## NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên: Trần Quang Toàn MSSV: 14141324

Nguyễn Minh Vũ MSSV: 14141374 Chuyên ngành: Công nghệ kỹ thuật điện - điện tử truyền thông Mã ngành: 41 Hệ đào tạo: Đại học chính quy Mã hệ: 1

Khóa: 2014 Lớp: 14141DT3B

14141DT1A

##### TÊN ĐỀ TÀI: THIẾT KẾ HỘP ĐIỀU KHIỂN THIẾT BỊ BẰNG SÓNG HỒNG NGOẠI

1. NHIỆM VỤ

Nội dung thực hiện:

* + Nội dung 1: Tìm hiểu về module ESP-12 8266, module 4 led 7 đoạn, micro SD card, led thu và phát hồng ngoại.
  + Nội dung 2: Giải mã hồng ngoại nhận từ điều khiển từ xa hồng ngoại.
  + Nội dung 3: Xuất tín hiệu hồng ngoại điều khiển một thiết bị.
  + Nội dung 4: Thiết kế và lập trình ứng dụng trên Android.
  + Nội dung 5: Thiết kế và thi công mạch điều khiển
  + Nội dung 6: Thiết kế mô hình sản phẩm.
  + Nội dung 7: Đánh giá kết quả thực hiện.

1. NGÀY GIAO NHIỆM VỤ: 02/04/2018
2. NGÀY HOÀN THÀNH NHIỆM VỤ: 02/07/2018
3. HỌ VÀ TÊN CÁN BỘ HƯỚNG DẪN: ThS. Phan Vân Hoàn

CÁN BỘ HƯỚNG DẪN BM. ĐIỆN TỬ CÔNG NGHIỆP – Y SINH

## LỊCH TRÌNH THỰC HIỆN ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên 1: Trần Quang Toàn

Lớp: 14141DT3B MSSV: 14141324

Họ tên sinh viên 2: Nguyễn Minh Vũ

Lớp: 14141DT1A MSSV: 14141374

##### Tên đề tài: Thiết kế hộp điều khiển thiết bị bằng sóng hồng ngoại

|  |  |  |
| --- | --- | --- |
| ***Tuần/ngày*** | ***Nội dung*** | ***Xác nhận GVHD*** |
| Tuần 1 (02-  08/04/2018) | Nhận đồ án , tìm hiểu đề tài |  |
| Tuần 2 (09-  15/04/2018) | Chọn và tìm hiểu đề tài. |  |
| Tuần 3 (16-  22/04/2018) | Tìm hiểu nội dung và hướng làm đề tài của  điều khiển bằng sóng hồng ngoại. |  |
| Tuần 4 (23/4 - 29 /04/2018) | Tìm hiểu cách thu phát một tín hiệu hồng ngoại |  |
| Tuần 5,6  (30/4 - 13/05/2018) | Tiến hành thiết kế phần cứng cho hệ thống điều khiển từ xa. |  |
| Tuần 7,8,9,10  (14/5 - 10/06/2018) | Tiến hành lập trình phần mềm cho hệ thống điều khiển từ xa. |  |
| Tuần 11,12 (11 -  24/06/2018) | Tìm hiểu thiết kế mô hình sản phẩm. |  |
| Tuần 13 (25/6 -  1/07/2018) | Viết báo cáo, kiểm ra các phần cứng |  |
| 02/07/2018 | Hoàn thành nhiệm vụ đồ án |  |

GV HƯỚNG DẪN

(Ký và ghi rõ họ và tên)

Đề tài này là do chúng tôi tự thực hiện dưới sự hướng dẫn của thầy ThS. Phan Vân Hoàn dựa vào một số tài liệu và không sao chép từ tài liệu hay công trình đã có trước đó. Nếu có bất kỳ sự gian lận nào chúng tôi xin chịu trách nhiệm về nội dung đồ án của mình.

Sinh viên thực hiện đồ án

Trần Quang Toàn Nguyễn Minh Vũ



Lời đầu tiên em xin chân thành cảm ơn các thầy cô giáo trong Trường Đại Học Sư Phạm Kỹ Thuật TP.HCM nói chung, các thầy cô trong khoa Điện – Điện Tử nói riêng đã dạy dỗ cho em kiến thức về các môn đại cương cũng như các môn chuyên ngành, giúp em có được cơ sở lý thuyết vững vàng và tạo điều kiện giúp đỡ em trong suốt quá trình học tập.

Em cũng xin gửi lời tri ân và biết ơn sâu sắc đến thầy ThS. Phan Vân Hoàn người đã tận tình hướng dẫn, chỉ bảo em trong suốt quá trình làm khoá luận.

Cuối cùng, em xin chân thành cảm ơn gia đình và bạn bè, đã luôn tạo điều kiện, quan tâm, giúp đỡ, động viên em trong suốt quá trình học tập và hoàn thành khoá luận tốt nghiệp.

Sinh viên thực hiện đồ án

Trần Quang Toàn Nguyễn Minh Vũ

## MỤC LỤC

Trang bìa **Error! Bookmark not defined.**

[Nhiệm vụ đồ án tốt nghiệp ii](#_bookmark0)

[Lịch trình thực hiện đồ án tốt nghiệp iii](#_bookmark1)

[Lời cam đoan iv](#_bookmark2)

[Lời cảm ơn v](#_bookmark3)

[Mục lục vi](#_bookmark4)

[Liệt kê hình vẽ ix](#_bookmark5)

[Liệt kê bảng xi](#_bookmark6)

[Tóm tắt xii](#_bookmark7)

[Chương 1. TỔNG QUAN 1](#_bookmark8)

* 1. [Đặt vấn đề 1](#_bookmark9)
  2. [Mục tiêu đề tài: 2](#_bookmark10)
  3. [Nội dung nghiên cứu: 2](#_bookmark11)
  4. [Giới hạn: 3](#_bookmark12)
  5. [Bố cục đồ án: 3](#_bookmark13)

[Chương 2. CƠ SỞ LÝ THUYẾT 4](#_bookmark14)

* 1. [Tổng quan về tia hồng ngoại: 4](#_bookmark15)
     1. [Định nghĩa: 4](#_bookmark16)
     2. [Nguồn phát: 4](#_bookmark17)
     3. [Tính chất: 4](#_bookmark18)
     4. [Phân loại: 4](#_bookmark19)
     5. [Ứng dụng: 5](#_bookmark20)
     6. [Tín hiệu hồng ngoại trên điều khiển từ xa: 5](#_bookmark21)
  2. [Giới thiệu phần cứng: 6](#_bookmark23)
     1. [Tổng quan về module ESP8266-12E (NodeMCU V1.0) 6](#_bookmark24)
     2. [Led thu hồng ngoại: 10](#_bookmark28)
     3. [Module micro SD Card: 11](#_bookmark30)
     4. [Module 4 led 7 đoạn: 13](#_bookmark32)
  3. [Giới thiệu về chuẩn giao thức spi: 13](#_bookmark34)
     1. [Khái niệm: 13](#_bookmark35)
     2. [Nguyên lý hoạt động: 15](#_bookmark37)
  4. [Giới thiệu về giao thức ntp: 16](#_bookmark39)
     1. [Khái niệm: 16](#_bookmark40)
     2. [Đặc trưng: 16](#_bookmark41)
     3. [Kiến trúc cơ bản: 16](#_bookmark42)
     4. [NTP timestamp: 17](#_bookmark43)
  5. [Giới thiệu vài nét về hệ điều hành android: 18](#_bookmark44)
     1. [Khái niệm: 18](#_bookmark45)
     2. [Đặc trưng: 18](#_bookmark46)
     3. [Kiến trúc cơ bản: 19](#_bookmark47)
     4. [Môi trường phát triển Android: 20](#_bookmark49)
  6. [Giới thiệu vài nét về firebase: 21](#_bookmark50)
     1. [Khái niệm: 21](#_bookmark51)
     2. [Đặc trưng cơ bản: 21](#_bookmark52)

[Chương 3. TÍNH TOÁN VÀ THIẾT KẾ 22](#_bookmark53)

* 1. [Tổng quan về yêu cầu thiết kế hệ thống 22](#_bookmark54)
  2. [Tính toán và thiết kế hệ thống: 22](#_bookmark55)
     1. [Thiết kế sơ đồ hệ thống: 22](#_bookmark56)
     2. [Tính toán và thiết kế 23](#_bookmark58)

[Chương 4. THI CÔNG HỆ THỐNG 28](#_bookmark63)

* 1. [Giới thiệu: 28](#_bookmark64)
  2. [Thi công hệ thống: 28](#_bookmark65)
     1. [Thi công bo mạch: 28](#_bookmark66)
     2. [Lắp ráp, kiểm tra và thi công mô hình: 30](#_bookmark71)
  3. [Lập trình hệ thống: 32](#_bookmark75)
     1. [Lưu đồ giải thuật: 32](#_bookmark76)
     2. [Phần mềm lập trình cho vi điều khiển: 36](#_bookmark80)
     3. [Chương trình trên Android Studio. 40](#_bookmark88)
  4. [Hướng dẫn sử dụng, thao tác với hệ thống: 49](#_bookmark103)

[Chương 5. KẾT QUẢ, NHẬN XÉT, ĐÁNH GIÁ 50](#_bookmark104)

* 1. [Sản phẩm sau khi hoàn thành: 50](#_bookmark105)
     1. [Sản phẩm: 50](#_bookmark106)
     2. [Kiến thức có được trong quá trình thiết kế sản phẩm: 50](#_bookmark108)
  2. [Kết quả chạy hệ thống: 51](#_bookmark109)
     1. [Quá trình chạy ứng dụng trên điện thoại: 51](#_bookmark110)
     2. [Quá trình vận hành trên phần cứng hệ thống: 53](#_bookmark114)

[Chương 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 56](#_bookmark120)

* 1. [Kết luận: 56](#_bookmark121)
  2. [Hướng phát triển: 56](#_bookmark122)

[TÀI LIỆU THAM KHẢO 57](#_bookmark123)

[PHỤ LỤC 58](#_bookmark124)

## LIỆT KÊ HÌNH VẼ

##### Hình Trang

[Hình 2.1 Hình ảnh remote thực tế 5](#_bookmark22)

[Hình 2.2 Sơ đồ chân của Node MCU 7](#_bookmark25)

[Hình 2.3 Hình ảnh thực tế và sơ đồ chân của ESP8266-12E 8](#_bookmark26)

[Hình 2.4 Hình ảnh thực tế và sơ đồ chân của TL 1838 10](#_bookmark29)

Hình 2.5 Sơ đồ góc và khoảng cách nhận được sóng 11

[Hình 2.6 Hình ảnh thực tế module micro SD card 12](#_bookmark31)

Hình 2.7 Các kích thước SD card 12

[Hình 2.8 Hình ảnh thực tế module 4 led 7 đoạn 13](#_bookmark33)

Hình 2.9 Kết nối SPI giữa hai thiết bị 14

[Hình 2.10 Kết nối SPI giữa nhiều thiết bị 14](#_bookmark36)

Hình 2.11 Các chế độ làm việc của SPI 15

[Hình 2.12 Truyền dữ liệu theo chuẩn SPI 16](#_bookmark38)

[Hình 2.13 Cấu trúc tổng quát của Android 20](#_bookmark48)

[Hình 3.1 Sơ đồ khối hệ thống 22](#_bookmark57)

[Hình 3.2 Sơ đồ thiết kế khối thu phát hồng ngoại 24](#_bookmark59)

[Hình 3.3 Sơ đồ thiết kế hiển thị thời gian 25](#_bookmark60)

[Hình 3.4 Sơ đồ thiết kế khối lưu dữ liệu 25](#_bookmark61)

[Hình 3.5 Sơ đồ nguyên lý toàn mạch 27](#_bookmark62)

[Hình 4.1 Mạch in của mạch điện. 28](#_bookmark67)

[Hình 4.2 Hình 3D mặt trên sơ đồ bố trí linh kiện 29](#_bookmark68)

[Hình 4.3 Hình 3D mặt dưới sơ đồ bố trí linh kiện 29](#_bookmark69)

[Hình 4.4 Mặt dưới mạch thi công thực tế 31](#_bookmark72)

[Hình 4.5 Lắp ráp các linh kiện vào mạch. 31](#_bookmark73)

[Hình 4.6 Hình ảnh mô hình 32](#_bookmark74)

[Hình 4.7 Lưu đồ chương trình trên vi điều khiển 33](#_bookmark77)

[Hình 4.8 Lưu đồ chương trình điều khiển trên Android 34](#_bookmark78)

[Hình 4.9 Lưu đồ hẹn giờ trên Android 35](#_bookmark79)

[Hình 4.10 Phần mềm lập trình Arduino IDE 36](#_bookmark81)

[Hình 4.11 Cách download phần mềm 37](#_bookmark82)

[Hình 4.12 Giao điện thêm board ESP trên Arduino IDE. 37](#_bookmark83)

[Hình 4.13 Tải thư viện cho Esp8266. 38](#_bookmark84)

[Hình 4.14 Danh mục các thư viện hiện có trên Arduino IDE. 38](#_bookmark85)

[Hình 4.15 Chọn Board nạp 39](#_bookmark86)

[Hình 4.16 Tải Java 41](#_bookmark89)

[Hình 4.17 Cài đặt Android Studio 41](#_bookmark90)

[Hình 4.18 Tạo project mới trong Android Studio. 42](#_bookmark91)

[Hình 4.19 Tạo project mới trong Android 42](#_bookmark92)

[Hình 4.20 Chọn phiên bản Android muốn lập trình 43](#_bookmark93)

[Hình 4.21 Chọn màn hình 44](#_bookmark94)

[Hình 4.22 Đặt tên cho màn hình 44](#_bookmark95)

[Hình 4.23 Chọn thiết bị đổ chương trình 45](#_bookmark96)

[Hình 4.24 Giao diện của project Android mới 45](#_bookmark97)

[Hình 4.25 Tạo một Project cho Firebase 46](#_bookmark98)

[Hình 4.26 Giao diện các hệ điều hành Firebase liên kết 46](#_bookmark99)

[Hình 4.27 Liên kết Firebase với Package name của app Android 47](#_bookmark100)

[Hình 4.28 Thêm tập tin google-services.json vào app Android 48](#_bookmark101)

[Hình 4.29 Thêm thư viện liên kết Firebase với app Android 48](#_bookmark102)

[Hình 5.1 Sản phẩm thi công 50](#_bookmark107)

[Hình 5.2 Giao diện app 51](#_bookmark111)

[Hình 5.3 Giao diện điều khiển 52](#_bookmark112)

[Hình 5.4 Giao diện hẹn thời gian 52](#_bookmark113)

[Hình 5.5 Giao diện trên Firebase 53](#_bookmark115)

[Hình 5.6 Học tín hiệu từ điều khiển 53](#_bookmark116)

[Hình 5.7 Điều khiển thiết bị 54](#_bookmark117)

[Hình 5.8 Dữ liệu từng nút trong thẻ nhớ 55](#_bookmark118)

[Hình 5.9 Dữ liệu của tín hiệu của một nút trong thẻ nhớ 55](#_bookmark119)

## LIỆT KÊ BẢNG

##### Bảng Trang

[Bảng 2.1 Chức năng các chân ESP8266-12E 9](#_bookmark27)

Bảng 2.2 Thông số TL1838 11

[Bảng 4.1 Liệt kê linh kiện 30](#_bookmark70)

[Bảng 4.2 Chức năng của các biểu tượng trên thanh công cụ 40](#_bookmark87)

## TÓM TẮT

Điều khiển từ xa (remote controller) là thành phần của một thiết bị điện từ, thường là tivi, đầu đĩa, máy hát, máy điều hòa, quạt, … và được sử dụng để điều khiển chúng từ một khoảng cách ngắn không qua dây dẫn. Những năm gần đây điều khiển từ xa đã liên tục được cải tiến, nâng cấp và phát triển.

Điều khiển từ xa thường sử dụng tia hồng ngoại giúp người dùng ra lệnh cho thiết bị chính thông qua một số nút nhấn để thay đổi các thiết lập khác nhau. Trong thực tế, tất cả các chức năng của đa số các thiết bị điện tử hiện nay đều có thể được điều chỉnh thông qua điều khiển từ xa, trong khi các nút trên thiết bị chính chỉ có một số ít các nút chính thiết yếu. Thông thường tín hiệu từ điều khiển từ xa được mã hóa và yêu cầu thiết bị chính phải cùng thuộc một dòng sản phẩm hay thương hiệu cụ thể. Đầu phát của điều khiển từ xa thường là một đèn Led (diode phát quang), khi điều khiển cần có một khoảng không không có vật cản chắn sang giữa nó và thiết bị chính, tín hiệu có thể bị phản xạ qua gương.

Ngày nay công nghệ trở nên hiện đại, xu hướng mọi thứ điều sẽ được kết nối và điều khiển thông qua mạng không dây wifi. Với ý tưởng giải quyết những bất cập của điều khiển từ xa, nhóm chúng em xin đưa ra đề tài: **“Thiết kế hộp điều khiển thiết bị bằng sóng hồng ngoại”**. Hệ thống có khả năng học được các tín hiệu hồng ngoại, nhận lệnh từ điện thoại thông minh phát ra tín hiệu vừa học được điều khiển các thiết bị hồng ngoại thông qua app Android.

Với đề tài này, nhóm hy vọng sẽ làm cơ sở nghiên cứu cho các nhóm sau có thể mở rộng, phát triển nữa. Nếu được điều chỉnh tốt, ý tưởng này kết hợp với ngôi nhà thông minh sẽ trở thành một hệ thống lớn đáp ứng nhu cầu điều khiển, quản lý tất cả các thiết bị trong nhà một cách thông minh, nâng cao đời sống tiện ích cho con người.

# Chương 1. TỔNG QUAN

#### ĐẶT VẤN ĐỀ:

Ngày nay cùng với sự phát triển mạnh mẽ của các ngành khoa học kỹ thuật, công nghệ kỹ thuật điện tử mà trong đó đặc biệt là kỹ thuật điều khiển tự động đóng vai trò quan trọng trong mọi lĩnh vực khoa học kỹ thuật, quản lí, công nghiệp, cung cấp thông tin... Do đó, là một sinh viên chuyên ngành Công Nghệ Kỹ Thuật Điện Tử

- Truyền Thông chúng ta phải biết nắm bắt xu hướng và vận dụng những kiến thức đã được học để phát triển nó một cách có hiệu quả nhằm góp phần vào sự phát triển nền khoa học kỹ thuật thế giới nói chung và trong sự phát triển kỹ thuật điện tử, truyền thông nói riêng. Bên cạnh đó còn là sự thúc đẩy sự phát triển của nền kinh tế nước nhà.

Như chúng ta đã biết, gần như các thiết bị tự động trong nhà máy, trong đời sống của các gia đình ngày nay đều hoạt động độc lập với nhau, mỗi thiết bị có một quy trình sử dụng khác nhau tuỳ thuộc vào sự thiết lập, cài đặt của nhà sản xuất và người sử dụng. Chúng chưa có một sự liên kết nào với nhau về mặt dữ liệu. Để đáp điều này các đồ án trước đây đã thiết kế một ngôi nhà thông minh hay còn gọi là ngôi nhà số là một giải pháp điều khiển tích hợp cho các căn hộ, tích hợp các thiết bị điện tử, nghe nhìn, truyền thông thành một hệ hoàn chỉnh và thống nhất, có thể tự vận hành tất cả các hệ thống một cách tự động theo chương trình đã cài đặt hoặc theo điều khiển từ xa của người dùng. Các hệ thống như chiếu sáng, máy lạnh, an ninh bảo vệ, âm thanh nghe nhìn, chuông hình, cửa tự động hay cả rèm cửa sẽ được phối hợp vận hành thành một hệ thống đồng nhất. Mỗi chức năng của ngôi nhà thông minh đều có khả năng tự vận hành hoặc dưới sự điều khiển của người dùng thông qua điện thoại di động sử dụng mạng 3G, 4G hay Internet và cung cấp nhiều chế độ sử dụng. Người dùng có thể truy cập từ xa vào hệ thống quản lý tại nhà để xem cửa ngõ qua video, tắt hệ thống đèn nếu lỡ quên khi ra khỏi nhà, tắt bớt các hệ thống đèn không cần thiết trong các khu vực trong nhà để tiết kiệm điện năng.

Tuy nhiên, hệ thống ngôi nhà thông minh này không có khả năng điều khiển các chức năng của những thiết bị sử dụng sóng hồng ngoại làm tín hiệu điều khiển (tivi,

máy lạnh, quạt,…), các thiết bị này có những chuẩn hồng ngoại khác nhau do các hãng sản xuất thiết bị quy định (Sony, Samsung, Sharp, LG…) điều này giúp tránh xung đột với các thiết bị điều khiển bằng sóng hồng ngoại khác nhưng cũng gây khó khăn trong việc điều khiển, vì mỗi khi muốn điều khiển thiết bị phải sử dụng điều khiển từ xa của đúng thiết bị đó, điều đó gây bất tiện khi trong nhà có nhiều thiết bị. Chính vì những bất cập trên cùng với xu hướng công nghệ, nhóm chúng em chọn đề tài **“*Thiết kế hộp điều khiển thiết bị bằng sóng hồng ngoại”*** để áp ứng nhu

cầu điều khiển các thiết bị hồng ngoại được thuận tiện hơn.

