|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI  **VIỆN ĐIỆN TỬ - VIỄN THÔNG**  logo_128  BÁO CÁO **ĐỒ ÁN TỐT NGHIỆP**  **Đề tài:**  **Thiết kế hệ thống giám sát và đảm bảo chất lượng nước cho bể cá**   |  |  | | --- | --- | | Sinh viên thực hiện: | Hoàng Đức Nghi | | MSSV: | 20172716 | | Lớp: | Điện tử 10 – K62 | | Giảng viên hướng dẫn: | PGS. TS. Trần Quang Vinh |   Hà Nội, 8-2022 |

**ĐÁNH GIÁ QUYỂN ĐỒ ÁN TỐT NGHIỆP**

(Dùng cho giảng viên hướng dẫn)

Tên giảng viên đánh giá:

Họ và tên Sinh viên: MSSV:

Tên đồ án:

**Chọn các mức điểm phù hợp cho sinh viên trình bày theo các tiêu chí dưới đây:**

Rất kém (1); Kém (2); Đạt (3); Giỏi (4); Xuất sắc (5)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Có sự kết hợp giữa lý thuyết và thực hành (20)** | | | | | | |
| 1 | Nêu rõ tính cấp thiết và quan trọng của đề tài, các vấn đề và các giả thuyết (bao gồm mục đích và tính phù hợp) cũng như phạm vi ứng dụng của đồ án | 1 | 2 | 3 | 4 | 5 |
| 2 | Cập nhật kết quả nghiên cứu gần đây nhất (trong nước/quốc tế) | 1 | 2 | 3 | 4 | 5 |
| 3 | Nêu rõ và chi tiết phương pháp nghiên cứu/giải quyết vấn đề | 1 | 2 | 3 | 4 | 5 |
| 4 | Có kết quả mô phỏng/thưc nghiệm và trình bày rõ ràng kết quả đạt được | 1 | 2 | 3 | 4 | 5 |
| **Có khả năng phân tích và đánh giá kết quả (15)** | | | | | | |
| 5 | Kế hoạch làm việc rõ ràng bao gồm mục tiêu và phương pháp thực hiện dựa trên kết quả nghiên cứu lý thuyết một cách có hệ thống | 1 | 2 | 3 | 4 | 5 |
| 6 | Kết quả được trình bày một cách logic và dễ hiểu, tất cả kết quả đều được phân tích và đánh giá thỏa đáng. | 1 | 2 | 3 | 4 | 5 |
| 7 | Trong phần kết luận, tác giả chỉ rõ sự khác biệt (nếu có) giữa kết quả đạt được và mục tiêu ban đầu đề ra đồng thời cung cấp lập luận để đề xuất hướng giải quyết có thể thực hiện trong tương lai. | 1 | 2 | 3 | 4 | 5 |
| **Kỹ năng viết quyển đồ án (10)** | | | | | | |
| 8 | Đồ án trình bày đúng mẫu quy định với cấu trúc các chương logic và đẹp mắt (bảng biểu, hình ảnh rõ ràng, có tiêu đề, được đánh số thứ tự và được giải thích hay đề cập đến trong đồ án, có căn lề, dấu cách sau dấu chấm, dấu phẩy v.v), có mở đầu chương và kết luận chương, có liệt kê tài liệu tham khảo và có trích dẫn đúng quy định | 1 | 2 | 3 | 4 | 5 |
| 9 | Kỹ năng viết xuất sắc (cấu trúc câu chuẩn, văn phong khoa học, lập luận logic và có cơ sở, từ vựng sử dụng phù hợp v.v.) | 1 | 2 | 3 | 4 | 5 |
| **Thành tựu nghiên cứu khoa học (5) (chọn 1 trong 3 trường hợp)** | | | | | | |
| 10a | Có bài báo khoa học được đăng hoặc chấp nhận đăng/đạt giải SVNC khoa học giải 3 cấp Viện trở lên/các giải thưởng khoa học (quốc tế/trong nước) từ giải 3 trở lên/ Có đăng ký bằng phát minh sáng chế | 5 | | | | |
| 10b | Được báo cáo tại hội đồng cấp Viện trong hội nghị sinh viên nghiên cứu khoa học nhưng không đạt giải từ giải 3 trở lên/Đạt giải khuyến khích trong các kỳ thi quốc gia và quốc tế khác về chuyên ngành như TI contest. | 2 | | | | |
| 10c | Không có thành tích về nghiên cứu khoa học | 0 | | | | |
| **Điểm tổng** | | **/50** | | | | |
| **Điểm tổng quy đổi về thang 10** | |  | | | | |

***Nhận xét khác*** *(về thái độ và tinh thần làm việc của sinh viên)*

|  |  |
| --- | --- |
|  | Ngày: … / … / 20…  **Người nhận xét**  (Ký và ghi rõ họ tên) |

**ĐÁNH GIÁ QUYỂN ĐỒ ÁN TỐT NGHIỆP**

(Dùng cho cán bộ phản biện)

Giảng viên đánh giá:

Họ và tên sinh viên: MSSV:

Tên đồ án:

**Chọn các mức điểm phù hợp cho sinh viên trình bày theo các tiêu chí dưới đây:**

Rất kém (1); Kém (2); Đạt (3); Giỏi (4); Xuất sắc (5)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Có sự kết hợp giữa lý thuyết và thực hành (20)** | | | | | | |
| 1 | Nêu rõ tính cấp thiết và quan trọng của đề tài, các vấn đề và các giả thuyết (bao gồm mục đích và tính phù hợp) cũng như phạm vi ứng dụng của đồ án | 1 | 2 | 3 | 4 | 5 |
| 2 | Cập nhật kết quả nghiên cứu gần đây nhất (trong nước/quốc tế) | 1 | 2 | 3 | 4 | 5 |
| 3 | Nêu rõ và chi tiết phương pháp nghiên cứu/giải quyết vấn đề | 1 | 2 | 3 | 4 | 5 |
| 4 | Có kết quả mô phỏng/thưc nghiệm và trình bày rõ ràng kết quả đạt được | 1 | 2 | 3 | 4 | 5 |
| **Có khả năng phân tích và đánh giá kết quả (15)** | | | | | | |
| 5 | Kế hoạch làm việc rõ ràng bao gồm mục tiêu và phương pháp thực hiện dựa trên kết quả nghiên cứu lý thuyết một cách có hệ thống | 1 | 2 | 3 | 4 | 5 |
| 6 | Kết quả được trình bày một cách logic và dễ hiểu, tất cả kết quả đều được phân tích và đánh giá thỏa đáng. | 1 | 2 | 3 | 4 | 5 |
| 7 | Trong phần kết luận, tác giả chỉ rõ sự khác biệt (nếu có) giữa kết quả đạt được và mục tiêu ban đầu đề ra đồng thời cung cấp lập luận để đề xuất hướng giải quyết có thể thực hiện trong tương lai. | 1 | 2 | 3 | 4 | 5 |
| **Kỹ năng viết quyển đồ án (10)** | | | | | | |
| 8 | Đồ án trình bày đúng mẫu quy định với cấu trúc các chương logic và đẹp mắt (bảng biểu, hình ảnh rõ ràng, có tiêu đề, được đánh số thứ tự và được giải thích hay đề cập đến trong đồ án, có căn lề, dấu cách sau dấu chấm, dấu phẩy v.v), có mở đầu chương và kết luận chương, có liệt kê tài liệu tham khảo và có trích dẫn đúng quy định | 1 | 2 | 3 | 4 | 5 |
| 9 | Kỹ năng viết xuất sắc (cấu trúc câu chuẩn, văn phong khoa học, lập luận logic và có cơ sở, từ vựng sử dụng phù hợp v.v.) | 1 | 2 | 3 | 4 | 5 |
| **Thành tựu nghiên cứu khoa học (5) (chọn 1 trong 3 trường hợp)** | | | | | | |
| 10a | Có bài báo khoa học được đăng hoặc chấp nhận đăng/đạt giải SVNC khoa học giải 3 cấp Viện trở lên/các giải thưởng khoa học (quốc tế/trong nước) từ giải 3 trở lên/ Có đăng ký bằng phát minh sáng chế | 5 | | | | |
| 10b | Được báo cáo tại hội đồng cấp Viện trong hội nghị sinh viên nghiên cứu khoa học nhưng không đạt giải từ giải 3 trở lên/Đạt giải khuyến khích trong các kỳ thi quốc gia và quốc tế khác về chuyên ngành như TI contest. | 2 | | | | |
| 10c | Không có thành tích về nghiên cứu khoa học | 0 | | | | |
| **Điểm tổng** | | **/50** | | | | |
| **Điểm tổng quy đổi về thang 10** | |  | | | | |

***Nhận xét khác của cán bộ phản biện***

|  |  |
| --- | --- |
|  | Ngày: … / … / 20…  **Người nhận xét**  (Ký và ghi rõ họ tên) |

**LỜI NÓI ĐẦU**

Ngày nay, sự phát triển vượt bậc của của khoa học công nghệ đã làm cho cuộc sống của con người ngày càng hiện đại. Việc ứng dụng khoa học công nghệ vào cuộc sống hàng ngày đem lại cho chúng ta rất nhiều lợi ích. Chúng ta phải nắm bắt và vận dụng chúng một cách hợp lý, nhằm góp phần vào sự phát triển của nền khoa học kỹ thuật nói chung và ngành kỹ thuật điện tử- viễn thông nói riêng để ứng dụng vào thực tiễn, cải thiện chất lượng cuộc sống.

Với việc sử dụng kit ESP32S có khả năng kết nối Internet, chúng ta có thể giám sát, thu thập dữ liệu và điều khiển các thiết bị khác. Các dữ liệu thông tin này được lưu và kiểm soát trên hoisting để chúng ta có thể theo dõi các thông số và điều khiển các thiết bị qua website hoặc app di động, giúp hệ thống trở nên tự động hoặc bán tự động tuỳ thuộc vào nhu cầu của người dùng

Đồ án có tên “**Thiết kế hệ thống giám sát và đảm bảo chất lượng nước cho bể cá** ” xây dựng một hệ thống giám sát các thông số của bể cá như nồng độ oxi hoà tan, độ PH của nước, nhiệt độ của nước và độ đục của nước thông qua 1 website. Từ các thông số thu được có thể giúp người dùng điều khiển các thiết bị đảm bảo chất lượng nước một cách tự động hoặc bán tự động

Trong thời gian thực hiện đồ án tốt nghiệp, em đã nhận được nhiều sự giúp đỡ tận tình từ thầy cô, bạn bè và gia đình.

Em xin được gửi lời cảm ơn chân thành nhất tới thầy hướng dẫn **PGS. TS. Trần Quang Vinh** đã trực tiếp hướng dẫn, chỉ bảo em về phương pháp cũng như hướng nghiên cứu để em có thể thực hiện thành công đồ án này.

Em xin chân thành cảm ơn!

**LỜI CAM ĐOAN**

Tôi là Hoàng Đức Nghi, mã số sinh viên 20172716, sinh viên lớp Điện tử 10 , khóa 62. Người hướng dẫn là PGS. TS. Trần Quang Vinh. Tôi xin cam đoan toàn bộ nội dung được trình bày trong đồ án “**Thiết kế hệ thống giám sát và đảm bảo chất lượng nước cho bể cá**” là kết quả quá trình tìm hiểu và nghiên cứu của tôi. Các dữ liệu được nêu trong đồ án là hoàn toàn trung thực, phản ánh đúng kết quả đo đạc thực tế. Mọi thông tin trích dẫn đều tuân thủ các quy định về sở hữu trí tuệ; các tài liệu tham khảo được liệt kê rõ ràng. Tôi xin chịu hoàn toàn trách nhiệm với những nội dung được viết trong đồ án này.

|  |  |
| --- | --- |
|  | Hà nội, ngày \_\_ tháng \_\_ năm 2022  **Người cam đoan**  **Hoàng Đức Nghi** |

**MỤC LỤC**

[DANH MỤC KÝ HIỆU VÀ CHỮ VIẾT TẮT i](#_Toc110326971)

[DANH MỤC HÌNH VẼ ii](#_Toc110326972)

[DANH MỤC BẢNG BIỂU v](#_Toc110326973)

[TÓM TẮT ĐỒ ÁN vi](#_Toc110326974)

[ABSTRACT vii](#_Toc110326975)

[CHƯƠNG 1. TỔNG QUAN 1](#_Toc110326976)

[1.1 Đặt vấn đề 1](#_Toc110326977)

[1.2 Phạm vi đề tài 1](#_Toc110326978)

[1.3 Phương pháp nghiên cứu 1](#_Toc110326979)

[1.4 Hướng nghiên cứu 2](#_Toc110326980)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT 3](#_Toc110326981)

[2.1 Kỹ thuật nuôi cá cảnh 3](#_Toc110326982)

[2.1.1 Môi trường nước nuôi cá cảnh 3](#_Toc110326983)

[2.1.2 Nhiệt độ nước 3](#_Toc110326984)

[2.1.3 Nồng độ oxi hoà tan 4](#_Toc110326985)

[2.1.4 Độ PH của nước 4](#_Toc110326986)

[2.1.5 Độ đục của nước 5](#_Toc110326987)

[2.2 Các công nghê kết nối 6](#_Toc110326988)

[2.2.1 WiFi 6](#_Toc110326989)

[2.3 Giao thức TCP/IP 7](#_Toc110326990)

[2.4 Giao thức HTTP 8](#_Toc110326991)

[2.5 Giao tiếp UART 9](#_Toc110326992)

[2.6 Tổng quan về website 10](#_Toc110326993)

[2.6.1 Tổng quan về ngôn ngữ HTML và CSS 10](#_Toc110326994)

[2.6.2 Bố cục cơ bản của HTML trong một trang web 11](#_Toc110326995)

[2.6.3 Các thẻ khác trong HTML 11](#_Toc110326996)

[2.6.4 Cấu trúc và bố cục của CSS 12](#_Toc110326997)

[2.6.5 Các thuộc tính thông dụng của CSS 12](#_Toc110326998)

[2.6.6 Tổng quan về NodeJS 13](#_Toc110326999)

[2.6.7 Ứng dụng của NodeJS trong các dự án IOT 13](#_Toc110327000)

[CHƯƠNG 3. THIẾT KẾ HỆ THỐNG 14](#_Toc110327001)

[3.1 Tổng quan hệ thống 14](#_Toc110327002)

[3.2 Yêu cầu chức năng của phần cứng 15](#_Toc110327003)

[3.3 Yêu cầu chức năng của Web Server 15](#_Toc110327004)

[3.4 Yêu cầu phi chức năng của thiết bị phần cứng 16](#_Toc110327005)

[CHƯƠNG 4. THIẾT KẾ PHẦN CỨNG 17](#_Toc110327006)

[4.1 Khối vi điều khiển 17](#_Toc110327007)

[4.2 Sơ đồ khối tổng quát 17](#_Toc110327008)

[4.3 Thiết kế khối nguồn 18](#_Toc110327009)

[4.3.1 IC LM2596 18](#_Toc110327010)

[4.3.2 Module Buck DC-DC sử dụng IC LM2596 19](#_Toc110327011)

[4.4 Thiết kế khối điều khiển Relay 21](#_Toc110327012)

[4.4.1 Relay SRD12VDC 21](#_Toc110327013)

[4.4.2 Transistor C1815 22](#_Toc110327014)

[4.4.3 Opto 23](#_Toc110327015)

[4.4.4 Led hiển thị trạng thái 23](#_Toc110327016)

[4.4.5 Sơ đồ nguyên lý khối điều khiển Relay 24](#_Toc110327017)

[4.5 Thiết kế khối giao tiếp với người dùng 24](#_Toc110327018)

[4.5.1 Khối nút bấm 24](#_Toc110327019)

[4.6 Thiết kế khối cảm biến và khối thời gian thực 25](#_Toc110327020)

[4.6.1 Khối cảm biến 25](#_Toc110327021)

[4.6.2 Khối thời gian thực 27](#_Toc110327022)

[4.7 Thiết kế layout 28](#_Toc110327023)

[CHƯƠNG 5. THIẾT KẾ FIRMWARE CHO HỆ THỐNG 31](#_Toc110327024)

[5.1 Thiết kế các chức năng ngoại vi 31](#_Toc110327025)

[5.1.1 Các hàm đọc giá trị cảm biến Analog: 31](#_Toc110327026)

[5.1.2 Hàm lấy dữ liệu thời gian thực từ module RTC 31](#_Toc110327027)

[5.1.3 Điều khiển thiết bị 32](#_Toc110327028)

[5.2 Protocols giao tiếp giữa phần cứng và Web Server 33](#_Toc110327029)

[5.2.1 Cấu trúc bản tin giao tiếp giữa Server và thiết bị 33](#_Toc110327030)

[5.2.2 Đồng bộ nút bấm 35](#_Toc110327031)

[5.2.3 Đồng bộ thời gian thực để hẹn giờ thiết bị 36](#_Toc110327032)

[5.2.4 Đồng bộ ngưỡng của thiết bị 36](#_Toc110327033)

[CHƯƠNG 6. THIẾT KẾ WEB SERVER VỚI NODEJS 38](#_Toc110327034)

[6.1 Sơ đồ trạng thái của trang Web 38](#_Toc110327035)

[6.2 Thiết kế cơ sở dữ liệu 38](#_Toc110327036)

[6.3 Thiết kế giao diện Web 38](#_Toc110327037)

[6.4 Deploy Website lên AWS CLOUD 38](#_Toc110327038)

[CHƯƠNG 7. KẾT QUẢ 39](#_Toc110327039)

[7.1 Triển khai 39](#_Toc110327040)

[7.2 Kiểm thử 41](#_Toc110327041)

[7.2.1 Kiểm thử các chức năng phần cứng 41](#_Toc110327042)

[7.2.2 Kiểm thử các chức năng phần mềm 48](#_Toc110327043)

[KẾT LUẬN 53](#_Toc110327044)

[Kết luận chung 53](#_Toc110327045)

[Hướng phát triển 53](#_Toc110327046)

[TÀI LIỆU THAM KHẢO 54](#_Toc110327047)

[PHỤ LỤC 55](#_Toc110327048)

[Phụ lục 1. Mã nguồn triển khai luồng hoạt động của Server trên Node-RED 55](#_Toc110327049)

[Phụ lục 2. Mã nguồn driver giao tiếp với module SIM7600 62](#_Toc110327050)

[Phụ lục 3. Mã nguồn driver giao tiếp WiFi 73](#_Toc110327051)

[Phụ lục 4. Mã nguồn firmware aplication 76](#_Toc110327052)

# DANH MỤC KÝ HIỆU VÀ CHỮ VIẾT TẮT

|  |  |
| --- | --- |
| **Ký hiệu và chữ viết tắt** | **Giải thích** |
| **DO** | Dissolved Oxygen |
| **HTTP** | Hypertext Transfer Protocol |
| **HTTPS** | Hypertext Transfer Protocol Secure |
| **IC** | Intergrated Circuit |
| **IEEE** | Institute of Electrical and Electronics Engineers |
| **IOT** | Internet of Thing |
| **JSON** | Javascript Object Notation |
| **NTU** | Nephelometric Turbidity Units |
| **OPTO** | Optocoupler |
| **PCB** | Printed Circuit Board |
| **RTC** | Real Time Clock |
| **UART** | Universal asynchronous receiver-transmitter |
| **WIFI** | Wireless Fidelity |

# DANH MỤC HÌNH VẼ

[Hình 2.1 Triển khai hệ thống V2X 4](#_Toc75885972)

[Hình 2.2 Kết nối C-V2X 7](#_Toc75885973)

[Hình 2.3 Logo WiFi 8](#_Toc75885974)

[Hình 2.4 4G LTE 10](#_Toc75885975)

[Hình 2.5 Mô hình Client – Server 11](#_Toc75885976)

[Hình 2.6 Mô hình 4 lớp TCP/IP 13](#_Toc75885977)

[Hình 2.7 Sơ đồ đường dây truyền UART giữa 2 thiết bị 14](#_Toc75885978)

[Hình 2.8 Cấu trúc của UART packet 15](#_Toc75885979)

[Hình 2.9 Sơ đồ truyền UART với ký tự ‘P’ 15](#_Toc75885980)

[Hình 2.10 Logo Node-RED 16](#_Toc75885981)

[Hình 2.11 Giao diện làm việc của Node-RED 17](#_Toc75885982)

[Hình 2.12 Các thành phần trên giao diện của Node-RED 17](#_Toc75885983)

[Hình 4.1 Sơ đồ khối thiết bị V2X-Tag 20](#_Toc75885984)

[Hình 4.2 Nguyên lý khối nguồn 3V3-250mA 21](#_Toc75885985)

[Hình 4.3 Nguyên lý khối nguồn 3V8 – 2.5A 22](#_Toc75885986)

[Hình 4.4 Sơ đồ nguyên lý khối dao động sử dụng thạch anh 22](#_Toc75885987)

[Hình 4.5 Sơ đồ nguyên lý bộ lọc thông thấp cho MCU 23](#_Toc75885988)

[Hình 4.6 Sơ đồ nguyên lý khối nạp code và Debug 23](#_Toc75885989)

[Hình 4.7 Sơ đồ nguyên lý khối UART Shift Level 24](#_Toc75885990)

[Hình 4.8 Sơ đồ nguyên lý nút bấm cảm ứng 24](#_Toc75885991)

[Hình 4.9 Sơ đồ nguyên lý khối led trạng thái 25](#_Toc75885992)

[Hình 4.10 Hình ảnh và đặc tính kỹ thuật của module SIM7600CE 26](#_Toc75885993)

[Hình 4.11 Sơ đồ nguyên lý tụ Decoupling cho module LTE 27](#_Toc75885994)

[Hình 4.12 Sơ đồ nguyên lý chân PWRKEY và RESET 27](#_Toc75885995)

[Hình 4.13 Sơ đồ nguyên lý giao tiếp chính của Module LTE 28](#_Toc75885996)

[Hình 4.14 Sơ đồ nguyên lý khối hiển thị trạng thái của module LTE 29](#_Toc75885997)

[Hình 4.15 Sơ đồ nguyên lý phối hợp trở kháng khối RF của module LTE 29](#_Toc75885998)

[Hình 4.16 Top layer của mạch 4G & Power 31](#_Toc75885999)

[Hình 4.17 Bottom layer của mạch 4G & Power 32](#_Toc75886000)

[Hình 4.18 Top layer của mạch MCU & Touch 32](#_Toc75886001)

[Hình 4.19 Bottom layer của mạch MCU & Touch 33](#_Toc75886002)

[Hình 4.20 Hình ảnh sản phẩm V2X-tag khi ghép 2 mạch 34](#_Toc75886003)

[Hình 4.21 Hình ảnh sản phẩm V2X-tag hoàn thiện 35](#_Toc75886004)

[Hình 5.1 Tập lệnh lấy thời gian từ mạng di động 37](#_Toc75886005)

[Hình 5.2 Tập lệnh lấy giữ liệu GPS 38](#_Toc75886006)

[Hình 5.3 Sơ đồ truyền nhận dữ liệu bằng TCP/IP 39](#_Toc75886007)

[Hình 5.4 Cấu trúc gói tin DataPacket (RAW DATA) 40](#_Toc75886008)

[Hình 5.5 Quy trình xử lý dữ liệu gửi/nhận qua Module LTE. 42](#_Toc75886009)

[Hình 5.6 Luồng hoạt động hệ thống V2X-TAG 42](#_Toc75886010)

[Hình 5.7 Các Led báo cho người dùng trên V2X-Tag 43](#_Toc75886011)

[Hình 5.8 Lưu đồ thuật toán giao thức truyền thông V2X – Tag 45](#_Toc75886012)

[Hình 6.1 Giao diện người dùng ở port 1880 49](#_Toc75886013)

[Hình 6.2 TCP in node 52](#_Toc75886014)

[Hình 6.3 TCP out node 52](#_Toc75886015)

[Hình 6.4 Function node 52](#_Toc75886016)

[Hình 6.5 MySQL node 53](#_Toc75886017)

[Hình 6.6 Switch node 53](#_Toc75886018)

[Hình 6.7 Change node 53](#_Toc75886019)

[Hình 6.8 Twilo out node 54](#_Toc75886020)

[Hình 6.9 Sơ đồ luồng hoạt động của Server 55](#_Toc75886021)

[Hình 6.10 Node-RED lắng nghe kết nối, bóc tách bản tin và lưu vào database 55](#_Toc75886022)

[Hình 6.11 Node-RED kiểm tra trạng thái đăng ký của user 56](#_Toc75886023)

[Hình 6.12 Node-RED kiểm tra trạng thái vào ra của xe 56](#_Toc75886024)

[Hình 6.13 Node-RED kiểm tra vị trí của V2X-Tag 56](#_Toc75886025)

[Hình 6.14 Node-RED xác nhận yêu cầu vào ra của V2X-Tag 56](#_Toc75886026)

[Hình 7.1 Trạng thái chờ V2X-Tag khi lấy được vị trí GPS 57](#_Toc75886027)

[Hình 7.2 Bản tin yêu cầu đỗ xe được gửi lên server 57](#_Toc75886028)

[Hình 7.3 Trạng thái Led sau khi gửi thành công yêu cầu đỗ xe 58](#_Toc75886029)

[Hình 7.4 Trạng thái xe đỗ được thể hiện trên trang quản trị 58](#_Toc75886030)

[Hình 7.5 Bản tin yêu cầu lấy xe được gửi lên server 58](#_Toc75886031)

[Hình 7.6 Trạng thái Led báo sau khi gửi yêu cầu lấy xe ra thành công 59](#_Toc75886032)

