

ĐẠI HỌC QUỐC GIA TP HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA ĐIỆN – ĐIỆN TỬ



BÁO CÁO ĐỒ ÁN 1
ĐỀ TÀI: THIẾT KẾ ROBOT TỰ HÀNH BÁM TƯỜNG
GVHD: HUỲNH THÁI HOÀNG
NHÓM: 4

Họ và Tên	MSSV
Trịnh Quang Huy	2110211
Nguyễn Đăng Khoa	2111529
Nguyễn Thanh Khôi	2111559

TP. Hồ Chí Minh, 25 tháng 5 năm 2024

MỤC LỤC

1. Giới thiệu đề tài	4
2. Lý thuyết điều khiển robot tự hành bám tường	4
2.1. Bộ điều khiển PID	4
2.2. Mô hình điều khiển.....	9
3. Thiết kế mô hình thực tế.....	13
3.1. Thiết kế phần cứng	13
3.1.1. Khối nguồn	13
3.1.2. Khối điều khiển	14
3.1.3. Khối động cơ	15
3.1.4. Khối cảm biến.....	17
3.1.5. Bo mạch phần cứng	18
3.1.6. Sản phẩm hoàn thiện	20
3.2. Thiết kế phần mềm	21
3.2.1. Lưu đồ giải thuật hành vi.....	21
3.2.2. Lưu đồ giải thuật điều khiển PID 2 vòng bám tường.....	22
3.2.3. Xử lý dữ liệu từ cảm biến	22
4. Kết quả và phương hướng phát triển đề tài	25
4.1. Kết quả đạt được.....	25
4.2. Hạn chế đề tài	27
4.3. Hướng phát triển đề tài	27
5. Kết luận.....	27
6. Tài liệu tham khảo	28

MỤC LỤC HÌNH ẢNH

Hình 1: Sơ đồ bộ điều khiển PID	5
Hình 2: Đáp ứng của hệ thống khi tăng dần K_P	6
Hình 3: Đáp ứng của hệ thống khi tăng dần K_I	7
Hình 4: Đáp ứng của hệ thống khi tăng dần K_D	8
Hình 5: Khoảng cách d và góc lệch θ khi $\theta > 0$	10
Hình 6: Khoảng cách d và góc lệch θ khi $\theta < 0$	11
Hình 7: Pin sạc 18650.....	13
Hình 8: IC ổn áp 7805	13
Hình 9: Sơ đồ Khối nguồn.....	14
Hình 10: Kit Phát Triển STM32F103C8T6.....	14
Hình 11: Động cơ DC giảm tốc GA12 N20	15
Hình 12: IC lái động cơ L298.....	16
Hình 13: Sơ đồ khối động cơ.....	16
Hình 14: Module cảm biến siêu âm SRF05	17
Hình 15: Nguyên lý hoạt động của cảm biến siêu âm.....	17
Hình 16: Encoder được gắn vào động cơ	18
Hình 18: Mạch PCB hoàn thiện.....	19
Hình 19: Sản phẩm Robot hoàn thiện.....	20
Hình 20: Lưu đồ giải thuật hành vi	21
Hình 21: Lưu đồ giải thuật điều khiển.....	22
Hình 22: Cấu hình xử lý tín hiệu cảm biến siêu âm	23
Hình 23: Cấu hình xử lý tín hiệu encoder	23
Hình 24: Cấu hình xuất xung PWM	24
Hình 25: đáp ứng của tốc độ động cơ.....	25
Hình 26: Robot giữ khoảng cách với tường	25
Hình 27: Robot rẽ trái khi gặp vật cản	26
Hình 28: Robot rẽ phải khi bên phải trống.....	26

1. Giới thiệu đề tài

Trong thời đại công nghiệp hóa, hiện đại hóa ngày nay, Robot ngày càng được sử dụng phổ biến trong sản xuất cũng như trong cuộc sống của con người. Chúng đóng vai trò quan trọng trong nhiều lĩnh vực, từ sản xuất đến y tế, giáo dục và thậm chí là các hoạt động hàng ngày. Robot giúp tăng năng suất lao động và thay thế con người để làm việc trong các điều kiện nguy hiểm khó khăn.

Từ tình hình thực tế đó, việc xây dựng các chương trình hoạt động cho các Robot là điều thiết yếu đặc biệt đối với các Robot di động. Bài toán Robot di động bám tường (wall-following problem) là một trong các bài toán thường gặp trong việc điều khiển Robot. Vì vậy nhóm chúng em quyết định chọn đề tài cho đồ án 1 là “**Thiết kế Robot tự hành bám tường**”.

Mục tiêu của đề tài là tập trung nghiên cứu, thiết kế và điều khiển thử nghiệm một Robot có khả năng tự động bám tường theo một khoảng cách cho trước và tránh các vật cản ở phía trước

2. Lý thuyết điều khiển robot tự hành bám tường

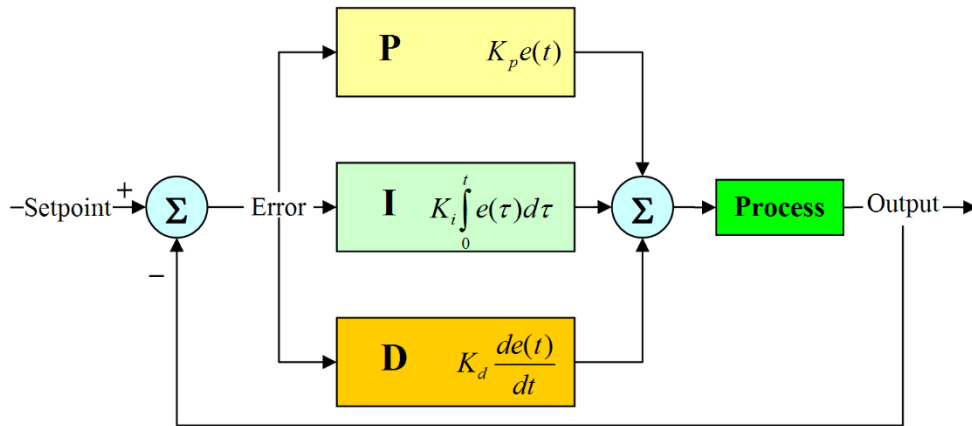
2.1. Bộ điều khiển PID

Một bộ điều khiển vi tích phân tỉ lệ (bộ điều khiển PID - Proportional Integral Derivative) là một cơ chế phản hồi vòng điều khiển (bộ điều khiển) tổng quát được sử dụng rộng rãi trong các hệ thống điều khiển – bộ điều khiển PID là bộ điều khiển được sử dụng nhiều nhất trong các bộ điều khiển phản hồi.

Bộ điều khiển PID sẽ tính toán giá trị "sai số" là hiệu số giữa giá trị đo thông số biến đổi và giá trị đặt mong muốn. Bộ điều khiển sẽ thực hiện giảm tối đa sai số bằng cách điều chỉnh giá trị điều khiển đầu vào. Trong trường hợp không có kiến thức cơ bản (mô hình toán học) về hệ thống điều khiển thì bộ điều khiển PID là sẽ bộ điều khiển tốt nhất.

Tuy nhiên, để đạt được kết quả tốt nhất, các thông số PID sử dụng trong tính toán phải điều chỉnh theo tính chất của hệ thống - trong khi kiểu điều khiển là giống nhau, các thông số phải phụ thuộc vào đặc thù của hệ thống.

Bộ điều khiển PID gồm 3 thông số đặc trưng: các giá trị tỉ lệ, tích phân và đạo hàm, viết tắt là P, I, và D. Giá trị tỉ lệ xác định tác động của sai số hiện tại, giá trị tích phân xác định tác động của tổng các sai số quá khứ, và giá trị vi phân xác định tác động của tốc độ biến đổi sai số.



