

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT  
THÀNH PHỐ HỒ CHÍ MINH**



**ĐỒ ÁN TỐT NGHIỆP  
NGÀNH ĐIỆN TỬ CÔNG NGHIỆP**

**NGHIÊN CỨU, CẢI TIẾN VÀ THI CÔNG BỘ  
ĐÈN LED TÍCH HỢP CẢM BIẾN QUANG,  
CHUYỂN ĐỘNG VÀ WIFI**

**GVHD: TRƯƠNG NGỌC ANH  
SVTT: NGUYỄN HOÀNG MỸ  
MSSV: 15141212**



**Tp. Hồ Chí Minh, tháng 07/2020**



Tp. HCM, ngày 30 tháng 03 năm 2020

## LỊCH TRÌNH THỰC HIỆN ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên: Nguyễn Hoàng Mỹ

Lớp: 15141DT2B

MSSV: 15141212

Tên đề tài: **Nghiên cứu, cải tiến và thi công bộ đèn led tích hợp cảm biến quang, chuyển động và wifi.**

<i>Tuần/ngày</i>	<i>Nội dung chính cần thực hiện</i>	<i>Xác nhận của GVHD</i>
Tuần 1 (09/03 – 15/03)	Gặp GV bộ môn để nghe phổ biến yêu cầu làm đồ án, nhận giấy giới thiệu làm đồ án. Gặp GVHD để chọn đề tài.	
Tuần 2 (16/03 – 22/03)	GVHD tiến hành xét duyệt đề tài và viết đề cương nộp lại.	
Tuần 3 (23/03 – 29/03)	Tham khảo các loại đèn led có cảm biến chuyển động trên thị trường, lựa chọn loại đèn phù hợp để cải tiến.	
Tuần 4 (30/03 – 05/04)	- Tìm hiểu cách kết nối, lập trình của các module cảm biến. - Kết nối các module ngoài thực tế để xem hoạt động.	
Tuần 5 (06/04 – 12/04)	- Tìm hiểu về module ESP8266, cách giao tiếp với các module cảm biến.	
Tuần 6 (13/04 – 19/04)	-Thiết kế sơ đồ khối, giải thích chức năng. - Tính toán lựa chọn linh kiện cho từng khối.	

Tuần 7,8 (20/04 – 03/05)	-Thiết kế sơ đồ nguyên lý. - Tìm hiểu về app Blynk và cách sử dụng.	
Tuần 9,10,11,12 (04/05 – 31/05)	-Thiết kế mạch PCB. -Thi công mạch, xây dựng mô hình. - Viết chương trình cho ESP8266.	
Tuần 13 (01/06 – 07/06)	- Kiểm tra phần cứng, chạy thử và sửa lỗi.	
Tuần 14 (08/06 – 14/06)	- Kiểm tra giao tiếp giữa phần cứng và ứng dụng trên điện thoại, chạy thử và sửa lỗi. - Viết báo cáo.	
Tuần 15,16 (15/06 – 28/06)	-Kiểm tra và hoàn thiện toàn bộ hệ thống. -Hoàn thiện, chỉnh sửa báo cáo gửi cho GVHD để xem xét góp ý lần cuối trước khi in báo cáo.	
Tuần 17 (29/06 – 05/07)	-Nộp quyền báo cáo và làm Slide báo cáo.	

GV HƯỚNG DẪN

(Ký và ghi rõ họ và tên)

# LỜI CAM ĐOAN

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

Người thực hiện

**Nguyễn Hoàng Mỹ**

# LỜI CẢM ƠN

Lời đầu tiên cho phép em gửi lời cảm ơn đến quý thầy cô trường Đại Học Sư Phạm Kỹ Thuật đã dạy dỗ trong suốt những năm tháng đại học. Trong đó phải kể đến quý thầy cô trong khoa Điện- Điện Tử đã truyền đạt những kiến thức cả chuyên môn lẫn kỹ năng để giúp em có những sự chuẩn bị chu đáo nhất. Những kiến thức ấy đã được em đúc kết lại thành đồ án cuối cùng, đồ án tốt nghiệp do chính tay mình tạo ra. Nó như là bàn đạp để em bước ra cánh cửa lớn hơn.

Đặc biệt, em xin gửi lời cảm ơn sâu sắc đến thầy Trương Ngọc Anh đã giúp đỡ em trong quá trình lựa chọn đề tài. Trong quá trình thực hiện cũng có nhiều điều khó khăn, thiếu sót và không thống nhất, nhưng nhờ sự đóng góp ý kiến của thầy, em đã hiểu được cách tìm hiểu những kiến thức ấy.

Em cũng cảm ơn đến bạn bè, anh chị đã có những chia sẻ, trao đổi kiến thức, kinh nghiệm thực tế của mọi người cho em khi gặp khó khăn.

Và cuối cùng là lời cảm ơn đến các bậc phụ huynh, người thân trong gia đình. Nhờ sự quan tâm, ủng hộ và những lời động viên của mọi người đã giúp em hoàn thành xong chương trình đại học để có những kiến thức để ứng dụng vào cuộc sống.

Xin chân thành cảm ơn!

Người thực hiện đề tài

**Nguyễn Hoàng Mỹ**

# MỤC LỤC

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP .....	i
LỊCH TRÌNH THỰC HIỆN ĐỒ ÁN TỐT NGHIỆP .....	ii
LỜI CAM ĐOAN .....	iv
LỜI CẢM ƠN .....	v
MỤC LỤC .....	vi
LIỆT KÊ HÌNH .....	x
LIỆT KÊ BẢNG .....	xiii
TÓM TẮT .....	xiv
CHƯƠNG 1: TỔNG QUAN.....	1
1.1 ĐẶT VẤN ĐỀ.....	1
1.2 MỤC TIÊU ĐỀ TÀI .....	1
1.3 NỘI DUNG THỰC HIỆN.....	1
1.4 GIỚI HẠN CỦA ĐỀ TÀI.....	2
1.5 BỐ CỤC ĐỒ ÁN .....	2
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT .....	4
2.1 GIỚI THIỆU VỀ LED .....	4
2.1.1 Cấu tạo và nguyên lý hoạt động .....	4
2.1.2 Các thông số kỹ thuật.....	5
2.1.3 Ưu điểm và nhược điểm của đèn LED .....	6
2.2 MỘT SỐ LOẠI ĐÈN CẢM ỨNG TRÊN THỊ TRƯỜNG.....	7
2.2.1 Đuôi đèn cảm ứng vi sóng Kawasan .....	7

2.2.2	Đuôi đèn cảm ứng hồng ngoại Kawasan .....	9
2.2.3	Đuôi đèn điều khiển từ xa bằng remote RF-E27 .....	10
2.2.4	Đuôi đèn điều khiển từ xa bằng wifi WL-LC01 .....	11
2.3	CHUẨN GIAO TIẾP UART .....	12
2.4	CẢM BIẾN CHUYỂN ĐỘNG.....	13
2.4.1	Định nghĩa .....	13
2.4.2	Nguyên lý hoạt động.....	14
2.4.3	Ưu điểm và nhược điểm của từng loại cảm biến .....	14
2.5	GIỚI THIỆU MODULE ESP8266.....	15
2.6	GIAO THỨC MQTT.....	20
2.7	MODULE CẢM BIẾN RADAR RCWL-0516.....	25
2.8	CẢM BIẾN ÁNH SÁNG .....	27
2.9	IC LM358 .....	28
2.10	MOC3021 .....	31
2.11	TRIAC BT137.....	32
2.12	IC ỔN ÁP ASM1117 .....	33
2.13	BIẾN TRỞ.....	34
<b>CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ .....</b>		<b>36</b>
3.1	GIỚI THIỆU .....	36
3.2	TÍNH TOÁN VÀ THIẾT KẾ HỆ THỐNG .....	36
3.2.1	Thiết kế sơ đồ khối hệ thống.....	36
3.2.2	Chức năng từng khối.....	37
3.3	TÍNH TOÁN VÀ THIẾT KẾ MẠCH.....	38
3.3.1	Khối cảm biến .....	38



3.3.2	Khối xử lý trung tâm.....	41
3.3.3	Khối biến trở.....	42
3.3.4	Khối OPTO-TRIAC .....	43
3.3.5	Khối điều khiển và giám sát từ xa.....	45
3.3.6	Khối nguồn .....	45
3.4	Sơ đồ nguyên lý toàn mạch .....	48
<b>CHƯƠNG 4: THI CÔNG HỆ THỐNG .....</b>		<b>49</b>
4.1	<b>GIỚI THIỆU .....</b>	<b>49</b>
4.2	<b>THI CÔNG HỆ THỐNG .....</b>	<b>49</b>
4.2.1	Thi công board mạch.....	49
4.2.2	Lắp ráp và kiểm tra.....	52
4.3	<b>ĐÓNG GÓI VÀ THI CÔNG MÔ HÌNH.....</b>	<b>54</b>
4.3.1	Đóng gói bộ điều khiển.....	54
4.3.2	Thi công mô hình .....	55
4.4	<b>LẬP TRÌNH HỆ THỐNG .....</b>	<b>57</b>
4.4.1	Lưu đồ giải thuật .....	57
4.4.2	Phần mềm lập trình cho vi điều khiển.....	63
4.5	<b>HƯỚNG DẪN SỬ DỤNG, THAO TÁC.....</b>	<b>69</b>
4.5.1	Viết tài liệu hướng dẫn sử dụng .....	69
4.5.2	Quy trình thao tác .....	70
<b>CHƯƠNG 5: KẾT QUẢ - NHẬN XÉT - ĐÁNH GIÁ.....</b>		<b>76</b>
5.1	<b>GIỚI THIỆU .....</b>	<b>76</b>
5.2	<b>KẾT QUẢ ĐẠT ĐƯỢC .....</b>	<b>76</b>
5.3	<b>KẾT QUẢ THỰC NGHIỆM.....</b>	<b>77</b>

5.3.1	Mô hình.....	77
5.3.2	Điều khiển, giám sát thiết bị thông qua ứng dụng Blynk .....	77
5.4	<b>NHẬN XÉT VÀ ĐÁNH GIÁ .....</b>	<b>80</b>
5.4.1	Nhận xét.....	80
5.4.2	Đánh giá.....	80
<b>CHƯƠNG 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN .....</b>		<b>82</b>
6.1	<b>KẾT LUẬN.....</b>	<b>82</b>
6.2	<b>HƯỚNG PHÁT TRIỂN .....</b>	<b>82</b>
<b>TÀI LIỆU THAM KHẢO .....</b>		<b>83</b>
<b>PHỤ LỤC .....</b>		<b>84</b>

# LIỆT KÊ HÌNH

Hình 2. 1: Cấu tạo của bóng đèn LED .....	4
Hình 2. 2: Sự di chuyển của hạt mang điện qua tiếp giáp P-N .....	5
Hình 2. 3: Đuôi đèn cảm ứng vi sóng Kawasan.....	8
Hình 2. 4: Đuôi đèn cảm ứng hồng ngoại Kawasan.....	9
Hình 2. 5: Bộ đuôi đèn điều khiển bằng remote RF-E27 .....	11
Hình 2. 6: Đuôi đèn điều khiển từ xa WL-LC01 .....	12
Hình 2. 7: Hệ thống truyền dữ liệu bất đồng bộ.....	13
Hình 2. 8: Đuôi đèn cảm ứng vi sóng (radar).....	15
Hình 2. 9: Sơ đồ chân của ESP8266MOD .....	16
Hình 2. 10: Sơ đồ chân của Module ESP-07 .....	19
Hình 2. 11: Mô hình giao thức MQTT .....	21
Hình 2. 12: Hoạt động của Client.....	22
Hình 2. 13: Hoạt động của topic .....	23
Hình 2. 14: Module Cảm Biến Radar RCWL-0516.....	27
Hình 2. 15: Cảm biến ánh sáng .....	28
Hình 2. 16: Sơ đồ chân IC LM358 .....	29
Hình 2. 17: Sơ đồ nguyên lý mạch chiếu sáng tự động dùng IC LM358 .....	30
Hình 2. 18: Sơ đồ chân Moc3021 .....	31
Hình 2. 19: Sơ đồ chân Triac BT317 .....	32
Hình 2. 20: IC ổn áp ASM1117 .....	33
Hình 2. 21: Các ký hiệu của biến trở.....	34
Hình 2. 22: Cấu tạo của biến trở .....	35
Hình 2. 23: Các loại biến trở .....	35
Hình 3. 1: Sơ đồ khối hệ thống .....	37
Hình 3. 2: Sơ đồ mạch cảm biến quang trở .....	39
Hình 3. 3: Sơ đồ mạch cảm biến chuyển động radar .....	40
Hình 3. 4: Module wifi ESP07 .....	42

Hình 3. 5: Sơ đồ nối dây module wifi ESP07 .....	42
Hình 3. 6: Sơ đồ nối dây biến trở VR1.....	43
Hình 3. 7: Biến trở vuông.....	43
Hình 3. 8: Sơ đồ nối dây ngõ ra opto-triac .....	44
Hình 3. 9: Sơ đồ nguyên lý khối nguồn .....	45
Hình 3. 10: Mạch chuyển đổi AC-DC 220V ra 5V.....	46
Hình 3. 11: Sơ đồ nguyên lý toàn mạch .....	48
Hình 4. 1: PCB mặt trên .....	50
Hình 4. 2: PCB mặt dưới.....	50
Hình 4. 3: Sơ đồ bố trí linh kiện mặt trên.....	51
Hình 4. 4: Sơ đồ bố trí linh kiện mặt dưới .....	51
Hình 4. 5: Hai mặt của mạch PCB đã gia công khi chưa lắp linh kiện.....	53
Hình 4. 6: Mặt trên board khi lắp linh kiện.....	53
Hình 4. 7: Mặt dưới board khi lắp linh kiện.....	54
Hình 4. 8: Board mạch sau khi được cho vào hộp đèn .....	55
Hình 4. 9: Mô hình mặt trước.....	56
Hình 4. 10: Mô hình mặt sau .....	56
Hình 4. 11: Lưu đồ điều khiển thiết bị .....	58
Hình 4. 12: Lưu đồ lựa chọn chế độ hoạt động của đèn .....	60
Hình 4. 13: Lưu đồ giải thuật Web Server .....	62
Hình 4. 14: Quy trình làm việc của Arduino.....	64
Hình 4. 15: Giao diện lập trình Arduino .....	64
Hình 4. 16: Giao diện menu arduino IDE .....	65
Hình 4. 17: Giao diện file menu arduino IDE.....	65
Hình 4. 18: Giao diện Examples menu .....	66
Hình 4. 19: Giao diện Sketch menu .....	67
Hình 4. 20: Giao diện Tool Menu Arduino IDE .....	67
Hình 4. 21: Board ESP8266 sử dụng .....	68
Hình 4. 22: Arduino Toolbar .....	69

Hình 4. 23: Chương trình nạp thành công.....	69
Hình 4. 24: Quy trình thao tác.....	70
Hình 4. 25: Giao diện Project điều khiển đèn .....	71
Hình 4. 26: Ấn nút chọn chế độ hoạt động của đèn (Auto/Manual) .....	72
Hình 4. 27: Giao diện đổi wifi kết nối với đèn.....	73
Hình 4. 28: Giao diện Web Server .....	74
Hình 4. 29: Giao diện đăng nhập.....	74
Hình 4. 30: Giao diện Web Server chọn chế độ và điều khiển đèn. ....	75
Hình 5. 1: Mô hình hộp đèn hoàn chỉnh.....	77
Hình 5. 2: Điều chỉnh mở đèn ở chế độ Manual .....	78
Hình 5. 3: Điều chỉnh tắt đèn ở chế độ Manual. ....	78
Hình 5. 4: Giao diện lúc đèn sáng ở chế độ Auto .....	79
Hình 5. 5: Giao diện lúc đèn tắt ở chế độ Auto.....	79

# LIỆT KÊ BẢNG

Bảng 2. 1: Bảng so sánh thông số ESP-01, ESP-07, Wemos D1 Mini.....	17
Bảng 2. 2: Thông số các chân của Board ESP-07 .....	19
Bảng 3. 1: Thông số, giá trị các linh kiện sử dụng.....	45
Bảng 4. 1: Danh sách linh kiện sử dụng.....	52
Bảng 5. 1: Số liệu thực nghiệm .....	81

# TÓM TẮT

Ngày nay công nghệ trở nên hiện đại, xu hướng mọi thứ đều sẽ được kết nối và giám sát thông qua mạng không dây wifi (Wireless Fidelity) và điều khiển các thiết bị theo tự động hóa. Với ý tưởng giải quyết những bất cập của điều khiển tự động, em xin đưa ra đề tài: ***“Nghiên cứu, cải tiến và thi công bộ đèn led tích hợp cảm biến quang, chuyển động và wifi.”***

Nội dung chính của đề tài:

- Sử dụng board ESP8266 làm mạch điều khiển trung tâm.
- Board kết hợp các cảm biến quang và chuyển động.
- Điều khiển đèn trên ứng dụng Blynk và giao diện Web thông qua Internet.
- Lưu trữ và giám sát trạng thái của đèn.

Với đề tài này, em hi vọng sẽ làm cơ sở nghiên cứu cho các nhóm sau có thể mở rộng, phát triển nữa.

# CHƯƠNG 1: TỔNG QUAN

## 1.1 ĐẶT VẤN ĐỀ

Ngày nay, khoa học công nghệ phát triển với tốc độ nhanh chóng. Cuộc sống của chúng ta vì vậy mà cũng thay đổi, văn minh và hiện đại hơn. Sự phát triển của ngành kỹ thuật điện tử đã tạo ra nhiều thiết bị gọn nhẹ, xử lý nhanh và rất chính xác. Cùng với sự phát triển vượt trội của cảm biến trong những năm gần đây, các thiết bị điện tử ứng dụng cảm biến cũng ra đời, với nhiều tính năng, góp phần nâng cao chất lượng cuộc sống.

Thiết nghĩ, việc giám sát và điều khiển thiết bị chiếu sáng trong các phòng được thực hiện một cách tự động là cần thiết và rất có ích cho cuộc sống sinh hoạt hàng ngày của mọi người.

Trên cơ sở kiến thức đã học và mong muốn ứng dụng những kiến thức đó vào trong đời sống hiện đại, em quyết định thực hiện đề tài ***“NGHIÊN CỨU, CẢI TIẾN VÀ THI CÔNG BỘ ĐÈN LED TÍCH HỢP CẢM BIẾN QUANG, CHUYỂN ĐỘNG VÀ WIFI”***

## 1.2 MỤC TIÊU ĐỀ TÀI

Nghiên cứu và cải tiến mô hình bộ đèn led gồm các mục tiêu cụ thể:

- Tìm hiểu về cảm biến quang trở và module ESP8266 từ đó nghiên cứu tích hợp vào mô hình bộ đèn led sử dụng cảm biến chuyển động.
- Viết được chương trình giao tiếp giữa các module ESP8266 và điện thoại.
- Viết được ứng dụng giám sát từ xa bằng điện thoại.
- Cập nhật được dữ liệu lên server giám sát và điều khiển được bóng đèn thông qua mạng Internet.

## 1.3 NỘI DUNG THỰC HIỆN

- Nội dung 1: Nghiên cứu về bộ đèn led có cảm ứng chuyển động.
- Nội dung 2: Tìm hiểu về module ESP8266.
- Nội dung 3: Tìm hiểu về LED.



## CHƯƠNG 1: TỔNG QUAN

---

- Nội dung 4: Tìm hiểu cách kết nối giữa bộ đèn led và module ESP8266.
- Nội dung 5: Nghiên cứu và kết nối được bộ đèn led có cảm ứng chuyển động với cảm biến quang và module ESP8266.
- Nội dung 6: Truyền và nhận được dữ liệu được qua Server.
- Nội dung 7: Lập trình và điều khiển thiết bị qua điện thoại.
- Nội dung 8: Thi công, chạy thử nghiệm và hiệu chỉnh sản phẩm.
- Nội dung 9: Đánh giá kết quả thực hiện.
- Nội dung 10: Viết báo cáo thực hiện.
- Nội dung 11: Tiến hành bảo vệ luận văn.

### 1.4 GIỚI HẠN CỦA ĐỀ TÀI

- Thiết kế trên mô hình với ít thiết bị được điều khiển chưa phát triển trên hệ thống lớn với số lượng nhiều thiết bị.
- Mô hình chỉ thực hiện trên bộ xử lý trung tâm là module ESP8266 chưa áp dụng cho các bộ xử lý trung tâm như PIC, ARM...
- Chưa điều chỉnh được độ sáng của đèn.

### 1.5 BỐ CỤC ĐỒ ÁN

- **Chương 1:** Tổng Quan.
- **Chương 2:** Cơ Sở Lý Thuyết.
- **Chương 3:** Tính Toán Và Thiết Kế Hệ Thống.
- **Chương 4:** Thi Công Hệ Thống.
- **Chương 5:** Kết Quả, Nhận Xét, Đánh Giá.
- **Chương 6:** Kết Luận và Hướng Phát Triển.

#### Chương 1: Tổng Quan

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

## **CHƯƠNG 1: TỔNG QUAN**

---

### **Chương 2: Cơ Sở Lý Thuyết**

Giới thiệu các linh kiện, thiết bị sử dụng thiết kế hệ thống, giao thức.

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

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

### **Chương 4: Thiết Kế Hệ Thống**

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

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

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

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

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

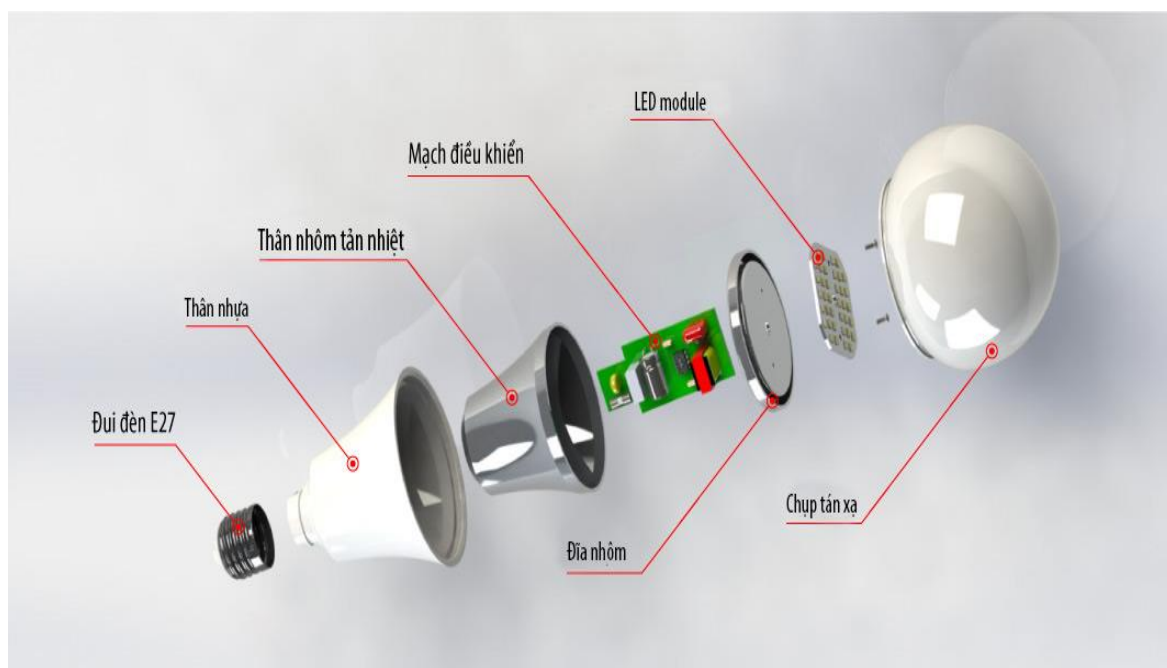
# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

## 2.1 GIỚI THIỆU VỀ LED

### 2.1.1 Cấu tạo và nguyên lý hoạt động

LED (Light-Emitting-Diode) là một thiết bị bán dẫn phát ra ánh sáng nhìn thấy được khi có dòng điện đi qua nó. Về cơ bản, có thể hiểu nó là một thiết bị bán dẫn hoạt động trái ngược với pin năng lượng mặt trời (thiết bị có khả năng hấp thụ ánh sáng để chuyển thành dòng điện).

Tùy thuộc vào cấu tạo của các chất bán dẫn mà thiết bị này sẽ cho ra các ánh sáng với bước sóng khác nhau, điều này đồng nghĩa với việc các nguồn ánh sáng sẽ có màu sắc khác nhau tùy thuộc vào chất bán dẫn cấu tạo nên đèn.

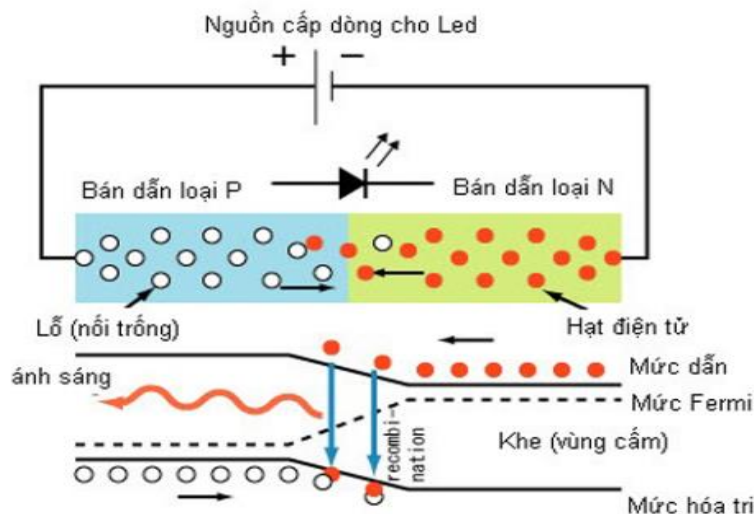


**Hình 2. 1: Cấu tạo của bóng đèn LED**

Đèn LED bao gồm hai loại vật liệu bán dẫn (loại P và loại N giống tế bào quang điện). Cả hai loại vật liệu loại P và N đều được pha các tạp chất để thay đổi một chút tính chất điện của chúng, vật liệu loại P chứa lỗ trống điện tử còn vật liệu loại N chứa điện tử

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

(electron). Các vật liệu loại P được tạo ra bằng cách sử dụng các nguyên tố (như Bo) có ít electron hóa trị hơn vật liệu nguyên chất ban đầu. Các vật liệu loại N thì được tạo ra bằng cách sử dụng các nguyên tố (như phốt pho) có nhiều electron hóa trị hơn vật liệu ban đầu. Khi kết hợp 2 vật liệu này tạo ra một tiếp giáp P-N với các thuộc tính hữu ích cho các ứng dụng điện tử.



