

TRƯỜNG ĐẠI HỌC KHOA HỌC HUẾ
KHOA CÔNG NGHỆ THÔNG TIN



TIỂU LUẬN

MÔN: PHÁT TRIỂN ỨNG DỤNG IOT

NHÓM: 4

Đề tài: Tạo máy chủ Web cục bộ với ESP32 để quản lý thiết bị IOT

Giảng viên hướng dẫn: Võ Việt Dũng

Huế, 04/2025

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

(Phần này sẽ được giảng viên hướng dẫn bổ sung sau khi đánh giá bài tiểu luận)

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

LỜI CẢM ƠN

Em xin gửi lời tri ân sâu sắc đến quý thầy cô trong khoa Công nghệ Thông tin, Trường Đại học Khoa học - Đại học Huế, đặc biệt là thầy Võ Việt Dũng, người đã tận tình hướng dẫn, chia sẻ kiến thức chuyên môn quý báu và tạo mọi điều kiện thuận lợi để em hoàn thiện bài tiểu luận này.

Em cũng xin bày tỏ lòng biết ơn chân thành đến các bạn sinh viên cùng khóa và các anh chị đi trước, những người đã nhiệt tình hỗ trợ, chia sẻ tài liệu và kinh nghiệm thực tiễn, góp phần quan trọng vào quá trình nghiên cứu của em.

Dù đã nỗ lực hoàn thiện bài viết với sự nghiêm túc và tâm huyết, em nhận thức rằng, do hạn chế về thời gian và kinh nghiệm, tiểu luận khó tránh khỏi những thiếu sót.

Em rất mong nhận được những ý kiến đóng góp và nhận xét quý giá từ quý thầy cô cùng các bạn để bài nghiên cứu được hoàn chỉnh và chất lượng hơn.

Một lần nữa, em xin trân trọng cảm ơn!

MỤC LỤC

DANH MỤC HÌNH ẢNH.....	5
DANH MỤC CÁC TỪ VIẾT TẮT.....	6
MỞ ĐẦU	7
1. Giới thiệu.....	7
2. Mục tiêu và phạm vi nghiên cứu	7
3. Nội dung và kế hoạch thực hiện	8
4. Bố cục	8
CHƯƠNG 1: CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ LIÊN QUAN	9
1.1. Tổng quan về ESP32 và các tính năng chính:	9
1.2. Giới thiệu về IoT và vai trò của giao diện quản lý	10
1.3. Tổng quan về nền tảng Blynk:.....	10
1.4. Giới thiệu về Wokwi và các tính năng mô phỏng ESP32	11
CHƯƠNG 2: XÂY DỰNG HỆ THỐNG QUẢN LÝ IOT	12
2.1. Thiết kế sơ đồ trên Wokwi	12
2.2. Lập trình hệ thống trên VSCode và PlatformIO	13
2.3. Tạo giao diện điều khiển trên Blynk	13
CHƯƠNG 3: THỬ NGHIỆM VÀ ĐÁNH GIÁ	15
3.1. Kết quả chạy thử nghiệm:.....	15
3.3. Thách thức trong quá trình mô phỏng và triển khai thực tế	17
3.2. Ưu nhược điểm của dự án:	18
CHƯƠNG 4: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	19
4.1. Kết luận	19
4.2. Hướng phát triển.....	19
PHỤ LỤC	20
TÀI LIỆU THAM KHẢO	21

DANH MỤC HÌNH ẢNH

Hình 1 Board ESP32 (ESP32-WROOM-32UE).....	9
Hình 2 Sơ đồ Wokwi.....	12
Hình 3 Thông tin từ Blynk cá nhân.....	13
Hình 4 Giao diện Blynk	14
Hình 5 Kết quả chạy thử nghiệm	15

DANH MỤC CÁC TỪ VIẾT TẮT

VIẾT TẮT	NỘI DUNG
IoT	Internet of Things
ESP32	Espressif Systems 32-bit microcontroller
API	Application Programming Interface
GUI	Graphical User Interface
VSCODE	Visual Studio Code
TM1637	4-digit 7-segment LED module
MCU	Microcontroller Unit

MỞ ĐẦU

1. Giới thiệu

Trong thời đại công nghệ 4.0, bối cảnh mà Internet vạn vật (IoT) ngày càng phát triển, nhu cầu quản lý và điều khiển các thiết bị thông minh từ xa trở nên cấp thiết. Việc xây dựng các hệ thống quản lý IoT cục bộ mang lại nhiều lợi ích về độ trễ, tính ổn định và bảo mật. Đề tài này tập trung vào việc mô phỏng một hệ thống quản lý thiết bị IoT cục bộ sử dụng vi điều khiển ESP32 và nền tảng Blynk, một nền tảng phát triển ứng dụng IoT trực quan. Việc sử dụng công cụ mô phỏng trực tuyến Wokwi cho phép chúng tôi thử nghiệm và đánh giá hệ thống một cách hiệu quả mà không cần đến phần cứng thực tế.