#### MỤC TIÊU ĐỀ TÀI:

Đồ án được nhóm nghiên cứu, khảo sát và thực hiện với mục đích áp dụng các kiến thức đã được học ở trường giúp cho những nhu cầu điều khiển thiết bị hồng ngoại trở nên tiện lợi hơn. Vì vậy nhóm chúng em thiết kế **“*Thiết kế hộp điều khiển thiết bị bằng sóng hồng ngoại*”** với mong muốn đem những kỹ thuật và công nghệ mới để làm đơn giản hóa việc điều khiển, không cần phải cầm những chiếc điều khiển từ xa của từng thiết bị nữa. Thiết bị tích hợp module wifi ESP 8266-12, micro SD card, module nhận, phát tín hiệu hồng ngoại, và module 4 led 7 đoạn. Thiết bị được điều khiển thông qua một chương trình trên điện thoại thông minh sử dụng hệ điều hành Android đồng thời có thể hẹn giờ để bật tắt thiết bị.

#### NỘI DUNG NGHIÊN CỨU:

* + - Nội dung 1: Tìm hiểu về module ESP-12 8266, module 4 led 7 đoạn, micro SD card, led thu và phát hồng ngoại.
    - Nội dung 2: Giải mã hồng ngoại nhận từ điều khiển từ xa hồng ngoại.
    - Nội dung 3: Xuất tín hiệu hồng ngoại điều khiển một thiết bị.
    - Nội dung 4: Thiết kế và lập trình ứng dụng trên Android.
    - Nội dung 5: Thiết kế và thi công mạch điều khiển
    - Nội dung 6: Thiết kế mô hình sản phẩm.
    - Nội dung 7: Đánh giá kết quả thực hiện.

#### GIỚI HẠN:

Điều khiển các thiết bị trong nhà (Tivi, điều hòa, camera, quạt, …) bằng sóng hồng ngoại qua giao diện App Android.

#### BỐ CỤC ĐỒ ÁN:

##### Chương 1: Tổng Quan.

Chương này trình bày tổng quan, lý do chọn đề tài, mục tiêu, nội dung nghiên cứu, các giới hạn và bố cục đồ án.

##### Chương 2: Cơ Sở Lý Thuyết.

Giới thiệu các linh kiện, thiết bị sử dụng thiết kế hệ thống, các chuẩn truyền, giao thức truyền, nhận dữ liệu.

##### Chương 3: Tính Toán Và Thiết Kế Hệ Thống.

Tính toán thiết kế, đưa ra sơ đồ nguyên lí của hệ thống.

##### Chương 4: Thiết Kế Hệ Thống.

Thiết kế hệ thống, lưu đồ, đưa ra giải thuật và chương trình.

##### Chương 5: Kết Quả, Nhận Xét, Đánh Giá.

Đưa ra kết quả đạt được sau một thời gian nghiên cứu, một số hình ảnh của hệ thống, đưa ra những nhận xét, đánh giá toàn bộ hệ thống.

##### Chương 6: Kết Luận và Hướng Phát Triển.

Trình bày những kết luận về hệ thống những phần làm rồi và chưa làm, đồng thời nếu ra hướng phát triển cho hệ thống.

# Chương 2. CƠ SỞ LÝ THUYẾT

#### TỔNG QUAN VỀ TIA HỒNG NGOẠI:

#### Định nghĩa:

Tia hồng ngoại là các bức xạ điện từ mà mắt ta không nhìn thấy được (còn gọi là các bức xạ ngoài vùng khả kiến) có bước sóng từ 700 nm đến 1 mm (lớn hơn bước sóng của ánh sáng đỏ và nhỏ hơn bước sóng của sóng vô tuyến cực ngắn).

#### Nguồn phát:

Mọi vật có nhiệt độ lớn hơn 0° K đều phát ra tia hồng ngoại.

#### Tính chất:

Tia hồng ngoại có tính chất cơ bản sau:

* + - * Tác dụng nhiệt.
      * Có thể gây ra hiện tượng [quang điện](https://vi.wikipedia.org/wiki/Hi%E1%BB%87u_%E1%BB%A9ng_quang_%C4%91i%E1%BB%87n) trong ở [chất bán dẫn](https://vi.wikipedia.org/wiki/Ch%E1%BA%A5t_b%C3%A1n_d%E1%BA%ABn).
      * Có thể tác dụng lên một số kính ảnh đặc biệt.
      * Có thể biến điệu như sóng điện từ cao tần.
      * Tia hồng ngoại tuân theo các định luật: truyền thẳng, phản xạ, và cũng gây được hiện tượng nhiễu xạ, giao thoa như ánh sáng thông thường.

#### Phân loại:

Tia hồng ngoại được phân chia theo bước sóng thành ba vùng chính:

* + - * Hồng ngoại gần: có kí hiệu là NIR, có bước sóng từ 0,78 µm đến 3 µm.
      * Hồng ngoại trung: có kí hiệu là MIR, có bước sóng từ 3 µm đến 50 µm.
      * Hồng ngoại xa: có kí hiệu là FIR, có bước sóng từ 50 µm đến 1000 µm.

#### Ứng dụng:

* + - * Dùng để sấy, sưởi.
      * Dùng để chụp ảnh hay quay phim ban đêm.
      * Dùng để truyền tín hiệu điều khiển trong các bộ điều khiển từ xa (remote)…

#### Tín hiệu hồng ngoại trên điều khiển từ xa:

Công nghệ chính được sử dụng trong điều khiển từ xa gia dụng là tia hồng ngoại (IR). Những xung [ánh sáng](https://vi.wikipedia.org/wiki/%C3%81nh_s%C3%A1ng) hồng ngoại này vô hình với [mắt](https://vi.wikipedia.org/wiki/M%E1%BA%AFt) người và có thể nhìn thấy bằng [máy ảnh](https://vi.wikipedia.org/wiki/M%C3%A1y_%E1%BA%A3nh) kỹ thuật số hay [máy quay phim](https://vi.wikipedia.org/wiki/M%C3%A1y_quay_phim). Đầu phát của điều khiển từ xa thường là một đèn [LED](https://vi.wikipedia.org/wiki/LED) (diode phát quang).

Vì điều khiển từ xa sử dụng tia hồng ngoại, cần có một khoảng không không có vật chắn sáng giữa nó và thiết bị chính. Tuy nhiên, tín hiệu có thể phản xạ qua [gương](https://vi.wikipedia.org/wiki/G%C6%B0%C6%A1ng) giống như những loại ánh sáng khác.

Điều khiển từ xa là một thiết bị phát sóng hồng ngoại, sử dụng trong các mục đích điều khiển từ xa (tầm 10m). Điều khiển từ xa nhận lệnh từ người điều khiển thông qua các phím bấm, sau đó xuất ra một khung dữ liệu ứng với phím được bấm. Có rất nhiều loại điều khiển được sử dụng như: Sony, LG, … mỗi loại có 1 cách mã hóa phím bấm khác nhau.



Hình 2.1 Hình ảnh remote thực tế

Số lượng bít dữ liệu được truyền đi khác nhau: có loại 7 bit (sony), loại 8 bit, 12 bit, 16 bit, 18 bit, 32 bit, 42 bit (AIWA). Cùng 1 hãng điện tử ví dụ sony thì số bít cũng có thể khác nhau, điều khiển tivi sony có 7 bit, còn dàn âm thanh sony là 16 bit. Mỗi bít sẽ mã hóa được 2 trạng thái 0 và 1, vậy n bit sẽ mã hóa được 2n trạng thái khác nhau , có nghĩa là với giao thức (protocol) 7 bít thì có thể có 27 = 128 lệnh tương ứng 128 nút trên điều khiển, với giao thức (protocol) 32 bít thì có thể có 232 = 4 294 967 296 lệnh tương ứng 4 294 967 296 nút trên điều khiển,với giao thức (protocol)

16 bít thì có thể có 216 = 65 536 lệnh tương ứng 65 536 nút trên điều khiển.

Tất nhiên nếu sử dụng giao thức có số lượng bít nhiều thì khả năng bị trùng phím với điều khiển khác là rất ít nhưng cũng không cần thiết lắm, trong khi điều này lại làm giảm tuổi thọ pin điều khiển.

Tùy vào mỗi nhà sản xuất mà sử dụng các cách mã hóa tín hiệu hồng ngoại khác nhau như:

* Loại điều chế độ rộng xung thấp là loại điều khiển từ xa có bit 0 và bit 1 khác nhau ở độ rộng xung thấp, điển hình là điều khiển sony.
* Loại điều chế độ rộng xung cao là loại điều khiển từ xa có bit 0 và bit 1 khác nhau ở độ rộng xung cao, điển hình là điều khiển Samsung.
* Ngoài ra cũng có giao thức hồng ngoại NEC và nhiều giao thức khác.

#### GIỚI THIỆU PHẦN CỨNG:

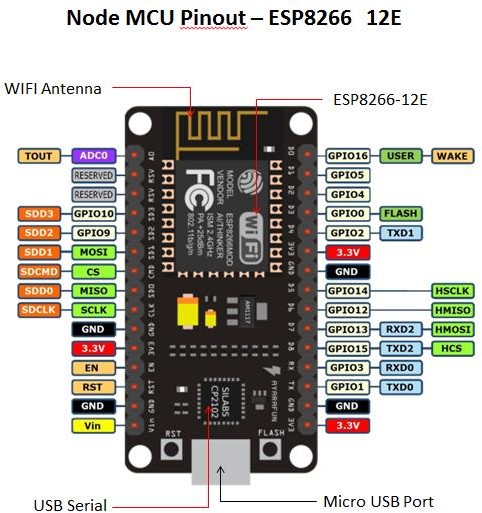
#### Tổng quan về module ESP8266-12E (NodeMCU V1.0):

##### Giới thiệu sơ lược về module ESP8266 NodeMCU V1.0:

* + NodeMCU V1.0 được phát triển dựa trên Chip WiFi ESP8266EX bên trong Module ESP-12E dễ dàng kết nối WiFi với một vài thao tác. 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 sẳn nút nhấn, led để tiện qua quá trình học, nghiên cứu.
  + Với kích thước nhỏ gọn, linh hoạt board dễ dàng liên kết với các thiết bị ngoại vi để tạo thành project, sản phẩm mẫu một cách nhanh chóng.

##### Thông số kỹ thuật:

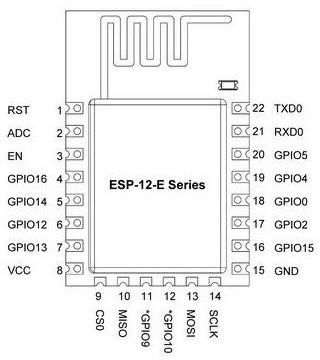
* + Chip: ESP8266EX
  + WiFi: 2.4 GHz hỗ trợ chuẩn 802.11 b/g/n
  + Điện áp hoạt động: 3.3V
  + Điện áp vào: 5V thông qua cổng USB
  + Số chân I/O: 13 (tất cả các chân I/O đều có Interrupt/PWM/I2C/One-wire, trừ chân D0)
  + Số chân Analog Input: 1 (điện áp vào tối đa 3.3V)
  + Bộ nhớ Flash: 4MB
  + Giao tiếp: [Cable Micro USB](https://iotmaker.vn/cable-micro-usb-b-80cm.html)
  + Hỗ trợ bảo mật: WPA/WPA2
  + Tích hợp giao thức TCP/IP
  + Lập trình trên các ngôn ngữ: C/C++, Micropython, NodeMCU - Lua



Hình 2.2 Sơ đồ chân của Node MCU

ESP8266 là một chip tích hợp cao, được thiết kế cho nhu cầu của một thế giới kết nối mới, thế giới Internet of Thing (IoT). Nó cung cấp một giải pháp kết nối mạng wifi đầy đủ và khép kín, cho phép nó có thể lưu trữ các ứng dụng hoặc để giảm tải tất cả các chứng năng kết nối mạng wifi từ một bộ xử lý ứng dụng.

ESP8266 có khả năng xử lý và lưu trữ mạnh mẽ cho phép nó được tích hợp các bộ cảm biến, vi điều khiển và các thiết bị ứng dụng cụ thể thông qua GPIO. ESP8266 có thể kết nối wifi hoặc làm một access point hay cũng có thể trở thành webserver đơn giản.



Hình 2.3 Hình ảnh thực tế và sơ đồ chân của ESP8266-12E Các thông số kỹ thuật:

* SDIO 2.0, SPI, UART.
* Tích hợp RF switch, balun, 24 dBm PA, DCXO và PMU.
* Tích hợp bộ xử lý RISC, giao diện bộ nhớ trong chip và ngoài chip.
* Tích hợp bộ vi xử lý MAC/baseband.
* Giao diện I2S cho các ứng dụng âm thanh chất lượng cao.
* Bộ điều chỉnh tuyến tính sụt áp trên chip cho tất cả nguồn nội.
* Tích hợp WEP, TKIP, AES và các công cụ WAPI.

**-** Wifi 802.11 b/g/n.

* Wifi Direct (P2P), soft AP.
* Công suất đầu ra 19.5 dBm ở chế độ 802.11b.
* Tích hợp CPU 32 bit công suất thấp có thể sử dụng như vi xử lý.
* Đánh thức và truyền gói dữ liệu trong < 2ms.
* Chế độ chờ tiêu thụ điện năng < 1.0 mW(DTIM3).

##### Chức năng các chân của ESP8266:

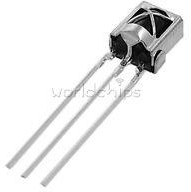
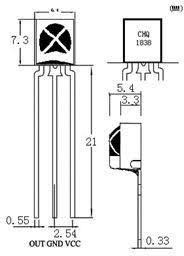
Chức năng các chân trong ESP8266-12E:

Bảng 2.1 Chức năng các chân ESP8266-12E

|  |  |  |
| --- | --- | --- |
| PIN | Chức năng | Mô tả |
| 1 | RST | Chân reset tích cực mức thấp.  Bỏ trống hoặc kết nối bên ngoài MCU. |
| 2 | ADC/OUT | Ngõ vào tương tự ADC 10 bit 0-1V |
| 3 | EN | Cho phép module, tích cực mức cao. |
| 4 | GPIO16 | GPIO(WAKE UP) |
| 5 | GPIO14 | GPIO |
| 6 | GPIO12 | GPIO |
| 7 | GPIO13 | GPIO .UART2 RXD |
| 8 | VDD | Nguồn cấp 3,3V |
| 9 | CS0 | Chọn lựa chip của chuẩn SPI |
| 10 | MISO | MISO chuẩn SPI |
| 11 | GPIO9 | GPIO |
| 12 | GPIO10 | GPIO |
| 13 | MOSI | MOSI chuẩn SPI |
| 14 | SCLK | Xung clock chuẩn SPI |
| 15 | GND | GND |
| 16 | GPIO15 | GPIO  UART2 TXD |
| 17 | GPIO2 | GPIO. Kết nối sẵn bên trong ESP8266 với LED |
| 18 | GPIO0 | GPIO |
| 19 | GPIO4 | GPIO |
| 20 | GPIO5 | GPIO |
| 21 | RXD0 | UART0 RXD |
| 22 | TXD0 | UART0 TXD |

#### Led thu hồng ngoại:

Hiện nay, trên thị trường có rất nhiều chủng loại led thu hồng ngoại khác nhau từ led thu 2 chân cho đến led thu 3 chân. Led thu 3 chân gọi là phototransistor là loại có 3 chân, nó có độ nhạy cao hơn. Đối với led thu 2 chân là photodiode thì cấp nguồn ngược, khi có ánh sáng hồng ngoại nó sẽ dẫn. Còn đối với transitor thì nó có 3 chân riêng biệt: V+, GND, out. Chân out là tín hiệu thu được.



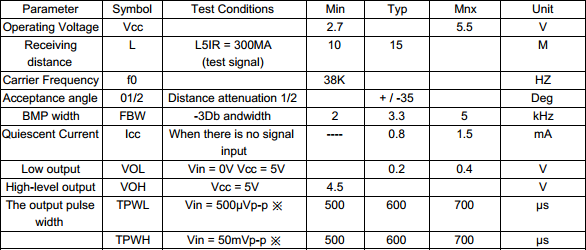
Hình 2.4 Hình ảnh thực tế và sơ đồ chân của TL 1838

Cảm biến hồng ngoại được sử dụng là TL1838. Một số thông số chính của TL1838:

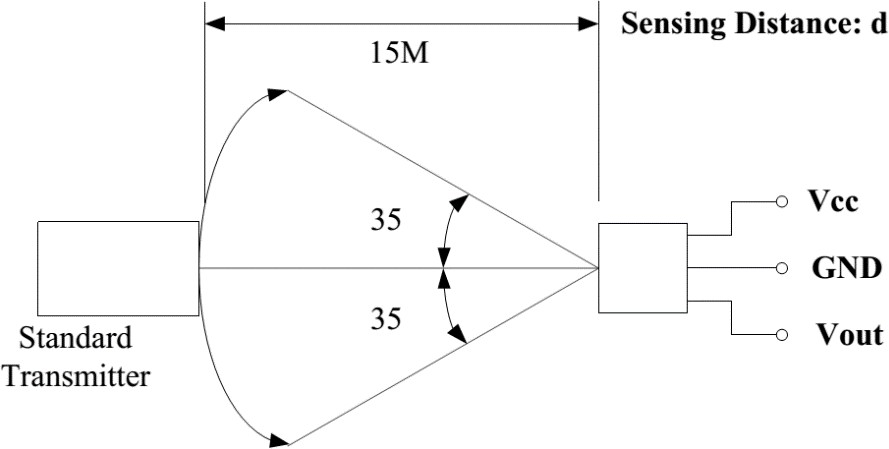
* Làm việc ở mức điện áp: 2,7 ~ 5,5V
* Tần số: 38 KHz
* Dòng tiêu thụ: 1,4 mA
* Khoảng cách nhận biết: khoảng 15m

**-** Góc: 45°

Bảng 2.2 Thông số TL1838



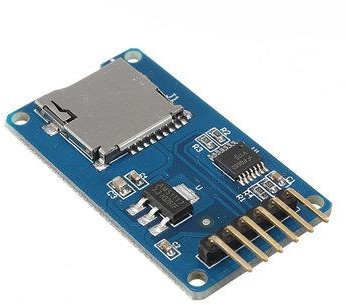
Cấu tạo bên trong của TL1838 được thể hiện ở ***Hình 2.5*** sau:



Hình 2.5 Sơ đồ góc và khoảng cách nhận được sóng

#### Module micro SD Card:

Module Micro SD card là module đọc/ghi thẻ nhớ micro SD dành cho Arduino sử dụng giao tiếp SPI, dễ dàng sử dụng với thư viện SPI.h và SD.h trên Arduino IDE. Có thể đọc/ghi dữ liệu từ micro SD. Cho phép thực hiện các dự án lưu trữ dữ liệu (data logging), phát nhạc MP3…

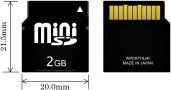
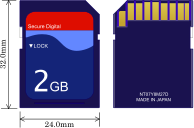


Hình 2.6 Hình ảnh thực tế module micro SD card

##### Thông số kĩ thuật:

* Hỗ trợ thẻ nhớ micro SD, micro SDHC
* Giao thức: SPI
* Các ngõ ra của module: CS, CLK, MOSI, MISO, VCC, GND.
* Thẻ nhớ Micro SD hỗ trợ định dạng FAT16 và FAT32
* Hỗ trợ việc recording và playback cho lượng âm thanh lớn
* Điện áp cung cấp: 5VDC
* Kích thước dài x rộng x cao: 42mm X 24mm X 12mm

Thẻ nhớ là thiết bị lưu trữ dữ liệu, sử dụng công nghệ flash để ghi xóa bộ nhớ. Trong đó thẻ micro SD, SD được sử dụng rộng rãi và được nhiều người biết đến nhất, ứng dụng trong các thiết bị cầm tay, máy ảnh kỹ thuật số, điện thoại thông minh, …



Hình 2.7 Các kích thước SD card

#### Module 4 led 7 đoạn:

Module LED 7 đoạn 4 số sử dụng IC TM1637, chỉ có hai đường tín hiệu nhưng có thể giúp cho MCU điều khiển hiển thị 4 led 7 đoạn anode.



Hình 2.8 Hình ảnh thực tế module 4 led 7 đoạn

##### Thông số kỹ thuật:

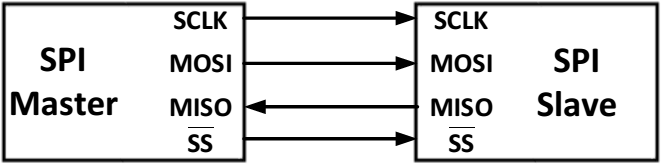
* IC sử dụng: TM1637
* Điện áp sử dụng: 5V hoặc 3.3V
* Gồm 4 led 7 đoạn loại anode chung
* Gồm 4 chân: GND, VCC, DIO, CLK

#### GIỚI THIỆU VỀ CHUẨN GIAO THỨC SPI:

#### Khái niệm:

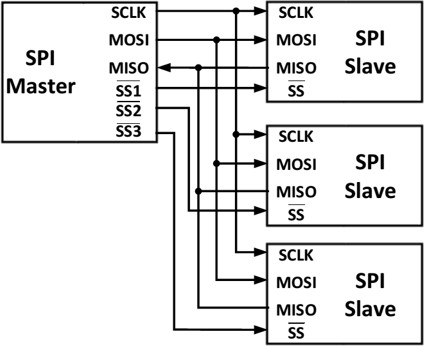
SPI viết tắt của Serial Peripheral Interface, SPI bus – Giao diện ngoại vi nói tiếp, bus SPI. Chuẩn SPI được phát triển bởi Motorola. Đây là một chuẩn đồng bộ nối tiếp để truyền dữ liệu ở chế độ song công toàn phần (full- duplex) tức trong cùng một thời điểm có thể xảy ra đồng thời quá trình truyền và nhận. Đôi khi SPI còn được gọi là chuẩn giao tiếp 4 dây (Four-wire). SPI là giao diện đồng bộ, bất cứ quá trình truyền nào cũng được đồng bộ hóa với tín hiệu clock chung. Tín hiệu này sinh ra bởi master.

