

ĐẠI HỌC HUẾ
TRƯỜNG ĐẠI HỌC KHOA HỌC



ĐẠI HỌC HUẾ
TRƯỜNG ĐẠI HỌC KHOA HỌC

Tên đề tài:

GIÁM SÁT NĂNG LƯỢNG TIÊU THỤ VỚI ESP32

Học Phần:

Phát Triển Ứng Dụng IoT – Nhóm 4

Giảng Viên Hướng Dẫn:

ThS. Võ Việt Dũng

Tp Huế, tháng 4 năm 2025

MỤC LỤC

LỜI MỞ ĐẦU	3
1.Lý do chọn đề tài	3
2. Mục tiêu của đề tài	3
3.Phương pháp nghiên cứu	4
4. Ý nghĩa của đề tài.....	4
NỘI DUNG	5
CHƯƠNG 1: TỔNG QUAN VỀ HỆ THỐNG GIÁM SÁT NĂNG LƯỢNG TIÊU THỤ VỚI ESP32	5
1. Mục tiêu của hệ thống giám sát năng lượng tiêu thụ	5
2. Vai trò và ứng dụng của ESP32 trong giám sát năng lượng	5
3.Tổng quan về ESP32	5
CHƯƠNG 2: THIẾT KẾ HỆ THỐNG GIÁM SÁT NĂNG LƯỢNG TIÊU THỤ	6
1. Kiến trúc tổng quan.....	6
2. Thành phần phần cứng	6
3.Thành phần phần mềm	7
CHƯƠNG 3: NGUYÊN LÝ HOẠT ĐỘNG CỦA HỆ THỐNG.....	10
1. Thu thập dữ liệu	10
2.Xử lý dữ liệu	10
3. Cơ chế cảnh báo	11
3. Sơ đồ khối hệ thống	11
CHƯƠNG 4: MÔ PHỎNG HỆ THỐNG TRÊN WOKWI.....	11
1. Mục tiêu mô phỏng	11
2. Thiết lập sơ đồ trên Wokwi	12
3. Quy trình thiết lập mô phỏng	13
4. Mô phỏng dữ liệu	13
CHƯƠNG 5: ĐÁNH GIÁ VÀ TRIỂN VỌNG	13
1. Ưu điểm của hệ thống.....	13
2. Hạn chế	13
3. Triển vọng phát triển	13
CHƯƠNG 6: ỨNG DỤNG THỰC TẾ VÀ TRIỂN KHAI	14

1. Ứng dụng trong đời sống.	14
2. Quy trình triển khai thực tế	14
KẾT LUẬN	15
TÀI LIỆU THAM KHẢO	16

LỜI MỞ ĐẦU

1. Lý do chọn đề tài

Trong bối cảnh nhu cầu sử dụng năng lượng ngày càng gia tăng cùng với yêu cầu cấp thiết về tiết kiệm và quản lý hiệu quả tài nguyên, việc giám sát năng lượng tiêu thụ đã trở thành một lĩnh vực nghiên cứu quan trọng và mang tính thực tiễn cao. Sự phát triển của công nghệ IoT (Internet of Things) đã mở ra nhiều cơ hội để xây dựng các hệ thống giám sát thông minh, cho phép theo dõi và phân tích dữ liệu năng lượng một cách chính xác, tiện lợi. Với sự hỗ trợ của các vi điều khiển mạnh mẽ như ESP32 và các cảm biến hiện đại như ACS712, việc đo lường công suất tiêu thụ không chỉ trở nên dễ dàng mà còn có thể tích hợp với các giải pháp lưu trữ dữ liệu linh hoạt, chẳng hạn như thẻ SD hoặc nền tảng đám mây.

Đề tài này tận dụng những ưu điểm đó để phát triển một giải pháp thực tiễn. Tiểu luận này tập trung vào nghiên cứu và triển khai hệ thống giám sát năng lượng tiêu thụ dựa trên vi điều khiển ESP32. Dữ liệu thu thập được sẽ được lưu trữ cục bộ trên thẻ SD hoặc truyền tải lên đám mây, tạo điều kiện cho việc phân tích và quản lý năng lượng hiệu quả.

Với tinh thần khám phá và sáng tạo, tiểu luận sẽ trình bày chi tiết quá trình thiết kế, triển khai và đánh giá hiệu quả của hệ thống, đồng thời thảo luận về những thách thức và hướng phát triển trong tương lai. Hy vọng rằng nội dung nghiên cứu sẽ mang lại giá trị tham khảo hữu ích và khơi nguồn cảm hứng cho các ứng dụng công nghệ IoT trong lĩnh vực quản lý năng lượng.

