

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC KHOA HỌC  
KHOA CÔNG NGHỆ THÔNG TIN**



**PHÁT TRIỂN ỨNG DỤNG IOT  
NHÓM 5**

**2024-2025.2.TIN4024.005**

**ĐIỀU KHIỂN QUẠT DỰA TRÊN  
NHIỆT ĐỘ VỚI ESP32**

Người thực hiện	: Ngô Văn Hiếu
Ngành	: Công nghệ thông tin
Khóa học	: 2021-2025
Giảng viên hướng dẫn	: Võ Việt Dũng

*Huế, tháng 4/2025*

# LỜI CẢM ƠN

Đầu tiên, em xin gửi lời cảm ơn chân thành đến Khoa Công nghệ Thông tin - Trường Đại học Khoa học – Đại học Huế, nơi đã mang đến cho em một môi trường học tập đầy cảm hứng và những điều kiện thuận lợi để em có thể thực hiện và hoàn thiện bài tiểu luận này. Nhờ sự hỗ trợ từ nhà trường, em đã có cơ hội trau dồi kiến thức chuyên sâu cũng như phát triển các kỹ năng quan trọng, làm nền tảng vững chắc cho hành trình học tập và sự nghiệp sau này.

Em xin bày tỏ lòng biết ơn đặc biệt đến thầy Võ Việt Dũng, người đã tận tâm hướng dẫn em trong suốt quá trình thực hiện đề tài. Với sự nhiệt tình, những góp ý quý giá và sự động viên không ngừng, thầy đã giúp em vượt qua những thử thách, định hướng rõ ràng và hoàn thành bài tiểu luận một cách trọn vẹn nhất. Em thực sự trân trọng sự đồng hành kiên nhẫn của thầy, điều đã để lại trong em những bài học sâu sắc không chỉ về kiến thức mà còn về tinh thần trách nhiệm.

Em cũng xin gửi lời tri ân đến tất cả quý thầy cô trong Khoa Công nghệ Thông tin - Trường Đại học Khoa học – Đại học Huế, những người đã luôn tận tụy giảng dạy, chia sẻ kinh nghiệm và hỗ trợ em trong suốt thời gian học tập. Sự nhiệt huyết và tâm huyết của các thầy cô không chỉ giúp em nâng cao năng lực học thuật mà còn truyền cảm hứng để em tự tin bước tiếp trên con đường mình đã chọn.

Cuối cùng, em xin gửi lời cảm ơn sâu sắc đến bạn bè, gia đình và những người thân yêu, những người đã luôn ở bên, tiếp thêm động lực và niềm tin để em hoàn thành tốt công việc của mình. Sự ủng hộ thầm lặng nhưng mạnh mẽ từ mọi người là nguồn sức mạnh lớn lao giúp em không ngừng nỗ lực và đạt được kết quả như hôm nay.

Em xin chân thành cảm ơn tất cả!

# MỤC LỤC

<b>MỞ ĐẦU .....</b>	<b>1</b>
1. Lý do chọn đề tài .....	1
2. Mục tiêu nghiên cứu.....	1
3. Phạm vi nghiên cứu.....	2
4. Phương pháp nghiên cứu.....	2
<b>NỘI DUNG .....</b>	<b>3</b>
<b>CHƯƠNG 1: TỔNG QUAN ĐIỀU KHIỂN QUẠT DỰA TRÊN NHIỆT ĐỘ ....</b>	<b>3</b>
1. Điều khiển quạt dựa trên nhiệt độ là gì? .....	3
2. Các phương pháp điều khiển quạt truyền thống và hiện đại .....	3
3. Lợi ích của việc ứng dụng IoT trong điều khiển quạt.....	3
<b>CHƯƠNG 2: GIỚI THIỆU CÔNG NGHỆ VÀ LINH KIỆN.....</b>	<b>4</b>
1. Wokwi – Công cụ mô phỏng phần cứng trực tuyến .....	4
2. ESP32 là gì? .....	4
3. Cảm biến nhiệt độ DHT22 .....	5
4. Module Relay.....	5
5. Nền tảng Blynk – Giám sát và điều khiển từ xa .....	6
<b>CHƯƠNG 3: THIẾT KẾ HỆ THỐNG ĐIỀU KHIỂN QUẠT DỰA TRÊN NHIỆT ĐỘ .....</b>	<b>6</b>
1. Xây dựng sơ đồ kết nối trên Wokwi.....	6
2. Thiết kế hệ thống điều khiển quạt tự động.....	10
<b>TỔNG KẾT .....</b>	<b>12</b>
1. Kết luận .....	12
2. Nhận xét .....	12
3. Đánh giá .....	13
4. Kiến nghị .....	13
<b>TÀI LIỆU THAM KHẢO.....</b>	<b>14</b>

# MỞ ĐẦU

## 1. Lý do chọn đề tài

Trong bối cảnh công nghệ phát triển, việc tự động hóa các thiết bị gia dụng như quạt điện không chỉ mang lại sự tiện lợi mà còn giúp tiết kiệm năng lượng và cải thiện chất lượng cuộc sống. Nhiệt độ môi trường là yếu tố quan trọng ảnh hưởng đến sự thoải mái của con người, nhưng các phương pháp điều khiển quạt truyền thống như bật/tắt thủ công thường không hiệu quả.

