

TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI
KHOA ĐIỆN – ĐIỆN TỬ



BÁO CÁO TỔNG KẾT

ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN
NĂM HỌC 2023-2024

NGHIÊN CỨU THIẾT KẾ HỆ THỐNG CẢNH BÁO
ĐẢM BẢO AN TOÀN TRÊN XE ĐƯA ĐÓN HỌC SINH

Sinh viên thực hiện: **Lê Văn Minh**
 Đỗ Văn Huy
 Trịnh Văn Mạnh
 Phạm Đăng Khôi
 Nguyễn Văn Cường

Lớp: Kỹ thuật điện tử viễn thông 1 – K62

Khoa: Điện - Điện tử

Người hướng dẫn: TS . TRẦN VĂN HÙNG

HÀ NỘI, 2024

TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI
KHOA ĐIỆN – ĐIỆN TỬ



BÁO CÁO TỔNG KẾT

ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN
NĂM HỌC 2023-2024

NGHIÊN CỨU THIẾT KẾ HỆ THỐNG CẢNH BÁO ĐẢM
BẢO AN TOÀN TRÊN XE ĐƯA ĐÓN HỌC SINH

Sinh viên thực hiện: **Lê Văn Minh**
 Đỗ Văn Huy
 Trịnh Văn Mạnh
 Phạm Đăng Khôi
 Nguyễn Văn Cường

Lớp: Kỹ thuật điện tử viễn thông 1 – K62
Khoa: Điện - Điện tử

Người hướng dẫn: TS . **TRẦN VĂN HƯNG**

HÀ NỘI, 2024

MỤC LỤC

MỤC LỤC	1
DANH MỤC HÌNH ẢNH	3
DANH MỤC CÁC BẢNG.....	5
MỞ ĐẦU	1
CHƯƠNG I. GIỚI THIỆU	2
1.1 Tổng quan tình hình nghiên cứu thuộc lĩnh vực của đề tài	2
1.2 Lý do chọn đề tài	3
1.3 Mục tiêu nghiên cứu	3
1.4 Nội dung nghiên cứu.....	4
1.5 Đối tượng và phạm vi nghiên cứu	4
1.6 Giới hạn đề tài.....	5
1.7 Cấu trúc báo cáo	5
CHƯƠNG II. CƠ SỞ LÝ THUYẾT	7
2.1 Giới thiệu hệ thống cảnh báo trẻ em bị bỏ quên trên xe đưa đón.....	7
2.2 Các thiết bị và linh kiện điện tử xây dựng hệ thống	8
2.2.1 Hệ thống các cảm biến.....	9
2.2.2 Board Arduino UNO R3	9
2.2.3 Module SIM 800A.....	11
2.2.4 ESP 32.....	13
2.2.5 Màn hình OLED 1.3 inch	14
2.2.6 Mạch Định Vị GPS NEO-6M V2.....	15
2.3 Các chuẩn giao tiếp.....	16
2.3.1 Chuẩn giao tiếp UART	16
2.3.2 Chuẩn giao tiếp I2C	17
2.4 Kết luận chương.....	18

CHƯƠNG III. THIẾT KẾ HỆ THỐNG	19
3.1 Chức năng của hệ thống thiết kế.....	19
3.2 Tính toán và thiết kế phần cứng hệ thống.....	19
3.2.1 Thiết kế sơ đồ khối hệ thống	19
3.2.2 Tính toán và thiết kế mạch.....	20
3.3 Xây dựng phần mềm cho hệ thống	28
3.3.1 Phân tích yêu cầu của phần mềm cho hệ thống.....	28
3.3.2 Các công cụ sử dụng để xây dựng phần mềm hệ thống	28
3.3.3 Xây dựng các lưu đồ giải thuật.....	32
3.4 Kết luận chương.....	36
CHƯƠNG IV. CÁC KẾT QUẢ VÀ ĐÁNH GIÁ	37
4.1 Mô hình sản phẩm của hệ thống	37
4.1.1 Lắp ráp và kiểm tra hệ thống	37
4.1.2 Sản phẩm mô hình hệ thống	40
4.2 Tài liệu hướng dẫn sử dụng	45
4.3 Các kết quả và đánh giá	46
4.3.1 Các kết quả đạt được.....	46
4.3.2 Thử nghiệm hệ thống.....	46
4.3.3 Đánh giá.....	47
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	48
A. Kết luận	48
B. Hướng phát triển	48
TÀI LIỆU THAM KHẢO	49
PHỤ LỤC.....	50

DANH MỤC HÌNH ẢNH

Hình 2. 1 Sơ đồ khối chung của hệ thống cảnh báo trẻ em bị bỏ quên trên xe.....	7
Hình 2. 2 Giải pháp đưa đón học sinh theo thời gian thực và ứng dụng di động của công ty PHX Smart School.....	8
Hình 2. 3 Board Arduino UNO R3	10
Hình 2. 4 Module SIM 800A.....	11
Hình 2. 5 ESP32.....	14
Hình 2. 6 Mạch định vị GPS NEO-6M V2.....	16
Hình 2. 7 Giản đồ truyền dữ liệu UART.....	17
Hình 2. 8 Giao tiếp I2C	17
Hình 3. 1 Sơ đồ khối hệ thống	19
Hình 3. 2 Khối xử lý trung tâm Arduino UNO R3.....	20
Hình 3. 3 Khối module sim 800A kết nối với UNO	22
Hình 3. 4 Khối module ESP32 kết nối với Oled và UNO.....	23
Hình 3. 5 Khối cảm biến kết nối với UNO.....	24
Hình 3. 6 Khối nút nhấn kết nối với UNO	25
Hình 3. 7 Module GPS kết nối ESP32	26
Hình 3. 8 Khối cảnh báo kết nối với UNO.....	27
Hình 3. 9 Sơ đồ toàn mạch	27
Hình 3. 10 Biên dịch trong Java.....	30
Hình 3. 11 Giao diện phần mềm lập trình Arduino IDE	32
Hình 3. 12 Lưu đồ đăng nhập hoặc đăng ký tài khoản.....	33
Hình 3. 13 Lưu đồ cập nhật dữ liệu từ cảm biến lên app	34

<i>Hình 3. 14 Lưu đồ thực hiện cuộc gọi thông báo tới tài xế.....</i>	<i>35</i>
<i>Hình 4. 1 Tiến hành gắn các linh kiện vào mô hình xe oto</i>	<i>37</i>
<i>Hình 4. 2 Kết nối các linh kiện với nhau</i>	<i>38</i>
<i>Hình 4. 3 Module sim 800A sau khi được cấp nguồn.....</i>	<i>38</i>
<i>Hình 4. 4 Uno sau khi được cấp nguồn và chạy thử</i>	<i>39</i>
<i>Hình 4. 5 Hình ảnh thực tế của module GPS khi được cấp nguồn.....</i>	<i>40</i>
<i>Hình 4. 6 Hình ảnh thực tế của module Sim 800A khi được cấp nguồn</i>	<i>41</i>
<i>Hình 4. 7 Sản phẩm khi chưa đóng gói</i>	<i>41</i>
<i>Hình 4. 8 Sản phẩm khi được cấp nguồn và chạy thử.....</i>	<i>42</i>
<i>Hình 4. 9 Sản phẩm khi được gắn hoàn chỉnh vào mô hình.....</i>	<i>42</i>
<i>Hình 4. 10 Đèn cảnh báo bên ngoài xe sáng lên khi trên xe vẫn còn học sinh....</i>	<i>43</i>
<i>Hình 4. 11 Cuộc gọi được module sim thực hiện để thông báo</i>	<i>43</i>
<i>Hình 4. 12 Giao diện đăng nhập của ứng dụng theo dõi</i>	<i>44</i>
<i>Hình 4. 13 Giao diện sau khi đăng nhập thành công</i>	<i>45</i>

DANH MỤC CÁC BẢNG

Bảng 2. 1 Các tham số cơ bản của Arduino UNO R3	10
Bảng 2. 2 Thông số kỹ thuật Module SIM800A	12
Bảng 2. 3 Chức năng các chân của Module SIM800A.....	12
Bảng 2. 4 Thông số màn hình Oled 1.3 inch	14
Bảng 4. 1 Thời gian số học sinh được cập nhật lên App (số liệu với 10 lần chạy thử)	47
Bảng 4. 2 Thời gian delay cho đến khi nhận được cuộc gọi (số liệu với 10 lần chạy thử)	47

MỞ ĐẦU

Trong thời gian gần đây, việc đưa đón học sinh bằng xe đưa đón đã trở thành một phần quan trọng trong hệ thống giáo dục. Tuy nhiên, sự việc đáng tiếc về học sinh tử vong do bị bỏ quên trên xe đưa đón đã đẩy mạnh nhu cầu và sự quan tâm đến việc tìm kiếm các giải pháp công nghệ hiện đại để ngăn chặn các sự cố tương tự và tạo ra một môi trường an toàn hơn cho việc đi lại của học sinh.

Trong dự thảo Quy chuẩn kỹ thuật quốc gia về chất lượng an toàn kỹ thuật và bảo vệ môi trường đối với xe ô tô, Bộ Giao thông vận tải đang lấy ý kiến góp ý (sửa đổi QCVN 09:2015/BGTVT) đã đưa ra các yêu cầu kỹ thuật riêng cho xe đưa đón học sinh. Theo đó, xe phải được sơn màu vàng đậm bên ngoài, phải có hệ thống còi báo động, âm thanh khẩn cấp hoặc liên lạc trực tiếp đến lái xe để cảnh báo khi có học sinh bị bỏ quên trên xe trong thời gian không quá 15 phút. Đề tài “Nghiên cứu thiết kế hệ thống cảnh báo an toàn trên xe đưa đón học sinh” là một vấn đề có ý nghĩa về mặt xã hội, cần được nghiên cứu để đưa ra các giải pháp hiệu quả đảm bảo an toàn cho học sinh trong quá trình đi lại.

Việc áp dụng công nghệ vào xe đưa đón học sinh không chỉ giúp tăng cường khả năng phát hiện các tình huống nguy hiểm mà còn tạo điều kiện thuận lợi cho việc giám sát và quản lý các xe trong hệ thống vận chuyển học đường. Đề tài này không chỉ là sự kết hợp giữa nhu cầu thực tiễn và sự phát triển của công nghệ, mà còn là một cơ hội để góp phần vào việc tạo ra một môi trường học tập an toàn và thú vị cho các em học sinh. Nước ta đang trong giai đoạn phát triển về nhiều mặt, đặc biệt là lĩnh vực giáo dục. Nhiều trường học Quốc tế được lập ra để đáp ứng nhu cầu học tập. Việc áp dụng phương pháp sử dụng xe đưa đón học sinh đem lại nhiều lợi ích, như giảm thời gian cho phụ huynh về việc đưa đón, giúp học sinh đi học đầy đủ, hạn chế kẹt xe khi tan học vào giờ cao điểm. Tuy nhiên, song song với những tích cực đó, còn tồn tại nhiều hạn chế, do số lượng học sinh nhiều nên người trông coi, giám hộ sẽ chủ quan, thiếu ý thức dẫn đến tình trạng bỏ quên học sinh trên xe đưa đón

CHƯƠNG I. GIỚI THIỆU

1.1 Tổng quan tình hình nghiên cứu thuộc lĩnh vực của đề tài

Nước ta đang trong giai đoạn phát triển về nhiều mặt: kinh tế, khoa học - công nghệ, ... đặc biệt là lĩnh vực giáo dục. Nhiều trường học Quốc tế được lập ra để đáp ứng nhu cầu học tập. Để có thể hỗ trợ cho phụ huynh về vấn đề đưa đón học sinh, nhiều trường đã áp dụng phương pháp sử dụng xe đưa đón học sinh. Điều đó đem lại rất nhiều tích cực như là giảm thời gian cho các bậc phụ huynh về việc đưa đón, giúp học sinh đi học đầy đủ, hạn chế kẹt xe khi tan học vào giờ cao điểm, ... Song song với những tích cực đó còn tồn tại nhiều hạn chế, do số lượng học sinh nhiều nên những người trông coi, giám hộ sẽ chủ quan, thiếu ý thức dẫn đến tình trạng bỏ quên học sinh trên xe đưa đón.

Trong khi nghiên cứu về hệ thống cảnh báo an toàn trên xe đưa đón học sinh đang phát triển, thị trường cũng đang trở nên ngày càng chú ý đến các giải pháp an toàn này. Với nhu cầu ngày càng tăng về an toàn giao thông và sự quan tâm đặc biệt đến việc bảo vệ học sinh trong quá trình đi lại, thị trường hệ thống cảnh báo an toàn trên xe đưa đón học sinh đang trở thành một phần không thể thiếu trong ngành công nghiệp vận tải và giáo dục.

Hướng phát triển của thị trường này tập trung vào việc cung cấp các giải pháp tích hợp và thông minh hơn. Các công ty và nhà sản xuất đang nỗ lực để phát triển các hệ thống cảnh báo an toàn trên xe đưa đón học sinh với tính linh hoạt cao, khả năng tương tác với các hệ thống quản lý trường học và cung cấp thông tin chi tiết và kịp thời cho phụ huynh và cơ quan quản lý. Đồng thời, các giải pháp mới cũng hướng đến việc sử dụng công nghệ tiên tiến như trí tuệ nhân tạo (AI), học máy, và Internet of Things (IoT) để nâng cao hiệu suất và tính hiệu quả của hệ thống.

- Công nghệ được sử dụng phổ biến trong các hệ thống cảnh báo an toàn trên xe đưa đón học sinh bao gồm:
- Công nghệ GPS: Sử dụng để theo dõi vị trí xe và học sinh trong thời gian thực.
- Cảm biến an toàn: Bao gồm cảm biến nhiệt độ, cảm biến cửa, cảm biến chuyển động, giúp phát hiện các tình huống nguy hiểm.
- Công nghệ GSM/GPRS: Được sử dụng để truyền dữ liệu vị trí và thông tin cảnh báo từ xe đến các hệ thống quản lý và phụ huynh.

- Trí tuệ nhân tạo (AI) và học máy: Sử dụng để phân tích dữ liệu và dự đoán các tình huống nguy hiểm, giúp cải thiện hiệu suất của hệ thống cảnh báo.

Tóm lại, thị trường hệ thống cảnh báo an toàn trên xe đưa đón học sinh đang phát triển với tốc độ nhanh chóng, tập trung vào việc cung cấp các giải pháp thông minh và tích hợp, sử dụng công nghệ tiên tiến để đảm bảo an toàn tối đa cho học sinh trong môi trường giao thông.

1.2 Lý do chọn đề tài

Ngay khi xảy ra trường hợp học sinh tử vong do bị bỏ quên trên xe đưa đón của nhà trường vào tháng 8-2019 ở Hà Nội, Bộ Giáo dục và Đào tạo, UBND thành phố Hà Nội đã chỉ đạo Sở Giáo dục và Đào tạo, các nhà trường tăng cường các giải pháp bảo đảm an toàn cho học sinh khi sử dụng dịch vụ đưa đón bằng xe ô tô. Sự kiện này đã đẩy mạnh nhu cầu và sự quan tâm đến việc tìm kiếm các giải pháp công nghệ hiện đại, như cảm biến an toàn, để ngăn chặn các sự cố tương tự và tạo ra một môi trường an toàn hơn cho việc đi lại của học sinh.

Trong dự thảo Quy chuẩn kỹ thuật quốc gia về chất lượng an toàn kỹ thuật và bảo vệ môi trường đối với xe ô tô mà Bộ Giao thông vận tải đang lấy ý kiến góp ý (sửa đổi QCVN 09:2015/BGTVT) có đưa nội dung: Xe đưa đón học sinh phải đáp ứng những yêu cầu kỹ thuật riêng, như bên ngoài xe phải sơn màu vàng đậm; có hệ thống còi báo động, âm thanh khẩn cấp hoặc liên lạc trực tiếp đến lái xe để cảnh báo khi có học sinh bị bỏ quên trên xe trong thời gian không quá 15 phút.

Vì vậy đề tài “**Nghiên cứu thiết kế hệ thống cảnh báo an toàn trên xe đưa đón học sinh**” là vấn đề có ý nghĩa về mặt xã hội, cần được nghiên cứu để có các giải pháp hiệu quả đảm bảo an toàn cho học sinh trên xe. Việc áp dụng công nghệ này vào xe đưa đón học sinh không chỉ giúp tăng cường khả năng phát hiện các tình huống nguy hiểm mà còn tạo điều kiện thuận lợi cho việc giám sát và quản lý các xe trong hệ thống vận chuyển học đường. Đề tài nghiên cứu thiết kế hệ thống cảnh báo an toàn trên xe đưa đón học sinh không chỉ là sự kết hợp giữa nhu cầu thực tiễn và sự phát triển của công nghệ mà còn là một cơ hội để góp phần vào việc tạo ra một môi trường học tập an toàn và thú vị cho các em học sinh.

