code nào được tải trên Arduino. Điều này rất hữu ích nếu code của bạn không có vòng lặp nhưng bạn lại muốn kiểm tra chương trình đó nhiều lần.

❖ **Đèn LED báo nguồn (Power LED Indicator)**

Đèn báo được nắp ngay bên phải của chữ UNO, đó là một đèn LED nhỏ được dán nhãn ON (11).

Đèn báo này có nhiệm vụ báo khi có nguồn cấp vào Arduino. Trong một số trường hợp đèn không sáng thì chắc chắn có vấn đề xảy ra. Bạn có thể kiểm tra lại dây cáp USB, nguồn cấp và cả mạch nữa.

❖ **LED TX và RX (TX RX LEDs)**

TX là LED hiển thị tín hiệu truyền đi và RX là hiển thị tín hiệu nhận về. Những tín hiệu này xuất hiện khá nhiều trong các thiết bị điện tử để có thể chỉ ra những chân thực hiện nhiệm vụ truyền tải nối tiếp. Trong trường hợp này, có 2 vị trí trên Arduino UNO là TX và RX (12).

Các LED này có nhiệm vụ thông báo cho người dùng bất cứ khi nào Arduino được nhận hoặc truyền dữ liệu đi. Ví dụ như tải một chương trình lên thì đèn sẽ hiển thị.

❖ **IC chủ (Main IC)**

IC chủ là vị trí số 13. Đây được coi là bộ não của Arduino. IC thường được sử dụng là dòng IC ATmega của công ty ATMEL sản xuất. Việc nhận biết được IC chủ cũng là điều rất quan trọng, vì bạn cần biết mạch của bạn đang sử dụng IC nào để bạn có thể nạp chương trình thích hợp từ phần mềm Arduino.

Thông tin về tên của IC thường được tìm thấy ở phía mặt trên. Nếu bạn muốn tìm hiểu sâu hơn về thông tin của IC bạn có thể đọc thêm tài liệu từ nhà sản xuất.

❖ **Bộ điều chỉnh điện áp (Voltage Regulator)**

Bộ điều chỉnh điện áp (14), không được sử dụng nhiều. Nhiệm vụ của nó là điều chỉnh điện áp, kiểm soát nguồn điện áp đưa vào mạch Arduino.

2.2.2. Tính năng

❖ Thông số kỹ thuật:

- Chip điều khiển: Atmega328P
- Điện áp hoạt động: 5V
- Điện áp đầu vào(khuyên dùng): 7-12V
- Điện áp đầu vào(gới hạn): 6-20V
- Số chân Digital: 14(trong đó 6 cung cấp đầu ra PWM)
- Số chân PWM Digital: 6
- Số chân Analog: 6
- Dòng điện DC trên mỗi chân I/O: 20 m A
- Dòn điện DC trên mỗi chân 3.3V: 50 m A
- Flash Memory: 32kb (Atmega328P) 0.5KB được sử dụng bởi bootloader
- SRAM: 2KB (Atmega328p)
- EEPROM: 1KB (Atmega328P)
- Tốc độ: 16 Mhz
- LED_BUILTIN: 13
- Chiều dài: 68.6mm
- Chiều rộng: 53.4 mm
- Trọng lượng: 25g

2.3. Arduino mega 2560

2.3.1.Cấu tạo

Hình 2.2 Hình ảnh về Arduino Mega 2560

Arduino Mega2560 là một vi điều khiển bằng cách sử dụng ATmega2560.

Bao gồm:

- 54 chân digital (15 có thể được sử dụng như các chân PWM)
- 16 đầu vào analog,
- 4 UARTs (cổng nối tiếp phần cứng),
- 1 thạch anh 16 MHz,
- 1 cổng kết nối USB,
- 1 jack cắm điện,
- 1 đầu ICSP,
- 1 nút reset.

Nó chứa tất cả mọi thứ cần thiết để hỗ trợ các vi điều khiển.

Arduino Mega2560 khác với tất cả các vi xử lý trước giờ vì không sử dụng FTDI chip điều khiển chuyển tín hiệu từ USB để xử lý. Thay vào đó, nó sử dụng ATmega16U2 lập trình như là một công cụ chuyển đổi tín hiệu từ USB. Ngoài ra, Arduino Mega2560 cơ bản vẫn giống Arduino Uno R3, chỉ khác số lượng chân và nhiều tính năng mạnh mẽ hơn, nên các bạn vẫn có thể lập trình cho con vi điều khiển này bằng chương trình lập trình cho Arduino Uno R3.

2.3.2. Tính năng

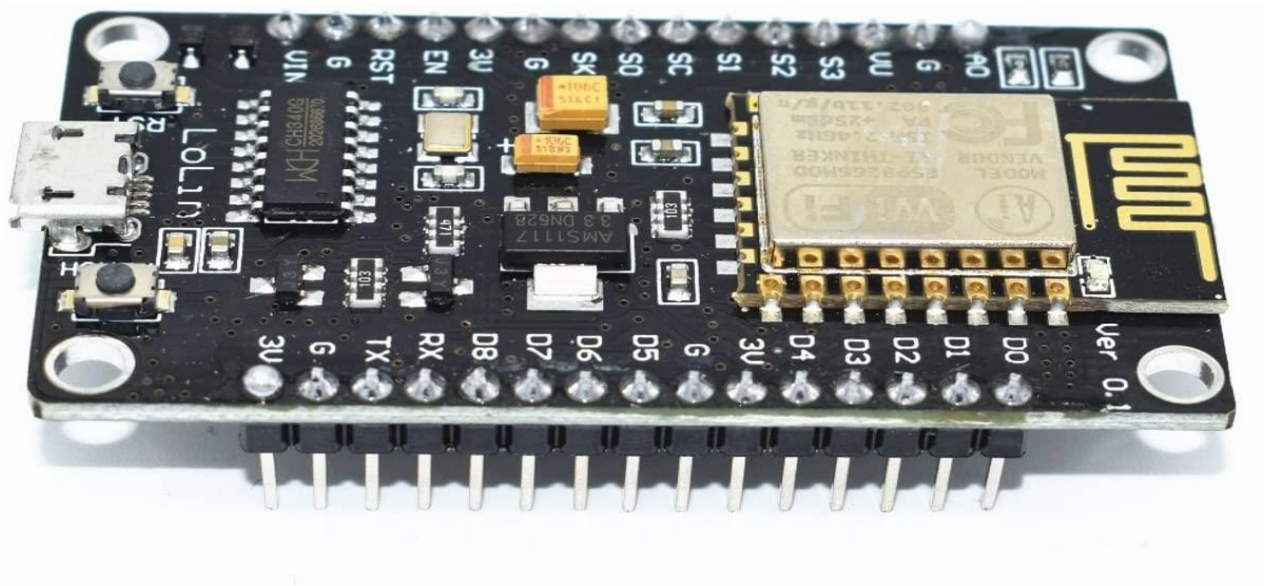
❖ Thông số kỹ thuật:

- Vi điều khiển: Atmega2560
- Điện áp hoạt động: 5V
- Điện áp khuyến nghị: 6-9V
- Số chân I/O: 54 chân(trong đó 15 chân PWM)
- Số chân in put analog: 16 chân
- Giao tiếp UART: 4 bộ

- Giao tiếp SPI: 1 bộ(chân 50-53), dùng thư viện SPI của Arduino
- Giao tiếp I2C : 1 bộ
- Cổng interput : 6 cổng
- Bộ nhớ trong : 256KB, 8KB sử dụng cho Bootloader
- SRAM : 8 KB
- EEPROM : 4 KB
- Giao động : 16 Mhz

2.4. Arduino ESP8266

ESP8266 là một vi mạch dạng SoC (System-on-a-chip) do hãng ESPRESSIF của Trung Quốc sản xuất và đã trở nên phổ biến trên toàn thế giới khi tích hợp được module WiFi vào vi mạch với giá rẻ. Với ESP8266, việc đưa kết nối WiFi vào các hệ thống nhúng trở nên vô cùng dễ dàng.



Hình 2.3 Hình ảnh thực tế của chip NODEMCU ESP8266

2.4.1. Cấu tạo

Module ESP8266 có các chân dùng để cấp nguồn và thực hiện kết nối.

Chức năng của các chân như sau:

+VCC: 3.3V lên đến 300Ma

+GND: Chân Nối đất .