Công nghệ IoT (Internet of Things) đã mở ra cơ hội xây dựng các hệ thống thông minh, trong đó vi điều khiển ESP32 nổi bật nhờ khả năng kết nối mạnh mẽ và chi phí thấp. Đề tài “Điều khiển quạt dựa trên nhiệt độ với ESP32” được thực hiện nhằm ứng dụng ESP32 kết hợp cảm biến nhiệt độ DHT22 và module relay để tự động bật/tắt quạt theo ngưỡng nhiệt độ cài đặt, đồng thời tích hợp khả năng giám sát từ xa qua nền tảng Blynk.

Nghiên cứu này không chỉ mang tính ứng dụng thực tiễn cao, giúp giải quyết nhu cầu điều chỉnh nhiệt độ một cách thông minh trong đời sống hàng ngày, mà còn là cơ hội để em khám phá sâu hơn về công nghệ IoT – một lĩnh vực đang ngày càng khẳng định vai trò quan trọng trong cách mạng công nghiệp 4.0. Thông qua việc thực hiện đề tài, em mong muốn không chỉ nắm vững kiến thức lý thuyết mà còn rèn luyện kỹ năng thực hành, từ lập trình, thiết kế mạch đến tích hợp các nền tảng công nghệ hiện đại, qua đó đóng góp một phần nhỏ vào việc thúc đẩy ứng dụng IoT trong các giải pháp cải thiện cuộc sống.

## 2. Mục tiêu nghiên cứu

- Xây dựng mô hình lý thuyết cho hệ thống điều khiển quạt tự động dựa trên nhiệt độ.
- Phân tích cách ESP32 thu thập dữ liệu từ cảm biến DHT22 và điều khiển quạt qua module relay.
- Thiết lập ngưỡng nhiệt độ để bật quạt ( $>30^{\circ}\text{C}$ ) và tắt quạt ( $<25^{\circ}\text{C}$ ).
- Tích hợp nền tảng Blynk để giám sát nhiệt độ và trạng thái quạt từ xa.
- Minh họa nguyên lý hoạt động của hệ thống thông qua sơ đồ kết nối trên Wokwi.

### 3. Phạm vi nghiên cứu

- Nghiên cứu lý thuyết và mô phỏng hệ thống trên nền tảng Wokwi.
- Ứng dụng trong môi trường nhỏ như phòng ở, văn phòng hoặc không gian kín.
- Sử dụng ESP32 làm trung tâm xử lý, kết hợp cảm biến DHT22 và module relay để điều khiển quạt.
- Tích hợp Blynk để hiển thị dữ liệu nhiệt độ và trạng thái quạt trên thiết bị di động hoặc máy tính.

### 4. Phương pháp nghiên cứu

- **Nghiên cứu tài liệu:** Tìm hiểu về ESP32, DHT22, Module Relay và Blynk.
- **Phân tích thiết kế:** Xây dựng mô hình lý thuyết về kết nối giữa các linh kiện.
- **Mô phỏng trên Wokwi:** Thiết kế sơ đồ mạch để kiểm tra nguyên lý hoạt động.
- **Thiết kế giao diện Blynk:** Tạo giao diện hiển thị nhiệt độ và trạng thái quạt.
- **Lập trình ESP32:** Viết mã điều khiển quạt dựa trên dữ liệu nhiệt độ từ DHT22.

# NỘI DUNG

## CHƯƠNG 1: TỔNG QUAN ĐIỀU KHIỂN QUẠT DỰA TRÊN NHIỆT ĐỘ

### 1. Điều khiển quạt dựa trên nhiệt độ là gì?

Điều khiển quạt dựa trên nhiệt độ là giải pháp tự động hóa sử dụng cảm biến để đo nhiệt độ môi trường, từ đó điều khiển quạt bật hoặc tắt theo các ngưỡng cài đặt sẵn. Nó giúp duy trì nhiệt độ ổn định, tiết kiệm năng lượng và giảm sự can thiệp của con người, phù hợp cho các không gian như nhà ở, văn phòng hoặc nhà xưởng nhỏ.

### 2. Các phương pháp điều khiển quạt truyền thống và hiện đại

#### 2.1. Phương pháp truyền thống

- **Điều khiển thủ công:** Người dùng bật/tắt quạt bằng công tắc hoặc điều chỉnh tốc độ bằng núm xoay.
- **Nhược điểm:** Tốn thời gian, không tối ưu năng lượng, không phản ứng kịp thời với thay đổi nhiệt độ.

