

**ĐẠI HỌC QUỐC GIA TP HCM
TRƯỜNG ĐẠI HỌC BÁCH KHOA**

-----oOoOo-----
KHOA ĐIỆN – ĐIỆN TỬ



**BÁO CÁO BÀI TẬP LỚN
MÔN HỌC: THIẾT KẾ HỆ THỐNG NHÚNG
ĐỀ TÀI: ĐÈN HẸN GIỜ THÔNG MINH ĐIỀU KHIỂN BẰNG
GIỌNG NÓI**

Giảng viên hướng dẫn: Trần Hoàng Quân

Lớp L01

Nhóm N5: SLED

Sinh viên thực hiện:

STT	Họ và tên	MSSV	Đóng góp	Điểm
1	Nguyễn Thế Bằng	2110048	100%	
2	Tô Minh Vũ	2110677	100%	
3	Nguyễn Đăng Cao Bằng	2110047	100%	
4	Nguyễn Hoàng Long Vũ	2112668	100%	
5	Trần Thái Dương	2110983	100%	

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023

MỤC LỤC

PHẦN I: GIỚI THIỆU ĐỀ TÀI	3
1. Tổng quan về đề tài	3
1.1. Tổng quan:	3
1.2 Mục đích:	3
1.3 Phân công công việc:	4
2. Đặc tả hệ thống (System Specification):	4
2.1. Product Specification:	4
2.2. Đặc điểm kĩ thuật (Engineering Specification):	4
2.3. Đặc điểm phần cứng (Hardware Specification):	5
2.4. Đặc tả phần mềm (Software Specification):	11
2.5. Test Specification:	12
3. Vấn đề thiết kế (Design Issues)	13
3.1. Constraints:	13
2.2. Functional issues:	13
2.3. Real-time issues:	13
2.4. Concurrent issues:	13
2.5. Reactive issues:	13
PHẦN II: THỰC HIỆN THIẾT KẾ MẠCH	14
1. Thiết kế và thực hiện phần cứng :	14
1.1. Block diagram:	14
1.2 Schematic:	15
2. Thiết kế phần mềm:	16
3. Kết quả thực hiện:	16
Thi công:	17
PHẦN III: KẾT LUẬN	18
Tài liệu tham khảo:	18

PHẦN I: GIỚI THIỆU ĐỀ TÀI

1. Tổng quan về đề tài

1.1. Tổng quan:

Đề tài "Bật tắt dãy đèn LED bằng giọng nói" tập trung vào việc phát triển một hệ thống giúp người dùng điều khiển đèn LED thông qua giọng nói mà không cần sử dụng các công tắc hoặc bộ điều khiển truyền thống. Thay vì phải tìm và nhấn các nút bật/tắt trên bảng điều khiển hoặc sử dụng điều khiển từ xa, người dùng có thể sử dụng giọng nói để ra lệnh cho hệ thống điều khiển đèn LED.

Để thực hiện điều này, đề tài đòi hỏi sự kết hợp giữa công nghệ nhận dạng giọng nói, xử lý ngôn ngữ tự nhiên để vi xử lý ESP32 có thể xử lý và truyền dữ liệu theo giao tiếp UART cho Atmega328p điều khiển đèn LED. Ngoài ra, hệ thống còn tích hợp thêm khả năng tự động bật khi có đèn khi có người. Hệ thống cần có khả năng nhận dạng và hiểu được lệnh bằng giọng nói từ người dùng. Điều này yêu cầu sử dụng các thuật toán và mô hình học máy để phân tích và dịch giọng nói thành các lệnh cụ thể như "bật đèn trong bao nhiêu giây," "tắt đèn trong thời gian bao nhiêu", hay có người ở trong phòng.

1.2 Mục đích:

- Giúp cho việc bật, tắt đèn trở nên nhanh gọn hơn
- Tiện lợi, dễ dàng sử dụng, điều khiển từ xa, điều khiển hẹn giờ. Tự động bật khi có người giúp tăng khả năng tiện lợi.
- Nghiên cứu về các tính năng, chân PORT, của 2 vi xử lý ESP32 và Atmega328p.
- Có thể sử dụng để bật tắt đèn từ xa, đặc biệt ở những nơi ẩm ướt như phòng tắm, nhà vệ sinh,... hạn chế việc phải chạm tay ướt vào công tắc, đảm bảo an toàn về điện.
- Chức năng hẹn giờ có thể được sử dụng để chiếu sáng theo yêu cầu tại các trang trại chăn nuôi, những khu vườn thông minh,... nhằm tiết kiệm điện năng mà vẫn đảm bảo năng suất.

