



DEPARTMENT OF COMPUTER AND COMMUNICATION
ENGINEERING
FACULTY OF ELECTRICAL AND ELECTRONICS ENGINEERING
HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY AND
EDUCATION

<http://cce.hcmute.edu.vn>

Email: cce@hcmute.edu.vn

TDD-FDP-CCE

Design Project (Technical Design Document)

Project Name	MÔ HÌNH HỆ THỐNG GIÁM SÁT THÂN NHIỆT VÀ CẢNH BÁO		
Student	[1] LÊ CÔNG BẮC [2] ĐÀO NGỌC MINH HUY	ID	20119199 20119045
Major	Electronics and Communication Engineering	Supervisor	Assoc. Prof. Phan Van Ca

MỤC LỤC

1. PHẦN MỞ ĐẦU.....	5
1.1. Nhận diện vấn đề cần giải quyết:.....	5
1.2. PHƯƠNG PHÁP NGHIÊN CỨU.....	5
2. MỤC TIÊU THIẾT KẾ VÀ ĐẶC TẢ YÊU CẦU	6
3. ĐẶC TẢ YÊU CẦU	6
3.1. Yêu cầu người dùng:.....	6
3.2. Yêu cầu kỹ thuật	6
4. THIẾT KẾ HỆ THỐNG	8
4.1. Thiết kế	8
4.1.1. Sơ đồ khối	8
4.2. Lựa chọn linh kiện và thi công hệ thống:.....	8
4.2.1. Vi điều khiển ESP32-WROOM-32 (Khối xử lý trung tâm): [1].....	8
4.2.2. Khối cảm biến.....	13
4.2.2.1. Cảm biến thân nhiệt hồng ngoại không tiếp xúc (MLX90614) [2].....	13
4.2.2.2. Cảm biến thân nhiệt chuyển động (HC-SR501) [3].....	15
4.2.3. Khối chấp hành (Hiển thị OLED và Buzzer).....	16
4.2.3.1. Màn hình OLED 0.96" [4]	16
4.2.3.2. Buzzer [5].....	18
4.2.3.3. Điện trở	19
4.2.3.4. Module Laser (KY-008) [6]	20
4.2.4. Khối điều khiển.....	21
4.3. Phân tích các khối chức năng.....	22
4.3.1. Khối điều khiển (Khối cấp nguồn).....	23
4.3.2. Khối xử lý trung tâm.....	24
4.3.3. Khối chấp hành	25
4.3.4. Khối cảm biến.....	28
4.4. Lưu đồ thuật toán:	29
4.5. Chương trình nạp vào ESP32.....	30
5. KẾT LUẬN.....	35
5.1. Kết quả và đánh giá.....	35
5.2. Hướng phát triển	35

6.	SẢN PHẨM CUỐI CÙNG	36
7.	PHỤ LỤC.....	38
7.1.	Bảng giá	38
7.2.	Biểu đồ làm việc	38
7.3.	Bảng phân công.....	39

TÀI LIỆU THAM KHẢO

- [1] "ALLDATASHEET," [Online]. Available: <https://pdf1.alldatasheet.com/datasheet-pdf/view/1243003/ESPRESSIF/ESP32.html>. [Accessed 12 2022].
- [2] Melexis, "ALLDATASHEET," [Online]. Available: <https://pdf1.alldatasheet.com/datasheet-pdf/view/218975/ETC2/MLX90614.html>. [Accessed 12 2022].
- [3] "COMPONENTS101," [Online]. Available: <https://components101.com/sensors/hc-sr501-pir-sensor>. [Accessed 12 2022].
- [4] "ARDUINOVN," [Online]. Available: <http://arduino.vn/bai-viet/1503-gioi-thieu-man-hinh-oled-096-inch-i2c>. [Accessed 12 2022].
- [5] "DatasheetPDF," [Online]. Available: <https://datasheetspdf.com/datasheet/KY-012.html>. [Accessed 12 2022].
- [6] "DatasheetPDF," [Online]. Available: <https://datasheetspdf.com/pdf/1415012/AZ-Delivery/KY-008/1>. [Accessed 12 2022].

MỤC LỤC BẢNG

Bảng 1:	Chức năng chân ESP32-WROOM-32	12
Bảng 2:	Bảng giá tiền sản phẩm	38
Bảng 3:	Biểu đồ làm việc	38
Bảng 4:	Bảng phân công.....	39

MỤC LỤC HÌNH ẢNH

Hình 1: Sơ đồ khối ESP32-WROOM-32 (8MB)	9
Hình 2: Sơ đồ chân ESP32	10
Hình 3: Sơ đồ khối MLX90614.....	13
Hình 4: Sơ đồ chân MLX90614.....	13
Hình 5: Sơ đồ chức năng chân MLX90614	14
Hình 6: Sơ đồ khối HC-SR501	15
Hình 7: Sơ đồ khối OLED 128x64 (.96").....	16
Hình 8: Màn hình OLED 128x62 (0.96")	17
Hình 9: Sơ đồ khối Buzzer	18
Hình 10: Buzzer (KY-012).....	18
Hình 11: Sơ đồ khối điện trở.....	19
Hình 12: Sơ đồ khối Module Laser (KY-008)	20
Hình 13: Sơ đồ chân Module Laser.....	20
Hình 14: Sơ đồ khối điều khiển LED	21
Hình 15: Sơ đồ chân LED	21
Hình 16: Sơ đồ kết nối các khối.....	22
Hình 17: Khối cấp nguồn	23
Hình 18: Adapter 12V.....	23
Hình 19: Mạch chuyển đổi điện áp 12V	24
Hình 20: Khối xử lý trung tâm	24
Hình 21: Sơ đồ kết nối khối chấp hành	25
Hình 22: Điều khiển LED.....	26
Hình 23: Buzzer	26
Hình 24: Màn hình OLED.....	27
Hình 25: Module Laser	27
Hình 26: PIR_HC-SR501.....	28
Hình 27: MLX90614.....	28
Hình 28: Lưu đồ thuật toán.....	29
Hình 29: Dữ liệu lên Firebase.....	36
Hình 30: Dữ liệu lên Google Sheet.....	36
Hình 31: Mô hình sản phẩm.....	37
Hình 32: Mô hình sản phẩm.....	37

1. PHẦN MỞ ĐẦU

1.1. Nhận diện vấn đề cần giải quyết:

Trong tình hình dịch bệnh truyền nhiễm đang diễn biến phức tạp nhiều nơi trên thế giới kể cả Việt Nam, việc một người tiềm ẩn dịch bệnh trong người đi lại ở những nơi đông người như sân bay, bến xe, bến tàu, bệnh viện, ... sẽ làm lây lan nhanh chóng và làm cho việc kiểm soát dịch bệnh trở nên phức tạp hơn. Vì vậy việc lập một vòng kiểm tra và kiểm soát nhiệt độ cơ thể (đa số dịch bệnh truyền nhiễm đều có triệu chứng sốt) ngay từ vòng ngoài để hạn chế bớt những đối tượng hành khách có nguy cơ là trung gian gây lây nhiễm cao.

Đứng trước những yêu cầu cấp thiết này, những người có trách nhiệm muốn trang bị tại những nơi có nguy cơ được kể trên muốn trang bị rất nhiều những thiết bị giám sát thân nhiệt ngay tại những lối vào sảnh chờ của mình nhưng đa số thiết bị giám sát trên thị trường hiện nay đều có hệ thống khá phức tạp, cồng kềnh, chi phí đắt đỏ, yêu cầu cho việc vận hành cao sẽ có rất nhiều khách hàng không đáp ứng được nhu cầu. Nắm bắt được vấn đề này nhóm chúng em đã quyết định tiến hành nghiên cứu và thiết kế một hệ thống bằng những cảm biến và vi điều khiển dựa trên các yêu cầu tối ưu hóa về chi phí, kích thước, độ phức tạp mà nhiều khách hàng đề ra.

1.2. PHƯƠNG PHÁP NGHIÊN CỨU

Trong đề tài “Thiết kế mô hình hệ thống giám sát thân nhiệt và cảnh báo” nhóm thực hiện việc tìm hiểu kiến thức về:

- Lý thuyết liên quan Firebase (Lưu trữ dữ liệu về thân nhiệt và số lượng người có thân nhiệt quá cao trong một ngày để nghiên cứu).
- Thiết kế, thi công hệ thống giám sát thân nhiệt dựa vào các cảm biến.
- Cài đặt ngưỡng giá trị phát ra thông tin cảnh báo cho hệ thống.
- Áp dụng kiến thức để thiết kế Website - Giúp đưa ra dữ liệu tham khảo để khách hàng tránh đến những nơi, những thời điểm có dữ liệu về số lượng người có thân nhiệt cao bất thường (khả năng lây nhiễm cao).

2. MỤC TIÊU THIẾT KẾ VÀ ĐẶC TẢ YÊU CẦU

Thiết kế được hệ thống giám sát thân nhiệt thông qua cảm biến đo thân nhiệt chuyển động. Truyền dữ liệu đo được lên máy tính, Firebase và màn hình LED bằng ESP32 (hoặc ESP8266), nếu giá trị thân nhiệt vượt quá mức quy định ESP32 sẽ gửi một tín hiệu cảnh báo lên thiết bị phát âm thanh cảnh báo đã được cài đặt sẵn.

3. ĐẶC TẢ YÊU CẦU

3.1. Yêu cầu người dùng:

- Thiết bị đo chính xác thân nhiệt của người với sai số không quá cao với thông số đo bằng cách thông thường.
- Tính tiện lợi cao, cho phép người dùng quan sát và thu thập được dữ liệu từ xa thông qua Firebase và Google Sheet.
- Nguyên lý hoạt động đơn giản, ai cũng có thể sử dụng được.
- Tính tự động cao, người dùng không cần thao tác hay tiếp xúc với máy đo.
- Cập nhật thông số nhiệt độ cơ thể người lên google sheet nhanh chóng và chính xác.
- Tính đơn giản: Dễ dàng lắp đặt, sử dụng, bảo hành, sửa chữa khi có sự cố và có thể nâng cấp, cải tạo một cách dễ dàng.
- Phân biệt được nhiệt độ môi trường và nhiệt độ vật thể.
- Dễ dàng theo dõi mức nhiệt đang đo được thông qua màn hình.
- Thiết bị phát ra cảnh báo chính xác ngay thời điểm ghi nhận mức nhiệt độ cao, không phát ra sai đối tượng.
- Sử dụng được bằng mức điện áp 220V ở hộ gia đình.
- Phân biệt được và ghi nhận các mức nhiệt độ của cơ thể người, loại bỏ các mức nhiệt của vật thể khác.
- Tính ứng dụng cao: Thiết bị có thể được ứng dụng trong các hoạt động chăm sóc sức khỏe ở trường học, bệnh viện, công ty, ...

3.2. Yêu cầu kỹ thuật

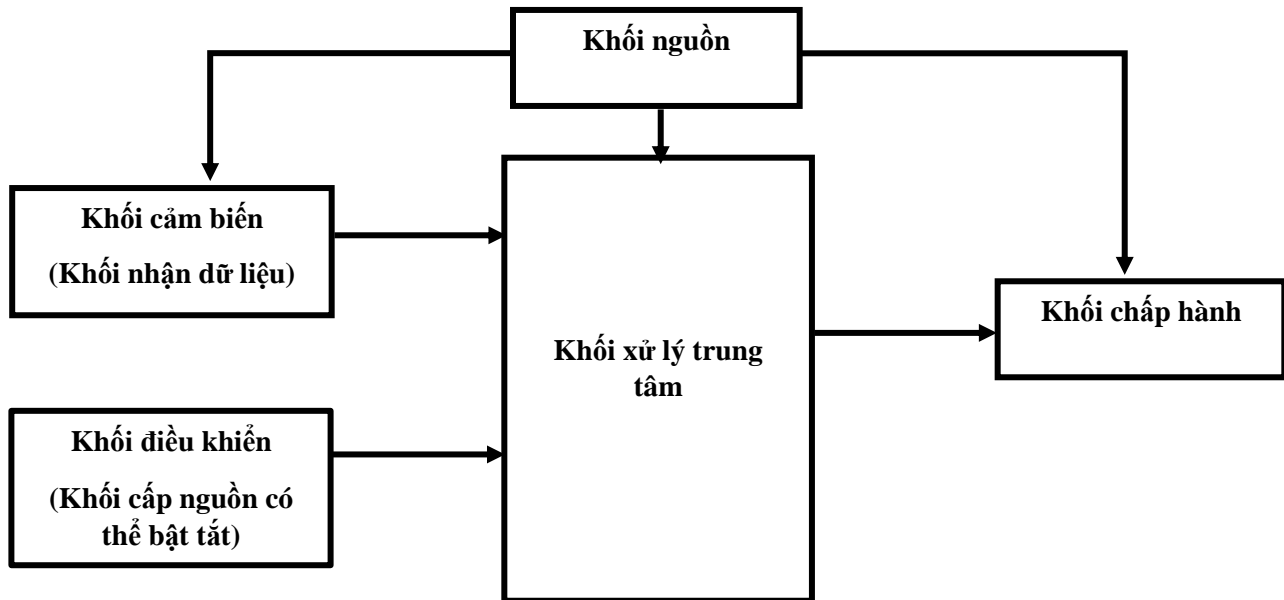
- Chức năng kỹ thuật:
 - Đo nhiệt độ cơ thể người thông qua cảm biến nhiệt hồng ngoại MLX-90614.
 - Hiển thị các thông số nhiệt độ đang đo được lên màn hình OLED với bố cục đơn giản.
 - Tiết kiệm năng lượng bằng cách chỉ cho phép đo thân nhiệt mỗi khi có phát hiện chuyển động thông qua cảm biến chuyển động HC-SR501.