Trong giao diện SPI có bốn tín hiệu số:



Hình 2.9 Kết nối SPI giữa hai thiết bị

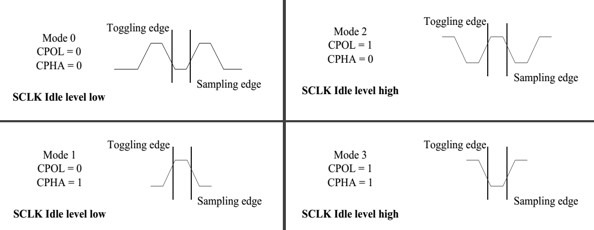
* MOSI hay SI – cổng ra của bên Master (Master Out Slave IN). Đây là chân dành cho việc truyền tín hiệu từ thiết bị chủ động đến thiết bị bị động.
* MISO hay SO – cổng ra bên Slave (Master in Slave Out). Đây là chân dành cho việc truyền dữ liệu từ Slave đến Master.
* SCLK hay SCK là tín hiệu clock đồng bộ (Serial Clock). Xung nhịp chỉ được tạo bởi Master.
* CS hay SS là tín hiệu chọn vi mạch (Chip Select hoặc Slave Select). SS sẽ ở mức cao khi không làm việc. Nếu Master kéo SS xuống thấp thì sẽ diễn ra quá trình giao tiếp. Chỉ có một đường SS trên mỗi Slave nhưng có thể có nhiều đường điều khiển SS trên Master, tùy thuộc vào thiết kế của người dùng.



Hình 2.10 Kết nối SPI giữa nhiều thiết bị

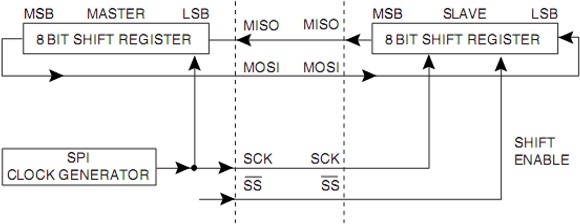
#### Nguyên lý hoạt động:

Để bắt đầu hoạt động thì kéo chân SS xuống thấp và kích hoạt clock ở cả Master và Slave. Mỗi chip Master hay Slave có một thanh ghi dữ liệu 8 bits. Cứ mỗi của xung nhịp do Master tạo ra trên đường giữ nhịp SCK, một bit trong thanh ghi dữ liệu của Master được truyền qua Slave trên đường MOSI, đồng thời một bit trong thanh ghi dữ liệu của chip Slave cũng được truyền qua Master trên đường MISO.



Hình 2.11 Các chế độ làm việc của SPI

Cực của xung giữ nhịp, phase và các chế độ hoạt động: cực của xung giữ nhịp (Clock Polarity) được gọi tắt là CPOL. Đây là khái niệm dùng chỉ trạng thái của chân SCK ở trạng thái nghỉ. Ở trạng thái nghỉ (Idle), chân SCK có thể được giữ ở mức cao (CPOL=1) hoặc thấp (CPOL=0). Phase (CPHA) dùng để chỉ cách mà dữ liệu được lấy mẫu (sample) theo xung giữ nhịp. Dữ liệu có thể được lấy mẫu ở cạnh lên của SCK (CPHA=0) hoặc cạnh xuống (CPHA=1). Sự kết hợp của CPOL và CPHA làm nên 4 chế độ hoạt động của SPI. Nhìn chung việc chọn 1 trong 4 chế độ này không ảnh hưởng đến chất lượng truyền thông mà chỉ cốt sao cho có sự tương thích giữa Master và Slave. Do 2 gói dữ liệu trên 2 chip được gởi qua lại đồng thời nên quá trình truyền dữ liệu này được gọi là “song công”.



Hình 2.12 Truyền dữ liệu theo chuẩn SPI

#### GIỚI THIỆU VỀ GIAO THỨC NTP:

#### Khái niệm:

NTP (Network Time Protocol - Giao thức đồng bộ thời gian mạng) là một giao thức để đồng bộ đồng hồ của các hệ thống máy tính thông qua mạng dữ liệu chuyển mạch gói với độ trễ biến đổi. Giao thức này được thiết kế để tránh ảnh hưởng của độ trễ biến đổi bằng cách sử dụng bộ đệm jitter.

#### Đặc trưng:

Trên mạng Internet, NTP đồng bộ đồng hồ của các hệ thống máy tính theo [UTC](https://vi.wikipedia.org/wiki/Gi%E1%BB%9D_ph%E1%BB%91i_h%E1%BB%A3p_qu%E1%BB%91c_t%E1%BA%BF); trong môi trường LAN độc lập, NTP cũng thường được sử dụng để đồng bộ với UTC, nhưng về nguyên tắc nó có thể được sử dụng để đồng bộ với một mốc thời gian khác, ví dụ như múi giờ tại chỗ.

#### Kiến trúc cơ bản:

NTP sử dụng kiến trúc phân cấp, phân lớp cho các cấp nguồn đồng bộ, mỗi một cấp trong phân cấp này được gọi là một "statum' và được gán một số của cấp bắt đầu từ 0 là cấp cao nhất. Cấp stratum chỉ ra nó đã qua bao nhiêu trung gian để đến được cấp tham chiếu và cấp stratum cũng giúp tránh tham chiếu vòng trong phân cấp. Chú ý rằng cấp stratum không có ý nghĩa chỉ chất lượng hay độ ổn định, dễ dàng tim thấy một nguồn đồng bộ "stratum 3" có chất lượng tốt hơn một nguồn "stratum 2" khác. Định nghĩa về statum này cũng khác với stratum dùng trong đồng bộ viễn thông.

* **Stratum 0**: Bao gồm những thiết bị như đồng hồ nguyên tử (atomic clock), đồng hồ GPS hay các đồng hồ vô tuyến khác. Thiết bị Stratum-0 thường không được kết nối trực tiếp vào mạng mà được kết nối với máy tính (ví dụ thông qua cổng RS-232 sử dụng tín hiệu xung).
* **Stratum 1**: Đây là các máy tính kết nối với thiết bị Stratum 0. Đây là nguồn đồng hồ tham chiếu cho các server Stratum 2. Các máy tính này còn được gọi là [time server](https://vi.wikipedia.org/w/index.php?title=Time_server&action=edit&redlink=1). Các server Stratum 1 (với NTPv3 hay trước đó) có thể không hoạt động với độ chính xác của cấp Stratum 1.
* **Stratum 2**: Là các máy tính gửi các yêu cầu NTP đến cho server Stratum 1. Thông thường máy tính Stratum 2 sẽ tham chiếu từ nhiều server Stratum 1 và sử dụng thuật toán NTP để thu thập thông tin chính xác nhất, và bỏ tham chiếu đến các server Stratum 1 hoạt động không chính xác. Các máy tính Stratum 2 sẽ liên lạc với các máy tính Stratum 2 khác để có được thời gian chính xác và ổn định hơn trong nhóm. Máy tính Stratum 2 theo phân cấp lại là nguồn tham chiếu cho các yêu cầu từ Stratum 3.
* **Stratum 3**: Các máy tính mày cũng thực hiện các chức năng như Stratum 2, và tương tự cũng là nguồn tham chiếu cho các cấp thấp hơn, có thể có tối đa 16 cấp. Tùy vào phiên bản, NTP có thể hỗ trợ đến 256 Stratum.

#### NTP timestamp:

Nhãn thời gian (timestamp) 64 bit của NTP bao gồm 32 bit chỉ giây và 32 bit chỉ phần chi tiết trong 1 giây, NTP timestamp có thể mô tả được một thời gian trong khoảng 232 giây (136 năm) và độ chi tiết đến 2−32 (233 pico giây).

NTP timestamp sẽ lặp lại mỗi 232 giây (136 năm). NTP lấy mốc thời gian vào tháng 1, năm 1900, vì thế nó sẽ lặp lại vào năm 2036, trước sự cố UNIX năm 2038.

Vì NTP hoạt động dựa trên chênh lệch giữa các time stamp và không bao giờ dựa trên giá trị tuyệt đối, việc lặp lại nhãn sẽ không có ảnh hưởng nếu giữa các nhãn sai lệnh không quá 68 năm. Điều này cũng có nghĩa việc lặp lại vào năm 2036 sẽ không ảnh hưởng đến các hệ thống đang hoạt động vì thông thường chênh lệch thời

gian là rất nhỏ. Tuy nhiên, khi hệ thống vừa khởi động, cần xác định ngày chính xác trong 68 năm.

#### GIỚI THIỆU VÀI NÉT VỀ HỆ ĐIỀU HÀNH ANDROID:

Phần này giới thiệu sơ lược về hệ điều hành Android, các thành phần cấu tạo, cấu trúc tổng quát của Android và các thành phần cơ bản của một ứng dụng được xây dựng trên nền Android. Nắm được một số hiểu biết nhất định về hệ điều hành Android và định hình được hướng phát triển một ứng dụng Android trong quá trình thực hiện khóa luận.

#### Khái niệm:

Android là tên một nền tảng mở cho thiết bị di động của Google dựa trên kernel Linux 2.6, gồm hệ điều hành, những phần mềm trung gian (middleware) và một số ứng dụng cơ bản mà người sử dụng cần đến. Bộ công cụ phát triển phần mềm Android SDK (Platform Software Developement Kit) cung cấp các công cụ và các giao diện lập trình ứng dụng API (Application Programming Interface) cần thiết để xây dựng và phát triển các ứng dụng trên nền Android bằng ngôn ngữ lập trình Java.

#### Đặc trưng:

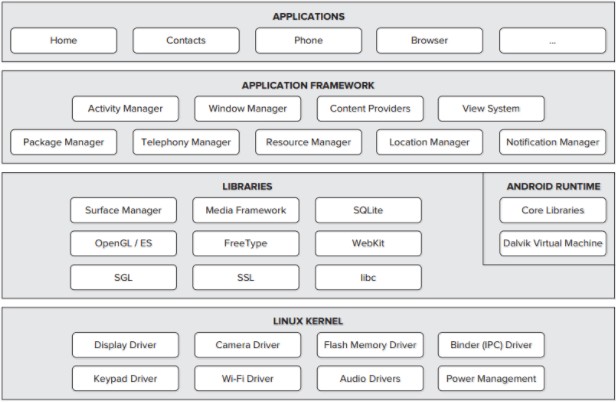
Một số đặc trưng của Android SDK:

* Hỗ trợ Wifi, EDGE, 3G, Bluetooth (phụ thuộc thiết bị phần cứng).
* Công nghệ GSM – GSM Telephony (phụ thuộc nền tảng phần cứng).
* Tích hợp trình duyệt web, cung cấp mã nguồn mở bộ phát triển trình duyệt web HTML5 WebKit.
* Tăng tốc đồ họa, cung cấp các thư viện đồ họa 2D, 3D sử dụng OpenGL ES 2.0.
* Nền tảng ứng dụng giúp sử dụng lại hoặc thay thế các thành phần của ứng dụng tích hợp sẵn trong thiết bị.
* Cấu trúc dữ liệu lưu trữ SQLite.
* Sử dụng các widget, thư mục, hình ảnh để tùy chỉnh giao diện màn hình chủ.
* Kho lưu trữ các dữ liệu dùng để chia sẻ giữa các ứng dụng.
* Hỗ trợ Camera, GPS, compass, accelerometer (phụ thuộc thiết bị phần cứng).
* Hỗ trợ đa phương tiện: hỗ trợ phần lớn các định dạng âm thanh, hình ảnh, video phổ biến như MP3, AAC, JPG, PNG, MPEG4, H.264.

#### Kiến trúc cơ bản:

Có 5 tầng cơ bản trong hệ điều hành Android: Application Framework, Android Runtime, Libraries, Linux Kernel, mỗi tầng làm việc đều nhờ sự giúp đỡ của tầng dưới.

* **Tầng Application:** bao gồm tất cả các ứng dụng có trong thiết bị chạy Android như: phone, contact, game, browser, …và một số ứng dụng chạy ngầm. Người dùng có quyền gỡ bỏ hay cài đặt các ứng dụng tùy thích ở tầng này.
* **Tầng Application Framework:** Là phần thể hiện các khả năng khác nhau của Android (kết nối, thông báo, truy xuất dữ liệu) cho nhà phát triển ứng dụng, chúng có thể được tạo ra để sử dụng trong các ứng dụng của họ.
* **Tầng Libraries:** bao gồm một số các thư viện C/C++ được sử dụng bởi các thành phần khác nhau của hệ thống Android, một số thư viện cơ bản như: System C Library, SQLite, Media Libraries, 3D Libraries, …
* **Tầng Runtime:** Là tầng cùng với lớp thư viện Android runtime cung cấp một tập các thư viện cốt lỗi để cho phép các lập trình viên phát triển viết ứng dụng bằng việc sử dụng ngôn ngữ lập trình Java. Mỗi ứng dụng Android chạy trên một tiến trình riêng của máy ảo Dalvik (vitural machine). Dalvik được viết để chạy nhiều máy ảo cùng lúc một cách hiệu quả trên thiết bị di động. Ngoài ra máy ảo còn giúp tối ưu năng lượng pin cũng như CPU của thiết bị Android
* **Tầng Linux Kernel:** Đây là nhân nền tảng mà hệ điều hành Android dựa vào nó để phát triển. Đâu là lớp chứa tất cả các thiết bị giao tiếp ở mức thấp dùng để điều khiển các phần cứng khác trên thiết bị Android.



Hình 2.13 Cấu trúc tổng quát của Android

#### Môi trường phát triển Android:

Một môi trường hoàn chỉnh để có thể phát triển các ứng dụng trên nền tảng Android gồm có:

* **JDK (Java Development Kit):** Là một môi trường phát triển dùng để viết các ứng dụng Java. JDK gồm một môi trường thực thi các ứng dụng Java (JRE), các công cụ hỗ trợ như: trình biên dịch, trình gỡ lỗi và các thư viện tiện ích giúp người lập trình xây dựng ứng dụng dễ dàng.
* **Android Studio:** là một IDE (Intergrated Development Environment) được google xây dựng và cung cấp miễn phí cho các nhà phát triển Android. Android studio dựa vào [IntelliJ IDEA](http://www.jetbrains.com/idea/), là một IDE tốt cho nhất Java hiện nay. Do đó Android Studio sẽ là môi trường phát triển ứng dụng tốt nhất cho Android.
* **Android SDK:** SDK là chữ viết tắt của Software Development Kit, là bộ công cụ phát triển phần mềm, mà cụ thể là phát triển ứng dụng cho hệ điều hành Android. Sản phẩm này do chính Google xây dựng và phát hành miễn phí cho tất cả những người muốn tìm hiểu và viết ứng dụng trên nền tảng này.
* **ADT Plugin:** ADT là chữ viết tắt của Android Development Tools. Đây là một thành phần mở rộng của Eclipse chứa những thành phần trong Android SDK và được thêm vào Eclipse để tạo một môi trường tích hợp xây dựng ứng dụng Android.

#### GIỚI THIỆU VÀI NÉT VỀ FIREBASE:

#### Khái niệm:

Firebase là một nền tảng ứng dụng di động và web với các công cụ và hạ tầng được thiết kế để giúp các lập trình viên xây dựng các ứng dụng chất lượng cao.

#### Đặc trưng cơ bản:

**Firebase Realtime Database**: Dữ liệu trong cơ sở dữ liệu Firebase của bạn được lưu trữ dưới dạng JSON và đồng bộ realtime đến mọi kết nối client. Khi bạn xây dựng những ứng dụng đa nền tảng như Android, IOS và JavaScrip SDKs, tất cả các client của bạn sẽ chia sẻ trên một cơ sở dữ liệu Firebase và tự động cập nhật với dữ liệu mới nhất.

**Xác thực người dùng**: Với Firebase, bạn có thể dễ dàng xác thực người dùng từ ứng dụng của bạn trên Android, iOS và JavaScript SDKs chỉ với một vài đoạn mã. Firebase đã xây dựng chức năng cho việc xác thực người dùng với Email, Facebook, Twitter, GitHub, Google, và xác thực nạc danh. Các ứng dụng sử dụng chức năng xác thực của FireBase có thể giải quyết được vấn đề khi người dùng đăng nhập, nó sẽ tiết kiện thời gian và rất nhiều các vấn đề phức tạp về phần backend. Hơn nữa bạn có thể tích họp xác thực người dùng với các chức năng backend đã có sẵn sử dụng [custom](https://www.firebase.com/docs/web/guide/login/custom.html) [auth tokens](https://www.firebase.com/docs/web/guide/login/custom.html).

**Firebase Hosting:** Phát triển ứng dụng web của bạn trong thời gian ngắn với các hosting tĩnh đã được cung cấp sẵn. Tất cả các kết nối được phân phối qua SSL từ CDN trên toàn thể giới của Firebase.

# Chương 3. TÍNH TOÁN VÀ THIẾT KẾ

#### TỔNG QUAN VỀ YÊU CẦU THIẾT KẾ HỆ THỐNG:

Mô hình mà nhóm thực hiện cũng chính là mô hình thực nghiệm, vậy nên mô hình thiết kế và thi công phải đáp ứng được các yêu cầu sau:

* + - Điều khiển được các thiết bị bằng sóng hồng ngoại như TV, điều hòa, … trong tầm khoảng 5 - 7 m.
    - Mô hình phải gọn nhẹ, đạt được sự ổn định và tính chính xác cao.
    - Phù hợp điều kiện kinh tế.

#### TÍNH TOÁN VÀ THIẾT KẾ HỆ THỐNG:

#### Thiết kế sơ đồ hệ thống:

**Khối lưu trữ**



**dữ liệu**

**Khối thu phát hồng ngoại**

**Khối xử lý**

**Khối nguồn**

**Khối hiển**

### thị thời gian

**Khối điều**

**khiển**

Hình 3.1 Sơ đồ khối hệ thống Các khối sử dụng trong mạch:

**Khối nguồn**: Sử dụng nguồn 5V để cấp cho toàn bộ hệ thống

**Khối xử lý**: Sử dụng Node MCU để xử lý và truyền nhận các dữ liệu từ khối thu phát hồng ngoại và khối lưu trữ dữ liệu.

**Khối thu phát hồng ngoại**: Có chức năng thu nhận các tín hiệu hồng ngoại bên ngoài hoặc phát ra tín hiệu hồng ngoại để điều khiển các thiết bị khi có yêu cầu từ khối điều khiển.

**Khối hiển thị thời gian**: Có chức năng giao tiếp với khối xử lý để thu dữ liệu từ khối điều khiển cập nhật thời gian thực và hiển thị trên 4 led 7 đoạn.

**Khối lưu trữ dữ liệu**: Sử dụng thẻ nhớ micro SD có khả năng lưu trữ dữ liệu nhận từ khối thu phát hồng ngoại khi có yêu cầu từ khối xử lý cũng như xuất lại dữ liệu cho khối xử lý.

**Khối điều khiển**: Là điện thoại thông minh sử dụng hệ điều hành Android cung cấp giao diện cho người sử dụng có thể điều khiển các thiết bị bằng cách gửi các dữ liệu thông qua mạng Wifi đến khối xử lý.

#### Tính toán và thiết kế:

##### Thiết kế khối thu phát hồng ngoại:

* + **Tính toán giá trị cho trở sử dụng cho phát hồng ngoại:**

Yêu cầu cho mỗi led phát hồng ngoại là 15 mA nên không thể cấp dòng trực tiếp thông qua vi điều khiển mà phải sử dụng transistor 2N2222 để kích dẫn. Công thức tính điện trở cho mỗi led hồng ngoại là:

𝑅 = 𝑉𝑐𝑐− 𝑉𝐿𝐸𝐷 = 5−1.3 = 246.67 Ω

##### (3.1)

𝐿𝐸𝐷

𝐼𝐿𝐸𝐷

0.015

Để transistor hoạt động ở chế độ dẫn bão hòa : 𝐼𝐵

≥ 𝐼𝐶

𝛽

##### (3.2)

Với điện áp ra ở chân vi điều khiển 𝑉𝐼𝑁 = 5V và transistor có 𝑉𝐵𝐸 =0.7, do giá trị β của 2N2222 từ 75 đến 300 nên ta có thể chọn giá trị nhỏ nhất để tính:

𝑅 = 𝑉𝐼𝑁−𝑉𝐵𝐸 = 5−0.7

= 403.13 Ω

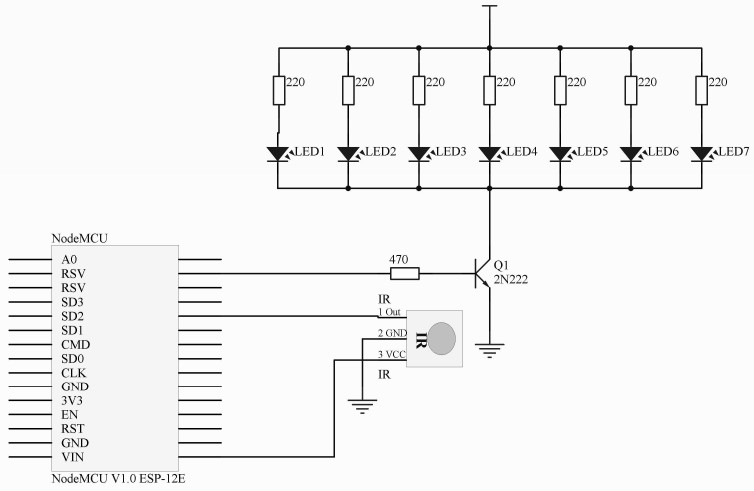
##### (3.3)

𝐵 𝐼𝐶/𝛽

0.8/75

Như vậy, ta sẽ chọn trở hạn dòng cho led là 220 Ω và trở 𝑅𝐵 là 470 Ω.