1.3 Mục tiêu nghiên cứu

Mục đích chính của đề tài là xây dựng được mô hình hệ thống cảnh báo để đảm bảo an toàn trên xe đưa đón học sinh.

- Nghiên cứu mô hình của hệ thống cảnh báo
- Xây dựng mô hình hệ thống cảnh báo: phần cứng và phần mềm
- Triển khai hệ thống và đánh giá

1.4 Nội dung nghiên cứu

- Đọc và tìm hiểu các nguồn tài liệu
- Tìm hiểu kết nối giữa Arduino với module Sim 800A, cảm biến hồng ngoại, module GPS và ESP 32, ESP 32 và UNO R3
- Viết chương trình cho khối điều khiển Arduino, khối module Sim 800A, khối cảm biến hồng ngoại và UNO R3
- Kết nối các khối lại với nhau
- Chạy thử nghiệm sản phẩm
- Viết báo cáo

1.5 Đối tượng và phạm vi nghiên cứu

Trong lĩnh vực nghiên cứu về hệ thống cảnh báo an toàn trên xe đưa đón học sinh, phạm vi của dự án tập trung vào việc phát triển và thử nghiệm một hệ thống có mục đích cảnh báo khi có trẻ em bị bỏ quên trên xe và tự động gọi điện báo cho lái xe. Hệ thống này sẽ sử dụng các thiết bị như ESP32 để truyền dữ liệu lên ứng dụng, Arduino Uno R3, cảm biến hồng ngoại, module SIM800A và mạch định vị GPS để theo dõi hành trình của xe.

Đối tượng chính của nghiên cứu là học sinh và lái xe đưa đón. Hệ thống sẽ được thiết kế để phát hiện sự hiện diện của trẻ em trong xe và tự động kích hoạt cảnh báo khi phát hiện trẻ em bị bỏ quên. Điều này đòi hỏi tích hợp cảm biến hồng ngoại để xác định sự có mặt của trẻ em trong xe và cảm biến nhiệt độ để phân biệt giữa trẻ em và các vật phẩm khác.

Hệ thống sẽ sử dụng module SIM800A để gửi thông báo cảnh báo đến lái xe thông qua điện thoại di động. Arduino Uno R3 sẽ là bộ điều khiển trung tâm của hệ thống, điều này giúp xử lý thông tin từ các cảm biến và kích hoạt hành động cảnh báo khi cần thiết. ESP32 sẽ được sử dụng để truyền dữ liệu về trạng thái của hệ thống lên ứng dụng trên điện thoại di động của lái xe, cho phép họ theo dõi và quản lý tình trạng an toàn của xe đưa đón.

Nghiên cứu này nhằm mục đích phát triển một hệ thống cảnh báo an toàn trên xe đưa đón học sinh, tập trung vào việc phát hiện và cảnh báo khi có trẻ em bị bỏ quên trên xe. Việc sử dụng các thiết bị như ESP8266, Arduino Uno R3, cảm biến hồng ngoại, module SIM800A và mạch định vị GPS sẽ giúp tạo ra một hệ thống hoạt động hiệu quả, giúp nâng cao mức độ an toàn cho học sinh khi sử dụng dịch vụ đưa đón.

1.6 Giới hạn đề tài

- **Thời gian và nguồn lực:** Do hạn chế về thời gian và nguồn lực, nghiên cứu sẽ tập trung vào việc phát triển một mô hình cơ bản của hệ thống cảm biến cảnh báo an toàn. Điều này có thể dẫn đến việc không thể thử nghiệm tất cả các tình huống giao thông thực tế hoặc không thể thử nghiệm hệ thống trên nhiều loại xe đưa đón học sinh khác nhau.
- **Hạn chế kỹ thuật:** Mặc dù cố gắng thiết kế và triển khai một hệ thống cảm biến cảnh báo an toàn có hiệu suất cao, nhưng có thể gặp phải một số hạn chế kỹ thuật như giới hạn của các linh kiện điện tử, độ chính xác của cảm biến, hoặc khả năng tích hợp giữa các thành phần phần cứng và phần mềm.

1.7 Cấu trúc báo cáo

Với đề tài thiết kế hệ thống cảnh báo an toàn trên xe đưa đón học sinh thì bố cục báo cáo như sau:

Chương 1: Giới thiệu

Trình bày lý do chọn đề tài, mục tiêu, nội dung nghiên cứu, các giới hạn, của đề tài.

Chương 2: Cơ sở lý thuyết

Trình bày các lý thuyết liên quan đến các vấn đề mà đề tài sẽ dùng để thực hiện thiết kế, thi công cho đề tài, bao gồm các thông số, hình ảnh các linh kiện sử dụng trong đề tài.

Chương 3: Thiết kế hệ thống

Trình bày sơ đồ khối, chức năng từng khối. Tính toán và thiết kế từng khối để có thể vẽ sơ đồ nguyên lý toàn mạch.

Trình bày quá trình lắp đặt và đóng gói phần cứng, các ứng dụng sử dụng lập trình phần mềm điều khiển và thiết bị phần cứng.

Chương 4: Các kết quả và đánh giá

Trình bày kết quả của cả quá trình nghiên cứu, thời gian thực hiện và kiến thức trong suốt thời gian nhận được trong quá trình làm đề tài nghiên cứu khoa học sinh viên và cuối cùng là đưa ra nhận xét, đánh giá khách quan của bản thân về sản phẩm tự tay mình làm.

Kết luận và hướng phát triển

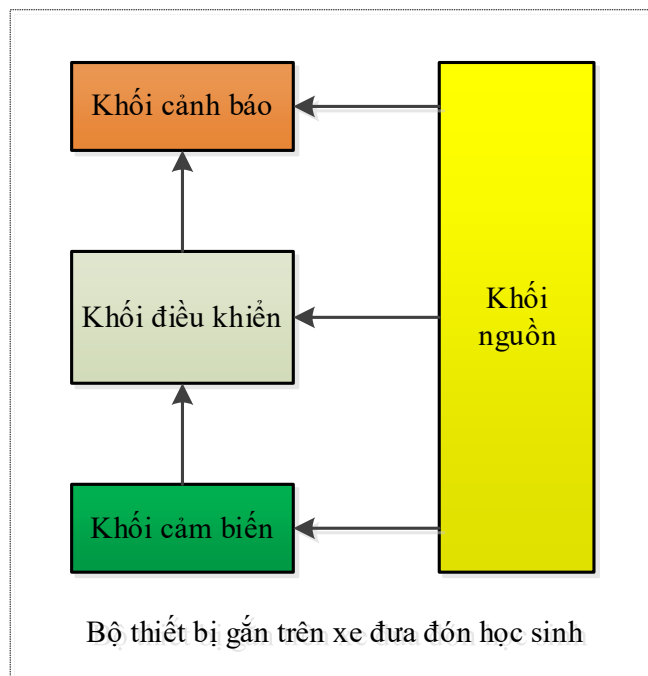
Trình bày kết quả thu được dựa vào phương pháp và mục đích ban đầu, các việc đã và chưa làm được.

CHƯƠNG II. CƠ SỞ LÝ THUYẾT

2.1 Giới thiệu hệ thống cảnh báo trẻ em bị bỏ quên trên xe đưa đón

Hiện tượng bỏ quên trẻ em trên xe đưa đón ra nếu xảy ra có nguy cơ để lại hậu quả rất nghiêm trọng. Điều này đã xảy ra trên thế giới cũng như ở Việt Nam. Những nguyên nhân dẫn đến tình trạng trên liên quan đến ý thức và trách nhiệm của người trông coi, giám hộ trẻ em trên xe đưa đón. Để nâng cao tính năng an toàn khi có trẻ em trên xe đưa đón học sinh, giải quyết triệt để tình trạng trẻ em bị bỏ quên trên xe đưa đón, tích hợp nhiều tiện ích hỗ trợ chủ phương tiện nhưng vẫn đảm bảo mức phí phù hợp với thị trường là vấn đề rất có ý nghĩa về mặt xã hội.

Sơ đồ khối chung của hệ thống cảnh báo như sau:



Hình 2. 1 Sơ đồ khối chung của hệ thống cảnh báo trẻ em bị bỏ quên trên xe

Hệ thống có bộ điều khiển trung tâm để nhận các tín hiệu cảm biến phát hiện trẻ em bị bỏ quên trên xe thông qua khối cảm biến. Bộ điều khiển trung tâm xử lý các thông tin cần thiết rồi đưa tín hiệu điều khiển sang khối cảnh báo. Khối cảnh báo đưa ra các cảnh báo cần thiết để giúp tài xế hoặc người bên ngoài xe đưa đón phát hiện ra trẻ em bị bỏ quên trên xe.

Hệ thống phát hiện trẻ em bị bỏ quên trên xe với các tín hiệu cảnh báo cho tài xế thông thường gồm có các tính năng tự động kích hoạt khi xe ngừng hoạt động, tài

xế xuống xe, cảnh báo khi phát hiện có trẻ em trong xe. Việc cảnh báo giúp tài xế phát hiện có trẻ em trong xe thông qua âm thanh còi báo, hoặc đèn cảnh báo giúp người xung quanh chú ý, có thể điều khiển hạ kính xe xuống tầm 10cm giúp lưu thông không khí, hạn chế rủi ro xảy ra khi tài xế nhận được cuộc gọi khi ở xa.

Trên thực tế đã triển khai một số giải pháp đưa đón học sinh theo thời gian thực và ứng dụng di động dành cho phụ huynh. Với giải pháp đưa đón học sinh theo thời gian thực kết hợp với ứng dụng di động dành cho phụ huynh, vấn đề này được giải quyết dễ dàng. Học sinh chỉ cần quét thẻ hoặc mã QR mỗi khi lên, xuống xe buýt trường, ra vào cổng trường. Các dữ liệu bao gồm thông tin học sinh và thời gian lên/xuống xe, điểm danh ra/vào trường học sẽ được đồng bộ ngay lập tức với hệ thống của trường. Sau đó, cha mẹ có thể nhận được thông báo tự động qua ứng dụng di động.



Hình 2. 2 Giải pháp đưa đón học sinh theo thời gian thực và ứng dụng di động của công ty PHX Smart School

Với giải pháp này có những đặc điểm như:

- Cha mẹ có thể nhận được thông báo tự động.
- Cha mẹ cũng có thể theo dõi vị trí xe buýt theo thời gian thực.
- Cán bộ xe dễ dàng kiểm tra thông tin của trẻ và thông báo trạng thái của trẻ cho trường học và phụ huynh.
- Cán bộ xe, nhà trường và gia đình có thể hợp tác để đối phó với các tình huống bất thường.

2.2 Các thiết bị và linh kiện điện tử xây dựng hệ thống

Các thiết bị và linh kiện điện tử sử dụng trong hệ thống bao gồm:

- Các cảm biến
- Thiết bị đầu vào: Nút nhấn đơn, module Sim 800A.
- Thiết bị đầu ra: module sim 800A, module GPS, đèn led , màn hình OLED.
- Thiết bị điều khiển trung tâm: Board Arduino UNO R3, ESP32.
- Các chuẩn truyền dữ liệu: UART, I2C.

2.2.1 Hệ thống các cảm biến

Trong bối cảnh quan trọng của việc đảm bảo an toàn cho hành khách trên xe đưa đón học sinh, việc sử dụng hệ thống cảm biến đóng vai trò vô cùng quan trọng. Hệ thống này không chỉ giúp phát hiện các học sinh bị bỏ quên trên xe mà còn thông báo và giúp lái xe và cơ quan quản lý có thể giải cứu học sinh trong xe kịp thời.

Cảm biến hồng ngoại dùng để đếm số lượng trẻ lên và xuống xe, nếu đã đủ hoặc còn thì sẽ gửi thông tin đến module.

2.2.2 Board Arduino UNO R3

Arduino thật ra là một bo mạch vi xử lý được dùng để lập trình tương tác với các thiết bị phân cứng như cảm biến, động cơ, đèn hoặc các thiết bị khác. Đặc điểm nổi bật của Arduino là môi trường phát triển ứng dụng cực kỳ dễ sử dụng, với một ngôn ngữ lập trình có thể học một cách nhanh chóng ngay cả với người ít am hiểu về điện tử và lập trình. Và điều làm nên hiện tượng Arduino chính là mức giá rất thấp và tính chất nguồn mở từ phần cứng tới phần mềm. Các ứng dụng nổi bật của board mạch Arduino như robot đơn giản, điều khiển nhiệt độ, phát hiện chuyển động, game tương tác. Board Arduino có rất nhiều phiên bản với hiệu năng và mục đích sử dụng khác nhau như: Arduino Mega, Arduino LilyPad. Trong số đó, Arduino Uno R3 là một trong những phiên bản được sử dụng rộng rãi nhất. Arduino UNO R3 là một dòng Arduino đủ mạnh về tính năng, 20 chân, bộ nhớ đủ cho hầu hết các ứng dụng thông thường.



Hình 2. 3 Board Arduino UNO R3

Cấu trúc tổng quát Arduino UNO R3 như sau:

Bảng 2. 1 Các tham số cơ bản của Arduino UNO R3

Vi điều khiển	ATmega328P
Điện áp hoạt động	5V
Điện áp đầu vào(khuyên dùng)	7-12V
Điện áp đầu vào(giới hạn)	6-20V
Chân digital I/O	14 (Với 6 chân PWM output)
Chân PWM digital I/O	6
Chân đầu vào analog	6
Dòng sử dụng I/O PIN	20 mA
Dòng sử dụng 3.3PIN	50 mA
Bộ nhớ Flash	32 KB (ATmega328P) với 0.5KB dùng bởi bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock speed	16 MHZ
LED_BUILTIN	13

Chiều dài	68.6 mm
Chiều rộng	53.2 mm
Trọng lượng	25

2.2.3 Module SIM 800A

Mô-đun SIM800A là một thiết bị GSM/GPRS nhỏ gọn, là phiên bản thay thế cho module SIM900A đã quá đỗi quen thuộc, được sản xuất bởi hãng SIMCOM. Được tích hợp sẵn SIM và có khả năng kết nối với mạng di động, mô-đun này đã trở thành lựa chọn phổ biến cho các ứng dụng IoT, giám sát từ xa, và nhiều ứng dụng khác. Module cải thiện về độ bền, độ ổn định và vẫn có các tính năng cơ bản của một chiếc điện thoại như gọi điện thoại, nhắn tin, truy cập GPRS...

Mô-đun tích hợp nguồn xung và ic đệm được thiết kế nhỏ gọn nhưng vẫn giữ được các yếu tố cần thiết của thiết kế như: mạch chuyển mức tín hiệu logic sử dụng Mosfet, IC giao tiếp RS232 MAX232, mạch nguồn xung dòng cao, khe sim chuẩn và các đèn led báo hiệu.



