

Hệ thống Giám sát Tủ điện và Điều khiển Thiết bị bằng ESP32 qua App Flutter

Nguyễn Thành Tín – MSSV: 2123060214



Lớp CCQ2306A, Chuyên ngành Kỹ thuật Điện – điện tử

Khoa Điện – Điện tử, Trường Cao Đẳng Công Thương TP.HCM

Email: tinnguyen12b2@gmail.com , nguyenkimsuyen@hitu.edu.vn

Tóm tắt

Bài báo trình bày thiết kế và triển khai mô hình giám sát tủ điện thông minh sử dụng vi điều khiển ESP32 kết hợp với ứng dụng Flutter. Hệ thống có khả năng đo lường điện năng theo thời gian thực, cảnh báo quá tải bằng LED và còi, điều khiển thiết bị phụ tải từ xa qua app, đồng thời lưu trữ dữ liệu điện năng để tính toán chi phí. Kết quả thực nghiệm cho thấy hệ thống hoạt động ổn định, phản hồi nhanh, giao diện trực quan. Hạn chế chính là quy mô nhỏ và chưa tích hợp AI dự báo. Hướng phát triển tập trung vào mở rộng IoT, tăng cường bảo mật và phân tích dữ liệu.

Từ khóa: ESP32, PZEM-004T, DHT22, Flutter, IoT, giám sát điện năng.

I. Giới thiệu

Trong bối cảnh nhu cầu quản lý điện năng ngày càng cao, việc xây dựng mô hình giám sát và điều khiển tủ điện thông minh có ý nghĩa quan trọng trong nghiên cứu và đào tạo. Đề tài này hướng đến việc mô phỏng nguyên lý hoạt động của một hệ thống quản lý điện năng tích hợp IoT, giúp sinh viên tiếp cận công nghệ điều khiển từ xa và giám sát thông minh.

Mục tiêu chính:

- Giám sát thông số điện năng theo thời gian thực (V, A, W, Hz, PF, kWh).
- Cảnh báo quá tải bằng LED và còi.
- Điều khiển thiết bị phụ tải (Quạt, Đèn) từ xa qua App Flutter.
- Lưu trữ dữ liệu điện năng và tính toán chi phí.
- Bảo mật bằng mã PIN khi thay đổi ngưỡng cảnh báo hoặc reset dữ liệu.

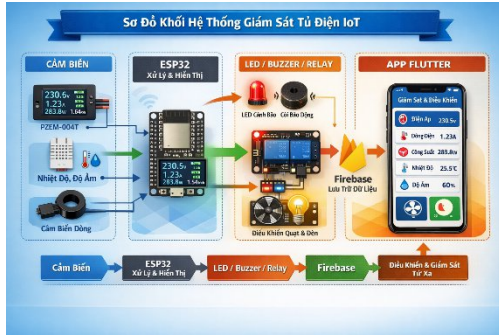
- **ESP32 DevKit V1**: vi điều khiển trung tâm, xử lý dữ liệu và kết nối Internet.
- **PZEM-004T v3.0**: đo điện áp, dòng điện, công suất và điện năng tiêu thụ.
- **Cuộn cảm biến dòng (CT)**: cảm ứng dòng điện cho PZEM.
- **DHT22 (AM2302)**: giám sát nhiệt độ và độ ẩm.
- **Màn hình OLED 0.96" (SSD1306)**: hiển thị thông số tại chỗ.
- **Module Relay 2 kênh**: điều khiển đóng/ngắt Quạt và Đèn.
- **Buzzer**: cảnh báo âm thanh khi vượt ngưỡng.
- **LED (Đỏ, Xanh)**: hiển thị trạng thái hệ thống.
- **Nút nhấn**: điều khiển thủ công, thiết lập ngưỡng và reset dữ liệu.
- **Nguồn Adapter 5V-2A**: cấp nguồn cho ESP32 và các module.
- **Hộp nhựa kỹ thuật**: chứa toàn bộ mạch điện.

2. Sơ đồ khối

Cảm biến → ESP32 → Xử lý & hiển thị → LED/còi/relay → App Flutter qua Firebase.

II. Kiến trúc hệ thống và phần cứng

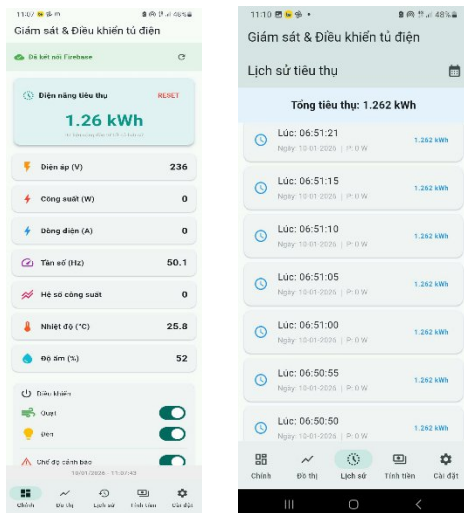
1. Thành phần chính



3. Giao diện App người dùng

Ứng dụng Flutter đóng vai trò là cầu nối giữa người dùng và hệ thống giám sát tủ điện. Giao diện được thiết kế trực quan, dễ sử dụng, gồm các thành phần chính:

- **Dashboard:** hiển thị thông số điện năng (V, A, W, kWh, PF, Hz) và trạng thái môi trường (nhiệt độ, độ ẩm) theo thời gian thực.
- **Biểu đồ (Chart):** vẽ đồ thị lịch sử tiêu thụ điện năng theo ngày, hỗ trợ người dùng theo dõi xu hướng và tính toán chi phí.
- **Điều khiển thiết bị:** các nút bật/tắt Quạt và Đèn, phản hồi gần như tức thì (<500 ms) nhờ Firebase Realtime Database.
- **Thông báo (Notification):** cảnh báo quá tải được đẩy trực tiếp về điện thoại, giúp người dùng xử lý kịp thời.
- **Cài đặt & Bảo mật:** mục thay đổi ngưỡng cảnh báo, reset dữ liệu điện năng, yêu cầu nhập mã PIN để xác thực.
- **Chế độ hiển thị:** hỗ trợ Dark/Light mode, tối ưu trải nghiệm người dùng.



III. Thuật toán điều khiển

1. Nguyên lý hoạt động

- **Thu thập dữ liệu:** PZEM đo thông số điện năng, DHT22 đo nhiệt độ/độ ẩm, ESP32 xử lý và hiển thị OLED.
- **Cảnh báo:** ESP32 so sánh công suất với ngưỡng Alert/Danger.
- **Alert:** LED đỏ nhấp nháy chậm, buzzer kêu ngắt quãng.
- **Danger:** LED đỏ nhấp nháy nhanh, buzzer kêu liên tục.
- **Điều khiển từ xa:** App Flutter gửi lệnh qua Firebase để đóng/ngắt relay.
- **Quản lý dữ liệu:** ESP32 đồng bộ thời gian NTP, lưu dữ liệu điện năng theo ngày.
- **Bảo mật:** xác thực PIN khi thay đổi ngưỡng hoặc reset dữ liệu.

2. Lưu đồ thuật toán

Đọc cảm biến → So sánh ngưỡng → Cảnh báo → Điều khiển relay → Cập nhật Firebase → Hiển thị OLED.



IV. Thực nghiệm và kết quả

A. Môi trường thử nghiệm

- Tủ điện mini với tải giả lập (Quạt, Đèn).
- ESP32 kết nối WiFi, đồng bộ thời gian NTP.
- Firebase Realtime Database làm trung gian dữ liệu.

- App Flutter hiển thị Dashboard và Chart.

B. Kết quả

- Hệ thống đo lường chính xác, hiển thị rõ ràng.
- LED/còi cảnh báo đúng ngưỡng.
- Relay đóng/ngắt thiết bị nhanh chóng qua App.
- Firebase đồng bộ dữ liệu với độ trễ < 500 ms.
- App Flutter cập nhật tự động, giao diện trực quan.



C. Hạn chế

- Quy mô nhỏ, chưa phản ánh thực tế.
- Độ chính xác phụ thuộc cảm biến PZEM và DHT22.
- Chưa tích hợp AI phân tích dữ liệu.

V. Thảo luận

A. Ý nghĩa khoa học

- Mô hình minh họa nguyên lý giám sát và điều khiển điện năng thông minh.
- Tích hợp nhiều công nghệ: cảm biến, vi điều khiển, IoT, Flutter.
- Nền tảng cho nghiên cứu quản lý năng lượng và tối ưu chi phí.

B. Thách thức

- Độ bền cảm biến.
- Nhiễu nguồn khi relay hoạt động.
- Bảo mật khi tích hợp IoT.

C. Ứng dụng thực tế

- Giáo dục: mô hình học tập IoT.
- Nghiên cứu: thử nghiệm thuật toán quản lý điện năng.
- Công nghiệp nhỏ: giám sát và điều khiển từ điện.

VI. Hướng phát triển

- Tích hợp IoT toàn diện (giám sát qua Web/App).
- Lưu dữ liệu lịch sử điện năng.
- Sử dụng nhiều cảm biến để tăng độ chính xác.
- Ứng dụng AI phân tích và dự báo tiêu thụ điện.

VII. Kết luận

Đề tài giám sát từ điện và điều khiển thiết bị bằng App Flutter đã hoàn thành mục tiêu: giám sát điện năng, cảnh báo và điều khiển từ xa. Mô hình có tính ứng dụng cao trong học tập và nghiên cứu, đồng thời mở ra hướng phát triển thành hệ thống quản lý điện năng thông minh trong thực tế.

Tài liệu tham khảo

- [1] Adafruit Industries, “Adafruit ST7735 TFT Display Library,” [Online]. Available: <https://github.com/adafruit/Adafruit-ST7735-Library> (github.com in Bing)
- [2] Espressif Systems, “ESP32 Technical Reference Manual,” [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf (espressif.com in Bing)
- [3] Arduino.cc, “Servo Library,” [Online]. Available: <https://www.arduino.cc/en/Reference/Servo> (arduino.cc in Bing)
- [4] STMicroelectronics, “ST7735S TFT LCD Controller Datasheet,” [Online]. Available: <https://www.st.com/resource/en/datasheet/st7735s.pdf> (st.com in Bing)
- [5] Hackster.io, “Water Level Monitoring using ESP32 and TFT Display,” [Online]. Available: <https://www.hackster.io>

