**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HCM**

**KHOA ĐIỆN - ĐIỆN TỬ**

**BỘ MÔN KỸ THUẬT MÁY TÍNH - TRUYỀN THÔNG**

**ĐỒ ÁN TỐT NGHIỆP**

**THIẾT KẾ Ổ ĐIỆN THÔNG MINH**

NGÀNH CÔNG NGHỆ KỸ THUẬT ĐIỆN TỬ TRUYỀN THÔNG

Sinh viên: **VÕ HOÀNG NGUYÊN**

MSSV: 12141151

**HUỲNH NGỌC THƯƠNG**

MSSV:12141227

**TP. HỒ CHÍ MINH** – **01/2017**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HCM**

**KHOA ĐIỆN - ĐIỆN TỬ**

**BỘ MÔN KỸ THUẬT MÁY TÍNH - TRUYỀN THÔNG**

**ĐỒ ÁN TỐT NGHIỆP**

**THIẾT KẾ Ổ ĐIỆN THÔNG MINH**

NGÀNH CÔNG NGHỆ KỸ THUẬT ĐIỆN TỬ TRUYỀN THÔNG

Sinh viên: **VÕ HOÀNG NGUYÊN**

MSSV: 12141151

**HUỲNH NGỌC THƯƠNG**

MSSV:12141227

Hướng dẫn: **Th.S HUỲNH HOÀNG HÀ**

**TP. HỒ CHÍ MINH** – **01/2017**

**LỜI CẢM ƠN**

*Lời đầu tiên, chúng tôi muốn gửi lời cảm ơn chân thành đến giáo viên, Th. Huỳnh Hoàng Hà, trong thời gian qua đã hướng dẫn và giúp đỡ chúng tôi trong quá trình thực hiện luận văn tốt nghiệp. Những lời nhận xét, góp ý và hướng dẫn của thầy đã giúp chúng tôi có một định hướng đúng trong quá trình thực hiện đề tài, giúp chúng tôi thấy được những ưu, khuyết điểm của đề tài và từng bước khắc phục để ngày một tốt hơn.*

*Đồng thời chúng tôi cũng muốn gửi lời cảm ơn sâu sắc đến gia đình và bạn bè đã động viên, cổ vũ tinh thần trong suốt quá trình học tập và luận án tốt nghiệp.*

*Bên cạnh đó, chúng tôi muốn nói lời cảm ơn đến những giáo viên, giảng viên đã đứng trên bục giảng truyền đạt kinh nghiệm, kiến thức trong suốt những năm học đã qua.*

*Một lần nữa, chúng tôi xin gửi lời cảm ơn đến bố mẹ, thầy cô và bạn bè, những người luôn ở cạnh, giúp đỡ và cho phép chúng tôi có thời gian để hoàn thành luận án này.*

Tác giả luận văn

***Võ Hoàng Nguyên***

***Huỳnh Ngọc Thương***

# 

# TÓM TẮT

Quyển luận án này gồm 5 chương, thể hiện đầy đủ quá trình nghiên cứu và thi công “Thiết kế ổ điện thông minh”. Với tính thương mại được tính đến, chúng tôi đã thiết kế một hệ thống hoàn chỉnh và hoàn toàn có thể phát triển trên thực tế.

Ổ cắm điện thông minh là phát minh tuyệt vời trong thời đại ngày nay. Từ ý tưởng ban đầu như kéo dài ổ cắm bằng các ổ cắm di động, tăng thêm số đầu ổ cắm, lắp thêm đèn báo điện áp trên ổ cắm, làm ổ cắm có cầu chì, làm ổ cắm có áp tô mát, làm ổ cắm đa năng…đến nay ổ cắm điện được bổ sung những công năng và hoàn thiện đến mức người ta phải coi ổ cắm điện là một khí cụ điện tinh xảo và đặt tên cho nó là “ Ổ cắm điện thông minh”. Nếu nhìn từ bên ngoài ổ cắm điện thông minh không khác ổ cắm điện thông thường là mấy, đa dạng giống như các ổ cắm điện bình thường nhưng tiện ích và khả năng làm việc của nó tiện lợi hơn nhiều, tự động đóng cắt được thiết bị điện, khả năng điều khiển từ xa dùng hệ không dây, đặc biệt giảm được khá nhiều tiền điện trong quá trình xử dụng.

**MỤC LỤC**

[DANH MỤC HÌNH viii](#_Toc470208487)

[DANH MỤC BẢNG x](#_Toc470208488)

[**CÁC TỪ VIẾT TẮT** xi](#_Toc470208489)

[CHƯƠNG 1 GIỚI THIỆU 1](#_Toc470208490)

[1.1 ĐẶT VẤN ĐỀ 1](#_Toc470208492)

[1.2 TỔNG QUAN TÌNH HÌNH NGHIÊN CỨU 2](#_Toc470208495)

[1.3 MỤC TIÊU VÀ NỘI DUNG CỦA ĐỀ TÀI 7](#_Toc470208497)

[*1.3.1 Mục tiêu của đề tài* 7](#_Toc470208498)

[*1.3.2 Nội dung của đề tài* 8](#_Toc470208499)

[1.4 ĐỐI TƯỢNG VÀ PHẠM VI ĐỀ TÀI 8](#_Toc470208500)

[*1.4.1 Đối tượng của đề tài* 8](#_Toc470208501)

[*1.4.2 Phạm vi nghiên cứu* 8](#_Toc470208502)

[1.5 BỐ CỤC LUẬN VĂN 9](#_Toc470208503)

[CHƯƠNG 2 CƠ SỞ LÝ THUYẾT 11](#_Toc470208504)

[2.1 GIỚI THIỆU VỀ WI-FI 11](#_Toc470208506)

2.2 ARDUINO PRO MINI 12

[*2.2.1 Giới thiệu* 12](#_Toc470208507)

[*2.2.2 Phần cứng của Arduino Pro Mini* 13](#_Toc470208508)

[*2.2.3 Môi trường lập trình trên Arduino*. 15](#_Toc470208509)

[2.3 MẠCH THU PHÁT WIFI ESP8266 16](#_Toc470208510)

[*2.3.1 Giới thiệu* 16](#_Toc470208511)

[*2.3.2 Thông số* 17](#_Toc470208513)

[*2.3.3 Sơ đồ chân* 18](#_Toc470208514)

[2.4 MẠCH ĐỌC THẺ NFC CR95HF 19](#_Toc470208515)

[*2.4.1 Giới thiệu* 19](#_Toc470208516)

[*2.4.2 Thông số* 20](#_Toc470208517)

[*2.4.3 Nguyên lý hoạt động mạch NFC CR95HF* 21](#_Toc470208518)

[2.5 CẢM BIẾN DÒNG ĐIỆN HALL ACS712 21](#_Toc470208519)

[*2.5.1 Giới thiệu* 21](#_Toc470208520)

[*2.5.2 Thông số* 22](#_Toc470208521)

[*2.5.3 Nguyên lý hoạt động* 23](#_Toc470208522)

[2.6 PHẦN MỀM ANDROID STUDIO 24](#_Toc470208524)

[*2.6.1 Giới thiệu* 24](#_Toc470208527)

[*2.6.2 Các bước để viết một ứng dụng* 25](#_Toc470208528)

[2.7 LẬP TRÌNH PHP VÀ MYSQL 28](#_Toc470208531)

[*2.7.1 Giới thiệu ngôn ngữ Php* 28](#_Toc470208532)

[*2.7.2 Cú pháp của Php* 29](#_Toc470208533)

[*2.7.3 Giới thiệu JSON* 29](#_Toc470208533)0

[*2.7.4 Giới thiệu MySQL* 30](#_Toc470208534)

[*2.7.5 Ưu điểm của MYSQL* 31](#_Toc470208535)

[*2.7.6 Giới thiệu HTTP* 32](#_Toc470208536)

[*2.7.7 Giới thiệu HTML* 32](#_Toc470208536)

CHƯƠNG 3 [THIẾT KẾ HỆ THỐNG 37](#_Toc470208537)

[3.1 SƠ ĐỒ KHỐI HỆ THỐNG 37](#_Toc470208538)

[3.2 KHỐI PHẦN CỨNG 38](#_Toc470208539)

[*3.2.1 Kit Arduino Pro Mini* 38](#_Toc470208540)

[*3.2.2 Cảm biến dòng điện Hall ACS712* 38](#_Toc470208541)

[*3.2.3 Cảm biến đọc thẻ NFC CR95HF* 40](#_Toc470208542)

[*3.2.4 Mạch thu phát Wifi ESP8266* 43](#_Toc470208543)

[*3.2.5 Giải thuật điều khiển phần cứng* 45](#_Toc470208544)

[3.3 THIẾT KẾ ỨNG DỤNG ANDROID 46](#_Toc470208546)

[*3.3.1 Phân tích yêu cầu đặt ra* 48](#_Toc470208547)

[*3.3.2 Xây dựng lưu đồ giải thuật* 48](#_Toc470208549)

[3.4 THIẾT KẾ WEB SEVER 52](#_Toc470208550)

[*3.4.1 Không gian lưu trữ và tên miền của Web server* 52](#_Toc470208551)

[*3.4.2 Thiết kế cơ sở dữ liệu MySQL* 53](#_Toc470208552)

[CHƯƠNG 4 KẾT QUẢ VÀ ĐÁNH GIÁ 56](#_Toc470208553)

[4.1 KẾT QUẢ PHẦN CỨNG 56](#_Toc470208555)

[4.2 KẾT QUẢ PHẦN MỀM 58](#_Toc470208556)

[*4.2.1 Ứng dụng ổ diện thông minh* 58](#_Toc470208557)

[*4.2.2 Web Application* 61](#_Toc470208558)

[CHƯƠNG 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 62](#_Toc470208559)

[5.1 KẾT LUẬN 62](#_Toc470208561)

[*5.1.1 Những vấn đề nghiên cứu* 62](#_Toc470208562)

[*5.1.3 Những hạn chế của đề tài* 62](#_Toc470208563)

[5.2 HƯỚNG PHÁT TRIỂN CỦA ĐỀ TÀI 62](#_Toc470208564)

[PHỤ LỤC A MÃ NGUỒN CHƯƠNG TRÌNH 64](#_Toc470208565)

[TÀI LIỆU THAM KHẢO 74](#_Toc470208566)

DANH MỤC HÌNH

**Hình 1.1** Công nghệ Internet of Thing 2

**Hình 1.2** Một số hình ảnh của ổ cắm 3

**Hình 1.3** Một số hình ảnh của sản phẩm 4

**Hình 1.4** Một số hình ảnh sản phẩm liên quan 4

**Hình 1.5** Một số hình ảnh sản phẩm liên quan 5

**Hình 1.6** Một số hình ảnh sản phẩm liên quan 6

**Hình 1.7** Một số hình ảnh sản phẩm liên quan 6

**Hình 1.8** Một số hình ảnh sản phẩm liên quan 7

**Hình 2.1** Board Arduino Pro Mini 13

**Hình 2.2** Sơ đồ chân của Arduino Pro Mini 15

**Hình 2.3** Giao diện phần mềm IDE 16

**Hình 2.4** Mạch thu phát wifi ESP8266 17

**Hình 2.5** Sơ đồ chân của ESP8266 18

**Hình 2.6** Mạch NFC CR95HF 20

**Hình 2.7** Cảm biến dòng điện Hall ACS712 22

**Hình 2.8** Sơ đồ nguyên lý ACS712 23

**Hình 2.9** Phần mềm Android Studio 24

**Hình 2.10** Các bước tạo ứng dụng 25

**Hình 2.11** Tạo SDK cho ứng dụng 26

**Hình 2.12** Quá trình hoạt động của App Android 27

**Hình 2.13** Một đoạn code Php 30

**Hình 2.14** Nguyên lý hoạt động của Php 26

**Hình 3.1** Sơ đồ khối hệ thống 37

**Hình 3.2** Sơ đồ khối cảm biến kết nối với kit 39

**Hình 3.3** Điện áp đầu ra so với dòng điện đầu vào 39

**Hình 3.4** Ảnh hưởng nhiệt độ đến độ ẩm của cảm biến 40

**Hình 3.5** Kết nối giữa ESP8266 với Arduino 43

**Hình 3.6** Chip ESP8266 44

**Hình 3.7** Lưu đồ hoạt động của Arduino 45

**Hình 3.8** Lưu đồ hoạt động của ESP 46

**Hình 3.9** Lưu đồ hoạt động của NFC khi kiểm tra thẻ 47

**Hình 3.10** Lưu đồ quá trình đăng nhập ứng dụng 48

**Hình 3.11** Lưu đồ hiển thị công suất từng ổ cắm 49

**Hình 3.12** Lưu đồ hiện thị tình trạng lỗ cắm 50

**Hình 3.13** Lưu đồ hẹn giờ đóng mở từng lỗ cắm 51

**Hình 3.14** Lưu đồ thay đổi password 52

**Hình 3.15** Lưu đồ mối quan hệ giữa Sever và hệ thống 54

**Hình 4.1** Mô hình ổ điện thông minh 57

**Hình 4.2** Màn hình giao diện Login và Register 58

**Hình 4.3** Màn hình giao diện Main Activity 59

**Hình 4.4** Màn hình giao diện đọc công suất và tên thiết bị 60

**Hình 4.5** Màn hình hẹn giờ và Change password 60

**Hình 4.6** Giao diện Web Application 61

# LIỆT KÊ BẢNG

**Bảng 2.1** Các thành phần của board Arduino Pro Mini 13

**Bảng 3.1** Bảng mô tả chân của module RFID-NFC CR95HF 41

**Bảng 3.2** Bảng danh sách các lệnh của CR95HF 42

**Bảng 3.3** Cách kết nối ESP8266 với Arduino 43

**Bảng 3.4** Cấu trúc dữ liệu của bảng users 53

**Bảng 3.5** Cấu trúc dữ liệu của bảng người dùng 53

CÁC TỪ VIẾT TẮT

ADC: Analog Digital Converter

CSDL: Cơ Sở Dữ Liệu

DSSS: Direct-Sequence Spread Spectrum

PHP: Hypertext Preprocessor

FHSS: Frequency Hopping Spread Spectrum

HTTP: Hypertext Transfer Protocol

IDE: Integrated Development Environment

MCU: Microcontroller

OFDM: Orthogonal Frequency-Division Multiplexing

PCB: Printed Circuit Board

SSID: Service Set Identifier

UART: Universal Asynchronous Receiver/Transmitter

WPA: Wireless Protected Area

# 

# Chương 1

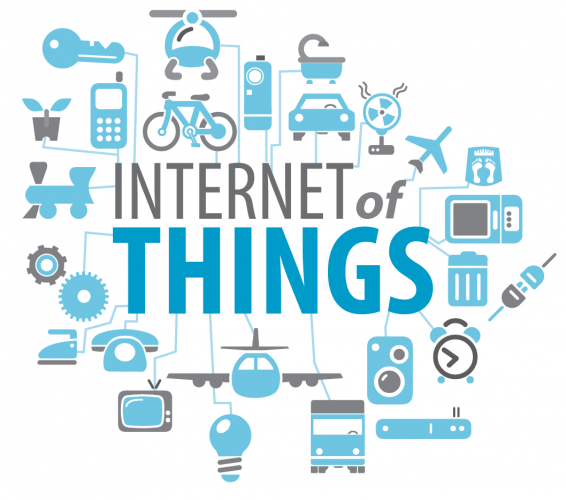
# GIỚI THIỆU

## ĐẶT VẤN ĐỀ

## Ngày nay, bên cạnh sự tiến bộ vượt bậc của kinh tế, khoa học kỹ thuật đã giúp cho đời sống con người được nâng cao. Chưa bao giờ trong lịch sử chúng ta sống trong môi trường tiện nghi như hiện nay. Nhu cầu sở hữu các thiết bị thông minh ở trong chính ngôi nhà của chúng ta[1].

## Các thiết bị thông minh được sử dụng trong ngôi nhà dường ngày cần tăng dần và dường như chúng ta phải chi trả tiền điện nhiều hơn trước. Trong khi chúng ta không cần sử dụng liên tục nhưng vẫn phải cung cấp diện. Mà hầu hết các đồ dùng điện đều sử dùng điện ở từ ổ cắm truyền thống nên dễ gây ra cháy nổ. Vì vậy để dễ dàng điều khiển và giám sát các thiết bị sử dụng điện một cách tiện lợi và hiệu quả nhất thì Ổ cắm điện thông minh ra đời. Nhằm thay thế các ổ cắm thuần túy đang sử dụng để giúp con người tiết kiệm thời gian và ổn định được độ chính xác trong điều khiển thiết bị. Thị trường hiện nay đã có nhiều loại ổ điện thông minh nhưng chưa đủ hết các tính năng đầy đủ và chưa đột phá nên chúng tôi lựa chọn đề tài này.

