

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM
KHOA ĐIỆN – ĐIỆN TỬ
NGÀNH CÔNG NGHỆ KỸ THUẬT MÁY TÍNH



BÁO CÁO CUỐI KỲ MÔN HỌC:
HỆ THỐNG NHÚNG (CNKT-MÁY TÍNH)

ĐỀ TÀI:
THIẾT KẾ MÁY PHUN SƯƠNG
THÔNG MINH

Mã môn học: EMSY435664
Giảng viên hướng dẫn: TS. Phan Văn Ca
Lớp: 08CLC, thứ 7, tiết 3-5

Danh sách thành viên nhóm thực hiện:

Họ và tên	Mã số sinh viên
Nguyễn Khắc Luân	22119099
Lê Hoàng Gia Khang	22119087
Trần Hoàng Hải	22119066
Lê Khải Hưng	22119086

TP. Hồ Chí Minh, tháng 12 năm 2024

MỤC LỤC

PHẦN A: XÁC ĐỊNH VĂN ĐỀ	5
I. Lý do chọn đề tài.....	5
II. Mục tiêu	5
PHẦN B: TIẾP CẬN VĂN ĐỀ.....	6
I. Phương pháp giải quyết	6
II. Cơ sở lý thuyết.....	6
PHẦN C: ĐẶC TẢ KỸ THUẬT	8
I. Chức năng hệ thống.....	8
II. Thông số kỹ thuật của hệ thống.....	8
PHẦN D: KIẾN TRÚC HỆ THỐNG	9
I. Sơ đồ tổng quát	9
II. Sơ đồ kết nối	9
PHẦN E: THIẾT KẾ CHI TIẾT	11
I. PHẦN CÚNG	11
1. Lựa chọn linh kiện và thông số của linh kiện.....	11
2. Sơ đồ khối tổng quan	20
II. PHẦN MỀM	26
1. Lưu đồ giải thuật.....	26
2. Code chương trình.....	27
PHẦN F: SẢN PHẨM CUỐI CÙNG.....	40
I. Hình ảnh sản phẩm thực tế.....	40
II. Đánh giá sản phẩm.....	42
KẾT LUẬN	43

TÀI LIỆU THAM KHẢO.....44

PHÂN CÔNG NHIỆM VỤ LÀM BÁO CÁO ĐỀ TÀI

- Tên đề tài: Thiết kế máy phun sương thông minh
- Bảng phân công nhiệm vụ

STT	Nhiệm vụ	Mô tả chi tiết	Ngày bắt đầu	Thời gian làm (ngày)	Thực hiện
1	Thiết kế hệ thống	Lên ý tưởng giải quyết vấn đề và thiết kế kiến trúc tổng quan của hệ thống	14/10/2024	7	Trần Hoàng Hải, Lê Khải Hưng, Lê Hoàng Gia Khang, Nguyễn Khắc Luân
2	Thiết kế chi tiết	Phân tích và tìm kiếm giải pháp phần cứng, phần mềm	21/10/2024	20	Trần Hoàng Hải, Nguyễn Khắc Luân, Lê Hoàng Gia Khang
3	Lập trình và ghép mẫu thử	Nạp mã và kiểm thử sản phẩm trên test board	10/11/2024	15	Trần Hoàng Hải, Lê Khải Hưng, Lê Hoàng Gia Khang, Nguyễn Khắc Luân
4	Thi công sản phẩm	Thi công mạch và thiết kế sản phẩm thực tế hoàn chỉnh	25/11/2024	4	Lê Khải Hưng, Lê Hoàng Gia Khang

Nhận xét của giảng viên:

.....

.....

.....

.....

.....

.....

.....

.....

.....

Chữ ký của giảng viên
(ký, ghi rõ họ tên)

PHẦN A: XÁC ĐỊNH VÂN ĐỀ

I. Lý do chọn đề tài

Kiểm soát độ ẩm phù hợp giúp duy trì sức khỏe tốt hơn, đặc biệt là trong các điều kiện thời tiết khô hanh hoặc môi trường sử dụng máy lạnh. Việc kiểm soát độ ẩm, nhiệt độ hợp lý có thể nâng cao chất lượng không khí trong nhà, giảm nguy cơ mắc các bệnh liên quan đến đường hô hấp. Máy phun sương được coi là một giải pháp tương đối tốt để đáp ứng các vấn đề trên.

Đồng thời sự phát triển mạnh mẽ của công nghệ nhúng, IoT và nhà thông minh mở ra tiềm năng lớn cho các thiết bị thông minh, trong đó máy phun sương có thể là một trong những sản phẩm tiên phong trong việc kết nối và tự động hóa môi trường sống. Vì những lý do trên, nhóm em quyết định chọn “Thiết kế máy phun sương thông minh” làm đề tài đồ án cuối kỳ.

II. Yêu cầu người dùng

- Máy phun sương dùng để tạo độ ẩm; tránh cảm giác khô trong phòng (khách, ngủ, máy lạnh,...).
- Máy phun sương hoạt động ở 2 chế độ: auto và manual
 - Auto: có khả năng tự động, điều chỉnh thời gian phun để đáp ứng độ ẩm theo nhu cầu của người dùng.
 - Tự động bật, tắt tùy theo trường hợp có người trong phòng hay không.
 - Manual: người dùng bật tắt máy phun tùy ý (bật hoặc tắt). Điều chỉnh tốc độ phun.
- Tự động tắt khi thiếu nước bất kể ở chế độ nào.
- Hiển thị thông số về nhiệt độ, độ ẩm, dung tích còn lại qua điện thoại, LCD.
- Người dùng có thể điều khiển thiết bị tùy ý, trực tiếp thông qua nút bấm hoặc điều khiển thông qua điện thoại.

PHẦN B: TIẾP CẬN VÂN ĐÈ

I. Phương pháp giải quyết

Để có thể kiểm soát được độ ẩm trong phòng, ta có thể điều chỉnh tốc độ phun và thời gian phun của máy phun sương. Nhóm em quyết định chọn điều chỉnh thời gian phun để kiểm soát vì đây là một yếu tố dễ tiếp cận và thực hiện hơn. Và để phù hợp với những căn phòng có kích thước khác nhau thì thời gian sẽ được điều chỉnh đôi chút. Cơ sở lý thuyết và cách thức tính toán sẽ được trình bày bên dưới đây.

II. Cơ sở lý thuyết

- **Độ ẩm tương đối** hay còn gọi là **độ ẩm tẩm ướt** (Relative Humidity, viết tắt: RH) là tỷ lệ giữa áp suất riêng phần của hơi nước và áp suất hơi bão hòa của nước ở cùng nhiệt độ. Không khí càng ẩm, độ ẩm tương đối càng cao. Độ ẩm tương đối phụ thuộc vào nhiệt độ và áp suất của hệ thống quan tâm. Cùng một lượng hơi nước sẽ dẫn đến độ ẩm tương đối trong không khí mát cao hơn không khí ấm.
- Ở mỗi nhiệt độ có độ bão hòa hơi nước khác nhau:

Temp (°F)	Mixing ratio (grams per kilogram)	Temp (°F)	Mixing ratio (grams per kilogram)	Temp (°F)	Mixing ratio (grams per kilogram)
0	0.96	42	5.69	72	17.37
3	1.10	44	6.16	74	18.64
6	1.26	46	6.65	76	20.00
9	1.44	48	7.18	78	21.45
12	1.64	50	7.75	80	22.99
15	1.87	52	8.36	82	24.64
18	2.13	54	9.01	84	26.40
21	2.42	56	9.71	86	28.27
24	2.75	58	10.46	88	30.27
27	3.11	60	11.26	90	32.41
30	3.52	62	12.12	92	34.68
32	3.82	64	13.03	94	37.10
34	4.15	66	14.01	96	39.68
36	4.49	68	15.06	98	42.44
38	4.86	70	16.17	100	45.37
40	5.26				

Hình 1: Bảng độ ẩm bão hòa

- Thông số tính toán:

- Tốc độ tạo sương: 300ml/giờ
- Khối lượng riêng không khí khô (density of dry air): 1.225 kg/m^3
- Khối lượng riêng của nước: $997 \text{ kg/m}^3 = 997 \text{ g/l}$

- Bài toán:

Giả sử căn phòng có diện tích 25m^2 , cao 4m (*ta sẽ ước lượng căn phòng lớn nhất có thể đáp ứng*)

$$\Rightarrow \text{Thể tích} = 120\text{m}^3$$

$$\Rightarrow \text{Khối lượng không khí khô} = 100 * 1.225 = 122.5\text{kg}$$

Giả sử giá trị nhiệt độ mà cảm biến trả về = $90^\circ\text{F} = 32.22222^\circ\text{C}$

Và độ ẩm từ cảm biến = 35%

$$\Rightarrow \text{Độ ẩm cần thêm} = 45\% - 40\% = 5\%$$

$$\Rightarrow \text{Lượng nước cần thêm (g)} = 5\% * 32,41 * 122.5 = 198.5\text{g} \Rightarrow 198.5 \text{ ml}$$

$$\Rightarrow \text{Thời gian phun} = (198.5 / 300) * 60 = 40 \text{ phút}$$

=> Chia đôi thời gian cần phun (vì ta lấy tham chiếu là căn phòng lớn nhất có thể đáp ứng)

=> Phun 20 phút trước rồi 5 phút sau đó lại.

Chú thích: 32,41 là thông số từ bảng trên.