- Tự động bật Led khi phát hiện chuyển động.
 - Thời gian cho phép đo sau khi phát hiện chuyển động là 2s
 - Tự động so sánh với giá trị chuẩn đã được cài đặt để phát ra tín hiệu cảnh báo tự động khi vượt quá mức thân nhiệt bình thường.
 - Đọc dữ liệu đo thân nhiệt và số lượng người thân nhiệt cao gửi lên Firebase thông qua WiFi trong thời gian thực, hiệu quả và chính xác.
 - Phân loại các dữ liệu đã đo được và gửi các mức nhiệt gần nhất với nhiệt độ cơ thể người (từ 34.5°C - 41°C) lên google sheet để ghi nhận lại.
 - Các mức thân nhiệt đo được không bị mất dữ liệu khi mất điện do data được gửi lên google sheet theo thời gian thực.
- Thông số kỹ thuật:
 - Dải đo: $34^{\circ}\text{C} \sim 45^{\circ}\text{C}$
 - Độ chính xác: $\pm 0.4^{\circ}\text{C}$
 - Thời gian gửi tín hiệu: 2s - 5s
 - Nguồn điện áp sử dụng: 5V DC – 2A

4. THIẾT KẾ HỆ THỐNG

4.1. Thiết kế

4.1.1. Sơ đồ khối

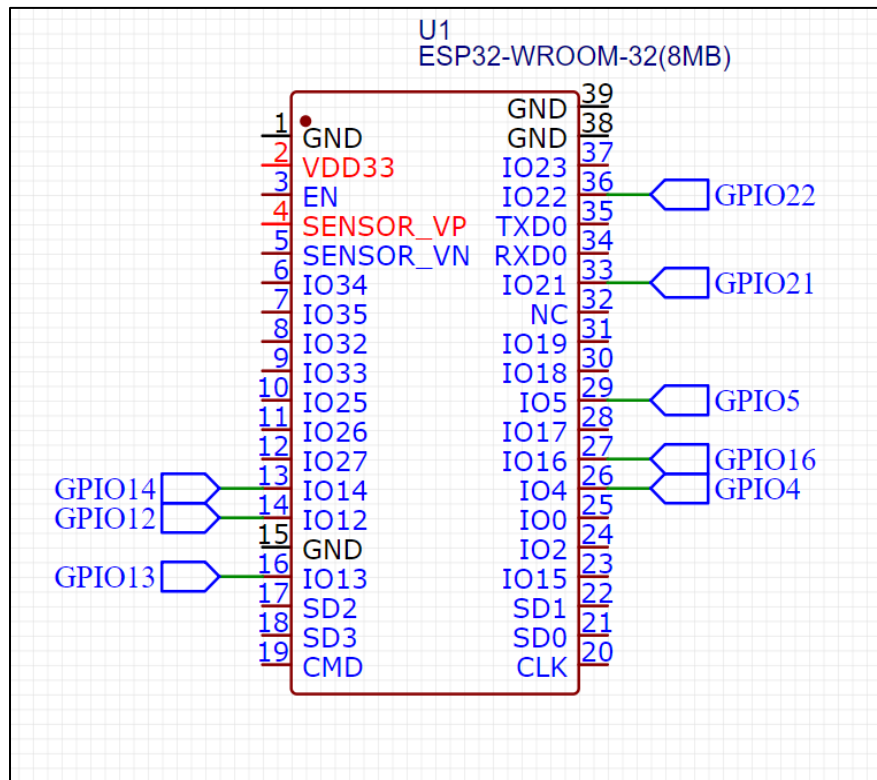


- Chức năng của từng khối:
 - Khối nguồn: Cấp nguồn cho cảm biến, vi điều khiển.
 - Khối cảm biến: Phát hiện đối tượng bằng cách phát hiện chuyển động, đo nhiệt độ bằng hồng ngoại. Khi phát hiện nhiệt độ cao hơn mức chuẩn (bức xạ hồng ngoại thu được thay đổi), cảm biến sẽ xuất dữ liệu nhiệt độ đưa vào vi điều khiển để thực hiện chức năng cảnh báo và gửi dữ liệu.
 - Khối chấp hành: Nhận tín hiệu từ khối xử lý để phát âm thanh cảnh báo và hiển thị ra màn hình OLED.
 - Khối xử lý trung tâm: lấy tín hiệu từ cảm biến, xử lý và gửi dữ liệu lên Google Firebase, Google Sheet và gửi yêu cầu cho khối chấp hành làm việc.
 - Khối điều khiển: Nhận tín hiệu điều khiển từ con người và gửi tín hiệu đến khối xử lý trung tâm.

4.2. Lựa chọn linh kiện và thi công hệ thống:

4.2.1. Vi điều khiển ESP32-WROOM-32 (Khối xử lý trung tâm): [1]

Vì điều khiển ESP32-WROOM-32 (8MB) được chọn làm khối xử lý trung tâm. Để giải quyết một cách tối ưu các yêu cầu đề ra của người dùng như dễ sử dụng, tối ưu các kết nối, chia đa luồng để xử lý các tác vụ, ESP32 còn hỗ trợ việc kết nối Bluetooth, WiFi để thực hiện việc gửi dữ liệu lên Firebase và Google Sheet từ những dữ liệu này người dùng có thể phân tích và đánh giá mức độ nguy hiểm của vùng đang xét. Thiết lập các khối khác nhau dựa trên các chuẩn giao tiếp khác nhau để dễ dàng hơn. Hình ... thể hiện các chân của các khối kết nối đến ESP32.



Hình 1: Sơ đồ khối ESP32-WROOM-32 (8MB)

Khối xử lý trung tâm (ESP32) có những yêu cầu đặt ra là phải đáp ứng về các mặt kỹ thuật gồm nhận dữ liệu từ các nút, dữ liệu nhận được sẽ được xử lý sau đó được gửi lên nơi lưu trữ là Firebase. Ngoài ra màn hình OLED có thể dùng để thực hiện việc hiển thị các số liệu từ sensor.

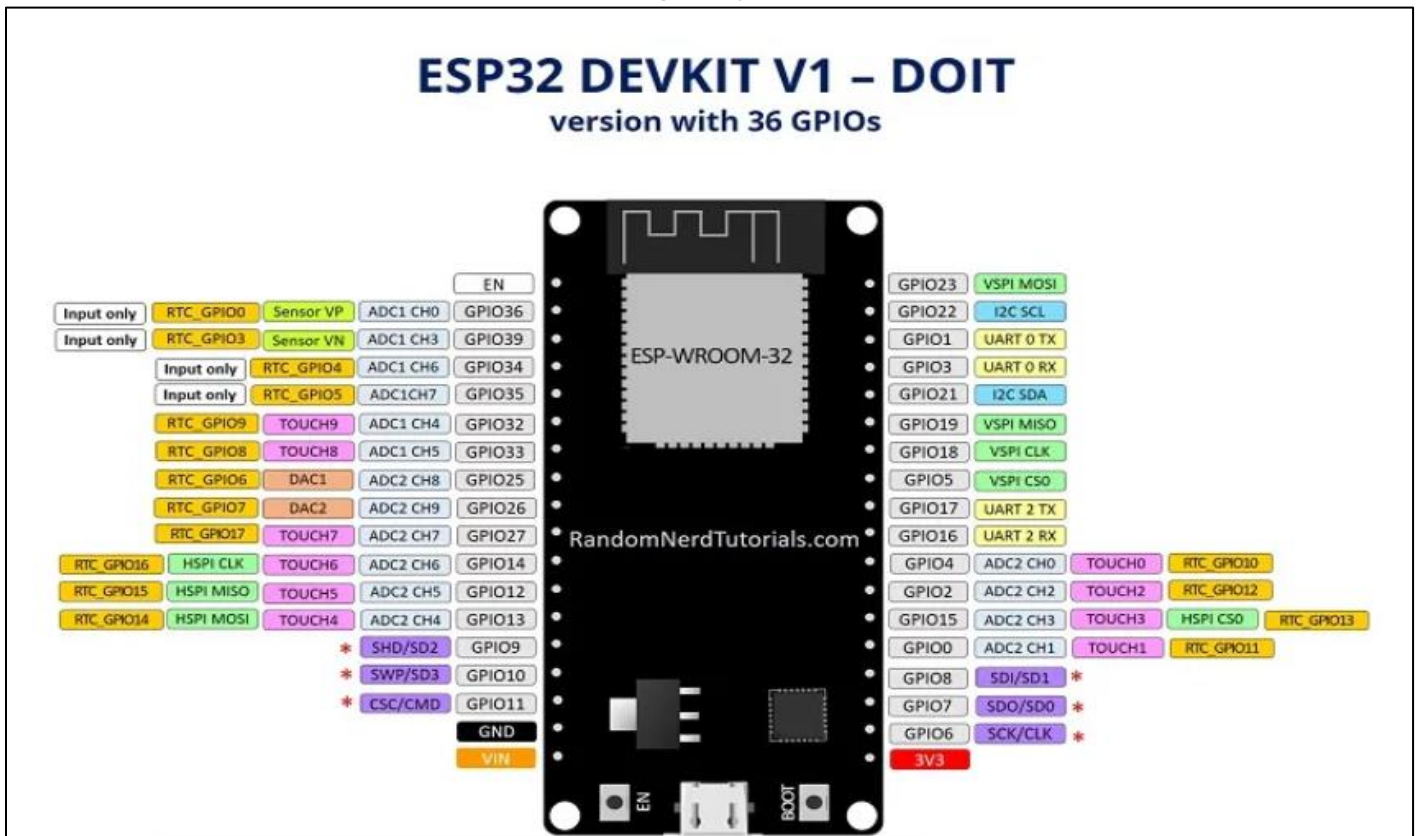
❖ Chức năng ESP32-WROOM-32:

- Là một module với nhiều tính năng cải tiến hơn các module dòng ESP8266 khi hỗ trợ thêm các tính năng Bluetooth và Bluetooth Low Energy (BLE) bên cạnh tính năng WiFi. Sản phẩm sử dụng chip ESP32-D0WDQ6 với 2 CPU có thể được điều khiển độc lập với tần số xung clock lên đến 240 MHz.

- 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),

...

- Cấu trúc, sơ đồ chân và chức năng từng chân:



Hình 2: Sơ đồ chân ESP32

- GPIO (General Purpose Input/Output) là một chức năng ngoại vi cơ bản của vi điều khiển, bao gồm các chân đầu vào và chân đầu ra tín hiệu số được lập trình và điều khiển bởi người dùng.
- Ô **USEFULL** màu xanh là các GPIO có thể sử dụng.
- Ô **USEFULL** màu vàng là có thể sử dụng nhưng cần phải chú ý để tránh xảy ra lỗi không cần thiết.

- Ô **X** màu đỏ là không được khuyến khích sử dụng làm INPUT hay OUTPUT.

