TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH

KHOA ĐIỆN ĐIỆN TỬ

BỘ MÔN KỸ THUẬT MÁY TÍNH – VIỄN THÔNG

…………o0o…………

ĐỒ ÁN 2

XE TỰ HÀNH DÙNG LIDAR

NGÀNH CÔNG NGHỆ KỸ THUẬT MÁY TÍNH

Sinh viên: NGUYỄN PHÚC TOÀN

MSSV: 19119139

VŨ THÙY PHƯƠNG TRINH

MSSV: 19119140

Hướng dẫn: PGS.TS. TRƯƠNG NGỌC SƠN

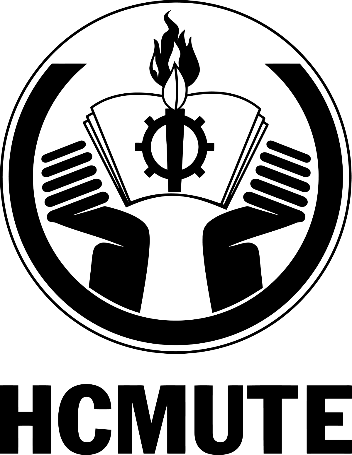
TP. HỒ CHÍ MINH 12/2024

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH

KHOA ĐIỆN ĐIỆN TỬ

**BỘ MÔN KỸ THUẬT MÁY TÍNH – VIỄN THÔNG**

…………o0o…………



ĐỒ ÁN 2

**XE TỰ HÀNH DÙNG LIDAR**

**NGÀNH CÔNG NGHỆ KỸ THUẬT MÁY TÍNH**

Sinh viên: **NGUYỄN PHÚC TOÀN**

MSSV: 19119139

**VŨ THÙY PHƯƠNG TRINH**

MSSV: 19119140

Hướng dẫn: **TS. TRƯƠNG NGỌC SƠN**

TP. HỒ CHÍ MINH 12/2024

**TRANG THÔNG TIN ĐỒ ÁN**

**1. Thông tin sinh viên**

|  |  |
| --- | --- |
| Họ và tên sinh viên: Nguyễn Phúc Toàn | MSSV: 19119139 |
| Email: 19119139@student.hcmute.edu.vn | Điện thoại: 0888322877 |
| Họ và tên sinh viên: Vũ Thùy Phương Trinh | MSSV: 19119140 |
| Email: 19119140@student.hcmute.edu.vn | Điện thoại: |

**2. Thông tin đề tài**

- Tên đề tài: XE TỰ HÀNH DÙNG LIDAR.

- Đơn vị quản lý: Bộ môn Kỹ Thuật Máy Tính – Viễn Thông, Khoa Điện – Điện Tử, Trường Đại học Sư phạm Kỹ Thuật Tp. Hồ Chí Minh.

- Thời gian thực hiện: Từ ngày 22/09/2024 đến ngày 30/12/2024.

- Thời gian bảo vệ: Ngày 30/12/2024

**3. Lời cam đoan của sinh viên**

“Tôi – Tên sinh viên I và Tên sinh viên II cam đoan đề tài này là do chính tôi nghiên cứu và thực hiện. Tôi không sao chép từ bất kì một bài viết nào đã được công bố mà không trích dẫn nguồn gốc. Nếu có bất kì một sự vi phạm nào, tôi xin chịu hoàn toàn trách nhiệm”.

Tp. Hồ Chí Minh, ngày 30 tháng 12 năm 2024

Sinh viên thực hiện đồ án

(Ký và ghi rõ họ tên)

Nguyễn Phúc Toàn Vũ Thùy Phương Trinh

|  |
| --- |
| Giảng viên hướng dẫn xác nhận quyễn báo cáo đã được chỉnh sửa theo đề nghị được ghi trong biên bản của Hội đồng đánh giá Khóa luận tốt nghiệp.  Tp. Hồ Chí Minh, ngày 30 tháng 12 năm 2024 .  Xác nhận của bộ môn Giáo viên hướng dẫn  (Ký và ghi rõ họ tên) |

|  |  |
| --- | --- |
| TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH  **KHOA ĐIỆN ĐIỆN TỬ**  **BỘ MÔN KT MÁY TÍNH – VIỄN THÔNG**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM  Độc lập – Tự do – Hạnh Phúc  \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

**Thành phố Hồ Chí Minh**, ngày 30 tháng 12 năm 2024.

**NHIỆM VỤ ĐỒ ÁN MÔN HỌC**

|  |  |  |
| --- | --- | --- |
| Họ và tên sinh viên I: | Nguyễn Phúc Toàn | MSSV: 19119139 |
| Chuyên ngành: | Công nghệ Kỹ Thuật Máy tính | Mã ngành: 119 |
| Hệ đào tạo: | CLC | Mã hệ: |
| Khóa: | K19 | Lớp: |
| Họ và tên sinh viên II: | Vũ Thùy Phương Trinh | MSSV: 19119140 |
| Chuyên ngành: | Công nghệ Kỹ Thuật Máy tính | Mã ngành: 119 |
| Hệ đào tạo: | CLC | Mã hệ: |
| Khóa: | K19 | Lớp: |

|  |  |
| --- | --- |
| TÊN ĐỀ TÀI: | XE TỰ HÀNH DÙNG LIDAR |

**I. NHIỆM VỤ:**

1. Các số liệu ban đầu:

* Số liệu 1.
* Số liệu 2.
* Số liệu 3.

1. Nội dung thực hiện:

* Nội dung 1.
* Nội dung 2.
* Nội dung 3.