Các hệ thống hiện đại này có khả năng liên kết không dây giữa các thành phần với nhau theo một xu hướng mới là ***Internet of Things.*** *Internet of Things* (IoT) là một xu hướng công nghệ mới mẻ, được nghiên cứu và phát triển rầm rộ trong thời gian gần đây do sự phát triển nhanh chóng của mạng Internet cũng như công nghệ tích hợp, kết nối không dây…[2]. Với sự giúp đỡ của IoT, chúng ta có thể kết nối bất cứ điều gì, truy cập từ bất cứ nơi nào và bất cứ lúc nào, có khả năng truy cập vào bất kỳ dịch vụ và thông tin về đối tượng nào. Hàng loạt những triển lãm công nghệ, hội thảo khoa học được tổ chức xoay quanh vấn đề quảng bá sản phẩm công nghệ IoT, hệ sinh thái khởi nghiệp IoT… Đây hứa hẹn sẽ là một cuộc cách mạng mới trong tương lai, đem đếm sự thay đổi mạnh mẽ trong nhiều lĩnh vực của cuộc sống.



**Hình 1.1** Công nghệ Internet of Things

* 1. **TỔNG QUAN TÌNH HÌNH NGHIÊN CỨU**

Trong bối cảnh tình hình nghiên cứu khoa học và ứng dụng khoa học công nghệ vào cuộc sống tại nước ta còn chậm khá nhiều so với các nước trong khu vực và trên thế giới. Rất nhiều các thiết bị đều được nhập về từ nước ngoài. Các nghiên cứu hệ thống thông minh, mô hình viễn thông rất hiếm gặp tại nước ta hiện nay. Tuy nhiên, trình trạng này đang được cải thiện dần, tại một số trường đại học và phòng nghiên cứu, các thiết bị thông minh trong gia đình và đang nhận được một cái nhìn chủ động hơn trong nghiên cứu và ứng dụng [3].

Ổ cắm điện thông minh là phát minh tuyệt vời trong thời đại ngày nay. Từ ý tưởng ban đầu như kéo dài ổ cắm bằng các ổ cắm di động, tăng thêm số đầu ổ cắm, lắp thêm đèn báo điện áp trên ổ cắm, làm ổ cắm có cầu chì, làm ổ cắm có áp tô mát, làm ổ cắm đa năng…đến nay ổ cắm điện được bổ sung những công năng và hoàn thiện đến mức người ta phải coi ổ cắm điện là một khí cụ điện tinh xảo và đặt tên cho nó là “ Ổ cắm điện thông minh”.Nếu nhìn từ bên ngoài ổ cắm điện thông minh không khác ổ cắm điện thông thường là mấy, đa dạng giống như các ổ cắm điện bình thường nhưng tiện ích và khả năng làm việc của nó tiện lợi hơn nhiều, tự động đóng cắt được thiết bị điện, khả năng điều khiển từ xa dùng hệ không dây, đặc biệt giảm được khá nhiều tiền điện trong quá trình xử dụng.

Các đề tài Ổ điện thông minh điều khiển sử dùng bluetooh hay hồng ngoại, đặt trước thời gian đã xuất hiện trên thị trường thương mại nhưng no chưa đầy đử các chức năng như :

*Ổ cắm điều khiển từ xa KONO KN-SK43***:** sản phẩm ổ điện thông minh của hãng KONO. Ứng dụng điều khiển từ xa thiết bị trong nhà như đèn, quạt, modum wifi, đèn quảng cáo, điều khiển xuyên tường, khoảng cách điều khiển từ 30~50m.



**Hình 1.2** Một số hình ảnh ổ cắm

*Ổ cắm và công tắc thông minh Handy Switch*: Giúp bạn tắt hoặc bật mọi bóng đèn từ các vị trí bạn muốn trong nhà. Đơn giản chỉ cần cắm điện đèn ngủ, đèn ban công, đèn vườn… hoặc các thiết bị khác thông qua ổ cắm Handy Switch sau đó dùng công tắc không dây đi kèm để tắt bật đèn và thiết bị từ mọi nơi bạn muốn.



**Hình 1.3** Một số hình ảnh của sản phầm

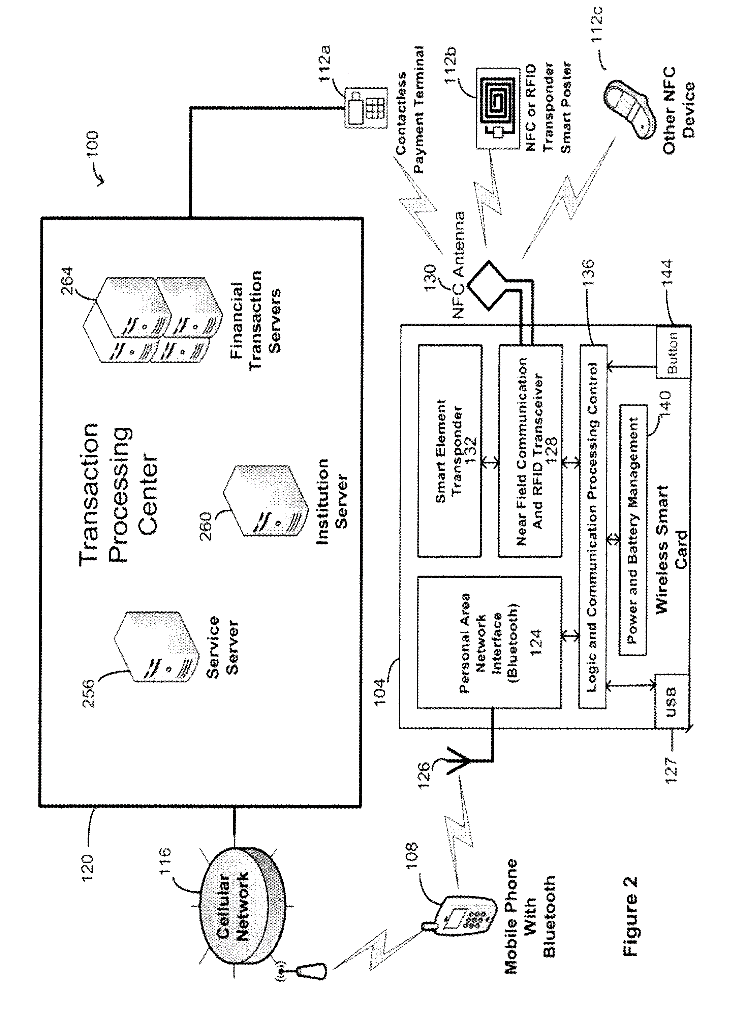
### [*Ổ cắm điện hẹn giờ KW-TG9*](https://thietbidienthongminh.wordpress.com/2010/08/25/%E1%BB%95-c%E1%BA%AFm-di%E1%BB%87n-h%E1%BA%B9n-gi%E1%BB%9D-kw-tg9/): Các thiết bị điện trong nhà sẽ tự động được tắt mở với thời gian bạn định trước, chỉ cần cài đặt 1 lần cho timer và vào ngày nào thiết bị của bạn sẽ được hẹn giờ tắt mở như lịch đã hẹn các thiết bị, và Timer sẽ làm việc đóng mở đúng thời gian bạn cài đặt, cụ thể bạn chỉ  cần gắn dây nguồn thiết bị vào Timer rồi gắn Timer vào điện lưới 220VAC nó sẽ giúp bạn canh thời gian và tắt mở thiết bị điện của bạn một cách hiệu quả.



**Hình 1.4** Một số hình ảnh của sản phẩm liên quan

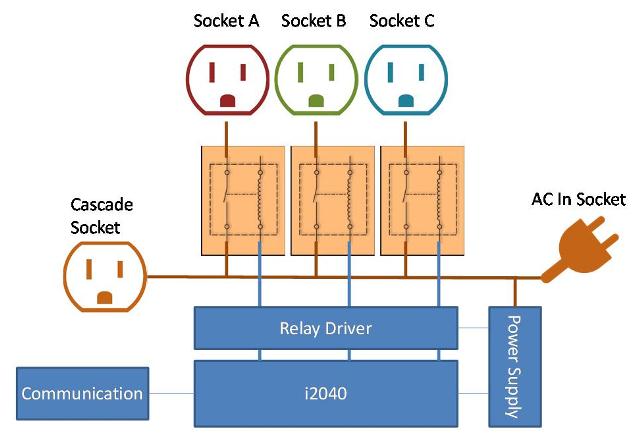
Những đề tài ngoài nước trên IEEEXPLORE.IEEE.ORG

* *A Wireless Power Outlet System for Smart Homes* của Guangming Song, Member, IEEE, Fei Ding, Weijuan Zhang and Aiguo Song, Member, IEEE



**Hình 1.5** Một số hình ảnh của sản phẩm liên quan

* *Smart Power Outlets with Cross-layer Communication* của Yukio Hira naka, Yusuke Sato, Toshihiro Taketa and Shinichi MiuraDepartment of Informatics,Yamagata University, Yonezawa, Japan Corresponding Author:[zioi@ieee.org](mailto:zioi@ieee.org)

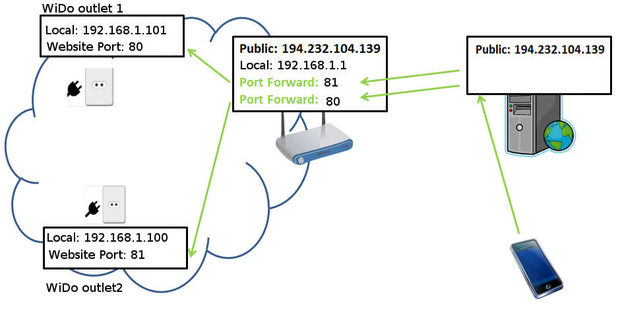


**Hình 1.6** Một số hình ảnh của sản phẩm liên quan

* *Dumb to Smart Power Outlets:* A Design Perspective on Smart Grids.
* World Wide WiDo Web power outlet ! ! The ultimate Arduino IoT DIY



**Hình 1.7** Một số hình ảnh của sản phẩm liên quan



**Hình 1.8** Một số hình ảnh của sản phẩm liên quan

Nhìn chung các đề tài trên đều là những sản phẩm của các hãng sản xuất và ứng dụng ngoài thị trường, nhưng mỗi sản phẩm chỉ làm một chức năng trọng tâm như: điều khiển từ xa, quản lý công suất…Tiếp nối những ứng dụng của những đề tài đã có, đề tài này em sẽ cải thiệt hiệu quả việc điều khiển từ xa và giám sát được ổ cắm có hay không thiết bị đang được sử dụng, qua đó cũng ứng dụng thêm công nghệ mới NFC.

## MỤC TIÊU VÀ NỘI DUNG CỦA ĐỀ TÀI

### 1.3.1 Mục tiêu của đề tài

***\*Mục tiêu bắt buộc hoàn thành:***

* **Có thể bật tắt nguồn điện từ bất cứ thiết bị nào và ở đâu trong nhà mình thông qua kết nối hệ thống mạng có sẵn trong nhà và ổ điện, điều khiển bằng App Android.**
* **Chức năng hẹn giờ để tự động đóng ngắt hoặc định sẵn thời gian hoạt động cho các đồ dùng điện trong gia đình.**
* **Có thể đo được công suất của thiết bị sử dụng, thông qua đó có thể nắm được điện năng tiêu thụ để điều chỉnh sinh hoạt cho hợp lý, hay có thể quản lý được hiện tượng quá tải của thiết bị nhằm bảo vệ thiết bị.**
* **Có thể nhận biết được loại thiết bị đang sử dụng qua đó có thể cài chế độ tự động hẹn giờ cho từng thiết bị thông qua hệ thống NFC, giúp người dùng hẹn giờ tự động cho thiết bị một cách tiện lợi nhất mà không cần dùng APP Android.**
* **Có cổng USB dùng cho xạc điện thoại.**

***\*Mục tiêu mở rộng nếu có thể:***

* **Tiến hành thương mại hóa sản phẩm.**
* **Cho phép đăng ký và quản lý người dùng trên giao diện chuyên nghiệp.**

### ****1.3.2 Nội dung của đề tài****

Cụ thể, các thông số được đo từ ổ điện thông minh liên tục cập nhật cảm biến lên server bằng kết nối Wifi, sau đó dữ liệu thô sẽ được xử lý và lưu trữ tại Web server. Android application do người dùng quản lý có khả năng truy cập giám sát dược các thiết bị. Từ các số liệu người dùng có thể dóng cắt nguồn điện trên từng lổ cắm trên ổ điện. Cách hiển thị trực quan này giúp người sử dụng dễ dàng nhận thấy sự thay đổi xem có đồ dùng nào bị hư hại hoặc hỏng trong thời gian dài, giúp họ có những điều chỉnh phù hợp tiết kiệm điện một cách tốt nhất.

Đề tài hướng đến thiết kế ổ điện có thể kết nối với Internet được giám sát với với điện thoại thông minh. Phần mềm trên hệ điều hành Android có tính năng xem được công suất và đưa ra cảnh báo. Bên cạnh đó công nghệ RFID NFC cũng là tính mới của đề tài.

## ĐỐI TƯỢNG VÀ PHẠM VI ĐỀ TÀI

### 1.4.1 Đối tượng của đề tài

* Kít Arduino Pro Mini.
* Mạch thu phát wifi ESP8266.
* Modem Wifi.
* Mạch NFC CR96HF.
* Cảm biến đòng điện ACS712.
* Phần mềm lập trình MySQL, Android studio, Php.
* …

### 1.4.2 Phạm vi nghiên cứu

* Nghiên cứu trong lĩnh vực lập trình nhúng trên Arduino Pro Mini.
* Nghiên cứu phần mềm hỗ trợ điều khiển trên Arduino Pro Mini và ESP8266 trên Arduino IDE.
* Cách kết nối vật lý giữa cảm biến, đọc thẻ với Arduino Pro Mini.
* Tìm hiểu nguyên lý hoạt động của các con cảm biến cần sử dụng.
* Nghiên cứu cách lập trình ứng dụng điều khiển Android.
* Tạo cơ sở dữ liệu MySQL trên internet.
* Lập trình Php.

## 1.5 BỐ CỤC LUẬN VĂN

Nội dung chính của đề tài được trình bày với năm chương như sau:

*Chương 1 Giới thiệu*

Trong chương này nêu ra được tình hình nghiên cứu hiện nay, lý do và mục tiêu chọn đề tài, đối tượng và phạm vi nghiên cứu của đề tài, phương pháp nghiên cứu và giới hạn của đề tài.

*Chương 2: Cơ sở lý thuyết*

Các lý thuyết chính liên quan đến các thành phần phần cứng và phần mềm cả hệ thống được trình bày như sau:

* Kít Arduino Pro Mini.
* Kít ESP8266.
* Các module cảm biến sử dụng như: ACS712, RFID NFC.
* Lập trình Android, Php và MySQL.
* …

*Chương 3: Phân tích và thiết kế hệ thống.*

* Trong chương này mục đích là thiết kế phần cứng cho hệ thống với những yêu cầu đặt ra với hệ thống. Từ sơ đồ khối tổng quát và sơ đồ khối chi tiết để tiến hành lựa chọn các linh kiện cho các khối.
* Xây dựng chương trình trên ESP để thực hiện quá trình truyền dữ liệu của các cảm biến về Sever thông qua wifi.
* Xây dựng cở sở dữ liệu và các giao thức liên lạc.

*Chương 4: Kết quả và đánh giá*

Trình bày về kết quả thi công phần cứng, phần mềm và phân thích. Đánh giá đề tài và đưa ra những hạn chế.

*Chương 5: Kết luận và hướng phát triển*

Đưa ra các kết luận về những vấn đề mà trong quá trình nghiên cứu đã đạt được và chưa đạt được. Đưa ra hướng phát triển đề tài trong tương lai.

# Chương 2

# CƠ SỞ LÝ THUYẾT

## 2.1 GIỚI THIỆU VỀ WI-FI

Wi-Fi là một công nghệ cho phép các thiết bị điện tử để kết nối với một mạng LAN không dây (WLAN), chủ yếu là sử dụng băng tần 2,4GHz và 5GHz. Một WLAN thường có mật khẩu bảo vệ, cho phép bất kỳ thiết bị trong phạm vi của nó có thể truy cập vào các nguồn tài nguyên của mạng WLAN.

