TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

& TRUYỀN THÔNG VIỆT - HÀN

KHOA KỸ THUẬT MÁY TÍNH VÀ ĐIỆN TỬ



CHUYÊN ĐỀ 4

LẬP TRÌNH Ô TÔ

ĐỀ TÀI

**THIẾT KẾ VÀ TRIỂN KHAI HỆ THỐNG HMI BẢN ĐỒ VÀ ĐIỀU HƯỚNG DÀNH CHO XE Ô TÔ**

Sinh viên thực hiện: Ngô Ngọc Sơn 21CE113

Giảng viên hướng dẫn: T.S Nguyễn Vũ Anh Quang

Giảng viên hướng dẫn: KS. Trần Viết An

*Đà Nẵng, 10 tháng 5 năm 2025*

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

& TRUYỀN THÔNG VIỆT - HÀN

KHOA KỸ THUẬT MÁY TÍNH VÀ ĐIỆN TỬ



CHUYÊN ĐỀ 4

LẬP TRÌNH Ô TÔ

ĐỀ TÀI

**THIẾT KẾ VÀ TRIỂN KHAI HỆ THỐNG HMI BẢN ĐỒ VÀ ĐIỀU HƯỚNG DÀNH CHO XE Ô TÔ**

Sinh viên thực hiện: Ngô Ngọc Sơn 21CE113

Giảng viên hướng dẫn: T.S Nguyễn Vũ Anh Quang

Giảng viên hướng dẫn: KS. Trần Viết An

*Đà Nẵng, 10 tháng 5 năm 2025*NHẬN XÉT

(Của giảng viên hướng dẫn)

……………………………………………………………………………

……………………………………………………………………………

……………………………………………………………………………

……………………………………………………………………………

……………………………………………………………………………

……………………………………………………………………………

……………………………………………………………………………

……………………………………………………………………………

……………………………………………………………………………

……………………………………………………………………………

……………………………………………………………………………

Đà Nẵng, ngày …. tháng …. năm 2025

Giảng viên hướng dẫn

**LỜI CẢM ƠN**

Chúng em xin gửi lời cảm ơn sâu sắc đến thầy Nguyễn Vũ Anh Quang và trợ giảng đã hỗ trợ và chỉ dẫn chúng em trong quá trình nghiên cứu và triển khai đề tài " Thiết kế và triển khai hệ thống HMI bản đồ và điều hướng dành cho xe ô tô".

Những lời khuyên và hướng dẫn từ anh không chỉ là nguồn động viên mà còn là nguồn cảm hứng lớn giúp chúng em vượt qua những thách thức và đạt được thành công trong đề tài này.

Chúng em cũng muốn bày tỏ lòng biết ơn đến bạn bè, gia đình và những người thân yêu đã luôn ở bên cạnh và động viên chúng em trong suốt thời gian qua. Một lần nữa, chúng em xin chân thành cảm ơn sự hỗ trợ và sự khích lệ từ thầy cô đã giúp chúng em hoàn thành đề tài này một cách thành công. Tuy nhiên với vốn kiến thức và kinh nghiệm chưa nhiều nên trong quá trình thực hiện đồ án sẽ có một số sai sót. Chúng em rất mong nhận được những nhận xét, đánh giá từ phía các thầy cô lẫn anh chị trợ giảng để bài báo cáo của chúng em được hoàn thiện một cách tốt nhất.

Em xin trân trọng cảm ơn!

**MỤC LỤC**

[MỞ ĐẦU……………………… 1](#_Toc197723625)

[CHƯƠNG 1: CƠ SỞ LÝ THUYẾT 3](#_Toc197723626)

[1. 1 Giới thiệu các linh kiện sử dụng 3](#_Toc197723627)

[1. 1. 1 Chip ESP32 wroom s3 3](#_Toc197723628)

[1. 1. 2 Màn Oled 0.96 inch i2c 4](#_Toc197723630)

[1. 1. 3 ASM1117 5](#_Toc197723632)

[1. 1. 4 Module CP2102 6](#_Toc197723634)

[1. 1. 5 Mạch Tăng Áp XL6009E1 7](#_Toc197723636)

[1. 2 Giới thiệu phần mềm sử dụng 8](#_Toc197723638)

[1. 2. 1 QT Creator 8](#_Toc197723639)

[1. 2. 2 Arduino 9](#_Toc197723641)

[CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 10](#_Toc197723643)

[2. 1 Phân tích yêu cầu hệ thống 10](#_Toc197723644)

[2. 1. 1 Yêu cầu chức năng: 10](#_Toc197723645)

[2. 1. 2 Yêu cầu phi chức năng: 10](#_Toc197723646)

[2. 2 Thiết kế hệ thống 11](#_Toc197723647)

[2. 2. 1 Kiến trúc hệ thống 11](#_Toc197723648)

[2. 2. 2 Giao diện người dùng (HMI) 12](#_Toc197723649)

[2. 2. 3 Tích hợp cảm biến và API 12](#_Toc197723650)

[2. 3 Sơ đồ khối hệ thống 13](#_Toc197723651)

[2. 4 Lưu đồ thuật toán 15](#_Toc197723653)

[2. 5 Nguyên lý hoạt động 16](#_Toc197723655)

[2. 5. 1 Nhận diện vị trí xe 16](#_Toc197723656)

[2. 5. 2 Tính toán lộ trình 16](#_Toc197723657)

[2. 5. 3 Hiển thị thông tin trên màn hình OLED 16](#_Toc197723658)

[2. 5. 4 Cập nhật và điều chỉnh lộ trình 17](#_Toc197723659)

[CHƯƠNG 3: THIẾT KẾ VÀ TRIỂN KHAI XÂY DỰNG 18](#_Toc197723660)

[3. 1 Thiết kế mạch nguyên lý 18](#_Toc197723661)

[3. 1. 1 Sơ đồ nguyên lý mạch 18](#_Toc197723662)

[3. 2 Mạch PCB và 3D MODEL 18](#_Toc197723664)

[3. 3 Kết quả giao diện Maps 19](#_Toc197723667)