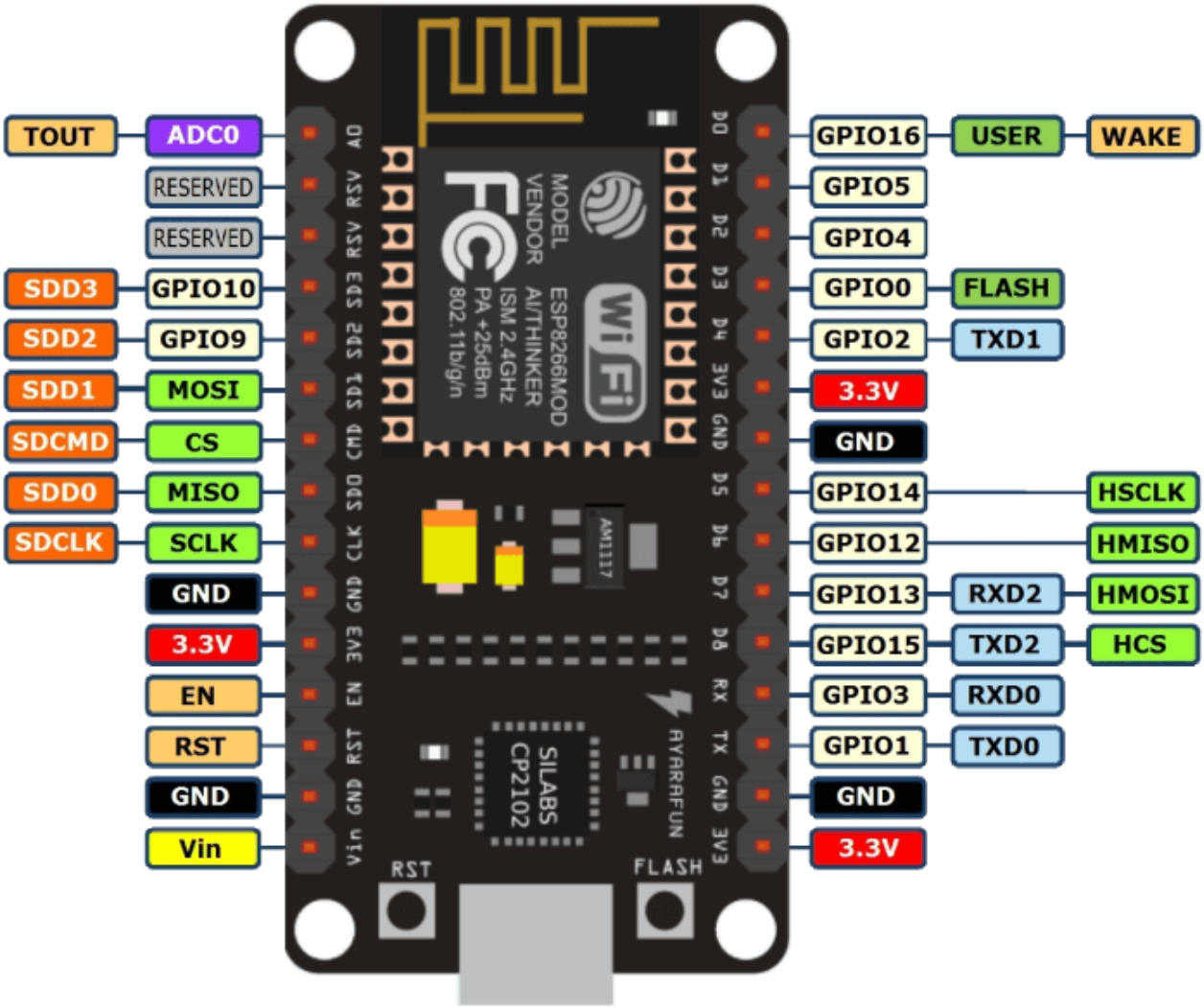
+Tx: Chân Tx của giao thức UART, kết nối đến chân Rx của vi điều khiển.

+Rx: Chân Rx của giao thức UART, kết nối đến chân Tx của vi điều khiển.

+RST: chân reset, kéo xuống mass để reset.

+10 chân GPIO từ D0 – D8, có chức năng PWM, IIC, giao tiếp SPI, 1-Wire và ADC trên chân A0

+Kết nối mạng wifi (có thể là sử dụng như điểm truy cập và/hoặc trạm máy chủ lưu trữ một, máy chủ web), kết nối internet để lấy hoặc tải lên dữ liệu.



Hình 2.4. Hình ảnh sơ đồ chân kết nối ESP8266

Module ESP-12 kết hợp với firmware ESP8266 trên Arduino và thiết kế phần cứng giao tiếp tiêu chuẩn đã tạo nên NodeMCU, loại Kit phát triển ESP8266 phổ biến nhất trong thời điểm hiện tại. Với cách sử dụng, kết nối dễ dàng, có thể lập trình, nạp chương trình trực tiếp trên phần mềm Arduino, đồng thời tương tích với các bộ thư viện Arduino sẵn có.

2.4.2. Tính năng

➢ Thông số kĩ thuật:

- IC chính: ESP8266 Wifi SoC
- Phiên bản firmware: NodeMCU Lua
- Chip nạp và giao tiếp UART: CP2102
- GPIO tương thích hoàn toàn với firmware Node MCU
- Cấp nguồn: 5VDC MicroUSB hoặc Vin
- GIPO giao tiếp mức 3.3VDC
- Tích hợp Led báo trạng thái, nút Reset, Flash
- Tương thích hoàn toàn với trình biên dịch Arduino
- Kích thước: 25 x 50 mm.

➢ **Ứng dụng của ESP8266:**

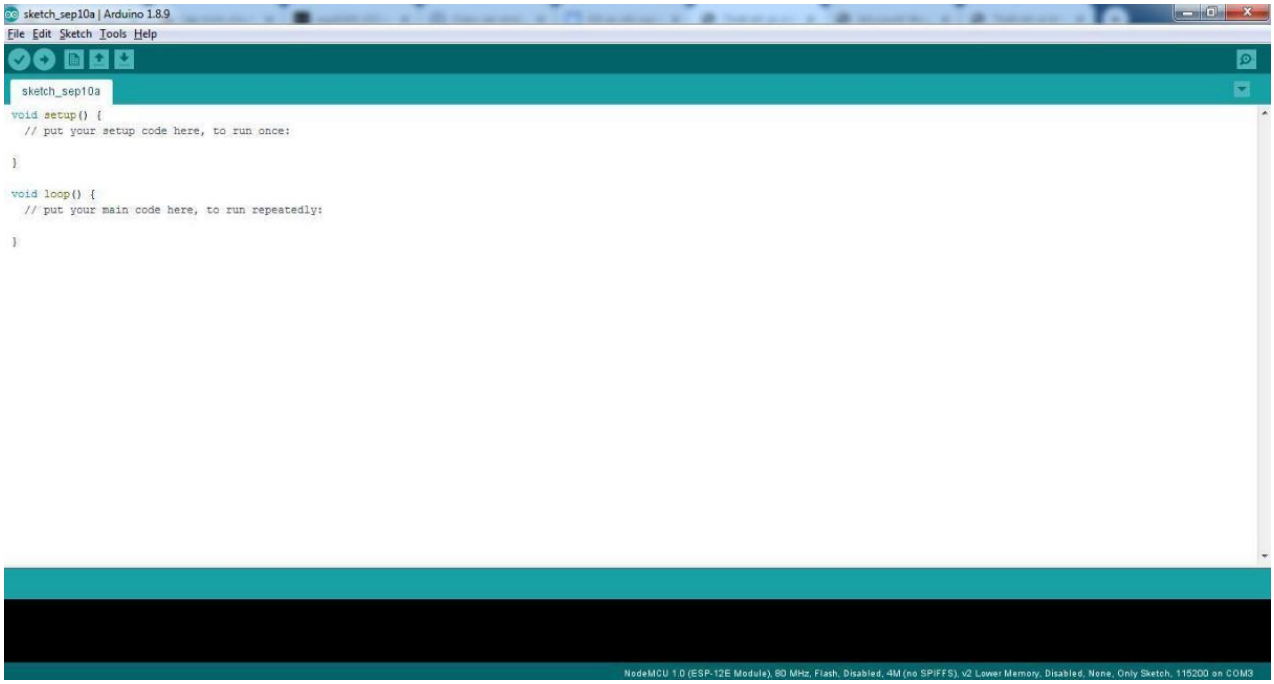
Với các tính năng kết nối wifi vượt trội ESP8266 được sử dụng rất nhiều trong cuộc sống. Thông qua kết nối wifi chúng ta có thể điều khiển các thiết bị từ xa như bật tắt bóng đèn, bật tắt quạt… Đặc biệt hiện nay được sử dụng rất nhiều trong các mô hình nhà thông minh. Qua đó chúng ta có thể dễ dàng quản lý và điều khiển tất cả các thiết bị trong nhà.

CHƯƠNG 3 : ỨNG DỤNG CHO ĐIỀU KHIỂN VÀ GIÁM SÁT HỆ THỐNG ĐIỆN

3.1. Giới thiệu phần mềm lập trình

Arduino IDE [15] là môi trường phát triển tích hợp mã nguồn mở, cho phép người dùng dễ dàng viết code và tải nó lên bo mạch. Môi trường phát triển được viếtbằng Java dựa trên ngôn ngữ lập trình xử lý và phần mềm mã nguồn mở khác. Phần mềm này có thể được sử dụng với bất kỳ bo mạch Arduino nào.