PHẦN C: ĐẶC TẨY KỸ THUẬT

I. Chức năng hệ thống

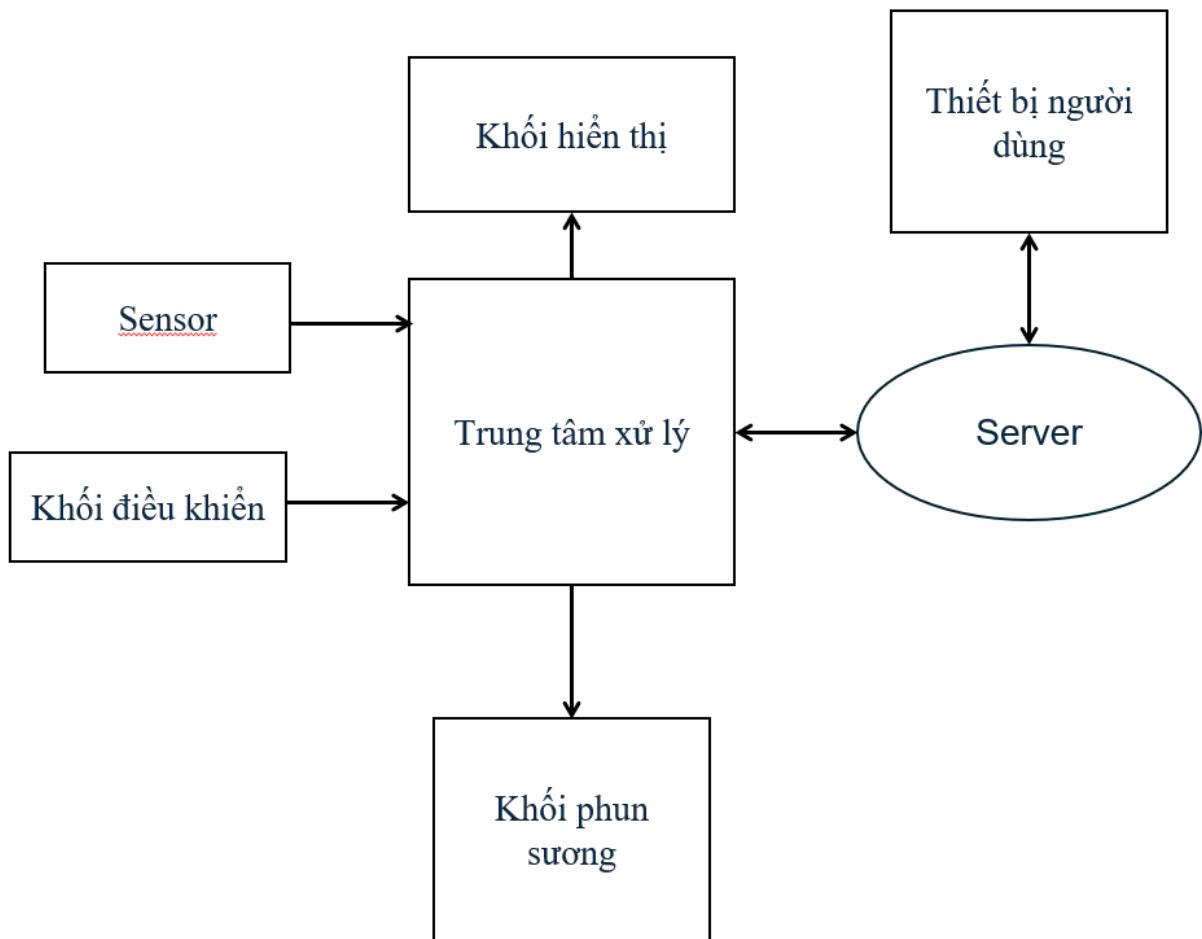
- **Chế độ auto:** Sử dụng dữ liệu từ cảm biến nhiệt độ, độ ẩm để tính toán ra lượng nước, chế độ cần phun. Sử dụng cảm biến hiện diện con người để kiểm tra. Từ đó tự động bật/tắt.
- **Chế độ manual:** Người dùng tùy chỉnh tắt/ bật.
- **Mực nước:** Sử dụng cảm biến siêu âm để kiểm tra lượng nước còn đủ để dùng hay không.
- **Hiển thị:** hiển thị thông số, dữ liệu trên LCD và qua điện thoại
- **Điều khiển trực tiếp:** nút bấm để người dùng có thể điều khiển.
- **Điều khiển qua điện thoại:** sử dụng Blynk, một nền tảng dễ tiếp cận nhanh.

II. Thông số kỹ thuật

- **Dung tích bình chứa nước:** 1 lít.
- **Công suất tối đa:** 5.3W
- **Lượng sương mù tối đa (lưu lượng phun):** 300 ml/h.
- **Diện tích sử dụng:** 10 - 25m²
- **Phù hợp:** Phòng ngủ, phòng khách, phòng làm việc vừa và nhỏ
- **Thời gian hoạt động:** 12 giờ - 24 giờ
- **Mức độ ẩm tương đối (RH):**
 - + 45%-55%: ngưỡng ổn.
 - + Tăng trưởng vi khuẩn (RH < 30% hoặc > 60%).
 - + Tăng trưởng, ổn định và khả năng lây nhiễm của vi-rút (RH < 50% hoặc > 70%).
 - + Tỷ lệ nấm mốc trong không khí nhà ở có liên quan khi RH > 60%.

PHẦN D: KIẾN TRÚC HỆ THỐNG

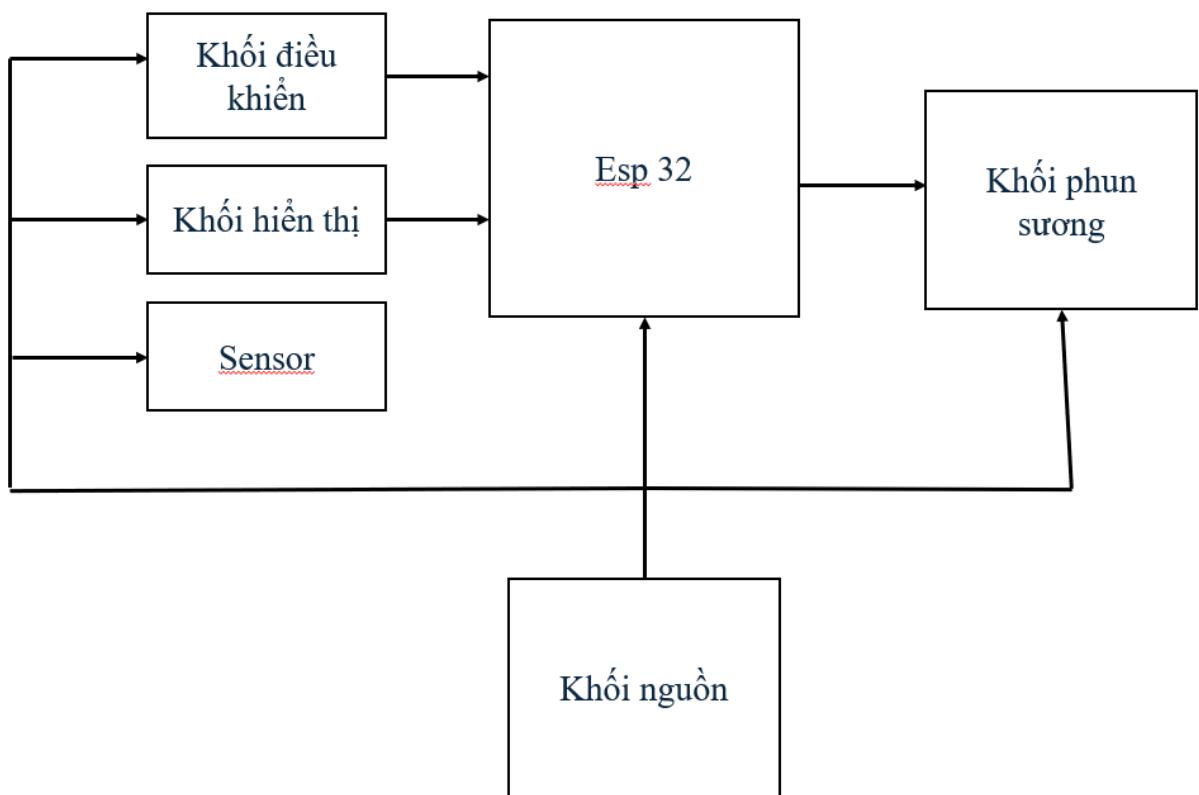
I. Sơ đồ tổng quát



Hình 2: Sơ đồ tổng quát của hệ thống

- **Khối điều khiển**: người dùng trực tiếp sử dụng mà không cần điện thoại để điều khiển
- **Sensor**: Cảm biến lấy dữ liệu từ môi trường, đưa thông tin về trung tâm xử lý
- **Trung tâm xử lý**: xử lý dữ liệu để điều khiển khói phun sương và hiển thị, gửi dữ liệu tới điện thoại
- **Khối hiển thị**: hiển thị trạng thái hiện tại, thông số
- **Thiết bị người dùng**: điều khiển thiết bị từ xa, quan sát dữ liệu

II. Sơ đồ kết nối



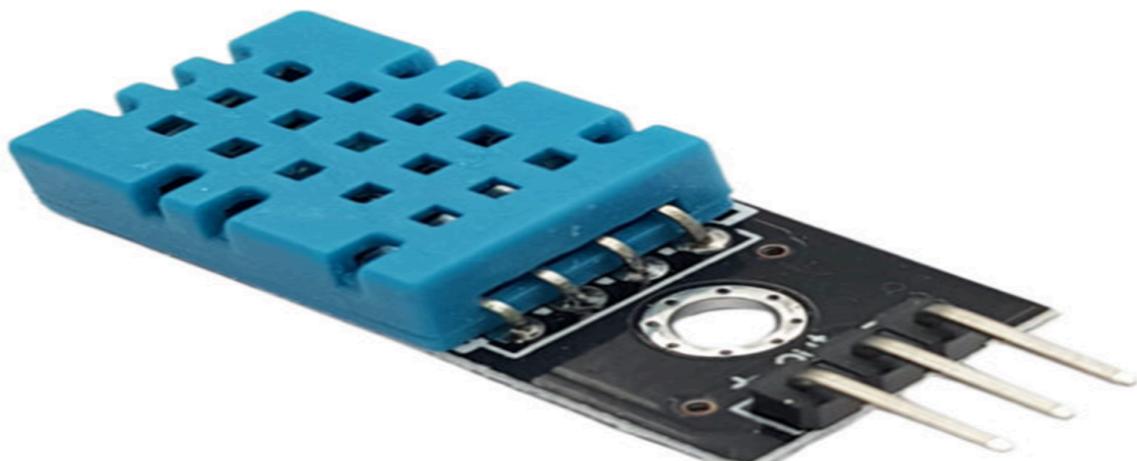
Hình 3: Sơ đồ kết nối

PHẦN E: THIẾT KẾ CHI TIẾT

I. Phần cứng

1. Lựa chọn linh kiện và thông số của linh kiện

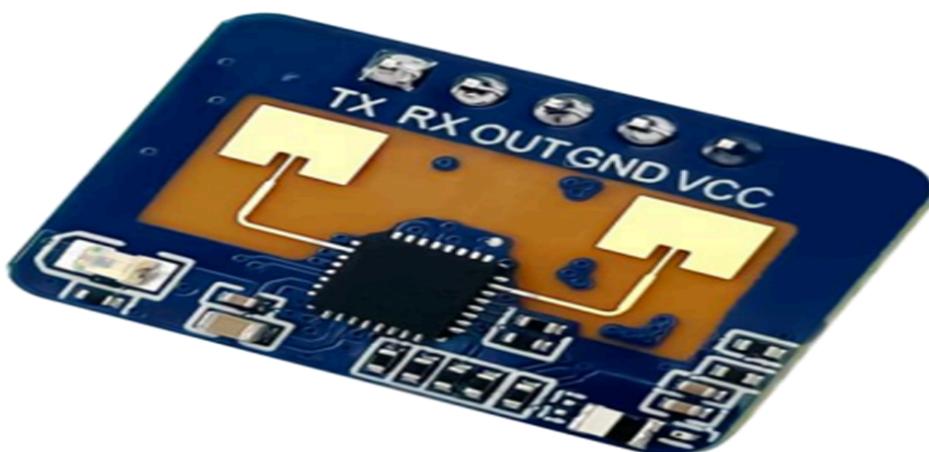
- Cảm biến DHT11:



Hình 4: Cảm biến DHT11

- Điện áp hoạt động: 5V.
- Khoảng đo độ ẩm: 20% – 80% RH, sai số 5% RH.
- Khoảng đo nhiệt độ: 0 - 50°C, sai số $\pm 2^{\circ}\text{C}$.
- Tần số lấy mẫu tối đa 1Hz (1 giây / lần).

- Cảm biến nhận điện con người HLK-LD2410C:



Hình 5: Cảm biến nhận điện con người HLK-LD2410C

- Điện áp hoạt động: 5V.
- Dòng hoạt động: 79mA.
- Tần số: 24GHz-24.25GHz.
- Góc phát hiện lớn, phạm vi bao phủ: $\pm 60^\circ$.
- Khoảng cách phát hiện dài tối: 0.75-6m (có thể điều chỉnh).
- Nhiệt độ hoạt động: -40°C - 85°C.

- Cảm biến siêu âm HY-SRF05:



Hình 6: Cảm biến siêu âm HY-SRF05.

