

**BỘ GIÁO DỤC & ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH  
KHOA ĐIỆN – ĐIỆN TỬ  
BỘ MÔN ĐIỆN TỬ CÔNG NGHIỆP – Y SINH**  
-----



# **ĐỒ ÁN TỐT NGHIỆP**

**NGÀNH CÔNG NGHỆ KỸ THUẬT ĐIỆN TỬ TRUYỀN THÔNG**  
**ĐỀ TÀI:**

**THIẾT KẾ, THI CÔNG MÔ HÌNH  
HỆ THỐNG ĐIỀU KHIỂN THIẾT BỊ ĐIỆN  
VÀ GIÁM SÁT NHÀ**

**GVHD: PGS. TS. NGUYỄN THANH HẢI**

**SVTH: Nguyễn Ngọc Lực**

**MSSV: 14141183**

**Tp. Hồ Chí Minh - 7/2018**

Tp. HCM, ngày 16 tháng 7 năm 2018

# NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên:	Nguyễn Ngọc Lục	MSSV:	14141183
Chuyên ngành:	Kỹ thuật Điện tử Truyền thông	Mã ngành:	41
Hệ đào tạo:	Đại học chính quy	Mã hệ:	1
Khóa:	2014	Lớp:	14141DT

I. TÊN ĐỀ TÀI: THIẾT KẾ, THI CÔNG MÔ HÌNH HỆ THỐNG ĐIỀU KHIỂN THIẾT BỊ ĐIỆN VÀ GIÁM SÁT NHÀ

## II. NHIỆM VỤ

### 1. Các số liệu ban đầu:

- [1] Nguyễn Văn Lem, “Thiết kế và xây dựng hệ thống mô hình nhà thông minh”, Đồ án tốt nghiệp, trường ĐHSPKT Tp.HCM, 2015.
- [2] Nguyễn Văn Hiệp – Đinh Quang Hiệp 2014, “Lập trình Android cơ bản”, Đại Học SPKT HCM.
- [3] Espressif Systems IOT Team, “ESP8266 Datasheet”, Espressif Systems, 2015.

## 2. Nội dung thực hiện:

- Tìm hiểu nhu cầu thực tiễn cũng như công nghệ của đề tài.
- Tìm hiểu lý thuyết liên quan, các giải pháp thiết kế hệ thống, thi công mô hình điều khiển các thiết bị điện.
- Tính toán và thiết kế hệ thống điều khiển.
- Thiết kế mô hình, giải pháp điều khiển thiết bị, lập trình cho hệ thống.
- Đánh giá kết quả đã thực hiện được và tiến tới hoàn thiện đề tài.
- Kết luận về đề tài, đưa ra hướng phát triển trong tương lai.

III. NGÀY GIAO NHIỆM VỤ: 20/3/2018

IV. NGÀY HOÀN THÀNH NHIỆM VỤ: 15/7/2018

V. HO VÀ TÊN CÁN BỘ HƯỚNG DẪN: PGS.TS. Nguyễn Thanh Hải

## CÁN BỘ HƯỚNG DẪN

BM. ĐIÊN TỬ CÔNG NGHIỆP – Y SINH

## LỊCH TRÌNH THỰC HIỆN ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên: Nguyễn Ngọc Lực  
Lớp: 14141DT2B

MSSV:14141183

Tên đề tài: THIẾT KẾ, THI CÔNG MÔ HÌNH HỆ THỐNG ĐIỀU KHIỂN THIẾT BỊ ĐIỆN VÀ GIÁM SÁT NHÀ

<i><b>Tuần</b></i>	<i><b>Nội dung</b></i>	<i><b>Xác nhận GVHD</b></i>
<b>Tuần 1</b>	Tìm hiểu công nghệ của đề tài đang thực hiện cũng như các lý thuyết liên quan.	
<b>Tuần 2</b>	Tính toán và thiết kế sơ đồ khối của hệ thống.	
<b>Tuần 3</b>	Tính toán và thiết kế sơ đồ nguyên lý của các mạch cần trong hệ thống, lựa chọn linh kiện cần thiết cho hệ thống.	
<b>Tuần 4</b>	Lập trình hệ thống báo động trên board điều khiển trung tâm Arduino.	
<b>Tuần 5</b>	Nghiên cứu, thiết kế website điều khiển.	
<b>Tuần 6</b>	Lập trình cho mô đun Wifi ESP8266 Node MCU.	
<b>Tuần 7</b>	Nghiên cứu, thiết kế phần mềm Android.	
<b>Tuần 8</b>	Chạy thử nghiệm hệ thống trên testboard, kit.	
<b>Tuần 9-10</b>	Vẽ mạch in, thi công mạch in hệ thống. Lắp ráp và kiểm tra	
<b>Tuần 11</b>	Thiết kế mô hình.	
<b>Tuần 12</b>	Đóng gói hệ thống và chạy thử nghiệm.	
<b>Tuần 13- Tuần 16</b>	Chạy thử nghiệm và cân chỉnh toàn hệ thống. Đánh giá kết quả đạt được, viết báo cáo.	

GV HƯỚNG DẪN

# LỜI CAM ĐOAN

Đề tài này là do chúng tôi tự thực hiện dựa vào một số tài liệu trước đó và không sao chép từ tài liệu hay công trình đã có trước đó.

Người thực hiện đề tài

**Nguyễn Ngọc Lực**

# LỜI CẢM ƠN

Nhóm chúng em xin gửi lời cảm ơn chân thành đến các Thầy Cô trong Khoa Điện - Điện Tử. Trong thời gian học tại trường, Thầy Cô đã tận tình dạy bảo, truyền đạt cho nhóm em kiến thức, kinh nghiệm và động lực trong quá trình học tập.

Em xin gửi lời cảm ơn sâu sắc đến Thầy Nguyễn Thanh Hải đã trực tiếp hướng dẫn và tận tình giúp đỡ tạo điều kiện để hoàn thành tốt đề tài. Trong quá trình thực hiện đồ án, Thầy luôn tạo điều kiện và có những hướng dẫn giúp chúng em làm việc khoa học và hiệu quả; giúp chúng em không những hoàn thành tốt đồ án này mà còn có những trải nghiệm, hình thành dần tác phong nghề nghiệp cho sau này.

Em cũng gửi lời đồng cảm ơn đến các bạn lớp 14141DT đã chia sẻ trao đổi kiến thức cũng như những kinh nghiệm quý báu trong thời gian thực hiện đề tài.

Cảm ơn đến cha mẹ.

Xin chân thành cảm ơn!

Người thực hiện đề tài

**Nguyễn Ngọc Lực**

# MỤC LỤC

Trang bìa .....	i
Nhiệm vụ đồ án .....	ii
Lịch trình .....	iii
Lời cam đoan.....	iv
Lời cảm ơn .....	v
Mục lục.....	vi
Liệt kê hình vẽ.....	ix
Liệt kê bảng.....	xi
Tóm tắt .....	xii
<b>Chương 1. TỔNG QUAN .....</b>	<b>1</b>
1.1 Đặt vấn đề.....	1
1.2 Mục tiêu.....	2
1.3 Nội dung nghiên cứu .....	2
1.4 Giới hạn .....	2
1.5 Bố cục.....	2
<b>Chương 2. CƠ SỞ LÝ THUYẾT.....</b>	<b>4</b>
2.1 Tổng quan về Internet of Things.....	4
2.1.1 Giới thiệu về Internet of Things.....	4
2.1.2 Ứng dụng của IoT .....	4
2.1.3 Đặc điểm công nghệ IoT.....	4
2.2 Tổng quan về mạng WiFi.....	5
2.3 Giới thiệu về Website, Webserver và PHP .....	5
2.3.1. Giới thiệu về Website .....	5
2.3.2 Máy chủ Web (Webserver).....	5
2.3.3 Giới thiệu PHP .....	6
2.4 Giới thiệu về tin nhắn SMS.....	6
2.5 Giới thiệu về Android .....	7
2.6 Giới thiệu phần cứng.....	7

2.6.1 Vi điều khiển.....	7
2.6.2 Cảm biến nhiệt, độ ẩm DHT11 .....	10
2.6.3 Mô đun cảm biến khí gas MQ2.....	11
2.6.4 Mô đun cảm biến chuyển động PIR HC – SR501.....	12
2.6.5 Mô đun WiFi ESP8266 Node MCU .....	13
2.6.6 Mô đun SIM900A .....	15
2.6.7 Giới thiệu LCD 16x2 .....	15
2.6.8 Giới thiệu OPTO PC817 .....	17
2.7 Chuẩn truyền thông UART .....	17
2.8 Phương thức GET .....	19
<b>Chương 3. TÍNH TOÁN VÀ THIẾT KẾ .....</b>	<b>21</b>
3.1 Giới thiệu.....	21
3.2 Tính toán và thiết kế hệ thống.....	21
3.2.1 Thiết kế sơ đồ khối hệ thống.....	21
3.2.2 Tính toán và thiết kế mạch .....	24
3.2.3 Sơ đồ nguyên lý của toàn mạch .....	39
<b>Chương 4. THI CÔNG HỆ THỐNG.....</b>	<b>40</b>
4.1 Giới thiệu.....	40
4.2 Thi công hệ thống.....	40
4.2.1 Mạch in lớp dưới .....	40
4.2.2 Mạch in lớp trên.....	41
4.2.3 Sơ đồ bố trí linh kiện.....	42
4.3 Thi công mô hình .....	44
4.4 Lập trình hệ thống .....	45
4.4.1 Lưu đồ giải thuật .....	45
4.4.2 Phần mềm lập trình cho Android .....	52
4.4.3 Phần mềm lập trình cho website .....	55
4.4.4 Phần mềm lập trình cho Arduino và Node MCU.....	58
4.5 Tài liệu hướng dẫn sử dụng .....	59
<b>Chương 5. KẾT QUẢ_NHẬN XÉT_ĐÁNH GIÁ .....</b>	<b>61</b>

5.1 Kết quả .....	61
5.2 Nhận xét, đánh giá .....	68
<b>Chương 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN .....</b>	<b>70</b>
6.1 Kết luận .....	70
6.2 Hướng phát triển.....	70
TÀI LIỆU THAM KHẢO.....	71
DANH SÁCH TỪ VIẾT TẮT .....	732
PHỤ LỤC.....	73



## LIỆT KÊ HÌNH VẼ

Hình	Trang
Hình 2.1 Board Arduino mega 2560 .....	8
Hình 2.2 Mô đun cảm biến DHT11 .....	10
Hình 2.3 Mô đun cảm biến khí gas MQ2 .....	11
Hình 2.4 Mô đun cảm biến chuyển động HC SR501 .....	12
Hình 2.5 Sơ đồ chân cảm biến HC SR501 .....	13
Hình 2.6 Sơ đồ chân Node MCU .....	14
Hình 2.7 Hình ảnh mô đun SIM900A .....	15
Hình 2.8 Sơ đồ chân LCD 16x2 .....	16
Hình 2.9 Opto PC817 .....	17
Hình 2.10 Giản đồ truyền dữ liệu UART .....	19
Hình 3.1 Sơ đồ khối hệ thống.....	22
Hình 3.2 Sơ đồ nguyên lí mạch điều khiển board Arduino .....	25
Hình 3.3 Sơ đồ nguyên lí nối dây mô đun Node MCU .....	26
Hình 3.4 Sơ đồ nối dây mô đun SIM900A.....	27
Hình 3.5 Sơ đồ nguyên lí khối cảm biến .....	28
Hình 3.6 Sơ đồ nguyên lí mạch kích Relay .....	29
Hình 3.7 Đặc tuyến điện áp ngõ ra $V_{CE}$ và dòng phân cực $I_F$ .....	30
Hình 3.8 Mạch nguyên lí kích hoạt Relay .....	30
Hình 3.9 Giao diện quản lí website sau khi đăng nhập .....	32
Hình 3.10 Giao diện website đã thiết kế trong DATN .....	33
Hình 3.11 Sơ đồ khối cấu tạo của hệ thống website .....	34
Hình 3.12 Sơ đồ hoạt động của ứng dụng Android.....	35
Hình 3.13 Sơ đồ nguyên lí mạch nguồn .....	37
Hình 4.1 Mạch in lớp dưới .....	40
Hình 4.2 Mạch in lớp trên .....	41
Hình 4.3 Sơ đồ bố trí linh kiện .....	42
Hình 4.4 Mô hình DATN .....	44
Hình 4.5 Lưu đồ mạch điều khiển Arduino.....	47
Hình 4.6 Lưu đồ chương trình con xử lí mất điện.....	47

<b>Hình 4.7</b>	Lưu đồ chương trình con nhận trạng thái đèn từ Node MCU .....	48
<b>Hình 4.8</b>	Lưu đồ chương trình con xử lý tin nhắn đến.....	49
<b>Hình 4.9</b>	Lưu đồ gửi tin nhắn phản hồi .....	50
<b>Hình 4.10</b>	Lưu đồ chương trình cho Node MCU .....	51
<b>Hình 4.11</b>	Giao diện phần mềm Android Studio.....	52
<b>Hình 4.12</b>	Giao diện ứng dụng Android trong ĐATN .....	54
<b>Hình 4.13</b>	Vị trí 2 file MainActivity.java và activity_main.xml .....	54
<b>Hình 4.14</b>	Tạo một file php mới .....	56
<b>Hình 4.15</b>	Giao diện ứng dụng DW .....	56
<b>Hình 4.16</b>	Mô tả board mạch hệ thống.....	60
<b>Hình 5.1</b>	Board mạch chính điều khiển.....	62
<b>Hình 5.2</b>	Mô hình ĐATN .....	63
<b>Hình 5.3</b>	Trạng thái các đèn, giá trị nhiệt độ và độ ẩm hiển thị trên LCD .....	64
<b>Hình 5.4</b>	Giao diện website điều khiển .....	64
<b>Hình 5.5</b>	Giao diện ứng dụng Android.....	65
<b>Hình 5.6</b>	Cảm biến chuyển động được lắp ở cửa ra vào nhà .....	66
<b>Hình 5.7</b>	Mô đun Sim gửi tin nhắn cảnh báo trộm.....	67
<b>Hình 5.8</b>	Cảm biến khí gas lắp tại phòng bếp .....	67
<b>Hình 5.9</b>	Tin nhắn cảnh báo có khí gas rò rỉ .....	68
<b>Hình 5.10</b>	Tin nhắn nhận được khi bị cúp điện 220VAC .....	68
<b>Hình 5.11</b>	Tin nhắn nhận được khi có điện trở lại .....	68

## LIỆT KÊ BẢNG

<b>Bảng</b>	<b>Trang</b>
Bảng 4.1 Danh sách các linh kiện .....	42
Bảng 4.2 Cấu tạo một project Android .....	53
Bảng 4.3 Một số nút chức năng trên Arduino IDE .....	58

## TÓM TẮT

Cuộc “Cách mạng Công nghiệp 4.0” đang dần diễn ra tại nhiều nước phát triển và đang phát triển trên toàn thế giới. Công nghệ về điều khiển thông minh cũng phát triển theo, chúng được ứng dụng trong nhiều lĩnh vực đời sống. Đặc biệt, các hệ thống điều khiển thiết bị điện trong nhà được phát triển mạnh mẽ và ngày càng trở nên phổ biến.

DATN này thiết kế, thi công một mô hình hệ thống điều khiển các thiết bị điện và giám sát nhà. Người sử dụng có thể điều khiển thiết bị điện ở bất cứ nơi nào có Internet, Wifi, 3G, 4G trên Website và ứng dụng Android được thiết kế. Các giá trị nhiệt độ, độ ẩm của căn nhà cũng được cập nhật lên giao diện này. Hệ thống tích hợp một hệ thống báo động bằng tin nhắn và chuông báo động trong các sự cố như: nhiệt độ phòng quá cao khi cháy, có khí gas rò rỉ, có trộm đột nhập. Đặc biệt các tính năng này của hệ thống vẫn hoạt động được ngay cả khi mất điện nhờ mạch chuyển sang Acqui dự trữ.

# Chương 1. TỔNG QUAN

## 1.1 ĐẶT VẤN ĐỀ

Với sự phát triển mạnh mẽ của khoa học và công nghệ thì các thiết bị, hệ thống thông minh cũng dần được tạo ra. Các hệ thống điều khiển thông minh được sử dụng rộng rãi trong đời sống, chẳng hạn như các hệ thống thông minh điều khiển thiết bị điện trong nhà.

Các dự án về ngôi nhà thông minh ứng dụng công nghệ IoT đang được quan tâm xây dựng rất nhiều và đang trở thành xu thế của xã hội hiện đại. Do giá thành vẫn còn cao nên việc triển khai vẫn còn hạn chế. Trên thực tế, nhiều hệ thống nhà thông minh từ Âu-Mỹ nếu lắp trọn bộ cho một căn biệt thự 1 trệt, 1 lầu rộng chừng 300m<sup>2</sup> nhiều khi lên đến 1 tỷ đồng, cá biệt có hệ thống lên đến 2 tỷ đồng. Một con số đủ gây choáng váng! Còn đối với các sản phẩm nhà thông minh từ các Công ty Việt Nam dao động từ 30 – 150 triệu đồng tùy theo gói lắp đặt, từ căn hộ chung cư, nhà phố cho đến biệt thự sân vườn. [1]

Những đề tài nghiên cứu trước đây đã thiết kế được những hệ thống mô hình điều khiển thông minh sử dụng nhiều công nghệ như: sử dụng sóng hồng ngoại, RF, bluetooth hay mạng WLAN[2]. Các hệ thống điều khiển thông minh không ngừng phát triển; đặc biệt trong thời kỳ mạng Internet phát triển mạnh mẽ, các hệ thống điều khiển mới yêu cầu điều khiển được qua mạng Internet, WiFi, 3G/4G và tiện dụng hơn là điều khiển bằng điện thoại thông minh.

Trong ĐATN này, thiết kế một mô hình điều khiển các thiết bị điện và giám sát nhà. Hệ thống sử dụng vi điều khiển trung tâm là board Arduino kết hợp mô đun WiFi ESP8266 Node MCU, mô đun SIM900A và các cảm biến nhiệt độ, cảm biến khí gas, cảm biến chuyển động. Người dùng có thể điều khiển thiết bị điện bằng smartphone chạy hệ điều hành Android hay truy cập vào trang web điều khiển, bất cứ nơi đâu có Internet hoặc WiFi. Đặc biệt, hệ thống có khả năng giám sát căn nhà từ xa như giá trị nhiệt độ, độ ẩm; có khả

năng báo động khi có khí gas rò rỉ, nhiệt độ cao hoặc trộm đột nhập qua tin nhắn và chuông báo. Hệ thống có pin dự phòng trong trường hợp mất điện và gửi tin nhắn cảnh báo mất điện.

### 1.2 MỤC TIÊU

Thiết kế và thi công một mô hình điều khiển các thiết bị điện là các đèn trên ứng dụng điện thoại Android, Website thông qua mạng WiFi và Internet. Hơn nữa, hệ thống này có khả năng giám sát nhiệt độ, độ ẩm căn nhà, phát cảnh báo khi phát hiện khí gas rò rỉ, nhiệt độ phòng quá cao hoặc trộm đột nhập. Hệ thống có khả năng hoạt động được khi mất điện nhờ vào mạch chuyển sang Acqui dự phòng.

#### 1.3.1 NỘI DUNG NGHIÊN CỨU

- Tìm hiểu board mạch Arduino.
- Tìm hiểu về mạng WiFi, mô đun WiFi Node MCU ESP8266.
- Tìm hiểu cảm biến nhiệt độ - độ ẩm, cảm biến khí gas và cảm biến chuyển động.
- Tìm hiểu mô đun SIM900A.
- Tìm hiểu, thiết kế và lập trình xây dựng website.
- Tìm hiểu về trình Android bằng Android Studio.

#### 1.4 GIỚI HẠN

- Thi công mô hình điều khiển các thiết bị điện là các đèn 220VAC công suất 5W.
- Khoảng cách phát hiện trộm đột nhập tối đa 5m, có thể hiệu chỉnh được nhỏ hơn.
- Không phát triển bảo mật của hệ thống.

#### 1.5 BỐ CỤC

##### **Chương 1: Tổng quan.**

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

### **Chương 2: Cơ Sở Lý Thuyết.**

Giới thiệu các cơ sở lý thuyết liên quan đến hệ thống; lựa chọn linh kiện, thiết bị sử dụng để thiết kế hệ thống và trình bày các chuẩn truyền, giao thức.

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

Thiết kế sơ đồ khối của hệ thống, đưa ra sơ đồ nguyên lý của các khối trong hệ thống và thực hiện tính toán thiết kế.

### **Chương 4: Thi Công Hệ Thống.**

Chương này trình bày về thiết kế lưu đồ, đưa ra giải thuật và viết chương trình. Thi công mô hình hệ thống hoàn chỉnh.

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

Đưa ra kết quả đạt được sau 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 đã làm và chưa làm, đồng thời nêu ra hướng phát triển cho hệ thống.

## **Chương 2. CƠ SỞ LÝ THUYẾT**

### **2.1 TỔNG QUAN VỀ INTERNET OF THINGS**

#### **2.1.1 Giới thiệu về Internet of Things**

Internet of Things (viết tắt IoT) mô tả mạng lưới kết nối vạn vật. Khi mà mỗi đồ vật hay cả con người được cung cấp một định danh riêng và tất cả có khả năng truyền tải, trao đổi thông tin, dữ liệu qua một mạng duy nhất. IoT phát triển từ sự kết hợp của công nghệ không dây, công nghệ vi cơ điện tử và Internet. Nói đơn giản IoT là một tập hợp các thiết bị có khả năng kết nối với nhau, với Internet và với thế giới bên ngoài để thực hiện một công việc nào đó.

#### **2.1.2 Ứng dụng của IoT**

IoT có ứng dụng trong nhiều lĩnh vực:

- Quản lý chất thải
- Quản lý và lập kế hoạch quản lý đô thị
- Quản lý môi trường
- Phản hồi trong các tình huống khẩn cấp
- Mua sắm thông minh
- Quản lý các thiết bị cá nhân
- Đồng hồ đo thông minh
- Tự động hóa ngôi nhà

#### **2.1.3 Đặc điểm công nghệ IoT**

Điểm quan trọng của IoT đó là các đối tượng phải có thể được nhận biết và định dạng (identifiable). Nếu mọi đối tượng, kể cả con người, được “đánh dấu” để phân biệt bản thân đối tượng đó với những thứ xung quanh thì chúng ta có thể hoàn toàn quản lý được nó thông qua máy tính. Việc đánh dấu này có thể được thực hiện thông qua nhiều công nghệ, chẳng hạn như RFID, NFC, mã vạch, mã QR,... Mạng kết nối thì có thể là qua Wi-Fi, mạng viễn thông băng rộng (3G, 4G), Bluetooth,



Ngoài những kỹ thuật nói trên, nếu nhìn từ thế giới web, chúng ta có thể sử dụng địa chỉ IP để xác định từng vật. Mỗi thiết bị sẽ có một IP riêng biệt không nhầm lẫn. Do đó mọi thứ có thể dễ dàng kết nối vào Internet cũng như kết nối với nhau.

### 2.2 Tổng quan về mạng WiFi

WiFi viết tắt từ Wireless Fidelity hay mạng 802.11 là hệ thống mạng không dây sử dụng sóng vô tuyến, giống như điện thoại di động, truyền hình và radio. Hệ thống cho phép truy cập Internet tại những khu vực có sóng của hệ thống này, hoàn toàn không cần đến cáp nối.

Sóng WiFi: Các sóng vô tuyến sử dụng cho WiFi gần giống với các sóng vô tuyến sử dụng cho thiết bị cầm tay, điện thoại di động và các thiết bị khác. Nó có thể chuyển và nhận sóng vô tuyến, chuyển đổi các mã nhị phân 1 và 0 sang sóng vô tuyến và ngược lại.

Tuy nhiên, sóng WiFi có một số khác biệt so với các sóng vô tuyến khác ở chỗ: chúng truyền và phát tín hiệu ở tần số 2.5 GHz hoặc 5GHz. Tần số này cao hơn so với các tần số sử dụng cho điện thoại di động, các thiết bị cầm tay và truyền hình. Tần số cao hơn cho phép tín hiệu mang theo nhiều dữ liệu hơn.

### 2.3 Giới thiệu về Website, Webserver và PHP

#### 2.3.1. Giới thiệu về Website

Website là một tập hợp các trang Web có liên quan với nhau. Mỗi trang Web là một tài liệu siêu văn bản. Tài liệu này có thể chứa văn bản, âm thanh, hình ảnh... Được mã hoá đặc biệt, sử dụng ngôn ngữ đánh dấu siêu văn bản – HTML (HyperText Markup Languages).

Mỗi WebSite được lưu trữ trên một máy phục vụ Web, vốn là các máy chủ Internet lưu trữ hàng các trang Web riêng lẻ.

#### 2.3.2 Máy chủ Web (Webserver)

Ở khía cạnh phần cứng, một Web server là một máy tính lưu trữ các file thành phần của một Website (ví dụ: các tài liệu HTML, các file ảnh, CSS) và

có thể phân phát chúng tới thiết bị của người dùng. Nó kết nối tới mạng Internet và có thể truy cập tới thông qua một tên miền.

Bất cứ khi nào một trình duyệt cần một file được lưu trữ trên một web server, trình duyệt yêu cầu file đó thông qua HTTP. Khi một yêu cầu tới đúng web server, HTTP server sẽ gửi tài liệu được yêu cầu trở lại, cũng thông qua HTTP.

### 2.3.3 Giới thiệu PHP

PHP (viết tắt của cụm từ Personal Home Page) là ngôn ngữ lập trình kịch bản (scripting language) mã nguồn mở được dùng phổ biến để ra tạo các ứng dụng web chạy trên máy chủ. PHP thực hiện các hàm hệ thống, ví dụ: từ các file trên một hệ thống, nó có thể tạo, mở, đọc, ghi và đóng chúng. PHP có thể xử lý các form, ví dụ: thu thập dữ liệu từ file, lưu dữ liệu vào một file, trả về dữ liệu tới người dùng. Bạn có thể thêm, xóa, sửa đổi các phần tử bên trong Database của bạn thông qua PHP.