#### 2.2. Phương pháp hiện đại

- **Sử dụng cảm biến và IoT:** Cảm biến đo nhiệt độ liên tục, vi điều khiển xử lý dữ liệu và điều khiển quạt qua relay.
- **Ưu điểm:** Tự động hóa hoàn toàn, giám sát từ xa qua mạng Wi-Fi, tiết kiệm năng lượng và tăng hiệu quả sử dụng.

### 3. Lợi ích của việc ứng dụng IoT trong điều khiển quạt

- **Tự động hóa:** Quạt hoạt động chỉ khi cần thiết, giảm lãng phí điện.
- **Giám sát từ xa:** Người dùng có thể theo dõi nhiệt độ và trạng thái quạt qua điện thoại.
- **Tiết kiệm năng lượng:** Tối ưu thời gian hoạt động của quạt dựa trên nhiệt độ thực tế.
- **Tăng độ bền thiết bị:** Giảm số lần bật/tắt thủ công, kéo dài tuổi thọ quạt.
- **Khả năng mở rộng:** Dễ dàng tích hợp thêm các chức năng như điều khiển tốc độ hoặc kết hợp với thiết bị khác.

# CHƯƠNG 2: GIỚI THIỆU CÔNG NGHỆ VÀ LINH KIỆN

## 1. Wokwi – Công cụ mô phỏng phần cứng trực tuyến

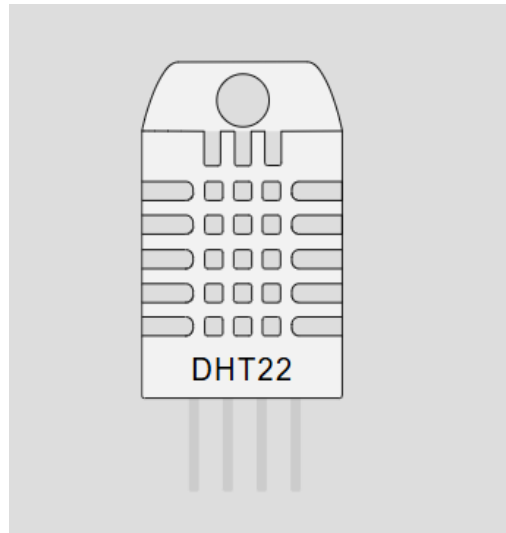
Wokwi là một ứng dụng trực tuyến giúp mô phỏng các dự án điện tử sử dụng vi điều khiển như Arduino, ESP32, Raspberry Pi Pico và nhiều linh kiện điện tử khác. Ứng dụng này cho phép người dùng thực hành viết mã và mô phỏng mạch điện mà không cần phần cứng thật, rất tiện lợi cho việc học tập và phát triển dự án.[1]

## 2. ESP32 là gì?



- ESP32 là một hệ thống vi điều khiển trên chip (SoC) giá rẻ của Espressif Systems, nhà phát triển của ESP8266 SoC. Nó là sự kế thừa của SoC ESP8266 và có cả hai biến thể lõi đơn và lõi kép của bộ vi xử lý 32-bit Xtensa LX6 của Tensilica với Wi-Fi và Bluetooth tích hợp.[2]
- Điểm tốt về ESP32, giống như ESP8266 là các thành phần RF tích hợp của nó như bộ khuếch đại công suất, bộ khuếch đại nhận tiếng ồn thấp, công tắc ăng-ten, bộ lọc và Balun RF. Điều này làm cho việc thiết kế phần cứng xung quanh ESP32 rất dễ dàng vì bạn cần rất ít thành phần bên ngoài.[2]
- Một điều quan trọng khác cần biết về ESP32 là nó được sản xuất bằng công nghệ 40 nm công suất cực thấp của TSMC. Vì vậy, việc thiết kế các ứng dụng hoạt động bằng pin như thiết bị đeo, thiết bị âm thanh, đồng hồ thông minh, ..., sử dụng ESP32 sẽ rất dễ dàng.[2]

### 3. Cảm biến nhiệt độ DHT22



DHT22 là cảm biến có thể đo nhiệt độ từ  $-40^{\circ}\text{C}$  đến  $80^{\circ}\text{C}$  và độ ẩm từ 0% đến 100% với độ chính xác  $\pm 1^{\circ}\text{C}$  và  $\pm 1\%$

#### Thông số kỹ thuật DHT22

- Điện áp hoạt động: 3.5V đến 5.5V
- Dòng hoạt động: 0,3mA (đo) 60uA (chế độ chờ)
- Đầu ra: Dữ liệu nối tiếp
- Phạm vi nhiệt độ:  $-40^{\circ}\text{C}$  đến  $80^{\circ}\text{C}$
- Phạm vi độ ẩm: 0% đến 100%
- Độ phân giải: Nhiệt độ và Độ ẩm đều là 16-bit
- Độ chính xác:  $\pm 0,5^{\circ}\text{C}$  và  $\pm 1\%$  [3]

### 4. Module Relay





- Module Relay 5V 1 Kênh được dùng như một công tắc điện , dùng để điều khiển các thiết bị công suất lớn ( đèn, động cơ, ...)[4]