**II. NGÀY GIAO NHIỆM VỤ:**

**III. NGÀY HOÀN THÀNH NHIỆM VỤ:**

**IV. HỌ VÀ TÊN CÁN BỘ HƯỚNG DẪN:**

|  |  |
| --- | --- |
| CÁN BỘ HƯỚNG DẪN | BM. KỸ THUẬT  MÁY TÍNH – VIỄN THÔNG |

|  |  |
| --- | --- |
| TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH  **KHOA ĐIỆN ĐIỆN TỬ**  **BỘ MÔN KT MÁY TÍNH – VIỄN THÔNG**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM  Độc lập – Tự do – Hạnh Phúc  \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

**Thành phố Hồ Chí Minh**, ngày 30 tháng 12 năm 2024.

**LỊCH TRÌNH THỰC HIỆN ĐỒ ÁN MÔN HỌC**

|  |  |  |
| --- | --- | --- |
| **Họ tên sinh viên:** | 1. Nguyễn Phúc Toàn | **MSSV:** 19119139 |
|  | 2. Vũ Thùy Phương Trinh | **MSSV:** 19119140 |
| **Lớp:** 19119 |  | **Đồ án:** Đồ án 2 |
| **Tên đề tài:** | XE TỰ HÀNH DÙNG LIDAR | |

|  |  |  |
| --- | --- | --- |
| **Tuần/ngày** | **Nội dung** | **Xác nhận GVHD** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**GV HƯỚNG DẪN**

(Ký và ghi rõ họ tên)

**BẢN NHẬN XÉT KHÓA LUẬN TỐT NGHIỆP**

(Dùng cho giảng viên hướng dẫn)

|  |  |  |
| --- | --- | --- |
| Để tài: | XE TỰ HÀNH DÙNG LIDAR | |
| Sinh viên thực hiện: | Nguyễn Phúc Toàn | MSSV: 19119139 |
|  | Vũ Thùy Phương Trinh | MSSV: 19119140 |
| Giáo viên hướng dẫn: | TS. Trương Ngọc Sơn |  |

**Nhận xét bao gồm các nội dung sau đây:**

1. Tính hợp lý trong các đặt và giải quyết vấn đề; ý nghĩa khoa học và thực tiễn:

Đặt vấn đề rõ ràng, mục tiêu cụ thể, đề tài có tính mới, cấp thiết có khả năng ứng dụng, tính sáng tạo.

2. Phương pháp thực hiện/phân tích/thiết kế:

Phương pháp hợp lý và tin cậy dựa trên cơ sở lý thuyết, có phân tích và đánh giá phù họp, có tính mới và sáng tạo.

3. Kết quả thực hiện/phân tích và đánh giá kết quả/kiểm định thiết kế:

Phù hợp với mục tiêu; phân tích và đánh giá / kiểm thử thiết kế hợp lý; có tính sáng tạo/kiểm định chặt chẽ và bảo đảm chất độ tin cậy

4.Kết luận và đề suất:

Kết luận phù hợp với cách đặt vấn đề, đề xuất mang tính cải tiến và thực tiễn; kết luận có đóng góp mới mẻ, đề suất sáng tạo và thuyết phục.

5. Hình thức trình bày, bố cục và chất lượng báo cáo:

Văn phong nhất quán, bố cục hợp lý, cấu trúc rõ ràng, đúng định dạng mẫu; có tính hấp dẫn, thể hiện năng lực tốt, văn bản trau chuốt.

6. Kỹ năng chuyên nghiệp và có tính sáng tạo:

Thể hiện kỹ năng giao tiếp, kỹ năng làm việc nhóm và các kỹ năng khác trong việc thực hiện đề tài.

7. Tài liệu trích dẫn:

Tính trung thực trong việc trích dẫn tài liệu tham khảo; tính phù hợp của các tài liệu trích dẫn; trích dẫn theo đúng chỉ dẫn APA.

8. Đánh giá về sự trùng lặp của đề tài:

Cần khẳng định đề tài có trùng lặp hay không? Nếu có, đề nghị ghi rõ mức độ, tên đề tài, nơi công bố, năm công bố của đề tài đã công bố.

9: Những nhược điểm và thiếu sót, những điểm cần được bổ sung và chỉnh sữa\*

10. Nhận xét tinh thần, thái độ học tập và nghiên cứu của sinh viên:

**Câu hỏi sinh viên phải trả lời trước hội đồng\*** (ít nhất 02 câu)

**Đánh giá chung**

- Điểm (Quy về thanh điểm 10 không làm tròn): ……../10.

- Xếp loại chung (Xuất sắc, Giỏi, Khá, Trung, Yếu, Kém):

**Đề nghị của giảng viên hướng dẫn**

Ghi rõ: “Báo cáo đạt/không đạt yêu cầu của một khóa luận tốt nghiệp kỹ sư, và được phép/không được phép bảo vệ khóa luận tốt nghiệp’

Tp.HCM, ngày 30 tháng 12 năm 2024.

**Người nhận xét**

# LỜI MỞ DẦU

# LỜI CẢM ƠN

|  |  |  |
| --- | --- | --- |
|  | **Người thực hiện đề tài** | |
|  | Nguyễn Phúc Toàn | Vũ Thùy Phương Trinh |

**MỤC LỤC**

[DANH MỤC HÌNH ix](#_Toc181476468)

[DANH MỤC BẢNG x](#_Toc181476469)

[DANH MỤC TỪ VIẾT TẮT VÀ THUẬT NGỮ xi](#_Toc181476470)

[CHƯƠNG 1 TỔNG QUAN 1](#_Toc181476471)

[1.1. ĐẶT VẤN ĐỀ 1](#_Toc181476472)

[1.2. MỤC TIÊU 1](#_Toc181476473)

[1.3. NỘI DUNG NGHIÊN CỨU 1](#_Toc181476474)

[1.4. GIỚI HẠN ĐỀ TÀI 1](#_Toc181476475)

[1.5. BỐ CỤC 1](#_Toc181476476)

[CHƯƠNG 2 CƠ SỞ LÝ THUYẾT 2](#_Toc181476477)

[2.1. TỔNG QUAN VỀ THỊ GIÁC MÁY 2](#_Toc181476478)

[2.1.1. Đặt vấn đề 2](#_Toc181476479)

[2.1.3. \*\*\*\*\*\*\*\*\*\*\*\* 2](#_Toc181476480)

[2.2. TỔNG QUAN VỀ THUẬT TOÁN SFM 2](#_Toc181476481)