Thiết bị có thể sử dụng công nghệ Wi-Fi bao gồm máy tính cá nhân, máy chơi game, điện thoại thông minh, máy ảnh kỹ thuật số, máy tính bảng và máy nghe nhạc kỹ thuật số. Thiết bị tương thích Wi-Fi có thể kết nối với Internet thông qua một mạng WLAN và một điểm truy cập không dây. Một điểm truy cập như vậy (hotpost ) có phạm vi khoảng 20 mét trong nhà và lớn hơn ở ngoài trời.

Kết nối Wi-Fi kém an toàn hơn so với kết nối có dây như Ethernet vì không cần đến một kết nối vật lý. Các trang web có sử dụng TLS thì an toàn, nhưng nếu truy cập Internet không được mã hóa có thể dễ dàng được phát hiện bởi những kẻ xâm nhập. Do đó, Wi-Fi đã sinh ra các mã hóa công nghệ khác nhau. Mã hóa đầu tiên là WEP đã được chứng minh dễ dàng bị qua mặt. Các giao thức chất lượng cao (WPA, WPA2) đã được sinh ra sau đó có chất lượng tốt hơn.

Chuẩn 802.11 được định nghĩa thông qua các đặc tả của WLAN. Nó định nghĩa các giao diện giữa các client và trạm phát hoặc giữa hai client với nhau.

Có vài đặc tả trong họ 802.11 như sau:

* 802.11 - Cho phép truyền dẫn 1 hoặc 2 Mbps sử dụng tần số 2.4 GHz của trải phố nhảy tần (FHSS) hoặc trải phổ trực tiếp (DSSS).
* 802.11a - Là chuẩn mở rộng của 802.11 gắn với mạng LAN và có tốc độ cao đến 54 Mbps trong dải tần 5 GHz. Chuẩn 802.11a triển khai dựa trên mã hóa trải phổ trực giao(OFDM).
* 802.11b - Là sự mở rộng tốc độ cao của chuẩn 802.11 cho tốc độ truyền dẫn đến 11 Mbps ở dải tần 2.4 GHz. Đặc tả 802.11b sử dụng trải phổ trực tiếp (DSSS).
* 802.11g - Truyền dẫn tốc độ lên đến 54Mbps trong tần số 2.4 GHz sử dụng mã hóa trải phổ trực giao (OFDM).
* 802.11n – Tốc độ truyền dẫn 72Mbps đối với băng tần 2.4GHz và 150Mbps đối với băng tần 5GHz. Sử dụng điều chế MIMO-OFDM.

Ngoài ra còn một số chuẩn không được đề cập ở đây là 802.11ac, 802.11ad.

## 2.2 ARDUINO PRO MINI

### 2.2.1 Giới thiệu

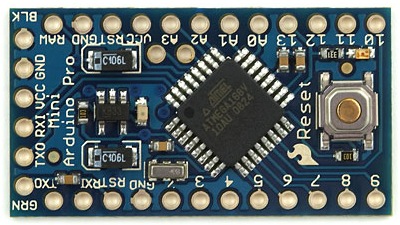
Arduino là một board mạch vi xử lý, nhằm xây dựng các ứng dụng tương tác với nhau hoặc với môi trường được thuận lợi hơn. Phần cứng bao gồm một board mạch nguồn mở được thiết kế trên nền tảng vi xử lý AVR Atmel 8bit, hoặc ARM Atmel 32-bit. Những Model hiện tại được trang bị gồm 1 cổng giao tiếp USB, 6 chân đầu vào analog, 14 chân I/O kỹ thuật số tương thích với nhiều board mở rộng khác nhau.

Được giới thiệu vào năm 2005, Những nhà thiết kế của Arduino cố gắng mang đến một phương thức dễ dàng, không tốn kém cho những người yêu thích, sinh viên và giới chuyên nghiệp để tạo ra những thiết bị có khả năng tương tác với môi trường thông qua các cảm biến và các cơ cấu chấp hành. Những ví dụ phổ biến cho những người yêu thích mới bắt đầu bao gồm các robot đơn giản, điều khiển nhiệt độ và phát hiện chuyển động. Đi cùng với nó là một môi trường phát triển tích hợp (IDE) chạy trên các máy tính cá nhân thông thường và cho phép người dùng viết các chương trình cho Aduino bằng ngôn ngữ C hoặc C++.

Arduino Mini là một board vi điều khiển nhỏ nguyên gốc dựa trên Atmega168, nhưng bây giờ với 328. Dự định để sử dụng trên breadboards và khi không gian là một hạn chế. Nó có 14 chân kỹ thuật số đầu vào / đầu ra (trong đó 6 có thể được sử dụng như là đầu ra PWM), 8 đầu vào analog và một bộ dao động tinh thể 16 MHz. Nó có thể được lập trình với các bộ chuyển đổi USB Serial hoặc USB hoặc RS232 để TTL adapter nối tiếp. Mini mới (phiên bản 05) có một gói phần mềm mới cho ATmega328, cho phép tất cả các thành phần để được trên đầu của bảng.

### 2.1.2 Phần cứng của Arduino Pro Mini

*Các thành phần chính của board:*



**Hình 2.1** Board Arduino Pro Mini

**Bảng 2.1** Các thành phần của board Arduino Pro Mini.

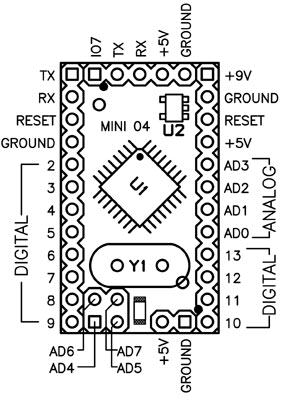
|  |  |
| --- | --- |
| Chip | **ATmega328** |
| Cấu trúc | **AVR** |
| Điện áp hoạt dộng | 5V |
| Bộ nhớ flash | 32 KB (trong đó 2KB sử dụng bởi Bootloader) |
| SRAM | 2KB |
| Kích thước | 85x56x17mm |
| CPU | 16 MHz |
| EEPROM | 1 KB |
| Analog I/O chân | 8 (trong đó 4 được chia ra thành các chân) |
| Dòng chân I/O | 40 mA |

***Input and Output:***

Mỗi trong số 14 chân digital trên Mini có thể được sử dụng như một đầu vào hay đầu ra. Chúng hoạt động ở 5 volts. Mỗi pin có thể cung cấp hoặc nhận được tối đa 40 mA và có một điện trở kéo lên (ngắt kết nối theo mặc định) 20-50 kOhms. Chân 3, 5, 6, 9, 10, và 11 có Kết quả xuất ra PWM; xem chi tiết ở các analogWrite () chức năng. Nếu bất cứ ngoài Mini USB (hay khác) adapter được kết nối với chân 0 và 1, nó sẽ gây trở ngại cho việc giao tiếp USB, ngăn chặn code mới từ được tải lên hoặc truyền qua lại với máy tính. Các Mini có 8 đầu vào analog mỗi trong số đó cung cấp 10 bit độ phân giải (ví dụ: năm 1024 giá trị khác nhau). Đầu vào 0-3 được chia ra trên chân; đầu vào 4-7 yêu cầu hàn vào các lỗ cung cấp. Theo mặc định các đầu vào analog đo từ mặt đất đến 5 volts, mặc dù là nó có thể thay đổi các đầu trên của phạm vi của nó bằng cách sử dụng pin Aref và một số mã ở mức độ thấp.

***Programming:***

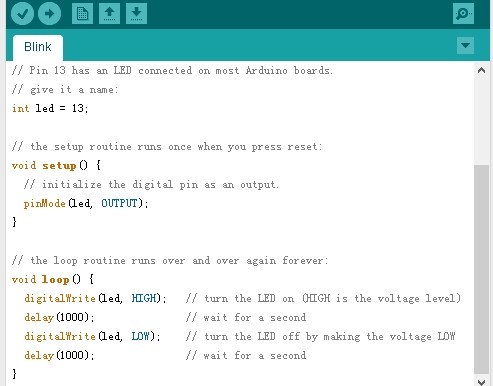
Arduino Mini có thể được lập trình với phần mềm Arduino. Để nạp chương trình Arduino Mini, bạn sẽ cần một bộ chuyển đổi USB Serial hoặc USB hoặc RS232 để TTL adapter nối tiếp. Các ATmega328 trên Arduino Mini đi kèm preburned với một Bootloader cho phép bạn tải lên mã mới cho nó mà không sử dụng một hệ thống lập trình trong. Các Bootloader giao tiếp bằng cách sử dụng giao thức STK500 gốc. Bạn cũng có thể bỏ qua các Bootloader và chương trình ATmega328 với ICSP.



**Hình 2.2** Sơ đồ chân của Arduino Pro Mini.

### 2.2.3 Môi trường lập trình trên Arduino

Môi trường phát triển tích hợp ([IDE](https://vi.wikipedia.org/wiki/IDE)) của Arduino là một ứng dụng  nền tảng được viết bằng [Java](https://vi.wikipedia.org/w/index.php?title=Java_(programming_language)&action=edit&redlink=1), và từ IDE này sẽ được sử dụng cho [Ngôn ngữ lập trình xử lý](https://vi.wikipedia.org/w/index.php?title=Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh_x%E1%BB%AD_l%C3%BD&action=edit&redlink=1) và project để viết. Nó được thiết kế để dành cho những người mới tập làm quen với lĩnh vực phát triển phần mềm. Nó bao gồm một chương trình code chỉnh sửa với các chức năng như đánh dấu cú pháp, tự động [tạo](https://vi.wikipedia.org/w/index.php?title=Brace_matching&action=edit&redlink=1) hàm, và tự động canh lề, cũng như biên dịch và upload chương trình lên board chỉ với 1 cú nhấp chuột. Một chương trình hoặc code viết cho Arduino được gọi là một bản phác họa.



**Hình 2.3** Giao diện phần mềm IDE

Các chương trình Arduino được viết bằng [C](https://vi.wikipedia.org/w/index.php?title=C_(programming_language)&action=edit&redlink=1) hoặc [C++](https://vi.wikipedia.org/wiki/C%2B%2B). Arduino IDE đi kèm với một [thư viện phần mềm](https://vi.wikipedia.org/w/index.php?title=Th%C6%B0_vi%E1%BB%87n_ph%E1%BA%A7n_m%E1%BB%81m&action=edit&redlink=1) được gọi là "[Wiring](https://vi.wikipedia.org/w/index.php?title=Wiring_(development_platform)&action=edit&redlink=1)", từ project Wiring gốc, có thể giúp các thao tác input/output được dễ dàng hơn. Người dùng chỉ cần định nghĩa 2 hàm để tạo ra một chương trình [vòng thực thi](https://vi.wikipedia.org/w/index.php?title=V%C3%B2ng_th%E1%BB%B1c_thi&action=edit&redlink=1) có thể chạy được:

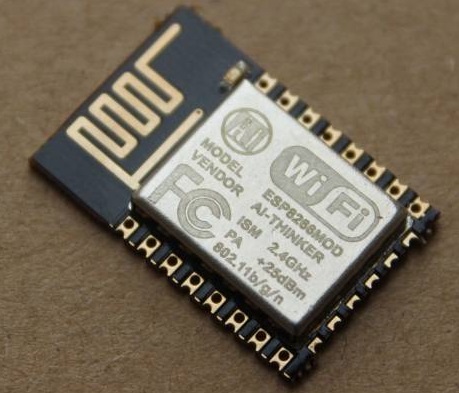
* setup(): hàm này chạy mỗi khi khởi động một chương trình, dùng để thiết lập các cài đặt
* loop(): hàm này được gọi lặp lại cho đến khi tắt nguồn board mạch

Vì nền tảng của Arduino là các vi điều khiển của Atmel, cho nên môi trường phát triển của Atmel, AVR Studio hoặc các phiên bản Atmel Studio mới hơn, cũng có thể được sử dụng để làm phần mềm phát triển cho Arduino.

## 2.3 MẠCH THU PHÁT WIFI ESP8266

### 2.3.1 Giới thiệu

Đây là module truyền nhận WiFi đơn giản dựa trên chip ESP8266 SoC (System on Chip) của hãng Espresif. Module ESP8266 thường được sử dụng trong các ứng dụng IOT (Internet of Thing). Module này đã được nập sẵn firmware giúp người dùng giao tiếp với WiFi rất đễ dàng qua tập lệnh AT thông qua giao tiếp UART hoặc có thể lập trình trực tiếp bằng phần mềm IDE (baudrate mặc định 9600, một sô mạch là 11520) quen thuộc.



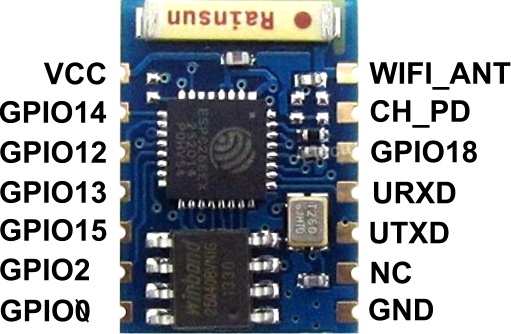
## Hình 2.4 Mạch thu phát wifi ESP8266

### 2.3.2 Thông số

Thông số kỹ thuật:

* Hỗ trợ chuẩn 802.11 b/g/n.
* Wi-Fi 2.4 Ghz, hỗ trợ WPA/WPA2.
* Chuẩn điện áp hoạt động: 3.3 V.
* Chuẩn giao tiếp UART với tốc độ Baud lên đến 115200.
* Có 3 chế độ hoạt động: Client, Access Point, Both Client and Acces Point.
* Hỗ trợ chuẩn bảo mật như: OPEN, WEB, WAP\_PSK, WPA2\_PSK, WPA\_WPA2\_PSK.
* Hỗ trợ cả 2 giao thức TCP và UDP.
* Làm việc như các máy chủ có thể kết nối 5 máy con.

### 2.3.3 Sơ đồ chân



**Hình 2.5** Sơ đồ chân của ESP8266

**Chức năng các chân:**

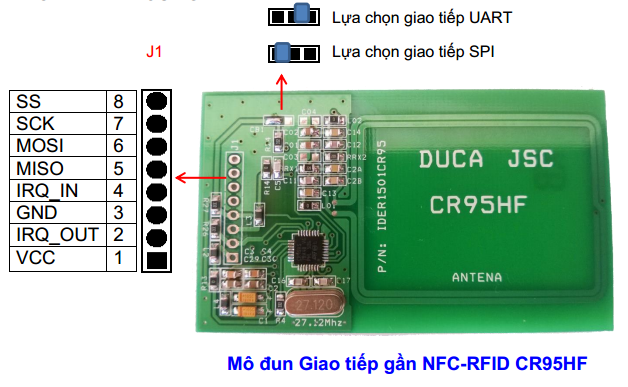
* VCC: 3.3 V, dòng có thể lên 330mA vì thế cần mạch nguồn riêng.
* GND: 0V
* Tx: Chân Tx của giao thức UART, kết nối đến chân Rx của vi điều khiển
* Rx: Chân Rx của giao thức UART, kết nối đến chân Tx của vi điều khiển
* RST: chân reset, kéo xuống mas để reset.
* CH\_PD: chân này nếu được kéo lên mức cao module sẽ bắt đầu phát WiFi kéo xuống mức thấp module dừng phát WiFi.
* GPIO 0: Kéo xuống thấp cho chế độ upgrade firmware.
* GPIO 2: không sử dụng.
* Giới thiệu tổng quan về Chip ESP8266
* Chip ESP8266 có giá thành rẻ và được đánh giá rất cao cho các ứng dụng liên quan đến Internet và Wifi cũng như các ứng dụng truyền nhận sử dụng thay thế cho các module RF khác.
* 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 Wi-Fi đầy đủ và khép kín, cho phép nó có thể lưu trữ các ứng dụng hoặc để giảm tải tất cả các chức năng kết nối mạng Wi-Fi từ một bộ xử lý ứng dụng.
* ESP8266 có tốc độ xử lý và khả năng lưu trữ mạnh mẽ cho phép nó được tích hợp với các bộ cảm biến, vi điều khiển và các thiết bị ứng dụng cụ thể khác thông qua GPIOs với một chi phí tối thiểu và một  PCB tối thiểu.
* Tính năng của Chip
* Tích hợp bộ xử lý RISC 32bit, bộ xử lý với tốc độ lên tới 80MHz, 64KB bộ nhớ lệnh RAM, 96KB bộ nhớ dữ liệu RAM
* Hỗ trợ chuẩn 802.11 b/g/n wifi, hỗ trợ bảo mật WEP hoặc WPA/WPA2 và mạng mở.
* Có bộ chuyển đổi ADC 10 bit
* 16 chân GPIO với mức điện áp ra 0-3.3V
* Thông số kỹ thuật
* Tương thích các chuẩn Wifi : 802.11 b/g/n.
* Hỗ trợ: Wi-Fi Direct (P2P), soft-AP.
* Tích hợp TCP/IP protocol stack.
* Tích hợp bộ nhân tần, ổn áp, quản lý nguồn.
* Trong chế độ 802.11b công suất phát lên tới +25dBm.
* Công suất tiêu thụ trạng thái nghỉ < 10µA.
* Thời gian đánh thức CPU < 2ms
* Công suất tiêu thụ ở chế độ standby < 1.0mW

## 2.4 MẠCH ĐỌC THẺ NFC CR95HF

### 2.4.1 Giới thiệu

NFC (Near-Field Communications) là công nghệ kết nối không dây phạm vi tầm ngắn trong khoảng cách 4 cm, sử dụng cảm ứng từ trường để thực hiện kết nối giữa các thiết bị khi có sự tiếp xúc trực tiếp hay để gần nhau. NFC được phát triển dựa trên nguyên lý nhận dạng bằng tín hiệu tần số vô tuyến (Radio-frequency identification - RFID), hoạt động ở dải băng tần 13.56 MHz và tốc độ truyền tải dữ liệu tối đa 424 Kbps.