2. Mục tiêu và phạm vi nghiên cứu

- Mục tiêu chính của đề tài là mô phỏng thành công hệ thống quản lý và điều khiển thiết bị IoT cục bộ bằng ESP32 và Blynk trên nền tảng Wokwi. Để đạt được mục tiêu chính này, đề tài tập trung vào các mục tiêu cụ thể sau:
 - + Nghiên cứu về nền tảng Blynk và cách tích hợp với ESP32.
 - + Tìm hiểu về các thành phần và tính năng mô phỏng của Wokwi.
 - + Thiết kế hệ thống mô phỏng với các thiết bị IoT ảo trên Wokwi.
 - + Lập trình ESP32 ảo trên Wokwi để tương tác với Blynk và các thiết bị ảo.
 - + Thử nghiệm và đánh giá khả năng quản lý thiết bị trong môi trường mô phỏng.
- Phạm vi nghiên cứu của đề tài giới hạn trong việc mô phỏng một số lượng giới hạn các thiết bị IoT cơ bản trên Wokwi (cụ thể là một cảm biến nhiệt độ/độ ẩm và một đèn LED) và quản lý chúng thông qua giao diện Blynk. Đề tài tập trung vào các chức năng cơ bản như hiển thị dữ liệu cảm biến và điều khiển actuator.

3. Nội dung và kế hoạch thực hiện

- **Nghiên cứu và tìm hiểu công nghệ:** Tìm hiểu chi tiết về nền tảng Blynk (kiến trúc, giao thức, thư viện), vi điều khiển ESP32 (khả năng kết nối mạng, tương tác phần cứng), và công cụ mô phỏng Wokwi (các thành phần, cách thức hoạt động).
- **Thiết lập môi trường mô phỏng:** Tạo tài khoản Blynk và dự án mới. Làm quen với giao diện Wokwi và tạo một dự án ESP32. Thêm các thành phần mô phỏng cần thiết (ESP32, cảm biến ảo, actuator ảo).
- **Thiết kế giao diện Blynk:** Xây dựng giao diện người dùng trên ứng dụng Blynk bằng cách sử dụng các widget (nút, thanh trượt, hiển thị giá trị). Thiết kế giao diện trực quan để quản lý và theo dõi các thiết bị IoT ảo
- **Tích hợp hệ thống điều khiển:** Viết code Arduino cho ESP32 ảo trên Wokwi để kết nối với mạng ảo của Wokwi và server Blynk. Lập trình để đọc dữ liệu từ cảm biến ảo và gửi lên Blynk thông qua Virtual Pins. Lập trình để nhận lệnh điều khiển từ các widget trên Blynk (thông qua Virtual Pins) và điều khiển actuator ảo
- **Thử nghiệm và đánh giá:** Chạy mô phỏng trên Wokwi và tương tác với giao diện Blynk để kiểm tra các chức năng (hiển thị dữ liệu, điều khiển thiết bị)

4. Bố cục

Chương 1: Cơ sở lý thuyết và công nghệ liên quan

Chương 2: Xây dựng hệ thống quản lý IoT

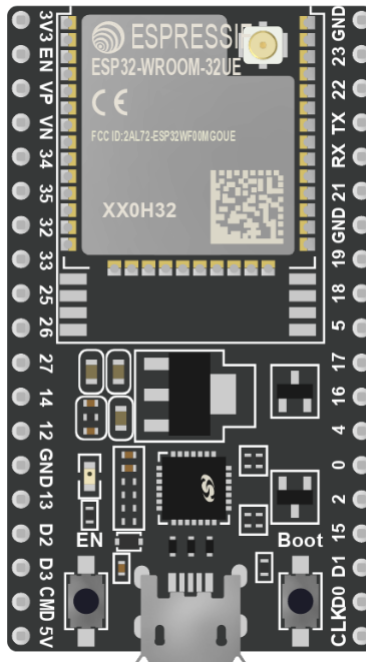
Chương 3: Thử nghiệm và đánh giá hệ thống mô phỏng

Chương 4: Kết luận và hướng phát triển.

CHƯƠNG 1: CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ LIÊN QUAN

1.1. Tổng quan về ESP32 và các tính năng chính:

- ESP32 là một vi điều khiển SoC (System on Chip) do Espressif Systems sản xuất, nổi bật với khả năng tích hợp Wi-Fi và Bluetooth. Điều này tạo điều kiện thuận lợi cho việc phát triển các ứng dụng IoT, nơi kết nối không dây là yếu tố then chốt.