1.3 Phân công công việc:

Nhóm đã lên kế hoạch và phân chia công việc cụ thể

SMART LED PROJECT					
STT	Công việc thực hiện	Người thực hiện	Ngày bắt đầu	Ngày kết thúc	Thời gian thực hiện (Ngày)
1	Xác định các mô tả hệ thống (System Specification + Design Issues)	Cao Bằng	14-10-2023	21-10-2023	7
2	Lập kế hoạch	Minh Vũ	14-10-2023	18-10-2023	4
3	Thiết kế phần cứng	Dương	21-10-2023	10-11-2023	20
4	Lập trình ATmega328P	Long Vũ, Minh Vũ	21-10-2023	10-11-2023	20
5	Lập trình giao tiếp ESP32 với MQTT	Dương, Cao Bằng, <u>Thế Bằng</u>	21-10-2023	10-11-2023	20
6	Lập trình thuật toán Speech to text	<u>Thế Bằng</u>	21-10-2023	30-10-2023	9
8	Thi công + Kiểm tra sản phẩm + Sửa lỗi	Cả nhóm	11-11-2023	20-11-2023	9
9	Tổng hợp + viết báo cáo	Long Vũ	20-11-2023	25-11-2023	5

2. Đặc tả hệ thống (System Specification):

2.1. Product Specification:

- Functions: Điều khiển bật/tắt đèn bằng giọng nói có tích hợp chức năng hẹn giờ chiếu sáng và tự động bật/tắt khi phát hiện có người.
- INPUT: Giọng nói đến laptop/điện thoại, sau đó được truyền lên MQTT và truyền xuống module ESP32 thông qua WI-FI, IC Atmega328P, Module RTC DS3231
- OUTPUT: Các mức logic:
 - + Mức 0: Đèn tắt
 - + Mức 1: Đèn sáng
- User interface: Led, micro.
- External interface: UART, I2C.

2.2. Đặc điểm kĩ thuật (Engineering Specification):

- Atmega328P, cảm biến âm thanh, ESP32-WROOM-32D, cảm biến chuyển động PIR, RTC DS3231, đèn dây tóc loại 220V (có thể dùng các loại đèn khác thay thế)

- Requirements:

- + Xử lý theo thời gian thực.
- + Có thể bật/tắt đèn và hẹn giờ chiếu sáng theo yêu cầu người dùng.
- + Tự động bật khi có người ở trong phòng
- + Truyền dữ liệu từ ESP32-WROOM-32D đến Atmega328P qua UART và truyền dữ liệu từ Atmega328p với RTC3231 qua I2C.

Chi phí:

Tên linh kiện	Giá thành
ESP32-WROOM-32D	126 000 đ
IC Atmega328P	69 000 đ
Cảm biến chuyển động PIR	16 000 đ
RTC3231	53 000 đ
LED màu trắng (5mm)	300 đ/cái
Điện trở các loại	56 đ/con
1 testboard	4700 đ/cái
Dây cắm board	15 000 đ/30 sợi
Pin 9V	9500 đ/cái
Tổng	301 800 đ

2.3. Đặc điểm phần cứng (Hardware Specification):

2.3.1 Mô tả hệ thống

Cảm biến âm thanh nhận diện tín hiệu tiếng nói của người dùng, sau đó thu dữ liệu qua micro và gửi lên Broker/Server. Dữ liệu từ Broker/Server giao tiếp với ESP32-WROOM-32D qua giao thức MQTT. Sau đó dữ liệu tiếp tục được truyền đến vi xử lý Atmega328P qua UART để thực hiện bật/tắt đèn, hẹn giờ chiếu sáng theo yêu cầu người dùng và tự động bật tắt đèn khi có người dùng.

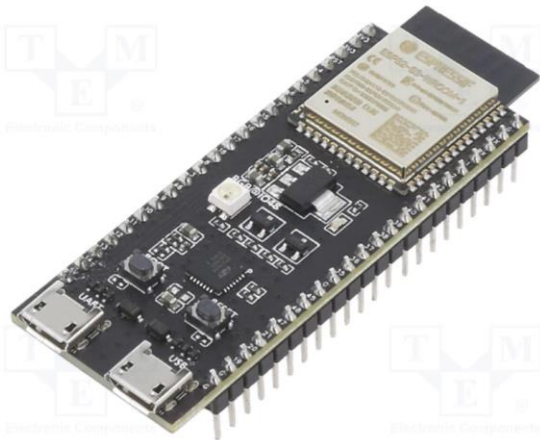
➤ Danh sách linh kiện sử dụng:

1. ATmega328P
2. ESP32-WROOM-32D
3. Micro
4. RTC3231
5. Cảm biến hồng ngoại

6. Micro
7. Điện trở 220 Ω
8. 2 breadboard
9. Dây cắm đực cái
10. Pin
11. Cáp nạp

➤ **Các hình ảnh và thông số linh kiện:**

Vi điều khiển ESP32-WROOM-32D



Vi điều khiển ESP32-WROOM-32D cung cấp cho người dùng:

- 448 KBytes ROM cho booting và các tính năng của lõi chip.
- 520 KBytes SRAM trên chip dùng cho dữ liệu và các lệnh instruction.