Do khoảng cách truyền dữ liệu khá ngắn nên giao dịch qua công nghệ NFC được xem là an toàn. Thiết bị được trang bị NFC thường là điện thoại di động, có thể giao tiếp với các thẻ thông minh, đầu đọc thẻ hoặc thiết bị NFC tương thích khác. Ngoài ra, NFC còn được kết hợp nhiều công nghệ sử dụng trong các hệ thống công cộng như bán vé, thanh toán hóa đơn…



**Hình 2.6** Mạch NFC CR95HF

### 2.4.2 Thông số

Thông số kỹ thuật:

* IC chính: CR95HF-VMD5T nhiều giao thức RF
* Anten PCB 46x34 mm.
* Giao tiếp SPI hoặc UART với MCU
* Chế độ hoạt động: Reader/Writer
* Giao tiếp NFC-RFID 13.56MHz
* ISO/IEC 14443 type A và B
* Khoảng cách giao tiếp trong tầm 10 cm

### 2.4.3 Nguyên lý hoạt động mạch NFC CR95HF

### Để NFC hoạt động, chúng ta buộc phải có 2 thiết bị, 1 là thiết bị khởi tạo (initiator) và thiết bị thứ 2 là mục tiêu (target). Bí mật của NFC nằm ở đây, initiator sẽ chủ động tạo ra những trường sóng radio (bản chất là bức xạ điện từ) đủ để cung cấp năng lượng cho target vốn hoạt động ở chế độ bị động. Target của NFC sẽ không cần điện năng, năng lượng để nó hoạt động lấy từ thiết bị initiator. Đây là một đặc điểm cực kỳ có ý nghĩa vì nó cho phép người ta chế tạo những thẻ tag, miếng dán, chìa khóa hay thẻ NFC nhỏ gọn hơn do không phải dùng pin.

### Loại hình NFC đang được ứng dụng hiện nay là thẻ nhận dạng NFC (NFC tag). Thẻ nhận dạng NFC có vai trò tương tự mã vạch hay mã QR. Thẻ NFC thường chứa dữ liệu chỉ đọc nhưng cũng có thể ghi đè được. Chúng có thể được tùy biến, mã hóa bởi nhà sản xuất hoặc sử dụng những thông số riêng do NFC Forum cung cấp. Thẻ NFC có thể lưu trữ an toàn các dữ liệu cá nhân như thông tin tài khoản tín dụng, tài khoản ghi nợ, dữ liệu ứng dụng, mã số PIN, mạng lưới danh bạ, …. NFC Forum đã phân ra 4 loại thẻ NFC trong đó mỗi loại lại có tốc độ giao tiếp và khả năng được tùy biến, bộ nhớ, bảo mật và thời hạn sử dụng khác nhau.

### NFC được phát triển dựa trên nhiều công nghệ không dây cự ly ngắn, khoảng cách thường dưới 4 cm. NFC hoạt động theo tần số 13.56 MHz và tốc độ truyền tải khoảng từ 106 kbit/s đến 848 kbit/s. NFC luôn yêu cầu một đối tượng khởi động và một đối tượng làm mục tiêu, chúng ta có thể hiểu nôm na là một máy sẽ đóng vai trò chủ động và máy còn lại bị động. Máy chủ động sẽ tạo ra một trường tần số vô tuyến (RF) để giao tiếp với máy bị động.

## 2.5 CẢM BIẾN DÒNG ĐIỆN HALL ACS712

### 2.5.1 Giới thiệu

IC ACS 712 là một IC cảm biến dòng tuyến tính dựa trên hiệu ứng Hall. ACS xuất ra 1 tín hiệu analog, Vout biến đổi tuyến tính theo sự thay đổi của dòng điện Ip được lấy mẫu thứ cấp DC (hoặc AC), trong phạm vi đã cho. CF được dùng với mục đích chống nhiễu và có giá trị tùy thuộc vào từng mục đích sử dụng.

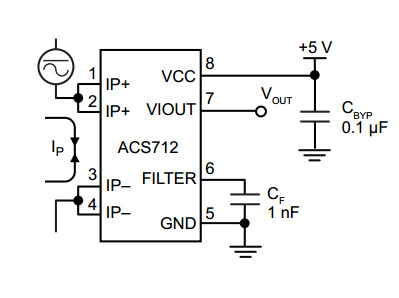


**Hình 2.7** Cảm biến dòng điện Hall ACS712

### 2.5.2 Thông số

* Thông số kỹ thuật:
* Đường tín hiệu analog có độ nhiễu thấp.
* Thời gian tăng của đầu ra để đáp ứng với đầu vào là 5µs.
* Điện trở dây dẫn trong là 1.2mΩ.
* Nguồn vận hành đơn là 5V.
* Độ nhạy đầu ra từ 63-190mV/A.
* Điện áp ra cực kỳ ổn định.
* ACS 712 sử dụng 20A (x20A)
* Dòng vào: -20A - 20A.
* Độ nhạy: 96 -104 mV/A.

### 2.5.3 Nguyên lý hoạt động



### Hình 2.8 Sơ đồ nguyên lý ACS712

* Đo dòng điện AC

Khi đo dòng điện Ac, đo dòng điện AC không có chiều nên không cần quan tâm chiều. Cấp nguồn 5V cho module khi chưa có dòng Ip (chưa có tải nối tiếp với domino), thì Vout = 2.5V. Khi có dòng điện Ip (dòng AC) do xoay chiều độ lớn thay đổi liên tục theo hình sin, nên điện thế Vout sẽ cũng là 1 hình sin có độ lớn tuyến tính với dòng AC, 0 đến 5V tương ứng với -20A đến 20A.

## 2.6 PHẦN MỀM ANDROID STUDIO

## C:\Users\boyxo\Desktop\studio-logo.png

## Hình 2.9 Phần mềm Android Studio

### 2.6.1 Giới thiệu

Google cung cấp một công cụ phát triển ứng dụng Android trên Website chính thức dựa trên nền tảng IntelliJ IDEA gọi là Android Studio. Bộ công cụ Android Studio cung cấp những trình soạn thảo riêng biệt tương ứng với hầu hết các file cấu hình và Layout của ứng dụng Android với định dạng XML. Với những file Layout giao diện, Android Studio cho phép các lập trình viên dễ dàng chuyển đổi chế độ chỉnh sửa giữa trình biên soạn nội dung XML hoặc trình biên soạn dưới dạng giao diện (GUI).

Ngoài ra, bộ phát triển Android Studio còn tích hợp bên trong những tiện ích hỗ trợ phát triển ứng dụng như sau:

* Bộ xây dựng ứng dụng Gradle với nhiều cấu hình linh động.
* Cho phép xây dựng ứng dụng tùy biến và tự động tạo ra file. apk tương thích với thông tin tùy biến.
* Bộ code mẫu giúp chúng ta xây dựng các chức năng phổ biến của các ứng dụng.
* Trình biên soạn Layout GUI cho ứng dụng Android phong phú và tiện lợi, cho phép người dùng dễ dàng tạo ra giao diện màn hình bằng cách kéo thả các Component mẫu có sẵn và chỉnh sửa giao diện themes (kích thướt, màu sắc, …) tùy ý.
* Tích hợp **lint** – Một ứng dụng tích hợp giúp các developer kiểm soát về hiệu suất (performance), tính khả dụng (usability), khả năng tương thích của các phiên bản API sử dụng, và những vấn đề tiềm ẩn bên trong có thể xảy ra lúc Runtime.
* ProGuard (tiện ích tối ưu và mã hóa code khi build ứng dụng) và Android app-signing.
* Bộ tích hợp hỗ trợ phát triển ứng dụng Android dễ dàng với các dịch vụ cùa nền tảng đám mây của Google

### 2.6.2 Các bước để viết một ứng dụng

### C:\Users\boyxo\Desktop\Android-01.jpg

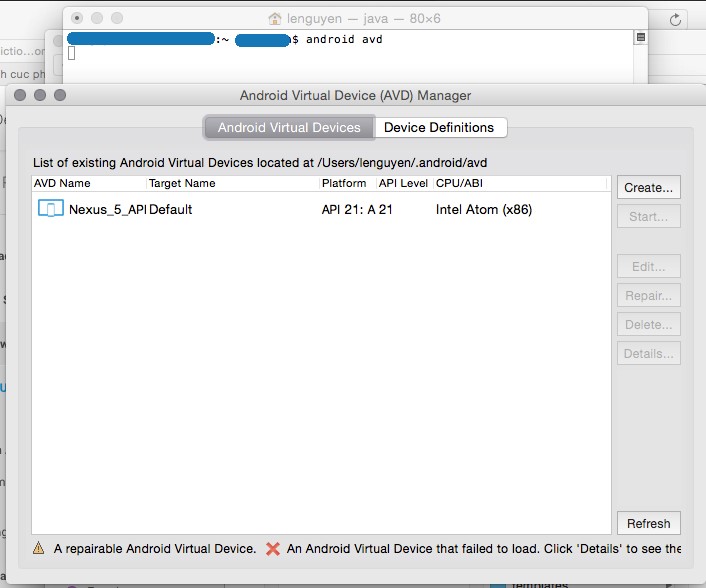
### Hình 2.10 Các bước tạo ứng dụng

#### Bước 1:**Cài đặt môi trường Environment Setup**

* Cài đặt Android SDK.
* Cài đặt Android Development Tools
* Cài đặt Android Platform

**Bước 2:** **Tạo và phát triển ứng dụng**

Android studio là IDE hỗ trợ lập trình. Ngôn ngữ lập trình sử dụng là dùng Java và XML (thiết kế layout). Bao gồm các cấu trúc, các resources, các file cấu hình…của một project



**Hình 2.11** Tạo SDK cho ứng dụng

#### ****Bước 3:** Building, Debugging và Testing**

Android Studio sử dụng bộ nhân để build hệ thống dựa trên Gradle sẽ giúp bạn build ứng dụng thành một gói. apk với chế độ debug tùy biến, độc lập và linh động.

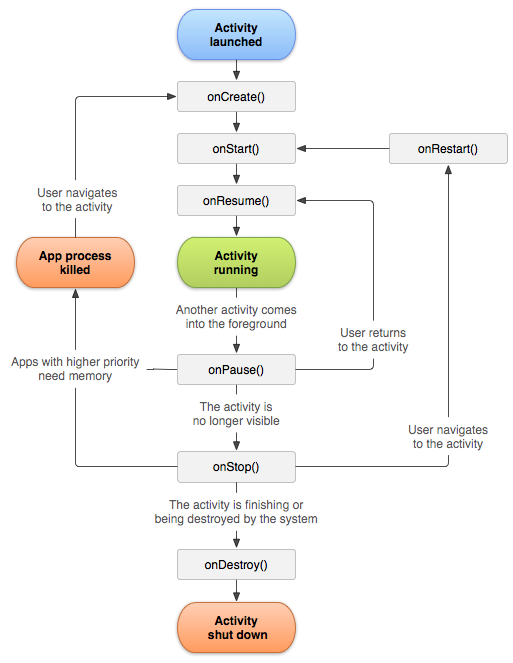
Bạn có thể debug ứng dụng của bạn với một bộ tiện ích trực quan “*Android Device Monitor*” và hệ thống Logcat được tích hợp bên trong Android Studio hoặc bộ Debugger được hỗ trợ chạy độc lập từ Android SDK.

#### ****Bước 4:**Signing và Publishing**

Publishing là một qúa trình làm cho ứng dụng của bạn sẵn sàng cho phép người dùng tải về và cài đặt lên thiết bị của họ và sử dụng. Khi publish một ứng dụng Android, bạn phải thực hiện 2 nhiệm vụ sau:

**Chuẩn bị cho Release**: cập nhật file cấu hình (*file manifest*) và các Resource cần thiết, biên dịch, đóng gói và Signing phiên bản code mà bạn muốn Release. Bên cạnh đó nếu ứng dụng của bạn còn có tương tác các dịch vụ, bạn phải chuẩn bị các hệ thống Server/Services tương ứng với phiên bản bạn muốn Release.

**Release ứng dụng đến với người dùng**: Có nhiều cách cho bạn phát hành ứng dụng và cho phép người dùng có thể tải về như Marketplace, Email, Website, Apps Store… Cách thông thường và phổ biến nhất là bạn Release trên Google Play (*một tập hợp rất nhiều ứng dụng trực tuyến của Google*).

****

**Hình 2.12** Quá trình hoạt động của một App Android

**2.6 LẬP TRÌNH PHP VÀ MYSQL**

### 2.6.1 Giới thiệu ngôn ngữ Php

Hiện nay có rất nhiều ngôn ngữ lập trình web, trong đó có thể kể đến các ngôn ngữ nổi tiếng như PHP, Perl, Ruby, Python, ASP.net… Các ngôn ngữ lập trình kể trên đều có vị thế riêng trong lĩnh vực thiết kế web, nổi bật trong số đó, PHP là một ngôn ngữ lập trình mã nguồn mở, với cộng đồng hỗ trợ mạnh mẽ ở khắp mọi nơi trên thế giới. Việc tìm hiểu ngôn ngữ lập trình này giúp tiết kiệm thời gian và chi phí một cách đáng kể, kèm theo đó đây là ngôn ngữ lập trình web mạnh mẽ và thông dụng nhất hiện nay.

**PHP** - viết tắt hồi quy của **"Hypertext Preprocessor"**, là một ngôn ngữ lập trình kịch bản được chạy ở phía server nhằm sinh ra mã html trên client. PHP đã trải qua rất nhiều phiên bản và được tối ưu hóa cho các ứng dụng web. Do được tối ưu hóa cho các [ứng dụng web](https://vi.wikipedia.org/wiki/%E1%BB%A8ng_d%E1%BB%A5ng_web), tốc độ nhanh, nhỏ gọn, cú pháp giống [C](https://vi.wikipedia.org/wiki/C_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)) và [Java](https://vi.wikipedia.org/wiki/Java_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)), dễ học và thời gian xây dựng sản phẩm tương đối ngắn hơn so với các ngôn ngữ khác nên PHP đã nhanh chóng trở thành một [ngôn ngữ lập trình](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh) web phổ biến nhất thế giới.

 PHP chạy trên môi trường Webserver và lưu trữ dữ liệu thông qua hệ quản trị cơ sở dữ liệu nên PHP thường đi kèm với Apache, MySQL và hệ điều hành Linux (LAMP).

* Apache là một phần mềm web server có nhiệm vụ tiếp nhận request từ trình duyệt người dùng sau đó chuyển giao cho PHP xử lý và gửi trả lại cho trình duyệt.
* MySQL cũng tương tự như các hệ quản trị cơ sở dữ liệu khác (Postgress, Oracle, SQL server...) đóng vai trò là nơi lưu trữ và truy vấn dữ liệu.
* Linux: Hệ điều hành mã nguồn mở được sử dụng rất rộng rãi cho các webserver. Thông thường các phiên bản được sử dụng nhiều nhất là RedHat Enterprise Linux, Ubuntu...