[2.3. TỔNG QUAN VỀ LIDAR VÀ ỨNG DỤNG 3](#_Toc181476482)

[2.4. TỔNG QUAN VỀ SÓNG SIÊU ÂM VÀ ỨNG DỤNG 3](#_Toc181476483)

[2.5. TỔNG QUAN VỀ XỬ LÝ ẢNH 3](#_Toc181476484)

[2.6. TỔNG QUAN VỀ XE TỰ HÀNH 3](#_Toc181476485)

[2.7. TỔNG QUAN VỀ THUẬT TOÁN SLAM 3](#_Toc181476486)

[2.7.1. Khái niệm 3](#_Toc181476487)

[2.7.2. Các biến thể SLAM 3](#_Toc181476488)

[2.8. TỔNG QUAN VỀ NVIDIA JETSON NANO 4](#_Toc181476489)

[CHƯƠNG 3 THIẾT KẾ XE TỰ HÀNH 5](#_Toc181476490)

[CHƯƠNG 4 THUẬT TOÁN SLAM 6](#_Toc181476491)

[CHƯƠNG 5 THUẬT TOÁN QUÉT LIDAR 3D 7](#_Toc181476492)

[CHƯƠNG 6 KIỂM TRA VÀ ĐÁNH GIÁ 8](#_Toc181476493)

[CHƯƠNG 7 HƯỚNG PHÁT TRIỂN 9](#_Toc181476494)

[TÀI LIỆU KHAM KHẢO 10](#_Toc181476495)

[TRÍCH DẪN 11](#_Toc181476496)

# DANH MỤC HÌNH

[Hình \*.\*: \*\*\*\*\*\*\*\*\*\*\*\*\*\* 2](#_Toc181476399)

[Hình 2.1 Minh họa thuật toán SfM 2](#_Toc181476400)

# DANH MỤC BẢNG

[Bảng \*: \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 2](#_Toc180072399)

# DANH MỤC TỪ VIẾT TẮT VÀ THUẬT NGỮ

**SLM**: (Structure From Motion) kỹ thuật sử lý ảnh để thực hiện tái dựng lại vật hay môi trường.

**SLAM**: (Simultaneous Localization And Mapping)

**vSLAM:** (vSLAM)

**Visual Odometry**: (Visual Odometry)

# CHƯƠNG 1 TỔNG QUAN

## ĐẶT VẤN ĐỀ

Tự động hóa vẫn luôn là chủ đề nghiên cứu thu hút nhiều sự quan tâm trong thời đại ngày nay. Những Tự động hóa vẫn luôn là chủ đề nghiên cứu thu hút nhiều sự quan tâm trong thời đại ngày nay. Những chiếc xe hay robot tự điều khiển mà không cần đến tác động của con người. Đây cũng là đề tài có tiềm năng rất lớn trong nhiều lĩnh vực khác nhau, là tiền đề cho các thiết bị hỗ trợ trong sinh hoạt sống của con người. Nhóm thực hiện đã dựa vào đề tài này để đề xuất mục tiêu cho đồ án 2, đó là hệ thống xe tự hành sử dụng LiDAR.

Ý tưởng của đề tài chính là thiết kế một chiếc xe tự hành sử dụng cảm biến LiDAR. LiDAR (Light Detection and Ranging) là công nghệ đo khoảng cách bằng cách sử dụng laser để tạo ra bản đồ 3D của môi trường xung quanh. Thiết bị này có thể vẽ lại bản đồ của khu vực để xác định phương hướng thích hợp cho việc di chuyển, và nhờ sử dụng những chùm tia laser mà thiết bị có thể đo được khoảng cách đến chướng ngại vật để né vật cản một cách chủ động.

## 1.2. MỤC TIÊU

Mục tiêu nghiên cứu

Nghiên đối tượng nào

Kết quả (sản phẩm) sau khi nghiên cứu

Ứng dụng của sản phẩm

Sản phẩm hệ thống xe tự hành cần phải đạt được một số yêu cầu của người dùng cũng như yêu cầu hệ thống. Xe tự hành có thể vận hành tự động mà không cần điều khiển. Thực hiện quét xung quanh để tạo bản đồ của khu vực, và từ bản đồ đó di chuyển né các chướng ngại vật trên đường đi. Hệ thống cần đạt đến độ chính xác cao, thời gian quét môi trường thấp và tối ưu trong việc di chuyển.

## 1.3. NỘI DUNG NGHIÊN CỨU

## 1.4. GIỚI HẠN ĐỀ TÀI

Từ mục đích nghiên cứu thì khả năng thực hiện có thể làm đến đâu

Tính năng của sản phẩm

Ngoài các chức năng của hệ thống, hệ thống vẫn có một số nhược điểm chưa thể được tối ưu. Có thể kể đến như cảm biến LiDAR có giá thành khá cao so với các cảm biến dò đường thông thường. Laser từ cảm biến khi gặp một số môi trường phản xạ kém có thể làm sai lệch về độ chính xác. Ngoài ra độ rộng mà cảm biến có thể quét được vẫn cần được tối ưu thêm.

## 1.5. BỐ CỤC

**Chương 1:** Tổng quan

Đặt vấn đề và giới thiệu tổng quát về đề tài, mục tiêu cũng như giới hạn đặt ra.

**Chương 2:** Cơ sở lý thuyết

Đưa ra một số lý thuyết cần thiết sử dụng trong đề tài.

**Chương 3:** Thiết kế hệ thống

Trình bày sơ đồ hệ thống và giải thích cơ chế hoạt động của hệ thống.

Các phương pháp lựa chọn phần cứng sao cho phù hợp với yêu cầu của đề tài.

**Chương 4:** Thuật toán SLAM

Đi sâu vào chi tiết của thuật toán và cách áp dụng thuật toán vào thiết kế.

**Chương 5:** Thuật toán quét LiDAR 3D

Đi sâu vào chi tiết của thuật toán và cách áp dụng thuật toán vào thiết kế.