Hình 1 Board ESP32 (ESP32-WROOM-32UE)

- Các tính năng chính của ESP32 bao gồm:
 - + Bộ vi xử lý lõi kép Xtensa LX6, cho phép xử lý đa nhiệm và hiệu suất cao.
 - + Bộ nhớ flash tích hợp và RAM, cung cấp không gian lưu trữ và xử lý dữ liệu.
 - + Các giao tiếp ngoại vi như UART, SPI, I2C, ADC, và DAC, cho phép kết nối với nhiều loại cảm biến và thiết bị khác nhau.
 - + Khả năng tiết kiệm năng lượng, phù hợp cho các ứng dụng IoT chạy bằng pin.
- ESP32 được lựa chọn vì khả năng xử lý mạnh mẽ, tích hợp Wi-Fi, và khả năng tương thích với nhiều giao thức và công nghệ web, đáp ứng yêu cầu của hệ thống quản lý thiết bị IoT cục bộ

1.2. Giới thiệu về IoT và vai trò của giao diện quản lý

- IoT (Internet of Things) là mạng lưới các thiết bị vật lý được nhúng cảm biến, phần mềm, và các công nghệ khác, cho phép chúng kết nối và trao đổi dữ liệu với nhau qua internet. Một hệ thống IoT hoàn chỉnh thường bao gồm:
 - + Phần cứng: Các cảm biến, bộ điều khiển (như ESP32), thiết bị ngoại vi.
 - + Phần mềm: Giao diện điều khiển, nền tảng quản lý từ xa.
 - + Kết nối mạng: WiFi, Bluetooth hoặc các giao thức IoT khác.
- Giao diện quản lý giúp người dùng dễ dàng giám sát và điều khiển thiết bị IoT mà không cần can thiệp trực tiếp vào phần cứng.

1.3. Tổng quan về nền tảng Blynk:

- Blynk là một nền tảng IoT Platform-as-a-Service (PaaS) được thiết kế để giúp người dùng nhanh chóng xây dựng giao diện người dùng (UI) trực quan cho các dự án IoT trên các thiết bị di động iOS và Android. Kiến trúc của Blynk bao gồm ba thành phần chính:
 - + Ứng dụng Blynk: Ứng dụng di động cho phép người dùng tạo giao diện điều khiển bằng cách kéo và thả các widget khác nhau (ví dụ: nút bấm, thanh trượt, đồng hồ đo, biểu đồ).
 - + Server Blynk: Server đóng vai trò là trung gian giao tiếp giữa ứng dụng di động và các thiết bị phần cứng. Blynk cung cấp một server công cộng miễn phí, nhưng người dùng cũng có thể thiết lập server riêng (Blynk Private Server hoặc Blynk Local Server).
 - + Thư viện Blynk: Thư viện này được cài đặt trên các thiết bị phần cứng (như ESP32) và cung cấp các hàm để kết nối với server Blynk và trao đổi dữ liệu thông qua các Virtual Pins.
- Virtual Pins là các cổng giao tiếp ảo giữa thiết bị phần cứng (ESP32) và ứng dụng Blynk, cho phép truyền dữ liệu hai chiều mà không cần kết nối vật lý. Chúng được đánh số từ 'V0' đến 'V255'.
- Virtual Pins hoạt động như các kênh dữ liệu ảo, cho phép ESP32 và ứng dụng Blynk trao đổi thông tin hai chiều mà không cần ánh xạ đến chân vật lý. Ví dụ: Dữ liệu cảm biến được gán vào V0, lệnh điều khiển từ app Blynk được nhận qua V1.

- Các tính năng chính của ứng dụng Blynk bao gồm khả năng tạo giao diện tùy chỉnh mà không cần kiến thức lập trình phức tạp, điều khiển thiết bị từ xa qua internet, hiển thị dữ liệu cảm biến theo thời gian thực, thiết lập thông báo và sử dụng tính năng Energy để quản lý mức độ sử dụng các widget. Virtual Pins là một khái niệm cốt lõi trong Blynk, chúng là các biến ảo trên server Blynk mà ứng dụng và thiết bị có thể đọc và ghi dữ liệu, cho phép tương tác giữa giao diện người dùng và phần cứng. Việc sử dụng Blynk giúp đơn giản hóa quá trình phát triển giao diện người dùng cho các dự án IoT, cho phép tập trung vào logic điều khiển và thu thập dữ liệu của thiết bị.