### 2.6.2 Cú pháp của Php

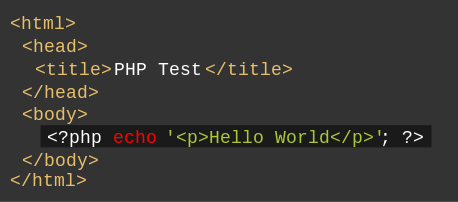
PHP chỉ phân tích các đoạn mã nằm trong những dấu giới hạn của nó. Bất cứ mã nào nằm ngoài những dấu giới hạn đều được xuất ra trực tiếp không thông qua xử lý bởi PHP. Các dấu giới hạn thường dùng nhất là <?php và ?>, tương ứngvới dấu giới hạn mở và đóng.

Các dấu giới hạn <script language="php"> và </script> cũng đôi khi được sử dụng. Cách viết dấu giới hạn dạng thẻ ngắn cũng có thể được dùng để thông báo bắt đầu đoạn mã PHP, là <? hay <?= (dấu này được sử dụng để in ra (echo) các [xâu ký tự](https://vi.wikipedia.org/w/index.php?title=X%C3%A2u_(m%C3%A1y_t%C3%ADnh)&action=edit&redlink=1) hay [biến](https://vi.wikipedia.org/wiki/Bi%E1%BA%BFn_s%E1%BB%91)) với thẻ thông báo kết thúc đoạn mã PHP là ?>.

Những thẻ này thường xuyên được sử dụng, tuy nhiên giống với những thẻ kiểu [ASP](https://vi.wikipedia.org/w/index.php?title=ASP&action=edit&redlink=1) (<%hay <%= và %>), chúng không có tính di động cao bởi có thể bị vô hiệu khi [cấu hình PHP](http://wiki.php.net/rfc/shortags). Bởi vậy, việc dùng các thẻ dạng ngăn hay các thẻ kiểu ASP không được khuyến khích. Mục đích của những dấu giới hạn này là ngăn cách mã PHP với những đoạn mã thuộc ngôn ngữ khác, gồm cả HTML. Mọi đoạn mã bên ngoài các dấu này đều bị hệ thống phân tích bỏ qua và được xuất ra một cách trực tiếp.

Các biến được xác định bằng cách thêm vào trước một [dấu đô la](https://vi.wikipedia.org/wiki/K%C3%BD_hi%E1%BB%87u_%C4%91%C3%B4_la) (*$*) và không cần xác định trước [kiểu dữ liệu](https://vi.wikipedia.org/w/index.php?title=Ki%E1%BB%83u_d%E1%BB%AF_li%E1%BB%87u_nguy%C3%AAn_th%E1%BB%A7y&action=edit&redlink=1). Không giống với tên hàm và lớp, tên biến là trường hợp nhạy cảm. Cả dấu ngoặc kép ("") và ký hiệu đánh dấu văn bản (<<<EOF EOF;) đều có thể dùng để truyền xâu và giá trị biến. PHP coi [xuống dòng](https://vi.wikipedia.org/w/index.php?title=Xu%E1%BB%91ng_d%C3%B2ng&action=edit&redlink=1) như một [khoảng trắng](https://vi.wikipedia.org/w/index.php?title=Kho%E1%BA%A3ng_tr%E1%BA%AFng_(khoa_h%E1%BB%8Dc_m%C3%A1y_t%C3%ADnh)&action=edit&redlink=1) theo kiểu như một [ngôn ngữ dạng tự do](https://vi.wikipedia.org/w/index.php?title=Ng%C3%B4n_ng%E1%BB%AF_d%E1%BA%A1ng_t%E1%BB%B1_do&action=edit&redlink=1) (*free-form language*) (trừ khi nó nằm trong trích dẫn xâu), và các phát biểu được kết thúc bởi một dấu chấm phẩy. PHP có ba kiểu [cú pháp chú thích](https://vi.wikipedia.org/w/index.php?title=Ch%C3%BA_th%C3%ADch_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)&action=edit&redlink=1): /\* \*/ cho phép một đoạn chú thích tùy ý, trong khi đó // và # cho phép chú thích trong phạm vi một dòng. Phát biểu echo là một trong những lệnh của PHP cho phép xuất văn bản (vd. ra một [trình duyệt web](https://vi.wikipedia.org/wiki/Tr%C3%ACnh_duy%E1%BB%87t_web)).

Về cú pháp các từ khóa và ngôn ngữ, PHP tương tự hầu hết các ngôn ngữ lập trình bậc cao có cú pháp kiểu C. Các phát biểu điều kiện *If* , vòng lặp *for* và *while*, các hàm trả về đều tương tự cú pháp của các ngôn ngữ như [C](https://vi.wikipedia.org/wiki/C), [C++](https://vi.wikipedia.org/wiki/C%2B%2B), [Java](https://vi.wikipedia.org/wiki/Java) và [Perl](https://vi.wikipedia.org/wiki/Perl).



**Hình 2.13** Một đoạn code Php

#### 2.6.3 Giới thiệu JSON

Để thuận tiện cho việc truyền tải dữ liệu từ Web server xuống ứng dụng điện thoại, nhóm có sử dụng thêm một dạng dữ liệu khác là JSON. Đây là một dạng dữ liệu tuân theo một quy luật nhất định mà hầu hết các ngôn ngữ lập trình hiện nay đều có thể đọc được, dữ liệu có thể sử dụng lưu nó vào một file, một record trong CSDL rất dễ dàng. JSON có định dạng đơn giản, dễ dàng sử dụng và truy vấn hơn XML rất nhiều nên tính ứng dụng của nó hiện nay rất là phổ biến.

Cú pháp của JSON rất đơn giản là mỗi thông tin dữ liệu sẽ có 2 phần đó là **key** và **value,**điều này tương ứng trong CSDL là tên field và giá trị của nó ở một record nào đó. Chính vì sự thuận tiện này, khi truyền tải dữ liệu từ web server xuống ứng dụng trên điện thoại, có thể dễ dàng lấy được dữ liệu và lưu trữ lại vào cơ sở dữ liệu.

**2.6.4 Giới thiệu MySQL**

**MySQL** là [hệ quản trị cơ sở dữ liệu](https://vi.wikipedia.org/wiki/H%E1%BB%87_qu%E1%BA%A3n_tr%E1%BB%8B_c%C6%A1_s%E1%BB%9F_d%E1%BB%AF_li%E1%BB%87u) [tự do nguồn mở](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_ngu%E1%BB%93n_m%E1%BB%9F) phổ biến nhất thế giới và được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng. Vì MySQL là cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Với tốc độ và tính bảo mật cao, MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên internet. MySQL miễn phí hoàn toàn cho nên bạn có thể tải về MySQL từ trang chủ. Nó có nhiều phiên bản cho các hệ điều hành khác nhau: phiên bản Win32 cho các hệ điều hành dòng [Windows](https://vi.wikipedia.org/wiki/Microsoft_Windows), [Linux](https://vi.wikipedia.org/wiki/Linux), [Mac OS X](https://vi.wikipedia.org/wiki/Mac_OS_X), [Unix](https://vi.wikipedia.org/wiki/Unix), [FreeBSD](https://vi.wikipedia.org/wiki/FreeBSD), [NetBSD](https://vi.wikipedia.org/w/index.php?title=NetBSD&action=edit&redlink=1), [Novell NetWare](https://vi.wikipedia.org/w/index.php?title=Novell_NetWare&action=edit&redlink=1), [SGI Irix](https://vi.wikipedia.org/w/index.php?title=SGI_Irix&action=edit&redlink=1), [Solaris](https://vi.wikipedia.org/wiki/Solaris), [SunOS](https://vi.wikipedia.org/w/index.php?title=SunOS&action=edit&redlink=1),...

**MySQL** là một trong những ví dụ rất cơ bản về Hệ Quản trị Cơ sở dữ liệu quan hệ sử dụng Ngôn ngữ truy vấn có cấu trúc (SQL).

**MySQL** được sử dụng cho việc bổ trợ [PHP](https://vi.wikipedia.org/wiki/PHP), [Perl](https://vi.wikipedia.org/wiki/Perl), và nhiều ngôn ngữ khác, nó làm nơi lưu trữ những thông tin trên các trang web viết bằng PHP hay Perl,...

**2.6.5 Ưu điểm của MYSQL**

Tốc độ: MySQL rất nhanh. Những nhà phát triển cho rằng MySQL là cơ sở dữ liệu nhanh nhất mà bạn có thể có.

Dễ sử dụng: MySQL tuy có tính năng cao nhưng thực sự là một hệ thống cơ sở dữ liệu rất đơn giản và ít phức tạp khi cài đặt và quản trị hơn các hệ thống lớn

Giá thành: MySQL là miễn phí cho hầu hết các việc sử dụng trong một tổ chức.

Hỗ trợ ngôn ngữ truy vấn: MySQL hiểu SQL, là ngôn ngữ của sự chọn lựa cho tất cả các hệ thống cơ sở dữ liệu hiện đại. Bạn cũng có thể truy cập MySQL bằng cách sử dụng các ứng dụng mà hỗ trợ ODBC (Open Database Connectivity -một giao thức giao tiếp cơ sở dữ liệu được phát triển bởi Microsoft).

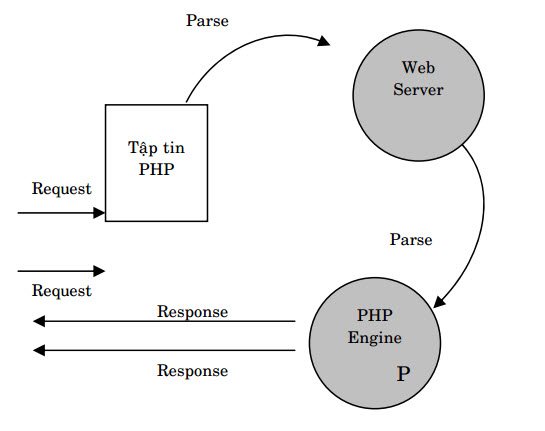
Năng lực: Nhiều client có thể truy cập đến server trong cùng một thời gian. Các client có thể sử dụng nhiều cơ sở dữ liệu một cách đồng thời. Bạn có thể truy cập MySQL tương tác với sử dụng một vài giao diện để bạn có thể đưa vào các truy vấn và xem các kết quả: các dòng yêu cầu của khách hàng, các trình duyệt Web…

Kết nối và bảo mật: MySQL được nối mạng một cách đầy đủ, các cơ sở dữ liệu có thể được truy cập từ bất kỳ nơi nào trên Internet do đó bạn có thể chia sẽ dữ liệu của bạn với bất kỳ ai, bất kỳ nơi nào. Nhưng MySQL kiểm soát quyền truy cập cho nên người mà không nên nhìn thấy dữ liệu của bạn thì không thể nhìn được.

Tính linh động: MySQL chạy trên nhiều hệ thống UNIX cũng như không phải UNIX chẳng hạn như Windows hay OS/2. MySQL chạy được các với mọi phần cứng từ các máy PC ở nhà cho đến các máy server.

Sự phân phối rộng: MySQL rất dễ dàng đạt được, chỉ cần sử dụng trình duyệt web của bạn. Nếu bạn không hiểu làm thế nào mà nó làm việc hay tò mò về thuật toán, bạn có thể lấy mã nguồn và tìm tòi nó. Nếu bạn không thích một vài cái, bạn có thể thay đổi nó.

Sự hỗ trợ: Bạn có thể tìm thấy các tài nguyên có sẵn mà MySQL hỗ trợ. Cộng đồng MySQL rất có trách nhiệm. Họ trả lời các câu hỏi trên mailing list thường chỉ trong vài phút. Khi lỗi được phát hiện, các nhà phát triển sẽ đưa ra cách khắc phục trong vài ngày, thậm chí có khi trong vài giờ và cách khắc phục đó sẽ ngay lập tức có sẵn trên Internet.

****

**Hình 2.14** Nguyên lý hoạt động PHP

Thông thường khi học MySQL thì chúng ta sẽ được học những câu truy vấn T-SQL như select, insert, update, delete. Chúng ta hãy xem PHP như là một con người bình thường, nghĩa là nó có thể viết ra các câu truy vấn và thực thi các câu truy vấn đó giống như bạn vậy. Nhưng điểm khác biệt ở đây là bạn sẽ bắt PHP thực hiện những câu truy vấn chứ không phải tự nó nghĩ ra.

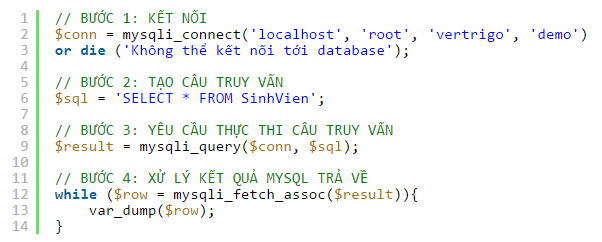
**Ví dụ**:  Mình muốn lấy danh sách tất cả sinh viên trong trường thì sẽ viết câu truy vấn như sau.



Kết quả ta thực thi câu truy vấn này trong PHP MY ADMIN là danh sách các sinh viên.

Như ta biết MySQL là một hệ quản trị CSDL nên nó có tính an toàn về bảo mật, nghĩa là nó có chức năng quản lý User, phân quyền User. Vì vậy trước khi vào PHP MY ADMIN cần phải đăng nhập.

Như vậy với PHP thì cũng phải có bước đăng nhập, sau đó sẽ là bước định nghĩa câu truy vấn và cuối cùng là thực thi và lấy kết quả.



Tóm lại PHP sẽ có nhiệm vục kết nối MySQL và yêu cầu MySQL thực thi các câu truy vấn và trả kết quả về cho PHP để từ đó PHP lấy kết quả để xử lý.

#### 2.6.6 Giới thiệu HTTP

Trong toàn bộ quá trình, dữ liệu được client gửi tới web server thông qua HTTP request và được server trả lời lại thông qua các HTTP response. Đây là hai loại thông điệp của giao thức HTTP cho phép client và server có thể giao tiếp được với nhau.

Thông tin trong một HTTP request bao gồm: request url, header, body (phần body có thể rỗng). Các method được hỗ trợ trong một HTTP request bao gồm:

* GET: phương thức hay dùng để lấy tài nguyên từ URL truy vấn. Khi truy vấn dùng phương thức này thì các tham số sẽ được nối thẳng vào URL. Phương thức GET không có phần thân hay phần thân rỗng. Phương thức này có nhiều hạn chế như việc bảo mật thông tin kém vì dữ liệu được hiển thị ngay trong URL, độ dài dữ liệu hạn chế (tối đa là 1024 byte).
* HEAD: lấy phần đầu của truy vấn, tức thông tin về tài nguyên.
* POST: gửi dữ liệu với độ dài không giới hạn tới web server. Khác với phương thức GET, phương thức POST dùng phần thân của câu truy vấn gửi dữ liệu lên web server.
* PUT: dùng để chỉ định thay thế hay upload một file…lên web server theo URL cung cấp.
* DELETE: xoá tài nguyên ở máy chủ được định vị bởi URL
* OPTIONS: trả về các phương thức mà server cung cấp cho một URL xác định
* TRACE: trả về thông tin truy vấn mà máy chủ nhận được. Phương thức này cho phép theo dõi một truy vấn để xem có những thay đổi hay thêm vào nào nếu có do nhưng máy chủ chung gian (intermidate server) thực hiện.

Thiết bị đeo tay theo dõi sức khỏe trong đề tài sử dụng phương thức POST để truyền tải dữ liệu với server, dữ liệu được nhóm lại với nhau kèm theo các tag và một số thông tin như: tên host, kiểu mã hóa, ngôn ngữ, độ dài dữ liệu … Một chuỗi Hearder HTTP request có dạng như sau:

POST / HTTP/1.1

User-Agent:

Host: www.giamsatsuckhoe.tk

Content-Type: application/x-www-form-urlencoded

Content-Length: length

Accept-Language: en-us

Accept-Encoding: gzip, deflate

Connection: Keep-Alive.

### 2.6.7 Giới thiệu về HTML

HTML là viết tắt của từ “Hyper Text Markup Language”, ngôn ngữ đánh dấu siêu văn bản này chỉ rõ một trang Web sẽ được hiển thị như thế nào trong một trình duyệt. HTML không phải là ngôn ngữ lập trình, đây là ngôn ngữ đánh dấu (markup language), bao gồm tập hợp của một nhóm các thẻ đánh dấu (các tag) để mô tả trang web. Một tài liệu html tạo thành một mã nguồn của trang Web. Khi được xem trên trình soạn thảo, tài liệu này là một chuỗi các thẻ và các phần tử, mà chúng xác định trang web hiển thị như thế nào. Trình duyệt đọc các file có đuôi .htm hay .html và hiển thị trang web đó theo các lệnh có trong đó. Tất cả các trang web dù xử lý phức tạp đến đâu đều phải trả về dưới dạng mã nguồn HTML để trình duyệt có thể hiểu và hiển thị lên được.