Arduino IDE là một môi trường phát triển tích hợp đa nền tảng, làm việc cùng với một bộ điều khiển Arduino để viết, biên dịch và tải code lên bo mạch. Phần mềm này cung cấp sự hỗ trợ cho một loạt các bo mạch Arduino như Arduino Uno, Nano, Mega, Pro hay Pro Mini, ... Ngôn ngữ tổng quát cho Arduino C và C++, do đó phần mềm phù hợp cho những lập trình viên đã quen thuộc với cả 2 ngôn ngữ này. Các tính năng như làm nổi bật cú pháp, thụt đầu dòng tự động, ... làm cho nó trở thành một sự thay thế hiện đại cho các IDE khác. Arduino IDE có thư viện code mẫu quá phong phú, viết chương trình trên Arduino IDE khá dễ dàng cộng thêm OpenSource viết riêng cho Arduino thì ngày càng nhiều.



Hình 3.1 Giao diện của phần mềm Arduino

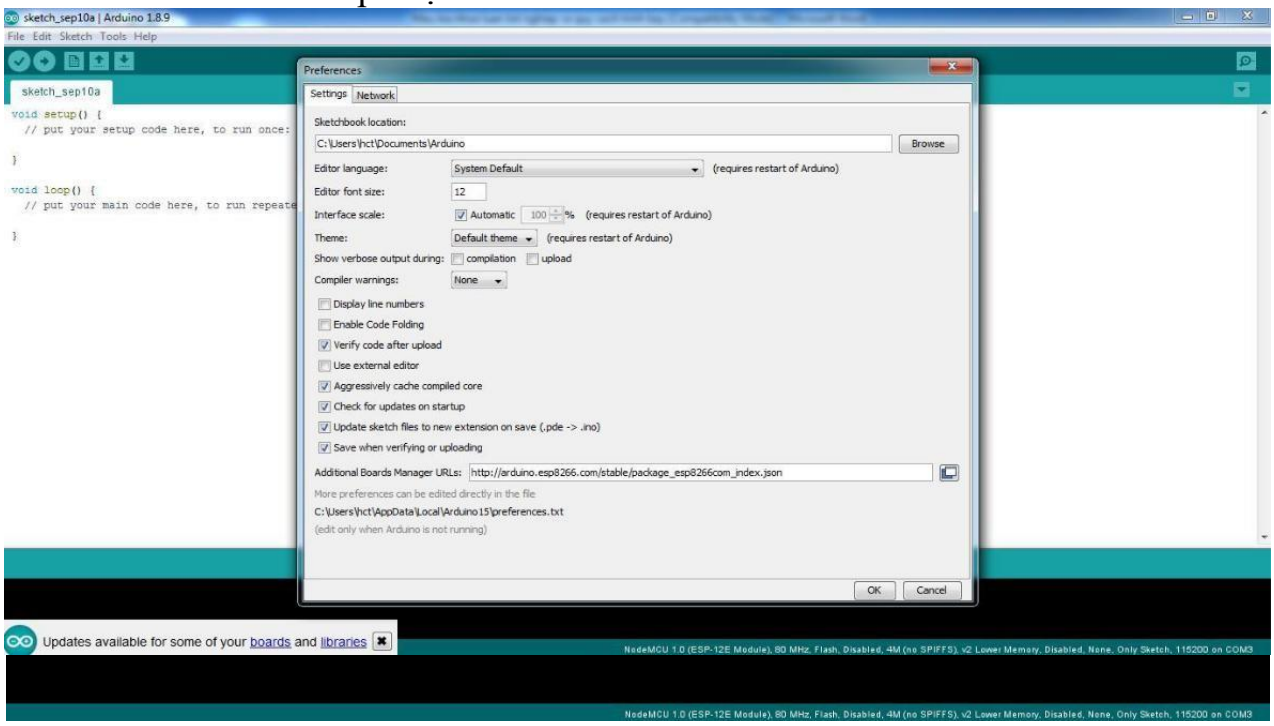
Đây là công cụ hỗ trợ viết code và nạp code cho các bo mạch Arduino cũng như các mạch NodeMCU. Truy cập vào trang chủ Arduino http://arduino.cc để

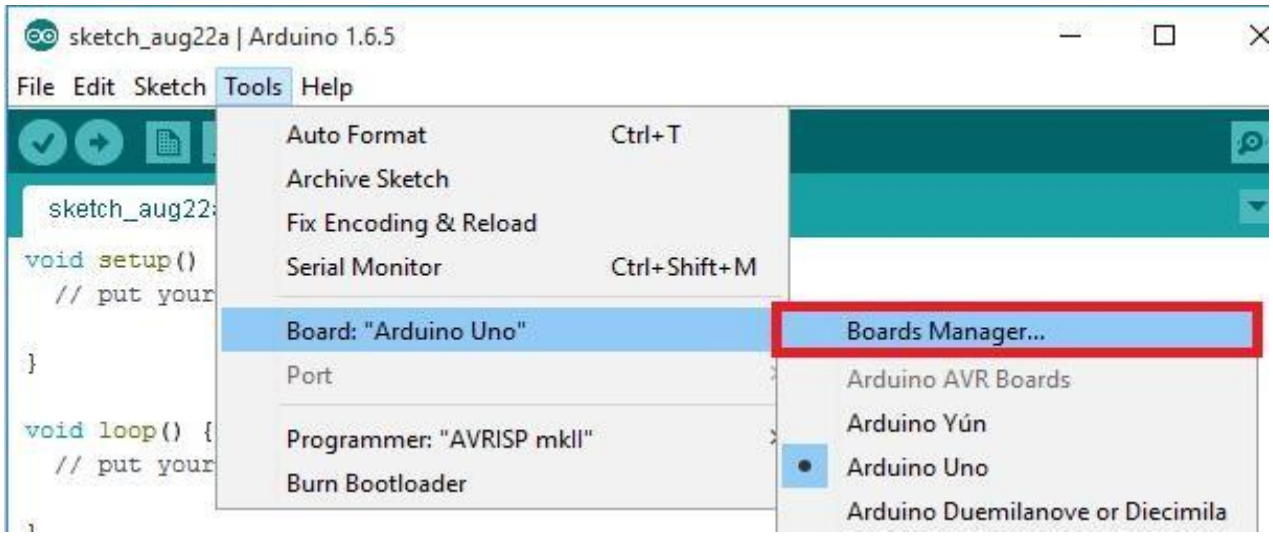
download phần mềm và cài đặt. Phần mềm được hỗ trợ miễn phí cho người dùng , với bản cập nhật mới nhất là Arduino 1.8.5.

Sau khi tải về thì cần thêm thư viện để có thể sử dụng với module NodeMCU ESP8266. Vào File → Preferences, vào textbox Additional Board Manage URL thêm đường link sau vào:

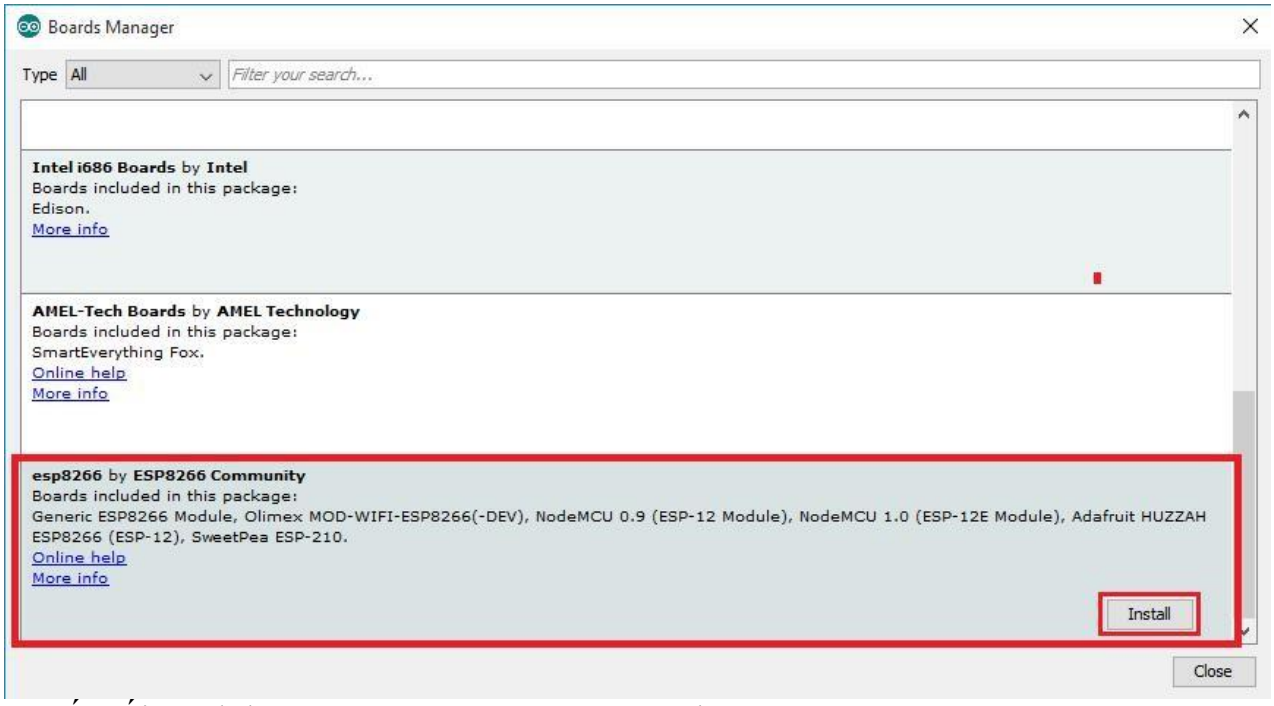
http://arduino.esp8266.com/stable/package_esp8266com_index.json

Sau đó click OK để chấp nhận.