Hình 2. 4 Module SIM 800A

Bảng 2. 2 Thông số kỹ thuật Module SIM800A

Điện áp hoạt động	5V-18V
Dòng điện hoạt động	100mA – 2A (Nên chọn nguồn trên 2A)
Dòng ở chế độ chờ	10mA
Công suất	10W
Chuẩn truyền	UART
Mức tín hiệu giao tiếp	TTL (3.3-5VDC) hoặc RS232
Tích hợp	Chuyển mức tín hiệu TTL Mosfet tốc độ cao
Tích hợp IC	Chuyển mức tín hiệu RS232 MAX232
Nhiệt độ hoạt động	-10°C – 60°C
Số chân	12
Loại	Module
Sử dụng khe	Micro Sim
Thiết kế	Nhỏ gọn, bền bỉ, chống nhiễu
Kích thước	50mm*49mm

Bảng 2. 3 Chức năng các chân của Module SIM800A

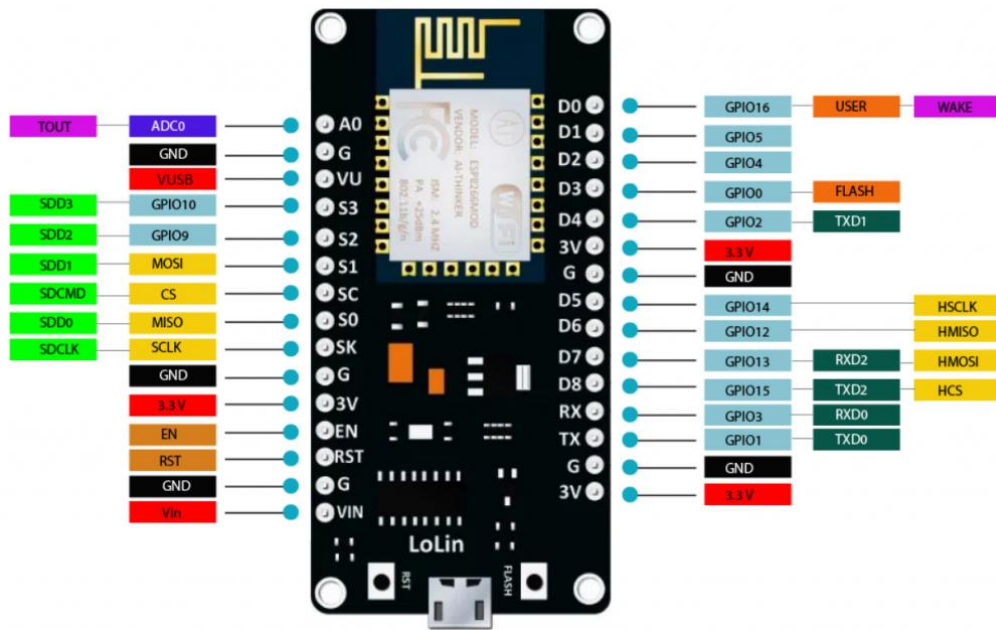
+ 5V	Nguồn dương từ 5VDC, lớn hơn 1A
GND	Mass, 0VDC

3.3V Transmit	Chân truyền tín hiệu TTL, chấp nhận mức 3.3V
3.3V Receive	Chân nhận tín hiệu TTL, chấp nhận mức 3.3V
5V Transmit	Chân truyền tín hiệu TTL, chấp nhận mức 5V
5V Receive	Chân nhận tín hiệu TTL, chấp nhận mức 5V
VCC MCU	Nguồn dương từ 5VDC, lớn hơn 1A
RS232 Transmit	Chân nhận tín hiệu RS232
RS232 Receive	Chân truyền tín hiệu RS232

2.2.4 ESP 32

ESP32 là một bộ vi điều khiển thuộc danh mục vi điều khiển trên chip công suất thấp và tiết kiệm chi phí. Hầu hết tất cả các biến thể ESP32 đều tích hợp Bluetooth và Wi-Fi chế độ kép, làm cho nó có tính linh hoạt cao, mạnh mẽ và đáng tin cậy cho nhiều ứng dụng.

Nó là sự kế thừa của vi điều khiển NodeMCU ESP8266 phổ biến và cung cấp hiệu suất và tính năng tốt hơn. Bộ vi điều khiển ESP32 được sản xuất bởi Espressif Systems và được sử dụng rộng rãi trong nhiều ứng dụng khác nhau như IoT, robot và tự động hóa.



Hình 2. 4 ESP32

ESP32 cũng được thiết kế để tiêu thụ điện năng thấp, lý tưởng cho các ứng dụng chạy bằng pin. Nó có hệ thống quản lý năng lượng cho phép nó hoạt động ở chế độ ngủ và chỉ thức dậy khi cần thiết, điều này có thể kéo dài tuổi thọ pin rất nhiều.

ESP32 được sử dụng để hiển thị trạng thái xe lên màn hình Oled đặt tại vị trí tài xế và đẩy dữ liệu lên ứng dụng trong đề tài.

2.2.5 Màn hình OLED 1.3 inch

Màn hình Oled 1.3 inch 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 oled 1.3 inch sử dụng giao tiếp I2C cho chất lượng đường truyền ổn định và rất dễ giao tiếp.

Màn hình OLED 1.3 inch được sử dụng làm màn hình hiển thị giao diện với người dùng trong đề tài.

Bảng 2. 4 Thông số màn hình Oled 1.3 inch

Thông số	Giá trị
Điện áp sử dụng	2.2~5.5VDC

Công suất tiêu thụ	0.04 W
Góc hiển thị	lớn hơn 160 độ
Số điểm hiển thị	128×64 điểm
Độ rộng màn hình	1.3 inch
Màu hiển thị	Trắng / Xanh Dương
Giao tiếp	I2C
Driver	SH1106 với 2 chân GPIO

2.3.6 Mạch Định Vị GPS NEO-6M V2

Mạch Định Vị GPS NEO-6M V2 là một trong những thành tựu đáng chú ý trong lĩnh vực định vị địa lý (GPS) và định thời gian. Được phát triển bởi công ty u-blox, mạch này là một trong những lựa chọn hàng đầu cho các ứng dụng cần độ chính xác cao và hiệu suất ổn định trong việc xác định vị trí trên toàn cầu.

Với thiết kế nhỏ gọn và hiệu suất mạnh mẽ, mạch NEO-6M V2 cung cấp khả năng định vị với độ chính xác cao, đồng thời tiết kiệm năng lượng và dễ dàng tích hợp vào các dự án điện tử khác nhau. Mạch được trang bị anten tích hợp và cổng giao tiếp UART, giúp dễ dàng kết nối với các vi điều khiển và thiết bị khác.

Với các tính năng như đa dạng giao thức giao tiếp, khả năng theo dõi nhiều vệ tinh cùng một lúc và độ chính xác cao trong việc xác định vị trí, mạch Định Vị GPS NEO-6M V2 là một công cụ lý tưởng cho các ứng dụng như định vị xe cộ, theo dõi thời gian thực và nhiều ứng dụng khác trong lĩnh vực IoT và các dự án điện tử. Với hiệu suất ổn định và tính linh hoạt cao, mạch NEO-6M V2 là một lựa chọn hàng đầu cho các nhà phát triển và kỹ sư trong việc tích hợp GPS vào các sản phẩm của họ.



Hình 2. 6 Mạch định vị GPS NEO-6M V2

2.3 Các chuẩn giao tiếp

2.3.1 Chuẩn giao tiếp UART

UART : Universal Asynchronous Receiver/Transmitter, là kiểu truyền thông tin nối tiếp không đồng bộ, UART thường được dùng trong máy tính công nghiệp, truyền thông, vi điều khiển, hay một số các thiết bị truyền tin khác.

Một số thông số:

Baud rate (tốc độ Baud) : Khi truyền nhận không đồng bộ để hai mô đun hiểu được nhau thì cần quy định một khoảng thời gian cho 1 bit truyền nhận, nghĩa là trước khi truyền thì tốc độ phải được cài đặt đầu tiên . Theo định nghĩa thì tốc độ baud là số bit truyền trong một giây.

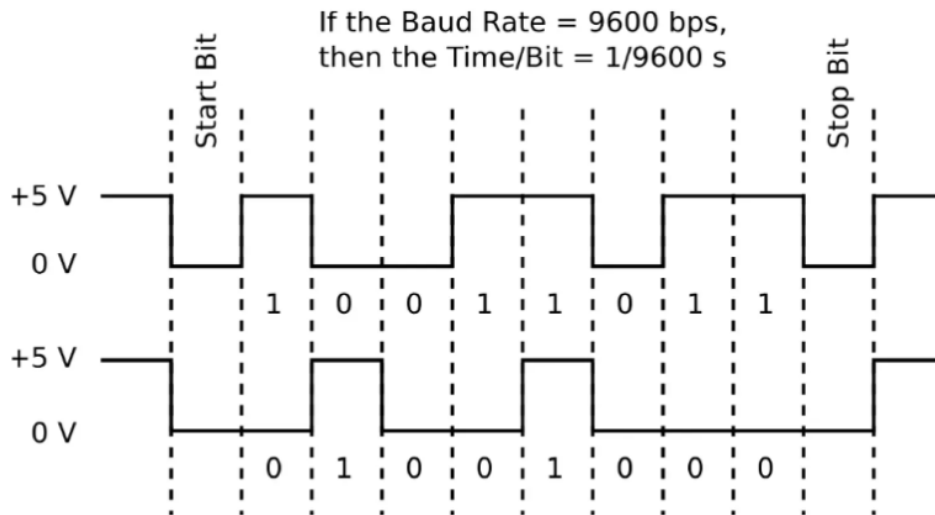
Frame (khung truyền) : Do kiểu truyền thông nối tiếp này rất dễ mất dữ liệu nên ngoài tốc độ , khung truyền cũng được cài đặt từ ban đầu để tránh bớt sự mất mát dữ liệu này . Khung truyền quy định số bit trong mỗi lần truyền, các bit báo như start, stop, các bit kiểm tra như parity, và số bit trong một data.

Bit Start : Là bit bắt đầu trong khung truyền Bit này nhằm mục đích báo cho thiết bị nhận biết quá trình truyền bắt đầu . trên AVR bit Start có trạng thái là 0.

Data : Dữ liệu cần truyền Data không nhất thiết phải 8 bit, có thể là 5, 6, 7, 8, 9. Trong UART bit LSB được truyền đi trước, Bit MSB được truyền đi sau.

Parity bit : Là bit kiểm tra dữ liệu đúng không . có 2 loại parity : chẵn (even parity), lẻ (old parity) . Parity chẵn là bit parity thêm vào để số bit 1 trong data + parity = chẵn, parity lẻ là bit parity thêm vào để số bit 1 trong data + parity = lẻ. Bit Parity là không bắt buộc nên có thể dùng hoặc không.

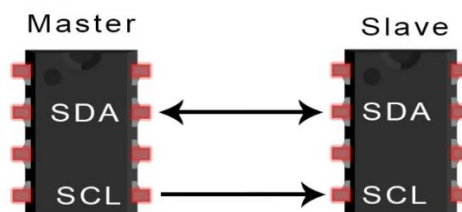
Stop : là bit báo cáo kết thúc khung truyền, thường là mức 5V và có thể có 1 hoặc 2 bit stop . Giản đồ trong hình 2.4 mô tả dữ liệu truyền đi bằng UART.



Hình 2. 7 Giản đồ truyền dữ liệu UART

2.3.2 Chuẩn giao tiếp I2C

I2C là 1 chuẩn truyền nối tiếp theo mô hình chủ – tớ. Một thiết bị chủ có thể giao tiếp với nhiều thiết bị tớ. Muốn giao tiếp với thiết bị nào, thiết bị chủ phải gửi đúng địa chỉ để kích hoạt thiết bị đó rồi mới được phép ghi hoặc đọc dữ liệu.



Hình 2. 8 Giao tiếp I2C

SDA và SCL của giao thức I2C sử dụng cơ chế truyền thông đồng bộ, trong đó tín hiệu (SCL) được Master điều khiển. Dữ liệu (bit) được truyền từng bit một qua đường SDA.

2.4 Kết luận chương

Chương này đã giới thiệu những vấn đề cơ bản để xây dựng hệ thống cảnh báo trẻ em bị bỏ quên trên xe đưa đón, bao gồm giới thiệu nguyên lý chung của hệ thống cũng như ví dụ hệ thống triển khai trên thực tế. Tiếp đó, chương này cũng giới thiệu sơ bộ các thiết bị và linh kiện điện tử sẽ được sử dụng để thiết kế hệ thống trong các chương sau.

CHƯƠNG III. THIẾT KẾ HỆ THỐNG

3.1 Chức năng của hệ thống thiết kế

Hệ thống cảnh báo an toàn cho xe đưa đón học sinh là một phần quan trọng của hệ thống giao thông trường học, giúp đảm bảo rằng học sinh được di chuyển một cách an toàn từ nhà đến trường và ngược lại. Hệ thống cảnh báo an toàn trên xe đưa đón học sinh bao gồm các chức năng:

+ Chức năng 1: Trước khi tài xế rời khỏi xe phải bấm vào nút bấm tại cửa xe, nếu còn học sinh trên xe thì còi, đèn lập tức kêu lên, báo cho tài xế biết còn trẻ bị bỏ quên.

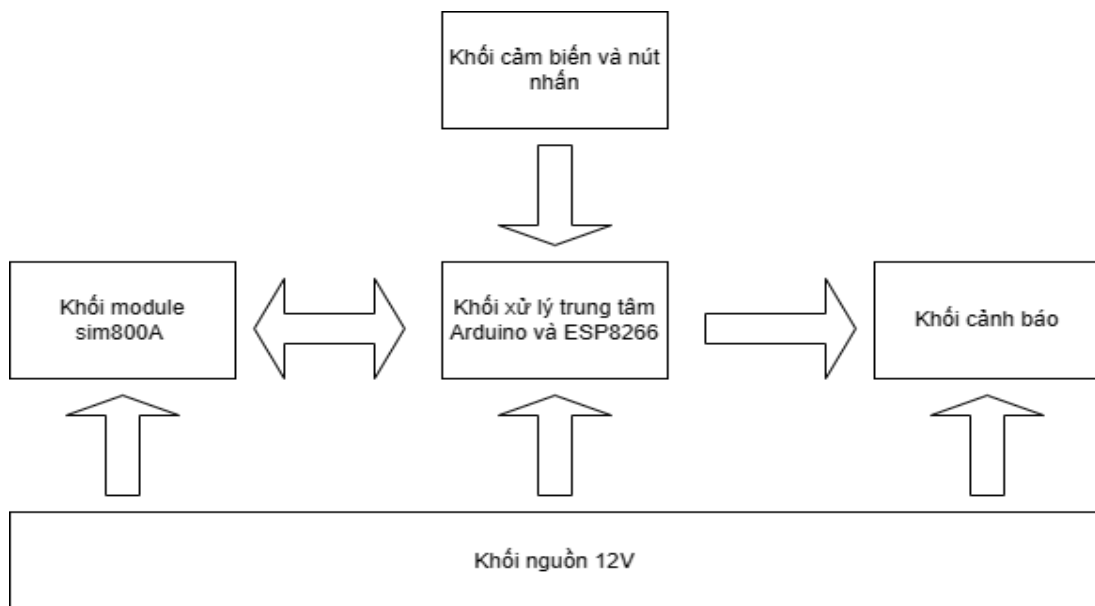
+ Chức năng 2: Cảnh báo quên học sinh trong xe:

Nếu như khi tài xế đã xuống xe mà quên bấm nút tại cửa xe. Khi phát hiện ra có học sinh bị bỏ quên sẽ thực hiện gọi cho lái xe quay trở lại xe. Điều này có thể giảm nguy cơ tai nạn và tránh được những hậu quả nghiêm trọng do học sinh bị bỏ quên trong xe trong thời tiết nóng.

3.2 Tính toán và thiết kế phần cứng hệ thống

3.2.1 Thiết kế sơ đồ khối hệ thống

Hệ thống cảnh báo an toàn trên xe đưa đón học sinh được thiết kế gồm 5 khối kết nối với nhau như hình 3.1 sau:



Hình 3. 1 Sơ đồ khối hệ thống

➤ **Chức năng các khối:**

- Khối xử lý trung tâm Arduino và ESP8266: dùng để xử lý các tín hiệu vào và xuất tín hiệu ra, điều khiển mọi hoạt động của hệ thống.
- Khối module sim: nhận tín hiệu từ vệ tinh gửi về điện thoại vị trí GPS và tin nhắn cảnh báo.
- Khối cảnh báo: Xử lý chuông kêu khi nhận tín hiệu từ khối Arduino.
- Khối nguồn : cung cấp nguồn cho các khối khác.
- Khối cảm biến và nút nhấn: Nút nhấn có chức năng bật/tắt mạch. Cảm biến khi được tác động sẽ gửi trạng thái cho khối xử lý trung tâm Arduino xử lý.