### 2.4 Giới thiệu về tin nhắn SMS

SMS (Short Message Services tạm dịch Dịch vụ tin nhắn ngắn) là một giao thức viễn thông cho phép gửi các thông điệp dạng văn bản ngắn (không quá 160 chữ cái). SMS hoạt động dựa trên 3 công nghệ mạng lớn đó là GSM, CDMA và TDMA. Một tin nhắn văn bản từ điện thoại di động sẽ được lưu trữ trong trung tâm lưu trữ tin nhắn SMC (Stored Message Central), sau đó sẽ chuyển tiếp đến đích cần đến. Tin nhắn SMS thường sử dụng một kênh riêng biệt để gửi và kiểm soát các tin nhắn. Vì vậy các cuộc gọi và các dạng tin nhắn khác sẽ không bị cản trở bởi SMS. Kênh điều khiển này được sử dụng để theo dõi vị trí điện thoại của bạn giúp tin nhắn có thể gửi chuẩn xác đến địa điểm mà bạn đang đứng.

Một số lợi ích của tin nhắn SMS có thể kể đến như:

- Các tin nhắn SMS có thể được gửi và đọc tại bất kỳ thời điểm nào.
- Ngày nay, hầu hết mọi người đều có điện thoại di động của riêng mình và mang nó theo người hầu như cả ngày. Với một điện thoại di động, bạn có thể gửi và đọc các tin nhắn SMS bất cứ lúc nào bạn muốn, sẽ

không gặp khó khăn gì khi bạn đang ở trong văn phòng hay trên xe bus hay ở nhà...

- Tin nhắn SMS có thể được gửi tới các điện thoại bị tắt nguồn.
- Nếu như không chắc cho một cuộc gọi nào đó thì bạn có thể gửi một tin nhắn SMS đến bạn của bạn thậm chí khi người đó tắt nguồn máy điện thoại trong lúc bạn gửi tin nhắn đó. Hệ thống SMS của mạng điện thoại sẽ lưu trữ tin nhắn đó rồi sau đó gửi nó tới người bạn đó khi điện thoại của người bạn này mở nguồn.
- Các tin nhắn SMS ít gây phiền phức trong khi bạn vẫn có thể giữ liên lạc với người khác.

### 2.5 Giới thiệu về Android

Android là một hệ điều hành có mã nguồn mở dựa trên nền tảng Linux, được thiết kế dành cho các thiết bị di động có màn hình cảm ứng như: điện thoại thông minh và máy tính bảng. Ban đầu, Android được phát triển bởi Tổng công ty Android, với sự hỗ trợ tài chính từ Google, sau này được chính Google mua lại vào năm 2005 và hệ điều hành Android đã ra mắt vào năm 2007. Android hiện là nền tảng di động rất phổ biến và được nhiều hãng sản xuất phần cứng sử dụng.

Chính mã nguồn mở cùng với giấy phép không có nhiều ràng buộc đã cho phép các nhà phát triển thiết bị, mạng di động và các lập trình viên được điều chỉnh và phân phối Android một cách tự do. Những yếu tố này đã giúp Android trở thành nền tảng điện thoại thông minh phổ biến nhất thế giới. Android chiếm 75% thị phần điện thoại thông minh trên toàn thế giới vào thời điểm quý 3 năm 2012, với tổng cộng 500 triệu thiết bị đã được kích hoạt và 1,3 triệu lượt kích hoạt mỗi ngày.

### 2.6 Giới thiệu phần cứng

#### 2.6.1 Vi điều khiển

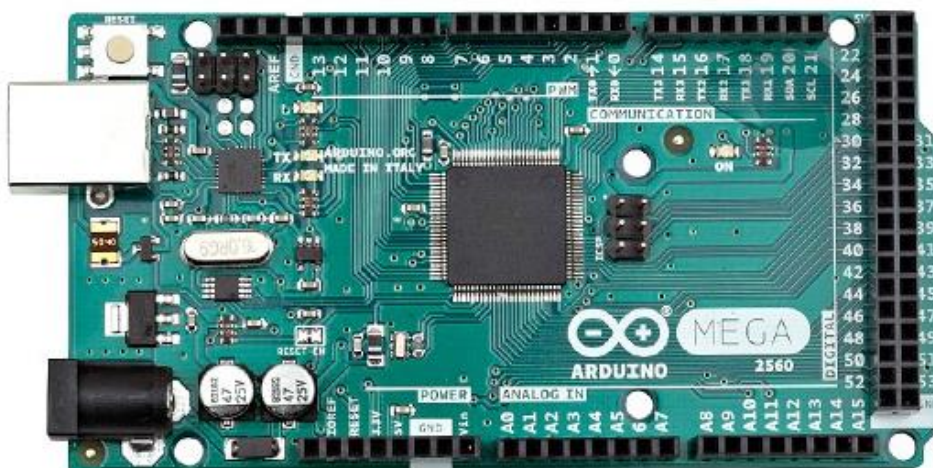
##### a. Giới thiệu về Arduino

Arduino là một bo mạch vi xử lý được dùng để lập trình, tương tác với các thiết bị phần cứng như: cảm biến, động cơ, đèn hoặc các thiết bị khác. Đặc điểm nổi bật của Arduino là môi trường phát triển ứng dụng cực kỳ dễ sử dụng, sử dụng ngôn ngữ lập trình có thể học rất dễ dàng và nhanh chóng. Arduino được bán ra với mức giá rất thấp. Phần cứng bao gồm một board mạch nguồn mở được thiết kế trên nền tảng vi xử lý megaAVR Atmel 8bit, hoặc ARM Atmel 32-bit. Những Model hiện tại thường 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, Arduino được ra đời để mang đến một cách thức tiếp cận dễ dàng, không tốn kém cho những người yêu thích, sinh viên và giới chuyên nghiệp để tạo ra những sản phẩm lập trình. Những ví dụ phổ biến khi mới bắt đầu học Arduino bao gồm: làm các robot đơn giản, điều khiển nhiệt độ và phát hiện chuyển động.

### **b. Giới thiệu về Arduino mega 2560**

Arduino mega 2560 (hình 2.1) là một board Arduino được trang bị chip ATmega2560, với bộ nhớ flash memory 256 KB, 8KB cho bộ nhớ SRAM, 4 KB cho bộ nhớ EEPROM.



**Hình 2.1** Board Arduino mega 2560

Board Arduino Mega2560 có thông số kỹ thuật như sau:

- Vi điều khiển ATmega 2560, với 256 KB flash memory, 8 KB bộ nhớ SRAM, 4 KB EEPROM.

- Điện áp hoạt động: 5V

- Điện áp vào: 6-20V

- Cường độ dòng điện trên mỗi I/O pin: 20mA.

- 54 chân digital (trong đó có 15 chân có thể được sử dụng như những chân PWM là từ chân số 2  $\rightarrow$  13 và chân 44 45 46).

- 6 ngắt ngoài: chân 2 (interrupt 0), chân 3 (interrupt 1), chân 18 (interrupt 5), chân 19 (interrupt 4), chân 20 (interrupt 3) và chân 21 (interrupt 2).

- 16 chân vào analog (từ A0 đến A15).

- 4 cổng Serial giao tiếp với phần cứng

- 1 bộ giao tiếp SPI.

- 1 bộ giao tiếp I2C.

- 1 thạch anh với tần số dao động 16 MHz.

- 1 cổng kết nối USB.

- Được tích hợp sẵn thư viện cho Matlab.

### **Nguồn nuôi**

Arduino có thể được cấp nguồn thông qua kết nối USB hoặc với nguồn cung cấp điện bên ngoài. Các nguồn sẽ được lựa chọn tự động. Hệ thống vi điều khiển có thể hoạt động bằng một nguồn cung cấp bên ngoài từ 6V đến 20V. Nên cung cấp với nguồn điện áp nhỏ hơn 7V để board mạch hoạt động ổn định.

- Chân  $V_{in}$ : Điện áp đầu vào Arduino khi chúng ta dùng nguồn điện bên ngoài. Chúng ta có thể cung cấp nguồn thông qua chân này.

- Chân 5V: Cung cấp nguồn vi điều khiển và các bộ phận khác trên bo mạch và cung cấp nguồn cho các thiết bị ngoại vi khi kết nối tới bo mạch.
- Chân 3V3: Cung cấp nguồn cho các thiết bị cảm biến.
- Chân GND : Chân nối đất..

### 2.6.2 Cảm biến nhiệt, độ ẩm DHT11

Hệ thống được thiết kế trong ĐATN này với mục tiêu sử dụng cảm biến nhiệt độ và độ ẩm giám sát nhiệt độ phòng của nhà, với tầm đo nhiệt độ từ 25°C - 45°C, độ chính xác dừng ở mức tương đối. Do đó, đề xuất sử dụng cảm biến nhiệt độ, độ ẩm DHT11. DHT11 là cảm biến nhiệt độ, độ ẩm rất thông dụng hiện nay vì chi phí rẻ và rất dễ lấy dữ liệu thông qua giao tiếp 1-dây. Cảm biến được tích hợp bộ tiền xử lý tín hiệu giúp dữ liệu nhận về được chính xác mà không cần phải qua bất kỳ tính toán nào.

Cảm biến DHT11 gồm 2 chân cấp nguồn và 1 chân tín hiệu trong hình 2.2:



**Hình 2.2** Mô đun cảm biến DHT11

Đặc điểm:

- Điện áp hoạt động : 3V - 5V (DC)
- Dải độ ẩm hoạt động : 20% - 90% RH, sai số  $\pm 5\%RH$
- Dải nhiệt độ hoạt động : 0°C ~ 50°C, sai số  $\pm 2^\circ C$
- Tần số lấy mẫu tối đa: 1 Hz

### 2.6.3 Mô đun cảm biến khí gas MQ2

Cảm biến khí ga MQ2 là một trong những loại cảm biến được sử dụng để nhận biết nhiều loại khí như: butan, Propane, Methane, Alcohol, khói và khí ga. Được thiết kế với độ nhạy cao, thời gian đáp ứng nhanh. Bộ phận cảm nhận của cảm biến khí MQ2 là  $\text{SnO}_2$ , có độ dẫn điện thấp trong không khí sạch. Khi phát hiện các khí dễ cháy, độ dẫn điện của cảm biến sẽ tăng khi nồng độ khí tăng. Chính nhờ đặc điểm này người ta biến đổi sự thay đổi này thành điện áp.

Mô đun MQ2 bao gồm cảm biến khí gas MQ2, một chiết áp điều chỉnh độ nhạy, LM393, đèn LED báo hiệu. Hình 2.3 là hình ảnh của mô đun cảm biến khí gas MQ2.



**Hình 2.3** Mô đun cảm biến khí gas MQ2

#### **Đặc điểm:**

- Nguồn cung cấp: 4.5V đến 5V DC
- Độ nhạy cao với khí Propane, khói và Butan
- Tuổi thọ cao, giá thành thấp
- Hỗ trợ đầu ra analog và digital
- Thời gian đáp ứng:  $\leq 10\text{s}$

- Thời gian phục hồi:  $\leq 30s$
- Trở kháng khi nóng:  $31\Omega \pm 3\Omega$
- Dòng tiêu thụ khi nóng:  $\leq 180mA$
- Tích hợp LED báo động trên modul
- Giao tiếp đơn giản với 4 chân VCC GND Analog và Digital

### 2.6.4. Mô đun cảm biến chuyển động PIR HC – SR501

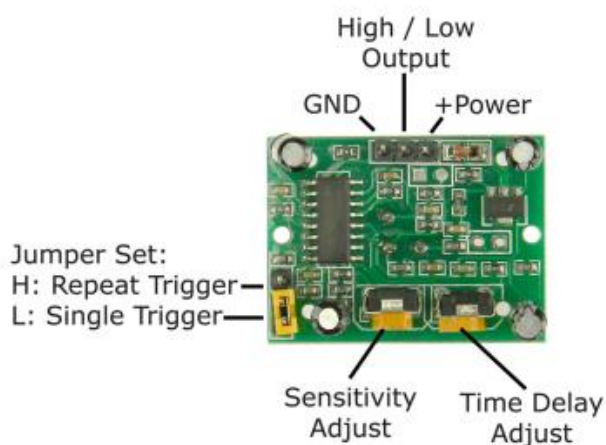
HC-SR501 là một mô đun cảm biến hồng ngoại, độ nhạy cao, độ tin cậy cao, chế độ hoạt động tiêu tốn ít năng lượng, được sử dụng rộng rãi trong các thiết bị điện cảm biến tự động khác nhau. Cảm biến PIR (Passive Infra-Red) phát hiện chuyển động bằng cách đo những thay đổi trong bức xạ hồng ngoại phát ra bởi các đối tượng. Khi phát hiện chuyển động cảm biến PIR sẽ xuất ra 1 xung ở mức cao, xung này được đọc bởi vi điều khiển để thực hiện chức năng mong muốn. Tín hiệu ngõ ra Digital, đọc trực tiếp trên chân vi điều khiển. Ngoài ra còn có 2 biến trở điều chỉnh độ nhạy chuyển động và thời gian giữ trạng thái kích hoạt của cảm biến. Hình 2.4 sau đây là hình ảnh của mô đun cảm biến HC SR501.



**Hình 2.4** Mô đun cảm biến chuyển động HC SR501

Hình 2.5 sau đây mô tả sơ đồ chân của mô đun cảm biến HC SR501:





**Hình 2.5** Sơ đồ chân cảm biến HC SR501

Đặc điểm:

- Điện áp hoạt động: 5VDC
- Dòng nghỉ:  $< 50\mu\text{A}$
- Nhiệt độ hoạt động:  $-15^{\circ}\text{C}$  đến  $70^{\circ}\text{C}$
- Góc quét:  $120^{\circ}$
- Tầm xa: 7m
- Thời gian trễ: điều chỉnh bằng biến trở, từ 5 giây đến 200 giây.
- Điện áp ra: 0 hoặc 3.3v
- Chế độ kích hoạt:
  - + Kích hoạt L Điện áp ra tự động chuyển về 0 khi hết thời gian trễ.
  - + Kích hoạt H (mặc định) Điện áp ra tự động giữ nguyên 3.3V cho đến khi không còn chuyển động.
- Tính năng hoạt động: khi có người đi vào vùng hoạt động của cảm biến, điện áp chân ra tự động đặt lên 3.3V, sau khi người đi ra khỏi vùng hoạt động, sau khi hết khoảng thời gian trễ (có thể đặt bằng biến trở) điện áp ra trở về 0.

### 2.6.5 Mô đun WiFi ESP8266 Node MCU

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

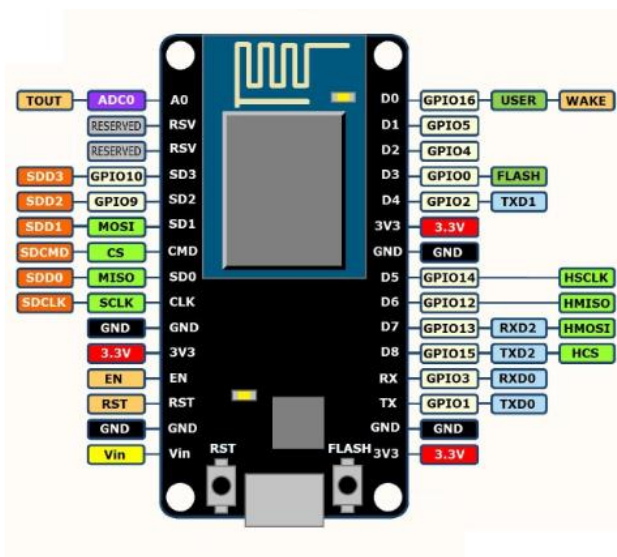
## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

NodeMCU (hình 2.6) được phát triển dựa trên Chip WiFi ESP8266EX bên trong. Board còn tích hợp IC CP2102, giúp dễ dàng giao tiếp với máy tính thông qua Micro USB để thao tác với board. Và có sẵn nút nhấn, led báo hiệu.

Với kích thước nhỏ gọn, linh hoạt board dễ dàng liên kết với các thiết bị ngoại vi để tạo thành project, sản phẩm một cách nhanh chóng.

Các tính năng chính:

- Có thể quét và kết nối đến một mạng WiFi bất kỳ để thực hiện các tác vụ như lưu trữ, truy cập dữ liệu từ server.
- Tạo điểm truy cập WiFi cho phép các thiết bị khác kết nối, giao tiếp và điều khiển.
- Là một server để xử lý dữ liệu từ các thiết bị sử dụng Internet khác.



- Bộ nhớ Flash: 4MB
- Giao tiếp: Cáp Micro USB
- Hỗ trợ bảo mật: WPA/WPA2
- Tích hợp giao thức TCP/IP

### 2.6.6 Mô đun SIM900A

Module SIM900A thiết kế dựa trên chipset SIM900A. Mô đun SIM900A(hình 2.7) được tích hợp các linh kiện, kết nối cần thiết, sẵn sàng cho người sử dụng để nghiên cứu, học tập hoặc làm demo nhanh chóng, tiện lợi cho các dự án có ứng dụng GSM.



**Hình 2.7** Hình ảnh mô đun SIM900A

Chức năng các chân:

- Chân 5V: chân cung cấp nguồn dương 5VDC để module sim900A hoạt động.
- Chân DTR: Chân đầu cuối dữ liệu.
- Chân TX: chân truyền UART(TX) dùng để truyền dữ liệu.
- Chân RX: chân nhận UART(RX) dùng để nhận và xử lý dữ liệu.
- Chân Headphone: chân xuất dữ liệu âm thanh ra loa thoại.
- Chân Microphone: chân kết nối với mic nếu muốn đàm thoại.
- Chân Reset: chân reset module sim.
- Chân GND: chân nối mass.

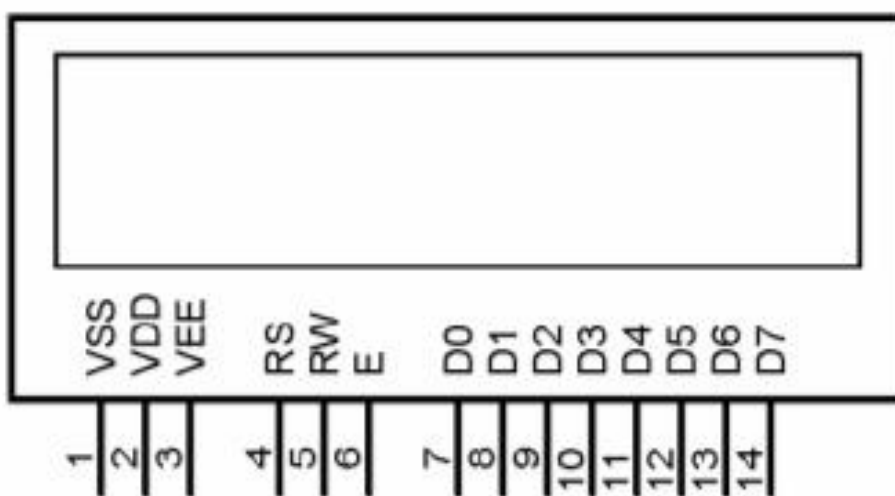
### 2.6.7 Giới thiệu LCD 16x2

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

LCD hiển thị được 2 hàng mỗi hàng hiển thị được 16 kí tự. Hình 2.8 mô tả sơ đồ các chân của LCD 16x2.

Thông số :

- Kích thước hiển thị : 16x2 dòng
- Màu hiển thị : đen trắng
- Chế độ giao tiếp : 8 hoặc 4 bit
- Cỡ chữ hiển thị : 5x7 hoặc 5x10
- Số chân : 16



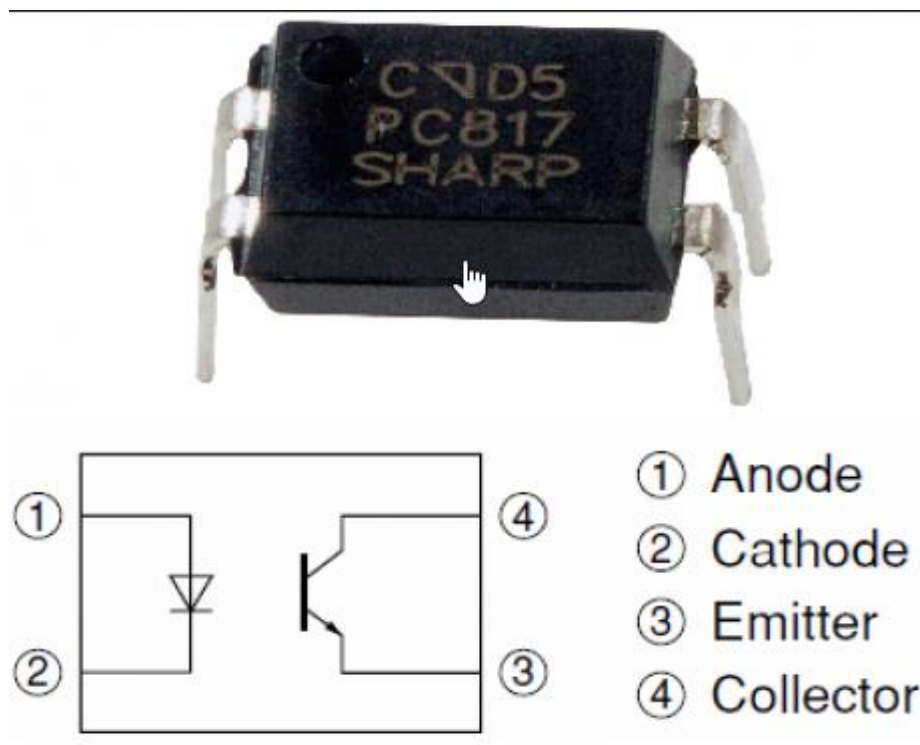
**Hình 2.8** Sơ đồ chân LCD 16x2

- VSS: Chân nối đất cho LCD, khi thiết kế ta nối chân này với GND của vi điều khiển.
- VDD: Chân cấp nguồn cho LCD, khi thiết kế ta nối chân này với VCC = 5V của mạch điều khiển.
- Vee: Dùng để điều chỉnh độ tương phản cho LCD.
- RS: Chân chọn thanh ghi (register select). Nối chân RS xuống mức logic '0' (GND) hay '1' (VDD) để chọn thanh ghi.
  - + Mức '0' : lệnh.
  - + Mức '1' : dữ liệu .
- R/W Chân chọn chế độ đọc/ghi (Read/Write). Nối chân R/W xuống mức logic '0' để LCD hoạt động ở chế độ ghi hoặc nối R/W lên mức logic '1' để LCD hoạt động ở chế độ đọc.

- E: Chân cho phép (Enable).
- D0- D7: Tám đường của bus dữ liệu dùng để trao đổi thông tin với MCU.

### 2.6.8 Giới thiệu OPTO PC817

Opto là một trong những linh kiện điện tử cơ bản được người dùng sử dụng rộng rãi trong lĩnh vực điện tử. Opto hay còn được gọi là bộ cách ly quang dùng để truyền tín hiệu điện bằng cách chuyển tín hiệu ánh sáng và sau đó mới truyền đi. Nó dùng để cách ly điện giữa đầu vào và đầu ra.



**Hình 2.9** Opto PC817

Opto này dùng để chống nhiễu (cách ly) giữa tải và mạch điều khiển MCU. Cấu tạo gồm diode quang và transistor. Khi diode dẫn thì sẽ có dòng phân cực qua transistor.

### 2.7 Chuẩn truyền thông UART

UART : Universal Asynchronous Receiver/Transmitter, là kiểu truyền thông tin nối tiếp không đồng bộ, UART thường được dùng trong máy tính

công nghiệp, truyền thông, vi điều khiển, hay một số các thiết bị truyền tin khác.

### **Một số thông số:**

**Baud rate (tốc độ Baud) :** Khi truyền nhận không đồng bộ để hai mô đun hiểu được nhau thì cần quy định một khoảng thời gian cho 1 bit truyền nhận, nghĩa là trước khi truyền thì tốc độ phải được cài đặt đầu tiên. Theo định nghĩa thì tốc độ baud là số bit truyền trong một giây.

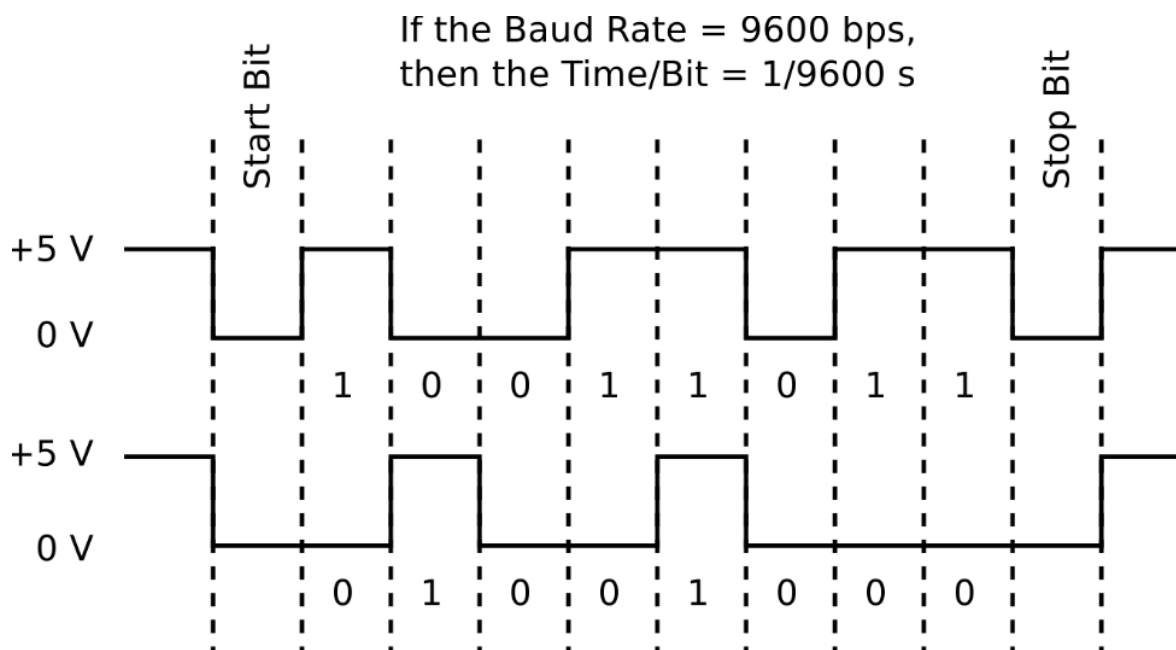
**Frame (khung truyền) :** Do khi truyền thông nối tiếp này rất dễ mất dữ liệu nên ngoài tốc độ, khung truyền cũng được cài đặt từ ban đầu để tránh bớt sự mất mát dữ liệu này. Khung truyền quy định số bit trong mỗi lần truyền, các bit báo như start, stop, các bit kiểm tra như parity, và số bit trong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

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

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

**Parity bit :** Là bit kiểm tra dữ liệu đúng không. có 2 loại parity : chẵn (even parity), lẻ (odd parity). Parity chẵn là bit parity thêm vào để số bit 1 trong data + parity = chẵn, parity lẻ là bit parity thêm vào để số bit 1 trong data + parity = lẻ. Bit Parity là không bắt buộc nên có thể dùng hoặc không.