Đợi một lát để chương trình tìm kiếm. Ta kéo xuống và click vào ESP8266 by ESP8266 Community, click vào Install. Chờ phần mềm tự động download và cài đặt.



Kết nối mudule USB-to-UART vào máy tính.

Vào Tool→Board→Generic ESP8266 Module, chọn cổng COM tương ứng với module USB-to-UART tương ứng. Tương tự ta nhập thư viện cho cảm biến đo nhiệt độ DHT11.

3.2. Relay

Các GPIO của board ESP8266 nhận tín hiệu sẽ điều khiển kích hoạt các Relay hoạt động, từ đó điều khiển các thiết bị điện.

Relay là một loại thiết bị điện tự động mà tín hiệu đầu ra thay đổi nhảy cấp khi tín hiệu đầu vào đạt những giá trị xác định. Relay là thiết bị điện dùng để đóng cắt mạch điện điều khiển, bảo vệ và điều khiển sự làm việc của mạch điện động lực.

Chọn Relay 5V, chỉ cần cung cấp nguồn 5VDC và dòng khoảng 80mA cho Relay là các tiếp điểm có thể đóng ngắt khi được kích. Bên cạnh đó, dòng điện tối đa mà Relay có thể chịu được là 10A, nên đảm bảo dòng của các thiết bị điện khi chạy qua các tiếp điểm của Relay sẽ an toàn.

Hjihsjkcljikldccdklcd

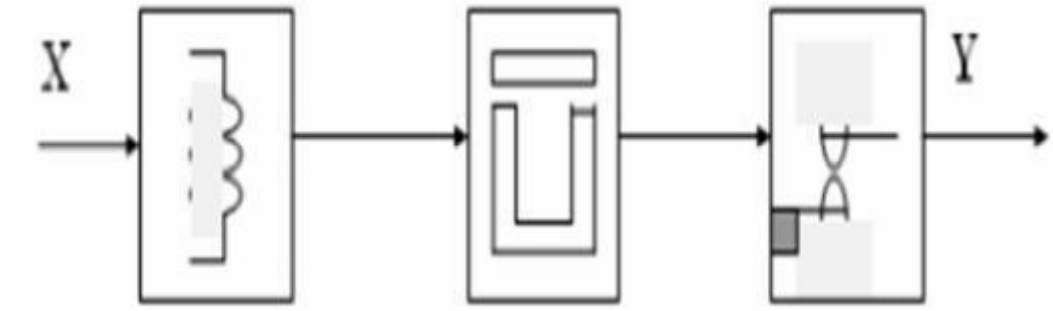
Hình 3.2. Relay 5V-10A

Các bộ phận chính của relay:

-Cơ cấu tiếp thu (khối tiếp thu): Có nhiệm vụ tiếp nhận những tín hiệu đầu vào và biến đổi nó thành đại lượng cần thiết cung cấp tín hiệu phù hợp cho khối trung gian.

-Cơ cấu trung gian (khối trung gian): Làm nhiệm vụ tiếp nhận những tín hiệu đưa đến từ khối tiếp thu và biến đổi nó thành đại lượng cần thiết cho rơle tác động.

-Cơ cấu chấp hành (khối chấp hành): Làm nhiệm vụ phát tín hiệu cho mạch điều khiển

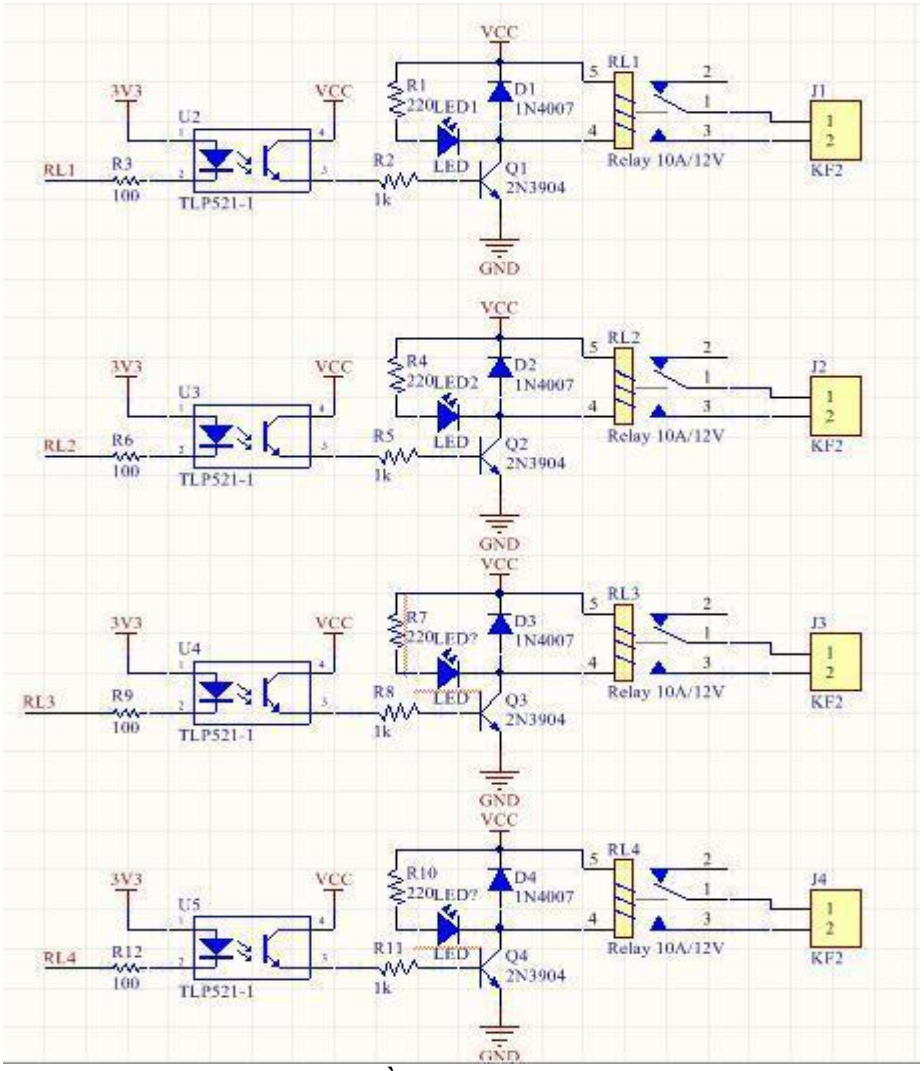


Hình 3.3. Sơ đồ khối của relay điện từ

Relay có 3 tiếp điểm đóng ngắt NO (thường mở), NC (thường đóng) và chân COM, ở trạng thái bình thường khi chưa được kích chân COM sẽ nối với NC, khi kích chân COM chuyển sang nối với NO, NC mất kết nối. Đồng thời có 2 chân nguồn DC để cấp nguồn cho Relay hoạt động.

Bảng 3.1 Thông số kỹ thuật của relay 5V-10A

Thiết kế



Hình 3.4. Sơ đồ nguyên lý của relay 5V

Model	SRD-05VDC
Điện áp kích hoạt	5VDC
Dòng tiêu thụ	80Ma
Dòng AC tối đa	10A-250/125VAC
Dòng DC tối đa	10A-30/28VDC
Số chân	5

Hình 3.5 Hình ảnh thực tế relay 4 kênh

Relay 4 Kênh 5V gồm 4 rơ le hoạt động tại điện áp 5VDC, chịu được hiệu điện thế lên đến 250VAC 10A. Relay 4 kênh 5V được thiết kế chắc chắn, khả năng cách điện tốt. Trên module đã có sẵn mạch kích relay sử dụng transistor và IC cách ly quang giúp cách ly hoàn toàn mạch điều khiển (vi điều khiển) với rơ le bảo đảm vi điều khiển hoạt động ổn định. Có sữn header rất tiện dụng khi kết nối với vi điều khiển.