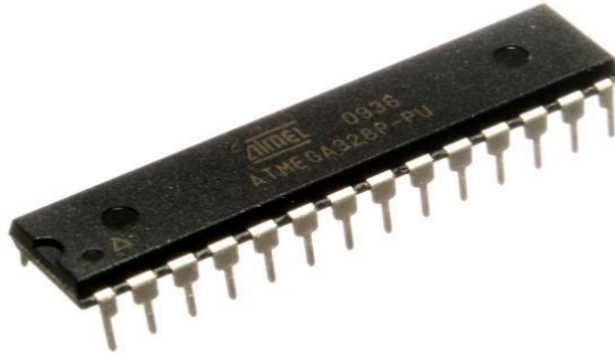
Giao tiếp ngoại vi:

- Bộ chuyển đổi ADC 12 bit, 16 kênh
- Bộ chuyển đổi 8-bits DAC: 2 kênh
- 10 chân để giao tiếp với cảm biến chạm (touch sensor)

Ngoài ra, ESP32-WROOM-32D là một module với nhiều tính năng cải tiến khi hỗ trợ thêm các tính năng Bluetooth và Bluetooth Low Energy (BLE) bên cạnh tính năng WiFi.

Module hỗ trợ các chuẩn giao tiếp SPI, UART, I2C và I2S và có khả năng kết nối với nhiều ngoại vi như các cảm biến, các bộ khuếch đại, thẻ nhớ (SD card),...

Vi điều khiển ATmega328P:



Bộ nhớ:

Vi điều khiển Atmega328 tiêu chuẩn cung cấp cho người dùng:

- 32KB bộ nhớ Flash: những đoạn lệnh bạn lập trình sẽ được lưu trữ trong bộ nhớ Flash của vi điều khiển.
- 2KB cho SRAM (Static Random Access Memory): giá trị các biến khai báo khi lập trình sẽ lưu ở đây.
- 1KB cho EEPROM (Electrically Erasable Programmable Read Only Memory): Đọc và ghi dữ liệu trên SRAM.
- Tốc độ xung nhịp 16 Mhz
- Nhiệt độ tối thiểu và tối đa -40 độ C đến 105 độ C.
- Số chân I/O kỹ thuật số: 14 chân
- Có ba bộ định thời (timer), hai bộ 8 bit và một 16 bit
- Số chân I/O: 23 chân
- Số kênh PWM: 6 chân
- Điện áp hoạt động: Từ 1.8V DC đến 5.5V DC

ATmega328P có cấu hình 28 chân với chức năng cụ thể như sau:

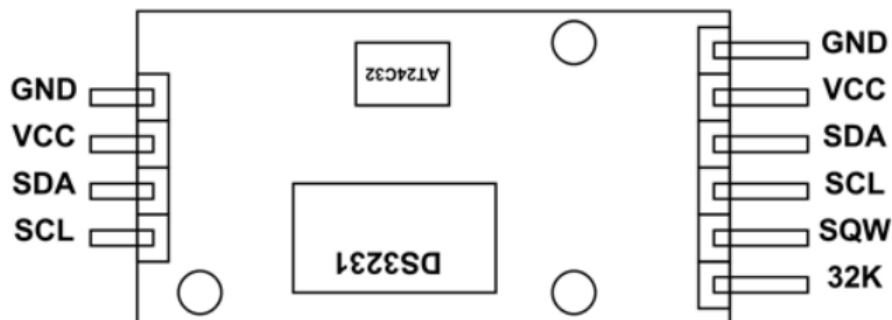
(PCINT14/RESET) PC6	1	28	PC5 (ADC5/SCL/PCINT13)
(PCINT16/RXD) PD0	2	27	PC4 (ADC4/SDA/PCINT12)
(PCINT17/TXD) PD1	3	26	PC3 (ADC3/PCINT11)
(PCINT18/INT0) PD2	4	25	PC2 (ADC2/PCINT10)
(PCINT19/OC2B/INT1) PD3	5	24	PC1 (ADC1/PCINT9)
(PCINT20/XCK/T0) PD4	6	23	PC0 (ADC0/PCINT8)
VCC	7	22	GND
GND	8	21	AREF
(PCINT6/XTAL1/TOSC1) PB6	9	20	AVCC
(PCINT7/XTAL2/TOSC2) PB7	10	19	PB5 (SCK/PCINT5)
(PCINT21/OC0B/T1) PD5	11	18	PB4 (MISO/PCINT4)
(PCINT22/OC0A/AIN0) PD6	12	17	PB3 (MOSI/OC2A/PCINT3)
(PCINT23/AIN1) PD7	13	16	PB2 ($\overline{\text{SS}}$ /OC1B/PCINT2)
(PCINT0/CLKO/ICP1) PB0	14	15	PB1 (OC1A/PCINT1)

DS3231



RTC có nghĩa là Đồng hồ thời gian thực. Mô-đun RTC chỉ đơn giản là các hệ thống ghi nhớ TIME và DATE có thiết lập pin trong trường hợp không có nguồn điện bên ngoài sẽ giữ cho mô-đun hoạt động. Điều này giúp TIME và DATE luôn được cập nhật. Vì vậy, chúng tôi có thể có THỜI GIAN và NGÀY chính xác từ mô-đun RTC bất cứ khi nào muốn.