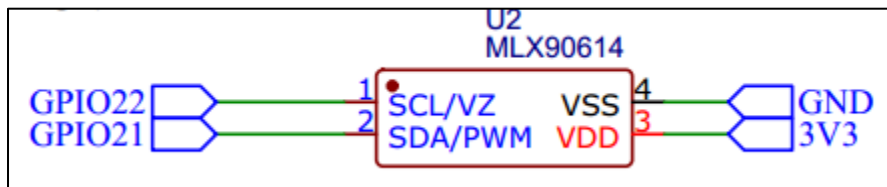
GPIO	INPUT	OUTPUT	FUNCTION
0 (T1)	PULL UP (HIGH)	USEFUL	Cấu hình Mode cho ESP32 phục vụ cho quá trình nạp Code bên trong Esp. Có thể sử dụng làm ADC. Có hỗ trợ chức năng RTC.
1	Nối với Tx của bộ UART0 (UART 0 Tx)	USEFUL	Nạp Code thông qua bộ UART này nên không nên sử dụng 2 chân này cho chức năng ngoại vi (I/O). Mức HIGH.
2 (T2)	USEFUL	USEFUL	Kết nối với LED vi điều khiển, định sẵn mức logic để phục vụ việc nạp Code, hạn chế sử dụng GPIO này. Có thể sử dụng làm ADC. Có hỗ trợ chức năng RTC.
3	USEFUL	RX pin (UART 0 Rx)	Nạp Code thông qua bộ UART này nên không nên sử dụng 2 chân này cho chức năng ngoại vi (I/O). Mức HIGH.
4 (T0)	USEFUL	USEFUL	Định sẵn mức logic để phục vụ việc nạp Code, hạn chế sử dụng GPIO này. Có thể sử dụng làm ADC. Có hỗ trợ chức năng RTC.
5	USEFUL	USEFUL	Định sẵn mức logic (HIGH) để phục vụ việc nạp Code, hạn chế sử dụng GPIO này. Dùng làm VSPI CSO. Mức HIGH.
6 (SCK/CLK)	X	X	Kết nối với SPI flash được tích hợp trên ESP32. Mức HIGH
7 (SDO/SD0)	X	X	Kết nối với SPI flash được tích hợp trên ESP32. Mức HIGH
8 (SDI/SD1)	X	X	Kết nối với SPI flash được tích hợp trên ESP32. Mức HIGH
9 (SHD/SD2)	X	X	Kết nối với SPI flash được tích hợp trên ESP32. Mức HIGH
10 (SWP/SD3)	X	X	Kết nối với SPI flash được tích hợp trên ESP32. Mức HIGH
11 (CSC/CMD)	X	X	Kết nối với SPI flash được tích hợp trên ESP32. Mức HIGH
12 (T5)	USEFUL	USEFUL	Định sẵn mức logic (LOW) để phục vụ việc nạp Code, hạn chế sử dụng GPIO này. Mức logic cao sẽ không khởi động được. Có thể sử dụng làm ADC. Có hỗ trợ chức năng RTC. Dùng làm HSPI MISO.
13 (T4)	USEFUL	USEFUL	Có thể sử dụng làm ADC. Có hỗ trợ chức năng RTC. Dùng làm HSPI MOSI.
14 (T6)	USEFUL	USEFUL	Có thể sử dụng làm ADC. Có hỗ trợ chức năng RTC. Dùng làm HSPI CLK. Mức HIGH.
15 (T3)	USEFUL	USEFUL	Định sẵn mức logic (HIGH) để phục vụ việc nạp Code, hạn chế sử dụng GPIO này. Có thể sử dụng làm ADC. Có hỗ trợ chức năng RTC. Dùng làm HSPI CSO. Mức HIGH.
16	USEFUL	USEFUL	UART 2 Rx.
17	USEFUL	USEFUL	UART 2 Tx.
18	USEFUL	USEFUL	VSPI CLK

19	USEFUL	USEFUL	VSPI MISO
21	USEFUL	USEFUL	I2C có thể sử dụng để đặt làm SDA hoặc SCL
22	USEFUL	USEFUL	I2C có thể sử dụng để đặt làm SDA hoặc SCL
23	USEFUL	USEFUL	VSPI MOSI
25	USEFUL	USEFUL	Có thể sử dụng làm ADC, DAC. Có hỗ trợ chức năng RTC
26	USEFUL	USEFUL	Có thể sử dụng làm ADC, DAC. Có hỗ trợ chức năng RTC
27 (T7)	USEFUL	USEFUL	Có thể sử dụng làm ADC. Có hỗ trợ chức năng RTC
32 (T9)	USEFUL	USEFUL	Có thể sử dụng làm ADC. Có hỗ trợ chức năng RTC
33 (T8)	USEFUL	USEFUL	Có thể sử dụng làm ADC. Có hỗ trợ chức năng RTC
34	USEFUL		Chỉ sử dụng được chức năng Input, Output không thể dùng. Có thể sử dụng làm ADC. Có hỗ trợ chức năng RTC. Tạo tín hiệu PWM.
35	USEFUL		Chỉ sử dụng được chức năng Input, Output không thể dùng. Có thể sử dụng làm ADC. Có hỗ trợ chức năng RTC. Tạo tín hiệu PWM.
36	USEFUL		Chỉ sử dụng được chức năng Input, Output không thể dùng. Có thể sử dụng làm ADC. Có hỗ trợ chức năng RTC. Tạo tín hiệu PWM. Cảm biến Sensor VP
39	USEFUL		Chỉ sử dụng được chức năng Input, Output không thể dùng. Có thể sử dụng làm ADC. Có hỗ trợ chức năng RTC. Tạo tín hiệu PWM. Cảm biến Sensor VP

Bảng 1: Chức năng chân ESP32-WROOM-32

4.2.2. Khối cảm biến

4.2.2.1. Cảm biến thân nhiệt hồng ngoại không tiếp xúc (MLX90614) [2]

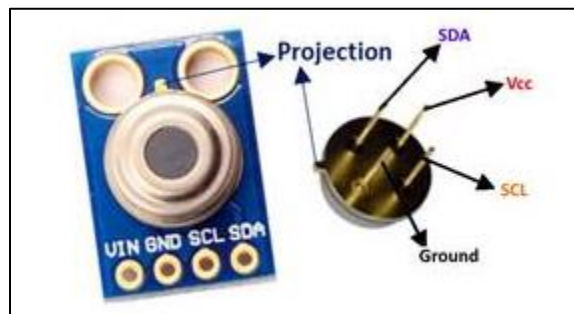


Hình 3: Sơ đồ khối MLX90614

- Sensor MLX90614 được dùng để thu nhận dữ liệu cảm biến thân nhiệt không tiếp xúc bằng hồng ngoại. Với thiết kế nhỏ, công suất thấp. Kết nối với ESP32 thông qua chuẩn giao tiếp I2C.

- Kết nối chân:

- Chân SCL được kết nối với GPIO22
- Chân SDA được kết nối với GPIO21
- Chân VSS được nối đất
- Chân VDD được nối với chân 3V3 của ESP32

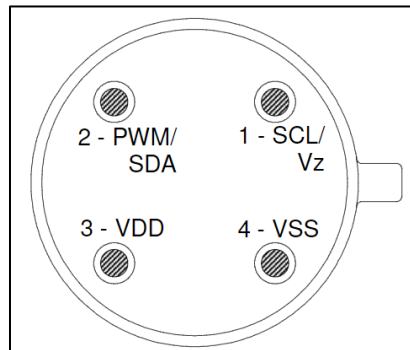


Hình 4: Sơ đồ chân MLX90614

- Giới thiệu và cấu trúc của MLX90614:

- MLX90614 có phạm vi đo từ -40°C đến 125°C với nhiệt độ môi trường và -70°C đến 382.2°C với một đối tượng cụ thể. Sai số ngõ ra 0.14°C .
- MLX90614 bao gồm 2 chip được phát triển và sản xuất bởi Melexis gồm: Chip đo nhiệt hồng ngoại MLX81101 và Bộ điều hòa tín hiệu ASSP MLX90302
- Sử dụng điện áp 5V hoặc pin 3V.
- Có bộ lọc quang học tích hợp giúp kháng ánh sáng mặt trời.

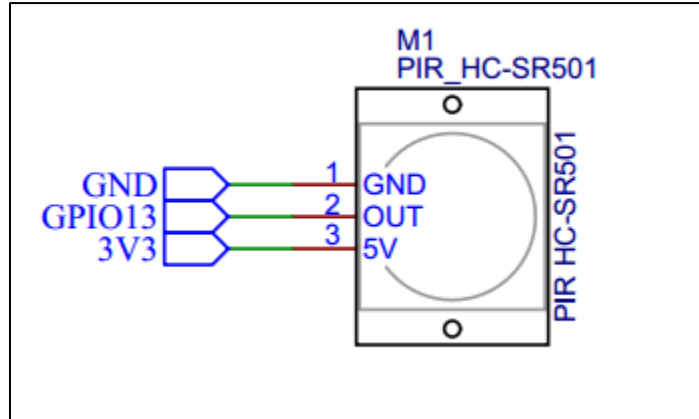
- Chức năng và sơ đồ chân của MLX90614:



Hình 5: Sơ đồ chức năng chân MLX90614

- SCL / Vz: ngõ vào nối tiếp cho các giao thức 2 dây. Zener 5.7V được sử dụng ở chân này để kết nối transistor lưỡng cực với MLX90614, cấp nguồn cho thiết bị từ nguồn 8-16V.
- PWM / SDA: Digital I/O, nhiệt độ của vật được đo xuất ra bằng chân này
- VSS: nối đất
- VDD: điện áp cung cấp ngoài.

4.2.2.2. Cảm biến thân nhiệt chuyển động (HC-SR501) [3]



Hình 6: Sơ đồ khối HC-SR501

- Cảm biến thân nhiệt chuyển động PIR (Passive infrared sensor) HC-SR501 được sử dụng để phát hiện chuyển động của các vật thể phát ra bức xạ hồng ngoại (con người, con vật, các vật phát nhiệt,...), cảm biến có thể chỉnh được độ nhạy để giới hạn khoảng cách bắt xa gần cũng như cường độ bức xạ của vật thể mong muốn, ngoài ra cảm biến còn có thể điều chỉnh thời gian kích trễ (giữ tín hiệu bao lâu sau khi kích hoạt) qua biến trở tích hợp sẵn.

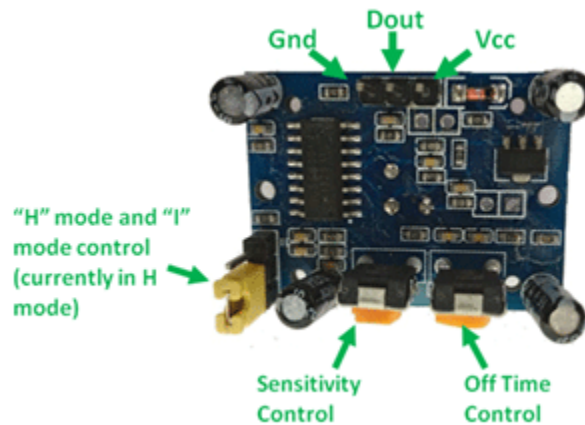
- Kết nối chân với ESP32:

- Chân GND nối đất.
- Chân Out nối với chân GPIO13.
- Chân nguồn nối với nguồn 3V3.

- Giới thiệu và cấu trúc HC-SR501:

- Khoảng điện áp hoạt động: 4.5-20V.
- Ngõ ra mức cao 3.3V và thấp 0V.
- Delay 5-200s (có thể tùy chỉnh).
- Kích thước: 32mm*24mm.
- Khoảng nhiệt độ hoạt động: -15 °C đến 70 °C.
- Có chế độ cảm quang.

- Sơ đồ và chức năng chân:



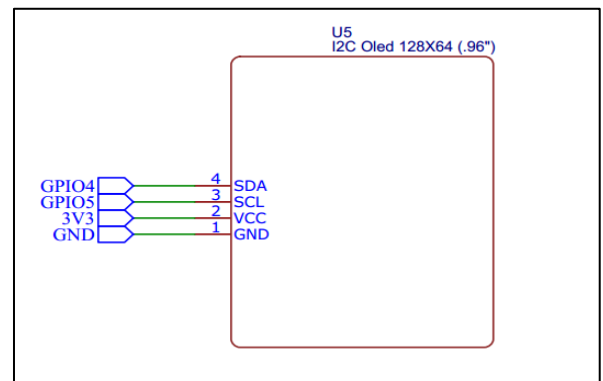
- GND: nối đất
- Dout: tín hiệu ngõ ra
- Vcc: điện áp cung cấp ngoài

4.2.3. Khối chấp hành (Hiển thị OLED và Buzzer)

4.2.3.1. Màn hình OLED 0.96" [4]

- OLED (viết tắt bởi Organic Light Emitting Diode: Diode phát sáng hữu cơ) đang trở thành đối thủ cạnh tranh cũng như ứng cử viên sáng giá thay thế màn hình LCD.