[Hình 7.7 Trạng thái xe ra được thể hiện trên trang quản trị 59](#_Toc75886033)

[Hình 7.8 Bộ cấp nguồn GW instek GPS-4 kênh đầu ra DC 60](#_Toc75886034)

[Hình 7.9 Dòng điện tiêu thụ thời điểm V2X-Tag đăng ký mạng 61](#_Toc75886035)

[Hình 7.10 Dòng điện tiêu thụ của V2X-Tag trạng thái chờ 62](#_Toc75886036)

[Hình 7.11 Dòng điện tiêu thụ của V2X-Tag khi người dùng tương tác 63](#_Toc75886037)

[Hình 7.12 Các tham số đặc tính S của mạng điện 2 cổng 63](#_Toc75886038)

[Hình 7.13 Máy đo phân tích mạng bằng phương pháp VNA 65](#_Toc75886039)

[Hình 7.14 Biểu đồ hệ số phản xạ S11 với độ cao anten 70cm so với mặt đất 65](#_Toc75886040)

[Hình 7.15 Biểu đồ hệ số phản xạ S11 với độ cao anten 50cm so với mặt đất 66](#_Toc75886041)

[Hình 7.16 Hiện tượng chạm giả trên cảm biến điện dung 67](#_Toc75886042)

[Hình 7.17 nút bấm sau xử lý đã không còn nhiễu 67](#_Toc75886043)

[Hình 7.18 Bản tin gửi từ V2X-Tag lên server 68](#_Toc75886044)

[Hình 7.19 Bản tin ACK được trả từ server xuồng V2X-TAG 68](#_Toc75886045)

[Hình 7.20 Bản tin ACK được trả từ V2X-Tag lên server 68](#_Toc75886046)

[Hình 7.21 Các bản tin đều được ghi lại trong cơ sở dữ liệu 69](#_Toc75886047)

[Hình 7.22 ID của V2X-Tag sẽ được so sánh với ID có trong danh sách đăng ký 69](#_Toc75886048)

[Hình 7.23 Vị trí ô tô sẽ được so sánh với vị trí bãi đỗ xe 70](#_Toc75886049)

[Hình 7.24 Trang quản trị theo dõi trạng thái đỗ xe 70](#_Toc75886050)

# DANH MỤC BẢNG BIỂU

[Bảng 4‑1 Tham số kỹ thuật của khối chuyển đổi 3V3 21](#_Toc75885963)

[Bảng 4‑2 Tham số kỹ thuật của khối nguồn 3V8 – 2.5A 22](#_Toc75885964)

[Bảng 4‑3 PCB Stackup của thiết bị V2X-tag 30](#_Toc75885965)

[Bảng 4‑4 Độ rộng đường tín hiệu RF cần kiểm soát trở kháng 30](#_Toc75885966)

[Bảng 5‑1 Mô tả cấu trúc bản tin trao đổi giữa SERVER và SMGW (LTE Packet) 40](#_Toc75885967)

[Bảng 5‑2 Định nghĩa trường MTY 40](#_Toc75885968)

[Bảng 5‑3 Định nghĩa trường DATA 40](#_Toc75885969)

[Bảng 5‑4 Định nghĩa trường STATE của bản tin gửi từ TAG về SERVER 41](#_Toc75885970)

[Bảng 5‑5 Định nghĩa trường STATE của bản tin gửi từ SERVER về TAG 41](#_Toc75885971)

# TÓM TẮT ĐỒ ÁN

Nuôi cá cảnh là một xu thế đã có từ rất lâu trên thế giới cũng như ở Việt Nam. Tuy nhiên việc nuôi cá cảnh có những khó khăn nhất định, một trong số đó là việc người nuôi không có đủ thời giàn để giám sát và đảm bảo chất lượng nước của bể cá. Với việc áp dụng IoT vào việc nuôi cá cảnh sẽ tự động hoá việc giám sát các thông số vật lý của bể cá và duy trì môi trường của bể cá luôn nằm trong vùng lý tưởng với những thay đổi cần thiết. Người sử dụng cũng có thể nắm bắt thông tin và tự điều chỉnh các thông số thiết bị qua Website

Nội dung trong đồ án gồm 7 chương sau:

*Chương 1: Tổng quan*

*Chương 2: Cơ sở lý thuyết*

*Chương 3: Thiết kế hệ thống*

*Chương 4: Thiết kế phần cứng*

*Chương 5: Thiết kế Driver và Firmware cho phần cứng*

*Chương 6: Thiết kế Website giám sát và điều khiển thiết bị*

*Chương 7: Kết quả*

# ABSTRACT

Raising ornamental fish is a trend that has existed for a long time in the world as well as in Viet Nam. However, this has certain difficulties, one of these difficulties is that the owner does not have enough time to monitor and ensure the walter quality of the aquarium. With the application of IOT in aquarium farming, it will automate the monitoring of the physical parameters of the aquarium and maintain the enviroment of the aquarium always in the ideal zone with the necessary changes. Users can also capture information and self-adjust device parameters via the Websites

The content of the project consists of the following 7 chapters:

*Chapter 1: Overview*

*Chapter 2: Theoretical Basis*

*Chapter 3: System Design*

*Chapter 4: Hardware Design*

*Chapter 5: Designing Drivers and Firmware for Hardware*

*Chapter 6: Designing website for monitoring and controlling devices*

*Chapter 7: Result*

# TỔNG QUAN

## Đặt vấn đề

Cho tới thời điểm này thì thú chơi cá cảnh đã trở nên phổ biến. Tuy nhiên để chất lượng nước trong bể luôn được đảm bảo là điều không dễ dàng, đòi hỏi người nuôi phải có sự am hiểu và kinh nghiệm chăm sóc. Để giảm thiểu tối đa rủi ro khi nuôi cá, người nuôi cần quan tâm tới môi trường nước của bể cá, chế độ ăn của cá.. Những điều đó sẽ giúp cho người nuôi đảm bảo việc chăm sóc bể cá của mình

Với sự phát triển của IOT ngày nay thì việc áp dụng công nghệ này trong việc theo dõi dữ liệu và điều khiển thiết bị là một điều tất yếu. Trong đồ án này em sẽ thực hiện thiết kế một hệ thống giúp theo dõi và đảm bảo các thông số của bể cá một cách tự động hoặc bán tự động. Toàn bộ việc giám sát vào điều khiển sẽ được người dùng thực hiện một cách dễ dàng qua một Website, đây là điểm nổi bật của hệ thống em thiết kế so với đại đa số các hệ thống trên thị trường

## Phạm vi đề tài

Về nội dung nghiên cứu, đề tài “Thiết kế hệ thống giám sát và đảm bảo chất lượng nước cho bể cá” sẽ tập trung vào thiết kế hệ thống cho việc giám sát và đảm bảo các thông số của một bể cá trong nhà. Website điều khiển và giám sát của người dùng sẽ giao tiếp với phần cứng của hệ thống thông qua Wifi

Hệ thống sẽ được chạy thử nghiệm với các bể nuôi cá trong nhà. Các thông số được giám sát là nhiệt độ nước, độ Oxi hoà tan, độ PH và độ đục của nước. Các thiết bị giúp đảm bảo các thông số trên sẽ là đèn cảnh báo PH, máy lọc nước, máy bơm oxi, máy sưởi và máy làm mát nước

## Phương pháp nghiên cứu

*Phương pháp phân tích hệ thống:* là phương pháp tiên quyết trong việc thực hiện thiết kế mọi hệ thống. Qua phương pháp này người thiết kế sẽ có được cái nhìn tổng quan nhất về hệ thống, từ đó phân tích và chọn ra giải pháp phù hợp

*Phương pháp đọc tài liệu hướng dẫn:* là phương pháp mà em cũng rất cần lưu ý. Ở trong đề tài này, em sẽ tìm đọc các tài liệu về datasheet của các linh kiện điện tử, các tài liệu hướng dẫn thiết kế một webserver dựa trên nền tảng NodeJS và các ngưỡng thông số cần đảm bảo về môi trường nước của bể cá

*Phương pháp thu thập số liệu:* là phương pháp thu thập những nguồn thông tin có sẵn từ các nguồn khác nhau. Phương pháp này giúp em tìm hiểu được các hệ thống đã có trên thị trường, từ đó rút ra ưu nhược điểm của các hệ thống đã có để từ đó tối ưu hệ thống của mình

*Phương pháp tham khảo chuyên gia:* là phương pháp hỏi ý kiến từ các thầy cô hướng dẫn, những người đã có kinh nghiệm, chuyên môn cao, giúp cho việc nghiên cứu và thực hiện đề tài trở nên thuận lợi

## Hướng nghiên cứu

* Tìm hiểu datasheet các con chip esp32, bo mạch Arduino Nano, các cảm biến đo thông số môi trường nước để có thể nắm được nguyên lý và cơ chế hoạt động của chúng
* Tìm hiểu quy trình gửi bản tin bằng WiFi (ESP32) thông qua giao thức TCP/IP.
* Thiết kế Firmware cho từng chức năng.
* Thiết kế 1 webserver giao tiếp với thiết bị phần cứng qua chuẩn HTTP
* Kiểm thử trong điều kiện thực tế.

# CƠ SỞ LÝ THUYẾT

## Kỹ thuật nuôi cá cảnh

### Môi trường nước nuôi cá cảnh

Vấn đề nguồn nước để nuôi cá là yếu tố tiên quyết mà người nuôi cần lưu ý. Tuỳ thuộc vào giống cá được nuôi mà phải sử dụng loại nước có thông số phù hợp. Nước nuôi cá có thể là nước ngọt, nước lợ, nước mặn. Với các bể cá trong nhà thì phần lớn môi trường nước là nước ngọt, vì cá nước ngọt dễ nuôi, nguồn nước cũng dễ tìm và dễ đảm bảo hơn. Vì vậy các nội dung tiếp theo sẽ áp dụng cho việc nuôi các loại cá trong môi trường nước ngọt

Hầu hết nước sử dụng trong bể cá hiện nay đều sử dụng nước máy. Vì vậy cẩn phải xử lý chất Clo trước khi sử dụng để nuôi cá. Nên để nước máy trong các bồn, chậu và bồn không có nắp đậy trong hơn 24 giờ để nước máy tự bay hơi hết clo. Để tối ưu, các thùng chứa nước này nên đặt ở vị trí thông thoáng, có đủ ánh nắng mặt trời và. Ngoài ra, người nuôi có thể sử dụng dung dịch khử clo trong nước máy bán tại các cửa hàng. Tuy nhiên, nên hạn chế lạm dụng và chỉ áp dụng khi cần cấp nước gấp hoặc khi không có thời gian dự trữ nước đã khử clo.

Nếu sử dụng nước giếng để nuôi cá thì cần lưu ý rằng nước giếng thường có độ pH thấp 4,5, cũng như oxy thấp, thậm chí một số vùng nước giếng bị nhiễm phèn nặng thì cần phải xử lý thêm. cẩn thận. Để xử lý nước giếng nuôi cá, cần trữ nước giếng trong bể, kết hợp sục khí oxy mạnh để tăng hàm lượng oxy và nâng pH. San hô vụn có thể được thêm vào hộp lọc để tăng độ pH.

Cách xử lý nước giếng khoan nhiễm phèn: Ngoài cách xử lý trên, cần cho than hoạt tính vào bồn chứa nước. Trung bình một lượng than bằng 1/3 thể tích của thùng nước.

Nuôi cá cảnh bằng nước mưa: Nước mưa ngọt sẽ kích thích cá bơi lội vào mùa hè, tuy nhiên nước mưa có độ pH thấp nên cần xử lý như nước thông thường, cần xử lý nước sạch như nước mưa và bổ sung thêm các yếu tố khác. Tuy nhiên, do nước mưa khiến bể cá nhanh mọc tảo nên bạn hạn chế sử dụng.

### Nhiệt độ nước

Nhiệt độ nước là một trong những yếu tố quan trọng nhất khi nuôi cá. Nhiệt độ tối ưu cho cá cảnh thường là 26 - 29 độ C, tùy thuộc vào từng loài cá. Nếu chênh lệch vài độ thì cá vẫn tốt. Nếu sống ở miền Nam, người nuôi không cần quá lo lắng về nhiệt độ, còn nếu ở miền Bắc và các tỉnh lân cận khí hậu lạnh, bà con nên lưu ý sử dụng nhiệt kế để đo nhiệt độ tổng của nước. Máy nước nóng phù hợp.

Nước ấm hơn bình thường cần được làm mát. Ngược lại, nước lạnh hơn bình thường cần được xử lý nhiệt để tránh cá rơi vào trạng thái sốc nhiệt. Sẽ dễ dàng hơn nếu người nông dân sử dụng các thiết bị điện tử. Trong các bể cá thường sử dụng sò làm mát và máy sưởi cho mục đích đảm bảo nhiệt độ của bể

### Nồng độ oxi hoà tan

Bất kể loài vật nào dù sống ở đâu cũng đều không thể thiếu oxi. Cá có thể ngoi lên mặt nước để lấy oxi. Nhưng nhiều loại cá thường nhát khi được nuôi, vì thế chúng rất hạn chế ngoi lên. Thay vào đó, chúng lấy oxi bằng cách lọc oxi có sẵn trong nước qua mang. Với những người nuôi cá thì cách giải quyết của họ là trang bị cho bể cá những bình thổi oxi.

Việc bơm oxi cho cá thì mỗi ngày bơm khoảng vài giờ là đủ cho cá. Không cần bật liên tục 24/24h mỗi ngày

### Độ PH của nước

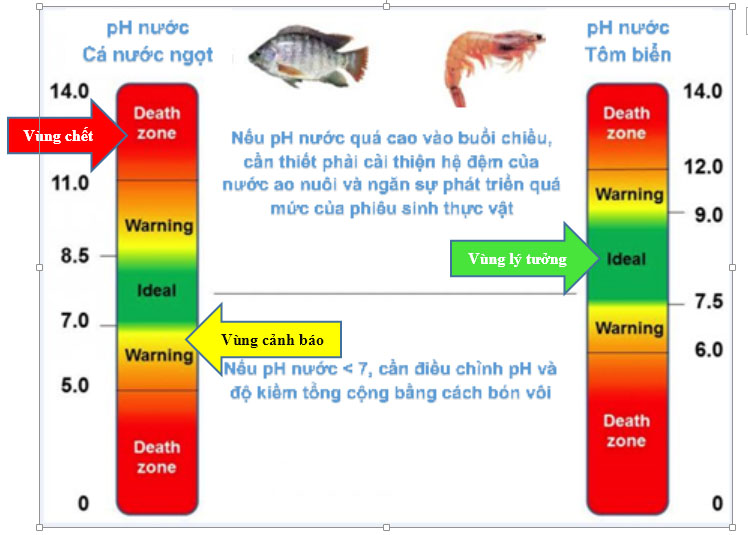
Độ pH của nước có ảnh hưởng quan trọng đến sự phát triển và sinh trưởng của cá. và cây thủy sinh trong bể. Vì vậy, cần duy trì độ pH sao cho ổn định và phù hợp với loại cá đang nuôi

Thông thường, cá sống và phát triển ở độ pH từ 6 đến 8. Tuy nhiên, mỗi loại cá cụ thể lại phát triển tốt nhất ở một độ pH nhất định. Vì vậy, cần tìm hiểu và điều chỉnh độ pH trong bể nuôi sao cho hợp lý

Trong trường hợp độ pH của bể cá là <5.5 thể hiện rằng nồng độ axit trong bể cá cao. Khi sống trong môi trường có tính axit cao có thể sẽ ảnh hưởng đến chất nhầy và hệ hô hấp của cá. Trong trường hợp nghiêm trọng hơn, cá sẽ chết.

Khi độ pH của bể cá> 8,5 thể hiện rằng độ kiềm trong bể cá cao. Khi sống trong môi trường có độ kiềm cao, nó có thể làm tổn thương da và mang. Điều này ảnh hưởng đến quá trình trao đổi oxy và chuyển hóa, kết quả là cá sẽ chậm lớn hơn bình thường.

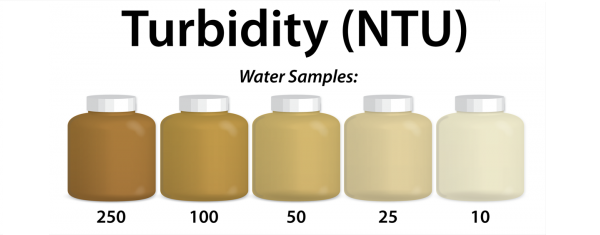
Do việc điều chỉnh PH cần phải thực hiện một cách chậm rãi và khá phức tạp để tránh cá bị chết do sốc PH đột ngột nên rất khó thực hiện tự động bằng các thiết bị, vì vậy trong đồ án này em chỉ sử dụng một đèn LED để cảnh báo cho người dùng khi độ PH nằm ngoài khoảng ngưỡng

****

**Hình 2.1: Các ngưỡng PH trong nuôi tôm cá**

### Độ đục của nước

Độ đục của nước (NTU) là một trong những chỉ số phổ biến và trực quan nhất của nước, nó thể hiện sự vẩn đục của nước mà người quan sát có thể dễ dàng nhận biết thông qua việc quan sát bằng mắt thường.



Hình 2.2 Độ đục của một số mẫu nước

Nước có độ đục cao thường có nhiệt độ cao hơn nước có độ đục thấp. Nguyên nhân là do các hạt lơ lửng trong nước hấp thu nhiệt nhiều hơn làm giảm nồng độ oxy hòa tan (DO).

Độ đục của nước cao cũng làm giảm lượng ánh sáng đi vào nước từ đó hạn chế sự quang hợp và sản xuất DO.

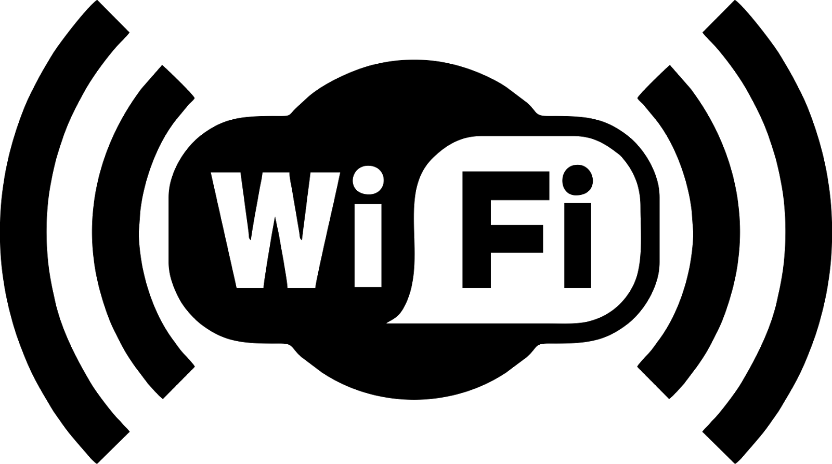
Độ đục cao đồng nghĩa với việc có nhiều chất lơ lửng trong nước, dẫn tới nhiều ảnh hưởng tiêu cực tới sự sinh trưởng và phát triển của các sinh vật sống trong đó: tắc nghẽn mang cá, tốc độ tăng trưởng và sức đề kháng bị suy giảm

Để đảm bảo độ đục của bể luôn phù hợp, các bể cá thường sử dụng máy lọc

## Các công nghê kết nối

### WiFi

WiFi (Wireless Fidelity) là hệ thống hoạt động dựa trên sóng vô tuyến không dây, hay còn được gọi là mạng IEEE.11. WiFi cho phép truy cập internet ở một khoảng cách xác định mà không cần kết nối vật lý. Sóng Wifi ngày nay được sử dụng rộng rãi phục vụ đời sống con người, đặc điểm của sóng Wifi là thu phát ở tần số từ 2.4 GHz đến 5 GHz



Hình 2.3 Logo WiFi

Có 6 chuẩn WiFi thông dụng hiện nay đó là a, b, g, n, ac, ad... Trong đó chuẩn ac được sử dụng rộng rãi nhất trên các thiết bị thông minh

* Chuẩn 802.11b là phiên bản đầu tiên trên thị trường. Đây là chuẩn chậm nhất và rẻ tiền nhất, vì thế trở nên ít phổ biến hơn so với các chuẩn khác. 802.11b phát tín hiệu ở tần số 2.4 GHz, có tốc độ xử lý lên đến 11 megabit/giây, sử dụng mã CCK (complimentary code keying).
* Chuẩn 802.11g cũng phát ở tần số 2.4 GHz, nhưng nhanh hơn so với chuẩn 802.11b, tốc độ xử lý lên tới 54 megabit/giây. Chuẩn 802.11g nhanh hơn vì nó sử dụng mã OFDM (orthogonal frequency-division multiplexing), một công nghệ mã hóa hiệu quả hơn.
* Chuẩn 802.11a phát ở tần số 5 GHz và có thể đạt đến 54 megabit/ giây. Nó cũng sử dụng mã OFDM. Những chuẩn mới hơn sau này như 802.11n còn nhanh hơn chuẩn 802.11a, nhưng 802.11n vẫn chưa phải là chuẩn cuối cùng.
* Chuẩn 802.11n cũng phát ở tần số 2.4 GHz, nhưng nhanh hơn so với chuẩn 802.11a, tốc độ truyền dữ liệu tối đa đạt 450 megabit/giây.
* Chuẩn 802.11ac phát ở tần số 5 GHz nhanh hơn so với chuẩn 802.11n, tốc độ truyền dữ liệu tối đa đạt đến 1.3 Gigabit/giây.
* Chuẩn 802.11ad phát ở tần số 60 GHz nhanh hơn so với chuẩn 802.11ac, tốc độ truyền dữ liệu tối đa đạt đến 4,6 Gigabit/giây.

WiFi có thể hoạt động trên cả ba tần số và có thể thay đổi qua lại giữa các tần số một cách nhanh chóng. Việc này giúp giảm thiểu sự nhiễu sóng và cho phép nhiều thiết bị kết nối không dây cùng một lúc.

## Giao thức TCP/IP

TCP/IP hoặc Transmission Control Protocol/Internet Protocol (Giao thức điều khiển truyền vận/giao thức mạng) là một bộ các giao thức trao đổi thông tin được sử dụng để kết nối các thiết bị mạng trên Internet. TCP/IP có thể được sử dụng như là một giao thức trao đổi thông tin trong một mạng riêng (intranet hoặc extranet)

TCP/IP là bộ giao thức cho phép kết nối các hệ thống mạng không đồng nhất với nhau. Ngày nay TCP/IP được dùng rộng rãi trong mạng cục bộ cũng như mạng toàn cầu. TCP/IP được xem như giản lược của mô hình tham chiếu OSI với 4 tầng như sau:

- Tầng liên kết (Datalink layer)

- Tầng mạng (Internet layer).

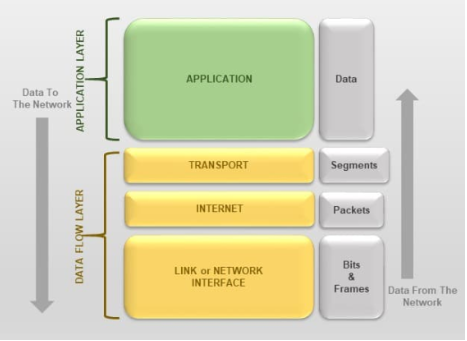
- Tầng giao vận (Transport layer).

- Tầng ứng dụng (Application layer).

Phương thức hoạt động của bộ giao thức TCP/IP:

Khi truyền dữ liệu, quá trình tiến hành từ tầng trên xuống tầng dưới (tầng ứng dụng truyền xuống tầng kiên kết), qua mỗi tầng dữ liệu được thêm vào thông tin điều khiển được gọi là Header.

Khi nhận dữ liệu thì quá trình này sẽ xảy ra ngược lại. Dữ liệu được truyền từ tầng dưới lên tầng trên (tầng liên kết truyền lên tầng ứng dụng) và qua mỗi tầng thì phần Header tương ứng sẽ được lấy đi và khi đến tầng cuối cùng thì dữ liệu không còn phần Header nữa.



Hình 2.4 Mô hình 4 lớp TCP/IP

TCP/IP có khả năng mở rộng cao và như một giao thức có thể định tuyến, nó có thể xác định đường dẫn hiệu quả nhất thông qua mạng.

## Giao thức HTTP

HTTP (Hyper Text Transfer Protocol) – Giao thức truyền siêu văn bản . Giao thức này nằm trong tầng Appication của giao thức TCP/IP, được sử dụng để truyền nội dung trang Web từ Web Server đến trình duyệt Web ở Client, là giao thức Client-Server phổ biến dùng trong Internet. Cơ chế hoạt động chính của HTTP Request-Response. Client sẽ gửi Request tới Server và nhận Response từ Server trả về

Các phương thức Request quen thuộc trong Http gồm có : GET, POST, PUT,DELETE, HEAD,CONNECT

Các trạng thái Response của Server trả về:

-1xx: Informal

-2xx- Success (200-OK,202-Accepted, 204-No content)

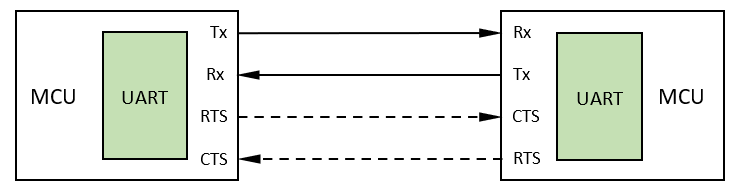
-3xx - Redirection (301 Moved Permanently: tài nguyên đã được chuyển hoàn toàn tới địa chỉ Location trong HTTP response. 303 See other: tài nguyên đã được chuyển tạm thời tới địa chỉ Location trong HTTP response. 304 Not Modified: tài nguyên không thay đổi từ lần cuối client request, nên client có thể sử dụng đã lưu trong cache.)