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 20](#_Toc197723669)

[TÀI LIỆU THAM KHẢO 22](#_Toc197723670)

**DANH MỤC HÌNH ẢNH**

[Hình 1- 1 ESP32-S3-WROOM-1 3](#_Toc197706984)

[Hình 1- 2 Màn OLED 0.96 inch I2C 4](#_Toc197706986)

[Hình 1- 3 ASM1117 5](#_Toc197706988)

[Hình 1- 4 Module CP2102 6](#_Toc197706990)

[Hình 1- 5 Mạch tăng áp XL6009E1 7](#_Toc197706992)

[Hình 1- 6 QT Creator 8](#_Toc197707258)

[Hình 1- 7 Arduino 9](#_Toc197706996)

[Hình 2- 1 Sơ đồ khối hệ thống 12](#_Toc197707006)

[Hình 2- 2 Lưu đồ thuật toán 14](#_Toc197707008)

[Hình 3- 1 Sơ đồ nguyên lý mạch 17](#_Toc197707017)

[Hình 3- 2 Mạch PCB 17](#_Toc197707283)

[Hình 3- 3 Mạch 3D 18](#_Toc197707284)

[Hình 3- 4 Giao diện Maps 19](#_Toc197723668)

# MỞ ĐẦU

1. **Lý do chọn đề tài**

* Nhu cầu thị trường: Các hệ thống điều hướng trong xe ô tô ngày càng được sử dụng rộng rãi, với nhu cầu về sự dễ sử dụng và tích hợp tính năng thông minh như dự đoán giao thông, giúp cải thiện trải nghiệm lái xe.
* Cải thiện an toàn: Hệ thống HMI hỗ trợ lái xe an toàn hơn bằng cách cung cấp thông tin chính xác và dễ hiểu, giảm thiểu sai sót khi lái xe.
* Tích hợp công nghệ: Hệ thống có thể kết hợp với các tính năng khác trong xe thông minh như hệ thống giải trí, cảnh báo an toàn, nhưng cần chú ý bảo mật và quyền riêng tư trong việc chia sẻ dữ liệu.
* Tiềm năng nghiên cứu: Đây là cơ hội nghiên cứu và phát triển hệ thống đa lĩnh vực, nhưng cũng cần chú ý đến tính cạnh tranh cao và yêu cầu nguồn lực lớn.

1. **Mục tiêu đồ án**

Phân tích và thiết kế hệ thống HMI điều hướng

* Nghiên cứu yêu cầu người dùng: Xác định các tính năng cần thiết như hiển thị bản đồ, tìm kiếm địa điểm, chỉ dẫn đường đi và cập nhật thông tin giao thông.
* Lựa chọn công nghệ phù hợp: Đánh giá và chọn lựa các công nghệ như API bản đồ (Google Maps API), màn hình OLED, và giao diện người dùng QML để xây dựng hệ thống.

Triển khai phần mềm và phần cứng

* Phát triển ứng dụng điều hướng: Lập trình ứng dụng trên nền tảng QML để hiển thị bản đồ, nhận diện vị trí và cung cấp chỉ dẫn đường đi.
* Tích hợp phần cứng: Kết nối các thiết bị như API, màn hình OLED, và các cảm biến để thu thập và hiển thị thông tin cần thiết.

Đánh giá hiệu năng và khả năng mở rộng

* Kiểm tra tính chính xác và độ trễ: Đánh giá độ chính xác của việc xác định vị trí và độ trễ trong việc cập nhật thông tin giao thông.
* Đánh giá khả năng mở rộng: Xác định khả năng mở rộng của hệ thống để tích hợp thêm các tính năng như hỗ trợ nhiều phương tiện, cảnh báo an toàn, hoặc tích hợp với các hệ thống xe thông minh khác.

Nghiên cứu và đề xuất hướng phát triển

* Phân tích xu hướng công nghệ: Nghiên cứu các xu hướng công nghệ mới như trí tuệ nhân tạo, học máy, và xe tự lái để đề xuất các cải tiến cho hệ thống.
* Đề xuất hướng phát triển: Đưa ra các đề xuất về việc mở rộng tính năng, cải thiện hiệu suất và tích hợp với các hệ thống khác trong tương lai.

1. **Phương pháp nghiên cứu**

* Nghiên cứu lý thuyết: Tìm hiểu các công nghệ điều hướng, API, và bản đồ số hiện có, đồng thời phân tích các yêu cầu về giao diện người dùng (HMI) và tính năng cần thiết cho hệ thống.
* Thu thập yêu cầu: Xác định yêu cầu người dùng về giao diện và tính năng, thu thập dữ liệu giao thông từ các dịch vụ API như Google Maps, OpenStreetMap.
* Thiết kế hệ thống: Lên kế hoạch cấu trúc hệ thống HMI, bao gồm giao diện người dùng, các module chức năng, và quy trình xử lý dữ liệu giao thông.
* Phát triển hệ thống: Lập trình và triển khai các tính năng chính của hệ thống như kết nối API, tính toán lộ trình, và tối ưu hóa tuyến đường.
* Kiểm thử và đánh giá: Thực hiện kiểm thử để đảm bảo độ chính xác, hiệu suất và tính ổn định của hệ thống.
* Tối ưu hóa và hoàn thiện: Dựa trên kết quả kiểm thử, tối ưu hóa hệ thống và khắc phục các vấn đề kỹ thuật.

