**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC NGUYỄN TẤT THÀNH**

**-------oOo-------**

**BÁO CÁO TỔNG KẾT ĐỀ TÀI**

**CHƯƠNG TRÌNH SINH VIÊN NCKH NĂM 2021**

**TÊN ĐỀ TÀI: Xe lăn điện**

**Mã số đề tài:**

**Chủ nhiệm đề tài**

**Giảng viên hướng dẫn:**

**Khoa:**

**Các thành viên tham gia:**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Họ và tên** | **MSSV** | **Lớp** |
| 1 |  |  |  |
| 2 |  |  |  |

*TP.* *Hồ Chí Minh, tháng năm 20*

**MỤC LỤC**

**CHƯƠNG 1. TỔNG QUAN**

**CHƯƠNG 2. NỘI DUNG VÀ PHƯƠNG PHÁP NGHIÊN CỨU**

**CHƯƠNG 3. KẾT QUẢ VÀ SẢN PHẨM ĐẠT ĐƯỢC**

**CHƯƠNG 4. KẾT LUẬN**

**TÀI LIỆU THAM KHẢO**

**DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT**

**DANH MỤC CÁC BẢNG BIỂU, SƠ ĐỒ, HÌNH ẢNH**

**CHƯƠNG 1. TỔNG QUAN**

* 1. Tính cấp thiết và mục tiêu nghiên cứu của đề tài.

An toàn là một vấn đề phổ biến liên quan đến việc sử dụng xe lăn, với một nghiên cứu nhấn mạnh rằng 55% người sử dụng xe lăn cho biết đã gặp ít nhất một vụ va chạm và 17% cho biết đã gặp hai vụ va chạm trở lên trong khoảng thời gian ba năm. Đặc biệt, người già là đối tượng xảy ra các tai nạn khi sử dụng xe lăn chiếm tỷ lệ rất cao. Các tai nạn này xảy ra có thể do nguyên nhân xe bị đổ dốc với tốc độ quá nhanh, xe di chuyển đến các điểm mù hoặc hố sâu dẫn đến mất thăng bằng và ngã xe. Một vấn đề nghiêm trọng khác chính là những người già ở một mình và không thể có sự trợ giúp kịp thời khi xe bị ngã.

Vì vậy nhu cầu đi lại cho người khuyết tật cùng với sự phát triển của công nghệ đã mở ra nhiều cơ hội mới để tạo ra các loại xe lăn điện thông minh và tiện ích hơn, giúp cải thiện tính di động và độc lập của người sử dụng. Tuy nhiên, hiện nay một số xe lăn điện trên thị trường thường chỉ có thể di chuyển 4 hướng hoặc 8 hướng, vì thế xe lăn điện này được làm ra để có thể cho người dùng cảm giác thoải mái nhất khi điều khiển. Bên cạnh đó, trong đề tài này, xe lăn cũng được tích hợp thêm các cảm biến cùng nhiều chế độ khác nhau nhằm tăng tính thuận tiện và an toàn cho người dùng:

* 1. Khả năng ứng dụng thực tiễn.

Xe lăn điện có rất nhiều ứng dụng vô cùng đa dạng và có ảnh hướng lớn đến nhiều lĩnh vực của cuộc sống. Điển hình là:

* Di Chuyển Cá Nhân: Xe lăn điện cung cấp phương tiện di chuyển cá nhân linh hoạt cho những người có khuyết tật hoặc hạn chế về sức khỏe. Chúng giúp họ dễ dàng di chuyển trong các không gian công cộng, như siêu thị, công viên, hoặc khu dân cư, mà không cần phụ thuộc vào sự giúp đỡ của người khác.
* Y Tế và Chăm Sóc Sức Khỏe: Xe lăn điện đóng vai trò quan trọng trong việc cung cấp các dịch vụ y tế và chăm sóc sức khỏe đến những người có khuyết tật hoặc khó khăn về việc di chuyển. Chúng được sử dụng trong các bệnh viện, phòng khám và cơ sở chăm sóc dài hạn để hỗ trợ di chuyển và vận chuyển bệnh nhân.

**CHƯƠNG 2. NỘI DUNG VÀ PHƯƠNG PHÁP NGHIÊN CỨU**

Bảng 1. Danh mục các thiết bị và chức năng được sử dụng trong đề tài.

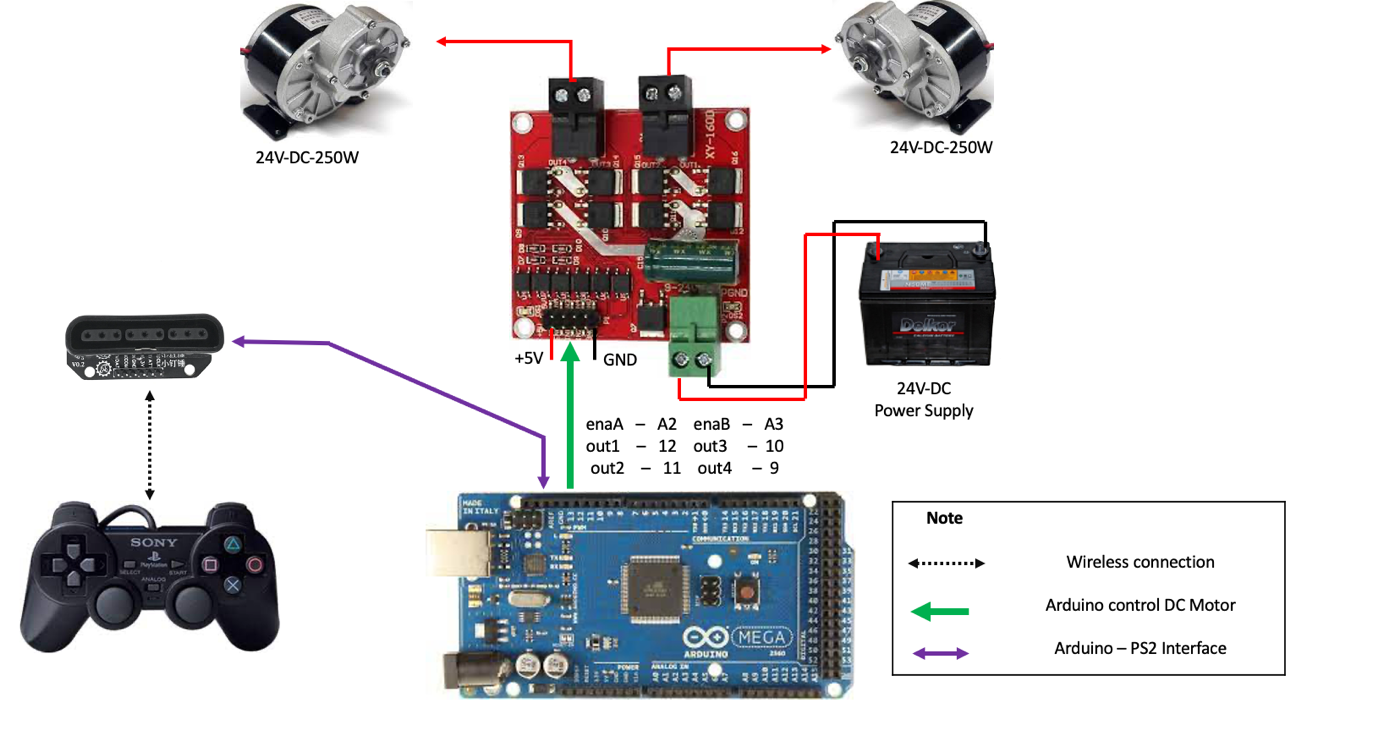
|  |  |  |  |
| --- | --- | --- | --- |
| TT | THIẾT BỊ | CHỨC NĂNG | HÌNH ẢNH THỰC TẾ |
| 1 | Arduino Mega | Dùng để kết nối với các cảm biến và đọc tín hiệu từ đó đưa ra các đầu ra tương ứng. |  |
| 2 | Motor (x2) | Dùng để làm xe chạy |  |
| 3 | Cảm biến siêu âm  (x) | Dùng để đo khoảng cách từ xe đến các vật xung quanh |  |
| 4 | Driver | Dùng để điều khiển hướng đi của xe và tốc độ của xe theo ý muốn của người sử dụng |  |
| 5 | Cảm biến dò line  TCRT5000 | Dùng để điều khiển xe sao cho xe chạy theo một đường đã được định sẵn |  |
| 6 |  |  |  |
| 7 |  |  |  |
| 8 |  |  |  |
|  |  |  |  |