Hình 2. 2: Sự di chuyển của hạt mang điện qua tiếp giáp P-N

Khi dòng điện chạy từ vật liệu loại P (cực âm) đến vật liệu loại N (cực dương), các electron sẽ lấp đầy các lỗ trống điện tử và từ đó sinh ra bức xạ ánh sáng nhìn thấy được.

### 2.1.2 Các thông số kỹ thuật

**Quang thông:** Hay còn gọi là thông lượng phát sáng là thước đo tổng lượng ánh sáng nhìn thấy được phát ra từ đèn, đơn vị là Lumen ( $Lm$ ). Do đó, tùy vào nhu cầu sử dụng ánh sáng của bạn mà lựa chọn đèn LED quang thông lớn để phát ra nhiều ánh sáng hay quang thông thấp để cung cấp nguồn sáng nhẹ nhàng hơn (ví dụ: đèn phòng ngủ).

**Chỉ số hoàn màu CRI:** Chỉ số này cho ta biết chất lượng ánh sáng của đèn và mức độ bị ảnh hưởng màu sắc, độ trung thực của vật khi được chiếu sáng. Đây là thông số rất quan trọng đối với đèn LED.

**Hiệu suất phát sáng:** Thông số này cho bạn biết được thiết bị chiếu sáng của bạn có tiết kiệm điện hay không?

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

---

**Nhiệt độ màu (đơn vị K):** Hiện nay trên thị trường ở Việt Nam, có 3 loại bóng LED phổ biến với các mức nhiệt độ màu khác nhau như sau:

- Màu ấm (vàng): nhiệt độ màu từ 2700K – 3500K tương đương với ánh sáng của đèn sợi đốt, rất thích hợp để trang trí và chiếu sáng phòng ngủ.
- Ánh sáng tự nhiên: nhiệt độ màu từ 4000K – 4500K. Đây là khoảng nhiệt độ màu trung bình, cho phép đèn LED phát ra ánh sáng vừa phải, hài hòa, rất thích hợp để lắp đặt ở phòng ăn.
- Ánh sáng trắng: nhiệt độ màu từ 5000K – 7000K, bóng LED siêu sáng, tương đương với ánh sáng mặt trời vào giữa trưa. Loại này rất thích hợp để chiếu sáng cho không gian văn phòng làm việc, phòng tiếp khách, sân thể thao trong nhà...

### 2.1.3 Ưu điểm và nhược điểm của đèn LED

#### ❖ Ưu điểm:

- Tuổi thọ lâu dài, tiết kiệm chi phí bảo trì.
- Hiệu quả cao hơn đáng kể so với đèn sợi đốt (ít tổn năng lượng hơn 80% khi phát sáng). Đèn LED có thể tạo ra 100 Lm ánh sáng với mỗi watt điện, trong khi cũng với 1W điện đèn halogen chỉ có thể tạo ra 10 – 15 Lumen.
- Tỏa nhiệt ít hơn rất nhiều so với đèn sợi đốt.
- Sản sinh cực kỳ ít hoặc không có tia hồng ngoại và tia cực tím, điều này làm giảm tối thiểu nguy cơ phát cháy.
- Chuyển đổi trạng thái bật/tắt đèn nhanh hơn các loại đèn khác và khả năng hư hỏng do bật/tắt liên tục thấp hơn rất nhiều.
- Đèn LED thường có cấu trúc cứng cáp hơn, khó bị vỡ hơn các loại đèn khác.
- Chúng thân thiện với môi trường hơn bất cứ loại đèn nào, không thải ra khí CO<sub>2</sub>, không chứa bất kỳ vật liệu độc hại nào trong sản xuất.
- Đa dạng màu sắc ánh sáng.

#### ❖ Nhược điểm:

## **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT**

---

- Các sản phẩm đèn công nghệ LED thường đắt tiền hơn để mua, nhưng hiện nay nhu cầu về dòng thiết bị này đang tăng cao nên giá thành của chúng đang giảm dần hàng năm.
- Dễ hư hỏng trong điều kiện nhiệt độ cao.
- CRI không phải lúc nào cũng tốt như đèn sợi đốt hoặc halogen, nhưng vấn đề này đang dần được cải thiện tốt hơn.
- Lượng sáng và chất lượng màu sắc có thể bị giảm theo tuổi thọ của đèn (đây là nhược điểm mà loại bóng đèn nào cũng có).
- Đây là công nghệ chiếu sáng được cho là mới nhất, nên việc tìm kiếm được các thông số kỹ thuật và lời khuyên chính xác có thể sẽ khó khăn với bạn.
- Đã có một số vấn đề được báo cáo về việc đèn led chớp nháy liên tục. Có nhiều nguyên nhân khiến điều này xảy ra như không tương thích với biến trở điều chỉnh độ sáng, không tương thích với máy biến áp... Tuy nhiên vấn đề này rất hiếm khi xảy ra.

Với những ưu điểm đã nêu ở trên em chọn đèn LED cho đề tài này.

### **2.2 MỘT SỐ LOẠI ĐÈN CẢM ỨNG TRÊN THỊ TRƯỜNG**

#### **2.2.1 Đuôi đèn cảm ứng vi sóng Kawasan**

- Đây là thiết bị cảm ứng sử dụng công nghệ vi sóng. Sử dụng sóng Radar tần số 5.8GHz nhận diện sự chuyển động thân nhiệt với độ nhạy rất cao, tính năng xuyên gỗ, kính, tường mỏng... Có tác dụng tự động mở đèn khi có người di chuyển trong vùng quét. Với chức năng điều chỉnh khoảng cách xa gần, kết hợp cùng với chức năng điều chỉnh được độ LUX (sáng/tối) của môi trường nếu đủ sáng thì không cần mở đèn để tiết kiệm điện.
- Với đặc tính phát hiện chuyển động liên tục nên thời gian mở đèn được di trì trong khoảng thời gian cài đặt đến khi không còn sự di chuyển nào trong vùng có sóng radar của thiết bị phát ra thì thiết bị sẽ tắt đèn.
- Giá thành khoảng 225.000VNĐ.



*Hình 2. 3: Đầu đèn cảm ứng vi sóng Kawsan*

### Thông số kỹ thuật:

- Dùng công nghệ sóng radar dải tần 5.8GHz.
- Điện áp hoạt động: 180-220VAC/50-60Hz.
- Góc quét: 360°/110°.
- Khoảng cách cảm ứng: 3-8 m (tương ứng với bán kính quét 1-5m).
- Thời gian điều chỉnh tự tắt: 20 giây đến 5 phút.
- Độ sáng điều chỉnh cảm biến hoạt động: từ 2 LUX(tối) đến 800 LUX(sáng).
- Công suất tiêu thụ: nhỏ hơn 0.2W.
- Công suất ra đèn đui xoắn E27 :100W (đèn compact).
- Nhiệt độ hoạt động: -10 đến 70°C.
- Độ cao lắp đặt: 1.5m~ 5m.
- Cấp độ kín nước: IP20 lắp trong nhà, chống bụi, hơi nước.

### Các ứng dụng thực tế:

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

---

- Ứng dụng cơ bản nhất của cảm ứng là dùng để tắt mở đèn cho nhà vệ sinh, phòng khách, phòng chờ, hành lang, cầu thang khách sạn hoặc nhà ở.
- Lắp đặt cho những cầu thang có khoảng cách xa, lắp cảm ứng vi sóng tại vị trí đèn cầu thang thì khi di chuyển từ trên xuống hoặc từ dưới lên thì cảm ứng cũng nhận được sóng và bật đèn kịp thời.
- Giải pháp cho các tòa nhà văn phòng hoặc nhà máy, phòng thay đồ nhà máy.
- Ta lắp đặt cảm ứng nối với bóng đèn và điều chỉnh sao cho góc quét hướng về phía có người di chuyển, sau đó điều chỉnh 2 nút: nút thời gian tắt đèn và độ sáng cảm ứng.

### 2.2.2 Đuôi đèn cảm ứng hồng ngoại Kawasan

Đây là thiết bị cảm biến nhận diện sự chuyển động của con người.

Tự động mở đèn khi có người di chuyển và thiếu ánh sáng (dưới 10 lux). Tự động mở đèn khi thiếu ánh sáng (dưới 10 lux) có người di chuyển trong vùng cảm ứng.

Giá thành khoảng 180.000VNĐ.



*Hình 2. 4: Đuôi đèn cảm ứng hồng ngoại Kawasan*

**Thông số kỹ thuật:**



## **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT**

---

- Điện áp nguồn: 180 ~ 230VAC.
- Tần số: 50Hz.
- Công suất tiêu thụ max: 36w hoặc 20w (LED).
- Thiết kế: Đui đèn chuẩn E27.
- Bán kính và góc quét cảm ứng: 2-4m/ 360 x 110 độ.
- Thời gian tự tắt: 120 giây.

### **Các ứng dụng thực tế:**

- Lắp đặt ở hành lang, cầu thang.
- Lắp đặt ở nhà vệ sinh, công.

### **2.2.3 Đui đèn điều khiển từ xa bằng remote RF-E27**

- Điều khiển bằng remote tiện lợi tạo nên sự tiện nghi và tinh tế.
- Điện áp 220v 50hz.
- Điều khiển xuyên tường, điều khiển đèn ở khoảng cách 20m.
- Đui đèn E27 kèm điều khiển từ xa (remote) bằng sóng Radio RF.
- Dễ sử dụng và lắp đặt. Đặc biệt điều khiển từ xa bằng sóng Radio nên rất ổn định, khoảng cách điều khiển xa từ 15-20m và điều khiển được cả khi xuyên tường hay có vật cản.
- Giá thành khoảng 60.000VNĐ.



*Hình 2. 5: Bộ đuôi đèn điều khiển bằng remote RF-E27*

### **2.2.4 Đuôi đèn điều khiển từ xa bằng wifi WL-LC01**

WL-LC01 là một đuôi đèn wifi thông minh có thể kết nối với bóng đèn cho phép người dùng điều khiển từ xa bật/tắt bóng đèn được kết nối với đuôi đèn thông qua app ứng dụng trên điện thoại ở bất cứ đâu có Internet.

Một tính năng khác của sản phẩm đó là hẹn giờ tắt bật đèn. Bạn có thể hẹn giờ đếm ngược hay đặt lịch trình tự động tắt bật cho bóng đèn nhằm tạo ra những ngữ cảnh thông minh cũng như tránh lãng phí điện năng do quên tắt đèn.

Giá thành khoảng 220.000VNĐ.



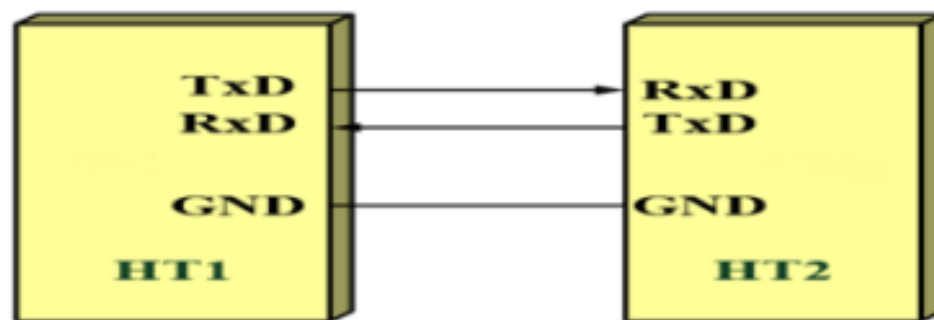
*Hình 2. 6: Đuôi đèn điều khiển từ xa WL-LC01*

### **Thông số kỹ thuật:**

- Điện áp nguồn: AC85-265V 50/60Hz.
- Công suất: Max 2A-200W (với bóng đèn sợi đốt).
- Wifi: 2.4 GHz IEEE802.11 b/g/n.
- Đui đèn: Chuẩn E27.

### **2.3 CHUẨN GIAO TIẾP UART**

Các tên đầy đủ UART là “Universal Asynchronous Receiver / Transmitter”, và nó là một vi mạch sẵn có trong một vi điều khiển nhưng không giống như một giao thức truyền thông (I2C & SPI). Chức năng chính của UART là truyền dữ liệu nối tiếp. Trong UART, giao tiếp giữa hai thiết bị có thể được thực hiện theo hai cách là giao tiếp dữ liệu nối tiếp và giao tiếp dữ liệu song song.



*Hình 2. 7: Hệ thống truyền dữ liệu bất đồng bộ*

Truyền thông nối tiếp và song song

Trong giao tiếp dữ liệu nối tiếp, dữ liệu có thể được truyền qua một cáp hoặc một đường dây ở dạng bit-bit và nó chỉ cần hai cáp. Truyền thông dữ liệu nối tiếp không đắt khi chúng ta so sánh với giao tiếp song song. Nó đòi hỏi rất ít mạch cũng như dây. Vì vậy, giao tiếp này rất hữu ích trong các mạch ghép so với giao tiếp song song.

Trong giao tiếp dữ liệu song song, dữ liệu có thể được truyền qua nhiều cáp cùng một lúc. Truyền dữ liệu song song tốn kém nhưng rất nhanh, vì nó đòi hỏi phần cứng và cáp bổ sung. Các ví dụ tốt nhất cho giao tiếp này là máy in cũ, PCI, RAM, v.v...

## 2.4 CẢM BIẾN CHUYỂN ĐỘNG

### 2.4.1 Định nghĩa

Cảm biến chuyển động là một thiết bị phát hiện chuyển động vật lý trên thiết bị hoặc trong môi trường. Nó có khả năng phát hiện và nắm bắt các chuyển động vật lý hoặc động học trong thời gian thực.

Một cảm biến chuyển động còn được gọi là một máy phát hiện chuyển động. Tùy thuộc vào khả năng của cảm biến chuyển động, nó có thể phát hiện chuyển động trong thiết bị được tích hợp hoặc môi trường xung quanh. Nó thường được kết nối với một hệ thống hoặc phần mềm xử lý chuyển động thành một hành động hoặc thông tin. Ví dụ: trong điện thoại thông minh, cảm biến chuyển động chủ yếu được sử dụng để lấy đầu vào từ người dùng trong các trò chơi được hỗ trợ và các ứng dụng khác.

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

---

Cảm biến chuyển động thường dùng trong sản xuất đèn cảm biến là cảm biến hồng ngoại và cảm biến radar.

### 2.4.2 Nguyên lý hoạt động

Công nghệ cảm biến radar (hay còn gọi là cảm biến vi sóng) hoạt động dựa trên hiệu ứng doppler, tần số 5.8 Ghz dưới 10m sẽ nhận tín hiệu khi người chuyển động qua và sẽ phản hồi lại đóng tiếp điểm mạch điện lại cấp nguồn cho đèn sáng. Cảm biến radar có khả năng cảm biến xuyên vật cản như tường mỏng, gỗ, kính....

Nguyên lý hoạt động của cảm biến hồng ngoại là nhờ đầu dò PIR và lăng kính fresnel để phát hiện chuyển động ngang của con người qua cảm ứng thân nhiệt. Cảm biến hồng ngoại không tự phát ra tia hồng ngoại mà nó nhận tia hồng ngoại phát ra từ thân thể người và vật. Khi có một đối tượng đi qua, PIR sẽ phát hiện được thay đổi của nhiệt độ.

### 2.4.3 Ưu điểm và nhược điểm của từng loại cảm biến

#### ❖ Ưu điểm:

- Nếu muốn dùng loại đèn có độ nhạy cao, có khả năng phát hiện chuyển động nhỏ xuyên tường mỏng, gỗ, nhựa... thì có thể chọn đèn dùng công nghệ cảm biến radar. Với công nghệ này, đèn có góc quét rộng (360 độ) không điểm chết và cảm ứng với khoảng cách 6 – 8m. Đèn áp dụng công nghệ radar sẽ không phụ thuộc cảm biến vào nhiệt độ môi trường.
- Đối với công nghệ cảm biến hồng ngoại, đèn có khả năng phân biệt được cả chuyển động của người và đồ vật. Bạn hoàn toàn có thể điều chỉnh được góc cảm ứng theo ý muốn của mình và sử dụng vách ngăn để tránh vùng không muốn cảm ứng.

#### ❖ Nhược điểm:

- Độ nhạy cao của cảm biến radar đôi khi cũng là nhược điểm của công nghệ này vì dễ bị nhầm lẫn khi phát hiện vật chuyển động.

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

- Đèn dùng công nghệ cảm biến hồng ngoại lại có yếu điểm là góc quét nhỏ, có điểm chết, không thể cảm biến xuyên vật cản. Chính vì công nghệ này phụ thuộc nhiều vào nhiệt độ để cảm biến nên tại môi trường có nhiệt độ cao thì đèn càng kém nhạy. Phạm vi đèn cảm biến được chỉ trong khoảng 2 – 3m.

Với những ưu, nhược điểm đã nêu và theo yêu cầu của đề tài, em quyết định chọn loại đèn sử dụng công nghệ cảm biến radar để thực hiện mô hình.



**Hình 2. 8: Đuôi đèn cảm ứng vi sóng (radar)**

### 2.5 GIỚI THIỆU MODULE ESP8266

ESP8266 là dòng chip tích hợp Wi-Fi 2.4Ghz có thể lập trình được, giá thành rẻ và được sản xuất bởi một công ty bán dẫn Trung Quốc: Espressif Systems.

Phiên bản đầu tiên được phát hành vào tháng 8 năm 2014, đóng gói đưa ra thị trường dạng Module ESP 01 do bên thứ 3 là AI-Thinker sản xuất ra. Module này có khả năng kết nối Internet qua mạng Wi-Fi một cách nhanh chóng và sử dụng rất ít linh kiện

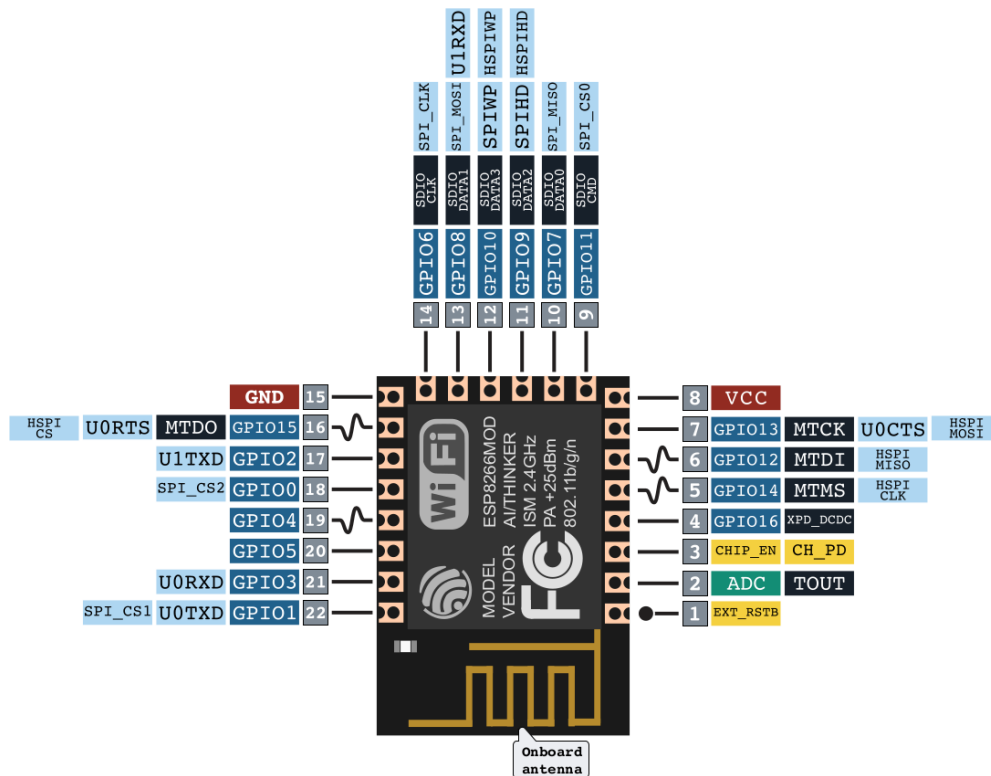
## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

đi kèm. Với giá cả phù hợp cũng như các tính năng vượt trội ESP8266 ngày càng được dùng nhiều trong cuộc sống đặc biệt là các dự án học tập nghiên cứu của sinh viên.

ESP8266 có một cộng đồng các nhà phát triển trên thế giới rất lớn, cung cấp nhiều Module lập trình mã mở giúp nhiều người có thể tiếp cận và xây dựng ứng dụng rất nhanh. Hiện nay, tất cả các dòng chip ESP8266 trên thị trường đều mang nhãn ESP8266EX, là phiên bản nâng cấp của ESP8266.

### ❖ Sơ đồ chân của ESP8266

Sơ đồ chi tiết các chân của ESP8266 được làm rõ trong hình 2. 2



Hình 2. 9: Sơ đồ chân của ESP8266MOD

### ❖ Thông số phần cứng

- Bộ xử lý lõi ESP8266 tích hợp MCU micro 32-bit công suất thấp Tensilica Xtensa LX106 chạy với tốc độ 80 MHz và 160Mhz.
- Bộ nhớ Flash ngoài từ 512KB đến 4MB.

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

---

- Chuẩn Wi-fi IEEE 802.11 b/g/n, Wi-Fi 2.4 GHz. Tích hợp giao thức TCP/IP.
- Có 16 chân GPIO.
- Tốc độ truyền UART lên đến 4Mbps.
- Hỗ trợ SDIO 2.0, UART, SPI, I2C, PWM, I2S với DMA.
- Một bộ chuyển đổi ADC có độ chính xác cao 10-bit.
- Dải nhiệt độ hoạt động rộng: -40C đến 125C.
- Có thể dùng tập lệnh AT.
- Hỗ trợ phát triển trên cả hai môi trường hệ điều hành Windows và Linux.

### ❖ Các loại Module ESP8266 trên thị trường:

Để đơn giản cho người sử dụng ở mọi lứa tuổi, ngành nghề, dự án trên thị trường xuất hiện rất nhiều Module và Board mạch phát triển toàn diện phù hợp với các yêu cầu trên. Một số Module ESP8266 điển hình như: ESP-01, ESP-02, ESP-03, ESP-07, ESP-12F, ... và một số Board mạch phát triển như: NodeMCU, Wemos, ... Trong đó, có ba loại được sử dụng nhiều và cũng khá là tốt như ESP-01, ESP-07 và Wemos D1 Mini. Với giá thành hợp lý, có thể đáp ứng nhu cầu các dự án tiên tiến và một điều đặc biệt là các Module và Board mạch trên còn tương thích với Arduino IDE rất tiện lợi cho việc lập trình để phát triển.

**Bảng 2. 1: Bảng so sánh thông số ESP-01, ESP-07, Wemos D1 Mini**

Thông số	ESP-01	ESP-07	Wemos D1 Mini
Số chân GPIO	4 (bao gồm TX và RX)	11	11
Số chân ADC	1	1	1
Bộ nhớ Plash	1MB (phiên bản nâng cấp)	4MB	4MB



## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

---

Chú thích các chân	Không	Có	Có
USB chuyển đổi	Không	Có	Có
Kích thước	24,75 x 14,5mm	21,2 x 16mm	34,2 x 25,6mm

Sau khi tìm hiểu một số loại Module và Board mạch phát triển em lựa chọn Module ESP-07. Với các đặc điểm đó là nhiều chân I/O, giá cả hợp lý, rất dễ tìm và mua ngoài thị trường, nạp chương trình điều khiển cho Board dễ dàng thông qua Arduino IDE và cổng COM của máy tính.

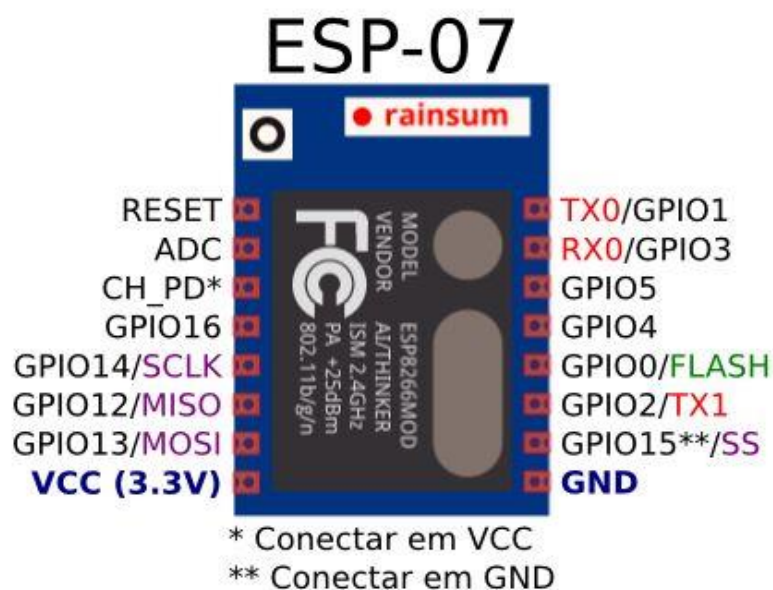
Tổng quan về Module ESP-07:

- Tích hợp sẵn anten ceramic và anten ngoài.
- Wifi: 2.4GHz, 802.11 b/g/n.
- Hỗ trợ bảo mật WPA/WPA2.
- Tích hợp giao thức TCP/IP.
- Số chân I/O: 11 (tất cả các chân I/O đều có Interrupt/PWM/I2C/One-wire, trừ chân D0).
- Số chân Analog Input: 1, 10bit (điện áp vào tối đa 3.3V).
- Hỗ trợ kết nối với smart phone (IOS và Android).
- Bộ nhớ Flash: 4MB.
- Tốc độ truyền Serial (Baurate): 115200 (Max).
- Điện áp hoạt động: 3.3V.
- Điện áp ra I/O: Tối đa 3.6V.
- Điện năng tiêu thụ ở chế độ chờ: nhỏ hơn 1.0mW.