2. Mục tiêu của đề tài

- **Xây dựng hệ thống đo lường công suất tiêu thụ chính xác:** Sử dụng cảm biến dòng điện ACS712 kết hợp với vi điều khiển ESP32 để thu thập dữ liệu về dòng điện và công suất tiêu thụ của các thiết bị điện một cách liên tục và đáng tin cậy.
- **Lưu trữ và quản lý dữ liệu năng lượng:** Phát triển giải pháp lưu trữ dữ liệu đo lường, bao gồm lưu trữ cục bộ trên thẻ SD để đảm bảo tính sẵn sàng và truyền tải dữ liệu lên nền tảng đám mây để hỗ trợ truy cập từ xa và phân tích lâu dài.
- **Tích hợp công nghệ IoT để giám sát thời gian thực:** Tận dụng khả năng kết nối Wi-Fi của ESP32 để gửi dữ liệu năng lượng tiêu thụ theo thời gian thực, giúp người dùng theo dõi và kiểm soát mức tiêu thụ năng lượng từ xa thông qua các ứng dụng hoặc giao diện web.
- **Đánh giá hiệu quả và tối ưu hóa hệ thống:** Phân tích hiệu suất của hệ thống giám sát, bao gồm độ chính xác của cảm biến, tốc độ truyền dữ liệu và khả năng mở rộng, từ đó đề xuất các cải tiến để nâng cao tính ứng dụng thực tiễn.
- **Đóng góp vào mục tiêu tiết kiệm năng lượng:** Cung cấp một giải pháp giám sát năng lượng chi phí thấp, dễ triển khai, hỗ trợ người dùng nhận biết và điều chỉnh thói quen tiêu thụ điện, góp phần vào việc sử dụng năng lượng bền vững và giảm thiểu lãng phí.

-> Đề tài này hướng đến việc thiết kế một hệ thống giám sát năng lượng tiêu thụ dựa trên ESP32, sử dụng cảm biến dòng điện ACS712 để đo công suất tiêu thụ của các thiết bị điện. Cụ thể, hệ thống sẽ:

- + Đo lường dòng điện và tính toán công suất tiêu thụ theo thời gian thực.
- + Lưu trữ dữ liệu cục bộ trên thẻ SD hoặc gửi lên đám mây thông qua các nền tảng như ThingSpeak, Blynk hoặc Telegram.
- + Mô phỏng toàn bộ hệ thống trên Wokwi để kiểm tra tính khả thi và minh họa cách hoạt động.

3. Phương pháp nghiên cứu

Để thực hiện đề tài, phương pháp nghiên cứu được xây dựng một cách có hệ thống, kết hợp giữa nghiên cứu lý thuyết, thiết kế mô phỏng và phân tích dữ liệu. Các bước cụ thể bao gồm:

- **Nghiên cứu lý thuyết:** Thu thập tài liệu về ESP32, cảm biến ACS712, các giao thức IoT (HTTP, MQTT) và nền tảng ThingSpeak/Blynk/Telegram.

- **Thiết kế và mô phỏng trên Wokwi:**

+ **Thiết kế phần cứng:** Xây dựng sơ đồ mạch trên Wokwi, sử dụng ESP32, nút bấm để mô phỏng hệ thống.

+ **Mô phỏng cảm biến:** Vì Wokwi không hỗ trợ ACS712, sử dụng Potentiometer hoặc giá trị ngẫu nhiên để tạo tín hiệu analog, đại diện cho dòng điện đo được.

+ **Tích hợp IoT:** Mô phỏng gửi dữ liệu đến ThingSpeak/Blynk/Telegram bằng cách xuất thông tin ra Serial Monitor với định dạng phù hợp (HTTP POST, Virtual Pin hoặc thông báo bot).

4. Ý nghĩa của đề tài

- Hệ thống này không chỉ mang lại lợi ích thực tiễn như giám sát năng lượng trong gia đình mà còn là một bài học điển hình về ứng dụng IoT trong đời sống. Ví dụ, người dùng có thể phát hiện thiết bị tiêu tốn điện bất thường (như quạt hỏng) và thay thế kịp thời. Kết quả nghiên cứu cũng có thể áp dụng vào các dự án thực tế hoặc làm nền tảng cho các cải tiến trong tương lai, như tích hợp trí tuệ nhân tạo để dự đoán mức tiêu thụ.

- Sự phát triển của Internet of Things (IoT) đã mở ra nhiều ứng dụng quan trọng trong đời sống, đặc biệt là trong lĩnh vực giám sát và quản lý năng lượng. Một trong những ứng dụng phổ biến là hệ thống giám sát năng lượng tiêu thụ sử dụng vi điều khiển ESP32 và cảm biến đo dòng điện như ACS712. Bài viết này trình bày cách xây dựng hệ thống giám sát năng lượng tiêu thụ, lưu trữ dữ liệu trên thẻ SD hoặc đám mây và mô phỏng trên nền tảng Wokwi.

NỘI DUNG

CHƯƠNG 1: TỔNG QUAN VỀ HỆ THỐNG GIÁM SÁT NĂNG LƯỢNG TIÊU THỤ VỚI ESP32

1. Mục tiêu của hệ thống giám sát năng lượng tiêu thụ

Hệ thống giám sát năng lượng tiêu thụ được thiết kế nhằm:

- Đo lường chính xác công suất tiêu thụ của các thiết bị điện.
- Lưu trữ dữ liệu tiêu thụ năng lượng để phân tích dài hạn.
- Hiển thị dữ liệu theo thời gian thực thông qua các nền tảng trực quan.
- Cảnh báo người dùng khi phát hiện mức tiêu thụ bất thường.
- Tích hợp vào các ứng dụng IoT để quản lý năng lượng thông minh.

2. Vai trò và ứng dụng của ESP32 trong giám sát năng lượng

2.1 Vai Trò của ESP32 trong giám sát năng lượng

ESP32 là một vi điều khiển mạnh mẽ với các đặc điểm nổi bật:

- Kết nối không dây: Hỗ trợ Wi-Fi và Bluetooth, cho phép gửi dữ liệu đến đám mây hoặc thiết bị di động.
- Hiệu suất cao: CPU lõi kép, tốc độ xử lý nhanh, phù hợp cho các ứng dụng thời gian thực.
- GPIO đa dạng: Dễ dàng kết nối với nhiều loại cảm biến và thiết bị ngoại vi.
- Tiết kiệm năng lượng: Các chế độ tiết kiệm năng lượng phù hợp cho ứng dụng giám sát liên tục.