-4xx - Client Error (400 - Bad Request, 401 - Unauthorized, 403 - Forbidden, 404 - Not Found, 405 - Method Not Allowed, 408 - Request Timeout, 429 - Too many requests)

-5xx - Server Error (500 - Internal Server Error, 503 - Service Unavailable, 504 - Gateway Timeout, 509 - Bandwidth Limit Exceed)

## Giao tiếp UART

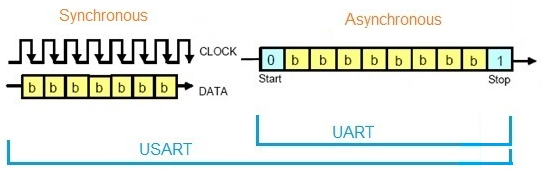
UART (Universal Asynchronous Receiver / Transmitter) là một trong những giao tiếp phổ biến của vi điều khiển. UART được gọi là truyền thông không đồng bộ. Chức năng chính của UART là truyền dữ liệu nối tiếp. Trong UART, việc truyền nhận dữ liệu giữa 2 thiết bị có 2 cách thức hiện là giao tiếp dữ liệu nối tiếp và giao tiếp dữ liệu song song.



Hình 2.5 Sơ đồ đường dây truyền UART giữa 2 thiết bị

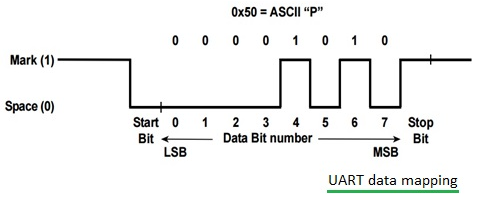
Các đặc trưng của giao tiếp UART:

* Hỗ trợ truyền tốc độ dữ liệu thấp
* UART là giao thức đơn giản, nó sử dụng start bit (trước các bit data), stop bit(một hoặc hai, sau các bit data), bit chẵn lẻ (chẵn hoặc lẻ) ở định dạng cơ sở của nó để định dạng dữ liệu. Bit chẵn lẻ giúp phát hiện lỗi một bit. Tức là UART packet = 1 bit start bit (mức thấp) + 8 bit dữ liệu gồm cả bit chẵn lẻ + 1 hoặc 2 stop bit (mức cao)



Hình 2.6 Cấu trúc của UART packet

* Dữ liệu được truyền từng byte.
* UART tạo xung nhịp bên trong và đồng bộ hóa nó với luồng dữ liệu với sự trợ giúp của quá trình chuyển đổi bit bắt đầu.
* Đối với giao tiếp khoảng cách xa, 5V UART được chuyển đổi thành điện áp cao hơn viz. + 12V cho mức logic 0 và -12V cho mức logic 1.



Hình 2.7 Sơ đồ truyền UART với ký tự ‘P’

## NTP Server

NTP là viết tắt của Network Time Protocol. Đó là Giao thức Internet tiêu chuẩn để đồng bộ hóa đồng hồ máy tính với một số tham chiếu qua mạng.

Giao thức có thể được sử dụng để đồng bộ hóa tất cả các thiết bị được nối mạng với Giờ phối hợp quốc tế (UTC) trong vòng vài mili giây (50 mili giây trên Internet công cộng và dưới 5 mili giây trong môi trường mạng LAN).

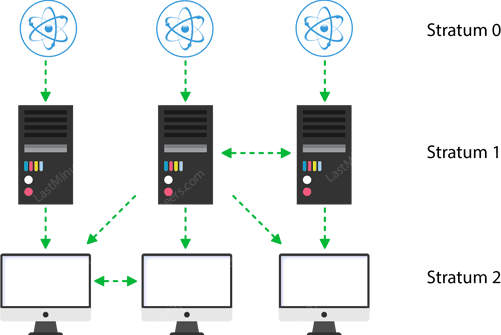
NTP đặt đồng hồ của máy tính thành UTC, bất kỳ độ lệch múi giờ địa phương hoặc thời gian tiết kiệm ánh sáng ngày nào cũng được ứng dụng khách áp dụng. Theo cách này, các máy khách có thể đồng bộ hóa với máy chủ bất kể sự khác biệt về vị trí và múi giờ.

### Kiến trúc của NTP

NTP sử dụng kiến ​​trúc phân cấp. Mỗi cấp trong hệ thống phân cấp được gọi là một tầng

Trên cùng là các thiết bị đo thời gian có độ chính xác cao, chẳng hạn như đồng hồ nguyên tử, GPS hoặc đồng hồ vô tuyến, được gọi là đồng hồ phần cứng tầng 0.

Máy chủ tầng 1 có kết nối trực tiếp với đồng hồ phần cứng tầng 0 và do đó có thời gian chính xác nhất.



**Hình 2.8: Kiến trúc NTP**

Mỗi tầng trong hệ thống phân cấp đồng bộ hóa với tầng trên và hoạt động như máy chủ cho các máy tính ở tầng thấp hơn.

### Giao tiếp giữa NTP và vi điều khiển ESP32

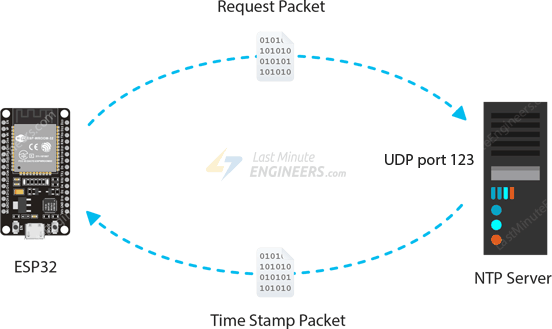
NTP có thể hoạt động theo một số cách. Cấu hình phổ biến nhất là hoạt động ở chế độ máy Server-Client Nguyên lý làm việc cơ bản như sau:

Client như ESP32 kết nối với Server bằng Giao thức dữ liệu người dùng (UDP) trên cổng 123.

Sau đó, một client sẽ truyền một gói yêu cầu (request) đến một Server NTP.

NTP server sau đó sẽ gửi một phản hồi (Response) là một gói thời gian, bao gồm các thông tin như mốc thời gian UNIX, độ trễ, múi giờ

Sau đó, Client có thể phân tích gói tin vừa được phản hồi về để thu được thời gian hiện tại



**Hình 2.9: Minh hoạ giao tiếp giữa ESP32 và NTP Server**

## Tổng quan về website

### Tổng quan về ngôn ngữ HTML và CSS

HTML (tiếng Anh, viết tắt cho HyperText Markup Language, hay là "Ngôn ngữ Đánh dấu Siêu văn bản") là một ngôn ngữ đánh dấu được thiết kế ra để tạo nênn các trang web với các mẩu thông tin được trình bày trên World Wide Web

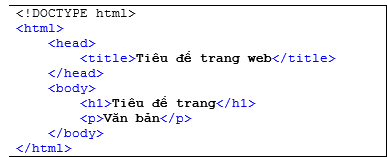
CSS (tiếng Anh, viết tắt cho Cascadding Style Sheets): là ngôn ngữ hỗ trợ cho việc hiển thị một tài liệu HTML. Đây là một ngôn ngữ đặc biệt hữu ích trong việc thiết kế Web, Nếu như HTML có vai trò giúp định dạng các phần tử trên website thì CSS đóng vai trò sẽ tạo ra các style cho các phần tử đó như bố cục, màu sắc, kích thước …



**Hình 2.10: Hiệu ứng khi có CSS trong HTML**

### Bố cục cơ bản của HTML trong một trang web

Bố cục cơ bản của một file HTML (file có phần mở rộng “.html”) trong một Website gồm có



**Hình 2.11: Bố cục cơ bản của HTML trong web**

Trong đó:

<!DOCTYPE>:Thẻ khai báo định dạng cho 1 file HTML

<html> và </html> : Có vai trò mở và đóng nội dung của trang web lại. Đây là một thẻ bắt buộc

<head> và </head>: Khai báo thông tin của trang web

<title> và <title/> : Khai báo tiêu đề của trang Web

<body> và </body>: Là 2 thẻ quan trọng nhất trong một file HTML, chứa những thẻ con giúp hiển thị nội dung của website như thẻ <h1> </h1> hay thẻ <p></p>

### Các thẻ khác trong HTML

Ngoài bố cục đã nêu ở trên thì HTML còn một số thẻ thông dụng khác như:

**Bảng 2.1: Các thẻ thông dụng trong HTML**

|  |  |
| --- | --- |
| Thẻ | Mô tả |
| <div> và <*/*div> | Được sử dụng để định nghĩa một khu vực trong tài liệu của bạn. Khi sử dụng thẻ <div> bạn có thể nhóm các khu vực lớn của các phần tử HTML với nhau và định dạng chúng với CSS. |
| <ul> và <*/* ul> | Xác định danh sách không thứ tự. |
| <li> và <*/* li> | Xác định danh sách item. |
| <a> và <*/* a> | Xác định một liên kết. |
| <button> và <*/* button > | Xác định một nút nhấn. |
| <p> và <*/* p> | Xác định một đoạn văn bản. |

### Cấu trúc và bố cục của CSS

Một đoạn CSS có cấu trúc như sau:

Vùng chọn {

Thuộc tính: giá trị,

Thuộc tính: giá trị,…..

}

Một đoạn CSS sẽ được khai báo bằng vùng chọn, các thuộc tính của vùng chọn nắm bên trong cặp dấu ngoặc nhọn “{}”. Mỗi thuộc tính luôn có một giá trị riêng, có thể là dạng số, hoặc có thể là các tên đã được định nghĩa sẵn của CSS. Phần thuộc tính và giá trị của thuộc tính phải được cách nhau bởi dấu hai chấm “:”, sử dụng dấu chấm phẩy “;” để kết thúc khai báo một thuộc tính

### Các thuộc tính thông dụng của CSS

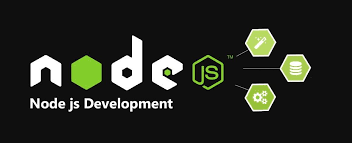
CSS có rất nhiều thuộc tính, bảng dưới đây là các thuộc tính phổ biến nhất của CSS trong thiết kế web

**Bảng 2.2: Các thuộc tính CSS cơ bản**

|  |  |  |
| --- | --- | --- |
| **Thuộc** **tính** | **Ví** **dụ** | **Mô** **tả** |
| [background](http://hocwebchuan.com/reference/cssSection/pr_bg.php) | background: #ff0000; | Định dạng nền (background) cho thành phần. |
| [border](http://hocwebchuan.com/reference/cssSection/pr_bdr.php) | border: 1px solid #ff0000; | Định dạng đường viền cho thành phần. |
| [bottom](http://hocwebchuan.com/reference/cssSection/pr_bottom.php) | bottom: 10px; | Xác định vị trí dưới cùng của thành phần được định vị trí. |
| [clear](http://hocwebchuan.com/reference/cssSection/pr_clear.php) | clear: both; | Xác định 2 bên của phần tử (left, right), nơi mà phần tử float không được cho phép. |
| [color](http://hocwebchuan.com/reference/cssSection/pr_color.php) | color: #ff0000; | Xác định màu sắc cho text. |
| [content](http://hocwebchuan.com/reference/cssSection/pr_content.php) | content: "." | Sử dụng kèm với bộ chọn ":before", ":after" để chèn nội dung được tạo. |
| [display](http://hocwebchuan.com/reference/cssSection/pr_display.php) | display: inline; | Xác định loại hiển thị của thành phần. |
| [float](http://hocwebchuan.com/reference/cssSection/pr_float.php) | float: left; | Xác định có hay không một thành phần được float. |
| [font](http://hocwebchuan.com/reference/cssSection/pr_font.php) | font:12px arial,sans- | Thiết lập font cho thành phần, bao gồm font |

### Tổng quan về NodeJS

NodeJS là một nền tảng được xây dựng trên V8 JavaScript Engine – là một framework giúp hực thi mã JavaScript. Framework này giúp xây dựng các ứng dụng web một cách đơn giản và dễ dàng mở rộng.



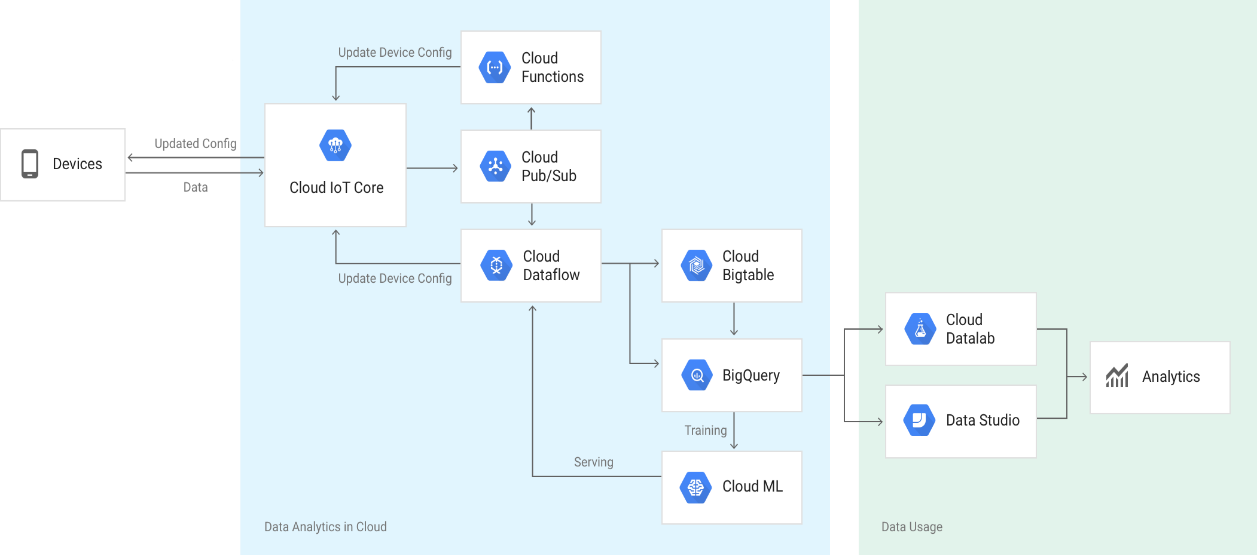
**Hình 2.12: Logo NodeJS**

Ưu điểm của NodeJS:

* Cộng đồng hỗ trợ ngôn ngữ Javascript rất lớn
* Tốc độ xử lý ngôn ngữ nhanh nhờ cơ chế bất đồng bộ
* Dễ dàng mở rộng tính năng

### Ứng dụng của NodeJS trong các dự án IOT

Một hệ thống Internet Of Things đầy đủ khá phức tạp. Bao gồm: thiết bị, Server xử lý kết nối, Server dữ liệu (Database), kèm theo đó là các hệ thống cân bằng tải, các hệ thống phân thích, báo cáo dữ liệu, …



**Hình 2.13: Mô hình IOT của Google IOT Core**

NodeJS hỗ trợ rất nhiều cho việc xây dựng Server nhờ những ưu điểm của nó và số lượng package hỗ trợ rất đa dạng. Server có thể xây dựng bẳng chuẩn Socket I/O hoặc HTTP, chạy được trên nhiều hệ điều hành khác nhau

# THIẾT KẾ HỆ THỐNG

## Tổng quan hệ thống

Hệ thống giám sát bể cá gồm các phân hệ sau:

* Phân hệ cảm biến: Phân hệ này bao gồm các cảm biến Analog đo các thông số của bể cá như: Nhiệt độ, độ oxi hoà tan, độ PH, độ đục. Các thông số cảm biến này sẽ được gửi từ thiết bị lên Server qua Wifi với chuẩn giao tiếp TCP/IP
* Phân hệ Server BackEnd: Phân hệ này sẽ xử lý các dữ liệu được gửi lên từ các module cảm biến
* Phân hệ vận hành, điều khiển: Phân hệ này có chức năng giám sát, điều khiển, cấu hình thiết bị từ xa.

## Yêu cầu chức năng của phần cứng

Phần cứng của hệ thống sẽ bao gồm 2 chức năng chính: Giám sát thông số và điều khiển thiết bị

Các thông số thu được từ các cảm biến sẽ được gửi lên server thông qua module truyền thông Wifi và hiển thị trên một giao diện Web

Các thiết bị đảm bảo thông số của bể cá gồm: Đèn cảnh báo PH, máy lọc nước, máy bơm oxi, máy làm mát, máy sưởi. Các thiết bị đó sẽ được nối vào các Relay để thực hiện bật tắt. Việc thực hiện bật tắt sẽ được thực hiện bằng nút bấm trên phần cứng hoặc qua webserver

Trên mạch điện sẽ có các điền hiển thị trạng thái của các thiết bị, khi thiết bị tắt thì đèn tắt, khi thiết bị được bật thì đèn sáng

Ngoài chế độ điều khiển bằng tay, phần cứng còn hỗ trợ chức năng điều khiển tự động dựa trên giá trị cảm biến thu được

## Yêu cầu chức năng của Web Server

Server sẽ nhận các bản tin từ mạch phần cứng, lưu vào cơ sở dữ liệu, xử lý bản tin và trả lại phản hồi cho mạch ứng với các trường hợp tương ứng

Server hỗ trợ giao thức HTTP để kết nối với thiết bị phần cứng

Các dữ liệu trên server có thể truy cập bởi các ứng dụng người dùng được phép. Bên thứ ba có thể dùng dữ liệu đó để phân tích, thiết kế và phát triển các ứng dụng liên quan.

Các tính năng cụ thể của trang Web bao gồm:

-Hiển thị các thông số cảm biến ở thời điểm mới nhất

-Hiển thị các đồ thị thông số cảm biến trong ngày và trong tuần

-Điều khiển thiết bị thủ công và tự động

-Hẹn giờ bật tắt thiết bị

## Yêu cầu phi chức năng của thiết bị phần cứng

Mạch điện sử dụng nguồn điện DC 12V , kích thước nhỏ gọn, tiết kiệm điện năng. Ngoài ra cần thiết bị một vỏ hộp giúp bảo quản được mạch và có thể dễ dàng lắp đặt trong môi trường bể cá

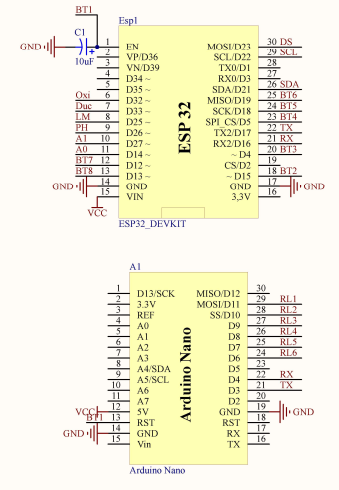
# THIẾT KẾ PHẦN CỨNG

## Khối vi điều khiển

Đầy là khối có vai trò thu thập các dữ liệu từ cảm biến, truyền các tín hiệu tới module thời gian thực, các nút bấm, các relay và giao tiếp với Webserver thông qua wifi

Các linh kiện chính trong khối gồm có : Kit ESP32 và bo mạch Arduino Nano, hoạt động khi được cấp nguồn 5V

Sơ đồ nguyên lý của khối:

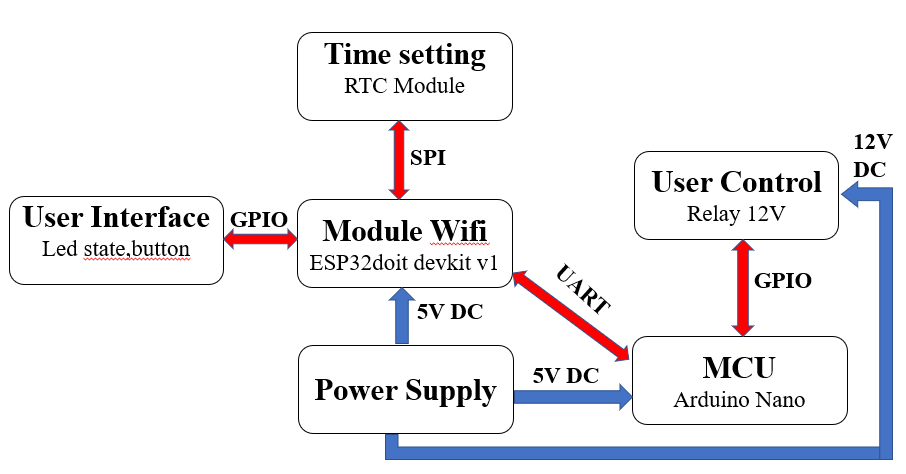


**Hình 4.17: Sơ đồ nguyên lý khối vi điều khiển**

## Sơ đồ khối tổng quát

Sơ đồ thiết kế tổng quát thiết bị giám sát và đảm bảo chất lượng nước bể cá gồm các khối chính sau

* *Khối nguồn*: Chuyển đổi nguồn cung cấp từ kết nối USB thành các mức điện áp khác nhau để duy trì hoạt động của toàn mạch.
* *Khối xử lý*: Khối xử lý sử dụng kit ESP32 có nhiệm vụ điều khiển tiếp nhận dữ liệu từ các cảm biến , đóng gói dữ liệu. Đồng thời cũng đóng vai trò làm module truyền thông qua WiFi.
* *Khối thực thi*: Khối thực thi sử dụng bo mạch Arduino Nano có nhiệm vụ là nhận các packet từ khối xử lý đề điểu khiển các thiết bị đảm bảo chất lượng nước bể cá
* *Khối tương tác với người dùng*: Khối này gồm nút nhấn cảm ứng và đèn LED hiển thị trạng thái cảnh báo, yêu cầu xác nhận và tương tác với người dùng.
* *Khối thời gian thực* : Sử dụng module thời gian thực RTC phục vụ cho chức năng hẹn giờ



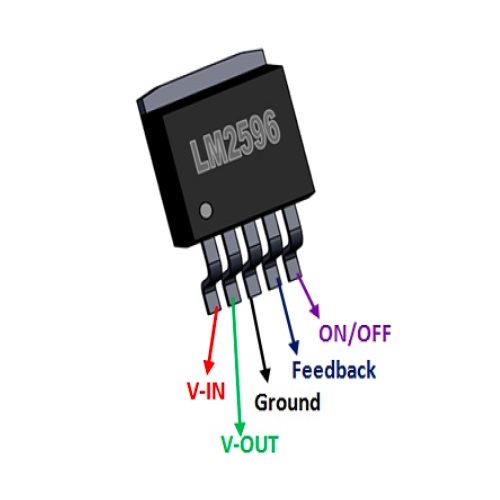
Hình . Sơ đồ khối của mạch phần cứng

## Thiết kế khối nguồn

Khối nguồn của thiết bị thực hiện chức năng chuyển đổi nguồn 12V DC sang các mức điện áp thấp hơn là 5V. Chính vì vậy ở khối này em sẽ sử dụng module hạ áp LM2596 để hạ điện áp đầu vào từ 12V về 5V

### IC LM2596

LM2596 là một IC hạ áp được sử dụng phổ biến. IC LM2596 có thể nhận điện áp đầu vào từ 4,5V đến 40V và chuyển đổi nó thành nguồn điện áp thay đổi với dòng điện liên tục lên tới 3A. IC này thường được sử dụng trong các module nguồn để cấp nguồn hoặc điều khiển tải.



Hình 4.2 Sơ đồ các chân của IC LM2596

Bảng 4.1 Mô tả các chân của IC LM296

|  |  |  |
| --- | --- | --- |
| **Số chân** | **Tên chân** | **Mô tả** |
| 1 | V-IN | Điện áp đầu vào của IC |
| 2 | V-OUT | Điện áp đầu ra của IC |
| 3 | GROUND | Nối đất |
| 4 | FEEDBACK | Điện áp hồi tiếp của IC |
| 5 | ON/OFF | Chân kích hoạt, phải được nối đất để IC hoạt động bình thường |

**Bảng 4.2: Đặc tính kỹ thuật của IC LM2596**

|  |  |
| --- | --- |
| **Thông số** | **Chi tiết** |
| Điện áp đầu vào | 4.5V tới 40V |
| Điện áp đầu ra tối thiểu | 3.16V |
| Dòng điện đầu ra liên tục | 3A |
| Dòng đầu ra đỉnh | 6.9A |
| Tần số chuyển mạch | 150KHz |

### Module Buck DC-DC sử dụng IC LM2596

Nhiệm vụ chính của khối hạ áp này là để cung cấp điện áp đầu ra 5V từ đầu vào 12 V cho các Kit ESP32 và Arduino Nano



Hình 4.3 Sơ đồ nguyên lý khối hạ áp 12V-5V

Nguyên lý hoạt động của module hạ áp:

Điện áp đầu vào sau khi được lọc qua tụ C1 và C2 sẽ đi vào chân Vin. Khi dòng điện đi vào chân Vin thì IC sẽ hoạt động ở nửa chu kỳ mở, cho dòng điện đi qua chân Output. Dòng điện đi ra khỏi chân Output rồi qua cuộn cảm và tụ lọc nhiễu C3 C4 để thu được điện áp đầu ra Vout. Khi Vout đạt 5V thì điện áp Feedback cũng trả về 5V để IC thực hiện chu kỳ đóng.