- Module Relay 5V 1 Kênh gồm 1 rơ le hoạt động tại điện áp 5VDC, 12VDC chịu được hiệu điện thế lên đến 250VAC 10A. Module relay 1 kênh được thiết kế chắc chắn, khả năng cách điện tốt.[4]

### THÔNG SỐ KỸ THUẬT:

- Điện áp hoạt động: 5V
- Dòng kích Relay: 5mA
- Kích thước: 43mm x 17.3mm x 17mm (dài x rộng x cao)
- Trọng lượng: 15g [4]

## 5. Nền tảng Blynk – Giám sát và điều khiển từ xa

Blynk IoT là một nền tảng đơn giản và mạnh mẽ, cho phép bạn kết nối và điều khiển các thiết bị IoT từ xa qua internet. Kết hợp với ESP32, một vi điều khiển tích hợp Wi-Fi, Blynk mang đến giải pháp hoàn hảo cho việc giám sát và điều khiển các thiết bị điện tử dễ dàng từ điện thoại di động hoặc máy tính.[5]

## CHƯƠNG 3: THIẾT KẾ HỆ THỐNG ĐIỀU KHIỂN QUẠT DỰA TRÊN NHIỆT ĐỘ

### 1. Xây dựng sơ đồ kết nối trên Wokwi

#### 1.1. Thành phần hệ thống

Hệ thống bao gồm:

- **ESP32 (wokwi-esp32-devkit-v1):** Trung tâm xử lý, điều khiển relay và kết nối Wi-Fi với Blynk.
- **DHT22:** Cảm biến đo nhiệt độ, kết nối với GPIO 4 (D4), giá trị nhiệt độ giả lập trong Wokwi là 48.6°C (theo JSON).
- **Module Relay (wokwi-relay-module):** Điều khiển quạt (mô phỏng bằng động cơ DC trong Wokwi), kết nối với GPIO 5 (D5).
- **LED:** Đại diện trạng thái quạt, kết nối với GPIO 2 (D2) qua điện trở 220Ω.
- **Động cơ DC (wokwi-dc-motor):** Mô phỏng quạt, kết nối qua relay.

- **Điện trở:**
  - 10k $\Omega$ : Kéo lên cho tín hiệu DATA của DHT22.
  - 220 $\Omega$ : Giới hạn dòng cho LED.
- **Nguồn điện:**
  - 3.3V từ ESP32 cấp cho DHT22.
  - 5V từ ESP32 cấp cho relay.

## 1.2. Sơ đồ kết nối

Sơ đồ mạch được xây dựng trên Wokwi dựa trên cấu hình từ code JSON:

### - DHT22:

- VCC  $\rightarrow$  3.3V (ESP32).
- GND  $\rightarrow$  GND (ESP32).
- SDA (DATA)  $\rightarrow$  GPIO 4 (ESP32), có điện trở kéo lên 10k $\Omega$ .

### - Module Relay:

- VCC  $\rightarrow$  5V (ESP32).
- GND  $\rightarrow$  GND (ESP32).
- IN  $\rightarrow$  GPIO 5 (ESP32).
- COM  $\rightarrow$  5V (ESP32).
- NO  $\rightarrow$  Động cơ DC (+).

### - Động cơ DC:

- (+)  $\rightarrow$  NO (relay).
- (-)  $\rightarrow$  GND (ESP32).

### - LED:

- Anode  $\rightarrow$  GPIO 2 (ESP32) qua điện trở 220 $\Omega$ .
- Cathode  $\rightarrow$  GND (ESP32).

### - ESP32:

- TX0 và RX0 kết nối với Serial Monitor để hiển thị thông tin debug.
- Kết nối Wi-Fi (Wokwi-GUEST) để gửi dữ liệu lên Blynk.