3.2.2 Tính toán và thiết kế mạch

a. Khối xử lý trung tâm

Arduino UNO sử dụng 3 vi điều khiển chính thuộc họ 8bit AVR là ATmega328, ATmega168, ATmega8. Với 3 dòng vi điều khiển này, mạch có thể xử lý những tác vụ đơn giản như điều khiển nhấp nháy đèn LED, thiết lập ứng dụng đo độ ẩm - nhiệt độ sau đó truyền và hiển thị lên LCD, tiếp nhận và xử lý tín hiệu cho ứng dụng điều khiển xe từ xa .. Nó có 16 chân digital I/O, 6 chân đầu vào tương tự (Analog Inputs), một thạch anh dao động 16 MHz, kết nối USB, một jack cắm điện, một đầu ICSP và một nút reset như trong hình:



Hình 3. 2 Khối xử lý trung tâm Arduino UNO R3

Trong quá trình kết nối các module và lập trình cho hệ thống, tổng số chân I/O sử dụng là 15 chân, công thức tính dòng tiêu thụ I có thể được tính như sau:

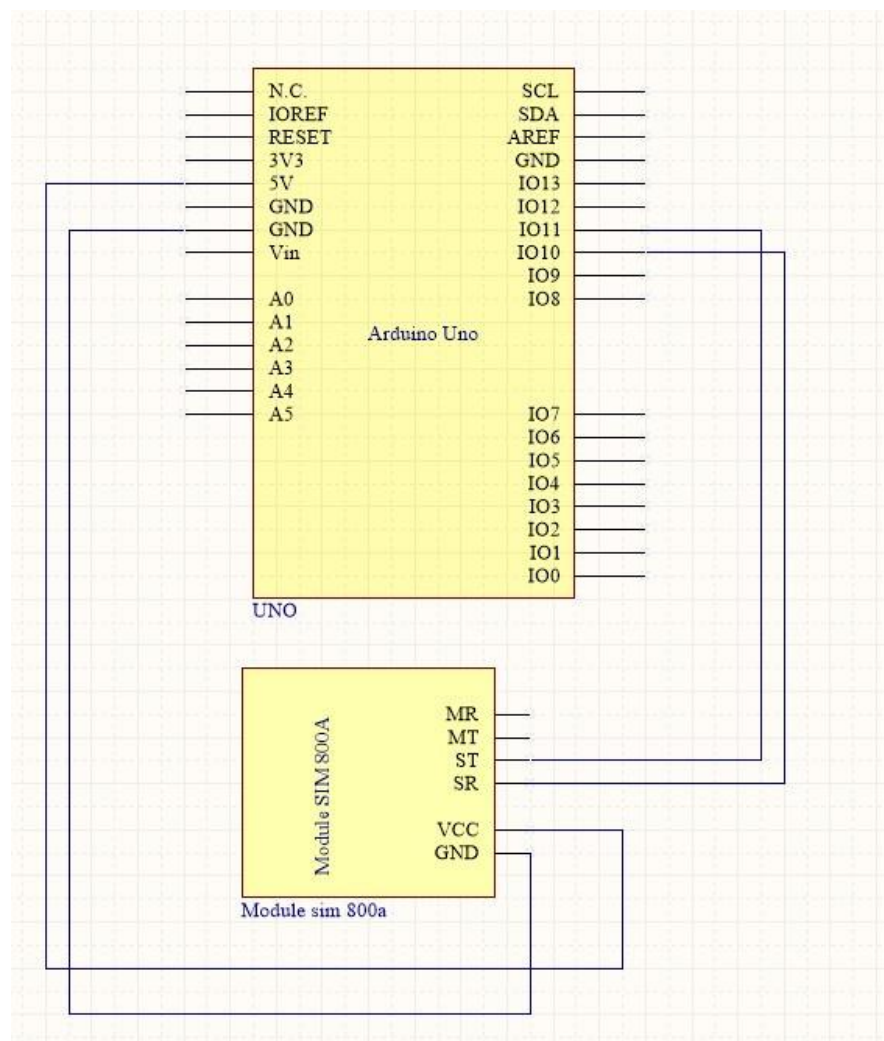
$$I = P * L = 15 * 20\text{mA} = 300\text{mA}$$

Trong đó, P là số chân và lo là dòng ngõ ra mỗi chân I/O.

b. Module SIM800A

Mạch GSM GPRS Sim800A được phát triển từ module sim 900A tích hợp nguồn xung và ic đệm được thiết kế cho các ứng dụng cần độ bền và độ ổn định cao. Vì module sim 800A chỉ sử dụng để gọi thông báo khi có học sinh bỏ quên trên xe nên ta chỉ sử dụng 4 chân của module là VCC , GND , ST , SR.

- Chân GND và chân VCC nối với chân 5V và chân GND tại UNO để cấp nguồn 5V để dòng điện cho sim hoạt động ổn định.
- Chân ST nối với chân IO11 để truyền tín hiệu về UNO.
- Chân SR nối với chân IO10 để nhận tín hiệu từ UNO.



Hình 3. 3 Khối module sim 800A kết nối với UNO

c. Module ESP32

Module esp 32 sử dụng để hiển thị trạng thái xe lên màn hình Oled đặt tại vị trí tài xế và đẩy dữ liệu lên ứng dụng.

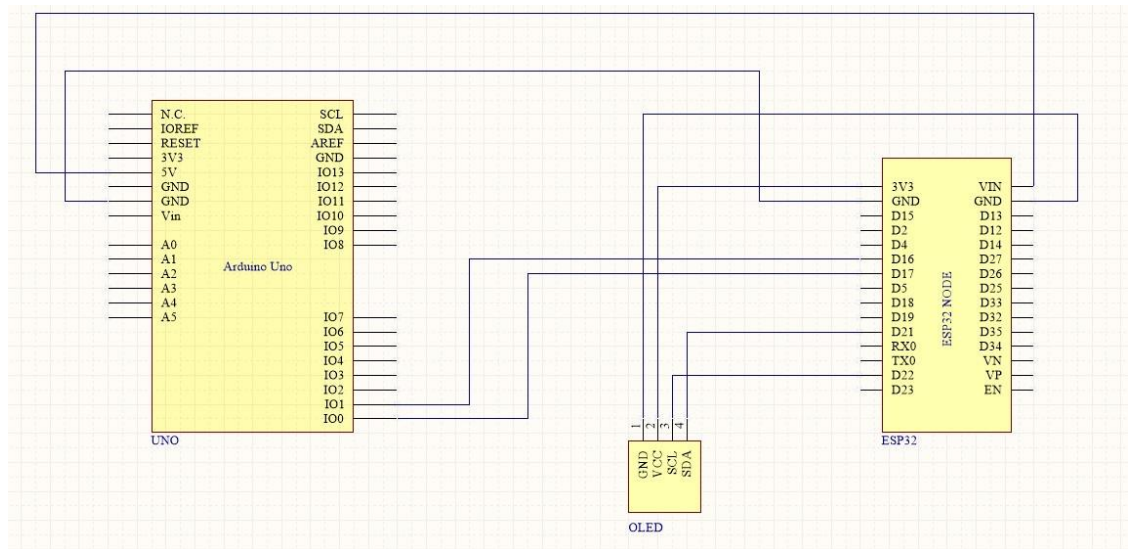
Các chân 3v3 , GND , D21 và D22 nối với màn hình led như sau:

- Chân SCL kết nối với D22 . Chân SCL là chân lock , có tác dụng đồng bộ hóa truyền dữ liệu giữa các thiết bị.

- Chân SDA kết nối với D21 thực hiện truyền dữ liệu.

Chân GND, D16, D17, VIN kết nối với Arduino R3 như sau:

- Chân VIN kết nối với chân 5V tại UNO nhận dòng điện để module hoạt động ổn định lâu dài.
- Chân D16 nối với IO1. Đây là chân nhận dữ liệu của Arduino.
- Chân D17 nối với IO0. Đây là chân truyền dữ liệu của Arduino.



Hình 3. 4 Khối module ESP32 kết nối với Oled và UNO

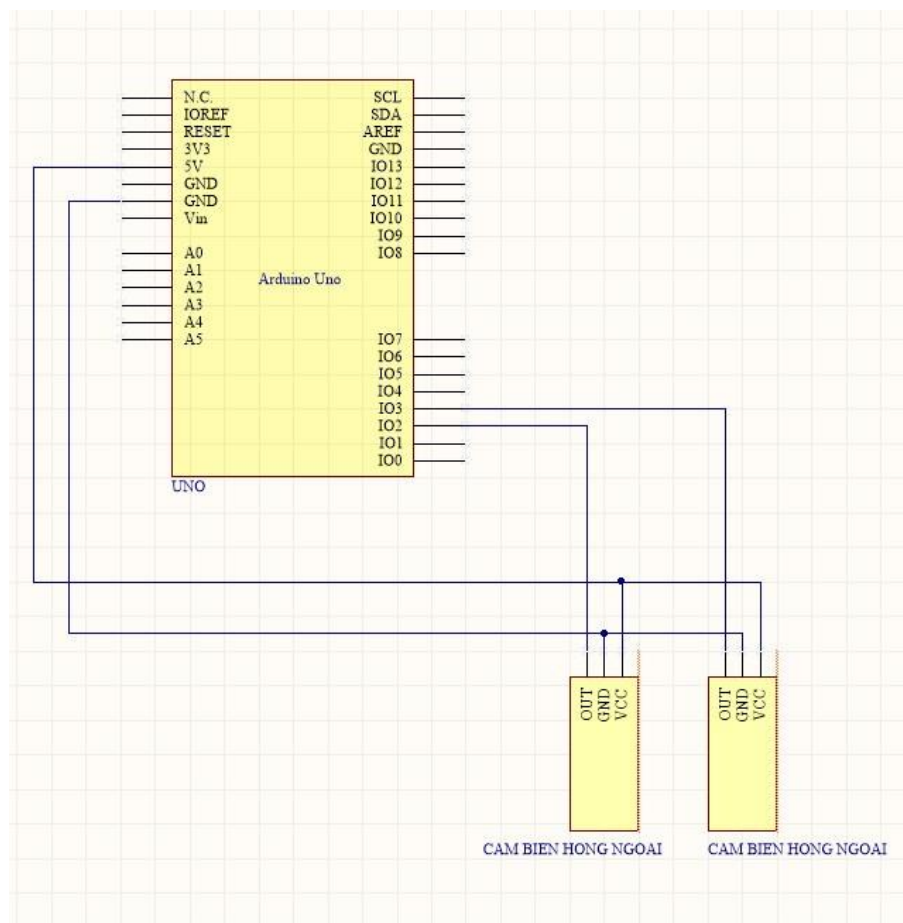
d. Khối cảm biến và nút nhấn

➤ Cảm biến hồng ngoại

- Điện áp làm việc: 3.3V – 5V DC.
- Kích thước: 3.2cm * 1.4cm.
- Phạm vi hoạt động 2 - 5cm
- Giao tiếp I2C

Cách kết nối của cảm biến hồng ngoại được kết nối với Arduino theo mô tả và hình dưới đây:

- Chân VCC kết nối với chân 5V tại Arduino , chân GND kết nối với chân GND tại Arduino để nhận dòng điện giúp module hoạt động ổn định lâu dài.
- Chân Out kết nối với chân IO2 thực hiện truyền dữ liệu.



Hình 3. 5 Khối cảm biến kết nối với UNO

➤ Khối nút nhấn

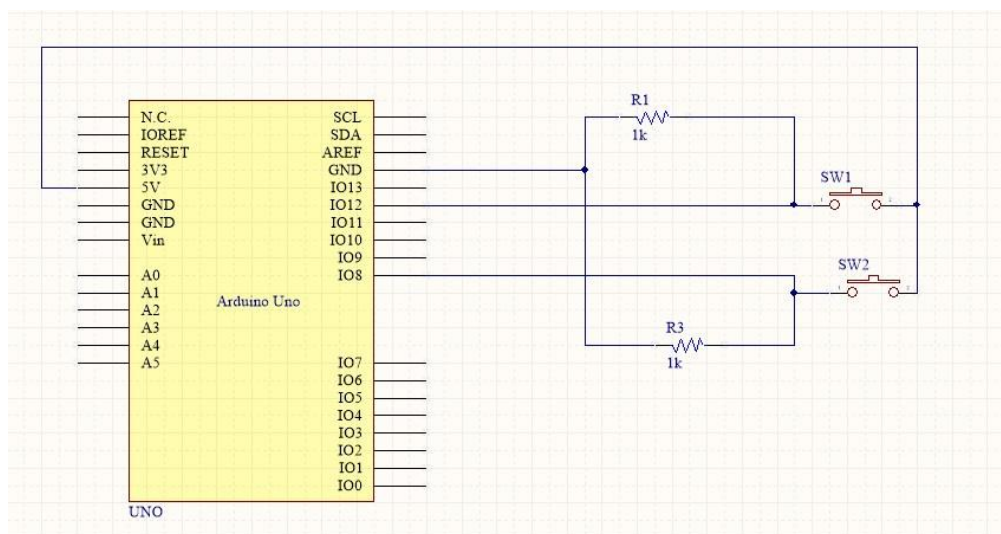
Khối nút nhấn có dòng đi qua là 0.4mA, điện áp đầu vào là 5V. Giá trị điện trở được tính bằng công thức:

$$R = \frac{v}{I_{in}} = \frac{5v}{0,4mA} = 12 \text{ k}\Omega$$

Trong đó, Rn là điện trở của nút nhấn, In là cường độ dòng điện qua nút nhấn.

Cách kết nối giữa khối nút nhấn và Arduino UNO R3, được thể hiện trong hình 3.7 với các chân được mô tả sau đây :

- Chân số 1 nối với IO12 và IO8 để điều khiển.
- Chân số 2 nối với nguồn 5V.



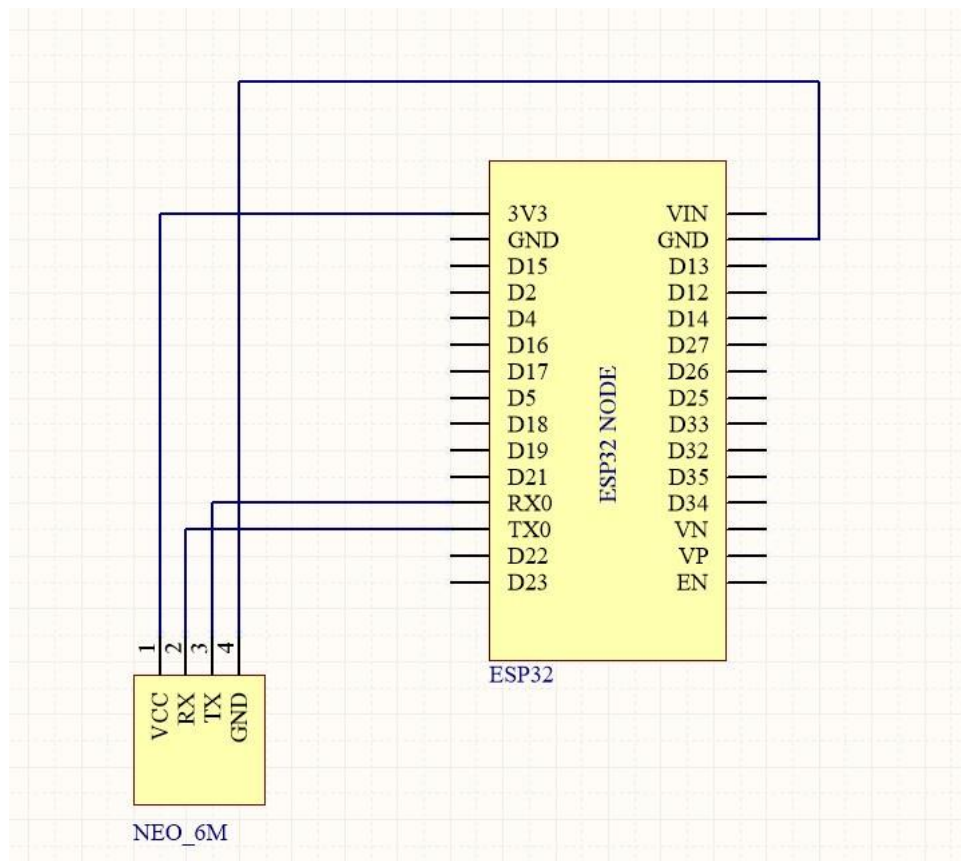
Hình 3. 6 Khối nút nhấn kết nối với UNO

e. Khởi định vị GPS

Mạch định vị GPS GY-NEO 6M V2 là module định vị toàn cầu sử dụng hệ thống vệ tinh GPS của Mỹ. Mạch định vị GPS NEO-6M cho tốc độ xác định vị trí nhanh và chính xác, có nhiều mức năng lượng hoạt động, phù hợp với các ứng dụng chạy pin. Trong hệ thống module GPS dùng để phụ huynh và người giám sát có thể theo dõi hành trình di chuyển của xe. Ngoài ra còn giúp tài xế xác định chính xác đường đi và điểm đón học sinh.

Kết nối của module GPS với ESP32 được mô tả như hình ảnh dưới đây:

- Chân GND và chân VCC nối với chân 3V3 và chân GND tại ESP32 để cấp nguồn 3.3V để dòng điện cho module hoạt động ổn định.
- Chân RX và TX của module lần lượt kết nối với chân RX0 TX0 của ESP32 để thực hiện truyền tải dữ liệu.



Hình 3. 7 Module GPS kết nối ESP32

f. Khởi nguồn

- Theo thông số kỹ thuật, Arduino hoạt động ở điện áp 5V, module hoạt động ở điện áp 5-18V. Vì vậy, nhóm sinh viên sử dụng nguồn adapter 12V 2A có tích hợp ổn áp trên module sim 808A đảm bảo cho arduino và module sim 808A hoạt động ổn định.