**Stop :** là bit báo cáo kết thúc khung truyền, thường là mức 5V và có thể có 1 hoặc 2 bit stop. Giả đồ trong hình 2.10 mô tả dữ liệu truyền đi bằng UART.



**Hình 2.10** Giải đồ truyền dữ liệu UART

### 2.8 Phương thức GET

GET là một phương thức của giao thức HTTP gửi dữ liệu từ máy khách(client) về server xử lý. Phương thức GET rất dễ nhận thấy, đó là trên URL sẽ kèm theo dữ liệu mà chúng ta muốn gửi.

#### a. Client gửi dữ liệu lên:

Phương thức GET là phương thức gửi dữ liệu thông qua đường dẫn URL nằm trên thanh địa chỉ của Browser. Server sẽ nhận đường dẫn đó và phân tích trả về kết quả. Server sẽ phân tích tất cả những thông tin đằng sau dấu hỏi (?) chính là phần dữ liệu mà Client gửi lên.

*Ví dụ: Với URL load.php?ND=12 thì Server sẽ nhận được giá trị ND là 12*

Để truyền nhiều dữ liệu lên Server ta dùng dấu "&" để phân cách giữa các cặp giá trị. Giả sử muốn truyền ND = 12 và title = 'tat' thì URL sẽ có dạng load.php?id=12&title=tat. Lưu ý vị trí các cặp giá trị không quan trọng, nghĩa là cặp title có thể nằm trước cặp ND cũng được.

#### b. Server nhận dữ liệu:

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

---

Tất cả các dữ liệu mà Client gửi lên bằng phương thức GET đều được lưu trong một biến toàn cục mà PHP tự tạo ra đó là biến `$_GET`, biến này là kiểu mảng kết hợp lưu trữ danh sách dữ liệu từ client gửi lên theo quy luật key => value. Ví dụ với URL `load.php?ND=12&title=tat` thì dữ liệu sẽ được lưu trong biến `$_GET` dưới dạng:

```
$_GET = array(  
    'ND' => '12',  
    'title' => 'tat'  
);
```

Vì thế để lấy dữ liệu thì ta chỉ cần làm như sau:

```
// Lấy ND  
$id = $_GET['id'];  
echo $id; // kết quả là 12  
  
// Lấy title  
$title = $_GET['title'];  
echo $title; // kết quả là tat
```



## **Chương 3. TÍNH TOÁN VÀ THIẾT KẾ**

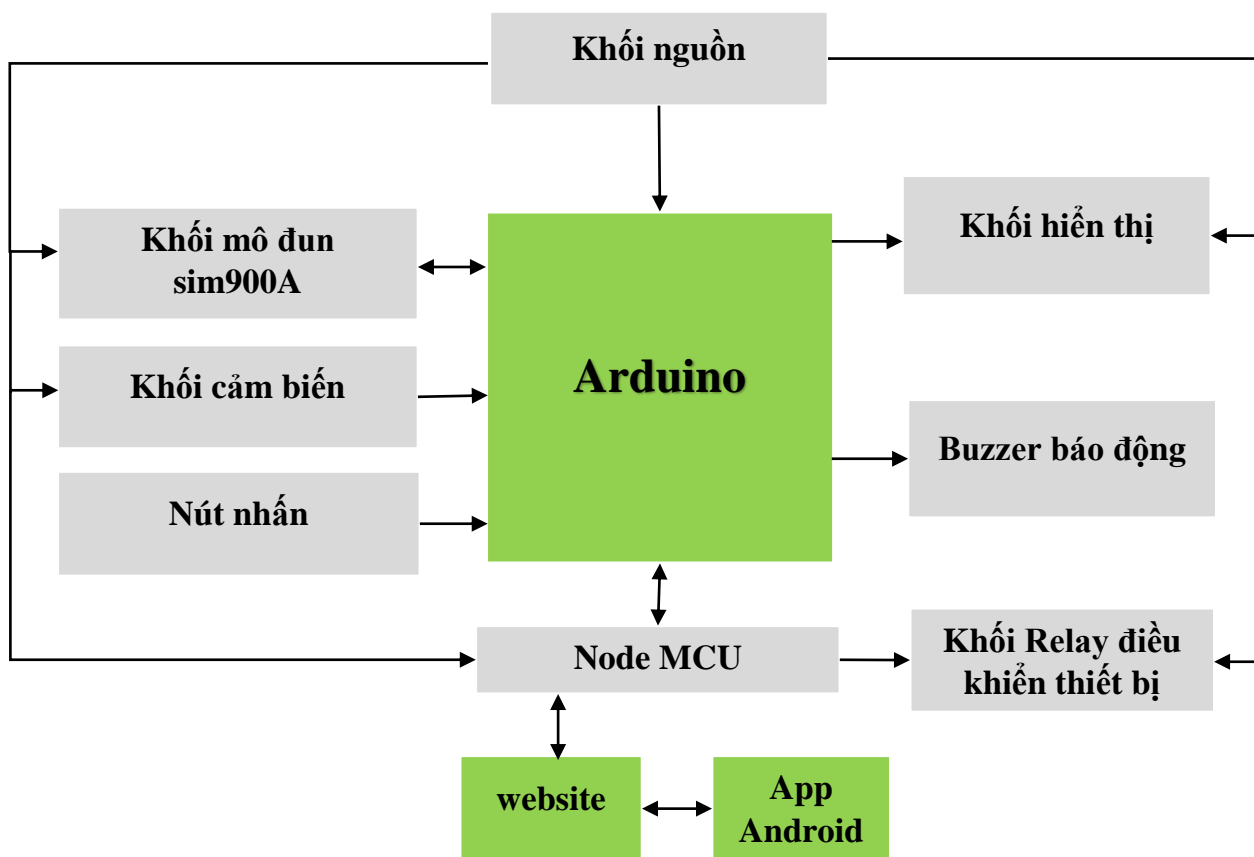
### **3.1 GIỚI THIỆU**

Trong đề tài này nhóm chúng tôi sẽ thiết kế một hệ thống điều khiển thiết bị điện ở bất cứ nơi đâu có Internet, Wifi, 3G bằng Website và ứng dụng Android. Người dùng cũng có thể xem giá trị nhiệt độ và độ ẩm căn nhà. Ngoài ra, còn có hệ thống báo động qua tin nhắn và chuông reo trong trường hợp có trộm đột nhập, khí gas rò rỉ hay nhiệt độ quá cao. Hệ thống có nguồn Acqui dự trữ nên vẫn hoạt động được trong trường hợp mất điện 220V. Hệ thống này sẽ bao gồm khối xử lý trung tâm – board Arduino, khối mô đun wifi, mô đun SIM900A, khối Relay điều khiển thiết bị, khối hiển thị, khối cảm biến và khối nguồn.

### **3.2 TÍNH TOÁN VÀ THIẾT KẾ HỆ THỐNG**

#### **3.2.1 Thiết kế sơ đồ khối hệ thống**

Sơ đồ khối của hệ thống được thể hiện trong hình 3.1 :



**Hình 3.1** Sơ đồ khối hệ thống

**Chức năng các khối như sau:**

**- Khối điều khiển - Arduino**

- Đọc các và xử các tín hiệu cảm biến (cảm biến chuyển động, cảm biến khí Gas, cảm biến nhiệt độ - độ ẩm ).
- Giao tiếp với mô đun SIM900A để gửi tin nhắn cảnh báo sự cố.
- Giao tiếp với mô đun wifi ESP8266 Node MCU để gửi dữ liệu nhiệt độ độ ẩm lên website, lấy trạng thái các thiết bị từ Node MCU để hiển thị trạng thái đèn ra LCD.
- Đưa giá trị nhiệt độ - độ ẩm và các trạng thái đèn ra khối hiển thị.
- Nhận tín hiệu nút nhấn để đặt và reset các chế độ hoạt động cho khối cảm biến.
- Kích hoạt buzzer báo động.
- Đọc tín hiệu báo mất điện vào nguồn.

#### - **Khối mô đun SIM900A**

Khối này chỉ gồm 1 mô đun sim để gửi tin nhắn cảnh báo sự cố (trộm , cháy, gas ) tới 1 số điện thoại. Nhận tin nhắn để bật hoặc tắt chế độ cảnh báo trộm và tắt âm thanh báo trộm.

#### - **Khối điều khiển - Mô đun wifi Node MCU**

Mô đun này có nhiệm vụ bắt sóng wifi và truy cập vào Sever đã được xây dựng sẵn, nhằm đưa dữ liệu nhiệt độ - độ ẩm lên website và lấy dữ liệu từ website xuống điều khiển các thiết bị điện bằng phương thức GET. Truyền UART với Arduino để lấy các giá trị nhiệt độ - độ ẩm từ Arduino, đưa trạng thái thiết bị tới Arduino hiển thị lên LCD.

#### - **Khối cảm biến**

Khối này gồm các cảm biến khí gas MQ2, cảm biến nhiệt độ-độ ẩm DHT11, cảm biến chuyển động PIR. Arduino sẽ đọc các tín hiệu từ các cảm biến này và xử lý.

#### - **Khối hiển thị**

Khối này dùng LCD 16X2 để hiển thị nhiệt độ - độ ẩm và các trạng thái hiện tại của thiết bị .

#### - **Khối Relay điều khiển thiết bị**

Khối này bao gồm 4 Relay và mạch kích Relay kết nối với 4 bóng đèn nhằm mục đích để điều khiển các bóng đèn thông qua website và ứng dụng Android.

#### - **Khối Website**

Website được xây dựng, nhận dữ liệu nhiệt độ độ ẩm gửi lên từ Node MCU, đồng thời trả về dữ liệu điều khiển các đèn xuống Node MCU.

#### - **Khối phần mềm Android**

Ứng dụng Android kết nối trực tiếp tới website để đọc giá trị nhiệt độ độ ẩm và hiển thị lên giao diện ứng dụng, đồng thời có khả năng thay đổi các giá trị trong trang web để gián tiếp điều khiển các đèn.

- **Khởi nút nhấn và buzzer**

Khởi này gồm hai nút nhấn (nút enable chức năng báo trộm và nút reset chuông buzzer) và buzzer. Hai nút nhấn này được xử lý là 2 ngắt ngoài của board Arduino.

- **Khởi nguồn**

Cấp nguồn cho các khối hoạt động. Khởi nguồn được thiết kế thêm mạch tự động chuyển sang nguồn Acqui dự trữ khi mất nguồn điện chính cấp trong mạch. Ngoài ra, có trích xuất điện áp và được đọc bởi 1 chân của Arduino; so sánh với trạng thái trong EPROM để xác định trạng thái mất điện của mạch và gửi tin nhắn cảnh báo.

### 3.2.2 Tính toán và thiết kế mạch

#### a. Khởi điều khiển Arduino

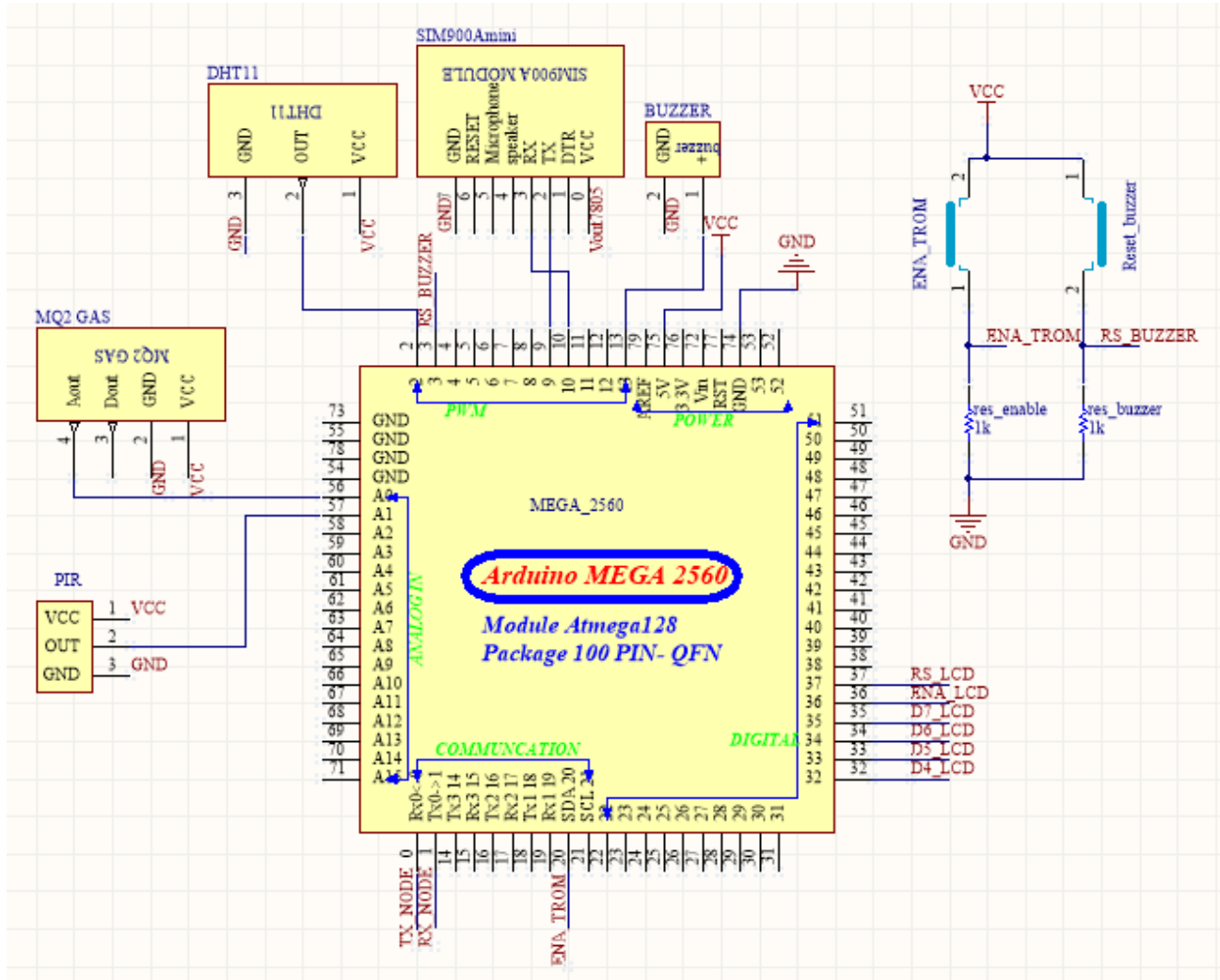
Khởi điều khiển là board Arduino Mega 2560, ưu điểm tích hợp 4 cổng giao tiếp UART, hỗ trợ 6 ngắt, lập trình đơn giản.

Board Arduino được cấp nguồn 6VDC vào chân Vin. Chi tiết kết nối giữa Board Arduino và các khối khác được thể hiện trong hình 3.2, cụ thể như sau:

- Node MCU: Tx Rx của Node MCU kết nối với Rx0 Tx0(chân 0 và 1) của Arduino.
- Mô đun Cảm biến MQ2: chân Aout kết nối với chân A0 của Arduino
- Mô đun Cảm biến DHT11: chân Dout kết nối với chân số 2 của Arduino
- Mô đun Cảm biến PIR: chân Aout kết nối với chân A1 của Arduino
- Mô đun SIM900A: Tx, Rx nối với chân 10 và 9 của Arduino.
- Nút nhấn Reset buzzer báo trộm nối với chân 20 của Arduino.
- Nút nhấn Enable/Disable tính năng báo trộm nối với chân số 3 của Arduino.

### CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

- RS, RW, D4, D5, D6, D7 của LCD lần lượt nối với 37, 36, 35, 34, 33, 32 của Arduino.



Hình 3.2 Sơ đồ nguyên lý mạch điều khiển board Arduino

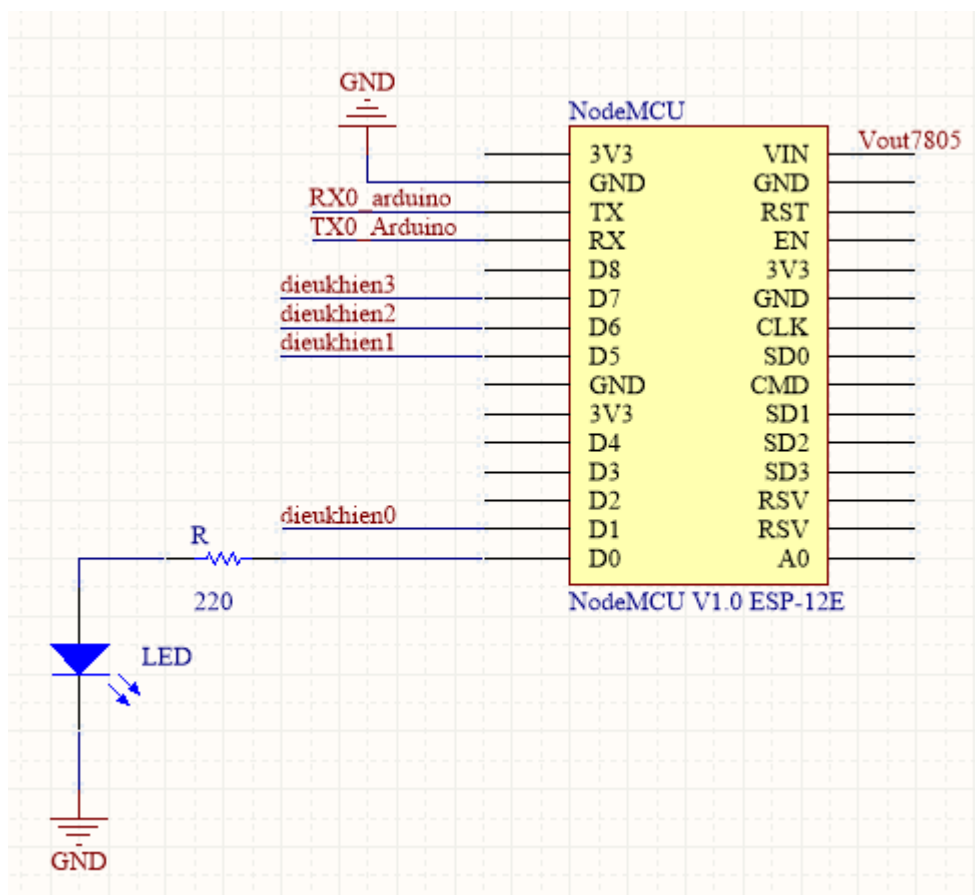
#### b. Khối điều khiển – Node MCU

Mục đích kết nối với vào mạng WiFi, gửi dữ liệu và đọc dữ liệu từ Website. Trực tiếp điều khiển khối Relay thiết bị. Với ưu điểm là dòng ESP 8266 V12 dễ dàng lập trình, tích hợp sẵn các chân GPIO, thích hợp cho điều khiển các thiết bị qua các chân này.

Kết nối chân giữa mô đun Node MCU đến các khối khác được mô tả như trong hình 3.3, cụ thể như sau:

### CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

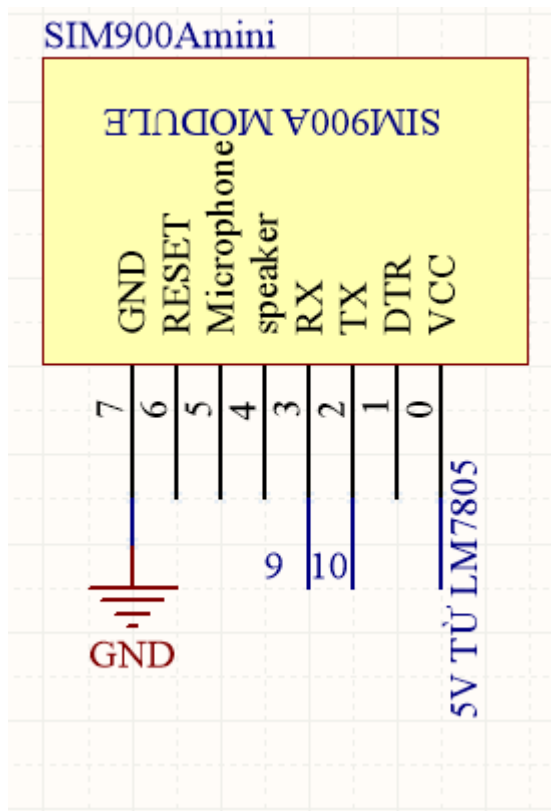
- Tx, Rx nối trực tiếp với Rx0 Tx0 board Arduino
- Chân D0 điều khiển 1 led báo trạng thái kết nối vào mạng wifi thành công.
- Chân D2, D5, D6, D7 kết nối với khối Relay điều khiển thiết bị điều khiển 4 đèn tương ứng.
- Chân Vin được cấp nguồn 5VDC từ Vout của IC7805.



**Hình 3.3** Sơ đồ nguyên lý nối dây mô đun Node MCU

#### c. Khối mô đun SIM900A

Mô đun SIM900A được sử dụng rộng rãi trong các ứng dụng giao tiếp với vi điều khiển xử lý tin nhắn và cuộc gọi, điều khiển dễ dàng thông qua tập lệnh AT. Với mục đích gửi các tin nhắn cảnh báo, và nhận các tin nhắn SMS để xử lý thì mô đun SIM900A là lựa chọn hợp lý. Sơ đồ nối dây của mô đun Sim với các khối khác được trình bày trong hình 3.4 sau:



**Hình 3.4** Sơ đồ nối dây mô đun SIM900A

Chân cấp nguồn Vcc của Mô đun SIM được cấp từ nguồn 5VDC Vout của IC7805. Chân Tx Rx của mô đun lần lượt nối với chân 9 và 10 của board Arduino.

#### d. Khối cảm biến

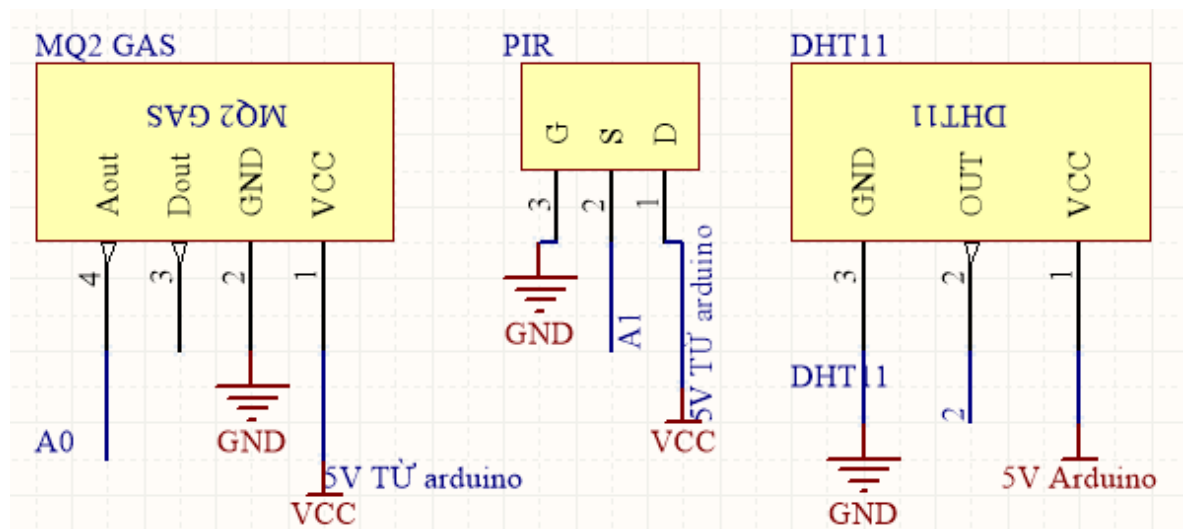
Khối cảm biến gồm các mô đun cảm biến khí gas MQ2, mô đun cảm biến chuyển động PIR, mô đun cảm biến DHT11.

Cảm biến đo nhiệt độ độ ẩm được lựa chọn sử dụng trong DATN này là DHT11, với giá thành rẻ, độ chính xác tương đối thích hợp cho ứng dụng đo nhiệt độ, độ ẩm phòng. Giao tiếp với vi điều khiển thông qua chuẩn 1 dây, đặc biệt có bộ tiền xử lý tín hiệu cảm biến, do đó dễ dàng đọc dữ liệu cảm biến. Ngoài ra tiêu tốn năng lượng rất thấp.

Mô hình trong DATN này cảm báo trộm bằng thuật toán đơn giản xử lý từ dữ liệu cảm biến hồng ngoại phát hiện chuyển động trong khoảng cách cho phép và độ chính xác tương đối, do đó lựa chọn mô đun PIR, dễ dàng điều

### CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

khuyến, chi phí rẻ. Sơ đồ nối dây chi tiết của khối cảm biến được trình bày trong hình 3.5.

