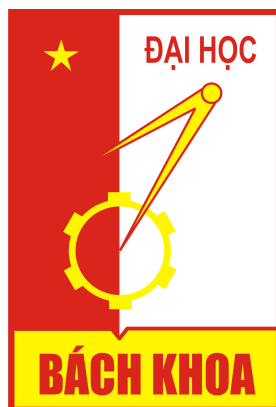


TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG



BÁO CÁO BÀI TẬP LỚN

MÔN: QUẢN TRỊ DỰ ÁN HỆ NHÚNG THEO CHUẨN KỸ NĂNG ITSS

Đề tài: Thiết kế robot dò đường

Giảng viên hướng dẫn: TS. Nguyễn Đình Thuận

Mã lớp: 122007

Nhóm sinh viên thực hiện:

- 1) Nguyễn Trung Thành - 20176874**
- 2) Bùi Như Phú - 20176844**
- 3) Đặng Bá Dũng - 20176731**

Hà Nội, tháng 6 năm 2021

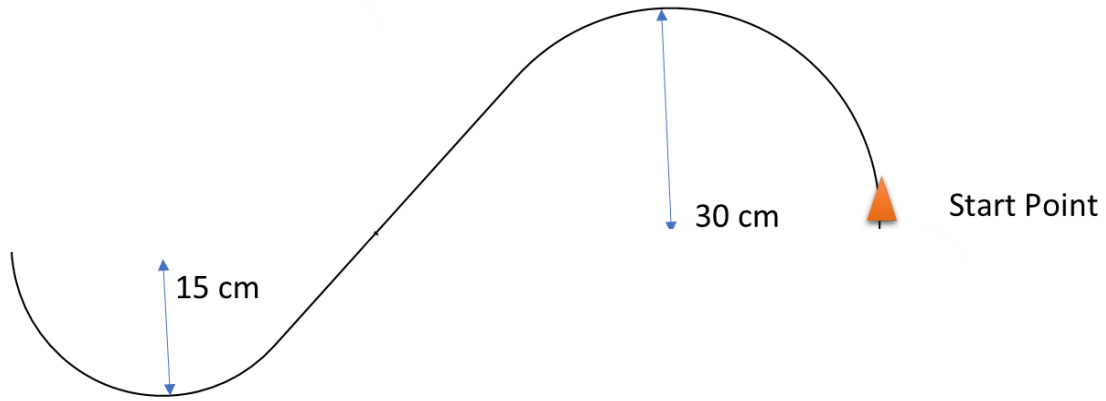
MỤC LỤC

CHƯƠNG 1: PHÂN TÍCH YÊU CẦU	3
1.1. Yêu cầu thiết kế	3
1.2. Yêu cầu phần cứng	3
1.3. Yêu cầu phần mềm	3
CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ	4
2.1. Phân tích thiết kế hệ thống	4
2.2. Module thiết kế phần cứng	4
2.2.2. Động cơ và Bánh xe	5
2.2.3. Cảm biến dò line 5 led	6
2.2.4. Arduino UNO	6
2.3. Thiết kế mạch phần cứng	7
2.4. Module thiết kế phần mềm	8
2.4.1. Thư viện và công cụ sử dụng	9
2.4.2. Mã nguồn phát triển	9
2.4.3. Phương pháp điều chế độ rộng xung (PWM)	9
2.4.4. Giải thuật điều khiển vi tích phân tỉ lệ (PID)	10
CHƯƠNG 3: QUẢN LÝ NHÂN SỰ	11
5.1. Danh sách thành viên	11
5.2. Phân công nhiệm vụ	11
CHƯƠNG 4: QUẢN LÝ THIẾT BỊ MUA SẮM	12
4.1. Danh sách các thiết bị dự kiến	12
4.2. Danh sách thiết bị đã mua	12
4.3. So sánh và đánh giá	13
CHƯƠNG 5: QUẢN LÝ RỦI RO	14
CHƯƠNG 6: KỊCH BẢN KIỂM THỬ	16
6.1. Thử nghiệm lập trình với phương pháp điều khiển tín hiệu xung PWM	16
6.2. Thử nghiệm lập trình hiệu chỉnh tham số PID	18
CHƯƠNG 7: ĐÁNH GIÁ KẾT QUẢ VÀ HƯỚNG PHÁT TRIỂN	19
7.1. Đánh giá kết quả	19
7.2. Hướng phát triển trong tương lai	19
CHƯƠNG 8: TÀI LIỆU THAM KHẢO	20

CHƯƠNG 1: PHÂN TÍCH YÊU CẦU

1.1. Yêu cầu thiết kế

Thiết kế phần cứng và phần mềm cho robot dò đường theo quỹ đạo cho sẵn. Quỹ đạo di chuyển của xe theo hình chữ S có chiều dài và kích thước như sau:



Hình 0: Quỹ đạo di chuyển của xe

1.2. Yêu cầu phần cứng

Thiết bị robot dò đường, mạch điều khiển xe chạy được.