Khi dòng chảy từ chân Vin sang chân output sẽ sinh ra từ trường trên cuộn cảm L1. Khi dòng chảy giảm dần thì điện áp trên cuộn cảm sẽ đảo chiều. Điện áp trên cuộn cảm lúc này sẽ đi qua cả Diode D1 giúp kín mạch



**HÌnh 4.4: Hình ảnh mạch hạ áp sử dụng IC LM2596**

Bảng 4.3: Tham số kỹ thuật của mạch hạ áp 12V-5V

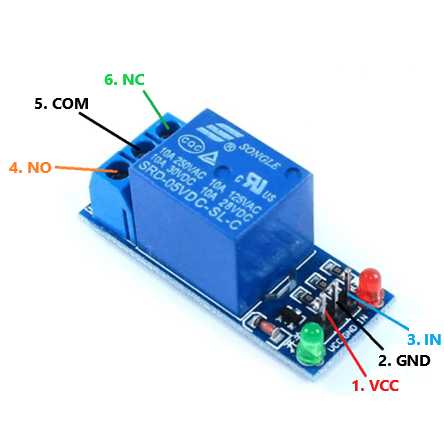
|  |  |
| --- | --- |
| **Thông số** | **Chi tiết** |
| Điện áp đầu vào | 3V-30V |
| Điện áp đầu ra | 1.5V-30V |
| Dòng đáp ứng tối đa | 3A |
| Hiệu suất | 92% |
| Công suất | 15W |
| Kích thước | 45x12x14 (mm) |

## Thiết kế khối điều khiển Relay

Khối điều khiển là một bộ phận quan trọng, giúp điều khiển các thiết bị đảm bảo chất lượng nước bể cá. Khối điều khiển Relay gồm những thành phần chính sau

### Relay SRD12VDC

Relay 12VDC nằm trung gian giữa nút bấm điều khiển và thiết bị cần điều khiển



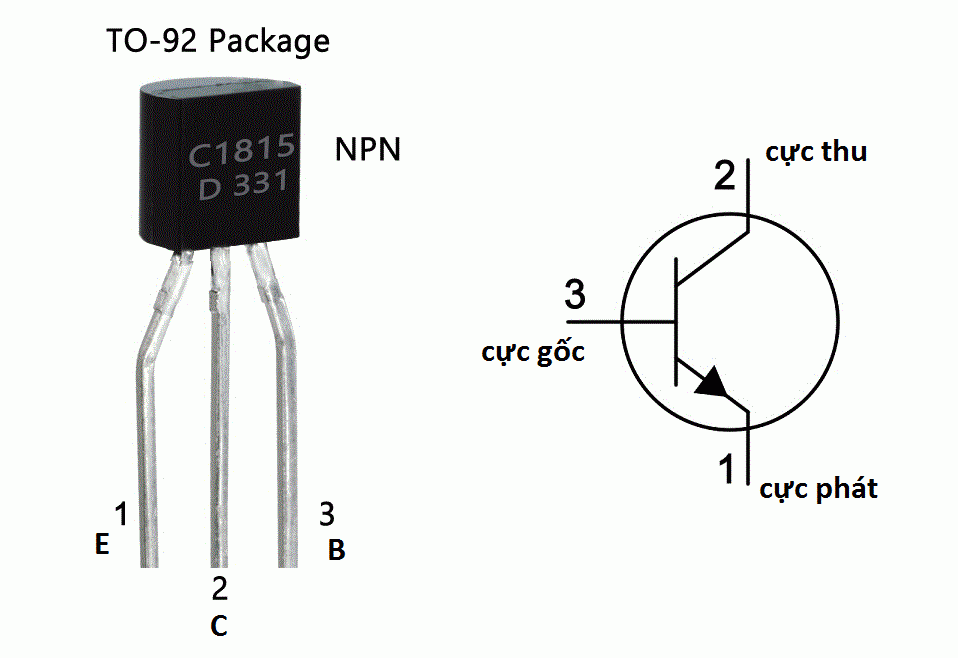
Hình 4.4 Sơ đồ các chân của Relay

**Bảng 4.4: Mô tả các chân của Relay**

|  |  |
| --- | --- |
| **Tên chân** | **Mô tả** |
| VCC | Cấp nguồn đầu vào cho mạch điều khiển |
| GND | Chân nối đất |
| IN | Nhận tín hiệu ngõ vào |
| NO | Tiếp điểm ngõ ra thường hở của Relay |
| COM | Tiếp điểm chung của Relay |
| NC | Tiếp điểm ngõ ra thường đóng của Relay |

### Transistor C1815

Transistor C1815 là transistor loại NPN, dùng để chuyển tín hiện điện, xung hoặc tín hiệu khuếch đại.



Hình 4.5 Sơ đồ chân của transistor C1815

Ba cực trên transistor gồm:

-Cực phát: Có vai trò là đầu ra cho nguồn

-Cực gốc: Hoạt động như cổng điều khiển cho đầu vào điện lớn hơn tại cực thu

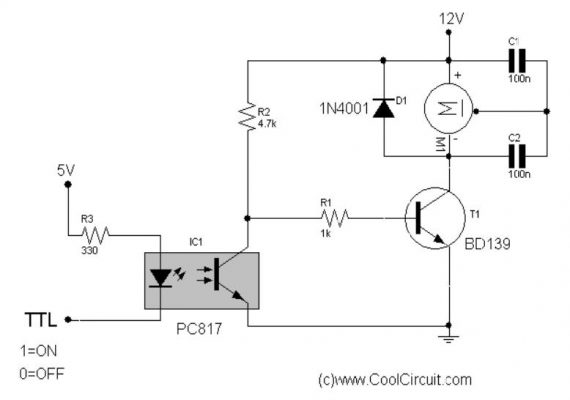
-Cực thu: Có vai trò thu thập năng lượng

**Bảng 4.5: Thông số kỹ thuật của Transistor C1815**

|  |  |
| --- | --- |
| **Thông số** | **Chi tiết** |
| Loại chân | TO-92 |
| Dòng điện tối đa (Ic) | 150mA |
| Điện áp cực đại C-E | 50V |
| Điện áp cực đại C-B | 60V |
| Tần số chuyển mạch tối đa | 80Mhz |
| Hệ số khuếch đại dòng | 70-700 |

### Opto

**Opto (optocoupler)** còn gọi là bộ **Opto cách ly quang**  là một linh kiện dùng để chuyển tín hiệu điện sang ánh sáng và sau đó mới truyền đi.



Hình 4.6 Sơ đồ cấu tạo của Opto

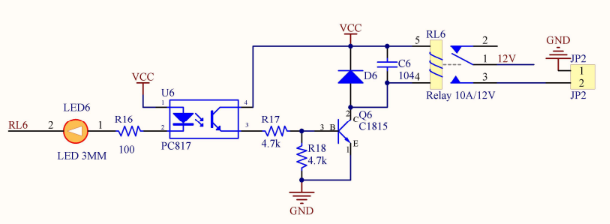
Khi có dòng nhỏ đi qua 2 đầu của led có trong opto làm cho LED phát sáng. Khi LED phát sáng làm thông hai cực của photo transitor hay photo diot mở cho dòng điện chạy qua.

### Led hiển thị trạng thái

Các Led hiển thị trạng thái sẽ thể hiện trạng thái bật tắt thiết bị, khi thiết bị bật thì đèn Led sẽ sáng, khi thiết bị tắt thì đèn Led sẽ tắt

### Sơ đồ nguyên lý khối điều khiển Relay

Sau khi lựa chọn các linh kiện đã nêu ở trên, em sẽ thiết kế sơ đồ nguyên lý của khối điều khiển Relay 12V như sau



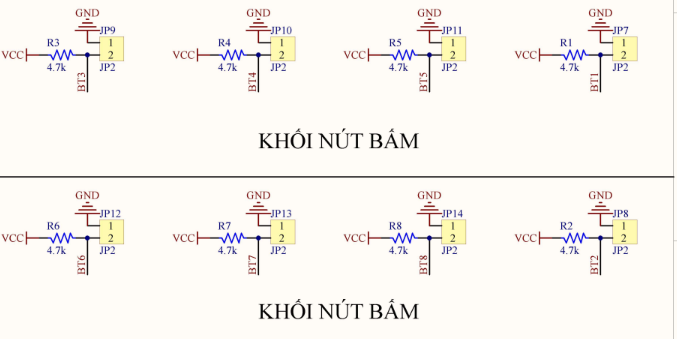
Hình . Sơ đồ nguyên lý khối điều khiển Relay

Nguyên lý hoạt động: Khi bật công tắc, dòng điện đi từ VCC tới chân RL trên board mạch Nano dẫn tới đèn trong Opto phát sáng, Opto sẽ cho dòng điện đi qua, dẫn tới dòng lúc này đi từ VCC qua 2 điện trở R17 và R18 rồi đi xuống đất, lúc này Relay sẽ có một đầu nối với VCC, một đầu nối với đất nên sẽ được kích hoạt

## Thiết kế khối giao tiếp với người dùng

### Khối nút bấm

Các nút bấm được thiết kế trong mạch với các mục đích: Điều khiển các Relay bật tắt thiết bị ,chuyển tiếp giữa 2 chế độ: Điều khiển tự động và điều khiển bằng tay và 1 nút bấm để reset mạch



Hình 4.8 Sơ đồ nguyên lý khối nút bấm

Các nút bấm được nối liền với các điện trở đóng vai trò là điện trở treo. Các điện trở treo đóng vai trò loại bỏ hiện tượng trôi nổi ở điện áp ngõ vào, tránh hiện tượng ngắn mạch cho mạch điện

## Thiết kế khối cảm biến và khối thời gian thực

### Khối cảm biến

Hệ thống sử dụng 3 cảm biến đo các thông số: Độ PH, nhiệt độ nước, độ oxi hoà tan, độ đục



Hình 4.12 Cảm biến đo độ PH và nhiệt độ nước

****

**Hình 4.13: Cảm biến đo độ oxi hoà tan**



**Hình 4.14: Cảm biến đo độ đục**

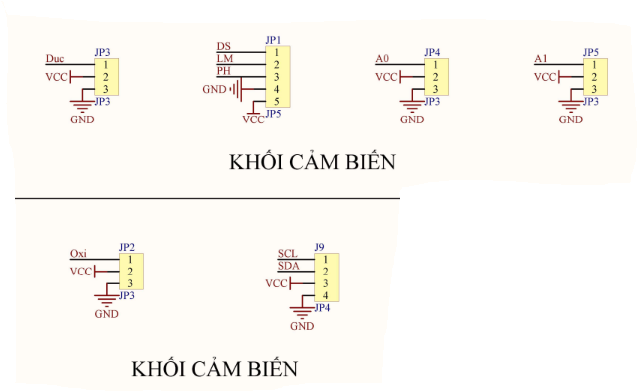
Ba cảm biến trên đều là cảm biến Analog có 3 chân như chính:

-1 Chân cấp nguồn 3.3V hoặc 5V

-1 Chân nối đất

-1 Chân Analog đọc tín hiệu cảm biến

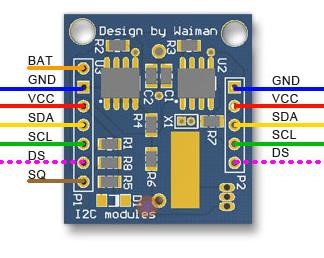
Dưới đây là sơ đồ nguyên lý của khối đọc giá trị của 3 cảm biến trên



**Hình 4.15: Sơ đồ nguyên lý của khối cảm biến**

### Khối thời gian thực

Thời gian thực tế có thể lấy bằng mạng wifi, nhưng phòng trường hợp mất wifi, cần có một module thời gian thực cho thiết bị. Nhờ module thời gian thực, thiết bị có thể hoạt động đúng so với thời gian thực, phục vụ cho chức năng bật tắt các thiết bị như máy bơm oxi, máy lọc nước, máy sưởi, sò làm mát, đèn cảnh báo PH.



Hình 4.16 Sơ đồ chân của module thời gian thực DS1307

**Bảng 4.6: Các chân của module RTC DS107**

|  |  |
| --- | --- |
| **Tên chân** | **Chi tiết** |
| VCC | Cấp nguồn 5V cho module |
| GND | Chân nối đất |
| SCL và SDA | Là hai bus dữ liệu của DS1307. Thông tin truyền và ghi đều được truyền qua 2 đường truyền này theo chuẩn I2C |

Board mạch ESP32\_DEVKIT hỗ trợ 2 chân SCL và SDA ở chân 21 và 22

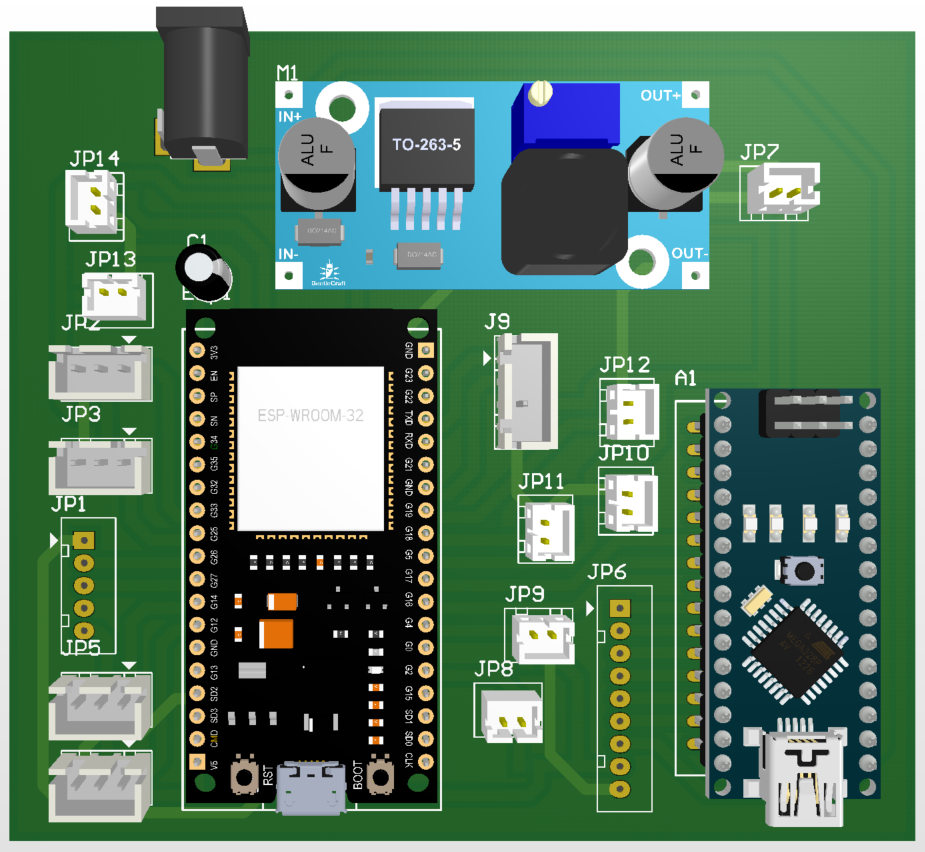
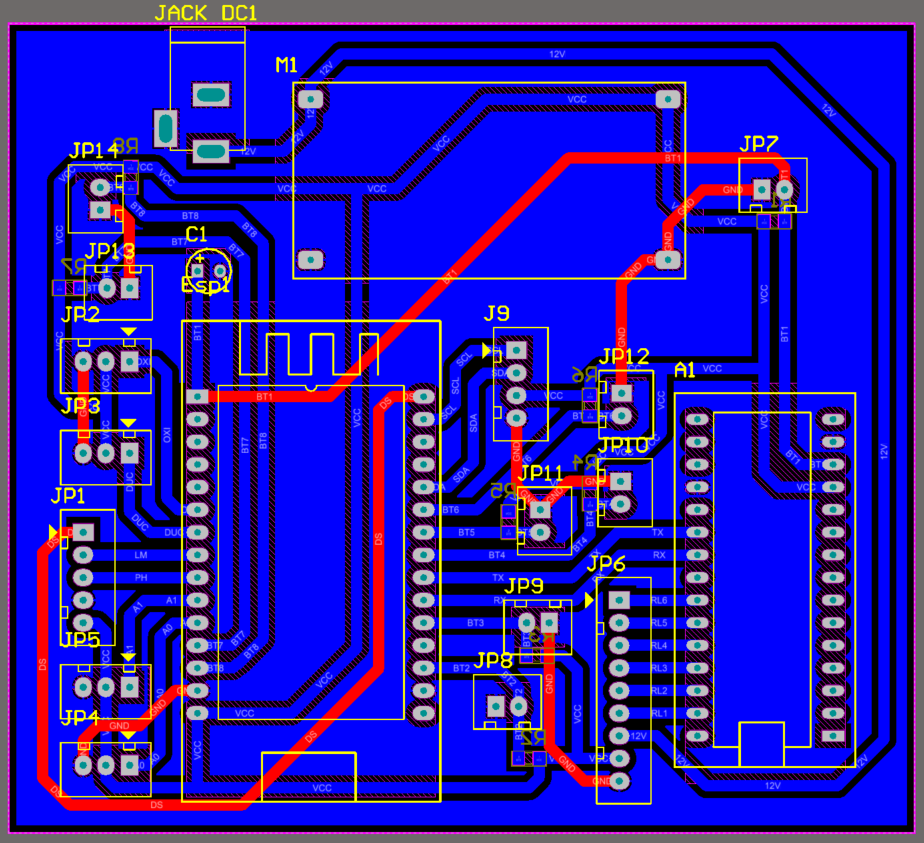
## Thiết kế layout

Do các yêu cầu chức năng của các khối đã nêu ở trên, em sẽ thiết kế hệ thống gồm 2 mạch chính: Mạch xử lý và mạch thực thi.

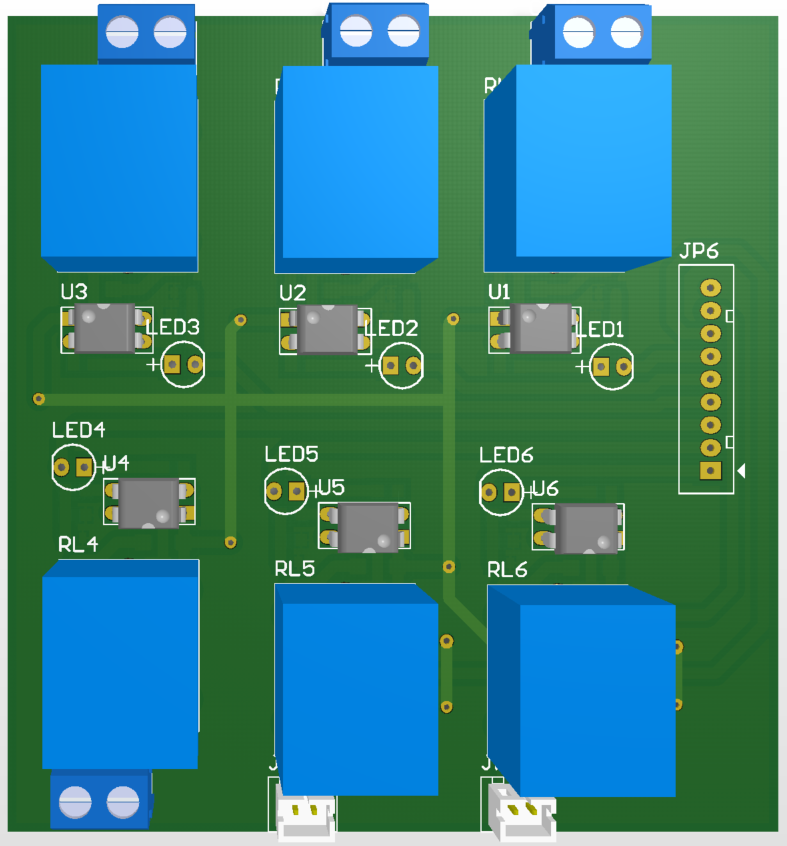
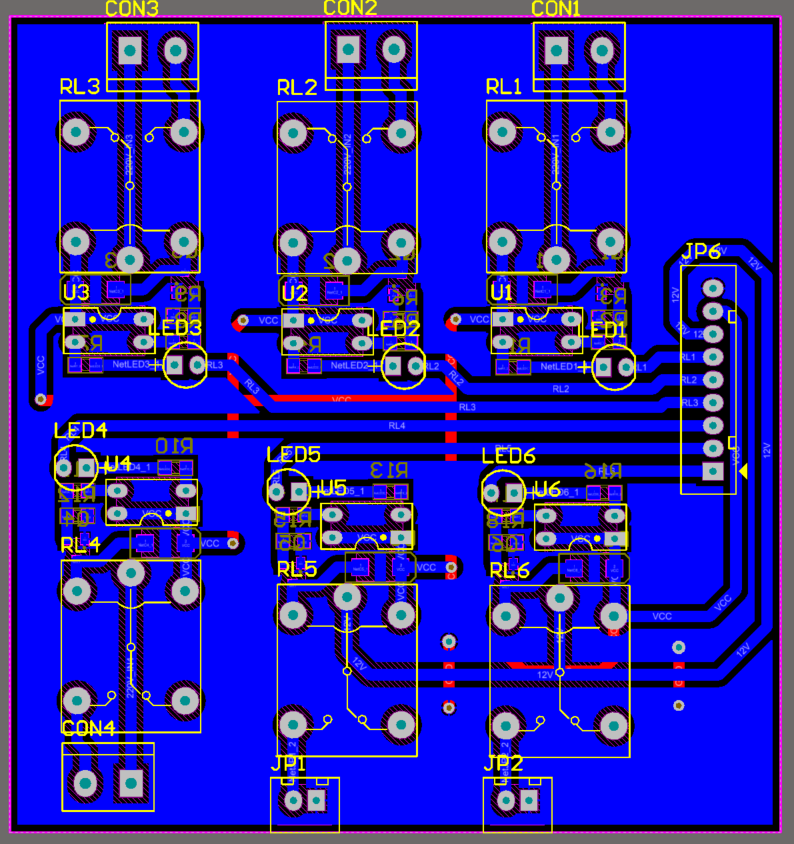
- Mạch xử lý: Bao gồm khối vi điều khiển, khối đọc cảm biến và thời gian thực, khối nguồn, được cấp nguồn đầu vào 12V

- Mạch thực thi: Gồm khối điều khiển Relay và khối nút bấm

Dưới đây là hình vẽ layout và 3D của 2 mạch



Hình 4.18: Layout và hình vẽ 3D của mạch xử lý



Hình 4.19:Layout và hình vẽ 3D của mạch thực thi

# THIẾT KẾ FIRMWARE CHO HỆ THỐNG

## Thiết kế các chức năng ngoại vi

### Các hàm đọc giá trị cảm biến Analog:

Hàm read\_ds() đọc giá trị nhiệt độ nước.

Hàm read\_ph() đọc giá trị PH.

Hàm read\_ph () đọc giá trị oxi hoà tan.