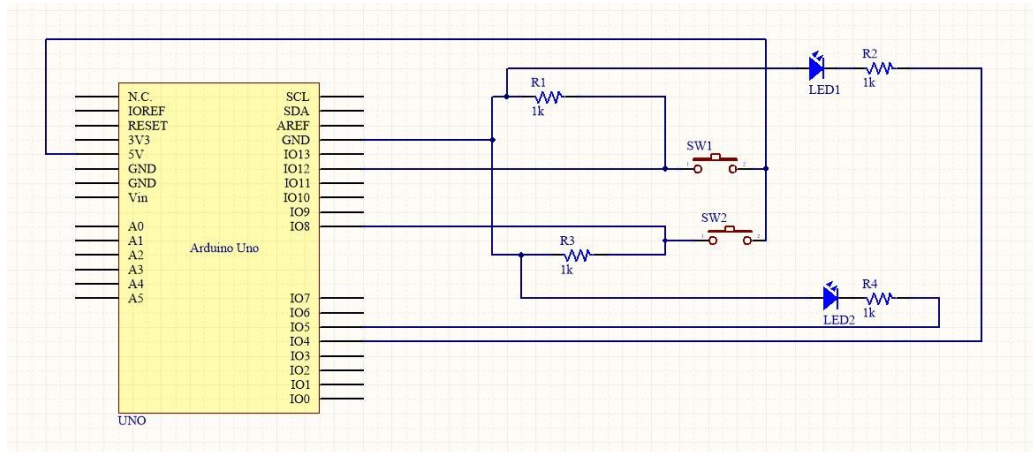
g. Khởi cảnh báo

Khởi cảnh báo sử dụng led 5mm, hoạt động ở mức 2.7V, dòng qua led hoạt động là 10mA. Điện áp đầu vào là 5V. Giá trị điện trở hạn dòng cho led được tính bằng công thức:

$$R_{led} = (V - V_{led}) / I_{led} = (5V - 2.7V) / 10mA = 0,23 \text{ k}\Omega = 230 \Omega$$

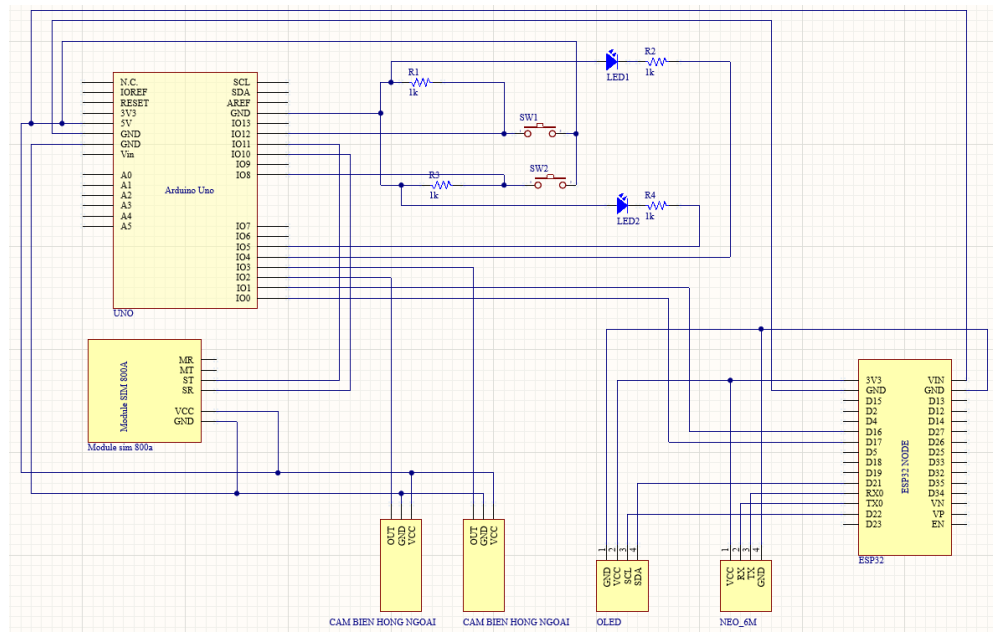
Chọn giá trị điện trở là 103Ω - nhằm giảm dòng điện thực tế xuống, giúp giảm độ chói của led và hạn chế việc led bị cháy. Dòng qua led thực tế tính bằng công thức sau:

$$I_{led} = (V - V_{led})/R = (5V - 2.7V)/ 103 = 22 \text{ mA}$$



Hình 3. 8 Khối cảnh báo kết nối với UNO

h. Sơ đồ toàn mạch



Hình 3. 9 Sơ đồ toàn mạch

3.3 Xây dựng phần mềm cho hệ thống

3.3.1 Phân tích yêu cầu của phần mềm cho hệ thống

- Giao diện người dùng thân thiện: Ứng dụng cần có một giao diện dễ sử dụng và thân thiện với người dùng. Đặc biệt cần xem xét đến việc sử dụng nút lớn, văn bản dễ đọc và màu sắc thu hút để thuận tiện cho tài xế và nhân viên trường.
- Thể hiện và chỉnh sửa được các thông tin người nhận cuộc gọi.
- Tích hợp với hệ thống cảm biến trên xe: Ứng dụng cần kết nối và tương tác với các cảm biến an toàn trên xe. Điều này giúp cung cấp thông tin đầy đủ về tình trạng an toàn của xe đến tài xế và quản lý.
- Thông báo và cảnh báo: Ứng dụng cần có khả năng thông báo ngay lập tức nếu có học sinh bị bỏ quên trên xe. Cảnh báo cần được hiển thị một cách rõ ràng và đầy đủ để tài xế và nhân viên trường có thể phản ứng kịp thời.

3.3.2 Các công cụ sử dụng để xây dựng phần mềm hệ thống

3.3.2.1 Ngôn ngữ Java

Java là một trong những ngôn ngữ lập trình hướng đối tượng. Nó được sử dụng trong phát triển phần mềm, trang web, game hay ứng dụng trên các thiết bị di động.

Java được khởi đầu bởi James Gosling và bạn đồng nghiệp ở Sun MicroSystem năm 1991. Ban đầu Java được tạo ra nhằm mục đích viết phần mềm cho các sản phẩm gia dụng, và có tên là Oak.

Java được phát hành năm 1994, đến năm 2010 được Oracle mua lại từ Sun MicroSystem.

Java được tạo ra với tiêu chí “Viết (code) một lần, thực thi khắp nơi” (Write Once, Run Anywhere – WORA). Chương trình phần mềm viết bằng Java có thể chạy trên mọi nền tảng (platform) khác nhau thông qua một môi trường thực thi với điều kiện có môi trường thực thi thích hợp hỗ trợ nền tảng đó.

Đặc điểm của ngôn ngữ lập trình Java

- Tương tự C++, hướng đối tượng hoàn toàn

Trong quá trình tạo ra một ngôn ngữ mới phục vụ cho mục đích chạy được trên nhiều nền tảng, các kỹ sư của Sun MicroSystem muốn tạo ra một ngôn ngữ dễ học và

quen thuộc với đa số người lập trình. Vì vậy họ đã sử dụng lại các cú pháp của C và C++.

Tuy nhiên, trong Java thao tác với con trỏ bị lược bỏ nhằm đảm bảo tính an toàn và dễ sử dụng hơn. Các thao tác overload, goto hay các cấu trúc như struct và union cũng được loại bỏ khỏi Java.

- Độc lập phần cứng và hệ điều hành

Một chương trình viết bằng ngôn ngữ Java có thể chạy tốt ở nhiều môi trường khác nhau. Gọi là khả năng “cross-platform”. Khả năng độc lập phần cứng và hệ điều hành được thể hiện ở 2 cấp độ là cấp độ mã nguồn và cấp độ nhị phân.

Ở cấp độ mã nguồn: Kiểu dữ liệu trong Java nhất quán cho tất cả các hệ điều hành và phần cứng khác nhau. Java có riêng một bộ thư viện để hỗ trợ vấn đề này. Chương trình viết bằng ngôn ngữ Java có thể biên dịch trên nhiều loại máy khác nhau mà không gặp lỗi.

Ở cấp độ nhị phân: Một mã biên dịch có thể chạy trên nhiều nền tảng khác nhau mà không cần dịch lại mã nguồn. Tuy nhiên cần có Java Virtual Machine để thông dịch đoạn mã này.

- Ngôn ngữ thông dịch

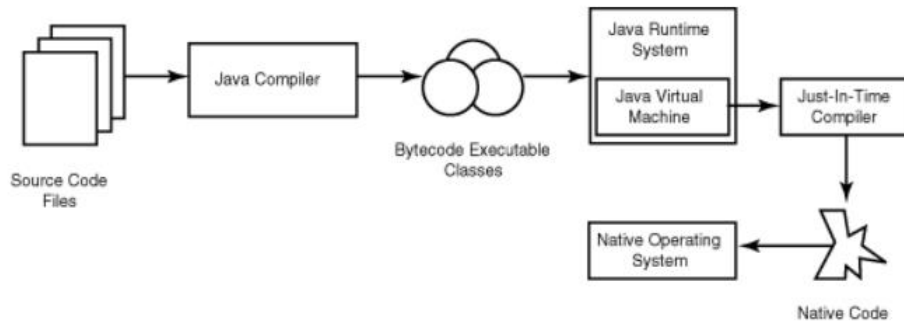
Ngôn ngữ lập trình thường được chia ra làm 2 loại (tùy theo các hiện thực hóa ngôn ngữ đó) là ngôn ngữ thông dịch và ngôn ngữ biên dịch.

- Thông dịch (Interpreter) : Nó dịch từng lệnh rồi chạy từng lệnh, lần sau muốn chạy lại thì phải dịch lại.

- Biên dịch (Compiler): Code sau khi được biên dịch sẽ tạo ra 1 file thường là .exe, và file .exe này có thể đem sử dụng lại không cần biên dịch nữa.

Ngôn ngữ lập trình Java thuộc loại ngôn ngữ thông dịch. Chính xác hơn, Java là loại ngôn ngữ vừa biên dịch vừa thông dịch. Cụ thể như sau:

Khi viết mã, hệ thống tạo ra một tệp .java. Khi biên dịch mã nguồn của chương trình sẽ được biên dịch ra mã byte code. Máy ảo Java (Java Virtual Machine) sẽ thông dịch mã byte code này thành machine code (hay native code) khi nhận được yêu cầu chạy chương trình.



Hình 3. 10 Biên dịch trong Java

- Ưu điểm : Phương pháp này giúp các đoạn mã viết bằng Java có thể chạy được trên nhiều nền tảng khác nhau. Với điều kiện là JVM có hỗ trợ chạy trên nền tảng này.

- Nhược điểm : Cũng như các ngôn ngữ thông dịch khác, quá trình chạy các đoạn mã Java là chậm hơn các ngôn ngữ biên dịch khác (tuy nhiên vẫn ở trong một mức chấp nhận được).

- Cơ chế thu gom rác tự động

Khi tạo ra các đối tượng trong Java, JRE sẽ tự động cấp phát không gian bộ nhớ cho các đối tượng ở trên heap.

Với ngôn ngữ như C C++, bạn sẽ phải yêu cầu hủy vùng nhớ mà bạn đã cấp phát, để tránh việc thất thoát vùng nhớ. Tuy nhiên vì một lý do nào đó, bạn không hủy một vài vùng nhớ, dẫn đến việc thất thoát và làm giảm hiệu năng chương trình.

Ngôn ngữ lập trình Java hỗ trợ cho bạn điều đó, nghĩa là bạn không phải tự gọi hủy các vùng nhớ. Bộ thu dọn rác của Java sẽ theo vết các tài nguyên đã được cấp. Khi không có tham chiếu nào đến vùng nhớ, bộ thu dọn rác sẽ tiến hành thu hồi vùng nhớ đã được cấp phát.

- Đa luồng

Java hỗ trợ lập trình đa tiến trình (multithread) để thực thi các công việc đồng thời. Đồng thời cũng cung cấp giải pháp đồng bộ giữa các tiến trình (giải pháp sử dụng priority...).

- Tính an toàn và bảo mật

- Tính an toàn

Ngôn ngữ lập trình Java yêu cầu chặt chẽ về kiểu dữ liệu.

Dữ liệu phải được khai báo tường minh.

Không sử dụng con trỏ và các phép toán với con trỏ.

Java kiểm soát chặt chẽ việc truy nhập đến mảng, chuỗi. Không cho phép sử dụng các kỹ thuật tràn. Do đó các truy nhập sẽ không vượt quá kích thước của mảng hoặc chuỗi.

Quá trình cấp phát và giải phóng bộ nhớ được thực hiện tự động.

Cơ chế xử lý lỗi giúp việc xử lý và phục hồi lỗi dễ dàng hơn.

- Tính bảo mật

Java cung cấp một môi trường quản lý chương trình với nhiều mức khác nhau.

Mức 1 : Chỉ có thể truy xuất dữ liệu cũng như phương thức thông qua giao diện mà lớp cung cấp.

Mức 2 : Trình biên dịch kiểm soát các đoạn mã sao cho tuân thủ các quy tắc của ngôn ngữ lập trình Java trước khi thông dịch.

Mức 3 : Trình thông dịch sẽ kiểm tra mã byte code xem các đoạn mã này có đảm bảo được các quy định, quy tắc trước khi thực thi.

Mức 4: Java kiểm soát việc nạp các lớp vào bộ nhớ để giám sát việc vi phạm giới hạn truy xuất trước khi nạp vào hệ thống.

3.3.2.2 Phần mềm lập trình cho vi điều khiển

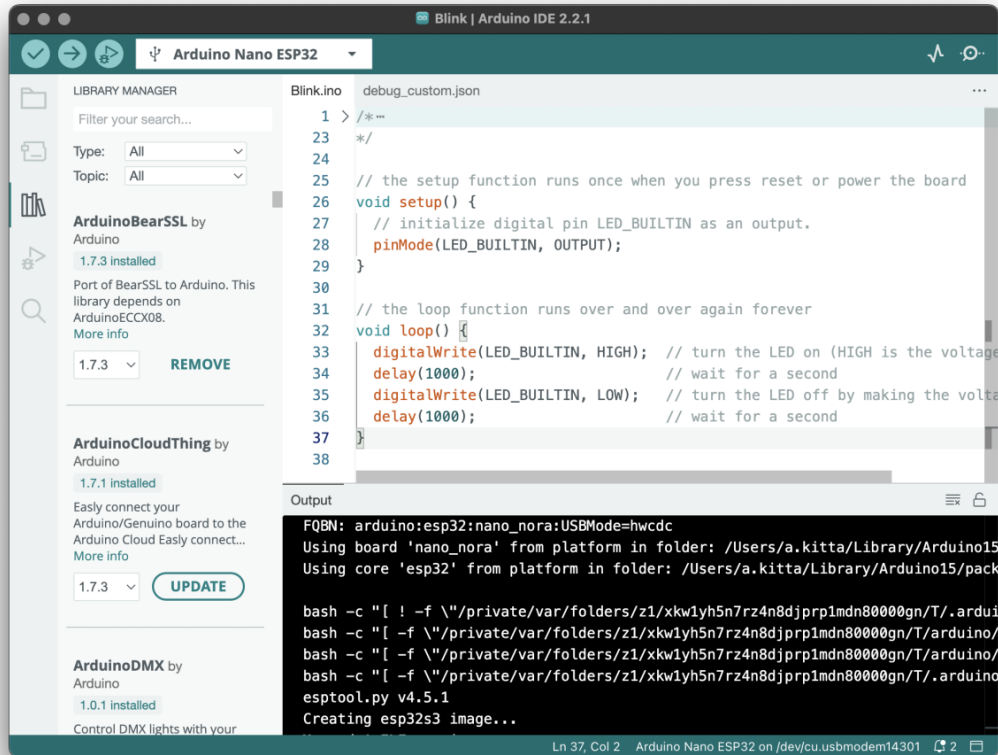
- Giới thiệu phần mềm lập trình Arduino IDE

Môi trường phát triển tích hợp Arduino IDE là một ứng dụng đa nền tảng được viết bằng Java, và được dẫn xuất từ IDE cho ngôn ngữ lập trình xử lý và các dự án lắp ráp. Do có tính chất mã nguồn mở nên môi trường lập trình này hoàn toàn miễn phí và có thể mở rộng thêm bởi người dùng có kinh nghiệm.

Người sử dụng chỉ cần định nghĩa hai hàm để thực hiện một chương trình hoạt động theo chu trình:

setup(): hàm chạy một lần duy nhất vào lúc bắt đầu của một chương trình dùng để khởi tạo các thiết lập.

loop(): hàm được gọi lặp lại liên tục cho đến khi bo mạch được tắt.