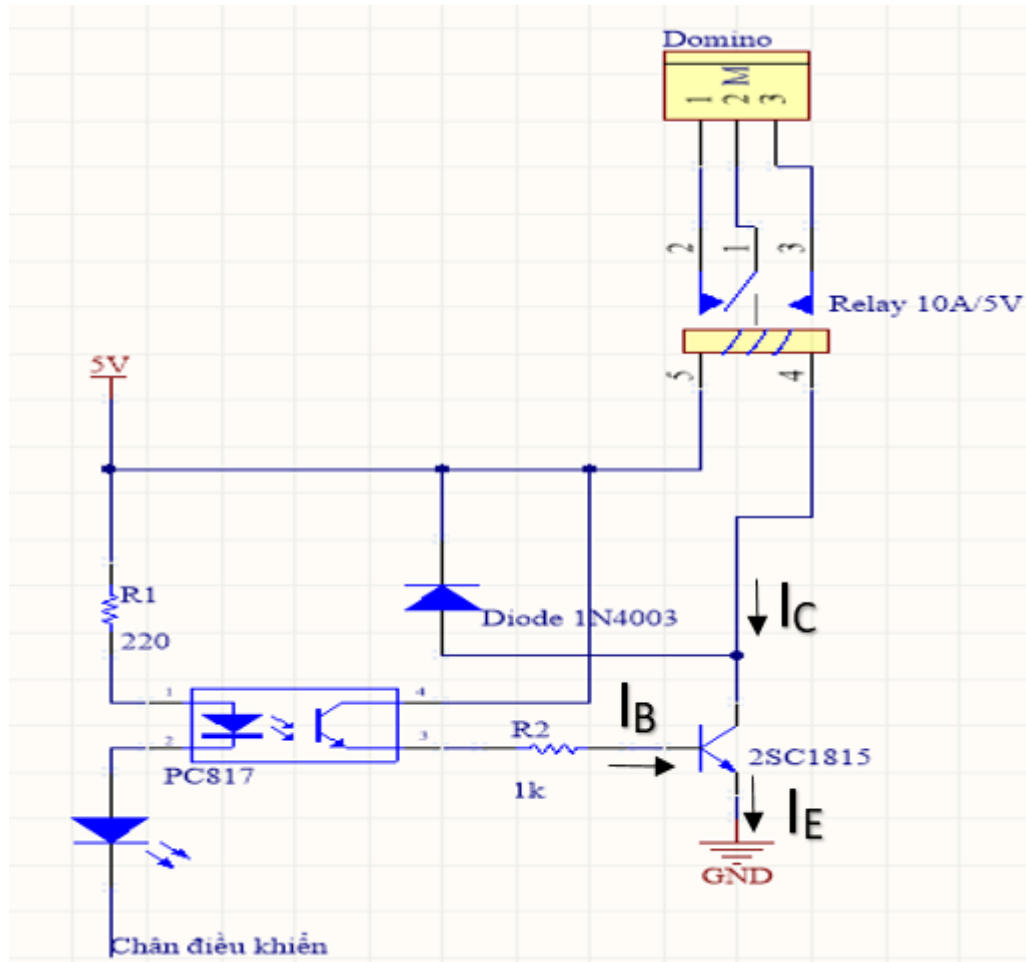


**Hình 3.5** Sơ đồ nguyên lí khối cảm biến

#### e. Khối Relay điều khiển thiết bị

Các chân D2, D5, D6, D7 của mô đun Node MCU sẽ điều khiển kích hoạt các Relay. Khối Relay này có sử dụng OPTO PC817 để cách ly giữa vi điều khiển và các tải (đèn 220VAC). Sơ đồ nguyên lí mạch Relay điều khiển thiết bị được thể hiện trong hình 3.6.





**Hình 3.6** Sơ đồ nguyên lý mạch kích Relay

**Tính toán trở  $R_2$  :**

$$R_{\text{RELAY}} = 69\Omega .$$

Chọn  $\beta = 70$  (Hệ số khuếch đại nhỏ nhất).

$$I_C = \frac{V_{CC} - V_{CE}}{R_{\text{Relay}}} = \frac{5 - 0.2}{69} = 69,56 \text{ (mA)} \quad (3.1).$$

$$\Rightarrow I_B = \frac{I_C}{\beta} = \frac{5}{69,56} = 1,03 \text{ (mA)} \quad (3.2).$$

( vậy chỉ cần  $I_B \geq 1.03\text{mA}$  thì transistor dẫn bão hòa) .

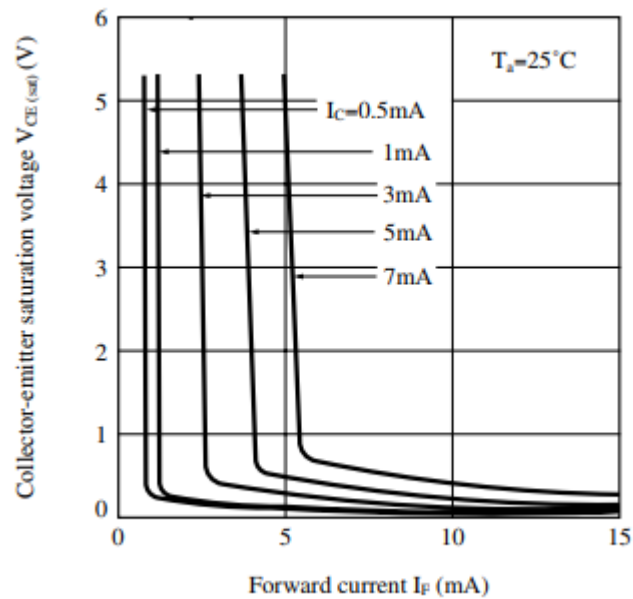
chọn  $I_B = 5\text{mA}$  .

Giá trị điện trở  $R_2$  là:

$$R_2 = \frac{V_{CC} - V_{BE}}{I_B} = \frac{5 - 0.7}{0.005} = 860 \text{ (}\Omega\text{)} \quad (3.3).$$

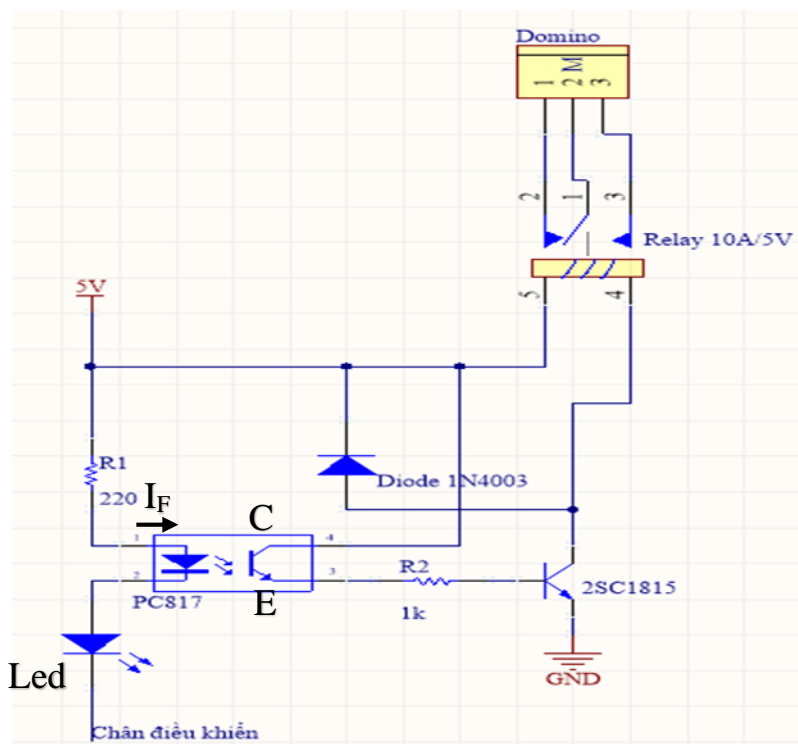
Chọn  $R_2 = 1\text{k}\Omega$ .

**Tính toán trở  $R_1$  cho opto :**



**Hình 3.7** Đặc tuyến điện áp bão hòa  $V_{CE}$  và dòng phân cực  $I_F$

Dựa vào đặc tuyến điện áp ngõ ra trong hình 3.7, nếu  $I_C = 5\text{mA}$  và dòng phân cực  $I_F$  từ 5 mA đến 15mA thì điện áp  $V_{CE}$  nằm trong vùng bão hòa với giá trị gần bằng 0V. Khi này Relay sẽ được kích hoạt.



**Hình 3.8** Mạch nguyên lý kích hoạt Relay

**Vled:** điện áp rơi trên led khi led hoạt động tầm 2V .

### CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

---

Chân điều khiển mức 0 là 0V (kích hoạt Relay), mức 1 (ngắt kích hoạt Relay ) là 3,3V.

Lựa chọn giá trị dòng  $I_F = 5\text{mA}$ , giá trị điện trở  $R_1$  cao nhất mà vẫn đảm bảo opto dẫn là:

$$R_1 = \frac{V_{CC} - V_{led}}{I_F} = \frac{5V - 2V}{5\text{mA}} = 600 (\Omega) \quad (3.4).$$

Khi  $I_F = 15\text{mA}$  Giá trị điện trở  $R_1$  nhỏ nhất mà vẫn đảm bảo opto dẫn là:

$$R_1 = \frac{V_{CC} - V_{led}}{I_F} = \frac{5V - 2V}{15\text{mA}} = 200 (\Omega) \quad (3.5).$$

Do đó lựa chọn  $200 \Omega < R_1 < 600 \Omega$

Chọn  $R_1 = 220 \Omega$ .

*Diode 1n4003* được sử dụng với mục đích bảo vệ transistor bằng cách dập xung điện áp, do cuộn dây Relay xả ra khi ngắt kích hoạt .

#### f. Khối hiển thị

khối hiển thị được sử dụng trong ĐATN gồm 1 LCD 16X2 hiển thị trạng thái bốn đèn và giá trị nhiệt độ, độ ẩm.

#### g. Led báo hiệu enable chế độ báo trộm, buzzer

Led báo hiệu cho biết chế độ báo trộm có đang ở chế độ enable hay không, còn buzzer thì phát ra tiếng kêu để cảnh báo. Giá trị điện trở mắc nối tiếp với Led báo hiệu là:

$$R_{led} = \frac{V}{I} = \frac{5V}{15\text{mA}} = 333.33 (\Omega) \quad (3.6).$$

=> chọn  $R_{led} = 220 \Omega$

Theo thông số kỹ thuật của buzzer, khi buzzer reo thì tổn dòng khoảng  $< 25\text{mA}$  nên dùng chân từ Arduino kích hoạt trực tiếp được, vì dòng cấp ra từ chân Arduino lên tới  $36\text{mA}$ .

#### h. Khối Website

Để tiện cho việc điều khiển thiết bị qua Internet nhóm đã thiết kế 1 website với giao diện web đơn giản gồm các nút điều khiển đèn ON OFF, và

### CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

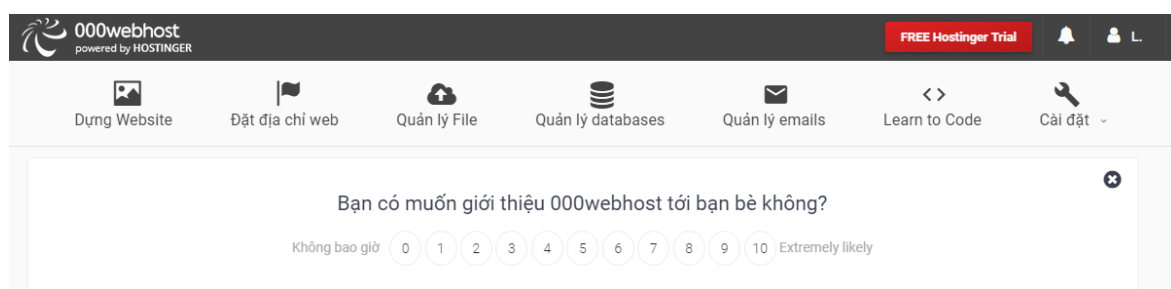
hiển thị giá trị nhiệt độ, độ ẩm. Để xây dựng Website này cần phải tạo được 1 webserver chứa dữ liệu của website và lập trình cho website với các tính năng như trên.

#### *Tạo 1 hosting(webserver)*

Tiến hành tạo 1 webserver miễn phí trên trang web <https://www.hostinger.vn/free-hosting>

Nhà cung cấp **Hosting Free** cho phép người dùng xây dựng được 1 server hoàn toàn miễn phí, người dùng sẽ được cấp 1000 MB dung lượng lưu trữ và không giới hạn băng thông. Không đặt bất kỳ giới hạn về kết nối web, là một giải pháp lớn trong việc học tập và sử dụng cho mục đích cá nhân. Các bước đăng ký một hosting miễn phí:

- Bước 1 : Đầu tiên truy cập vào địa chỉ <https://www.hostinger.vn/free-hosting> này và tạo 1 tài khoản.
- Bước 2: Sau khi đăng kí xong ta tiếp tục truy cập vào địa chỉ <https://www.000webhost.com> và tiến hành đăng nhập vào tài khoản vừa tạo và được hình như bên dưới.



**Hình 3.9** Giao diện quản lý website sau khi đăng nhập

Trong giao diện quản lý như hình 3.9, người dùng có thể upload nội dung của trang web trong mục “Quản lý File”. Hình 3.10 sau đây trình bày giao diện đã thiết kế trong DATN.

**ĐỒ ÁN TỐT NGHIỆP 2018**

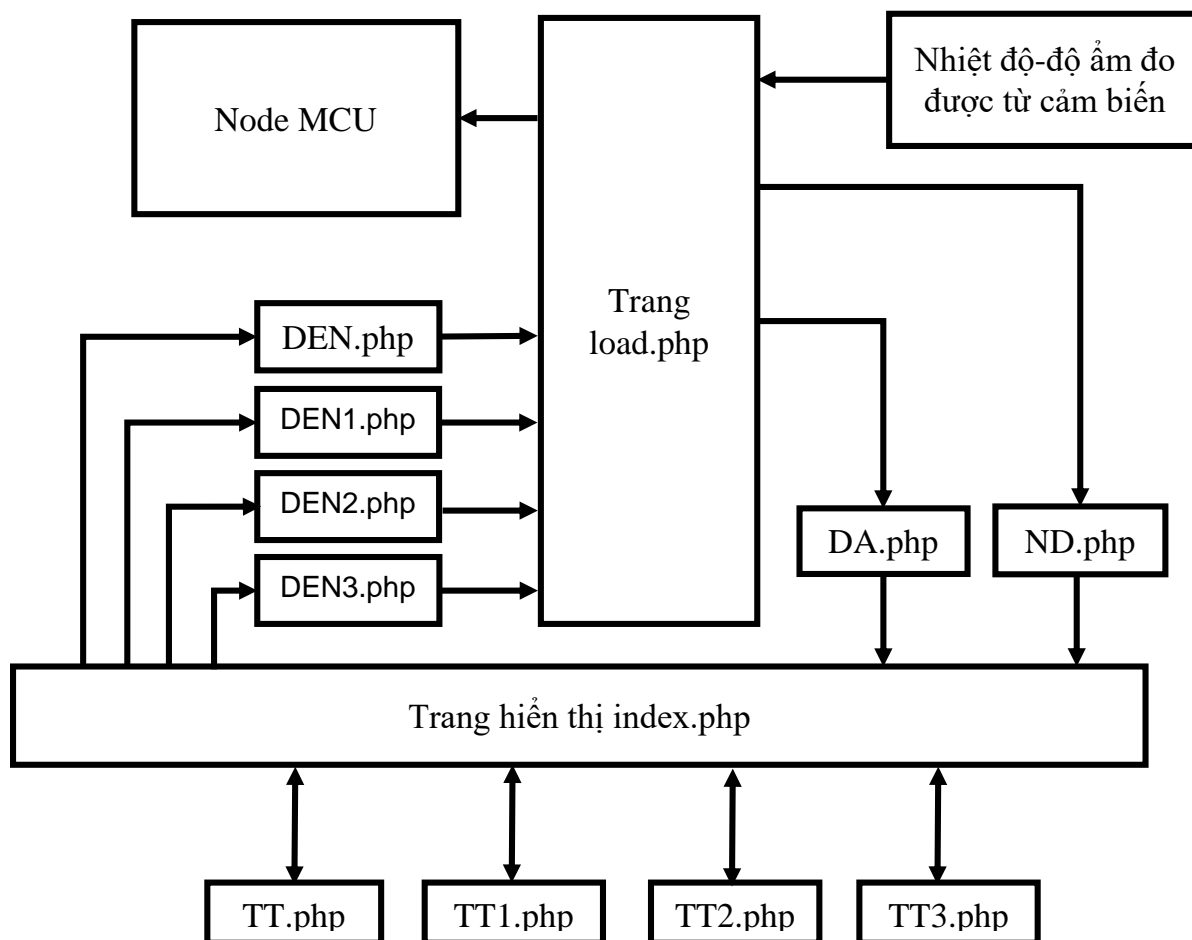
Nhiệt Độ(°C)		Độ Ẩm(%)	
32		91	

ON		OFF	
	Bật Đèn 0		Tắt Đèn 0
	Bật Đèn 1		Tắt Đèn 1
	Bật Đèn 2		Tắt Đèn 2
	Bật Đèn 3		Tắt Đèn 3
	Bật Tất Cả		Tắt Tất Cả

ĐÈN 0	ĐÈN 1	ĐÈN 2	ĐÈN 3
ON	ON	ON	ON

**Hình 3.10** Giao diện website đã thiết kế trong DATN

Sơ đồ khối hình 3.11 sau đây sẽ mô tả cách thức hoạt động và cách xây dựng website trong đồ án này :



**Hình 3.11** Sơ đồ khối cấu tạo của hệ thống website

Cấu tạo, chức năng từng trang trong sơ đồ:

- Trang Index.php: là trang hiển thị giao diện chính của trang web, trang này chứa code tạo giao diện web như tạo nút nhấn, tạo các label gắn hiển thị nhiệt độ, độ ẩm, các trạng thái đèn. Các nút nhấn ON0, OFF0, ON1, OFF1 ON2, OFF2, ON3, OFF3 khi được nhấn trên trang giao diện này sẽ thực hiện mở các trang tương ứng den.php, den1.php, den2.php, den3.php TT.php, TT1.php, TT2.php, TT3.php và lưu vào giá trị đèn là bật hay tắt vào.

- Trang den.php, den1.php, den2.php, den3.php: lưu trạng thái đèn từ nút nhấn trên trang index.php.

- Trang TT.php, TT1.php, TT2.php, TT3.php: lưu trạng thái đèn từ nút nhấn, và được gán cho các label trạng thái đèn ngoài trang index.php để hiển thị trạng thái đèn.

### CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

---

- Trang load.php: trang có nhiệm vụ lấy các giá trị cảm biến đưa lên từ Node MCU qua phương thức GET và ghi vào trang ND.php, DA.php. Ngoài ra trang này còn đọc các giá trị điều khiển đèn trong các trang den.php, den1.php, den2.php, den3.php, sau đó gửi phản hồi về Node MCU, nhờ đó điều khiển được các thiết bị qua website.

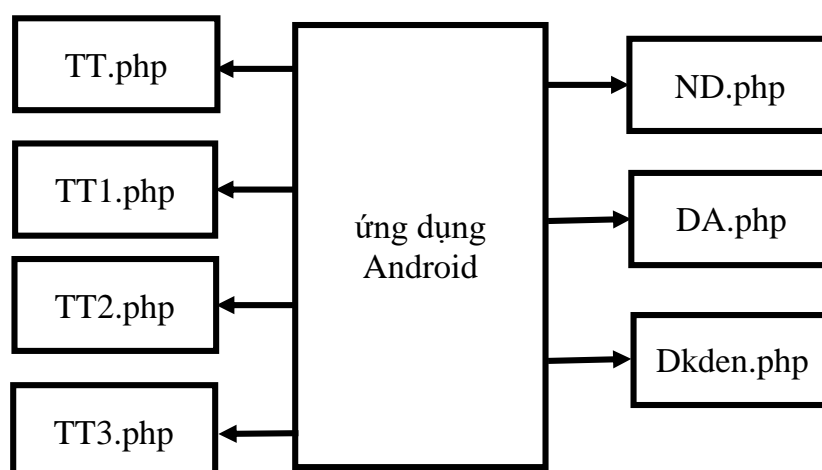
- Trang ND.php, DA.php: lưu các giá trị nhiệt độ, độ ẩm, được gán cho các label hiển thị nhiệt độ, độ ẩm ở trang index.php.

#### j. Ứng dụng Android

Để thuận tiện hơn nữa trong việc điều khiển các thiết bị nhóm cũng đã xây dựng thêm một app Android. Ứng dụng này hoạt động dựa trên các trang nền tảng của website.

Ứng dụng Android được xây dựng với giao diện gồm các nút nhấn bật đèn 0, tắt đèn 0, bật đèn 1, tắt đèn 1, bật đèn 2, tắt đèn 2, bật đèn 3, tắt đèn 3 và các label để hiển thị trạng thái đèn, nhiệt độ, độ ẩm.

Sơ đồ khối hình 3.12 mô tả hoạt động của ứng dụng Android.:



**Hình 3.12** Sơ đồ hoạt động của ứng dụng Android

Giải thích sơ đồ :

Ứng dụng Android sẽ truy cập vào các trang ND.php, DA.php để lấy các giá trị nhiệt độ, độ ẩm và hiển thị lên giao diện. Tương tự trạng thái đèn hiển thị trên app cũng được lấy từ các trang TT.php, TT1.php, TT2.php, TT3.php.

Các nút nhấn trên giao diện ứng dụng khi được kích hoạt tương ứng sẽ gửi các trạng thái đèn đến trang dkden.php. Trang dkden.php kiểm tra nút nhấn nào được nhấn sẽ mở các trang den.php, den1.php, den2.php, den3.php tương ứng và ghi trạng thái mới của đèn vào.

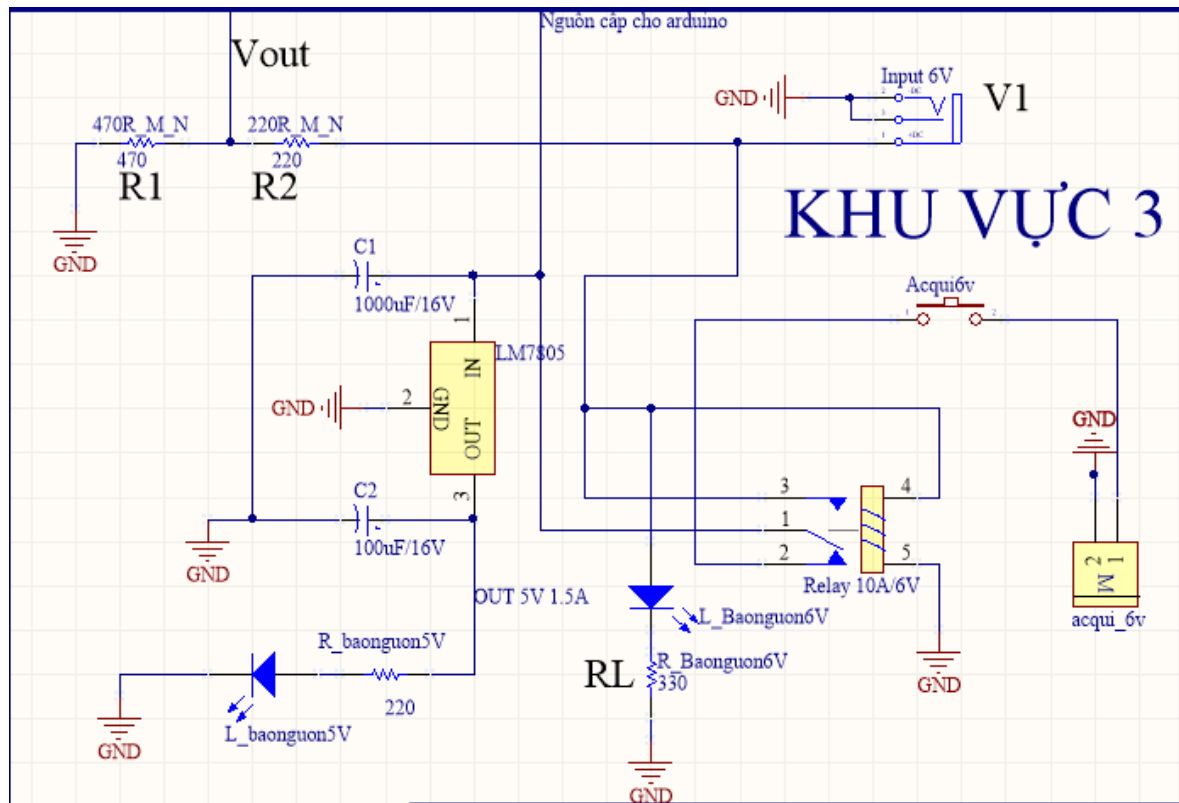
#### **j. Khối nguồn**

Mạch nguồn có chức năng cấp nguồn cho các khối hoạt động. Khối nguồn được thiết kế thêm mạch tự động chuyển sang nguồn Acqui dự trữ khi mất nguồn điện chính cấp trong mạch. Ngoài ra, có trích xuất điện áp từ apdapter và được đọc bởi 1 chân của Arduino; sau đó, so sánh với trạng thái thiết bị lưu trong bộ nhớ EPROM xác định trạng thái mất điện của mạch, nhờ đó gửi thông báo cho người sử dụng. Khối nguồn được thiết kế như hình 3.13.

#### **Cách hoạt động của mạch nguồn:**

- Chân báo mất nguồn: Chân này là Vout của mạch phân áp từ nguồn 6VDC của adapter cấp cho mạch. Khi mất điện vào adapter thì giá trị sẽ chân này là 0V. Được nối đến chân 22 của board Arduino, Arduino đọc giá trị chân này so sánh với giá trị đã lưu trước của chân này trong bộ nhớ EPROM, nhờ đó xác định được trạng thái của hệ thống đang dùng nguồn dự phòng hay nguồn điện chính 220VAC. Giá trị R1 và R2 (hình 3.13) trong mạch phân áp này được lựa chọn sao cho giá trị Vout là mức 1 đối với Arduino.





**Hình 3.13** Sơ đồ nguyên lí mạch nguồn

R1 và R2 được lựa chọn lần lượt là 470  $\Omega$  và 220  $\Omega$ . Theo như sơ đồ trên khi có điện thì áp chân báo mất nguồn là:

$$V_{out} = V_1 \times \frac{R_1}{R_1 + R_2} = 6 \times \frac{470}{220 + 470} = 4,08 \text{ (mức 1 đối với Arduino)} \quad (3.7).$$

$V_1$ : điện áp 6VDC từ adapter.

Giá trị điện trở cho led báo nguồn 6VDC từ adapter là :

$$R_L = \frac{V_1 - V_{led}}{I_{led}} = \frac{6V - 2V}{10mA} = 360 \text{ } (\Omega) \quad (3.8).$$

=> chọn  $R_L = 330 \Omega$ .

- Kichhoatnguồn6V là một switch cho phép dùng nguồn dự trữ acqui 6v hay không (khi mất điện sẽ chuyển sang nguồn dự phòng).