1. Bố cục báo cáo

Chương 1: Cơ sở lý thuyết

Chương 2: Phân tích thiết kế hệ thống

Chương 3: Thiết kế và triển khai xây dựng

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

TÀI LIỆU THAM KHẢO

# CƠ SỞ LÝ THUYẾT

## Giới thiệu các linh kiện sử dụng

### Chip ESP32 wroom s3

ESP32-S3-WROOM-1 là mô-đun vi điều khiển (MCU) đa năng của Espressif Systems, tích hợp Wi-Fi 2.4 GHz và Bluetooth Low Energy (BLE) 5.0. Được trang bị bộ vi xử lý dual-core Xtensa 32-bit LX7 với tốc độ tối đa 240 MHz, mô-đun này phù hợp cho các ứng dụng IoT, tự động hóa và các hệ thống yêu cầu kết nối không dây hiệu suất cao.​.​



Hình 1- 1 ESP32-S3-WROOM-1

#### Tính năng

* Hiệu suất cao: Tốc độ xử lý nhanh, phù hợp cho các ứng dụng yêu cầu tính toán và phản hồi nhanh.
* Kết nối không dây ổn định: Hỗ trợ Wi-Fi và Bluetooth LE, phù hợp cho các ứng dụng IoT và tự động hóa.
* Giao diện ngoại vi phong phú: Hỗ trợ nhiều giao diện, dễ dàng tích hợp với các cảm biến và thiết bị ngoại vi.
* Khả năng xử lý tín hiệu: Hỗ trợ các ứng dụng như nhận dạng giọng nói, nhận dạng khuôn mặt, điều khiển thông minh.
* Tiết kiệm năng lượng: Có chế độ tiết kiệm năng lượng, phù hợp với các thiết bị hoạt động liên tục.​

#### Thông số kỹ thuật

|  |  |
| --- | --- |
| Thông Số | Giá Trị |
| Bộ vi xử lý | Xtensa® 32-bit LX7 dual-core, tối đa 240 MHz |
| Bộ nhớ | 384 KB ROM, 512 KB SRAM, tối đa 8 MB PSRAM (tùy phiên bản) |
| Flash | 4 MB, 8 MB hoặc 16 MB (Quad SPI) |
| Tốc độ truyền dữ liệu | Tối đa 150 Mbps |
| Kết nối | Wi-Fi 802.11 b/g/n, Bluetooth 5.0 LE |
| Giao diện ngoại vi | |  | | --- | |  |  |  | | --- | | 36 GPIOs, hỗ trợ ADC, DAC, PWM, I2C, SPI, UART, I2S, USB OTG, CAN, SDIO, IR, cảm biến chạm | |
| Nhiệt độ hoạt động | |  | | --- | |  |  |  | | --- | | -40 ~ 85°C (phiên bản tiêu chuẩn), -40 ~ 105°C (phiên bản H4) | |

### Màn Oled 0.96 inch i2c

Màn hình Oled 0.96 giao tiếp I2C cho khả năng hiển thị đẹp, sang trọng, rõ nét vào ban ngày và khả năng tiết kiệm năng lượng tối đa với mức chi phí phù hợp, màn hình sử dụng giao tiếp I2C cho chất lượng đường truyền ổn định và rất dễ giao tiếp chỉ với 2 chân GPIO.



Hình 1- 2 Màn OLED 0.96 inch I2C

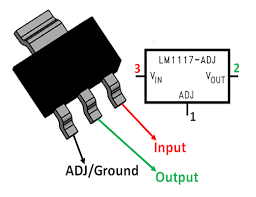
* Thông số kĩ thuật

|  |  |
| --- | --- |
| Thông Số | Giá Trị |
| Điện áp sử dụng | 2.2~5.5VDC |
| Công suất tiêu thụ | 0,04w |
| Góc hiển thị | lớn hơn 160 độ |
| Số điểm hiển thị | 128×64 điểm |
| Độ rộng màn hình | 0.96 inch |
| Giao tiếp | I2C |
| Driver | SSD1306 |

* Tính năng
* Độ phân giải 128x64 pixels: Hiển thị rõ ràng, sắc nét.
* Tiết kiệm năng lượng: Tiêu thụ ít điện năng.
* Giao tiếp I2C: Dễ dàng kết nối với vi điều khiển qua 2 dây tín hiệu.
* Độ tương phản cao, góc nhìn rộng: Hình ảnh sắc nét và dễ xem từ nhiều góc độ.
* Không cần đèn nền: Tự phát sáng, tiết kiệm không gian và năng lượng.
* Hiển thị màu đen sâu: Tạo độ tương phản tốt hơn.
* Dễ tích hợp: Phù hợp cho các dự án DIY, thiết bị điện tử nhỏ gọn.

### ASM1117

Mạch cấp nguồn DC-DC Power Supply 3.3VDC ASM1117 được sử dụng để tạo điện áp 3.3VDC / 800mA cấp cho mạch điện với đầu vào linh hoạt từ 4.5~9VDC, thường được sử dụng với các mạch module ESP8266, vi điều khiển sử dụng điện áp 3.3VDC,...



Hình 1- 3 LM1117 3.3v

* Thông số kỹ thuật

|  |  |
| --- | --- |
| Thông số | Giá trị |
| Điện áp vào | 4.5~9VDC. |
| Điện áp ra | 3.3VDC, 800mA |
| Kích thước | 25 x 11 mm |

* Tính năng
* Cung cấp dòng điện lên đến 1A và điện áp đầu ra cố định hoặc điều chỉnh (1.25V đến 12V).
* Bảo vệ ngắn mạch và quá nhiệt.
* Dễ sử dụng với ba chân (Input, Output, Ground).
* Tiết kiệm chi phí và phổ biến trong các mạch điện tử DIY.
* Ổn định cao khi hoạt động với tụ điện đầu vào và đầu ra.