**Chương 5:** Kết quả thực hiện

Trình bày kết quả đã thực hiện về phần cứng và phần mềm.

**Chương 6:** Kết quả và tổng hợp

Đưa ra các kết quả thực nghiệm và đánh giá hệ thống.

**Chương 7:** Kết luận và hướng phát triển

Kết luận về hệ thống và hướng phát triển.

# CHƯƠNG 2 CƠ SỞ LÝ THUYẾT

## 2.1. TỔNG QUAN VỀ XE TỰ HÀNH

## 2.2. TỔNG QUAN VỀ LIDAR VÀ ỨNG DỤNG

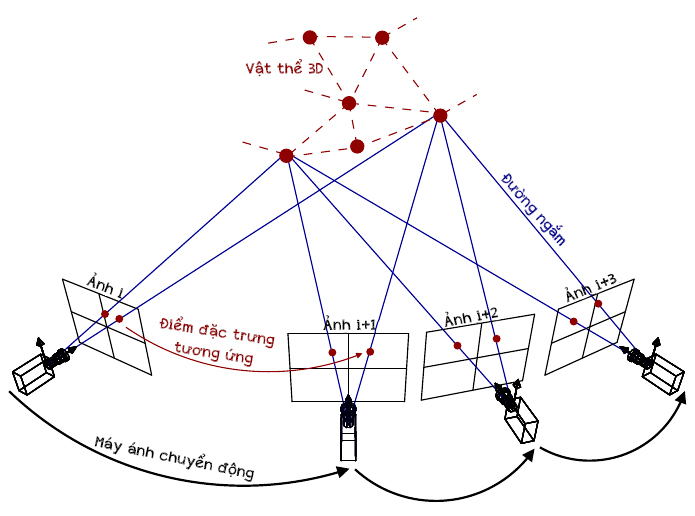
2.2.1. Giới thiệu chung về Lidar

2.2.2. Khả năng và giới hạn của lidar

## 2.3. TỔNG QUAN VỀ THUẬT TOÁN SFM

### 2.3.1. Giới thiệu chung

SFM (Structure from Motion) là một lĩnh vực trong thị giác máy tính hướng đến việc tái tạo lại cấu trúc ba chiều của môi trường từ một chuỗi hình ảnh hai chiều. Công nghệ này có thể được ứng dụng trong đa dạng các lĩnh vực như robot, khảo cổ học và quy hoạch đô thị. Trọng tâm của SfM là một tập hợp các thuật toán phức tạp cho phép trích xuất thông tin không gian từ một loạt các hình ảnh.



##### Hình 2.1 Minh họa thuật toán SfM

Thuật toán SfM cho phép tính toán đồng thời các ma trận chiếu và các điểm 3D bằng cách sử dụng các điểm tương ứng trong mỗi khung nhìn. Đây là các bước liên quan đến quá trình tái tạo cảnh 3D:

* Trích xuất điểm đặc trưng.
* Tương thích điểm đặc trưng.
* Tính toán các điểm 3D.
* Tái cấu trúc 3D.

### 2.3.2 Ứng dụng

Thuật toán SfM đã và đang được áp dụng rộng rãi trong nhiều lĩnh vực khác nhau với các ứng dụng như:

+ Trò chơi điện tử: Thuật toán được sử dụng để tạo môi trường 3D và các đối tượng trong thế giới ảo, trò chơi điện tử và ứng dụng thực tế ảo. Bằng cách ước lượng cấu trúc 3D từ các hình ảnh, SfM giúp xây dựng môi trường 3D chân thực và sống động.

+ Địa lý và bản đồ số: Trong lĩnh vực địa lý, SfM được sử dụng để tạo ra bản đồ 3D của môi trường từ dữ liệu hình ảnh thu thập từ máy bay không người lái (UAV) kết hợp với xử lý ảnh.

+ Điều khiển thiết bị tự hành: SfM được sử dụng để hỗ trợ việc định vị và xây dựng bản đồ cho robot tự động và xe tự hành.

+ Giám sát và an ninh: SfM có thể được áp dụng trong giám sát và an ninh để theo dõi và phát hiện chuyển động, xây dựng mô hình 3D của một khu vực và phân tích hành vi.

## 2.4. TỔNG QUAN VỀ THUẬT TOÁN SLAM

### 2.4.1. Khái niệm

SLAM (Simultaneous Localization And Mapping) là thuật toán ứng dụng cho các thiết bị xe tự hành, thuật toán có thể xây dựng bản đồ cũng như xác định vị trí của thiết bị trong cùng một lúc. Thuật toán này cũng cho phép các thiết bị tự hành nhận biết bản đồ trong các môi trường khác nhau.

SLAM đã trở thành đề tài nghiên cứu trong nhiều năm qua. Với những cải tiến về tốc độ xử lý của máy tính và sự sẵn có của các cảm biến giá rẻ như máy ảnh và máy đo phạm vi như laser, thuật toán SLAM hiện được ứng dụng trong ngày càng nhiều lĩnh vực khác nhau.

Quy trình hoạt động của SLAM gồm hai bước chính:

+ Định vị (Localization): Dựa trên laser của cảm biến để xác định vị trí của thiết bị theo thời gian.

+ Lập bản đồ (Mapping): Nhờ vào cảm biến để thu được môi trường xung quanh, sau đó di chuyển để lấy được thông tin trực quan hơn, trích xuất các đặc điểm để có thể xây dựng thành một bản đồ.

Ứng dụng: Thuật toán SLAM hiện được ứng dụng trong một số thiết bị tự vận hành như robot quét dọn, xe hơi tự động hay robot tự hành.

Một số nhược điểm của thuật toán:

+ Lỗi sai lệch định vị gây ra chênh lệch đáng kể so với giá trị thực tế. Thuật toán SLAM chuyển động tuần tự bao gồm những vị trí sai số. Lỗi này tích lũy theo thời gian, gây ra sai lệch đáng kể so với giá trị thực tế. Nó cũng có thể khiến dữ liệu bị biến dạng gây khó khăn cho việc xác định bản đồ.