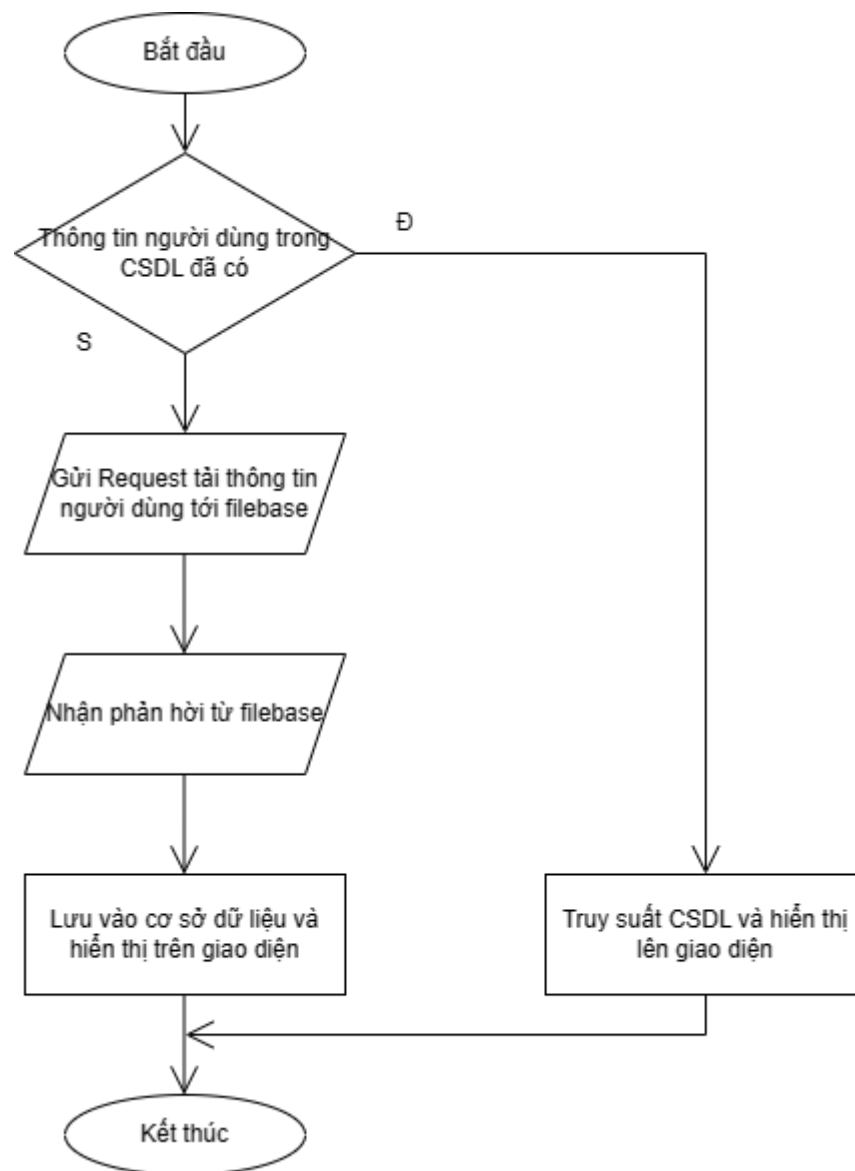


Hình 3. 11 Giao diện phần mềm lập trình Arduino IDE

3.3.3 Xây dựng các lưu đồ giải thuật

3.3.3.1 Đăng nhập hoặc đăng ký tài khoản

Để có thể triển khai trong thực tế, cần thiết lập hệ thống có khả năng quản lý người sử dụng rõ ràng, vì vậy mỗi người dùng được cung cấp một tài khoản riêng khi đăng nhập, chỉ có những người sử dụng thiết bị mới được cung cấp tài khoản.



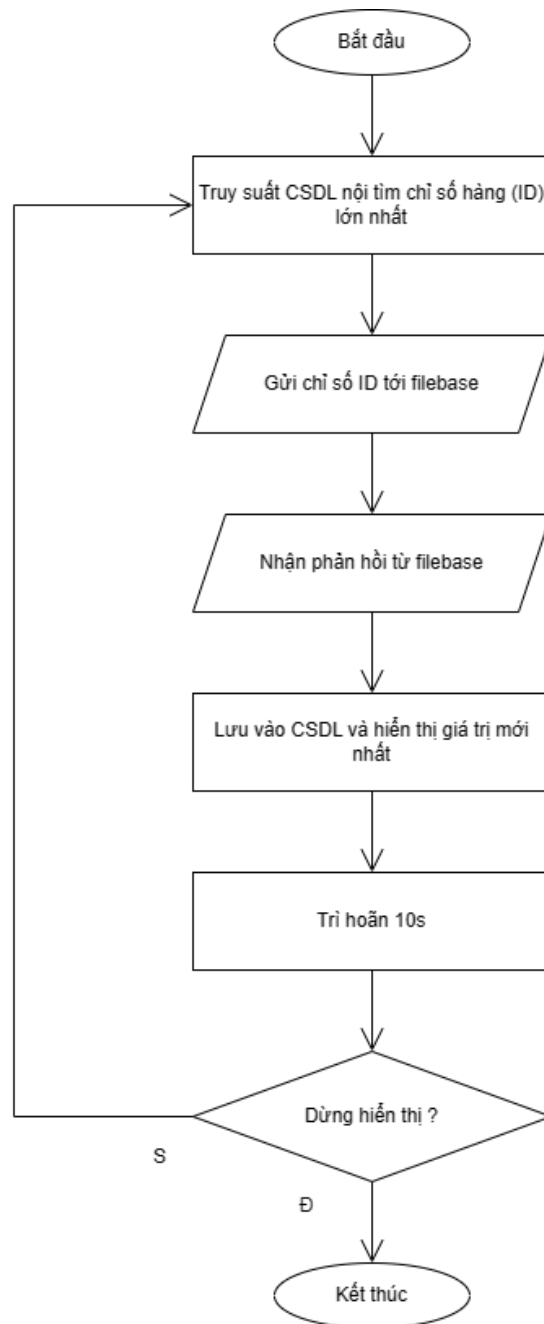
Hình 3. 12 Lưu đồ đăng nhập hoặc đăng ký tài khoản

Sau khi đăng nhập thành công, ứng dụng sẽ hiển thị thông tin trên chuyển xe.

- Thẻ hiển thị tên người dùng.
- Số người trên xe.
- Tên trường, số xe, tên tài xế, trạng thái xe.

3.3.3.2 Cập nhật dữ liệu từ cảm biến lên App

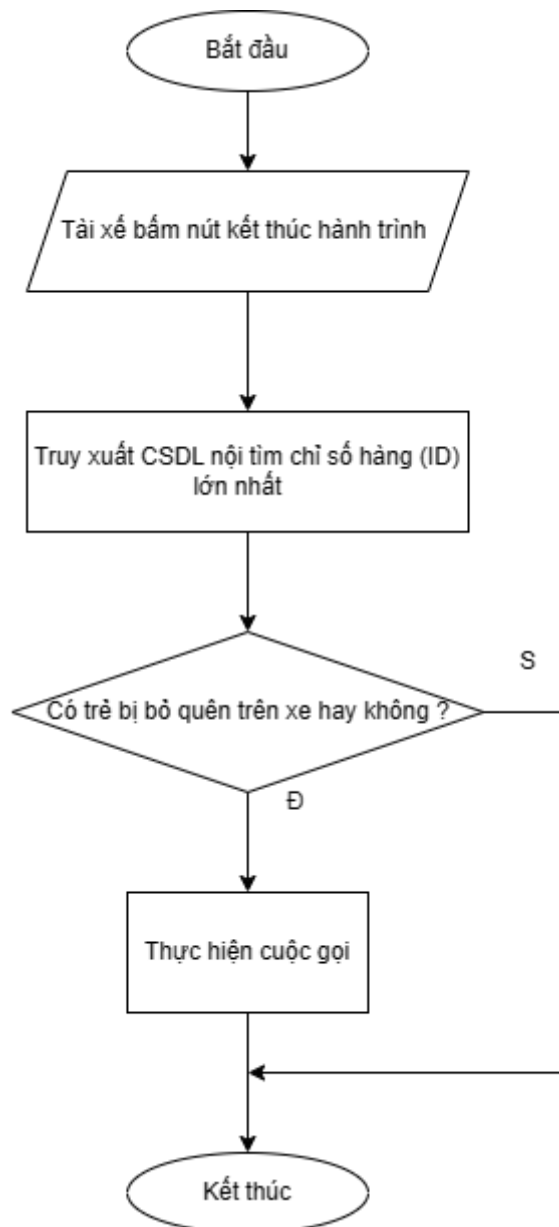
Thông tin số học sinh trên xe sẽ được cập nhật lên app liên tục với chu kỳ 10s 1 lần và bắt đầu cập nhật khi tài xế bấm nút bắt đầu và dừng cập nhật khi nhấn nút kết thúc hành trình.



Hình 3. 13 Lưu đồ cập nhật dữ liệu từ cảm biến lên app

3.3.3.3 Thực hiện cuộc gọi thông báo tới tài xế

Sau khi kết thúc hành trình, trước khi xuống xe tài xế phải bấm vào nút **kết thúc hành trình**, lúc này hệ thống sẽ kiểm tra nếu trên xe còn học sinh lập tức thực hiện cuộc gọi đến tài xế, nếu trên xe không còn ai hệ thống sẽ kết thúc.



Hình 3. 14 Lưu đồ thực hiện cuộc gọi thông báo tới tài xế

3.4 Kết luận chương

Chương này đã tiến hành thiết kế chi tiết hệ thống bao gồm cả phần cứng và phần mềm. Thiết kế đi từ phân tích yêu cầu về chức năng của hệ thống để đưa ra sơ đồ khối. Trên cơ sở sơ đồ khối nhóm tác giả đã tính toán thiết kế chi tiết cho các khối phần cứng của hệ thống. Chương này cũng tiến hành xây dựng các thuật toán cho phần mềm điều khiển hệ thống, bao gồm phần mềm điều khiển trên hệ thống phần cứng và phần mềm ứng dụng trên điện thoại thông minh.

CHƯƠNG IV. CÁC KẾT QUẢ VÀ ĐÁNH GIÁ

4.1 Mô hình sản phẩm của hệ thống

Mô hình thực tế chính là kết quả thu được sau khi chúng ta tính toán và thiết kế mạch. Khi bắt tay vào thi công, chúng ta cần phải có những bước thực hiện hợp lý, tuần như vậy mới chính xác được.

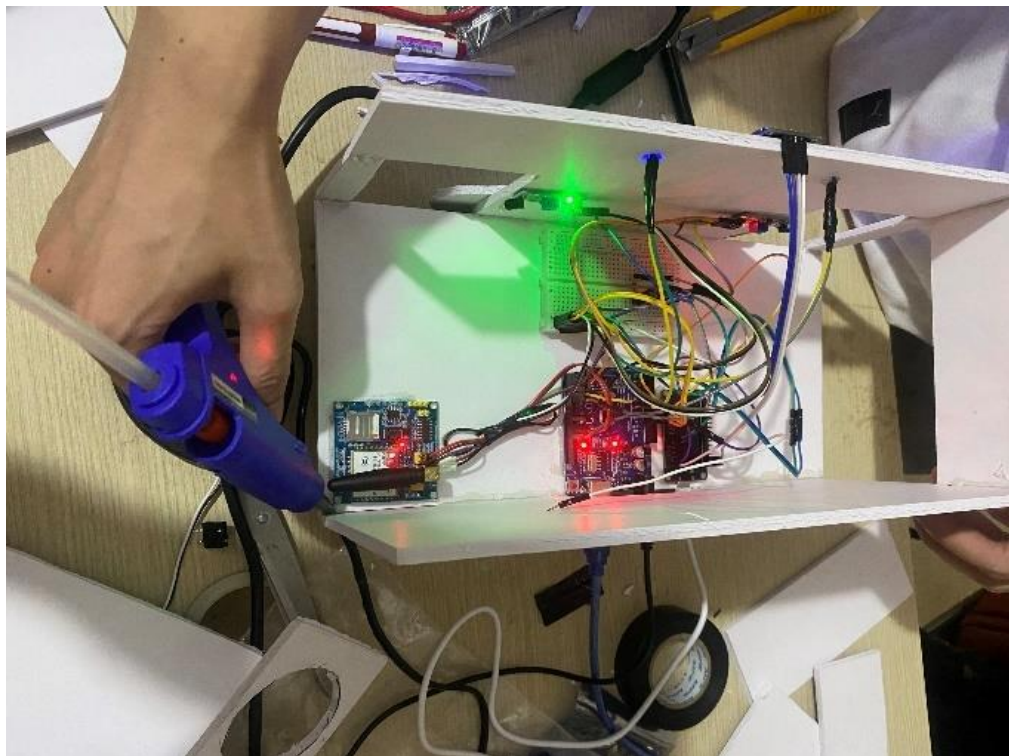
Quá trình thực hiện thi công trải qua các bước sau:

- Thi công chế tạo mô hình, lắp ráp linh kiện.
- Tiến hành nạp chương trình cho hệ thống trên mô hình, chạy thử, tinh chỉnh.

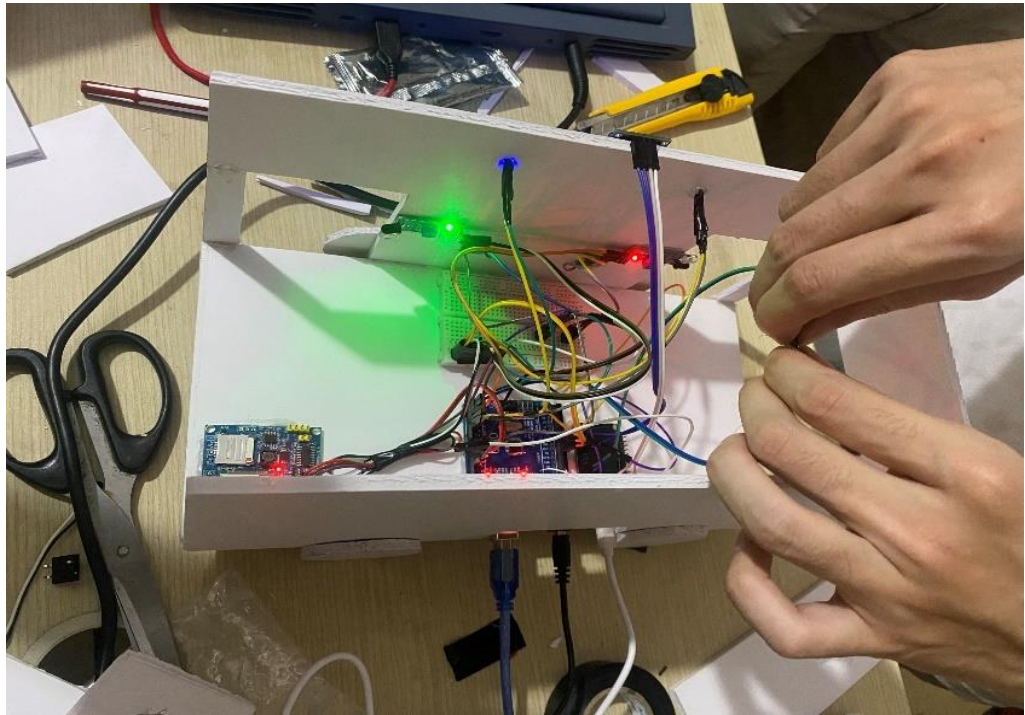
4.1.1 Lắp ráp và kiểm tra hệ thống

Quy trình lắp ráp:

Bước 1: Gắn đầu bus của các module sim 800A, ESP8266, Arduino. Nút nhấn, các điện trở. Đo kiểm tra từng chân của các thiết bị đã kết nối hết chưa.