##### Sơ đồ nguyên lý khối thu phát hồng ngoại:



Hình 3.2 Sơ đồ thiết kế khối thu phát hồng ngoại

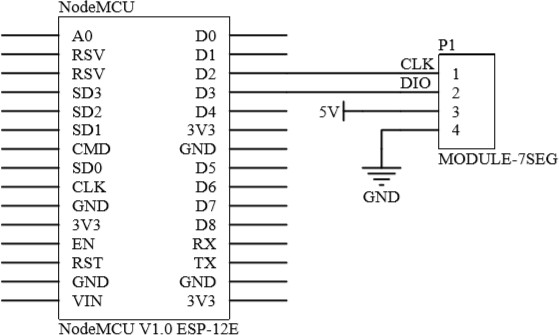
Tín hiệu hồng ngoại nhận được từ remote thực tế thông qua Led thu hồng ngoại TL1838, tín hiệu đó sẽ được xuất ra để điều khiển các thiết bị thông qua 7 con led phát.

##### Thiết kế khối hiển thị thời gian:

Để quan sát được thời gian thực của hệ thống, ta sử dụng led 7 đoạn để hiển thị. Để đơn giản và chi phí thấp ta sử dụng module led 7 đoạn 4 số. Module sử dụng nguồn 5 Vdc.

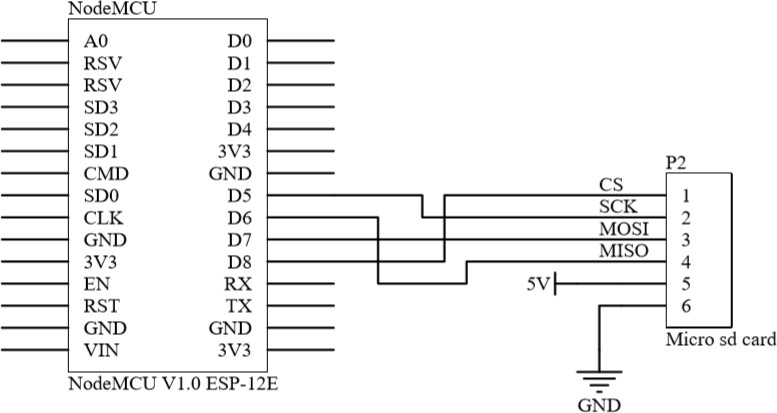
Để giao tiếp với module 4 led 7 đoạn, ta quan tâm chủ yếu đến 2 chân là CLK và DIO sẽ được nối với các chân I/O trên vi điều khiển:

* Chân CLK: được kết nối với chân D2 của vi điều khiển.
* Chân DIO: kết nối với chân D3 của vi điều khiển.



Hình 3.3 Sơ đồ thiết kế khối hiển thị thời gian

##### Thiết kế khối lưu trữ dữ liệu:

Để lưu trữ dữ liệu tín hiệu nhận từ led thu hồng ngoại ta sử dụng SD card. Để đơn giản ta sử dụng module micro SD card tích hợp sẵn. Vì dữ liệu lưu trữ không quá lớn ta sử dụng thẻ SD có dung lượng là 2 GB.

Hình 3.4 Sơ đồ thiết kế khối lưu dữ liệu

Để giao tiếp với micro SD ta sử dụng chuẩn truyền SPI. Ta quan tâm chủ yếu đến 4 chân là CS, MOSI, MISO và SCK sẽ được nối với các chân I/O trên vi điều khiển:

* Chân CS: được kết nối với chân D8 của vi điều khiển.
* Chân MOSI: kết nối với chân D7 của vi điều khiển.
* Chân MISO: kết nối với chân D6 của vi điều khiển.
* Chân SCK: kết nối với chân D5 của vi điều khiển.

##### Thiết kế khối xử lý:

Khối xử lý là khối quan trọng nhất có chức năng truyền, nhận dữ liệu và xử lý tín hiệu thông qua Internet. Vì thế ta chọn Module Node MCU có tích hợp Chip WiFi ESP8266EX bên trong dễ dàng kết nối được wifi chỉ với một vài thao tác.

##### Thiết kế khối nguồn

Nguồn cấp cho khối thu phát hồng ngoại gồm 1 led thu hồng ngoại TL1838 và 7 led phát hồng ngoại. Dòng cấp cho led thu khoảng 1,4mA, mỗi con led phát cần dòng tối đa là 20mA nên tổng dòng cấp cho khối thu phát là: I = 7 x 20 + 1,4≈ 141 mA

Module 4 led 7 đoạn cần dòng cho mỗi led 7 đoạn là 20 mA nên cần dòng là: I= 4x20= 80 mA.

ESP8266 khi hoạt động có thể sử dụng dòng lên đến hơn 550 mA.

Vậy nguồn cấp cho toàn bộ mạch hoạt động cần khoảng I = 141 + 80 + 550

= 771 mA. Nên ta cần chọn nguồn có dòng ra 1A để mạch hoạt động ổn định.

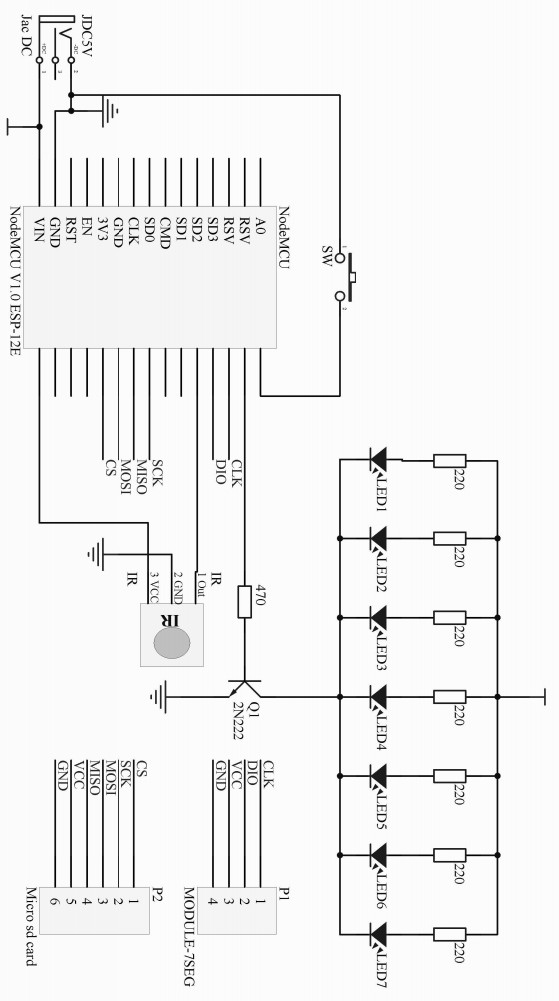
Như vậy nguồn cho hệ thống ta chọn Adapter 5V, 1A để cung cấp nguồn cho hệ thống.

##### Thiết kế khối điều khiển:

Để điều khiển các thiết bị bằng sóng hồng ngoại ta có thể sử dụng nút nhấn cơ, điều khiển qua giao diện Wed, điều khiển qua app Android trên điện thoại, … Ở đây khối điều khiển nhóm sử dụng điện thoại thông minh (smartphone) thông qua một ứng dụng để điều khiển, ứng dụng được viết bằng ngôn ngữ chính là java dựa trên hỗ trợ của phần mềm Android Studio. Giao diện cơ bản của ứng dụng gồm một

màn hình giao diện app, một màng hình điều khiển và các màn hình giao diện hẹn giờ điều khiển.

##### Sơ đồ nguyên lý của toàn mạch:



Hình 3.5 Sơ đồ nguyên lý toàn mạch

# Chương 4. THI CÔNG HỆ THỐNG

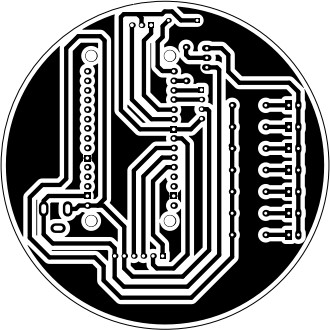
#### GIỚI THIỆU:

Quá trình thi công để được một hệ thống hoàn chỉnh gồm các bước:

* + - Vẽ sơ đồ mạch in.
    - Làm mạch in.
    - Kiểm tra và chỉnh sửa mạch.
    - Lắp ráp các mạch lại với nhau, hoàn thiện sản phẩm.

#### THI CÔNG HỆ THỐNG:

#### Thi công bo mạch:

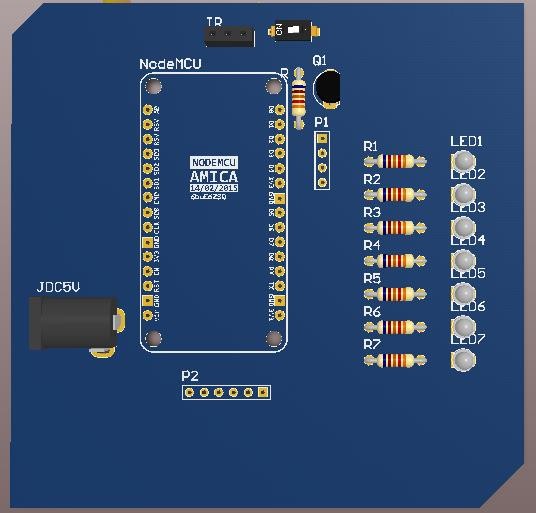


Hình 4.1 Mạch in của mạch điện.

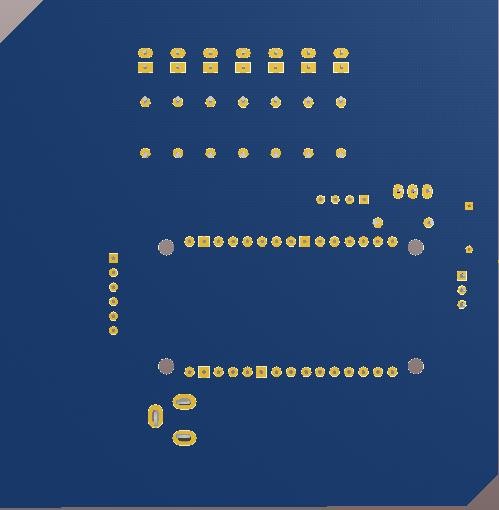
Sau khi in mạch ra board đồng, cần kiểm tra các đường dây có bị hở hay không? Khi khoan lỗ để hàn chân linh kiện cũng phải dùng mũi khoan sao cho hợp lý với

chân linh kiện, điều này giúp linh kiện khi lắp vào mạch được chắc chắn, việc hàn linh kiện cũng dễ dàng hơn.

Sơ đồ bố trí linh kiện để quan sát và gắn linh kiện cho đúng chiều:



Hình 4.2 Hình 3D mặt trên sơ đồ bố trí linh kiện



Hình 4.3 Hình 3D mặt dưới sơ đồ bố trí linh kiện*.*

Bảng liệt kê linh kiện:

Bảng 4.1 Liệt kê linh kiện

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên linh kiện** | **Thông số, giá trị** | **Số lượng** |
| 1 | Node MCU | 𝑉𝐼𝑁 = 5𝑉 | 1 |
| 2 | TL1838 | 𝑉𝐼𝑁 = 2.7 ~ 5.5𝑉 | 1 |
| 3 | Led phát hồng ngoại | 𝑉𝐼𝑁 = 1.3𝑉, 5mm | 7 |
| 4 | 2N2222 | 𝐼𝑐𝑚𝑎𝑥 = 800𝑚𝐴  𝛽 = 75 ~ 300 | 1 |
| 5 | Trở | 220Ω  470 Ω | 7  1 |
| 6 | Module 4 led 7 đoạn TM1637 | 𝑉𝐼𝑁 = 5𝑉 | 1 |
| 7 | Module micro SD Card | 𝑉𝐼𝑁 = 5𝑉, giao tiếp SPI | 1 |
| 8 | Nút nhấn | 2 chân, 3mm | 1 |
| 9 | Adapter | 5V, 1A | 1 |

#### Lắp ráp, kiểm tra và thi công mô hình:

Phần này trình bày quá trình lắp ráp các linh kiện vào trong mạch bao gồm mạch nguồn và mạch PCB đã thiết kế. Do mạch được thiết kế đơn giản nên phần vẽ mạch và kiểm tra chi tiết được chính xác và ổn định.



Hình 4.4 Mặt dưới mạch thi công thực tế Tiến hành lắp ráp và hàn các linh kiện vào mạch.

Hình 4.5 Lắp ráp các linh kiện vào mạch.

Sau đó tiến hành đóng gói, sơn và hoàn thiện mô hình như hình sau:



Hình 4.6 Hình ảnh mô hình

#### LẬP TRÌNH HỆ THỐNG:

#### Lưu đồ giải thuật:

* + - * **Lưu đồ chính của hệ thống:** (Hình 4.7)

##### Giải thích lưu đồ:

Bước 1: Ban đầu ta cấu hình cho hệ thống kết nối với Wifi tại nơi sử dụng. Bước 2: Lấy thời gian thực trên server và đọc tất cả các giá trị từng nút trên

Firebase.

Bước 3: Nếu ta gạt nút cho phép học lệnh thì dữ liệu từng nút sẽ được đọc và lưu vào thẻ SD sau đó quay lại Bước 2, nếu không cho phép học lệnh thì tiếp tục thực hiện Bước 4.

Bước 4: Nếu có nhấn nút từ app điều khiển thì hệ thống đọc dữ liệu đã lưu trong thẻ SD và gửi tín hiệu đó để điều khiển thiết bị và tiếp tục quay lại Bước 2. Còn nếu không nhấn nút từ app thì tiếp tục thực hiện Bước 5.

Bước 5: Kiểm tra thời gian đã đặt trước được lưu trên Firebase với thời gian thực nếu giống nhau thì gửi dữ liệu điều khiển của nút thứ nhất, sau khoảng thời gian 30s thì tiếp tục gửi dữ liệu điều khiển của nút thứ hai. Sau đó quay lại bước 2.

Bắt đầu

Cấu hình cho hệ

thống

Đ

Học lệnh ?

S

Đ

Có nhấn nút

S

S

Hẹn giờ

Đ

Gửi mã nút thứ 1 Gửi mã nút thứ 2

Đọc data từ

SD và gửi tín hiệu đó đi

Đọc và lưu data vào SD

Lấy thời gian từ

Server, đọc tất cả các

giá trị trên Firebase

Hình 4.7 Lưu đồ chương trình trên vi điều khiển

##### Lưu đồ chương trình điều khiển trên Android Studio: (Hình 4.8)

**Giải thích lưu đồ:**

Ban đầu ta khai báo các thư viện cần thiết, cấu hình cho firebase. Tiếp theo kiểm tra có nhấn nút nào không, nếu không thì tiếp tục kiểm tra. Còn nếu có thì kiểm tra nút đó phải là nút TIMER không nếu phải thì chuyển sang màn hình hẹn giờ và kết thúc. Nhưng nếu nút nhấn đó không phải là nút TIMER thì thiết lập biến tương ứng trên Firebase và quay lại kiểm tra có nhấn nút hay không.

Bắt đầu

Khai báo thư viện, Cấu hình Firebase

S

Có nhấn nút?

Đ

S

Có nhấn TIMER ?

Đ

Kết thúc

Hẹn giờ

Set biến tương ứng trên Firebase

Hình 4.8 Lưu đồ chương trình điều khiển trên Android

* + - * **Lưu đồ hẹn giờ trên Android:** (Hình 4.9)

##### Giải thích lưu đồ:

Ban đầu ta khai báo các thư viện cần thiết và cấu hình cho firebase. Sau đó kiểm tra xem có nhấn nút ENTER hay không. Nếu không thì tiếp tục kiểm tra. Còn nếu có thì kiểm tra tiếp đã nhập giờ, nhập phút hay không nếu không thì tiếp tục kiểm tra. Nếu đã nhập giờ, nhập phút rồi thì kiểm tra xem đã nhập nút điều khiển chưa. Nếu chưa nhập nút thì tiếp tục kiểm tra, nếu đã nhấn nút thì kiểm tra giờ có đúng giá trị từ 0-23, phút tử 0-59 hay không. Nếu sai thì quay lại bước kiểm tra có nhấn ENTER chưa và tiếp tục kiểm tra tiếp. Ngoài ra nếu đã kiểm tra đúng hết các điều kiện thì thiết lập dữ liệu trên firebase và kết thúc thuật toán.

Bắt đầu

Khai báo thư viện, Cấu hình Firebase

S

Enter ?

Đ

S

Đã nhập giờ ?

Đ

S

Đã nhập phút ?

Đ

S

Đã nhập nút ?

Đ

Giờ 0-23 ?

Phút 0-59 ?

S

Đ

Hình 4.9 Lưu đồ hẹn giờ trên Android

Kết thúc

Thiết lập dữ liệu trên firebase

#### Phần mềm lập trình cho vi điều khiển:

##### Giới thiệu phần mềm lập trình Arduino IDE:

Arduino IDE 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 viết bằng 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

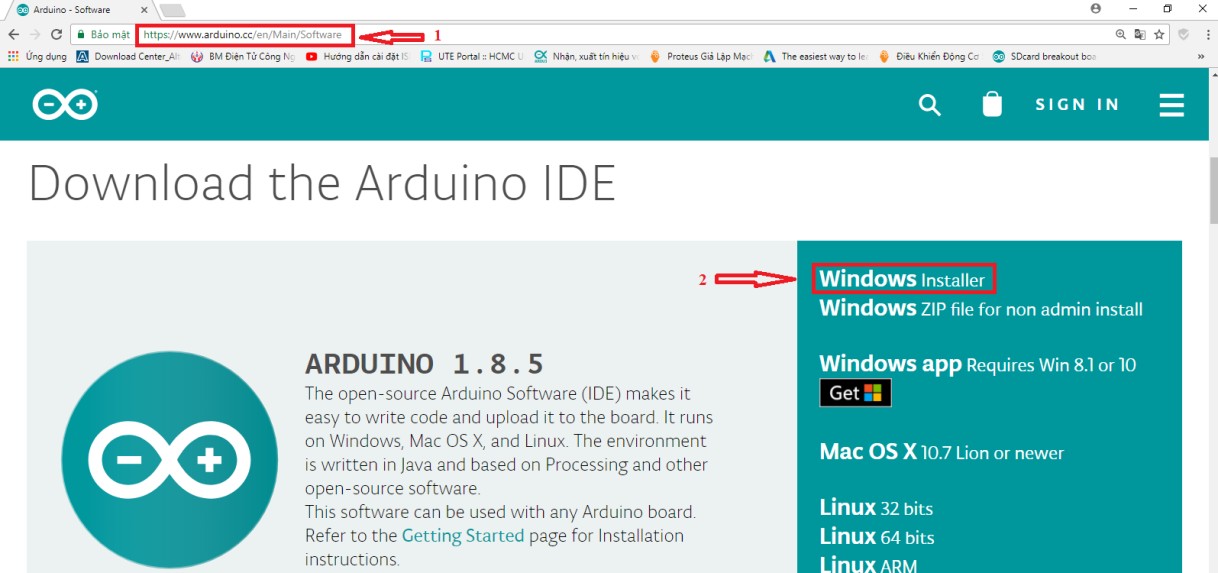
sự thay thế hiện đại cho các IDE khác. Arduino IDE có thư viện code mẫu quá phong phú, viết chương trình trên Arduino IDE khá dễ dàng cộng thêm OpenSource viết riêng cho Arduino thì ngày càng nhiều.



Hình 4.10 Phần mềm lập trình Arduino IDE

Đầu tiên, truy cập vào <https://www.arduino.cc/en/Main/Software>để dowload chương trình theo hệ điều hành muốn cài đặt (ở đây là phiên bản cài trên Windows).

Tiếp theo, nhấn vào Windows Installer nếu muốn cài đặt phần mềm cho Windown.