ESP32 đóng vai trò trung tâm trong việc thu thập dữ liệu từ cảm biến, xử lý thông tin, lưu trữ và truyền dữ liệu đến các nền tảng như ThingSpeak hoặc Blynk.

2.2 Ứng dụng của ESP32 trong giám sát năng lượng

ESP32, với các tính năng vượt trội như khả năng kết nối Wi-Fi/Bluetooth, hiệu suất xử lý cao, và khả năng giao tiếp với nhiều loại cảm biến, đã trở thành một công cụ lý tưởng trong các hệ thống giám sát năng lượng tiêu thụ.

3. Tổng quan về ESP32

ESP32 là một vi điều khiển nhỏ gọn, mạnh mẽ do Espressif Systems phát triển, được sử dụng rộng rãi trong các dự án IoT, điện tử và tự động hóa. Dưới đây là những điểm chính về ESP32:

Chức năng chính:

- + **CPU mạnh:** Lõi kép, tốc độ lên đến 240 MHz, xử lý nhanh các tác vụ.
- + **Kết nối không dây:** Có Wi-Fi và Bluetooth tích hợp, dễ dàng kết nối mạng và thiết bị khác.
- + **Nhiều chân giao tiếp:** Hỗ trợ kết nối với cảm biến, màn hình, thẻ SD, v.v. qua các giao thức như SPI, I2C, UART.

+ **Tiết kiệm năng lượng:** Có chế độ ngủ (sleep mode) để giảm tiêu thụ điện, phù hợp cho thiết bị chạy pin.

- **Ưu điểm:**

- + Giá rẻ, dễ mua, cộng đồng hỗ trợ lớn.
- + Linh hoạt, phù hợp cho cả người mới học và dự án chuyên nghiệp.
- + Có thể lập trình dễ dàng bằng Arduino IDE, PlatformIO hoặc ESP-IDF.

- **Ứng dụng phổ biến:**

- + Giám sát năng lượng (như đo công suất tiêu thụ).
- + Nhà thông minh (điều khiển đèn, ổ cắm).
- + Thiết bị đeo (wearables), cảm biến IoT, robot nhỏ.

CHƯƠNG 2: THIẾT KẾ HỆ THỐNG GIÁM SÁT NĂNG LƯỢNG TIÊU THỤ

1. Kiến trúc tổng quan

Hệ thống bao gồm 4 khối chính:

- **Khối đo lường:** ACS712 đo dòng điện, ESP32 xử lý tín hiệu và tính công suất.
- **Khối lưu trữ:** Thẻ SD lưu dữ liệu cục bộ, ThingSpeak lưu dữ liệu từ xa.
- **Khối truyền thông:** ESP32 gửi dữ liệu qua Wi-Fi đến ThingSpeak, Blynk hoặc Telegram.
- **Khối bổ sung:** Cảm biến DHT11 đo nhiệt độ và độ ẩm để phân tích mối liên hệ với năng lượng tiêu thụ.

2. Thành phần phần cứng

2.1. ESP32

Hệ thống giám sát năng lượng tiêu thụ được thiết kế với các thành phần phần cứng tối ưu để đảm bảo khả năng đo lường, xử lý, lưu trữ và truyền dữ liệu một cách hiệu quả. Các thành phần được chọn dựa trên tính khả thi, chi phí hợp lý và khả năng mô phỏng trên nền tảng Wokwi.

2.2. Cảm biến dòng điện ACS712

Mô tả: ACS712 là cảm biến dòng điện hoạt động dựa trên hiệu ứng Hall, có khả năng đo dòng điện xoay chiều (AC) hoặc một chiều (DC) với các phiên bản khác nhau (5A, 20A, 30A).

Vai trò trong hệ thống:

- Đo dòng điện chạy qua thiết bị tiêu thụ (như bóng đèn, quạt, hoặc động cơ).

- Cung cấp tín hiệu analog tỷ lệ với giá trị dòng điện, được ESP32 đọc qua chân ADC để tính toán công suất tiêu thụ.

2.3. Cảm biến nhiệt độ và độ ẩm DHT22

Mô tả: DHT22 (hoặc AM2302) là cảm biến kỹ thuật số đo nhiệt độ và độ ẩm, sử dụng giao thức một dây (single-wire protocol) để giao tiếp với vi điều khiển.

Vai trò trong hệ thống:

- Cung cấp dữ liệu bổ sung về điều kiện môi trường (nhiệt độ và độ ẩm), giúp phân tích mối liên hệ giữa môi trường và mức tiêu thụ năng lượng.-
- Dữ liệu từ DHT22 được sử dụng để cảnh báo người dùng nếu môi trường bất lợi (ví dụ: nhiệt độ quá cao có thể làm tăng tiêu thụ điện của điều hòa).

2.4. Module thẻ SD

Mô tả: Module thẻ SD là thiết bị giao tiếp với thẻ nhớ SD/microSD, sử dụng giao thức SPI để lưu trữ dữ liệu cục bộ.