Bảng 2. Danh mục các chân được sử dụng và chức năng

| TT | ĐỊA CHỈ | TÊN | CHỨC NĂNG CỦA CHÂN |
| --- | --- | --- | --- |
| 1 | A2 | enaA | Phát xung cho motor trái |
| 2 | A3 | enaB | Phát xung cho motor phải |
|  | 14 | out1 | Điều khiển motor trái |
|  | 15 | out2 | Điều khiển motor trái |
|  | 16 | out3 | Điều khiển motor phải |
|  | 17 | out4 | Điều khiển motor phải |
|  | 8 | lineSensor1 | Đọc tín hiểu của cảm biến dò line 1 |
|  | 7 | lineSensor2 | Đọc tín hiểu của cảm biến dò line 2 |
|  | 6 | lineSensor3 | Đọc tín hiểu của cảm biến dò line 3 |
|  | 5 | lineSensor4 | Đọc tín hiểu của cảm biến dò line 4 |
|  | 4 | lineSensor5 | Đọc tín hiểu của cảm biến dò line 5 |
|  | 12 | PS2\_DAT |  |
|  | 13 | PS2\_CLK |  |
|  | 11 | PS2\_SEL |  |
|  | 10 | PS2\_CMD |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

2.2 Hệ thống điều khiển bằng Play Station 2 ( PS2)

Trong phần này trình bày nguyên lí vận hành xe lăn được điều khiển bởi tay cầm ps2. Xe có thể di chuyển tiến hoặc lùi và có thể thay đổi tốc độ dựa trên 2 joystick của ps2. Sơ đồ nối dây thiết bị được trình bày trong hình. ….

****

**Sơ đồ nối dây của vi điều khiển với bộ điều khiển không dây ps2**

****

Các nút điều khiển hai bánh xe của xe lăn

Nguyên lí

* Mỗi một joystick có thể một bánh xe tiến lùi hoặc tăng tốc và giảm tốc.

Cách điều khiển 2 bánh xe được mô tả ngắn gọn trong bảng ….

* Khi người dùng muốn xe đi thẳng thì đẩy cả hai joystick trái và phải cùng về phía trước. Đoạn code sau xử lí cho phép xe đi thẳng:

if ( leftStickY <= 127 ) {

forwardMotor1();

speedMotorA = map(abs(127-leftStickY), 0, 127, 0, 170);

}

* Ngược lại, khi người dùng muốn xe đi lùi thì đẩy cả hai joystick trái và phải cùng về phía sau.

if ( leftStickY >= 127 ) {

backwardMotor1();

speedMotorA = map(abs(127-leftStickY), 0, 127, 0, 170);

}

* Khi người dùng muốn xe rẽ trái thì đẩy joystick trái về phía sau và joystick phải về phía trước hoặc chỉ đẩy mỗi joystick phải về phía trước còn joystick trái giữ nguyên.

if ( rightStickY <= 127) {

forwardMotor2();

speedMotorB = map(abs(127-rightStickY), 0, 127, 0, 170);

}

* Tương tự đối với rẽ phải, khi người dùng muốn xe rẽ phải thì đẩy joystick phải về phía sau và joystick trái về phía trước hoặc chỉ đẩy mỗi joystick trái về phía trước còn joystick phải giữ nguyên

if ( rightStickY >= 127 ) {

backwardMotor2();