Relay 4 kênh sử dụng chân kịch mức Thấp (0V), khi có tín hiệu 0V vào chân IN thì relay sẽ nhảy qua thường Hở của Relay. Ứng dụng với relay module khá nhiều bao gồm cả điện DC hay AC

Module được kết nối với các board điều khiển bằng 6 chân header như sau:

- VCC cung cấp nguồn cho các opto.
- GND kết nối với GND của board điều khiển.
- IN1, IN2, IN3, IN4 dùng để điều khiển relay 1, relay 2, relay 3, relay4 tích cực mức thấp.

Ngoài ra còn một 3 chân header được dùng để cấp nguồn cho relay, header này sẽ có một jumper dùng để kết nối chân VCC với chân RY_VCC mục đích dùng chung nguồn VCC (5V) từ header 4 chân cho relay, thông thường jumper được nối lại với nhau. Nếu như muốn cách ly tín hiệu điều khiển với nguồn cấp cho relay thì có thể bỏ jumper này ra và cấp nguồn riêng 5V cho chân RY_VCC.

❖ **Tính năng:**

- Đóng ngắt được dòng điện cao: AC250V 10A, DC30V 10A
- 4 led báo trạng thái relay
- Điện áp điều khiển: 5V
- Mạch cách ly bằng opto
- Kích thước: 76x56 mm

3.3.Lựa chọn thiết bị điều khiển

➢ Bóng đèn

3.4. Chọn app đưa lên internet

3.4.1. Giới thiệu về Blynk

Blynk là một nền tảng có ứng dụng iOS, Android cho phép điều khiển Arduino, Raspberry Pi, ESP8266. Bạn có thể xây dựng ứng dụng điều khiển bằng cách kéo, thả các Widget. Blynk được thiết kế cho IoT, nó có thể điều khiển phần cứng từ xa, hiển thị dữ liệu cảm biến, lưu trữ dữ liệu và có thể làm một vài điều khác khá thú vị.

3.4.2. Cách cài đặt Blynk

- Bắt đầu với ứng dụng và thư viện Blynk
- Phần thiết bị phải phù hợp đáp ứng với Blynk. Nếu phần cứng không có module kết nối, có thể sử dụng các module hỗ trợ như Ethernet, WiFi, GPRS và nhiều thiết bị hỗ trợ khác. Blynk cũng làm việc với Bluetooth và USB.
- Tải ứng dụng blynk trên android hoặc trên ios.

- Với android truy cập vào : https://play.google.com/store/apps/details?id=cc.blynk

-Với ios truy cập vào :https://itunes.apple.com/us/app/blynk-control-arduino-raspberry/id808760481?ls=1&mt=8

-Lấy mã Auth Token

- Auth Token là loại mã dùng để xác thực kết nối giữa chương trình và phần cứng
- Tạo tài khoản trên ứng dụng Blynk của máy chủ Việt Nam. Phần Custom điền (IP: cloud.blynk.vn và Port: 8443)
- Tạo một dự án mới. Sau đó, chọn bảng kết nối sẽ sử dụng.
- Khi tạo dự án mới thành công, sao chép hoặc gửi mã xác thực Auth Token qua tài khoản gmail.
- Kiểm tra gmail trong hộp thư đến và tìm mã xác thực Auth Token

Cài đặt thư viện Blynk

- Tải thư viện Blynk.zip.
- Giải nén thư viện Blynk.zip.
- Sao chép thư mục Blynk vào thư viện libraries trên Arduino IDE (thư mục libraries trên Arduino IDE.
- Dán mã xác thực Auth Token

- Sửa mã xác thực YourAuthToken, mã xác thực nằm trong gmail khi tạo dự án trong ứng dụng Blynk.

3.4.3 Cách hoạt động của Blynk

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

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

- Gửi dữ liệu từ App đến hardware

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

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

```
BLYNK-WRITE(V1)
{
Int pinData=param.asInt();
}
```

- Gửi mảng dữ liệu

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

```
BLYNK_WRITE(V1)
{
Int x=param[0].asInt();
Int y=param[1].asInt();
Int z=param[N].asInt();
}
```

- Lấy dữ liệu từ Hardware

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

Widget yêu cầu hardware:

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

```
BLYNK_READ(V0)
{
Blynk.virtualWrite(5,millis()/1000);
}
```

Hardware gửi dữ liệu đến Widget:

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

- Đồng bộ đối với Hardware

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

```
BLYNK_CONNECTED() {
Blynk.syncAll();
}
BLYNK_WRITE(V0){
 ….
}
```

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

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

- Đối với App

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

- Điều khiển nhiều thiết bị

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

- Trạng thái online của thiết bị

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

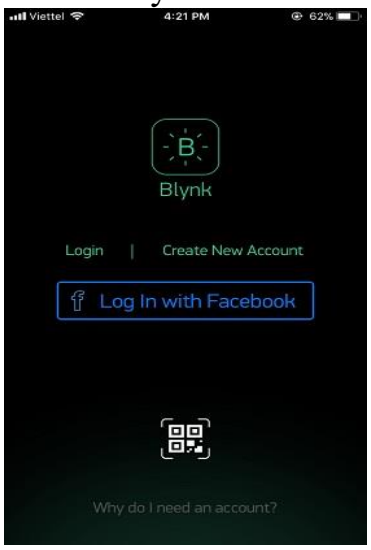
Blynk sử dụng HEARTBEAT. Cách tiếp cận này được thực hiện bằng cách hardware định kỳ gửi lệnh ping (mặc định là 10s). Trong trường hợp, Server không nhận được lệnh ping nào từ hardware trong 10s và thêm 5s, server sẽ cho rằng hardware mất kết nối đến Server.

- Hạn chế và khuyến nghị:

Không thực hiện Blynk.virtualWrite và Blynk.* trong void loop()- điều này có thể gây ra hardware mất kết nối. Sử dụng các hàm có Timer như BlynkTimer. Tránh sử dụng delay() – nó có thể gây ra mất kết nối. Không gửi quá 100 giá trị/s – dẫn đến Flood Error. Khi sử dụng ESP8266 lưu ý không gửi quá nhiều lệnh Blynk.virtualWrite do sự hạn chế của thiết bị này là số lượng request có thể xử lý.

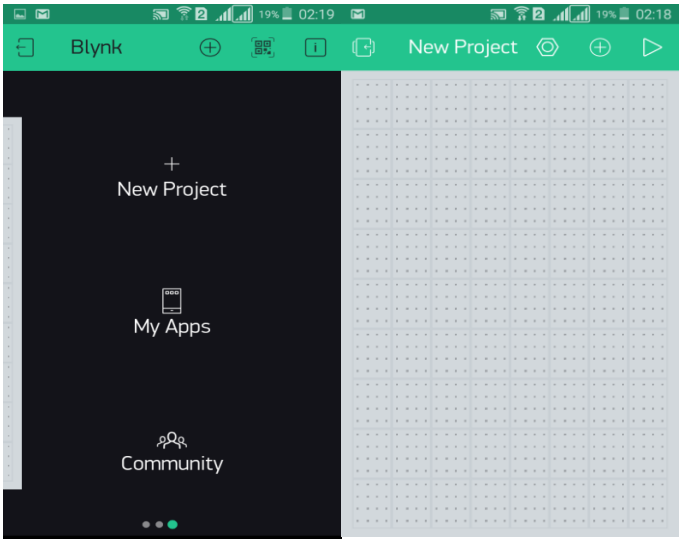
- Bắt đầu với App Blynk

Trên điện thoại sử dụng iOS hoặc Android tải phần mềm Blynk về điện thoại. Sau khi cài đặt đăng ký một Account Blynk.



Giao diện đăng nhập Blynk

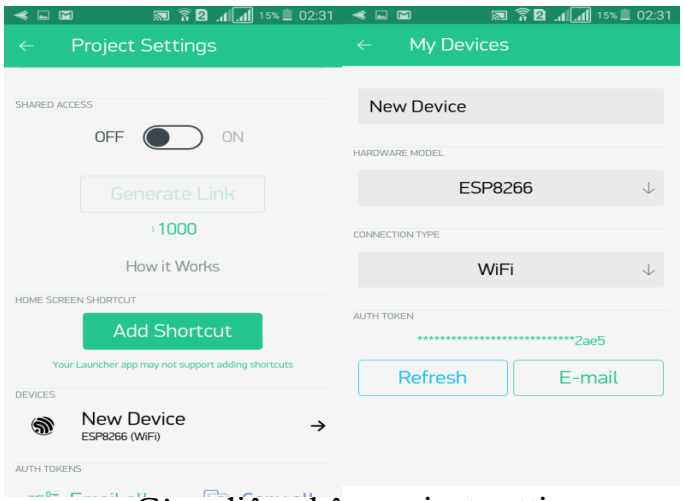
Tại đây ta có thể tạo một tài khoản mới hoặc đăng nhập tài khoản có sẵn. Sau khi đăng nhập màn hình hiện ra.