Cấu hình chân DS3231 RTC



Tên chân	Mô tả
VCC	Kết nối với cực dương của nguồn
GND	Kết nối với mặt đất
SDA	Chân dữ liệu nối tiếp
SCL	Các chân đồng hồ nối tiếp
SQW	Chân đầu ra sóng vuông
32K	Đầu ra dao động 32K

Cảm biến chuyển động PIR



Phạm vi phát hiện : góc 360 độ hình nón, độ xa tối đa 6m.

Nhiệt độ hoạt động : 32-122 ° F (050 ° C)

Điện áp hoạt động : DC 3.8V - 5V

Mức tiêu thụ dòng: $\leq 50 \mu A$

Thời gian báo: 30 giây có thể tùy chỉnh bằng biến trở.

Điện trở các giá trị 1k Ω , 4.7k Ω , 300 Ω



Thông số kỹ thuật:

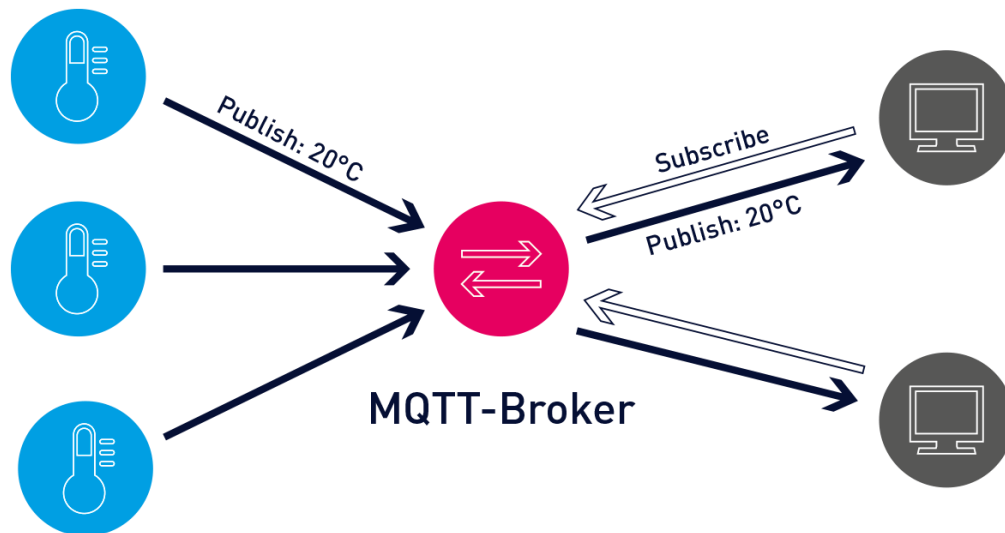
- Model: 220 Ω , 1/4W
- Nhiệt độ hoạt động: -55°C – 155°C

- Linh kiện xuyên lỗ: 0.5mm
- Loại: Điện trở cố định

2.4. Đặc tả phần mềm (Software Specification):