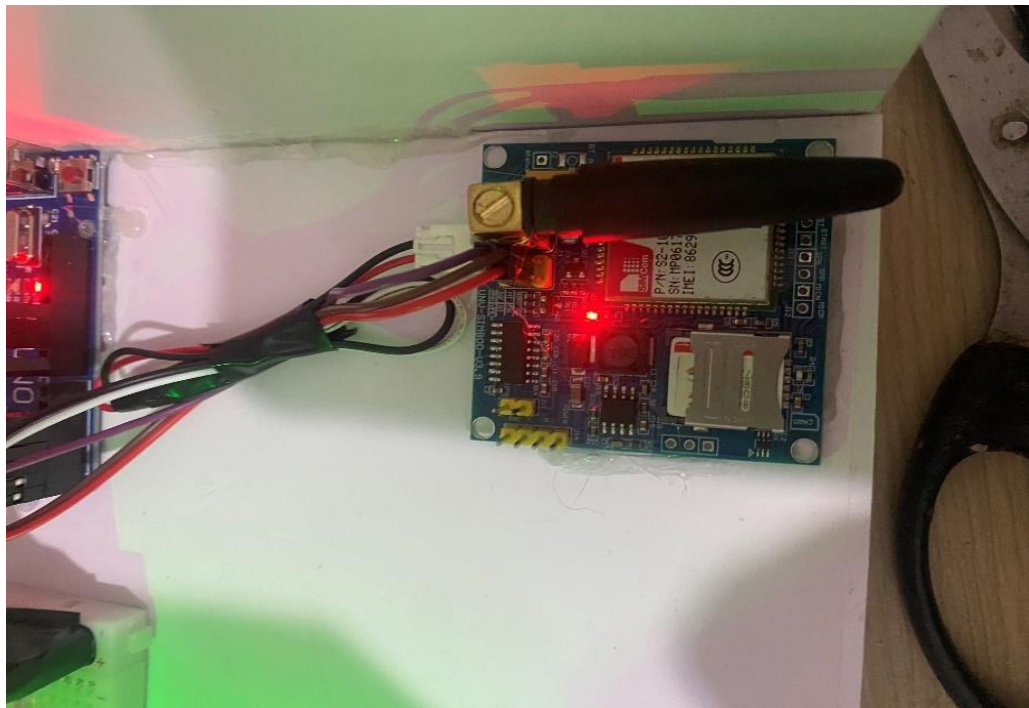


Hình 4. 1 Tiến hành gắn các linh kiện vào mô hình xe oto

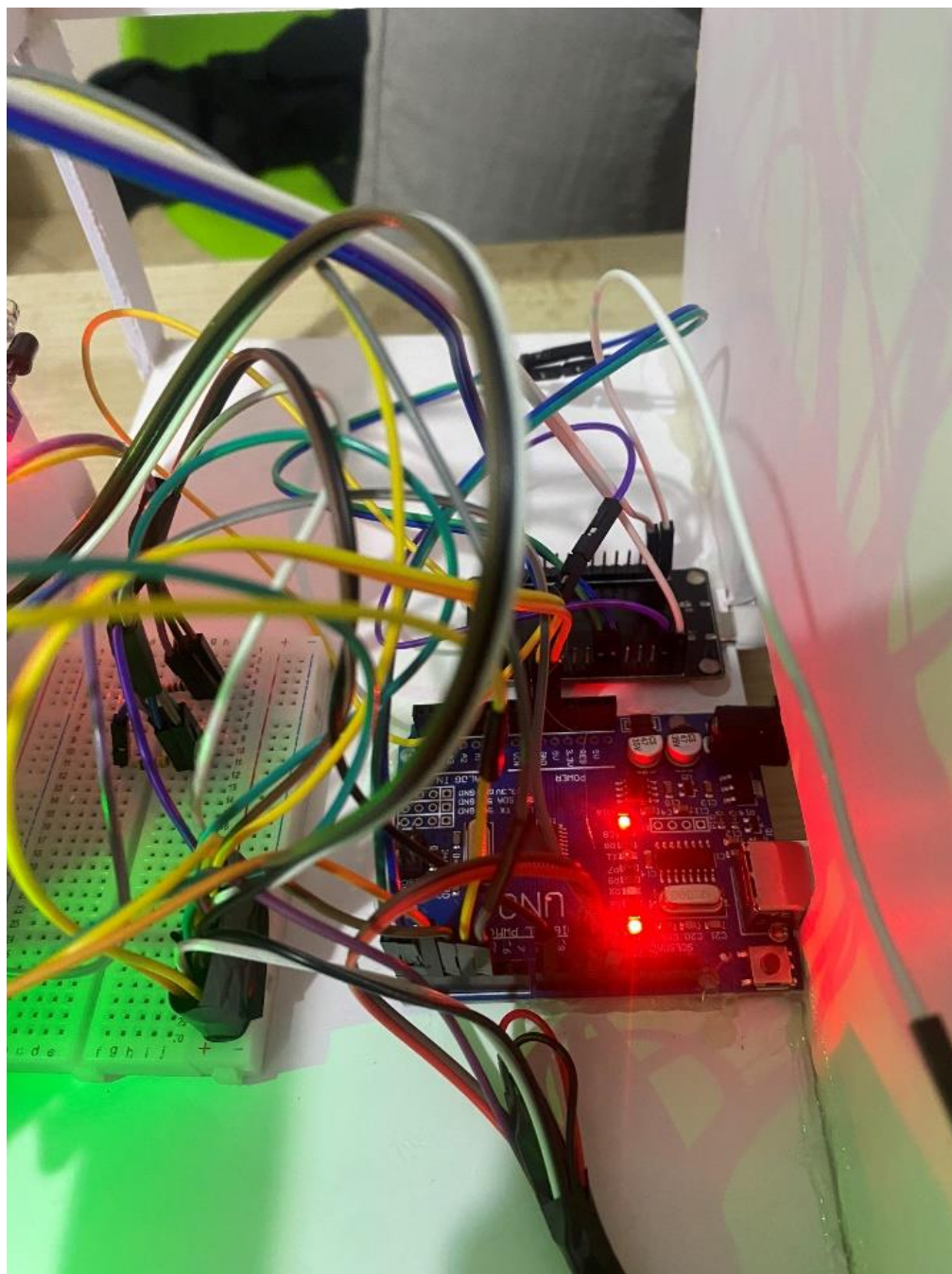


Hình 4. 2 Kết nối các linh kiện với nhau

Bước 2: Cấp nguồn 12V vào Arduino, ESP32, cảm biến hồng ngoại và module sim 800A.



Hình 4. 3 Module sim 800A sau khi được cấp nguồn

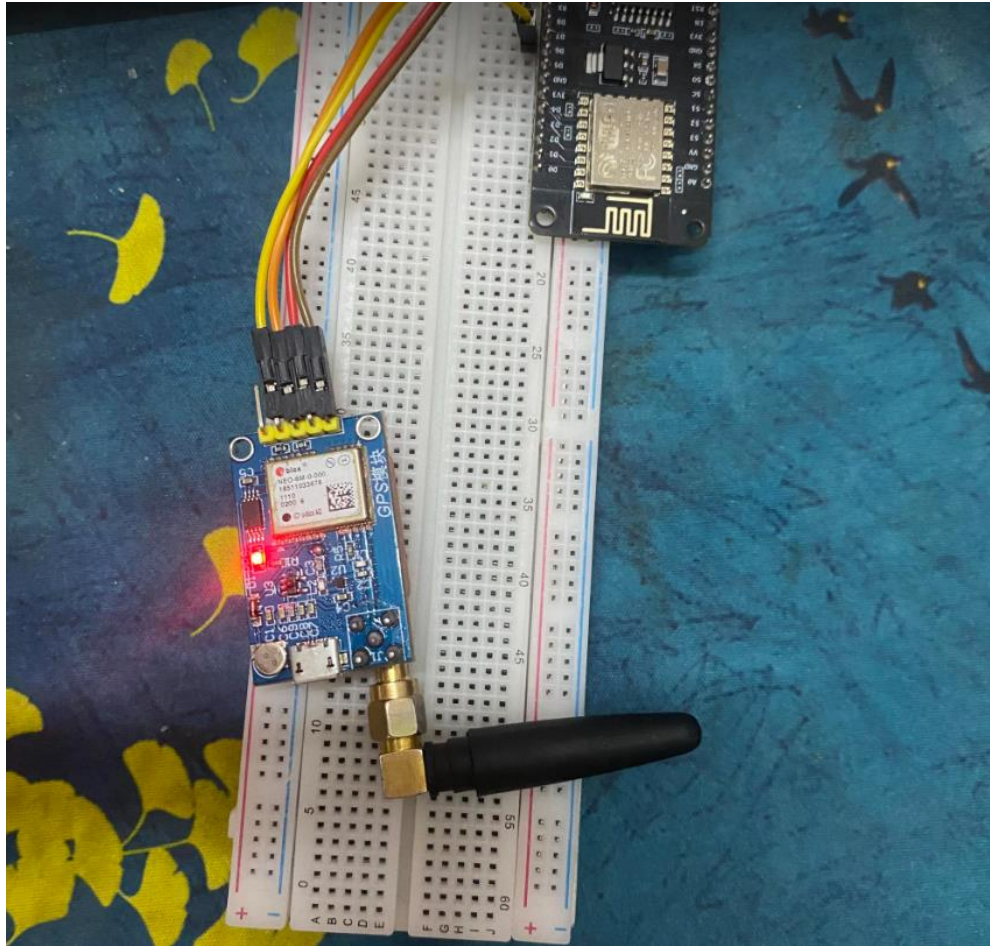


Hình 4. 4 Uno sau khi được cấp nguồn và chạy thử

Bước 3: Cuối cùng nạp chương trình và test chương trình có đạt như yêu cầu ban đầu không.

4.1.2 Sản phẩm mô hình hệ thống

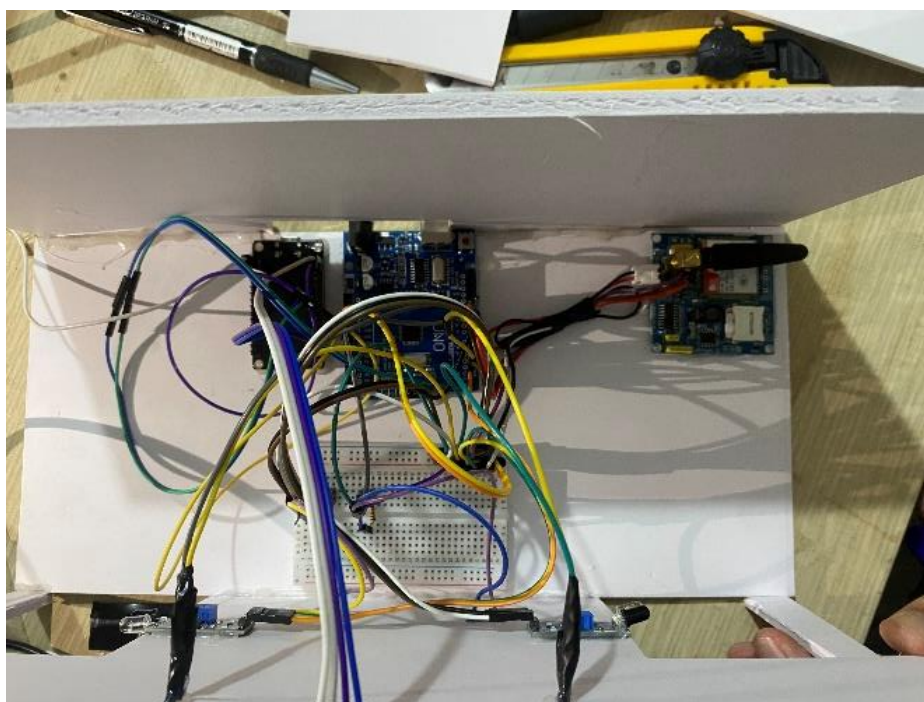
Sau đây là một số hình ảnh kết quả đạt được về phần cứng:



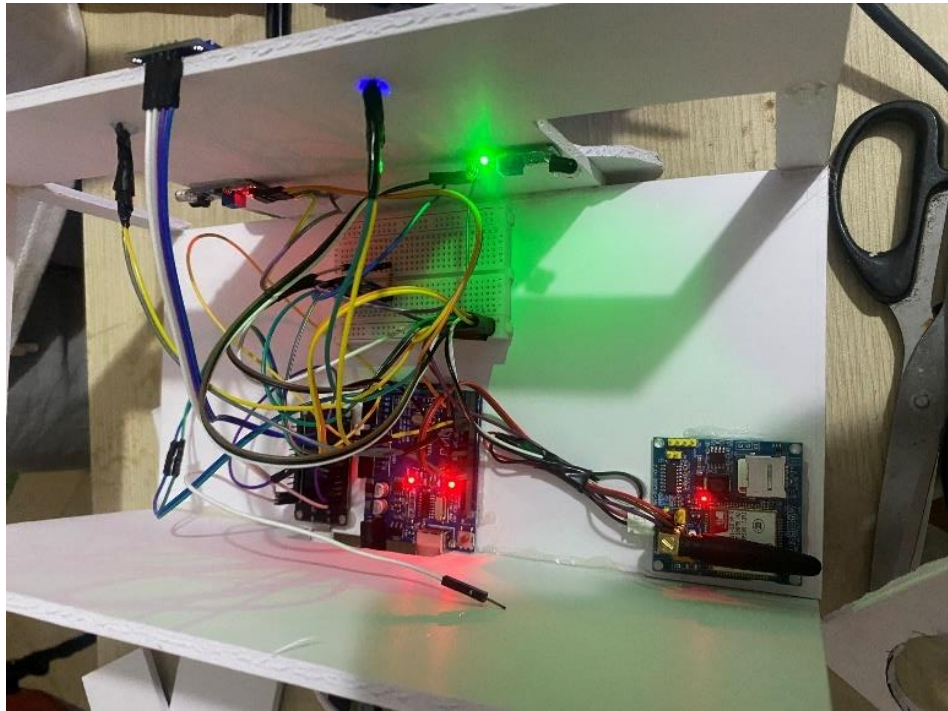
Hình 4. 5 Hình ảnh thực tế của module GPS khi được cấp nguồn



Hình 4. 6 Hình ảnh thực tế của module Sim 800A khi được cấp nguồn



Hình 4. 7 Sản phẩm khi chưa đóng gói

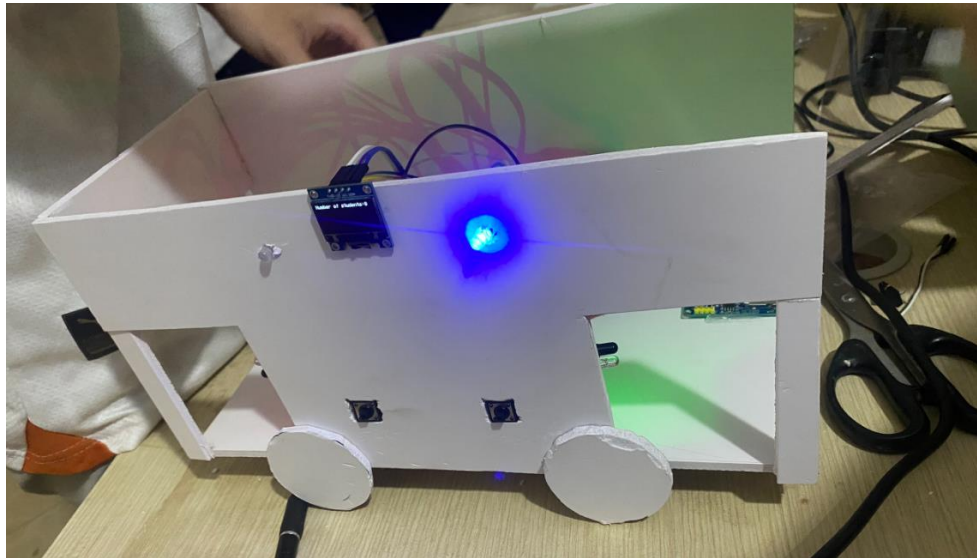


Hình 4. 8 Sản phẩm khi được cấp nguồn và chạy thử



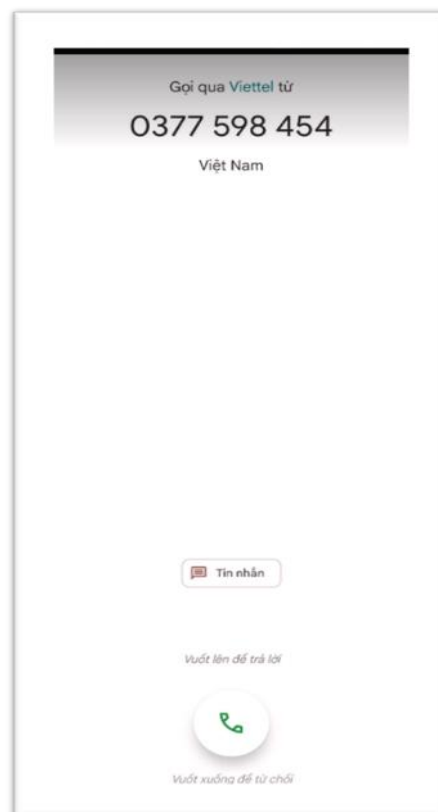
Hình 4. 9 Sản phẩm khi được gắn hoàn chỉnh vào mô hình

Khi tài xế kết thúc hành trình nhưng trên xe vẫn còn học sinh đèn cảnh báo lập tức sáng lên phát ra cảnh báo :



Hình 4. 10 Đèn cảnh báo bên ngoài xe sáng lên khi trên xe vẫn còn học sinh

Trong khi đèn sáng module sim cũng đồng thời thực hiện cuộc gọi thông báo còn trẻ bị bỏ quên trên xe đến tài xế:



Hình 4. 11 Cuộc gọi được module sim thực hiện để thông báo

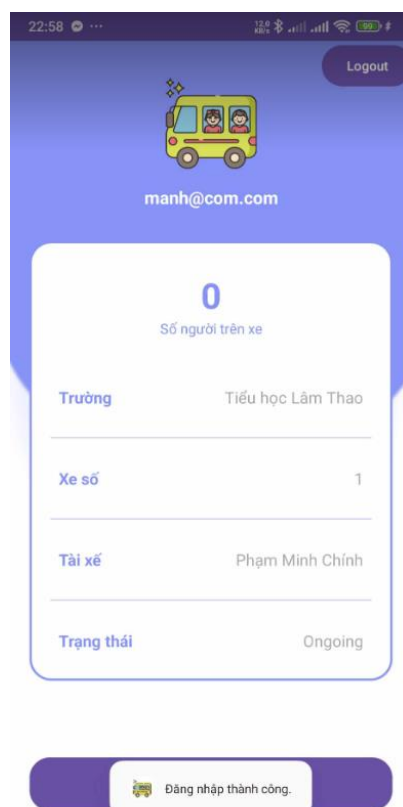
Về kết quả đạt được của phần mềm:

- Thiết kế được app có tính năng đăng nhập, đăng ký:



Hình 4. 12 Giao diện đăng nhập của ứng dụng theo dõi

Sau khi đăng nhập thành công ứng dụng sẽ chuyển đến giao diện hiển thị số người trên xe, tên tài xế, trạng thái của xe, tên tài xế, tên trường, số xe:



Hình 4. 13 Giao diện sau khi đăng nhập thành công

4.2 Tài liệu hướng dẫn sử dụng

Việc vận hành hệ thống rất đơn giản thông qua các bước sau:

Bước 1: Khi tài xế lên xe, trước khi xe bắt đầu tiến hành nhấn nút **start** để khởi động hệ thống. Khi khởi động hệ thống sẽ được cấp nguồn 12V , chờ 1p để hệ thống khởi động hoàn toàn và bắt đầu hành trình.