- Cách tự động chuyển mạch nguồn dùng Relay như sau: khi có điện tại 6Vinput(nguồn điện từ Adapter) thì Relay kích hoạt nối chân 1 với 3 lại với nhau. Ngược lại, khi mất điện tại adapter 6V Relay ngắt, kích hoạt chân 1 với 2 nối lại với nhau lúc này acqui 6V được sử dụng. IC LM7805 được cấp Vin,

### CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

---

ở ngõ ra IC LM7805 ta thu được điện áp 5V cấp cho mô đun sim và Node MCU (cả 2 mô đun hoạt động ở điện áp 5v).

- Nguồn cấp cho Arduino là 6V .

#### Tính dòng và chọn linh kiện mạch nguồn :

Theo các thông số kỹ thuật của các cảm biến :

- Cảm biến PIR dùng áp 5v và tiêu tốn dòng khoảng  $\leq 50\mu A$ .
- Cảm biến DHT11 dùng áp 5V tiêu tốn dòng khoảng  $< 2.5mA$ .
- Cảm biến khí Gas dùng áp 5V tiêu tốn dòng khoảng 150mA.
- LCD 16x2 dùng áp 5V và tốn dòng khoảng 1mA .

➔ Tổng dòng cấp cho 3 cảm biến và LCD là:

$$I_1 = I_{PIR} + I_{Gas} + I_{DHT11} = 2.5 + 150 + 0.05 + 1 = 153,55 \text{ (mA)} \quad (3.9).$$

- Board Arduino Mega2560 điều khiển 1 buzzer và 1 led báo hiệu nên tốn :

$$I_3 = I_{LED} + I_{Buzzer} = 15 + 25 = 50 \text{ (mA)} \quad (3.10).$$

- Board Arduino Mega2560 cấp dòng tổng tại chân 5V khoảng 500mA và tiêu thụ dòng  $I_{Arduino} = 30mA$ .

➔ Chọn Board Arduino Mega2560 cấp nguồn cho 3 cảm biến, buzzer và LCD.

Dựa vào thông số kỹ thuật của các linh kiện :

- Node MCU dùng điện áp 5V và tiêu tốn dòng  $\leq 170mA$  .
- Mô đun SIM900A dùng áp 5V và tiêu tốn dòng tối đa 200mA.
- 4 Relay 5V tốn : 280mA. .
- 1 led báo nguồn 5v tốn 15mA.

Tổng dòng cấp cho Node MCU, SIM900A, 4 Relay và 1 led báo nguồn là:

$$I_2 = I_{Node} + I_{Sim} + I_{Relay} + I_{Led} = 170 + 280 + 200 + 20 = 670 \text{ (mA)} \quad (3.11).$$

➔ Chọn IC LM7805 có Vout 5V và dòng tối đa lên đến 1,5A (nếu tản nhiệt tốt) cấp cho Node MCU, mô đun SIM900A, 4 Relay và 1 led báo nguồn.

Tổng dòng điện tiêu tốn của các linh kiện trong hệ thống là:

$$I = I_1 + I_2 + I_3 + I_{\text{led6V}} + I_{\text{Arduino}} = 153,55 + 670 + 50 + 13 + 30 = 915,55 \text{ (mA)}$$

(3.12).

⇒ Chọn adapter 6v/2A cấp cho toàn hệ thống.

⇒ Nguồn dự phòng chọn acqui 6V/4.5 Ah.

#### 3.2.3 Sơ đồ nguyên lý của toàn mạch

##### **Giải thích sơ đồ nguyên lý toàn mạch :**

**Sơ đồ nguyên lý toàn mạch được trình bày trong hình sau, chi tiết các khối được mô tả như sau :**

Khu vực 1: Là khu vực bao gồm các cảm biến GAS, PIR , DHT11, LCD , nút nhấn reset, nút nhấn enable, buzzer và đèn báo hiệu. Các cảm biến khu vực này dùng nguồn 5V từ Arduino được Arduino đọc các giá trị cảm biến. Led , buzzer và LCD được điều khiển trực tiếp bằng các chân digital của Arduino.

Khu vực 2: Bao gồm Node MCU và mạch Relay, Node MCU lấy dữ liệu từ website và xuất các giá trị tại các chân D2 D7 D6 D5 để điều khiển mạch Relay. Node dùng nguồn 5V từ LM7805.

Khu vực 3: Là khu vực mạch nguồn, mạch chuyển nguồn dự phòng, và mạch báo mất nguồn. Điện áp cấp cho mạch nguồn là adapter 6V/2A. Nguồn dự phòng là 1 acqui 6V/4.5 Ah.

Còn riêng mô đun SIM900A(dùng nguồn LM7805) để gửi tin nhắn cảnh báo tới người sử dụng (Arduino đọc các giá trị cảm biến và xử lý sau đó truyền UART tới mô đun sim để gửi tin nhắn).



## **Chương 4. THI CÔNG HỆ THỐNG**

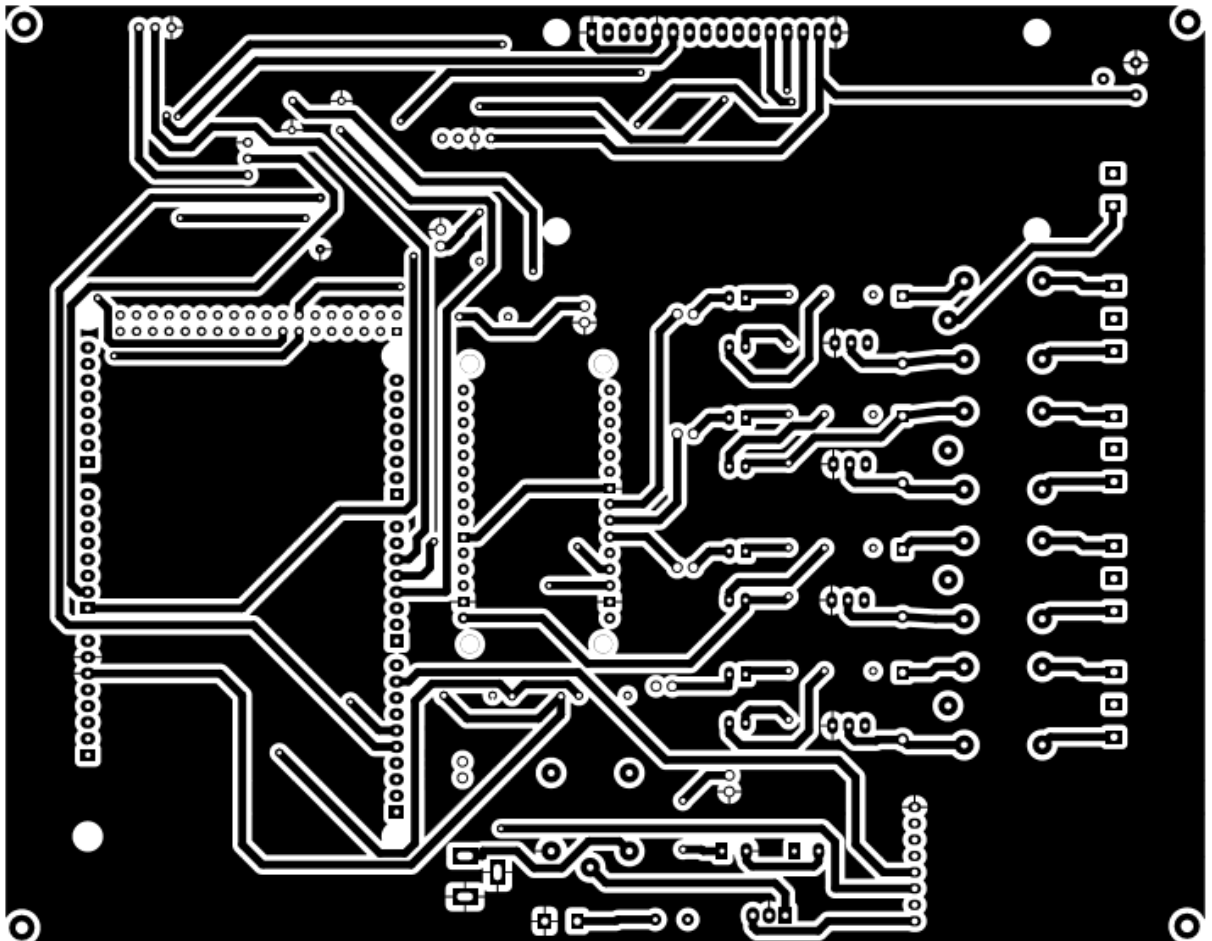
### **4.1 GIỚI THIỆU**

Trong chương này nhóm sẽ trình bày về sơ đồ mạch in lớp trên, lớp dưới, sơ đồ bố trí linh kiện, lập danh sách các linh kiện, lắp ráp và kiểm tra, lập trình hệ thống, viết tài liệu hướng dẫn sử dụng.

### **4.2 THI CÔNG HỆ THỐNG**

#### **4.2.1 Mạch in lớp dưới**

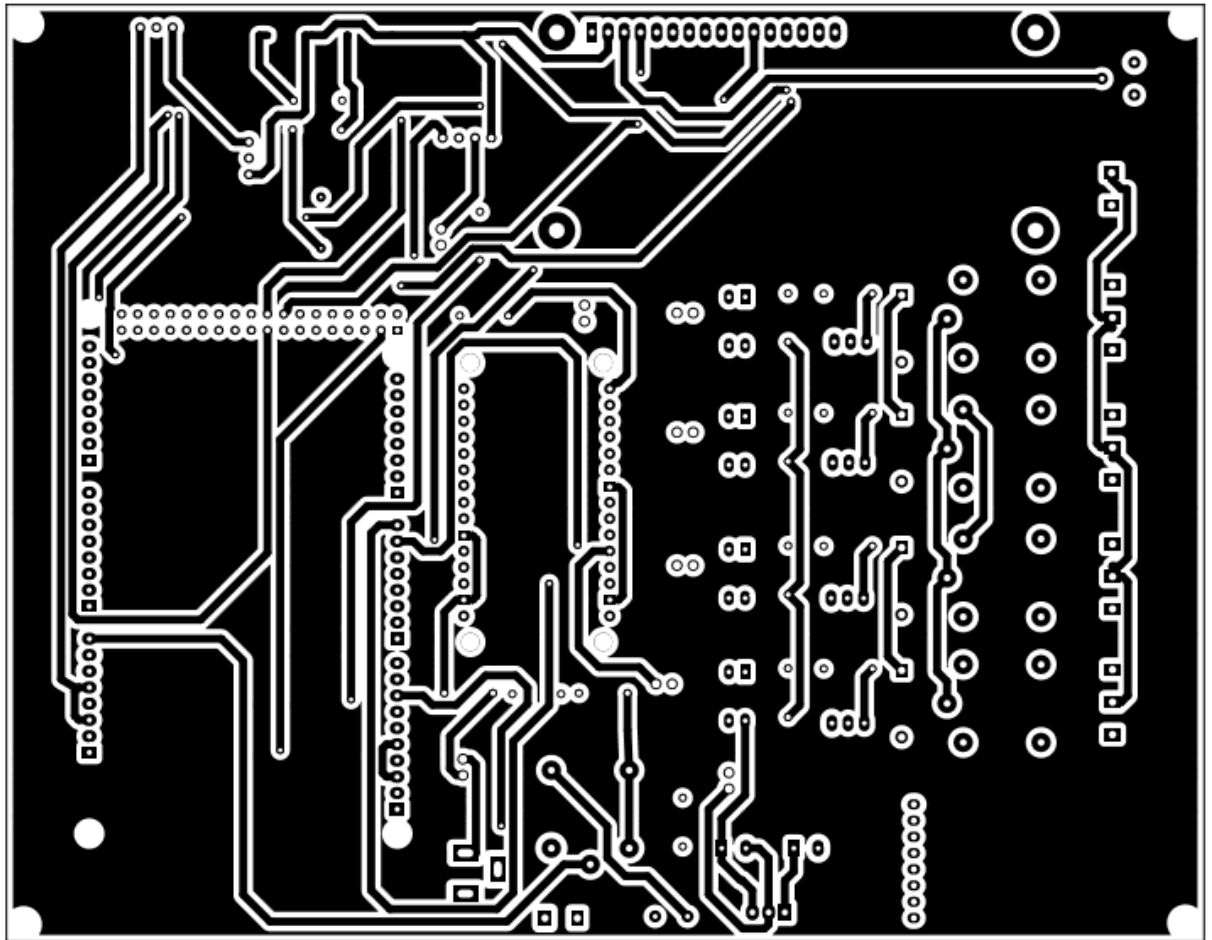
Mạch in trong DATN này được thiết kế bằng phần mềm Altium designer, hình 4.1 là mạch in lớp dưới.



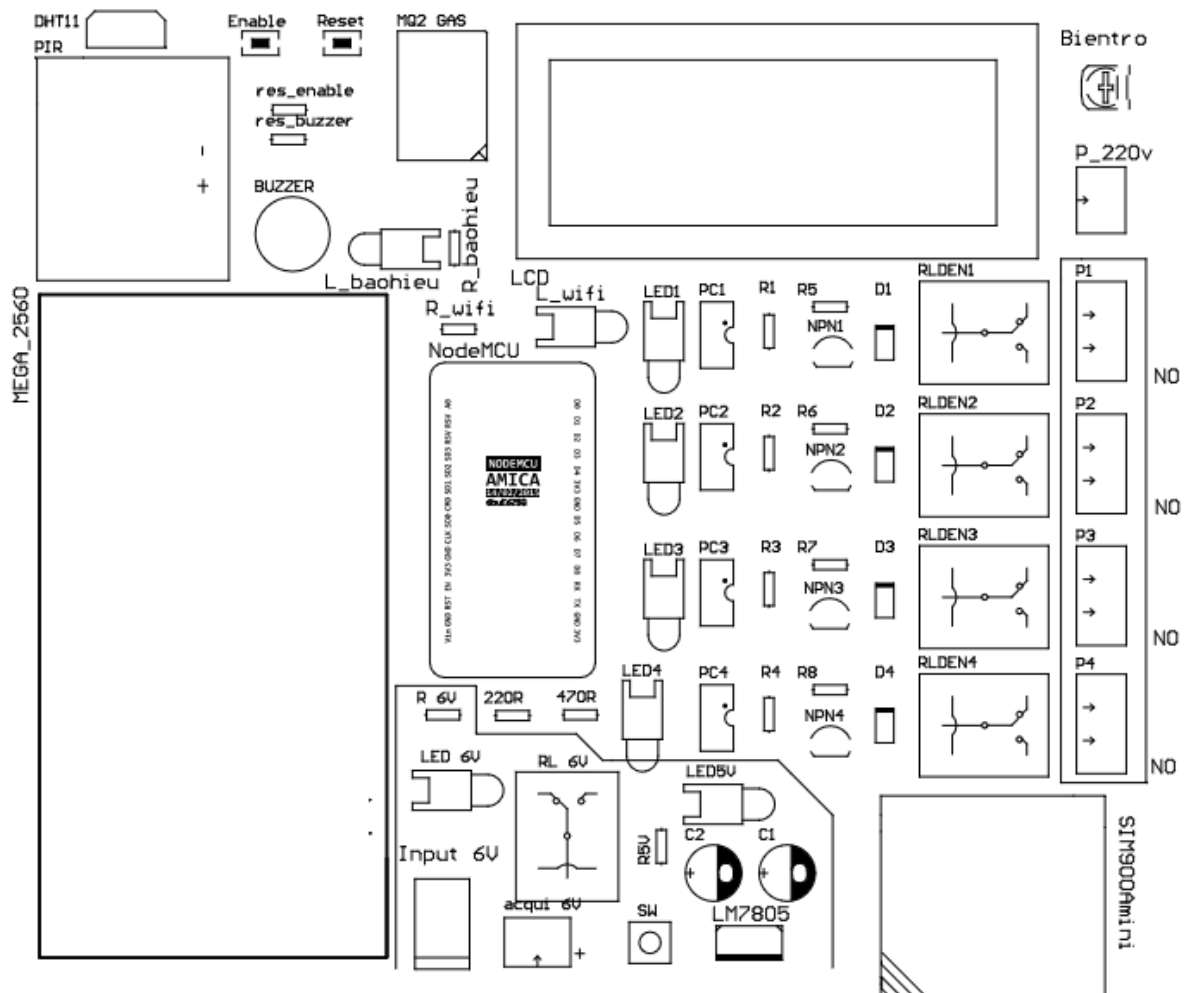
**Hình 4.1** Mạch in lớp dưới

### 4.2.2 Mạch in lớp trên

Hình 4.2 sau đây là mạch in lớp trên.



**Hình 4.2** Mạch in lớp trên



STT	Tên linh kiện	Giá trị	Chú thích
1	Board Arduino mega 2560	R3	Arduino mega 2560
2	Node MCU	V1.0	Mô đun wifi
3	Mô đun Sim 900A	MiniV2	

#### CHƯƠNG 4. THI CÔNG HỆ THỐNG

4	PIR		Cảm biến chuyển động
5	Mô đun GAS MQ2	MQ2	Cảm biến khí Gas
6	Mô đun DHT11		Cảm biến nhiệt độ-độ ẩm
7	LCD	16X2	Nền xanh da trời
8	Nút nhấn	2 chân	Nút nhấn dán
9	Trở	220 $\Omega$ , 680 $\Omega$ ,470 $\Omega$ ,1k $\Omega$ .	
10	LED	Trắng, đỏ, xanh	
11	Diode	1N3607	
12	Domino	2 chân , 3 chân	
13	Relay	5V/10A, 6V/10A	
14	Transistor	2SC1815	
15	Opto	PC817	
16	Buzzer	5V	
17	Biến trở tròn	5k $\Omega$	
18	Jack DC	5.5x2.1mm	

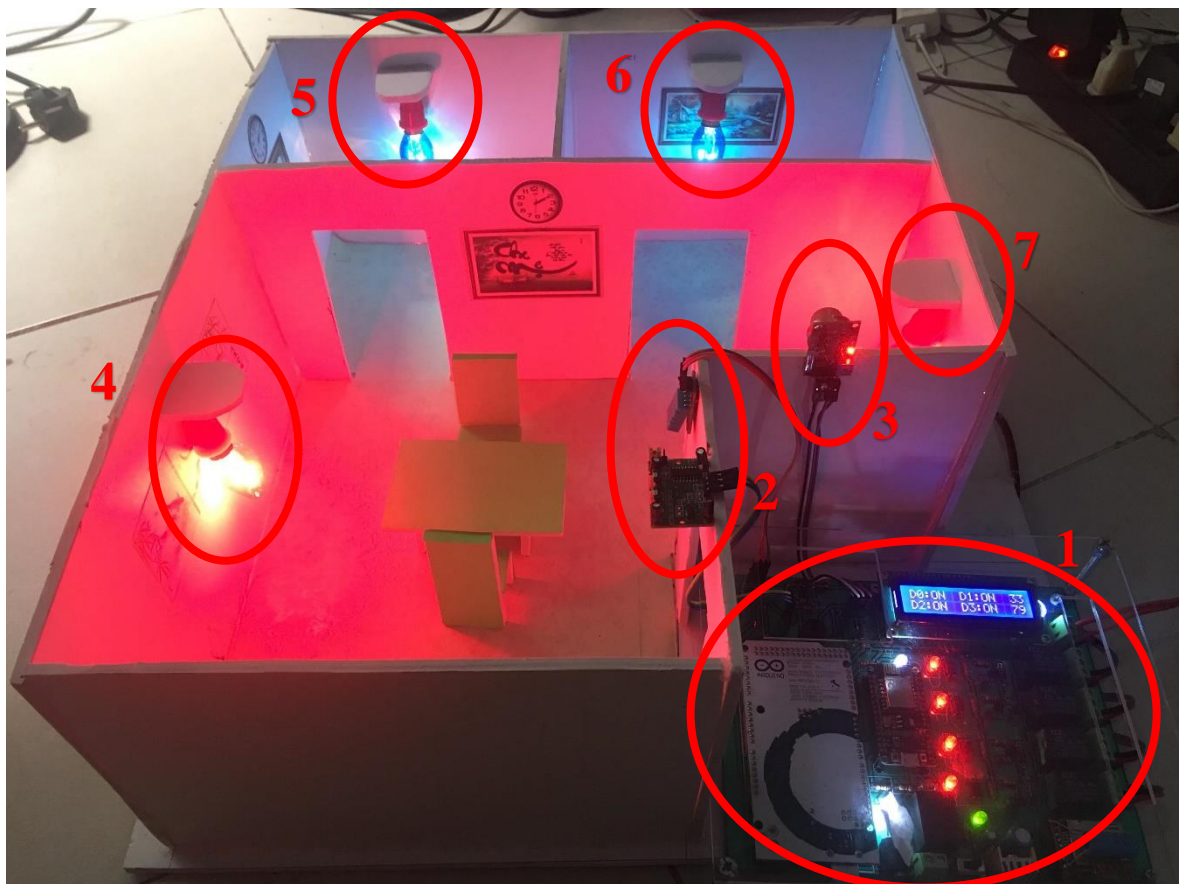


## CHƯƠNG 4. THI CÔNG HỆ THỐNG

19	Tụ	1000uF , 100uF	
20	LM7805	1.5A	Có tản nhiệt
21	Switch	2 chân	
22	Acqui	6V/4.5Ah	

### 4.3 THI CÔNG MÔ HÌNH

Mô hình của hệ thống được thể hiện trong hình 4.4 sau:



**Hình 4.4** Mô hình DATN

Mô hình này bao gồm:

- Vị trí số 1: board mạch chính điều khiển của hệ thống.
- Vị trí số 2: cảm biến chuyển động được đặt tại cửa ra vào để phát hiện trộm, cảm biến nhiệt độ đặt tại phòng khách để đo nhiệt độ phòng.
- Vị trí số 3: cảm biến khí gas đặt tại phòng bếp để phát hiện khí gas rò rỉ.
- Vị trí số 4,5,6,7: lần lượt là các đèn phòng khách, phòng ngủ 1, 2 và đèn phòng bếp.

Cụ thể chi tiết về mô hình sẽ được trình bày trong phần kết quả.

### 4.4 LẬP TRÌNH HỆ THỐNG

#### 4.4.1 Lưu đồ giải thuật

Hoạt động của cả hệ thống từ khi được cấp điện cho đến khi hệ thống ngừng hoạt động như sau:

- Lần đầu cấp điện cho mạch thì chế độ báo trộm không được kích hoạt, các chế độ báo cháy, báo gas rò rỉ hoạt động bình thường, mô đun SIM900A được khởi tạo và chờ nhận tin nhắn điều khiển. Board Arduino sẽ đọc các giá trị cảm biến, truyền UART giá trị nhiệt độ, độ ẩm đến Node MCU. LCD hiển thị nhiệt độ - độ ẩm và các trạng thái đèn hiện tại nhận từ Node MCU.

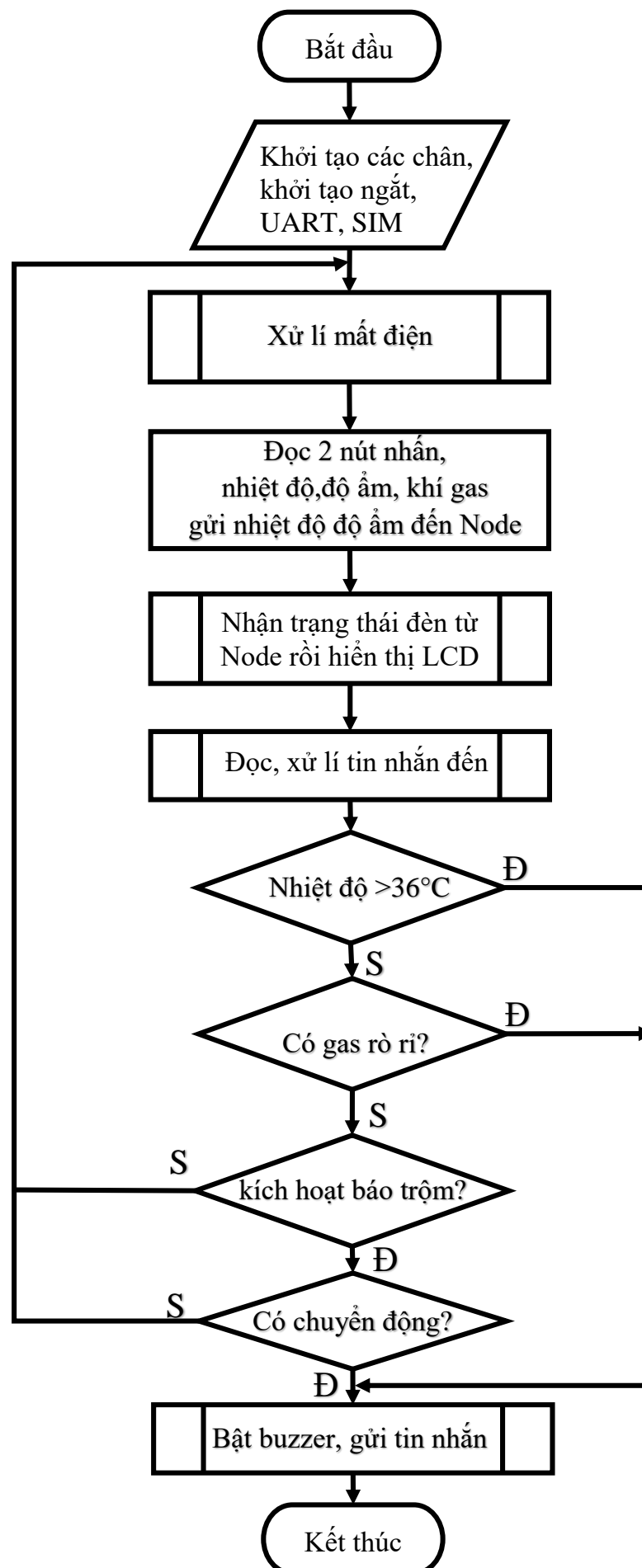
- Arduino đọc giá trị điện áp từ mạch phân áp từ adapter so sánh với giá trị trạng thái hệ thống trước đó từ bộ nhớ EPROM của Arduino đánh giá nếu có mất điện hay có điện sẽ gửi tin nhắn thông báo. Sau đó, lưu trạng thái mới của hệ thống vào bộ nhớ EPROM.