- Màn hình sử dụng giao tiếp I2C cho chất lượng đường truyền ổn định và rất dễ giao tiếp chỉ với 2 chân GPIO.



Hình 7: Sơ đồ khối OLED 128x64 (.96")

- Kết nối chân:
 - Chân SDA nối với chân GPIO4.
 - Chân SCL nối với chân GPIO5.
 - Chân VCC nối với chân 3V3.
 - Chân GND nối với chân GND.

- Giới thiệu và cấu trúc OLED:
 - Khoảng điện áp: 3.3V – 5V
 - Công suất tiêu thụ: 0.04W
 - Độ phân giải: 128*64 pixel
 - Độ rộng: 0.96 inch
 - Giao tiếp: I2C
- Sơ đồ và chức năng chân:

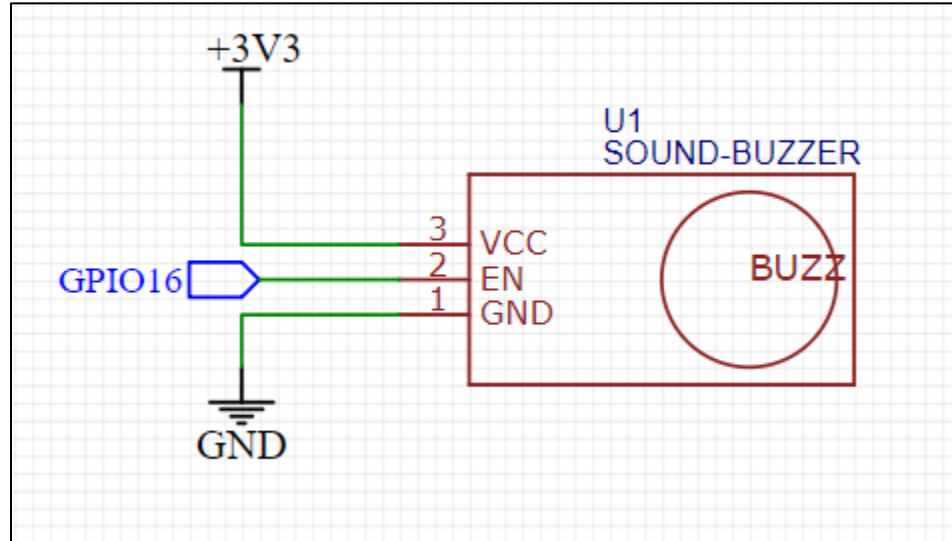


Hình 8: Màn hình OLED 128x62 (0.96")

- VCC: điện áp cung cấp ngoài
- GND: nối đất
- SCL: xung clock
- SDA: truyền dữ liệu

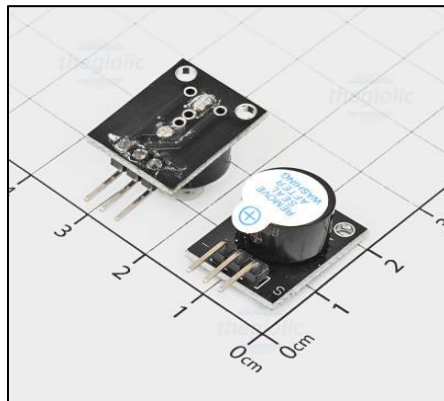
4.2.3.2. Buzzer [5]

- Để có thể cảnh báo nhanh nhất đến cho người dùng nhóm thực hiện sử dụng mô-đun buzzer. Với kích thước nhỏ với khả năng phát ra tiếng cảnh báo lớn và có thể dễ dàng sử dụng cũng như lắp đặt.



Hình 9: Sơ đồ khối Buzzer

- Kết nối các chân của Buzzer trong sơ đồ:
 - Chân GND của Buzzer được nối với GND
 - Chân VCC nối với chân 3V3 của ESP32
 - Chân EN được nối với GPIO16



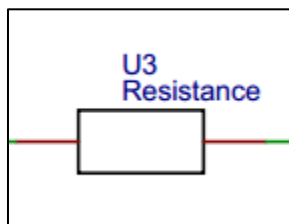
Hình 10: Buzzer (KY-012)

- Chức năng và cấu trúc của Buzzer (KY-012):
 - KY-012 là một mạch buzzer lý tưởng để tăng thêm âm thanh cho dự án của bạn và tương thích khi hoạt động với các bộ vi điều khiển như Arduino. Mạch buzzer này bao gồm một bộ rung áp điện hoạt động, nó tạo ra âm thanh có tần số gần 2.5kHz khi tín hiệu ở mức cao.
 - Để hoạt động, chân EN phải nhận được điện áp để kích hoạt bộ rung. Điều này có thể được thực hiện thông qua lập trình.
- Thông số kỹ thuật:
 - Điện áp hoạt động: 3.5V ~ 5.5V
 - Dòng hoạt động: 30mA / 5VDC
 - Tần số cộng hưởng 2500Hz \pm 300Hz
 - Đầu ra âm thanh tối thiểu 85Db @ 10cm
 - Nhiệt độ làm việc: -20°C ~ 70°C
 - Kích thước: 18.5 x 15 mm

4.2.3.3. Điện trở

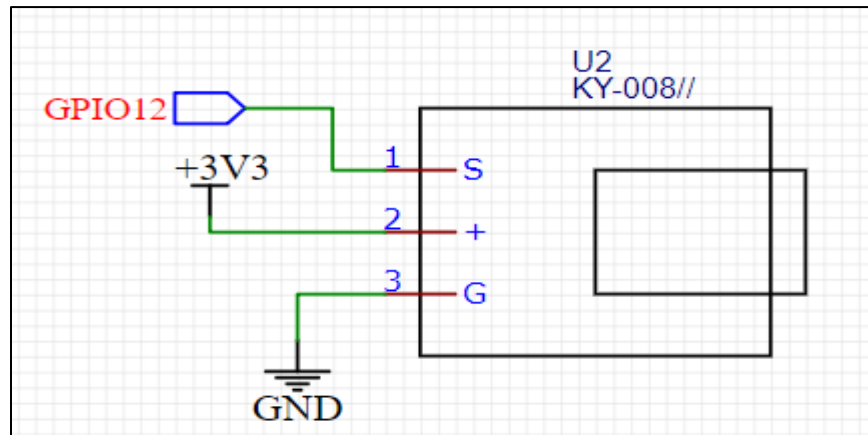
- Điện trở hay Resistor là một linh kiện điện tử thụ động gồm 2 tiếp điểm kết nối, thường được dùng để hạn chế cường độ dòng điện chảy trong mạch, điều chỉnh mức độ tín hiệu, dùng để chia điện áp, kích hoạt các linh kiện điện tử chủ động như transistor, tiếp điểm cuối trong đường truyền điện và có trong rất nhiều ứng dụng khác.

- Ký hiệu điện trở:



Hình 11: Sơ đồ khối điện trở

4.2.3.4. Module Laser (KY-008) [6]

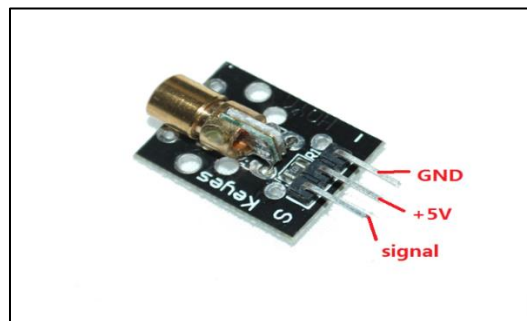


Hình 12: Sơ đồ khối Module Laser (KY-008)

- Module phát Laser KY-008 cho Arduino, phát ra một chùm tia laser màu đỏ. Bộ module phát laser KY-008 bao gồm một đầu diode laser màu đỏ và một điện trở. Cần sử dụng cẩn thận, không nhìn trực tiếp vào đầu laze.

- Kết nối chân:

- Chân S được nối với GPIO12
- Chân + được nối với nguồn 3V3
- Chân G nối với chân GND

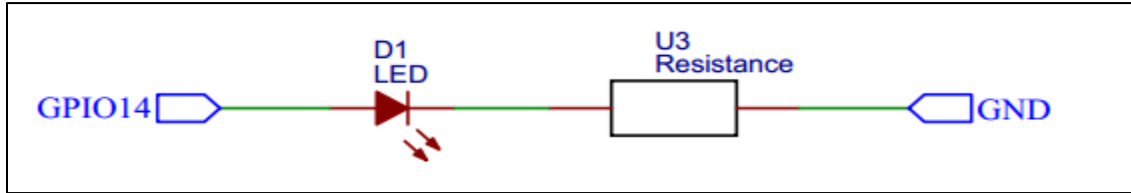


Hình 13: Sơ đồ chân Module Laser

- Thông số kỹ thuật:

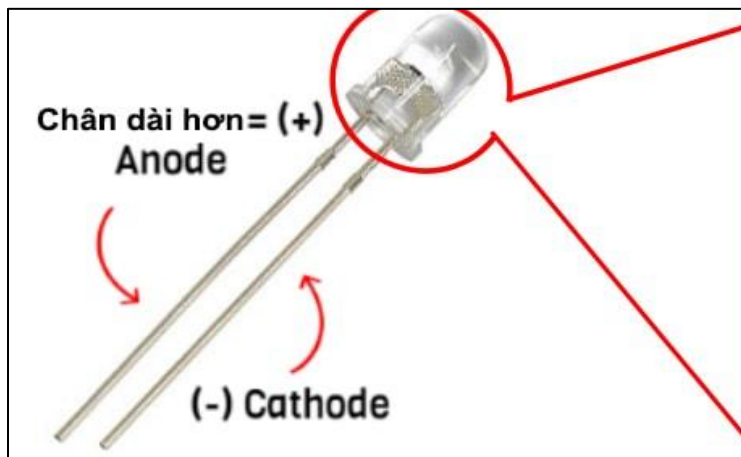
- Điện áp hoạt động 5V
- Công suất ra 5mW
- Bước sóng 650nm
- Dòng hoạt động dưới 40mA
- Nhiệt độ hoạt động: -10 ~ 40 độ C
- Kích thước: 18.5mm x 15mm

4.2.3.5. Led 5mm



Hình 14: Sơ đồ khối khối điều khiển LED

- Một chân của LED được nối với GPIO14, chân còn lại được nối với trở sau đó trở được nối với GND.
- LED là một biến thể trên diode (điốt) cơ bản. Diode là một thành phần điện tử chỉ dẫn điện theo một hướng. Nó xác định độ chênh điện áp nhỏ nhất giữa Anode (+) và Cathode (-). LED là cơ bản giống như một Diode, sự khác biệt ở đây là nó tạo ra ánh sáng khi dòng điện đi qua.
- Mỗi đèn LED có một mức điện áp đầu vào nhất định để tạo ra ánh sáng. Nếu bạn cung cấp ít hơn mức này, nó sẽ không phát sáng. Nếu bạn có nhiều đèn được nối tiếp trong một mạch, bạn phải tính mức điện áp cần cấp cho loạt này.
- Dòng điện cũng là trị số quan trọng hàng đầu cần quan tâm. Nhiệt độ là kẻ thù của công nghệ LED, nếu bạn cung cấp một dòng điện vượt qua ngưỡng cho phép, tương đương với việc làm gia tăng nhiệt độ và làm đèn nhanh chóng bị hỏng.