- Điện áp hoạt động: 5V.
- Dòng tiêu thụ: 10-40mA
- Tần số phát sóng: 40Khz
- Góc độ đo lường: 15° .
- Khoảng cách đo được: 2-450cm. Sai số 0.3cm

- Transistor BC547:



Hình 7: Transistor BC547

- Loại transistor: NPN.
- Dòng điện cực thu: $I_C = 100\text{mA}$.
- Điện áp Collector-Base: $V_{CB} = 50\text{V}$
- Điện áp Collector-Emitter: $V_{CE} = 45\text{V}$
- Điện áp Emitter-Base: $V_{EB} = 6\text{V}$
- Nhiệt độ hoạt động và lưu trữ: -65°C đến 150°C .
- Độ lợi dòng điện h_{FE} : 110-800.

- Opto PC817:



Hình 8: Opto PC817.

- Dòng điện Collector: $I_C = 50\text{mA}$
- Điện áp Collector-Emitter: $V_{CEO} = 35\text{V}$
- Điện áp Emitter-Collector: $V_{ECO} = 6 \text{ V}$
- Tổng công suất tiêu tán: $P_{tot} = 200\text{mW}$
- Nhiệt độ hoạt động: -30°C đến 100°C .
- Nhiệt độ lưu trữ: -55°C đến 125°C .

- Relay 5V:



Hình 9: Relay 5V.

- Điện áp hoạt động: 5VDC
- Khả năng chịu tải:

30VDC 10A

a 250VAC 10A

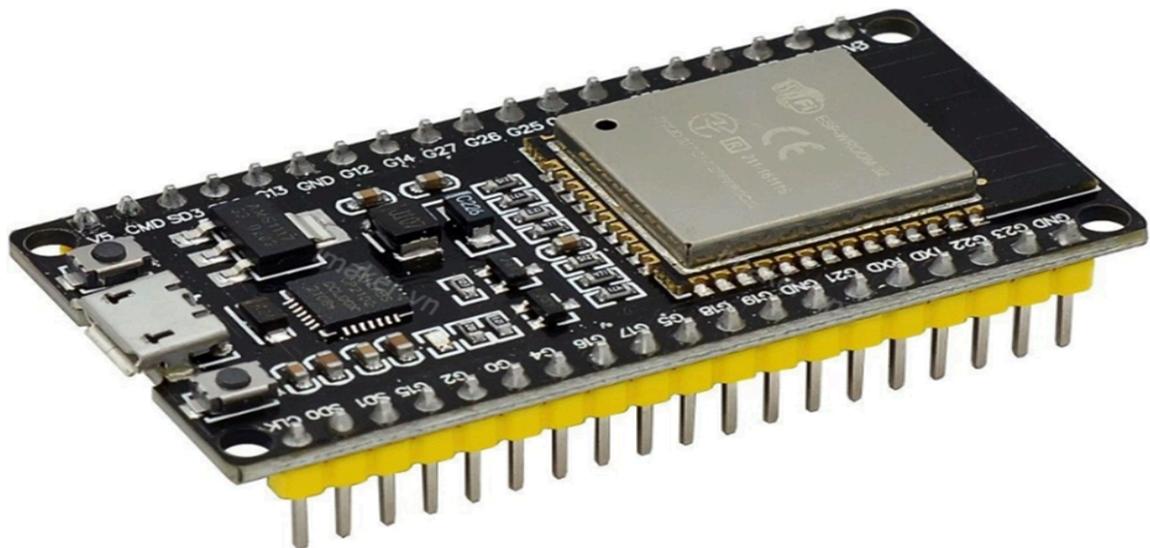
- LCD I2C:



Hình 10: LCD I2C.

- Điện áp hoạt động: 5V.
- Giao diện truyền thông: I2C.
- Loại màn hình: Màn hình LCD.
- Chip điều khiển: AiP31068L.
- Kích thước màn hình: 64,5 x 16,0 mm.
- Kích thước phác thảo: 87,0 x 32,0 x 13,0 (mm).
- Dòng điện hoạt động: 26mA(5V).

- ESP32:



Hình 11: *ESP32 38 chân.*

- Tích hợp crystal: 40MHz crystal.
 - Tích hợp SPI flash: 4MB.
 - Điện áp hoạt động/nguồn cấp: 3V-3.6V.
 - Dòng điện hoạt động: 80mA.
 - Nhiệt độ hoạt động: -40°C đến 85°C.
 - Module giao tiếp: SD card, UART, SPI, SDIO, I2C, LED PWM, Motor PWM, I2S, IR, pulse counter, GPIO, capacitive touch sensor, ADC, DAC, Two-Wire Automotive Interface (TWAI®), compatible with ISO11898-1 (CAN Specification 2.0).

- Bộ phun sương:



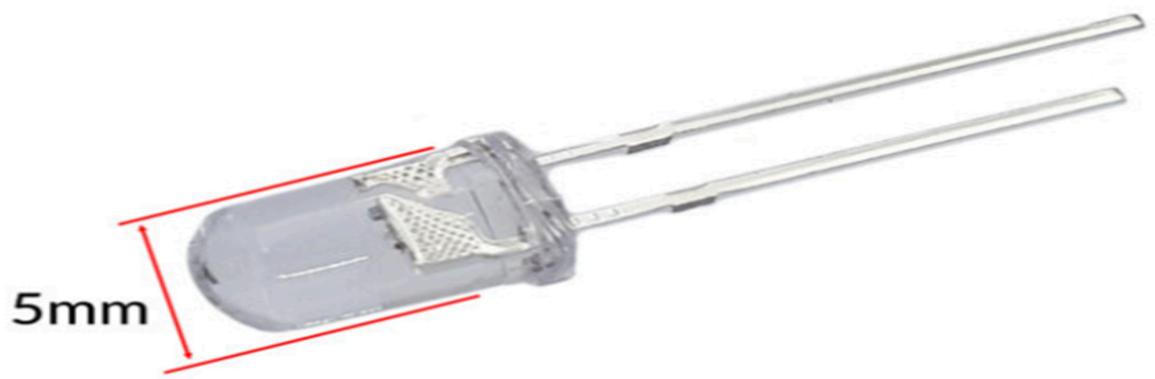
Hình 12: Bô phun sương.

- Cáp nguồn: 5V Type C hoặc 5V trên mạch.
- Dòng tiêu thụ: 300 mA.
- Công suất: 2W.
- Tần số loa siêu âm: 108 - 110 KHz.
- Kích thước board mạch: 30x20x8 mm.
- Kích thước loa:

Đường kính ngoài: 20mm.

Đường kính trong: 12mm.

- LED:



Hình 13: LED.

- Màu: đỏ và xanh lá.
- Dòng hoạt động: 10-20mA.
- Điện áp led: 1.8-2V(đỏ) và 2.8-3.2V(xanh lá).

- Diode 1N4007:



Hình 14: Diode 1N4007.

- Điện áp nghịch đỉnh: 1000 V.
- Dòng điện thuận trung bình: 1A.
- Dòng điện thuận đỉnh không lặp lại: 30A.
- Nhiệt độ hoạt động: -55°C - 175°C.
- Công suất tiêu thụ: 3 W.
- Điện áp thuận: 1.1 V.
- Dòng điện nghịch: 5 uA.

- **Điện trở 240Ω và 1KΩ:**



Hình 15: Điện trở 240Ω (trái) và điện trở 1KΩ (phải)

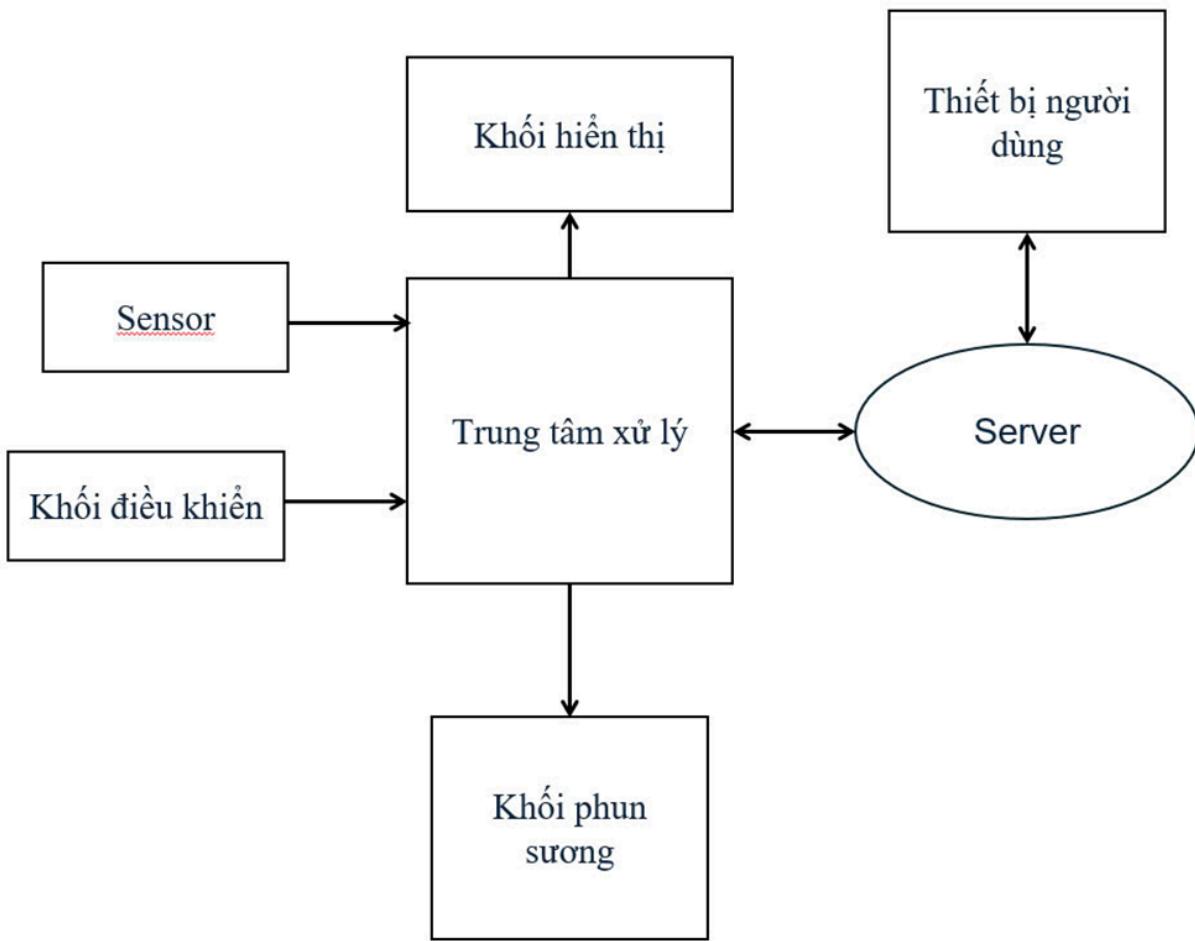
- **Nút nhấn:**



Hình 16: Nút nhấn 2 chân.

- Dòng định mức: 50mA.
- 2 chân.

2. Sơ đồ khối tổng quan



Hình 17: Sơ đồ khái tổng quan.

a. Khối cảm biến:

- Khối cảm biến bao gồm các cảm biến: cảm biến nhiệt độ và độ ẩm DHT11, cảm biến nhận diện con người, cảm biến siêu âm.