- Nhấn nút enable được nhấn thì sẽ kích hoạt chế độ báo trộm, đèn báo hiệu sáng. Nhấn lần nữa sẽ hủy kích hoạt (ta có thể bật và tắt kích hoạt chống trộm bằng tin nhắn). Nhấn nút reset chuông thì sẽ reset buzzer và đặt các chế độ cảnh báo về trạng thái ban đầu như lúc mới cấp điện (ta cũng có thể reset bằng tin nhắn). 2 nút nhấn này được xử lý là 2 ngắt ngoài của Arduino.

- Khi có phát hiện trộm(phát hiện chuyển động), nhiệt độ cao(nhiệt độ ngưỡng đặt mức báo động là 36°C), phát hiện rò khí gas thì buzzer sẽ reo và sau đó mô đun SIM900A gửi tin nhắn cảnh báo về 1 số điện thoại đã được cài sẵn.

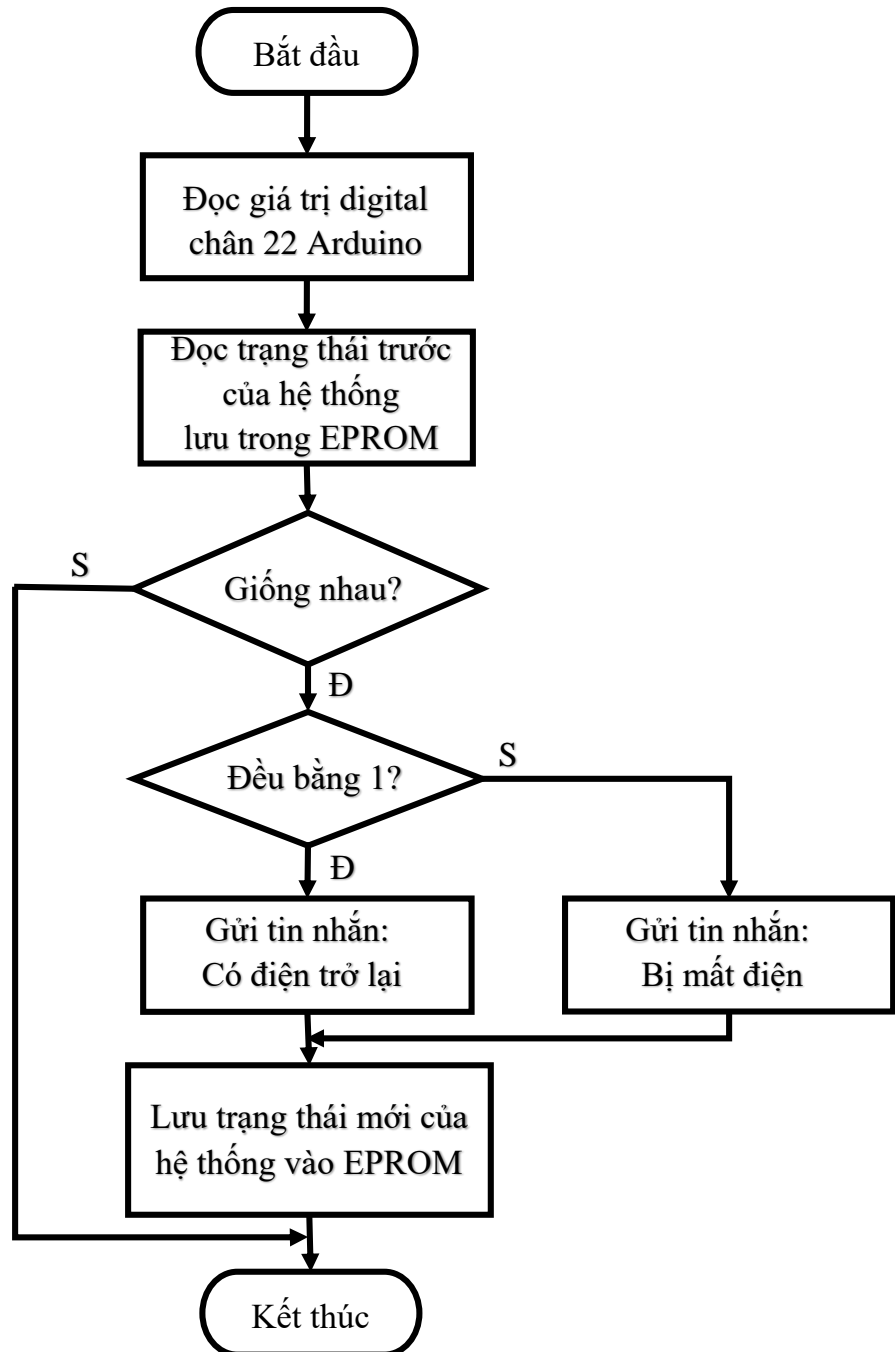
- Khi mất điện chính thì Relay nguồn nhà sang nguồn acqui dự phòng, sẽ có tin nhắn gửi cảnh báo mất nguồn chính. Đến khi nguồn chính có điện thì Relay sẽ nhảy sang dùng nguồn chính và sẽ gửi tin nhắn báo có điện.

**Lưu đồ chương trình chính cho board Arduino được trình bày trong hình 4.5 sau:**



**Hình 4.5** Lưu đồ mạch điều khiển Arduino

Lưu đồ chương trình con xử lý mất điện được trình bày như trong hình 4.6:

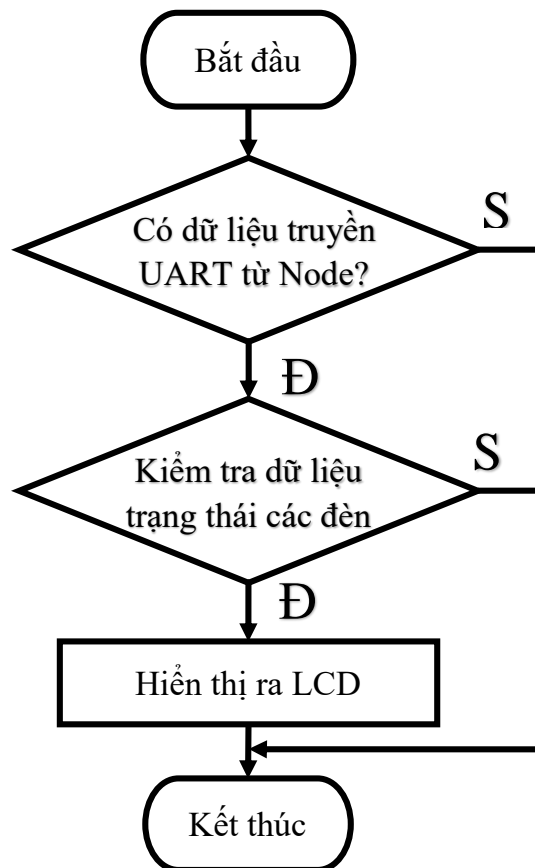
**Hình 4.6** Lưu đồ chương trình con xử lý mất điện

**Giải thích:** Arduino đọc giá trị digital từ chân 22, chân này là Vout của mạch phân áp từ adapter, sau đó so sánh với giá trị trạng thái hệ thống trước đó từ bộ nhớ EPROM của Arduino. Giá trị chân này bằng 1 khi dùng nguồn từ

## CHƯƠNG 4. THI CÔNG HỆ THỐNG

adapter và bằng 0 khi dùng nguồn từ acqui. Đánh giá nếu có mất điện hay có điện sẽ gửi tin nhắn thông báo. Sau đó, lưu trạng thái mới của hệ thống vào bộ nhớ EPROM.

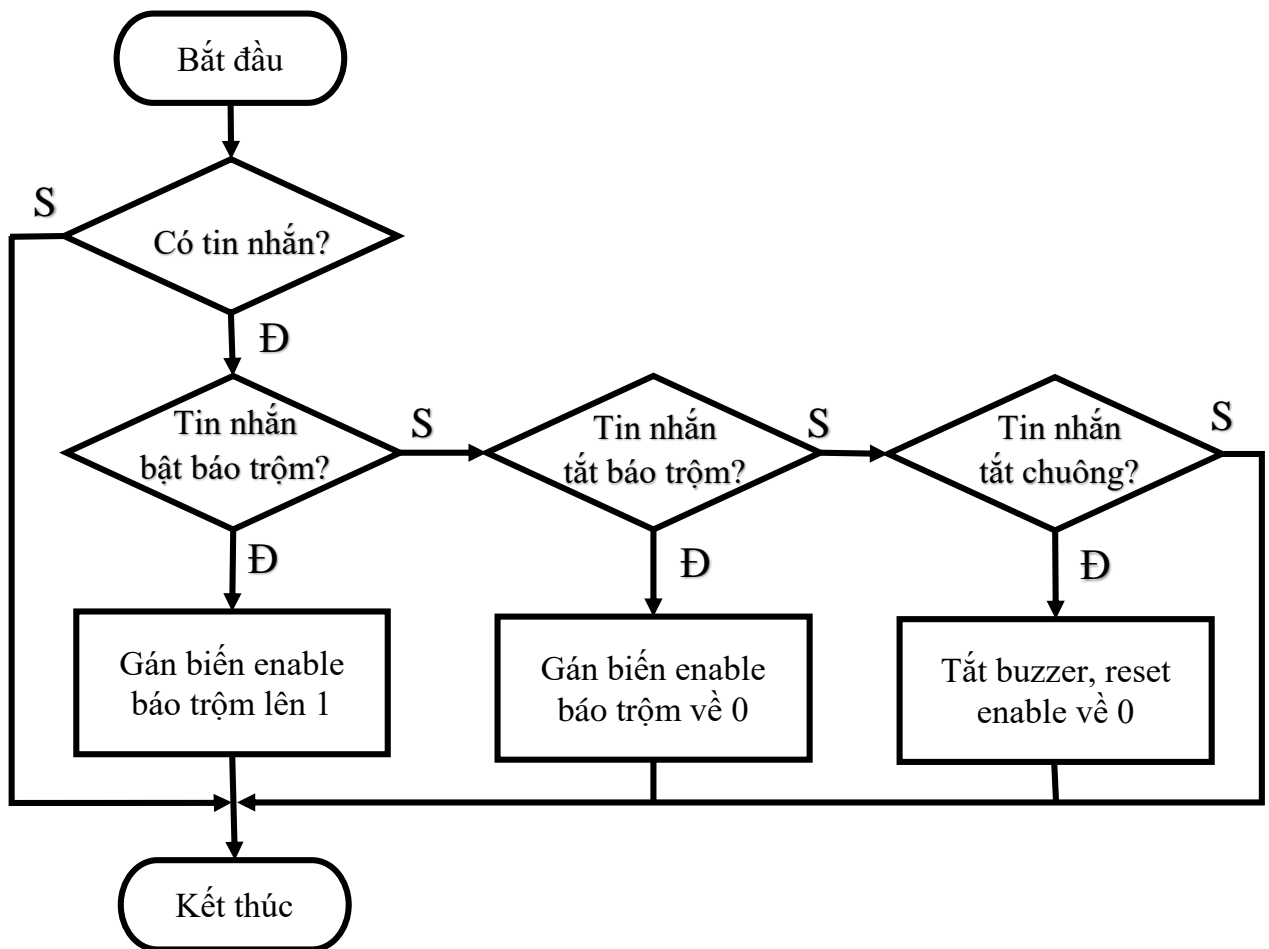
**Lưu đồ chương trình con nhận trạng thái đèn từ Node MCU và hiển thị ra LCD được trình bày ở hình 4.7.**



**Hình 4.7** Lưu đồ chương trình con nhận trạng thái đèn từ Node MCU và hiển thị ra LCD

**Giải thích:** Khi có dữ liệu truyền về từ Node MCU, Arduino sẽ xử lý chuỗi này và biết được trạng thái đèn từ dữ liệu nhận được này, sau đó sẽ hiển thị trạng thái đèn tương ứng ra LCD.

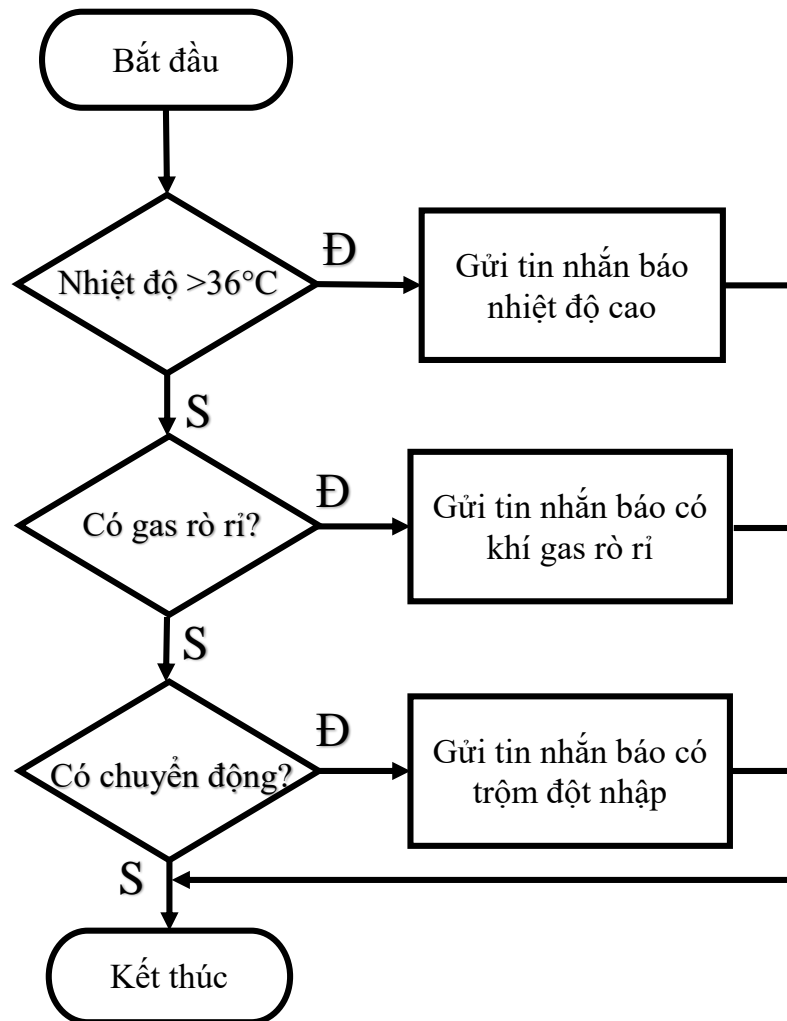
Lưu đồ chương trình con xử lý tin nhắn đến được trình bày trong hình 4.8 sau:



**Hình 4.8** Lưu đồ chương trình con xử lý tin nhắn đến

**Giải thích:** chương trình này kiểm tra cú pháp tin nhắn đến, so sánh với lệnh trong code để đưa ra các điều khiển tương ứng. Cụ thể: là thực thi khi có các tin nhắn bật tính năng báo trộm, tắt tính năng báo trộm và tắt chuông buzzer.

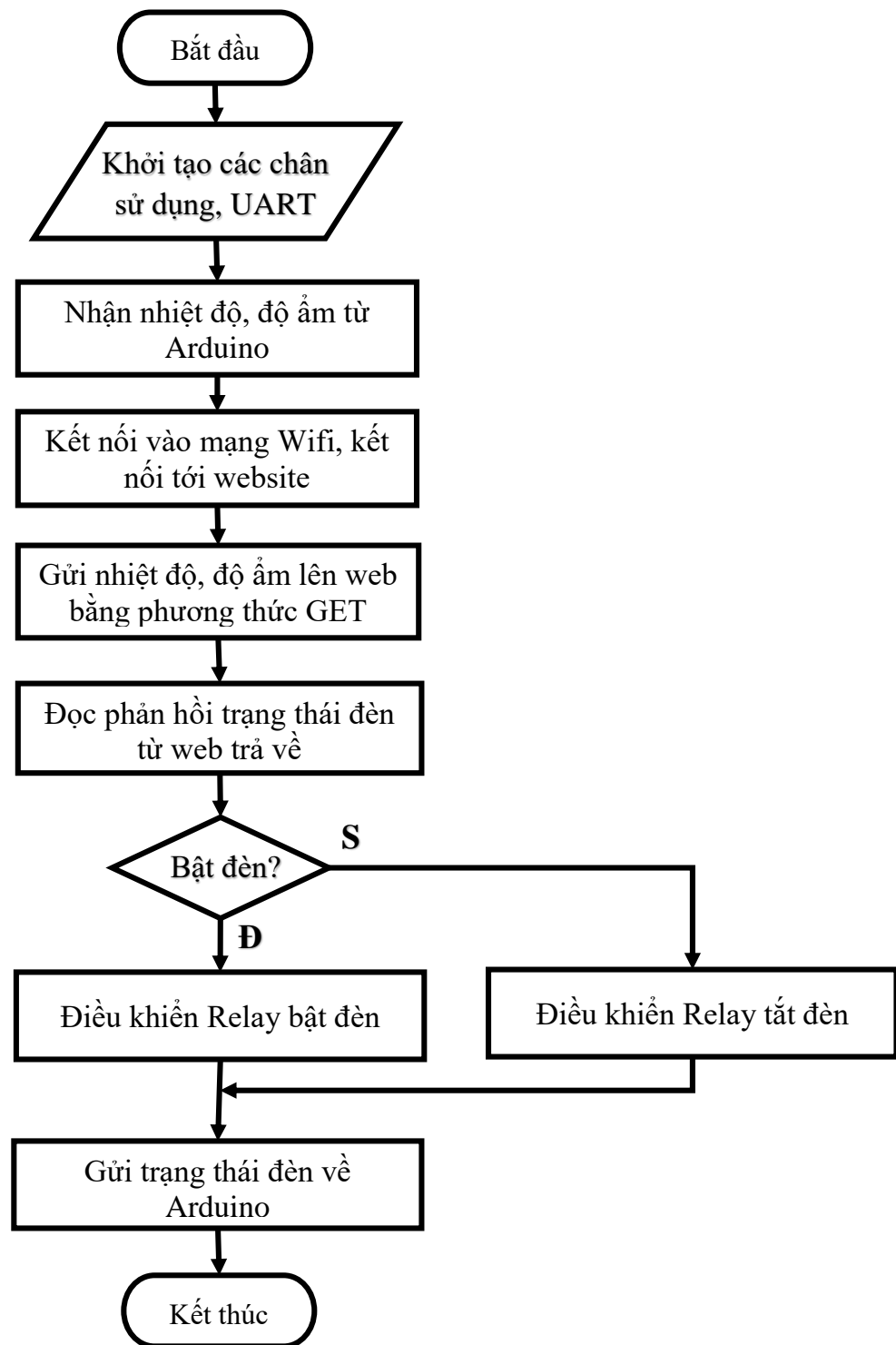
Lưu đồ chương trình con gửi tin nhắn phản hồi (hình 4.9).



**Hình 4.9** Lưu đồ gửi tin nhắn phản hồi

**Giải thích:** chương trình kiểm tra hệ thống đang ở trong sự cố nào và gửi tin nhắn tương ứng với từng trường hợp. Trong trường hợp nhiệt độ quá cao vượt mức ngưỡng 36°C thì gửi tin nhắn báo nhiệt độ cao. Tương tự cho trường hợp có khí gas rò rỉ và có trộm đột nhập sẽ gửi tin nhắn cảnh báo.

Lưu đồ chương trình cho Node MCU được trình bày trong hình 4.10:



**Hình 4.10** Lưu đồ chương trình cho Node MCU

**Giải thích:** Node MCU sau khi kết nối vào mạng wifi thành công sẽ sáng led báo hiệu. Nhận giá trị nhiệt độ, độ ẩm từ Arduino qua cổng giao tiếp UART. Sau đó, kết nối tới web dùng giao thức GET gửi các giá trị này lên trang web



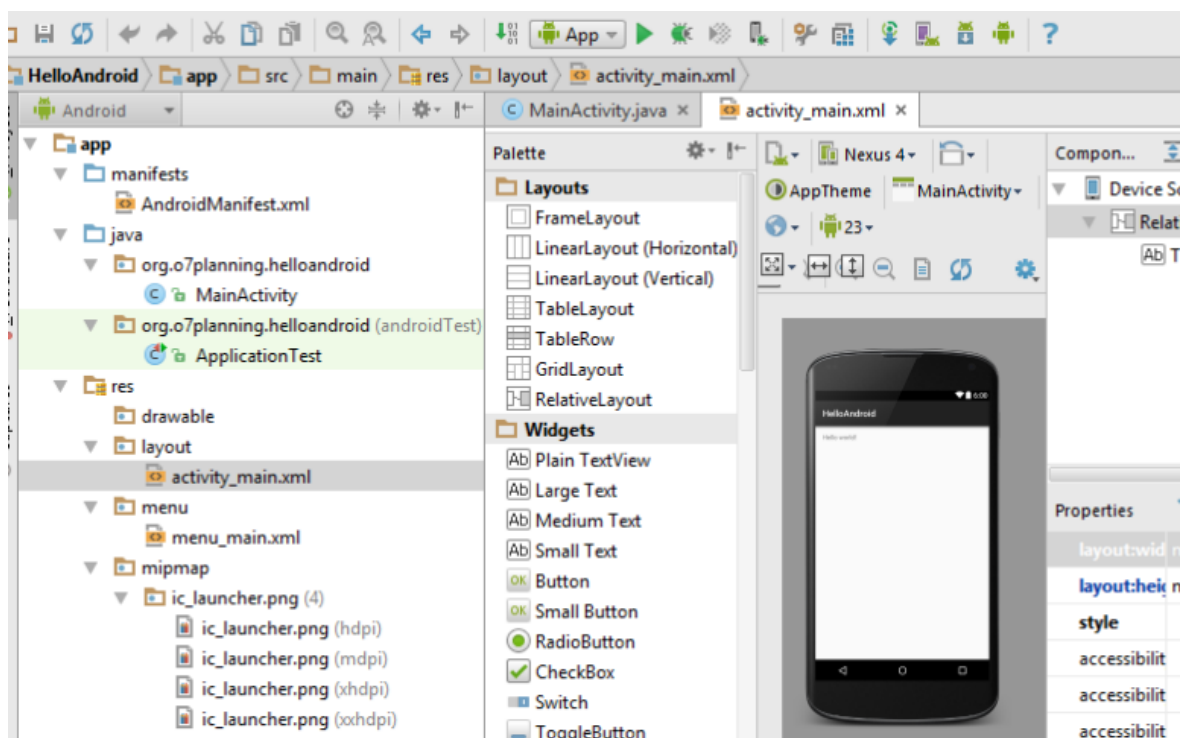
## CHƯƠNG 4. THI CÔNG HỆ THỐNG

load.php, dùng để hiển thị lên trang web. Đồng thời sẽ nhận được các phản hồi từ web là các trạng thái đèn, sử dụng các trạng thái này để điều khiển mạch Relay. Có gửi các trạng thái đèn này về Arduino, để hiển thị lên LCD.

### 4.4.2 Phần mềm lập trình cho Android

#### a. Giới thiệu Android Studio

Android Studio là môi trường phát triển tích hợp (IDE) chính thức dành cho phát triển nền tảng Android, ra mắt vào năm 2013 tại hội nghị Google I/O. Android Studio được phát hành miễn phí. Dựa trên phần mềm IntelliJ IDEA của JetBrains, Android Studio được thiết kế đặc biệt để phát triển ứng dụng Android. Nó hỗ trợ các hệ điều hành Windows, Mac OS X và Linux, và là IDE chính thức của Google để phát triển ứng dụng Android gốc để thay thế cho Android Development Tools (ADT) dựa trên Eclipse.



**Hình 4.11** Giao diện phần mềm Android Studio

Một project Android bao gồm các file sau:

## CHƯƠNG 4. THI CÔNG HỆ THỐNG

Bảng 4.2 Cấu tạo một project Android

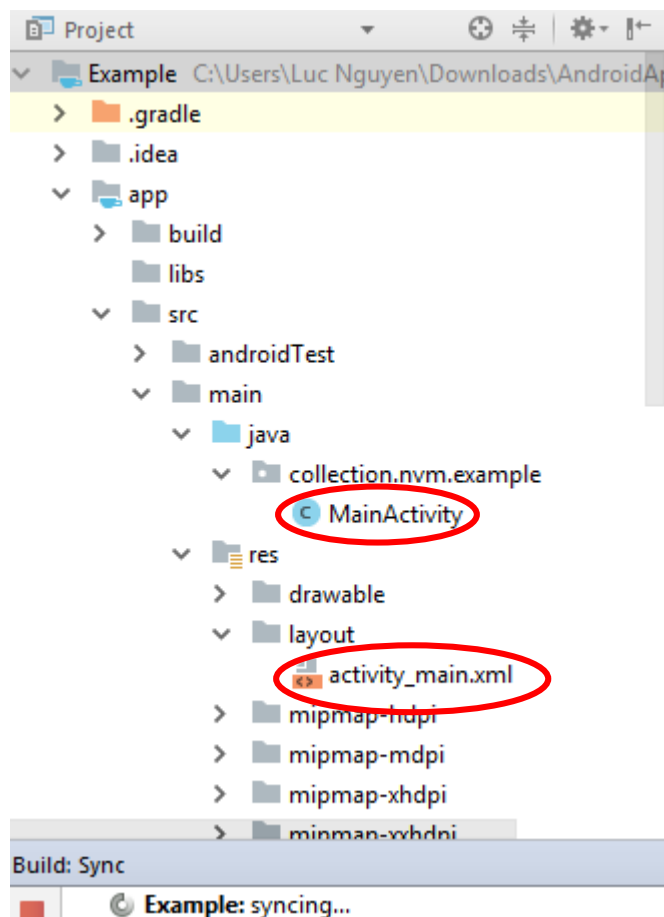
File hoặc thư mục	Mô tả
AndroidManifest.xml	Đây là file manifest mô tả các đặc điểm cơ bản của ứng dụng và xác định từng thành phần của nó.
java	Thư mục này có chứa các file nguồn java cho dự án của bạn. Theo mặc định, nó bao gồm một tập tin nguồn MainActivity.java một lớp hoạt động (activity) chạy khi ứng dụng của bạn được khởi động.
res/drawable	Các phiên bản Android trước đây sử dụng thư mục này để chứa ảnh, các phiên bản hiện tại sử dụng thư mục mipmap thay thế làm nơi chứa ảnh. Thư mục này gần như không còn sử dụng.
res/layout	Thư mục này chứa các file định nghĩa giao diện người dùng.
res/menu	Thư mục này chứa các file xml, định nghĩa các menu sẽ hiển thị trên Action Bar.
res/mipmap	Chứa các ảnh 'mipmap'.
res/values	Đây là một thư mục cho các tập tin XML khác nhau có chứa một tập hợp các nguồn, chẳng hạn như các chuỗi (String) và các định nghĩa màu sắc.