Hình 1: Sơ đồ bộ điều khiển PID

Gọi e là sai số của quá trình, được xác định:

$$e = SP - PV$$

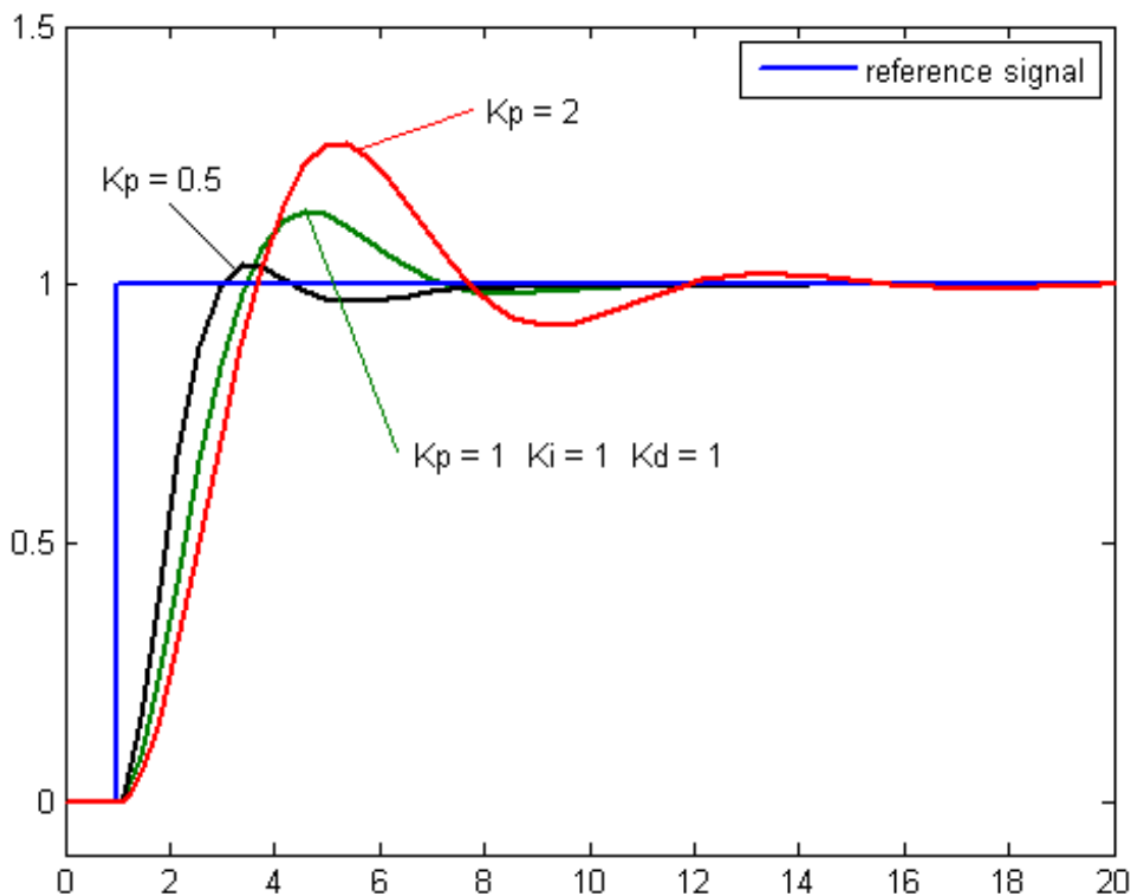
Trong đó: SP là giá trị đặt, PV trị tín hiệu đầu ra

Nhiệm vụ của bộ điều khiển phản hồi là điều chỉnh giá trị đầu vào theo giá trị e đo được để giá trị đầu ra tiến tới giá trị đặt. Tuy nhiên nếu chỉ cộng hoặc trừ đi e vào giá trị điều khiển thì thời gian đáp ứng của hệ thống có thể sẽ nhanh hơn hoặc chậm hơn thời gian đáp ứng mong muốn.

Do đó cần đến khâu tỉ lệ P. Khâu tỉ lệ làm thay đổi giá trị đầu ra, tỉ lệ với giá trị sai số hiện tại. Đáp ứng tỉ lệ có thể được điều chỉnh bằng cách nhân sai số đó với một hằng số K_p , được gọi là hệ số tỉ lệ.

Khâu tỉ lệ được cho bởi: $P = K_p \times e$

Nếu hệ số tỉ lệ quá cao, hệ thống sẽ không ổn định. Nếu hệ số tỉ lệ quá thấp, tác động điều khiển có thể sẽ quá bé khi đáp ứng với các nhiễu của hệ thống



Hình 2: Đáp ứng của hệ thống khi tăng dần K_P

Khi sai số tồn tại trong một khoảng thời gian dài về một phía của giá trị đặt, ta cần tăng tốc chuyển động của quá trình tới điểm đặt. Nhờ có khâu tích phân ta có thể giải quyết được vấn đề này.

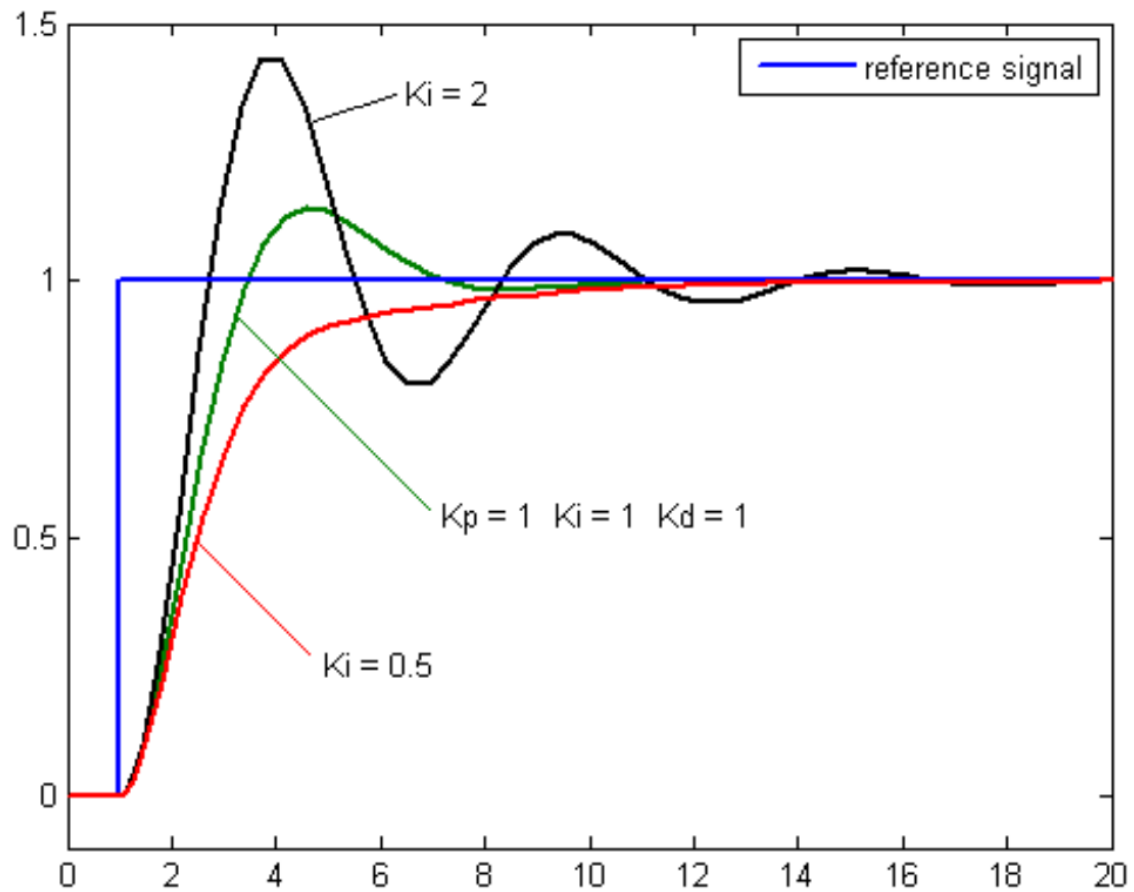
Khâu tích phân tỉ lệ thuận với cả biên độ sai số lẫn quãng thời gian xảy ra sai số. Tổng sai số tức thời theo thời gian cho ta tích lũy bù đã được hiệu chỉnh trước đó. Tích lũy sai số sau đó được nhân với độ lợi tích phân và cộng với tín hiệu đầu ra của bộ điều khiển. Biên độ phân phối của khâu tích phân trên tất cả tác động điều chỉnh được xác định bởi hệ số tích phân K_I .

$$\text{Thừa số tích phân được cho bởi: } I = K_I \int_0^t e(\tau) d\tau$$

Trong đó: K_I là hệ số tích phân

$e(\tau)$ là hàm sai lệch theo thời gian

Tuy nhiên, vì khâu tích phân là đáp ứng của sai số tích lũy trong quá khứ, nó có thể khiến giá trị hiện tại vượt quá giá trị đặt (ngang qua điểm đặt và tạo ra một độ lệch với các hướng khác), do đó cần phải chú ý khi sử dụng khâu tích phân.