+ Chi phí tính toán cao trong việc xử lý hình ảnh và tối ưu hóa.

+ Chi phí các cảm biến như LiDAR, camera phân giải tốt có giá thành cao.

### 2.4.2. Các biến thể SLAM

SLAM trực quan (Visual SLAM hoặc vSLAM) sử dụng hình ảnh thu được từ máy ảnh và các cảm biến hình ảnh khác. Visual SLAM có thể được sử dụng bởi các loại máy ảnh đơn giản (máy ảnh góc rộng, mắt cá và hình cầu), máy ảnh mắt ghép (âm thanh nổi và đa máy ảnh), máy ảnh RGB-D (máy ảnh độ sâu và máy ảnh thời gian bay ToF). Visual SLAM có thể được triển khai với chi phí tương đối thấp. Ngoài ra, vì camera cung cấp một lượng lớn thông tin nên chúng có thể được sử dụng để phát hiện các điểm mốc (các vị trí đã đo trước đó). Việc phát hiện cột mốc cũng có thể được kết hợp với tối ưu hóa dựa trên biểu đồ, mang lại sự linh hoạt trong việc triển khai SLAM. SLAM một mắt (Monocular SLAM) là một thuật toán vSLAM sử dụng một camera duy nhất làm cảm biến, điều này khiến việc xác định độ sâu trở nên khó khăn. Vấn đề này có thể được giải quyết bằng cách phát hiện điểm đánh dấu AR, bàn cờ hoặc các vật thể đã biết khác trong ảnh để định vị hoặc bằng cách kết hợp thông tin camera với một cảm biến khác như đơn vị đo quán tính (IMU), có thể đo các đại lượng vật lý như vận tốc và hướng. Công nghệ liên quan đến vSLAM bao gồm cấu trúc từ chuyển động (SfM), đo hình ảnh trực quan và điều chỉnh bó.

LiDAR SLAM (Light Detection And Ranging SLAM) là phương pháp chủ yếu sử dụng cảm biến laser (hoặc cảm biến khoảng cách). So với máy ảnh bình thường, máy ảnh ToF và các cảm biến khác, tia laser có độ chính xác cao hơn và được sử dụng cho các ứng dụng có phương tiện di chuyển tốc độ cao như ô tô tự lái và máy bay không người lái. Các giá trị đầu ra từ cảm biến laser thường là dữ liệu điểm 2D (x, y) hoặc 3D (x, y, z). Cảm biến laser cung cấp các phép đo khoảng cách có độ chính xác cao và hoạt động hiệu quả trong việc xây dựng bản đồ bằng thuật toán SLAM. Tính toán chuyển động (quãng đường đã di chuyển) được sử dụng để định vị phương tiện. Ngoài ra có thể sử dụng phương pháp tiếp cận dựa trên tính năng như Lidar Odometry And Mapping (LOAM) hoặc Fast Global Registration (FGR), dựa trên các tính năng của FPFH. Bản đồ đám mây điểm 2D hoặc 3D có thể được biểu diễn dưới dạng bản đồ lưới hoặc bản đồ ba chiều voxel. Đối với các ứng dụng robot trong nhà, SLAM lidar 2D thường được sử dụng, trong khi SLAM 3D thường được sử dụng cho phương tiện không người lái (drone, flycam,...) và xe tự hành.

## 2.x. TỔNG QUAN VỀ ĐỘNG CƠ ENCODER

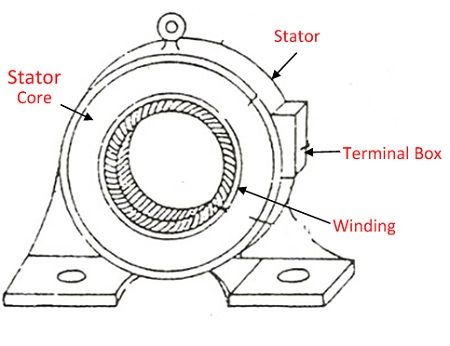
### 2.x.1. Khái niệm về động cơ dc

Động cơ DC (direct current motor) là một động cơ có cấu tạo đơn giản từ 2 bộ phận trục quay động cơ (Rotor) và lớp vỏ tĩnh (Sator) bao quanh Rotor. Sator có vai trò bộ phận được cố định trong một động cơ để từ đó rotor tạo ra mô men xoắn từ đó có thể quay.

***Sator*** được chia được 3 bộ phận chính:

+ Phần vỏ - khung ( ***Terminal box*** ): có chức năng cố định phần lõi sator để từ có thể tạo ra vùng từ trường biến thiên mạnh từ đó cho phép rotor quay với tốc độ cao. Cũng như là phần khung để đi dây đồng cho cuộn dây ( nếu có ).

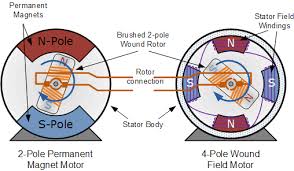
+ Lõi sator ( ***Core*** ) với các cuộn dây ( ***Winding*** ): Phần lõi của sator thường là cấu tạo từ các nam châm vĩnh cửu với các cực của nam châm xem kẻ nhau điều này giúp cho phần lõi đơn giản hơn nhưng không thế tạo ra từ trường mạnh. Còn lại là khi lõi được tạo ra lõi sắt non và các cuộn dây lúc bấy giờ khi từ trường của nam châm điện lúc này mạch hơn rất nhiều từ đó động cơ sẽ có lực kéo mạnh hơn hay mô men xoắn cao hơn.