### Module CP2102

Mạch chuyển USB UART CP2102 Mini được dùng để chuyển giao tiếp từ USB sang UART TTL và ngược lại, Driver của mạch có thể nhận trên tất cả các hệ điều hành hiện nay: Windows, Mac, Linux,…



Hình 1- 4 Module CP2102

* Thông số kỹ thuật

|  |  |
| --- | --- |
| Thông số | Giá trị |
| Giao tiếp USB | USB 2.0 Full-Speed (12 Mbps) |
| Giao tiếp UART | Hỗ trợ các tín hiệu RXD, TXD, RTS, CTS, DTR, DSR, DCD, RI |
| Tốc độ truyền dữ liệu | Từ 300 bps đến 1 Mbps |
| VDD (Đầu vào) | 3.0V đến 3.6V |
| VDDOUT (Đầu ra) | 3.3V |
| Nhiệt độ hoạt động | Từ -40°C đến +85°C |

* Tính năng
* Chuyển đổi USB sang UART: CP2102 giúp kết nối các thiết bị như vi điều khiển (ví dụ: Arduino, ESP32) với máy tính qua cổng USB, sử dụng giao tiếp UART (TTL - 3.3V hoặc 5V).
* Cung cấp tín hiệu điều khiển: Hỗ trợ các tín hiệu như RTS, CTS, DTR, DSR, giúp giao tiếp nối tiếp giữa các thiết bị được ổn định và dễ dàng lập trình.
* Tương thích với nhiều hệ điều hành: Module CP2102 hoạt động trên Windows, Linux, và macOS mà không cần cài đặt driver phức tạp, vì nó có hỗ trợ driver tự động.
* Dễ sử dụng và tích hợp: Với giao diện đơn giản và dễ sử dụng, CP2102 là lựa chọn phổ biến cho các dự án điện tử và lập trình viên khi cần giao tiếp nối tiếp qua USB.
* Tốc độ truyền dữ liệu cao: Hỗ trợ tốc độ truyền dữ liệu lên đến 1 Mbps, giúp truyền tải dữ liệu nhanh chóng và hiệu quả.

### Mạch Tăng Áp XL6009E1

Mạch tăng áp DC-DC Boost Converter XL6009 được sử dụng để tăng điện áp DC, hiệu suất chuyển đổi của mạch lên đến 94%, dòng đầu ra tối đa 4A (10W), là sự lựa chọn phù hợp cho các ứng dụng tăng áp.



Hình 1- 5 Mạch tăng áp XL6009E1

* Thông số kỹ thuật

|  |  |
| --- | --- |
| Thông số | Giá trị |
| Điện áp đầu vào | 3V-32VDC |
| Điện áp đầu ra | 5V-35VDC |
| Dòng tải | 3A (Max) |
| Hiệu suất | <94% |
| Tần số chuyển đổi | 400KHz |
| Nhiệt độ hoạt động | -40 ℃ ~ +85 ℃ |
| Kích thước | 43 x 21 x 14mm |

* Tính năng
* Hiệu suất cao: Giảm tổn thất năng lượng, tiết kiệm năng lượng cho các ứng dụng di động hoặc năng lượng tái tạo.
* Điều chỉnh linh hoạt: Cho phép điều chỉnh điện áp đầu ra dễ dàng thông qua biến trở tích hợp.
* Kích thước nhỏ gọn: Phù hợp với các thiết kế yêu cầu tiết kiệm không gian.
* Ứng dụng rộng rãi: Thích hợp cho các dự án DIY, nguồn cấp cho mạch Arduino, Raspberry Pi, hoặc các thiết bị điện tử khác.

## Giới thiệu phần mềm sử dụng

### QT Creator

Qt Creator là một môi trường phát triển tích hợp (IDE) đa nền tảng C++ , JavaScript , Python và QML giúp đơn giản hóa quá trình phát triển ứng dụng GUI. Nó là một phần của SDK cho khuôn khổ phát triển ứng dụng GUI Qt và sử dụng Qt API, đóng gói các lệnh gọi hàm GUI của hệ điều hành máy chủ.



Hình 1- 6 QT Creator

#### Tính năng:

* IDE mạnh mẽ: Hỗ trợ phát triển ứng dụng đa nền tảng (Windows, Linux).
* Trình soạn thảo thông minh: Hỗ trợ syntax highlighting, autocompletion và refactoring.
* Tích hợp Qt Designer: Thiết kế GUI trực quan cho ứng dụng Qt.
* Hệ thống quản lý dự án linh hoạt: Hỗ trợ qmake và CMake.
* Hỗ trợ ngôn ngữ: C++, QML
* Kiểm thử tự động: Hỗ trợ unit testing và TDD.
* Quản lý phiên bản: Tích hợp Git.

### Arduino

Arduino là một nền tảng mã nguồn mở (open-source) dùng để phát triển các ứng dụng điện tử nhúng, bao gồm phần cứng (board vi điều khiển) và phần mềm (IDE lập trình). Nó được thiết kế để giúp người dùng dễ dàng tạo ra các thiết bị tương tác với thế giới thực thông qua cảm biến, động cơ, đèn LED, và nhiều thiết bị ngoại vi khác.



Hình 1- 7 Arduino

* Tính năng:
* Phần cứng mở: Sơ đồ mạch và thiết kế board có sẵn, cho phép người dùng tùy chỉnh và phát triển thêm.​
* Phần mềm miễn phí: Arduino IDE hỗ trợ lập trình bằng ngôn ngữ C/C++, với thư viện phong phú và dễ sử dụng.​
* Giao tiếp đa dạng: Hỗ trợ nhiều giao thức như UART, SPI, I2C, PWM, ADC, giúp kết nối với nhiều loại cảm biến và thiết bị ngoại vi.​
* Tương thích đa nền tảng: Arduino IDE chạy trên Windows, macOS và Linux, hỗ trợ nhiều loại board khác nhau như Arduino Uno, Mega, Nano, ESP32, STM32.​
* Cộng đồng lớn: Nền tảng này có cộng đồng người dùng rộng lớn, chia sẻ mã nguồn, dự án và hỗ trợ kỹ thuật.

# PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## Phân tích yêu cầu hệ thống

### Yêu cầu chức năng:

Hiển thị bản đồ:

* Cung cấp giao diện bản đồ số, cho phép người dùng phóng to, thu nhỏ và di chuyển bản đồ.
* Hiển thị các địa điểm, tuyến đường và thông tin giao thông liên quan trong khu vực.

Tìm kiếm vị trí:

* Cho phép người dùng nhập tên địa điểm hoặc tọa độ GPS để tìm kiếm vị trí và hiển thị chúng trên bản đồ.
* Cung cấp chức năng tìm kiếm theo địa chỉ hoặc tên địa điểm.

Chỉ dẫn đường đi:

* Cung cấp chỉ dẫn đường đi chi tiết từ điểm xuất phát đến điểm đến.
* Tính toán và hiển thị lộ trình tối ưu, bao gồm các chỉ dẫn về rẽ trái, rẽ phải, chạy thẳng, và khoảng cách.