Hàm read\_ntu() đọc giá trị độ đục

**A: Các bước đọc giá trị cảm biến độ đục :**

-Bước 1: Calib cảm biến

-Bước 2: Đọc giá trị tín hiệu Analog thu được 10 lần, 2 lần liên tiếp cách nhau 10ms. Sau đó tính giá trị trung bình của tín hiệu (Ký hiệu là volt)

-Bước 3-Đo giá trị ntu=-2572.2 \* volt \* volt + 8700.5 \* volt - 4352.9

**B: Các bước đọc giá trị cảm biến độ oxi hoà tan:**

Bước 1: Calib cảm biến để thu được giá trị điện áp đi qua cảm biến(CAL1\_V) khi đã ổn định và nhiệt độ nước (CAL1\_T) tại lần Calib đầu tiên

Bước 2: Tính điện áp bão hoà theo công thức:

V\_saturation = CAL1\_V + 35 \* temperature\_c - CAL1\_T \* 35;

Bước 3-Giá trị DO=voltage\_mv \* DO\_Table[temperature\_c] / V\_saturation

**C: Các bước đọc giá trị PH**

Bước 1: Calib cảm biến

Bước 2: Đọc giá trị tín hiệu Analog thu được 10 lần, 2 lần liên tiếp cách nhau 10ms. Sau đó tính giá trị trung bình của tín hiệu (Ký hiệu là Avg)

Bước 3-Giá trị PH và DO= AVG/4095\*3.3.3+Offset

### Hàm lấy dữ liệu thời gian thực từ module RTC

Sử dụng thư viện "RTClib.h" của nhà sản xuất để hỗ trợ cho việc lấy thời gian thực bằng giao thức SPI

Hàm rtc.begin()để khởi động module thời gian thực RTC

Hàm rtc.adjust()để chỉnh sửa thời gian thực và nạp vào thiết bị

Hàm rtc.now()để lấy giá trị thời gian hiện tại

Hàm readrtc()để thực hiện hẹn giờ bật tắt thiết bị

### Điều khiển thiết bị

Khi bấm các nút bấm tương ứng, vi điều khiển ESP32 sẽ gửi các bản tin sang bo mạch Arduino Nano thông qua giao tiếp Uart để thực hiện bật tắt

**Bảng 5.1: Nội dung bản tin UART để bật tắt thiết bị**

|  |  |  |
| --- | --- | --- |
| **Nội dung bản tin** | | **Ý nghĩa** |
| Byte thứ nhất | Byte thứ hai |  |
| 1 | 0 | Tắt đèn cảnh báo PH |
| 1 | Bật đèn cảnh báo PH |
| 2 | 0 | Tắt máy bơm oxi |
| 1 | Bật máy bơm oxi |
| 3 | 0 | Tắt máy lọc nước |
| 1 | Bật máy lọc nước |
| 4 | 0 | Tắt máy sưởi |
| 1 | Bật máy sưởi |
| 5 | 0 | Tắt quạt làm mát |
| 1 | Bật quạt làm mát |

4 bit cuối trong bản tin gửi bằng UART ở byte thứ nhất và byte thứ hai sẽ không dùng đến trong đề tài

Thư viện của Arduino IDE hỗ trợ các hàm sau giúp thuận tiện cho việc lập trình:

Hàm digitalRead()để đọc trạng thái các nút bấm qua GPIO

Hàm Serial2.Write()để thực hiện gửi bản tin qua giao tiếp Uart

Hàm mySeiral.read()để đọc các bản tin Uart vừa nhận được

Hàm digitalWrite()để điều chỉnh trạng thái thiết bị qua GPIO

Ngoài các hàm đã được thư viện hỗ trợ sẵn ở trên, em viêt thêm các hàm:

Switch\_auto() để chuyển đổi trạng thái điều khiển bằng nút bấm trên mạch và điều khiển tự động

Auto\_mode() để thực hiện điều khiển tư động

Việc điều khiển tự động sẽ dựa trên việc người dùng đặt ngưỡng thiết bị qua website và các thông số thu được từ cảm biến

**Bảng 5.2: Các trường hợp của chế độ điều khiển tự động**

|  |  |
| --- | --- |
| Trường hợp thông số | Bật/Tắt thiết bị |
| Nồng độ Oxi đo được <Nồng độ Oxi Min | Bật máy bơm Oxi |
| Nồng độ Oxi đo được >Nồng độ Oxi Max | Tắt máy bơm oxi |
| Độ đục đo được >Độ đục Max | Bật máy lọc nước |
| Độ PH đo được > Độ PH Max | Bật đèn cảnh báo PH |
| Độ PH đo được < Độ PH Min | Bật đèn cảnh báo PH |
| Nhiệt độ nước > Nhiệt độ Max | Bật quạt làm mát |
| Nhiệt độ nước < Nhiệt độ Min | Bật máy sưởi |

## Protocols giao tiếp giữa phần cứng và Web Server

### Cấu trúc bản tin giao tiếp giữa Server và thiết bị

Thiết bị sẽ gửi các bản tin lên server ở 3 mode với các data cụ thể như sau:

**Bảng 5.3: Data gửi từ thiết bị lên Server theo từng mode**

|  |  |
| --- | --- |
| **Mode** | **Data gửi lên** |
| 0 | buff\_data |
| 1 | time\_send, day\_of\_week, hour\_send, buff\_data |
| 2 | status\_b2,status\_b3, status\_b4,status\_b5, status\_b6, status\_b8 |

**Bảng 5.4: Chi tiết về các data gửi đi**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên data** | **Định dạng** | **Số byte tối đa** | **Mô tả** | **Ví dụ** |
| buff\_data | Char | 50 | Các thông số nhiệt độ nước, độ PH, độ Oxy hoà tan, độ đục thu được từ các cảm biến | “temp”: “31”, “oxi”: “4”, “ph”: “7.3”,  ”duc”: “30000” |
| Time\_send | Char | 20 | Ngày gửi dữ liệu | “date”: “22/7/2022” |
| day\_of\_week | Char | 11 | Thứ gửi dữ liệu | “day”: “Fri” |
| hour\_send | Char | 11 | Giờ gửi dữ liệu trong ngày | “time”: “15” |
| status\_b2,status\_b3, status\_b4,status\_b5, status\_b6, status\_b8 | Bool | 1 | Trạng thái các thiết bị | True hoặc False |

Ở chiều ngược lại, Server sẽ trả về cho thiết bị các bản tin được xuất ra dưới dạng Object, thông qua thao tác của người dùng trên Web, các bản tin được Server được trình bày cụ thể ở bảng dưới đây

**Bảng 5.5: Các bản tin được Server gửi về thiết bị**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên bản tin** | **Các phần tử** | **Mô tả** | **Ví dụ** |
| settingTime | id, start,stop | Thông tin về thiết bị dược hẹn giờ, giờ bật và tắt của thiết bị | {“id”: “maybom”, “start”: “10:04”, “stop”: “00:53”} |
| deviceStatus | id, value | Trạng thái bật/tắt các thiết bị | {{“id”: “maybom”, “value”: false, } |
| deviceRanger | id, value | Ngưỡng của các thiết bị | {“id”: “oxiRangeMax”, “value”: “10”} |

### Đồng bộ nút bấm

Để phục vụ cho việc có thể điều khiển thiết bị thông qua cả nút bấm trên mạch và giao diện web, bản tin khi được gửi lên khi bấm nút ở mạch và bản tin được trả về khi bấm nút điều khiển trên web sẽ được xử lý qua 2 phương thức HTTP sau:

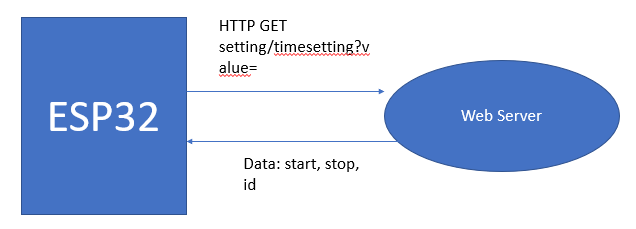


**Hình 5.1: Luồng giao tiếp đồng bộ nút bấm**

Khi bấm nút trên mạch, vi điều khiển ESP32 sẽ gửi bản tin trạng thái các thiết bị qua phương thức HTTP PUT, còn khi bấm nút trên web, ESP32 sẽ nhận được bản tin trạng thái thiết bị gửi từ web xuống thông qua phương thức HTTP GET. 2 luồng giao tiếp này sẽ giúp cho việc điều khiển bằng nút bấm trên mạch và trên web luôn được đồng bộ với nhau

### Đồng bộ thời gian thực để hẹn giờ thiết bị

Để thực hiện được chức năng hẹn giờ bật tắt thiết bị, thiết bị cần nhận được thời gian thực lấy từ người dùng khi thao tác trên website thông qua phương thức HTTP GET sau:

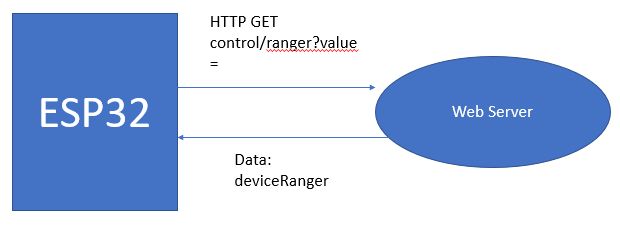


**Hình 5.2: Luồng giao tiếp để lấy thời gian thực**

Sau khi lấy được thời gian thực với Response trả về từ server, em sẽ thực hiện tách chuỗi để lấy được thời gian bật, tắt và thiết bị cần được hẹn giờ qua hàm get\_time\_set() và viết hàm ctl\_time\_auto()để thực hiện hẹn giờ bật tắt thiết bị

### Đồng bộ ngưỡng của thiết bị

Để thực hiện chức năng đồng bộ ngưỡng của thiết bị, các ngưỡng của thiết bị sau khi được người dùng điều chỉnh trực tiếp qua trang web sẽ được gửi về thiết bị qua phương thức HTTP sau



**Hình 5.3: Luồng giao tiếp lấy ngưỡng thiết bị**

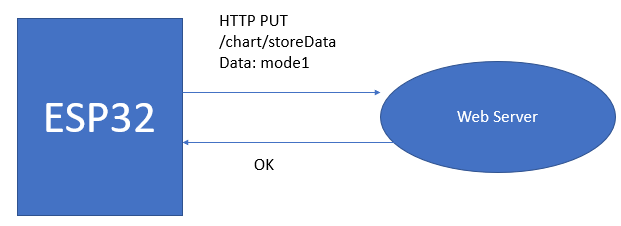
Từ phương thức HTTP GET ở hình 5.3, thiết bị sẽ xử lý bản tin nhận được để thu được các giá trị ngưỡng: Nồng độ Oxy lớn nhất, nồng đọ Oxy nhỏ nhất, nhiệt độ nước lớn nhất, nhiệt độ nước nhỏ nhất, độ PH lớn nhất, độ PH nhỏ nhất, độ đục lớn nhất. Các thông số đó sẽ được sử dụng cho việc điều khiển tự động các thiết bị ở hàm Auto\_mode()đã nêu ở phần 5.1.3

### Hiển thị thông số cảm biến dưới dạng đồ thị

Để hiển thị được thông số cảm biến dưới dạng đồ thị, trước hết thiết bị cần lấy được thông số thời gian thực qua giao tiếp với NTP Server, giao tiếp giữa vi điều khiển ESP32 và NTP Server đã được trình bày ở phần 2.6.2

Trong thư viện “time.h“ của Arduino IDE đã hỗ trợ hàm getLocaltime()để thực hiện việc giao tiếp này

Sau khi lấy được thời gian thực đóng gói một bản tin thành dạng mode 1 được đề cập ở phần 5.2.1, thiết bị sẽ giao tiếp với server qua phương thức HTTP ở hình sau:



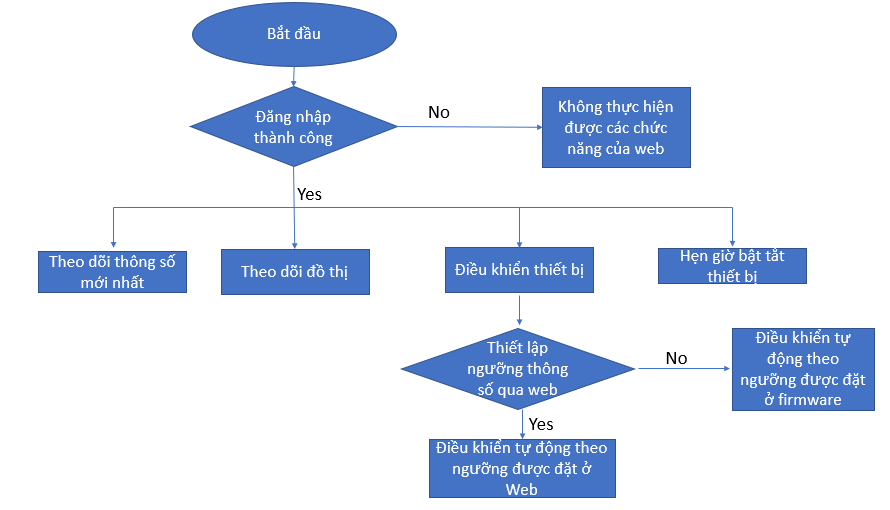
**Hình 5.4: Luồng giao tiếp để hiển thị đồ thị dữ liệu**

Sau khi bản tin được đẩy lên Server, Server sẽ cập nhật giá trị gửi lên 3 tiếng một lần để đưa vào đồ thị các thông số cảm biến trong ngày, sau mỗi một ngày thì sẽ tính giá trị trung bình trong ngày của từng thông số để từ đó đưa vào đồ thị giá trị các thông số cảm biến trong tuần

# THIẾT KẾ WEB SERVER VỚI NODEJS

## Sơ đồ chức năng của trang Web

Sơ đồ trạng thái các chức năng của trang web được thể hiện ở hình vẽ dưới đây



**Hình 6.1: Sơ đồ chức năng của trang Web**

Để có thể sử dụng được các chức năng của trang web, người dùng cần tạo tài khoản ở giao diện đăng ký và đăng nhập tài khoản được tạo. Khi thực hiện điều khiển tự động các thiết bị, nếu người dùng không đặt ngưỡng các thông số qua giao diện Web thì thiết bị sẽ được điều khiển tự động dựa theo các thông số đã được thiết lập trong firmware

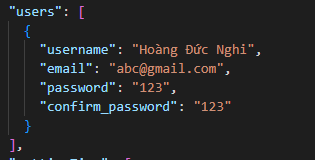
## Thiết kế cơ sở dữ liệu

Với cơ sở dữ liệu không quá phức tạp và các dữ liệu trong đồ án không có quan hệ với nhau thì em sẽ sử dụng package LowDB trong NodeJS để lưu trữ cơ sở dữ liệu

LowDB là một JSON database nhỏ gọn, được cung cấp bởi Lodash. Cơ sở dữ liệu này cho phép người dùng lưu trữ database dưới dạng JSON hoặc cũng có thể là Local Storage

Để cài đặt package LowDB trong NodeJS, người dùng sử dụng câu lệnh:

|  |
| --- |
| $ npm install LowDB |



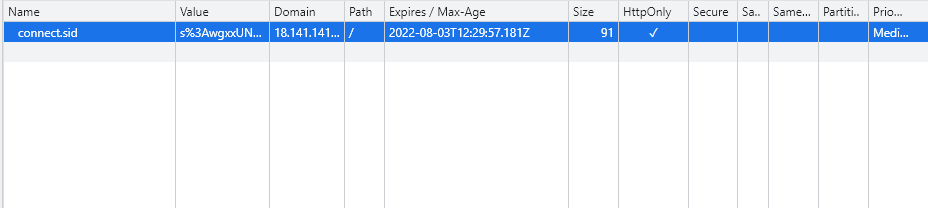
**Hình 6.2: Minh hoạ 1 cơ sở dữ liệu trong LowDB**

**Bảng 6.1: Các dữ liệu của trang Web**

|  |  |  |
| --- | --- | --- |
| **Đối tượng** | **Các thuộc tính của đối tượng** | **Ý nghĩa của đối tượng** |
| **Users** | username, email, password, confirm\_password | Thông tin tài khoản người dùng |
| settingTime | id,start, stop | Giờ bật/ tắt thiết bị |
| deviceRanger | id, value | Ngưỡng thông số của các cảm biến |
| devcieStatus | id,value | Trạng thái bật/ tắt của thiết bị |
| dataSensor | Date, day, hour,value | Thông số các cảm biến cùng thời gian gửi các thông số đó |

## Thiết kế chức năng đăng nhập

Khi người dùng đăng nhập thành công vào trang web, thông tin người dùng vừa đăng nhập sẽ được mã hoá bằng hệ mật **Bcrypt**, sau đó được lưu trong Cookie của trình duyệt web với thời gian một ngày, khi hết thời hạn của Cookie thì tài khoản của người dùng sẽ tự động đăng xuất

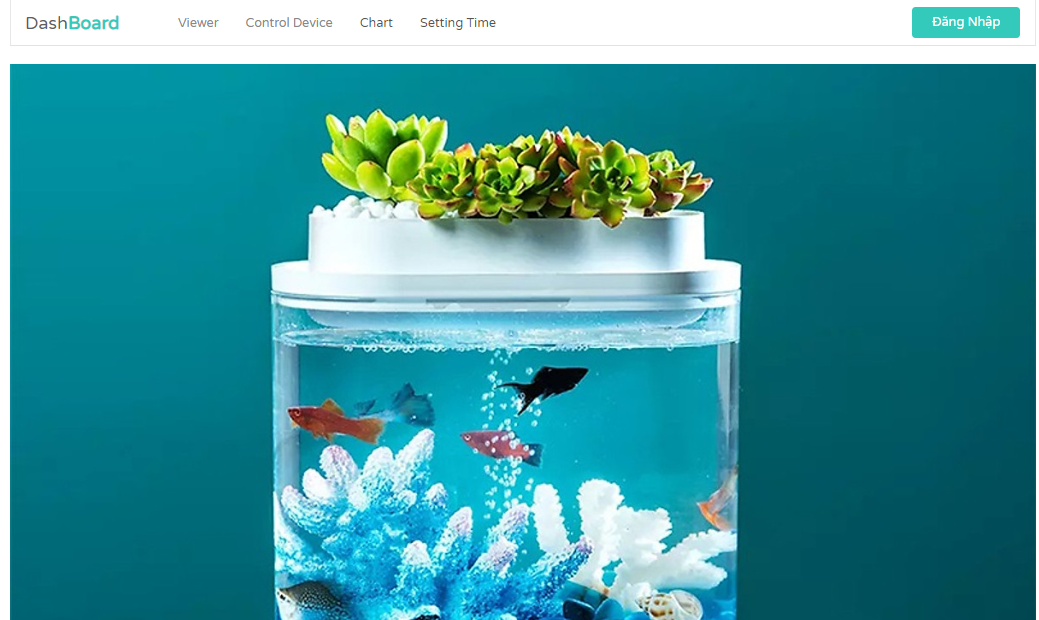


**Hình 6.3: Minh hoạ thông tin người dùng vừa đăng nhập trên Cookie**

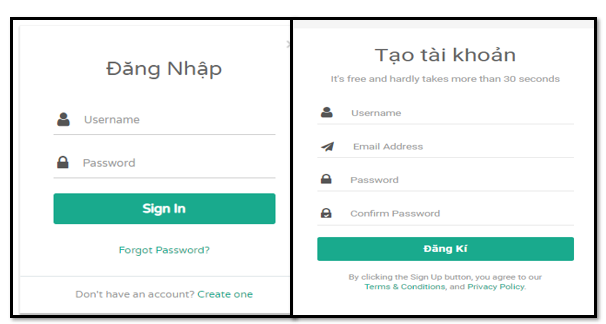
Để giúp cho việc chỉ người dùng đã đăng nhập mới có thể thực hiện các tính năng của trang web, em viết hàm isLoggedin đóng vai trò là đoạn mã trung gian (Middleware) giữa Reqest và Response. Nó nhận các request, thi hành các mệnh lệnh tương ứng trên request đó. Sau khi hoàn thành nó response (trả về) hoặc chuyển kết quả ủy thác cho một **Middleware** khác trong hàng đợi.

## Thiết kế giao diện

Ngoài các luồng giao tiếp đã nêu ở phần 5.2 và cơ sở dữ liệu đã trình bày ở phần 6.2, em sẽ thiết kế giao diện cho các chức năng của trang web bằng ngôn ngữ HTML và CSS, giao diện trang web phải tương thích với cả điện thoại và máy tính, đồng thời thân thiện với người dùng



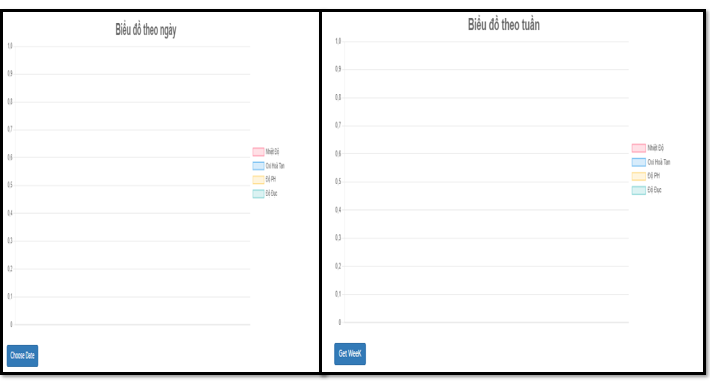
**Hình 6.4: Giao diện trang chủ**



**Hình 6.5: Giao diện đăng nhập và đăng ký người dùng**



**Hình 6.6: Giao diện hiển thị các thông số**



**Hình 6.7: Giao diện hiển thị đồ thị**



**Hình 6.8: Giao diện điều khiển thiết bị**



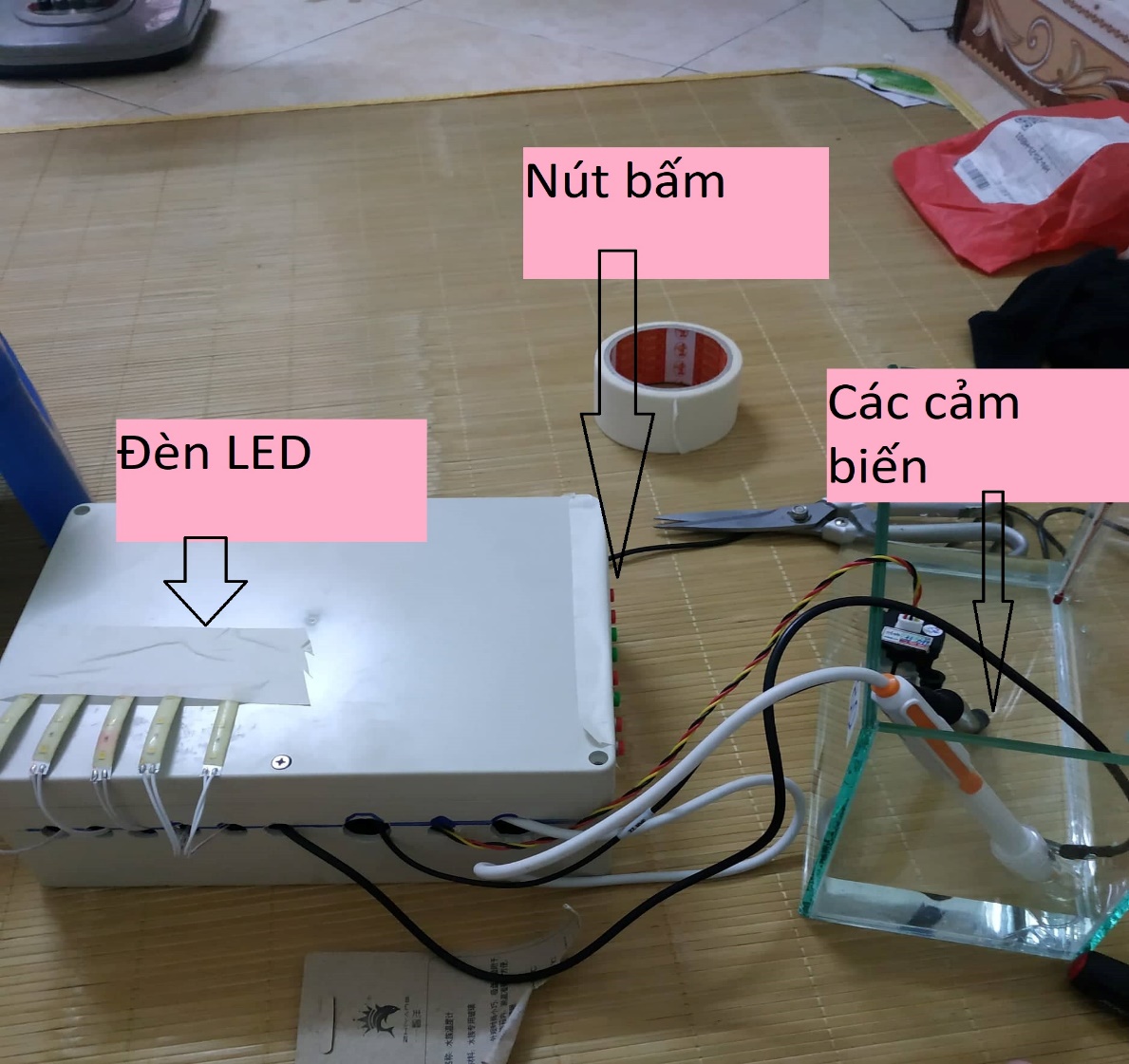
**Hình 6.9: Giao diện hẹn giờ thiết bị**

## Deploy Website lên AWS

# KẾT QUẢ

## Triển khai

Trong phạm vi của đồ án, để thuận tiện cho việc lắp đặt và kiểm tra sản phẩm, em sẽ thay thế các thiết bị đèn cảnh báo PH, máy bơm oxi, máy lọc nước, sò làm mát, máy sưởi bằng các đèn Led 12V. 2 board mạch xử lý và mạch thực thi sẽ được đựng trong một hộp nhựa chống nước chuẩn IP 65

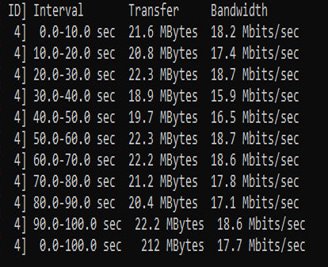


**Hình 7.1: Lắp đặt thiết bị**

## Kiểm thử

### Kiểm thử tốc độ truyền tin bằng wifi

Bằng cách sử dụng công cụ Iperf, một công cụ hỗ trợ đo các thông số về truyền tin trong frameworkd ESP-IDF, em cấu hình thiết bị thành một TCP client và truyền những gói tin với dung lượng xấp xỉ 21MB tới server, giữa 2 lần truyền tin liên tiếp cách nhau 10s, em thu được kết quả về tốc độ truyền tin của thiết bị như sau

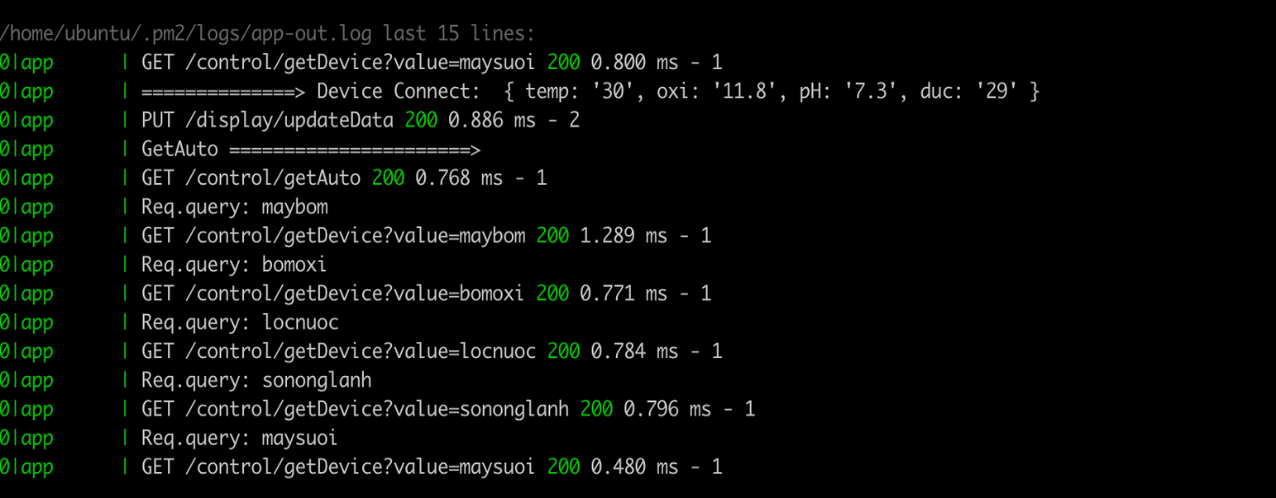


**Hình 7.2: Khảo sát tốc độ truyền tin bằng Wifi**

Từ hình 7.2, thu được kết quả về tốc độ truyền tin bằng Wifi của thiết bị rơi vào khoảng 17 tới 18Mbits/S