##### Hình 2.x: Bảng vẽ cơ bảng của sator



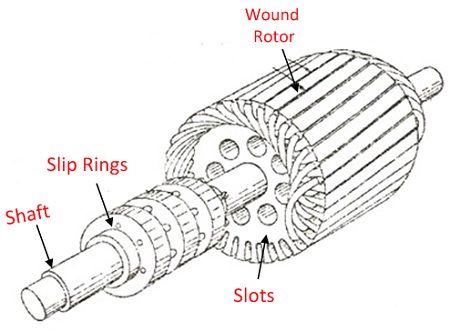
##### Hình 2.x: Ảnh thực tế của một lõi sator

 Phần lõi Sator sẽ được chia nhỏ thành các mảng nhỏ và được quấn dây đồng xen kẻ nhau từ đó làm cho phần từ trường có thế dày đặc và mạch hơn cho phép động cơ đạt được số vòng quay RPM lớn hơn hoặc động cơ có thể kéo được một mức tải lớn hơn.

##### Hình 2.z: So sánh lõi Sator từ nam châm vĩnh cửu và nam châm điện

Cũng từ phần lõi Sator ta có thể suy ra rằng phần lõi rotor cũng có thể chia thành 2 loại chính. Một loại là một nam châm vĩnh cữu với ưu điểm là đơn giản nhỏ gọn nhưng dẫn đến sức mạng của động cơ phụ thuộc nhiều vào từ trường của sator và yêu cầu phỉa thiết kế phần lõi sator phức tạp hơn.

Loại 2 là được cấu thành từ lõi thép và cuộn cảm với độ phức tạp cao do yêu cầu phải kết thêm phần truyền điện cho cuộn cảm hoặc rotor được cấu thành từ nam châm vĩnh cửu.



##### Hình 2.x: Bảng vẻ cấu tạo của Rotor

Chức năng các bộ trong rotor:

+ Trục dẫn ( ***Shaft*** ): thường dùng để truyền tài mô men xoắn từ phần lõi đến bánh xe hoặc băng truyền hoặc cánh quạt …

+ Vòng trượt ( ***Slip rings*** ): thường xuất hiện trên các rotor cấu thành từ nhiều cuộn cảm có chức năng truyền điện đến phần lõi của rotor để tạo ra từ cảm.

+ Lõi rotot ( ***Wound rotor*** ): Lõi được cấu tạo từ các lá thép với các cuộn dây được quấn theo một chiều cố định để từ đó có thể tạo ra dòng từ trường cần thiết để giúp cho rotor quay, các rảnh ( ***Slot*** ) trên phần thân rotor thường được thêm vào để rotor nhẹ hơn và không cần nhiều điện để rotor chạy hơn.

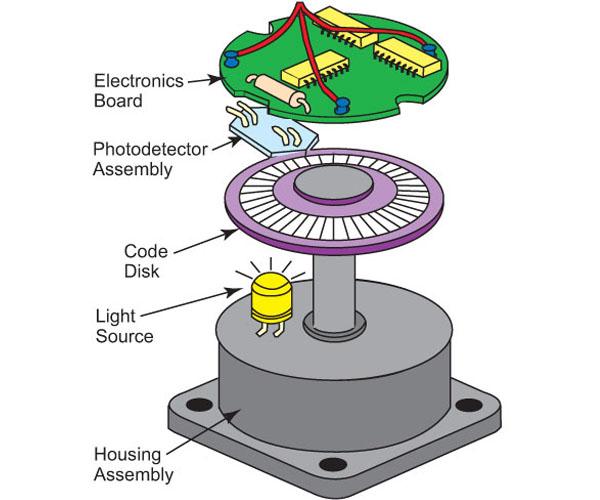


##### Hình 2.x: Ảnh chụp lõi của rotor

Thông qua phần ảnh trên chủng ta có thể thấy được rẳng phần phần vòng trượt của rotor cũng có thế được chia thành cách rảnh từ đó nhà sản suất có thế thế kế lõi rotor với nhiều cực cũng như cho phép rotor nhẹ hơn từ đó có thế đạt được tốc độ quay cao hơn.

### 2.x.2. Cấu tạo cơ bản của bộ giải mã cho động cơ encoder

Với việc động cơ DC chỉ chạy theo chiều và cường độ của dòng điện nên việc xác định tốc độ quay ( RPM ) rất khó khăn là việc dừng motor đúng thời điểm hay ở khoảng cách cố định là gần như không thể, dẫn đến việc điều khiển chính xác trong các ứng dụng cách tay robot hay servo trở nên không khả thi, như nếu người dùng có thế xác định được góc quay của rotor thì những việc trên trở nên khả thi hơn.



##### Hình 2.x. : Thiết kế cơ bản của bộ encoder cho động cơ DC

Encoder thường được có 2 loại chính

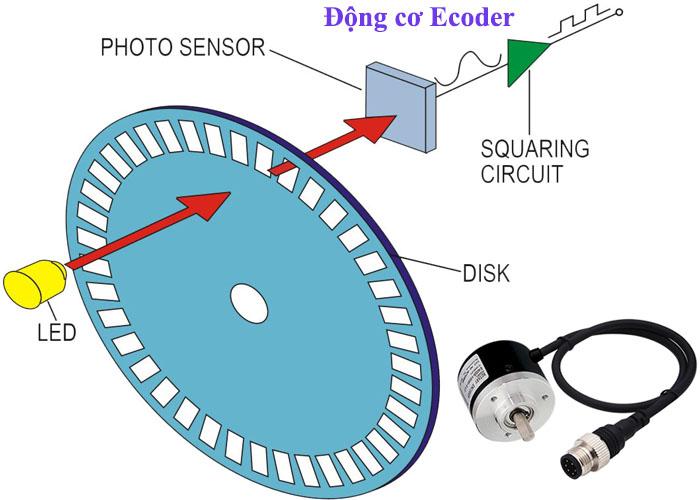
\_ Ecoder trường của rotor để xác định tốc độ của động cơ với độ bền cao có thể hoạt động trong các môi trường khắc nghiệt những khi hoạt động trong gần các thiết bị điện tử thì từ trường sẽ gây nhiễu có thiết bị điện tử thậm chí là hư hỏng link kiện điện tử.