- Cảm biến nhiệt độ và độ ẩm DHT11:
 - + Dùng để đo nhiệt độ và độ ẩm căn phòng.
 - + Theo dõi độ ẩm căn phòng để quyết định bật/tắt máy phun sương duy trì độ ẩm lý tưởng.
 - + Tránh phun sương quá mức làm tăng độ ẩm gây mốc gây hại đến sức khỏe.
- Cảm biến nhận diện con người HLK-LD2410C:

+ Phát hiện nhận diện con người, máy phun sương khi ở chế độ auto sẽ được kích hoạt khi phát hiện có người ở trong phòng và tắt khi không có ai giúp tiết kiệm năng lượng và nước.

+ Tăng tính tiện lợi, máy phun sương sẽ tự động bật ngay lập tức mà không cần thao tác thủ công.

- Cảm biến siêu âm HY-SRF05:

+ Dùng để theo dõi dung tích trong bình: Đảm bảo máy phun sương có đủ nước để hoạt động. Khi nước trong bình đạt ngưỡng báo động 0.5 cm, cảm biến phát hiện và ngắt hoạt động máy phun sương tránh làm hư bộ phun.

+ Tự động cảnh báo khi mực nước đạt ngưỡng cảnh báo tới thiết bị người dùng, giúp người dùng có thể nhận biết và thay nước.

b. Khối phun sương:

- Khối phun sương sẽ bao gồm: 2 bộ phun sương, 2 relay, 2 transistor, 2 diode, 2 opto, 2 điện trở.

- Opto PC817:

+ Dùng để bảo vệ mạch khỏi nhiễu điện và hỏng hóc do điện áp cao.

+ Chuyển đổi tín hiệu điện thành tín hiệu quang và sau đó trở lại tín hiệu điện.

- Transistor BC547:

+ Dùng để khuếch đại tín hiệu từ Opto để điều khiển các linh kiện khác.

+ Hoạt động như một công tắc điện tử điều khiển dẫn dòng, cấp điện cho relay.

- Diode:

+ Bảo vệ transistor khỏi điện áp ngược sinh ra bởi cuộn dây relay khi ngừng hoạt động.

+ Khi dòng điện qua cuộn dây bị ngắt, năng lượng cảm ứng trong cuộn dây sẽ sinh ra một xung điện áp cao. Diode này dẫn xung điện áp đó để bảo vệ transistor.

- Relay 5V: Hoạt động như công tắc đóng hoặc cung cấp nguồn cho bộ phun sương dựa trên tín hiệu điều khiển của transistor.
- Bộ phun sương: Dùng để hoạt động phun sương.
- Điện trở: Hạn chế dòng điện vào chân base của transistor, tránh làm hỏng transistor. Đảm bảo dòng điện cung cấp ở mức phù hợp kích hoạt transistor.

c. Khối điều khiển:

- Khối điều khiển bao gồm: 2 nút nhấn. Dùng để thay đổi trạng thái ON/OFF hoặc chế độ MANUAL/AUTO.

d. Khối hiển thị:

- Khối hiển thị bao gồm: 1 LCD giao tiếp bằng I2C, 2 LED, 2 điện trở.

- LCD: Dùng để hiển thị thông tin về nhiệt độ, độ ẩm....
- LED: Dùng để hiển thị trạng thái ON/OFF hoặc hiển thị chế độ MANUAL/AUTO.
- Điện trở: Giới hạn dòng điện đi qua LED, đảm bảo LED không bị cháy, và duy trì độ sáng phù hợp.

e. SERVER và thiết bị người dùng:

- Server: Sử dụng Blynk cloud và app Blynk để làm giao diện thao tác giữa thiết bị người dùng tới thiết bị phun sương điều khiển qua app Blynk và sử dụng Blynk cloud làm lưu trữ.

- Thiết bị người dùng: những đồ điện tử thông minh điện thoại, máy tính bảng, laptop...

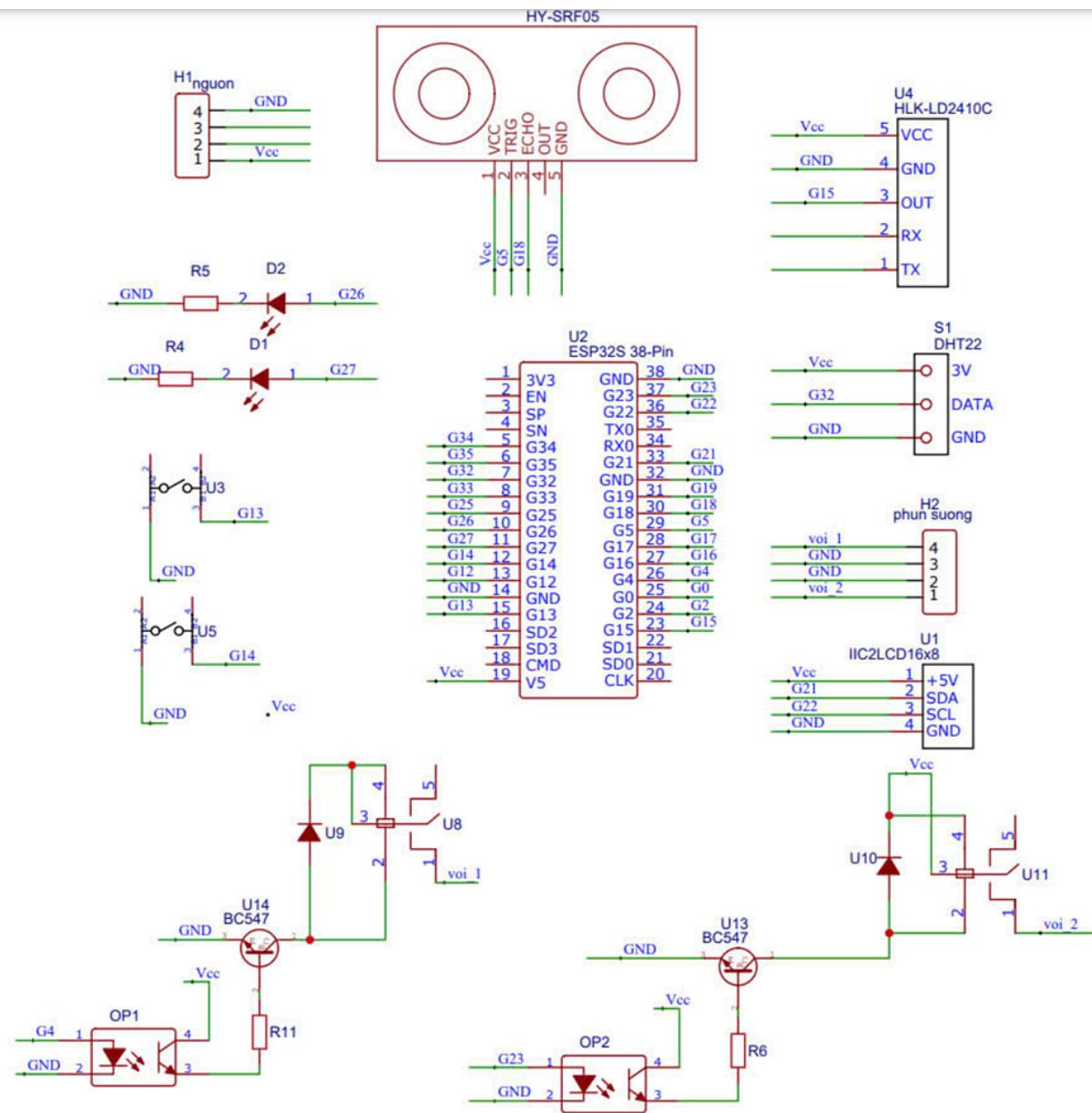
f. Trung tâm xử lý:

- ESP32 đóng vai trò là bộ điều khiển trung tâm, cung cấp các tín hiệu điều khiển.

- 1 nguồn cung cấp 5V.

Tổng hợp:

- Sơ đồ mạch linh kiện tổng quan:



Hình 18: Sơ đồ mạch linh kiện tổng quan.

- Bảng tổng hợp các linh kiện:

Bảng 1: Tổng hợp thông số và nhiệm vụ của các linh kiện

ID	Tên linh kiện	Thông số	Nhiệm vụ
1	ESP32 NodeMCU LuaNode32 Module	Điện áp hoạt động: 5V Dòng cung cấp: 0,5A	Xử lý tín hiệu, điều khiển hệ thống.
2	DHT11	Điện áp hoạt động: 5V Dòng cung cấp: 2,6mA	Cảm biến đo nhiệt độ, độ ẩm.

3	Opto PC817	Dòng tối đa qua collector (Ic): 50 mA	Hỗ trợ điều khiển BJT.
4	Transistor BC547	Dòng Base tối đa (IB): 5 mA Hệ số khuếch đại dòng (hFE): 110 - 800 (tùy điều kiện hoạt động)	Hỗ trợ điều khiển relay.
5	Relay 5V	Công suất: 0.45W	Điều khiển mạch phun sương.
6	LCD I2C	Điện áp hoạt động: 5V Dòng cung cấp: 26mA	Hiển thị thông số.
7	Bộ phun sương	Điện áp hoạt động: 5V Dòng cung cấp: 0,4A	Điều khiển phun sương.
8	Cảm biến HY-SRF05	Điện áp hoạt động: 5V Dòng cung cấp: 40mA	Đo mực lượng nước.
9	Điện trở 1kΩ	Trở kháng: 1kΩ, 1/2W	Hạn dòng cho BJT.
10	Điện trở 240Ω	Trở kháng: 240Ω, 1/2W	Hạn dòng cho led.
11	Led xanh lá	Điện áp cấp: 2,2V	Hiển thị trạng thái on/off.
12	Led đỏ	Điện áp cấp: 2,2V	Hiển thị trạng thái auto/manual.
13	Nút nhấn	Dòng định mức tiếp điểm: 50 mA	Điều khiển.
14	Cảm biến HLK LD 2410c	Điện áp hoạt động: 5V Dòng cung cấp: 79mA	Cảm biến hiện diện con người.
15	Diode 1N4007	Điện áp ngược lặp lại tối đa là: 1000 V	Tránh điện áp ngược.