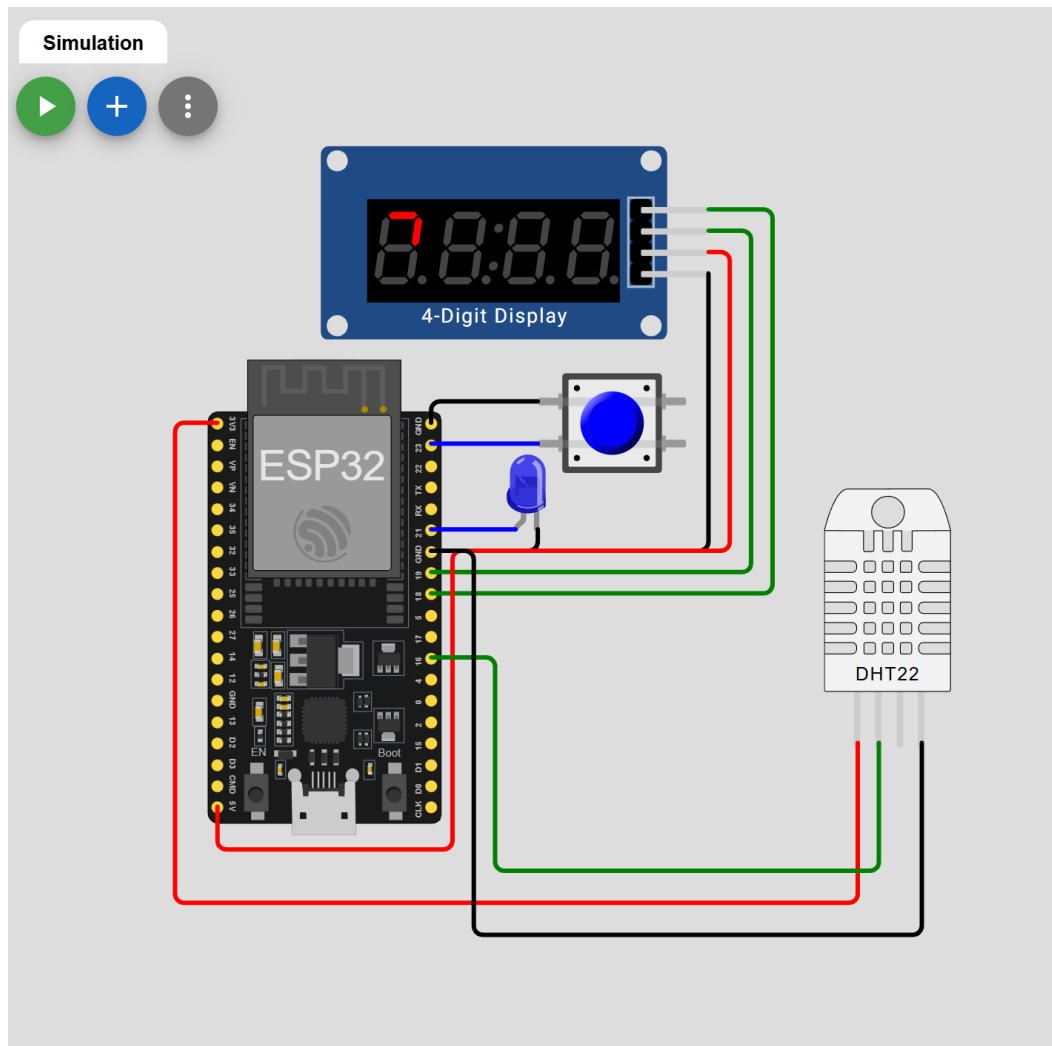
1.4. Giới thiệu về Wokwi và các tính năng mô phỏng ESP32

- Wokwi là một trình mô phỏng trực tuyến miễn phí và mạnh mẽ cho Arduino và ESP32, hoạt động trực tiếp trên trình duyệt web. Nó cung cấp một môi trường tích hợp để viết code, xây dựng mạch điện ảo và mô phỏng hành vi của phần cứng mà không cần đến các linh kiện vật lý. Wokwi hỗ trợ một loạt các thành phần điện tử phổ biến, bao gồm ESP32, Arduino, các loại cảm biến (ví dụ: DHT22, cảm biến ánh sáng), actuator (ví dụ: LED, servo) và nhiều module khác.
- Đối với ESP32, Wokwi mô phỏng các chân GPIO, cho phép bạn kết nối các thành phần ảo. Nó cũng cung cấp khả năng mô phỏng kết nối Wi-Fi, cho phép ESP32 ảo kết nối với internet ảo và tương tác với các dịch vụ đám mây như server Blynk. Serial Monitor trong Wokwi cho phép bạn xem các thông báo debug và dữ liệu được in ra từ code ESP32 ảo.
- Một lợi ích quan trọng của Wokwi là khả năng chia sẻ dự án dễ dàng thông qua URL, giúp cộng tác và trình bày ý tưởng một cách thuận tiện. Việc sử dụng Wokwi trong đề tài này cho phép chúng tôi thử nghiệm và đánh giá hệ thống quản lý thiết bị IoT với Blynk mà không cần mua và cấu hình phần cứng thực tế, tiết kiệm thời gian và chi phí.

CHƯƠNG 2: XÂY DỰNG HỆ THỐNG QUẢN LÝ IOT

2.1. Thiết kế sơ đồ trên Wokwi

Thiết bị	ESP32 Pin	Ghi chú chức năng
LED màu xanh (LED đơn)	GPIO 21	Điều khiển bật/tắt bằng nút nhấn
Nút nhấn màu xanh	GPIO 23	Đọc trạng thái nhấn, đảo trạng thái LED
TM1637 (4-digit display)	CLK: GPIO 18 DIO: GPIO 19	Hiển thị thời gian hoạt động của hệ thống
Cảm biến DHT22	DATA: GPIO 16	Đọc nhiệt độ và độ ẩm từ môi trường