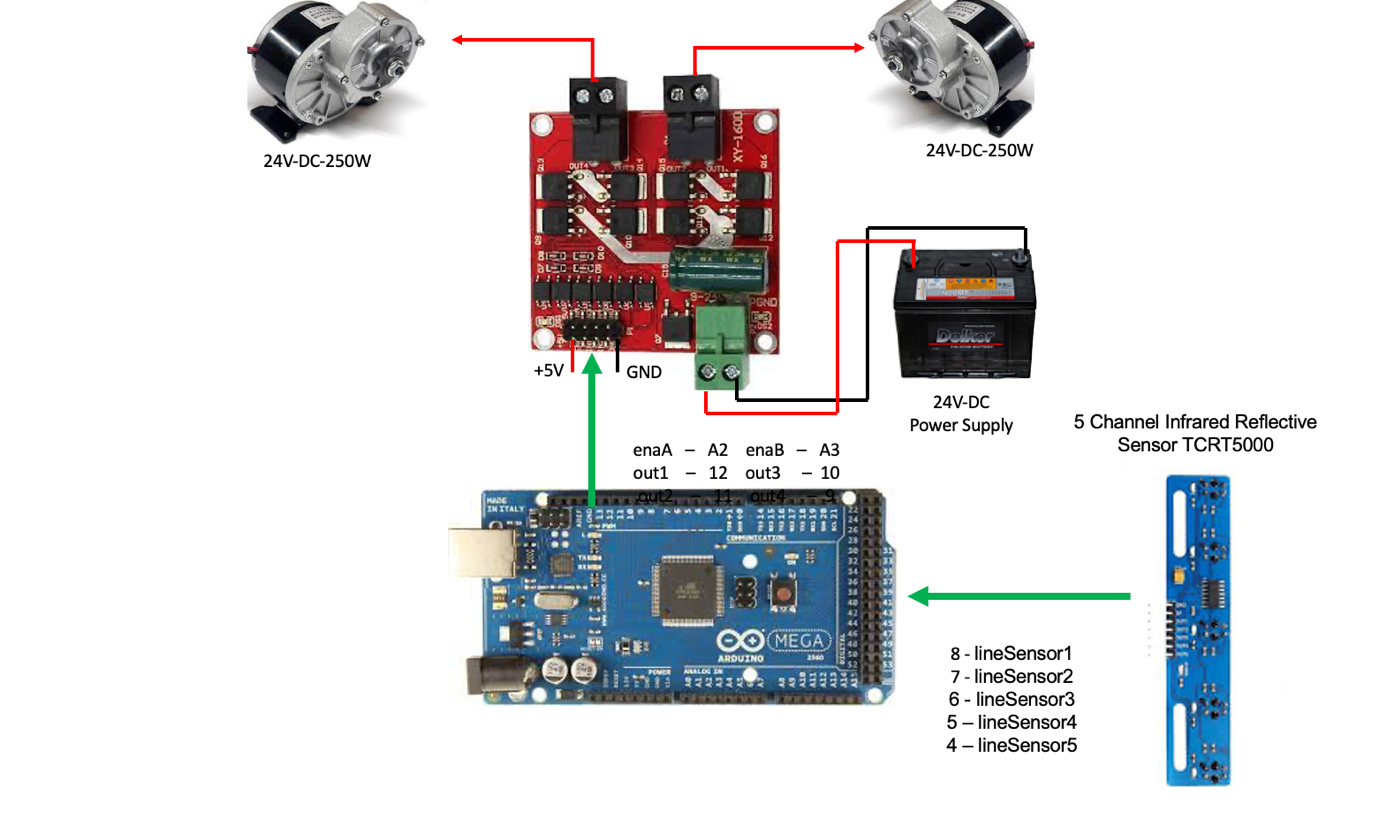
speedMotorB = map(abs(127-rightStickY), 0, 127, 0, 170);

}

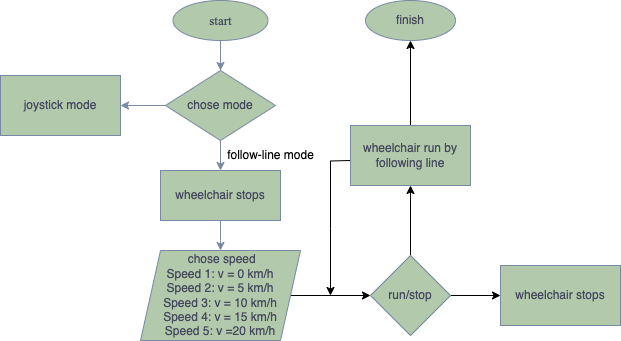
Tốc độ xe sẽ phụ thuộc vào khoảng từ núm đến gốc ( càng đẩy ra xa gốc, thì tốc độ xe sẽ càng nhanh).

**2.3 Hệ thống điều khiển xe tự hành theo line**

**2.3.1 Sơ đồ đấu dây và lưu đồ giải thuật của chế độ tự hành theo line.**



**Sơ đồ nối dây của vi điều khiển với cảm biến dò line TCRT5000.**



**Lưu đồ giải thuật nguyên lý vận hành xe ở chế độ tự hành theo line.**

Trong hình số , sơ đồ trình bày nguyên lý vận hành xe khi ở chế độ tự hành theo line. Khi khởi động hệ thống thì cho phép lựa chọn chế độ bằng tay ( sử dụng joystick ) hoặc chế độ tự hành theo line. Nếu người dùng lựa chọn chế độ tự hành theo line thì khi bật lên xe sẽ có tốc độ ban đầu bằng 0 ( xe dừng ). Lúc này, người dùng có thể lựa chọn tốc độ bằng 2 nút nhấn up ( tăng ) và down ( giảm ) với 5 cấp tốc độ cụ thể như sau:

* Cấp tốc độ 1: tốc độ xe bằng 0 ( xe dừng ).
* Cấp tốc độ 2: tốc độ xe bằng 5 km/h.
* Cấp tốc độ 3: tốc độ xe bằng 10 km/h
* Cấp tốc độ 4: tốc độ xe bằng 15 km/h.
* Cấp tốc độ 5: tốc độ xe bằng 20 km/h.

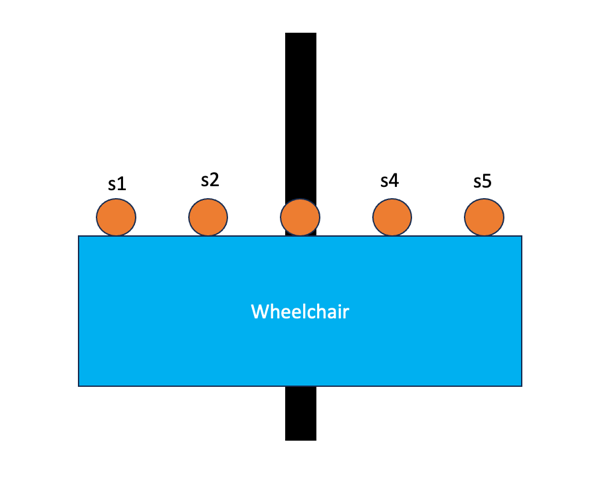
Sau khi chọn tốc độ thì người sử dụng có thể cho phép xe chạy bằng cách chuyển công tắc sang chế độ run ( chạy ). Trong quá trình xe chạy, có thể tiếp tục tăng hoặc giảm tốc độ bằng 2 nút nhấn up ( tăng ) và down ( giảm ) . Nếu đang chạy mà nhấn stop ( dừng ) thì tốc độ động cơ sẽ giảm dần về 0 trong vòng 5s.

**2.3.2 Nguyên lý điều hướng cho xe sử dụng cảm biến dò line TCRT5000**

Phần này trình bày nguyên lý cảm biến TCRT5000 phát hiện đường line dựa trên giá trị trả về từ 5 kênh cảm biến. Nếu như cảm biến phát hiện đường line, cảm biến sẽ trả về giá trị mức 0, nếu cảm biến không phát hiện đường line thì sẽ trả về mức 1.