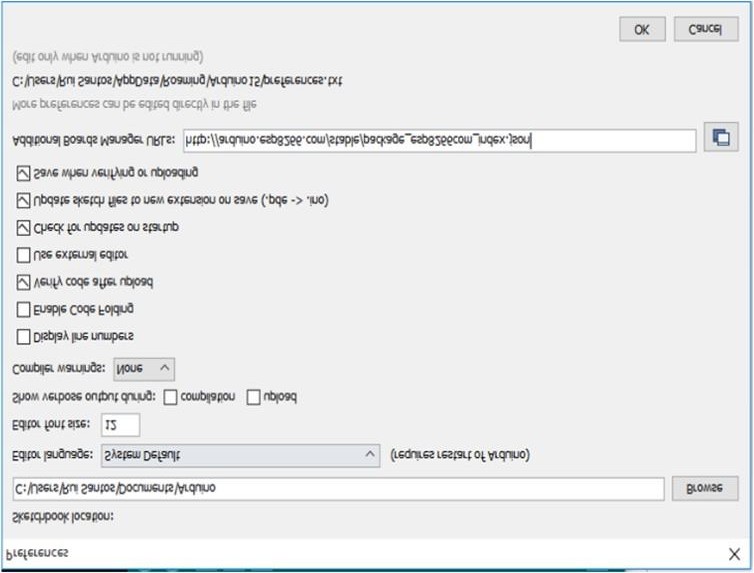


Hình 4.11 Cách download phần mềm

Để có thể nạp được code cho Module Node MCU chúng ta phải thêm board và cài đặt các thư viện cần thiết.

Vào **File→ Preferences**, vào textbox **Additional Board Manager URLs** thêm vào đường link sau: [*http://arduino.esp8266.com/stable/package\_esp8266com\_index.json*](http://arduino.esp8266.com/stable/package_esp8266com_index.json)

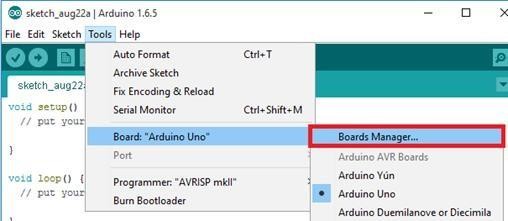
Click OK để chấp nhận.



Hình 4.12 Giao điện thêm board ESP trên Arduino IDE.

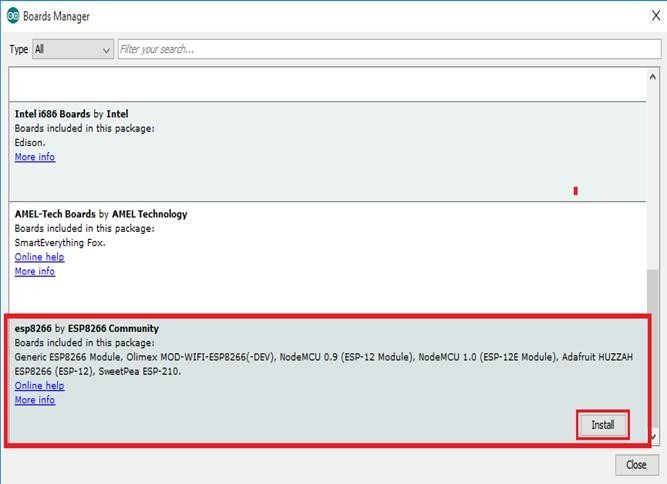
Tiếp theo, sau khi thêm đường link vào phần mềm, chúng ta vào ***Tool > Board***

***> Boards Manager*** để tải bộ thư viện cho board ***Esp8266***.



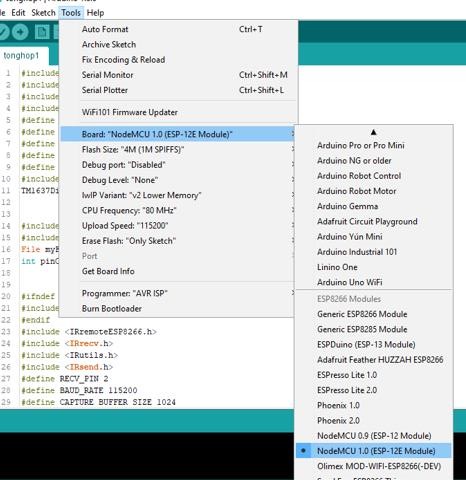
Hình 4.13 Tải thư viện cho Esp8266.

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.



Hình 4.14 Danh mục các thư viện hiện có trên Arduino IDE.

Kết nối mudule USB-to-UART vào máy tính. Vào **Tool**→**Board**→ **NodeMCU 1.0 (ESP-12EModule),** chọn cổng COM tương ứng với module USB- to-UART tương ứng.



Hình 4.15 Chọn Board nạp

Bảng 4.2 Chức năng của các biểu tượng trên thanh công cụ:

|  |  |
| --- | --- |
| Biểu tượng | Chức năng |
|  | Biên dịch chương trình đang soạn thảo để kiểm tra các lỗi lập trình. |
|  | Biên dịch và upload chương trình đang soạn thảo. |
|  | Mở một trang soạn thảo mới. |
|  | Mở các chương trình đã lưu. |
|  | Lưu chương trình đang soạn. |
|  | Mở cửa sổ Serial Monitor để gửi và nhận dữ liệu giữa máy tính và board Arduino |

##### Viết chương trình cho hệ thống:

Chương trình được bổ sung vào phần phụ lục.

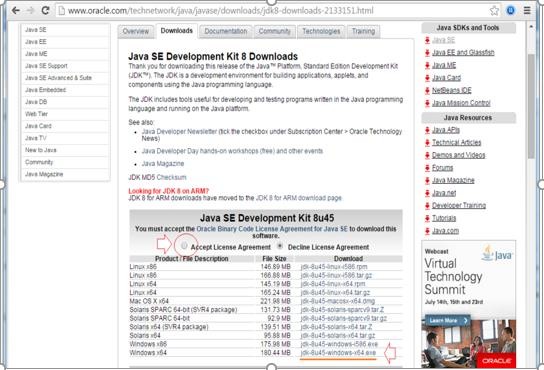
#### Chương trình trên Android Studio.

##### Giới thiệu:

Song song với Eclipse, Android Studio là công cụ lập trình mới dành cho Android do Google phát triển, nó hổ trợ tốt hơn việc bố cục ứng dụng cho nhiều thiết bị khác nhau, đảm bảo app viết ra có thể tương thích với màn hình của cả smartphone (điện thoại thông minh) và tablet (máy tính bảng). Google bổ sung thêm khả năng kéo thả các thành phần đồ họa để quá trình xây dựng ứng dụng trở nên dễ dàng và nhanh chóng hơn. Đồ án sử dụng Android Studio để phát triển ứng dụng do đây là phần mềm IDE phổ biến được Google hỗ trợ các gói ứng dụng thuận lợi cho việc lập trình.

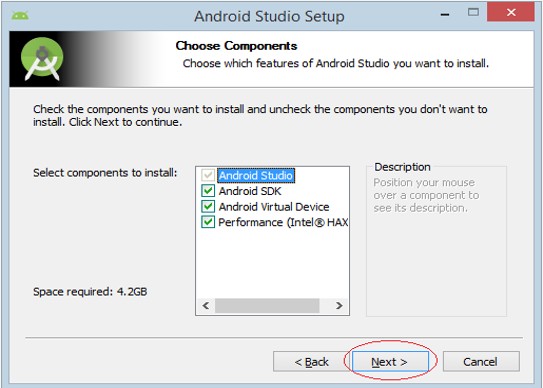
Để cài đặt phần mềm Android Studio đầu tiên tải Android Studio từ trang chủ của nó: [*https://developer.android.com/sdk/index.html*](https://developer.android.com/sdk/index.html).

Lưu ý máy tính phải cài java trước khi cài Android Studio, để cài java vào trang: [*http://www.oracle.com/technetwork/java/javase/downloads/index.html*](http://www.oracle.com/technetwork/java/javase/downloads/index.html)*.* Sau khi tải về tín hành cài đặt như phần mềm bình thường, riêng đối với Window 7 phải cấu hình cho môi trường java.



Hình 4.16 Tải Java

Sau khi cài xong java xong, tiến hành cài đặt Android Studio, trong quá trình cài đặt sẽ xuất hiện màng hình như hình dưới.



Hình 4.17 Cài đặt Android Studio

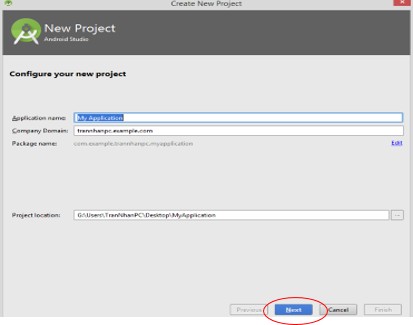
Trước tiên vào phần mềm Android Studio, sẽ thấy màn hình như hình dưới và chọn theo dấu mũi tên màu đỏ.



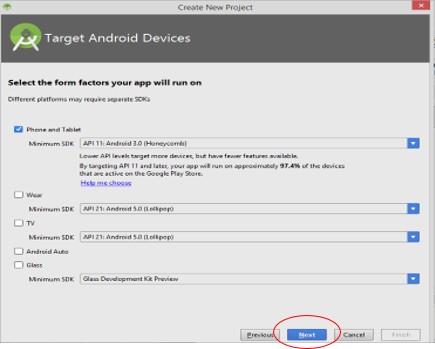
Hình 4.18 Tạo project mới trong Android Studio.

Nhập vào:

* Application Name: tên ứng dụng của muốn lập trình.
* Company Domain: có thể đặt theo định dạng sau *tên mình.tên công ty.tên ứng dụng mình đang làm.*
* Project location: đường dẫn chứa tệp tin chương trình.

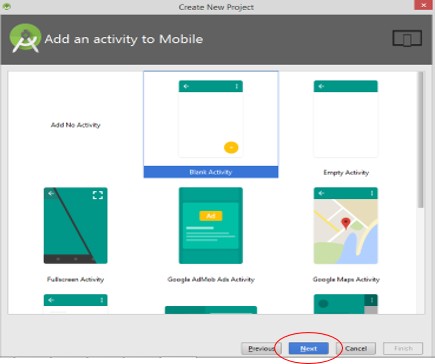


Hình 4.19 Tạo project mới trong Android

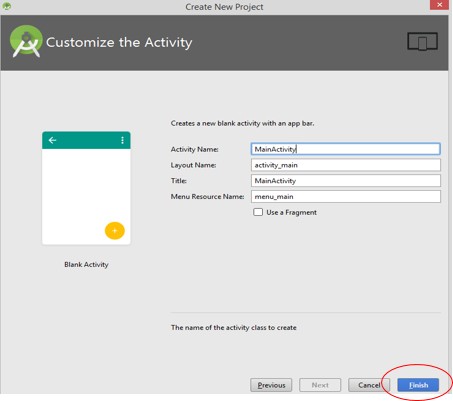
Sau khi đặt tên xong nhấn Next để qua bước tiếp theo, bước này xuất hiện màn hình chọn phiên bản Android để lập trình, lưu ý khi chọn phiên bản Android càng thấp thì ứng dụng tạo ra sẽ chạy được trên nhiều thế hệ.

Hình 4.20 Chọn phiên bản Android muốn lập trình

Sau khi đặt tên xong nhấn Next để qua bước tiếp theo, bước này xuất hiện màn hình chọn phiên bản Android để lập trình, lưu ý khi chọn phiên bản Android càng thấp thì ứng dụng tạo ra sẽ chạy được trên nhiều thế hệ.

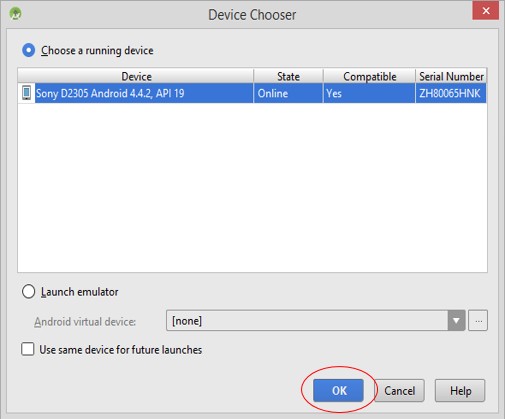


Hình 4.21 Chọn màn hình

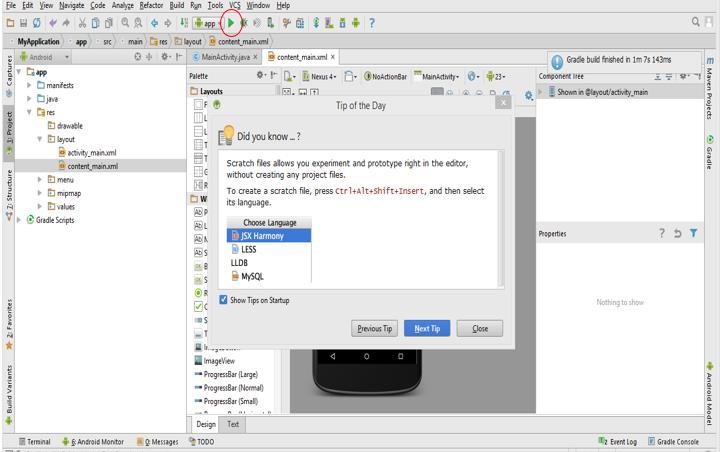


Hình 4.22 Đặt tên cho màn hình

Để chạy thử chương trình nhấn nào nút “Run” chổ khung tròn màu đỏ. Sau đó cấm điện thoại máy tính, chọn tên điện thoại muốn đổ chương trình xuống rồi nhấn nút **“Ok”.**



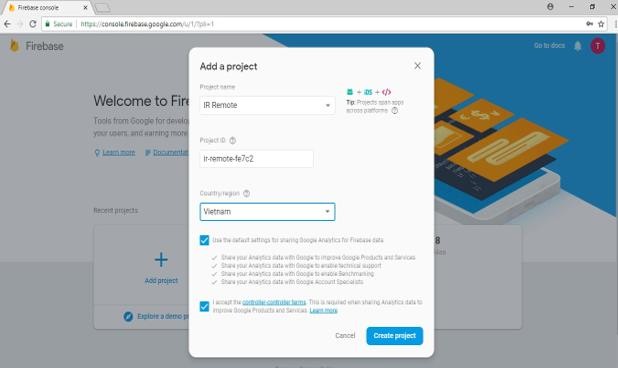
Hình 4.23 Chọn thiết bị đổ chương trình



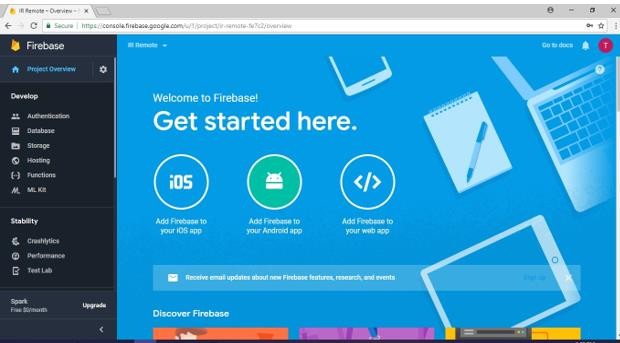
Hình 4.24 Giao diện của project Android mới

##### Tích hợp firebase với app Android:

Để tạo một Firebase project mới, các bạn truy cập vào [**Firebase Console**](https://console.firebase.google.com/), đăng nhập với tài khoản Google của mình, và nhấn **Create new project** và tạo project.

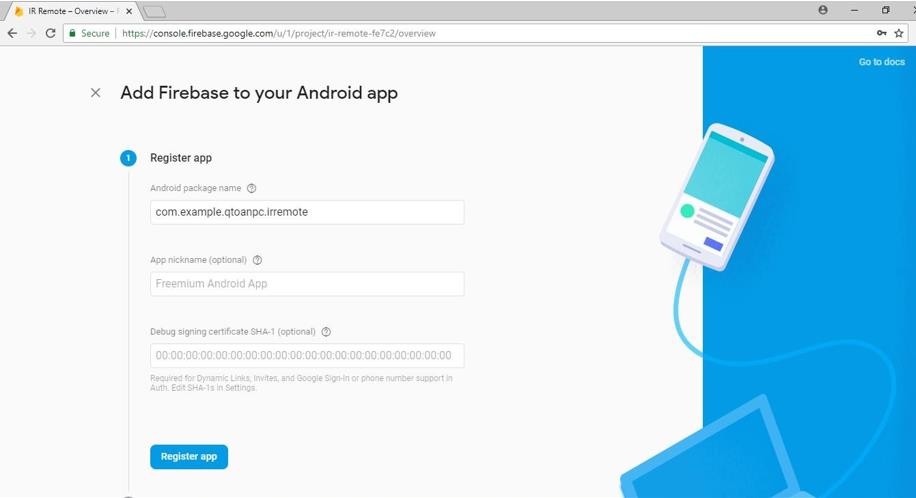


Hình 4.25 Tạo một Project cho Firebase Sau khi tạo xong, ta sẽ thấy giao diện sau:



Hình 4.26 Giao diện các hệ điều hành Firebase liên kết

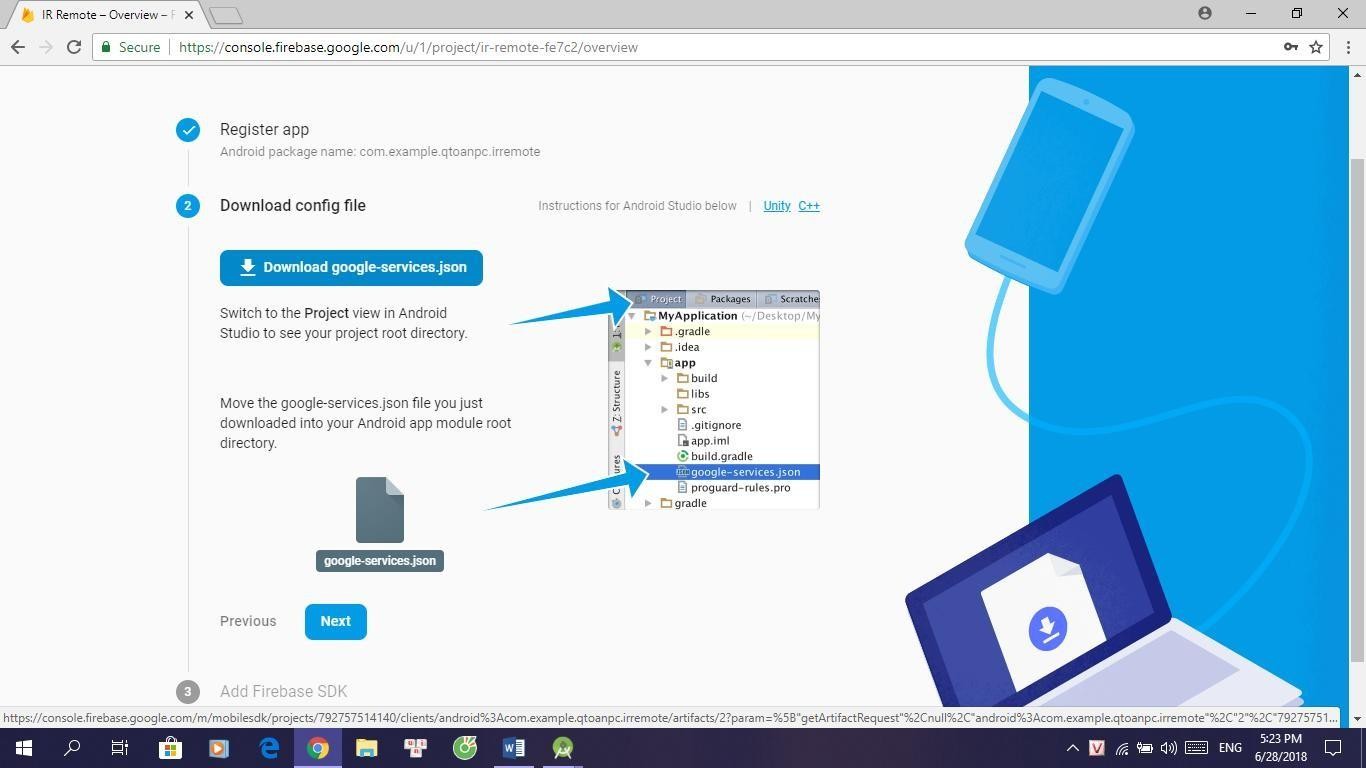
Ở giao diện trên ta chọn **Add Firebase to your Android app**, sẽ xuất hiện giao diện sau:



Hình 4.27 Liên kết Firebase với Package name của app Android

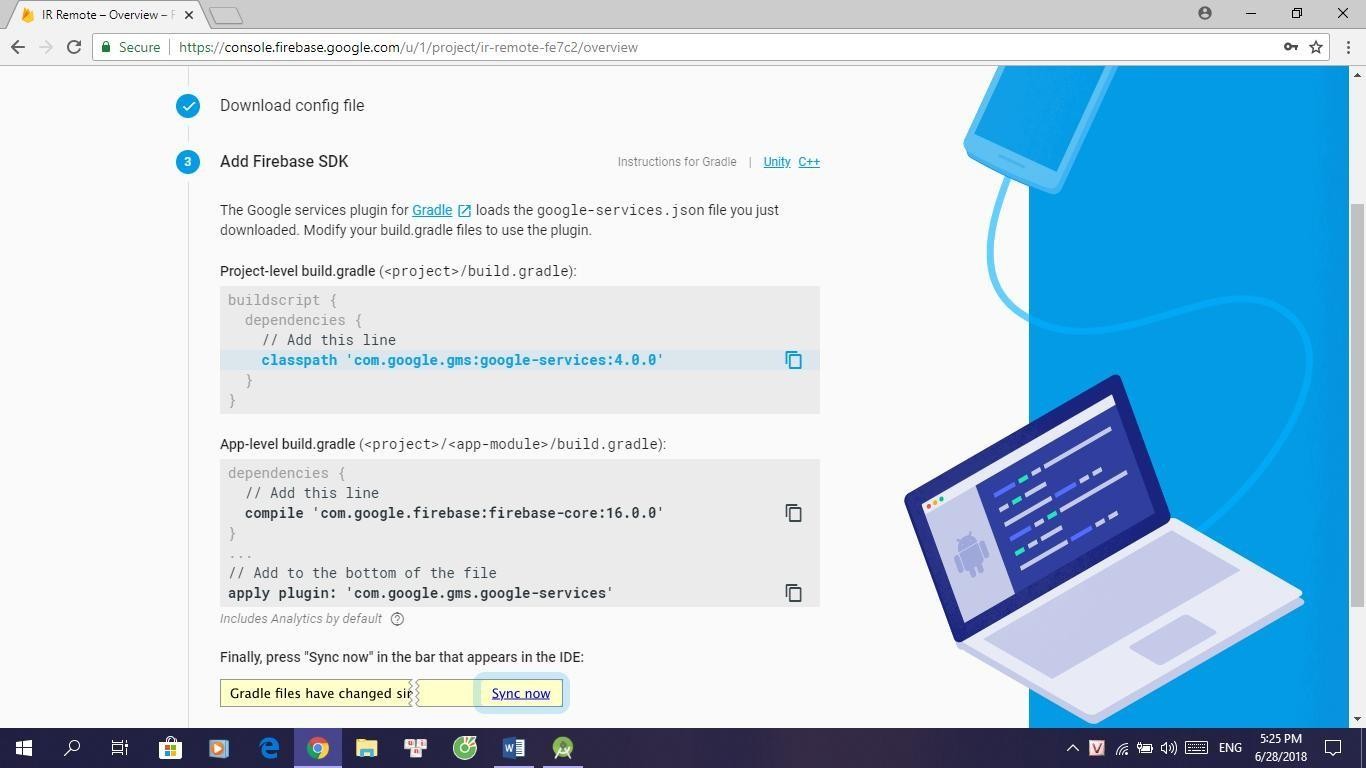
Ở phần Package name, đó chính là tên gói ứng dụng mà bạn đã chọn khi tạo Android project. Phần App nickname, đó là tên mà bạn dùng để quản lí ứng dụng.