## 1.3. Nguyên lý hoạt động

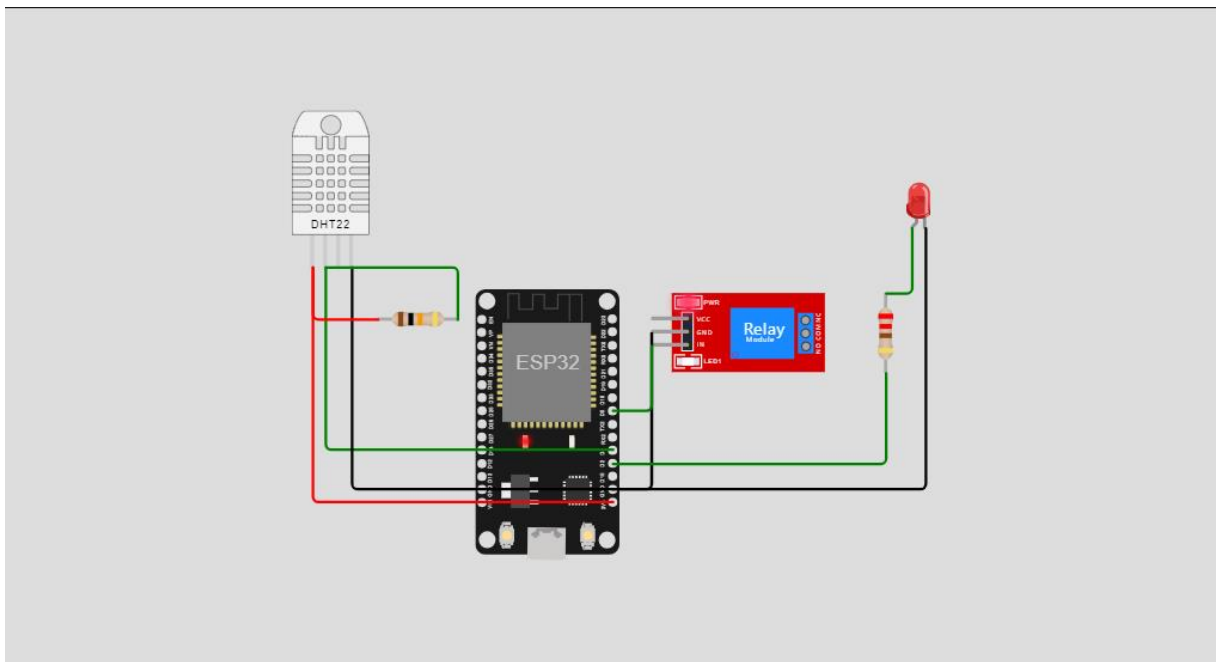
- **Đo nhiệt độ:** DHT22 đo nhiệt độ môi trường và gửi dữ liệu đến ESP32 qua GPIO 4 mỗi 2 giây.
- **Xử lý dữ liệu:** ESP32 so sánh nhiệt độ với ngưỡng:

- Nếu nhiệt độ  $> 30^{\circ}\text{C}$ : ESP32 gửi tín hiệu HIGH đến GPIO 5, kích hoạt relay để bật quạt (động cơ DC quay, LED sáng).
- Nếu nhiệt độ  $\leq 30^{\circ}\text{C}$ : ESP32 gửi tín hiệu LOW, ngắt relay để tắt quạt (động cơ DC dừng, LED tắt).

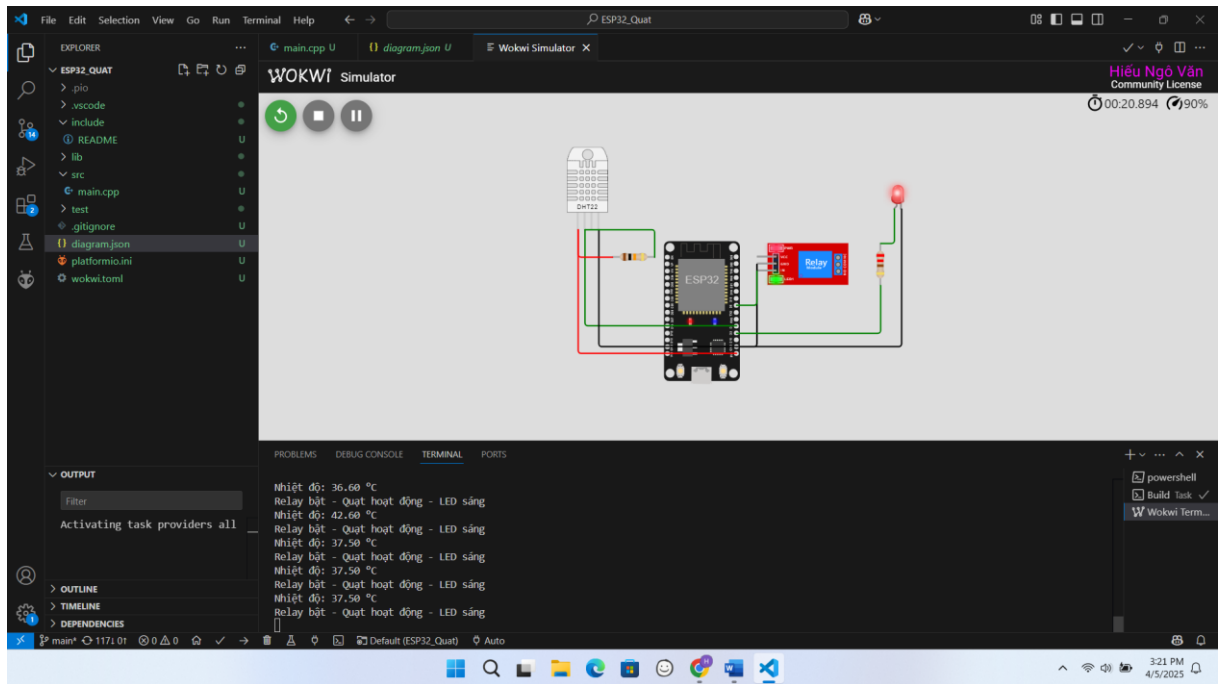
**- Giám sát:**

- Nhiệt độ được gửi lên Blynk (V1) mỗi 2 giây.
- Thời gian hoạt động (uptime) được gửi lên Blynk (V0) mỗi giây.