Tạo New Project

Kéo màn hình sang trái, giao diện tạo Project:

Chọn New Project, thiết lập tên Project, chọn thiết bị. Trong phần này sử dụng ESP8266 vì vậy sẽ chọn ESP8266 và kiểu kết nối là wifi. Sau khi tạo Project, Blynk app sẽ gửi Auth Token đến gmail đăng ký. Auth Token được sử dụng để xác thực các thiết bị. Sau khi tạo xong project, vào phần Project Setting :

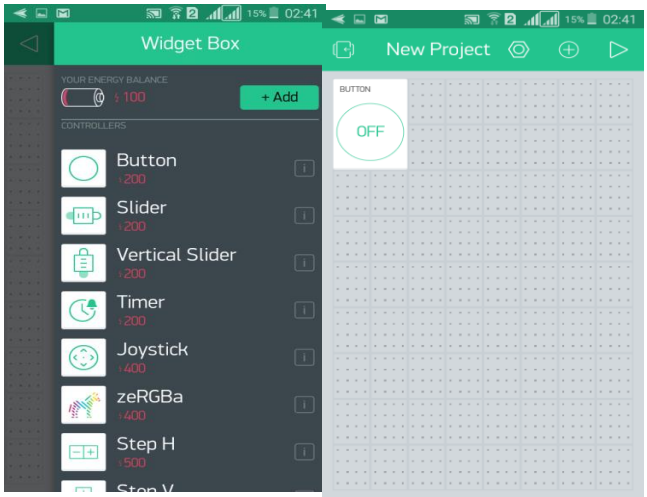


Giao diện phần project settings

Mỗi account khi đăng ký sử dụng dịch vụ đám mây Blynk sẽ được cấp một số Energy. Với mỗi Widget tạo ra trong Project sẽ tốn một số tài nguyên Energy, nếu người dùng muốn sử dụng thêm thì phải bỏ thêm tiền mua Energy. Blynk có mã nguồn mở, tuy nhiên để duy trì hoạt động và tái phát triển cho Blynk, dịch vụ đám mây Blynk cũng có thu phí theo nhu cầu của người sử dụng. Khi kích vào Devices

có thể thêm bớt các Device. Mỗi Device sẽ có một Auth Token dùng để xác thực thiết bị đến Blynk Server.

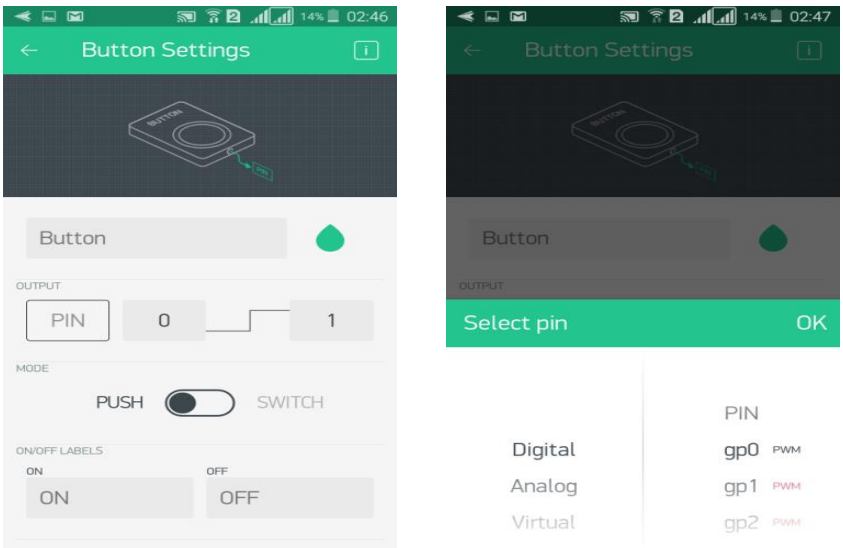
- Thêm Widget



Thêm Widget

Ban đầu bảng vẽ là trống, kích vị trí bất kỳ trên bảng vẽ, hộp các Widget sẽ hiển thị ra. Thêm một Button vào bảng vẽ.

Sau khi thêm Button, có thể kích vào button, giữ và di chuyển button đến vị trí thích hợp. Kích vào Button để thiết lập, trong phần Output thiết lập là Digital và chọn GP4 tương ứng GPIO4 của ESP8266. Sau khi thiết lập xong, thực hiện Run để bắt đầu hoạt động của Project. Khi sửa đổi Project thì cần STOP chương trình lại và có thể thêm các Widget khác.

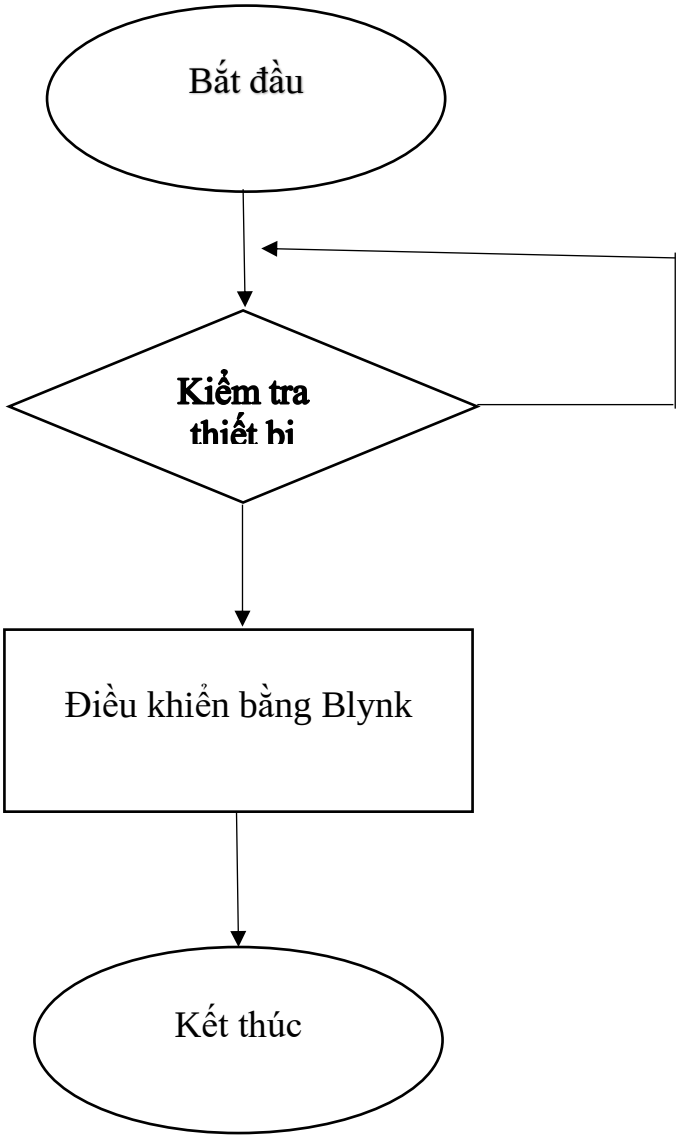


Chọn pin cho Widget

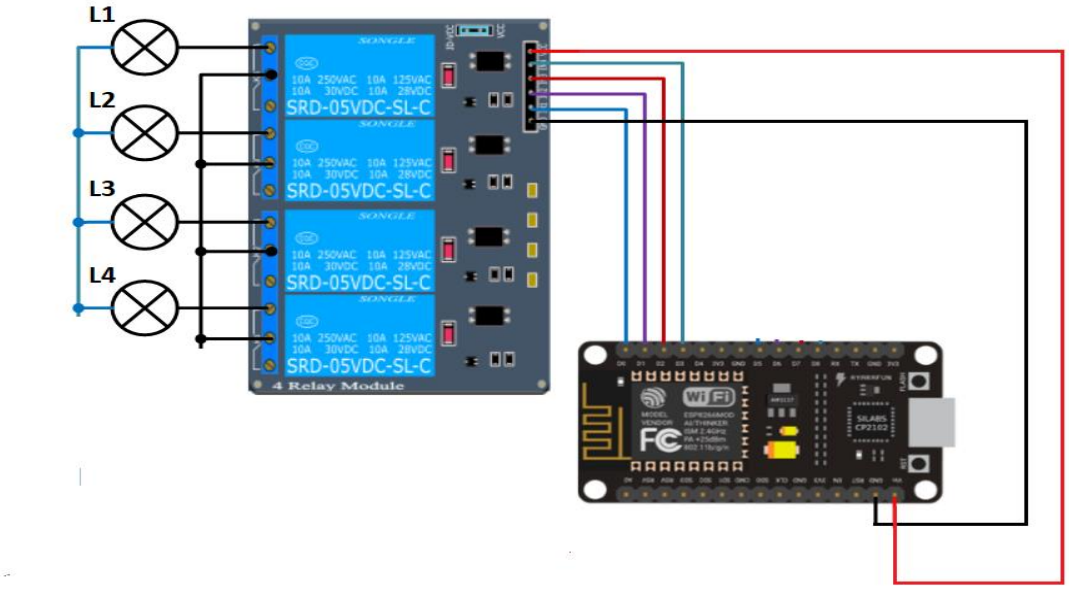
Sau khi chọn xong pin gpio thì tiến hành điều khiển thiết bị mình đã cài đặt.