Vai trò trong hệ thống:

- Lưu trữ dữ liệu đo được từ cảm biến (dòng điện, công suất, nhiệt độ, độ ẩm) dưới dạng tệp CSV.
- Đảm bảo dữ liệu không bị mất khi kết nối Wi-Fi không ổn định.

2.5. Nguồn điện

Mô tả: Nguồn điện cung cấp năng lượng cho ESP32 và các cảm biến/module trong hệ thống.

Vai trò trong hệ thống:

- Cấp nguồn 5V cho ESP32 (qua cổng USB hoặc pin ngoài).
- Cấp nguồn 5V cho ACS712 và module thẻ SD.
- Cấp nguồn 3.3V cho DHT22 (từ chân 3.3V của ESP32).

3. Thành phần phần mềm

3.1. Phần mềm mô phỏng trên Wokwi

Mô tả: Wokwi là nền tảng mô phỏng IoT trực tuyến, hỗ trợ ESP32, cảm biến DHT22, module thẻ SD, và các thành phần khác.

Vai trò trong hệ thống:

- Mô phỏng hoạt động của ESP32 khi đọc dữ liệu từ cảm biến ACS712 (giả lập tín hiệu analog), cảm biến DHT22, và ghi dữ liệu vào thẻ SD.
- Giả lập kết nối Wi-Fi để gửi dữ liệu đến ThingSpeak, Blynk, hoặc Telegram.
- Cung cấp giao diện trực quan để thiết kế sơ đồ mạch và kiểm tra tính khả thi của hệ thống.

3.2. Phần mềm cho cảm biến và module

Để giao tiếp với các thành phần phần cứng, hệ thống cần các thư viện phần mềm tương ứng. Dưới đây là các thư viện chính:

- Thư viện ACS712:

+ **Mô tả:** Không có thư viện ACS712 chính thức trong Arduino, nhưng tín hiệu analog từ ACS712 có thể được xử lý bằng các hàm ADC cơ bản của ESP32.

+ **Vai trò:** Đọc tín hiệu analog từ ACS712 (giả lập trên Wokwi bằng giá trị ngẫu nhiên) và chuyển đổi thành giá trị dòng điện (A).

+ **Lý do lựa chọn:** Đơn giản hóa việc xử lý tín hiệu analog trong mô phỏng.

-Thư viện DHT:

+ **Mô tả:** Thư viện “DHT sensor library” của Adafruit hỗ trợ giao tiếp với cảm biến DHT22.

+ **Vai trò:** Đọc dữ liệu nhiệt độ (°C) và độ ẩm (%) từ DHT22, cung cấp thông tin môi trường bổ sung cho hệ thống.

+ **Lý do lựa chọn:** Thư viện này được hỗ trợ trực tiếp trên Wokwi và tương thích với ESP32.

-Thư viện SPI và SD:

+ **Mô tả:** Thư viện “SD” và “SPI” có sẵn trong Arduino IDE dùng để giao tiếp với module thẻ SD qua giao thức SPI.

+ **Vai trò:** Ghi dữ liệu đo được (dòng điện, công suất, nhiệt độ, độ ẩm) vào thẻ SD dưới dạng tệp CSV, với định dạng các cột như: thời gian, dòng điện, công suất, nhiệt độ, độ ẩm.

+ **Lý do lựa chọn:** Đảm bảo lưu trữ cục bộ đáng tin cậy, phù hợp với yêu cầu lưu trữ dữ liệu khi kết nối mạng không ổn định.

3.3. Phần mềm quản lý kết nối Wi-Fi

- **Mô tả:** ESP32 sử dụng thư viện WiFi.h (có sẵn trong Arduino IDE) để thiết lập kết nối không dây với mạng Wi-Fi.

- Vai trò trong hệ thống:

+ Kết nối ESP32 với mạng Wi-Fi giả lập trên Wokwi (ví dụ: mạng “Wokwi-GUEST” không cần mật khẩu).

+ Cho phép truyền dữ liệu từ ESP32 đến các nền tảng đám mây (ThingSpeak, Blynk) hoặc gửi thông báo qua Telegram.

3.4. Nền tảng lưu trữ và hiển thị dữ liệu đám mây

Hệ thống sử dụng các dịch vụ IoT để lưu trữ và hiển thị dữ liệu trên đám mây. Các nền tảng được chọn bao gồm:

ThingSpeak:

- **Mô tả:** ThingSpeak là nền tảng IoT mã nguồn mở, hỗ trợ lưu trữ và phân tích dữ liệu thông qua các kênh (channels) và API HTTP.

- **Vai trò trong hệ thống:**

+ Lưu trữ dữ liệu đo được (dòng điện, công suất, nhiệt độ, độ ẩm) dưới dạng biểu đồ thời gian thực.

+ Cung cấp giao diện web để người dùng theo dõi mức tiêu thụ năng lượng từ xa.

- **Tính năng chính:**

+ Hỗ trợ nhiều trường dữ liệu (fields) trong mỗi kênh (ví dụ: một kênh có thể lưu công suất, nhiệt độ, độ ẩm).

+ Tích hợp MATLAB để phân tích dữ liệu nâng cao (nếu cần).

+ Tần suất gửi dữ liệu tối thiểu 15 giây (cho tài khoản miễn phí).

- **Lý do lựa chọn:** ThingSpeak miễn phí, dễ tích hợp với ESP32, và phù hợp cho việc hiển thị dữ liệu trong mô phỏng.