### Kiểm tra thời gian phản hồi của Server tới Client

Sau khi deploy web server lên cloud của AWS, em sẽ kiểm tra thời gian phản hồi của server khi thực hiện các phương thức HTTP put và get

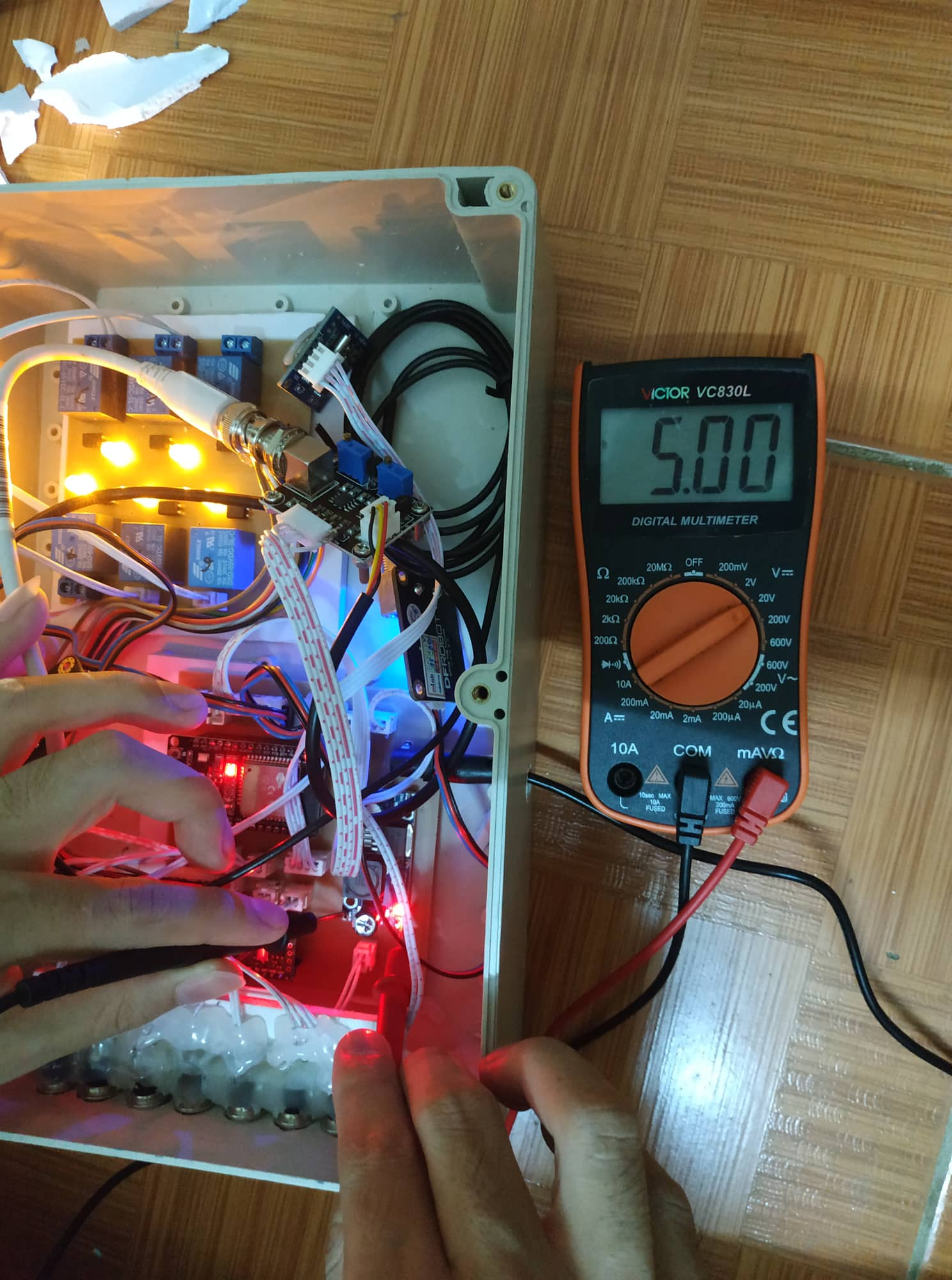


**Hình 7.3: Kiểm tra tốc độ phản hồi của Server**

Từ kết quả khảo sát ta thu được tốc độ phản hồi từ Server tới thiết bị tương đối ổn định, từ 0.5-1.3 ms cho một lần phản hồi

### Kiểm thử độ ổn định của điện áp trong mạch

Em sẽ kiểm tra độ ổn định của điện áp đi vào vi điều khiển ESP32 ở trường hợp mạch điều khiển cùng lúc cả 5 relay và đọc giá trị của 4 cảm biến đã nêu ở trên, kết quả được đo sau khi mạch đã chạy liên tục 12 tiếng



**Hình: Điện áp của ESP32 khi điều khiển 5 thiết bị**

Sau khi hoạt động ở trạng thái điều khiển thiết bị trong thời gian dài, điện áp đi vào vi điều khiển ESP32 vẫn duy trì ổn định ở mức 5V

### Kiểm thử giao tiếp của trang web và thiết bị

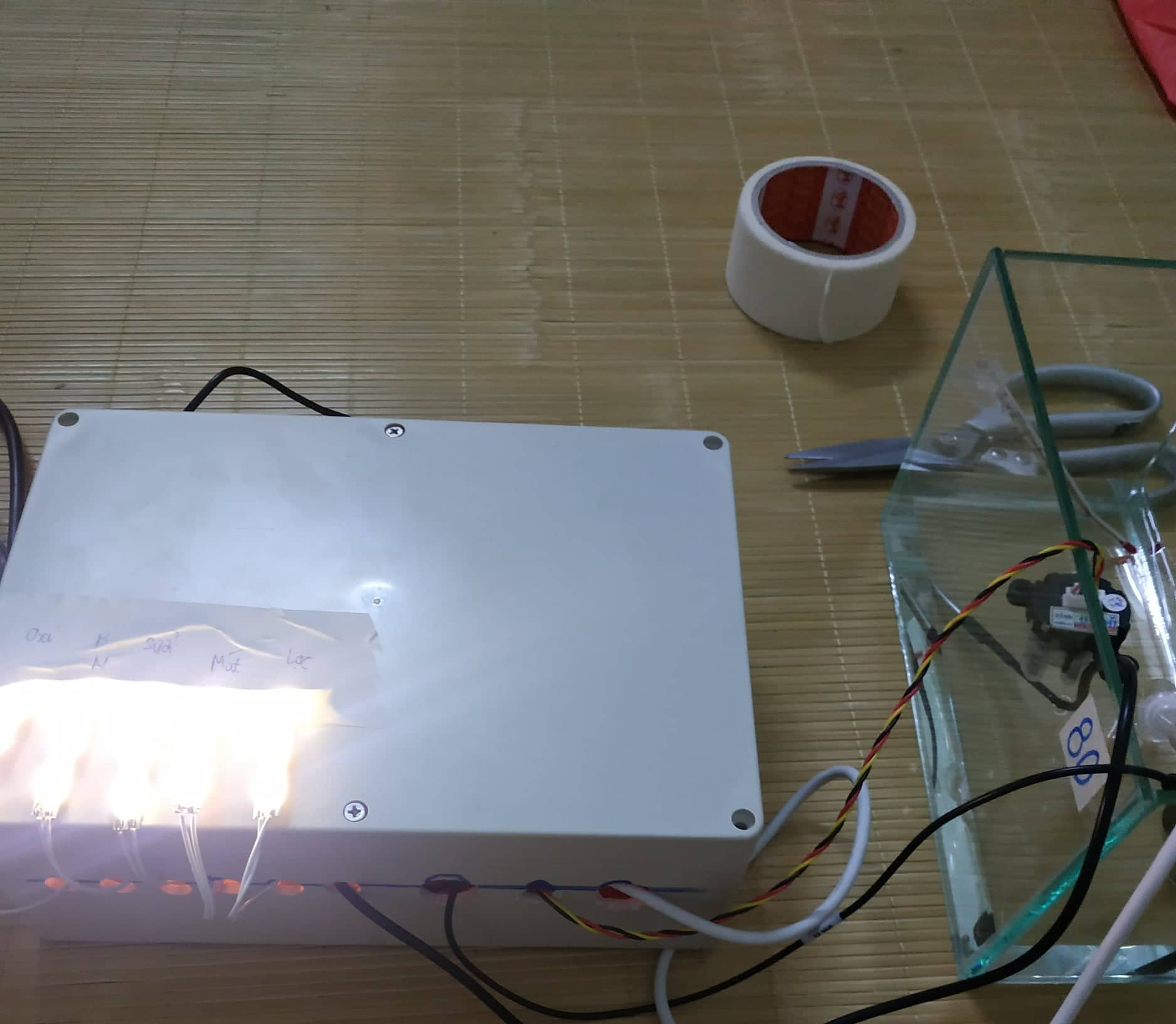
**A: Kiểm thử tính năng bật tắt thủ công**

Em sẽ thực hiện việc bật cả 5 thiết bị bằng nút bấm trên web



**Hình: thực hiện bật tắt thiết bị thủ công**

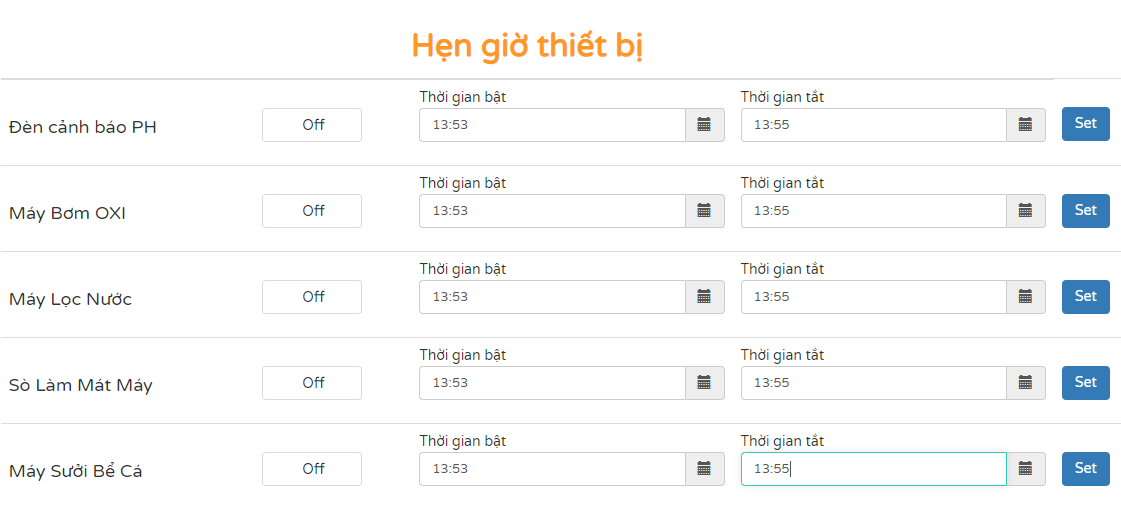
Kết quả thu được là cả 5 đèn led tượng trưng cho 5 thiết bị đều sáng



**B:Kiểm thử tính năng bật tắt tự động**

**C: Kiểm thử tính năng hẹn giờ thiết bị**

Em sẽ thực hiện hẹn giờ bật hết các thiết bị trong khoảng thời gian từ 13h52 tới 13h55 trong ngày



**Hình : Thực hiện hẹn giờ các thiết bị**

Em sẽ thu được kết quả là cả 5 đèn Led tượng trưng cho 5 thiết bị sẽ sáng trong khoảng thời gian đó



**Hình: Trạng thái thiết bị khi hẹn giờ**

Như vậy thiết bị đã được hẹn giờ bật/ tắt theo đúng khoảng thời gian mà người dùng đã đặt từ trang web

# KẾT LUẬN

## Kết luận chung

Đề tài ““**Thiết kế hệ thống giám sát và đảm bảo chất lượng nước cho bể cá**” đã đạt được một số kết quả nhất định. Mạch chạy ổn định, phản hồi giữa mạch và website điều tương đối nhanh, các thông số cảm biến thu được tương đối sát so với thực tế. Trong thời gian tới, để thiết bị có thể hoạt động ổn định, đạt chất lượng như mong đợi, em sẽ cố gắng hoàn thiện thêm chức năng và kiểm thử với nhiều trường hợp hơn trong môi trường không tốt như nhiều vật cản

## Hướng phát triển

Thời gian tới, em sẽ cải thiện thêm chất lượng của hệ thống, bổ sung thêm tính năng của của Server điều khiển, bổ sung thêm các phương thức truyền thông có phạm vi truyền được xa hơn Wifi như Lora, GSM, đồng thời tìm thêm các cảm biến có các thông số cần thiết cho việc giám sát môi trường nuôi cá

# TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | GSMA, "C-2VX Enabling Intelligent Transport". |
| [2] | T. Instruments, TLV755p Datasheet, 2018. |
| [3] | Richtek, RT8068A Datasheet, 2011. |
| [4] | SIMCOM, SIM7600 Series Hardware Design, Shanghai P.R.China. |
| [5] | E. Systems, ESP32\_D0WDQ6\_datasheet\_en, 2021. |
| [6] | SIMCOM, AT Command Manual - SIM7500\_SIM7600 Series, Shanghai P.R. China. |
| [7] | Node-RED. [Online]. Available: https://nodered.org/docs/. [Accessed 24 6 2021]. |
| [8] | [Online]. Available: https://github.com/nodesource/distributions. [Accessed 12 6 2021]. |

# PHỤ LỤC

## Phụ lục 1. Mã nguồn triển khai luồng hoạt động của Server trên Node-RED

Import đoạn mã nguồn sau vào Node-RED.