3.5. Thiết kế xây dựng hệ thống

3.5.1. Lưu đồ giải thuật


Bắt đầu
Kiểm tra thiết bị
Điều khiển bằng Blynk
Kết thúc

3.5.2.Sơ đồ mô phỏng mạch



Hình 3.6 Hình ảnh sơ đồ toàn hệ thống

- Sơ đồ nối chân :

ESP8266	Relay 4 kênh
GND	GND
VU	VCC
D1	IN1
D2	IN2
D3	IN3
D4	IN4

Bảng 3.2 Sơ đồ nối chân hệ thống

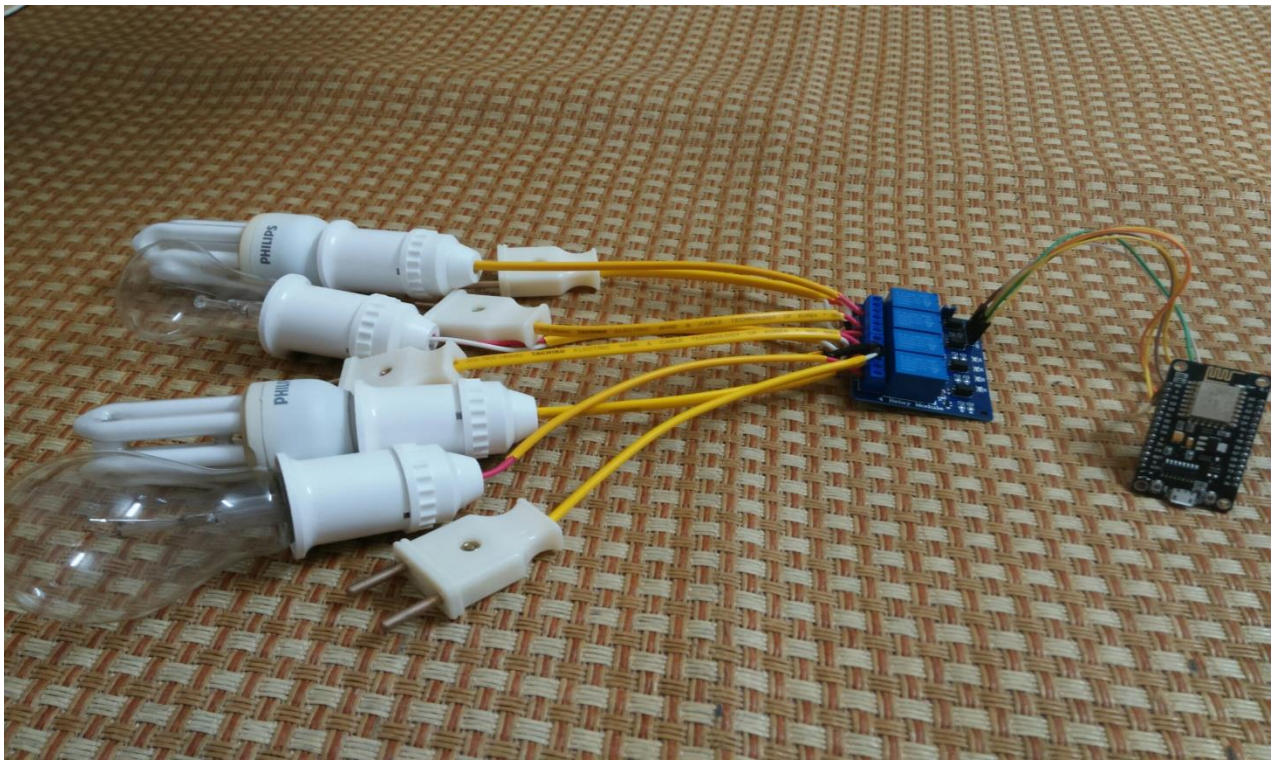
3.5.3. Thực hiện lắp ráp mạch và ghép nối modul

B1: Nối các chân của esp8266 với relay 4 kênh

B2: Nối 2 nửa dây nóng của bóng đèn cho lần lượt cổng COM và cổng NO của relay

B3: Kết nối arduino với esp8266

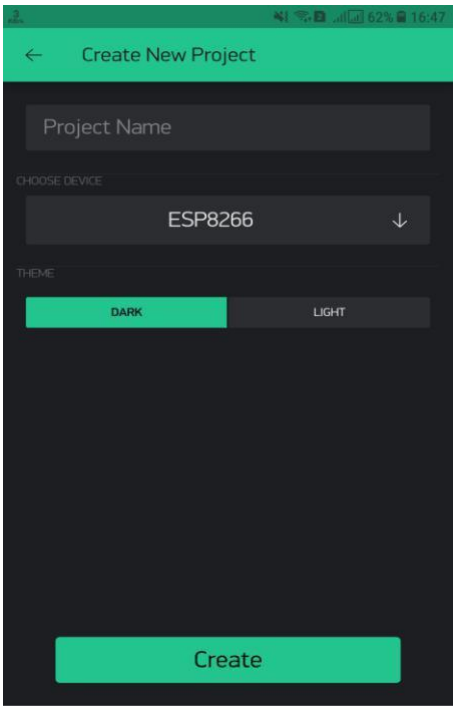
B4: Cấp nguồn cho mạch 5v
B5: Nạp chương trình và kiểm tra lại xem có đạt yêu cầu không



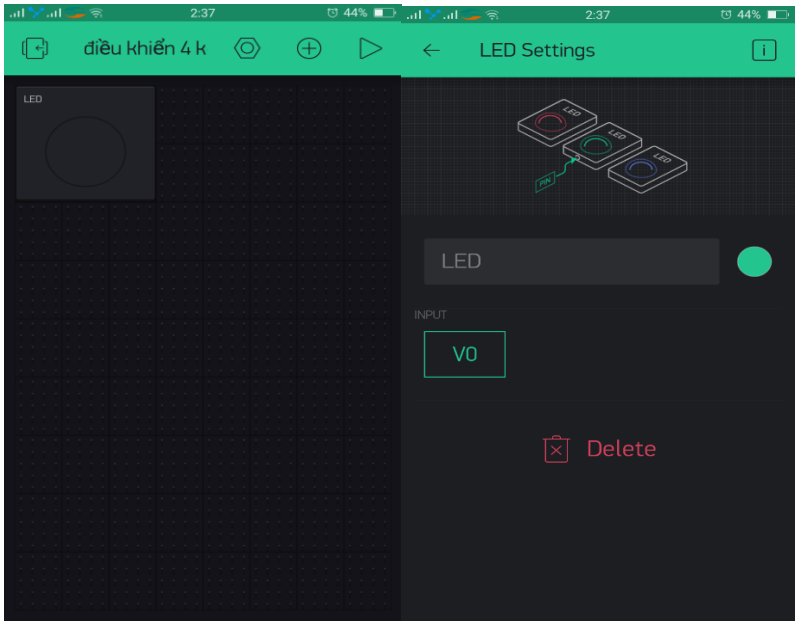
Hình 3.7 Mô hình thực tế của mạch

- Đưa dữ liệu lên blynk

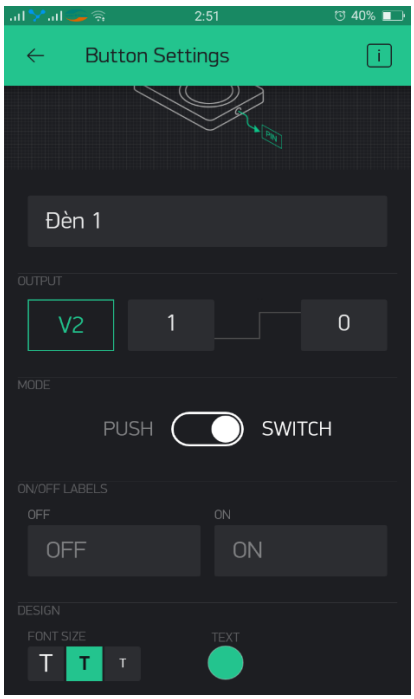
Trước tiên sẽ là tạo giao diện thông qua app Blynk trên smartphone có các chức năng sau