Hình 3: Đáp ứng của hệ thống khi tăng dần K_I

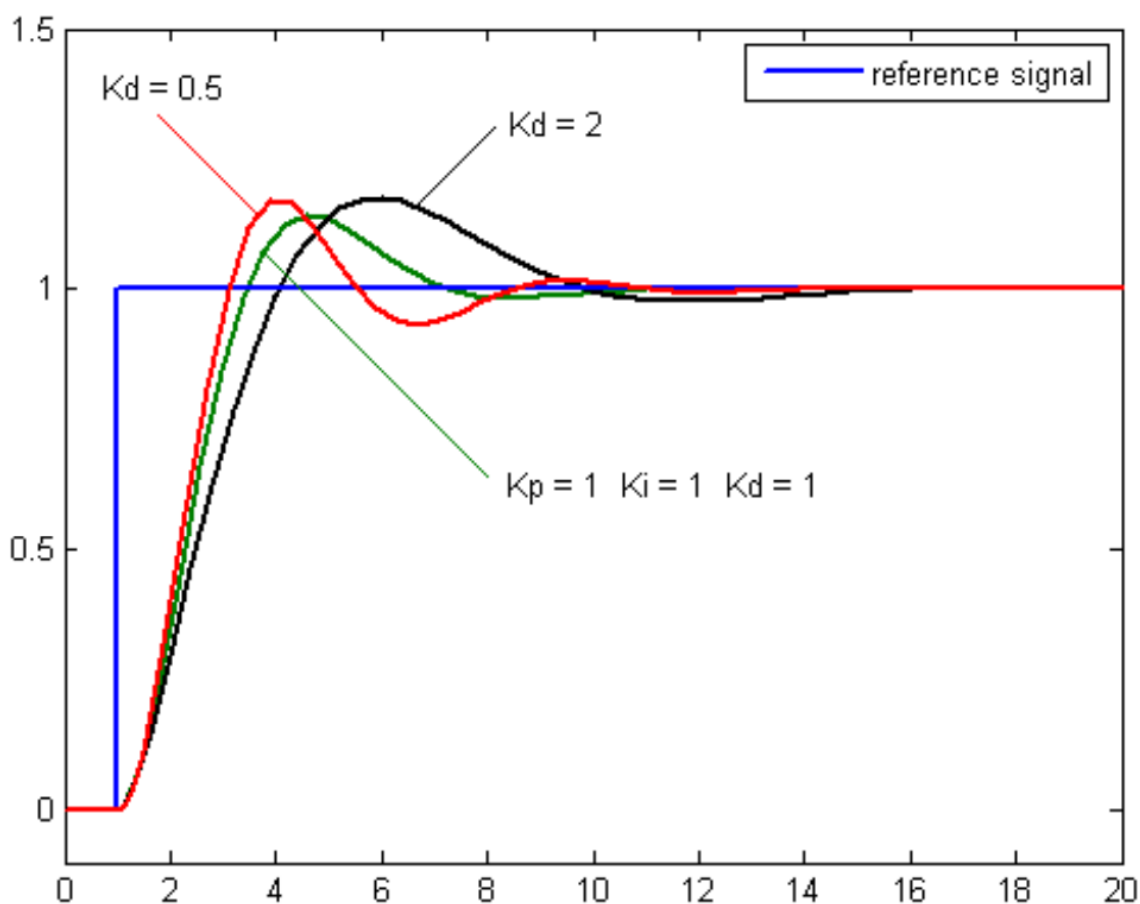
Tốc độ thay đổi của sai số qua trình được tính toán bằng cách xác định độ dốc của sai số theo thời gian (tức là đạo hàm bậc một theo thời gian) và nhân tốc độ này với hệ số vi phân K_D .

Thừa số vi phân được cho bởi: $D = K_D \frac{d}{dt} e(t)$

Trong đó: K_D là hệ số vi phân

$e(\tau)$ là hàm sai lệch theo thời gian

Khâu vi phân làm chậm tốc độ thay đổi của đầu ra bộ điều khiển và đặc tính này là đang chú ý nhất để đạt tới điểm đặt của bộ điều khiển. Từ đó, điều khiển vi phân được sử dụng để làm giảm biên độ vọt lố được tạo ra bởi thành phần tích phân và tăng cường độ ổn định của bộ điều khiển hỗn hợp.



Hình 4: Đáp ứng của hệ thống khi tăng dần K_D

Khâu tỉ lệ, tích phân, vi phân được cộng lại với nhau để tính toán đầu ra của bộ điều khiển PID. Gọi u là tín hiệu điều khiển, ta có:

$$u = K_p \times e + K_I \int_0^t e(\tau) d\tau + K_D \frac{d}{dt} e(t)$$

Nhờ những ưu điểm đã nêu trên, ta có thể ứng dụng bộ điều khiển PID để giải quyết bài toán xe tự hành bám tường.

Công thức tính toán bộ điều khiển PID rời rạc:

Ta sử dụng công thức lấy xấp xỉ với chu kì lấy mẫu là T :

$$\int_0^t e(t) dt = \sum_{k=0}^k \frac{e(kT) + e(kT - T)}{2}$$

$$\frac{de(t)}{dt} = \frac{e(kT) - e(kT - T)}{T}$$

Từ đó, ta được:

$$u(kT) = u(kT - T) + K_p \times [e(kT) - e(kT - T)] + K_I \times \frac{e(kT) - e(kT - T)}{2T} + K_D \times \frac{e(kT) - 2e(kT - T) + e(kT - 2T)}{T}$$

2.2. Mô hình điều khiển

Sử dụng hai cảm biến siêu âm để thu thập hai khoảng cách s_1 và s_2 .