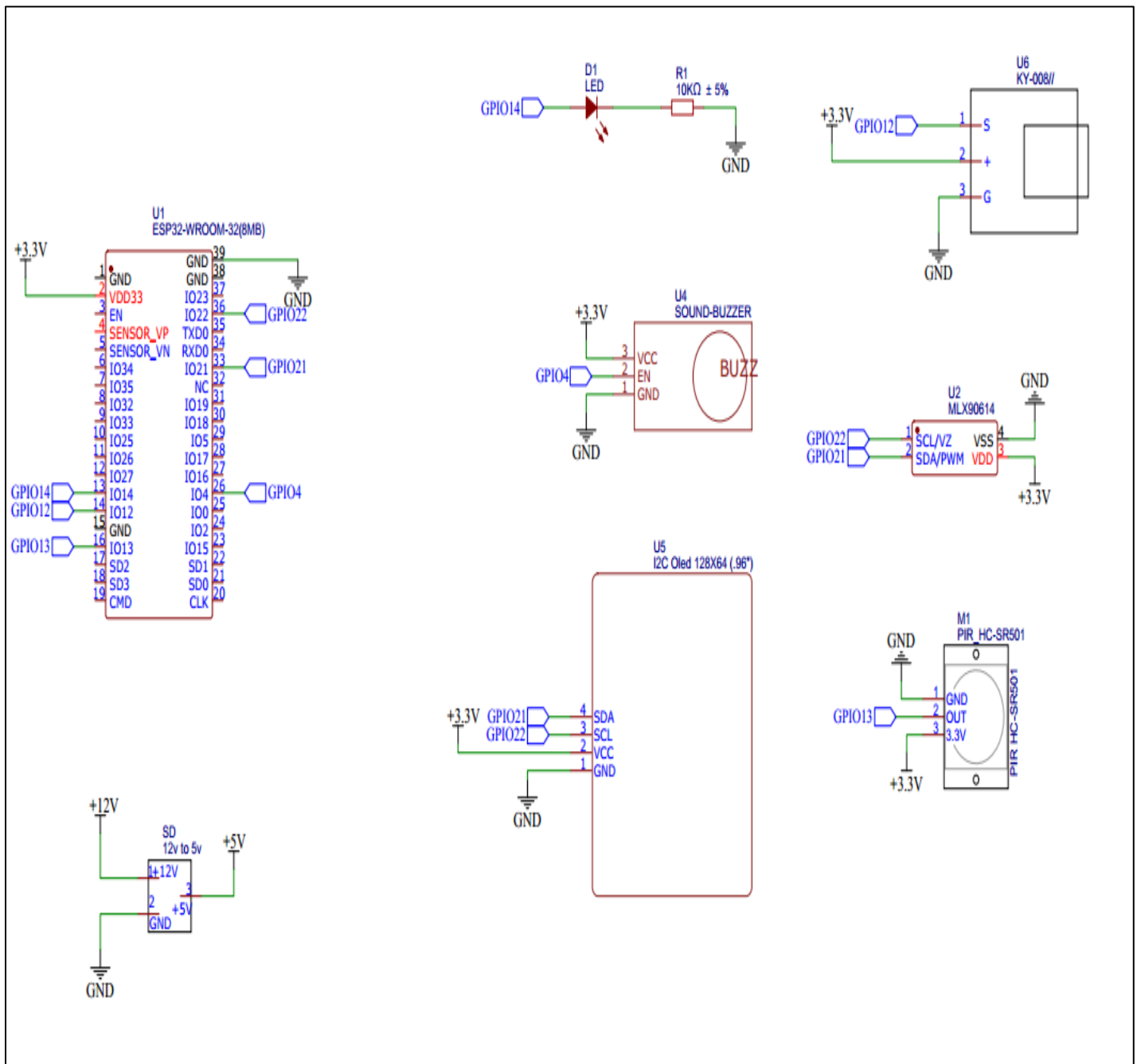


Hình 15: Sơ đồ chân LED

4.2.4. Khối điều khiển

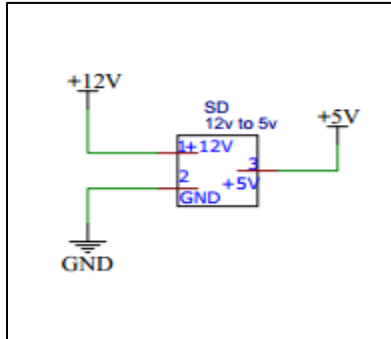
- Dùng để điều khiển hoạt động bật tắt của hệ thống.

4.3. Phân tích các khối chức năng



Hình 16: Sơ đồ kết nối các khối

4.3.1. Khối điều khiển (Khối cấp nguồn)



Hình 17: Khối cấp nguồn

- ESP sử dụng nguồn điện áp 5V, sau khi nạp code từ máy tính để cấp nguồn hoạt động tự do không chịu hạn chế của máy tính, ta sử dụng một khối cấp nguồn biến đổi điện áp bình thường (220V) thành áp 12V thông qua Adapter sau đó qua một bộ chuyển đổi điện áp 12V thành 5V, 3V3 và Vin theo yêu cầu. Sự kết hợp của hai thiết bị trên tạo thành khối cấp nguồn.

- Adapter (được cắm với nguồn 12V) tạo ngõ ra 12V:



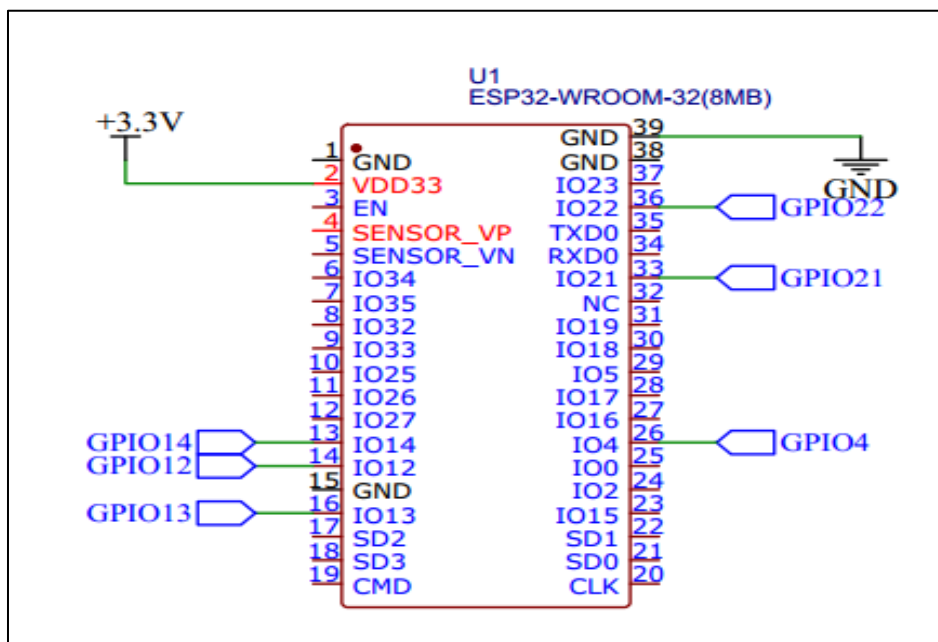
Hình 18: Adapter 12V

- Bộ chuyển đổi điện áp 12V thành điện áp theo yêu cầu:



Hình 19: Mạch chuyển đổi điện áp 12V

4.3.2. Khối xử lý trung tâm



Hình 20: Khối xử lý trung tâm

- ESP32, một vi điều khiển lõi kép của Espressif Systems với Wi-Fi và Bluetooth tích hợp. Là một thiết bị tuyệt vời dành cho người mới bắt đầu, chưa có hoặc đã có kinh nghiệm về IOT hoặc hệ thống nhúng. Dự án của nhóm được xây dựng và lập trình bởi Arduino IDE, một nền tảng cơ bản, dễ sử dụng cho người mới bắt đầu có thể thực hiện dự án theo hướng thương mại bằng Arduino IDE.

- Sử dụng các chân GPIO để nhận dữ liệu cảm biến, gửi dữ liệu điều khiển đã qua xử lý đến các khối chức năng khác nhau tùy thuộc vào nhu cầu.

- Các tín hiệu cảm biến được đưa vào ESP32 đều sẽ được xử lý thành các tín hiệu điều khiển hoạt động xử lý dữ liệu, kết nối WiFi, gửi dữ liệu lên Firebase, Google Sheet và các tín hiệu điều khiển hoạt động của các khối chức năng khác (khởi chấp hành).

- GPIO21 và GPIO22 là 2 chân giao tiếp I2C được sử dụng để nhận dữ liệu cảm biến của MLX90614 (Thân nhiệt hồng ngoại) và cũng dùng để gửi dữ liệu ra màn hình OLED.

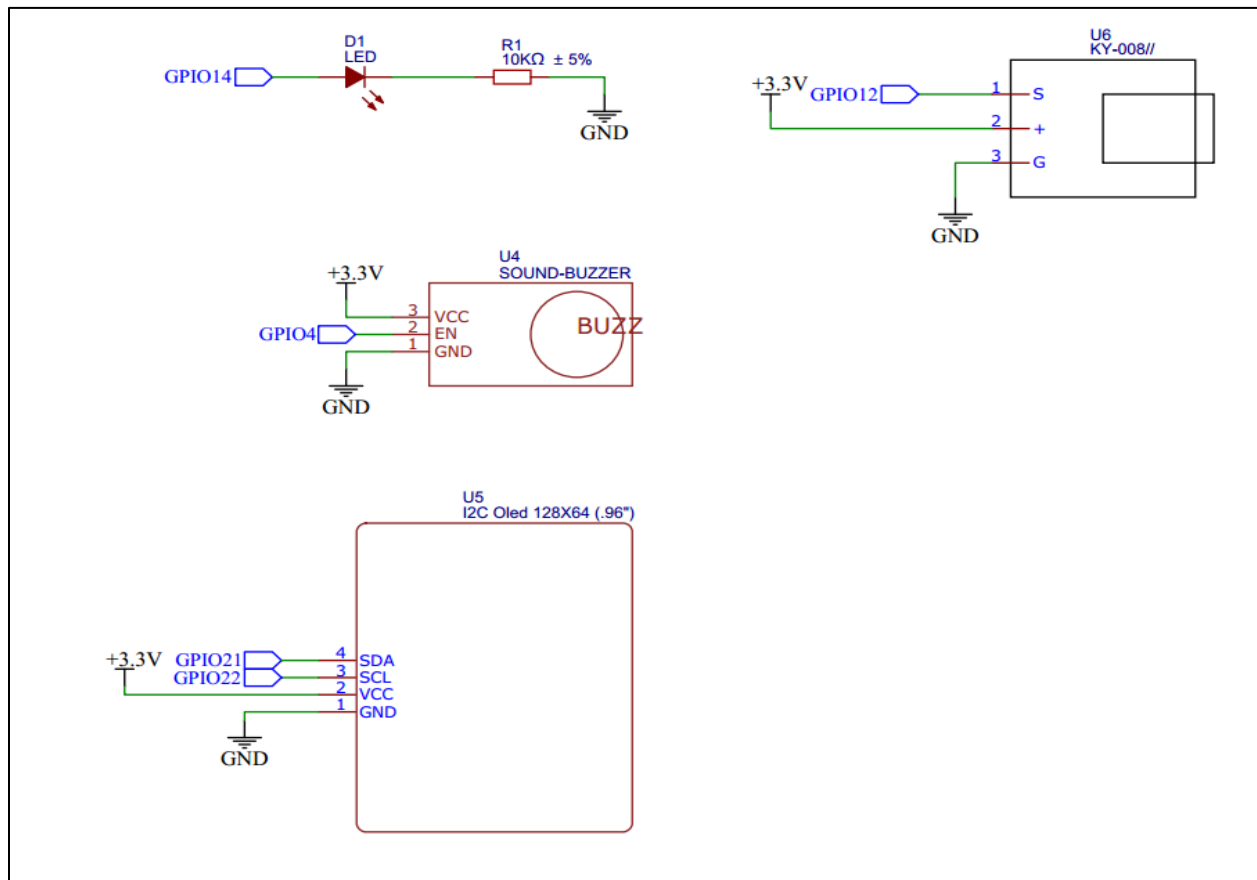
- GPIO13 là chân nhận dữ liệu cảm biến của HC-SR501 (Cảm biến chuyển động hồng ngoại) dùng làm ngắt để bật tắt hoạt động của dự án.

- GPIO14 là chân đưa tín hiệu điều khiển LED.

- GPIO4 là chân đưa tín hiệu điều khiển Buzzer.

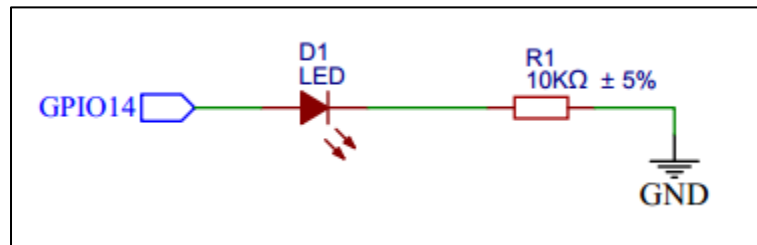
- GPIO12 là chân đưa tín hiệu điều khiển Laser module.