Hình 2 Sơ đồ Wokwi

2.2. Lập trình hệ thống trên VSCode và PlatformIO

- Viết chương trình mô phỏng trên Wokwi với các chức năng: xử lý cảm biến DHT22, điều khiển LED, đọc trạng thái nút nhấn, hiển thị thời gian lên TM1637.
- Sau khi hoàn chỉnh chức năng mô phỏng trên Wokwi, bắt đầu tích hợp thư viện Blynk vào chương trình.
- Thêm các thông tin từ Blynk sau khi đã tạo từ mục 2.3:

```
#define BLYNK_TEMPLATE_ID "TMPL6pEvA_RXf"
#define BLYNK_TEMPLATE_NAME "BLYNK"
#define BLYNK_AUTH_TOKEN "VYupZeVSLopv1KEBTu9_tK1X5pI3rSYr"
```

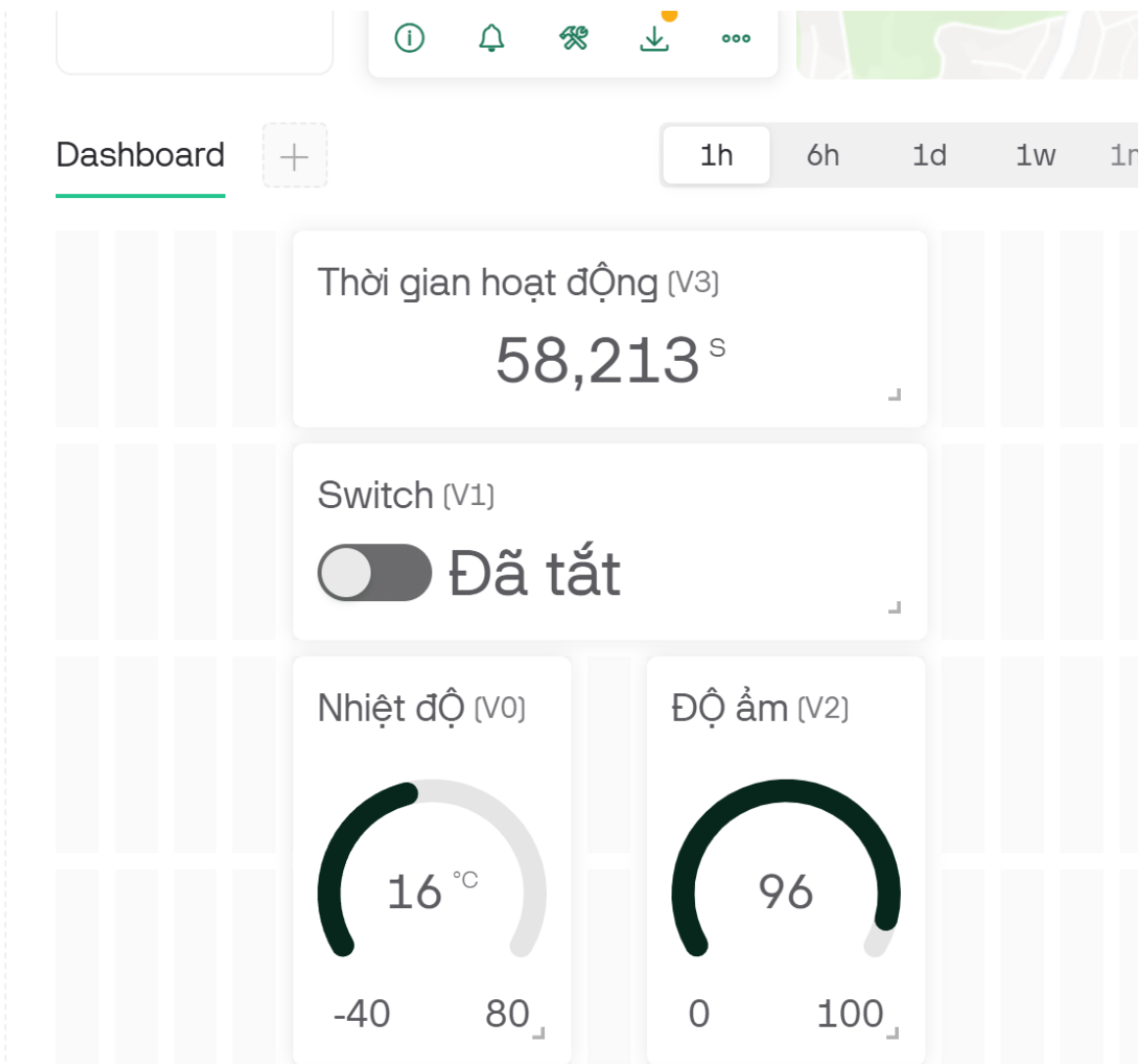
Hình 3 Thông tin từ Blynk cá nhân

- Kết nối các virtual pin với các chức năng tương ứng (V0 - nhiệt độ, V2 - độ ẩm, V1 - trạng thái LED, V3 - thời gian hoạt động).