Tín hiệu đặt bao gồm 2 tín hiệu:

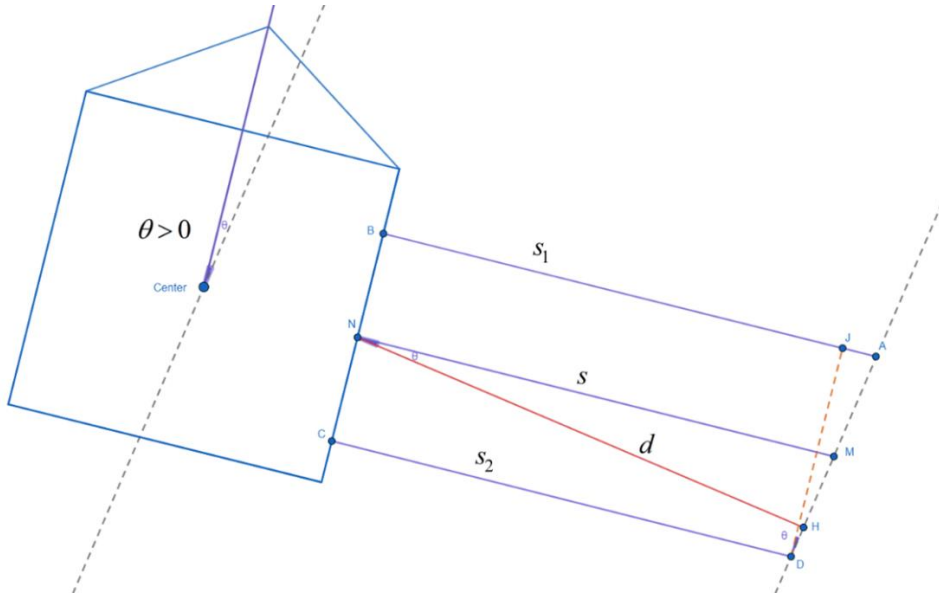
- + Tốc độ mong muốn của robot khi chạy theo đường thẳng: v_{sp}
- + Khoảng cách mong muốn giữa robot và tường: d_{sp}

Dữ liệu thu được s_1 và s_2 sẽ được xử lý để tính toán:

- + Khoảng cách hiện tại robot và tường: d
- + Sai số về khoảng cách hiện tại với khoảng cách mong muốn: e_d
- + Góc lệch hiện tại của robot so với tường cần bám: θ
- + Tốc độ góc tức thời mà robot cần phải xoay: ω
- + Tốc độ xoay hai bánh xe trái và phải để đáp ứng ω : ω_l, ω_r

Tính toán khoảng cách hiện tại d và góc lệch θ từ s_1, s_2

Trường hợp 1: $\theta > 0$



Hình 5: Khoảng cách d và góc lệch θ khi $\theta > 0$

Sử dụng cách chứng minh hình học, ta có:

$$s = \frac{s_1 + s_2}{2}$$

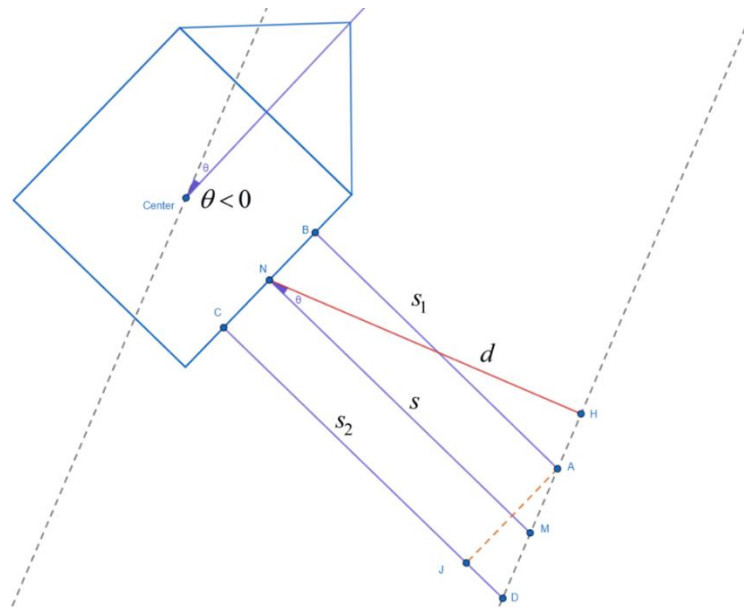
$$BC = ds$$

$$\theta = \tan^{-1}\left(\frac{s_1 - s_2}{ds}\right), \text{ do } s_1 > s_2 \text{ nên } \theta > 0$$

$$d = s \times \cos \theta$$

$$e_d = d - d_{sp}$$

Trường hợp 2: $\theta < 0$



Hình 6: Khoảng cách d và góc lệch θ khi $\theta < 0$

Sử dụng cách chứng minh hình học, ta có:

$$s = \frac{s_1 + s_2}{2}$$

$$BC = ds$$

$$\theta = \tan^{-1}\left(\frac{s_1 - s_2}{ds}\right), \text{ do } s_1 < s_2 \text{ nên } \theta < 0$$

$$d = s \times \cos \theta$$

$$e_d = d - d_{sp}$$

Tính toán tốc độ góc bánh xe ω_r, ω_l để điều khiển robot

Ta có công thức tính toán tốc độ xoay bánh xe dựa vào tốc độ góc và tốc độ dài mong muốn của robot như sau:

$$\begin{bmatrix} \omega_r \\ \omega_l \end{bmatrix} = \begin{bmatrix} \frac{1}{r} & \frac{b}{r} \\ \frac{1}{r} & \frac{-b}{r} \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}$$

Với: ω_r : tốc độ góc bánh xe bên phải của robot

ω_l : tốc độ góc bánh xe bên trái của robot

ω : tốc độ góc mong muốn của bánh xe robot

v : tốc độ mong muốn của robot

r : bán kính bánh xe

b : khoảng cách từ tâm robot đến trục bánh xe

Do $v = v_{sp}$ là giá trị người dùng mong muốn đặt và các thông số r và b cố định phụ thuộc vào phần cứng. Như vậy chỉ có giá trị ω là thay đổi để ω_r và ω_l thay đổi theo.

Giá trị ω sẽ được tính dựa vào giá trị của bộ PID đo khoảng cách, với các cận giới hạn là từ $-\pi/2$ rad/s đến $\pi/2$ rad/s.

3. Thiết kế mô hình thực tế

3.1. Thiết kế phần cứng

3.1.1. Khối nguồn

Pin sạc 18650 là lựa chọn phổ biến cho nhiều ứng dụng di động và nhúng nhờ vào kích thước tiêu chuẩn, dung lượng lớn và hiệu suất ổn định dễ dàng tìm thấy trên thị trường, và có thể sạc bằng các bộ sạc tiêu chuẩn. Dưới đây là một số đặc điểm chính của pin sạc 18650:

- Điện áp : 3.7V, khi sạc đầy có thể đạt đến 4.2v
- Dung lượng : 2200 – 2600 mAh



Hình 7: Pin sạc 18650

IC ổn áp 7805 giúp tạo nguồn áp cố định cho vi điều khiển và các cảm biến.

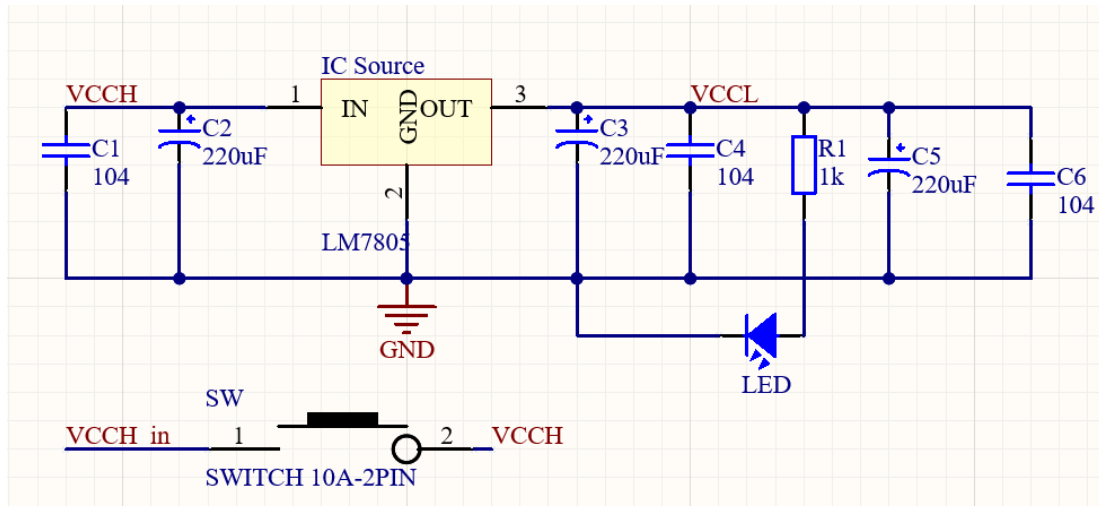
- Điện áp đầu vào: 7V – 35V
- Điện áp đầu ra: cố định ở 5V
- Dòng điện ra: tối đa 1A



Hình 8: IC ổn áp 7805

Sơ đồ khối nguồn:

Sử dụng 3 viên pin 18650 mắc nối tiếp, ta có điện áp nguồn đầu vào VCCH_in trong khoảng 11.1V ~ 12.6V. Sử dụng các tụ để lọc nhiễu và ổn định điện áp đầu vào và đầu ra.