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

- Kích thước: 16mm x 21.2mm.

Dưới đây là sơ đồ chân của Module ESP-07:



*Hình 2. 10: Sơ đồ chân của Module ESP-07*

Chức năng chi tiết từng chân của Board ESP-07 được làm rõ trong bảng 2-2:

*Bảng 2. 2: Thông số các chân của Board ESP-07*

STT	Chân	Mô tả
1	RST	Reset module
2	ADC	Chuyển đổi analog/digital
3	CH_PD	Chân cho phép hoạt động, cấp nguồn 3.3VDC
4	GPIO16	GPIO16
5	GPIO14	GPIO14; HSPI_CLK
6	GPIO12	GPIO12; HSPI_MISO
7	GPIO13	GPIO13; HSPI_MOSI

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

---

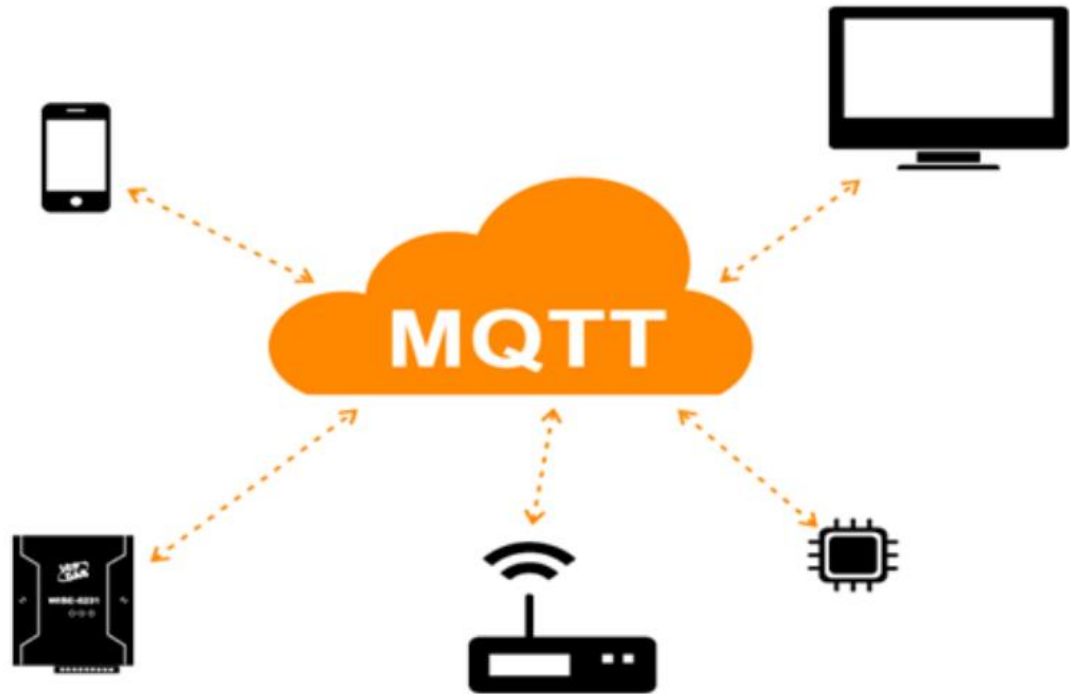
8	VCC	Chân cấp nguồn 3.3VDC
9	GND	Nối đất
10	GPIO15	GPIO15, MTDO, HSPICS, Nối mức thấp.
11	GPIO2	GPIO2
12	GPIO0	GPIO0 (UART_low, Flash Boot_high)
13	GPIO4	GPIO4
14	GPIO5	GPIO5
15	RX0	UART0-RXD
16	TX0	UART0-TXD

### 2.6 GIAO THỨC MQTT

ESP8266 là module wifi có chức năng thu và phát sóng wifi, được ứng dụng nhiều trong các dự án IOT. Và để sử dụng ESP8266 một cách triệt để, thì cần kết hợp với giao thức MQTT.

MQTT (Message Queue Telemetry Transport) là một giao thức truyền thông điệp (message) dạng publish/subscribe (xuất bản/theo dõi) sử dụng cho các thiết bị Internet of Things với băng thông thấp, độ tin cậy cao và khả năng được sử dụng trong mạng lưới không ổn định.

Là một giao thức rất nhẹ do đó được sử dụng để giao tiếp các thiết bị (M2M - Machine to Machine), WSN (Wireless Sensor Networks) và phổ biến nhất trong các dự án IoT.



*Hình 2. 11: Mô hình giao thức MQTT*

❖ **Lịch sử:**

MQTT được tạo ra bởi tiến sĩ Andy Stanford-Clark của IBM và Arlen Nipper của Arcom vào năm 1999 như là một cách hiệu quả, đáng tin cậy để kết nối các thiết bị được sử dụng trong ngành dầu khí với các máy chủ doanh nghiệp từ xa nhằm điều khiển giám sát và thu thập dữ liệu từ các cảm biến.

❖ **Ưu điểm:**

- Chuyên thông tin hiệu quả hơn.
- Tăng khả năng mở rộng.
- Giảm đáng kể tiêu thụ băng thông mạng.
- Giảm tốc độ cập nhật xuống giây.
- Rất phù hợp cho điều khiển và đo thám.
- Tối đa hóa băng thông có sẵn.

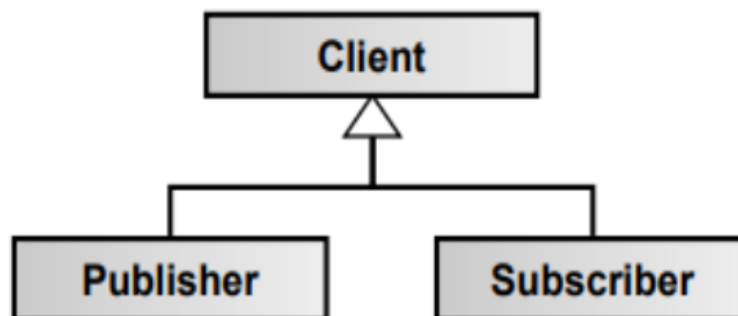
## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

---

- Chi phí cực nhẹ.
- Rất an toàn với bảo mật dựa trên sự cho phép.
- Được sử dụng bởi ngành công nghiệp dầu khí, Amazon, Facebook và các doanh nghiệp lớn khác.
- Tiết kiệm thời gian phát triển.
- Giao thức publish/subscribe thu thập nhiều dữ liệu hơn với ít băng thông hơn so với giao thức cũ.

### ❖ Các khái niệm đáng chú ý:

- Message: Trong giao thức MQTT, message còn được gọi là "message payload", có định dạng mặc định là plain-text (chữ viết người đọc được), tuy nhiên người sử dụng có thể cấu hình thành các định dạng khác.
- Client (Publisher, Subscriber): Client (khách hàng) đăng ký một chủ đề để gửi và nhận message:
  - Khi một client muốn gửi dữ liệu cho Broker: đây là hoạt động Publisher.
  - Khi một client muốn nhận dữ liệu từ Broker: đây là hoạt động Subscriber. Vì vậy Publisher và Subscriber đóng vai trò đặc biệt của client.



*Hình 2. 12: Hoạt động của Client*

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

---

- Server (Broker): Server trong MQTT gọi là một Broker được xem là trung tâm, là điểm giao của các kết nối đến từ client. Nhiệm vụ chính của Broker là nhận đăng ký từ các client về các chủ đề (topic), nhận tin nhắn (message), sắp xếp các message theo hàng đợi rồi chuyển chúng đến một địa chỉ dựa trên việc đăng ký của client. Nhiệm vụ phụ của Broker là có thể đảm nhận thêm một vài tính năng liên quan đến truyền thông như: bảo mật message, lưu trữ message.
- Topic: Topic có thể coi như một "đường truyền" logic giữa 2 điểm là publisher và subscriber. Về cơ bản, khi message được publish vào một topic thì tất cả những subscriber của topic đó sẽ nhận được message này. Giao thức MQTT cho phép khai báo các topic kiểu phân cấp.



*Hình 2. 13: Hoạt động của topic*

- Publish, subscribe: Trong một hệ thống sử dụng giao thức MQTT, nhiều node trạm (gọi là mqtt client - gọi tắt là client) kết nối tới một MQTT server (gọi là broker). Mỗi client sẽ đăng ký một vài kênh (topic), ví dụ như "/client1/channel1", "/client1/channel2". Quá trình đăng ký này gọi là "subscribe", giống như chúng ta đăng ký nhận tin trên một kênh Youtube vậy. Mỗi client sẽ nhận được dữ liệu khi bất kỳ trạm nào khác gửi dữ liệu và kênh đã đăng ký. Khi một client gửi dữ liệu tới kênh đó, gọi là "publish".
- QoS: Ở đây có 3 tùy chọn QoS (Qualities of service) khi "publish" và "subscribe". Một gói tin có thể được gửi ở bất kỳ QoS nào, và các client cũng có thể subscribe với bất kỳ yêu cầu QoS nào. Có nghĩa là client sẽ lựa chọn QoS tối đa mà nó có để nhận tin. Ví dụ, nếu 1 gói dữ liệu được publish với QoS2, và client subscribe với QoS0, thì gói dữ liệu được nhận về client này sẽ

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

---

được broker gửi với QoS0, và 1 client khác đăng ký cùng kênh này với QoS 2, thì nó sẽ được Broker gửi dữ liệu với QoS2.

- **QoS0** Broker/client sẽ gửi dữ liệu đúng 1 lần, quá trình gửi được xác nhận bởi chỉ giao thức TCP/IP, giống kiểu đem con bỏ chợ.
  - **QoS1** Broker/client sẽ gửi dữ liệu với ít nhất 1 lần xác nhận từ đầu kia, nghĩa là có thể có nhiều hơn 1 lần xác nhận đã nhận được dữ liệu.
  - **QoS2** Broker/client đảm bảo khi gửi dữ liệu thì phía nhận chỉ nhận được đúng 1 lần, quá trình này phải trải qua 4 bước bắt tay.
- **Retain:** Retain là một cờ (flag) được gắn cho một message của giao thức MQTT. Retain chỉ nhận giá trị 0 hoặc 1 (tương ứng 2 giá trị logic false hoặc true). Nếu retain = 1, broker sẽ lưu lại message cuối cùng của 1 topic kèm theo mức QoS tương ứng. Khi client bắt đầu subscribe topic có message được lưu lại đó, client ngay lập tức nhận được message.
- ❖ **Bảo mật:** MQTT được thiết kế một cách nhẹ và linh hoạt nhất có thể. Do đó nó chỉ có 1 lớp bảo mật ở tầng ứng dụng: bảo mật bằng xác thực (xác thực các client được quyền truy cập tới broker). Tuy vậy, MQTT vẫn có thể được cài đặt kết hợp với các giải pháp bảo mật đa tầng khác như kết hợp với VPN ở tầng mạng hoặc SSL/TLS ở tầng transport.
- ❖ **Ứng dụng:**
    - Facebook Messenger: Facebook đã sử dụng các khía cạnh của MQTT trong Facebook Messenger để trò chuyện trực tuyến. Tuy nhiên, không rõ MQTT được sử dụng bao nhiêu hoặc để làm gì.
    - Amazon Web Services đã công bố Amazon IoT dựa trên MQTT vào năm 2015.
    - Các tổ chức không gian địa lý SensorThings API đặc điểm kỹ thuật tiêu chuẩn có một phần mở rộng MQTT trong tiêu chuẩn như một giao thức thông báo bổ

sung ràng buộc. Nó đã được chứng minh trong một thí điểm IoT của Bộ An ninh Nội địa Hoa Kỳ.

- Các dịch vụ của Cơ sở hạ tầng thượng nguồn OpenStack được kết nối bằng một bus tin nhắn hợp nhất MQTT với Mosquitto là broker MQTT.
- Adafruit đưa ra một MQTT miễn phí dịch vụ đám mây cho thí nghiệm IOT và người học gọi Adafruit IO trong năm 2015.
- Microsoft Azure IoT Hub sử dụng MQTT làm giao thức chính cho các tin nhắn từ xa.
- XIM, Inc. đã ra mắt ứng dụng khách MQTT có tên MQTT Buddy vào năm 2017. Đây là ứng dụng MQTT dành cho Android và iOS, nhưng không phải là F-Droid, người dùng có sẵn bằng tiếng Anh, tiếng Nga và tiếng Trung Quốc.
- Node-RED hỗ trợ các nút MQTT kể từ phiên bản 0.14, để định cấu hình đúng các kết nối TLS.
- Nền tảng tự động hóa phần mềm nguồn mở Home Assistant được bật MQTT và cung cấp bốn tùy chọn cho các broker MQTT.

### 2.7 MODULE CẢM BIẾN RADAR RCWL-0516

Module Cảm Biến Radar RCWL-0516 dùng để phát hiện vật cản bằng sóng radar với khoảng cách phát hiện từ 5m đến 9m, sóng có thể xuyên một số vật cản bằng nhựa, gỗ mỏng... nên có thể đặt module trong hộp để bảo vệ. Cảm biến có độ nhạy và độ bền cao.

#### ❖ Thông số kỹ thuật:

- IC: RCWL-9196.
- Điện áp hoạt động: 4 đến 28VDC.
- Dòng tiêu thụ: 2.8mA, 3mA (max).
- Khoảng cách hoạt động: 5 - 9m.



## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

---

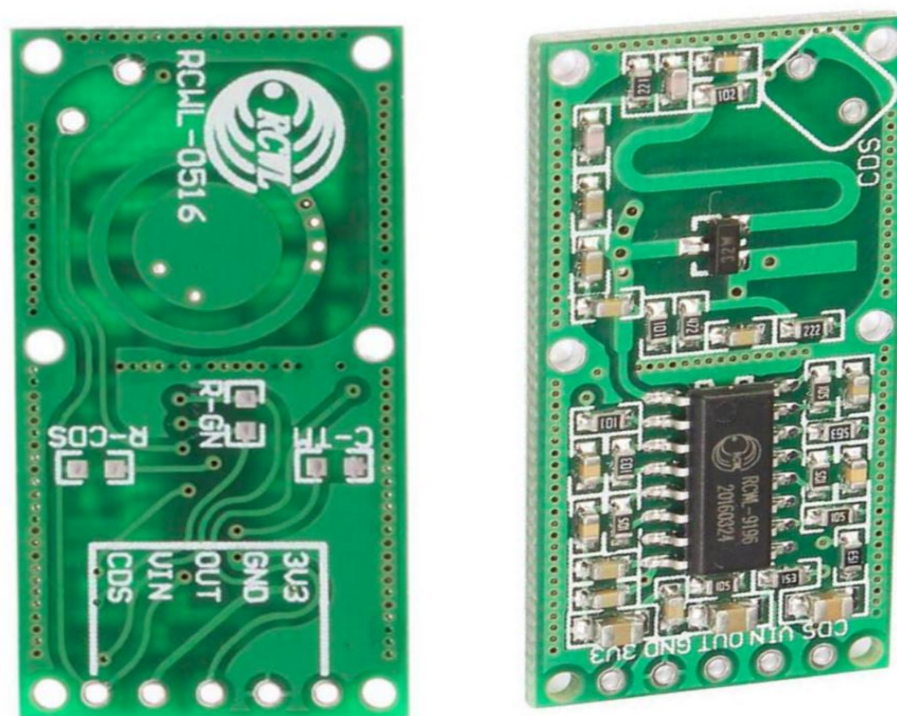
- Năng lượng truyền: 20mW (chuẩn); 30mW (max).
- Điện áp ra: 3.2 - 3.4V.
- Dòng ra: 100mA.
- Nhiệt độ hoạt động: -20 đến 80°C.

### ❖ Hướng dẫn nối chân:

- 3.3V: ngõ ra 3.3V, I<sub>max</sub>: 100mA.
- GND: Chân Mass, 0V.
- Vin: Điện áp cung cấp (4 - 28V).
- OUT: Điều khiển tín hiệu ngõ ra. Khi phát hiện đối tượng -> Ngõ ra mức cao (3.3V). Ngược lại -> Ngõ ra mức thấp (0V).
- CDS: Điều khiển tín hiệu kích hoạt (nhỏ hơn 0.7V - ngõ ra OUT thấp, lớn hơn 0.7V - hoạt động bình thường). Kết nối LDR có thể tắt chức năng phát hiện vào ban đêm.

### ❖ Chức năng điều chỉnh:

- C<sub>TM</sub>: Đây là vị trí gắn tụ điện để điều chỉnh thời gian kích, mặc định nếu không có tụ điện thì thời gian kích là 2s, giá trị của tụ điện càng lớn thì thời gian kích càng lâu.
- R<sub>GN</sub>: Đây là vị trí gắn điện trở chỉnh khoảng cách phát hiện vật cản, nếu không có điện trở thì khoảng cách mặc định là 7m, nếu nối điện trở 1MΩ thì khoảng cách phát hiện vật cản giảm xuống còn 5m, giá trị điện trở càng nhỏ thì khoảng cách phát hiện càng gần.
- R<sub>CDS</sub>: Đây là vị trí gắn quang trở, nếu gắn quang trở vào vị trí này thì cảm biến sẽ không hoạt động vào buổi tối.



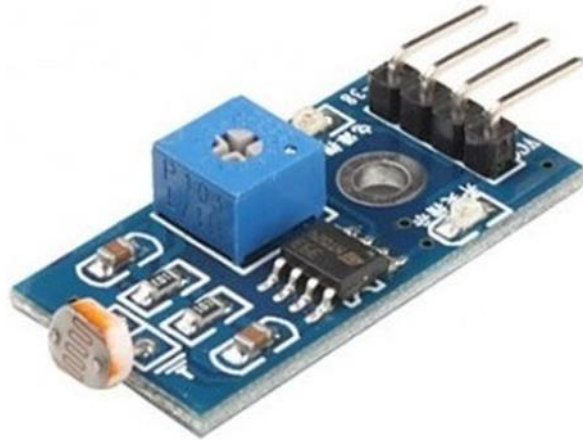
*Hình 2. 14: Module Cảm Biến Radar RCWL-0516*

### 2.8 CẢM BIẾN ÁNH SÁNG

Cảm biến ánh sáng sử dụng quang trở có khả năng thay đổi điện trở theo cường độ ánh sáng chiếu vào. Tín hiệu xuất ra của cảm biến là digital HIGH (5V) và LOW (0V) tương trưng cho các trạng thái bật, tắt thiết bị điện tự động mà bạn không cần phải thao tác vào.

Cảm biến quang trở CDS có 4 chân là: DO, VCC, GND, AO.

Ngõ ra DO trên cảm biến được dùng để xác định cường độ sáng của môi trường, khi ở ngoài sáng ngõ ra DO là giá trị 0, khi ở trong tối ngõ ra DO là 1. Trên cảm biến có 1 biến trở để điều chỉnh cường độ sáng phát hiện, khi vặn cùng chiều kim đồng hồ thì sẽ làm giảm cường độ sáng nhận biết của cảm biến, tức là môi trường phải ít sáng hơn nữa thì cảm biến mới đọc giá trị digital là 1.



*Hình 2. 15: Cảm biến ánh sáng*

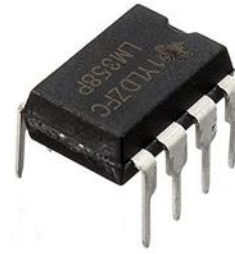
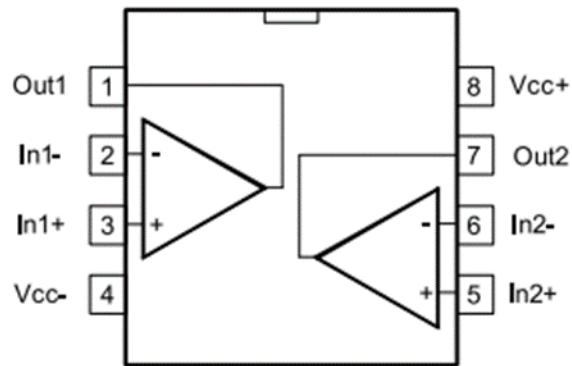
❖ **Thông số kỹ thuật:**

- Điện áp đầu vào: 3.3V - 5V.
- Đầu ra: có đầu ra số và đầu ra tương tự tương ứng D0 và A0.
- Có chiết áp điều chỉnh cường độ sáng.
- Kích thước: 3.2cm x 1.4cm.
- Tải đầu ra số D0: 15mA.

### 2.9 IC LM358

LM358 là bộ khuếch đại thuật toán kép công suất thấp, bộ khuếch đại này có ưu điểm hơn so với bộ khuếch đại thuật toán chuẩn trong các ứng dụng dùng nguồn đơn.

IC LM358 có thể hoạt động ở nguồn điện áp thấp 3V hoặc cao lên tới 32V. Có công suất cực mánh thấp, tuy nhiên có độ lợi cao 100dB. Cấu tạo bên trong gồm 2 bộ khuếch đại thuật toán, tương thích với nhiều loại mạch logic khác nhau.



*Hình 2. 16: Sơ đồ chân IC LM358*

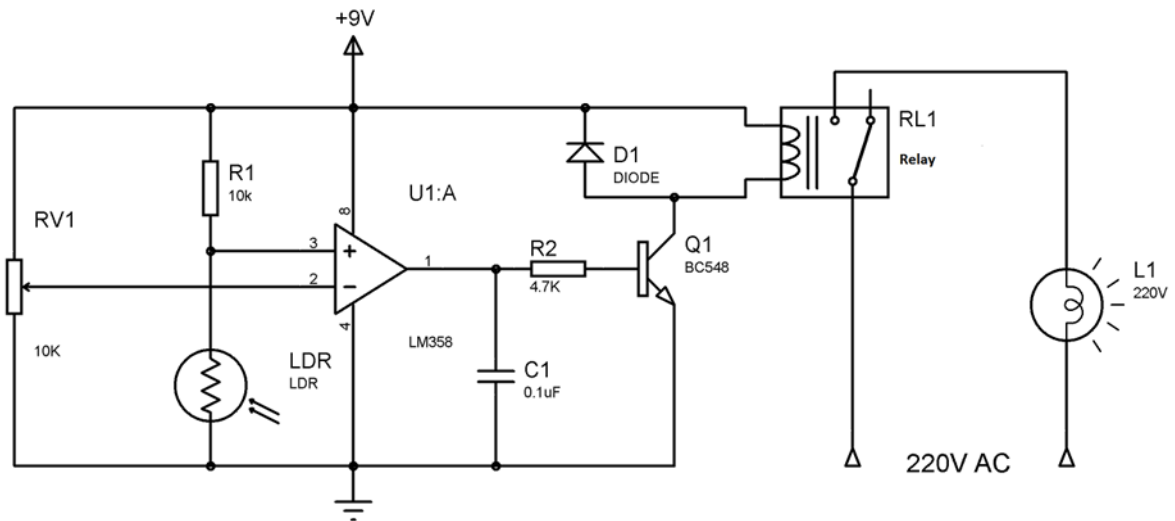
❖ **Thông số kỹ thuật:**

- Số chân: 14 chân.
- Điện áp: 3-32V với nguồn đơn, 1.5-16V với nguồn đôi.
- Dải nhiệt độ hoạt động: 0 đến 70 độ C.
- Độ lợi khuếch đại DC 100dB.
- Điện áp ngõ ra: 0V đến VCC(+) -1.5V.

❖ **Các tính năng của khuếch đại thuật toán:**

- Bảo vệ quá áp lỗi ra.
- Tầng khuếch đại vi sai lỗi vào.
- Dòng cung cấp lỗi vào thấp.
- Dải tín hiệu cùng pha mở rộng tới nguồn âm.

❖ **Ứng dụng của IC LM358:** Mạch chiếu sáng tự động.



Hình 2. 17: Sơ đồ nguyên lý mạch chiếu sáng tự động dùng IC LM358

### ➤ Nguyên lý hoạt động:

Khi điện áp thay đổi trên 2 đầu vào (2) và (3), O-A nhận biết 2 tín hiệu này và thực hiện phép so sánh.

- $V_2, V_3$  là điện thế của các điểm này: Nếu  $V_2 > V_3$  thì  $V_{out} = V_4$  ( $V_{out}$  chính là  $V_1$ ).
- Ngược lại: Nếu  $V_2 < V_3 \Rightarrow V_{out} = V_8$ .
- Lúc này, tín hiệu đưa ra Transistor chắc chắn là 1 trong 2 giá trị:  $V_8$  hoặc  $V_4$
- $V_{out} = V_4 = 0 \Rightarrow$  Transistor không dẫn  $\Rightarrow$  Relay ở trạng thái mở  $\Rightarrow$  đèn tắt.
- $V_{out} = V_8 \neq 0 \Rightarrow$  Transistor dẫn  $\Rightarrow$  Relay ở trạng thái đóng  $\Rightarrow$  đèn mở.

Như vậy, việc thay đổi điện trở thụ động của quang trở phụ thuộc vào ánh sáng chiếu vào nó. Quang trở thay đổi điện trở làm điện áp thay đổi liên tục và Op-Amp sử dụng các tín hiệu điện áp này để điều khiển điện áp ra... Từ đó quyết định việc đóng mở của Transistor và relay...