Bảng 2: Tổng hợp chi phí và số lượng các linh kiện

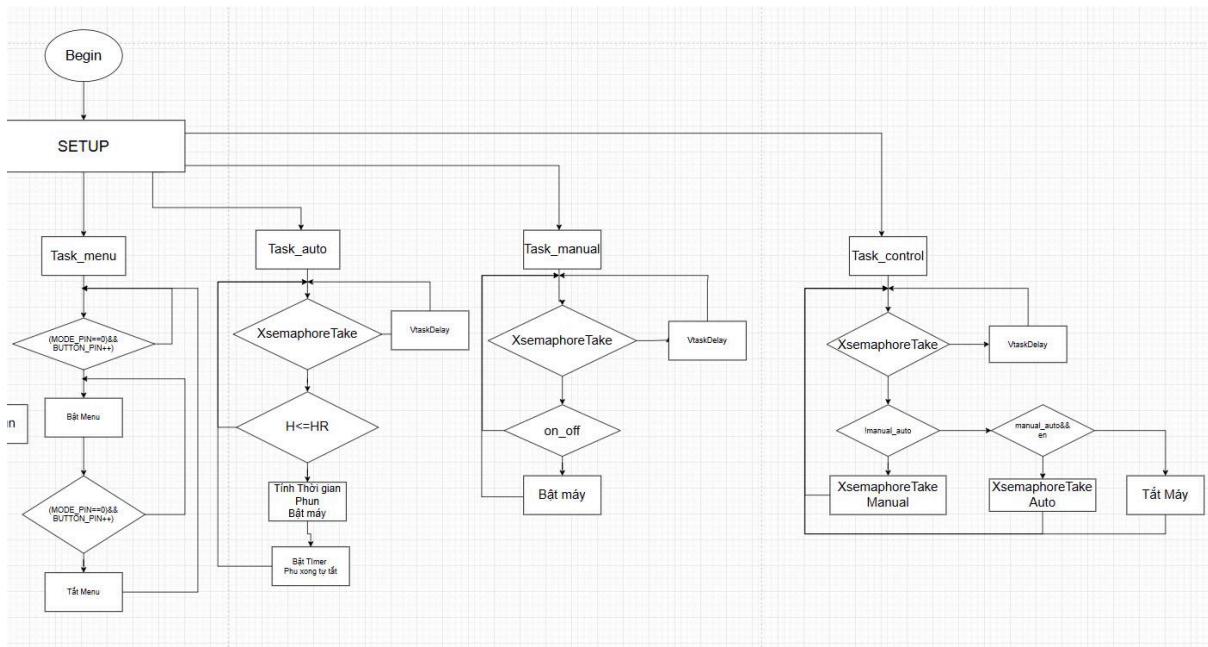
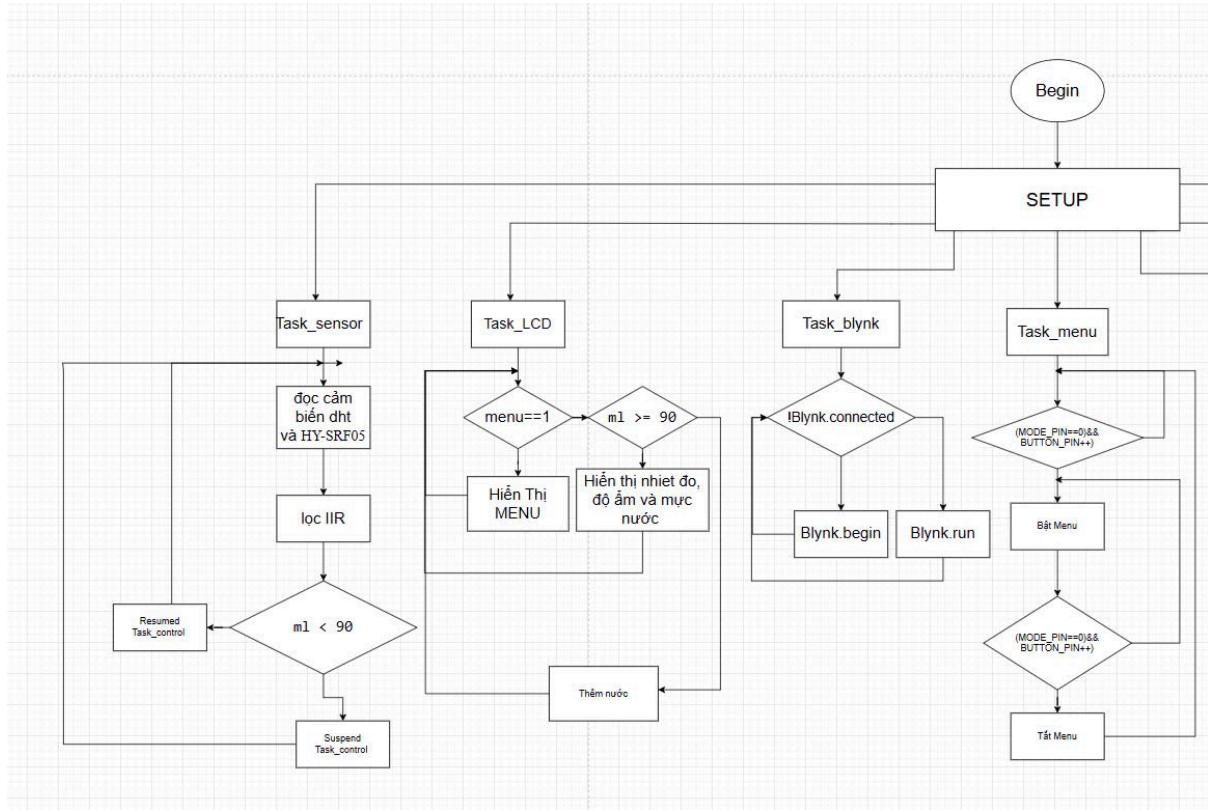
ID	Tên linh kiện	Giá tiền (đồng)/đơn vị	Số lượng	Tổng tiền(đồng)
1	ESP32 NodeMCU LuaNode32 Module	110k	1	110k

2	DHT11	14k	1	14k
3	Opto PC817	2,6k	2	5k2
4	Transistor BC547	300	2	600
5	Relay 5V	7k	2	14k
6	LCD I2C	56k	1	56k
7	Bộ phun sương	22k	2	44k
8	Cảm biến HY-SRF05	24k	1	24k
9	Điện trở 1kΩ	500	2	1k
10	Điện trở 240Ω	200	2	400
11	Led xanh lá	250	1	250
12	Led đỏ	200	1	200
13	Nút nhấn	600	2	1k2
14	Cảm biến HLK LD 2410c	120k	1	120k
15	Diode 1N4007	200	2	400

Tổng chi phí cho các linh kiện là 390k đồng.

II. Phần mềm

1. Lưu đồ giải thuật



Hình 19: Lưu đồ giải thuật

2. Code chương trình

```

#define BLYNK_TEMPLATE_ID "TMPL6hxy5H-jj"
#define BLYNK_TEMPLATE_NAME "haq"
#define BLYNK_AUTH_TOKEN "2fYYz9ECNTT5eUXboKxN8zXQ-r-EtI36"

#include <DHT.h>
#include <Wire.h>
#include <BlynkSimpleEsp32.h>
#include <LiquidCrystal_I2C.h>
#include <Adafruit_Sensor.h>
#include <EEPROM.h>

#define DHTPIN 32
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

LiquidCrystal_I2C lcd(0x27,16,2);
const char* ssid = "Luan";
const char* pass = "12345678";
unsigned long last_time;
unsigned long last1;
float muc_nuoc;
unsigned long lastDebounceTime = 0;
float nhietdo=0.0;
float doam=0.0;
float filterednhietdo = 0.0;
float filtereddoam = 0.0;
float ml;
#define trig 5
#define echo 18
#define MODE 26
#define EN_PIN 15
#define MODE_PIN 13
#define BUTTON_PIN 14
#define LED_PIN 27
#define RELAY1_PIN 4
#define RELAY2_PIN 23
float ratio[] = {
    3.82, 4.15, 4.49, 4.86, 5.26, 5.26,
    5.69, 6.16, 6.65, 7.18, 7.75, 8.36,
    9.01, 9.71, 10.46, 10.46, 11.26, 12.12,
    13.03, 14.01, 15.06, 16.17, 17.37, 18.64,
    20.00, 20.00, 21.45, 22.99, 24.64, 26.40,
    28.27, 30.27, 32.41, 34.68, 37.10, 37.10,
    39.68, 42.44, 45.37
};

```

```

TaskHandle_t myTaskHandle = NULL;
TimerHandle_t myTimer; // Handle của timer
volatile bool on_off = false;
volatile bool manual_auto = false;
volatile bool en = false;
SemaphoreHandle_t sema_auto;
SemaphoreHandle_t sema_manual;
SemaphoreHandle_t control;
bool menu = 0;
// Hệ số alpha cho bộ lọc IIR thông thấp
float alpha = 0.1;
unsigned long tmenu=0;
unsigned char m1=0;
char HR= 45;
unsigned char k2=0;
unsigned char m2=0;
unsigned char m3=0;
unsigned char m4=0;
unsigned char m5=0;

void TimerCallback(TimerHandle_t xTimer) {
    if((manual_auto)&&( digitalRead(LED_PIN))) {
        Serial.println("callback tắt máy");
        on_off=0;
        digitalWrite(LED_PIN,on_off);
        digitalWrite(RELAY1_PIN,on_off);
        digitalWrite(RELAY2_PIN,on_off);
        xSemaphoreGive(control);
    }
}

void IRAM_ATTR button_on_off() { // ISR xử lý nút nhấn
    BaseType_t xHigherPriorityTaskWoken = pdFALSE;
    if (millis() - lastDebounceTime > 70) {
        xSemaphoreGiveFromISR(control, &xHigherPriorityTaskWoken); // Cấp
semaphore từ ngắt
        on_off = !on_off; // Đảo trạng thái
        //if (xHigherPriorityTaskWoken == pdTRUE) {
            lastDebounceTime = millis(); // Cập nhật thời gian ngắt cuối

        if(menu){
            if(m1&&!m2&&!m3&&!m4&&!m5){
                m1++;if(m1==3)m1=1;
            }else if(m2&&!m3&&!m4&&!m5&&(m1==1)){
                m2++;if(m2==5)m2=1;
            }
        }
    }
}

```

```

}else if(m3&&!m2&&!m4&&!m5&&(m1==2)){
    m3++;if(m3==5)m3=1;
}
}

}

last_time=millis();
portYIELD_FROM_ISR(); // Đảm bảo task có thể chạy ngay lập tức nếu
cần
}
void IRAM_ATTR mode_manual_auto() {
    BaseType_t xHigherPriorityTaskWoken = pdFALSE;
    if ( millis() - lastDebounceTime > 70) {
        xSemaphoreGiveFromISR(control, &xHigherPriorityTaskWoken); // Cấp
semaphore từ ngắt
        manual_auto = !manual_auto; // Đảo trạng thái
        if(menu){
            if(m1==0)m1=1;
            else if(m1==1){
                if(m2==0)m2=1;
                else if(m2==1)HR++;if(HR>=101)HR=0;
                else if(m2==3)k2=1;
                else if(m2==4)m2=0;
            }else if(m1==2){
                if(m3==0)m3=1;
                else if(m3==4)m3=0;
            }
        }
        lastDebounceTime = millis();
    }
    last_time=millis();
    portYIELD_FROM_ISR();
}
void IRAM_ATTR en_auto() {
    BaseType_t xHigherPriorityTaskWoken = pdFALSE;
    if ( millis() - lastDebounceTime > 100) {
        xSemaphoreGiveFromISR(control, &xHigherPriorityTaskWoken); // Cấp
semaphore từ ngắt
        lastDebounceTime = millis(); // Cập nhật thời gian ngắt cuối
        en = !en; // Đảo trạng thái
        //if (xHigherPriorityTaskWoken == pdTRUE) {
        //}
        last_time=millis()-60001;
        portYIELD_FROM_ISR(); // Đảm bảo task có thể chạy ngay lập tức nếu

```

```

cần
}

BLYNK_WRITE(V0) { // Nhận tín hiệu từ virtual pin V1
    bool blynkState = param.asInt(); // Lấy trạng thái từ Blynk
    // Cấp semaphore để thay đổi trạng thái máy
    //if (xSemaphoreTake(xStateSemaphore, portMAX_DELAY)) {
        on_off = blynkState; // Cập nhật trạng thái máy
        xSemaphoreGive(control); // Trả lại semaphore
        Serial.println(blynkState ? "on_off: ON" : "on_off: OFF");
        vTaskDelay(1);
    //}
}

BLYNK_WRITE(V3) { // Nhận tín hiệu từ virtual pin V1
    bool blynkState = param.asInt(); // Lấy trạng thái từ Blynk
    // Cấp semaphore để thay đổi trạng thái máy
    //if (xSemaphoreTake(xStateSemaphore, portMAX_DELAY)) {
        manual_auto = blynkState; // Cập nhật trạng thái máy
        xSemaphoreGive(control); // Trả lại semaphore
        Serial.println(blynkState ? "mode: manual" : "mode: auto");
    //}
    vTaskDelay(1);
}

// Task xử lý trạng thái máy
void task_manual(void* arg){
    while (1) {
        if (xSemaphoreTake(sema_manual, portMAX_DELAY)){
            digitalWrite(LED_PIN, on_off);
            digitalWrite(RELAY1_PIN, on_off);
            digitalWrite(RELAY2_PIN, on_off);
            Blynk.virtualWrite(V3, manual_auto);
            Blynk.virtualWrite(V0, on_off);
            Serial.println(on_off ? "manual ON" : "manual OFF");

        }
    }
}

void task_menu(void* arg){
    while (1) {
        tmenu=millis();

        while((digitalRead(MODE_PIN)==0)&&(digitalRead(BUTTON_PIN)==0)&&men