Điều hướng trên App QML và OLED:

* Trên App QML: Giao diện người dùng hỗ trợ cảm ứng và thao tác dễ dàng, hiển thị các chỉ dẫn điều hướng theo thời gian thực.
* Trên OLED: Hiển thị thông tin tóm tắt như thời gian còn lại, khoảng cách, và các bước tiếp theo trong lộ trình.

### Yêu cầu phi chức năng:

Hiệu suất:

* Hệ thống cần có thời gian phản hồi nhanh, xử lý các thao tác phóng to/thu nhỏ, tìm kiếm và tính toán lộ trình mượt mà, không gây độ trễ.

Độ chính xác:

* Cung cấp các chỉ dẫn và thông tin giao thông chính xác, kịp thời, đặc biệt trong việc xác định vị trí xe và tính toán lộ trình chính xác.

Độ tin cậy:

* Hệ thống cần hoạt động ổn định và không bị gián đoạn, đặc biệt là khi cung cấp thông tin về chỉ dẫn đường và tình trạng giao thông.

Bảo mật:

* Đảm bảo bảo mật thông tin người dùng và dữ liệu giao thông, tránh các nguy cơ tấn công hoặc rò rỉ thông tin.

Khả năng tương thích:

* Hệ thống cần có khả năng tương thích với các thiết bị phần cứng khác nhau, bao gồm màn hình OLED.

Dễ sử dụng:

* Giao diện người dùng phải dễ sử dụng, giúp người lái dễ dàng thao tác mà không bị phân tâm trong khi lái xe.

Khả năng mở rộng:

* Hệ thống cần có khả năng mở rộng và tích hợp với các dịch vụ bổ sung trong tương lai, như hỗ trợ trợ giúp lái xe tự động hoặc cập nhật giao thông thời gian thực từ các nguồn khác.

## Thiết kế hệ thống

### Kiến trúc hệ thống

Hệ thống được thiết kế theo mô hình phân lớp, bao gồm:

* Lớp giao diện người dùng (HMI): Cung cấp giao diện trực quan để người dùng tương tác với hệ thống, bao gồm màn hình cảm ứng và các nút điều khiển vật lý.
* Lớp xử lý ứng dụng: Xử lý các yêu cầu từ người dùng, tính toán lộ trình, cập nhật thông tin giao thông và quản lý dữ liệu bản đồ.
* Lớp dữ liệu: Lưu trữ và quản lý dữ liệu bản đồ, lộ trình, thông tin giao thông và dữ liệu người dùng.
* Lớp phần cứng: Gồm các thiết bị như API, cảm biến, màn hình hiển thị và bộ xử lý trung tâm.​

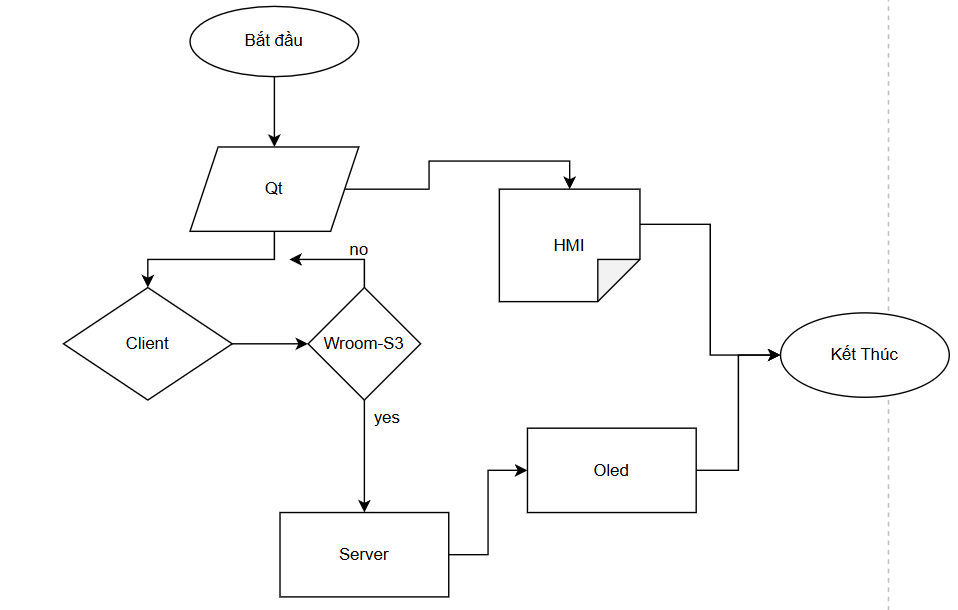
### Giao diện người dùng (HMI)

* Giao diện người dùng được thiết kế với các tiêu chí:
* Đơn giản và trực quan: Giúp người dùng dễ dàng thao tác mà không gây mất tập trung khi lái xe.
* Phản hồi nhanh: Đảm bảo hệ thống phản hồi kịp thời với các thao tác của người dùng.
* Hiển thị thông tin rõ ràng: Sử dụng màu sắc và biểu tượng phù hợp để hiển thị thông tin quan trọng.
* Tùy biến: Cho phép người dùng tùy chỉnh giao diện theo sở thích cá nhân.​

### Tích hợp cảm biến và API

* API: Cung cấp dữ liệu và dịch vụ từ bên ngoài, như bản đồ số, thông tin giao thông thời gian thực, và các dịch vụ hỗ trợ điều hướng.
* Cảm biến: Nhận dữ liệu từ các cảm biến trong xe như cảm biến tốc độ, cảm biến hướng, cảm biến gia tốc, v.v., để cung cấp thông tin bổ sung và hỗ trợ tính toán lộ trình.