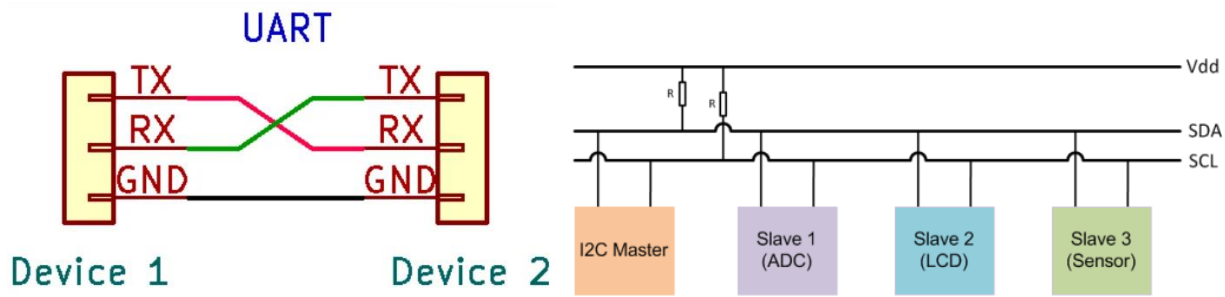
- Sử dụng mô thức MQTT, I2C, UART

Mô thức MQTT là một giao thức nhắn tin dựa trên các tiêu chuẩn hoặc một bộ các quy tắc được sử dụng cho việc giao tiếp máy với máy. Cảm biến thông minh, thiết bị đeo trên người và các thiết bị Internet vạn vật (IoT) khác thường phải truyền và nhận dữ liệu qua mạng có tài nguyên và băng thông hạn chế. Các thiết bị IoT này sử dụng MQTT để truyền dữ liệu vì giao thức này dễ triển khai và có thể giao tiếp dữ liệu IoT một cách hiệu quả. MQTT hỗ trợ nhắn tin giữa các thiết bị với đám mây và từ đám mây đến thiết bị.



Mô thức UART và I2C về cơ bản, trong giao tiếp UART, có 2 dây kết nối là RX và TX tương ứng với chức năng nhận dữ liệu và truyền dữ liệu, vì chỉ có 1 dây nên dữ liệu sẽ lần lượt truyền từng bit, từng bit 1, nên nó có kiểu truyền dữ liệu nối tiếp và truyền từ 1 khung dữ liệu cần truyền, ta sẽ đưa thêm vào đầu và cuối của khung này bit bắt đầu, bit chẵn lẻ (để kiểm tra xem dữ liệu có gửi đủ không), bit kết thúc. Khi đó 1 khung truyền sẽ được hình thành và đưa lên đường dây TX và bắt đầu quá trình truyền nối tiếp.

Đối với Giao tiếp I2C bao gồm quá trình truyền nhận dữ liệu giữa các thiết bị chủ tớ, hay Master - Slave. Master và Slave được kết nối với nhau bằng hai đường bus SCL và SDA đều hoạt động ở chế độ Open Drain, nghĩa là bất cứ thiết bị nào kết nối với mạng I2C này cũng chỉ có thể kéo 2 đường bus này xuống mức thấp (LOW), nhưng lại không thể kéo được lên mức cao.



- Nhận và xử lý tín hiệu âm thanh.

Sau khi âm thanh đã được xử lý và được gửi dữ liệu âm thanh qua kết nối MQTT từ web server đến ESP. Một chủ đề MQTT sẽ được sử dụng để định danh cho dữ liệu âm thanh và ESP có thể đăng ký nhận dữ liệu từ chủ đề đó.

Khi ESP nhận được dữ liệu âm thanh từ web server MQTT, âm thanh trên ESP có thể xử lý theo yêu cầu của ứng dụng của người dùng. Điều này có thể bao gồm phân tích, giải mã hoặc tái tạo âm thanh trên ESP.

- Điều khiển relay bật/tắt đèn và hẹn giờ chiếu sáng.

2.5. Test Specification:

Test Specification mô tả chi tiết về các yêu cầu kiểm thử và các kịch bản kiểm thử cần thực hiện để đảm bảo rằng một hệ thống, ứng dụng hoặc sản phẩm phần mềm hoạt động đúng theo các yêu cầu đã định. Ở đây nhóm chúng em đã thực hiện các bài kiểm tra thử để đánh giá chất lượng hoạt động của hệ thống:

- + Kiểm tra cảm biến trong cả hai trường hợp môi trường yên tĩnh và môi trường có âm thanh.
- + Kiểm tra Relay, LCD, RTC DS3231.
- + Kiểm tra độ ổn định của đèn.
- + Kiểm tra giao thức MQTT.

- + Kiểm tra Atmega328p, ESP có hoạt động bình thường
- + Kiểm tra độ chính xác của cảm biến chuyển động PIR và hiệu chỉnh phù hợp.

3. Vấn đề thiết kế (Design Issues)

3.1. Constraints

- Hệ thống có giá thành thấp (< 500.000 VNĐ).
- Thời gian đáp ứng điều khiển < 1ms.
- Nhận diện và tín hiệu tiếng nói của người dùng khoảng cách 10m.
- Phát hiện có người trong phạm vi bán kính 10m (chế độ tự động)

3.2. Functional issues

- Hệ thống phải có chế độ bảo vệ khi ngắn mạch.
- Hệ thống phải có cầu chì bảo vệ hệ thống khi quá dòng.
- Các sự cố dẫn đến hư hỏng đèn.

3.3. Real-time issues

- Hệ thống có real-time/soft real-time: delay < 1ms.

3.4. Concurrent issues

- Nhận diện và đọc dữ liệu tiếng nói của người dùng.
- Gửi dữ liệu lên Broker/Server.
- Nhận dữ liệu từ Broker/Server qua giao thức MQTT.
- Ghi dữ liệu vào bộ nhớ.
- Phân tích lệnh và điều khiển bật/tắt đèn theo yêu cầu.