### b. Viết ứng dụng Android

Ứng dụng Android được xây dựng với giao diện gồm các nút nhấn bật tắt đèn, các label để hiển thị trạng thái đèn và nhiệt độ, độ ẩm. Hình 4.12 là giao diện điều khiển của ứng dụng Android trong DATN này.



**Hình 4.12** Giao diện ứng dụng Android trong ĐATN



**Hình 4.13** Vị trí 2 file MainActivity.java và activity\_main.xml

Tiến hành lập trình tạo giao diện cho ứng dụng Android trên file layout/activity\_main.xml. Và các hoạt động của ứng dụng sẽ được lập trình trên file MainActivity.java (như hình 4.12).

### 4.4.3 Phần mềm lập trình cho website

#### a. Giới thiệu phần mềm lập trình

Để lập trình thiết kế giao diện web nhóm sử dụng phần mềm Adobe Dreamweaver CS6 .

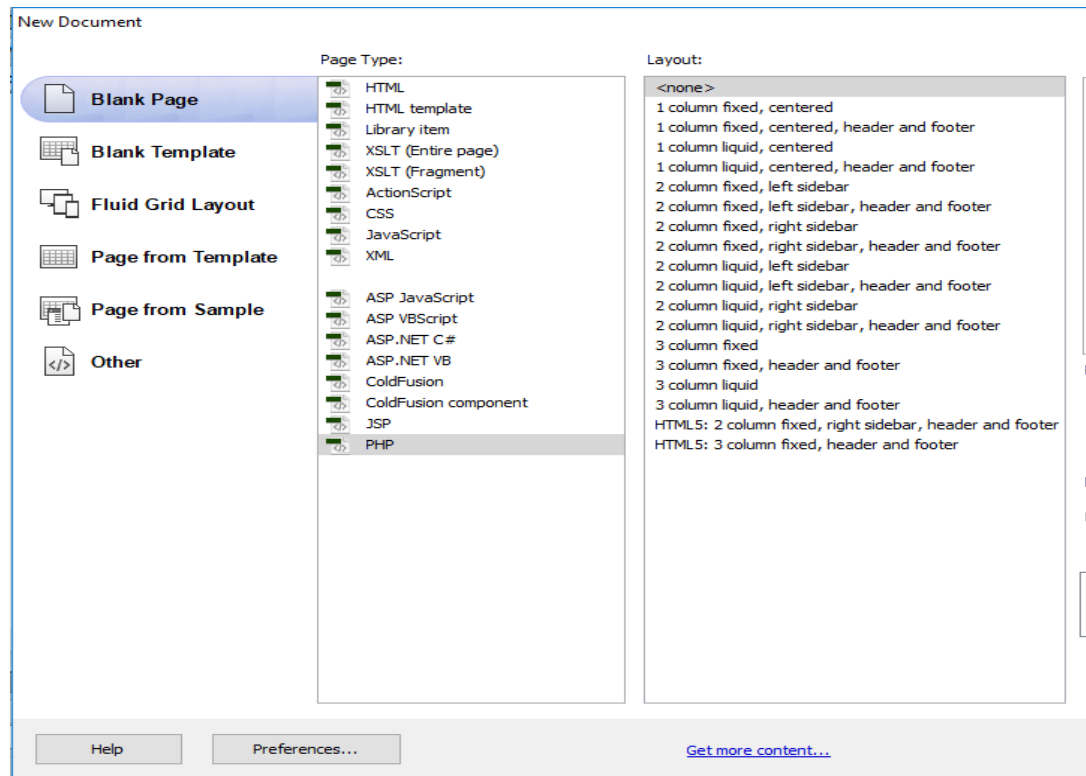
Adobe Dreamweaver CS6 là công cụ xử lý cho những người thiết kế web, viết code và những nhà phát triển ứng dụng ở mọi cấp độ khác nhau..

Với tính năng Fluid Grid Layout của chương trình này cho phép chúng ta thiết kế website với giao diện đẹp với mọi thiết bị, từ pc cho đến mobile tablet. Adobe Dreamweaver có khả năng tạo website tiêu chuẩn hiện đại với HTML5 và CSS3.

Điểm nổi bật của Dreamweaver CS6 chính là tính năng xem trước trong khi bạn đang code. Điều này tạo điều kiện thuận lợi cho công việc của các lập trình viên, tránh được những sai sót đáng tiếc trong quá trình làm việc, giúp tăng hiệu quả làm việc.

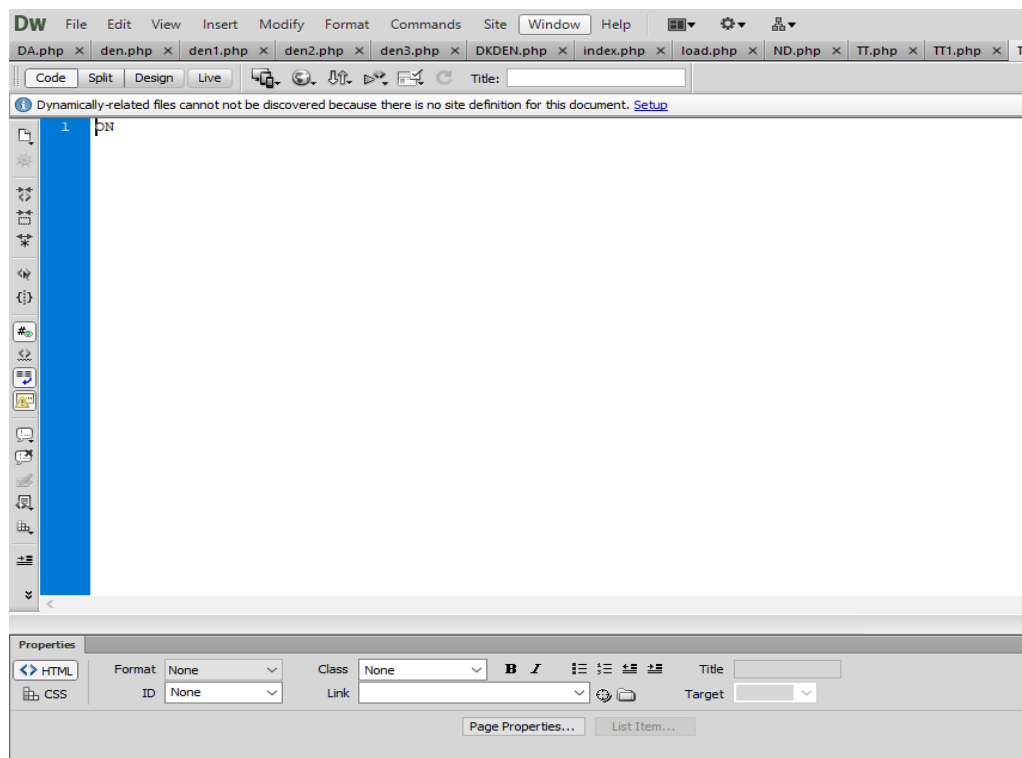
Cách tạo một file mới(hình 4.14): nhấn Create New, Dream liệt kê các file bạn thường sử dụng như File HTML, File CSS, File PHP... bấm nút Create để tạo. Xuất hiện nội dung File vừa chọn.

## CHƯƠNG 4. THI CÔNG HỆ THỐNG



Hình 4.14 Tạo một file php mới

Hình 4.15 sau đây là màn hình làm việc của Dream sau khi tạo file.



Hình 4.15 Giao diện ứng dụng DW

- Thanh Menu là thanh nằm trên cùng chứa các Menu.

- Thanh Toolbars : Nằm dưới thanh Menu. Chứa danh sách các Icons tắt. Giúp cho bạn thực hiện nhanh các chức năng như Preview,
- Hệ thống các panels bên phải. Liệt kê các panel phổ biến như CSS, Quản Lý Files (Panels Files), AP Elements. Panels Insert là Panel mà bạn thường dùng nhất giúp bạn Insert.
- Màn hình làm việc : Màn hình trắng lớn bên trái là Màn hình làm việc chính của bạn hay gọi là Document Windows.
- Tag Selectors: nằm dưới Màn hình làm việc. Các Tags này giúp các bạn chọn nhanh các Thẻ.
- Properties Inspector: định dạng cách hiển thị của bất kỳ thành phần nào mà bạn đã lựa chọn trong Documents Windows.

### **b. Viết chương trình hệ thống**

Để thiết kế được giao diện điều khiển thiết bị cho đề tài này , nhóm đã tạo ra 13 file php (ND, DA, den, den1, den2, den3, TT, TT1, TT2, TT3, Index, load, DKDEN).

Các file php sau khi tạo, được để trống để lưu giá trị điều khiển và hiển thị lên web ( ND, DA, den, den1,den2, den3, TT, TT1, TT2, TT3).

Chi tiết các trang như sau:

- Trang Index.php: là trang hiển thị giao diện chính của trang web, trang này chứa code tạo giao diện web như tạo nút nhấn, tạo các label gắn hiển thị nhiệt độ, độ ẩm, các trạng thái đèn. Các nút nhấn ON0, OFF0, ON1, OFF1 ON2, OFF2, ON3, OFF3 khi được nhấn trên trang giao diện này sẽ thực hiện mở các trang tương ứng den.php, den1.php, den2.php, den3.php TT.php,TT1.php, TT2.php, TT3.php và lưu vào giá trị đèn là bật hay tắt vào.
- Trang den.php, den1.php, den2.php, den3.php: lưu trạng thái đèn từ nút nhấn trên trang index.php.
- Trang TT.php, TT1.php,TT2.php, TT3.php: lưu trạng thái đèn từ nút nhấn, và được gán cho các label trạng thái đèn ngoài trang index.php để hiển thị trạng thái đèn.

## CHƯƠNG 4. THI CÔNG HỆ THỐNG






- Trang load.php: trang có nhiệm vụ lấy các giá trị cảm biến đưa lên từ Node MCU qua phương thức GET và ghi vào trang ND.php, DA.php. Ngoài ra trang này còn đọc các giá trị điều khiển đèn trong các trang den.php, den1.php, den2.php, den3.php, sau đó gửi phản hồi về Node MCU, nhờ đó điều khiển được các thiết bị qua website.
- Trang ND.php, DA.php: lưu các giá trị nhiệt độ, độ ẩm, được gán cho các label hiển thị nhiệt độ, độ ẩm ở trang index.php.

### 4.4.4 Phần mềm lập trình cho Arduino và Node MCU

Trong đề tài này nhóm dùng phần mềm Arduino IDE để lập trình cho cả 2 board Arduino và Node MCU. Arduino IDE là một trình soạn thảo văn bản, giúp bạn viết code để nạp vào bo mạch Arduino.

Một chương trình viết bởi Arduino IDE được gọi là sketch, sketch được lưu dưới định dạng .ino.

Bảng 4.3 Một số nút chức năng trên Arduino IDE

	Verify Kiểm tra lỗi và biên dịch code
	Upload Dịch và upload code vào bo mạch đã được cài đặt sẵn
	New Tạo sketch mới
	Open Mở một sketch có sẵn
	Save Lưu sketch
	Serial Mở <u>serial monitor</u> .
	Monitor

### Nạp Sketch

Trước khi nạp Sketch bạn phải chọn tên bo mạch, và cổng com:

Từ menu chính chọn Tools > Board và Tools > Port

### Libraries

Là những thư viện bổ sung thêm những chức năng để sử dụng trong sketch, ví dụ làm việc với màn hình LCD bạn phải cần có một thư viện để điều khiển nó. Để dùng library, bạn chọn Sketch->import Library.

### Serial Monitor

Là hộp thoại để hiển thị dữ liệu được gửi từ bo mạch Arduino, hoặc gửi dữ liệu từ máy tính tới bo mạch của bạn.

## 4.5 TÀI LIỆU HƯỚNG DẪN SỬ DỤNG

**Bước 1 :** cấp nguồn cho hệ thống, hệ thống sử dụng nguồn từ adapter 6VDC/2A, khi cấp nguồn thì đèn báo hiệu có điện sáng lên.

Lưu ý: trên board mạch có 1 switch(hình 4.16) , có chức năng lựa chọn có sử dụng nguồn dự phòng trong trường hợp mất điện hay không. Gạt switch sang trái sẽ vào chế độ không dùng nguồn dự phòng khi mất điện. Ngược lại, gạt switch sang phải sẽ vào chế độ dùng nguồn dự phòng khi mất điện. Trước khi cấp nguồn cho mạch thì cấp acqui 6V tại domino acqui 2 chân.

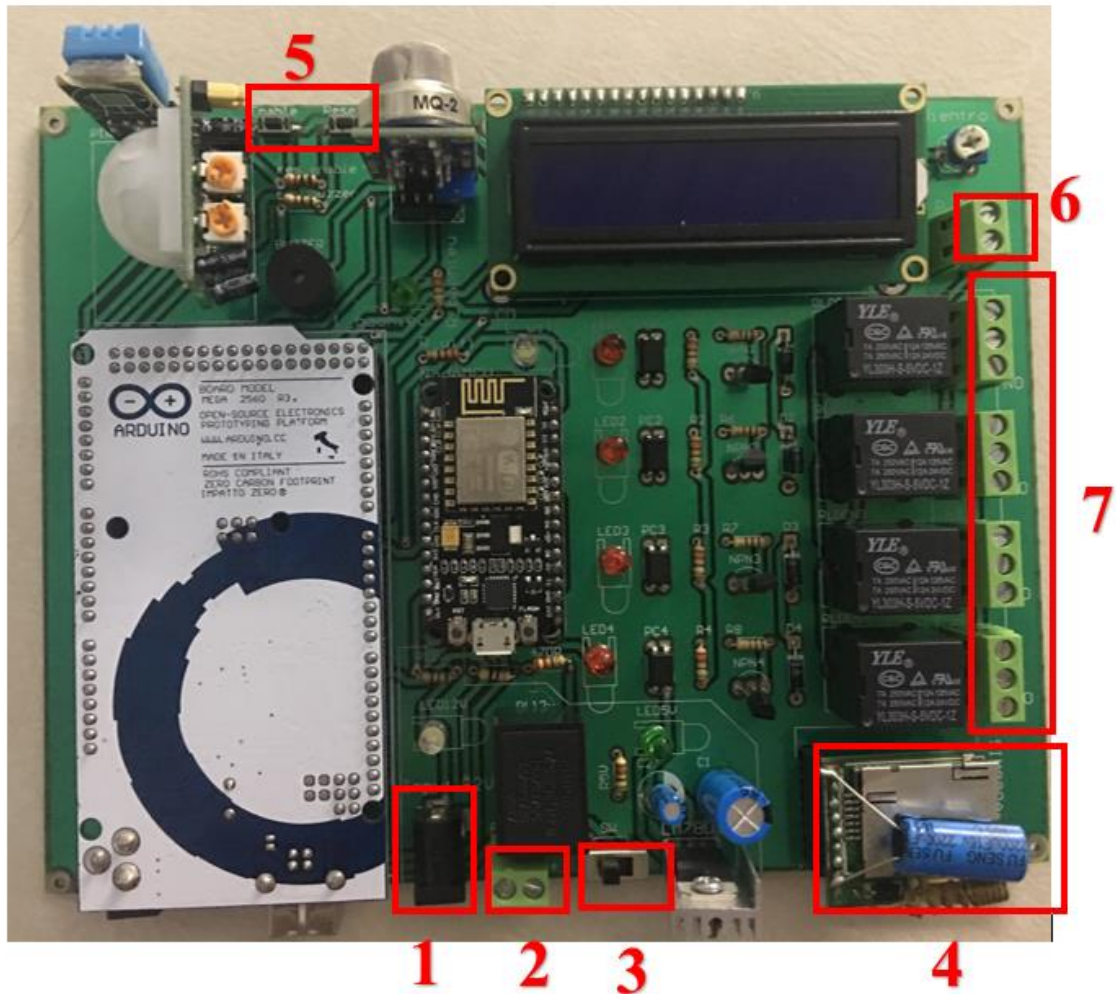
**Bước 2:** Khởi động chế độ báo trộm bằng cách nhấn nút enable trên mạch hoặc gửi tin nhắn “kichhoat” tới số điện thoại được cài trên board.(chế độ báo cháy, báo khí gas rò rỉ mặc định luôn hoạt động). Nhấn “huy”, để hủy tính năng này.

**Bước 3:** Truy cập trang web <https://huynhthan1996.000webhostapp.com/> hoặc mở app Android được cài trên điện thoại điều khiển các thiết bị điện. Nhấn vào các nút trên giao diện, sẽ điều khiển tương ứng thiết bị. Ví dụ nhấn bật 1, thì đèn 1 sẽ sáng.

**Bước 4:** Để xem giá trị nhiệt độ-độ ẩm và các trạng thái hiện tại của thiết bị, ta xem trên LCD hoặc truy cập website , app Android ở trên.

**Bước 5:** Khi có sự cố thì chuông sẽ kêu lên và sẽ có tin nhắn gửi cảnh báo. Để tắt cảnh báo thì nhấn nút reset trên board hoặc nhấn tin “reset”.





**Hình 4.16** Mô tả board mạch hệ thống

1. Vị trí cắm jack nguồn Adapter 6VDC.
2. Vị trí cắm Acqui dự phòng 6VDC.
3. Vị trí Switch lựa chọn sử dụng nguồn acqui.
4. Vị trí bỏ thẻ SIM.
5. Vị trí 2 nút nhấn reset chuông và enable báo trộm.
6. Ngõ vào 220VAC.
7. Ngõ ra nối vào các đèn 220VAC.

## **Chương 5. KẾT QUẢ\_NHẬN XÉT\_ĐÁNH GIÁ**

### **5.1 Kết quả**

Trong quá trình thực hiện DATN, nhóm đã đạt được những thành công nhất định thông qua việc tìm hiểu, nghiên cứu các đối tượng liên quan đến đồ án; thiết kế được một board mạch điều khiển gọn đẹp và xây dựng được một mô hình thực tế. Cụ thể như sau:

- Nghiên cứu và hiểu được nguyên lý hoạt động của các cảm biến nhiệt độ-độ ẩm DHT11, cảm biến khí gas MQ5, cảm biến chuyển động PIR.

- Nghiên cứu và biết sử dụng Board Arduino, ứng dụng hiệu quả board mạch này trong DATN.

- Nghiên cứu, tìm hiểu về mạng mạng Wifi.

- Tìm hiểu và ứng dụng được các mô đun wifi ESP8266, SIM900A.

- Tìm hiểu về lập trình web, xây dựng website và lập trình Android.

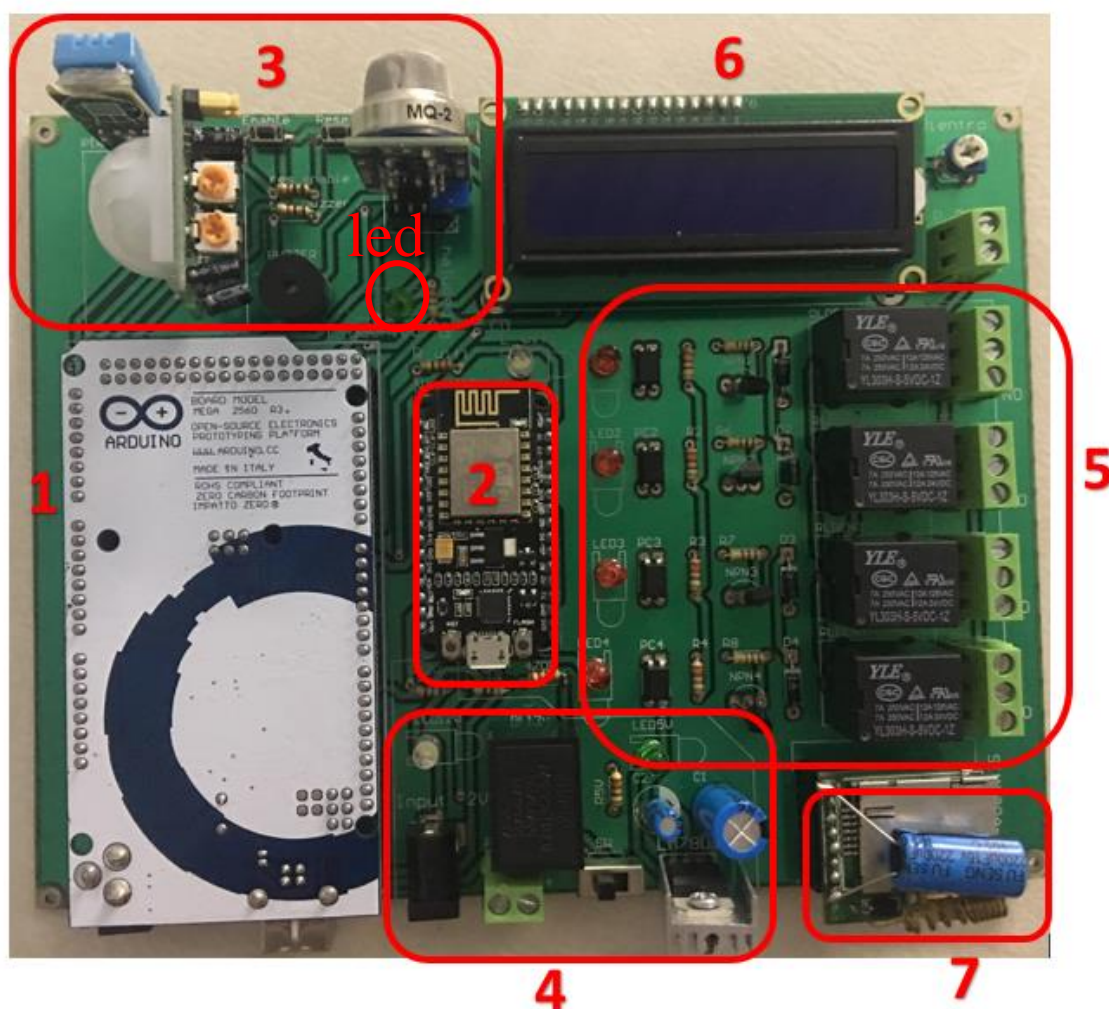
- Thiết kế được mạch ứng dụng từ Relay điều khiển các thiết bị công suất nhỏ.

- củng cố lại chuẩn giao tiếp UART, chuẩn giao tiếp 1 dây đã được học. Tiếp cận được phương thức GET - gửi, nhận dữ liệu giữa Client và máy chủ.

- Tiếp cận sử dụng được nhiều phần mềm mới như: Altium, Android Studio, Dreamweaver.

- Tính toán, thiết kế, chọn linh kiện để hoàn thiện được board mạch chính đủ dòng áp chạy đáp ứng được yêu cầu đề tài.

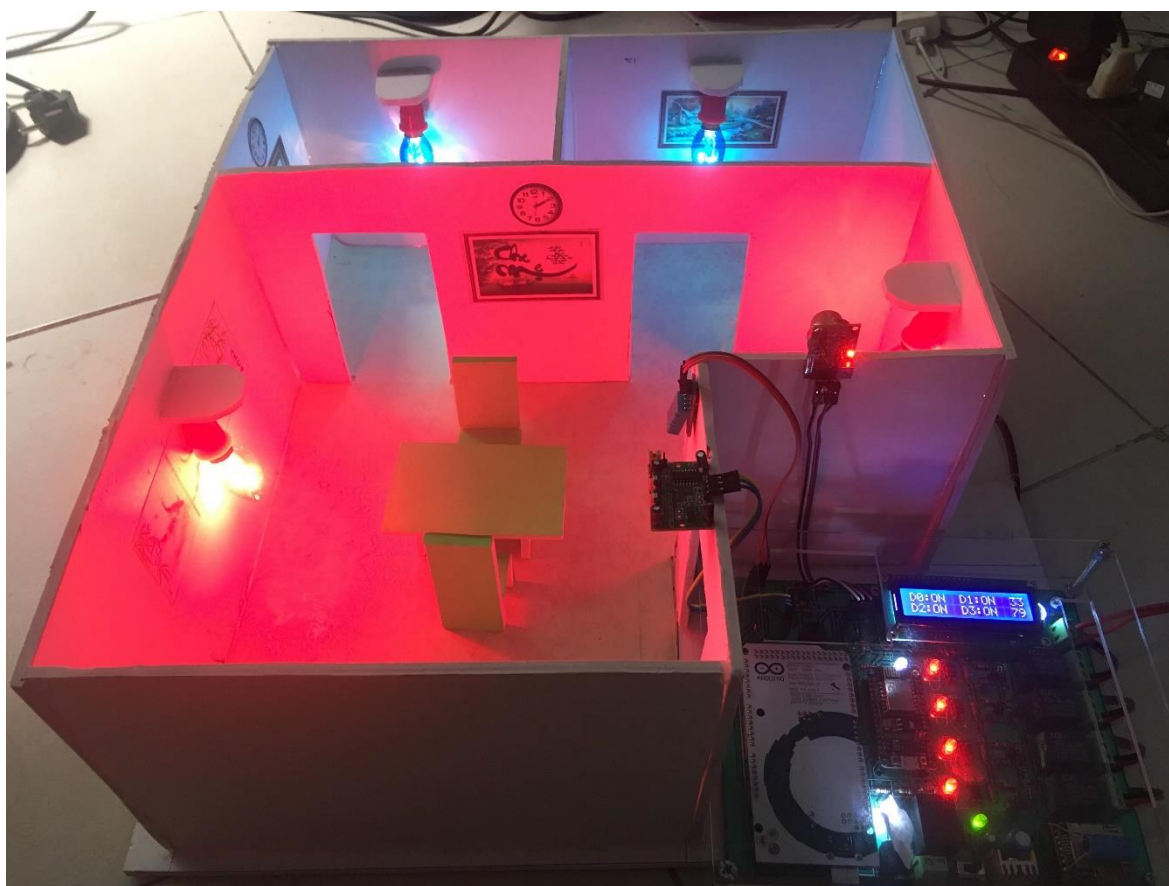
Hình 5.1 trình bày chi tiết các khối trên board mạch chính.