\_ Encoder dùng đèn led thường là led hồng ngại và quang trở với thiết kế đơn giản giá thành rẻ độ chính xác cao cấu tạo 4 bộ phận chính:

+ Led và quang trở ( ***Photodetector Assembly*** ) với vai trờ như một nút nhấn kéo lên ( pull-up button ) và khi có vật cản đi qua thì mất đi nguồn sáng của led lúc này giá trị quang trở tăng làm cho dòng điện đi qua yếu hoặc ngắt.

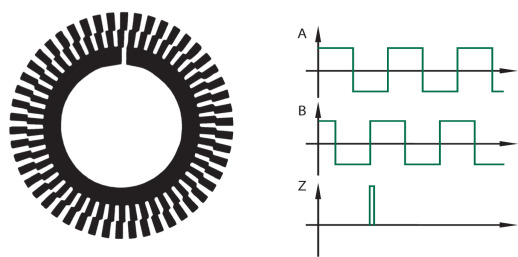
+ Đĩa quay ( ***Code disk*** ) đóng vai trò tạo nguồn xung cho quang trở bằng việc thực hiện cho để cho ánh sáng đi đến quang trở thông qua các rảnh được khắc sẵn trên đĩa. Đĩa được nối với trục quay của rotor nên số vòng quay giữa bánh đĩa và trục xoay ( bánh xe ) sẽ hoàn toàn giống nhau nếu độc cơ không phải đi qua hộp số, nếu có thì người dùng cần tìm hiểu tỷ lệ giảm tốc của hộp số.

+ Mạch điện ( ***Electronics Board*** ) với vai trò là chuyền năng lượng đến led cũng như thực hiện giải mã tính hiệu dòng điện thông qua quang trở từ tính hiệu Analog sang tín hiệu số để tiện cho vi điều khiển xử lý.



##### Hình 2.x. : Sơ đồ hoạt động của Encoder

### 2.x.3. Lý thuyết cơ bản trong ứng dụng

 Với việc thông qua quang trở nên tín hiệu cần được xử lý thông qua trở nên tính hiệu là dạng Analog chúng ta có thế biểu hiện nó như một sóng sin để có thế nhận biết được tín hiệu tín hiệu số cho vi điều khiển có thể xử lý thì tín hiện phải đi qua op-amp tín hiệu đi qua lúc này sẽ là tín hiệu xung.

##### Hình 2.x. : Bảng vẻ đĩa che với 2 xung A và B

Trên đĩa quay khi khoét các lỗ **r** cách tâm một khoảng **R** thì mọi điểm nằm trên đường thẳng từ tâm đều r đều sẽ di chuyển cùng nhau dẫn lúc này người dùng không thế xác định chiều quay của motor để có thể làm được điều đó nhà sản xuất đã thêm vào hệ thống rảnh thứ 2 và được đặt lệch vài độ với phần rảnh thứ 1 từ đó khi 2 tính hiệu led sẽ luôn lệch nhau 1 khoảng thời gian **n.**

Với khoảng thời gian lệch **n** một trong 2 tín hiệu xung sẽ luôn đến sớm hơn:

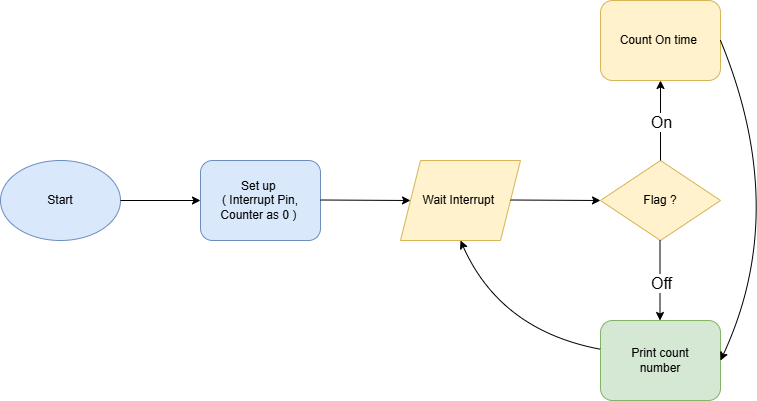
+ Xung tín hiệu **A** sẽ sớm hơn tín hiệu B một khoảng khi phần đen của rảnh trên di chuyển nhanh hơn rảnh đen dưới một khoảng **n** nhưng cũng từ đó mà phần rảnh đen của A cũng lớn hơn.

+ Khi động cơ quay ngược lại thì lúc này do thanh chắn của rảnh lớn hơn khi đã bị làm lệch sẽ làm cảm biến phía trên sẽ quay chậm dẫn đến tín hiệu xung B xuất hiện sớm hơn một khoảng **n**.

### 2.x.4. Các công thức tính và sơ đồ khối của động cơ

2.x.4.1. Công thức tính toán cho động cơ có encoder

Một trong những số liệu quan trọng cho phép người dùng kiểm soát được góc quay của motor trong các ứng dụng yêu cầu động cơ phải chuyển động với độ chính xác cao đó tống số xung cho một vòng quay. Từ tổng số xung cho một vòng quay thì người dùng có thế biết được trong tốc độ góc quay của động cơ là bao nhiêu ( raido/s ), cũng từ đây người dùng sẽ có thông tin về quảng đường mà động cơ có thế kéo được với một số lượng xung cố định.

Số xung cho 1 vòng quay của động cơ – độ phân giải tối thiểu ( Pulse Per Revolution - PPR ), để xác định được thì người dùng có thể đọc datasheet của nhà sản xuất về động cơ hoặc dùng vdk để đọc dữ liệu bằng việc xoay tay động cơ.

##### Hình 2.x. : Sơ đồ khối thiết kế code đọc xung cho vdk

Góc quay nhỏ nhất của động cơ: thông số này tính toán góc quay nhỏ nhất của motor trong một ứng nhất định nào đó. Với việc động cơ quay trò, ta có.

|  |  |
| --- | --- |
|  | N: số xung cho motor quay 360 độ (PPR)  I: độ quay nhỏ nhất ( ⁰ , radian ) |

VD: Với việc ứng dụng cho cánh tay robot người dùng cần độ quay nhỏ nhất là 4 độ.