Cấu trúc của một tài liệu HTML bao gồm 3 thành phần cơ bản:

* Phần html: Mọi tài liệu html phải bắt đầu bằng thẻ mở html <html> và kết thúc bằng thẻ đóng html </html>. Thẻ html báo cho trình duyệt biết nội dung giữa hai thẻ này là một tài liệu html.
* Phần tiêu đề: Phần tiêu đề bắt đầu bằng thẻ <head> và kết thúc bởi thẻ </head>. Phần này chứa tiêu đề mà được hiển thị trên thanh điều hướng của trang Web. Tiêu đề nằm trong thẻ title, bắt đầu bằng thẻ <title> và kết thúc là thẻ </title>. Tiêu đề là phần khá quan trọng. Khi người dùng tìm kiếm thông tin, tiêu đề của trang Web cung cấp từ khóa chính yếu cho việc tìm kiếm.
* Phần thân: phần này nằm sau phần tiêu đề. Phần thân bao gồm văn bản, hình ảnh và các liên kết hiển thị trên trang web. Phần thân bắt đầu bằng thẻ <body> và kết thúc bằng thẻ </body>.

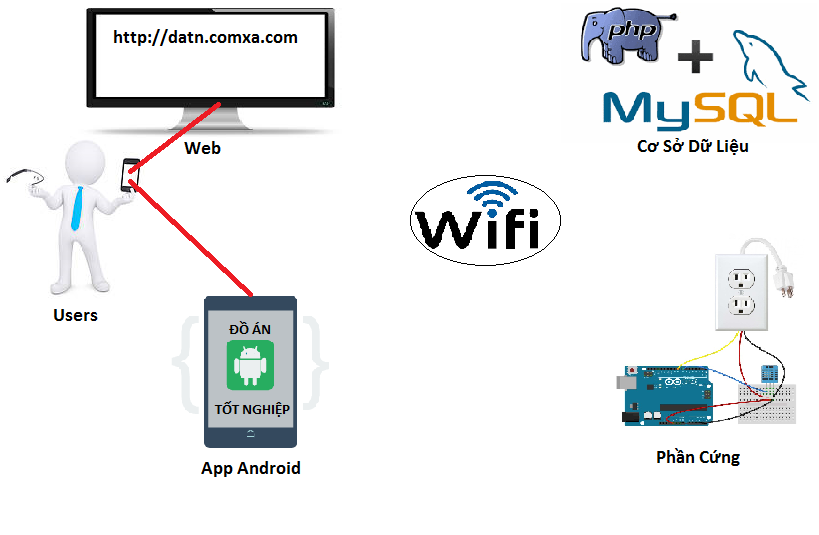
HTML được tạo thành từ rất nhiều các thẻ khác nhau, các thẻ này ghép nối với nhau để tạo thành một trang web hoàn chỉnh. Một trang web được thiết kế thường bao gồm các loại thẻ thông dụng: thẻ tiêu đề (heading) gồm 6 loại từ thẻ h1 đến h6, thẻ đoạn văn (paragraph) và các loại thẻ định dạng văn bản khác.

Trong nội dung đề tài, nhóm thực hiện đã sử dụng kết hợp ngôn ngữ lập trình PHP và HTML để thiết kế Web Application dành cho bác sĩ, cũng như thiết kế Web server cho toàn bộ hệ thống.

Chương 3

# THIẾT KẾ HỆ THỐNG

## 3.1 SƠ ĐỒ KHỐI HỆ THỐNG

**Hình 3.1** Sơ đồ khối hệ thống

Từng khối trong hệ thống này sẽ được giải quyết các vấn đề về thiết kế như sau:

* *Khối phần cứng*

Khối này thực hiện đọc dữ liệu thu được từ các cảm biến xử lý các thông số nhận được rồi gởi cho ESP truyền lên Sever. Giao tiếp với hệ thống qua wifi và điều khiển các relay đóng mở lỗ cắm điện.

* *Khối Server SQL*

Khối này thực hiện việc lưu trữ thông tin hẹn giờ hệ thống, tình trạng, công suất và thông tin người dùng .. vào cơ sở dữ liệu SQL Server. Ngoài ra khối server còn nhận dữ liệu từ ESP và Ứng dụng điện thoại để gởi xuống lệnh điều khiển hay lấy dữ liệu trên CSDL.

* *Khối người dùng*

*Ứng dụng* được xây dựng trên điện thoại sử dụng hệ điều hành Android, mục đích của ứng dụng là giúp người sử dụng có thể theo dõi các công suất thu thập từ phần cứng hoặc gởi lệnh điều khiển xuống.

*Web Application* được xây dựng với mục đích hỗ trợ trong việc theo dõi tình trạng ổ cắm và cho phép điều khiển.

## 3.2 KHỐI PHẦN CỨNG

### 3.2.1 Kit Arduino Pro Mini

Arduino Pro Mini đóng vai trò là bộ điều khiển trung tâm, xử lý tín hiệu điện áp từ ba cảm biến đo dòng điện và mạch đọc thẻ NFC, đồng thời điều khiển ba relay đóng mở. Sau quá trình xử lý và thu thập dữ liệu hoàn thành Kit sẽ bắt đầu gởi dữ liệu qua ESP để kết nối với Web Server. Các linh kiện trên được gộp chung lại một mạch nhỏ và đặt trong một ổ điện do chúng tôi thiết kế.

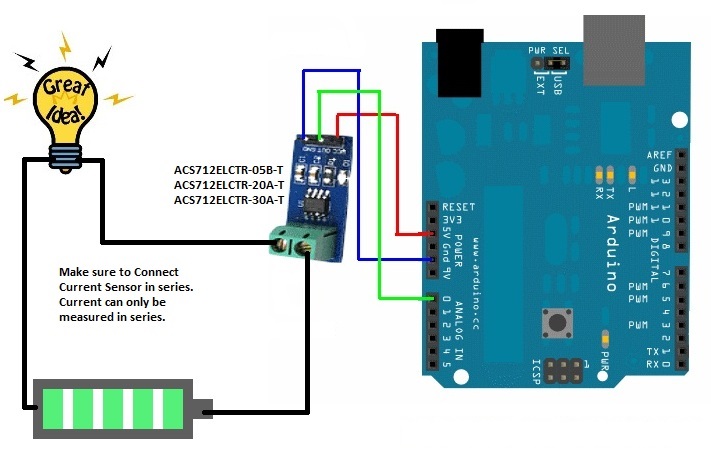
Chúng tôi sử dụng chân A0, A1, A2 analog trên Mini để đọc ba cảm biến đo dòng. Mỗi pin có thể cung cấp hoặc nhận được tối đa 40 mA và có một điện trở kéo lên (ngắt kết nối theo mặc định) 20-50 kOhms. Chân 10, 11 và 12 là ba chân kích relay. Chân 2,3 lần lượt là chân Tx và Rx để truyền-nhận dữ liệu cho ESP, chân 4,5 là Tx, Rx truyền nhận NFC. Arduino Mini có thể được lập trình với phần mềm IDE. Để nạp chương trình Arduino Mini, chúng tôi sử dụng một bộ chuyển đổi USB UART ..

### 3.2.2 Cảm biến dòng điện Hall ACS712

IC ACS 712 là một IC cảm biến dòng tuyến tính dựa trên hiệu ứng Hall. ACS xuất ra 1 tín hiệu analog, Vout biến đổi tuyến tính theo sự thay đổi của dòng điện Ip. Từ đó có thể dễ dàng tính ra công suất thiết bị độ chính xác khoảng 95%. Tôi chọn ACS712 vì đặc tính:

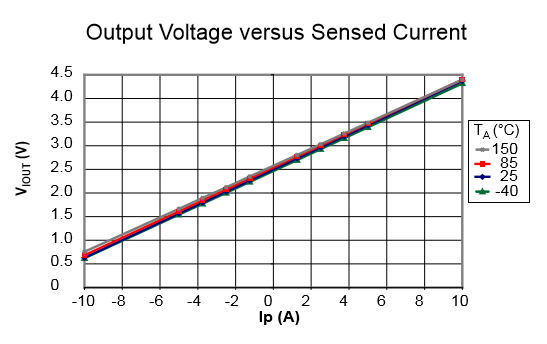
* Đường tín hiệu analog có độ nhiễu thấp.
* Thời gian tăng của đầu ra để đáp ứng với đầu vào là 5µs.
* Điện trở dây dẫn trong là 1.2mΩ.
* Nguồn vận hành đơn là 5V.
* Độ nhạy đầu ra từ 63-190mV/A.
* Điện áp ra cực kỳ ổn định.
* ACS 712 sử dụng 20A (x20A):
* Dòng vào: -20A - 20A.
* Độ nhạy: 96 -104 mV/A.

Kết nối cảm biến ACS712 với Arduino



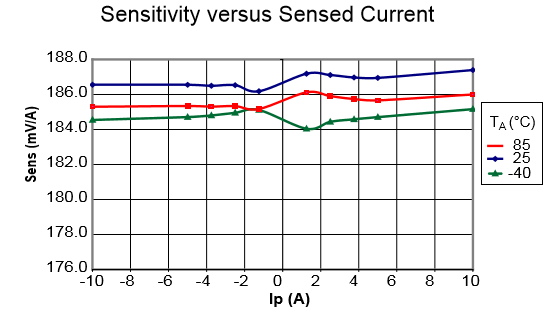
**Hình 3.2** Sơ đồ kết nối cảm biến với kit

Cách tính dòng điện ASC712 (10 A)



**Hình 3.3** Điện áp đầu ra so với dòng điện đầu vào

Độ nhạy đầu ra từ 185 mV/A và ít thay đổi bởi nhiệt độ



**Hình 3.4** Ảnh hưởng nhiệt độ đến độ nhạy của cảm biến

ASC712 hoạt động như sau:

Điện áp ra được thay đổi tuyến tính theo sự thay đổi của dòng điện Ip lấy mẫu thứ cấp DC (hoặc AC). Tại 2.5V thì dòng điện bằng 0. Từ 0 -> 2.5 V dòng âm. 2.5V -> 5V lá dòng dương.

-   Để đo dòng điện ACS712 ta sử dung ADC để đọc tín hiệu analog từ chân Vout của ACS712.

-  Công thức tính dòng điện:

+   Voltage0 = (data\_adc0 / 1023.0) \* 5; // biến đổi điện áp đọc được từ ADC

+   Ip0=(((Voltage0 + 2) – 2.5)\*1000)/100; //100 la do nhay dien ap 100mV/A

### **3.2.3** Cảm biến đọc thẻ NFC CR95HF

Loại hình NFC đang được ứng dụng hiện nay là thẻ nhận dạng NFC (NFC tag). Thẻ nhận dạng NFC có vai trò tương tự mã vạch hay mã QR. Thẻ NFC thường chứa dữ liệu chỉ đọc nhưng cũng có thể ghi đè được. Chúng có thể được tùy biến, mã hóa bởi nhà sản xuất hoặc sử dụng những thông số riêng do NFC

Với Thông số kỹ thuật:

* IC chính: CR95HF-VMD5T nhiều giao thức RF
* Anten PCB 46x34 mm.
* Giao tiếp SPI hoặc UART với MCU
* Chế độ hoạt động: Reader/Writer
* Giao tiếp NFC-RFID @13.56MHz
* ISO/IEC 14443 type A và B
* Khoảng cách giao tiếp trong tầm 10 cm

Nó phù hợp với phạm vi nghiên cứu mà đề tài đang thực hiện nên tôi chọn CR95HF làm thiết bị đọc thẻ.

**Bảng 3.1** Bảng mô tả chân của module RFID-NFC CR95HF

|  |  |  |  |
| --- | --- | --- | --- |
| Chân | Tên | Tín hiệu | Mô tả |
| 1 | VCC | VPS and VPS\_TX | Main power supply/power supply for RF drivers, 3.3÷5V |
| 2 | IRQ\_OUT | IRQ\_OUT and UART\_TX | UART transmit pin/interrupt output for CR95HF |
| 3 | GND | Ground | Ground |
| 4 | IRQ\_IN | IRQ\_IN and UART\_RX | UART receive pin/interrupt input for CR95HF |
| 5 | MISO | SPI\_MISO | SPI data, slave output |
| 6 | MOSI | SPI\_MOSI | SPI data, slave input |
| 7 | SCK | SPI\_SCK | SPI serial clock |
| 8 | SS | SPI\_SS | SPI slave select |

Sau đây là một số lệnh của CR95HF khi lập trình

* 0x01: Thông tin về CR95HF và phiên bản hiện tại.
* 0x02: Chọn giao thức RF giao tiếp và xác định một số thông số giao thức liên quan.
* 0x04: Gửi dữ liệu bằng cách sử dụng giao thức chọn trước đó và nhận được phản hồi.
* 0x07: Chuyển CR95HF vào mức tiêu thụ năng lượng thấp. Chờ trạng thái tiếp theo
* 0x0A:Thiết lập tốc độ truyền UART.
* …

**Bảng 3.2** Bảng danh sách các lệnh CR95HF

|  |  |  |
| --- | --- | --- |
| *Code* | *Command* | *Description* |
| 0x01 | IDN | Thông tin về CR95HF và phiên bản hiện tại. |
| 0x02 | PROTOCOL  SELECT | Chọn giao thức RF giao tiếp và xác định một số thông số giao thức liên quan |
| 0x04 | SENDRECV | Gửi dữ liệu bằng cách sử dụng giao thức chọn trước đó và nhận được phản hồi |
| 0x07 | IDLE | Chuyển CR95HF vào mức tiêu thụ năng lượng thấp. Chờ trạng thái tiếp theo |
| 0x08 | RDREG | Đăng ký sự kiện Wake-up hoặc Đăng ký Cấu hình Analog |
| 0x09 | WRREG | Ghi Analog cấu hình đăng ký (ARC\_B) đăng ký hoặc viết chỉ số của địa chỉ ARC\_B đăng ký. Ghi Timer Window (TimerW) |
| 0x0A | BAUDRATE | Thiết lập tốc độ truyền UART |
| 0x55 | ECHO | CR95HF trả về một hàm ECHO |
| Other codes | |  |

### 3.2.4 Mạch thu phát Wifi ESP8266

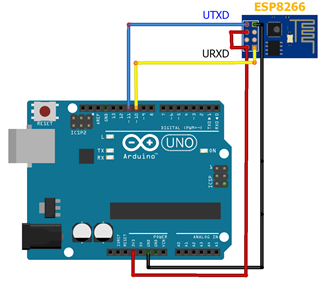
ESP8266 là một chip tích hợp cao - System on Chip (SoC), có khả năng xử lý và lưu trữ tốt, cung cấp khả năng vượt trội để trang bị thêm tính năng wifi cho các hệ thống khác hoặc đóng vai trò như một giải pháp độc lập.

Module wifi ESP8266 v12 cung cấp khả năng kết nối mạng wifi đầy đủ và khép kín, bạn có thể sử dụng nó để tạo một web server đơn giản hoặc sử dụng như một access point.

**Bảng 3.3** Cách kết nối ESP8266 với Arduino.

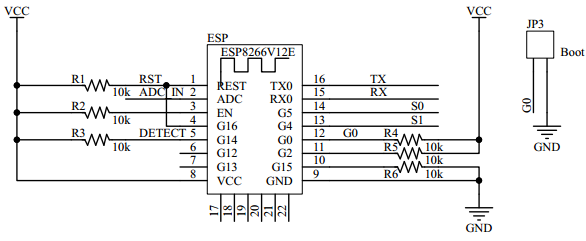
|  |  |
| --- | --- |
| ESP8266 | Arduino |
| VCC  GND  TX  RX | 3.3 V  GND  RX  TX |

Các chân GPIO 0, GIPO 2, GIPO PD được sử dụng để lựa chọn chế độ Boot cho chip, G2 và G0 được đặt mức cao, G15 được đặt mức thấp chip sẽ khởi động ở chế độ chạy bình thường, ngược lại chip sẽ khởi động ở chế độ Flash firmware khi đặt G0 ở mức thấp. Chân URXD(Rx) và UTXD(Tx) để truyền tín hiệu trong giao tiếp UART với Pro Mini.



**Hình 3.5** Kết nối giữa ESP8266 với Arduino

Có thể giao tiếp và truyền nhận dữ liệu giữa Esp8266 và Arduino thông qua chân Tx, Rx. ở đây tôi dùng chân 2,3 của Arduino làm Tx, Rx thông qua hàm SofwareSerial. Chân 2,3 sẽ được kết nối với chân Tx, Rx của Esp8266.