Blynk:

- **Mô tả:** Blynk là nền tảng IoT cung cấp ứng dụng di động và máy chủ đám mây để giám sát và điều khiển thiết bị.

- **Vai trò trong hệ thống:**

+ Hiển thị dữ liệu thời gian thực qua các widget như biểu đồ, đồng hồ đo (gauge), hoặc nhãn (label).

+ Gửi thông báo đến người dùng khi công suất vượt ngưỡng (ví dụ: >1000W).

+ Cho phép tùy chỉnh giao diện người dùng trên ứng dụng di động.

- **Tính năng chính:**

+ Giao tiếp qua giao thức TCP/IP, đảm bảo tốc độ truyền nhanh.

+ Hỗ trợ cấu hình ngưỡng cảnh báo và điều khiển từ xa (nếu cần).

- **Lý do lựa chọn:** Blynk thân thiện với người dùng, phù hợp cho ứng dụng di động, và có thể mô phỏng trên Wokwi bằng cách giả lập kết nối đám mây.

Telegram

- **Mô tả:** Telegram là ứng dụng nhắn tin hỗ trợ bot, cho phép gửi thông báo tự động thông qua Telegram Bot API.

- Vai trò trong hệ thống:

+ Gửi tin nhắn cảnh báo đến người dùng khi phát hiện mức tiêu thụ năng lượng bất thường (ví dụ: công suất vượt quá 1500W).

+ Cung cấp thông tin tóm tắt về trạng thái hệ thống (như công suất hiện tại, nhiệt độ, độ ẩm).

- Tính năng chính:

+ Giao tiếp qua giao thức HTTPS, dễ dàng tích hợp với ESP32.

+ Cho phép tạo bot bằng BotFather trên Telegram và sử dụng token để gửi tin nhắn.

+ Hỗ trợ gửi thông báo tức thời, không cần ứng dụng bổ sung.

- Thư viện liên quan: Thư viện “UniversalTelegramBot” (trong Arduino IDE) được sử dụng để giao tiếp với Telegram Bot API.

- Lý do lựa chọn: Telegram miễn phí, dễ thiết lập, và cung cấp phương thức giao tiếp nhanh chóng với người dùng, phù hợp cho mô phỏng và ứng dụng thực tế.

CHƯƠNG 3: NGUYÊN LÝ HOẠT ĐỘNG CỦA HỆ THỐNG

1. Thu thập dữ liệu

- Cảm biến bằng ACS712: Cảm biến ACS712 được kết nối với mạch điện của thiết bị tiêu thụ (tải giả lập trên Wokwi, như bóng đèn LED hoặc điện trở). ESP32 đọc tín hiệu điện áp từ ACS712

- Cảm biến nhiệt độ và độ ẩm bằng DHT22:

+ Cảm biến DHT22 được kết nối với ESP32 qua một chân GPIO (ví dụ: GPIO 4) và sử dụng giao thức một dây (single-wire protocol) để truyền dữ liệu.

+ DHT22 cung cấp giá trị nhiệt độ (°C) và độ ẩm (%) của môi trường xung quanh. Dữ liệu này được thu thập để phân tích mối liên hệ giữa điều kiện môi trường và mức tiêu thụ năng lượng (ví dụ: nhiệt độ cao có thể làm tăng tiêu thụ điện của điều hòa).

+ Trên Wokwi, cảm biến DHT22 được hỗ trợ trực tiếp, cho phép mô phỏng giá trị nhiệt độ và độ ẩm thực tế hoặc ngẫu nhiên.

2. Xử lý dữ liệu

- Xử lý tín hiệu từ ACS712: Sau khi thu thập dữ liệu từ cảm biến ACS712 (hoặc tín hiệu giả lập trong Wokwi), hệ thống sử dụng ESP32 để xử lý dữ liệu nhằm tính toán công suất tiêu thụ và chuẩn bị cho việc hiển thị, lưu trữ hoặc gửi lên các nền tảng IoT như ThingSpeak, Blynk hoặc Telegram.