Bước 2: Kiểm tra hiển thị trên ứng dụng bằng cách truy cập vào ứng dụng, đăng nhập tài khoản, lúc ban đầu (chưa có học sinh lên xe) số học sinh trên xe được thiết lập là 0. Khi có học sinh lên xe sẽ đi qua cảm biến lúc này số người trên xe sẽ tăng lên 1, học sinh xuống xe đi qua cảm biến số học sinh trên xe sẽ giảm 1.

Bước 3: Trước khi tài xế xuống xe sẽ bấm vào nút **stop** để thông báo kết thúc hàng trình, nếu lúc này trên xe vẫn còn học sinh đèn cảnh báo sẽ sáng và hệ thống sẽ khởi tạo cuộc gọi đến tài xế.

4.3 Các kết quả và đánh giá

4.3.1 Các kết quả đạt được

Sau thời gian tiến hành tìm hiểu, nghiên cứu các tài liệu chuyên ngành, tìm hiểu thêm qua mạng Internet, tổng hợp lại các kiến thức đã được học trong trường cũng như được sự hướng dẫn của thầy GVHD Ts.Trần Văn Hưng. Nhóm chúng em cũng đã hoàn thành được đề tài nghiên cứu khoa học sinh viên “THIẾT KẾ HỆ THỐNG CẢNH BÁO AN TOÀN TRÊN XE ĐƯA ĐÓN HỌC SINH”.

Sau đề tài này, nhóm em cũng đã nghiên cứu và tích lũy được thêm nhiều hiểu biết, kiến thức mới như:

- Hiểu biết sâu hơn về sử dụng và các tính năng của Arduino như giao tiếp giữa Arduino với các module mở rộng như: cảm biến hồng ngoại, module Sim 800A, mạch Định Vị GPS NEO-6M V2, ESP 8266.
- Nghiên cứu và biết cách kết nối giữa Arduino với các cảm biến hồng ngoại, module Sim 800A, ESP32
- Nghiên cứu biết được cách sử dụng ,nguyên lý hoạt động của cảm biến, các thông số kỹ thuật, các tính năng của cảm biến.
- Nghiên cứu biết được cách sử dụng module Sim 800A, nguyên lý hoạt động, các thông số kỹ thuật, tính năng của Sim 800A. Biết được cách thiết lập cho module sim, sử dụng tính năng gọi điện đến điện thoại

4.3.2 Thử nghiệm hệ thống

Thử nghiệm hệ thống được thực hiện khi hoàn tất thiết kế phần cứng và phần mềm. Nhóm đã thử nghiệm với các chức năng cập nhật dữ liệu học sinh lên xuống xe từ hệ thống phần cứng lên App, chức năng gọi điện đến số điện thoại cài đặt trước khi phát hiện trẻ bị bỏ quên trên xe. Tính năng cập nhật lộ trình của phương tiện theo thời gian thực vẫn chưa được hoàn thiện.

Sau khi chạy thử hệ thống nhóm đưa ra được số liệu theo bảng sau:

Bảng 4. 1 Thời gian số học sinh được cập nhật lên App (số liệu với 10 lần chạy thử)

Lần chạy thử	Thời gian cập nhật lên ứng dụng	Lần chạy thử	Thời gian cập nhật lên ứng dụng
Lần 1	14s	Lần 6	12s
Lần 2	16s	Lần 7	11s
Lần 3	20s	Lần 8	15s
Lần 4	18s	Lần 9	17s
Lần 5	22s	Lần 10	20s

Bảng 4. 2 Thời gian delay cho đến khi nhận được cuộc gọi (số liệu với 10 lần chạy thử)

Lần chạy thử	Thời gian delay cho đến khi có cuộc gọi	Lần chạy thử	Thời gian delay cho đến khi có cuộc gọi
Lần 1	7s	Lần 6	8s
Lần 2	6s	Lần 7	7s
Lần 3	5s	Lần 8	8s
Lần 4	6s	Lần 9	9s
Lần 5	8s	Lần 10	9s

Chạy thử được thực hiện bên trong PTN nên tín hiệu Wifi không ổn định. Thời gian cập nhật số lượng học sinh và thực hiện cuộc gọi có thể nếu ở điều kiện tín hiệu wifi tốt hơn.

4.3.3 Đánh giá

Đề tài đã thiết kế hoàn thiện được mô hình phần cứng và phần mềm điều khiển bao gồm phần mềm nhúng và phần mềm trên điện thoại. Đề tài đã đạt được mục tiêu ban đầu đề ra.

Hệ thống dễ dàng kiểm tra, và gửi tin nhắn về điện thoại đã kết nối, các cảm biến hoạt động tốt, đèn Led cảnh báo hoạt động ổn định đúng chức năng yêu cầu. Người dùng thao tác một cách đơn giản, dễ sử dụng.

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

A. Kết luận

Sau thời gian tìm hiểu, nghiên cứu và thực hiện mô hình, nhiệm vụ đề tài cơ bản được hoàn thành. Bằng sự cố gắng của bản thân, học hỏi từ Thầy Cô, bạn bè, nhóm đã hoàn thành đề tài “**Thiết kế hệ thống cảnh báo an toàn trên xe đưa đón học sinh**”. Đề tài đã hoàn thành mục tiêu đề ra, cụ thể:

- Xây dựng được mô hình phần cứng, phần mềm điều khiển cho hệ thống.
- Hệ thống chạy ổn định, hoàn thành 100% mục tiêu ban đầu.
- Phản hồi của hệ thống cảnh báo nhanh.
- Tương tác giữa điện thoại và module sim ổn định.

B. Hướng phát triển

- Tích hợp hệ thống và đóng gói sản phẩm: làm sản phẩm có kích thước nhỏ gọn, an toàn khi gắn trên phương tiện, nâng cao tính thẩm mỹ cho hệ thống.
- Hoàn thiện tính năng giám sát vị trí và lộ trình phương tiện thông qua Module GPS, cập nhật lên bản đồ thời gian thực.
- Thêm các AI giúp dự báo nguy hiểm từ các phương tiện khác.
- Nâng cấp hệ thống cảm biến bằng hệ thống nhận diện khuôn mặt và kiểm soát trên xe thông qua camera.

TÀI LIỆU THAM KHẢO

- ĐỀ TÀI TỐT NGHIỆP NGÀNH CÔNG NGHỆ KỸ THUẬT ĐIỆN TỬ TRUYỀN THÔNG ĐỀ TÀI: GIÁM SÁT VÀ CẢNH BÁO HOẠT ĐỘNG PHƯƠNG TIỆN VẬN TẢI Ô TÔ GVHD: Th.S Nguyễn Ngô Lâm

- ĐỀ TÀI TỐT NGHIỆP NGÀNH ĐIỆN TỬ TRUYỀN THÔNG ĐỀ TÀI : THIẾT KẾ HỆ THỐNG THÔNG MINH CẢNH BÁO CHO XE MÁY NGƯỜI THỰC HIỆN NGUYỄN QUANG HUY , DƯƠNG VĂN HUÂN

- <https://topdev.vn/blog/tong-quan-ve-ngon-ngu-lap-trinh-java>

- Wikipedia, Hệ thống định vị toàn cầu (sửa lần cuối vào 24 tháng 4 năm 2018), <http://vi.wikipedia.org/>

- Tài liệu tham khảo về datasheet linh kiện, <https://www.alldatasheet.com/>

PHỤ LỤC

CODE NHÚNG

1. Đọc giá trị cảm biến

```
giatri1 = digitalRead(cambien1);

if (giatri1 == LOW && pre1 == HIGH) {

    dem++;

    delay(1000);

}

pre1 = giatri1;

giatri2 = digitalRead(cambien2);

if (giatri2 == LOW && pre2 == HIGH) {

    dem--;

    delay(1000);

}

pre2 = giatri2;

if (dem < 0) dem = 0;

pre_dem = dem;
```

2. Truyền dữ liệu từ UNO sang ESP32

```
if (Serial.availableForWrite()) {

    Serial.println(dem);

}
```

3. Xét trạng thái các nút bấm

```
buttonState1 = digitalRead(button1);

if (buttonState1 != lastButtonState1) {

    if (buttonState1 == HIGH) {

        digitalWrite(led1, HIGH);

        digitalWrite(led2, LOW);

        button2State = false;

        if (!isButtonPressed1 && dem != 0) {

            isButtonPressed1 = true;

            callUp(phone_no);

        }else {

            if (isButtonPressed1) {

                isButtonPressed1 = false;

                button2State = false;

            }

        }

    }

    else {

        isButtonPressed1 = false;

        digitalWrite(led1, LOW);

    }

    delay(50);

}
```

```
buttonState2 = digitalRead(button2);  
if (buttonState2 != lastButtonState2) {  
    if (buttonState2 == HIGH) {  
        if (button2State == false) {  
            digitalWrite(led2, HIGH);  
            dem = 0;  
            pre1 = HIGH;  
            pre2 = HIGH;  
            called = false;  
            pre_dem = 0;  
            button2State = true;  
        }  
    }  
    else {  
        digitalWrite(led2, LOW);  
        button2State = false;  
    }  
}  
delay(50);  
}  
  
lastButtonState2 = buttonState2;  
  
lastButtonState1 = buttonState1;
```

4. Hàm setup UNO

```
void setup() {  
    pinMode(button1, INPUT);  
    pinMode(led1, OUTPUT);  
    digitalWrite(led1, LOW);  
    pinMode(button2, INPUT);  
    pinMode(led2, OUTPUT);  
    digitalWrite(led2, LOW);  
    Serial.begin(9600);  
    GSM.begin(9600);  
    pinMode(cambien1, INPUT);  
    pinMode(cambien2, INPUT);  
    Serial.println("Initializing....");  
    initModule("AT", "OK", 1000);  
}
```

5. Hàm setup cho ESP32

```
void setup(){  
    Serial.begin(9600);  
    Serial2.begin(9600, SERIAL_8N1, RXp2, TXp2);  
    if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {  
        Serial.println(F("SSD1306 allocation failed"));  
        for (;;);  
    }
```

```
}

display.display();

WiFi.begin(WIFI_SSID, WIFI_PASSWORD);

Serial.print("Connecting to Wi-Fi");

while (WiFi.status() != WL_CONNECTED){

    Serial.print(".");

    delay(300);

}

Serial.println();

Serial.print("Connected with IP: ");

Serial.println(WiFi.localIP());

Serial.println();

config.api_key = API_KEY;

config.database_url = DATABASE_URL;

if (Firebase.signUp(&config, &auth, "", "")){

    Serial.println("ok");

    signupOK = true;

}

else

    Serial.printf("%s\n", config.signer.signupError.message.c_str());

config.token_status_callback = tokenStatusCallback;

Firebase.begin(&config, &auth);

Firebase.reconnectWiFi(true);
```

```
    delay(5000);  
}
```

6. Hàm gửi dữ liệu từ ESP32 lên Firebase

```
void send_data_firebase () {  
  
    if (Firebase.ready() && signupOK && (millis() - sendDataPrevMillis >  
    15000 || sendDataPrevMillis == 0)){  
  
        sendDataPrevMillis = millis();  
  
        if (Firebase.RTDB.setInt(&fbdo, "xe1/onbus", so_hoc_sinh)){  
  
            Serial.println("PASSED" );  
  
            Serial.println("PATH: " + fbdo.dataPath());  
  
            Serial.println("TYPE: " + fbdo.dataType());  
  
        }  
  
        else {  
  
            Serial.println("FAILED");  
  
            Serial.println("REASON: " + fbdo.errorReason());  
  
        }  
  
    }  
  
}
```

7. Đọc giá trị từ UNO

```
if (Serial2.available() > 0) { // Kiểm tra xem có dữ liệu nhận được không
```

```
so_hoc_sinh = Serial2.parseInt(); // Đọc giá trị được gửi từ Board Arduino
#1

Serial.print("Number of students: ");

Serial.println(so_hoc_sinh); // In giá trị đã nhận lên Serial Monitor

}
```

8. Hiện thị lên OLED

```
Serial.print("Number of students: ");

Serial.println(so_hoc_sinh);

display.clearDisplay();

display.setTextSize(1);

display.setTextColor(WHITE);

display.setCursor(0, 0);

display.print(F("Number of students:"));

display.println(so_hoc_sinh);

display.display();

send_data_firebase();

delay(500);
```

CODE JAVA

1. Phần đăng ký

```
private void registerUser(String email, String password) {

    auth.createUserWithEmailAndPassword(email,
password).addOnCompleteListener(new
OnCompleteListener<AuthResult>() {

        @Override

        public void onComplete(@NonNull Task<AuthResult> task) {

            if (task.isSuccessful()) {

                // Sign in success, update UI with the signed-in user's information

                //FirebaseUser user = mAuth.getCurrentUser();

                Toast.makeText(RegisterActivity.this, "Tạo tài khoản thành
công.", Toast.LENGTH_SHORT).show();

                Intent intent = new
Intent(getApplicationContext(),PeoDisplayActivity.class);

                startActivity(intent);

                finish();

            } else {

                // If sign in fails, display a message to the user.

                Toast.makeText(RegisterActivity.this, "Tạo tài khoản thất bại.",
Toast.LENGTH_SHORT).show();

            }

        }

    });
}
```



```
}
```

2. Phần đăng nhập

```
private void loginUser(String email, String password) {  
    auth.signInWithEmailAndPassword(email,  
password).addOnCompleteListener(new OnCompleteListener<AuthResult>()  
{  
    @Override  
    public void onComplete(@NonNull Task<AuthResult> task) {  
        if (task.isSuccessful()) {  
            Toast.makeText(LoginActivity.this, "Đăng nhập thành  
công.", Toast.LENGTH_SHORT).show();  
            Intent intent = new  
Intent(getApplicationContext(),PeoDisplayActivity.class);  
            startActivity(intent);  
            finish();  
        } else {  
            Toast.makeText(LoginActivity.this, "Email hoặc mật khẩu  
không chính xác.", Toast.LENGTH_SHORT).show();  
        }  
    }  
});  
}
```

3. Đọc dữ liệu từ Firebase

```
private void readDatabaseN(TextView display_data, String child, String
parent){

    // Read from the database

    FirebaseDatabase database = FirebaseDatabase.getInstance();

    DatabaseReference myRef = database.getReference(parent).child(child);

    myRef.addValueEventListener(new ValueEventListener() {

        @Override

        public void onDataChange(@NonNull DataSnapshot dataSnapshot) {

            // This method is called once with the initial value and again

            // whenever data at this location is updated.

            Integer value = dataSnapshot.getValue(Integer.class);

            // Hiển thị giá trị lên TextView

            display_data.setText(String.valueOf(value));

        }

        @Override

        public void onCancelled(@NonNull DatabaseError error) {

            // Failed to read value

        }

    });

}
```