2.3. Tạo giao diện điều khiển trên Blynk

- Tạo các widget như nút bấm để điều khiển LED, màn hình hiển thị nhiệt độ và độ ẩm

Virtual Pin	Chức năng	Ghi chú
V0	Hiển thị nhiệt độ từ cảm biến DHT22	Đọc dữ liệu nhiệt độ từ ESP32 gửi về.
V1	Điều khiển trạng thái đèn LED	Khi nhấn Switch, gửi trạng thái để bật/tắt LED.
V2	Hiển thị độ ẩm từ cảm biến DHT22	Đọc dữ liệu độ ẩm từ ESP32 gửi về.
V3	Hiển thị thời gian hoạt động	Đếm số giây hệ thống đã chạy.

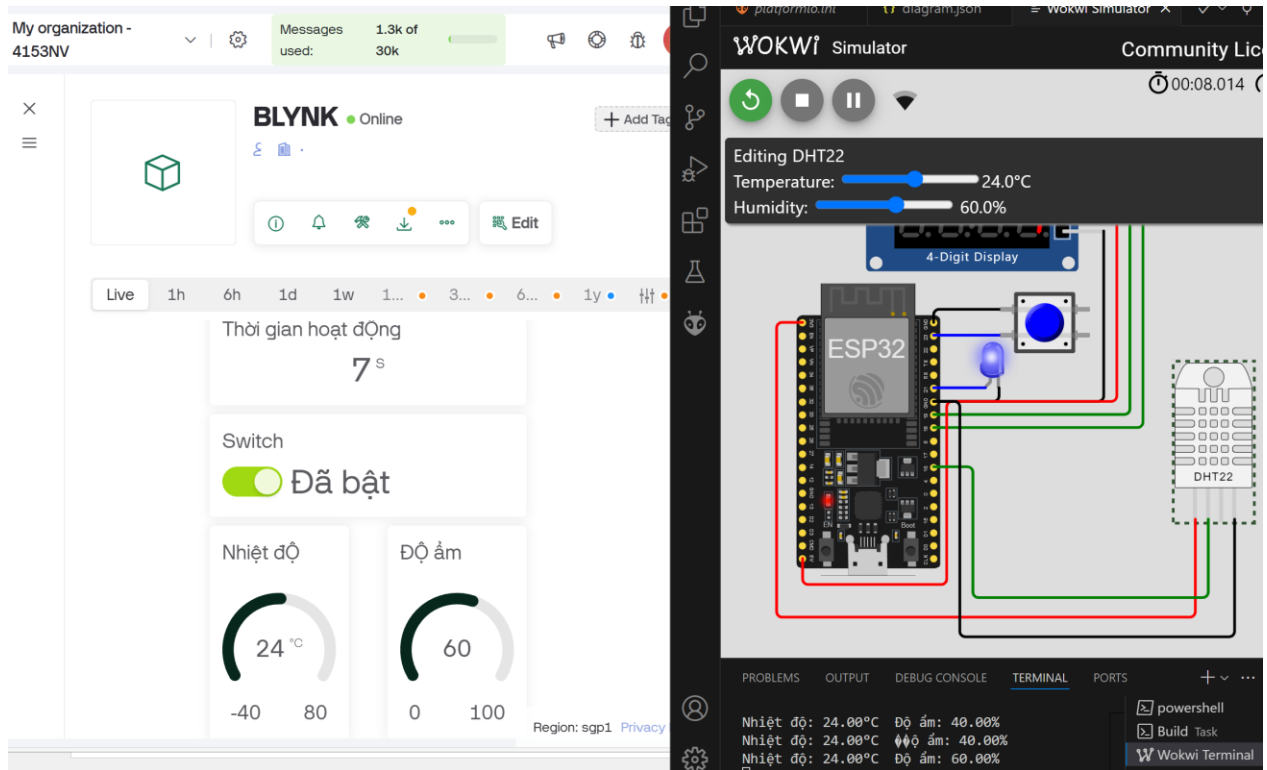


Hình 4 Giao diện Blynk

- Lưu mã Auth Token từ Blynk để sử dụng trong mã nguồn

CHƯƠNG 3: THỬ NGHIỆM VÀ ĐÁNH GIÁ

3.1. Kết quả chạy thử nghiệm:



Hình 5 Kết quả chạy thử nghiệm

STT	Chức năng	Kết quả
1	Đọc nhiệt độ từ DHT22	Giá trị nhiệt độ cập nhật đúng định kỳ
2	Đọc độ ẩm từ DHT22	Giá trị độ ẩm cập nhật đúng định kỳ
3	Điều khiển LED bằng nút nhấn vật lý	LED thay đổi trạng thái và trạng thái này cũng được cập nhật lên Blynk (V1)
4	Điều khiển LED từ Blynk	Dữ liệu gửi từ Blynk tới ESP32 điều khiển được trạng thái LED
5	Hiển thị thời gian hoạt động trên TM1637	Màn hình hiển thị đúng thời gian hoạt động của thiết bị (tính bằng giây). Khi tắt LED bằng nút nhấn, màn hình cũng tắt.
6	Gửi dữ liệu lên Blynk	Giá trị nhiệt độ, độ ẩm, trạng thái LED và thời gian hoạt động đều được cập nhật thường xuyên lên app Blynk qua các Virtual Pin V0 – V3
7	Kết nối WiFi và Blynk	ESP32 kết nối thành công với mạng Wokwi-GUEST và duy trì ổn định trong suốt thời gian chạy thử