1.3. Yêu cầu phần mềm

Thiết kế chương trình điều khiển xe chạy đúng đường đã thiết kế (theo hình vẽ), xe chạy ổn định với tốc độ tối ưu.

CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ

2.1. Phân tích thiết kế hệ thống

Robot dò đường bao gồm 2 module chính:

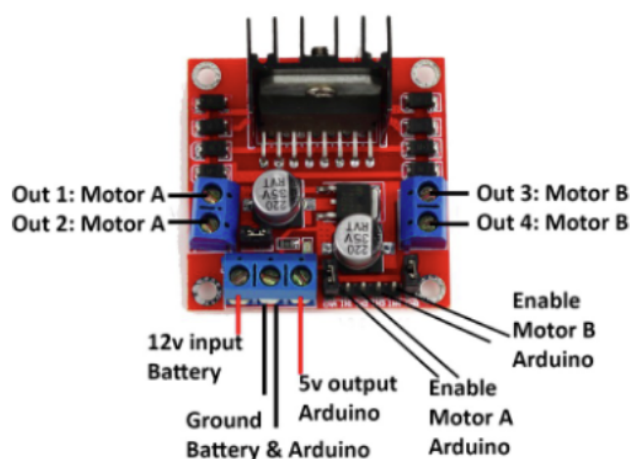
- Phần 1: Module thiết kế phần cứng.
- Phần 2: Module thiết kế phần mềm.

2.2. Module thiết kế phần cứng

Các module phần cứng sử dụng thiết kế robot bao gồm các thành phần như sau:

STT	Tên thiết bị	Chức năng
1	Arduino UNO	Bộ xử lý trung tâm của robot, chứa các mã lệnh điều khiển chương trình của robot
2	Module dò đường (5 cặp hồng ngoại)	Cảm biến phát hiện đường đi cho robot
3	Module điều khiển động cơ	Module dùng để điều chỉnh tốc độ dòng điện của motor, từ đó điều khiển tốc độ quay của bánh xe
4	Khung xe Robot	Khung xe của robot
5	Module Bluetooth	Điều khiển robot thông qua bluetooth
6	Pin sạc + Khay Pin	Nguồn điện của robot
7	Dây cắm board	Kết nối các phần cứng với nhau
8	Bộ sạc Pin	Nạp lại nguồn cho pin

2.2.1. Module điều khiển động cơ



Hình 1: Module điều khiển LD298

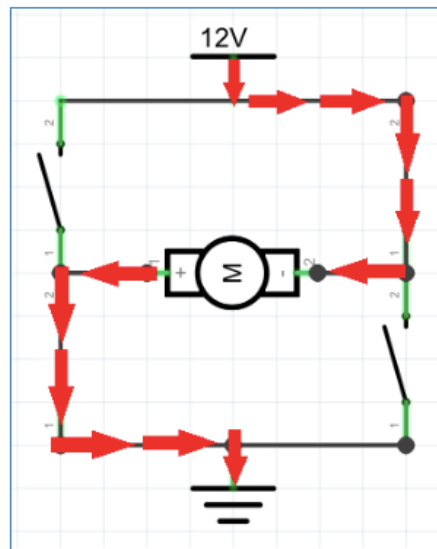
- Dùng để điều chỉnh điện áp cho arduino UNO và động cơ.
- Tích hợp sẵn 2 mạch cầu H
- Nguồn vào: 12V
- Cung cấp nguồn ra 5V cho UNO
- Dòng tối đa cho mỗi cầu H là 2A
- Điện áp của tín hiệu điều khiển: 5-7V
- Dòng tín hiệu điều khiển: 0-36mA

2.2.2. Động cơ và Bánh xe



Hình 2: Động cơ 1 chiều + bánh xe

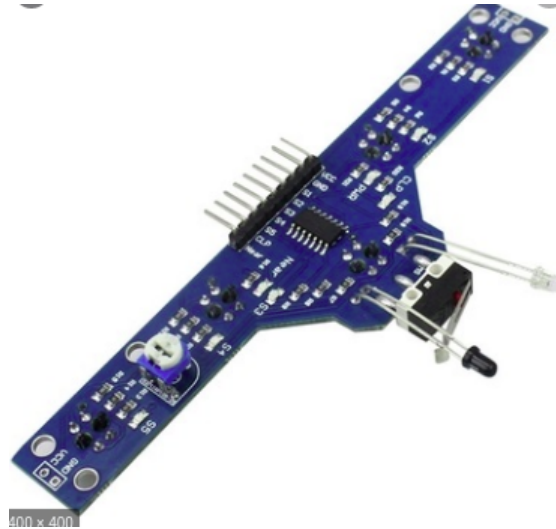
Nguyên lý hoạt động: Động cơ được điều khiển thông qua mạch cầu H của module 1298n.
Mạch cầu H:



Hình 3: Sơ đồ mạch cầu H điều khiển động cơ

Mỗi động cơ sẽ được nối với mạch cầu H của module 1298n. Khi tín hiệu in là LOW thì bánh xe không quay. Ngược lại khi tín hiệu in là HIGH thì bánh sẽ quay. Tốc độ quay được điều chỉnh bằng cách điều chỉnh điện áp đặt vào 2 đầu mạch cầu này.