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

### 2.10 MOC3021

MOC3021 là một loại Opto - Triac là linh kiện được sử dụng để cách ly giữa các khối chệnh lệch nhau về điện hay công suất.

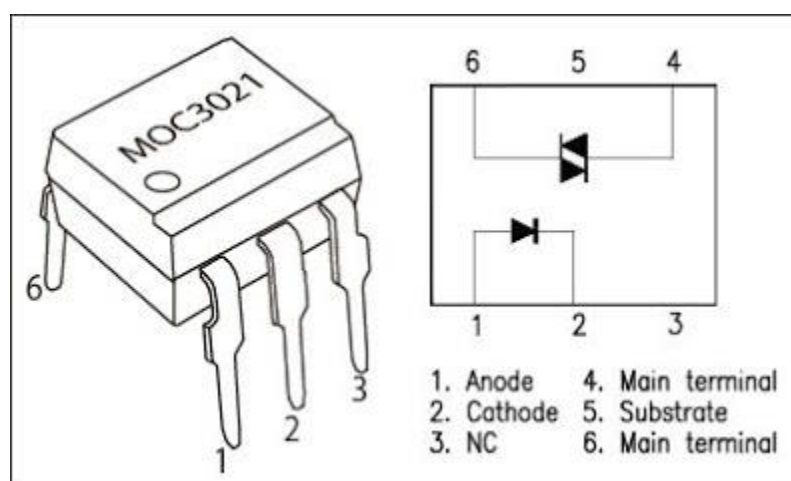
Cách ly điều khiển giữa hai tầng mạch điện khác nhau.

Ngăn các xung điện áp cao hay các phần tử công suất lớn có thể làm hư hỏng các ngõ điều khiển công suất nhỏ trên board mạch.

Với giá thành rẻ, hữu ích nên rất được ưa dùng trong mạch điện tử.

#### ❖ Thông số kỹ thuật:

- Điện áp trạng thái tắt: 400V.
- Tốc độ dữ liệu: 10Mbps.
- Điện áp cách ly: 5300Vrms.
- Nguồn dòng ngõ kích hoạt: 15mA.
- Dòng điện chuyển tiếp DC: 60mA.
- Nhiệt độ ổn định: -40 đến 85 độ C.



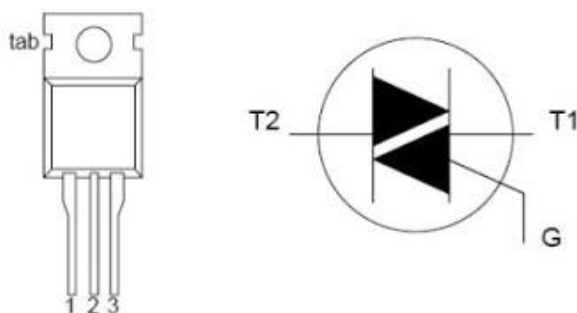
Hình 2. 18: Sơ đồ chân Moc3021

### 2.11 TRIAC BT137

Triac BT137 là một linh kiện bán dẫn có ba cực năm lớp, làm việc như 2 Thyristor mắc song song ngược chiều, có thể dẫn điện theo hai chiều.

❖ **Thông số kỹ thuật:**

- Điện áp cực đại: 600V.
- Dòng điện thuận cực đại: 8A.
- Điện áp điều khiển kích dẫn: 1.5V.
- Dòng điện điều khiển kích dẫn: 100mA.
- Nhiệt độ làm việc: -40 đến 125 độ C.



PIN	DESCRIPTION
1	main terminal 1
2	main terminal 2
3	gate
tab	main terminal 2

*Hình 2. 19: Sơ đồ chân Triac BT317*

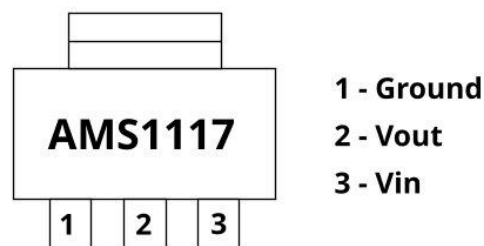
### 2.12 IC ỔN ÁP ASM1117

Tạo nguồn 3.3V, kích thước mạch nhỏ gọn, sử dụng đơn giản.

Cung cấp điện áp cho các mạch sử dụng nguồn 3.3V.

❖ **Thông số kỹ thuật:**

- Điện áp vào: Từ 1.5VDC đến 12VDC.
- Điện áp ra: 3.3V.
- Dòng ra tối đa: 1A.
- Nhiệt độ hoạt động: -40 đến 125 độ C.



*Hình 2. 20: IC ổn áp ASM1117*



### 2.13 BIẾN TRỞ

Biến trở là các thiết bị có điện trở thuần có thể biến đổi được theo ý muốn. Chúng có thể được sử dụng trong các mạch điện để điều chỉnh hoạt động của mạch điện.

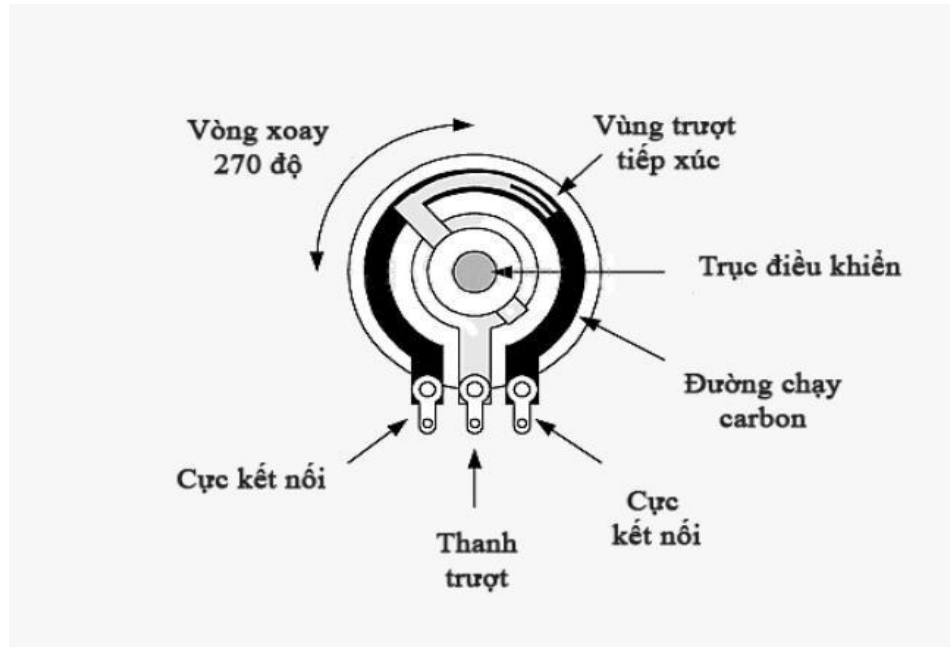


*Hình 2. 21: Các ký hiệu của biến trở*

#### ❖ Cấu tạo:

Biến trở có cấu tạo gồm ba bộ phận chính:

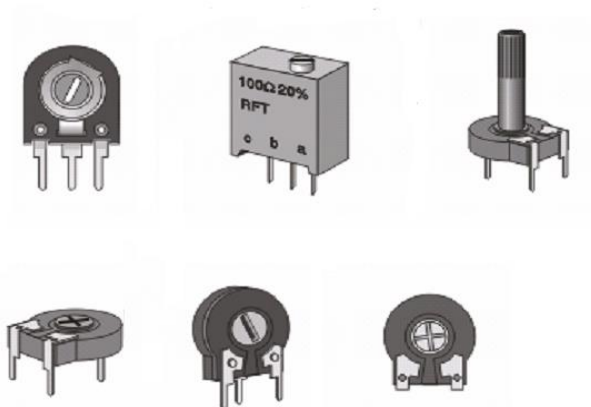
- Cuộn dây được làm bằng hợp kim có điện trở suất lớn.
- Con chạy/Chân chạy: cho khả năng chạy dọc cuộn dây để làm thay đổi giá trị trở kháng.
- Chân ngõ ra gồm có 3 chân (3 cực): Trong số ba cực này, có hai cực được cố định ở đầu của điện trở. Các cực này được làm bằng kim loại. Cực còn lại là một cực di chuyển và thường được gọi là cần gạt. Vị trí của cần gạt này trên dải điện trở sẽ quyết định giá trị của biến trở.



*Hình 2. 22: Cấu tạo của biến trở*

Biến trở được phân loại thành:

- Biến trở tay quay.
- Biến trở con chạy.
- Biến trở than.
- Biến trở dây quấn.



*Hình 2. 23: Các loại biến trở*

# CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

## 3.1 GIỚI THIỆU

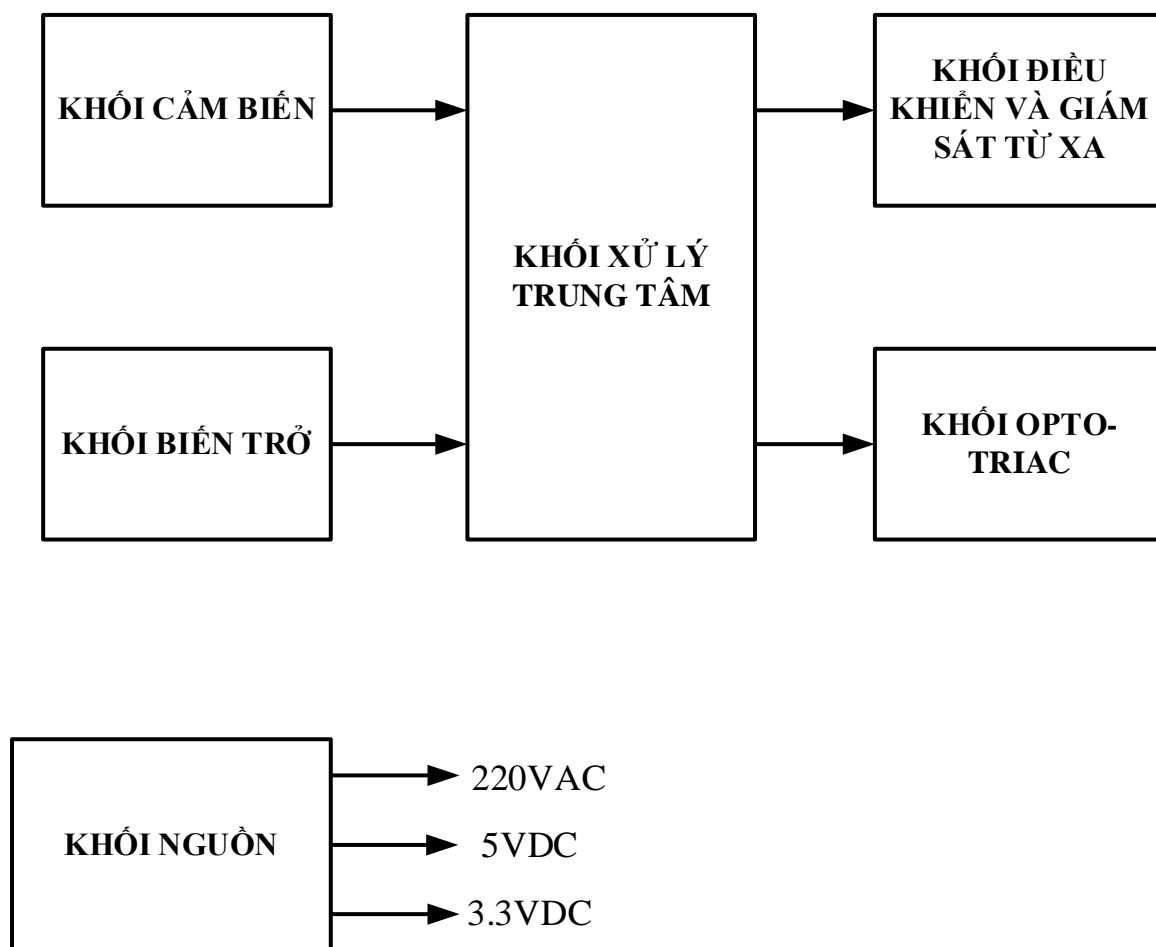
Đối với đề tài “*Nghiên cứu, cải tiến và thi công bộ đèn led tích hợp cảm biến quang, chuyển động và wifi*” yêu cầu đặt ra của đề tài gồm những yêu cầu sau:

- Thiết kế, tính toán mạch giao tiếp với các cảm biến để điều khiển đèn.
- Thiết kế, tính toán dòng áp cung cấp cho các thiết bị của hệ thống hoạt động đúng yêu cầu.
- Tìm hiểu sơ đồ chức năng chân của ESP8266 kết nối với các module.

## 3.2 TÍNH TOÁN VÀ THIẾT KẾ HỆ THỐNG

### 3.2.1 Thiết kế sơ đồ khối hệ thống

Với mỗi đề tài để biết được các thành phần, cấu tạo cũng như hoạt động của chúng để tạo nên một sản phẩm hoàn thiện chúng ta cần khảo sát, tìm hiểu theo từng bước. Sơ đồ khối giúp ta có cái nhìn tổng quan hơn về đề tài. Từ đó hình dung được những công việc cần làm và phân chia thời gian cụ thể, hợp lý trong quá trình thực hiện. Chính vì vậy, để làm được đề tài “*Nghiên cứu, cải tiến và thi công bộ đèn led tích hợp cảm biến quang, chuyển động và wifi*” em sẽ khảo sát từng vấn đề theo sơ đồ khối ở các mục dưới đây. Theo yêu cầu đặt ra em thiết kế sơ đồ khối như hình 3-1.



*Hình 3. 1: Sơ đồ khối hệ thống*

### 3.2.2 Chức năng từng khối

**Khối nguồn:** Đây là khối không thể thiếu trong các mạch điện tử, có khối này mới cung cấp điện áp và dòng điện cho mạch hoạt động.

**Khối cảm biến:** Bao gồm cảm biến ánh sáng và cảm biến chuyển động:

- Cảm biến ánh sáng: Kiểm tra môi trường có ánh sáng hay không có ánh sáng.
- Cảm biến chuyển động: Kiểm tra và phát hiện các chuyển động có trong khoảng cách hoạt động.

**Khối xử lý trung tâm:** Là khối điều khiển trung tâm cho toàn hệ thống, có nhiệm vụ kết nối với các thành phần trong hệ thống. Nhận dữ liệu từ các cảm biến, biến trở và

## CHƯƠNG 3: TÍNH TOÁN VÀ THIẾT KẾ

---

gửi lên cơ sở dữ liệu để lưu trữ, giám sát thông qua điện thoại, nhận dữ liệu từ khối điều khiển và giám sát để điều khiển và giám sát thiết bị.

**Khối điều khiển và giám sát từ xa:** Truyền nhận dữ liệu 2 chiều giữa khối xử lý trung tâm và cơ sở dữ liệu thông qua giao thức MQTT. Điều khiển và giám sát thiết bị từ xa thông qua wifi và sử dụng app trên điện thoại.

**Khối biến trở:** Có chức năng điều chỉnh độ nhạy sáng của quang trở.

**Khối OPTO-TRIAC:** Khối này nhận lệnh từ khối xử lý trung tâm để điều khiển đóng, cắt và giảm độ sáng của đèn.

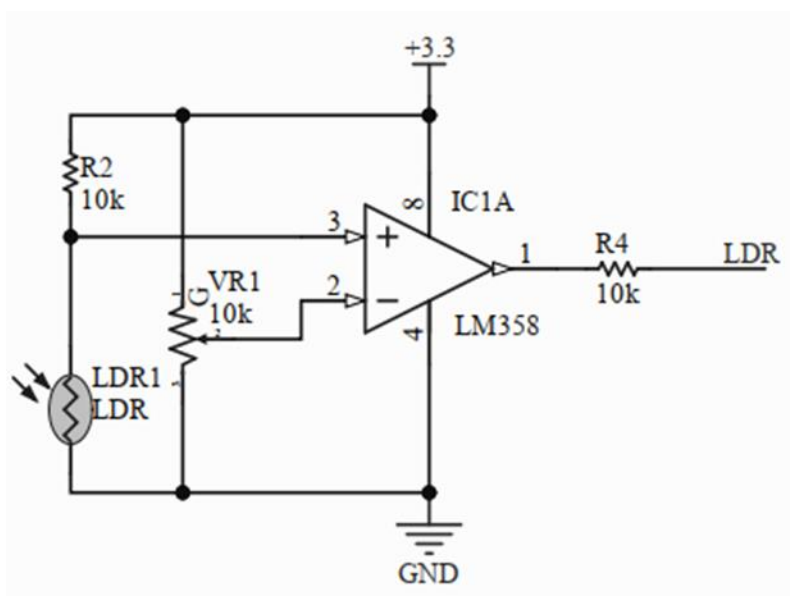
### 3.3 TÍNH TOÁN VÀ THIẾT KẾ MẠCH

#### 3.3.1 Khối cảm biến

**Cảm biến quang trở:**

Mạch cảm biến ánh sáng dùng quang trở có ưu điểm: tương đối nhỏ gọn, độ chính xác cao, các thành phần phụ của mạch thương được gắn đầy đủ và nếu không may bị hỏng hóc thì rất dễ thay thế.

Ngoài ra mạch cảm biến ánh sáng dùng quang trở còn có thể điều chỉnh được độ nhạy của mạch chỉ bằng một biến trở. Có thể tăng độ nhạy của mạch cảm biến với ánh sáng bằng cách vặn về bên tay trái làm cho chỉ cần xuất hiện ánh sáng có cường độ nhỏ thì đã ngắt mạch. Còn nếu vặn về bên tay phải thì cần lượng ánh sáng với cường độ mạnh hơn mới có thể khiến cho mạch bị tắt vì đã làm giảm độ nhạy sáng của mạch.



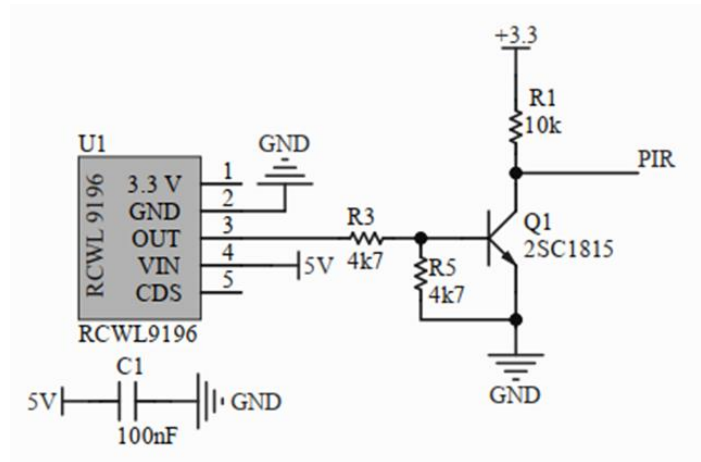
**Hình 3. 2:** Sơ đồ mạch cảm biến quang trở

Khi điện áp thay đổi trên 2 đầu vào (2) và (3), O-A nhận biết 2 tín hiệu này và thực hiện phép so sánh.

- $V_2, V_3$  là điện thế của các điểm này: Nếu  $V_2 > V_3$  thì  $V_{out} = V_4$  ( $V_{out}$  chính là  $V_1$ ).
- Ngược lại: Nếu  $V_2 < V_3 \Rightarrow V_{out} = V_8$ .
- Lúc này, tín hiệu đưa ra chân LDR chắc chắn là 1 trong 2 giá trị:  $V_8$  hoặc  $V_4$
- $V_{out} = V_4 = 0$ .
- $V_{out} = V_8 \neq 0$ .

Vì ngõ ra của quang trở ở dạng digital (tín hiệu số) nên có thể nối ngõ ra LDR của mạch cảm biến ánh sáng trực tiếp vào chân số 5 (GPIO14) của module wifi ESP07 để đọc giá trị trả về.

### Cảm biến chuyển động:



Hình 3. 3: Sơ đồ mạch cảm biến chuyển động radar

Tính toán chọn điện trở:

Transistor ở trạng thái dẫn bão hòa:  $V_{CC} - I_B R_B - V_{BE} = 0$

$$\Rightarrow 5 - I_B R_B - 0,7 = 0 \Rightarrow I_B R_B = 4,3$$

$$\text{Mà } I_C = \beta I_B = 70 \text{ mA (Chọn } \beta = 100) \Rightarrow I_B = \frac{70}{100} = 0,7 \text{ mA}$$

$$U_B = 3,3 \text{ (V)} \Rightarrow R_B = \frac{U_B}{I_B} = \frac{3,3}{0,7} = 4,7 \text{ k}\Omega$$

Chọn điện trở  $R_3 = R_5 = 4,7 \text{ k}\Omega$

#### ❖ Thông số kĩ thuật:

- IC: RCWL-9196.
- Điện áp hoạt động: 4 đến 28VDC.
- Dòng tiêu thụ: 2.8mA, 3mA (max).
- Khoảng cách hoạt động: 5m - 9m.
- Năng lượng truyền: 20mW (chuẩn); 30mW (max).
- Điện áp ra: 3.2 - 3.4V.

- Dòng ra: 100mA.
- Nhiệt độ hoạt động: -20 đến 80°C.

### ❖ Nối chân:

- 3.3V: ngõ ra 3.3V, I<sub>max</sub>: 100mA.
- GND: Chân Mass, 0V.
- Vin: Điện áp cung cấp 12VDC.
- OUT (PIR): Điều khiển tín hiệu ngõ ra. Khi phát hiện đối tượng -> Ngõ ra mức cao (3.3V). Ngược lại -> Ngõ ra mức thấp (0V). Vì ngõ ra của cảm biến chuyển động ở dạng digital (tín hiệu số) nên có thể nối ngõ ra PIR của cảm biến trực tiếp vào chân số 6 (GPIO12) của module wifi ESP07 để đọc giá trị trả về.

### 3.3.2 Khối xử lý trung tâm

Hiện nay trên thị trường có rất nhiều dòng vi điều khiển khác nhau như PIC, AVR, 8051, Raspberry, Arduino... Tất cả đều có thể đáp ứng được yêu cầu đặt ra nhưng em chọn ESP8266 vì nó có những ưu điểm sau:

- Giá thành rẻ, dễ sử dụng, là module hoàn chỉnh sử dụng vi điều khiển.
- Có tích hợp module wifi.
- Kích thước nhỏ gọn.
- Là dòng vi điều khiển mã nguồn mở, có nhiều thư viện hỗ trợ cho các module chức năng khác nhau.
- Trình biên dịch đơn giản, dễ sử dụng.
- Tích hợp sẵn anten ceramic và anten ngoài, trong trường hợp bắt sóng yếu có thể dùng thêm anten ngoài.

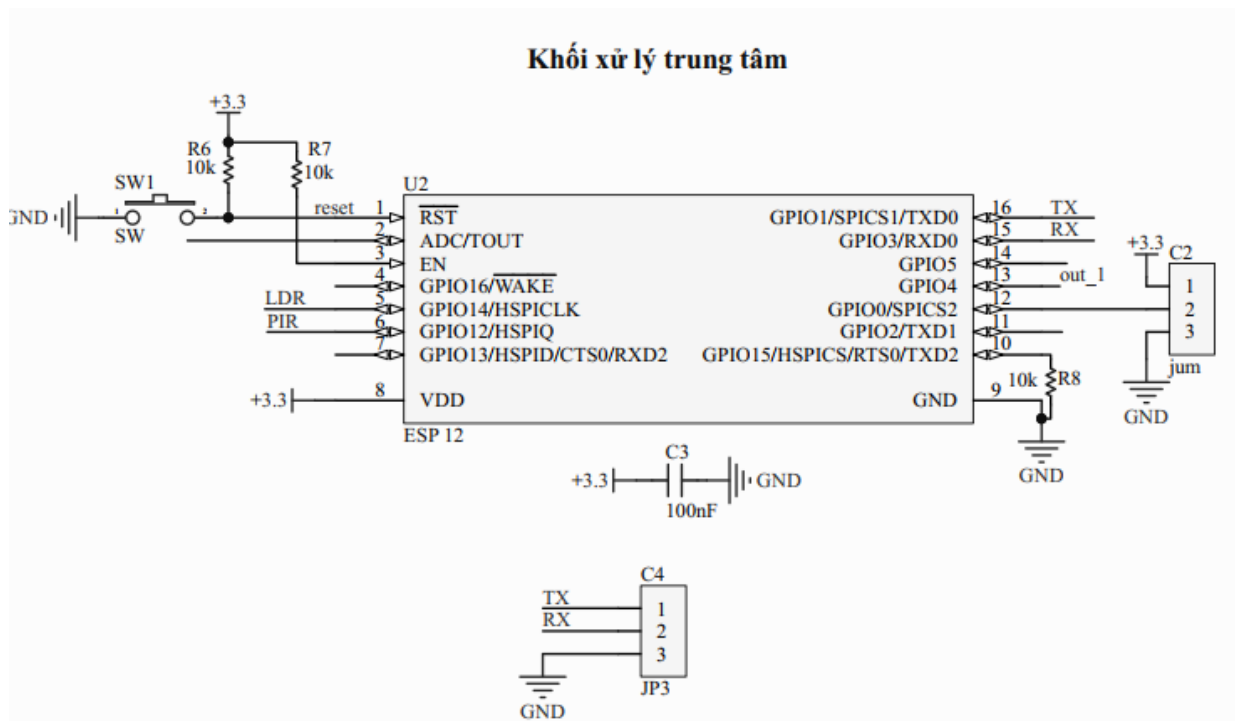


## CHƯƠNG 3: TÍNH TOÁN VÀ THIẾT KẾ



Hình 3. 4: Module wifi ESP07

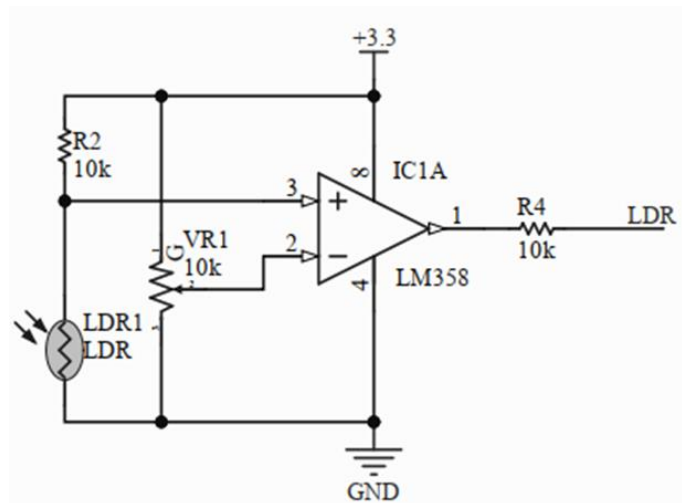
Với những ưu điểm được trình bày ở trên và ở phần 2.4 em quyết định chọn Module ESP07.