Nguyên lý bám theo line của xe lăn tự hành gồm có 5 trường hợp sau đây.

* **Xe đi thẳng:**

Trong trường hợp này, cảm biến chân s3 phát hiện line thì trả về mức 0, các cảm biến còn lại trả về mức 1, hình mô tả tình trạng của xe so với line .   


**Xe đi thẳng**

Chương trình xử lý tín hiệu cảm biến và điều khiển 2 bánh xe như sau:

if((s1 == 1) && (s2 == 1) && (s3 == 0) && (s4 == 1) && (s5 == 1)) {

analogWrite(e1, speed);

analogWrite(e2, speed);

digitalWrite(m1, HIGH);

digitalWrite(m2, LOW);

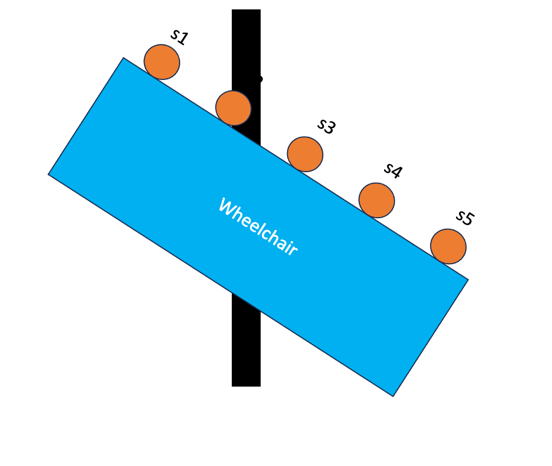
digitalWrite(m3, HIGH);

digitalWrite(m4, LOW);

}

Trong đoạn code trên khi cảm biến giữa phát hiện line thì sẽ điều khiển 2 bánh cùng đi thẳng.

* **Xe lệch phải cấp 1:**

Trong trường hợp này, cảm biến chân s2 phát hiện line thì trả về mức 0, các cảm biến còn lại trả về mức 1, hình mô tả tình trạng của xe so với line .   


**Xe chạy lệch phải cấp 1**

Chương trình xử lý tín hiệu cảm biến và điều khiển 2 bánh xe như sau:

if((s1 == 1) && (s2 == 0) && (s3 == 1) && (s4 == 1) && (s5 == 1)) {

analogWrite(e1, speed);

analogWrite(e2, speed);

digitalWrite(m1, LOW);

digitalWrite(m2, LOW);

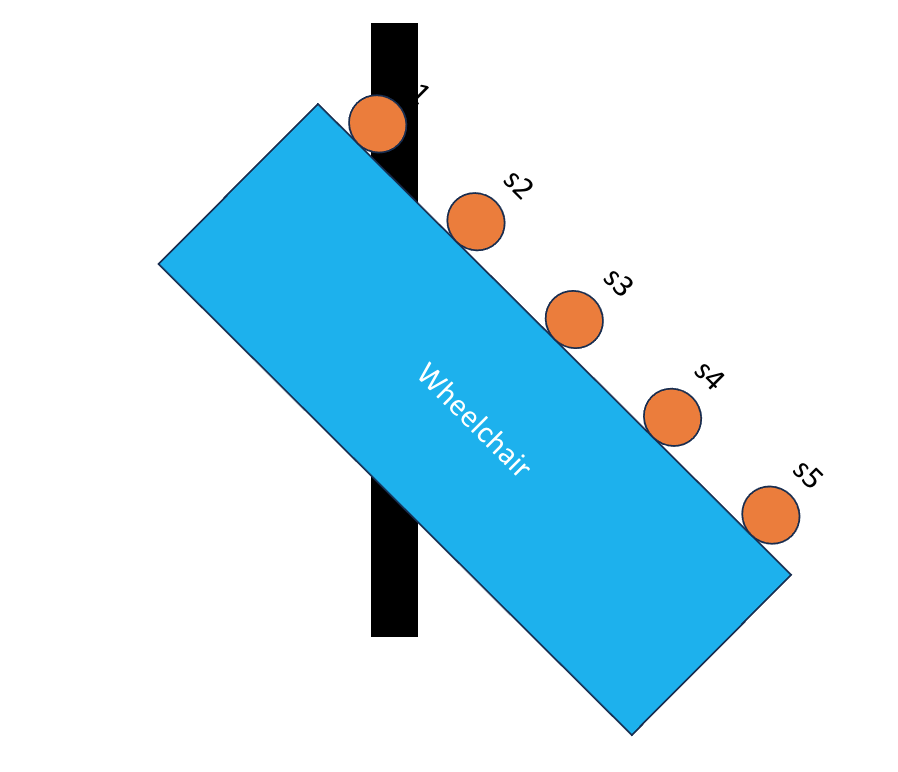
digitalWrite(m3, HIGH);

digitalWrite(m4, LOW);

}

Trong đoạn code trên khi xe bị lệch sang phải thì động cơ bên trái dừng còn động cơ bên phải chạy để xe điều hướng sang bên trái.

* **Xe lệch phải cấp 2:**

Trong trường hợp này, cảm biến chân s1 phát hiện line thì trả về mức 0, các cảm biến còn lại trả về mức 1, hình mô tả tình trạng của xe so với line .   


**Xe lệch phải cấp 2**

Chương trình xử lý tín hiệu cảm biến và điều khiển 2 bánh xe như sau:

if((s1 == 0) && (s2 == 1) && (s3 == 1) && (s4 == 1) && (s5 == 1)) {

analogWrite(e1, speed);

analogWrite(e2, speed);

digitalWrite(m1, LOW);

digitalWrite(m2, HIGH);