## Sơ đồ khối hệ thống

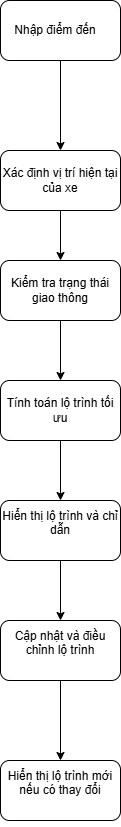


Hình 2- 1 Sơ đồ khối hệ thống

1. Quy trình khởi đầu tại điểm "Bắt đầu" và chuyển sang thành phần code trong Qt.
2. Qt là một framework phát triển ứng dụng đa nền tảng, đóng vai trò xử lý chính trong quy trình này. Từ đây, luồng chia làm hai nhánh:
   * Một nhánh kết nối trực tiếp đến HMI (Human Machine Interface)
   * Nhánh còn lại chuyển đến thành phần Client
3. HMI đại diện cho giao diện người dùng, nơi người dùng tương tác với hệ thống.
4. Client kết nối đến điểm quyết định Wroom-S3 làm vi xử lí.
5. Tại điểm Wroom-S3:
   * Nếu kết quả là "no", luồng quay trở lại Qt để xử lý lại.
   * Nếu kết quả là "yes", luồng tiếp tục đến Server để khởi tạo API Server
6. Server thực hiện xử lý dữ liệu hoặc yêu cầu từ Client và kết nối đến thành phần Oled.
7. Oled (màn hình OLED) hiển thị thông tin hoặc kết quả xử lý từ Server.
8. Cuối cùng, cả hai nhánh từ HMI và Oled đều dẫn đến "Kết Thúc", hoàn tất quy trình xử lý.

Sơ đồ thể hiện kiến trúc một hệ thống IoT hoặc hệ thống điều khiển với các thành phần giao diện người dùng (Qt, HMI), kết nối không dây (Wroom-S3), xử lý (Server) và hiển thị (Oled).

## Lưu đồ thuật toán



Hình 2- 2 Lưu đồ thuật toán

Bước 1: Nhập điểm đến: Người dùng nhập điểm đến cần di chuyển.

Bước 2: Xác định vị trí hiện tại của xe: Hệ thống xác định vị trí hiện tại thông qua API.

Bước 3: Kiểm tra trạng thái giao thông: Kiểm tra tình trạng giao thông trên tuyến đường, như ùn tắc hoặc tai nạn.

Bước 4: Tính toán lộ trình tối ưu: Dựa trên thông tin giao thông và vị trí hiện tại, hệ thống tính toán lộ trình tối ưu.

Bước 5: Hiển thị lộ trình và chỉ dẫn: Lộ trình và các chỉ dẫn được hiển thị cho người lái xe.

Bước 6: Cập nhật và điều chỉnh lộ trình: Hệ thống sẽ liên tục cập nhật thông tin giao thông và điều chỉnh lộ trình khi cần thiết.

Bước 7: Hiển thị lộ trình mới nếu có thay đổi: Nếu có sự thay đổi về giao thông, hệ thống sẽ cập nhật và hiển thị lộ trình mới cho người lái xe.

## Nguyên lý hoạt động

### Nhận diện vị trí xe

* Module GPS: Nhận tín hiệu từ các vệ tinh để xác định vị trí hiện tại của xe. Dữ liệu vĩ độ và kinh độ được truyền đến ESP32-WROOM-1 qua giao thức UART.​
* Xử lý dữ liệu: ESP32-WROOM-1 xử lý dữ liệu GPS để xác định vị trí chính xác và chuẩn bị cho việc hiển thị trên màn hình LCD.​

### Tính toán lộ trình

* Nhập điểm đến: Người lái xe nhập điểm đến thông qua giao diện người dùng (có thể là màn hình cảm ứng hoặc nút bấm).​
* Thuật toán định tuyến: Hệ thống sử dụng thuật toán định tuyến (ví dụ: A\* hoặc Dijkstra) để tính toán lộ trình tối ưu từ vị trí hiện tại đến điểm đến, dựa trên dữ liệu bản đồ số.

### Hiển thị thông tin trên màn hình OLED

* Bản đồ và chỉ dẫn: Màn hình OLED 0.96 inch hiển thị bản đồ khu vực xung quanh, vị trí hiện tại của xe, và các chỉ dẫn đường đi như "rẽ phải", "tiếp tục đi thẳng", v.v.​
* Phản hồi người dùng: Hệ thống cung cấp phản hồi trực quan và âm thanh để hỗ trợ người lái trong việc theo dõi và thực hiện các chỉ dẫn.​

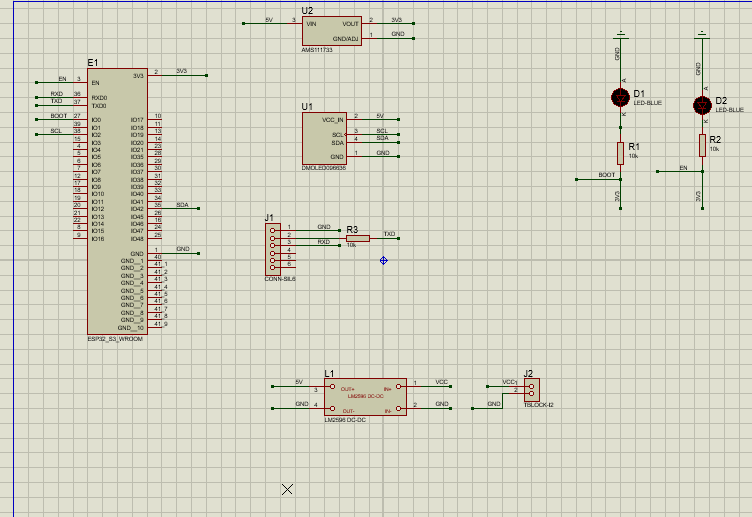
### Cập nhật và điều chỉnh lộ trình

* Cập nhật vị trí: Dữ liệu AOU được cập nhật liên tục, cho phép hệ thống theo dõi vị trí xe theo thời gian thực.​
* Điều chỉnh lộ trình: Khi có thay đổi về điều kiện giao thông hoặc khi người lái xe thay đổi điểm đến, hệ thống tự động tính toán và cung cấp lộ trình mới.