Hình 3. 5: Sơ đồ nối dây module wifi ESP07

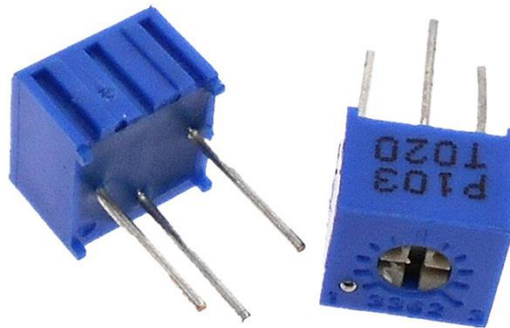
### 3.3.3 Khối biến trở

**Biến trở VR1:** Dùng để điều chỉnh độ nhạy ánh sáng của cảm biến quang trở.



*Hình 3. 6: Sơ đồ nối dây biến trở VR1*

Biến trở trong đề tài em sử dụng là biến trở vuông 10KΩ.



*Hình 3. 7: Biến trở vuông*

### 3.3.4 Khối OPTO-TRIAC

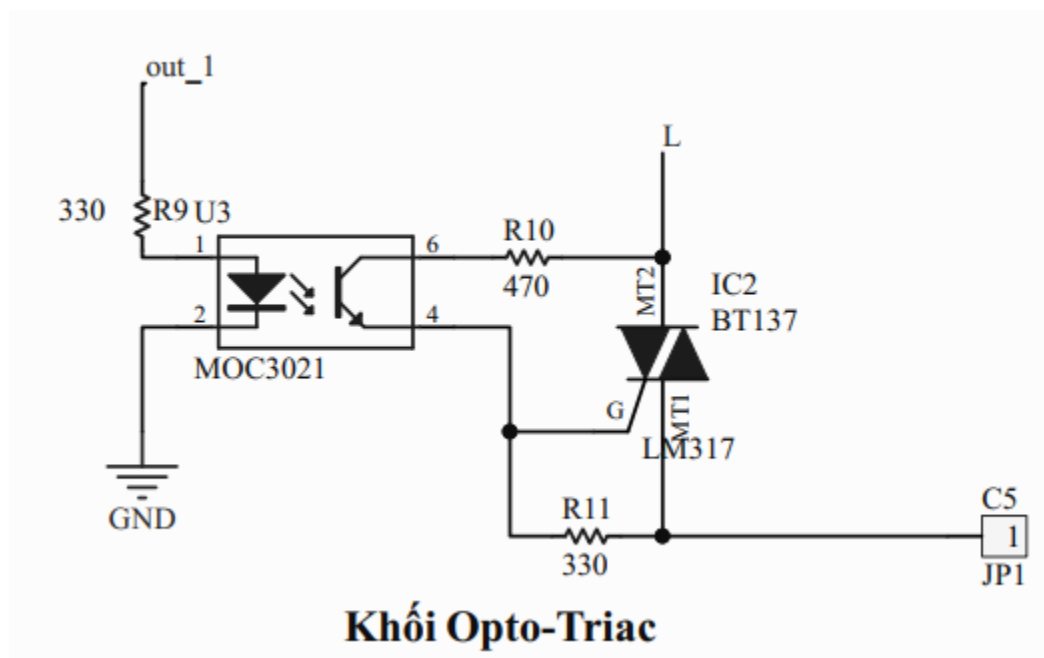
Ở đề tài này em chọn cách ly opto-triac thay cho cách ly relay vì relay chỉ là thiết bị đóng cắt ON/OFF bình thường, và kích thước lớn hơn so với opto-triac.

Ngoài ra cách ly opto-triac còn có những ưu điểm sau:

- Có thể sử dụng cho cả điện xoay chiều vào một chiều.
- Không tạo tiếng ồn.
- Không gây nhiễu sóng hài, không tạo ra hồ quang điện.

## CHƯƠNG 3: TÍNH TOÁN VÀ THIẾT KẾ

- An toàn điện do được cách li giữa khối điều khiển và khối công suất bằng Opto.
- Có thể sử dụng cho các ứng dụng yêu cầu việc đóng cắt nhanh.



*Hình 3. 8: Sơ đồ nối dây ngõ ra opto-triac*

### Cấu tạo và nguyên lý hoạt động:

Opto gồm một đèn LED hồng ngoại và một photodetector (chẳng hạn như một diode quang, transistor quang, Darlington, SCR hoặc triac), cả hai được tích hợp nằm bên trong một vỏ bọc kín.

Nó có một đèn LED ở phía đầu vào và một diode quang ở đầu ra. Điện áp nguồn bên trái và điện trở nối tiếp tạo dòng điện qua đèn LED. Sau đó, ánh sáng từ đèn LED bắt đầu đến diode quang và tạo nên một dòng trong mạch đầu ra. Điện áp đầu ra bằng với điện áp đầu ra trừ đi điện áp trên điện trở tải. Khi điện áp đầu vào thay đổi, lượng ánh sáng sẽ dao động.

## CHƯƠNG 3: TÍNH TOÁN VÀ THIẾT KẾ

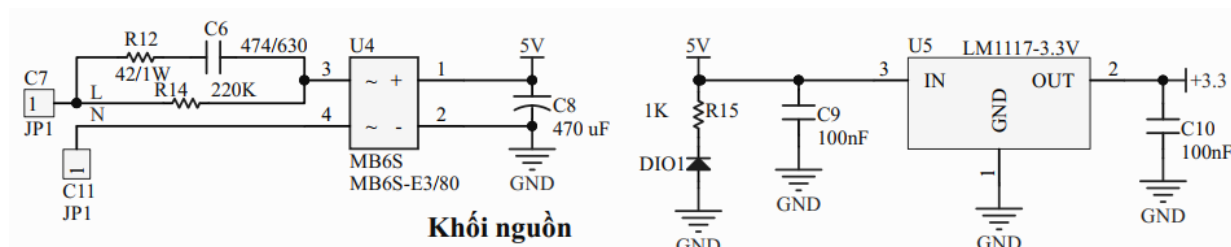
### 3.3.5 Khối điều khiển và giám sát từ xa

Nhận dữ liệu điều khiển từ khối xử lý trung tâm, tiến hành lưu trạng thái của thiết bị vào Server, để biết được lịch sử điều khiển thiết bị. Người dùng có thể tác động vào phần giao diện để điều khiển theo ý muốn. Đồng thời dữ liệu điều khiển được đồng bộ với ứng dụng trên điện thoại và hệ thống.

Xử lý và gửi tín hiệu điều khiển đến vi điều khiển, điều khiển đèn trực tiếp trên điện thoại.

### 3.3.6 Khối nguồn

Khối nguồn đóng vai trò quan trọng trong việc đảm bảo cho hệ thống sau khi hoàn thành có hoạt động ổn định hay không, đồng thời là khối quan trọng nhất cần phải được tính toán chính xác.



Hình 3. 9: Sơ đồ nguyên lý khối nguồn

Bảng 3. 1: Thông số, giá trị các linh kiện sử dụng

STT	Tên linh kiện	Số lượng	Điện áp (VDC)	Dòng điện (mA)
1	Module ESP8266	1	3.3	80
2	Cảm biến radar	1	5	3
3	Cảm biến ánh sáng	1	3.3	8.3
4	MOC 3021	1	3.3	15
5	BT137	1	3.3	100
Tổng dòng tiêu thụ của hệ thống				203.3

## CHƯƠNG 3: TÍNH TOÁN VÀ THIẾT KẾ

---

Với giá trị tính được từ bản trên tiến hành lựa chọn mạch nguồn AC-DC vào 220VAC ra 5VDC 700mA để cấp nguồn cho cảm biến radar, cảm biến ánh sáng.

Sau đó qua mạch giảm áp LM1117 hạ áp xuống còn 3.3V cung cấp cho các linh kiện khác hoạt động.



*Hình 3. 10: Mạch chuyển đổi AC-DC 220V ra 5V*

❖ **Thông số kỹ thuật:**

- Điện áp đầu vào: AC 85 đến 265V 50/60 hz hoặc DC 100 đến 370V.
- Dòng vào: 0.0273A (AC110V) 0.014A (AC220V).
- Dòng khởi động: 20A.
- Điện áp đầu ra: DC 5V (+/- 0.2 V).
- Dòng ra: 700 mA.
- Công suất: 3,5W.
- Nhiệt độ hoạt động: -20 đến 60°C.

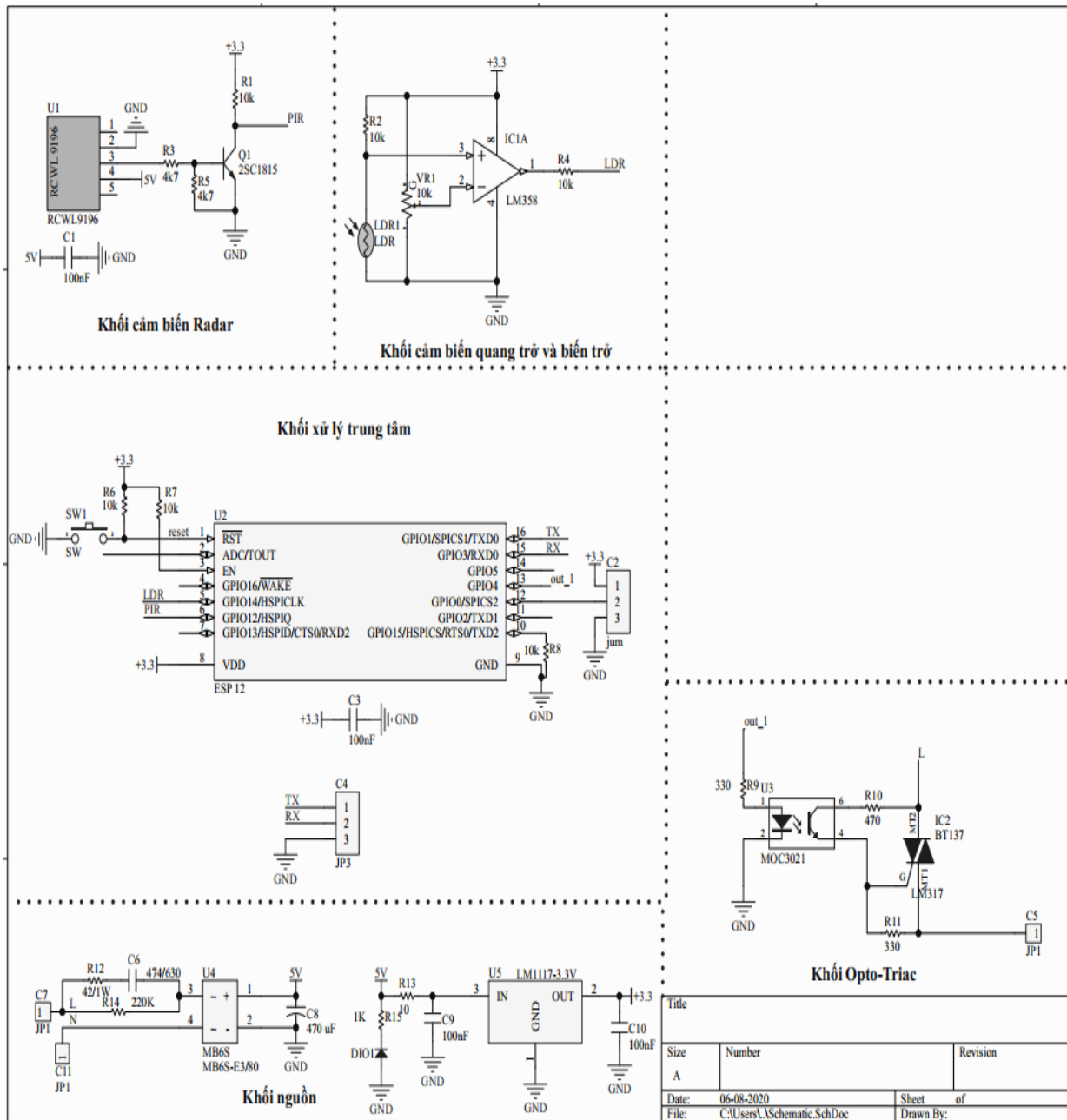
### **CHƯƠNG 3: TÍNH TOÁN VÀ THIẾT KẾ**

---

- Độ ẩm tương đối: 40-90% RH.
- Đầu ra 5V, dòng ra (min) 0A, dòng ra (max) 700mA, dòng đỉnh: 800mA, phạm vi điện áp ra: 4.8 đến 5.2V.
- Công suất đầu ra: 0 đến 4W (dòng điện DC).
- Hiệu suất đầu ra: 80%.
- Bảo vệ quá dòng và quá áp: Có.
- Kích thước: 3 x 2 x 1,8cm.

# CHƯƠNG 3: TÍNH TOÁN VÀ THIẾT KẾ

## 3.4 Sơ đồ nguyên lý toàn mạch



Hình 3. 11: Sơ đồ nguyên lý toàn mạch

## **CHƯƠNG 4: THI CÔNG HỆ THỐNG**

### **4.1 GIỚI THIỆU**

Mô hình bao gồm 2 phần thi công: phần cứng và phần mềm.

#### **Phần cứng:**

Bao gồm 1 mạch PCB chung cho toàn bộ hệ thống với kích thước 520mm x 550mm. Mạch được in hai lớp (Top và Bottom).

Sử dụng hộp đèn có sẵn trên thị trường.

#### **Phần mềm:**

Lập trình cho ESP8266, điều khiển tắt mở đèn và truyền nhận dữ liệu từ Server.

Sử dụng app Blynk điều khiển đèn.

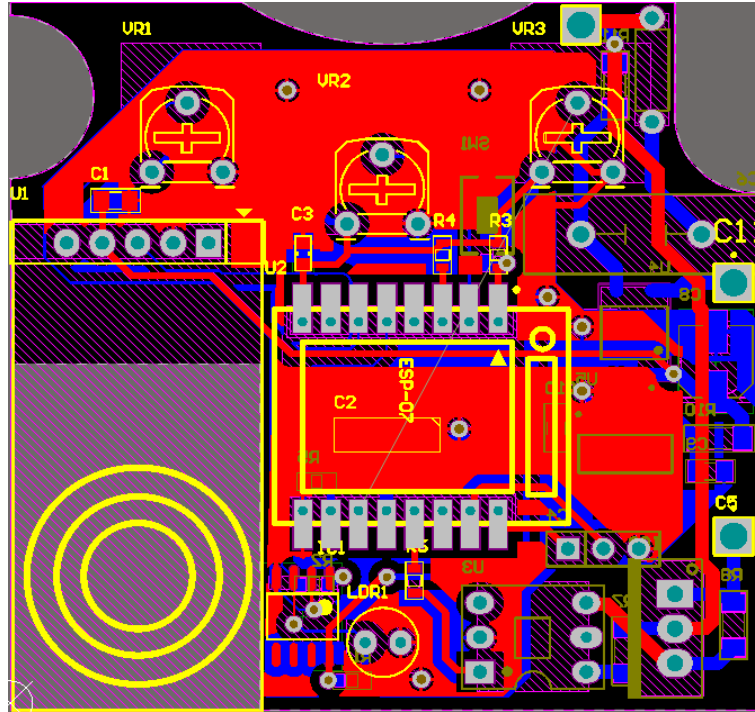
### **4.2 THI CÔNG HỆ THỐNG**

#### **4.2.1 Thi công board mạch**

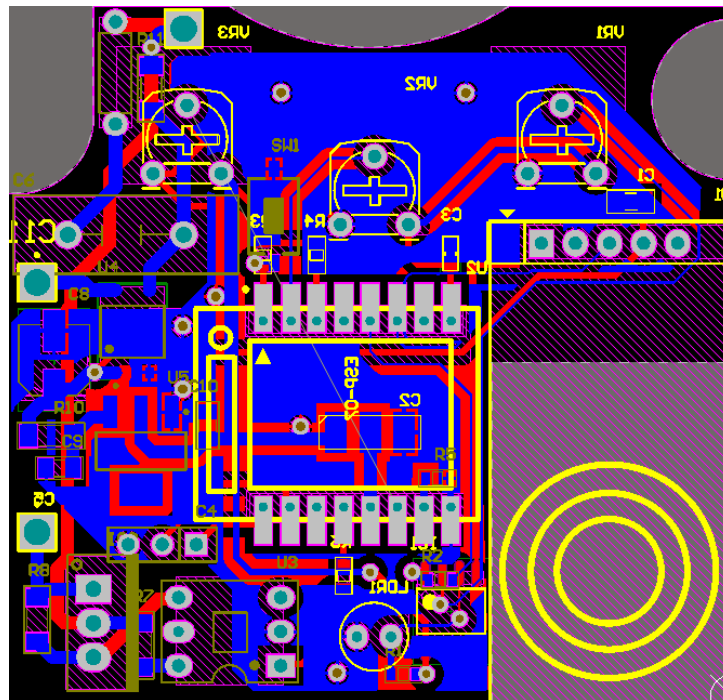
Các module, cảm biến được tích hợp trên 1 board mạch duy nhất.

Board mạch được thiết kế bằng phần mềm Altium.

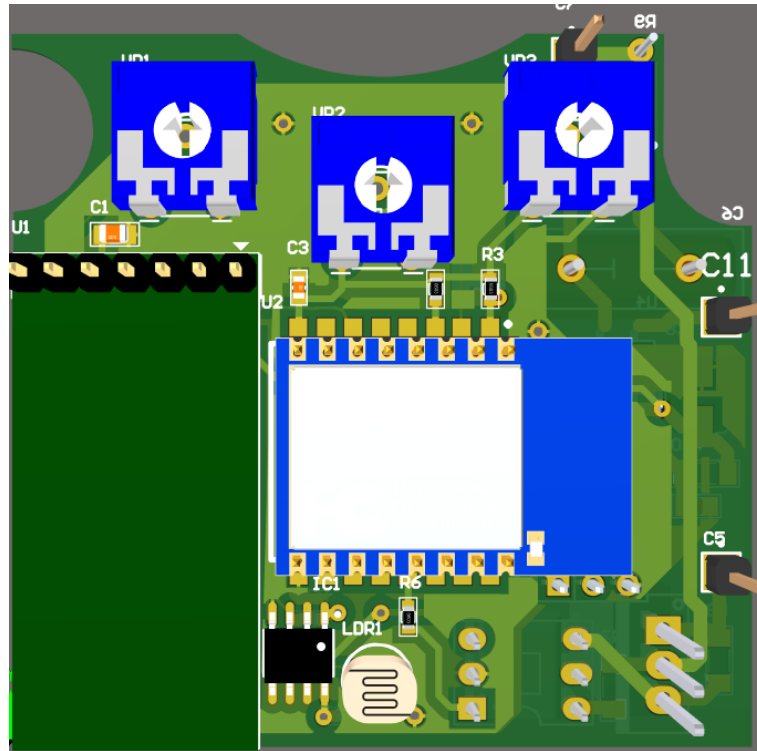




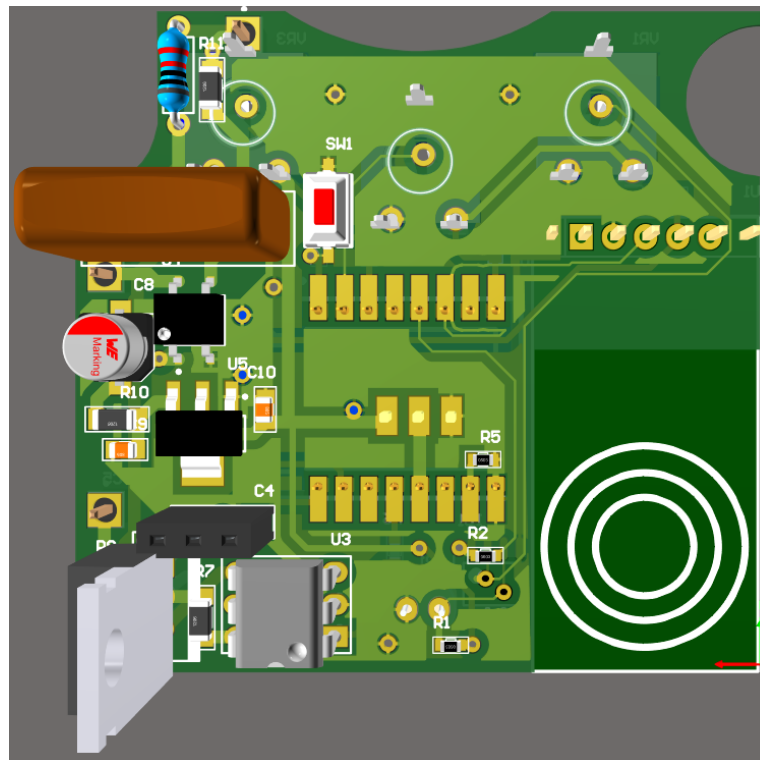
Hình 4. 1: PCB mặt trên



Hình 4. 2: PCB mặt dưới



*Hình 4. 3: Sơ đồ bố trí linh kiện mặt trên*



*Hình 4. 4: Sơ đồ bố trí linh kiện mặt dưới*

*Bảng 4. 1: Danh sách linh kiện sử dụng*

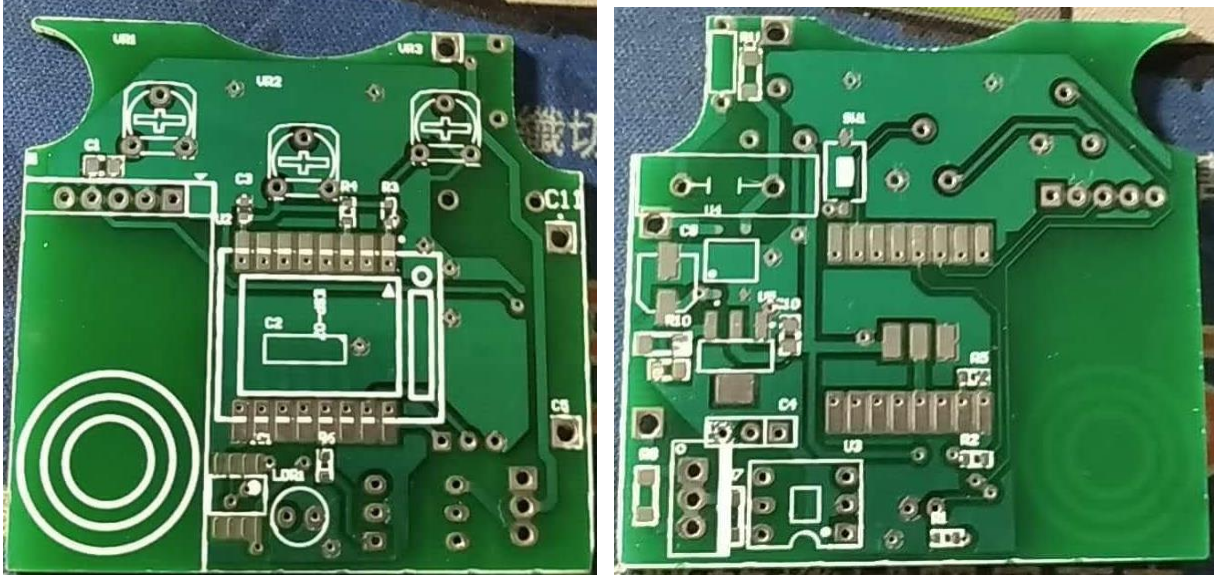
STT	Tên linh kiện	Giá trị
1	ESP-07	3.3V
2	Cảm biến radar	5V
3	Cảm biến quang trở	3.3V
4	Biến trở	10k $\Omega$
5	Điện trở	4.7k $\Omega$ , 10k $\Omega$ , 330 $\Omega$ , 470 $\Omega$
6	Transistor C1815	5V
7	LM358	3.3V
8	LM1117	3.3V
9	Moc3021	3.3V
10	BT137	3.3V

### 4.2.2 Lắp ráp và kiểm tra

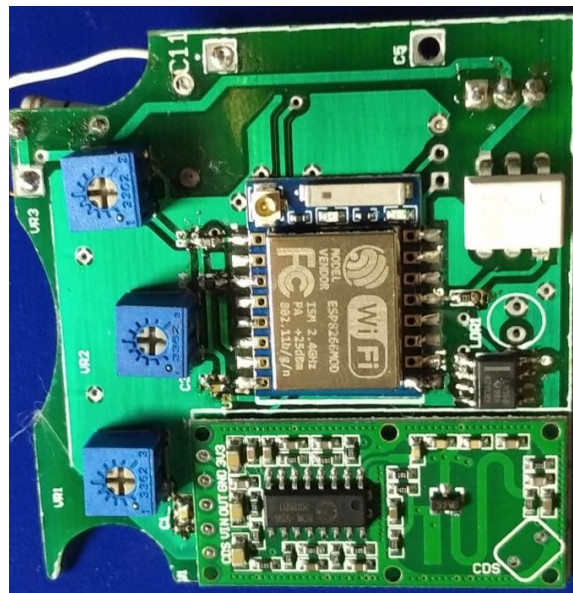
Phần này trình bày lắp ráp các linh kiện vào trong mạch, bao gồm mạch nguồn và mạch PCB đã thiết kế. Sau khi thiết kế xong PCB em tiến hành đặt gia công mạch in. Ta lắp linh kiện vào đúng vị trí trên sơ đồ và tiến hành hàn.