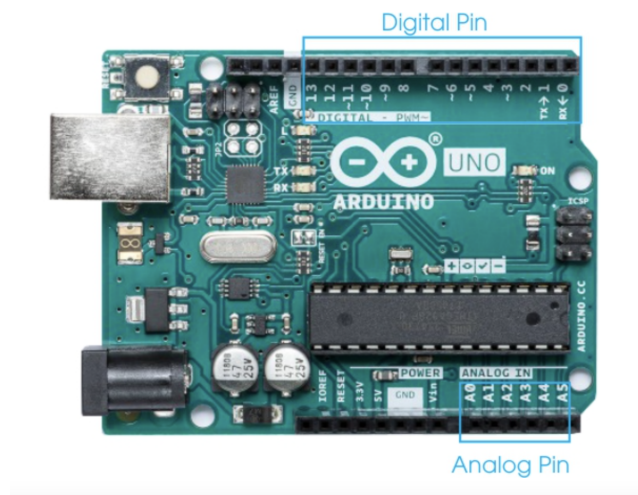
2.2.3. Cảm biến dò line 5 led



Hình 4: Module cảm biến dò line 5 led

- Phát hiện vùng màu đen và vùng màu sáng
- Hoạt động dựa trên nguyên lý của tia hồng ngoại (bị hấp thụ khi chiếu vào bề mặt màu đen và phản xạ với các bề mặt màu sáng)
- 5 cảm biến hồng ngoại hướng xuống đất, cảm biến nào nằm trên vùng màu đen sẽ cho giá trị 0, ngược lại là 1
- Nguồn hoạt động: 3.3V từ UNO

2.2.4. Arduino UNO

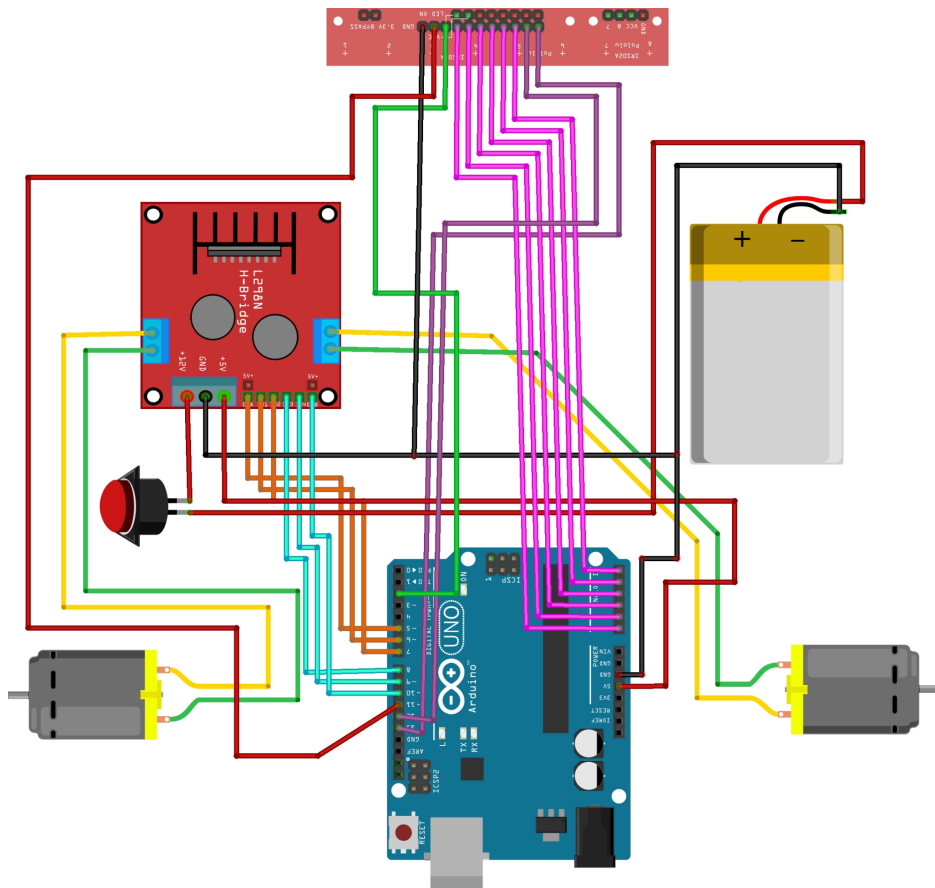


Hình 5: Module Arduino Uno

Trên Board Arduino Uno có 14 chân Digital được sử dụng để làm chân đầu vào và đầu ra và chúng sử dụng các hàm pinMode(), digitalWrite(), digitalRead(). Giá trị điện áp trên mỗi chân là 5V, dòng trên mỗi chân là 20mA và bên trong có điện trở kéo lên là 20-50 ohm. Dòng tối đa trên mỗi chân là I/O không vượt quá 40mA để tránh trường hợp gây hỏng board mạch. Ngoài ra, một số chân Digital có chức năng đặc biệt:

- **Serial:** 0(RX) và 1(TX): Được sử dụng để nhận dữ liệu (RX) và truyền dữ liệu (TX) TTL.
 - **Ngắt ngoài:** Chân 2 và 3
 - **PWM:** 3, 5, 6, 9 và 11 cung cấp đầu ra xung PWM với độ phân giải 8 bit bằng hàm analogWrite().
 - **SPI:** Có 1 LED được tích hợp trên bảng mạch và được nối vào chân D13. Khi chân có giá trị mức cao (HIGH) thì LED sẽ sáng và LED tắt khi ở mức thấp (LOW).
 - **TWI/I2C:** A4 (SDA) và A5 (SCL) hỗ trợ giao tiếp I2C/TWI với các thiết bị khác.
- Arduino Uno R3 có 6 chân Analog từ A0 đến A5, đầu vào cung cấp độ phân giải là 10 bit.

2.3. Thiết kế mạch phần cứng



Hình 6: Sơ đồ nối dây mạch robot dò line

Sơ đồ lắp ghép mạch phần cứng được thể hiện như hình bên trên, cụ thể như sau:

- Các cổng đầu ra của cảm biến dò line 5 led được gắn thứ tự lần lượt vào các cổng A0, A1, A2, A3, A4 trên mạch Arduino Uno
- Cổng 11 trên mạch Arduino nối với cổng EnA của mạch điều khiển động cơ để điều khiển tốc độ của bánh phải
- Cổng 10 trên mạch Arduino nối với cổng EnB của mạch điều khiển động cơ LD298 để điều khiển tốc độ của bánh phải

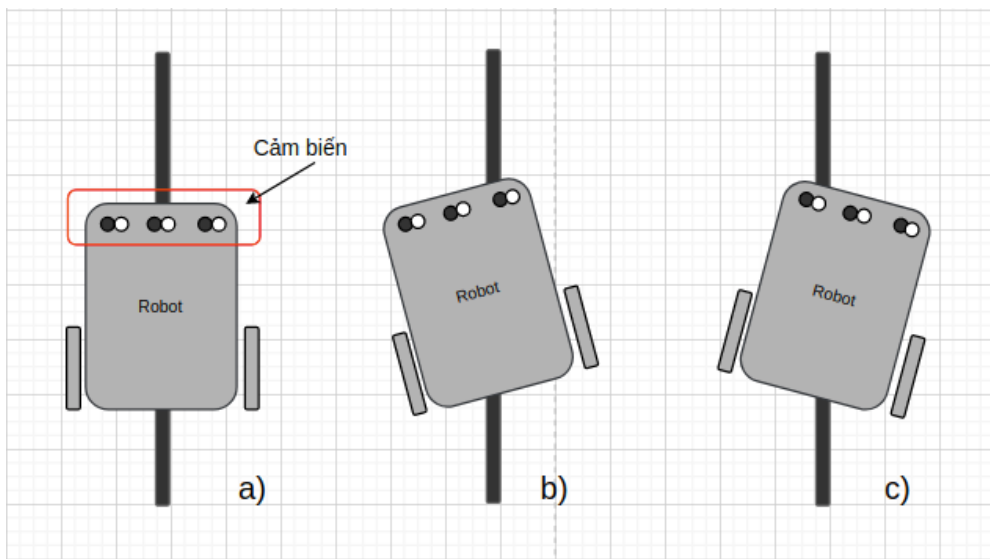
- Các cổng 8, 9, 12, 13 trên mạch Arduino nối với các cổng input của mạch điều khiển động cơ LD298
- Các cổng output của mạch điều khiển động cơ LD298 nối với 2 motor để điều khiển bánh xe trái và phải
- Nguồn điện: Cực dương nối với cổng (+5V) trên mạch Arduino và mạch điều khiển động cơ LD298, cực âm nối với ground để cấp nguồn cho mạch hoạt động.

2.4. Module thiết kế phần mềm

Thuật toán dò đường thường sử dụng trong thiết kế các robot dò đường là đặt một dãy các cảm biến sắp xếp đều nhau phía trước robot. Việc lệch đường đi trong quá trình di chuyển sẽ được robot phát hiện thông qua các trạng thái của cảm biến, cụ thể:

- Nếu hai cảm biến ở giữa cảm nhận được đường đi thì robot đang tiến thẳng về phía trước (hình 7.a).
- Nếu các cảm biến bên phải cảm nhận được đường đi thì robot di chuyển lệch về bên trái (hình 7.b).
- Nếu các cảm biến bên trái cảm nhận được đường đi thì robot đang di chuyển lệch sang phải (hình 7.c).