Tiếp theo, nút **ADD APP** chính nơi tiếp theo để bạn nhấn. Trình duyệt sẽ tự động download tập tin google-services.json. Bạn chuyển góc nhìn từ Android sang Project trong Studio, và paste tập tin kia vào thư mục app. Tức là google-services.json sẽ ngang hàng với tập tin build.gradle module app có sẵn.



Hình 4.28 Thêm tập tin google-services.json vào app Android

Trong tập tin build.gradle module app và build.gradle project bạn thêm các dòng để tập tin tương tự như sau:



Hình 4.29 Thêm thư viện liên kết Firebase với app Android

Cuối cùng, cho sync Gradle lại. Như vậy là bạn đã tích hợp và liên kết ứng dụng của bạn tới Firebase project rồi đó. Việc tiếp theo là viết code.

##### Viết chương trình trên Android:

Code được bổ sung ở phần phụ lục.

#### HƯỚNG DẪN SỬ DỤNG, THAO TÁC VỚI HỆ THỐNG:

**Bước 1:** Cấp nguồn cho hệ thống, hệ thống sử dụng nguồn 5V DC.

**Bước 2:** Đăng nhập vào Wifi của esp. Với tên là: WIFI\_ESP và không có password. **Bước 3:** Tiến hành config wifi cho ESP để kết nối với Wifi và password tại nơi mình đang sử dụng cho lần đầu tiên đăng nhập.

**Bước 4:** Học lệnh:

Gạt Switch Set ở app lên mức ‘1’ cho phép bắt đầu học lệnh. Nhấn nút muốn học lệnh từ remote của thiết bị muốn điều khiển hướng vào mắt thu hồng ngoại TL 1838 của hộp điều khiển, sau đó nhấn nút muốn gán trên App Android để học lệnh cho nút đó.

Sau đó gạt Switch Set về ‘0’ kết thúc chế độ học lệnh.

**Bước 5:** Hẹn giờ điều khiển:

Ở màn hình của Remote nhấn vào nút TIMER để mở giao diện mới để hẹn

giờ.

Ở giao diện này có 4 giá trị: giờ (giá trị 0-23), phút (0-59), nút thứ nhất và nút thứ hai (có giá trị nhập là:0-9, sao, thang, on1, on2, on3, on4, up, dw, back, next, menu, mute, exit, mode).

Khi đúng giá trị thời gian đặt trước thì gửi tín hiệu điều khiển nút thứ nhất sau thời gian đặt trước 30s thì tiếp tục gửi tín hiệu điều khiển của nút thứ hai.

**Bước 6:** Khi đã gán hoàn tất ta chỉ việc nhấn các nút mà mình đã gán dữ liệu khi đó phần cứng sẽ phát tín hiệu hồng ngoại tương ứng đến thiết bị mà mình muốn điều khiển.

***Lưu ý:*** Khi xảy ra sự cố như không có wifi từ kit, ta nhấn nút reset trên thiết bị hoặc rút nguồn ra cắm lại.

# Chương 5. KẾT QUẢ, NHẬN XÉT, ĐÁNH GIÁ

#### SẢN PHẨM SAU KHI HOÀN THÀNH:

#### Sản phẩm:



Hình 5.1 Sản phẩm thi công

Do vật liệu không thể phát tán hồng ngoại nên ta cần đưa led phát hồng ngoại ra các hướng khác nhau để có thể truyền đến các thiết bị ở nhiều hướng, mặt hộp được thiết theo hình nón cụt để có thể dễ dàng bố trí led và tăng thêm tính thẩm mĩ cho sản phẩm.

Tỉ lệ học lệnh thành công lên đến hơn 95%.

Sản phẩm đã điều khiển được tivi của một số hãng như Samsung, LG, Toshiba, Sony, … với tốc độ đáp ứng khoảng 0,5s.

Có thể hiển thị thời gian thực và hẹn giờ điều khiển thiết bị thông qua module 4 led 7 đoạn.

Chỉ cần đặt hộp điều khiển cách Tivi khoảng 5m, ta có thể điều khiển Tivi thông qua App Android trên điện thoại không giới hạn khoảng cách chỉ cần điện thoại đó có kết nối mạng.

#### Kiến thức có được trong quá trình thiết kế sản phẩm:

Kiến thức về vi xử lý thì học được cách giao tiếp và lập trình cho Node MCU, giao tiếp Module thẻ nhớ thông qua chuẩn SPI.

Biết được cách lập trình ứng dụng trên Android, cách tạo một giao diện, ánh xạ các phần tử bên giao diện qua giao diện lập trình, hiểu thêm được về giao thức mạng cũng như là cách giao tiếp truyền nhận dữ liệu thông qua Realtime Database của Firebase.

Hiểu thêm được các kiến thức về các chuẩn truyền hồng ngoại cũng như là cách các thiết bị nhận biết được tín hiệu hồng ngoại từ điều khiển từ xa gửi đến thiết bị.

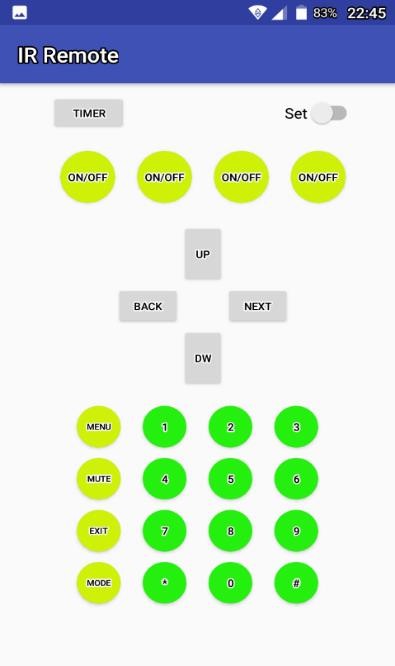
Có thể sử dụng phần mềm công cụ Altium để thiết kế sơ đồ nguyên lý và bố trí linh kiện trên mạch.

#### KẾT QUẢ CHẠY HỆ THỐNG:

#### Quá trình chạy ứng dụng trên điện thoại:



Hình 5.2 Giao diện app



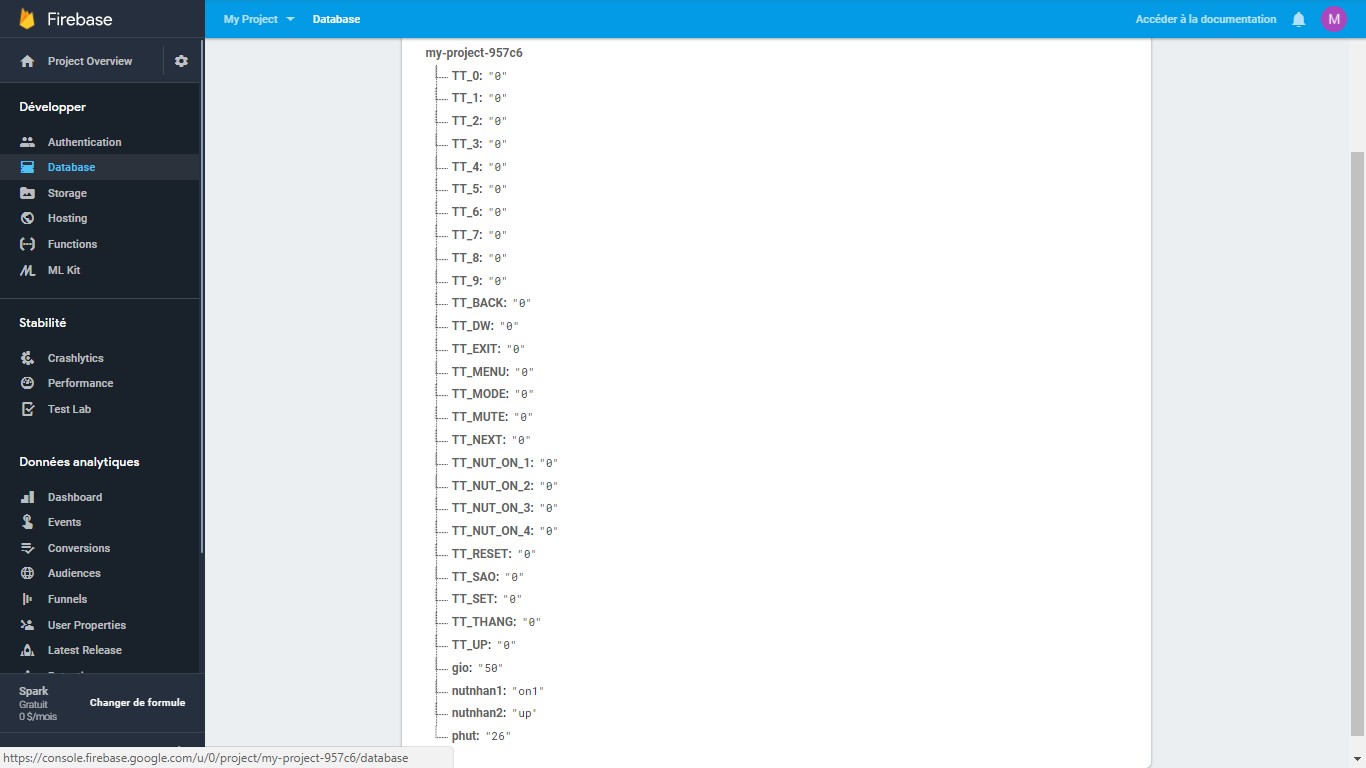
Hình 5.3 Giao diện điều khiển



Hình 5.4 Giao diện hẹn thời gian

#### Quá trình vận hành trên phần cứng hệ thống:

Sau khi nhận được dữ liệu cho phép bắt đầu học tín hiệu thì sẽ phát tín hiệu từ điều khiển từ xa của thiết bị muốn học. Trên Firebase các giá trị sẽ thay đổi theo khi ta nhấn nút trên app Android (giá trị ban đầu là ‘0’ khi ta nhấn nút trên app Android của điện thoại thì giá trị sẽ lên ‘1’).



Hình 5.5 Giao diện trên Firebase

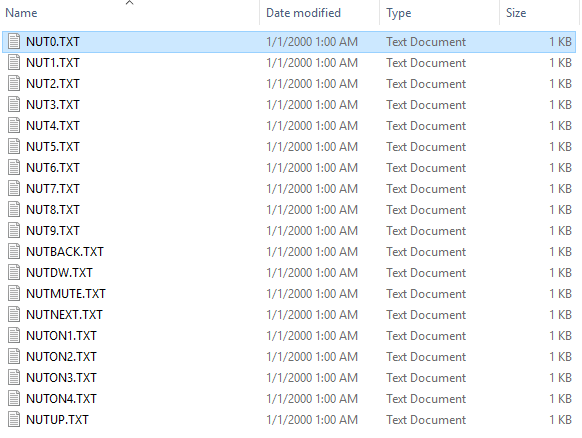


Hình 5.6 Học tín hiệu từ điều khiển

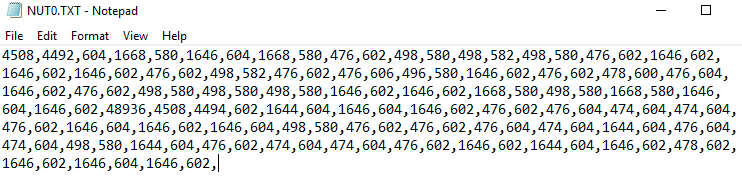


Hình 5.7 Điều khiển thiết bị

Sau khi đã học lệnh thì trong thẻ nhớ sẽ có các thông số của tín hiệu từng nút.



Hình 5.8 Dữ liệu từng nút trong thẻ nhớ



Hình 5.9 Dữ liệu của tín hiệu của một nút trong thẻ nhớ

Từ dữ liệu tín hiệu của một nút lưu trong thẻ nhớ ta có thể biết được số phần tử của dữ liệu, độ rộng xung của từng bit phần tử (đơn vị là µs).

# Chương 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

#### KẾT LUẬN:

* + - Ưu điểm:
      * Hệ thống học được tín hiệu của hầu hết các loại điều khiển từ xa sử dụng tín hiệu hồng ngoại và phát lại tín hiệu hồng ngoại đã học được để điều khiển thiết bị thông qua app Android trên điện thoại.
      * Về phần ứng dụng điện thoại thông minh, đã tạo ra giao diện các nút tương tự như một Remote thực tế
    - Nhược điểm:
      * Hệ thống chỉ hoạt động khi có kết nối Internet, khả năng truyền nhận dữ liệu đôi khi đáp ứng chậm.
      * Khả năng điều khiển máy điều hòa còn hạn chế, kém ổn định.
      * Giao diện app điều khiển còn đơn giản chưa chia ra từng loại thiết bị.

#### HƯỚNG PHÁT TRIỂN:

Tạo cơ sở dữ liệu riêng cho từng loại thiết bị, từng hãng thiết bị để khi điều khiển ta không cần học lệnh nữa mà chỉ cần chọn thiết bị sau đó chọn hãng của thiết bị là có thể điều khiển thiết bị đó một cách đơn giản và hiệu quả.

Có thể kết hợp với ngôi nhà thông minh sẽ trở thành một hệ thống lớn đáp ứng nhu cầu điều khiển, quản lí tất cả các thiết bị trong nhà một cách thông minh, nâng cao đời sống tiện ích cho con người.

#### Sách Tham Khảo

**TÀI LIỆU THAM KHẢO**

1. PGS.TS Trần Thu Hà, ThS. Trương Thị Bích Ngà, TS. Nguyễn Thị Lưỡng, ThS. Bùi Thị Tuyết Đan, ThS. Phù Thị Ngọc Hiếu, ThS. Dương Thị Cẩm Tú, “Giáo trình điện tử cơ bản”, NXB Đại Học Quốc Gia Tp.HCM, 2013.
2. Nguyễn Văn Hiệp, Đinh Quang Hiệp, “Giáo trình lập trình Android Cơ Bản”, NXB Đại Học Quốc Gia Tp.HCM, 2015.
3. Nguyễn Việt Hùng, Nguyễn Ngô Lâm, Nguyễn Văn Phúc, Đặng Phước Hải Trang, “Giáo trình Truyền Số Liệu”, NXB Đại Học Quốc Gia Tp.HCM, 2013.

#### Trang Web Tham Khảo

1. Giao thức đồng bộ thời gian mạng, <https://vi.wikipedia.org/wiki/NTP>
2. Tia hồng ngoại, https://vi.wikipedia.org/wiki
3. Chuẩn giao tiếp SPI, https://kienltb.wordpress.com/2015/04/05/chuan-giao-tiep- spi/
4. Giới thiệu về hệ điều hành Android, https://androidcoban.com/
5. https://firebase.google.com/
6. [https://www.al](http://www.alldatasheet.com/)ldata[sheet.co](http://www.alldatasheet.com/)m/
7. https://github.com/

**PHỤ LỤC**

##### Code Arduino IDE cho vi điều khiển Node MCU:

*#include <ESP8266WiFi.h> #include <WiFiUdp.h> #include <TM1637Display.h> #include <FirebaseArduino.h>*

*#define FIREBASE\_HOST "my-project-957c6.firebaseio.com" #define FIREBASE\_AUTH ""*

*#define CLK D2 #define DIO D3*

*#define time\_zone 3600\*7 #include <WiFiManager.h>*

*TM1637Display display(CLK, DIO); #include <SD.h>*

*#include <SPI.h> File myFile;*

*int pinCS = 15; #ifndef UNIT\_TEST #include <Arduino.h> #endif*

*#include <IRremoteESP8266.h> #include <IRrecv.h>*

*#include <IRutils.h> #include <IRsend.h> #define RECV\_PIN 2*

*#define BAUD\_RATE 115200*

*#define CAPTURE\_BUFFER\_SIZE 1024*

*#define IR\_LED 5 IRsend irsend(IR\_LED);*

*extern uint16\_t array\_[300]; int i=0;*

*int a=0;*

*int tggio=0,tgphut=0;*

*String nutset="",nutreset="";*

*String nut0="",nut1="", nut2="", nut3="", nut4="", nut5="", nut6="", nut7="", nut8="", nut9="", nutsao="", nutthang="";*

*String nuton1="", nuton2="", nuton3="", nuton4="", nutup="", nutdw="", nutback="", nutnext="", nutmenu="", nutmute="", nutexit="", nutmode="";*

*String giohen="", phuthen="", nutnhanhen1="", nutnhanhen2=""; String path = "/";*

*byte s, trunggian=0; #if DECODE\_AC*

*#define TIMEOUT 50U #else // DECODE\_AC #define TIMEOUT 15U #endif // DECODE\_AC*

*#define MIN\_UNKNOWN\_SIZE 12*

*const int button\_config = D0;*

*IRrecv irrecv(RECV\_PIN, CAPTURE\_BUFFER\_SIZE, TIMEOUT, true); decode\_results results;*

*unsigned long epoch; int hh;*

*int mm; int ss;*

*int force\_update = 1; bool showdot = false;*

*uint32\_t ts, ts1, ts2, ts3, ts4; unsigned int localPort = 2390;*

*int ip = 0;*

*String IP\_of\_Server[5] = {"122.155.169.213", "129.6.15.28", "158.108.212.149",*

*"203.158.118.2", "129.250.35.250"};*

*IPAddress timeServer;*

*const int NTP\_PACKET\_SIZE = 48;*

*byte packetBuffer[ NTP\_PACKET\_SIZE]; WiFiUDP udp;*

*unsigned long sendNTPpacket(IPAddress& address)*

*{*

*Serial.println("sending NTP packet..."); memset(packetBuffer, 0, NTP\_PACKET\_SIZE); packetBuffer[0] = 0b11100011; packetBuffer[1] = 0;*

*packetBuffer[2] = 6; packetBuffer[3] = 0xEC; packetBuffer[12] = 49; packetBuffer[13] = 0x4E; packetBuffer[14] = 49;*

*packetBuffer[15] = 52;*

*udp.beginPacket(address, 123); udp.write(packetBuffer, NTP\_PACKET\_SIZE); udp.endPacket();*

*}*

*void NTP\_get(void)*

*{*

*sendNTPpacket(timeServer); delay(1000);*

*int cb = udp.parsePacket(); if (!cb) {*

*force\_update = 1;*

*}*

*else {*

*force\_update = 0;*

*udp.read(packetBuffer, NTP\_PACKET\_SIZE); // read the packet into the buffer unsigned long highWord = word(packetBuffer[40], packetBuffer[41]); unsigned long lowWord = word(packetBuffer[42], packetBuffer[43]); unsigned long secsSince1900 = highWord << 16 | lowWord;*

*const unsigned long seventyYears = 2208988800UL; epoch = secsSince1900 - seventyYears + time\_zone;*

*}*

*}*

*void setup() {*

*Serial.begin(BAUD\_RATE, SERIAL\_8N1, SERIAL\_TX\_ONLY);*

*while (!Serial) // Wait for the serial connection to be establised. delay(50);*

*Serial.println();*

*Serial.print("IRrecvDumpV2 is now running and waiting for IR input on Pin "); Serial.println(RECV\_PIN);*