**Hình 3.6** Chip ESP8266

### 3.2.5 Giải thuật điều khiển phần cứng

### 

**Hình 3.7** Lưu đồ hoạt động của phần cứng

Khi đã kiểm tra nút đã nhấn hay không tức là thiết bị đã được cắm hay chưa. Nếu có thì Arduino sẽ lập tức đọc analog từ các cảm biến. Đọc thẻ NFC từ đó gán tên thiết bị được cắm theo ID đã cài đặt sẵn trên thẻ đồng thời tình trạng ổ cắm được gởi là đang “ON” tiếp đó đọc công suất đo được kể thì khi bật đến khi tắt thiết bị và tiếp theo truyền dữ liệu ESP qua Uart.



**Hình 3.8** Lưu đồ hoạt động của ESP

Trước tiên, ESP với một mạng Wifi để tham gia một client của Server, sau đó giao tiếp với Arduino bằng chân Tx và Rx để nhận chuỗi dữ liệu gởi qua nếu đúng “*tag gởi dữ liệu*” thì tiến hành gởi dữ liệu đến Server. Chờ nhận phản hồi thành công và nhận các lệnh điều khiển từ ứng dụng điện thoại gởi lên cho Server. Để quá trình thực thi nhanh chóng ESP phải gởi và nhận liên tục.



**Hình 3.9** Lưu đồ của NFC khi kiểm tra thẻ

Khi sử dụng NFC để cấu hình cho module hoạt động cần có các lệnh để cấu hình. 2 ID thẻ mà chúng tôi đọc là Card1="8802E200" và Card2="8804EB1D" là quạt được setup sằn khi đọc đúng mã thẻ thì tên thiết bị tương ứng với mã thẻ là “*bóng đèn*” hay “*quạt điện*”.

## 3.3 THIẾT KẾ ỨNG DỤNG ANDROID

### 3.3.1 Phân tích yêu cầu đặt ra

* Ứng dụng phải có khả năng cung cấp cho người sử dụng các công cụ trực quan dễ hiểu.
* Có chức năng đăng nhập bằng tài khoản người sử dụng.
* Đăng ký thông tin người sử dụng mới, bao gồm tên người sử dụng, email, tên tài khoản và mật khẩu.
* Thể hiện được các giá trị cảm biến mới nhất mà ổ điện thông minh thu thập được, đồng thời cũng cho biết tình trạng ổ cắm.
* Có thể hẹn giờ bật tắt ổ cắm do người sử dụng nhập vào

### 3.3.2 Xây dựng lưu đồ giải thuật

Để có thể triển khai trong thực tế, chúng tôi cần thiết lập hệ thống có khả năng quản lý người sử dụng rõ ràng, vì vậy mỗi người dùng được cung cấp một tài khoản riêng khi đăng nhập, chỉ có những người sử dụng thiết bị mới được cung cấp tài khoản.



**Hình 3.10** Lưu đồ quá trình đăng nhập ứng dụng

Sau khi đăng nhập thành công, ứng dụng sẽ hiển thị các chức năng ở dạng chuyển màn hình, có bốn thẻ được thiết kế:

* Màn hình 1 hiển thị công suất từng lỗ cắm.
* Màn hình 2 tình trạng từng lỗ cắm (Empty/Busy).
* Màn hình 3 cho phép điều khiển và hẹn giờ bật tắt ổ cắm.
* Màn hình 4 hiển thị tên đồ dùng được cắm vào.
* Màn hình 5 cho cho người dùng thay đổi mật khẩu.



**Hình 3.11** Lưu đồ hiển thị công suất của từng ổ cắm

Ở đây hiển thị cồng suất của thiết bị cũng như là điện năng tiêu thụ từ lúc thiết bị sử dụng.

Màn hình 2 tình trạng từng lỗ cắm (Empty/Busy).



**Hình 3.12** Lưu đồ hiển thị tình trạng của từng lổ cắm

Màn hình 3 cho phép hẹn giờ và bật tắt ổ cắm.



**Hình 3.13** Lưu đồ hẹn giờ đóng mở từng lỗ cắm

Màn hình 5 cho cho người dùng thay đổi mật khẩu



**Hình 3.14** Lưu đồ thay đổi password

* 1. **THIẾT KẾ WEB SEVER**

### Không gian lưu trữ và tên miền của Web server

Dữ liệu từ các cảm biến và máy đo được Chip ESP8266 thu thập và gửi lên mạng Internet, công việc tiếp theo là xây dựng một Web Server có khả năng nhận dữ liệu được gửi lên, xử lý và lưu trữ lại. Để thực hiện được quá trình này cần có một không gian lưu trữ trên mạng Internet để lưu dữ liệu và một tên miền để tham chiếu tới không gian đang sử dụng.

Hiện nay có rất nhiều nhà cung cấp dịch vụ Hosting, chỉ cần đăng kí một Hosting và Domain là chúng ta đã có một không gian lưu trữ trên mạng Internet. Các dịch vụ này có thể tính phí hoặc miễn phí, tùy thuộc vào các tiện ích và chất lượng gói dịch vụ chúng ta đăng kí sử dụng. Trong quá trình xây dựng và thử nghiệm đồ án, chúng tôi sử dụng dịch vụ Web Hosting của nhà cung cấp dịch vụ mạng https://www.000webhost.com/, tên miền được đăng kí miễn phí tại địa chỉ web http://www.dot.tk/. Sau khi đăng kí xong Hosting, ta đã có một không gian lưu trữ mã nguồn và có thể tạo cơ sở dữ liệu MySQL để lưu trữ thông tin.

* + 1. **Thiết kế cơ sở dữ liệu MySQL**

MySQL là hệ quản trị dữ liệu miễn phí, được tích hợp sử dụng chung với apache, PHP. Chính yếu tố phát triển trong cộng đồng mã nguồn mở nên MySQL đã qua rất nhiều sự hỗ trợ của những lập trình viên yêu thích mã nguồn mở. MySQL cũng có cùng một cách truy xuất và mã lệnh tương tự với ngôn ngữ SQL. Nhưng Mysql không bao quát toàn bộ những câu truy vấn cao cấp như SQL. Về bản chất MySQL chỉ đáp ứng việc truy xuất đơn giản trong quá trình vận hành của website nhưng hầu hết có thể giải quyết các bài toán trong PHP.

Cơ sở dữ liệu MySQL Server được chia làm 2 nhóm:

* Nhóm bảng chứa thông tin người dùng.
* Nhóm bảng chứa thông số của người sử dụng.

a) Nhóm bảng chứa thông tin người dùng.

Yêu cầu xây dựng các bảng có khả năng lưu trữ đầy đủ các thông tin của người dùng gồm tên tài khoản, mật khẩu, tên thật người sử dụng, ngày sinh và email người sự dụng.

Sau khi phân tích các yêu cầu đặt ra, bảng chứa thông tin người dùng được xây dựng như bảng dưới.

**Bảng 3.4** Cấu trúc dữ liệu của bảng users

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TEN\_TAI  \_KHOAN | MAT\_  KHAU | CREATED AT | FIRST NAME | LAST NAME | EMAIL |
| cha | 123 | 23/10/2016 | huynh | thuong | cha@gmail.com |
| … |  |  |  |  |  |

b) Nhóm bảng chứa thông số của người sử dụng

Mỗi bảng chưa 1 dữ liệu gồm email, công suất ổ (1;2;3), tên thiết bị đang sử dụng, trạng thái on/off của từng ổ, thời gian bật/tắt từng ổ điện…

**Bảng 3.5** Cấu trức dữ liệu của bảng người dùng

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Email | Kw1 | On/Off1 | Start1  Stop1 | Dvname1 | Command1 | Kw2 | On/Off2 |  |
| cha@gmail.com | 35W | On | 6:35-  9:50 | Quạt | Bật/Tắt | 50W | Off |  |

Lưu đồ tổng quan mối quan hệ giữa Server và các phần còn lại hệ thống



**Hình 3.15** Lưu đồ mối quan hệ của Sever và hệ thống

Trong ứng dụng cần thiết phải xậy dựng một cơ sở dữ liệu tương tự trên Sever để giúp giảm bớt khối lượng dữ truyền tải giữa Sever và ứng dụng. Các thông tin người dùng và tin thông báo được đồng bộ với Sever, sau đó lưu trữ trên điện thoại, khi ứng dựng được khởi động lại sẽ không cần thiết phải tải xuống. Cơ sở dữ liệu được dùng trên điện thoại là SQLite, có cấu trúc tương tự MySQL.

Ứng dụng trên điện thoại android chủ yếu thực hiện ba request tới Server dưới dạng chuỗi JSON, đây là một dạng dữ liệu tuân theo một quy luật nhất định mà hầu hết các ngôn ngữ lập trình hiện nay đều có thể đọc được. Trong phần thiết kế ứng dụng trên điện thoại Android ta sẽ đi sâu hơn về cấu trúc của một chuỗi JSON, các API hỗ trợ định dạng và tách dữ liệu từ chuỗi JSON.

Các Request bao gồm:

* Yêu cầu đăng nhập, yêu cầu này của ứng dụng cũng là một http post method request như quá trình tải dữ liệu của thiết bị đeo. Server sẽ xác minh sự tồn tại của tài khoản và chả về cho ứng dụng thông tin tưng ứng của người dùng sau khi nhận tiếp một Request tải thông tin người dùng sau đó.
* Yêu cầu tải dữ liệu mới nhất, một http post method request chứa thông tin về số lượng hàng của cở sở dữ liệu trên ứng dụng được gửi tới Server, Server sẽ so sánh và gửi lại tất cả những dữ liệu mới nhất trở lại cho ứng dụng.
* Yêu cầu tải lên dữ liệu lên Server khi người sử dụng muốn điều khiển ổ điện. Tương tự như các Request trên, tuy nhiên có sự khác biệt trong cách xử lí của Server đối với hai request trên, thay vì đọc dữ liệu từ cơ sở dữ liệu và gửi xuống cho ứng dụng, yêu cầu tải lên dữ liệu được Server tách lấy các dữ liệu kèm theo và lưu vào cơ sở dữ liệu trên Server.

# Chương 4

# KẾT QUẢ VÀ ĐÁNH GIÁ

Như đã trình bày ở phần trước, nhiệm vụ chính của luận văn là thực hiện “Hệ thống ổ điện thông minh” với các phần như sau:

* Thiết bị đo các thông số cảm biến và điều khiển đóng mở relay (ba cảm biến đo dòng, cảm biến đọc thẻ NFC).
* Ứng dụng theo dõi và điều khiển đóng cắt từng lỗ cắm trên hệ điều hành Android.
* Sever lưu trữ dữ liệu.
* Web application là giao diện web của ứng dụng.

Để đánh giá kết quả của luận văn, chúng tôi đã tổng kết lại kết quả thông qua từng thành phần của hệ thống, cùng với đó là kết quả tổng quan trong quá trình thi công.

## 4.1 KẾT QUẢ PHẦN CỨNG

**Có thể đo được công suất của thiết bị sử dụng, thông qua đó có thể nắm được điện năng tiêu thụ để điều chỉnh sinh hoạt cho hợp lý, hay có thể quản lý được hiện tượng quá tải của thiết bị nhằm bảo vệ thiết bị.**

**Có thể nhận biết được loại thiết bị đang sử dụng qua đó có thể cài chế độ tự động hẹn giờ cho từng thiết bị thông qua hệ thống NFC, giúp người dùng hẹn giờ tự động cho thiết bị một cách tiện lợi nhất mà không cần dùng APP Android.**



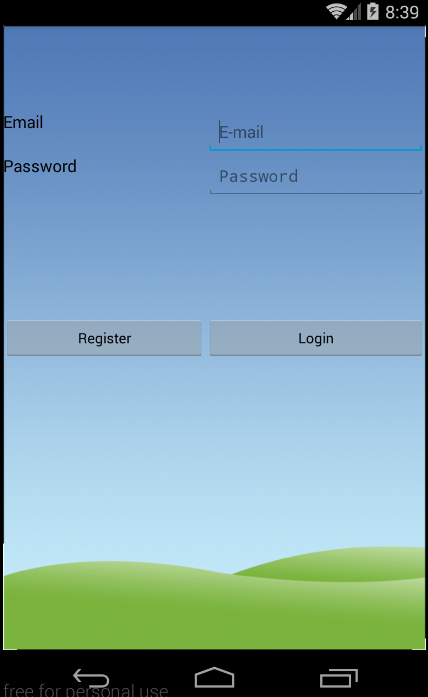
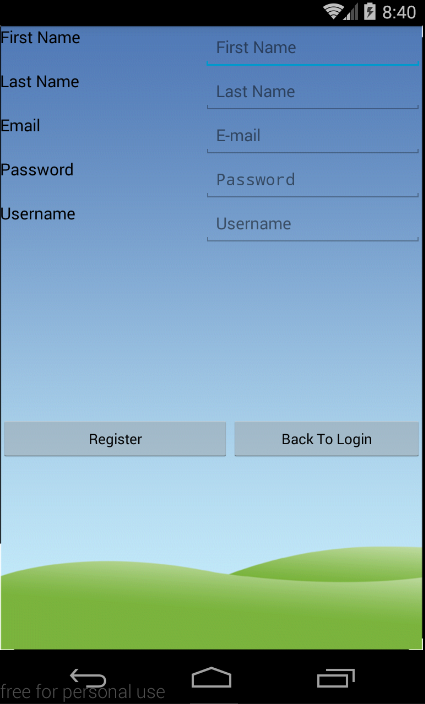
**Hình 4.1** Mô hình ổ điện thông minh

## 4.2 KẾT QUẢ PHẦN MỀM

### 4.2.1 Ứng dụng ổ diện thông minh

Ở tại màn hình đăng nhập có 2 Edit text cho phép nhập *Email* và *Password*.

Khi nhấn nút *Login* thì Email và Password sẽ được so sánh với thông số mà người dùng đã đắng ký trước trong Sever nếu đúng thì chuyển sang màn hình MainActivity.xml. Ngoài ra còn có nút đăng ký khi đó những thông số người dùng nhập vào sẽ được thêm vào CSDL.

**Hình 4.2** Màn hình Login và Register.

Tiếp theo khi đăng nhập thành công hiên ra màn hình MainActivity.xml hiện ra

Ở đây có 4 *Imagebutton* có chức năng:

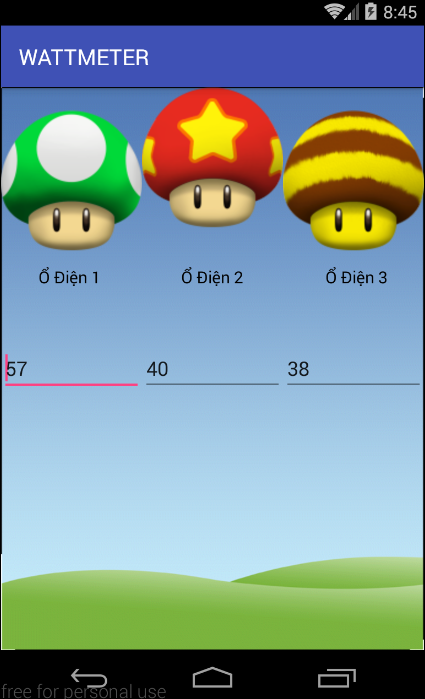
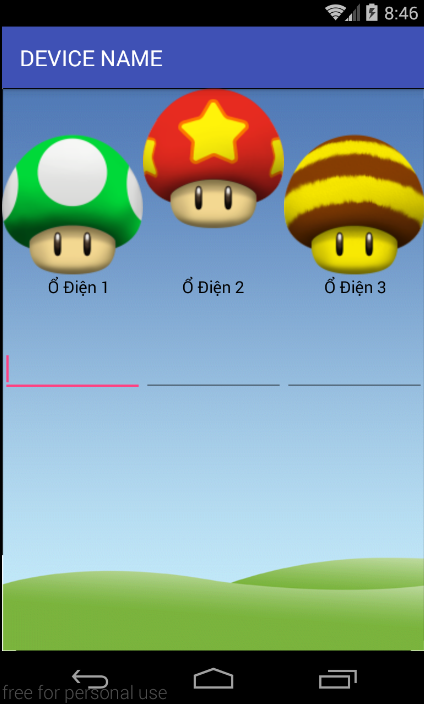
* Hiện thị công suất
* Đo lượng điện tiêu thụ của thiết bị từ khi bật đến lúc tắt
* Tình trạng từng lỗ cắm
* Hẹn giờ bật tắt thiết bị
* Thay đổi mật khẩu của người dùng ổ điện thông minh
* Tên thiết bị cắm vào lỗ (sử dụng NFC)