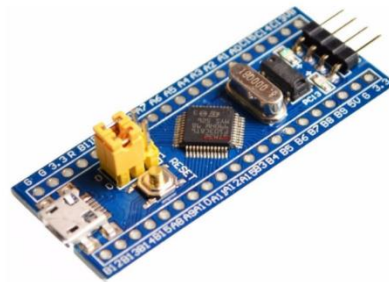
Hình 9: Sơ đồ Khối nguồn

3.1.2. Khối điều khiển

Kit Phát Triển STM32F103C8T6 là một bộ công cụ phát triển dành cho vi điều khiển STM32F103C8T6 của STMicroelectronics. STM32F103C8T6 là một vi điều khiển ARM Cortex-M3 mạnh mẽ và linh hoạt, được sử dụng rộng rãi trong các ứng dụng nhúng.

Bộ kit này được sử dụng để phát triển các ứng dụng nhúng đa dạng như thu thập dữ liệu từ các cảm biến từ đó phát tín hiệu điều khiển robot, xe tự hành.

- Điện áp hoạt động từ 2.0 → 3.6V
- Nguồn sử dụng: 5VDC từ cổng MiniUSB hoặc chân GPIO
- Thạch anh nội dùng dao động RC ở mode 8Mhz
- Tích hợp cổng USB, Thạch anh 32.768 KHz
- Tích hợp mạch chuyển USB-UART
- 64 kbytes bộ nhớ Flash
- 20 kbytes SRAM



Hình 10: Kit Phát Triển STM32F103C8T6

3.1.3. Khối động cơ

Động cơ DC giảm tốc GA12 N20 là một loại động cơ điện có kích thước nhỏ gọn và công suất phù hợp cho nhiều ứng dụng nhúng và thiết bị cần chuyển động. Dòng động cơ này thường được sử dụng trong robot nhỏ, máy in 3D, thiết bị tự động hóa nhỏ.

Có thể điều chỉnh tốc độ và lực xoắn của động cơ thông qua việc thay đổi điện áp đầu vào hoặc sử dụng các bộ điều chỉnh tốc độ và hướng điều khiển.

Dưới đây là một số đặc điểm chính của Động cơ DC giảm tốc GA12 N20:

- Tỷ số truyền: 1:100.
- Điện áp sử dụng: 1.5 ~ 12VDC
- Đường kính động cơ: 12mm.
- Chiều dài động cơ: 34mm.
- Đường kính trục: 3mm (trục dạng chữ D).
- Chiều dài trục: 10mm
- Tốc độ qua hộp số: 75rpm ~ 300rpm.



Hình 11: Động cơ DC giảm tốc GA12 N20

IC lái động cơ L298 là một IC được sử dụng để điều khiển động cơ DC hoặc các tải khác như các bơm, van, hoặc cảm biến. Nó được thiết kế dựa vào sơ đồ cầu H, có khả năng cung cấp dòng lớn đến động cơ và có thể được điều khiển bằng các tín hiệu điện tử từ mạch điều khiển

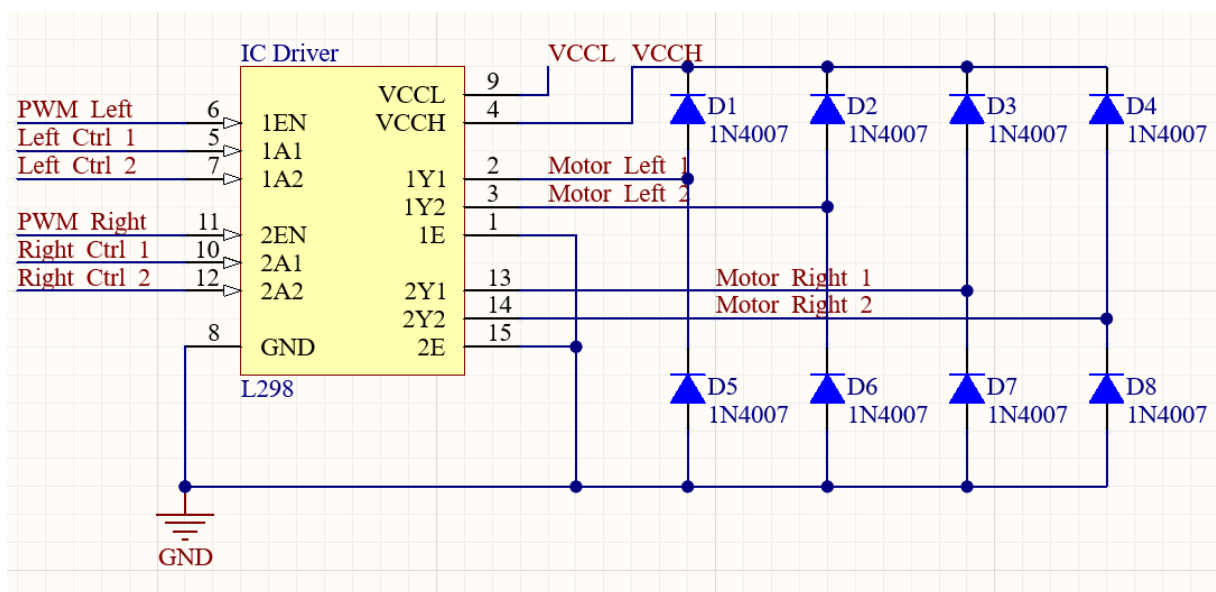
- Điện áp cấp cho động cơ (Tối đa): 46V
- Dòng điện cấp động cơ (tối đa): 2A
- Điện áp logic: 3.3 - 5V
- Điện áp hoạt động của IC: 5-35V
- Dòng điện hoạt động IC: 2A

- Dòng logic: 0-36mA
- Công suất tối đa (W): 25W
- Có tản nhiệt cho hiệu suất tốt hơn



Hình 12: IC lái động cơ L298

Sơ đồ khối động cơ:



Hình 13: Sơ đồ khối động cơ

3.1.4. Khối cảm biến

Module cảm biến siêu âm SRF05 là một thiết bị sử dụng sóng siêu âm để đo khoảng cách từ chính nó đến các vật thể xung quanh. Nó được sử dụng rộng rãi trong các ứng dụng đòi hỏi đo khoảng cách từ xa và chính xác.

SRF05 sử dụng sóng siêu âm để phát ra và sau đó nhận lại các tín hiệu phản xạ từ vật thể. Dựa trên thời gian mà sóng siêu âm đi và quay lại, nó có thể tính toán khoảng cách từ cảm biến đến vật thể.

- Điện áp: 5 V DC
- Dòng hoạt động: < 2mA
- Mức cao: 5V
- Mức thấp: 0V
- Góc tối đa: 15 độ
- Khoảng cách: 2cm – 450cm (4.5m)
- Độ chính xác: 3mm



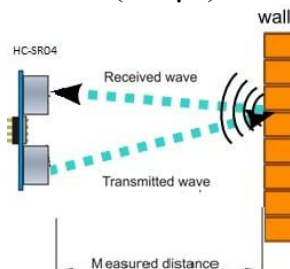
Hình 14: Module cảm biến siêu âm SRF05

Công thức để tính khoảng cách sử dụng cảm biến siêu âm:

$$d = \frac{v \times t}{2} = \frac{0.034 \times t}{2} = 0.017 \times t$$

Trong đó: d : Khoảng cách từ cảm biến đến vật cản (cm)

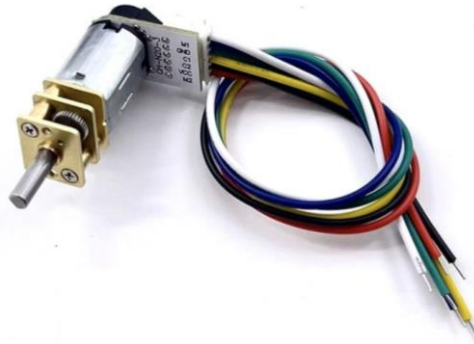
v : vận tốc âm thanh 0.034 (cm/ μ s)



Hình 15: Nguyên lý hoạt động của cảm biến siêu âm

Encoder dùng để tính toán tốc độ động cơ, được tích hợp sẵn vào động cơ GA12 N20.