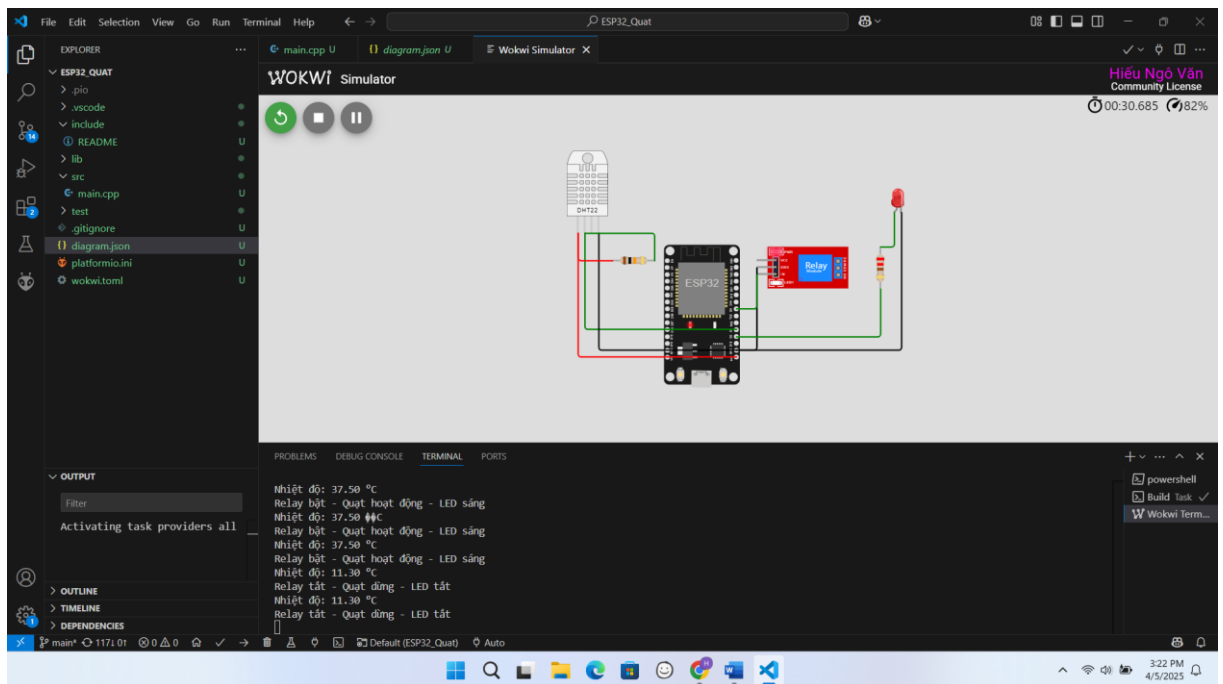
#### 1.4. Giao diện điều khiển quạt dựa trên nhiệt độ với ESP32 trên Wokwi



Hình ảnh sơ đồ trên wokwi.



Hình ảnh khi nhiệt độ  $> 30^{\circ}\text{C}$ : Relay bật, LED sáng.



Hình ảnh khi nhiệt độ  $\leq 30^{\circ}\text{C}$ : Relay tắt, LED tắt.

## 2. Thiết kế hệ thống điều khiển quạt tự động

### 2.1. Giao diện trên Blynk

Giao diện Blynk được thiết kế để giám sát hệ thống:

- **Gauge Widget (V1):** Hiển thị giá trị nhiệt độ từ DHT22, đơn vị °C.
- **Value Display Widget (V0):** Hiển thị thời gian hoạt động của hệ thống (tính bằng giây).

### 2.2. Cấu hình

a) **Tạo dự án trên Blynk:**

- Chọn loại vi điều khiển ESP32.

b) **Thêm widget và gán chân ảo:**

- V0: Hiển thị thời gian chạy (uptime).
- V1: Gauge để hiển thị nhiệt độ.

c) **Kết nối với ứng dụng Blynk bằng mã xác thực (Auth Token).**

d) **Lập trình ESP32:**

- Sử dụng thư viện DHT.h để đọc nhiệt độ từ DHT22 qua GPIO 4.
- Điều khiển relay qua GPIO 5 dựa trên ngưỡng nhiệt độ ( $> 30^{\circ}\text{C}$  bật,  $\leq 30^{\circ}\text{C}$  tắt).
- Gửi dữ liệu lên Blynk:
  - Nhiệt độ (V1) mỗi 2 giây.
  - Thời gian chạy (V0) mỗi giây.
- Kết nối Wi-Fi với SSID Wokwi-GUEST và mật khẩu trống.