3.3. Thách thức trong quá trình mô phỏng và triển khai thực tế

- Trong quá trình xây dựng hệ thống trên Wokwi, một số thách thức đã xuất hiện. Đầu tiên, việc cấu hình Virtual Pins trên Blynk đòi hỏi sự chính xác trong việc ánh xạ giữa giao diện ứng dụng và thiết bị ảo. Một lỗi nhỏ trong việc gán Virtual Pin (ví dụ: nhầm V0 thành V1) có thể khiến dữ liệu không hiển thị đúng. Để khắc phục, nhóm đã kiểm tra kỹ lưỡng các thông số cấu hình và sử dụng Serial Monitor của Wokwi để debug.
- Thứ hai, kết nối WiFi ảo trên Wokwi đôi khi không ổn định, đặc biệt khi mô phỏng chạy trong thời gian dài. Điều này yêu cầu khởi động lại mô phỏng để đảm bảo ESP32 duy trì liên kết với server Blynk. Mặc dù Wokwi cung cấp môi trường mô phỏng lý tưởng, hạn chế này gợi ý rằng trong thực tế, việc chọn module WiFi chất lượng cao và tối ưu hóa mã nguồn sẽ rất quan trọng.
- Khi xem xét triển khai trên phần cứng thực tế, một số thách thức tiềm ẩn bao gồm:
 - + **Sai số cảm biến:** Cảm biến DHT22 thực tế có sai số khoảng $\pm 2^{\circ}\text{C}$ cho nhiệt độ và $\pm 5\%$ cho độ ẩm, đòi hỏi hiệu chỉnh trong quá trình xử lý dữ liệu.
 - + **Chi phí và độ bền:** Việc mua các linh kiện như ESP32, DHT22, và TM1637 có thể tăng chi phí, trong khi môi trường thực tế (nhiệt độ cao, độ ẩm lớn) có thể ảnh hưởng đến độ bền của thiết bị.
 - + **Nguồn điện:** Hệ thống thực tế cần nguồn điện ổn định, đặc biệt khi sử dụng lâu dài, có thể yêu cầu tích hợp pin hoặc bộ chuyển đổi.
- Những thách thức này nhấn mạnh giá trị của việc thử nghiệm trên Wokwi để phát hiện lỗi sớm, đồng thời cung cấp bài học kinh nghiệm cho việc triển khai thực tế, bao gồm tối ưu hóa cấu hình mạng và lựa chọn linh kiện đáng tin cậy.

3.2. Ưu nhược điểm của dự án:

* Ưu điểm:

- Dễ triển khai và tiết kiệm chi phí: Việc mô phỏng toàn bộ hệ thống trên nền tảng Wokwi giúp giảm chi phí phần cứng và tránh rủi ro hỏng hóc thiết bị thực trong quá trình thử nghiệm.
- Tích hợp Blynk dễ dàng: Nhờ vào nền tảng Blynk IoT, việc giao tiếp và điều khiển thiết bị từ xa qua điện thoại, web trở nên trực quan, dễ sử dụng, không cần lập trình giao diện phức tạp.
- Mã nguồn gọn gàng, rõ ràng: Mã chương trình được tổ chức hợp lý, tách biệt từng chức năng như xử lý cảm biến, điều khiển LED, cập nhật dữ liệu,... giúp dễ dàng bảo trì và phát triển thêm.
- Hiện thị trực quan trên màn hình TM1637: Người dùng có thể dễ dàng quan sát thời gian hoạt động của thiết bị ngay trên phần cứng mô phỏng.
- Tính tương tác cao: Có thể điều khiển LED bằng cả nút nhấn vật lý và ứng dụng Blynk, tạo sự linh hoạt và phản hồi hai chiều giữa phần cứng và phần mềm.

* Nhược điểm:

- Chỉ mới mô phỏng, chưa triển khai thực tế: Dự án mới dừng ở mức mô phỏng trên Wokwi, chưa được kiểm chứng với phần cứng thật nên có thể có sai số khi vận hành ngoài thực tế (ví dụ: cảm biến DHT22 thực có sai số $\pm 2\sim 5\%$).
- Phụ thuộc vào WiFi và máy chủ Blynk: Nếu mất kết nối Internet hoặc máy chủ Blynk gặp sự cố, chức năng điều khiển từ xa sẽ không hoạt động.
- Bảo mật chưa cao: Dự án chưa tích hợp các phương pháp mã hóa hoặc xác thực nâng cao khi giao tiếp giữa thiết bị và nền tảng Blynk.
- Giao diện Blynk còn đơn giản: Ứng dụng Blynk mới chỉ hiển thị và điều khiển các thông số cơ bản, chưa có các tính năng như biểu đồ, cảnh báo hay nhật ký hệ thống.