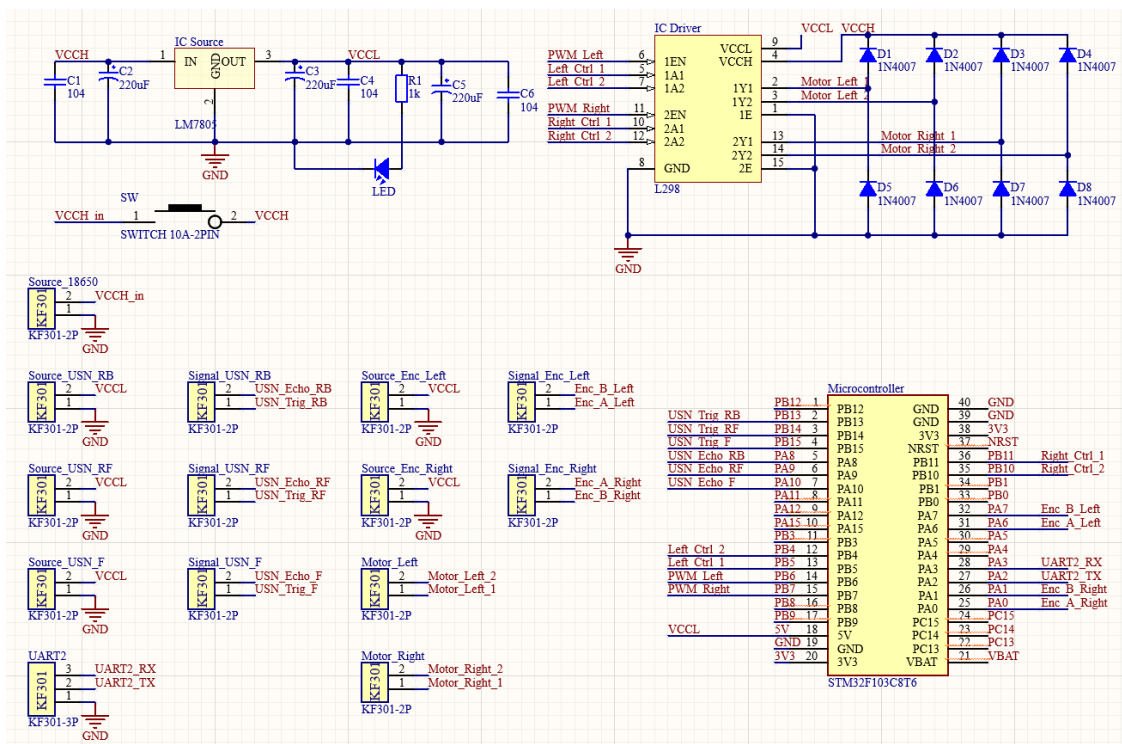
- Điện áp: 3.3-5 V DC
- Số xung mỗi vòng: 700 xung/vòng



Hình 16: Encoder được gắn vào động cơ

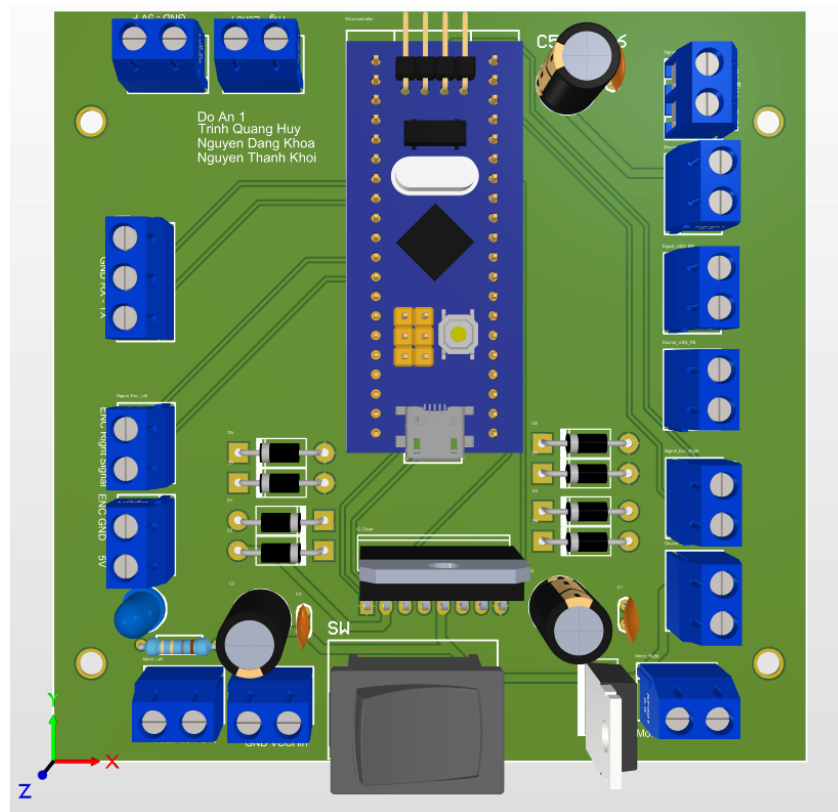
3.1.5. Bo mạch phần cứng

Sơ đồ schematic:



Hình 17: Sơ đồ schematic

Mạch PCB hoàn thiện:



Hình 18: Mạch PCB hoàn thiện

3.1.6. Sản phẩm hoàn thiện

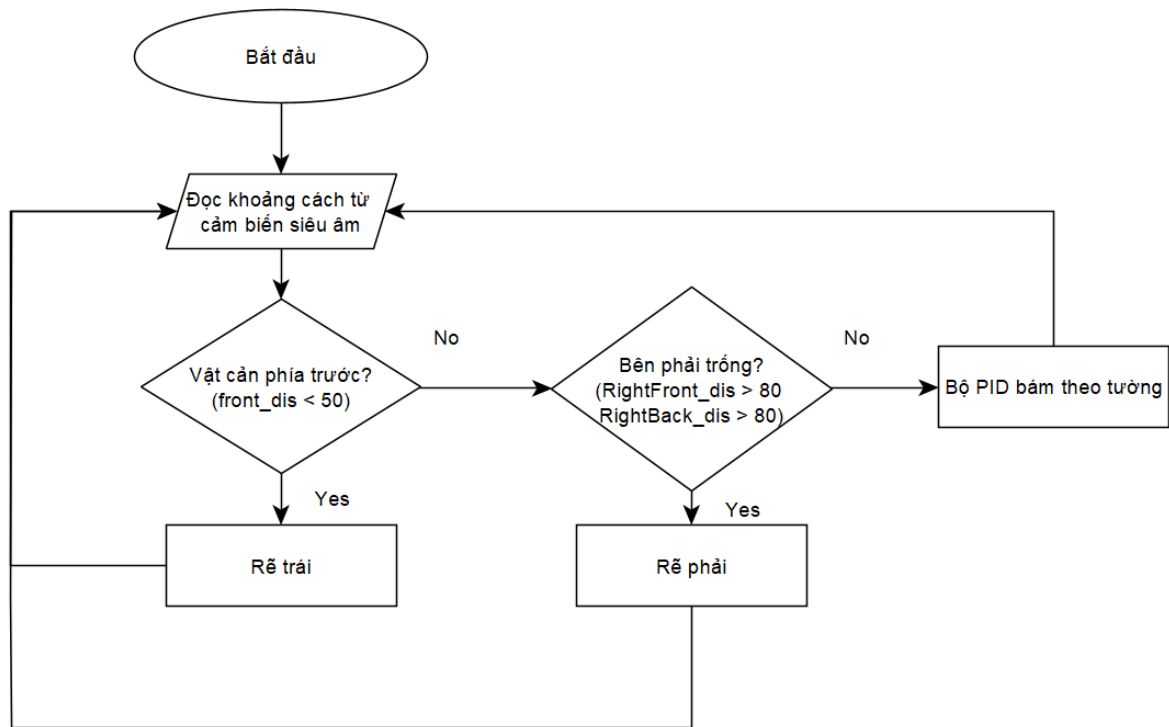
Sản phẩm robot hoàn thiện:



Hình 19: Sản phẩm Robot hoàn thiện

3.2. Thiết kế phần mềm

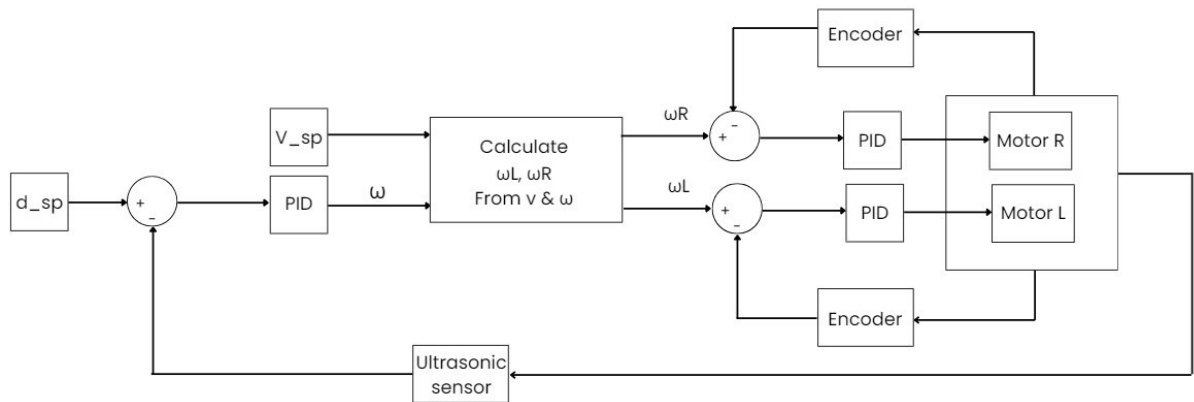
3.2.1. Lưu đồ giải thuật hành vi



Hình 20: Lưu đồ giải thuật hành vi

Khi khởi động robot, dữ liệu từ cảm biến siêu âm sẽ được đọc để xác định khoảng cách giữa robot và vật cản phía trước, bức tường cần bám theo. Xếp theo thứ tự ưu tiên, nếu như có vật cản phía trước thì robot sẽ rẽ trái để tránh vật cản. Nếu robot phát hiện bên phải có khoảng trống rộng thì sẽ rẽ phải. Nếu không rơi vào hai trường hợp trên, thì sẽ thực hiện giải thuật PID 2 vòng bám tường.

3.2.2. Lưu đồ giải thuật điều khiển PID 2 vòng bám tường



Hình 21: Lưu đồ giải thuật điều khiển

Bộ điều khiển gồm 2 vòng PID:

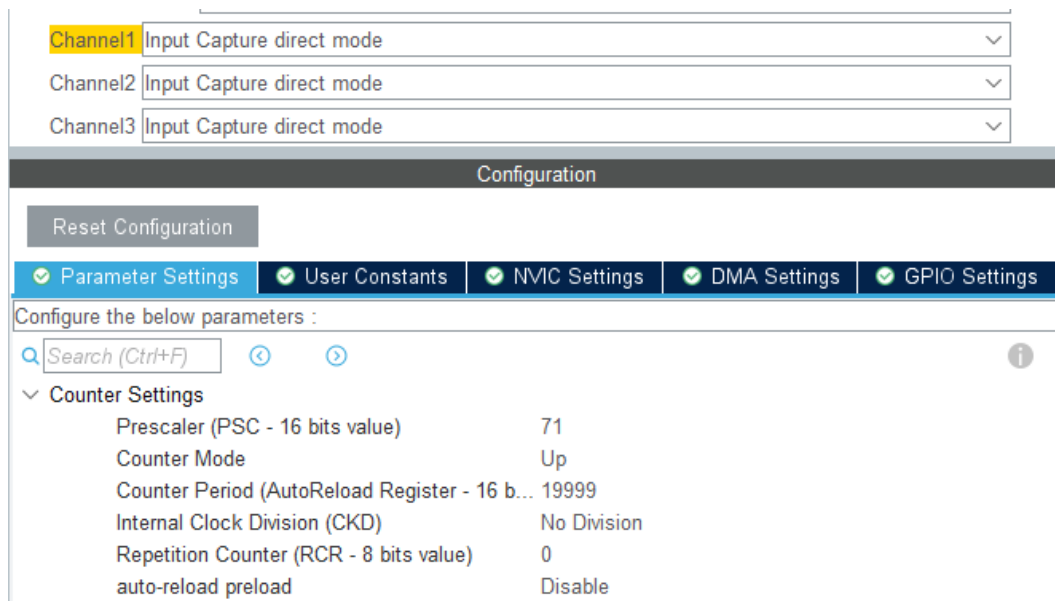
- Vòng ngoài: Tín hiệu vào là khoảng cách mong muốn giữa robot và tường, tín hiệu phản hồi là khoảng cách hiện tại giữa robot và tường (tính toán từ dữ liệu của hai cảm biến siêu âm bên phải). Lấy hiệu hai tín hiệu trên, ta có sai số về khoảng cách và đưa vào bộ PID để tính toán tốc độ góc cần xoay của robot (ω). Từ tốc độ xoay của robot và tốc độ dài đã đặt (v_{sp}), ta tính ra tốc độ xoay của bánh xe trái và phải (ω_l và ω_r).
- Vòng trong: Tín hiệu vào là tốc độ xoay của hai bánh xe trái và phải từ vòng PID ngoài, cùng với tín hiệu phản hồi là tốc độ xoay hiện tại của hai bánh xe. Lấy hiệu hai tín hiệu trên, ta có sai số về tốc độ xoay và đưa vào bộ PID để xuất xung PWM điều khiển động cơ.

3.2.3. Xử lý dữ liệu từ cảm biến

3.2.3.1. Cấu hình vi điều khiển xử lý tín hiệu cảm biến siêu âm

Sử dụng Timer 1 của STM32F103C8T6 đọc dữ liệu từ cảm biến siêu âm. Gồm 3 channel theo thứ tự channel 1, channel 2, channel tương ứng với Ultrasonic_Front, Ultrasonic_RightFront, Ultrasonic_RightBack.

Cấu hình Timer 1 cùng 3 channel ở chức năng Input Capture direct mode như sau:



Hình 22: Cấu hình xử lý tín hiệu cảm biến siêu âm

3.2.3.2. Cấu hình vi điều khiển xử lý tín hiệu từ encoder

Sử dụng Timer 2 và Timer 3 để đo tốc độ xoay của động cơ phải và động cơ trái.