Thuật toán này được ứng dụng khá thành công trong việc thiết kế robot dò đường.



Hình 7. Sự lệch trong quá trình di chuyển của robot.
(a) Robot đi thẳng, (b) Robot lệch trái, (c) Robot lệch phải

Ý tưởng chính để điều khiển robot dò đường:

- Khi robot di chuyển lệch sang trái của đường đi thì robot sẽ giảm dòng điện động cơ phải (giảm tốc độ bánh phải) trong khi động cơ trái vẫn hoạt động bình thường, điều này sẽ giúp cho robot quay lại đường đi ban đầu;
- Khi robot di chuyển lệch sang phải của đường đi thì robot sẽ giảm dòng điện động cơ trái (giảm tốc độ bánh trái) trong khi động cơ phải vẫn hoạt động bình thường để giúp robot quay trở lại đường đi ban đầu.

2.4.1. Thư viện và công cụ sử dụng

Ngôn ngữ phát triển: ngôn ngữ lập trình Arduino.

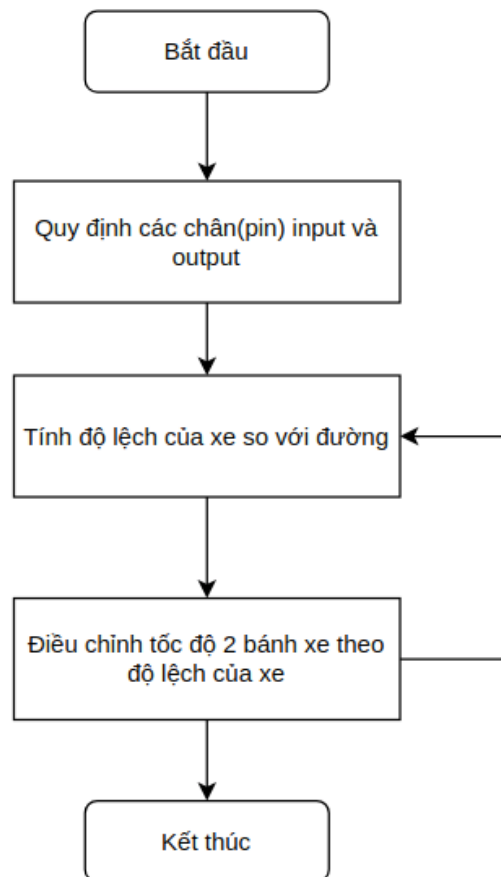
IDE phát triển: phần mềm chuyên dụng Arduino IDE.

Quản lý mã nguồn: Github.

2.4.2. Mã nguồn phát triển

Liên kết lưu trữ mã nguồn: <https://github.com/thanhhhff/arduino-embedded-system-projects>

Lưu đồ xử lý trong mã nguồn:



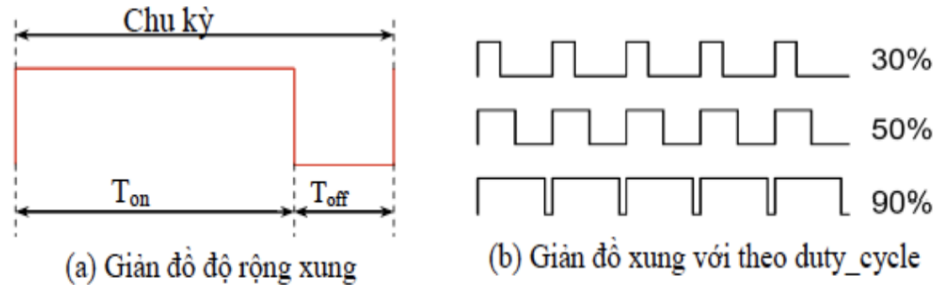
Hình 8: Lưu đồ thuật toán mã nguồn điều khiển xe dò line

2.4.3. Phương pháp điều chế độ rộng xung (PWM)

- Tốc độ quay của động cơ một chiều tỉ lệ thuận với điện áp đầu vào. Do đó, cách đơn giản nhất để điều khiển tốc độ động cơ là thay đổi mức điện áp đặt vào động cơ.
- Nguyên tắc cơ bản để thay đổi tốc độ đó là phương pháp PWM, cụ thể là giữ nguyên giá trị điện áp vào và thay đổi thời gian đặt điện áp vào động cơ.
- Khi thời gian mức cao T_{on} trong một chu kỳ của xung ngõ vào động cơ càng lớn thì điện áp trung bình đặt vào động cơ càng cao và ngược lại khi mức thời gian thấp ta có T_{off} .