Dùng đồng hồ VOM để kiểm tra toàn bộ hệ thống mạch điều khiển bao gồm chân nguồn, chân điều khiển xem có đủ điện áp, bị hở hay chập mạch không.

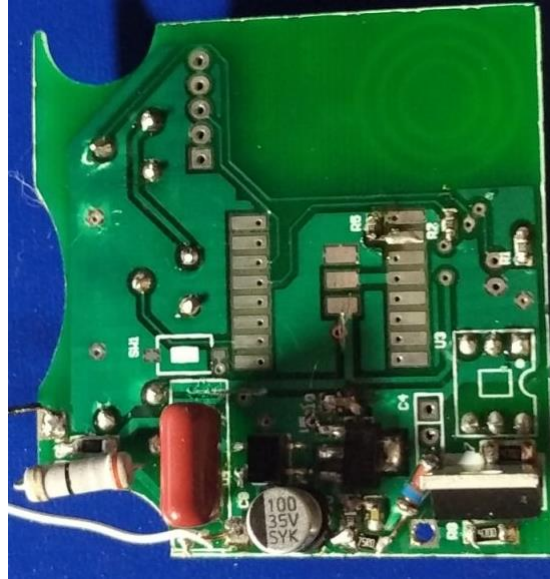
## CHƯƠNG 4: THI CÔNG HỆ THỐNG



*Hình 4. 5: Hai mặt của mạch PCB đã gia công khi chưa lắp linh kiện*



*Hình 4. 6: Mặt trên board khi lắp linh kiện*



*Hình 4. 7: Mặt dưới board khi lắp linh kiện*

### 4.3 ĐÓNG GÓI VÀ THI CÔNG MÔ HÌNH

#### 4.3.1 Đóng gói bộ điều khiển

Sau khi thi công và kiểm tra mạch đã chạy, tiến hành đặt mạch vào vỏ hộp đèn.  
Em sử dụng lại vỏ hộp đèn có sẵn trên thị trường.



*Hình 4. 8: Board mạch sau khi được cho vào hộp đèn*

### 4.3.2 Thi công mô hình

Em đã thiết kế mạch phù hợp với mô hình hộp đèn có sẵn trên thị trường. Mạch nguồn và mạch điều khiển được đặt gọn trong mô hình này.

## CHƯƠNG 4: THI CÔNG HỆ THỐNG

---



*Hình 4. 9: Mô hình mặt trước*



*Hình 4. 10: Mô hình mặt sau*

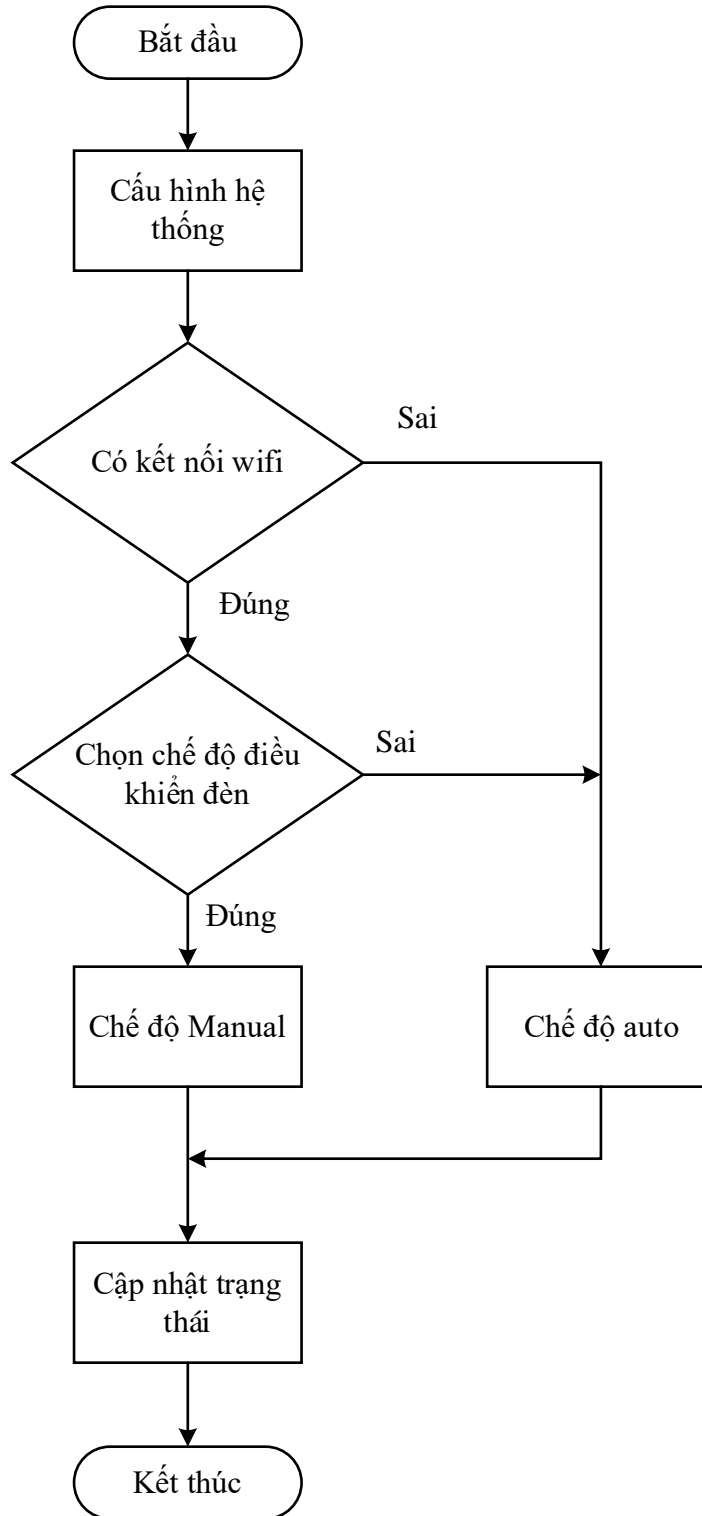
### **4.4 LẬP TRÌNH HỆ THỐNG**

#### **4.4.1 Lưu đồ giải thuật**

Yêu cầu điều khiển: sau khi kết nối wifi thành công, ta mở ứng dụng Blynk hay truy cập giao diện Web Server. Sau đó tiến hành điều khiển bật tắt thiết bị bằng thao tác trên giao diện để tác động đến hệ thống, đồng thời giám sát được trạng thái của thiết bị. Có hai chế độ cho đèn hoạt động là auto và manual.

#### **Chương trình chính**





*Hình 4. 11: Lưu đồ điều khiển thiết bị*

### **Giải thích lưu đồ:**

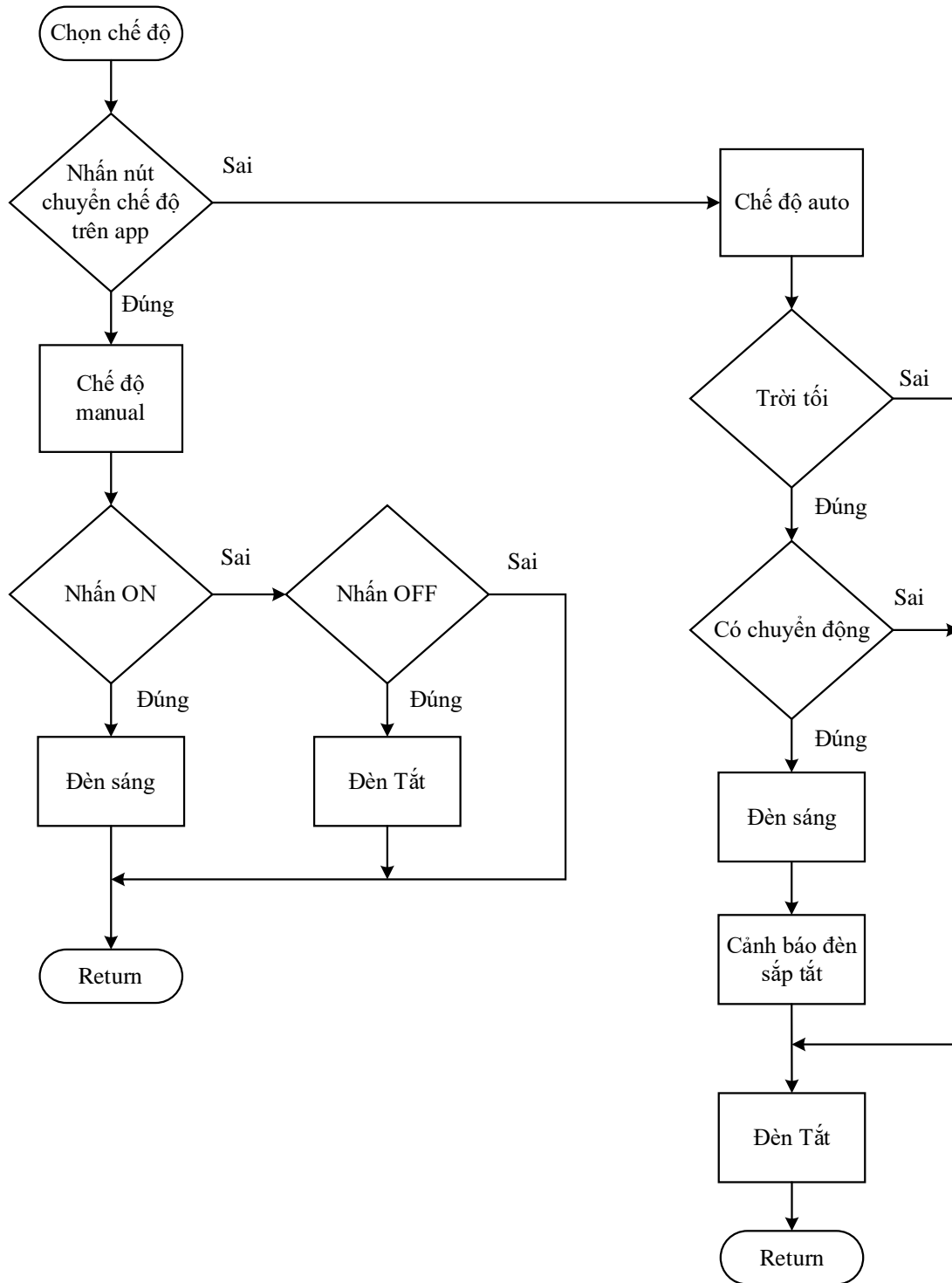
ESP8266 sẽ tiến hành kết nối Internet (wifi), và thiết lập kết nối với Server. Đợi khi kết nối thành công. Nếu có trao đổi với Server (người dùng tác động vào giao diện ứng dụng Blynk hoặc có tín hiệu từ Server gửi xuống), thì đèn sẽ được điều khiển theo yêu cầu người dùng. Ở bất kỳ trường hợp điều khiển nào thì trạng thái của đèn cũng đều được lưu lại và hiển thị trên giao diện ứng dụng và giao diện Web.

Nếu ESP8266 không kết nối wifi thì đèn sẽ mặc định ở chế độ auto.

Khi người dùng tác động vào giao diện trên ứng dụng Blynk, thì đèn sẽ được bật tắt, đồng thời sẽ gửi trạng thái của đèn lên Server.

Khi người dùng tác động vào giao diện web, thì ESP sẽ nhận tín hiệu từ web, đèn sẽ được bật tắt tương ứng, đồng thời sẽ gửi trạng thái của đèn qua Server.

### **Chương trình chọn chế độ hoạt động của đèn**



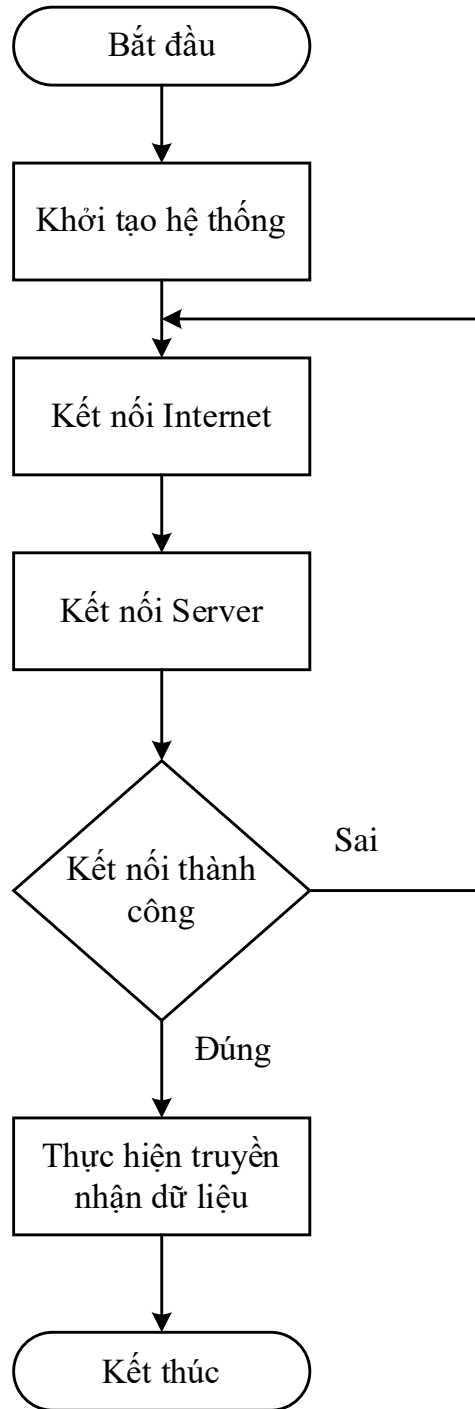
*Hình 4. 12: Lưu đồ lựa chọn chế độ hoạt động của đèn*

### Giải thích lưu đồ:

Đèn có hai chế độ điều khiển là auto và manual:

- Ở chế độ auto ESP8266 sẽ đọc dữ liệu trả về từ các cảm biến, đèn chỉ sáng khi thỏa mãn cùng lúc hai điều kiện đó là khi trời tối và có người chuyển động trong phạm vi quét của cảm biến. Sau khi hết thời gian delay để đèn sáng, đèn sẽ chớp sau đó sáng tiếp thêm 3 giây để cảnh báo cho người dùng biết đèn sắp tắt.
- Ở chế độ manual người dùng có thể trực tiếp điều khiển trạng thái hoạt động của đèn mà không cần xét điều kiện của hai cảm biến. Người dùng có thể điều chỉnh qua app Blynk hay qua giao diện Web.
- Đèn chỉ hoạt động ở chế độ auto khi cả trên app Blynk và trên giao diện Web điều chọn chế độ auto và khi đèn không kết nối được wifi.

### Lưu đồ giải thuật Web Server



*Hình 4. 13: Lưu đồ giải thuật Web Server*

### Giải thích lưu đồ:

Người dùng truy cập vào Web Server sau khi hệ thống đã được cấu hình, và kết nối thành công với Server. Hệ thống sẽ liên tục gửi yêu cầu lên Server để xem có hoạt động truyền nhận dữ liệu hay không. Nếu có thì dữ liệu sẽ được truyền nhận, đồng thời trạng thái của thiết bị được lưu trữ lại để có thể giám sát được.

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

#### Gới thiệu phần mềm Arduino IDE

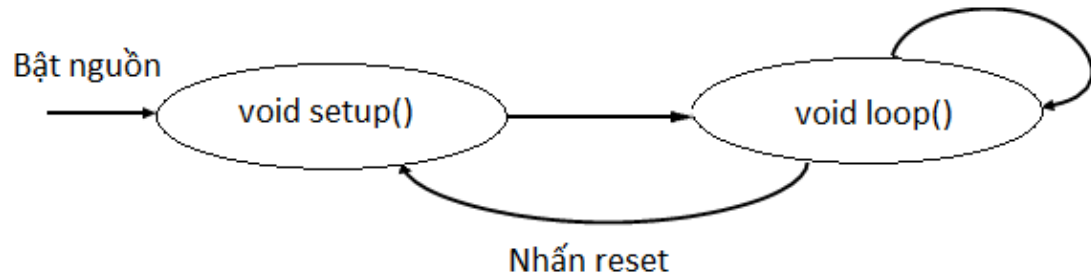
Môi trường phát triển tích hợp (IDE) của Arduino là một ứng dụng nền tảng được viết bằng Java, và từ IDE này sẽ được sử dụng cho Ngôn ngữ lập trình xử lý (Processing programming language) và project Wiring. Công cụ này được đội ngũ kỹ sư của Arduino phát triển và có thể chạy trên Windows, MAC OS X và Linux. Nó được thiết kế để dành cho những người mới tập tành làm quen với lĩnh vực phát triển phần mềm. Nó bao gồm một chương trình code editor với các chức năng như đánh dấu cú pháp, tự động brace matching, và tự động canh lề, cũng như compile (biên dịch) và upload chương trình lên board chỉ với 1 cú click chuột. Một chương trình hoặc code viết cho Arduino được gọi là một sketch.

Các chương trình Arduino được viết bằng C hoặc C++. Arduino IDE đi kèm với một thư viện phần mềm được gọi là "Wiring", từ project Wiring gốc, có thể giúp các thao tác input/output được dễ dàng hơn.

Người sử dụng chỉ cần định nghĩa hai hàm để thực hiện một chương trình hoạt động theo chu trình:

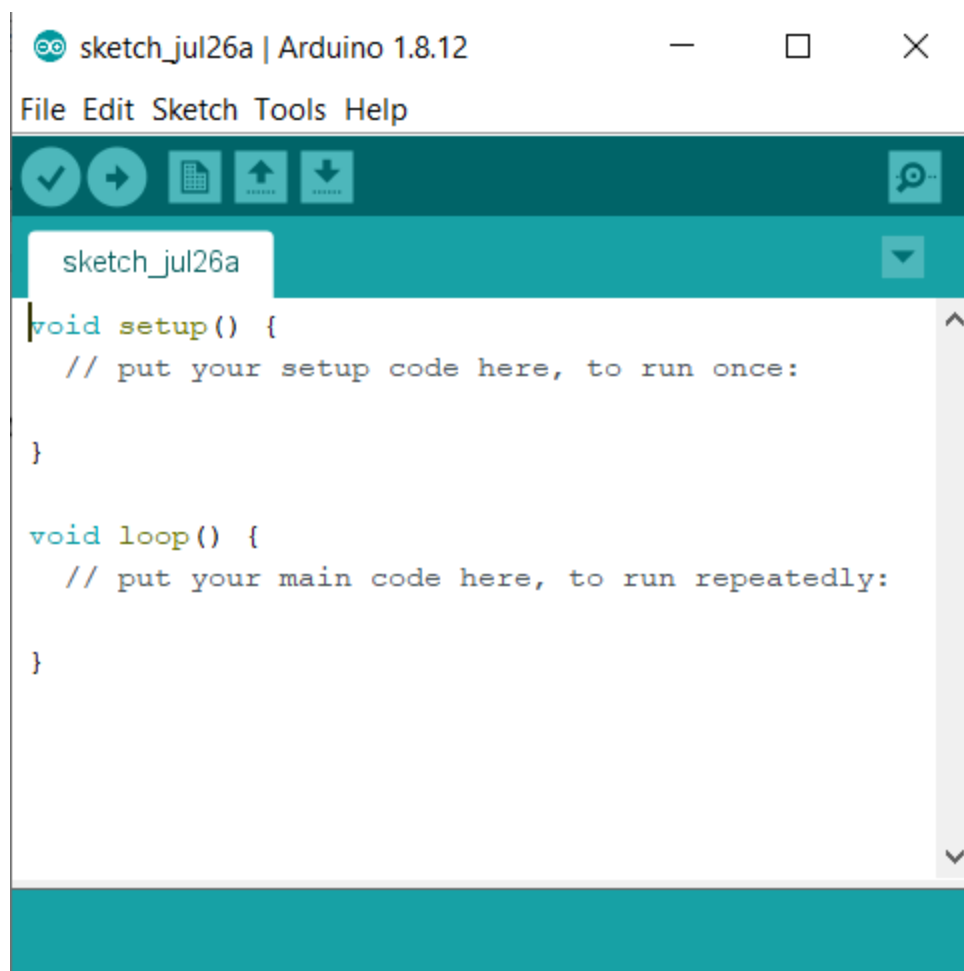
- Setup(): hàm chạy một lần duy nhất vào lúc bắt đầu của một chương trình dùng để khởi tạo các thiết lập.
- Loop(): hàm được gọi lặp lại liên tục cho đến khi bo mạch được tắt.

Chu trình đó có thể mô tả trong hình dưới đây:



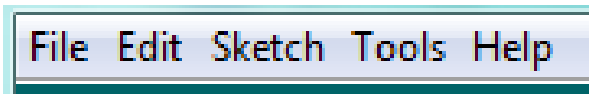
*Hình 4. 14: Quy trình làm việc của Arduino*

Arduino IDE là nơi để soạn thảo code, kiểm tra lỗi và upload code.



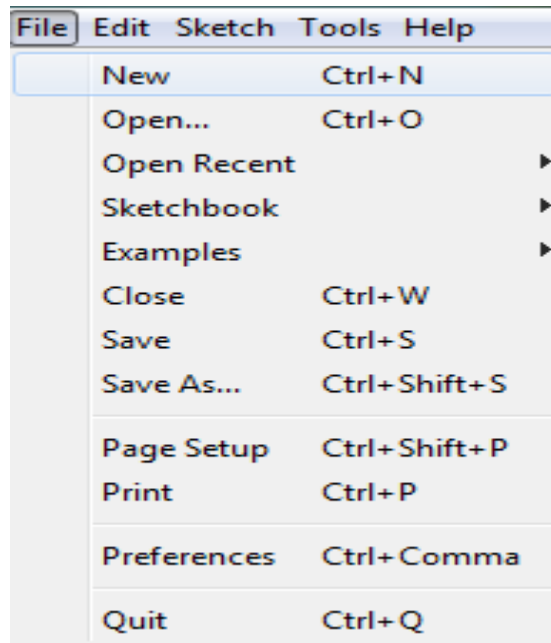
*Hình 4. 15: Giao diện lập trình Arduino*

Arduino IDE Menu:



*Hình 4. 16: Giao diện menu arduino IDE*

Filemenu:

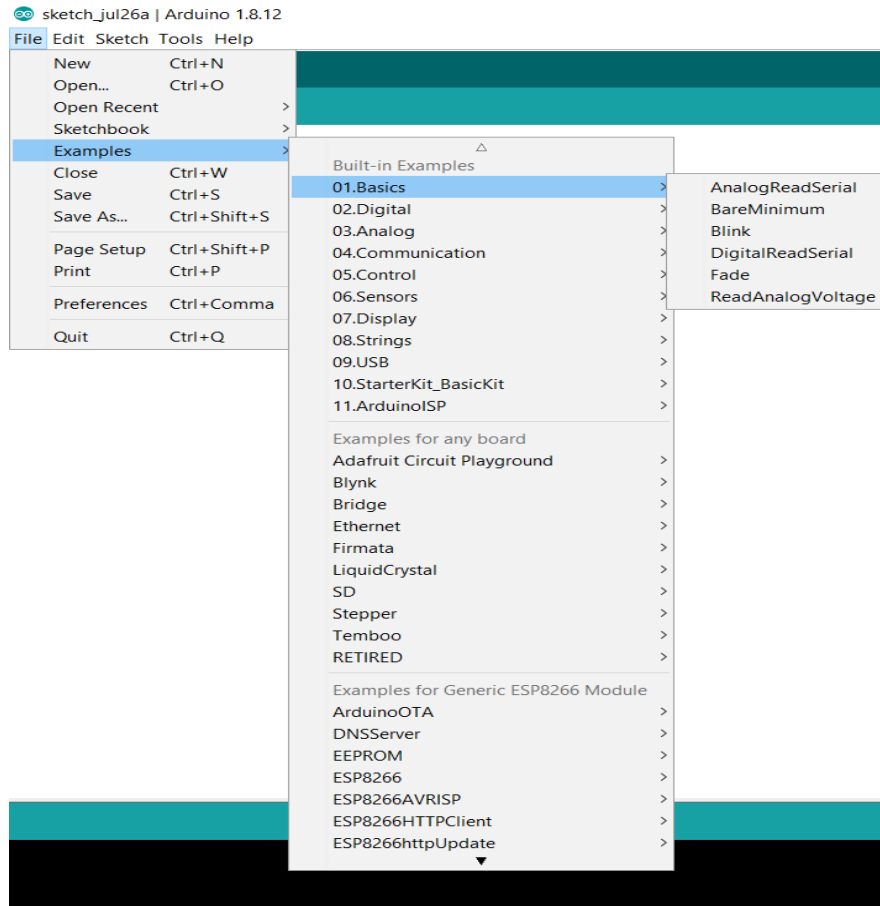


*Hình 4. 17: Giao diện file menu arduino IDE*

Trong file menu chúng ta cần quan tâm tới mục Examples, vì đây là nơi chứa các code chương trình mẫu như điều khiển nhấp nháy led, cách sử dụng các chân digital, analog, cảm biến...