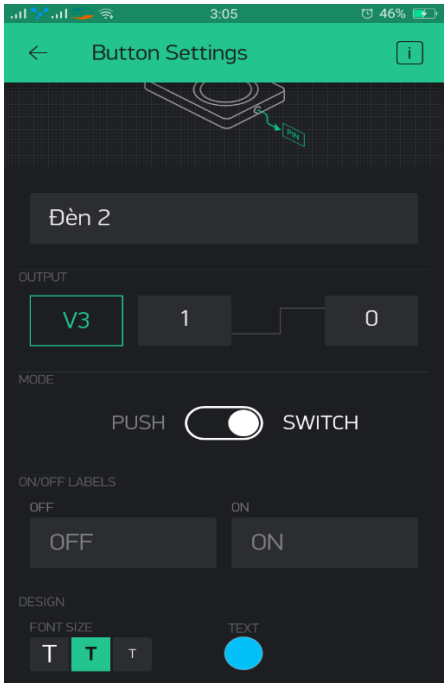
Đọc và hiển thị toàn bộ dữ liệu từ cảm biến
Ta sẽ tạo ra 5 tab để hiển thị thông tin trên :



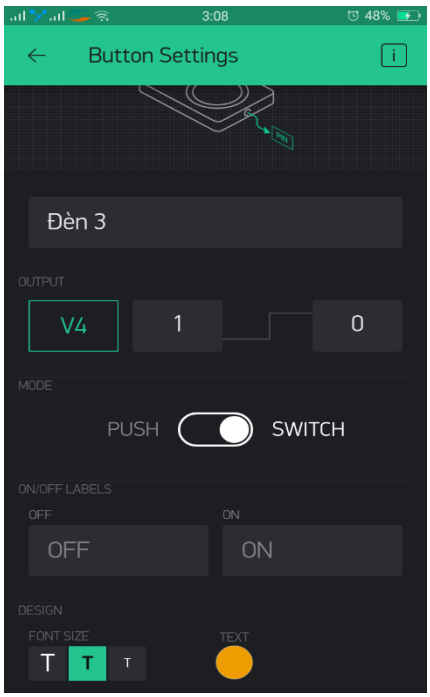
Nút led: Input virtual –V0



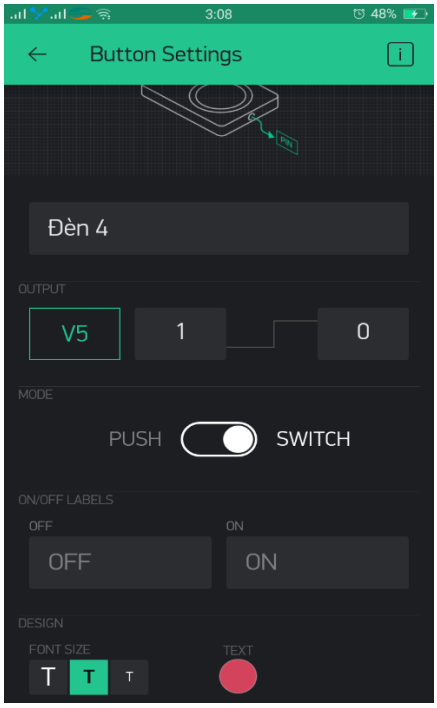
Đèn 1: output là virtual V2 1-0, mode : switch, on/off labels: off-on, text: xanh lá



Đèn 2: output là virtual V3 1-0, mode : switch, on/off labels: off-on, text: xanh da trời



Đèn 3: output là virtual V4 1-0, mode : switch, on/off labels: off-on, text: vàng đất



Đèn 4: output là virtual V5 1-0, mode : switch, on/off labels: off-on, text: đỏ

3.5.4. Kết luận chương

Trong chương này đã trình bày một cách rõ ràng cách để tiến hành xây dựng mô hình giám sát bao gồm các bước, sơ đồ thuật toán, sơ đồ mô phỏng đến thực

hiện chế tạo. Giới thiệu một cách tổng quan về ứng dụng Blynk dùng để điều khiển các thiết bị. Quá trình chế tạo và khảo sát cho thấy bộ điều khiển hoạt động chính xác và ổn định.

3.6. Viết chương trình chạy hệ thống

```
#define BLYNK_PRINT Serial

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>


char auth[] = "CnODloyWjIWQFwT73fzhFNyKuiBB0kQo";


char ssid[] = "ngoc khanh";

char pass[] = "19980121sss";


WidgetLED LedConnect(V0);

BlynkTimer timer;


void blinkLedWidget()

{

  if (LedConnect.getValue()) {

    LedConnect.off();

  } else {

    LedConnect.on();

  }

}
```

```
BLYNK_WRITE(V2)
{
  int pinValue = param.asInt();
  digitalWrite(D1,pinValue);
}
BLYNK_WRITE(V3)
{
 int pinValue = param.asInt();
  digitalWrite(D2,pinValue);
}
BLYNK_WRITE(V4)
{
 int pinValue = param.asInt();
  digitalWrite(D3,pinValue);
}
BLYNK_WRITE(V5)
{
 int pinValue = param.asInt();
  digitalWrite(D4,pinValue);
}
void setup()
{
```

```
  Serial.begin(9600);
  Blynk.begin(auth, ssid, pass);
  timer.setInterval(1000L, blinkLedWidget);

  pinMode(D1,OUTPUT);
  digitalWrite(D1,HIGH);
   pinMode(D2,OUTPUT);
  digitalWrite(D2,HIGH);
  pinMode(D3,OUTPUT);
  digitalWrite(D3,HIGH);
  pinMode(D4,OUTPUT);
  digitalWrite(D4,HIGH);
}
void loop()
{
  Blynk.run();
  timer.run();
}
```

KẾT LUẬN

4.1. Kết luận

Sau thời gian nghiên cứu, thi công thì đồ án tốt nghiệp của em với đề tài "Điều khiển và giám sát thiết bị điện qua mạng internet" đã hoàn thiện, đáp ứng được những yêu cầu ban đầu đặt ra.

4.1.1. Ưu điểm

Mạch điều khiển nhỏ gọn, hoạt động khá ổn định, thời gian đáp ứng khá nhanh.

Giao diện giám sát dễ sử dụng, thân thiện người dùng.

Mô hình hệ thống có độ chính xác, tính an toàn và dễ dàng thao tác với người dùng.

Phù hợp cho các hệ thống điện trong phòng học, hộ gia đình.

Nhìn chung, mô hình đã hoạt động tương đối ổn định, có thể làm việc liên tục, đạt 100% yêu cầu đề ra ban đầu

4.1.2. Khuyết điểm

Tuy nhiên, do sự hạn chế về kiến thức và thời gian thực hiện, nguồn tài liệu tham khảo chủ yếu thông qua internet nên đề tài không tránh khỏi sai sót và còn một số hạn chế:

Hệ thống phụ thuộc tốc độ truy cập mạng Internet. Hoạt động chủ yếu tại môi trường có phủ sóng wifi.mô hình còn mang tính tượng trưng. Kích thước sản phẩm còn thô, thiếu tính thẩm mỹ. Số lượng thiết bị còn hạn chế. Giới hạn về thời gian, kiến thức nên hệ thống chưa được tối ưu. Nhìn chung hệ thống điều khiển về cơ bản hoạt động tốt.

Hệ thống hiện tại đã đáp ứng được việc giám sát các thiết bị. Trong quá trình thực hiện, em thấy rằng đề tài này rất phổ biến, có tính ứng dụng rất cao trong nhiều dự án thực tế. Vì vậy em đưa ra một số đề xuất nhằm cải tiến và nâng cấp hệ thống:

Giám sát nơi điều khiển bằng camera, cảnh báo chống trộm, báo cháy. Điều chỉnh độ sáng đèn, tốc độ quạt, nhiệt độ điều hòa, ...

Thiết lập hệ thống điều khiển thiết bị tự động nhằm tối ưu hóa việc sử dụng và tiết kiệm điện năng.

TÀI LIỆU THAM KHẢO

Sách tham khảo

1. Lập trình điều khiển với Arduino – Phạm Quang Huy & Lê Cảnh Trung.
2. Nguyễn Tất Bảo Thiện, Phạm Quang Huy , Lập trình IoT với Arduino ESP8266, Nhà xuất bản Thanh Niên.
3. Nguyễn Đình Phú – Nguyễn Trường Duy, Giáo trình Kỹ thuật số, Nhà xuất bản ĐH Quốc gia Tp. Hồ Chí Minh.

Các Webside tham khảo

4. https://tailieu.vn/
5. https://www.youtube.com/
6. https://esp8266.vn/
7. https://123docz.net
8. http://kdientu.duytan.edu.vn/media/49635/tt002-dieu-khien-thiet-bi-qua-internet.pdf
9. https://tapit.vn/internet-things-dieu-khien-giam-sat-thiet-bi-gia-dinh-tu-xa