**Hình 4.3** Màn hình MainActivity

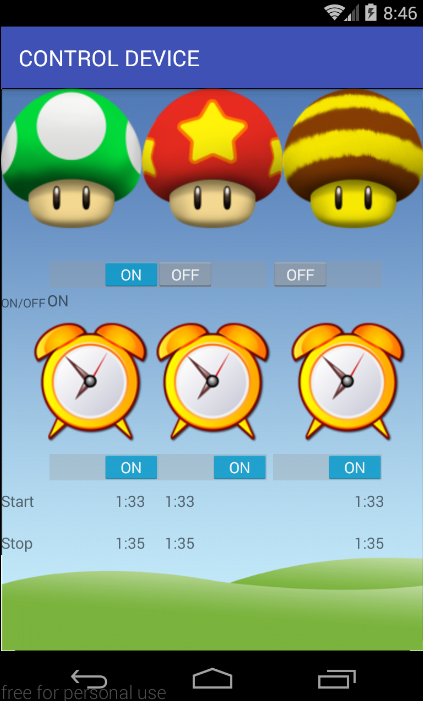
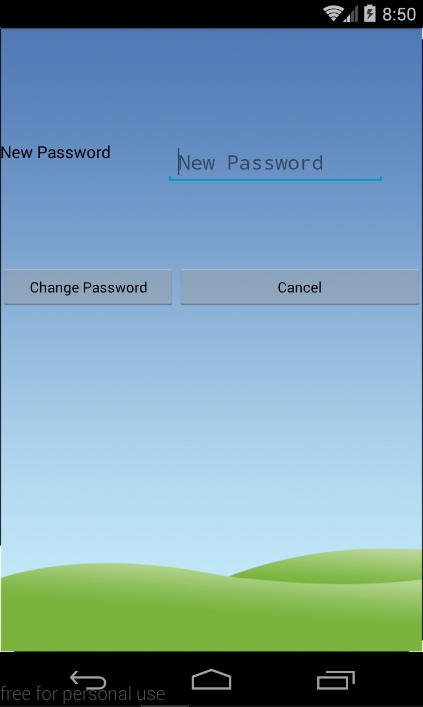
Sau màn hình Login, màn hình MainActivity tại đây ta có thể xem công suất của thiết bị và công suất tiêu thụ từ khi bật đến khi tắt thiết bị đi tại *“Wattmetter”* Cho phép bật tắt thiết bị điện nằm ở *“Control Device”* hay nhận diện thiết bị ở trên *“What device?”.* Trạng thái ổ điện đóng hay mở thì ở tại *“Status Device” ,*

*“Change Password”* để đổi mật khẩu.

**Hình 4.4** Màn hình đọc công suất và hiện thị tên thiết bị

Giao diện hiển thị công suất và tên thiết bị sử dụng trên ứng dụng. Đơn vị là Kwh trên *“Wattmeter”* và *“Device name”*

**Hình 4.5** Màn hình hẹn giờ và Change password

Ở màn hình “Control Device’ có các switch để đóng cắt các ổ điện hay cũng như cho phép hẹn giờ. Phần “Change password’ cho dùng để Update mật khẩu người dùng mới trên cơ sở dữ liệu.

Qua quá trình chạy thực nghiệm, ứng dụng *ổ điện thông minh* trên điện thoại android hoạt động tốt, ổn định trong thời gian dài, giao diện thân thiện dễ sử dụng. Điều này chứng tỏ các thành phần của hệ thống hoạt động tốt và đạt được các yêu cầu thiết kế đã đặt ra. Tuy nhiên phần giao diện có tính thẩm mĩ chưa cao, các thành phần hiển thị còn khá thô và thiếu tính uyển chuyển.

### 4.2.2 Web Application

Một file giao diện Web bằng ngôn ngữ html cho để giúp xem thông số ổ điện mà không cần thông qua Android.

Được đăng ký tên miền là [*http://www.datn.coxma.com*](http://www.datn.coxma.com)

**

**Hình 4.6** Giáo diện Web Application

# Chương 5

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## 5.1 KẾT LUẬN

### 5.1.1 Những vấn đề nghiên cứu

Sau quá trình thực hiện luận văn tốt nghiệp về đề tài “*Thiết kế* *Ổ điện thông minh*”, các mục tiêu đặt ra đều đã được hoàn thành tốt. Ngoài ra, chúng tôi còn thu được một số kinh nghiệm nhất định về lập trình web, lập trình android, thiết kế một sản phẩm thực tế và các bước xây dựng một hệ thống IoT đơn giản.

Một số các ưu điểm của “Thiết kế Ổ điện thông minh”:

* Cơ bản đã thiết kế được thiết bị hoàn chỉnh, hoạt động tương đối tốt.
* Thông tin của người sử dụng được cập nhật nhanh và liên tục.
* Ứng dụng trên điện thoại trực quan, dễ sử dụng, cung cấp nhiều thông tin hữu ích.
* Cho phép đăng ký được nhiều người sử dụng.
* Biết công suất thiết bị đang dùng và lượng điện năng tiêu thụ khi hoạt động
* Có thể điều khiển thiết bị từ xa thông qua App
* Cho phép người dùng nhận dạng thiết bị đang dùng
* Có thể hẹn giờ bật tắt thiết bị chính xác

### Những hạn chế của đề tài

Tuy nhiên, do thời gian và kinh phí thực hiện đề tài có hạn, sản phẩm của nhóm không tránh khỏi những khuyết điểm như:

* Các thành phần được thiết kế còn khá thô, tính thẩm mĩ chưa cao, chưa thu hút.
* Hệ thống chưa có tính bảo mật.
* Hệ thống hoạt động chưa ổn định.
* Tình trạng treo của Server online khi sử dụng nhiều giờ do đăng ký domain miễn phí.
* Độ chính xác phụ thuộc vào các module có sẵn.
* Không chống nước.

**5.2 HƯỚNG PHÁT TRIỂN CỦA ĐỀ TÀI**

Hệ thống giám sát sức khỏe thông minh được nhóm xây dựng với hi vọng có thể giúp giải quyết phần nào các vấn đề nêu trên. Để có thể thực hiện được những nhu cầu thực tế trên đề tài cần có những cải tiến như:

* Tăng thêm nhiều cảm biến cho thiết bị.
* Xây hệ thống khách hàng với nhiều hơn nữa các ổ điện thông minh.
* Thêm hệ thống học để xử lý thông tin và đưa ra các cảnh báo.
* Nhận dạng nhiều thiết bị cùng một lúc và có thể tự động hẹn giờ.
* Nhắn tin cảnh báo tới điện thoại cá nhân khi sảy sự cố.
* Cho phép giám sát nhiều ổ điện kết nối tới Web server.

# PHỤ LỤC A MÃ NGUỒN CHƯƠNG TRÌNH

***Mã nguồn Arduino Pro Mini***

#include <stdio.h>

#include <SoftwareSerial.h>

#include <Filters.h>

SoftwareSerial swSer(2, 3);//2RX

SoftwareSerial mySerial(4, 5); // RX, TX

#define Status1 6

#define Status2 7

#define Status3 8

#define Relay1 10

#define Relay2 11

#define Relay3 12

char setPro[]={0x02,0x02,0x02,0x00};

char senrecv[]={0x04,0x02,0x26,0x07};

String ID="";

char IDCard[]={0x80,0x5,0x4,0x0,0x28,0,0};

String sendRecv();

float kwh1=2.0,kwh2=3.0,kwh3=4.0;

String stt1,stt2,stt3;

String reply;

String dv1,dv2="None",dv3="None";

int socket1,socket2,socket3;

// Wattmeter

float testFrequency = 50.0;

float windowLength = 20.0/testFrequency;

float KWH\_1 = 0.0;

float KWH\_2 = 0.0;

float KWH\_3 = 0.0;

float intercept = -0.116;

float slope = 0.04;

float currentAMP1=0.0, currentAMP2=0.0,currentAMP3=0.0;

unsigned long printPeriod = 1000;

unsigned long previousMillis = 0;

unsigned long time1=0;

unsigned long time2=0;

unsigned long time3=0;

unsigned long time4=0;

unsigned long time5=0;

unsigned long time6=0;

boolean toogle1 = false;

boolean toogle2 = false;

boolean toogle3 = false;

void setup() {

Serial.begin(9600);

swSer.begin(9600);

mySerial.begin(57600);

delay(500);

pinMode(Status1, INPUT\_PULLUP);

pinMode(Status2, INPUT\_PULLUP);

pinMode(Status3, INPUT\_PULLUP);

pinMode(Relay1, OUTPUT);

pinMode(Relay2, OUTPUT);

pinMode(Relay3, OUTPUT);

digitalWrite(Relay1, LOW);

digitalWrite(Relay2, LOW);

digitalWrite(Relay3, LOW);

delay(100);

while(mySerial.available()>0) mySerial.read(); // cho phép đọc sạch bộ nhớ

while(swSer.available()>0) swSer.read();

//Serial.println("END setup");

}

void loop() {

// NFC

mySerial.listen();

if(mySerial.isListening()){

String ID = sendRecv();

if(ID.equals(IDCard))

{

dv1 = "LOAD";

}

Serial.println(dv1);

}

// Status && wattmeter

RunningStatistics inputStats;

inputStats.setWindowSecs( windowLength );

socket1 = digitalRead( Status1 );

socket2 = digitalRead( Status2 );

socket3 = digitalRead( Status3 );

if (socket1 == 0)

{

delay(20);

if (socket1 == 0){

if ( toogle1 == false )

{

time1 = millis();

stt1 = "Busy";

toogle1 = true;

}

}

}

else //nếu chưa nhấn

{

stt1 = "Empty1";

if( toogle1 == true )

{

time2 = millis();

unsigned long time12 = (unsigned long)(time1 - time2);

KWH\_1 = analogRead(A0);

inputStats.input(KWH\_1);

currentAMP1 = (intercept + slope \* inputStats.sigma());

kwh1 = (currentAMP1 \* 220.0)\*(time12 / (2.778/100000000)) ;

toogle1 = false;

}

}

if (socket2 == 0)

{

delay(20);

if (socket2 == 0){

if ( toogle2 == false )

{

time3 = millis();

stt2 = "Busy";

toogle2 = true;

}

}

}

else { //nếu chưa nhấn

stt2 = "Empty2";

if( toogle2 == true ){

time4 = millis();

unsigned long time34 = (unsigned long)(time3 - time4);

KWH\_2 = analogRead(A1);

inputStats.input(KWH\_2);

currentAMP2 = (intercept + slope \* inputStats.sigma());

kwh2 = (currentAMP2 \* 220.0) \*(time34 / (2.778/100000000));

toogle2 = false;

}

}

if(socket3 == 0) {

delay(20);

if(socket3 == 0) {

if ( toogle3 == false )

{

time5 = millis();

stt3 = "Busy";

toogle3 = true;

}

}

}

else { //nếu chưa nhấn

stt3 = "Empty3";

if( toogle3 == true ){

time6 = millis();

unsigned long time56 = (unsigned long)(time5 - time6);

KWH\_3 = analogRead(A2);

inputStats.input(KWH\_3);

currentAMP3 = (intercept + slope \* inputStats.sigma());

kwh3 = (currentAMP3 \* 220.0) \*(time56 / (2.778/100000000));

toogle3 = false;

}

}

// ESP8266

swSer.listen();

if(swSer.isListening())

{

);

//swSer.print(buf);

swSer.print("tag=espup&email=cha@gmail.com&kwh1=");swSer.print(kwh1);

swSer.print("&stt1=");swSer.print(stt1);

swSer.print("&dvname1=");swSer.print(dv1);

swSer.print("&kwh2=");swSer.print(kwh2);

swSer.print("&stt2=");swSer.print(stt2);

swSer.print("&dvname2=");swSer.print(dv2);

swSer.print("&kwh3=");swSer.print(kwh3);

swSer.print("&stt3=");swSer.print(stt3);

swSer.print("&dvname3=");swSer.print(dv3);

swSer.print("\n");

while(swSer.available()<=0);

String reply=swSer.readStringUntil('\n');

delay( 200);

Serial.println(reply);

// Xác định tín hiệu ON/OFF điều khiển relay

while(swSer.available()>0) swSer.read();

if(reply.indexOf("\"onoff1\":\"ON\"")!=-1){ // tìm kiếm có chuỗi onoff1=ON từ Android gửi về

//Serial.println( "ON1" );

digitalWrite( Relay1, HIGH);

}else if(reply.indexOf("\"onoff1\":\"OFF\"")!=-1){

//Serial.println( "OFF1" );

digitalWrite( Relay1, LOW);

}

if(reply.indexOf("\"onoff2\":\"ON\"")!=-1){

digitalWrite( Relay2, HIGH);

//Serial.println( "ON2" );

}else if(reply.indexOf("\"onoff2\":\"OFF\"")!=-1){

digitalWrite( Relay2, LOW);

//Serial.println( "OFF2" );

}

if(reply.indexOf("\"onoff3\":\"ON\"")!=-1){

digitalWrite( Relay3, HIGH);

//Serial.println( "ON3" );

}else if(reply.indexOf("\"onoff3\":\"OFF\"")!=-1){

//Serial.println( "OFF3" );

digitalWrite( Relay3, LOW);

}

}

}

/\*

String getTag(String mbody,String tmptag){

if(mbody.length()<=0 || tmptag.length()<=0) return "";

String mtag= "\"" + tmptag + "\":";

int start=mbody.indexOf(mtag);

if(start==-1) return "";

for(int i=mbody.length();i>=0;i--){

if(mbody.charAt(i)=='\n' || mbody.charAt(i)=='\r' || mbody.charAt(i)=='}')

mbody.remove(i);

}

return mbody.substring(start+mtag.length(),mbody.indexOf(',',start));

}

\*/

String sendRecv(){

for( int i=0; i< sizeof(senrecv); i++) // gửi respone để đọc thẻ

{

mySerial.write( senrecv[i]);

}

delay(100);

String ID="";

while(mySerial.available()>0)

{

char r= mySerial.read();

ID+= r;

}

return ID;

}

***Mã nguồn ESP8266***

**#**include <ESP8266WiFi.h>

#include <ESP8266HTTPClient.h>

#include <SoftwareSerial.h>

SoftwareSerial swSer(14, 12);//r,t

#define LED 2

const char\* ssid = "Messi";

const char\* password = "10122010";

const char\* host = "192.168.1.69";

const int port = 80;

const char\* uri = "/nguyen.php";

void setup() {

Serial.begin(9600);

delay(10);

swSer.begin(9600);

pinMode(LED,OUTPUT);

digitalWrite(LED,1);

//Serial.print("\nConnecting to ");

//Serial.println(ssid);

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED) {

delay(500);

}

digitalWrite(LED,0);

//Serial.println("OK");

}

void loop() {

ESP.wdtDisable();

String cmd = "tag=espup&kwh1=111";

if(cmd.indexOf("tag")!=-1)

{

String r = sendForResponse(host,port,uri,cmd);

Serial.println(r);

//swSer.printf("TRAN:%s\n",r.c\_str());

}

}

String sendForResponse(String mhost, int mport, String muri,String mbody){

HTTPClient http;

http.begin(mhost,mport,muri); //HTTP

http.addHeader("Host",host);

http.addHeader("Content-Type","application/x-www-form-urlencoded");

http.addHeader("Accept","\*/\*");

http.addHeader("Connection","close");

int httpCode = http.POST(mbody); // Gửi dữ liệu đi

if(httpCode > 0) {

return http.getString(); //Dữ liệu trả về từ web ( thuong.php)

//Serial.println(http.getString());

}

return "";

}

# 

# TÀI LIỆU THAM KHẢO

1. S. S. Bhunia, S. K. Dhar and N. Mukherjee, "iHealth: A fuzzy approach for provisioning intelligent health-care system in smart city," *2014 IEEE 10th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Larnaca, 2014, pp. 187-193.
2. J. Gubbi et al, “Internet of Things (IoT): A vision, architectural elements,  
   and future directions”, Future Generation Computer Systems Vol. 29,  
   No.7, 2013, pp. 1645-1660.
3. K. Ullah, M. A. Shah and S. Zhang, "Effective ways to use Internet of Things in the field of medical and smart health care," *2016 International Conference on Intelligent Systems Engineering (ICISE)*, Islamabad, 2016, pp. 372-379.
4. R. R. Fletcher, M. Z. Poh and H. Eydgahi, "Wearable sensors: Opportunities and challenges for low-cost health care," *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, Buenos Aires, 2010, pp. 1763-1766.