# THIẾT KẾ VÀ TRIỂN KHAI XÂY DỰNG

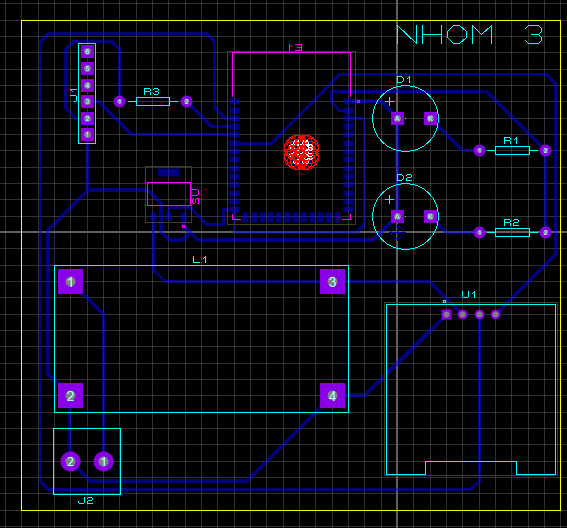
## Thiết kế mạch nguyên lý

### Sơ đồ nguyên lý mạch

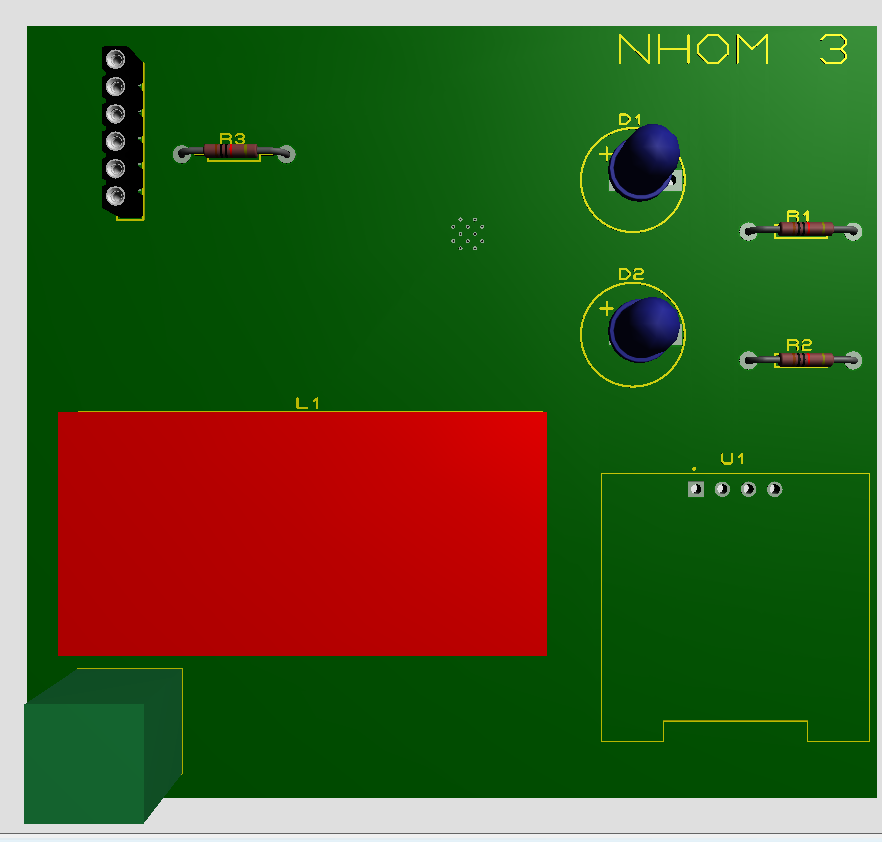


Hình 3- 1 Sơ đồ nguyên lý mạch

## Mạch PCB và 3D MODEL

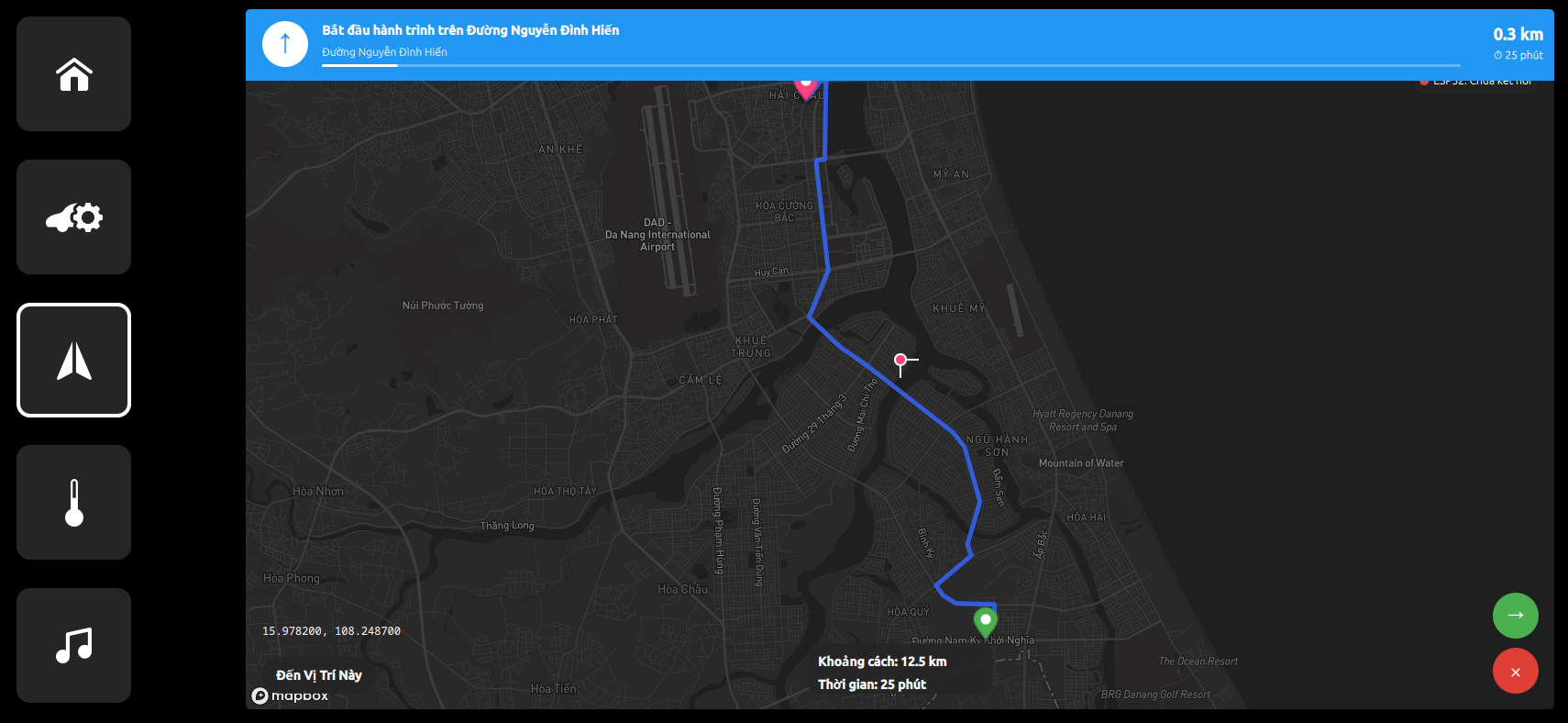


Hình 3- 2 Mạch PCB



Hình 3- 3 Mạch 3D

## Kết quả giao diện Maps



Hình 3- 4 Giao diện Maps

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

**Kết luận**

Đề tài "Thiết kế và triển khai hệ thống HMI bản đồ và điều hướng dành cho xe ô tô" đã xây dựng một hệ thống điều hướng thông minh, tích hợp các tính năng như hiển thị bản đồ, tìm kiếm vị trí, chỉ dẫn đường đi, và cập nhật thông tin giao thông thời gian thực. Hệ thống sử dụng API bản đồ từ các dịch vụ nổi bật như Google Maps API, HERE Maps API, hoặc OpenStreetMap API, giúp cung cấp thông tin chính xác về các tuyến đường, vị trí, và tình trạng giao thông, từ đó hỗ trợ người lái xe trong việc di chuyển an toàn và hiệu quả.

Cấu trúc hệ thống gồm các thành phần chính như API, vi xử lý (ESP), giao diện người dùng (GUI) và màn hình OLED. Các chức năng như chỉ dẫn đường, tính toán lộ trình tối ưu và cập nhật giao thông đã được triển khai và hoạt động ổn định, giúp người lái xe dễ dàng theo dõi và điều chỉnh lộ trình.

**Hướng phát triển**

Tích hợp các dịch vụ giao thông thời gian thực:

* Phát triển hệ thống để tích hợp thêm các dịch vụ API giao thông từ các nền tảng như Waze hoặc TomTom, giúp người lái xe nhận thông tin về tình trạng giao thông, tai nạn, và các sự kiện giao thông ảnh hưởng đến lộ trình. Điều này sẽ giúp tối ưu hóa lộ trình không chỉ về thời gian mà còn về điều kiện giao thông thực tế.

Tính năng học máy (Machine Learning):

* Học máy có thể được tích hợp để phân tích dữ liệu giao thông lịch sử, giúp dự đoán tình trạng giao thông và tự động điều chỉnh lộ trình một cách thông minh. Hệ thống có thể học hỏi từ thói quen của người lái và dự báo các điểm ùn tắc tiềm ẩn trong suốt hành trình.