4.3.3. Khởi chấp hành



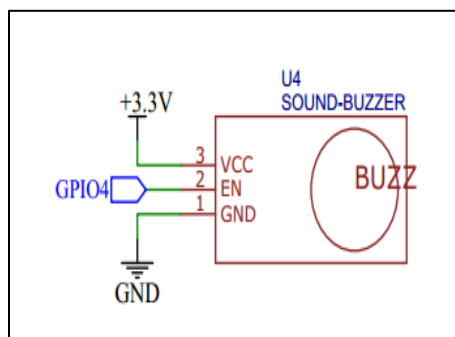
Hình 21: Sơ đồ kết nối khối chấp hành

- LED: Được điều khiển bằng tín hiệu hoạt động ngắt thông qua HC-SR501, để thể hiện thông tin có người và mạch đang hoạt động.



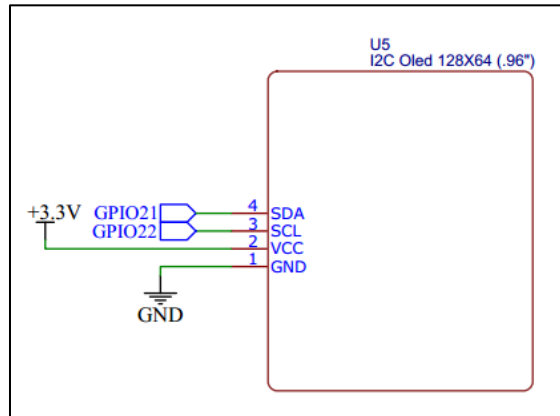
Hình 22: Điều khiển LED

- Buzzer: Được điều khiển bằng tín hiệu của ESP32 nếu nhiệt độ của người được đo cao hơn mức được cho phép Buzzer sẽ phát ra âm thanh để báo hiệu cho người giám sát.



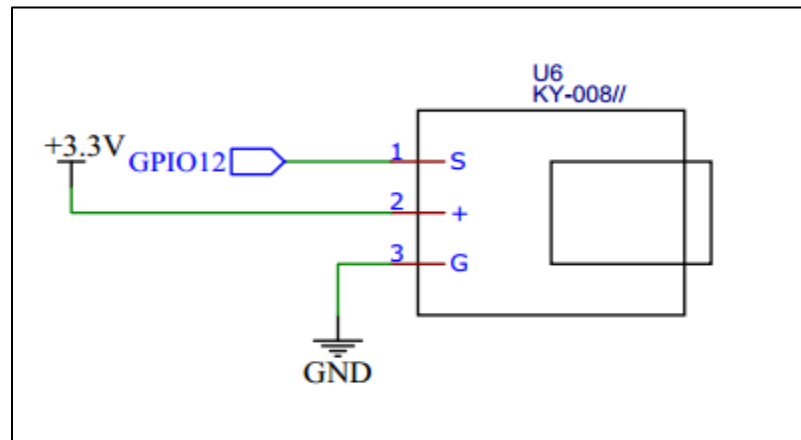
Hình 23: Buzzer

- Màn hình OLED: Được cấp nguồn hoạt động 3V3, dữ liệu nhận vào bằng 2 chân GPIO21 và GPIO22, ghi dữ liệu nhiệt độ của người được đo hoặc vật thể được đo ra ngoài màn hình. Điều khiển hoạt động ghi thông qua HC-SR501, có chuyển động mới ghi dữ liệu mới ra màn hình nếu không sẽ vẫn là dữ liệu cũ.



Hình 24: Màn hình OLED

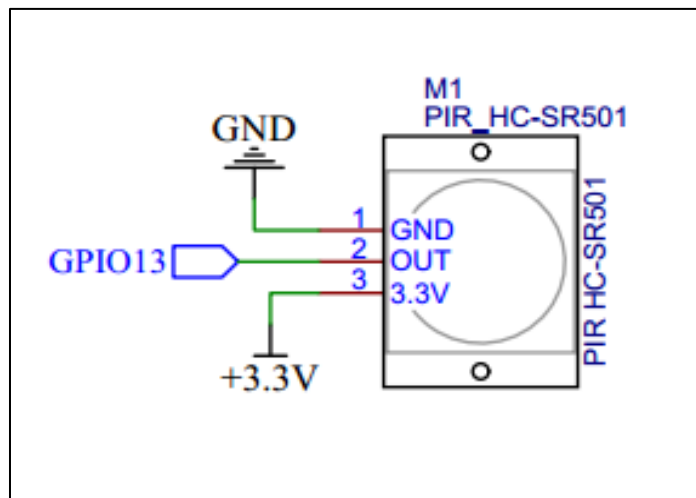
- Laser Module: Được cấp nguồn 3V3, tín hiệu điều khiển thông qua chân GPIO12 và được bật tắt bằng ngắt (HC-SR501), thể hiện hướng của máy đo đang hướng tới.



Hình 25: Module Laser

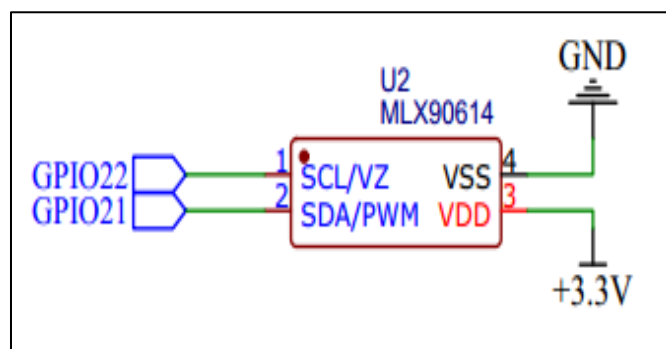
4.3.4. Khối cảm biến

- Cảm biến chuyển động hồng ngoại (HC-SR501): Nhận biết chuyển động của người thông qua hồng ngoại, trong dự án của nhóm HC-SR501 đóng vai trò là một ngắt, khi có tín hiệu chuyển động (tín hiệu ngắt) thì mạch mới bắt đầu hoạt động – các cảm biến mới bắt đầu truyền dữ liệu về ESP32 để xử lý, ESP sau khi xử lý dữ liệu mới gửi tín hiệu đến các khối chấp hành, có thể nói HC-SR501 đóng vai trò như một công tắc bật tắt tự động.
- Được cấp nguồn 3V3, gửi dữ liệu thông qua chân GPIO13.



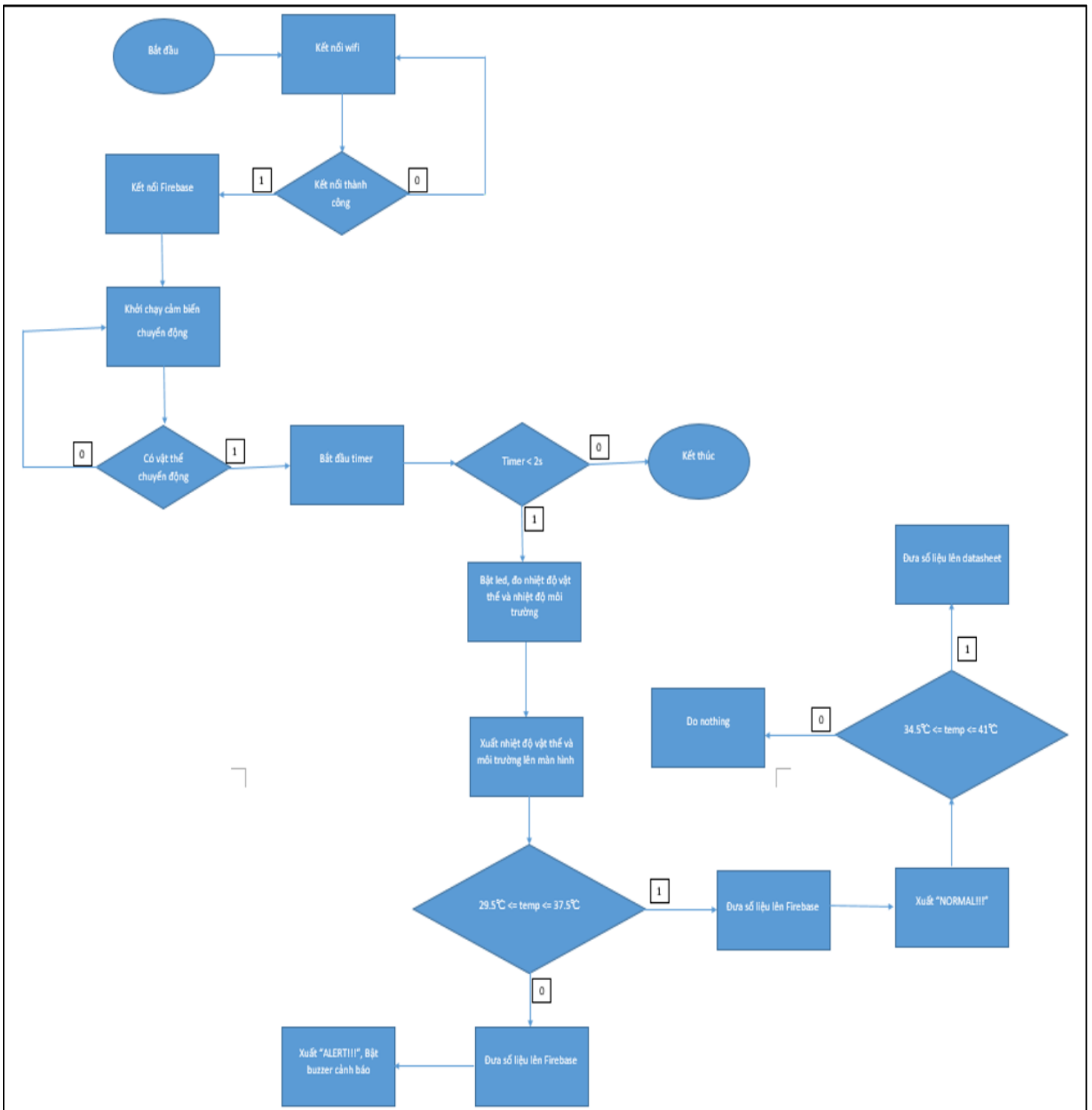
Hình 26: PIR_HC-SR501

- Cảm biến nhiệt độ hồng ngoại MLX90614: Đo nhiệt độ vật thể và nhiệt độ xung quanh bằng hồng ngoại và truyền về ESP bằng hai chân GPIO22 và GPIO21 thông qua giao tiếp I2C. Cũng như các khối chức năng khác, MLX90614 cũng được điều khiển hoạt động bật tắt thông qua HC-SR501.