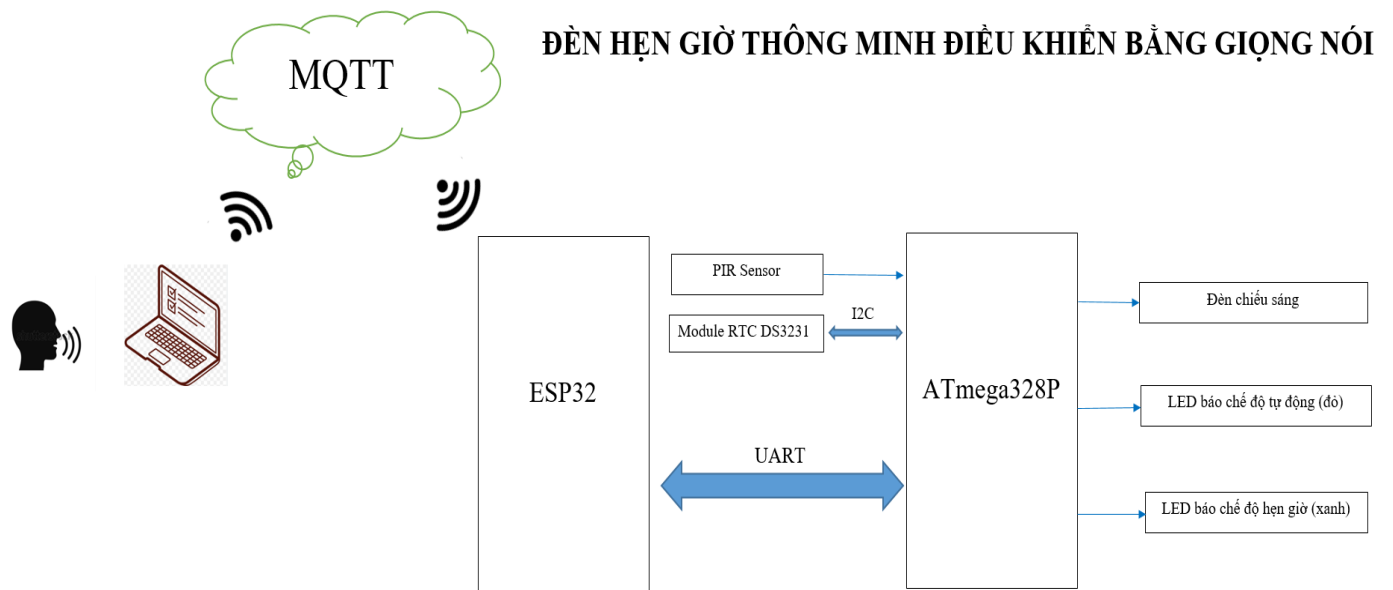
3.5. Reactive issues:

- Hệ thống luôn bật.
- Tương tác liên tục với môi trường hoạt động.
- Bắt đầu thực hiện đáp ứng yêu cầu của người dùng như nhận lệnh.
- Các sự kiện mang tính định kỳ: Hệ thống cần một lịch trình để nắm bắt đầy đủ các sự kiện.

PHẦN II: THỰC HIỆN THIẾT KẾ MẠCH

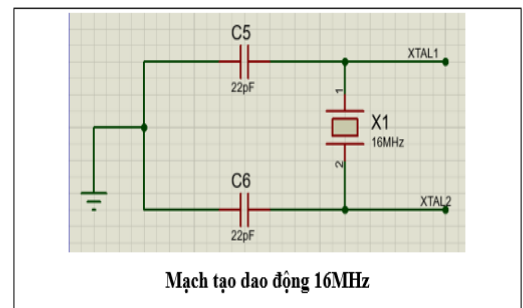
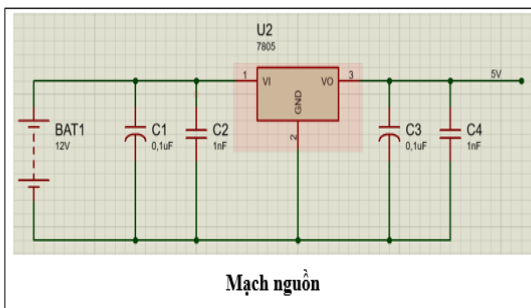
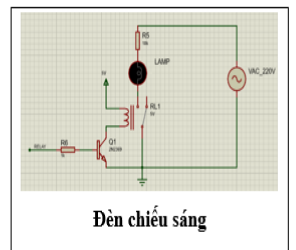
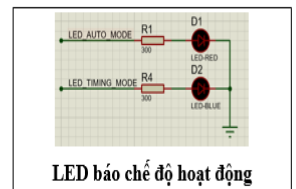
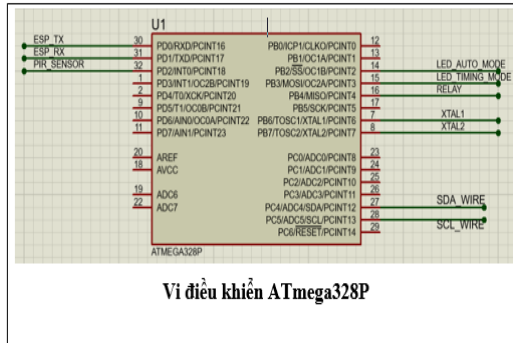
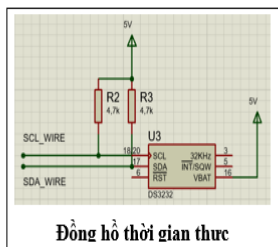
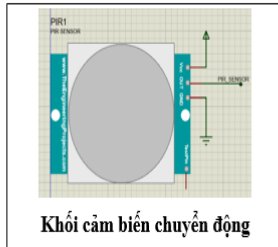
1. Thiết kế và thực hiện phần cứng :