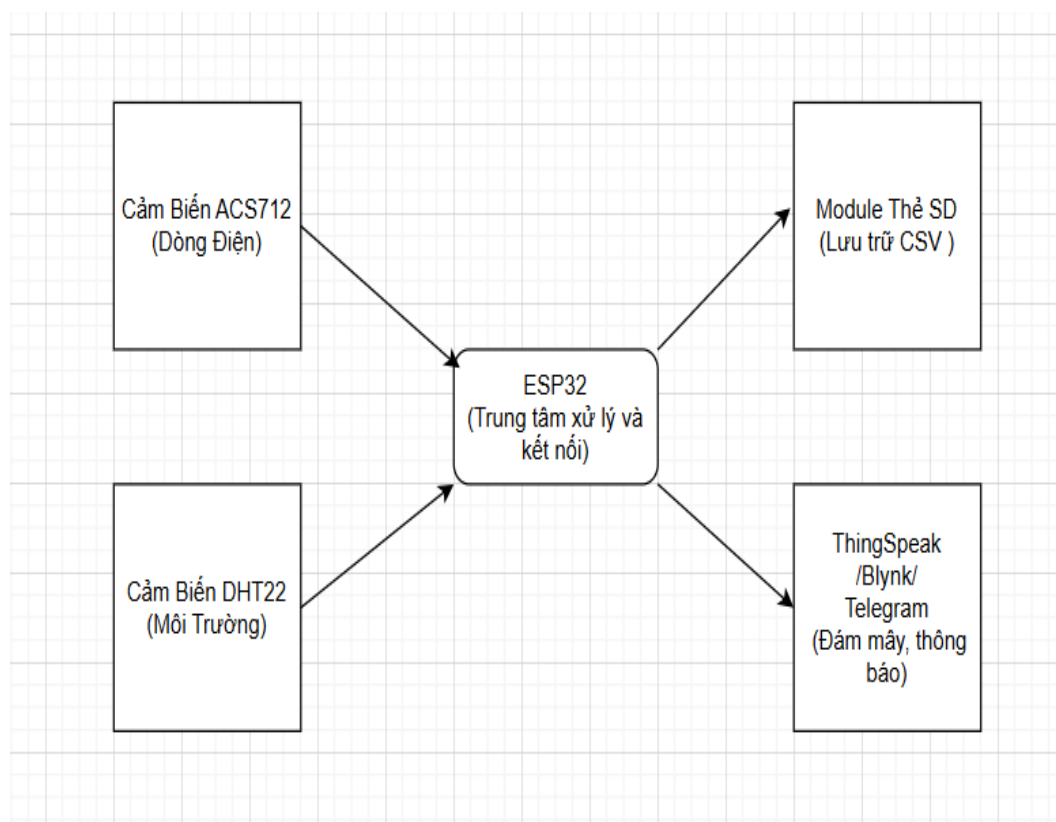
- **Xử lý dữ liệu từ DHT22:** Dữ liệu nhiệt độ và độ ẩm từ DHT22 được đọc qua giao thức một dây và chuyển đổi thành các giá trị số trực tiếp. ESP32 kiểm tra tính hợp lệ của dữ liệu (ví dụ: loại bỏ giá trị bất thường) để đảm bảo độ chính xác.

3. Cơ chế cảnh báo

- Cơ chế cảnh báo là một phần quan trọng của hệ thống giám sát năng lượng tiêu thụ, giúp phát hiện các tình trạng bất thường liên quan đến mức tiêu thụ năng lượng hoặc điều kiện môi trường, từ đó thông báo kịp thời đến người dùng để có hành động khắc phục.

- Hệ thống sử dụng ESP32 để phân tích dữ liệu từ cảm biến (ACS712, DHT22), so sánh với các ngưỡng định trước, và gửi cảnh báo qua các nền tảng như ThingSpeak, Blynk, hoặc Telegram.

3. Sơ đồ khối hệ thống



CHƯƠNG 4: MÔ PHỎNG HỆ THỐNG TRÊN WOKWI

1. Mục tiêu mô phỏng

- Xây dựng hệ thống giám sát năng lượng tiêu thụ dựa trên ESP32.
- Giả lập tín hiệu từ ACS712 để đo dòng điện và tính công suất.

- Hiển thị công suất trên LCD I2C, giả lập lưu trữ trên thẻ SD (qua Serial Monitor), và gửi dữ liệu đến ThingSpeak, Blynk hoặc Telegram. Đảm bảo hệ thống hoạt động ổn định trong môi trường mô phỏng.

2. Thiết lập sơ đồ trên Wokwi

Sơ đồ mô phỏng bao gồm:

ESP32:

- GPIO34: Nối với chân Output của Potentiometer (giả lập tín hiệu ACS712).
- GPIO21 (SDA) và GPIO22 (SCL): Nối với LCD I2C để hiển thị.
- GPIO2: Nối với cực dương của LED (qua điện trở 220Ω) để báo trạng thái.
- GPIO4: Nối với nút bấm (chân còn lại nối GND) để reset hoặc kích hoạt.
- 3.3V và GND: Cấp nguồn cho Potentiometer và LCD.

Potentiometer (giả lập ACS712):

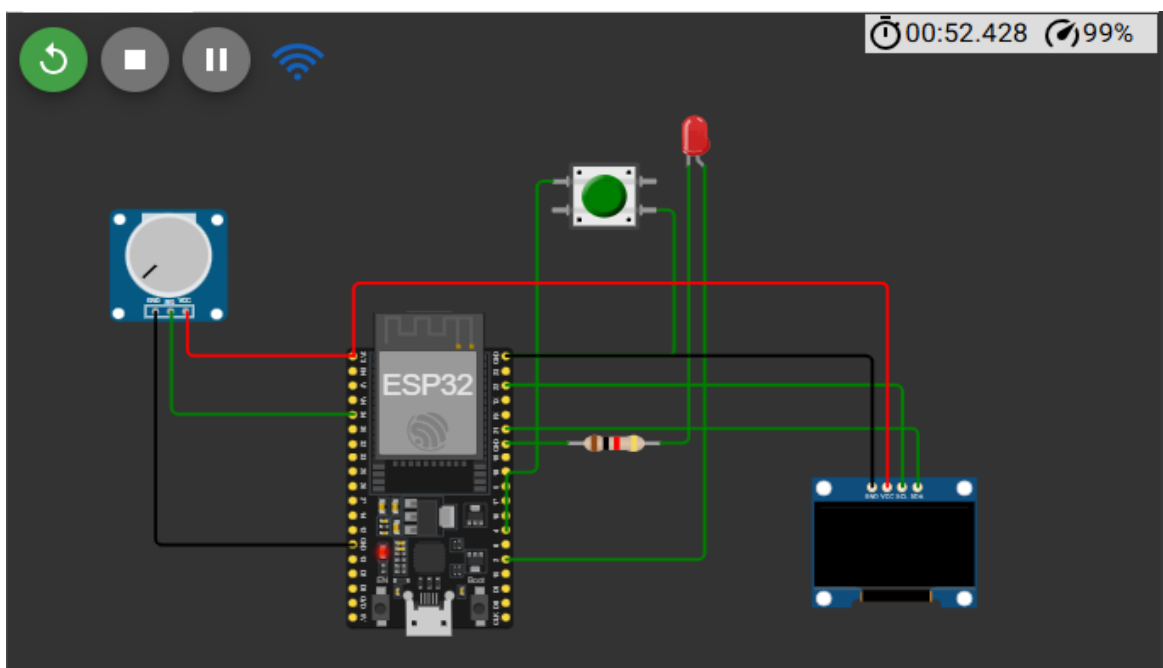
- Chân giữa (Output): Nối GPIO34.
- Chân trái: Nối 3.3V.
- Chân phải: Nối GND.