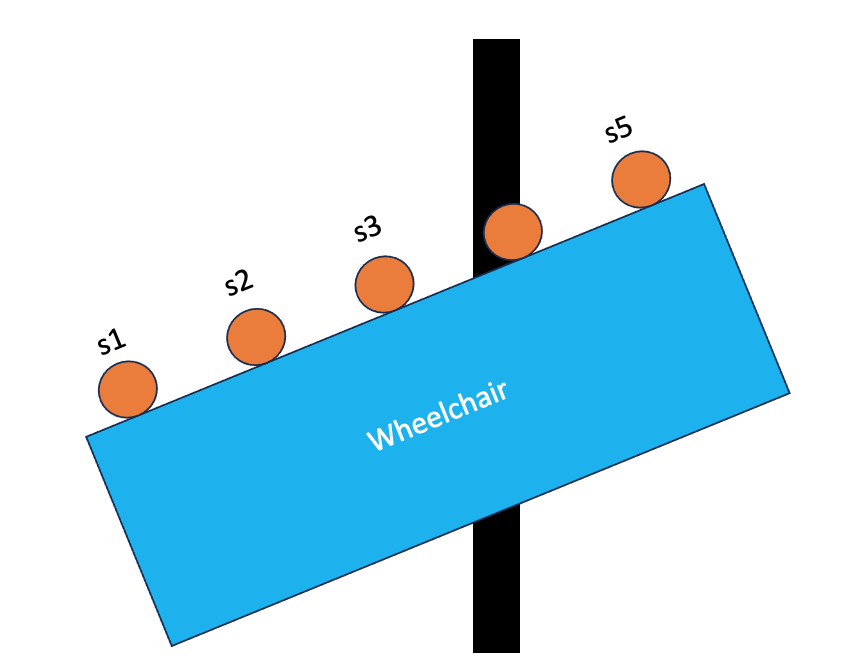
digitalWrite(m3, HIGH);

digitalWrite(m4, LOW);

}

Trong đoạn code trên khi xe bị lệch hoàn toàn sang bên phải thì động cơ 1 sẽ chạy lùi và động cơ 2 sẽ chạy thẳng để điều hướng xe quay phải.

* **Xe lệch trái cấp 1:**

Trong trường hợp này, cảm biến chân s4 phát hiện line thì trả về mức 0, các cảm biến còn lại trả về mức 1, hình mô tả tình trạng của xe so với line .   


**Xe lệch trái cấp 1**

Chương trình xử lý tín hiệu cảm biến và điều khiển 2 bánh xe như sau:

if((s1 == 1) && (s2 == 1) && (s3 == 1) && (s4 == 0) && (s5 == 1)) {

analogWrite(e1, speed);

analogWrite(e2, speed);

digitalWrite(m1, HIGH);

digitalWrite(m2, LOW);

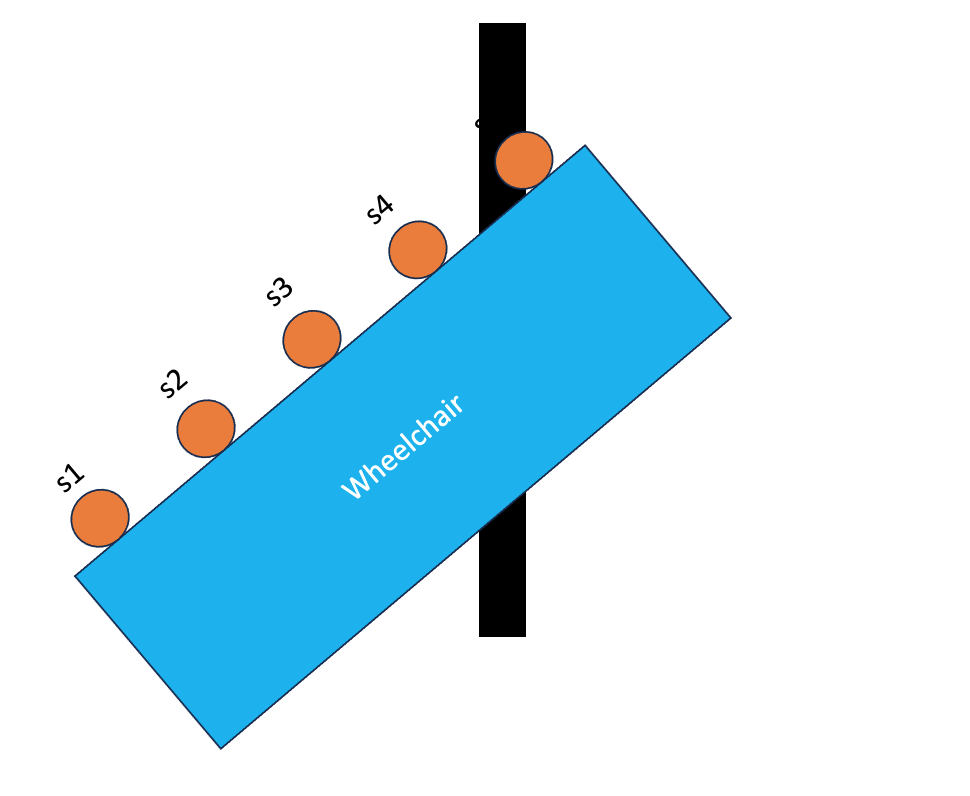
digitalWrite(m3, LOW);

digitalWrite(m4, LOW);

}

Trong đoạn code trên khi xe bị lệch nhẹ về phía bên trái thì động cơ 1 sẽ chạy còn động cơ 2 sẽ dừng để điều hướng xe quay nhẹ về phía bên phải

* **Xe lệch trái cấp 2:**

Trong trường hợp này, cảm biến chân s5 phát hiện line thì trả về mức 0, các cảm biến còn lại trả về mức 1, hình mô tả tình trạng của xe so với line .   


**Xe lệch trái cấp 2**

Chương trình xử lý tín hiệu cảm biến và điều khiển 2 bánh xe như sau:

if((s1 == 1) && (s2 == 1) && (s3 == 1) && (s4 == 1) && (s5 == 0)) {

analogWrite(e1, speed);

analogWrite(e2, speed);

digitalWrite(m1, HIGH);

digitalWrite(m2, LOW);

digitalWrite(m3, LOW);