**Hình 5.1** Board mạch chính điều khiển

**Chú thích:**

1. Board điều khiển Arduino.
2. Mô đun Wifi Node MCU giúp kết nối board mạch vào mạng Wifi.
3. Khối cảm biến: gồm cảm biến DHT11, khí gas MQ2, cảm biến chuyển động, nút nhấn reset chuông, nút nhấn enable báo trộm. Có một led báo hiệu có đang kích hoạt báo trộm hay không(sáng là đang kích hoạt, tắt là không kích hoạt).
4. Khối nguồn: gồm mạch nguồn IC7805, relay chuyển sang nguồn acqui.
5. Khối relay: gồm 4 mạch kích relay, điều khiển các đèn.
6. LCD hiển thị.
7. Mô đun SIM900A.



**Hình 5.2** Mô hình ĐATN

Các đèn được điều khiển bật hết như hình 5.2 khi nhấn các nút nhấn bật đèn trên giao diện website hoặc trên ứng dụng Android. Các đèn này là các đèn 220VAC được điều khiển qua mạch Relay. Có thể xem được các trạng thái đèn hiện tại trên giao diện website, Android và màn hình LCD trên board mạch. Ngoài ra, còn có các giá trị nhiệt độ, độ ẩm cũng được cập nhật và hiển thị. Màn hình hiển thị như hình 5.3.





**Hình 5.3** Trạng thái các đèn, giá trị nhiệt độ và độ ẩm hiển thị trên LCD

### ĐỒ ÁN TỐT NGHIỆP 2018

Nhiệt Độ(°C)		Độ Ẩm(%)	
32		91	

ON		OFF	
	Bật Đèn 0		Tắt Đèn 0
	Bật Đèn 1		Tắt Đèn 1
	Bật Đèn 2		Tắt Đèn 2
	Bật Đèn 3		Tắt Đèn 3
	Bật Tất Cả		Tắt Tất Cả

ĐÈN 0	ĐÈN 1	ĐÈN 2	ĐÈN 3
ON	ON	ON	ON

**Hình 5.4** Giao diện website điều khiển

Giao diện website điều khiển (hình 5.4) gồm các nút điều khiển bật đèn, tắt đèn, các giá trị nhiệt độ, độ ẩm được cập nhật và các trạng thái đèn. Website này có thể được truy cập từ bất cứ nơi đâu có Internet, Wifi, 3G.



**Hình 5.5** Giao diện ứng dụng Android

Ứng dụng Android (hình 5.5) được thiết kế với giao diện đơn giản gồm các nút bật đèn, tắt đèn, có tích hợp hiển thị nhiệt độ - độ ẩm căn nhà và các trạng thái đèn.



**Hình 5.6** Cảm biến chuyển động được lắp ở cửa ra vào nhà

Cảm biến chuyển động được lắp tại cửa ra vào để phát hiện báo trộm. Bên cạnh cảm biến chuyển động còn có cảm biến nhiệt độ độ ẩm cũng được lắp tại phòng khách để đo nhiệt độ, độ ẩm. Để kích hoạt chế độ báo trộm có thể nhấn nút nhấn enable trước khi ra khỏi nhà hoặc nhấn tin có cú pháp “kichhoat”. Ngược lại để tắt, nhấn nút nhấn này một lần nữa hoặc nhấn tin “huy”. Có led báo hiệu trên board mạch để nhận biết có đang kích hoạt báo trộm hay không. Khi chế độ báo trộm được kích hoạt và có phát hiện chuyển động của vật thể, sẽ có báo động buzzer và gửi tin nhắn cảnh báo. Chuông buzzer chỉ tắt khi nhấn nút reset chuông trên board mạch hoặc nhấn tin nhắn với cú pháp “tat”. Tin nhắn cảnh báo nhận được sẽ giống như hình 5.7



**Hình 5.7** Mô đun Sim gửi tin nhắn cảnh báo trộm

ĐATN này còn có tính năng cảnh báo rò rỉ khí gas. Vị trí cảm biến khí gas được lắp như trong hình 5.8.



**Hình 5.8** Cảm biến khí gas lắp tại phòng bếp

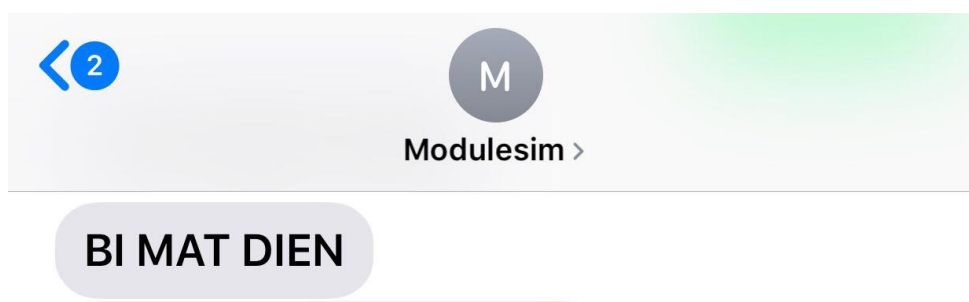
Cảm biến khí ra được lắp tại phòng bếp của mô hình. Mặc định, chế độ báo rò rỉ khí gas luôn hoạt động. khi có khí gas rò rỉ, lập tức buzzer reo chuông cảnh báo, đồng thời gửi tin nhắn cảnh báo về. Để tắt chuông buzzer nhấn nút reset chuông trên board mạch hoặc gửi tin nhắn với cú pháp “tat”. Tin nhắn nhận được khi phát hiện rò rỉ khí gas sẽ giống như hình 5.9.





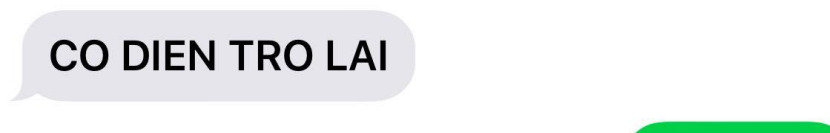
**Hình 5.9** Tin nhắn cảnh báo có khí gas rò rỉ

Hệ thống có Acqui dự phòng. Để xác định trạng thái board mạch đang sử dụng nguồn từ Adapter hay Acqui, Arduino đọc giá trị digital từ chân 22. Chân này là Vout của mạch phân áp từ adapter 6VDC, sau đó so sánh với giá trị trạng thái hệ thống trước đó từ bộ nhớ EPROM của Arduino. Giá trị chân này bằng 1 khi dùng nguồn từ adapter và bằng 0 khi dùng nguồn từ acqui. Đánh giá nếu có mất điện hay có điện sẽ gửi tin nhắn thông báo. Sau đó, lưu trạng thái mới của hệ thống vào bộ nhớ EPROM. Khi mất điện sẽ có tin nhắn báo về điện thoại như hình 5.10 sau:



**Hình 5.10** Tin nhắn nhận được khi bị cúp điện 220VAC

Ngược lại khi có điện 220V trở lại thì người dùng sẽ nhận được tin nhắn thông báo như hình 5.11



**Hình 5.11** Tin nhắn nhận được khi có điện trở lại

### 5.2 Nhận xét, đánh giá

Kết quả thực nghiệm của toàn bộ mô hình về cơ bản đã hoàn thành được yêu cầu DATN đề ra, đó là thực hiện mô hình điều khiển được các đèn qua mạng Wifi và Android, tích hợp theo dõi nhiệt độ, độ ẩm căn nhà; có báo động

và gửi tin nhắn trong trường hợp có sự cố: rò rỉ gas, đột nhập, nhiệt độ cao. Có còi buzzer báo động kêu to, rõ. Lập trình cho hệ thống hoạt động ổn định qua nhiều lần kiểm tra, xử lý triệt để các tình huống mạch hoạt động sai so với yêu cầu. Hệ thống điều khiển từ xa trên giao diện ứng dụng điện thoại và trang web dễ dàng sử dụng. Cảm biến hoạt động chính xác, đáp ứng trong các trường hợp sự cố nhanh, hiệu quả. Tuy nhiên vẫn tồn tại nhiều điểm yếu chưa khắc phục được. Chẳng hạn ĐATN không phát triển tính bảo mật trong hệ thống

## **Chương 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

### **6.1 KẾT LUẬN**

Sau thời gian tìm hiểu, nghiên cứu và thực hiện mô hình, nhiệm vụ đề tài cơ bản được hoàn thành. Bằng sự cố gắng của bản thân, học hỏi từ Thầy Cô, bạn bè, nhóm đã hoàn thành đề tài, đạt được yêu cầu đặt ra là thiết kế và thi công mô hình điều khiển thiết bị điện qua website và ứng dụng, có hệ thống cảnh báo qua tin nhắn, hệ thống cảm biến.

Đề tài đã hoàn thành cụ thể:

- Kết quả thi công hoàn thiện.
- Hệ thống chạy tương đối ổn định.
- Độ phản hồi của hệ thống cảnh báo sự cố tương đối nhanh.
- Giao diện web và ứng dụng Android thân thiện, dễ sử dụng. Điều khiển thiết bị tương đối nhanh.
- Board mạch điều khiển thiết kế gọn và hợp lý, dễ sửa chữa.
- Có thiết kế được một mạch chuyển sang dùng Acqui dự trữ trong trường hợp mất điện.
- Có khả năng báo mất điện qua tin nhắn.

### **6.2 HƯỚNG PHÁT TRIỂN**

Để đảm bảo sự ổn định, độ bền và tính thực tế cho sản phẩm, nhóm có kiến nghị hướng phát triển hệ thống như sau:

- Thêm hệ thống camera giám sát từ xa, tích hợp nhận diện khuôn mặt phát hiện người lạ khi đột nhập
- Tăng tính bảo mật, xây dựng trang đăng nhập vào hệ thống.
- Tích hợp trợ lý ảo giúp thực hiện yêu cầu, xây dựng smarthome.

## **TÀI LIỆU THAM KHẢO**

### **Sách tham khảo**

- [1] Nguyễn Đình Phú, “Giáo trình vi xử lý II”, NXB ĐH Quốc Gia Tp.HCM, 2007.
- [2] Nguyễn Văn Lem, “Thiết kế và xây dựng hệ thống mô hình nhà thông minh”, Đồ án tốt nghiệp, trường ĐHSPKT Tp.HCM, 2015.
- [3] Trần Thu Hà, “Điện tử cơ bản”, NXB ĐH Quốc Gia Tp.HCM, 2013.
- [4] Nguyễn Văn Hiệp, Đinh Quang Hiệp, “Lập trình Android cơ bản”, NXB ĐH Quốc Gia Tp.HCM, 2015.

### **Trang web tham khảo**

- [1] Nguyễn An, “ACIS Smart Home người lãnh ấn tiên phong trong giải pháp nhà thông minh Việt Nam”, 2017.

<https://acis.com.vn/acis-smart-home-nguoi-linh-tien-phong-trong-giai-phap-nha-thong-minh-viet-nam/>.

- [2] Nguyễn Sỹ Thanh Sơn, “Đo nhiệt độ, độ ẩm và gửi dữ liệu lên cloud”, 2016.

“<https://www.linkedin.com/pulse/ngày-thứ-2-học-iot-đo-nhiệt-độ-ẩm-và-gửi-dữ-liệu-lên-cloud-thanh-son>”.

- [3] Modi, “Internet controlled led using Node MCU”, 2017.

<http://www.instructables.com/id/Internet-Controlled-LED-Using-NodeMCU/>.

- [4] Nguyễn Linh, “Giới thiệu về hệ điều hành Android”, 2015.

<https://androidcoban.com/gioi-thieu-ve-dieu-hanh-android.html>.

**DANH SÁCH TỪ VIẾT TẮT**

Tên từ viết tắt	Tên đầy đủ
ĐATN	Đồ Án Tốt Nghiệp
IoT	Internet of Things
ĐH	Đại học
UART	Universal Asynchronous Receiver – Transmitter
HTTP	Hypertext Transfer Protocol
SMS	Short Message Services
PHP	Hypertext Preprocessor

## PHỤ LỤC

### Code chương trình cho board Arduino:

```
#include <SoftwareSerial.h>
#include <LiquidCrystal.h>
#include <EEPROM.h>
#include "DHT.h"
LiquidCrystal lcd(37, 36, 32, 33, 34, 35);
#define MATNGUON 23 // chân báo mất nguồn
#define btn2 4 // nút tắt chuông báo
#define relay 13 // buzzer
int addr = 0;
#define led_enable 12 // led báo kích hoạt chống trộm
const int DHTPIN = 2; // Đọc dữ liệu từ DHT11 ở chân 2 trên mạch
Arduino
const int DHTTYPE = DHT11; // Khai báo loại cảm biến, có 2 loại là DHT11
và DHT22
DHT dht(DHTPIN, DHTTYPE);
int CHAY = 0; // có cháy =1 ; không cháy =0
int TROM = 0; // có trộm =1 ; không trộm =0
int GAS = 0; // có gas =1 ; không gas =0
int i = 0; // biến dùng để đếm số lần gửi tin nhắn
int j = 0; // biến dùng cho nút nhấn
int TT = 0; // biến GAS
int TT1 = 0; // biến PIR
int TT2 = 0; // báo Cháy DHT11
int KH = 0; // biến kích hoạt chế độ báo trộm
int MN = 0; // biến báo mất nguồn
int t=0;
int h=0;
int Sgas = A0;
int Spir = A1;
int address = 0;
int BTN_ENA;
int BTN_chuong;
String textMessage, den;
SoftwareSerial SIM900(10, 9); // RX, TX

void btn_reset()
{
    TT=0;
    TT1=0;
    TT2=0;
    i=0;
    KH=0;
    CHAY=0;
    GAS=0;
    TROM=0;
    digitalWrite(led_enable, LOW);
    //Serial.println("đã tắt báo động");
    digitalWrite(relay, LOW);
}

void btn_enable()
{
    j++;
    if (j>1)
    { j=0; }

    if (j==0)
    { KH=0; digitalWrite(led_enable, LOW); }
    if (j==1)
```

```
{KH=1;digitalWrite(led_enable, HIGH);}

}

void setup() {
  Serial.begin(9600); //Mở cổng Serial để giao tiếp
  lcd.begin(16, 2); //Thông báo đây là LCD 1602
  dht.begin(); // Khởi động cảm biến
  pinMode(relay, OUTPUT);
  pinMode(led_enable, OUTPUT);
  pinMode(btn2, INPUT); //Cài đặt chân D4 ở trạng thái đọc dữ liệu
  pinMode(MATNGUON, INPUT); // Cài đặt chân 22 ở trạng thái đọc dữ liệu
  digitalWrite(relay, LOW); //chuông báo
  digitalWrite(led_enable, LOW); //led báo enable che do bao trom
  attachInterrupt(1, btn_enable, RISING);
  attachInterrupt(3, btn_reset, HIGH);
  SIM900.begin(9600);
  xulymatdien();
}

void loop() {
  float t = dht.readTemperature(); //Đọc nhiệt độ
  float h = dht.readHumidity(); //Đọc độ ẩm
  int value = analogRead(Sgas); //đọc giá trị điện áp ở chân A0 - chân
  cảm biến GAS
  int value1 = analogRead(Spir); // PIR
  hienthi(t, h);
  Serial.print('D');
  Serial.print(round(t));
  Serial.print('!');
  Serial.print('E');
  Serial.print(round(h));
  Serial.print('%');

  if (t>35){
    CHAY = 1;}
  else
    CHAY = 0;
  if (value>50)
    {GAS = 1; }
  else
    GAS = 0;
  if (value1>500)
    {TROM = 1;}
  else TROM=0;
  //////////////////////////////////////
  if (GAS==1)
  {TT=1; //trang thai tam
    //Serial.print("co GAS");
    digitalWrite(relay, HIGH);
    i++; //dem de gui tin nhan 1 lan
  }
  else if ((TROM)&(KH)) //chi bao trom khi da kích hoạt che do bao trom
  {TT1=1; //trang thai tam
    digitalWrite(relay, HIGH);
    //Serial.print("co trom");
    i++; //dem de gui tin nhan 1 lan
  }
  else if (CHAY==1)
  {TT2=1; //trang thai tam
    //Serial.print("co CHAY");
    digitalWrite(relay, HIGH);
    i++; //dem de gui tin nhan 1 lan
  }
}
```

```
}

////////////////////////////////////

if ((TT)&(i==1))//gui tin nhan bao RÒ GAS chỉ 1 lần
{
    sendSMS("PHAT HIEN CO KHI GAS");//bo dau comment de gui tin nhan
    i++;}

else if ((TT1)&(i==1))//gui tin  nhan bao trom
{
    sendSMS("PHAT HIEN TROM");
    i++;}
else if ((TT2)&(i==1))//gui tin  nhan bao cháy
{
    sendSMS("PHAT HIEN CHAY");
    i++;}

////////////////////////////////////
if(SIM900.available()>0){//doc tin nhan
    textMessage = SIM900.readString();
    //Serial.print(textMessage);
    delay(10);
}
if(textMessage.indexOf("Kichhoat")>=0){

    KH=1;
    digitalWrite(led_enable, HIGH);

    j=1;
    textMessage = "";
}

////////////////////////////////////
if(textMessage.indexOf("Huy")>=0){

    KH=0;

    j=0;
    digitalWrite(led_enable, LOW);
    textMessage = "";
}
if(textMessage.indexOf("Tat")>=0){

    TT=0;
    TT1=0;
    TT2=0;
    i=0;
    KH=0;
    CHAY=0;
    GAS=0;
    TROM=0;
    digitalWrite(led_enable, LOW);
    //Serial.println("da tat bao dong");
    digitalWrite(relay, LOW);
}

if(den.indexOf("one")>=0)
{
    lcd.setCursor(3, 0);
```



```
    lcd.print("ON ");
}
if(den.indexOf("two")>=0)
{
    lcd.setCursor(3, 0);
    lcd.print("OFF");
}

while (Serial.available())
{
    den = Serial.readStringUntil('\R');

    if(den.indexOf("three")>=0)
    {
        lcd.setCursor(10, 0);
        lcd.print("ON ");
    }
    if(den.indexOf("four")>=0)
    {
        lcd.setCursor(10, 0);
        lcd.print("OFF");
    }
    if(den.indexOf("five")>=0)
    {
        lcd.setCursor(3, 1);
        lcd.print("ON ");
    }
    if(den.indexOf("six")>=0)
    {
        lcd.setCursor(3, 1);
        lcd.print("OFF");
    }

    if(den.indexOf("seven")>=0)
    {
        lcd.setCursor(10, 1);
        lcd.print("ON ");
    }
    if(den.indexOf("eight")>=0)
    {
        lcd.setCursor(10, 1);
        lcd.print("OFF");
    }

}

////////////////////////////////////
void sendSMS(String message) {
    SIM900.println("AT+CMGF=1");
    delay(1000);
    SIM900.println("AT+CMGS=\"+841678532566\"\\r");
    delay(1000);
    SIM900.println(message);
    delay(1000);
    SIM900.println((char)26);
    delay(100);
    SIM900.println();
}
}
```

```
////////////////////////////////////////
void hienthi(int t, int h){
    lcd.setCursor(0, 0); // cột 0 dòng 1
    lcd.print("D0:");
    lcd.setCursor(7, 0); // cột 0 dòng 1
    lcd.print("D1:");
    lcd.setCursor(0, 1); // cột 0 dòng 1
    lcd.print("D2:");
    lcd.setCursor(7, 1); // cột 0 dòng 1
    lcd.print("D3:");
    lcd.setCursor(14, 0); // cột 0 dòng 1
    lcd.print(t);
    lcd.setCursor(14, 1); // cột 0 dòng 2
    lcd.print(h);

}
void xulymatdien()
{int buttonStatus = digitalRead(MATNGUON);
  int L = EEPROM.read(addr);
  delay(500);
  EEPROM.write(addr,buttonStatus);
  delay(500);
  if((L!=buttonStatus)&&(buttonStatus==1))
  sendSMS("CO DIEN TRO LAI");
  if((L!=buttonStatus)&&(buttonStatus==0))
  sendSMS("BI MAT DIEN");
}
```

### Code cho Node MCU:

```
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
#define RELAY0 14
#define RELAY1 12
#define RELAY2 13
#define RELAY3 4
String nd_da = "";
String line = "";
boolean stringComplete= false;
//UART Nhan;
uint8_t t=0;
uint8_t h=0;
const char* ssid = "Charlie Nguyen";
const char* password = "ngocluc96";
WiFiClient client;
void setup(void)
{
    Serial.begin(9600);
    //Nhan.begin(9600);
    pinMode(RELAY0, OUTPUT);
    pinMode(RELAY1, OUTPUT);
    pinMode(RELAY2, OUTPUT);
    pinMode(RELAY3, OUTPUT);
    pinMode(16, OUTPUT);
    digitalWrite(RELAY0, HIGH);
    digitalWrite(RELAY1, HIGH);
    digitalWrite(RELAY2, HIGH);
    digitalWrite(RELAY3, HIGH);
    digitalWrite(16, LOW);
    //Serial.println("Humidity and temperature\n\n");
}
```

```
delay(700);
WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    digitalWrite(16, HIGH); //led bao hieu ket noi wifi thanh cong
}
digitalWrite(16, HIGH); //led bao hieu ket noi wifi thanh cong
}
void loop() {

    //nhan_du_lieu(&t, &h);
    serialEvent();
    if (stringComplete) {
        String giatriA = nd_da.substring(nd_da.indexOf("D") + 1,
nd_da.indexOf("!"));
        t = giatriA.toInt();
        String giatriB = nd_da.substring(nd_da.indexOf("E") + 1,
nd_da.indexOf("%"));
        h = giatriB.toInt();
        nd_da = "";
        stringComplete = false;
    }
    if (WiFi.status() == WL_CONNECTED)
    {
        HTTPClient http; //Declare an object of class HTTPClient
        if (client.connect("huynhthan1996.000webhostapp.com",80)) {
            client.print(String("GET /load.php?") + "field1=" +
String(t) + "&field2=" + String(h) + " HTTP/1.1\r\n" +
"Host: " + "huynhthan1996.000webhostapp.com" + "\r\n" +
"Connection: close\r\n\r\n");
            delay(500);
            while (client.available()) {
                char bufer = client.read();
                line+=bufer;
            }
            Serial.print(line);
            if(line.indexOf("one")!=-1)
            {
                digitalWrite(RELAY0, LOW);
            }
            else if(line.indexOf("two")!=-1)
            {
                digitalWrite(RELAY0, HIGH);
            }
            if(line.indexOf("three")!=-1)
            {
                digitalWrite(RELAY1, LOW);
            }
            else if(line.indexOf("four")!=-1)
            {
                digitalWrite(RELAY1, HIGH);
            }
            if(line.indexOf("five")!=-1)
            {
                digitalWrite(RELAY2, LOW);
            }
            else if(line.indexOf("six")!=-1)
            {
                digitalWrite(RELAY2, HIGH);
            }
            if(line.indexOf("seven")!=-1)
```

```
        {
            digitalWrite(RELAY3, LOW);
        }
        else if(line.indexOf("eight")!=-1)
        {
            digitalWrite(RELAY3, HIGH);
        }
        line="";
    }
}

void serialEvent() {
    while (Serial.available()) {
        char inChar2 = (char)Serial.read();
        nd_da += inChar2;
        if (inChar2 == '%') {
            stringComplete = true;
        }
    }
}
```

**Code để tạo label hiển thị độ ẩm:**

```
<TextView

    Android:id="@+id/tvDA"

    Android:layout_width="match_parent"

    Android:layout_height="wrap_content"

    Android:layout_marginTop="60dp"

    Android:textSize="@dimen/text_size" />
```

**Code để tạo 1 nút nhấn:**

```
<Button

    Android:id="@+id/btOn"

    Android:layout_width="wrap_content"

    Android:layout_height="wrap_content"

    Android:layout_margin="@dimen/margin"

    Android:text="on" />
```

**Code để xử lý sự kiện 1 nút nhấn được nhấn:**

## PHỤ LỤC

---

Nút nhấn được nhấn sẽ gửi trạng thái đèn bằng phương thức GET đến trang dkden.php

```
public void onClick(){
    btOff.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            mClient.get(url: DEN + STATE_OFF, new
            TextHttpResponseHandler() {
                @Override
                public void onFailure(int statusCode, Header[] headers, String
                responseString, Throwable throwable) {
                    Toast.makeText(MainActivity.this, responseString,
                    Toast.LENGTH_SHORT).show();
                }
            }
        }
    })
}
```

**Code đọc dữ liệu gửi về từ trang web:**

```
public void receiveData(){
    AsyncHttpClient DoAm = new AsyncHttpClient();
    DoAm.get(DO_AM, new TextHttpResponseHandler() {
        @Override
        public void onFailure(int statusCode, Header[] headers, String
        responseString, Throwable throwable) {
            Toast.makeText(MainActivity.this, "Fail",
            Toast.LENGTH_SHORT).show();
        }
    })
}
```

Chương trình load :

```
<?php
session_start();
error_reporting(E_ALL & ~E_NOTICE & ~E_DEPRECATED);
```

```
$ND = $_GET['field1'];// lấy giá trị nhiệt độ gửi từ Node mcu

$DA = $_GET['field2'];// lấy giá trị độ ẩm
if($ND)
{
    $file = fopen("ND.php", "w") or die("can't open file");
    fwrite($file, $ND);
    fclose($file);
}
if($DA)
{
    $file1 = fopen("DA.php", "w") or die("can't open file");
    fwrite($file1, $DA);//
    fclose($file1);
}
$myfile = fopen("den.php", "r") or die("error");//mở file
$x = fread($myfile,filesize("den.php"));//đọc file
echo $x; // in ra màn hình
fclose($myfile);

$myfile = fopen("den1.php", "r") or die("error");//mở file
$y = fread($myfile,filesize("den1.php"));//đọc file
echo $y; // in ra màn hình
fclose($myfile);

$myfile = fopen("den2.php", "r") or die("error");//mở file
$z = fread($myfile,filesize("den2.php"));//đọc file
echo $z; // in ra màn hình
fclose($myfile);

$myfile = fopen("den3.php", "r") or die("error");//mở file
$h = fread($myfile,filesize("den3.php"));//đọc file
echo $h; // in ra màn hình
```

```
fclose($myfile);
```

```
?>
```