```

```

u==0){
    Serial.println("Task is Suspend");
    digitalWrite(LED_PIN, 0);
    digitalWrite(RELAY1_PIN,0);
    digitalWrite(RELAY2_PIN,0);
    Serial.println("ktra menu");
    vTaskDelay(1000);
    if((millis()-tmemo >1000)&&menu==0){
        vTaskSuspend(myTaskHandle);
        vTaskDelay(1);
        Serial.println("bat menu");
        menu=1;
        while(1){
            Serial.println("menu");
            Serial.println(menu);
            vTaskDelay(1000);

if((digitalRead(MODE_PIN)==0)&&(digitalRead(BUTTON_PIN)==0)&&menu==1){
            vTaskDelay(10);
            if(millis()-tmemo >1000){
                xSemaphoreGive(control);
                vTaskResume(myTaskHandle);
                Serial.println("Task is resumed");
                break;
            }else{ tmemo=millis(); }
        }

        }
    }
}

menu=0;tmemo=0;m1=0;k2=0;m2=0;m3=0;m4=0;m5=0;
vTaskDelay(100);
}

}

void task_control(void* arg){
    while (1) {
        if (xSemaphoreTake(control, portMAX_DELAY) == pdTRUE) {
            if (!manual_auto){
                digitalWrite(MODE,manual_auto);
                Serial.println(on_off);
                Serial.println("GIVE sema_manual");
            }
        }
    }
}

```

```

        xSemaphoreGive(sema_manual);
    }else if ((manual_auto)&&(digitalRead(15))){
        Blynk.virtualWrite(V3, manual_auto);
        Blynk.virtualWrite(V0, on_off);
        digitalWrite(MODE,manual_auto);
        xSemaphoreGive(sema_auto);
        Serial.println("GIVE sema_auto");
    }else if((manual_auto)&&(!digitalRead(15))){
        on_off=0;
        Blynk.virtualWrite(V3, manual_auto);
        Blynk.virtualWrite(V0, on_off);
        digitalWrite(LED_PIN, on_off);
        digitalWrite(RELAY1_PIN,on_off);
        digitalWrite(RELAY2_PIN,on_off);
        digitalWrite(MODE,manual_auto);
    }
}
vTaskDelay(10);

}

void task_auto(void* arg) {
    HR =EEPROM.readChar(1);
    last_time = millis();
    while (1) {
        if (xSemaphoreTake(sema_auto, portMAX_DELAY)) {
            if(millis()- (last_time + 60000) > 0){
                if(filtereddoam <= HR){
                    Serial.println(en);
                    Serial.println(on_off);
                    unsigned int t
=((float)HR-filtereddoam)*(1.0/100)*(ratio[(int)nhietdo])*(245.0/2)*(1.0/2)*(1.0/3
00)*(60*60*1000);
                    Serial.printf("thoi gian phun: "); Serial.print(t); Serial.println("ms");
                    on_off=digitalRead(15);
                    digitalWrite(LED_PIN, on_off);
                    digitalWrite(RELAY1_PIN,on_off);
                    digitalWrite(RELAY2_PIN,on_off);
                    Blynk.virtualWrite(V0, on_off);
                    Serial.println(en);
                    Serial.println(on_off);
                    Serial.println(on_off ? "auto ON" : "auto OFF");
                    last_time = t + millis();
                }
            }
        }
    }
}

```

```

if (myTimer != NULL) {
    xTimerDelete(myTimer, 0); // Xóa timer cũ
}
myTimer = xTimerCreate("MyTimer", pdMS_TO_TICKS(t), pdFALSE,
(void*)0, TimerCallback);
if (myTimer == NULL) {
    Serial.println("Không thể tạo timer.");
    return;
}
if (xTimerStart(myTimer, 0) != pdPASS) {
    Serial.println("Không thể bắt đầu timer.");
    return;
}
}

void blynk_Task(void* arg) {
last1=millis();
while (1) {
    if (!Blynk.connected()) {
        Serial.println("Blynk disconnected. Reconnecting...");
        Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);
    }
    Blynk.run();
    if (millis() - last1 >= 3000){
        Blynk.virtualWrite(V2, filterednhietdo);
        Blynk.virtualWrite(V1, filtereddoam);
        Blynk.virtualWrite(V4, ml);
        last1 = millis();
    }
    vTaskDelay(10);
}
}

void LCD_Task(void* arg) {
lcd.setCursor(0,0);
lcd.printf("xin chao!toi la:");
lcd.setCursor(0,1);
lcd.printf("<may phun suong>");
lcd.display();
vTaskDelay(4000);
unsigned int tt;
}

```

```

while (1){
if(menu){
Serial.println("*****");
Serial.println(m1);
Serial.println(m2);
Serial.println(m3);
Serial.println(m4);
Serial.println(m5);
Serial.println("_____");
switch(m1){
case 0:
//lcd.clear();
lcd.setCursor(0,0);
lcd.printf(" N1 & N2 EXIT ");
lcd.setCursor(0,1);
lcd.printf("N1:Menu/N2:XUONG");
lcd.display();
break;
case 1:
switch(m2){
case 0:
lcd.setCursor(0,0);
lcd.printf(">DO AM MONG MUON");
lcd.setCursor(0,1);
lcd.printf("THONG SO      ");
lcd.display();
break;
case 1:
tt=millis();
while(digitalRead(MODE_PIN)==0){
if(millis()-tt>50){
HR++;
if(HR>=101)HR=0;
lcd.setCursor(0,0);
lcd.printf(">SET H = %d%%    ",HR);
lcd.setCursor(0,1);
lcd.printf("Khuyen dung 45%%");
lcd.display();
tt=millis();
}
vTaskDelay(50);
}
lcd.setCursor(0,0);
lcd.printf(">SET H = %d%%    ",HR);
lcd.setCursor(0,1);
}
}
}

```

```

lcd.printf("Khuyen dung 45%%");
lcd.display();
break;
case 2:
lcd.setCursor(0,0);
lcd.printf(">Khuyen dung 45%");
lcd.setCursor(0,1);
lcd.printf("SAVE<--      ");
lcd.display();
break;
case 3:
if(!k2){
lcd.setCursor(0,0);
lcd.printf(">SAVE<--      ");
lcd.setCursor(0,1);
lcd.printf("BACK-->      ");
lcd.display();
tt=millis();
}else if(k2){
if(millis()-tt>1500){k2=0;}
lcd.setCursor(0,0);
lcd.printf("  DA LUU  ");
lcd.setCursor(0,1);
lcd.printf("H = %d%%      ",HR);
lcd.display();
EEPROM.writeChar(1,HR);
EEPROM.commit();
}
break;
case 4:
lcd.setCursor(0,0);
lcd.printf("SAVE<--      ");
lcd.setCursor(0,1);
lcd.printf(">BACK-->      ");
lcd.display();
break;
}
break;
case 2:
switch(m3){
case 0:
lcd.setCursor(0,0);
lcd.printf("DO AM MONG MUON ");
lcd.setCursor(0,1);
lcd.printf(">THONG SO      ");

```

```

lcd.display();
break;
case 1:
lcd.setCursor(0,0);
lcd.printf("Dung tich:1,5lit");
lcd.setCursor(0,1);
lcd.printf("Cong suat:11,5W ");
lcd.display();
break;
case 2:
lcd.setCursor(0,0);
lcd.printf(" Toc do phun  ");
lcd.setCursor(0,1);
lcd.printf(" toi da 300ml/h ");
lcd.display();
break;
case 3:
lcd.setCursor(0,0);
lcd.printf("Dien tich sudung");
lcd.setCursor(0,1);
lcd.printf(" 10 - 25m2  ");
lcd.display();
break;
case 4:
lcd.setCursor(0,0);
lcd.printf(">BACK-->      ");
lcd.setCursor(0,1);
lcd.printf("-----");
lcd.display();
break;
}
break;
}
}
else{
if(ml>=90){
lcd.setCursor(0,0);
lcd.printf("T:%.1f°C/H:%.1f%%",filterednhietdo,filtereddoam);
lcd.setCursor(0,1);
lcd.printf("Con %.2f ml  ",ml);
lcd.display();
}else{
//lcd.clear();
lcd.setCursor(0,0);
lcd.printf("  VUI LONG  ");
lcd.setCursor(0,1);
}
}

```

```

lcd.printf(" THEM NUOC!! ");
lcd.display();
}
}

vTaskDelay(100);
}

}

void read_sensor(void* arg) {
bool ab=0;
float thoi_gian;
while (1) {
nhietdo = dht.readTemperature();
doam = dht.readHumidity();

if (isnan(filterednhietdo) || isnan(filtereddoam)) {
filterednhietdo = dht.readTemperature();
filtereddoam = dht.readHumidity();
}
//Áp dụng bộ lọc thông thấp IIR cho dữ liệu đọc
filterednhietdo = alpha*nhietdo + (1.0-alpha)*filterednhietdo;
filtereddoam = alpha*doam+(1.0-alpha)*filtereddoam;
digitalWrite(trig, 0);
delayMicroseconds(2);
digitalWrite(trig, 1);
delayMicroseconds(10);
digitalWrite(trig, 0);
thoi_gian = pulseIn(echo, HIGH);
muc_nuoc= (thoi_gian/2/29.412);
ml=(6.0-muc_nuoc)*18.0*10.0;
vTaskDelay(1);
if ((ml < 90 )&&(ab==0)){
ab=1;
Serial.println("Task is Suspend");
digitalWrite(LED_PIN, 0);
digitalWrite(RELAY1_PIN,0);
digitalWrite(RELAY2_PIN,0);
vTaskSuspend(myTaskHandle);
}else if ((ml >=90 ) &&(ab==1)){
ab=0;
xSemaphoreGive(control);
vTaskResume(myTaskHandle);
Serial.println("Task is resumed");
}
vTaskDelay(1000);
}