* Tính năng hỗ trợ xe tự lái:
* Phát triển tính năng trợ giúp lái xe tự động bằng cách tích hợp hệ thống này với các hệ thống điều khiển xe tự động. API có thể cung cấp thông tin về lộ trình, giao thông và các yếu tố an toàn, hỗ trợ khả năng điều khiển tự động cho xe.
* Cải thiện giao diện người dùng (UI/UX):
* Giao diện người dùng có thể được cải tiến với các tính năng như thông báo bằng giọng nói, giúp người lái không cần nhìn vào màn hình khi điều hướng. Điều này sẽ nâng cao độ an toàn khi lái xe, giảm thiểu sự phân tâm.

Hỗ trợ nhiều phương tiện giao thông:

* Mở rộng hệ thống để hỗ trợ nhiều loại phương tiện khác nhau như xe đạp, xe buýt, đi bộ, v.v., giúp hệ thống trở nên linh hoạt hơn và phục vụ cho nhiều nhu cầu di chuyển khác nhau.

Tích hợp với các hệ thống xe thông minh:

* Phát triển hệ thống để tích hợp với các công nghệ thông minh trong xe, chẳng hạn như hệ thống giải trí, hệ thống cảnh báo va chạm, và hệ thống hỗ trợ lái xe. Hệ thống có thể chia sẻ dữ liệu với các thiết bị khác trong xe, tạo ra một trải nghiệm điều khiển và điều hướng toàn diện.

Tối ưu hóa hiệu suất hệ thống:

* Hệ thống cần được tối ưu hóa về hiệu suất, giảm thiểu độ trễ trong việc hiển thị bản đồ và cập nhật lộ trình, đồng thời xử lý nhanh chóng các thay đổi về thông tin giao thông.

Tăng cường bảo mật:

* Hệ thống cần được phát triển với các biện pháp bảo mật mạnh mẽ, bảo vệ dữ liệu người dùng và thông tin giao thông tránh bị xâm nhập hoặc lạm dụng. Các phương pháp như mã hóa dữ liệu và xác thực mạnh mẽ sẽ giúp bảo vệ người dùng và hệ thống.

# TÀI LIỆU THAM KHẢO

[1] <https://developers.google.com/maps>.

[2][https://www.researchgate.net/publication/265933115\_An\_Automotive\_HMI\_Architecture\_Based\_on\_a\_Mobile\_Operating\_System](https://www.researchgate.net/publication/265933115_An_Automotive_HMI_Architecture_Based_on_a_Mobile_Operating_System%20). [3]<https://www.autoweek.com/article/car-news/8-ways-hmi-design-can-improve-driving>.