- Đại lượng mô tả mối quan hệ giữa khoảng thời gian T_{on} và T_{off} được gọi là độ xung (duty_cycle), được tính theo công thức sau:

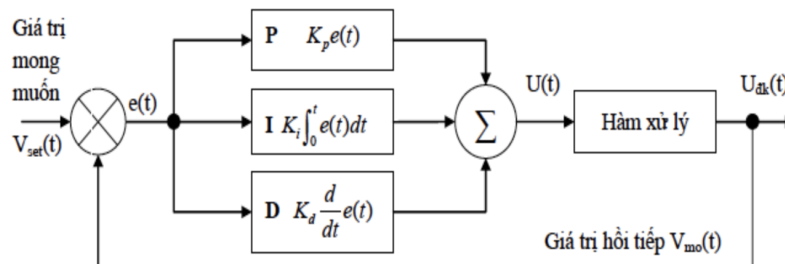
$$\text{duty_cycle} = \frac{T_{on}}{T_{on} + T_{off}} \times 100$$



Hình 9. Giản đồ thời gian của xung PWM

2.4.4. Giải thuật điều khiển vi tích phân tỉ lệ (PID)

- Giải thuật điều khiển PID phân tích tỷ lệ hiệu chỉnh sai số giữa giá trị biến đo được và giá trị mong muốn đạt đến bằng cách tính toán và xuất ra một “điều chỉnh” nhanh chóng để giữ cho lỗi ở mức nhỏ nhất có thể được.
- Bộ điều khiển này gồm 3 tham số: Proportional (P), Integral (I), Derivative (D), được thể hiện qua sơ đồ khối và phương trình tổng quát như hình sau:



Hình 10. Sơ đồ khối giải thuật PID

- Phương trình PID tổng quát:

$$U(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{d}{dt} e(t)$$

- Hàm lỗi $e(t)$ tại thời điểm t : độ lệch giữa giá trị mong muốn và giá trị đo được.

$$e(t) = V_{set} - V_{mo}(t)$$
- Proportional $P(t)$ (hàm tỷ lệ): điều khiển tỉ lệ với giá trị lỗi hiện tại.
- Integral $I(t)$ (hàm tích phân): điều khiển tương ứng với mức lỗi được tích lũy theo thời gian.
- Derivative $D(t)$ (hàm vi phân): điều khiển tương ứng với tốc độ thay đổi của lỗi, hay độ dốc lỗi theo thời gian.

CHƯƠNG 3: QUẢN LÝ NHÂN SỰ**5.1. Danh sách thành viên**

STT	Họ và tên	MSSV	Vai trò
1	Nguyễn Trung Thành	20176874	Quản lý dự án, Phát triển mã nguồn
2	Bùi Như Phú	20176844	Quản lý rủi ro, Phát triển mã nguồn
3	Đặng Bá Dũng	20176731	Quản lý mua sắm, Phát triển mã nguồn

5.2. Phân công nhiệm vụ

STT	Nhiệm vụ	Thành viên	Đánh giá	Ghi chú
1	Tìm hiểu và mua thiết bị phần cứng	Dũng	100%	Đã mua đầy đủ các thiết bị
2	Lắp ráp mô hình hoàn thiện về toàn bộ thiết bị phần cứng	Thành, Dũng, Phú	100%	Xe lắp ráp hoàn thiện
3	Code các module mô phỏng trình dò line bằng PWM	Thành	100%	Code chạy ổn định
4	Thực nghiệm kiểm thử các tham số khác nhau cho PWM	Thành, Phú	100%	Đã lựa chọn được bộ tham số ổn định
5	Code thử nghiệm với thuật toán PID	Dũng	70%	Chạy ổn định với đường cong kín và hình chữ S (chưa quay đầu được)
7	Hoàn thành tài liệu báo cáo	Thành, Dũng, Phú	100%	

CHƯƠNG 4: QUẢN LÝ THIẾT BỊ MUA SẴM

4.1. Danh sách các thiết bị dự kiến

STT	Tên thiết bị	Số lượng	Đơn giá	Link tham khảo
1	Arduino UNO	1	109.000đ	Link shopee
2	IR Sensor	1	68.000đ	Link shopee
3	Motor Driver (L298N)	1	26.000đ	Link shopee
4	Khung xe	1	140.000đ	Tham khảo tại cửa hàng 3M
5	Bánh xe sau	2		
6	Bánh dẫn	1		
7	Motors	1		
8	Dây điện	20 dây		
9	Case đựng pin	1	8.000đ	Link shopee
10	Pin	2	24.000đ	Link shopee
11	Module Bluetooth	1	90.000đ	Link shopee
Tổng tiền dự kiến			489.000đ	

4.2. Danh sách thiết bị đã mua

STT	Tên thiết bị	Số lượng	Đơn giá	Thành tiền
1	Arduino UNO	1	125,000đ	125,000đ
2	Module dò đường (5 cặp hồng ngoại)	1	95,000đ	95,000đ
3	Module điều khiển động cơ	1	37,000đ	37,000đ
4	Khung xe Robot	1	119,000đ	119,000đ
5	Module Bluetooth	1	115,000đ	115,000đ
6	Pin sạc + Khay Pin	1	45,000đ	45,000đ
7	Dây cắm board	2	13,000đ	26,000đ
8	Bộ sạc Pin	1	40,000đ	40,000đ
Tổng				602,000đ