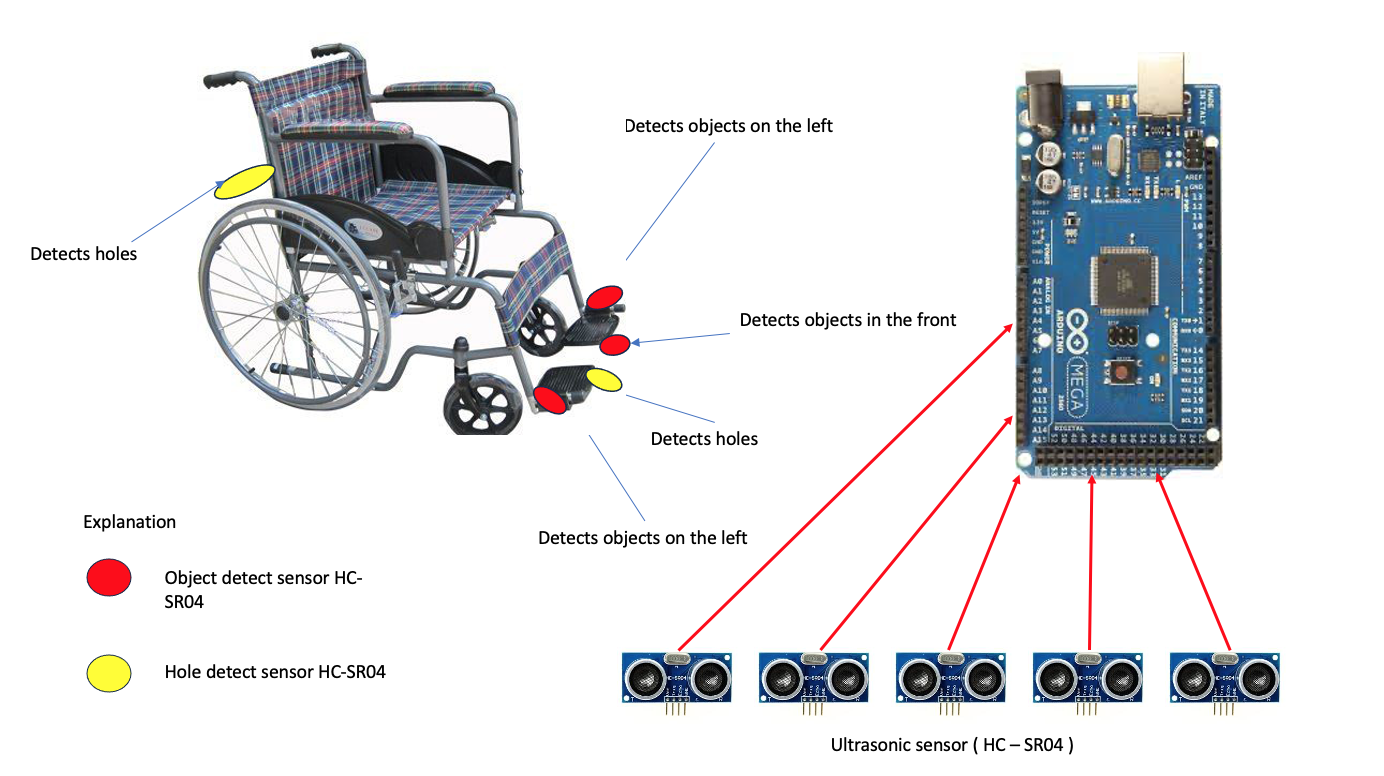
digitalWrite(m4, HIGH);

}

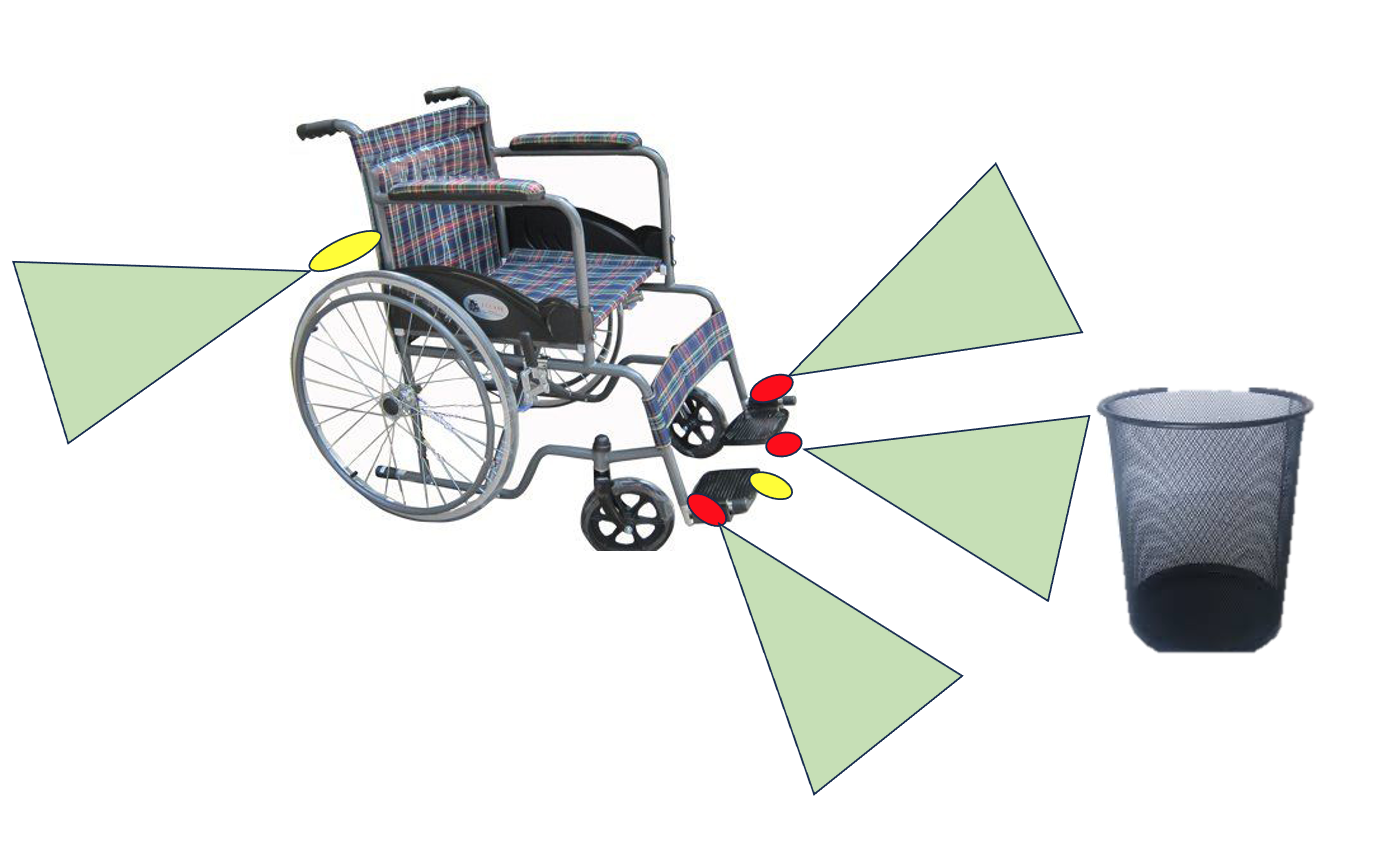
Trong đoạn code trên khi xe bị lệch hoàn toàn về phía bên trái thì động cơ 1 sẽ chạy còn động cơ 2 sẽ đi lùi để điều hướng xe quay về phía bên phải.