Cấu hình Timer 2 và Timer 3 ở chế độ Encoder Mode, chế độ 4x như sau:

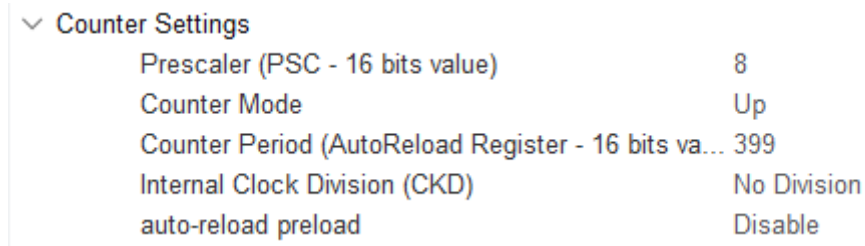


Hình 23: Cấu hình xử lý tín hiệu encoder

3.2.3.3. Cấu hình vi điều khiển xuất xung PWM

Sử dụng Timer 4 của STM32F103C8T6 để xuất xung PWM, cấp vào chân Enable của Driver L298N để điều khiển động cơ.

Nhóm cấu hình Timer4 điều chế xung PWM với tần số 10 kHz điều khiển động cơ.



Counter Settings	
Prescaler (PSC - 16 bits value)	8
Counter Mode	Up
Counter Period (AutoReload Register - 16 bits va...)	399
Internal Clock Division (CKD)	No Division
auto-reload preload	Disable

Hình 24: Cấu hình xuất xung PWM

$$f_{Timer4} = \frac{f_{APB1} / 2}{prescale + 1} = \frac{72 / 2}{8 + 1} = 4MHz$$

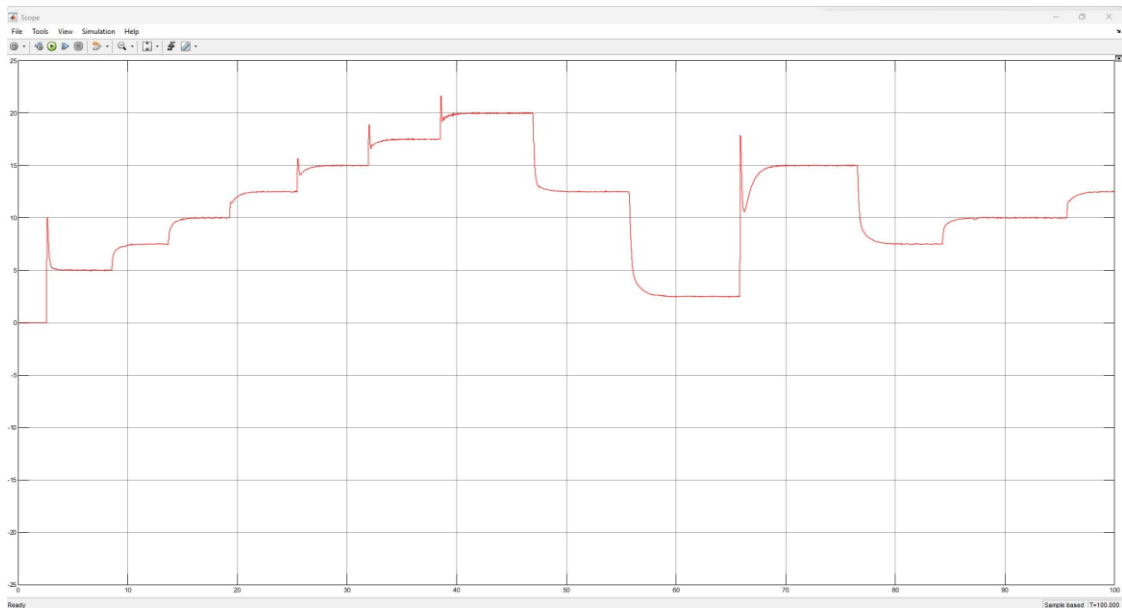
Timer 4 có tần số 4 MHz đồng nghĩa với việc khi ta cài đặt thanh ghi ARR = 399, khi đó bộ đếm của Timer 4 sẽ đếm từ 0 đến 399 rồi quay lại 0. Vòng đếm đó sẽ có chu kỳ là 10 kHz. Như vậy giá trị so sánh để điều chế xung PWM sẽ ở trong khoảng 0 – 399.

4. Kết quả và phương hướng phát triển đề tài

4.1. Kết quả đạt được

Nhóm đã thành công trong việc điều khiển động cơ đáp ứng theo tốc độ xoay mong muốn và giữ khoảng cách với bức tường, rẽ phải cũng như rẽ trái đúng theo mong muốn.

Điều khiển tốc độ động cơ:



Hình 25: đáp ứng của tốc độ động cơ

Giữ khoảng cách với tường:



Hình 26: Robot giữ khoảng cách với tường

Rẽ trái khi gặp vật cản:



Hình 27: Robot rẽ trái khi gặp vật cản

Rẽ phải khi bên phải trống:



Hình 28: Robot rẽ phải khi bên phải trống

4.2. Hạn chế đề tài

- Đáp ứng khi rẽ phải đôi khi có độ trễ lớn.
- Bánh xe sau bị trượt đối với mặt sàn ma sát không lớn.
- Bánh mắt trâu đa hướng phía trước dễ bị mắc kẹt vào khe ở giữa những tấm gạch.
- Thiếu chính xác về mặt cơ khí khi lắp đặt động cơ và cảm biến siêu âm bên phải xe.

4.3. Hướng phát triển đề tài

Ta có thể khắc phục được những hạn chế nêu trên để phát triển đề tài và cải thiện chất lượng Robot như sau:

Sử dụng bộ điều khiển tự chỉnh STR để điều khiển Robot, nhờ đó hệ thống có thể tự thay đổi bộ thông số trong quá trình vận hành để giữ vững chất lượng điều khiển khi có sự thay đổi từ nhiều yếu tố.

Thiết kế lại phần bánh xe để xe có thể di chuyển trên nhiều loại địa hình khác nhau.

Thiết kế hệ thống thị giác để robot có thể nhận dạng môi trường xung quanh một cách chính xác.

Tích hợp các công nghệ SLAM (Simultaneous Localization and Mapping) để robot có thể tự xây dựng bản đồ môi trường.

5. Kết luận

Từ những yêu cầu đặt ra của đề tài "**Thiết kế Robot tự hành bám tường**" bằng việc áp dụng kiến thức đã học được trong môn Cơ sở Điều khiển Tự động, Lý thuyết Điều khiển Nâng cao,... Nhóm chúng em đã thành công trong việc tự thiết kế và tạo ra Robot đúng như yêu cầu đề tài và đạt những kết quả mong muốn.

Mặc dù đề tài vẫn còn một số hạn chế như: Đáp ứng khi rẽ phải đôi khi có độ trễ lớn, bánh xe của Robot không chạy ổn định trong môi trường ma sát không lớn.

Ngoài việc khắc phục những điểm thiếu sót, nhóm cũng xin đề xuất phương hướng cải tiến cho Robot như: Thiết kế hệ thống thị giác để robot có thể nhận dạng môi trường xung quanh một cách chính xác; Tích hợp các công nghệ SLAM (Simultaneous Localization and Mapping) để robot có thể tự xây dựng bản đồ môi trường.

Chúng em xin chân thành cảm ơn thầy PGS TS. Huỳnh Thái Hoàng đã tận tình hướng dẫn, chỉ dạy và đã truyền đạt cho chúng em đầy đủ kiến thức để có thể hoàn thành được đề tài này.

6. Tài liệu tham khảo

- [1] Doãn Minh Đăng, “Robot di động theo dấu tường”, Luận văn tốt nghiệp Đại học Bách Khoa TP. Hồ Chí Minh.
- [2] Lê Duy Anh, “Thiết kế xe tự hành di động bám tường”, Đại học Bách Khoa Hà Nội.
- [3] Muamad Hamam Iqbal, Wahyu Sapto Aji, “Wall Following Control System with PID Control and Ultrasonic Sensor for KRAI 2018 Robot”, International Journal of Robotics and Controls.
- [4] Phan Văn Hoàn, “Nghiên cứu và kiểm chứng thuật toán bám tường cho robot tự hành”, Đại học Sư Phạm Kỹ Thuật TP. Hồ Chí Minh.