4.3. So sánh và đánh giá

Hình thức	Ưu điểm	Nhược điểm
Mua sắm online	+ Giá thành rẻ hơn so với mua trực tiếp tại cửa hàng.	+ Thời gian nhận hàng có thể bị chậm, ảnh hưởng đến kế hoạch phát triển. + Chất lượng sản phẩm có thể không đảm bảo. + Nếu sản phẩm bị hỏng, có vấn đề thì rất khó để bảo hành.
Mua trực tiếp tại cửa hàng	+ Đảm bảo chất lượng các linh kiện. + Dễ dàng bảo hành, đổi trả nếu có vấn đề xảy ra. + Mua hàng trực tiếp có thể đảm bảo có ngay các linh kiện để lắp ráp, không bị ảnh hưởng đến kế hoạch đặt ra.	+ Chi phí cao hơn so với mua hàng online.

Kết luận: Qua các đánh giá trên nhóm quyết định mua hàng trực tiếp tại cửa hàng để đảm bảo chất lượng các linh kiện, dễ giải quyết khi phát sinh vấn đề liên quan đến linh kiện. Khi mua sắm trực tiếp được tư vấn để bổ sung các linh kiện còn thiếu trong dự kiến ban đầu, chênh lệch chi phí dự kiến và thực tế nằm trong khoảng cho phép.

CHƯƠNG 5: QUẢN LÝ RỦI RO

Danh sách quản lý các rủi ro đã xảy ra và các hướng đề xuất:

STT	Tên rủi ro	Nguyên do
1	Arduino cháy diot kiểm soát nguồn đầu vào	Đoản mạch vì cắm sai dây

Các phương án đề xuất cho rủi ro 1:

Phương án	Phương hướng giải quyết	Nhận xét
1	Thay vi mạch Arduino mới	Tốn chi phí của dự án và thời gian chờ mua vi mạch mới
2	Thay diot kiểm soát nguồn đầu vào	Chi phí thấp, tốn thời gian chờ thay diot mới
3	Cắm trực tiếp nguồn vào vi mạch Arduino	Nguồn 7.4V sẽ lớn hơn nguồn 5V của vi mạch sẽ gây hỏng mạch trong lâu dài
4	Sử dụng nguồn 5V từ Module điều khiển động cơ chuyển sang vi mạch Arduino	Đảm bảo về thời gian, chi phí và mức độ ổn định

=> Lựa chọn giải quyết theo **phương án 4**.

STT	Tên rủi ro	Nguyên do
2	Thiết bị cảm biến sensor có độ phủ quá rộng	Do nhà sản xuất

Các phương án đề xuất cho rủi ro 2:

Phương án	Phương hướng giải quyết	Nhận xét
1	Thay thế thiết bị cảm biến sensor khác có độ phủ thấp hơn	Tốn chi phí của dự án và thời gian mua cảm biến mới
2	Hàn lại cảm biến hướng dọc quay sang ngang	Rủi ro cao có thể gây hỏng cảm biến khi hàn
3	Bao phủ lại vùng cảm biến của Sensor bằng băng dính đen	Đảm bảo tín hiệu thu nhận ổn định

=> Lựa chọn giải quyết theo **phương án 3**.

Kết luận: Thông qua những rủi ro trên nhóm đã ghi nhận được các trường hợp có thể xảy ra rủi ro đối với dự án của nhóm. Sau đó đã đưa ra được các giải pháp khắc phục kịp thời và hiệu quả để hạn chế chi phí, thời gian để đảm bảo tiến độ và chất lượng dự án.

CHƯƠNG 6: KỊCH BẢN KIỂM THỬ

Các tín hiệu của Sensor được mô tả như sau:

STT	Mã	Độ lỗi	Ghi chú
1	0b11110	-4	Xe lệch trái độ 4
2	0b11100	-3	Xe lệch trái độ 3
3	0b11101	-2	Xe lệch trái độ 2
4	0b11001	-1	Xe lệch trái độ 1
5	0b11011	0	Xe ở giữa trung tâm
6	0b10011	1	Xe lệch phải độ 1
7	0b10111	2	Xe lệch phải độ 2
8	0b00111	3	Xe lệch phải độ 3
9	0b01111	4	Xe lệch phải độ 4
10	0b11111	-5	Không thu được tín hiệu từ sensor, xe đang lệch trái
		5	Không thu được tín hiệu từ sensor, xe đang lệch phải

6.1. Thử nghiệm lập trình với phương pháp điều khiển tín hiệu xung PWM

Thí nghiệm 1: Thử với xung PWM cao

Độ lỗi	Xung PWM cho bánh trái	Xung PWM cho bánh phải
-5	200	50
-4	220	100
-3	230	150
-2	230	180
-1	240	200
0	255	255

1	200	240
2	180	230
3	150	230
4	100	220
5	50	200

Nhận xét: Tốc độ của xe quá lớn dẫn đến việc tăng tốc quá nhanh trượt khỏi đường đi chuyển. Đặc biệt là khi quay đầu xe, xe có xu hướng đi phi khỏi đường đi.
=> Cần phải giảm tốc độ xuống mức trung bình.