Hình 27: MLX90614

4.4. Lưu đồ thuật toán:



Hình 28: Lưu đồ thuật toán

4.5. Chương trình nạp vào ESP32

```
//Khai báo thư viện cần thiết cho ESP32
#include <Wire.h>
#include <Adafruit_MLX90614.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <WiFi.h>
#include <FirebaseESP32.h>
#include <WiFiClient.h>
#include <HTTPClient.h>

//Information GG Firebase
#define FIREBASE_HOST "https://webofhuy-default-rtdb.firebaseio.com/"
#define FIREBASE_AUTH "kEw0jTfd14NXXctHZetZ4nYmGmTt8oXd9fsP0T0p"
char ssid[] = "BAD_Group";//Tên đăng nhập WiFi
char pass[] = "deocodau";//Mật khẩu WiFi

WiFiClientSecure client;//Biến Client của WiFiClientSecure để gửi
thông tin lên Google Sheet

/*****Google Sheets Definations*****/
String GOOGLE_SCRIPT_ID =
"AkfycbzQzg9t3yVNDHahm4D1K9wcKhtLdEyNDbFlHICnXO4eRUGlT4OhauePn50KnCkOB
aNXgg";

//Firebase Data
FirebaseData fbdo1;
FirebaseData fbdo2;
FirebaseData fbdo3;

//LCD OLED Parameter
#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 64 // OLED display height, in pixels
#define OLED_RESET -1 // Reset pin # (or -1 if sharing Arduino
reset pin)
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire,
OLED_RESET);//Khai báo cho màn hình SSD1306 được kết nối với I2C (chân
SDA, SCL)

//GPIO laser
#define laser 12
```

```
//Khai báo biến mlx là MLX90614 của Adafruit
Adafruit_MLX90614 mlx = Adafruit_MLX90614();

//Biến Temperature
double temp_amb;
double temp_obj;

//LED, cảm biến HC-SR501 và buzzer GPIO
const int led_pin = 14;
const int sensor_pin = 13;
const int buzzer_pin = 4;

//Khoảng thời gian của ngắt
const long interval = 2000; //Khoảng thời gian là 2000ms
unsigned long current_time = millis(); //Thời gian tính bằng ms
unsigned long last_trigger = 0;
boolean timer_on = false; //Tắt timer

//Interrupt function
void IRAM_ATTR movement_detection() {
    Serial.println("Motion was detected"); //In dòng: "Motion was
detected"
    digitalWrite(led_pin, HIGH); //Bật led thể hiện phát hiện chuyển động
    timer_on = true; //Bật timer (Bắt đầu đếm ngược)
    last_trigger = millis();
}

void setup() {
    Serial.begin(115200); //Tốc độ baud của Serial Monitor là 115200

    //MLX90614
    mlx.begin(); //Khởi tạo MLX90614 bắt
đầu hoạt động
    display.begin(SSD1306_SWITCHCAPVCC, 0x3C); //khởi tạo với I2C addr
0x3C (128x64)

    //Kết nối WiFi và Firebase
    WiFi.begin(ssid, pass); //Bắt đầu kết nối với WiFi với thông tin về
tên đăng nhập và mật khẩu
    Serial.print("Connect Wi-Fi"); //In ra dòng: "Connect Wi-Fi"
    while (WiFi.status() != WL_CONNECTED) //Nếu không kết nối được WiFi
    {
```

```

    Serial.print("."); //In ra dấu "."
    delay(300); //Độ trễ 0.3s
}
Serial.println();
Serial.print("WiFi IP: ");
Serial.println(WiFi.localIP()); //Nếu kết nối thành công in ra địa
chỉ IP WiFi đang kết nối
Serial.println();
Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH); //Kết nối tới Firebase
Firebase.reconnectWiFi(true); //Nếu mất kết nối tự động kết nối lại

//Interrupt
pinMode(sensor_pin, INPUT_PULLUP); //Cài đặt chân đầu vào và pull_up
cho HC-SR501
attachInterrupt(digitalPinToInterrupt(sensor_pin),
movement_detection, RISING); //Tạo kết nối với hàm ngắt, chân nhận tín
hiệu ngắt và xung cạnh lên
pinMode(led_pin, OUTPUT); //Gán chân Led là OUTPUT
digitalWrite(led_pin, LOW); //Chân led khởi đầu mức thấp
pinMode(buzzer_pin, OUTPUT);
digitalWrite(buzzer_pin, LOW);

Serial.println("Temperature Sensor MLX90614"); //In ra dòng:
"Temperature Sensor MLX90614"

pinMode(laser, OUTPUT); // Connect LASER

//Cài đặt thông số Laser module
pinMode(laser, OUTPUT);
digitalWrite(laser, LOW);

//Cài đặt thông số bắt đầu cho OLED
display.clearDisplay();
display.setCursor(25, 15);
display.setTextSize(1);
display.setTextColor(WHITE);
display.println(" Thermometer");
display.setCursor(25, 35);
display.setTextSize(1);
display.print("Initializing");
display.display();
delay(5000);
}

```



```

void loop() {
  //Cài đặt đơn vị thời gian là ms
  current_time = millis();

  //Interrupt ON và Timer sẽ đếm xuống
  if (timer_on && (current_time - last_trigger <= interval)) {
    //Đọc nhiệt độ môi trường và vật thể
    //nếu muốn đọc nhiệt độ dưới dạng độ F thì dùng
    //mlx.readAmbientTempF() , mlx.readObjectTempF() )
    temp_amb = mlx.readAmbientTempC();
    temp_obj = mlx.readObjectTempC()+2.65;

    //Bật laser module
    digitalWrite(laser, HIGH);

    //Gửi dữ liệu ra Serial Monitor (Arduino IDE)
    Serial.print("Room Temp = "); //In ra dòng: "Room Temp = "
    Serial.println(temp_amb); //Nhiệt độ phòng
    Serial.print("Object temp = "); //In ra dòng: "Object temp = "
    Serial.println(temp_obj); //Nhiệt độ vật thể

    //Hiển thị nhiệt độ vật thể được đo ra màn hình
    display.clearDisplay(); //xóa toàn bộ chữ có trên màn hình
    display.setCursor(25, 10); //Vị trí con trỏ
    display.setTextSize(1); //Độ lớn ký tự
    display.setTextColor(WHITE); //Màu ký tự
    display.println(" Temperature"); //Hiện ra dòng: " Temperature"
    display.setCursor(25, 30); //Vị trí con trỏ
    display.setTextSize(2); //Độ lớn ký tự
    display.print(temp_obj); //Độ lớn nhiệt độ vật thể
    display.print((char)247); //Ký tự độ
    display.print("C"); //Ký tự C
    display.display();

    //Gửi dữ liệu về nhiệt độ Object và Ambient lên Firebase
    Firebase.setInt(fbdo1, "/HTN/Object", temp_obj);
    Firebase.setInt(fbdo2, "/HTN/Ambient", temp_amb);

    //Nếu nhiệt độ người được đo cập nhật trạng thái bình thường ngược
    lại là Alert
    if(temp_obj <= 37.5 || temp_obj >= 34.5){
      Firebase.setString(fbdo3, "/HTN/Status", "NORMAL!!");
    } else {
      Firebase.setString(fbdo3, "/HTN/Status", "ALERT!!!");
    }
  }
}

```

//Điều kiện về nhiệt độ của con người để Buzzer báo động (Khởi chấp hành)

```
if (temp_obj >= 37.5) {
    digitalWrite(buzzer_pin, HIGH);
    Serial.print("ALERT!!!!\n ");
}
```

//Điều kiện phân biệt về nhiệt độ của con người gửi nhiệt độ lên GG Sheet

```
if (temp_obj >= 35.75 && temp_obj <= 41) {
    sendData(temp_obj,temp_amb);
}
delay(2000);
}
```

//Interrupt OFF khi Timer lớn hơn interval

//Tắt hết các hệ thống khi Interrupt ngắt

```
if (timer_on && (current_time - last_trigger > interval)) {
    Serial.println("Motion has stopped");
    digitalWrite(led_pin, LOW); //LED tắt
    digitalWrite(buzzer_pin, LOW); //Buzzer tắt
    digitalWrite(laser, LOW); //laser tắt
    timer_on = false; //Tắt timer
    display.clearDisplay(); //xóa màn hình oled
}
```

```
}
```

//Tạo hàm sendData để gửi dữ liệu lên Google Sheet

```
void sendData(float temp_obj, float temp_amb) {
    HTTPClient http;
    String
url="https://script.google.com/macros/s/"+GOOGLE_SCRIPT_ID+"/exec?value1="+temp_obj+"&value2="+temp_amb; //Đường dẫn tự động gửi dữ liệu lên Sheet
    Serial.print(url); //In ra đường dẫn
    Serial.print("Making a request"); //In ra dòng "Making a request"
    http.begin(url.c_str()); //Chỉ định URL và certificate
    int httpCode = http.GET(); //Lấy httpCode
    http.end(); //Kết thúc kết nối http
    Serial.println(": done\n "+httpCode); //In ra dòng: "done" + httpCode
}
```

5. KẾT LUẬN

5.1. Kết quả và đánh giá

Sau khi nạp chương trình vào Board ESP32, thiết bị đã sẵn sàng để sử dụng. Bây giờ ta bật nút nguồn để mở hoạt động mạch chuyển động điện áp để bắt đầu cấp nguồn hoạt động cho hệ thống, hệ thống sẽ mất một khoảng thời gian, ít nhất là 5 giây để bắt đầu hoạt động. Khi mạch hoạt động chúng ta có thể bắt đầu đo.

Kết quả và đánh giá: Khi phát hiện chuyển động của người được đo, mạch nhận biết chuyển động gửi tín hiệu điều khiển bắt đầu hoạt động đo của hệ thống, laser và led được bật lên, led thể hiện hệ thống đang hoạt động, laser chiếu hướng hệ thống đang đo, sau khi đo nhiệt độ thành công hiển thị nhiệt độ được lên màn hình OLED. Kiểm thử bằng nhiệt độ cao hơn nhiệt độ bình thường thì Buzzer phát ra âm thanh cảnh báo. Dữ liệu về nhiệt độ của MLX90614 được gửi thành công lên Google Firebase và Google Sheet. Mạch hoạt động ổn định, hoạt động đo chính xác và độ trễ thấp. Mạch đi dây khá rườm rà, hộp đựng bảo vệ các linh kiện chưa phù hợp nên dễ ảnh hưởng đến hoạt động của hệ thống.

5.2. Hướng phát triển

Hướng phát triển cho sản phẩm là thiết kế một mạch hoàn chỉnh tích hợp toàn bộ ESP, các cảm biến và các mạch chức năng vào cùng nhau không cần sử dụng đến dây dẫn tránh rắc rối. Đóng gói sản phẩm chắc chắn để bảo vệ hệ thống mạch trước các ảnh hưởng của môi trường (nắng, mưa, ...).

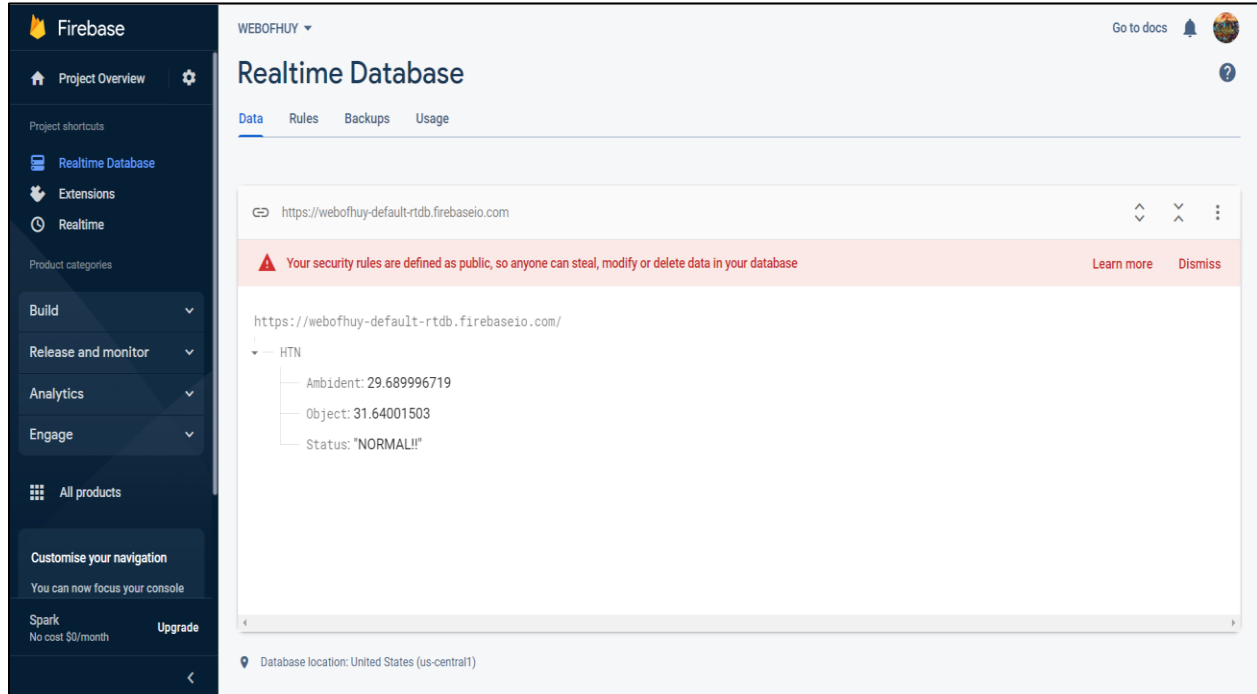
Kết luận: Sau khi trải qua quá trình nghiên cứu, thiết kế và thực hiện mạch dưới sự hướng dẫn, góp ý từ thầy Phan Văn Ca nhóm chúng em đã hoàn thành đề tài "Hệ thống giám sát thân nhiệt và cảnh báo tại nơi đông người qua lại".

Chúng em đã gặp nhiều khó khăn trong quá trình này từ việc nghiên cứu đến thực hiện thi công mạch thực tế còn nhiều sai sót nhưng cũng vì thế mà nhóm em đã rút ra được kinh nghiệm kịp thời và cho ra được sản phẩm đúng với mong muốn.

Sản phẩm thực hiện tốt chức năng đã đề ra tuy nhiên vẫn còn những điểm cần cải thiện. Việc kết hợp với một khối điều khiển không dây để người dùng có thể điều khiển từ xa, tạo thành một mạch hoàn chỉnh không cần đến dây dẫn rườm rà và thiết kế phần vỏ bảo vệ sẽ là hướng phát triển để hoàn thiện sản phẩm.