```

```

}

}

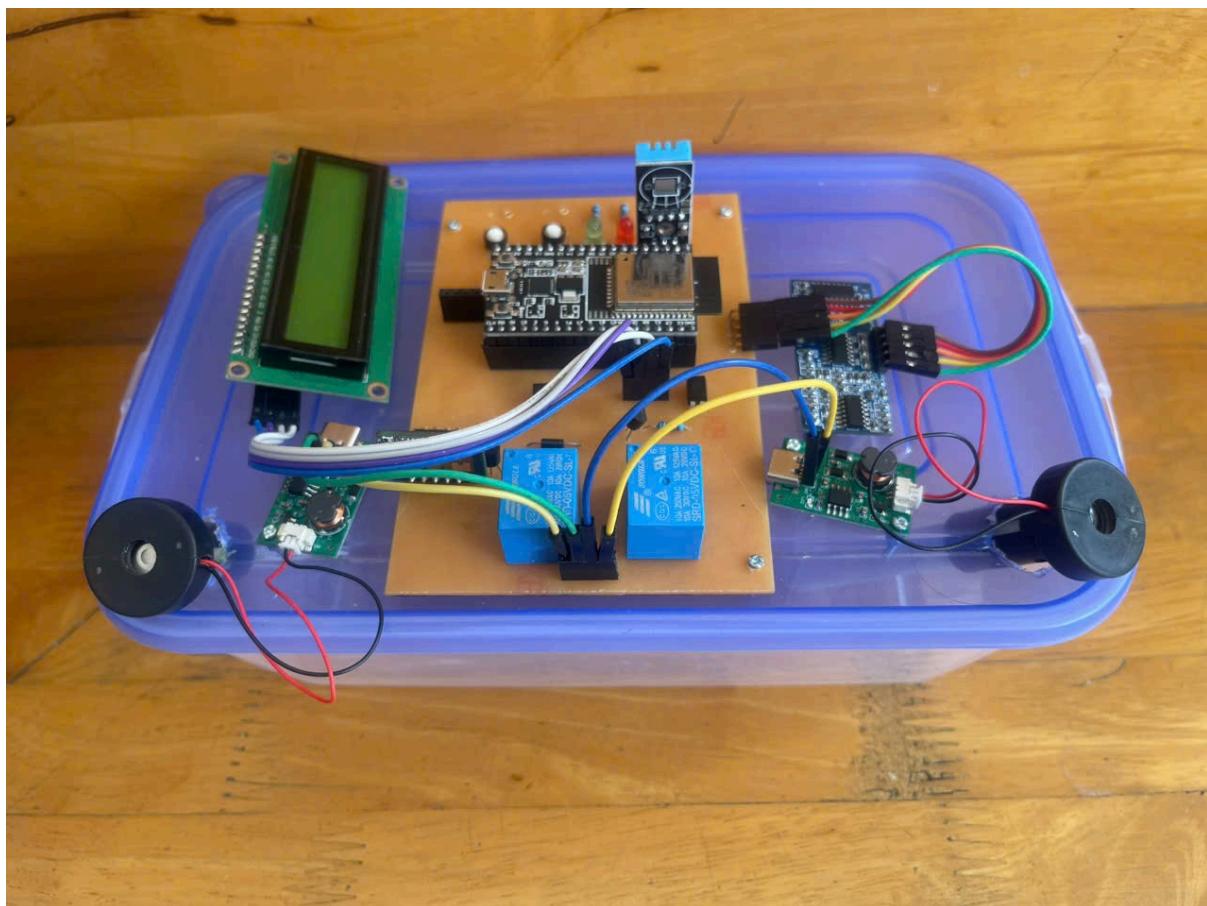
void setup(){
    Serial.begin(9600);
    pinMode(EN_PIN, INPUT_PULLUP);
    pinMode(MODE, OUTPUT);
    pinMode(MODE_PIN, INPUT_PULLUP);
    pinMode(BUTTON_PIN, INPUT_PULLUP);
    pinMode(LED_PIN, OUTPUT);
    pinMode(RELAY1_PIN, OUTPUT);
    pinMode(RELAY2_PIN, OUTPUT);
    pinMode(trig,OUTPUT);
    pinMode(echo,INPUT);
    lcd.init();
    lcd.backlight();
    dht.begin();
    //Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);
    EEPROM.begin(8); // ESP32 cần khai báo dung lượng EEPROM
    filterednhietdo = dht.readTemperature();
    filtereddoam = dht.readHumidity();
    sema_auto = xSemaphoreCreateBinary();
    sema_manual= xSemaphoreCreateBinary();
    control = xSemaphoreCreateBinary();
    //on_off= xSemaphoreCreateBinary();
    attachInterrupt(BUTTON_PIN, button_on_off, FALLING); // Gắn ngắt ngoài cho
    nút nhấn
    attachInterrupt(MODE_PIN, mode_manual_auto, FALLING); // Gắn ngắt ngoài
    cho nút nhấn
    attachInterrupt(EN_PIN, en_auto, CHANGE);
    HR= EEPROM.readChar(1);
    xTaskCreate(task_control,"Task_control",4098,NULL,4, &myTaskHandle );
    xTaskCreate(task_manual,"Task_on_off",4098,NULL,3,NULL);
    xTaskCreate(task_auto,"task_auto",4098,NULL,3,NULL);
    xTaskCreate(read_sensor,"Task_sensor",2048,NULL,2,NULL);
    xTaskCreate(LCD_Task,"Task_LCD",4096,NULL,2,NULL);
    xTaskCreate(task_menu,"Task_menu",4096,NULL,4,NULL);
    xTaskCreate(blynk_Task,"Task_blynk",4096,NULL,3,NULL);
    Serial.println("Setup complete.");
}

void loop() {}

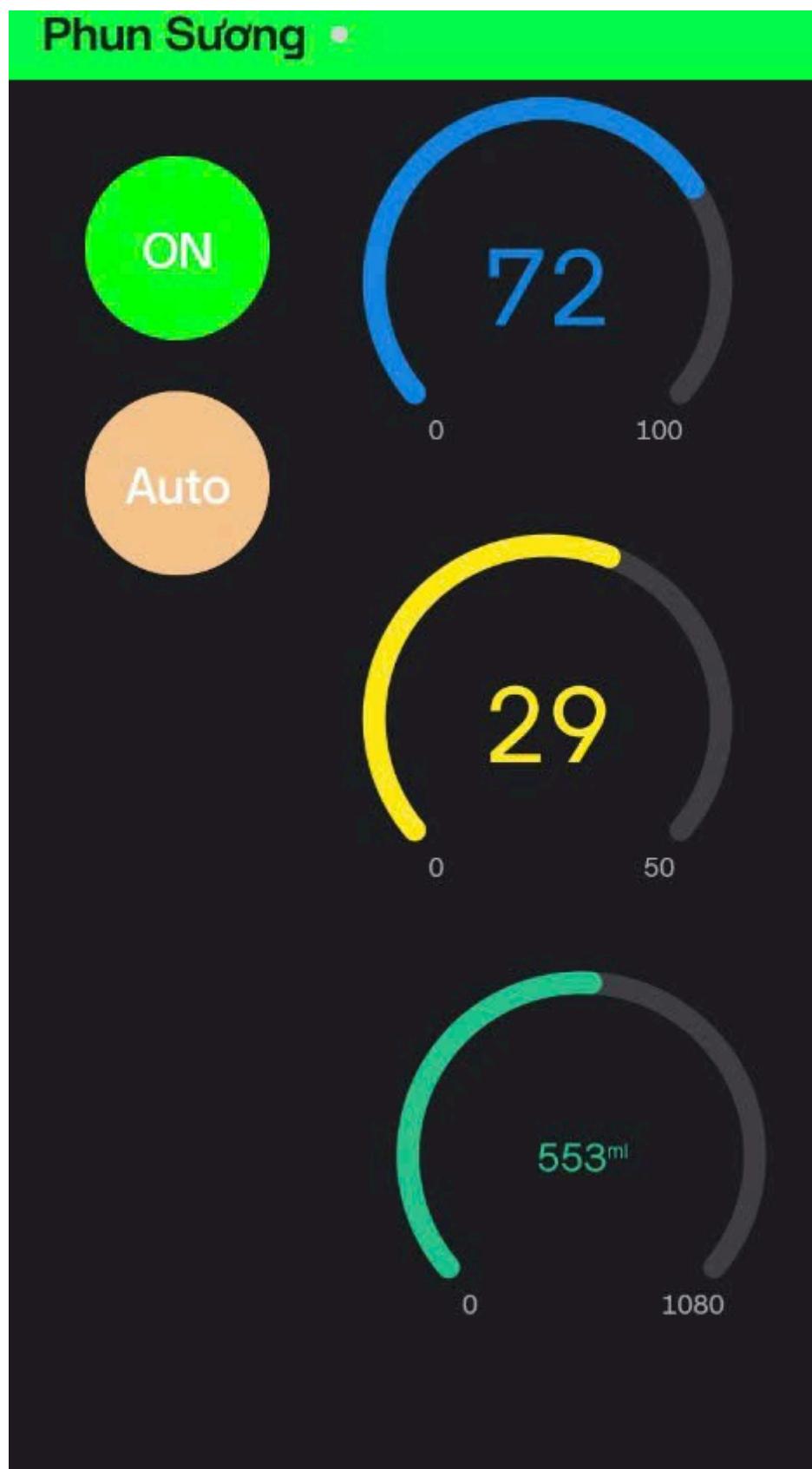
```

PHẦN F: SẢN PHẨM CUỐI CÙNG

I. Hình ảnh sản phẩm thực tế



Hình 20: Sản phẩm thực tế



Hình 21: Giao diện phần mềm điều khiển

II. Đánh giá sản phẩm

- Hệ thống hoạt động đúng với các yêu cầu đã đề ra.
- Công suất tiêu thụ: 5,3W
- Điện năng tiêu thụ để phun hết 1 lít nước: $A = P.t = 5,3/1000*10/3 = 0,0177 \text{ kWh}$

***Hạn chế:** Quá trình kiểm thử chưa đa dạng trong nhiều môi trường khác nhau, phần cứng hệ thống chưa tối ưu hết cỡ.

Bảng 3: Thống kê kết quả đo và sai số

Lần đo	Kết quả đo (%)	DELTA	Nhiệt độ (độ C)	DELTA
1	40,2	2.8	27	1
2	40,1	2.9	27.2	0.8
3	40,1	2.9	27.6	0.4
4	40,5	2.5	27.9	0.1
5	41,1	1.9	28.1	0.1
6	41,3	1.7	28.3	0.3
7	41,7	1.3	28.7	0.7
8	42	1	28.9	0.9
9	42	1	29	1
10	42,8	0.2	29.1	1.1
11	43,2	0.2	29.3	1.3
12	43,4	0.4	29.4	1.4
13	43,9	0.9	29.7	1.7
14	44,1	1.1	29.8	1.8
15	44,3	1.3	30	2

16	44,6	1.6	30.2	2.2
17	45,1	2.1	30.4	2.4
18	45,5	2.5	30.7	2.7
19	46	3	30.8	2.8
20	45,9	2.9	31	3
Sai số	1.71		1.385	

PHẦN KẾT LUẬN

Nhìn chung, hệ thống phun sương do nhóm thiết kế đã hoạt động đúng với những yêu cầu đề ra ban đầu dù còn nhiều hạn chế, quá trình kiểm thử chưa đa dạng trong nhiều môi trường khác nhau. Định hướng tiếp theo của nhóm là sẽ khắc phục được các vấn đề còn tồn động và có thể phát triển thêm những tính năng mới. Đây cũng là kinh nghiệm quý giá của nhóm để hướng tới thiết kế và phát triển những hệ thống hay và phức tạp hơn trong tương lai.