*irsend.begin();*

*pinMode(button\_config, INPUT\_PULLUP);*

*// device\_uid += String(ESP.getChipId());*

*// ssid\_apmode += String(ESP.getChipId());*

*// -- begin configuration of the portal --*

*if ( digitalRead(button\_config) == LOW ){ Serial.println("VAO TRANG CHON WIFI");*

*WiFiManager wifiManager;*

*// set the callback used for notifying if there was a change*

*//wifiManager.setSaveConfigCallback(saveConfigCallback);*

*// add custom parameters to portal§*

*WiFiManagerParameter custom\_text0("<p><strong> SMART CONTROL </p>*

*</strong>"); wifiManager.addParameter(&custom\_text0); wifiManager.setTimeout(1000);*

*//ssid\_apmode += String(ESP.getChipId());*

*//if (!wifiManager.startConfigPortal(ssid\_apmode.c\_str(),pass\_apmode)) if (!wifiManager.startConfigPortal("WIFI\_ESP",""))*

*{*

*Serial.println("failed to connect and hit timeout"); Serial.println("6s");*

*delay(3000);*

*//reset and try again, or maybe put it to deep sleep ESP.reset();*

*delay(3000); Serial.println("test aaaa");*

*}*

*}*

*else{*

*Serial.println("KHONG VAO TRANG CHON WIFI");*

*}*

*// -- end configuration --*

*while (WiFi.status() != WL\_CONNECTED) { Serial.print(".");*

*delay(500);*

*}*

*Serial.println();*

*Serial.print("connected: "); Serial.println(WiFi.localIP()); Firebase.begin(FIREBASE\_HOST, FIREBASE\_AUTH); #if DECODE\_HASH*

*// Ignore messages with less than minimum on or off pulses. irrecv.setUnknownThreshold(MIN\_UNKNOWN\_SIZE); #endif // DECODE\_HASH*

*irrecv.enableIRIn(); // Start the receiver*

*Serial.println("Starting UDP"); udp.begin(localPort); Serial.print("Local port: "); Serial.println(udp.localPort());*

*//Get package from NTP while (force\_update == 1) { if (ip >= 5) ip = 0;*

*timeServer.fromString(IP\_of\_Server[ip]); Serial.println(timeServer);*

*ip += 1; NTP\_get();*

*}*

*delay(100);*

*hh = (epoch % 86400L) / 3600; mm = (epoch % 3600) / 60;*

*ss = (epoch % 60);*

*ts = ts1 = ts2 = ts3 = ts4 = millis(); digitalWrite(2, HIGH); //SSID connect ready*

*pinMode(pinCS, OUTPUT); SD.begin();*

*}*

*void getRealTime()*

*{*

*ts = millis();*

*if (WiFi.status() == WL\_CONNECTED) {*

*/////////////////////////check time every 1 Hr.////////////////////////////////*

*if ((WiFi.status() == WL\_CONNECTED) && (ts - ts3 >= 3600000) && (force\_update == 0)) {*

*NTP\_get(); ts3 = millis();*

*}*

*}*

*if ( ts - ts1 >= 1000 ) { epoch++; //Add a second*

*hh = (epoch % 86400L) / 3600; mm = (epoch % 3600) / 60;*

*ss = (epoch % 60);*

*ts1 += 1000; // increment counter by 1 every 1sec*

*}*

*if ( ts - ts4 >= 500 ) { display.dotShow(showdot = !showdot); ts4 = millis();*

*}*

*if ( ts - ts2 >= 5 ) { display.write(hh / 10, 0);*

*display.write(hh % 10, 1);*

*display.write(mm / 10, 2);*

*display.write(mm % 10, 3);*

*}*

*}*

*void getBTNvalue()*

*{*

*FirebaseObject object = Firebase.get(path); nutset =object.getString("TT\_SET"); nutreset=object.getString("TT\_RESET");*

*nut0 =object.getString("TT\_0"); nut1 =object.getString("TT\_1"); nut2 =object.getString("TT\_2"); nut3 =object.getString("TT\_3"); nut4 =object.getString("TT\_4"); nut5 =object.getString("TT\_5"); nut6 =object.getString("TT\_6"); nut7 =object.getString("TT\_7"); nut8 =object.getString("TT\_8"); nut9 =object.getString("TT\_9"); nutsao =object.getString("TT\_SAO");*

*nutthang=object.getString("TT\_THANG");*

*nuton1 =object.getString("TT\_NUT\_ON\_1"); nuton2 =object.getString("TT\_NUT\_ON\_2"); nuton3 =object.getString("TT\_NUT\_ON\_3"); nuton4 =object.getString("TT\_NUT\_ON\_4");*

*nutup =object.getString("TT\_UP"); nutdw =object.getString("TT\_DW"); nutback =object.getString("TT\_BACK"); nutnext =object.getString("TT\_NEXT");*

*nutmenu =object.getString("TT\_MENU"); nutmute =object.getString("TT\_MUTE"); nutexit =object.getString("TT\_EXIT"); nutmode =object.getString("TT\_MODE");*

*giohen =object.getString("gio"); phuthen =object.getString("phut");*

*nutnhanhen1=object.getString("nutnhan1"); nutnhanhen2=object.getString("nutnhan2");*

*}*

*void loop()*

*{*

*getRealTime(); getBTNvalue(); choosefileSD(); hengio();*

*if(nutset=="1")*

*{*

*if (irrecv.decode(&results))*

*{*

*// Display a crude timestamp. uint32\_t now = millis();*

*Serial.printf("Timestamp : %06u.%03u\n", now / 1000, now % 1000); if (results.overflow)*

*Serial.printf("WARNING: IR code is too big for buffer (>= %d). " "This result shouldn't be trusted until this is resolved. "*

*"Edit & increase CAPTURE\_BUFFER\_SIZE.\n", CAPTURE\_BUFFER\_SIZE);*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Serial.print(resultToHumanReadableBasic(&results));*  *yield(); // Feed the WDT as the text output can take a while to print. Serial.println(resultToSourceCode(&results));*  *}*  *a=getCorrectedRawLength(&results); writetoSD();*  *}*  *else ReadFromSD();*  *}*  *void choosefileSD()*  *{*  *s=0;*  *if(nutset=="1")*  *{* | | | | |
| *if(nut0=="1")* | *{SD.remove("nut0.txt");* | *myFile* | *=* | *SD.open("nut0.txt",* |
| *FILE\_WRITE);s++;}* |  |  |  |  |
| *if(nut1=="1")* | *{SD.remove("nut1.txt");* | *myFile* | *=* | *SD.open("nut1.txt",* |
| *FILE\_WRITE);s++;}* |  |  |  |  |
| *if(nut2=="1")* | *{SD.remove("nut2.txt");* | *myFile* | *=* | *SD.open("nut2.txt",* |
| *FILE\_WRITE);s++;}* |  |  |  |  |
| *if(nut3=="1")* | *{SD.remove("nut3.txt");* | *myFile* | *=* | *SD.open("nut3.txt",* |
| *FILE\_WRITE);s++;}* |  |  |  |  |
| *if(nut4=="1")* | *{SD.remove("nut4.txt");* | *myFile* | *=* | *SD.open("nut4.txt",* |
| *FILE\_WRITE);s++;}* |  |  |  |  |
| *if(nut5=="1")* | *{SD.remove("nut5.txt");* | *myFile* | *=* | *SD.open("nut5.txt",* |
| *FILE\_WRITE);s++;}* |  |  |  |  |
| *if(nut6=="1")* | *{SD.remove("nut6.txt");* | *myFile* | *=* | *SD.open("nut6.txt",* |
| *FILE\_WRITE);s++;}* |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *if(nut7=="1")*  *FILE\_WRITE);s++;}* | *{SD.remove("nut7.txt");* | *myFile* | *=* | *SD.open("nut7.txt",* |
| *if(nut8=="1")* | *{SD.remove("nut8.txt");* | *myFile* | *=* | *SD.open("nut8.txt",* |
| *FILE\_WRITE);s++;}* |  |  |  |  |
| *if(nut9=="1")* | *{SD.remove("nut9.txt");* | *myFile* | *=* | *SD.open("nut9.txt",* |
| *FILE\_WRITE);s++;}* |  |  |  |  |
| *if(nutsao=="1") {SD.remove("nutsao.txt"); myFile = SD.open("nutsao.txt", FILE\_WRITE);s++;}*  *if(nutthang=="1") {SD.remove("nutthang.txt");myFile = SD.open("nutthang.txt", FILE\_WRITE);s++;}*  *if(nuton1=="1") {SD.remove("nuton1.txt"); myFile = SD.open("nuton1.txt", FILE\_WRITE);s++;}*  *if(nuton2=="1") {SD.remove("nuton2.txt"); myFile = SD.open("nuton2.txt", FILE\_WRITE);s++;}*  *if(nuton3=="1") {SD.remove("nuton3.txt"); myFile = SD.open("nuton3.txt",*  *FILE\_WRITE);s++;}* | | | | |
| *if(nuton4=="1")* | *{SD.remove("nuton4.txt");* | *myFile = SD.open("nuton4.txt",* | | |
| *FILE\_WRITE);s++;}* |  |  | | |
| *if(nutup=="1")* | *{SD.remove("nutup.txt");* | *myFile = SD.open("nutup.txt",* | | |
| *FILE\_WRITE);s++;}* |  |  | | |
| *if(nutdw=="1")* | *{SD.remove("nutdw.txt");* | *myFile = SD.open("nutdw.txt",* | | |
| *FILE\_WRITE);s++;}*  *if(nutback=="1") {SD.remove("nutback.txt"); myFile = SD.open("nutback.txt", FILE\_WRITE);s++;}*  *if(nutnext=="1") {SD.remove("nutnext.txt"); myFile = SD.open("nutnext.txt", FILE\_WRITE);s++;}*  *if(nutmenu=="1") {SD.remove("nutmenu.txt"); myFile = SD.open("nutmenu.txt", FILE\_WRITE);s++;}* | | | | |

*if(nutmute=="1") {SD.remove("nutmute.txt"); myFile = SD.open("nutmute.txt", FILE\_WRITE);s++;}*

*if(nutexit=="1") {SD.remove("nutnexit.txt");myFile = SD.open("nutexit.txt", FILE\_WRITE);s++;}*

*if(nutmode=="1") {SD.remove("nutmode.txt"); myFile = SD.open("nutmode.txt", FILE\_WRITE);s++;}*

*}*

*else if(nutset=="0")*

*{*

|  |  |  |
| --- | --- | --- |
| *if(nut0=="1")* | *{myFile = SD.open("nut0.txt");* | *s++;}* |
| *if(nut1=="1")* | *{myFile = SD.open("nut1.txt");* | *s++;}* |
| *if(nut2=="1")* | *{myFile = SD.open("nut2.txt");* | *s++;}* |
| *if(nut3=="1")* | *{myFile = SD.open("nut3.txt");* | *s++;}* |
| *if(nut4=="1")* | *{myFile = SD.open("nut4.txt");* | *s++;}* |
| *if(nut5=="1")* | *{myFile = SD.open("nut5.txt");* | *s++;}* |
| *if(nut6=="1")* | *{myFile = SD.open("nut6.txt");* | *s++;}* |
| *if(nut7=="1")* | *{myFile = SD.open("nut7.txt");* | *s++;}* |
| *if(nut8=="1")* | *{myFile = SD.open("nut8.txt");* | *s++;}* |
| *if(nut9=="1")* | *{myFile = SD.open("nut9.txt");* | *s++;}* |

*if(nutsao=="1") {myFile = SD.open("nutsao.txt"); s++;} if(nutthang=="1") {myFile = SD.open("nutthang.txt"); s++;}*

|  |  |  |
| --- | --- | --- |
| *if(nuton1=="1")* | *{myFile = SD.open("nuton1.txt");* | *s++;}* |
| *if(nuton2=="1")* | *{myFile = SD.open("nuton2.txt");* | *s++;}* |
| *if(nuton3=="1")* | *{myFile = SD.open("nuton3.txt");* | *s++;}* |
| *if(nuton4=="1")* | *{myFile = SD.open("nuton4.txt");* | *s++;}* |

*if(nutup=="1") {myFile = SD.open("nutup.txt"); s++;} if(nutdw=="1") {myFile = SD.open("nutdw.txt"); s++;} if(nutback=="1") {myFile = SD.open("nutback.txt"); s++;}*

*if(nutnext=="1") {myFile = SD.open("nutnext.txt"); s++;} if(nutexit=="1") {myFile = SD.open("nutexit.txt"); s++;} if(nutmenu=="1") {myFile = SD.open("nutmenu.txt"); s++;} if(nutmode=="1") {myFile = SD.open("nutmode.txt"); s++;} if(nutmute=="1") {myFile = SD.open("nutmute.txt"); s++;}*

*}*

*}*

*void writetoSD()*

*{*

*if(s>0)*

*{*

*if (myFile)*

*{*

*for(i=0;i<a;i++)*

*{*

*myFile.print(String(array\_[i])); myFile.print(",");*

*}*

*myFile.close(); // close the file*

*}*

*}*

*}*

*void ReadFromSD()*

*{*

*if(s>0)*

*{*

*uint16\_t mang[400]; uint16\_t k=0, moc; String myString;*

*if (myFile)*

*{*

*while (myFile.available())*

*{*

*char c= myFile.read(); myString= myString + String(c);*

*}*

*myFile.close(); Serial.println(" " + myString);*

*Serial.print("Dodaichuoi: ");Serial.print(myString.length()); Serial.println();*

*for (uint16\_t n = 0; n < myString.length(); n++)*

*{*

*if (myString.charAt(n) == ',')*

*{*

*moc = n;*

*mang[k]= myString.toInt(); myString.remove(0,moc+1); k++;*

*n=0;*

*}*

*}*

*Serial.print("Dodaimang: ");Serial.print(k); Serial.println();*

*// for(uint16\_t m=0;m<k;m++)*

*// {*

*// Serial.print(mang[m]); Serial.print(" ");*

*// }*

*// Serial.println(); irsend.sendRaw(mang, k, 38);*

*Serial.println("da gui ma IR");*

*}*

*else Serial.println("error opening tenfile.txt");*

*}*

*}*

*void hengio()*

*{*

*tggio =giohen.toInt(); tgphut =phuthen.toInt();*

*if((tggio==hh)&&(tgphut==mm))*

*{*

*s=1;*

*Serial.println("lam viec thoi"); Serial.println(nutnhanhen1); timnutnhanhengio(); Firebase.setString("gio", "50"); ReadFromSD();*

*s=0;*

*}*

*if((trunggian==1)&&(ss==30))*

*{*

*s=1;*

*Serial.println("lam viec lan 2"); Serial.println(nutnhanhen2); timnutnhanhengio2(); ReadFromSD();*

*s=0;*

*}*

*}*

*///////////////////////////////////////////////////////////// void timnutnhanhengio()*

*{*

*if(nutnhanhen1=="0")*

*{*

*Serial.println("mo file so 0"); myFile = SD.open("nut0.txt");*

*}*

*if(nutnhanhen1=="1")*

*{*

*Serial.println("mo file so 1"); myFile = SD.open("nut1.txt");*

*}*

*if(nutnhanhen1=="2")*

*{*

*Serial.println("mo file so 2"); myFile = SD.open("nut2.txt");*

*}*

*if(nutnhanhen1=="3")*

*{*

*Serial.println("mo file so 3"); myFile = SD.open("nut3.txt");*

*}*

*if(nutnhanhen1=="4")*

*{*

*Serial.println("mo file so 4"); myFile = SD.open("nut4.txt");*

*}*

*if(nutnhanhen1=="5")*

*{*

*Serial.println("mo file so 5"); myFile = SD.open("nut5.txt");*

*}*

*if(nutnhanhen1=="6")*

*{*

*Serial.println("mo file so 6"); myFile = SD.open("nut6.txt");*

*}*

*if(nutnhanhen1=="7")*

*{*

*Serial.println("mo file so 7"); myFile = SD.open("nut7.txt");*

*}*

*if(nutnhanhen1=="8")*

*{*

*Serial.println("mo file so 8"); myFile = SD.open("nut8.txt");*

*}*

*if(nutnhanhen1=="9")*

*{*

*Serial.println("mo file so 9"); myFile = SD.open("nut9.txt");*

*}*

*if(nutnhanhen1=="sao")*

*{*

*Serial.println("mo file sao"); myFile = SD.open("nutsao.txt");*

*}*

*if(nutnhanhen1=="thang")*

*{*

*Serial.println("mo file thang"); myFile = SD.open("nutthang.txt");*

*}*

*if(nutnhanhen1=="on1")*

*{*

*Serial.println("mo file on1"); myFile = SD.open("nuton1.txt");*

*}*

*if(nutnhanhen1=="on2")*

*{*

*Serial.println("mo file on2"); myFile = SD.open("nuton2.txt");*

*}*

*if(nutnhanhen1=="on3")*

*{*

*Serial.println("mo file on3"); myFile = SD.open("nuton3.txt");*

*}*

*if(nutnhanhen1=="on4")*

*{*

*Serial.println("mo file on4"); myFile = SD.open("nuton4.txt");*

*}*

*if(nutnhanhen1=="up")*

*{*

*Serial.println("mo file up");*

*myFile = SD.open("nutup.txt");*

*}*

*if(nutnhanhen1=="dw")*

*{*

*Serial.println("mo file dw"); myFile = SD.open("nutdw.txt");*

*}*

*if(nutnhanhen1=="back")*

*{*

*Serial.println("mo file back"); myFile = SD.open("nutback.txt");*

*}*

*if(nutnhanhen1=="next")*

*{*

*Serial.println("mo file next"); myFile = SD.open("nutnext.txt");*

*}*

*if(nutnhanhen1=="menu")*

*{*

*Serial.println("mo file menu"); myFile = SD.open("nutmenu.txt");*

*}*

*if(nutnhanhen1=="mute")*

*{*

*Serial.println("mo file mute"); myFile = SD.open("nutmute.txt");*

*}*

*if(nutnhanhen1=="exit")*

*{*

*Serial.println("mo file exit");*

*myFile = SD.open("nutexit.txt");*

*}*

*if(nutnhanhen1=="mode")*

*{*

*Serial.println("mo file mode"); myFile = SD.open("nutmode.txt");*

*}*

*trunggian=1;*

*}*

*//////////////////////////////////////////////// void timnutnhanhengio2()*

*{*

*if(nutnhanhen2=="0")*

*{*

*Serial.println("mo file so 0"); myFile = SD.open("nut0.txt");*

*}*

*if(nutnhanhen2=="1")*

*{*

*Serial.println("mo file so 1"); myFile = SD.open("nut1.txt");*

*}*

*if(nutnhanhen2=="2")*

*{*

*Serial.println("mo file so 2"); myFile = SD.open("nut2.txt");*

*}*

*if(nutnhanhen2=="3")*

*{*

*Serial.println("mo file so 3"); myFile = SD.open("nut3.txt");*

*}*

*if(nutnhanhen2=="4")*

*{*

*Serial.println("mo file so 4"); myFile = SD.open("nut4.txt");*

*}*

*if(nutnhanhen2=="5")*

*{*

*Serial.println("mo file so 5"); myFile = SD.open("nut5.txt");*

*}*

*if(nutnhanhen2=="6")*

*{*

*Serial.println("mo file so 6"); myFile = SD.open("nut6.txt");*

*}*

*if(nutnhanhen2=="7")*

*{*

*Serial.println("mo file so 7"); myFile = SD.open("nut7.txt");*

*}*

*if(nutnhanhen2=="8")*

*{*

*Serial.println("mo file so 8"); myFile = SD.open("nut8.txt");*

*}*

*if(nutnhanhen2=="9")*

*{*

*Serial.println("mo file so 9"); myFile = SD.open("nut9.txt");*

*}*

*if(nutnhanhen2=="sao")*

*{*

*Serial.println("mo file sao"); myFile = SD.open("nutsao.txt");*

*}*

*if(nutnhanhen2=="thang")*

*{*

*Serial.println("mo file thang"); myFile = SD.open("nutthang.txt");*

*}*

*if(nutnhanhen2=="on1")*

*{*

*Serial.println("mo file on1"); myFile = SD.open("nuton1.txt");*

*}*

*if(nutnhanhen2=="on2")*

*{*

*Serial.println("mo file on2"); myFile = SD.open("nuton2.txt");*

*}*

*if(nutnhanhen2=="on3")*

*{*

*Serial.println("mo file on3"); myFile = SD.open("nuton3.txt");*

*}*

*if(nutnhanhen2=="on4")*

*{*

*Serial.println("mo file on4"); myFile = SD.open("nuton4.txt");*

*}*

*if(nutnhanhen2=="up")*

*{*

*Serial.println("mo file up"); myFile = SD.open("nutup.txt");*

*}*

*if(nutnhanhen2=="dw")*

*{*

*Serial.println("mo file dw"); myFile = SD.open("nutdw.txt");*

*}*

*if(nutnhanhen2=="back")*

*{*

*Serial.println("mo file back"); myFile = SD.open("nutback.txt");*

*}*

*if(nutnhanhen2=="next")*

*{*

*Serial.println("mo file next"); myFile = SD.open("nutnext.txt");*

*}*

*if(nutnhanhen2=="menu")*

*{*

*Serial.println("mo file menu"); myFile = SD.open("nutmenu.txt");*

*}*

*if(nutnhanhen2=="mute")*

*{*

*Serial.println("mo file mute"); myFile = SD.open("nutmute.txt");*

*}*

*if(nutnhanhen2=="exit")*

*{*

*Serial.println("mo file exit"); myFile = SD.open("nutexit.txt");*

*}*

*if(nutnhanhen2=="mode")*

*{*

*Serial.println("mo file mode"); myFile = SD.open("nutmode.txt");*

*}*

*trunggian=0;*

*}*

##### Code trên Android studio:

//activity\_dieukhien.xml////////////////////////////////////////////////////////////////