1.1. Block diagram



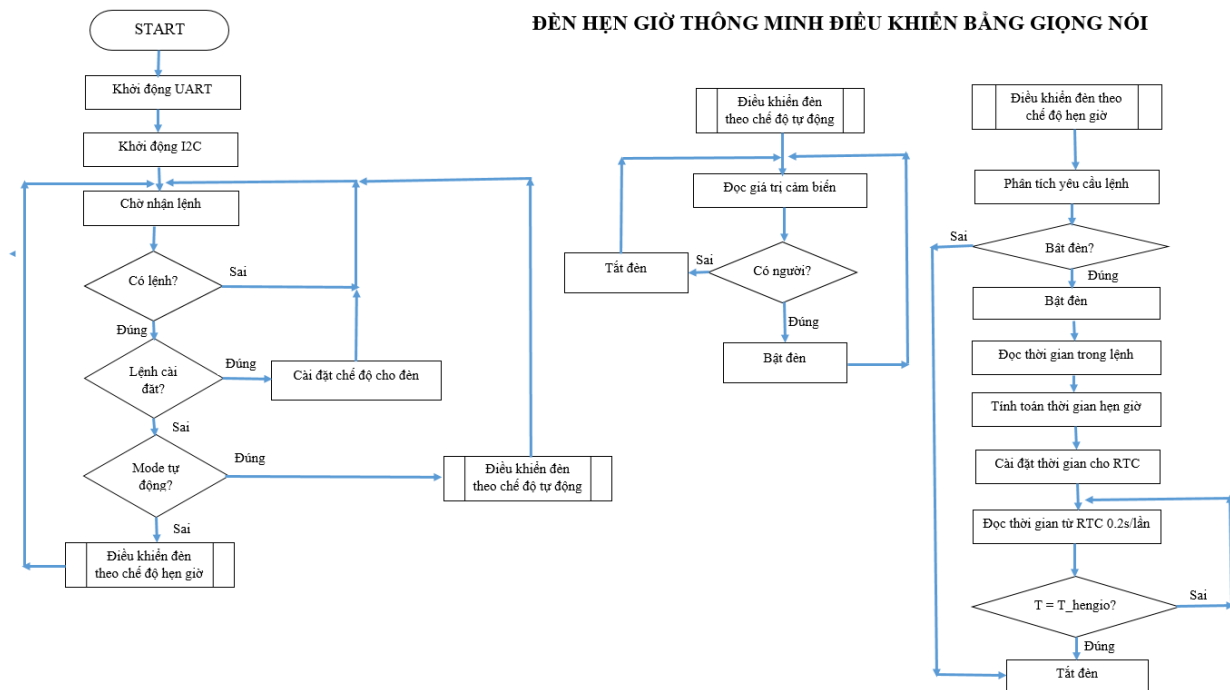
1.2 Schematic

ĐÈN HẸN GIỜ THÔNG MINH ĐIỀU KHIỂN BẰNG GIỌNG NÓI



2. Thiết kế phần mềm:

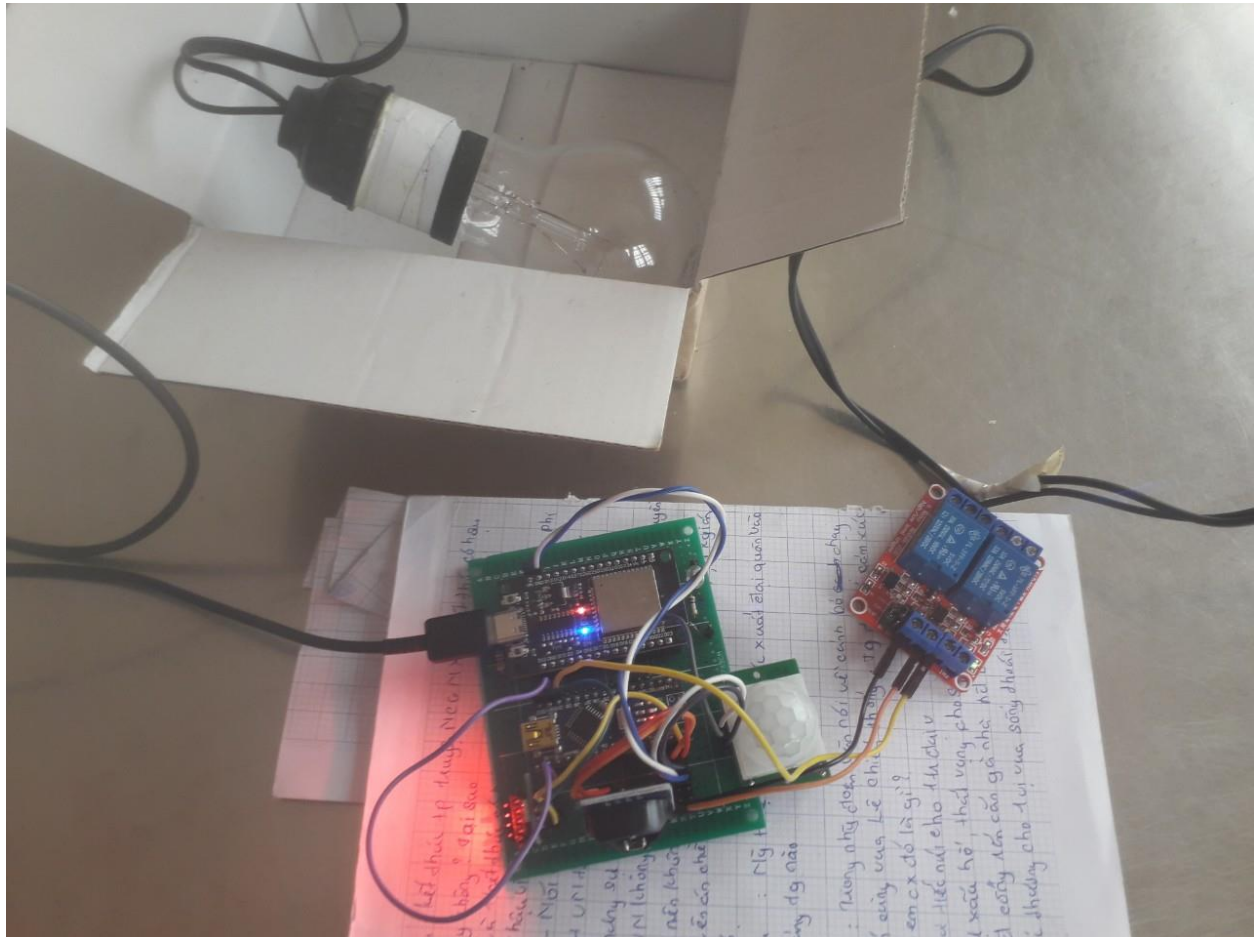
❖ Lưu đồ giải thuật (Flow chart)



❖ Code xem tại Github: https://github.com/bangthe2222/ESD_Project_HCMUT

3. Kết quả thực hiện

❖ Thi công



Sản phẩm thực tế

❖ Link Video demo sản phẩm: <https://youtu.be/dPY-e3a7Rt4>

PHẦN III: KẾT LUẬN

Trong báo cáo này, chúng ta đã xem xét và phân tích chi tiết về thiết kế của Hệ thống đèn hẹn giờ thông minh điều khiển bằng giọng nói. Qua quá trình nghiên cứu và phát triển, chúng ta đã đạt được một hệ thống linh hoạt và tiện ích, giúp tối ưu hóa việc sử dụng đèn và tạo ra trải nghiệm thoải mái cho người sử dụng.

Hệ thống không chỉ đáp ứng được nhu cầu cơ bản của việc điều khiển đèn mà còn tích hợp thêm chức năng điều khiển từ xa bằng giọng nói, giúp người dùng tương tác một cách tự nhiên và thuận tiện. Điều này mang lại sự hiệu quả cao về tiện ích trong việc sử dụng hiệu quả và tiết kiệm năng lượng.

Không chỉ tiết kiệm năng lượng thông qua chức năng hẹn giờ thông minh, hệ thống còn được tích hợp thêm chức năng tự động bật/tắt đèn khi phát hiện có người, từ đó giúp cho việc sử dụng đèn được hiệu quả và đa dạng hơn.

TÀI LIỆU THAM KHẢO

1. Link: <https://arduino.vn/bat-tat-den-bang-tieng-vo-tay-su-dung-arduino/>
2. Data sheet ATmega328P
https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf
3. Cùng học AVR: http://www.hocavr.com/2018/06/home_12.html