2.4 Hệ thống phát hiện vật cản và phát hiện hố sâu.

Trong phần này, tác giả xây dựng một hệ thống cho phép giám sát và đưa ra cảnh báo khi phát hiện vật cản phía trước hoặc phát hiện có hố sâu.

Trường hợp phát hiện có vật cản, nếu như người vận hành tiếp tục cho phép đi tới thì sẽ không có tác dụng và hệ thống sẽ cảnh báo hoặc yêu cầu điều hướng sang phía không có vật cản. Ngoài ra, nếu người vận hành xe di chuyển đến khu vực có hố thì hệ thống sẽ cảnh báo và kích hoạt cho xe dừng lại và kích hoạt cơ chế phanh xe chống bị ngã xuống hố.

2.4.1 Hệ thống phát hiện vật cản và yêu cầu điều hướng



Hình mô tả vùng hoạt động của các cảm biến

* Nguyên lý hoạt động của cảm biến phát hiện vật cản:

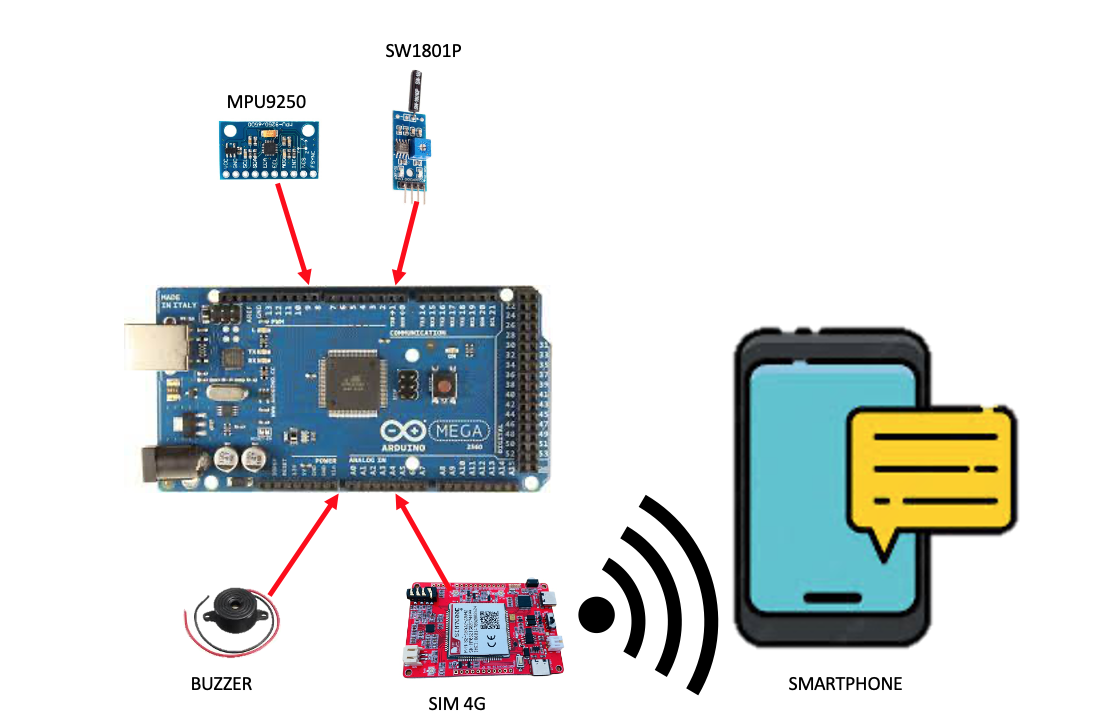
Khi xe di chuyển phát hiện vật cản ở phía trước nếu khoảng cách từ cảm biến đến vật thể nhỏ hơn 2m ( < 2m) thì xe sẽ giảm tốc nếu bên trái hoặc bên phải không có vật cản thì xe sẽ cảnh báo cho người vận hành. Trường hợp người vận hành vì lí do nào đó vẫn tiếp tục giữ nút điều khiển xe tiếp tục đi thẳng thì nếu khoảng cách nhỏ hơn 1m ( < 1m ) thì xe sẽ tự dừng.

* Nguyên lý hoạt động của cảm biến phát hiện hố sâu:

Khi xe đang di chuyển mà phát hiện có hố ở phía trước thì xe sẽ cảnh báo và tự dừng lại. Lúc này tiếp tục điều khiển di chuyển về phía trước sẽ không có tác dụng. Tương tự cho cảm biến phát hiện hố ở phía sau.

2.5 Hệ thống cảnh báo tẽ ngã

Ở phần này, tác giả đã xây dựng một hệ thống thu thập dữ liệu nhằm mục đích phát hiện xe lăn bị nghiêng hoặc người dùng bị té hoặc bị ngã, từ đó hệ thống sẽ tự động phát còi cảnh báo và gửi một tin nhắn về người chăm sóc hoặc người thân trong gia đình của người dùng. Với việc kết hợp giữa cảm biến và vi điều khiển, hệ thống có khả năng phản ứng nhanh chóng và tự động trong các tình huống nguy hiểm, tăng cường sự an toàn cho người sử dụng.

Để thực hiện chức năng này tác giả sử dụng một cảm biến rung động ( SW1801P) kết hợp với một cảm biến góc nghiêng (MPU9250). Trong trường hợp xe té ngã, cảm biến sẽ phát hiện góc nghiêng của trục y đã được giới hạn trước và cảm biến rung động phát tín hiệu. Sử dụng hai tín hiệu này để kích hoạt một chuông cảnh báo ngay trên xe và gửi một tin nhắn thông qua module sim đến số điện thoại người ứng cứu. Sơ đồ đấu dây các thiết bị được trình bày trong hình ….. 