Nhóm chúng em xin chân thành cảm ơn sự giúp đỡ, các ý kiến góp ý từ thầy Phan Văn Ca đã giúp nhóm hoàn thành đề tài này.

6. SẢN PHẨM CUỐI CÙNG

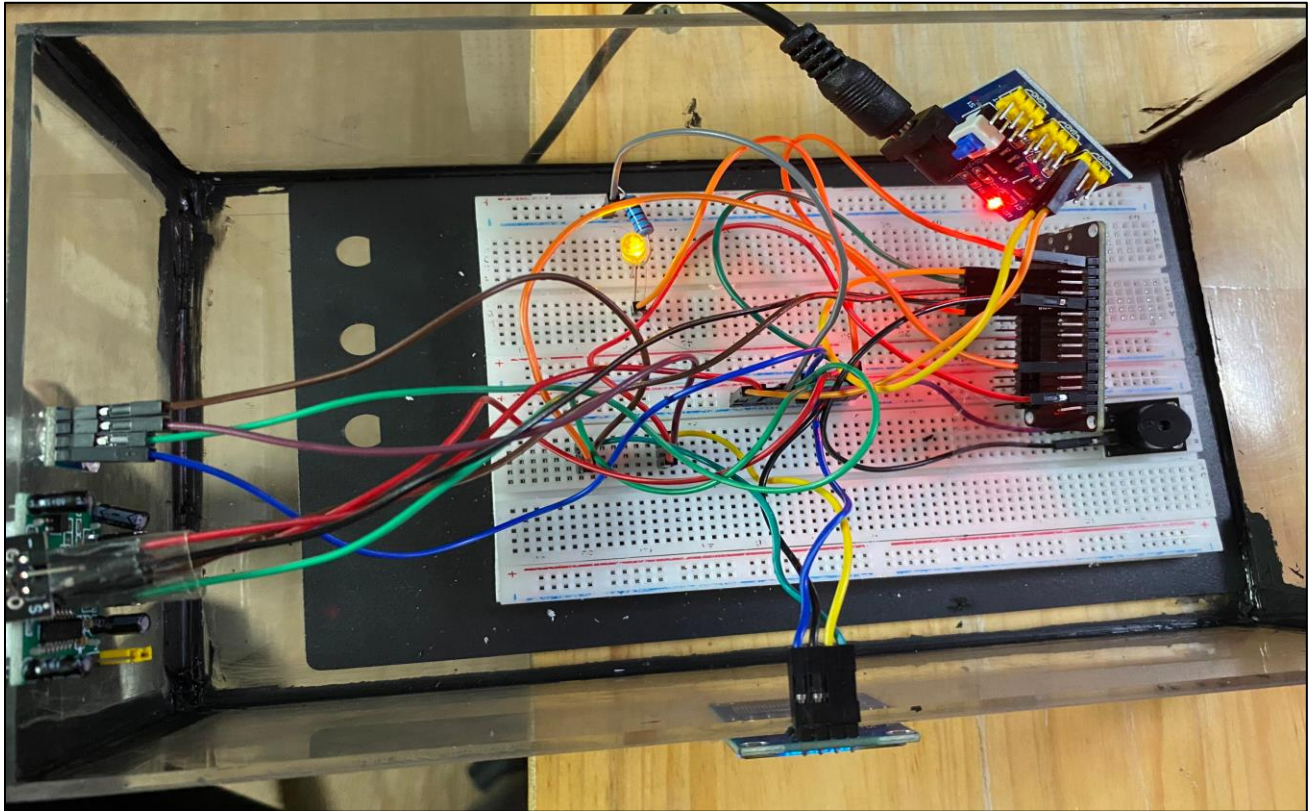


Hình 29: Dữ liệu lên Firebase

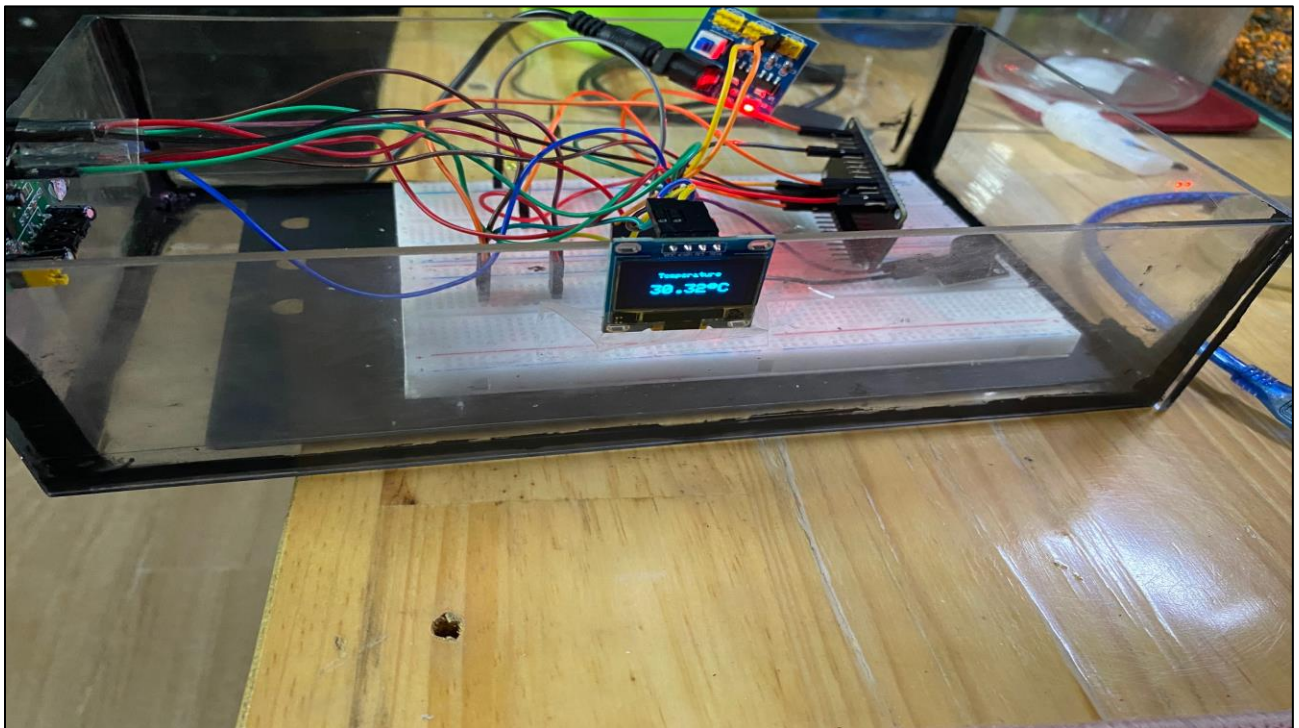
The screenshot shows a Google Sheet titled 'HTN'. The sheet contains a table with columns A through N. The data is organized into rows, with the first column (A) containing dates (15/12/2022) and the subsequent columns (B through N) containing numerical values. The data is as follows:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
124	15/12/2022	21:18:03	35.17	30.73										
125	15/12/2022	21:18:17	35.39	30.73										
126	15/12/2022	21:18:29	36.23	30.67										
127	15/12/2022	21:18:54	35.33	30.69										
128	15/12/2022	21:19:05	35.61	30.67										
129	15/12/2022	21:19:17	35.93	30.67										
130	15/12/2022	21:52:01	37.43	30.03										
131	15/12/2022	21:53:00	37.09	30.17										
132	15/12/2022	21:53:28	37.25	30.43										
133	15/12/2022	22:09:12	35.89	30.09										
134	15/12/2022	22:09:20	36.33	30.17										
135	15/12/2022	22:09:43	35.97	30.09										
136	15/12/2022	22:10:05	36.23	30.07										
137	15/12/2022	22:10:52	35.49	30.07										
138	15/12/2022	22:11:06	35.27	30.07										
139														
140														
141														
142														
143														
144														
145														
146														
147														
148														

Hình 30: Dữ liệu lên Google Sheet



Hình 31: Mô hình sản phẩm



Hình 32: Mô hình sản phẩm

7. PHỤ LỤC

7.1. Bảng giá

ID	Parts/Components	Amount	Price per Unit	Total
1	ESP32-WROOM 32	1	170.000	170.000
2	HC-SR501	1	20.000	20.000
3	OLED 0.96 inch	1	76.000	76.000
4	KY-008 (laser module)	1	6.000	6.000
5	Cảm biến nhiệt MLX-90614	1	235.000	235.000
6	Adapter 12V	1	12.000	12.000
7	Mạch chuyển đổi điện áp	1	27.000	27.000
8	Buzzer	1	12.000	12.000
9	Dây bus	50	15.000	15.000
10	Hàng rào 1x20	20	6.000	6.000
11	Tấm mica	1	100.000	100.000

Bảng 2: Bảng giá tiền sản phẩm

7.2. Biểu đồ làm việc

Công việc	ngày bắt đầu	ngày kết thúc	số ngày	Tuần 16							Tuần 17						
				2	3	4	5	6	7	CN	2	3	4	5	6	7	CN
Lên ý tưởng thực hiện đề tài, xây dựng mô hình lý thuyết	05/12/2022	06/12/2022	2														
Đặc tả thiết kế, vẽ sơ đồ khối, tìm hiểu yêu cầu, tìm linh kiện	07/12/2022	09/12/2022	3														
Thiết kế kiến trúc hệ thống	09/12/2022	11/12/2022	3														
Thiết kế phần mềm hệ thống	12/12/2022	14/12/2022	3														
Thi công sản phẩm	14/12/2022	15/12/2022	2														
Viết báo cáo	15/12/2022	17/12/2022	2														

Bảng 3: Biểu đồ làm việc

7.3. Bảng phân công

STT	Công việc	Phân công	Mô tả cụ thể	Nguồn	Điều kiện tiên quyết
1	Lên ý tưởng thực hiện đề tài. Xây dựng mô hình trên lý thuyết. Đặc tả yêu cầu.	Cả nhóm	- Xây dựng các chức năng của hệ thống dựa trên những yêu cầu của người dùng.	Tham khảo các nguồn trên Internet	Không có.
2	Đặc tả về thiết kế, vẽ sơ đồ khối Thông số các linh kiện	Cả nhóm	Nêu được mục tiêu của thiết kế Những yêu cầu và thông số linh kiện	Tham khảo nguồn internet và datasheet	Tiến trình 1.
3	Thiết kế kiến trúc hệ thống	Cả nhóm	Vẽ ra được khái quát sơ đồ khối hệ thống	Tham khảo các mô hình minh họa có sẵn	Tiến trình 1, 2
4	Xây dựng hệ thống cảm biến chuyển động và đo nhiệt độ, cảnh báo qua buzzer.	Lê Công Bắc	- Ghi dữ liệu nhiệt độ khi có chuyển động, quá ngưỡng còi sẽ kêu	ESP32 Buzzer KY-012 MLX-90614 (cảm biến nhiệt độ) HCSR-501	Tiến trình 3.
5	Xây dựng hệ thống cảm biến nhiệt độ hiển thị trên màn OLED và chỉ hướng đo bằng laser	Đào Ngọc Minh Huy	- Ghi dữ liệu nhiệt độ từ môi trường và một đối tượng hiển thị được dữ liệu trên OLED	OLED 0.96 inch Module Laser KY-008	Tiến trình 3.
6	Liên kết với ứng dụng Firebase để có thể gửi dữ liệu qua wifi	Lê Công Bắc	- Tạo một hệ thống hoàn chỉnh. Giao tiếp thông qua VDK để gửi cảnh báo qua internet.	Firebase	Tiến trình 3.
7	Liên kết firebase qua google sheet	Đào Ngọc Minh Huy	Gửi các dữ liệu nhiệt độ nằm trong khoảng nhiệt độ con người lên google sheet	Firebase Google sheet	Tiến trình 3
8	Thử nghiệm hệ thống	Cả nhóm	Kiểm tra các trường hợp có thể xảy ra. Kiểm tra lỗi và khắc phục.	- Các hệ thống nhỏ đã xây dựng trước đó.	Tiến trình 4.
9	Xây dựng mô hình	Cả nhóm	- Đóng gói mô hình hoàn chỉnh để có tạo ra 1 khối đạt được những yêu cầu đưa ra	Các hệ thống hoàn chỉnh đã xây dựng xong	Tiến trình 5
10	Viết báo cáo	Cả nhóm	- Tổng hợp và viết báo cáo	Mô hình hệ thống hoàn chỉnh	Tiến trình 6.

Bảng 4: Bảng phân công