LED:

- Cực dương: Nối GPIO2 qua điện trở 220Ω.
- Cực âm: Nối GND.

Nút bấm:

- Một chân nối GPIO4, chân còn lại nối GND.



3. Quy trình thiết lập mô phỏng

- ESP32 đọc giá trị ngẫu nhiên từ “ACS712” (0-5A), tính công suất ($P = 5 \times I$).
- Ghi dữ liệu vào thẻ SD mỗi 5 giây.
- Gửi dữ liệu qua Wi-Fi đến ThingSpeak.
- DHT22 cung cấp nhiệt độ và độ ẩm (giả lập).

4. Mô phỏng dữ liệu

- Dữ liệu mô phỏng nằm trong phạm vi hợp lý (0-1000W), phản ánh mức tiêu thụ của thiết bị gia dụng.
- Dù không có ACS712 thực tế, Potentiometer hoặc giá trị ngẫu nhiên đủ để kiểm tra lý thuyết.

CHƯƠNG 5: ĐÁNH GIÁ VÀ TRIỂN VỌNG

1. Ưu điểm của hệ thống

Chi phí thấp: ESP32 (~150.000 VNĐ), ACS712 (~50.000 VNĐ), thẻ SD (~100.000 VNĐ)

phù hợp cho dự án nhỏ.

Linh hoạt: Lưu trữ cục bộ (thẻ SD) và từ xa (ThingSpeak) đáp ứng nhiều nhu cầu, từ offline đến online.

Tính mở rộng: Dễ dàng thêm cảm biến (như áp suất, ánh sáng) hoặc tích hợp AI để phân tích dữ liệu.

2. Hạn chế

Kết nối: Wi-Fi không ổn định (như ở vùng nông thôn) có thể làm gián đoạn truyền dữ liệu lên đám mây.

Mô phỏng: Wokwi thiếu tính thực tế do không có cảm biến vật lý, chỉ phù hợp để kiểm tra ý tưởng ban đầu.

3. Triển vọng phát triển

- Hệ thống giúp người dùng theo dõi công suất tiêu thụ của thiết bị gia dụng (như đèn, quạt, máy lạnh) theo thời gian thực, từ đó nhận biết thiết bị tiêu tốn nhiều điện và điều chỉnh thói quen sử dụng để tiết kiệm chi phí hóa đơn điện.

- Thay vì mô phỏng trên Wokwi, hệ thống có thể được xây dựng với ACS712 thực tế và thẻ SD để lưu trữ dữ liệu. Điều này giúp kiểm tra hiệu suất thực tế, bao gồm nhiều tín hiệu và tốc độ ghi dữ liệu. Ngoài ra, phát triển ứng dụng di động riêng với giao diện tùy chỉnh, thay vì phụ thuộc Blynk, để tăng tính chuyên nghiệp.

CHƯƠNG 6: ỨNG DỤNG THỰC TẾ VÀ TRIỂN KHAI

1. Ứng dụng trong đời sống.

- Đối với hộ gia đình:

- + Giám sát thiết bị gia dụng: Hệ thống đo công suất tiêu thụ của các thiết bị như đèn, quạt, máy lạnh, hoặc tủ lạnh, giúp người dùng nhận biết thiết bị nào tiêu tốn nhiều điện. Ví dụ: Phát hiện máy nước nóng tiêu thụ 800W liên tục, người dùng có thể giảm thời gian sử dụng để tiết kiệm điện.
- + Tiết kiệm chi phí: Dữ liệu công suất hiển thị trên LCD hoặc qua Blynk/Telegram giúp người dùng điều chỉnh thói quen sử dụng, giảm hóa đơn tiền điện.

- Đối với Văn phòng và cửa hàng nhỏ:

- + Quản lý năng lượng tập trung: Theo dõi tiêu thụ của hệ thống chiếu sáng, máy tính, hoặc điều hòa trong văn phòng, hỗ trợ tối ưu hóa vận hành.
- + Lập kế hoạch sử dụng: Dữ liệu từ ThingSpeak (biểu đồ) giúp quản lý xác định thời điểm tiêu thụ cao (giờ cao điểm) để chuyển sang giờ thấp điểm nếu có thể.