Thí nghiệm 2: Thử với xung PWM trung bình

Độ lỗi	Xung PWM cho bánh trái	Xung PWM cho bánh phải
-5	100	10
-4	120	50
-3	120	60
-2	120	70
-1	130	80
0	150	150
1	80	130
2	70	120
3	60	120
4	50	120
5	10	100

Nhận xét: Xe chạy bám đường và quay đầu ổn định. Tuy nhiên, xe vẫn lắc nhẹ khi di chuyển, cần chỉnh lại xung cho tối ưu hơn.

Thí nghiệm 3: Tinh chỉnh lại xung PWM từ thí nghiệm 2

Độ lỗi	Xung PWM cho bánh trái	Xung PWM cho bánh phải
-5	100	10

-4	120	20
-3	100	20
-2	100	60
-1	120	100
0	140	140
1	100	120
2	60	100
3	20	100
4	20	120
5	10	100

Nhận xét: Xe chạy ổn định hơn thí nghiệm 02 (đỡ rung lắc và quay đầu ổn định).

6.2. Thử nghiệm lập trình hiệu chỉnh tham số PID

Dựa trên các thí nghiệm tại phần 6.1, nhóm sử dụng thuật toán PID để mô phỏng gần đúng và tối ưu tốc độ, độ rung lắc của xe với các giá trị của P, I, D được liệt kê dưới bảng đây:

Tham số	Giá trị
P	6.5
I	0
D	2.8

Nhận xét: Xe chạy ổn định và không rung lắc trên đường chữ S. Tuy nhiên, chưa tối ưu được để xe có thể quay đầu trở lại đường đi.

***Phân tích lý do:** khoảng cách giữa các cảm biến của Sensor từ 2.5cm đến 3cm dẫn đến ngay cả trên đường cho trước vẫn nhiều trường hợp không bắt được tín hiệu của Sensor nào. Điều này dẫn đến khó khăn trong việc tìm được tham số vừa tối ưu được trên đường đi của Sensor vừa có thể quay đầu được trở lại đường đi.

Kết luận: Thông qua 2 phương pháp hiệu chỉnh các tham số khác nhau, nhóm nhận thấy các tham số trong **Phần 6.1 - Thí nghiệm 3** thỏa mãn yêu cầu bài toán đặt ra.

CHƯƠNG 7: ĐÁNH GIÁ KẾT QUẢ VÀ HƯỚNG PHÁT TRIỂN

7.1. Đánh giá kết quả

- Sau một thời gian nghiên cứu dưới sự hướng dẫn của thầy GV. Nguyễn Đình Thuận và tìm hiểu từ các nguồn thông tin hữu ích trên mạng. Nhóm đã tận dụng tốt những thông tin và tri thức liên quan đến xe dò line để áp dụng và phát triển cho sản phẩm của nhóm.
- Trong quá trình từ khi lắp ráp thiết bị đến khi thực hiện code phần mềm điều khiển thiết bị, nhóm cũng gặp một số khó khăn liên quan đến phần cứng (cháy diot điều chỉnh nguồn đầu vào; mắt dò line quá nhạy), và một số khó khăn vì tình hình dịch bệnh nên cả nhóm không làm việc chung được để phát triển sản phẩm.
- Trước những thuận lợi và khó khăn trên, nhóm cũng đã cố gắng hoàn thiện sản phẩm ở mức độ khá tốt. Xe có thể chạy dò line theo yêu cầu được đề ra.
- Ngoài ra, nhóm học được kỹ năng làm việc nhóm, xây dựng và quản lý một dự án nhúng theo chuẩn kỹ năng ITSS.

7.2. Hướng phát triển trong tương lai

- Như đã được định hướng từ trước, sản phẩm của nhóm có thể được phát triển thêm các module tiếp theo như là ứng dụng các công nghệ thông minh AI để giúp xe thực hiện được một số chức năng thông minh khác. Những chức năng ấy có thể là nhận biết các loại tín hiệu biển báo giao thông để có thể đưa ra các hành động phù hợp với từng loại tín hiệu giao thông đó.

CHƯƠNG 8: TÀI LIỆU THAM KHẢO

[1] Hướng dẫn lắp robot dò đường

Liên kết: https://www.youtube.com/watch?v=bH2dDkPj1_s

[2] Tài liệu môn học Quản trị dự án hệ nhúng theo chuẩn kỹ năng ITSS - IT4947

[3] Tài liệu tự học Arduino

Liên kết: <http://arduino.vn/reference/howto>