|  |
| --- |
| [{"id":"a29260e0.a8d92","type":"tab","label":"Flow 1","disabled":false,"info":""},{"id":"3b74303f.593ec","type":"tcp in","z":"a29260e0.a8d92","name":"","server":"server","host":"192.168.1.98","port":"5544","datamode":"stream","datatype":"utf8","newline":"","topic":"","base64":false,"x":140,"y":240,"wires":[["32220009.5ea58","eb731ec0.6886f"]]},{"id":"8162273d.6d44f8","type":"tcp out","z":"a29260e0.a8d92","host":"192.168.1.98","port":"5544","beserver":"reply","base64":false,"end":false,"name":"Return ACK","x":1950,"y":180,"wires":[]},{"id":"5c5a67a4.33b9d8","type":"function","z":"a29260e0.a8d92","name":"Split Msg Recieved","func":"var msgSplit = msg.payload.split(\",\");\n\nvar rawdata = {};\nrawdata.SIGN = msgSplit[1];\nrawdata.SEQ = msgSplit[2];\nrawdata.MTY = msgSplit[3];\nrawdata.ID = msgSplit[4];\nrawdata.LAT = msgSplit[5];\nrawdata.LONG = msgSplit[6];\nrawdata.DATE = msgSplit[7];\nrawdata.TIME = msgSplit[8];\nrawdata.STATE = msgSplit[9];\n\nmsg.raw = rawdata;\n\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":610,"y":240,"wires":[["1aec5967.2ff127","d84be2e4.7dce6","dae0cc55.6ddee"]]},{"id":"32220009.5ea58","type":"debug","z":"a29260e0.a8d92","d":true,"name":"original message","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"payload","targetType":"msg","statusVal":"","statusType":"auto","x":130,"y":300,"wires":[]},{"id":"1aec5967.2ff127","type":"debug","z":"a29260e0.a8d92","d":true,"name":"msg object","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"true","targetType":"full","statusVal":"","statusType":"auto","x":590,"y":300,"wires":[]},{"id":"839c4de1.f3a4f","type":"mysql","z":"a29260e0.a8d92","mydb":"ebfc643a.a07248","name":"database","x":1040,"y":340,"wires":[["ee619d57.a04ab"]]},{"id":"d84be2e4.7dce6","type":"function","z":"a29260e0.a8d92","name":"INSERT into 'rawdata'","func":"var insertQuery = {};\ninsertQuery.topic = \"INSERT INTO rawdata VALUE (null,'\"\n +msg.timestamp +\"','\"\n +msg.raw.SIGN +\"','\"\n +msg.raw.SEQ +\"','\"\n +msg.raw.MTY +\"','\"\n +msg.raw.ID +\"','\"\n +msg.raw.LAT +\"','\"\n +msg.raw.LONG +\"','\"\n +msg.raw.DATE +\"','\"\n +msg.raw.TIME +\"','\"\n +msg.raw.STATE +\"');\";\n\nreturn insertQuery;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":840,"y":340,"wires":[["839c4de1.f3a4f","cedae1d0.79526"]]},{"id":"4fcff1e3.2b37e","type":"comment","z":"a29260e0.a8d92","name":"msg \"205.205.205.6,SIGN,...\"","info":"bản tin có thêm đầu 205.205.205.6 như server của lab","x":160,"y":200,"wires":[]},{"id":"cedae1d0.79526","type":"debug","z":"a29260e0.a8d92","d":true,"name":"insert query","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"topic","targetType":"msg","statusVal":"","statusType":"auto","x":810,"y":400,"wires":[]},{"id":"eb731ec0.6886f","type":"function","z":"a29260e0.a8d92","name":"Add Timestamp to msg","func":"const currentDate = new Date();\nvar day = currentDate.getDate(),\n month = currentDate.getMonth()+1,\n year = currentDate.getFullYear();\n if (month < 10) month = \"0\" + month;\n if (day < 10) day = \"0\" + day;\nvar hour = currentDate.getHours();\n min = currentDate.getMinutes();\n sec = currentDate.getSeconds();\n if (hour < 10) hour = \"0\" + hour;\n if (min < 10) min = \"0\" + min;\n if (sec < 10) sec = \"0\" + sec;\n\nmsg.timestamp = day+\"-\"+month+\"-\"+year+\" \"+hour+\":\"+min+\":\"+sec;\n\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":380,"y":240,"wires":[["5c5a67a4.33b9d8"]]},{"id":"dae0cc55.6ddee","type":"function","z":"a29260e0.a8d92","name":"Check ID Tag đã được đăng ký chưa?","func":"msg.topic = \"select \* from users WHERE ID = '\"+msg.raw.ID+\"'\";\n\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":890,"y":240,"wires":[["52c5067.b35f3f8"]]},{"id":"52c5067.b35f3f8","type":"mysql","z":"a29260e0.a8d92","mydb":"ebfc643a.a07248","name":"database","x":1140,"y":240,"wires":[["f315d41d.6dc9c8"]]},{"id":"f315d41d.6dc9c8","type":"switch","z":"a29260e0.a8d92","name":"Check ∃ ID trong 'users'","property":"payload.length","propertyType":"msg","rules":[{"t":"eq","v":"0","vt":"str"},{"t":"else"}],"checkall":"true","repair":false,"outputs":2,"x":1350,"y":240,"wires":[["8983cbe0.fce238"],["95a1585e.6979d8"]]},{"id":"a2a20ed2.ee25e","type":"comment","z":"a29260e0.a8d92","name":"if (msg.payload.length==1) Tiếp tục luồng","info":"","x":1740,"y":300,"wires":[]},{"id":"8983cbe0.fce238","type":"function","z":"a29260e0.a8d92","name":"∄ ID trong 'users' --> return ACK, với STATE =2","func":"var mMTY = 1; // Xác nhận đã nhận được gói tin\nvar mSTATE = 2; // TAG chưa đăng ký trong hệ thống\n\nmsg.payload = msg.raw.SIGN +\",\"+\n msg.raw.SEQ +\",\"+\n mMTY +\",\"+\n msg.raw.ID +\",\"+\n mSTATE;\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":1680,"y":180,"wires":[["8162273d.6d44f8"]]},{"id":"58905630.dd3a98","type":"switch","z":"a29260e0.a8d92","name":"Check STATE (rawdata) == STATE (filtereddata)","property":"raw.STATE","propertyType":"msg","rules":[{"t":"eq","v":"payload[0].STATE","vt":"msg"},{"t":"else"}],"checkall":"true","repair":false,"outputs":2,"x":2180,"y":240,"wires":[["e7778e0.df9e57"],["e8a03eab.1c246"]]},{"id":"95a1585e.6979d8","type":"function","z":"a29260e0.a8d92","name":"Lấy STATE mới nhất trong 'filtereddata'","func":"msg.topic = \"select STATE from filtereddata WHERE ID = '\"+\n msg.raw.ID +\n \"' ORDER BY STT DESC LIMIT 1;\";\n\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":1650,"y":240,"wires":[["ec7e033e.4a7e3"]]},{"id":"ec7e033e.4a7e3","type":"mysql","z":"a29260e0.a8d92","mydb":"ebfc643a.a07248","name":"database","x":1900,"y":240,"wires":[["58905630.dd3a98"]]},{"id":"7ffea17f.c69ec","type":"tcp out","z":"a29260e0.a8d92","host":"192.168.1.98","port":"5544","beserver":"reply","base64":false,"end":false,"name":"Return ACK","x":2970,"y":160,"wires":[]},{"id":"e7778e0.df9e57","type":"function","z":"a29260e0.a8d92","name":"STATE (rawdata)==STATE(filtereddata) --> return ACK, với STATE = 4","func":"var mMTY = 1; // Xác nhận đã nhận được gói tin\nvar mSTATE = 4; // TAG chưa đăng ký trong hệ thống\n\nmsg.payload = msg.raw.SIGN +\",\"+\n msg.raw.SEQ +\",\"+\n mMTY +\",\"+\n msg.raw.ID +\",\"+\n mSTATE;\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":2630,"y":160,"wires":[["7ffea17f.c69ec"]]},{"id":"9ac8314c.fbcb7","type":"switch","z":"a29260e0.a8d92","name":"STATE (rawdata) ==?","property":"raw.STATE","propertyType":"msg","rules":[{"t":"eq","v":"0","vt":"num"},{"t":"eq","v":"1","vt":"str"}],"checkall":"true","repair":false,"outputs":2,"x":3280,"y":320,"wires":[["39a831ce.ac8bee"],["b3ce4f93.8e7eb"]]},{"id":"e8a03eab.1c246","type":"function","z":"a29260e0.a8d92","name":"Lấy tọa độ bãi xe trong 'parkinglocation'","func":"msg.topic = \"select \* from parkinglocation\";\n\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":2560,"y":240,"wires":[["e609f4d7.738908"]]},{"id":"e609f4d7.738908","type":"mysql","z":"a29260e0.a8d92","mydb":"ebfc643a.a07248","name":"database","x":2820,"y":240,"wires":[["1eee616c.1491df"]]},{"id":"1eee616c.1491df","type":"function","z":"a29260e0.a8d92","name":"Check trùng tọa độ bãi đỗ xe","func":"var i;\nvar mLAT = msg.raw.LAT;\nvar mLONG = msg.raw.LONG;\nfor(i=0;i<msg.payload.length;i++){\n var park\_Lat = msg.payload[i].LAT;\n var park\_Long = msg.payload[i].LONG;\n var park\_Radius = msg.payload[i].RADIUS;\n var r = (mLAT-park\_Lat)\*(mLAT-park\_Lat) + (mLONG-park\_Long)\*(mLONG-park\_Long);\n if(r<=(park\_Radius\*park\_Radius)) return [null,msg];\n}\nreturn [msg,null];","outputs":2,"noerr":0,"initialize":"","finalize":"","libs":[],"x":3020,"y":240,"wires":[["30d4dd55.9524d2"],["9ac8314c.fbcb7"]]},{"id":"7247c2f4.6d1e1c","type":"tcp out","z":"a29260e0.a8d92","host":"192.168.1.98","port":"5544","beserver":"reply","base64":false,"end":false,"name":"Return ACK","x":3670,"y":80,"wires":[]},{"id":"30d4dd55.9524d2","type":"function","z":"a29260e0.a8d92","name":"Không trùng tọa độ --> return ACK, với STATE = 3","func":"var mMTY = 1; // Xác nhận đã nhận được gói tin\nvar mSTATE = 3; // TAG chưa đăng ký trong hệ thống\n\nmsg.payload = msg.raw.SIGN +\",\"+\n msg.raw.SEQ +\",\"+\n mMTY +\",\"+\n msg.raw.ID +\",\"+\n mSTATE;\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":3370,"y":80,"wires":[["7247c2f4.6d1e1c"]]},{"id":"24cf5fb4.4a676","type":"mysql","z":"a29260e0.a8d92","mydb":"ebfc643a.a07248","name":"database","x":3740,"y":260,"wires":[["1b08adf6.690552","3be04e51.919eb2"]]},{"id":"7ad088ce.c7ff98","type":"function","z":"a29260e0.a8d92","name":"INSERT into DB","func":"msg.topic = \"INSERT INTO filtereddata VALUE (null,'\"\n +msg.timestamp +\"','\"\n +msg.raw.SIGN +\"','\"\n +msg.raw.SEQ +\"','\"\n +msg.raw.MTY +\"','\"\n +msg.raw.ID +\"','\"\n +msg.raw.LAT +\"','\"\n +msg.raw.LONG +\"','\"\n +msg.raw.DATE +\"','\"\n +msg.raw.TIME +\"','\"\n +msg.raw.STATE +\"');\";\n\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":3560,"y":260,"wires":[["24cf5fb4.4a676"]]},{"id":"8a2b2e42.a13f2","type":"tcp out","z":"a29260e0.a8d92","host":"192.168.1.98","port":"5544","beserver":"reply","base64":false,"end":false,"name":"Return ACK","x":4210,"y":260,"wires":[]},{"id":"1b08adf6.690552","type":"function","z":"a29260e0.a8d92","name":"Thông báo xe ra, với STATE = 0","func":"var mMTY = 1; // Xác nhận đã nhận được gói tin\nvar mSTATE = 0; // TAG chưa đăng ký trong hệ thống\n\nmsg.payload = msg.raw.SIGN +\",\"+\n msg.raw.SEQ +\",\"+\n mMTY +\",\"+\n msg.raw.ID +\",\"+\n mSTATE;\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":3970,"y":260,"wires":[["8a2b2e42.a13f2"]]},{"id":"e84d5d12.f2ff3","type":"mysql","z":"a29260e0.a8d92","mydb":"ebfc643a.a07248","name":"database","x":3740,"y":380,"wires":[["f106e35b.75d8a","3be04e51.919eb2"]]},{"id":"b3ce4f93.8e7eb","type":"function","z":"a29260e0.a8d92","name":"INSERT into DB","func":"msg.topic = \"INSERT INTO filtereddata VALUE (null,'\"\n +msg.timestamp +\"','\"\n +msg.raw.SIGN +\"','\"\n +msg.raw.SEQ +\"','\"\n +msg.raw.MTY +\"','\"\n +msg.raw.ID +\"','\"\n +msg.raw.LAT +\"','\"\n +msg.raw.LONG +\"','\"\n +msg.raw.DATE +\"','\"\n +msg.raw.TIME +\"','\"\n +msg.raw.STATE +\"');\";\n\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":3560,"y":380,"wires":[["e84d5d12.f2ff3"]]},{"id":"ef7bed6a.02281","type":"tcp out","z":"a29260e0.a8d92","host":"192.168.1.98","port":"5544","beserver":"reply","base64":false,"end":false,"name":"Return ACK","x":4230,"y":380,"wires":[]},{"id":"f106e35b.75d8a","type":"function","z":"a29260e0.a8d92","name":"Thông báo xe vào, với STATE = 1","func":"var mMTY = 1; // Xác nhận đã nhận được gói tin\nvar mSTATE = 1; // TAG chưa đăng ký trong hệ thống\n\nmsg.payload = msg.raw.SIGN +\",\"+\n msg.raw.SEQ +\",\"+\n mMTY +\",\"+\n msg.raw.ID +\",\"+\n mSTATE;\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":3980,"y":380,"wires":[["ef7bed6a.02281"]]},{"id":"cef3cf7e.2856c","type":"ui\_table","z":"a29260e0.a8d92","group":"995073cc.561d9","name":"Bảng SQL","order":3,"width":30,"height":6,"columns":[],"outputs":1,"cts":true,"x":910,"y":580,"wires":[["12953012.93ebb"]]},{"id":"cdb9dc6f.287c9","type":"mysql","z":"a29260e0.a8d92","mydb":"ebfc643a.a07248","name":"database","x":740,"y":580,"wires":[["cef3cf7e.2856c"]]},{"id":"13b4f622.ea144a","type":"function","z":"a29260e0.a8d92","name":"10 newest rawdata","func":"msg.topic = \"select \* from rawdata ORDER BY STT DESC LIMIT 10\";\n\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":550,"y":580,"wires":[["cdb9dc6f.287c9"]]},{"id":"7d8a3747.9159d8","type":"ui\_switch","z":"a29260e0.a8d92","name":"","label":"","tooltip":"","group":"995073cc.561d9","order":2,"width":2,"height":1,"passthru":true,"decouple":"false","topic":"topic","topicType":"msg","style":"","onvalue":"true","onvalueType":"bool","onicon":"","oncolor":"","offvalue":"false","offvalueType":"bool","officon":"","offcolor":"","animate":true,"x":210,"y":580,"wires":[["6a54748d.a94a6c"]]},{"id":"6a54748d.a94a6c","type":"switch","z":"a29260e0.a8d92","name":"","property":"payload","propertyType":"msg","rules":[{"t":"true"},{"t":"false"}],"checkall":"true","repair":false,"outputs":2,"x":350,"y":580,"wires":[["13b4f622.ea144a"],["fdb59289.fe8ac"]]},{"id":"fdb59289.fe8ac","type":"function","z":"a29260e0.a8d92","name":"Bỏ truy vấn","func":"msg.topic = \"select \* from rawdata ORDER BY STT DESC LIMIT 0\";\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":530,"y":640,"wires":[["cdb9dc6f.287c9"]]},{"id":"e0d12bab.ca1848","type":"ui\_template","z":"a29260e0.a8d92","group":"995073cc.561d9","name":"table TITLE","order":1,"width":28,"height":1,"format":"<div><h3>10 bản tin mới nhất</h3></div>","storeOutMessages":true,"fwdInMessages":true,"resendOnRefresh":true,"templateScope":"local","x":150,"y":740,"wires":[[]]},{"id":"ff9e65ad.cd5648","type":"ui\_template","z":"a29260e0.a8d92","group":"995073cc.561d9","name":"state TITLE","order":4,"width":30,"height":1,"format":"<div><h3>Theo dõi trạng thái xe</h3></div>","storeOutMessages":true,"fwdInMessages":true,"resendOnRefresh":true,"templateScope":"local","x":150,"y":780,"wires":[[]]},{"id":"933d240b.f2ec98","type":"ui\_text\_input","z":"a29260e0.a8d92","name":"","label":"Type here","tooltip":"ID Tag cần theo dõi","group":"995073cc.561d9","order":6,"width":9,"height":1,"passthru":true,"mode":"text","delay":"30","topic":"topic","topicType":"msg","x":1360,"y":580,"wires":[["c378144c.58dd98"]]},{"id":"afb56611.bfea18","type":"ui\_button","z":"a29260e0.a8d92","name":"","group":"995073cc.561d9","order":7,"width":8,"height":1,"passthru":false,"label":"Theo dõi","tooltip":"","color":"Black","bgcolor":"yellow","icon":"","payload":"textInput","payloadType":"flow","topic":"topic","topicType":"msg","x":2120,"y":580,"wires":[["12e07ddd.8b7d92"]]},{"id":"12953012.93ebb","type":"function","z":"a29260e0.a8d92","name":"Lấy ID khi bấm vào bản ghi","func":"msg.payload = msg.payload.ID;\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":1140,"y":580,"wires":[["933d240b.f2ec98"]]},{"id":"990967c0.332e08","type":"ui\_template","z":"a29260e0.a8d92","group":"995073cc.561d9","name":"Nhập ID TITLE","order":5,"width":2,"height":1,"format":"<div><h4>Nhập ID: </h4></div>","storeOutMessages":true,"fwdInMessages":true,"resendOnRefresh":true,"templateScope":"local","x":160,"y":820,"wires":[[]]},{"id":"ee619d57.a04ab","type":"link out","z":"a29260e0.a8d92","name":"out- insert into rawdata","links":["a8d6b6e4.9fc5a8"],"x":1155,"y":340,"wires":[]},{"id":"a8d6b6e4.9fc5a8","type":"link in","z":"a29260e0.a8d92","name":"","links":["ee619d57.a04ab"],"x":95,"y":580,"wires":[["7d8a3747.9159d8"]]},{"id":"995d6a07.855ac8","type":"switch","z":"a29260e0.a8d92","name":"is text empty?","property":"textInput","propertyType":"flow","rules":[{"t":"empty"},{"t":"else"}],"checkall":"true","repair":false,"outputs":2,"x":1760,"y":580,"wires":[["e7e97851.f3add8"],["3cea5e1b.c456d2"]]},{"id":"c378144c.58dd98","type":"change","z":"a29260e0.a8d92","name":"","rules":[{"t":"set","p":"textInput","pt":"flow","to":"payload","tot":"msg"}],"action":"","property":"","from":"","to":"","reg":false,"x":1550,"y":580,"wires":[["995d6a07.855ac8"]]},{"id":"e7e97851.f3add8","type":"change","z":"a29260e0.a8d92","name":"enabled false","rules":[{"t":"set","p":"enabled","pt":"msg","to":"false","tot":"bool"}],"action":"","property":"","from":"","to":"","reg":false,"x":1950,"y":560,"wires":[["afb56611.bfea18"]]},{"id":"3cea5e1b.c456d2","type":"change","z":"a29260e0.a8d92","name":"enabled true","rules":[{"t":"set","p":"enabled","pt":"msg","to":"true","tot":"bool"}],"action":"","property":"","from":"","to":"","reg":false,"x":1950,"y":600,"wires":[["afb56611.bfea18"]]},{"id":"12e07ddd.8b7d92","type":"function","z":"a29260e0.a8d92","name":"SELECT from DB","func":"msg.topic = \"select \* from users where ID = '\"+msg.payload +\"';\";\n\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":2310,"y":580,"wires":[["b2949078.9dff9"]]},{"id":"b2949078.9dff9","type":"mysql","z":"a29260e0.a8d92","mydb":"ebfc643a.a07248","name":"database","x":2500,"y":580,"wires":[["11d9e1c5.ca47be"]]},{"id":"8d8ba2c9.326cc","type":"function","z":"a29260e0.a8d92","name":"follow user with ID","func":"var ID={}, SDT={}, BSX={}, Ten={}, DiaChi={};\nID.payload = msg.payload[0].ID;\nSDT.payload = msg.payload[0].SDT;\nBSX.payload = msg.payload[0].BSX;\nTen.payload = msg.payload[0].Ten;\nDiaChi.payload = msg.payload[0].DiaChi;\n\nreturn [ID,SDT,BSX,Ten,DiaChi];","outputs":5,"noerr":0,"initialize":"","finalize":"","libs":[],"x":2850,"y":580,"wires":[["4d81c6fb.6bd278"],["8407fa39.dac938"],["7dc7a6d1.31e1b8"],["3cb53007.6dbff"],["6c7b265a.a6a548"]]},{"id":"4d81c6fb.6bd278","type":"ui\_text","z":"a29260e0.a8d92","group":"995073cc.561d9","order":9,"width":5,"height":1,"name":"ID\_label","label":"<i>ID: </i>","format":"{{msg.payload}}","layout":"row-left","x":3100,"y":480,"wires":[]},{"id":"8407fa39.dac938","type":"ui\_text","z":"a29260e0.a8d92","group":"995073cc.561d9","order":10,"width":6,"height":1,"name":"SDT\_label","label":"<i>Số điện thoại: </i>","format":"{{msg.payload}}","layout":"row-left","x":3110,"y":520,"wires":[]},{"id":"7dc7a6d1.31e1b8","type":"ui\_text","z":"a29260e0.a8d92","group":"995073cc.561d9","order":16,"width":11,"height":1,"name":"BSX\_label","label":"<i>Biển số xe: </i>","format":"{{msg.payload}}","layout":"row-left","x":3110,"y":560,"wires":[]},{"id":"3cb53007.6dbff","type":"ui\_text","z":"a29260e0.a8d92","group":"995073cc.561d9","order":21,"width":11,"height":1,"name":"Ten\_label","label":"<i>Tên: </i>","format":"{{msg.payload}}","layout":"row-left","x":3100,"y":600,"wires":[]},{"id":"6c7b265a.a6a548","type":"ui\_text","z":"a29260e0.a8d92","group":"995073cc.561d9","order":26,"width":11,"height":1,"name":"Address\_label","label":"<i>Địa chỉ: </i>","format":"{{msg.payload}}","layout":"row-left","x":3120,"y":640,"wires":[]},{"id":"e183e18.b0fce2","type":"ui\_toast","z":"a29260e0.a8d92","position":"dialog","displayTime":"3","highlight":"","sendall":true,"outputs":1,"ok":"OK","cancel":"","raw":true,"topic":"Thất bại!","name":"Thông báo không tồn tại ID","x":2880,"y":680,"wires":[[]]},{"id":"11d9e1c5.ca47be","type":"function","z":"a29260e0.a8d92","name":"","func":"if(msg.payload.length==0) \n{\n msg.payload = \"Không tồn tại ID!\";\n return [null, msg];\n}\nelse return [msg,null];","outputs":2,"noerr":0,"initialize":"","finalize":"","libs":[],"x":2660,"y":580,"wires":[["8d8ba2c9.326cc","883ceb0c.5d1548","34ae95b5.4f83ba"],["e183e18.b0fce2"]]},{"id":"883ceb0c.5d1548","type":"change","z":"a29260e0.a8d92","name":"get flow.ID\_follow","rules":[{"t":"set","p":"ID\_follow","pt":"flow","to":"payload[0].ID","tot":"msg"}],"action":"","property":"","from":"","to":"","reg":false,"x":2850,"y":480,"wires":[[]]},{"id":"7209c027.da8","type":"link in","z":"a29260e0.a8d92","name":"","links":["1bd26d7e.eab323","3be04e51.919eb2"],"x":3055,"y":720,"wires":[["d9508b4.924b078"]]},{"id":"332e3d95.57dca2","type":"function","z":"a29260e0.a8d92","name":"follow\_ID","func":"var SEQ={}, MTY={}, LAT={}, LONG={}, DATE={}, TIME={}, STATE={};\nSEQ.payload = msg.raw.SEQ;\nMTY.payload = msg.raw.MTY;\nLAT.payload = msg.raw.LAT;\nLONG.payload = msg.raw.LONG;\nDATE.payload = msg.raw.DATE;\nTIME.payload = msg.raw.TIME;\nSTATE.payload = Boolean(msg.raw.STATE);\n\nreturn [SEQ,MTY,LAT,LONG,DATE,TIME,STATE];","outputs":7,"noerr":0,"initialize":"","finalize":"","libs":[],"x":3320,"y":720,"wires":[["4cbbef88.bfbb5"],["ead297e5.b57cd8"],["155abd67.51d143"],["305e3fa.18678c"],["ac1c2574.dac558"],["16c580d0.90076f"],["6045113b.1c93b","61793766.8c6f58"]]},{"id":"d9508b4.924b078","type":"switch","z":"a29260e0.a8d92","name":"","property":"ID\_follow","propertyType":"flow","rules":[{"t":"eq","v":"raw.ID","vt":"msg"}],"checkall":"true","repair":false,"outputs":1,"x":3170,"y":720,"wires":[["332e3d95.57dca2"]]},{"id":"3be04e51.919eb2","type":"link out","z":"a29260e0.a8d92","name":"out\_follow\_tag","links":["7209c027.da8"],"x":3855,"y":320,"wires":[]},{"id":"34ae95b5.4f83ba","type":"function","z":"a29260e0.a8d92","name":"SELECT latest follow from DB","func":"msg.topic = \"select \* from filtereddata where ID = '\"+\n msg.payload[0].ID +\n \"' ORDER BY STT DESC LIMIT 1;\";\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":2890,"y":420,"wires":[["9760b8d8.52e828"]]},{"id":"9760b8d8.52e828","type":"mysql","z":"a29260e0.a8d92","mydb":"ebfc643a.a07248","name":"database","x":3100,"y":420,"wires":[["abb38416.e02928"]]},{"id":"4cbbef88.bfbb5","type":"ui\_text","z":"a29260e0.a8d92","group":"995073cc.561d9","order":11,"width":4,"height":1,"name":"SEQ\_label","label":"<i>SEQ: </i>","format":"{{msg.payload}}","layout":"row-left","x":3570,"y":460,"wires":[]},{"id":"ead297e5.b57cd8","type":"ui\_text","z":"a29260e0.a8d92","group":"995073cc.561d9","order":12,"width":4,"height":1,"name":"MTY\_label","label":"<i>MTY: </i>","format":"{{msg.payload}}","layout":"row-left","x":3570,"y":500,"wires":[]},{"id":"155abd67.51d143","type":"ui\_text","z":"a29260e0.a8d92","group":"995073cc.561d9","order":17,"width":8,"height":1,"name":"Lat\_label","label":"<i>Vĩ độ: </i>","format":"{{msg.payload}}","layout":"row-left","x":3560,"y":540,"wires":[]},{"id":"305e3fa.18678c","type":"ui\_text","z":"a29260e0.a8d92","group":"995073cc.561d9","order":22,"width":8,"height":1,"name":"Long\_label","label":"<i>Kinh độ: </i>","format":"{{msg.payload}}","layout":"row-left","x":3570,"y":580,"wires":[]},{"id":"ac1c2574.dac558","type":"ui\_text","z":"a29260e0.a8d92","group":"995073cc.561d9","order":28,"width":4,"height":1,"name":"Date\_label","label":"<i>Ngày: </i>","format":"{{msg.payload}}","layout":"row-left","x":3570,"y":620,"wires":[]},{"id":"abb38416.e02928","type":"function","z":"a29260e0.a8d92","name":"follow message with ID","func":"var SEQ={}, MTY={}, LAT={}, LONG={}, DATE={}, TIME={}, STATE={};\nSEQ.payload = msg.payload[0].SEQ;\nMTY.payload = msg.payload[0].MTY;\nLAT.payload = msg.payload[0].LAT;\nLONG.payload = msg.payload[0].LONG;\nDATE.payload = msg.payload[0].DATE;\nTIME.payload = msg.payload[0].TIME;\nSTATE.payload = Boolean(msg.payload[0].STATE);\n\nreturn [SEQ,MTY,LAT,LONG,DATE,TIME,STATE];","outputs":7,"noerr":0,"initialize":"","finalize":"","libs":[],"x":3320,"y":560,"wires":[["4cbbef88.bfbb5"],["ead297e5.b57cd8"],["155abd67.51d143"],["305e3fa.18678c"],["ac1c2574.dac558"],["16c580d0.90076f"],["6045113b.1c93b","61793766.8c6f58"]]},{"id":"16c580d0.90076f","type":"ui\_text","z":"a29260e0.a8d92","group":"995073cc.561d9","order":27,"width":4,"height":1,"name":"Time\_label","label":"<i>Thời gian: </i>","format":"{{msg.payload}}","layout":"row-left","x":3570,"y":660,"wires":[]},{"id":"6045113b.1c93b","type":"ui\_led","z":"a29260e0.a8d92","order":19,"group":"995073cc.561d9","width":3,"height":1,"label":"<h4>Xe vào</h4>","labelPlacement":"left","labelAlignment":"left","colorForValue":[{"color":"#008000","value":"true","valueType":"bool"}],"allowColorForValueInMessage":false,"shape":"circle","showGlow":true,"name":"xe\_vao","x":3560,"y":740,"wires":[]},{"id":"61793766.8c6f58","type":"ui\_led","z":"a29260e0.a8d92","order":24,"group":"995073cc.561d9","width":3,"height":1,"label":"<h4>Xe ra</h4>","labelPlacement":"left","labelAlignment":"left","colorForValue":[{"color":"#ff0000","value":"false","valueType":"bool"}],"allowColorForValueInMessage":false,"shape":"circle","showGlow":true,"name":"xe\_ra","x":3550,"y":780,"wires":[]},{"id":"d5202bc.a62bbd8","type":"ui\_template","z":"a29260e0.a8d92","group":"995073cc.561d9","name":"STATE\_Label","order":14,"width":4,"height":1,"format":"<div><h4>Trạng thái đỗ xe: </h4></div>","storeOutMessages":true,"fwdInMessages":true,"resendOnRefresh":true,"templateScope":"local","x":160,"y":860,"wires":[[]]},{"id":"854c6861.0057c8","type":"mysql","z":"a29260e0.a8d92","mydb":"ebfc643a.a07248","name":"database","x":3740,"y":140,"wires":[["10c8d8d.0d06c27"]]},{"id":"39a831ce.ac8bee","type":"function","z":"a29260e0.a8d92","name":"SELECT latest parked DB","func":"msg.topic = \"select DATE,TIME from filtereddata WHERE ID = '\"+\n msg.raw.ID+\n \"' ORDER BY STT DESC LIMIT 1\";\n\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":3530,"y":140,"wires":[["854c6861.0057c8"]]},{"id":"10c8d8d.0d06c27","type":"function","z":"a29260e0.a8d92","name":"Get DATE, TIME","func":"msg.lastDATE= msg.payload[0].DATE;\nmsg.lastTIME= msg.payload[0].TIME;\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":3930,"y":140,"wires":[["7ad088ce.c7ff98","c691c4c1.7ef488"]]},{"id":"c691c4c1.7ef488","type":"function","z":"a29260e0.a8d92","name":"get bill","func":"var mLastDate = msg.lastDATE.split(\"/\");\nvar mLastTime = msg.lastTIME.split(\":\");\nvar currentDate = msg.raw.DATE.split(\"/\");\nvar currentTime = msg.raw.TIME.split(\":\");\nvar dYear = parseInt(currentDate[2]) - parseInt(mLastDate[2]);\nvar dMonth = parseInt(currentDate[1]) - parseInt(mLastDate[1]);\nvar dDay = parseInt(currentDate[0]) - parseInt(mLastDate[0]);\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":4110,"y":140,"wires":[[]]},{"id":"ebfc643a.a07248","type":"MySQLdatabase","name":"","host":"127.0.0.1","port":"3306","db":"v2x\_tag","tz":"","charset":"UTF8"},{"id":"995073cc.561d9","type":"ui\_group","name":"Main Group","tab":"73c1de5.b86492","order":2,"disp":false,"width":"30","collapse":false},{"id":"73c1de5.b86492","type":"ui\_tab","name":"Hệ thống quản lý bãi đỗ xe","icon":"dashboard","order":1,"disabled":false,"hidden":false}] |