TÀI LIỆU THAM KHẢO

[1] Tables of Saturation Mixing Ratios: Bảng độ ẩm bão hòa:
<https://iuearth.com/e144/index.php?id=mixing-ratio>

[2] Datasheet ESP32: Espressif Systems, *ESP32-WROOM-32*, 2023

[3] Datasheet HLK LD2410c: Shenzhen Hi-Link Electronic, *HLK-LD2410C Human Presence Motion Module User Manual*, 2022

[4] Datasheet PC817: Sharp Corporation, *PC817*, 1995

[5] Datasheet BC547: Fairchild Semiconductor, *BC547, NPN Epitaxial Silicon Transistor*, 2002

[6] Datasheet DHT11: Lady Ada, *DHT11, DHT22 and AM2302 Sensors*, 2024

[7] Datasheet HY-SRF 05: *HY-SRF05 Precision Ultrasonic Sensor*

[8] Datasheet Relay 5V: Single Relay, *RELAY ISO9002*

[9] Thông số LCD i2c module:

https://www.waveshare.com/wiki/LCD1602_I2C_Module

[10] Thông số mạch phun sương:

<https://nshopvn.com/product/mach-phun-suong-sieu-am-108khz-type-c/>

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập – Tự do - Hạnh phúc

Thành phố Hồ Chí Minh, ngày 17 tháng 11 năm 2024

BIÊN BẢN

HỌP BÁO CÁO TIẾN ĐỘ

- **Thời gian:** lúc 20h00' ngày 17 tháng 11 năm 2024
- **Địa điểm:** Họp online qua Google Meet
- **Thành phần tham dự:** 4/4 thành viên

Chức vụ	Họ tên	MSSV
Chủ trì	Trần Hoàng Hải	22119066
Thư ký	Lê Khải Hưng	22119086
Thành viên nhóm	Lê Hoàng Gia Khang	22119087
Thành viên nhóm	Nguyễn Khắc Luân	22119099

- **Người lập biên bản:** Lê Khải Hưng

* NỘI DUNG

1. BÁO CÁO TIẾN ĐỘ CỦA DỰ ÁN:

Trần Hoàng Hải	<ul style="list-style-type: none">- Đã tìm hiểu về các phần cứng để thiết kế mạch.- Đã hoàn thành được công việc tìm hiểu và kết nối linh kiện với vi điều khiển, đồng thời tìm hiểu các chuẩn giao tiếp của từng cảm biến tới vi điều khiển- Đang nghiên cứu về thiết kế giao diện của người dùng trên Blynk
-----------------------	---

Lê Khải Hưng	<ul style="list-style-type: none"> - Đã chuẩn bị các linh kiện và dụng cụ cần thiết dự kiến để thi công mạch, nếu sau này có phát sinh thêm linh kiện hay vấn đề gì thì sẽ mua bổ sung sau. - Đã tìm hiểu và thực hành thử nghiệm các công đoạn thiết kế mạch.
Lê Hoàng Gia Khang	<ul style="list-style-type: none"> - Đã tìm hiểu được các thư viện của cảm biến và các câu lệnh trong đó - Đã hoàn thành được công việc tìm hiểu và kết nối linh kiện với vi điều khiển, đồng thời tìm hiểu các chuẩn giao tiếp của từng cảm biến tới vi điều khiển - Đang tìm hiểu về cách giao tiếp LCD I2C.
Nguyễn Khắc Luân	<ul style="list-style-type: none"> - Đã hoàn thành được công việc tìm hiểu và kết nối linh kiện với vi điều khiển, đồng thời tìm hiểu các chuẩn giao tiếp của từng cảm biến tới vi điều khiển - Đã tìm hiểu được các thư viện của cảm biến và các câu lệnh trong đó - Đang tìm hiểu về cách giao tiếp LCD I2C.

2. Đóng góp ý kiến trong buổi họp

- Thông qua 1 số trao đổi ý kiến:
 - + Các thành viên đã nắm bắt được các phần cứng để thiết kế mạch.
 - + Mỗi người đang tìm hiểu về phần của mình với tiến độ ổn.
 - + Mọi người trao đổi cho nhau về các kiến thức mà mình tìm hiểu được.
 - + Quyết định ngày thi công mạch.
- Nhất trí biên bản 4/4, tỉ lệ 100%. Biên bản kết thúc lúc 22h00' cùng ngày.

THU KÝ

(ký, ghi rõ họ tên)

CHỦ TRÌ

(ký, ghi rõ họ tên)

Lê Khải Hưng

Trần Hoàng Hải

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập – Tự do - Hạnh phúc

Thành phố Hồ Chí Minh, ngày 24 tháng 9 năm 2024

BIÊN BẢN

V/v họp chọn đề tài đồ án môn học Hệ thống nhúng

- Thời gian : vào lúc 19h ngày 24 tháng 9 năm 2024

- Địa điểm : Google Meet

Diễn ra cuộc họp với mục đích: Chọn đề tài đồ án môn học Hệ thống nhúng

I. Thành phần tham dự:

- | | |
|--------------------------------|-----------------------------|
| - Chủ trì : Trần Hoàng Hải | ; Mã số sinh viên: 22119066 |
| - Thư ký : Lê Khải Hưng | ; Mã số sinh viên: 22119086 |
| - Tham dự : Lê Hoàng Gia Khang | ; Mã số sinh viên: 22119087 |
| - Tham dự : Nguyễn Khắc Luân | ; Mã số sinh viên: 22119099 |

II. Nội dung cuộc họp:

1. Đề xuất đề tài đồ án của từng thành viên

Trần Hoàng Hải	<ul style="list-style-type: none">Máy phun sương thông minh.Hệ thống định vị vật nuôi thông qua GPS.
Lê Khải Hưng	<ul style="list-style-type: none">Cửa khóa vân tay chống trộm.Hệ thống báo cháy trong chung cư.Hệ thống giám sát và cảnh báo rò rỉ khí gas.Hệ thống phân loại rác thông minh.
Lê Hoàng Gia Khang	<ul style="list-style-type: none">Hệ thống giám sát số lượng người ra vào tòa nhà thông qua cảm biến hồng ngoại.Hệ thống quản lý điện năng cho văn phòng.Hệ thống điều khiển Robot phục vụ trong nhà hàng.Hệ thống điều khiển đèn thông qua cường độ ánh sáng hiện tại.
Nguyễn Khắc Luân	<ul style="list-style-type: none">Hệ thống nhận diện giọng nói để điều khiển thiết bị.Hệ thống pha chế đồ uống tự động.Thiết bị theo dõi, định vị xe máy và báo cáo sự cố từ xa.Hệ thống kiểm soát và giám sát pin năng lượng mặt trời.

2. Biểu quyết

Sau khi trao đổi và thông qua biểu quyết, đề tài đồ án được nhóm lựa chọn là: **Máy phun sương thông minh.**

- Số phiếu tán thành: 4/4, chiếm tỉ lệ 100%
- Số phiếu không tán thành: 0/4, chiếm tỉ lệ 0%

III. Kết luận

Tất cả thành viên trong nhóm nhất trí lựa chọn đề tài **Máy phun sương thông minh** cho đồ án môn học Hệ thống nhúng sắp tới.

Cuộc họp kết thúc vào lúc 21h cùng ngày.

Biên bản có hiệu lực kể từ ngày ký.

THUẾ KÝ

(Ký, ghi rõ họ tên)

Lê Khải Hưng

CHỦ TRÌ

(Ký, ghi rõ họ tên)

Trần Hoàng Hải

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập - Tự do - Hạnh phúc

Thành phố Hồ Chí Minh, ngày 29 tháng 9 năm 2024

BIÊN BẢN

V/v họp đồ án môn học Hệ thống nhúng

- Thời gian: vào lúc 19h ngày 29 tháng 9 năm 2024

- Địa điểm : Google Meet

Diễn ra cuộc họp với mục đích: Thảo luận về yêu cầu người dùng và đặc tả sơ bộ của hệ thống.

Đề tài đồ án: Thiết kế và phát triển máy phun sương thông minh

I. Thành phần tham dự

- | | |
|--------------------------------|-----------------------------|
| - Chủ trì : Trần Hoàng Hải | ; Mã số sinh viên: 22119066 |
| - Thư ký : Lê Khải Hưng | ; Mã số sinh viên: 22119086 |
| - Tham dự : Lê Hoàng Gia Khang | ; Mã số sinh viên: 22119087 |
| - Tham dự : Nguyễn Khắc Luân | ; Mã số sinh viên: 22119099 |

II. Nội dung cuộc họp

1. Thảo luận về yêu cầu của người dùng đối với máy phun sương thông minh

- Ý kiến đóng góp của từng thành viên:

Trần Hoàng Hải	<ul style="list-style-type: none">Tiết kiệm năng lượngĐơn giản, thân thiện với người dùngThông báo người dùngThời gian thựcCó thể tích hợp với mô hình nhà thông minh
Lê Khải Hưng	<ul style="list-style-type: none">Đáp ứng thời gian tương đối, chấp nhận độ trễThay đổi theo mùaKhử khuẩn
Lê Hoàng Gia Khang	<ul style="list-style-type: none">Sản phẩm bềnHoạt động ít tiếng ồnTiết kiệm năng lượng

	<ul style="list-style-type: none"> - Dung tích: 1,5 lít
Nguyễn Khắc Luân	<ul style="list-style-type: none"> - Hiển thị thông số - Chế độ phun tự động điều chỉnh, đồng thời có thể chỉnh thủ công - Độ ẩm phù hợp với nhiệt độ môi trường. - Hẹn giờ phun sương - Dung tích: 1 lít

- **Quyết định các ý kiến cuối cùng về yêu cầu người dùng:**
 - + Tiết kiệm năng lượng
 - + Đơn giản, thân thiện với người dùng
 - + Thông báo người dùng
 - + Hiển thị thông số
 - + Đáp ứng thời gian tương đối, chấp nhận độ trễ không quá 3 phút
 - + Có thể tích hợp với mô hình nhà thông minh
 - + Chế độ phun tự động điều chỉnh, đồng thời có thể chỉnh thủ công
 - + Dung tích: 1 lít

2. Đặc tả hệ thống của máy phun sương thông minh

a. Bàn luận về thông số của hệ thống

Đặc điểm	Thông số	Người đóng góp ý kiến
Dung tích	1 lít	Nguyễn Khắc Luân
Dung tích	2,5 lít	Lê Hoàng Gia Khang
Công suất tiêu thụ hệ thống (Dự kiến)	25W-30W	Trần Hoàng Hải
Tốc độ vòi phun sương	300 ml/h	Lê Khải Hưng
Tốc độ vòi phun sương	450 ml/h	Trần Hoàng Hải
Tốc độ vòi phun sương	500ml/h	Lê Hoàng Gia Khang

b .Bàn luận về tính năng của hệ thống

Các chế độ phun sương	- Tự động - Thủ công	Trần Hoàng Hải
Hệ thống cảnh báo	Hiện trên màn hình thiết bị, app trên điện thoại, máy tính,..	Lê Khải Hưng
Đo đồng thời nhiệt độ kèm, độ ẩm	Dựa vào tham số nhiệt độ để tính ra Heat Index => Cảnh báo người dùng	Nguyễn Khắc Luân
Dựa vào chỉ số RH để quyết định	40%-60%: chấp nhận, không phun. 50%-54%: lý tưởng. <40%: khô, cần phun. >60%: cảnh báo người dùng, không phun.	Lê Hoàng Gia Khang

III. Kết luận

- Cuộc họp này đã bàn luận tương đối đầy đủ về yêu cầu, chức năng, thông số kỹ thuật trong đề tài “Thiết kế và phát triển máy phun sương thông minh” của nhóm. Từ những yêu cầu, chức năng, thông số trên có thể triển khai, mô hình hóa sản phẩm và hoạt động của thiết bị. Đồng thời sau đó có thể tính tới những thành phần, linh kiện sẽ sử dụng.

- Cuộc họp kết thúc vào lúc 21h20' cùng ngày.

- Biên bản có hiệu lực kể từ ngày ký.

THƯ KÝ

(Ký, ghi rõ họ tên)

CHỦ TRÌ

(Ký, ghi rõ họ tên)

Lê Khải Hưng

Trần Hoàng Hải