*<?xml version="1.0" encoding="utf-8"?>*

*<RelativeLayout android:layout\_width="match\_parent" android:layout\_height="match\_parent" xmlns:android="*[*http://schemas.android.com/apk/res/android"*](http://schemas.android.com/apk/res/android)*>*

*<Button android:id="@+id/Buttontimers" android:layout\_width="70dp" android:layout\_height="37dp"*

*android:layout\_alignParentStart="true" android:layout\_alignParentTop="true" android:layout\_marginStart="46dp"*

*android:layout\_marginTop="10dp" android:textSize="10dp" android:text="Timer" />*

*<Switch android:id="@+id/switchs"*

*android:layout\_width="wrap\_content" android:layout\_height="37dp" android:layout\_alignParentEnd="true" android:layout\_alignTop="@+id/Buttontimers" android:layout\_marginEnd="33dp" android:text="Set" />*

*<Button android:id="@+id/buttonon1"*

*android:layout\_below="@+id/Buttontimers" android:layout\_marginTop="18dp" android:layout\_marginLeft="55dp" android:background="@drawable/tkbt" android:layout\_width="50dp" android:layout\_height="50dp" android:textSize="10dp" android:text="ON/OFF" />*

*<Button android:id="@+id/buttonon2" android:layout\_width="50dp" android:layout\_height="50dp"*

*android:layout\_toRightOf="@id/buttonon1" android:layout\_marginTop="65dp" android:layout\_marginLeft="20dp" android:background="@drawable/tkbt" android:text="ON/OFF" android:textSize="10dp" />*

*<Button android:id="@+id/buttonon3"*

*android:layout\_width="50dp" android:layout\_height="50dp" android:layout\_toRightOf="@id/buttonon2" android:layout\_marginTop="65dp" android:layout\_marginLeft="20dp" android:background="@drawable/tkbt" android:text="ON/OFF"*

*android:textSize="10dp" />*

*<Button android:id="@+id/buttonon4"*

*android:layout\_width="50dp" android:layout\_height="50dp" android:layout\_marginLeft="20dp" android:layout\_marginTop="65dp" android:layout\_toRightOf="@id/buttonon3" android:background="@drawable/tkbt" android:text="ON/OFF" android:textSize="10dp" />*

*<Button android:id="@+id/buttonup" android:layout\_width="40dp" android:layout\_height="60dp"*

*android:layout\_below="@+id/buttonon1" android:layout\_marginTop="19dp" android:layout\_toEndOf="@+id/buttonback" android:text="UP"*

*android:textSize="10dp" />*

*<Button android:id="@+id/buttondw" android:layout\_width="40dp" android:layout\_height="60dp"*

*android:layout\_alignStart="@+id/buttonup" android:layout\_below="@+id/buttonnext" android:text="DW" android:textSize="10dp" />*

*<Button android:id="@+id/buttonnext" android:layout\_width="60dp" android:layout\_height="40dp"*

*android:layout\_below="@+id/buttonup" android:layout\_toEndOf="@+id/buttondw" android:text="NEXT" android:textSize="10dp" />*

*<Button android:id="@+id/buttonback" android:layout\_width="60dp"*

*android:layout\_height="40dp" android:layout\_below="@+id/buttonup" android:layout\_toEndOf="@+id/buttonon1" android:text="BACK" android:textSize="10dp" />*

*<Button android:id="@+id/buttonmenu"*

*android:layout\_width="40dp" android:layout\_height="40dp" android:layout\_below="@+id/buttonon1" android:layout\_marginTop="195dp"*

*android:background="@drawable/tkbt" android:text="menu" android:layout\_marginLeft="70dp" android:textSize="8dp" />*

*<Button android:id="@+id/buttonp1" android:layout\_width="40dp" android:layout\_height="40dp"*

*android:layout\_toRightOf="@id/buttonmenu" android:layout\_marginTop="310dp" android:layout\_marginLeft="20dp" android:background="@drawable/tkbtdo" android:text="1"*

*android:textSize="10dp" />*

*<Button android:id="@+id/buttonp2" android:layout\_width="40dp" android:layout\_height="40dp"*

*android:layout\_toRightOf="@id/buttonp1" android:layout\_marginTop="310dp" android:layout\_marginLeft="20dp" android:background="@drawable/tkbtdo" android:text="2"*

*android:textSize="10dp" />*

*<Button android:id="@+id/buttonp3" android:layout\_width="40dp" android:layout\_height="40dp" android:layout\_marginLeft="20dp"*

*<Button android:id="@+id/buttonmute"*

*android:layout\_width="40dp" android:layout\_height="40dp" android:layout\_below="@+id/buttonmenu" android:layout\_marginTop="10dp" android:layout\_marginLeft="70dp" android:background="@drawable/tkbt" android:text="mute" android:textSize="8dp" />*

*<Button android:id="@+id/buttonp4" android:layout\_width="40dp" android:layout\_height="40dp"*

*android:layout\_toRightOf="@id/buttonmute" android:layout\_marginTop="360dp" android:layout\_marginLeft="20dp" android:background="@drawable/tkbtdo" android:text="4"*

*android:textSize="10dp" />*

*<Button android:id="@+id/buttonp5" android:layout\_width="40dp" android:layout\_height="40dp"*

*android:layout\_toRightOf="@id/buttonp4" android:layout\_marginTop="360dp" android:layout\_marginLeft="20dp" android:background="@drawable/tkbtdo" android:text="5"*

*android:textSize="10dp" />*

*<Button android:id="@+id/buttonp6" android:layout\_width="40dp" android:layout\_height="40dp" android:layout\_marginLeft="20dp"*

*<Button android:id="@+id/buttonexit"*

*android:layout\_width="40dp" android:layout\_height="40dp" android:layout\_below="@+id/buttonmute" android:layout\_marginTop="10dp" android:layout\_marginLeft="70dp" android:background="@drawable/tkbt" android:text="exit" android:textSize="8dp" />*

*<Button android:id="@+id/buttonp7" android:layout\_width="40dp" android:layout\_height="40dp"*

*android:layout\_toRightOf="@id/buttonexit" android:layout\_marginTop="410dp" android:layout\_marginLeft="20dp" android:background="@drawable/tkbtdo" android:text="7"*

*android:textSize="10dp" />*

*<Button android:id="@+id/buttonp8" android:layout\_width="40dp" android:layout\_height="40dp"*

*android:layout\_toRightOf="@id/buttonp7" android:layout\_marginTop="410dp" android:layout\_marginLeft="20dp" android:background="@drawable/tkbtdo" android:text="8"*

*android:textSize="10dp" />*

*<Button android:id="@+id/buttonp9" android:layout\_width="40dp" android:layout\_height="40dp" android:layout\_marginLeft="20dp"*

*android:layout\_marginTop="410dp" android:layout\_toRightOf="@id/buttonp8" android:background="@drawable/tkbtdo"*

*android:layout\_width="40dp" android:layout\_height="40dp" android:layout\_below="@+id/buttonexit" android:layout\_marginTop="10dp" android:layout\_marginLeft="70dp" android:background="@drawable/tkbt" android:text="mode" android:textSize="8dp" />*

*<Button android:id="@+id/buttonpsao" android:layout\_width="40dp" android:layout\_height="40dp"*

*android:layout\_toRightOf="@id/buttonmode" android:layout\_marginTop="460dp" android:layout\_marginLeft="20dp" android:background="@drawable/tkbtdo" android:text="\*"*

*android:textSize="10dp" />*

*<Button android:id="@+id/buttonp0" android:layout\_width="40dp" android:layout\_height="40dp"*

*android:layout\_toRightOf="@id/buttonpsao" android:layout\_marginTop="460dp" android:layout\_marginLeft="20dp" android:background="@drawable/tkbtdo" android:text="0"*

*android:textSize="10dp" /> android:textSize="10dp" />*

*<Button android:id="@+id/buttonpthang" android:layout\_width="40dp" android:layout\_height="40dp" android:layout\_marginLeft="20dp" android:layout\_marginTop="460dp"*

*android:layout\_toRightOf="@id/buttonp0" android:background="@drawable/tkbtdo" android:text="#"*

*</RelativeLayout>*

*//dieukhien.java////////////////////////////////////////////////////////////// package com.example.admin.myapp;*

*import android.content.Intent;*

*import android.support.v7.app.AppCompatActivity; import android.os.Bundle;*

*import android.view.MotionEvent; import android.view.View;*

*import android.widget.Button;*

*import android.widget.CompoundButton; import android.widget.Switch;*

*import android.widget.Toast;*

*import com.google.firebase.database.DatabaseReference; import com.google.firebase.database.FirebaseDatabase;*

*public class dieukhien extends AppCompatActivity {*

*Button bt1,bt2,bt3,bt4,bt5,bt6,bt7,bt8,bt9,bt10,bt11,bt12,bt13,bt14,bt15,bt16,bt17,bt18,bt19*

*,bt20,bt21,bt22,bt23,bt24,bthengio; Switch sw2;*

*FirebaseDatabase csdl;*

*DatabaseReference on1,on2,on3,on4,up,dw,next,back,b0,b1,b2,b3,b4,b5,b6,b7,b8,b9,bsao,bthang,mode*

*,menu,exit,mute,set; int ks=0,krs=0; @Override*

*protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_dieukhien); anhxa();*

*csdl = FirebaseDatabase.getInstance();*

*on1 = csdl.getReference("TT\_NUT\_ON\_1"); on2 = csdl.getReference("TT\_NUT\_ON\_2"); on3 = csdl.getReference("TT\_NUT\_ON\_3"); on4 = csdl.getReference("TT\_NUT\_ON\_4"); up = csdl.getReference("TT\_UP");*

*dw = csdl.getReference("TT\_DW");*

*back = csdl.getReference("TT\_BACK"); b0 = csdl.getReference("TT\_0");*

*b1 = csdl.getReference("TT\_1"); b2 = csdl.getReference("TT\_2"); b3 = csdl.getReference("TT\_3"); b4 = csdl.getReference("TT\_4"); b5 = csdl.getReference("TT\_5"); b6 = csdl.getReference("TT\_6"); b7 = csdl.getReference("TT\_7"); b8 = csdl.getReference("TT\_8"); b9 = csdl.getReference("TT\_9");*

*bsao = csdl.getReference("TT\_SAO"); bthang = csdl.getReference("TT\_THANG"); mode = csdl.getReference("TT\_MODE"); menu = csdl.getReference("TT\_MENU"); exit = csdl.getReference("TT\_EXIT");*

*mute = csdl.getReference("TT\_MUTE"); set = csdl.getReference("TT\_SET");*

*bt1.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*on1.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*on1.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt2.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*on2.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*on2.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt3.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*on3.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*on3.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt4.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*on4.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*on4.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt5.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*up.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*up.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt6.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*dw.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*dw.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt7.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*next.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*next.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt8.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*back.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*back.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt9.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*menu.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*menu.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt10.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*mute.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*mute.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt11.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*exit.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*exit.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt12.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*mode.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*mode.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt13.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*b0.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*b0.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt14.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*b1.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*b1.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt15.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*b2.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*b2.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt16.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*b3.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*b3.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt17.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*b4.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*b4.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt18.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*b5.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*b5.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt19.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*b6.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*b6.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt20.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*b7.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*b7.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt21.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*b8.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*b8.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt22.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*b9.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*b9.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt23.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*bsao.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*bsao.setValue("0");*

*}*

*return false;*

*}*

*});*

*bt24.setOnTouchListener(new View.OnTouchListener() { @Override*

*public boolean onTouch(View v, MotionEvent event) { if(event.getAction()==MotionEvent.ACTION\_DOWN)*

*{*

*bthang.setValue("1");*

*}*

*else if(event.getAction()==MotionEvent.ACTION\_UP)*

*{*

*bthang.setValue("0");*

*}*

*return false;*

*}*

*});*

*sw2.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {*

*@Override*

*public void onCheckedChanged(CompoundButton buttonView, boolean b) { if(b)*

*{*

*set.setValue("1");*

*Toast.makeText(dieukhien.this,"Đang học lệnh",Toast.LENGTH\_LONG).show();*

*}*

*else*

*{*

*set.setValue("0");*

*}*

*}*

*});*

*bthengio.setOnClickListener(new View.OnClickListener() { @Override*

*public void onClick(View v) {*

*});*

*}*

*Intent manhinh=new Intent(dieukhien.this,hengio.class); dieukhien.this.startActivity(manhinh);*

*}*

*public void anhxa()*

*{*

*bthengio=(Button) findViewById(R.id.Buttontimers) ;*

*sw2=(Switch) findViewById(R.id.switchs); bt1=(Button) findViewById(R.id.buttonon1); bt2=(Button) findViewById(R.id.buttonon2); bt3=(Button) findViewById(R.id.buttonon3); bt4=(Button) findViewById(R.id.buttonon4);*

*bt5=(Button) findViewById(R.id.buttonup); bt6=(Button) findViewById(R.id.buttondw); bt7=(Button) findViewById(R.id.buttonnext); bt8=(Button) findViewById(R.id.buttonback);*

*bt9=(Button) findViewById(R.id.buttonmenu); bt10=(Button) findViewById(R.id.buttonmute); bt11=(Button) findViewById(R.id.buttonexit); bt12=(Button) findViewById(R.id.buttonmode);*

*bt13=(Button) findViewById(R.id.buttonp0); bt14=(Button) findViewById(R.id.buttonp1); bt15=(Button) findViewById(R.id.buttonp2); bt16=(Button) findViewById(R.id.buttonp3);*

*bt17=(Button) findViewById(R.id.buttonp4); bt18=(Button) findViewById(R.id.buttonp5); bt19=(Button) findViewById(R.id.buttonp6); bt20=(Button) findViewById(R.id.buttonp7);*

*bt21=(Button) findViewById(R.id.buttonp8); bt22=(Button) findViewById(R.id.buttonp9); bt23=(Button) findViewById(R.id.buttonpsao); bt24=(Button) findViewById(R.id.buttonpthang);*

*}*

*}*

*//activity\_hengio.xml///////////////////////////////////////////////////////////*

*<?xml version="1.0" encoding="utf-8"?>*

*<RelativeLayout android:layout\_height="match\_parent" android:layout\_width="match\_parent" android:background="@drawable/nen" xmlns:android="*[*http://schemas.android.com/apk/res/android"*](http://schemas.android.com/apk/res/android)*>*

*<TextView android:id="@+id/textView"*

*android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_alignParentStart="true" android:layout\_alignParentTop="true" android:layout\_marginStart="43dp" android:layout\_marginTop="64dp" android:text="Hour" android:textColor="@color/red" android:textSize="30dp" android:textStyle="bold" />*

*<EditText android:id="@+id/editTexthour" android:layout\_width="200dp" android:layout\_height="wrap\_content"*

*android:layout\_alignBottom="@+id/textView" android:layout\_marginStart="12dp" android:layout\_toEndOf="@+id/textView3" android:ems="10"*

*android:hint="Write here. "*

*android:inputType="number" android:textColor="#0e0e0e" android:textSize="20dp" android:textStyle="bold" />*

*<TextView android:id="@+id/textView2" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content"*

*android:layout\_alignStart="@+id/textView" android:layout\_below="@+id/textView"*

*android:layout\_marginTop="47dp" android:text="Minute" android:textColor="@color/red" android:textSize="30dp" android:textStyle="bold" />*

*<EditText android:id="@+id/editTextminute" android:layout\_width="200dp" android:layout\_height="wrap\_content"*

*android:layout\_alignBottom="@+id/textView2" android:layout\_alignStart="@+id/editTexthour" android:ems="10"*

*android:hint="Write here" android:inputType="number" android:textColor="#151414" android:textSize="20dp" android:textStyle="bold" />*

*<TextView android:id="@+id/textView3" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content"*

*android:layout\_alignStart="@+id/textView2" android:layout\_below="@+id/textView2" android:layout\_marginTop="47dp" android:textSize="30dp" android:textColor="@color/blue" android:textStyle="bold" android:text="Button 1" />*

*<EditText android:id="@+id/editTextlenh1" android:layout\_width="200dp" android:layout\_height="wrap\_content"*

*android:layout\_alignBottom="@+id/textView3" android:layout\_alignStart="@+id/editTextminute" android:ems="10"*

*android:hint="Write here. "*

*android:inputType="textPersonName" android:textColor="#0e0e0e" android:textSize="20dp" android:textStyle="bold" />*

*<TextView*

*android:id="@+id/textView4" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_alignStart="@+id/textView3" android:layout\_below="@+id/textView3" android:layout\_marginTop="44dp" android:text="Button 2" android:textColor="#2e0dea" android:textSize="30dp" android:textStyle="bold" />*

*<EditText android:id="@+id/editTextlenh2" android:layout\_width="200dp" android:layout\_height="wrap\_content"*

*android:layout\_alignBottom="@+id/textView4" android:layout\_alignStart="@+id/editTextlenh1" android:ems="10"*

*android:hint="Write here. "*

*android:inputType="textPersonName" android:textColor="#0e0e0e" android:textSize="20dp" android:textStyle="bold" />*

*<Button android:id="@+id/buttonEnter" android:layout\_width="wrap\_content"*

*android:layout\_height="wrap\_content" android:layout\_below="@+id/textView4" android:layout\_marginTop="82dp" android:layout\_toEndOf="@+id/textView4" android:text="ENTER" />*

*</RelativeLayout>*

*//hengio.java////////////////////////////////////////////////////*

*package com.example.admin.myapp;*

*import android.support.v7.app.AppCompatActivity; import android.os.Bundle;*

*import android.view.View;*

*import android.widget.Button; import android.widget.EditText; import android.widget.Toast;*

*import com.google.firebase.database.DatabaseReference; import com.google.firebase.database.FirebaseDatabase;*

*public class hengio extends AppCompatActivity { EditText edtgio,edtphut,edtbt1,edtbt2;*

*Button btnEnter; FirebaseDatabase csdl2;*

*DatabaseReference gio,phut,lenh1,lenh2; @Override*

*protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_hengio);*

*csdl2 = FirebaseDatabase.getInstance(); edtbt1=(EditText) findViewById(R.id.editTextlenh1); edtbt2=(EditText) findViewById(R.id.editTextlenh2); edtgio=(EditText) findViewById(R.id.editTexthour); edtphut=(EditText) findViewById(R.id.editTextminute); btnEnter=(Button) findViewById(R.id.buttonEnter);*

*gio = csdl2.getReference("gio"); phut = csdl2.getReference("phut");*

*lenh1 = csdl2.getReference("nutnhan1"); lenh2 = csdl2.getReference("nutnhan2");*

*btnEnter.setOnClickListener(new View.OnClickListener() { @Override*

*public void onClick(View v) {*

*String hg= edtgio.getText().toString().trim(); String hp= edtphut.getText().toString().trim();*

*String b1= edtbt1.getText().toString().trim(); String b2= edtbt2.getText().toString().trim();*

*//Toast.makeText(hengio.this,"Please,set hour",Toast.LENGTH\_SHORT).show();*

*if(hg.length()==0)*

*{*

*Toast.makeText(hengio.this,"Please,set hour",Toast.LENGTH\_SHORT).show();*

*edtgio.requestFocus();*

*}*

*else if(hp.length()==0)*

*{*

*Toast.makeText(hengio.this,"Please,set*

*minute",Toast.LENGTH\_SHORT).show();*

*edtphut.requestFocus();*

*}*

*else if(b1.length()==0 && b2.length()==0)*

*{*

*Toast.makeText(hengio.this,"Please,set button",Toast.LENGTH\_SHORT).show();*

*edtbt1.requestFocus();*

*}*

*else if(b1.length()!=0 && b2.length()==0)*

*{*

*int gg=Integer.parseInt(hg); int pp=Integer.parseInt(hp); if (gg>=24)*

*{*

*Toast.makeText(hengio.this,"Please,Hour value <24 ",Toast.LENGTH\_SHORT).show();*

*edtgio.setText("");*

*edtgio.requestFocus();*

*}*

*else if (pp>=60)*

*{*

*edtphut.setText(""); Toast.makeText(hengio.this,"Please, Minute value*

*<60",Toast.LENGTH\_SHORT).show();*

*edtphut.requestFocus();*

*}*

*else {*

*gio.setValue(hg); phut.setValue(hp); lenh1.setValue(b1); lenh2.setValue(" ");*

*Toast.makeText(hengio.this, "Complete for 1 command", Toast.LENGTH\_SHORT).show();*

*}*

*}*

*else if(b1.length()==0 && b2.length()!=0)*

*{*

*gio.setValue(hg); phut.setValue(hp);*

*lenh1.setValue(" "); lenh2.setValue(b2);*

*Toast.makeText(hengio.this,"Complete for 1 command",Toast.LENGTH\_SHORT).show();*

*}*

*else*

*{*

*gio.setValue(hg); phut.setValue(hp); lenh1.setValue(b1); lenh2.setValue(b2);*

*Toast.makeText(hengio.this,"Complete for 2 command",Toast.LENGTH\_SHORT).show();*

*}*

*}*

*});*

*}*

*}*