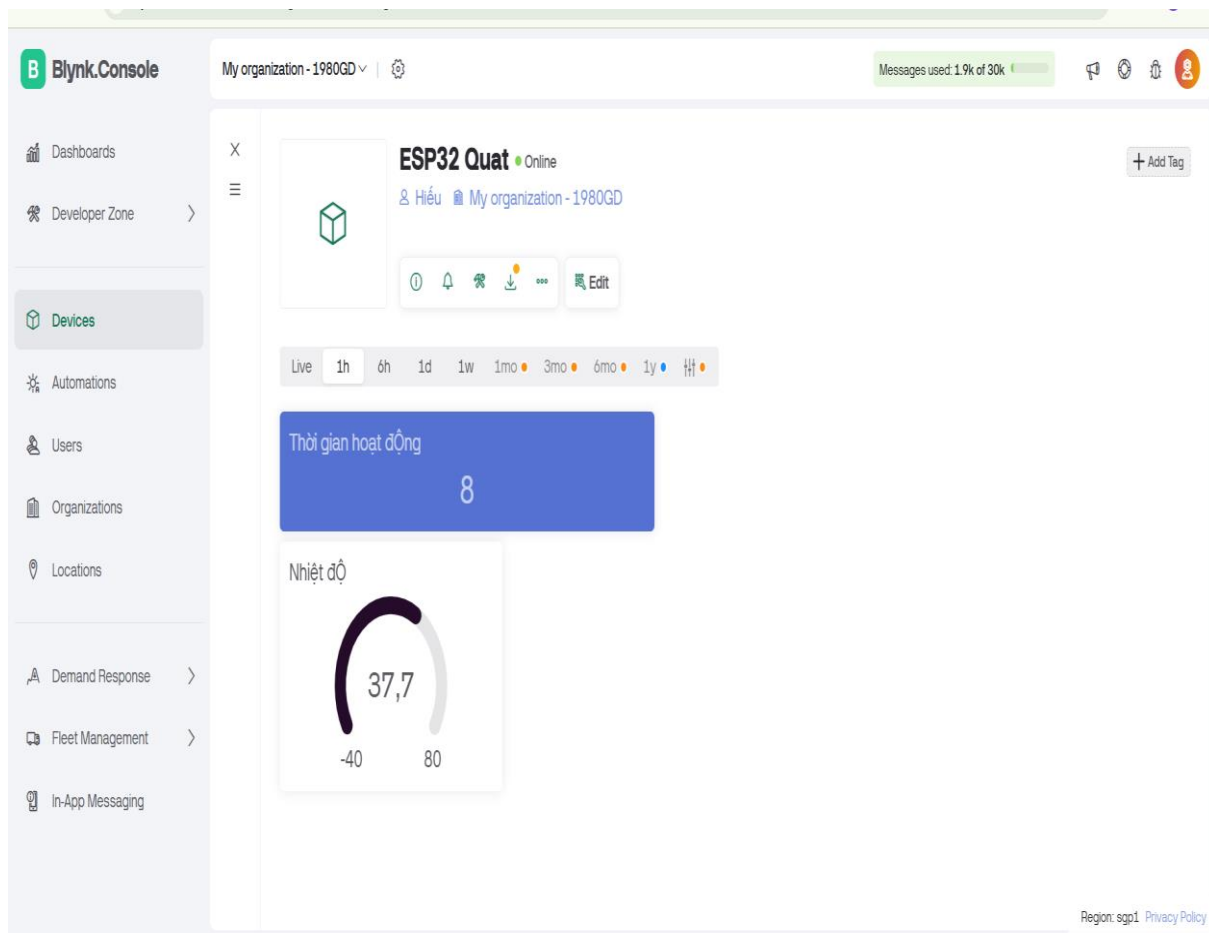
### 2.3. Kiểu dữ liệu

- **Nhiệt độ:** Double (giá trị thập phân từ DHT22, ví dụ  $28.3^{\circ}\text{C}$ ).
- **Trạng thái quạt (relayState):** Integer (0: tắt, 1: bật, tương ứng với LOW/HIGH).
- **Thời gian chạy (uptime):** Unsigned long (tính bằng giây từ khi khởi động).

### 2.4. Giới hạn nhiệt độ

- **Nhiệt độ hoạt động:** từ  $-40^{\circ}\text{C}$  đến  $80^{\circ}\text{C}$ .

## 2.5. Giao diện giám sát điều khiển quạt dựa trên nhiệt độ với ESP32 trên Blynk



# TỔNG KẾT

## 1. Kết luận

Hệ thống điều khiển quạt dựa trên nhiệt độ sử dụng ESP32 đã hoàn thành tốt các mục tiêu đề ra trong quá trình nghiên cứu và thiết kế. Thông qua việc tích hợp cảm biến DHT22 để đo nhiệt độ môi trường, vi điều khiển ESP32 xử lý dữ liệu và điều khiển quạt qua Module Relay theo ngưỡng nhiệt độ cài đặt (bật khi  $> 30^{\circ}\text{C}$ , tắt khi  $\leq 30^{\circ}\text{C}$ ), hệ thống đã chứng minh khả năng tự động hóa hiệu quả. Đồng thời, việc kết nối với nền tảng Blynk qua Wi-Fi cho phép người dùng giám sát nhiệt độ và trạng thái quạt từ xa, cũng như điều khiển thủ công khi cần thiết thông qua ứng dụng di động.