## Phụ lục 2. Mã nguồn driver giao tiếp với module SIM7600

SIM7600.cpp

|  |
| --- |
| #include "SIM7600.h"  char Global\_buffer[BUFFER\_MAX\_LEN];  uint16\_t Global\_bufferLen = 0;  bool Global\_tcpConnected = false;  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ AT CORES  void GSM\_SendAT(const char\* at)  {  while (mSerial.available())  {  mSerial.read(); // Clean the input buffer  }  memset(Global\_buffer, '\0', sizeof(Global\_buffer));  Global\_bufferLen = 0;  //Serial.printf("\"%s\"\n",at);  mSerial.println(at);  }  AT\_Response GSM\_ReadAT(const char\* foundResponse, unsigned long timeout)  {  delay(100);  unsigned long previous = millis();  while(millis()-previous < timeout)  {  while(mSerial.available())  {  char c = mSerial.read();  Global\_buffer[Global\_bufferLen++] = c;  if (strstr(Global\_buffer, foundResponse) != nullptr)  {  Serial.println(Global\_buffer);  Serial.printf("Found \"%s\"\n",foundResponse);  return AT\_OK;  }  }  }  Serial.println(Global\_buffer);  Serial.printf("Time out, Not Found \"%s\"\n",foundResponse);  return AT\_ERROR;  }  AT\_Response GSM\_ReadAT\_NonDebug(const char\* foundResponse, unsigned long timeout)  {  delay(100);  unsigned long previous = millis();  while(millis()-previous < timeout)  {  while(mSerial.available())  {  char c = mSerial.read();  Global\_buffer[Global\_bufferLen++] = c;  if (strstr(Global\_buffer, foundResponse) != nullptr)  {  return AT\_OK;  }  }  }  return AT\_ERROR;  }  void GSM\_Init()  {  Serial.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Init LTE");  delay(10);  mSerial.begin(115200, SERIAL\_8N1, RXD2, TXD2);  GSM\_SendAT("AT");  if (GSM\_ReadAT("OK", 1000L) == AT\_ERROR)  {  Serial.println("[INIT LTE FAILED]");  delay(1000);  GSM\_Init();  }  else  {  Led\_TurnOn(LED\_LTE\_RED);  Serial.printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [INIT LTE SUCCESS]\n\n");  }  }  void GSM\_TimeZoneUpdate(Time\_Data \*mTime)  {  Serial.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ TIME ZONE UPDATE");  memset(mTime->\_date,0,9);  memset(mTime->\_time,0,9);  delay(200);  GSM\_SendAT("AT+COPS=2");  while(GSM\_ReadAT("OK", 1000L) == AT\_ERROR)  {  Serial.println("Deregister - Failed");  delay(200);  GSM\_SendAT("AT+COPS=2");  }  Serial.println("Deregister - Success");  delay(200);  GSM\_SendAT("AT+CTZU=1");  while(GSM\_ReadAT("OK", 1000L) == AT\_ERROR)  {  Serial.println("Automatic Time Zone Update - Failed");  delay(200);  GSM\_SendAT("AT+CTZU=1");  }  Serial.println("Automatic Time Zone Update - Success");  delay(200);  GSM\_SendAT("AT+COPS=0");  while(GSM\_ReadAT("OK", 1000L) == AT\_ERROR)  {  Serial.println("Select Operation Automatic Mode - Failed");  delay(200);  GSM\_SendAT("AT+COPS=0");  }  Serial.println("Select Operation Automatic Mode - Success");  delay(200);  GSM\_SendAT("AT+CCLK?");  if(GSM\_ReadAT("OK", 1000L) == AT\_ERROR)  {  Serial.println("Get time - failed");  }  Serial.printf("\n\_\_\_\n%s\n\_\_\_\n", Global\_buffer);  Serial.println("Get time - Success");  Serial.printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [TIME ZONE UPDATE SUCCESS]\n\n");  }  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ TCP/IP  bool GSM\_TCPClose()  {  Global\_tcpConnected = false;  delay(200);  GSM\_SendAT("AT+CIPCLOSE=1");  GSM\_ReadAT("OK", 500L);  delay(200);  GSM\_SendAT("AT+NETCLOSE");  GSM\_ReadAT("OK", 500L);  return 1;  }  bool GSM\_TCPOpen(const char\* host, uint16\_t port)  {  // Configure PDP context  delay(200);  GSM\_SendAT("AT+CGDCONT");  if (GSM\_ReadAT("OK", 1000L) == AT\_ERROR)  {  GSM\_SendAT("AT+CGDCONT");  if (GSM\_ReadAT("OK", 1000L) == AT\_ERROR)  {  return false;  }  }  // Non transparent  // default AT+CIPMODE=0  delay(200);  GSM\_SendAT("AT+CIPMODE=0");  if (GSM\_ReadAT("OK", 1000L) == AT\_ERROR)  {  GSM\_SendAT("AT+CIPMODE=0");  if (GSM\_ReadAT("OK", 1000L) == AT\_ERROR)  {  return false;  }  }  // Activate PDP context  delay(200);  GSM\_SendAT("AT+NETOPEN");  if (GSM\_ReadAT("OK", 1000L) == AT\_ERROR)  {  GSM\_SendAT("AT+NETOPEN");  if (GSM\_ReadAT("OK", 1000L) == AT\_ERROR)  {  return false;  }  }  // set the way to get the network data manually AT+CIPRXGET=1 --> Buffer access  delay(200);  GSM\_SendAT("AT+CIPRXGET=1");  if (GSM\_ReadAT("OK", 1000L) == AT\_ERROR)  {  GSM\_SendAT("AT+CIPRXGET=1");  if (GSM\_ReadAT("OK", 1000L) == AT\_ERROR)  {  GSM\_TCPClose();  return false;  }  }  //Open Connection Socket AT+CIPOPEN  char buf[56];  memset(buf, '\0', sizeof(buf));  sprintf(buf, "AT+CIPOPEN=1,\"TCP\",\"%s\",%d", host, port);  delay(500);  GSM\_SendAT(buf);  if (GSM\_ReadAT("OK", 3500L) == AT\_ERROR)  {  GSM\_TCPClose();  return false;  }  Global\_tcpConnected = true;  return true;  }  bool GSM\_TCPSend(const char\* host, uint16\_t port, const char\* data, bool listenACK)  {  char buf[64];  unsigned short len = strlen(data);  //CHECK CONNECT  if (Global\_tcpConnected == false)  {  GSM\_TCPOpen(host, port);  }  delay(300);  //SEND DATA  memset(buf, '\0', sizeof(buf));  sprintf(buf, "AT+CIPSEND=1,%d", len);  GSM\_SendAT(buf);  if (GSM\_ReadAT(">", 3500L) == AT\_ERROR)  {  GSM\_TCPClose();  return false;  }  delay(200);  GSM\_SendAT(data);  if (GSM\_ReadAT("OK", 3500L) == AT\_ERROR)  {  GSM\_TCPClose();  return false;  }  //LISTEN ACK  if(listenACK==true)  {  char \*AckBuffer = new char[64];  memset(AckBuffer,0,64);  delay(1500);  GSM\_SendAT("AT+CIPRXGET=2,1,100");  if (GSM\_ReadAT("OK", 3500L) == AT\_ERROR)  {  GSM\_TCPClose();  return false;  }  sscanf(Global\_buffer,"%\*[^\n]\n%\*[^\n]\n%[^\n]%\*s",AckBuffer);  Serial.printf("ACK is: %s\n",AckBuffer);  // set the way to get the network data manually AT+CIPRXGET=1 --> Buffer access  delay(200);  GSM\_SendAT("AT+CIPRXGET=1");  if (GSM\_ReadAT("OK", 1000L) == AT\_ERROR)  {  GSM\_TCPClose();  }  ACK\_FromServer ackFromSv;  sscanf(AckBuffer,"%[^,],%d,%d,%[^,],%d",&ackFromSv.\_SIGN,&ackFromSv.\_SEQ,&ackFromSv.\_MTY,&ackFromSv.\_ID,&ackFromSv.\_STATE);  delete [] AckBuffer;  if(ackFromSv.\_STATE==0||ackFromSv.\_STATE==1)  {  char ackToSv[26];  sprintf((char\*)(ackToSv),"%s,",ackFromSv.\_SIGN);  sprintf((char\*)(ackToSv+strlen((char\*)ackToSv)),"%d,",ackFromSv.\_SEQ);  sprintf((char\*)(ackToSv+strlen((char\*)ackToSv)),"%d,",ackFromSv.\_MTY);  sprintf((char\*)(ackToSv+strlen((char\*)ackToSv)),"%s",ackFromSv.\_ID);  GSM\_TCPSend(host, port, ackToSv);  Serial.printf("Xac nhan: STATE = %d\n", ackFromSv.\_STATE);  return true;  }  else  {  Serial.printf("Yeu cau bi tu choi: STATE = %d\n", ackFromSv.\_STATE);  return false;  }  }  return true;  }  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ GPS  void GSM\_EnableGPS()  {  // Truoc khi enable GPS nen goi ham GSM\_Init() truoc  Serial.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Enable GPS");  GSM\_SendAT("AT+CGPS=1,1");  if(GSM\_ReadAT("OK", 1500L)==AT\_OK)  {  Led\_TurnOn(LED\_GPS\_RED);  Serial.printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [ENABLE GPS SUCCESS]\n\n");  }  else  {  GSM\_SendAT("AT+CGPS=0");  Serial.println("[ENABLE GPS FAILED]");  delay(1000);  GSM\_EnableGPS();  }  }  // Hàm xử lý tọa độ phục vụ cho hàm GSM\_getDataGPS()  double convert\_coordinate(double coord){ // coord = ddmm.mmmmmm, convert to dd+ mm.mmmmmm/60  int dd = coord/100;  double mm\_mm = coord- dd\*100;  double res = dd+mm\_mm/60;  char arr[16];  memset(arr,0,16);  sprintf(arr,"%.8lf",res);  sscanf(arr,"%lf",&res);  return res;  }  void GSM\_getDataGPS(GNSS\_Data \*mgpsData)  {  char gpsBuffer[128];  delay(200);  GSM\_SendAT("AT+CGPSINFO");  if (GSM\_ReadAT\_NonDebug("OK", 1500L) == AT\_OK)  {  sscanf(Global\_buffer,"%\*[^\n]\n%[^\n]%\*s",gpsBuffer);  Serial.println(gpsBuffer);  sscanf(gpsBuffer,"+CGPSINFO:%lf,%\*c,%lf,%\*s",&mgpsData->\_lat, &mgpsData->\_long);  mgpsData->\_lat = convert\_coordinate(mgpsData->\_lat);  mgpsData->\_long = convert\_coordinate(mgpsData->\_long);  }  }  bool GSM\_isOkeyGPS(GNSS\_Data mgpsData)  {  return mgpsData.\_lat!=0;  }  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ TIME  void GSM\_getDataTime(Time\_Data \*timeData)  {  char timeBuffer[128];  delay(200);  GSM\_SendAT("AT+CCLK?");  if(GSM\_ReadAT\_NonDebug("OK", 1500L) == AT\_OK)  {  printf("Global\_buffer: [%s]\n",Global\_buffer);  sscanf(Global\_buffer,"%\*[^\n]\n%[^\n]%\*s",timeBuffer);  printf("Time Buffer: [%s]\n",timeBuffer);  sscanf(timeBuffer,"+CCLK: \"%[^,],%[^+]+%\*s",&timeData->\_date,&timeData->\_time);  char tmp = timeData->\_date[0];  timeData->\_date[0]= timeData->\_date[6];  timeData->\_date[6]= tmp;  tmp = timeData->\_date[1];  timeData->\_date[1]= timeData->\_date[7];  timeData->\_date[7]= tmp;  printf("date: [%s]\n",timeData->\_date);  printf("time: [%s]\n\n",timeData->\_time);  }  }  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ PACKING DATA  void GSM\_PackingData(char\* buffer, int seq, unsigned char mty, int id, GNSS\_Data gpsData, Time\_Data timeData, unsigned char state)  {  int len = strlen("205.205.205.6,");  sprintf((char\*)(buffer),"205.205.205.6,");  sprintf((char\*)(buffer+len),"@V2X,");  sprintf((char\*)(buffer+strlen((char\*)buffer)),"%d,",seq);  sprintf((char\*)(buffer+strlen((char\*)buffer)),"%d,",mty);  sprintf((char\*)(buffer+strlen((char\*)buffer)),"%015d,",id);  sprintf((char\*)(buffer+strlen((char\*)buffer)),"%.8lf,%.8lf,",gpsData.\_lat,gpsData.\_long);  sprintf((char\*)(buffer+strlen((char\*)buffer)),"%s,%s,",timeData.\_date,timeData.\_time);  sprintf((char\*)(buffer+strlen((char\*)buffer)),"%d",state);  }  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ PACKING DATA  void getDatafromEEPROM()  {  int address = 0;  latest\_Lat = EEPROM.readDouble(address);  Serial.printf("Lat from MEM: %.8lf\n",latest\_Lat);  address += sizeof(double);  latest\_Long = EEPROM.readDouble(address);  Serial.printf("Long from MEM: %.8lf\n",latest\_Long);  address += sizeof(double);  latest\_State = EEPROM.readUChar(address);  Serial.printf("State from MEM: %d\n",latest\_State);  address += sizeof(unsigned char);  if(latest\_State == 1) // state = 1: xe dang do  {  Led\_TurnOn(LED\_PARK\_GREEN);  Led\_TurnOff(LED\_PARK\_RED);  }  else // sate = 0: xe dang ko do  {  Led\_TurnOn(LED\_PARK\_RED);  Led\_TurnOff(LED\_PARK\_GREEN);  }  }  void SaveDataToEEPROM(GNSS\_Data mGpsData, unsigned char mState)  {  int address = 0;  latest\_Lat = mGpsData.\_lat;  EEPROM.writeDouble(address, mGpsData.\_lat);  address += sizeof(double);  latest\_Long = mGpsData.\_long;  EEPROM.writeDouble(address, mGpsData.\_long);  address += sizeof(double);  latest\_State = mState;  EEPROM.writeUChar(address, mState);  address += sizeof(unsigned char);  EEPROM.commit();  }  void getLatestGPS(GNSS\_Data \*mGpsData)  {  mGpsData->\_lat = latest\_Lat;  mGpsData->\_long = latest\_Long;  }  void ClearDataToEEPROM() // latest\_Lat=0, lastest\_Long=0, latest\_State=0, clear data in EEPROM  {  int address = 0;  latest\_Lat = 0;  EEPROM.writeDouble(address, 0);  address += sizeof(double);  latest\_Long = 0;  EEPROM.writeDouble(address, 0);  address += sizeof(double);  latest\_State = 0;  EEPROM.writeUChar(address, 0);  address += sizeof(unsigned char);  EEPROM.commit();  } |

## Phụ lục 3. Mã nguồn driver giao tiếp WiFi

WiFi\_TCP.cpp

|  |
| --- |
| #include "WiFi\_TCP.h"  WiFiClient mclient;  bool isConnectedWiFi(unsigned long timeout)  {  if(WiFi.status() != WL\_CONNECTED)  {  Serial.println("Reconnect WiFi");  WiFi.begin(WIFI\_NAME, WIFI\_PASSWORD);  unsigned long previous = millis();  while ((WiFi.status() != WL\_CONNECTED)&&(millis()-previous < timeout)) {  delay(200);  Serial.print(".");  }  if(WiFi.status() == WL\_CONNECTED)  {  Serial.println("Reconnect Success!");  return true;  }  else  {  Serial.println("Reconnect Fail!");  return false;  }  }  else  {  Serial.println("WiFi Connected");  return true;  }  }  bool WiFi\_TCPSend(const char\* host, uint16\_t port, const char\* data, bool listenACK )  {  //CONNECT  WiFiClient client;  uint16\_t AckBufferLen = 0;  //ACK\_Test \*ack\_Test = new ACK\_Test;  if (!mclient.connect(host, port))  {  Serial.println("TCPSEND: Connection to host failed");  return false;  }  Serial.println("Connection to host successful");  //SEND DATA  if(mclient.print(data)==0)  {  Serial.println("Send data failed!");  return false;  }  Serial.println("Send data successful!");  //LISTEN ACK  if(listenACK == true)  {  char\* AckBuffer = new char[64];  memset(AckBuffer,0,64);  unsigned long previous = millis();  while (mclient.available()==0)  {  if(millis()-previous > 3000L)  {  Serial.println("Listen ACK Timeout");  mclient.stop();  return false;  }  }  while (mclient.available())  {  char c = mclient.read();  AckBuffer[AckBufferLen++] = c;  }  Serial.printf("ACK is: %s\n",AckBuffer);  ACK\_FromServer ackFromSv;  sscanf(AckBuffer,"%[^,],%d,%d,%[^,],%d",&ackFromSv.\_SIGN,&ackFromSv.\_SEQ,&ackFromSv.\_MTY,&ackFromSv.\_ID,&ackFromSv.\_STATE);  delete [] AckBuffer;  if(ackFromSv.\_STATE==0||ackFromSv.\_STATE==1)  {  char ackToSv[30];  sprintf((char\*)(ackToSv),"%s,",ackFromSv.\_SIGN);  sprintf((char\*)(ackToSv+strlen((char\*)ackToSv)),"%d,",ackFromSv.\_SEQ);  sprintf((char\*)(ackToSv+strlen((char\*)ackToSv)),"%d,",ackFromSv.\_MTY);  sprintf((char\*)(ackToSv+strlen((char\*)ackToSv)),"%s",ackFromSv.\_ID);  WiFi\_TCPSend(host, port, ackToSv);  Serial.printf("Xac nhan: STATE = %d\n", ackFromSv.\_STATE);  return true;  }  else  {  Serial.printf("Yeu cau bi tu choi: STATE = %d\n", ackFromSv.\_STATE);  return false;  }  }  return true;  } |

## Phụ lục 4. Mã nguồn firmware aplication

V2X\_Tag\_V2.ino

|  |
| --- |
| #include <WiFi.h>  #include "src/utility/GSM/SIM7600.h"  #include "src/utility/WiFi/WiFi\_TCP.h"  #include "src/utility/User\_Interface/User\_Interface.h"  #include "src/config/config.h"  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ FreeRTOS CONFIG  #define ESP32\_RUNNING\_CORE\_0 0  #define ESP32\_RUNNING\_CORE\_1 1  // TaskHandle of Tasks  TaskHandle\_t xHandle\_Touch;  TaskHandle\_t xHandle\_GetGPS;  TaskHandle\_t xHandle\_BlinkLed;  TaskHandle\_t xHandle\_ScanWiFi;  // Declare 4 task  void CheckTouchPressed(void \*pvParameters);  void GetGPS(void \*pvParameters);  void BlinkLed(void \*pvParameters);  void ScanWiFi(void \*pvParameters);  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ GLOBAL VARIABLE  GNSS\_Data gpsData;  Time\_Data timeData;  char sendBuffer[256];  void setup() {  delay(200);  Serial.begin(115200);  Led\_TurnOff(LED\_WIFI\_GREEN);  Led\_TurnOn(LED\_LTE\_RED);  Led\_TurnOff(LED\_GPS\_GREEN);  Led\_TurnOn(LED\_GPS\_RED);  if (!EEPROM.begin(20)) {  Serial.println("Failed to initialise EEPROM");  Serial.println("Restarting...");  delay(1000);  ESP.restart();  }  getDatafromEEPROM(); // get lat, long, state saved in eeprom    // Led\_TurnOff(LED\_PARK\_GREEN);  // Led\_TurnOff(LED\_PARK\_RED);  xTaskCreatePinnedToCore(  CheckTouchPressed // Function that should be called  , "CheckTouchPressed" // Name of the task  , 8192 // This stack size (bytes)  , NULL // Parameter to pass  , 3 // Priority, with 3  , &xHandle\_Touch  , ESP32\_RUNNING\_CORE\_1);  xTaskCreatePinnedToCore(  GetGPS // Function that should be called  , "GetGPS" // Name of the task (for debugging)  , 4096 // This stack size (bytes)  , NULL // Parameter to pass  , 3 // Priority, with 3  , &xHandle\_GetGPS  , ESP32\_RUNNING\_CORE\_0);  xTaskCreatePinnedToCore(  BlinkLed // Function that should be called  , "BlinkLed" // Name of the task (for debugging)  , 1024  , NULL // Parameter to pass  , 3 // Priority, with 3  , &xHandle\_BlinkLed  , ESP32\_RUNNING\_CORE\_0);  xTaskCreatePinnedToCore(  ScanWiFi // Function that should be called  , "ScanWiFi" // Name of the task (for debugging)  , 2048 // This stack size (bytes)  , NULL // Parameter to pass  , 3 // Priority, with 3  , &xHandle\_ScanWiFi  , ESP32\_RUNNING\_CORE\_0);  }  void loop() {  delay(100);  }  /\*--------------------------------------------------\*/  /\*---------------------- Tasks ---------------------\*/  /\*--------------------------------------------------\*/  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ TOUCH  void CheckTouchPressed(void \*pvParameters) // This is a task.  {  //CONTROL POWER  pinMode(MCU\_PWK, OUTPUT); // default = LOW  pinMode(MCU\_RST, OUTPUT); // default = LOW  delay(100);  // POWER ON SIM7600  Serial.println("\*\*\*\*\*POWER ON SIM7600\*\*\*\*\*");  digitalWrite(MCU\_PWK, HIGH); // Power on SIM7600  vTaskDelay(500); // T\_on: [Min = 100ms, Typ. = 500ms, Max = --]  digitalWrite(MCU\_PWK, LOW); // Power on SIM7600  // initializing WiFi  WiFi.begin(WIFI\_NAME, WIFI\_PASSWORD);  vTaskDelay(12000); // wait for UART (SIM) ready: T\_on(uart) = 12s (Typ.)  /\*  // POWER OFF SIM7600  Serial.println("\*\*\*\*\*POWER OFF SIM7600\*\*\*\*\*");  digitalWrite(MCU\_PWK, HIGH); // Power off SIM7600  vTaskDelay(3750); // T\_off: [Min = 2.5s, Typ. = --, Max = 5.0s]  digitalWrite(MCU\_PWK, LOW); // Power off SIM7600  vTaskDelay(15000); // wait for UART port off: T\_off(uart) = 15s (Typ.)  \*/  /\*  // RESET SIM7600  Serial.println("\*\*\*\*\*RESET SIM7600\*\*\*\*\*");  digitalWrite(MCU\_RST, HIGH); // Reset SIM7600  vTaskDelay(200); // T\_reset: [Min = 100ms, Typ. = 200ms, Max = 500mss]  digitalWrite(MCU\_RST, LOW); // Reset SIM7600  vTaskDelay(500); // wait for Reset  \*/  // initializing SIM  GSM\_Init();  GSM\_EnableGPS();  GSM\_TimeZoneUpdate(&timeData);  GSM\_TCPOpen(HOST, PORT);  // Bat dau lay gps lien tuc va xoa task nhay led khi nam trong vung do xe  vTaskResume(xHandle\_ScanWiFi);  vTaskResume(xHandle\_GetGPS);  vTaskDelete(xHandle\_BlinkLed);  for (;;) // A Task shall never return or exit.  {  // Serial.println("$ TASK TOUCH");  int t = touchRead(TOUCH);  if (t < threshold && !touchStarted) // Touched the Button  {  touchStarted = true;  touchTime = millis();  }  else if (t >= threshold && touchStarted) // Untouched the Button  {  touchStarted = false;    // LONG PRESS <Thong bao xe ra>    if (millis() - touchTime > touchDurationLim2)  {  Serial.printf(" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");  Serial.printf(" | Long Touch: %dms |\n", millis() - touchTime);  Serial.printf(" |\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\n");  vTaskSuspend(xHandle\_GetGPS);  vTaskSuspend(xHandle\_ScanWiFi);  BlinkLedFunc(LED\_PARK\_RED);  // <-- begin process -->  if(latest\_State==1) // Trang thai DA GUI XE thi moi cho phep gui yeu cau XE RA  {  GSM\_getDataTime(&timeData);  getLatestGPS(&gpsData);  //PACKING DATA  memset(sendBuffer,0,sizeof(sendBuffer));  GSM\_PackingData(sendBuffer, seq, 1, ID, gpsData, timeData, 0); //MTY = 1; STATE = 0    //SEND DATA  if(TAG\_TCPSend(HOST, PORT, sendBuffer, true))  {  ClearDataToEEPROM(); // latest\_Lat=0, lastest\_Long=0, latest\_State=0, clear data in EEPROM  seq++;  Led\_TurnOn(LED\_PARK\_RED);  Led\_TurnOff(LED\_PARK\_GREEN);  }  else  {  Led\_TurnOn(LED\_PARK\_GREEN);  Led\_TurnOff(LED\_PARK\_RED);  }  }  else  {  Serial.printf("Release Failed! You haven't Parked your car\n");  }  // <-- end process -->    vTaskResume(xHandle\_GetGPS);  vTaskResume(xHandle\_ScanWiFi);  }  // SHORT PRESS <Thong bao xe vao>    else if (millis() - touchTime > touchDurationLim1)  {  Serial.printf(" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");  Serial.printf(" | Touched: %dms |\n", millis() - touchTime);  Serial.printf(" |\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\n");  vTaskSuspend(xHandle\_GetGPS);  vTaskSuspend(xHandle\_ScanWiFi);  BlinkLedFunc(LED\_PARK\_GREEN);  // <-- begin process -->  if(latest\_State==0) // Trang thai CHUA GUI XE thi moi cho phep gui yeu cau XE VAO  {  GSM\_getDataTime(&timeData);  GSM\_getDataGPS(&gpsData);  //PACKING DATA  memset(sendBuffer,0,sizeof(sendBuffer));  GSM\_PackingData(sendBuffer, seq, 1, ID, gpsData, timeData, 1); //MTY = 1; STATE = 1    //SEND DATA  if(TAG\_TCPSend(HOST, PORT, sendBuffer, true))  {  SaveDataToEEPROM(gpsData,1); // (GNSS\_Data, state); state = 1: yeu cau gui xe thanh cong va xe dang do  seq++;  Led\_TurnOn(LED\_PARK\_GREEN);  Led\_TurnOff(LED\_PARK\_RED);  }  else  {  Led\_TurnOn(LED\_PARK\_RED);  Led\_TurnOff(LED\_PARK\_GREEN);  }  }  else  {  Serial.printf("Register Failed! You have Parked your car\n");  }  // <-- end process -->    vTaskResume(xHandle\_GetGPS);  vTaskResume(xHandle\_ScanWiFi);  }  }  // Serial.print("Stack Size TOUCH: ");  // Serial.println(uxTaskGetStackHighWaterMark(NULL));  vTaskDelay(100);  }  }  void BlinkLedFunc(int typeLed)  {  for(int i=1;i<=8;i++)  {  Led\_Toggle(typeLed); // if parked, toggle led Green else toggle led red  vTaskDelay(100);  }  }  bool TAG\_TCPSend(const char\* host, uint16\_t port, const char\* mdata, bool listenACK)  {  if(currentTypeCommunication == WIFI\_T)  {  return WiFi\_TCPSend(host,port,mdata,listenACK);  }  else  {  return GSM\_TCPSend(host,port,mdata,listenACK);  }  }  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ GET GPS  void GetGPS(void \*pvParameters) // This is a task.  {  vTaskSuspend(xHandle\_GetGPS);  for (;;)  {  // Serial.println("$ TASK GET GPS");  //GET GPS  GSM\_getDataGPS(&gpsData);  if(GSM\_isOkeyGPS(gpsData))  {  Led\_TurnOn(LED\_GPS\_GREEN);  Led\_TurnOff(LED\_GPS\_RED);  }  else  {  Led\_TurnOn(LED\_GPS\_RED);  Led\_TurnOff(LED\_GPS\_GREEN);  }  // Serial.print("Stack Size Get GPS: ");  // Serial.println(uxTaskGetStackHighWaterMark(NULL));  vTaskDelay(5000); // one tick delay (15ms) in between reads for stability  }  }  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ BLINK  void BlinkLed(void \*pvParameters) // This is a task.  {  //vTaskSuspend(xHandle\_BlinkLed);  for (;;)  {  // Serial.println("$ TASK BLINK LED");  Led\_Toggle(pinCurrentBlink);  // Serial.print("Stack Size Blink: ");  // Serial.println(uxTaskGetStackHighWaterMark(NULL));  vTaskDelay(250);  }  }  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ SCAN WIFI  void ScanWiFi(void \*pvParameters) // This is a task.  {  vTaskSuspend(xHandle\_ScanWiFi);  for (;;) // A Task shall never return or exit.  {  if(isConnectedWiFi(1000L))  {  Led\_TurnOn(LED\_WIFI\_GREEN);  Led\_TurnOff(LED\_LTE\_RED);  currentTypeCommunication = WIFI\_T; // nếu bắt được WIFI thì truyền gói tin theo đường WIFI  }  else  {  Led\_TurnOn(LED\_LTE\_RED);  Led\_TurnOff(LED\_WIFI\_GREEN);  currentTypeCommunication = GSM\_T; // còn không bắt được WIFI thì truyền gói tin theo đường SIM  }  // Serial.print("Stack Size Scan WiFi: ");  // Serial.println(uxTaskGetStackHighWaterMark(NULL));  vTaskDelay(5000);  }  } |