## CHƯƠNG 4: THI CÔNG HỆ THỐNG

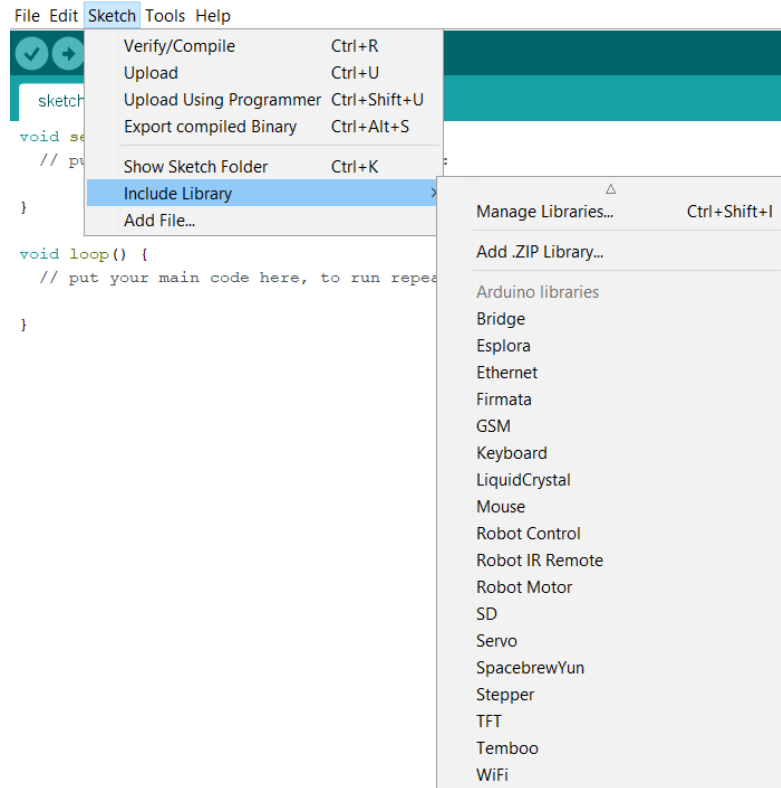


*Hình 4. 18: Giao diện Examples menu*

### **Arduino IDE Sketch menu:**

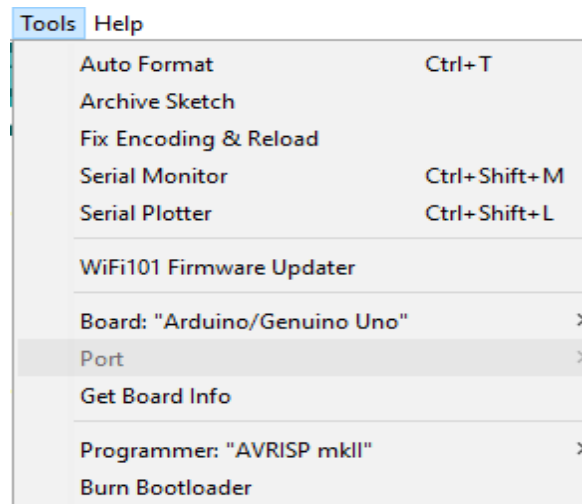
- Verify/ Compile: chức năng kiểm tra lỗi code.
- Show Sketch Folder: hiển thị nơi code được lưu.
- Add File: thêm vào một Tập code mới.
- Import Library: thêm thư viện từ bên ngoài cho IDE.

## CHƯƠNG 4: THI CÔNG HỆ THỐNG



*Hình 4. 19: Giao diện Sketch menu*

**Tool menu:**

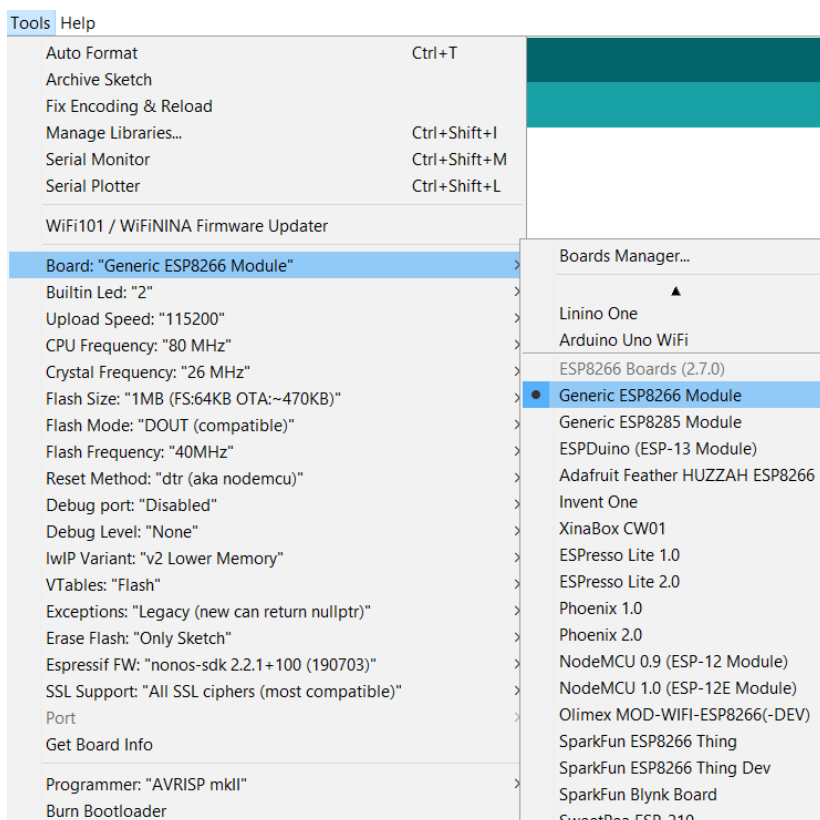


*Hình 4. 20: Giao diện Tool Menu Arduino IDE*

Trong mục Tools ta quan tâm các mục Board và Serial Port. Trong mục Board, cần phải lựa chọn board mạch cho phù hợp với loại board sử dụng. Nếu sử dụng loại

## CHƯƠNG 4: THI CÔNG HỆ THỐNG

board khác thì phải chọn đúng loại board mà mình đang có, nếu sai thì khi Upload chương trình vào chip sẽ bị lỗi. Vì sử dụng Module ESP8266 cho đề tài này nên chúng ta sẽ chọn Board: “Generic ESP8266 Module”.



**Hình 4. 21:** Board ESP8266 sử dụng

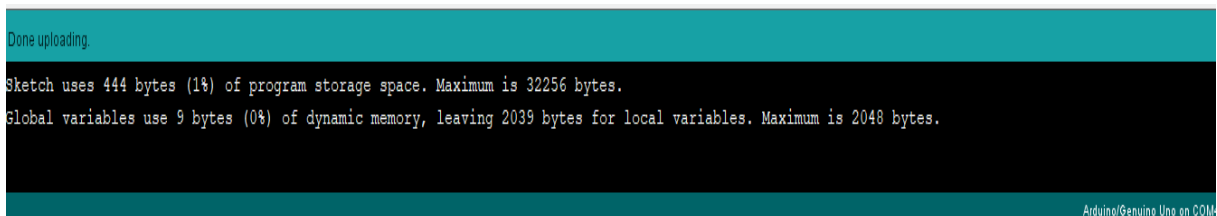
Serial Port: đây là nơi lựa chọn cổng Com của Arduino. Khi chúng ta cài đặt driver thì máy tính sẽ hiện thông báo tên cổng Com của Arduino là bao nhiêu, ta chỉ việc vào Serial Port chọn đúng cổng Com để nạp code, nếu chọn sai thì không thể nạp code cho Arduino được.

Arduino Toolbar có một số button và chức năng của chúng như sau:



*Hình 4. 22: Arduino Toolbar*

- Verify (1): kiểm tra code có lỗi hay không.
- Upload (2): nạp code đang soạn thảo vào Arduino.
- New, Open, Save (3): Tạo mới, mở và lưu sketch.
- Serial Monitor (4): Đây là màn hình hiển thị dữ liệu từ Arduino gửi lên máy tính.



*Hình 4. 23: Chương trình nạp thành công*

### **Viết chương trình hệ thống**

Phần này sẽ được trình bày ở phần phụ lục.

## **4.5 HƯỚNG DẪN SỬ DỤNG, THAO TÁC**

### **4.5.1 Viết tài liệu hướng dẫn sử dụng**

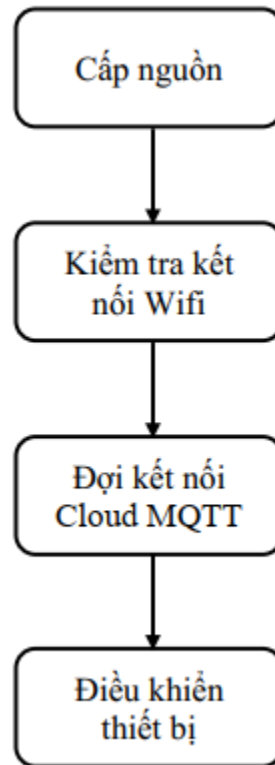
**Bước 1:** Cấp nguồn cho hệ thống. Cấp nguồn 220VAC cho ngõ ra mạch công suất, nguồn 220VAC qua cầu chỉnh lưu và mạch giảm áp để tạo nguồn DC cấp cho mạch điều khiển.

## CHƯƠNG 4: THI CÔNG HỆ THỐNG

---

**Bước 2:** Chờ đến khi ESP8266 kết nối với wifi, mở ứng dụng Blynk, truy cập Web Server. Chọn chế độ hoạt động của đèn, tiến hành điều khiển, giám sát trực tiếp trạng thái của đèn trên giao diện.

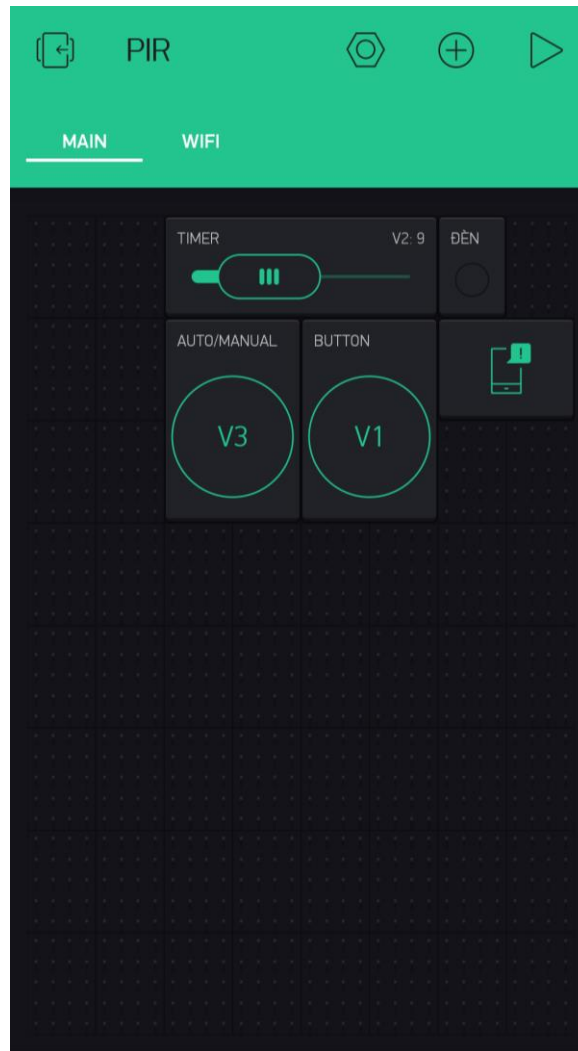
### 4.5.2 Quy trình thao tác



*Hình 4. 24: Quy trình thao tác*

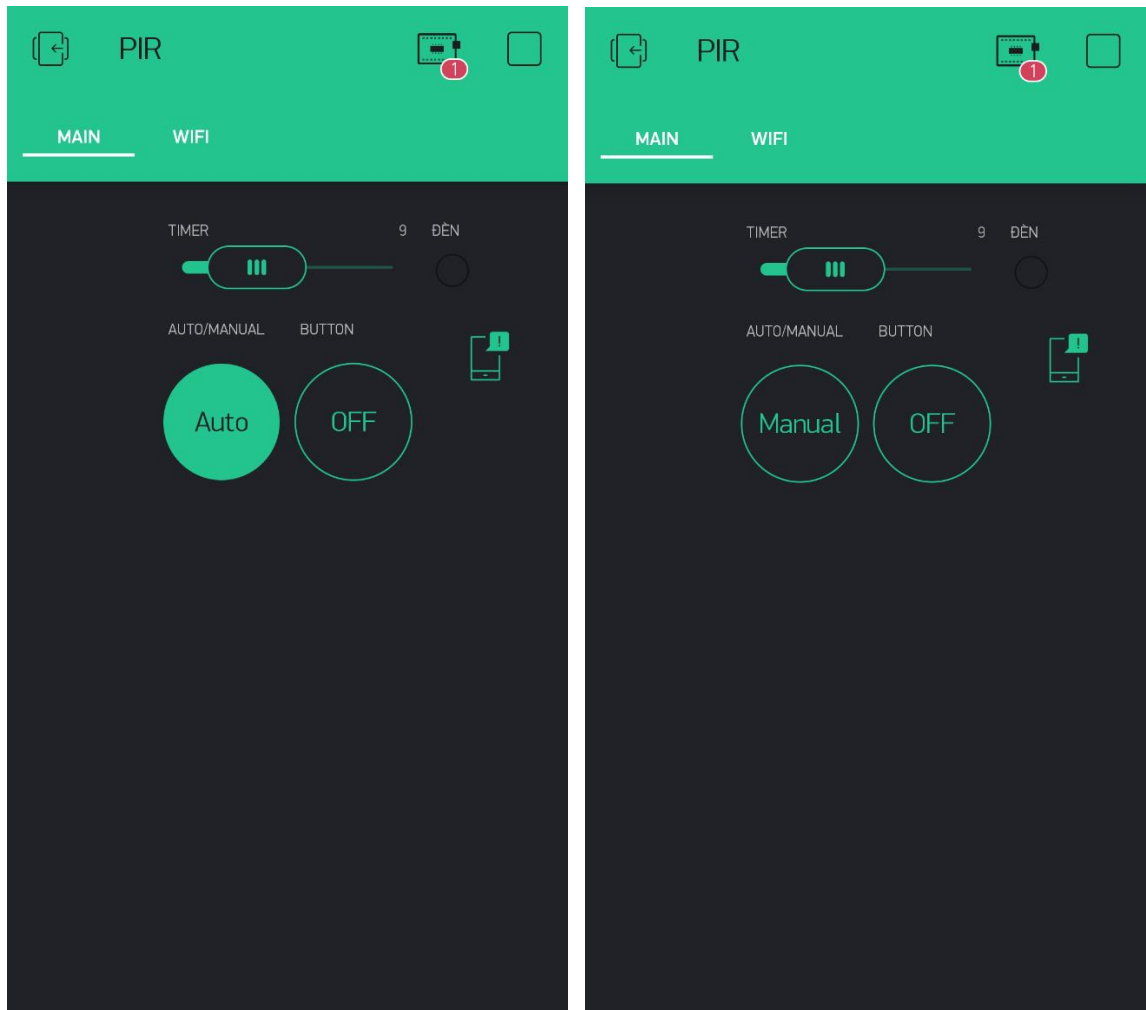
Sau khi tiến hành cấp nguồn, hệ thống sẽ hoạt động như sau:

- Điều khiển qua app Blynk: Người sử dụng mở app Blynk. Đăng ký tài khoản (đối với người sử dụng chưa có tài khoản). Sau đó quét mã QR code đã được dán trên hộp đèn để vào Project. Phát wifi từ điện thoại với tên mặc định: “Do An Tot Nghiep” mật khẩu: “12345678”.



*Hình 4. 25: Giao diện Project điều khiển đèn*

Sau đó nhấn vào biểu tượng hình  $\Delta$  trên góc phải để khởi chạy. Chọn chế độ hoạt động của đèn, có thể điều chỉnh thời gian delay sáng của đèn trực tiếp trên app bằng cách kéo điều chỉnh thanh TIMER.



**Hình 4. 26:** Ấn nút chọn chế độ hoạt động của đèn (Auto/Manual)

Ngoài ra còn có thể đổi wifi truy cập của đèn trực tiếp trên ứng dụng bằng cách nhấn vào thẻ “WIFI” trên ứng dụng. Cú pháp đổi wifi là “abc:123”, với “abc” là tên của wifi và “123” mật khẩu của wifi đó.

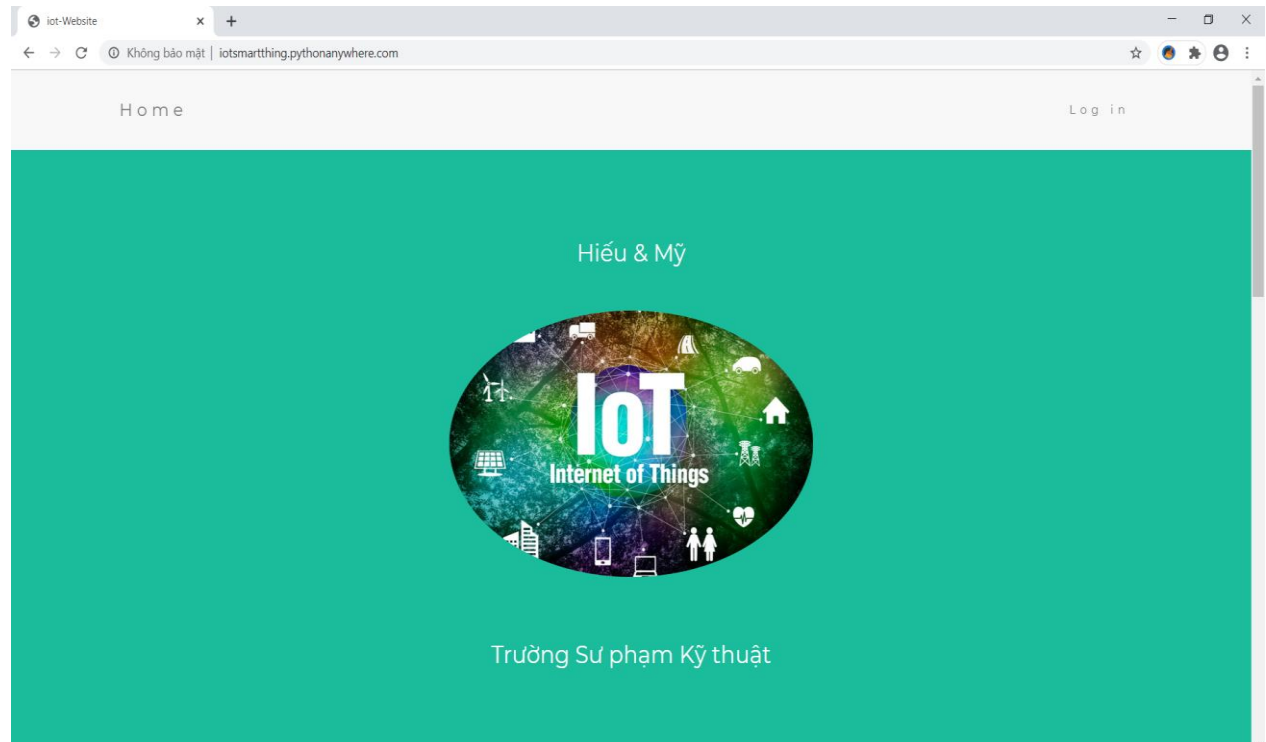


*Hình 4. 27: Giao diện đổi wifi kết nối với đèn*

- Điều khiển qua giao diện Web Server: Người sử dụng sử dụng trình duyệt trên điện thoại hay máy tính truy cập đường dẫn “<http://iotsmartthing.pythonanywhere.com/>”.

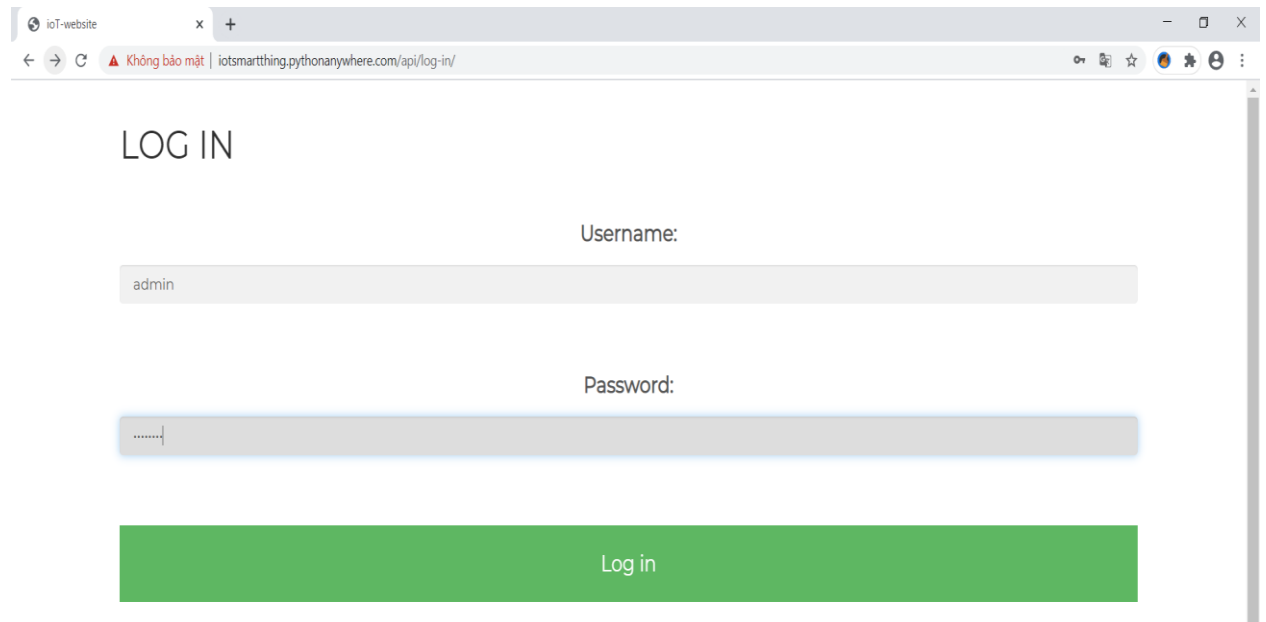


## CHƯƠNG 4: THI CÔNG HỆ THỐNG



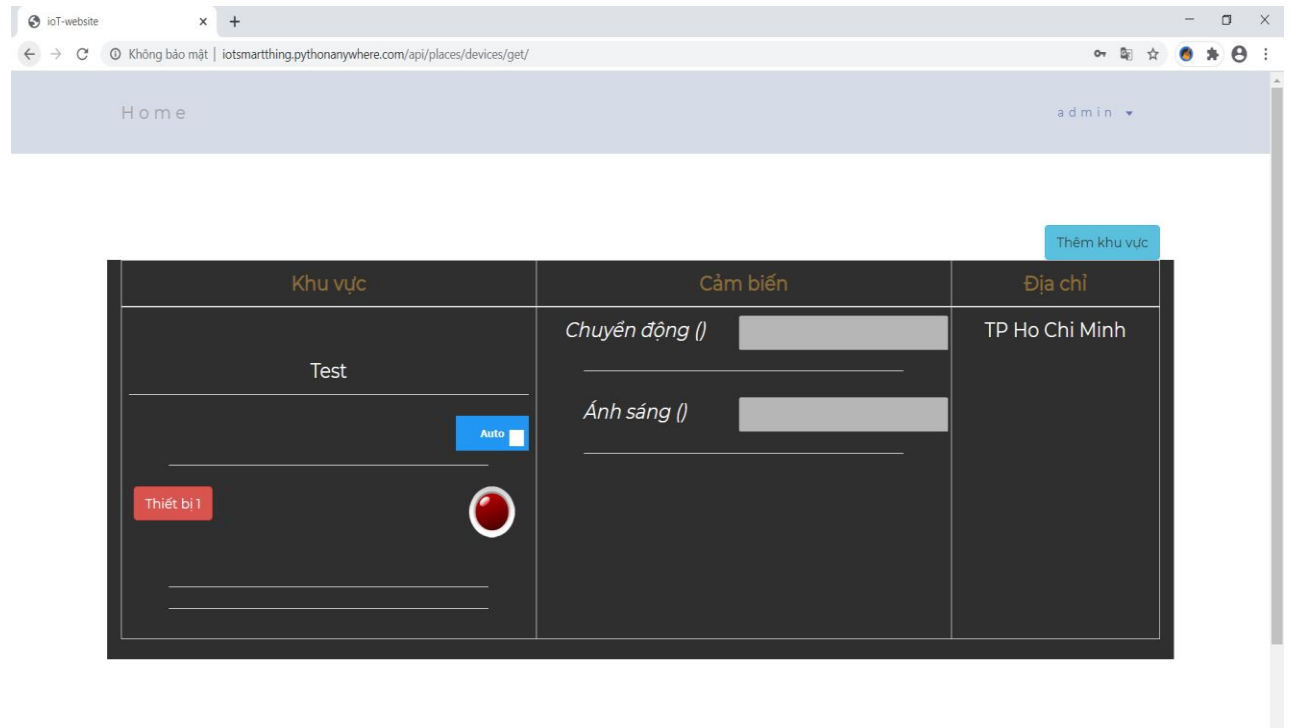
*Hình 4. 28: Giao diện Web Server*

- Sau đó nhấn vào Log in với User name: “admin” và Password: “iot@2020” để đăng nhập chọn chế độ và điều khiển đèn.



*Hình 4. 29: Giao diện đăng nhập*

## CHƯƠNG 4: THI CÔNG HỆ THỐNG



*Hình 4. 30: Giao diện Web Server chọn chế độ và điều khiển đèn.*

# CHƯƠNG 5: KẾT QUẢ - NHẬN XÉT - ĐÁNH GIÁ

## 5.1 GIỚI THIỆU

Chương này trình bày kết quả của cả quá trình nghiên cứu làm đề tài trong thời gian 17 tuần. Bên cạnh đó là nhận xét, đánh giá và đề xuất hướng phát triển của sản phẩm mô hình để hoàn thiện và có thể đi vào thực tế.