## 2. Nhận xét

### - Ưu điểm:

- Tự động hóa cao: Hệ thống tự động bật/tắt quạt dựa trên nhiệt độ mà không cần sự can thiệp thủ công liên tục, giúp tiết kiệm thời gian và công sức.
- Giao diện trực quan: Ứng dụng Blynk cung cấp giao diện thân thiện, dễ sử dụng, cho phép người dùng theo dõi nhiệt độ, trạng thái quạt và thời gian hoạt động một cách rõ ràng, ngay cả khi không có kiến thức kỹ thuật sâu.
- Dễ mở rộng: Thiết kế linh hoạt của hệ thống cho phép tích hợp thêm các tính năng như điều khiển từ xa qua Telegram, thêm cảm biến khác (độ ẩm, ánh sáng), hoặc kết nối với các thiết bị thông minh khác trong hệ sinh thái IoT.
- Tính thực tiễn: Hệ thống có thể áp dụng trong nhiều bối cảnh như nhà ở, văn phòng, hoặc nhà xưởng nhỏ, nơi cần duy trì nhiệt độ ổn định mà không gây lãng phí điện năng.

### - Nhược điểm:

- Phụ thuộc vào kết nối Wi-Fi: Hệ thống yêu cầu kết nối Internet ổn định để gửi dữ liệu lên Blynk và nhận lệnh điều khiển từ xa. Nếu Wi-Fi bị gián đoạn, chức năng giám sát và điều khiển từ xa sẽ không hoạt động, dù chế độ tự động cục bộ vẫn có thể duy trì.
- Độ chính xác của cảm biến: Cảm biến DHT22 có thể bị ảnh hưởng bởi các yếu tố môi trường như độ ẩm cao, bụi bẩn hoặc vị trí lắp đặt không tối ưu, dẫn đến sai số trong dữ liệu nhiệt độ và ảnh hưởng đến quyết định bật/tắt quạt.
- Hạn chế mô phỏng: Trong Wokwi, việc thiếu mô hình quạt thực tế buộc phải thay thế bằng relay và kiểm tra logic gián tiếp, làm giảm khả năng đánh giá toàn

diện hiệu suất thực tế của hệ thống.

### 3. Đánh giá

Hệ thống phù hợp cho các ứng dụng gia đình hoặc văn phòng nhỏ, với tiềm năng mở rộng như điều chỉnh tốc độ quạt hoặc tích hợp thêm cảm biến khác.

### 4. Kiến nghị

Để nâng cao hiệu quả và tính ứng dụng của hệ thống, một số kiến nghị được đề xuất như sau:

- **Cải thiện độ ổn định kết nối Wi-Fi:** Nên bổ sung cơ chế lưu trữ dữ liệu cục bộ trên ESP32 khi mất kết nối Internet, đồng thời tự động đồng bộ lại với Blynk khi Wi-Fi được khôi phục. Điều này đảm bảo hệ thống không bị gián đoạn hoàn toàn trong các tình huống thực tế.
- **Tăng cường bảo mật dữ liệu truyền qua Blynk:** Đề xuất áp dụng mã hóa dữ liệu (encryption) cho các gói tin gửi từ ESP32 đến Blynk, đồng thời sử dụng cơ chế xác thực nâng cao để ngăn chặn truy cập trái phép, đặc biệt khi hệ thống được triển khai trong môi trường công cộng hoặc quy mô lớn.
- **Thử nghiệm thực tế và tối ưu hóa:** Nên triển khai hệ thống với quạt 220V thực tế và thử nghiệm trong các điều kiện môi trường khác nhau (nhiệt độ cao, độ ẩm lớn, không gian kín/mở) để đánh giá hiệu suất, từ đó điều chỉnh ngưỡng nhiệt độ và cải thiện độ bền của linh kiện.
- **Tích hợp thêm tính năng thông báo:** Có thể kết hợp với Telegram để gửi cảnh báo khi nhiệt độ vượt quá ngưỡng nguy hiểm (ví dụ:  $>35^{\circ}\text{C}$ ), giúp người dùng phản ứng kịp thời trong các tình huống bất thường.



# TÀI LIỆU THAM KHẢO

- [1] <https://www.scribd.com/document/797929559/H%C6%B0%E1%BB%9Bng-d%E1%BA%ABn-s%E1%BB%AD-d%E1%BB%A5ng-Wokwi>
- [2] <https://dientutuonglai.com/esp32-la-gi.html>
- [3] <https://mecsuvn/ho-tro-ky-thuat/dht22-cam-bien-nhiet-do-va-do-am.vad>
- [4] <https://www.dosangtao.vn/module-relay-5v-1-kenh>
- [5] <https://dienthongminhesmart.com/lap-trinh-esp32/blynk-iot-va-esp32/>