CHƯƠNG 4: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

4.1. Kết luận

- Qua quá trình nghiên cứu và triển khai mô phỏng, đề tài đã đạt được các mục tiêu đề ra ban đầu. Cụ thể:
- Hệ thống được xây dựng và mô phỏng hoàn chỉnh trên nền tảng Wokwi, giúp kiểm thử các chức năng mà không cần phần cứng thực tế.
- ESP32 đã hoạt động ổn định, thực hiện được các chức năng cơ bản như: đọc dữ liệu từ cảm biến DHT22, điều khiển LED bằng nút nhấn, hiển thị thời gian hoạt động lên màn hình TM1637, và đặc biệt là gửi – nhận dữ liệu với ứng dụng Blynk thông qua các virtual pin.
- Giao diện điều khiển được xây dựng trên ứng dụng Blynk, trực quan và dễ thao tác. Người dùng có thể giám sát và điều khiển thiết bị từ xa thông qua Internet.
- Việc sử dụng nền tảng Visual Studio Code kết hợp PlatformIO đã giúp đơn giản hóa quá trình viết mã và tổ chức dự án, đồng thời dễ dàng tích hợp thư viện bên ngoài như Blynk, DHT, TM1637.

4.2. Hướng phát triển

Đề tài hiện tại mới chỉ dừng ở mức mô phỏng cơ bản và thực hiện các chức năng điều khiển, giám sát đơn lẻ. Trong tương lai, hệ thống có thể được mở rộng theo các hướng sau:

- Triển khai trên phần cứng thực tế: Sử dụng ESP32, cảm biến DHT22, module hiển thị TM1637 và LED để thử nghiệm ngoài đời thật, đảm bảo tính ổn định trong môi trường thực tế.
- Phát triển thêm tính năng tự động: Ví dụ, hệ thống có thể tự động bật/tắt thiết bị khi nhiệt độ/độ ẩm vượt ngưỡng cho phép.
- Lưu trữ dữ liệu lịch sử: Kết nối Blynk với nền tảng lưu trữ đám mây hoặc Firebase để theo dõi và phân tích dữ liệu cảm biến theo thời gian.
- Tối ưu giao diện người dùng: Thiết kế giao diện trực quan hơn, bổ sung biểu đồ thời gian thực, cảnh báo bằng notification.
- Bảo mật kết nối: Áp dụng các cơ chế bảo mật khi giao tiếp giữa ESP32 và ứng dụng, nhằm đảm bảo an toàn cho dữ liệu và thiết bị.

PHỤ LỤC

- **Link Source code dự án:**

https://github.com/nguyenthibaongoc1603/TieuLuan_ESP32

TÀI LIỆU THAM KHẢO

- [1] Wokwi Documentation. <https://docs.wokwi.com/>
- [2] Blynk IoT Platform. <https://blynk.io/>
- [3] Arduino ESP32 Guide. <https://docs.espressif.com/>
- [4] PlatformIO Documentation. <https://docs.platformio.org/>
- [5] Tài liệu khoa học về IoT và vi điều khiển.
- [6] Chợ Trời Hà Nội. "ESP32 – Vi điều khiển đa năng cho dự án IoT." <https://chotroi.vn/esp32-vi-dieu-khien-da-nang-cho-du-an-iot>
- [7] SriTu Hobby. "How to Set Up the New Blynk App with an ESP32 Board." <https://srituhobby.com/how-to-set-up-the-new-blynk-app-with-an-esp32-board/>
- [8] Điện Thông Minh E-Smart. "Blynk IoT và ESP32." <https://dienthongminhesmart.com/lap-trinh-esp32/blynk-iot-va-esp32/>
- [9] IoT Zone. "Top dự án lập trình ESP32 IoT hay nhất 2024." <https://www.iotzone.vn/esp32/top-du-an-esp32-iot/>
- [10] Arduino Kit. "Thiết kế nhà thông minh bằng Arduino sử dụng ESP32 và Blynk 2.0." <https://arduino.vn/thiet-ke-nha-thong-minh-bang-arduino-su-dung-esp32-va-blynk-2-0/>
- [11] IoT Zone. "Dự án ESP32 Blynk – Bật tắt đèn LED qua WiFi." <https://www.iotzone.vn/esp32/du-an-esp32-blynk-bat-tat-den-led-qua-wifi/>
- [12] Điện Thông Minh E-Smart. "Điều khiển thiết bị qua internet dùng ESP32 và Blynk IoT." <https://dienthongminhesmart.com/du-an-iot/dieu-khien-qua-internet/>
- [13] FUVITECH. "Xây dựng trạm thời tiết ứng dụng IoT với ESP32." <https://fuvitech.vn/2021/08/23/xay-dung-tram-thoi-tiet-ung-dung-iot-voi-esp32/>
- [14] Deviot. "Lập trình IoT chip ESP32-IDF." <https://deviot.vn/products/lap-trinh-iot-lap-trinh-iot-chip-esp32-idf>