Hình ảnh sơ đồ nối dây của hệ t

2.5.1

CHƯƠNG 3. KẾT QUẢ VÀ SẢN PHẨM ĐẠT ĐƯỢC

<Nêu rõ các kết quả thu được trong quá trình thực hiện đề tài>

**CHƯƠNG 4. KẾT LUẬN**

**TÀI LIỆU THAM KHẢO**

**BÀI BÁO (Nếu có)**

**Code cho hệ thống điều khiển xe tự hành theo line**

**#define m1 4 //Right Motor MA1**

**#define m2 5 //Right Motor MA2**

**#define m3 2 //Left Motor MB1**

**#define m4 3 //Left Motor MB2**

**#define e1 9 //Right Motor Enable Pin EA**

**#define e2 10 //Left Motor Enable Pin EB**

**//\*\*\*\*\*\*\*\*\*\*5 Channel IR Sensor Connection\*\*\*\*\*\*\*\*\*\*//**

**#define ir1 A0**

**#define ir2 A1**

**#define ir3 A2**

**#define ir4 A3**

**#define ir5 A4**

**//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//**

**void setup() {**

**pinMode(m1, OUTPUT);**

**pinMode(m2, OUTPUT);**

**pinMode(m3, OUTPUT);**

**pinMode(m4, OUTPUT);**

**pinMode(e1, OUTPUT);**

**pinMode(e2, OUTPUT);**

**pinMode(ir1, INPUT);**

**pinMode(ir2, INPUT);**

**pinMode(ir3, INPUT);**

**pinMode(ir4, INPUT);**

**pinMode(ir5, INPUT);**

**}**

**void loop() {**

**//Reading Sensor Values**

**int s1 = digitalRead(ir1); //Left Most Sensor**

**int s2 = digitalRead(ir2); //Left Sensor**

**int s3 = digitalRead(ir3); //Middle Sensor**

**int s4 = digitalRead(ir4); //Right Sensor**

**int s5 = digitalRead(ir5); //Right Most Sensor**

**//if only middle sensor detects black line**

**if((s1 == 1) && (s2 == 1) && (s3 == 0) && (s4 == 1) && (s5 == 1))**

**{**

**//going forward with full speed**

**analogWrite(e1, 255); //you can adjust the speed of the motors from 0-255**

**analogWrite(e2, 255); //you can adjust the speed of the motors from 0-255**

**digitalWrite(m1, HIGH);**

**digitalWrite(m2, LOW);**

**digitalWrite(m3, HIGH);**

**digitalWrite(m4, LOW);**

**}**

**//if only left sensor detects black line**

**if((s1 == 1) && (s2 == 0) && (s3 == 1) && (s4 == 1) && (s5 == 1))**

**{**

**//going right with full speed**

**analogWrite(e1, 255); //you can adjust the speed of the motors from 0-255**

**analogWrite(e2, 255); //you can adjust the speed of the motors from 0-255**

**digitalWrite(m1, HIGH);**

**digitalWrite(m2, LOW);**

**digitalWrite(m3, LOW);**

**digitalWrite(m4, LOW);**

**}**

**//if only left most sensor detects black line**

**if((s1 == 0) && (s2 == 1) && (s3 == 1) && (s4 == 1) && (s5 == 1))**

**{**

**//going right with full speed**

**analogWrite(e1, 255); //you can adjust the speed of the motors from 0-255**

**analogWrite(e2, 255); //you can adjust the speed of the motors from 0-255**

**digitalWrite(m1, HIGH);**

**digitalWrite(m2, LOW);**

**digitalWrite(m3, LOW);**

**digitalWrite(m4, HIGH);**

**}**

**//if only right sensor detects black line**

**if((s1 == 1) && (s2 == 1) && (s3 == 1) && (s4 == 0) && (s5 == 1))**

**{**

**//going left with full speed**

**analogWrite(e1, 255); //you can adjust the speed of the motors from 0-255**

**analogWrite(e2, 255); //you can adjust the speed of the motors from 0-255**

**digitalWrite(m1, LOW);**

**digitalWrite(m2, LOW);**

**digitalWrite(m3, HIGH);**

**digitalWrite(m4, LOW);**

**}**

**//if only right most sensor detects black line**

**if((s1 == 1) && (s2 == 1) && (s3 == 1) && (s4 == 1) && (s5 == 0))**

**{**

**//going left with full speed**

**analogWrite(e1, 255); //you can adjust the speed of the motors from 0-255**

**analogWrite(e2, 255); //you can adjust the speed of the motors from 0-255**

**digitalWrite(m1, LOW);**

**digitalWrite(m2, HIGH);**

**digitalWrite(m3, HIGH);**

**digitalWrite(m4, LOW);**

**}**

**//if middle and right sensor detects black line**

**if((s1 == 1) && (s2 == 1) && (s3 == 0) && (s4 == 0) && (s5 == 1))**

**{**

**//going left with full speed**

**analogWrite(e1, 255); //you can adjust the speed of the motors from 0-255**

**analogWrite(e2, 255); //you can adjust the speed of the motors from 0-255**

**digitalWrite(m1, LOW);**

**digitalWrite(m2, LOW);**

**digitalWrite(m3, HIGH);**

**digitalWrite(m4, LOW);**

**}**

**//if middle and left sensor detects black line**

**if((s1 == 1) && (s2 == 0) && (s3 == 0) && (s4 == 1) && (s5 == 1))**

**{**

**//going right with full speed**

**analogWrite(e1, 255); //you can adjust the speed of the motors from 0-255**

**analogWrite(e2, 255); //you can adjust the speed of the motors from 0-255**

**digitalWrite(m1, HIGH);**

**digitalWrite(m2, LOW);**

**digitalWrite(m3, LOW);**

**digitalWrite(m4, LOW);**

**}**

**//if middle, left and left most sensor detects black line**

**if((s1 == 0) && (s2 == 0) && (s3 == 0) && (s4 == 1) && (s5 == 1))**

**{**

**//going right with full speed**

**analogWrite(e1, 255); //you can adjust the speed of the motors from 0-255**

**analogWrite(e2, 255); //you can adjust the speed of the motors from 0-255**

**digitalWrite(m1, HIGH);**

**digitalWrite(m2, LOW);**

**digitalWrite(m3, LOW);**

**digitalWrite(m4, LOW);**

**}**

**//if middle, right and right most sensor detects black line**

**if((s1 == 1) && (s2 == 1) && (s3 == 0) && (s4 == 0) && (s5 == 0))**

**{**

**//going left with full speed**

**analogWrite(e1, 255); //you can adjust the speed of the motors from 0-255**

**analogWrite(e2, 255); //you can adjust the speed of the motors from 0-255**

**digitalWrite(m1, LOW);**

**digitalWrite(m2, LOW);**

**digitalWrite(m3, HIGH);**

**digitalWrite(m4, LOW);**

**}**

**//if all sensors are on a black line**

**if((s1 == 0) && (s2 == 0) && (s3 == 0) && (s4 == 0) && (s5 == 0))**

**{**

**//stop**

**digitalWrite(m1, LOW);**

**digitalWrite(m2, LOW);**

**digitalWrite(m3, LOW);**

**digitalWrite(m4, LOW);**

**}**

**}**

|  |  |
| --- | --- |
| **Chủ nhiệm đề tài**  *(Ký và ghi rõ họ tên)* | **Giảng viên hướng dẫn**  *(Ký và ghi rõ họ tên)* |