## 5.2 KẾT QUẢ ĐẠT ĐƯỢC

Qua quá trình nghiên cứu và thực hiện đề tài, em nhận thấy được đề tài có khả năng ứng dụng vào thực tiễn cao, có thể giám sát trực tiếp trên ứng dụng điện thoại và có thể điều hệ thống trực tiếp trên web, đáp ứng được nhu cầu xã hội hiện đại ngày nay và đi đúng với xu hướng của thời đại. Đồng thời đề tài này cũng là một nguồn tài liệu có giá trị cho các bạn sinh viên những khóa tiếp theo có thể tham khảo khi nghiên cứu những đề tài có liên quan cũng như là phát triển thêm đề tài từ nền tảng có sẵn mà em đã nghiên cứu. Bên cạnh đó, em cũng bổ sung cho mình những kiến thức hay và bổ ích như:

- Biết cách sử dụng phần mềm altium để thiết kế sơ đồ nguyên lý của hệ thống, thiết kế mạch PCB 2 lớp.
- Nghiên cứu và biết cách kết nối giữa ESP8266 với các cảm biến.
- Hiểu được cấu tạo, chức năng, chuẩn kết nối, giao tiếp của module ESP8266 qua đó hiểu được hơn về lĩnh vực IoTs, truyền thông công nghiệp đang rất có tiềm năng phát triển.
- Nâng cao được kỹ năng thi công mạch (hàn linh kiện, kiểm tra các thành phần trong mạch).
- Biết cách tính toán các giá trị điện áp dòng điện trong hệ thống để chọn nguồn phù hợp.
- Biết kết nối các linh kiện với nhau sao cho phù hợp tạo thành một hệ thống hoàn chỉnh.
- Biết lập trình cho arduino, ESP8266 sử dụng phần mềm adduino IDE.

### 5.3 KẾT QUẢ THỰC NGHIỆM

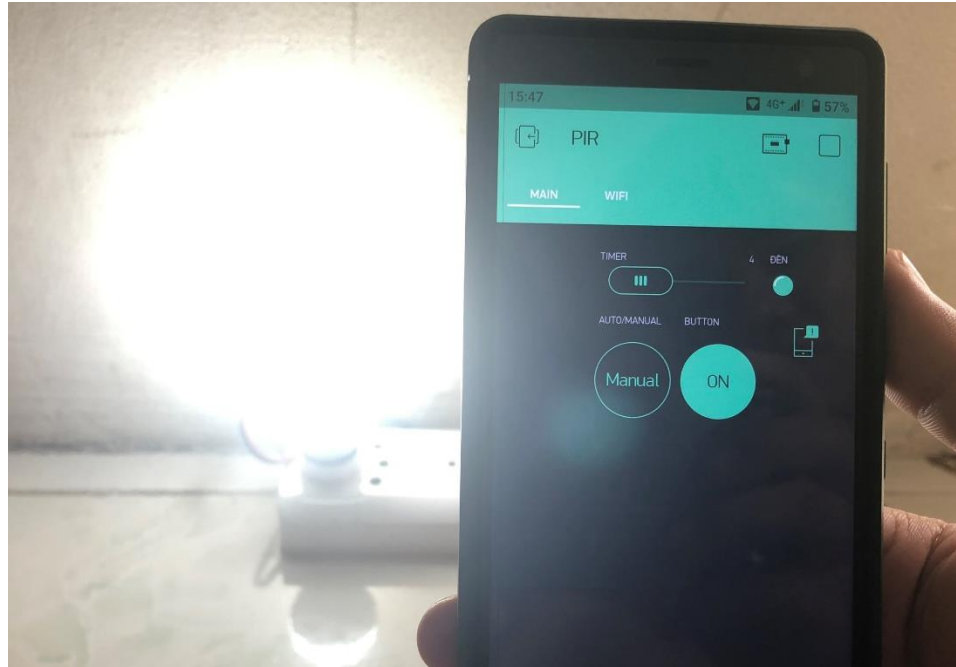
#### 5.3.1 Mô hình



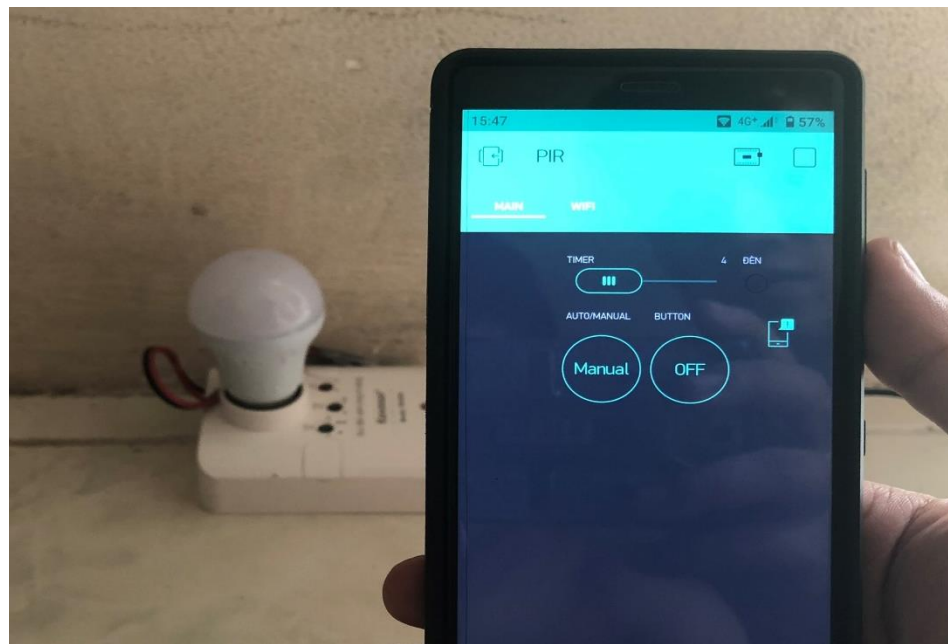
*Hình 5. 1: Mô hình hộp đèn hoàn chỉnh*

#### 5.3.2 Điều khiển, giám sát thiết bị thông qua ứng dụng Blynk

Giao diện điều khiển và giám sát trạng thái của đèn ở chế độ Manual.



*Hình 5. 2: Điều chỉnh mở đèn ở chế độ Manual*



*Hình 5. 3: Điều chỉnh tắt đèn ở chế độ Manual*

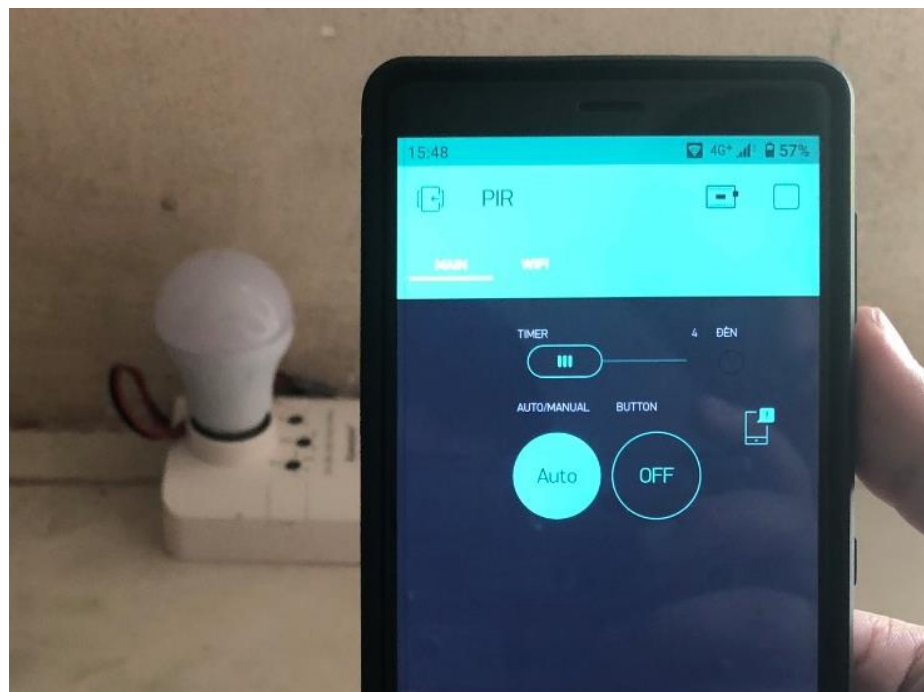
## CHƯƠNG 5: KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ

---

Giao diện điều khiển và giám sát trạng thái của đèn ở chế độ Auto.



*Hình 5. 4: Giao diện lúc đèn sáng ở chế độ Auto*



*Hình 5. 5: Giao diện lúc đèn tắt ở chế độ Auto*

### 5.4 NHẬN XÉT VÀ ĐÁNH GIÁ

#### 5.4.1 Nhận xét

Sau thời gian 17 tuần nghiên cứu và thực hiện đề tài, hệ thống cơ bản đáp ứng được yêu cầu thiết kế ban đầu. Dưới đây là một số nhận xét:

- Nhìn chung mô hình hoạt động đạt 100% yêu cầu đề ra ban đầu.
- Hệ thống hoạt động ổn định qua nhiều lần thử nghiệm.
- Hệ thống sử dụng nguồn cấp nhỏ từ 5V trở xuống nên an toàn cho người sử dụng trước nguy cơ điện giật.
- Ứng dụng điện thoại giám sát hệ thống có giao diện dễ sử dụng và quan sát. Ứng dụng trên điện thoại hoạt động tốt có thể dễ dàng điều khiển từ xa theo mong muốn của người dùng.
- Tốc độ điều khiển tương đối nhanh.
- Giám sát được trạng thái của thiết bị.
- Dễ dàng sử dụng, lắp đặt và bảo dưỡng.
- Tuy nhiên hệ thống vẫn còn nhiều hạn chế như:
  - Hệ thống phụ thuộc vào tốc độ mạng wifi, và sự ổn định của Server.
  - Sử dụng lại hộp đèn có sẵn trên thị trường.

#### 5.4.2 Đánh giá

Trong quá trình vận hành hệ thống, em đã ghi nhận lại kết quả được tổng hợp.

## CHƯƠNG 5: KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ

---

*Bảng 5. 1: Số liệu thực nghiệm*

<b>Công việc</b>	<b>Số lần thao tác</b>	<b>Số lần thành công</b>	<b>Đánh giá</b>
Có chuyển động trong phạm vi quét của cửa cảm biến	30	30	Đạt
Điều khiển đèn qua ứng dụng	30	28	Đạt
Cập nhật trạng thái đèn	30	28	Đạt
Giám sát cảm biến	Ổn định	Ổn định	Đạt

Qua những số liệu được thống kê ở bảng trên, em đánh giá hệ thống về cơ bản đã đáp ứng được mục tiêu đặt ra. Hệ thống hoạt động ổn định sau nhiều lần chạy, kiểm tra thử trong nhiều trường hợp. Mô hình nhỏ gọn, thẩm mỹ, nhưng vẫn đảm bảo tính an toàn cao, dễ dàng lắp đặt và sử dụng.



# CHƯƠNG 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## 6.1 KẾT LUẬN

Sau thời gian nghiên cứu, tìm hiểu và thực hiện em đã hoàn thành đề tài “**NGHIÊN CỨU, CẢI TIẾN VÀ THI CÔNG BỘ ĐÈN LED TÍCH HỢP CẢM BIẾN QUANG, CHUYỂN ĐỘNG VÀ WIFI**”. Đạt được yêu cầu ban đầu đề ra, cụ thể là:

### ❖ Ưu điểm:

- Cải tiến được bộ đèn có sẵn trên thị trường.
- Có thể giám sát trạng thái và điều khiển đèn thông qua ứng dụng Blynk.
- Giao diện điều khiển và giám sát dễ sử dụng, thân thiện người dùng.
- Toàn bộ mô hình có thể hoạt động trong thời gian dài, tương đối ổn định, đạt kết quả tốt.

### ❖ Nhược điểm:

- Chưa có hộp riêng cho sản phẩm.
- Chưa tích hợp được nhiều tính năng mới.
- Sử dụng dịch vụ Server, nên còn hạn chế về sự ổn định.
- Giới hạn về thời gian, kiến thức nên hệ thống chưa được tối ưu.

## 6.2 HƯỚNG PHÁT TRIỂN

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

- Thêm chức năng điều khiển độ sáng của đèn.
- Thêm chức năng điều khiển bằng giọng nói.
- Thiết kế hộp đựng riêng cho đèn.

## **TÀI LIỆU THAM KHẢO**

- [1] Nguyễn Đình Phú, “*Giáo trình Vi xử lý*”, Trường ĐHSPKT, Tp.HCM, 2013.
- [2] Nguyễn Đình Phú, “*Giáo trình: Kỹ thuật số*”, Nhà xuất bản ĐH Quốc Gia, Tp.HCM, 2013.
- [3] Nguyễn Văn Hiệp – Đinh Quang Hiệp, “*Lập trình android cơ bản*”, Nhà xuất bản ĐH Quốc Gia, Tp.HCM, 2015.
- [4] Phạm Văn Huy, “*Hệ thống điều khiển và giám sát các thiết bị trong nhà*”, Đồ án tốt nghiệp, trường ĐHSPKT Tp.HCM, 2017.
- [5] Nguyễn Ngọc Lực, “*Thiết kế, thi công mô hình hệ thống điều khiển thiết bị điện và giám sát nhà*”, Đồ án tốt nghiệp, trường ĐHSPKT Tp.HCM, 2018.
- [6] Trần Minh Luân – Lâm Thành Đạt, “*Thiết kế và thi công hệ thống thiết bị điều khiển nhà thông minh*”, Đồ án tốt nghiệp, trường ĐHSPKT Tp.HCM, 2018.

### **Website tham khảo**

- [1] <https://arduino.vn>
- [2] <https://wikipedia.org>
- [3] <https://alldatasheet.com>
- [4] <https://iotmaker.vn>
- [5] <https://phocode.com>

## **PHỤ LỤC**

### **Code chương trình**

```
#define BLYNK_PRINT Serial

#include <ESP8266WiFi.h>

#include "EEPROM.h"

#include <BlynkSimpleEsp8266.h>

#include <PubSubClient.h>

#include <SimpleTimer.h>

#include <ArduinoJson.h>

WidgetLED ALARM(V0);

WidgetTerminal terminal(V20);

#define PIR 12

#define LDR 14

#define LED 4

char auth[] = "vA9_xMZeFqa5HjVSViQ7powY4ODg4eSy";

char ssid_default[] = "Do An Tot Nghiep";

char pass_default[] = "12345678";

char ssid_new[25] = "" ;

char pass_new[25] = "" ;

char ssid_buff[25] = "" ;
```

## **PHỤ LỤC**

---

```
char pass_buff[25] = "" ;
```

```
boolean switchStatus = 1;
```

```
boolean switchStatusW = 1;
```

```
boolean ledStatus = 0;
```

```
boolean valPIR ;
```

```
boolean valLDR ;
```

```
boolean sttPIR ;
```

```
boolean sttLDR ;
```

```
boolean flag1 = 0 ;
```

```
boolean flag2 = 0 ;
```

```
int valueTimer = 5;
```

```
SimpleTimer timer;
```

```
unsigned long time1 = 0;
```

```
unsigned long time2 = 0;
```

```
unsigned long time_delay = 0;
```

## PHỤ LỤC

---

```
#define mqtt_server "m16.cloudmqtt.com"

#define mqtt_topic_pub "subscribeTopic2"

#define mqtt_topic_sub "publishTopic2"

#define mqtt_user "vwlfeugw"

#define mqtt_pwd "YM_PrssCv2lu"

char sever[] = "";

char mqtt2[200] = "";

boolean wifiStatus = 0;

const uint16_t mqtt_port = 19932;

WiFiClient espClient;

PubSubClient client(espClient);

String receiver = "";

String sttOn = "gateway2-ld1:a-statusA";

String sttOff = "gateway2-ld1:b-statusA";

String getValue(String data, char separator, int index)

{

    int found = 0;
```

## PHỤ LỤC

---

```
int strIndex[] = {0, -1};

int maxIndex = data.length()-1;

for(int i=0; i<=maxIndex && found<=index; i++){
    if(data.charAt(i)==separator || i==maxIndex){
        found++;
        strIndex[0] = strIndex[1]+1;
        strIndex[1] = (i == maxIndex) ? i+1 : i;
    }
}

return found>index ? data.substring(strIndex[0], strIndex[1]) : "";
}

void delete_eeprom(){
    for (int i = 0; i <64; i++) {
        EEPROM.write(i, 0);
        delay(5);
    }
    EEPROM.commit();
}
```

## PHỤ LỤC

---

```
void callback(char* topic, byte* payload, unsigned int length) {

    for (int i = 0; i < length; i++) {

        char receivedChar = (char)payload[i];

        receiver += receivedChar;

    }

    //Serial.println();

    // Serial.print("Chuoi nhan duoc: ");

    // Serial.println(receiver);

    check_control();

}

void reconnect() {

    // Chờ tới khi kết nối

    while (!client.connected()) {

        // Thực hiện kết nối với mqtt user và pass

        if (client.connect("ESP8266Client",mqtt_user, mqtt_pwd)) {

            Serial.println("connected");

            client.publish(mqtt_topic_pub, "ESP_reconnected");

            client.subscribe(mqtt_topic_sub);

        } else {

            Serial.print("failed, rc=");
```

```
    Serial.print(client.state());

    Serial.println(" try again in 1 seconds");

    delay(1000);

}

}

}

void setup() {

    Serial.begin(9600);

    pinMode(PIR, INPUT);

    pinMode(LDR, INPUT);

    pinMode(LED, OUTPUT);

    Serial.println("Start");

    EEPROM.begin(64);

    //delete_eeprom();

    Serial.println("Da xoa eeprom");

    EEPROM.get(1,ssid_new);

    delay(100);

    EEPROM.get(30,pass_new);

    String st1 = ssid_new;

    String st2 = pass_new;

    Serial.print("ID:");
```



```
Serial.println(st1);

Serial.print("Length:");

Serial.println(st1.length());

Serial.print("Pass:");

Serial.println(st2);

Serial.print("Length:");

Serial.println(st2.length());

if ((st1.length() > 2)&& (st2.length() > 2)){

WiFi.begin(ssid_new, pass_new);

int wifi_ctr = 0;

while (WiFi.status() != WL_CONNECTED) {

  for (int i =0 ; i <20 ; i++)

  {

    if (WiFi.status() == WL_CONNECTED){

      Serial.println("WiFi new connected");

      Blynk.begin(auth, ssid_new, pass_new);

      wifiStatus = 1;

      break;

    }

    Serial.print(".");

    delay(500);

  }

}
```

```
break ;  
  
}  
  
}  
  
if (wifiStatus == 0) {  
  
    WiFi.begin(ssid_default, pass_default);  
  
    while (WiFi.status() != WL_CONNECTED) {  
  
        for (int i =0 ; i <20 ; i++)  
  
        {  
  
            if (WiFi.status() == WL_CONNECTED){  
  
                Serial.println("WiFi default connected");  
  
                Blynk.begin(auth, ssid_default, pass_default);  
  
                wifiStatus = 1;  
  
                break;  
  
            }  
  
            Serial.print(".");  
  
            delay(500);  
  
        }  
  
        break ;  
  
    }  
  
}  
  
ALARM.off();  
  
analogWrite(LED, 0);
```

## PHỤ LỤC

---

```
Blynk.notify("Device Start!");

terminal.clear();

terminal.println(F("He thong da khoi dong"));

terminal.println(F("Moi nhap SSID va PASS wifi"));

terminal.println(F("Theo dinh dang abc;123"));

terminal.println(F("abc la SSID, 123 la PASS "));

terminal.flush();

client.setServer(mqtt_server, mqtt_port);

client.setCallback(callback);

timer.setInterval(500, readSensor);

}

void loop() {

  timer.run();

  Blynk.run();

  if ((switchStatus == 1)&&(switchStatusW == 1))

  {

    if ((valPIR == 0) && (valLDR == 1))

    {

      analogWrite(LED, 1024);

      ALARM.on();

      update_status(1);

    }

  }

}
```

```
//Serial.println("Bat");

flag2 = 1;

time1 = millis();

}

//else flag2 = 0;

if ((valPIR == 0)&&(flag2 == 1))

{

    if (flag1 == 1)

    {

        time2 = millis();

    }

    else time1 = millis();

    //Serial.println("Reset time");

}

time_delay = valueTimer * 1000 ;

//Serial.println(time_delay);

if (( (unsigned long) (millis() - time1) > time_delay ) && (flag1 == 0) && (flag2 ==

1))

{

    //Serial.println("Tat");

    analogWrite(LED, 0);

    turnOff();

}
```

```
    ALARM.off();

    update_status(2);

    delay(1000);

    analogWrite(LED, 1024);

    ALARM.on();

    update_status(1);

    time2 = millis();

    flag1 = 1;

}

if (( (unsigned long) (millis() - time2) > 3000 )&&(flag1 == 1) && (flag2 == 1))

{

    analogWrite(LED, 0);

    turnOff();

    ALARM.off();

    update_status(2);

    flag1 = 0;

    flag2 = 0;

}

}

if (wifiStatus == 1)

{

    if (!client.connected()) {
```

```
    reconnect();  
  }  
  client.loop();  
}  
  
}  
BLYNK_WRITE(V1)  
{  
  int i = param.asInt();  
  if (switchStatus == 0)  
  {  
    if (i == 1)  
    {  
      analogWrite(LED, 1024);  
      ALARM.on();  
      update_status(1);  
    }  
  }  
  else  
  {  
    analogWrite(LED, 0);  
    ALARM.off();  
    update_status(2);  
  }  
}
```

## PHỤ LỤC

---

```
}  
  
}  
  
}  
BLYNK_WRITE(V3)  
{  
    int i = param.asInt();  
    if (i == 1)  
    {  
        switchStatus = 1;  
    }  
    else  
    {  
        switchStatus = 0;  
    }  
}  
BLYNK_WRITE(V2)  
{  
    valueTimer = param.asInt();  
    //analogWrite(14, value);  
}  
BLYNK_WRITE(V6)  
{
```

```
int i = param.asInt();

if (i == 1)

{ delete_eeprom();

  delay(50);

  terminal.println(ssid_buff);

  terminal.println(pass_buff);

  EEPROM.put(1, ssid_buff);

  EEPROM.put(30, pass_buff);

  EEPROM.commit();

  terminal.println("CAP NHAT THANH CONG");

  terminal.println("Auto reboot...");

  terminal.flush();

  delay(2000);

  ESP.restart();

}

else

{

  ;

}

}

BLYNK_WRITE(V7)

{
```



## PHỤ LỤC

---

```
int i = param.asInt();

if (i == 1)

{ delete_eeprom();

  delay(50);

  terminal.println("Da xoa bo nho");

  terminal.flush();

}

else

{

;

}

}

BLYNK_WRITE(V20)

{

String t = param.asStr();

String id = getValue(t, ';', 0);

String pass = getValue(t, ';', 1);

terminal.print("SSID :");

terminal.println(id);

terminal.print("PASS :");

terminal.println(pass);

terminal.flush();
```

## PHỤ LỤC

---

```
id.toCharArray(ssid_buff,25);

pass.toCharArray(pass_buff,25);

}

void readSensor()

{

    valPIR = digitalRead(PIR);

    valLDR = digitalRead(LDR);

    sendmqtt();

// Serial.print("Sensor PIR: ");

// Serial.print(valPIR);

// Serial.print(" ");

// Serial.print("Sensor LDR: ");

// Serial.println(valLDR);

}

void turnOff()

{

    for ( int j = 1000; j >= 0; j = j - 50)

    {

        analogWrite(LED, j);

        delay(50);

    }

    analogWrite(LED, 0);
```

```
}  
  
void sendmqtt() { // Hàm truyền dữ liệu lên server mqtt  
  
    DynamicJsonBuffer jsonBuffer;  
  
    if (valPIR == 0){  
        sttPIR = 1;  
    }  
  
    else sttPIR = 0;  
  
    if (valLDR == 0){  
        sttLDR = 1;  
    }  
  
    else sttLDR = 0;  
  
    String stringMoti = String("{\"serial\":\"sen4\", \"type\":1, \"value\":") + String(sttPIR)  
+ String("}");  
  
    String stringLig = String("{\"serial\":\"sen5\", \"type\":1, \"value\":") + String(sttLDR)  
+ String("}");  
  
    JsonObject& root3 = jsonBuffer.parseObject(stringMoti);  
  
    JsonObject& root4 = jsonBuffer.parseObject(stringLig);  
  
    char outputC[200];  
  
    char outputP[200];  
  
  
    root3.printTo(outputC);  
  
    root4.printTo(outputP);  
  
    //Serial.println(output);
```

## PHỤ LỤC

---

```
// client.publish(mqtt_topic_sub, stringsta);
client.publish(mqtt_topic_pub, outputC);
client.publish(mqtt_topic_pub, outputP);
// client.publish(mqtt_topic_pub, mqtt);
}

void update_status(int i)
{
switch(i){
case 1: sttOn.toCharArray(mqtt2,100);
        client.publish(mqtt_topic_pub, mqtt2);
        break;
case 2: sttOff.toCharArray(mqtt2,100);
        client.publish(mqtt_topic_pub, mqtt2);
        break;
default: break;
}
}

void check_control()
{
receiver.trim();

String check1 = getValue(receiver, '-', 0);

Serial.println(check1);
```

```
if (check1 == "gateway2")
{
    if (receiver.length() == 15)
    {
        String value = getValue(receiver, '-', 1);
        Serial.println(value);
        if (value == "mode:a")
        {
            switchStatusW = 1;
            //Serial.println(switchStatusW);
        }
        if (value == "mode:b")
        {
            switchStatusW = 0;
            //Serial.println(switchStatusW);
        }
    }
}

if (receiver.length() == 23)
{
    String value = getValue(receiver, '-', 1);
    Serial.println(value);
}
```

```
//Serial.println(switchStatusW);  
  
if ((value == "ld1:a")&& (switchStatusW == 0))  
{  
    analogWrite(LED, 1024);  
    ALARM.on();  
    update_status(1);  
}  
  
if ((value == "ld1:b")&& (switchStatusW == 0))  
{  
    analogWrite(LED, 0);  
    ALARM.off();  
    update_status(2);  
}  
}  
  
receiver = "";  
}
```

## PHỤ LỤC

---