- Đối với giáo dục và nghiên cứu:

- + Học tập thực hành: Hệ thống là công cụ lý tưởng cho sinh viên học về IoT, cảm biến ACS712, và xử lý dữ liệu, đặc biệt với chi phí thấp và tính linh hoạt.
- + Dự án nghiên cứu: Dùng làm nền tảng để thử nghiệm các ý tưởng mới, như tích hợp AI hoặc thêm cảm biến bổ sung.

- Đối với bảo vệ môi trường:

- + Giảm khí thải carbon: Tiết kiệm năng lượng thông qua giám sát hiệu quả giúp giảm lượng điện tiêu thụ, từ đó giảm khí thải từ nhà máy điện.
- + Thúc đẩy lối sống bền vững: Dữ liệu minh bạch từ hệ thống khuyến khích người dùng sử dụng năng lượng hợp lý, góp phần vào mục tiêu phát triển bền vững.

2. Quy trình triển khai thực tế

- Lắp đặt:

- + Gắn hệ thống vào thiết bị cần giám sát
- + Đặt LCD ở vị trí dễ nhìn (như trên bàn làm việc) để người dùng theo dõi.

- Kiểm tra:

- + Chạy thử để đảm bảo ACS712 đo chính xác (so sánh với đồng hồ đo điện chuẩn).
- + Kiểm tra lưu trữ trên thẻ SD và dữ liệu gửi qua IoT có đúng định dạng.

- Vận hành:

+ Người dùng theo dõi công suất qua LCD, ứng dụng Blynk, hoặc thông báo Telegram.

+ Dữ liệu trên ThingSpeak được phân tích để lập biểu đồ tiêu thụ hàng ngày/tuần.

- Bảo trì và nâng cấp:

+ **Bảo trì:** Kiểm tra định kỳ kết nối ACS712 và thẻ SD để tránh lỗi đo hoặc ghi dữ liệu, cập nhật phần mềm để cải thiện hiệu suất (giả định).

+ **Nâng cấp:** Tích hợp rơ-le để tự động bật/tắt thiết bị khi vượt ngưỡng công suất, dùng AI để dự đoán xu hướng tiêu thụ dựa trên dữ liệu thẻ SD/đám mây.

KẾT LUẬN

Hệ thống giám sát năng lượng tiêu thụ sử dụng ESP32 với cảm biến dòng điện ACS712 đã chứng minh được tính hiệu quả và tiềm năng trong việc theo dõi, quản lý và tối ưu hóa sử dụng năng lượng. Bằng cách kết hợp khả năng xử lý mạnh mẽ và kết nối không dây của ESP32, hệ thống thu thập dữ liệu từ cảm biến ACS712 để đo lường dòng điện và tính toán công suất tiêu thụ một cách chính xác. Dữ liệu bổ sung từ cảm biến nhiệt độ, độ ẩm (như DHT22) cung cấp cái nhìn toàn diện về môi liên hệ giữa điều kiện môi trường và mức tiêu thụ năng lượng.

Việc lưu trữ dữ liệu linh hoạt trên thẻ SD hoặc các nền tảng đám mây như ThingSpeak và Blynk đảm bảo tính bền vững và dễ dàng truy cập thông tin từ xa. Mô phỏng trên Wokwi đã khẳng định tính khả thi của hệ thống, cho phép kiểm tra thiết kế mà không cần phần cứng vật lý, phù hợp cho nghiên cứu và ứng dụng thực tiễn.

Đề tài không chỉ mang lại giải pháp công nghệ chi phí thấp, dễ triển khai mà còn góp phần thúc đẩy ý thức sử dụng năng lượng bền vững trong bối cảnh tài nguyên ngày càng khan hiếm. Hệ thống có tiềm năng ứng dụng rộng rãi trong các lĩnh vực như hộ gia đình, công nghiệp, nông nghiệp thông minh, và giáo dục. Trong tương lai, việc tích hợp trí tuệ nhân tạo để dự đoán mức tiêu thụ, sử dụng cảm biến tiên tiến hơn, hoặc phát triển ứng dụng di động độc quyền sẽ giúp nâng cao hiệu suất và mở rộng phạm vi ứng dụng của hệ thống.

Vậy nên, đề tài “Giám sát năng lượng tiêu thụ với ESP32” đã đạt được mục tiêu xây dựng một hệ thống giám sát hiệu quả, linh hoạt và thân thiện với người dùng, đồng thời mở ra nhiều hướng phát triển mới cho các nghiên cứu và ứng dụng IoT trong quản lý năng lượng.

TÀI LIỆU THAM KHẢO

1. <https://jetking.fpt.edu.vn/toi-uu-hieu-nang-esp32/>
2. <http://arduino.vn/bai-viet/1183-huong-dan-su-dung-cam-bien-dong-dien-ac712-voi-arduino>
3. <https://mecosun.vn/tin-tuc/giam-sat-nang-luong-va-tam-quan-trong-trong-cong-nghiep-145.html>
4. <https://huynhnhattung.com/dht22-giao-tiep-esp32-cam-bien-nhiet-do-do-am-oled-esp/>
5. <https://datainsight.vn/du-an/trien-khai-he-thong-giam-sat-nang-luong-dien-tieu-thu-cho-nha-may-vsip-2/>
6. Tài liệu kỹ thuật ESP32 từ Espressif Systems
7. Thông số kỹ thuật cảm biến ACS712 và DHT22.
8. <https://vnatech.com.vn/uu-diem-phan-mem-giam-sat-nang-luong/>
9. <https://fptshop.com.vn/tin-tuc/danh-gia/thingspeak-181528>