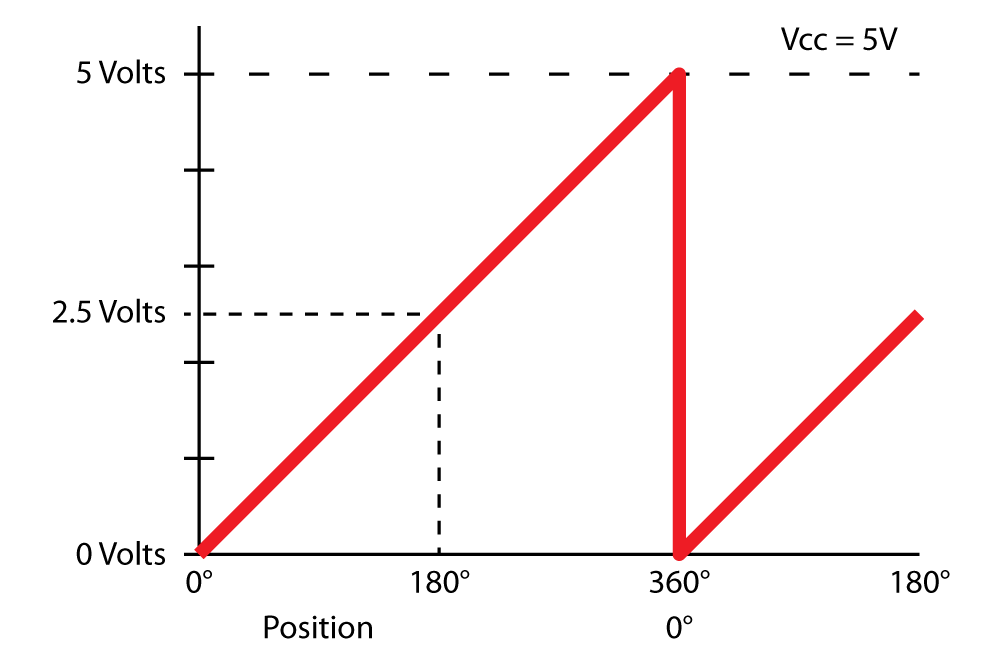
* Từ đó người dùng có thể suy ra bản thân cần một động cơ cung cấp 120 xung cho một vòng quay, nếu người dùng chỉ có động cơ 20-40 xung thì cần được lắp đặt thêm bánh răng giảm tốc với tỷ lệ 60:1 hay 30:1.

Trong trường hợp người dùng có một bộ giải mã động cơ với độ chính xác rất cao như cũng tại ra số lượng xung rất lớn cũng như cần được quay nhanh và chính xác.

Để giải quyết vấn đề trên thì nhà sản xuất có thể biến dạng dữ liệu vị trí quay của động cơ ở dạng analog, với các giá trị hiêu điện thế có thể hiện vị trí của động cơ từ đó người dùng có dùng giá trị hiệu thế để điều khiển động cơ đến vị trí chính xác.

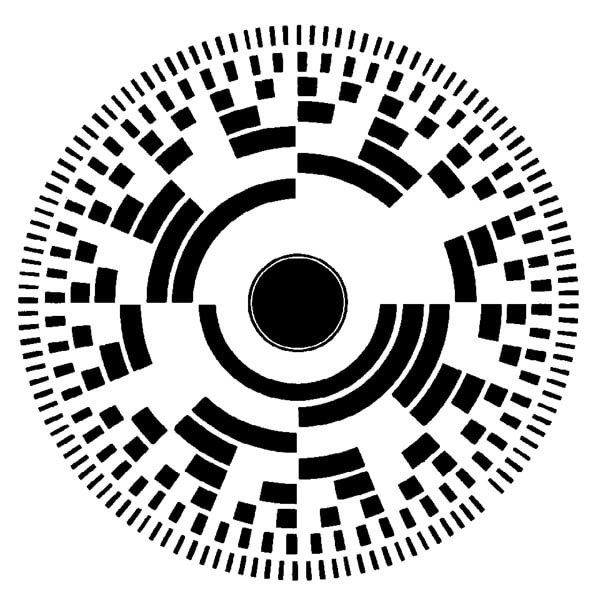
|  |  |
| --- | --- |
|  | M: số vị trí trên đĩa quay  P: khoảng cách giữa các vị trí ( ⁰ , radian )  X: Số bit cần thiết để mã hóa dữ liệu theo hệ nhị phân |

VD: Nếu người dùng cần động cơ có thế chuyển động chính xác **4000** vị trí.

* Khoảng cách của các vị trí này **0.09⁰** , cũng như để có thế biểu thị giá trị thì người dùng cần có **12 bit** để mã hóa được dữ liệu **ADC** này.

##### Hình 2.x : Mã hóa góc quay của motor với hiện điện thế

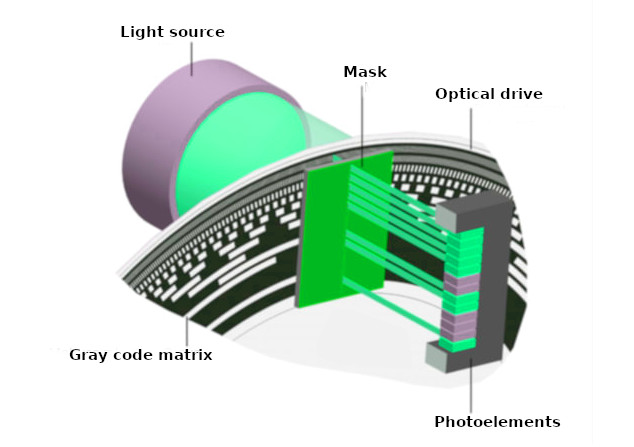
Để có được dữ liệu vị trí chính xác thì nhà sản xuất sẽ thực hiện mã hóa các vòng dữ liệu dựa trên các rảnh của đĩa che với số vòng có rảnh là số bit hay độ phân giải của encoder. Mỗi rảnh được đặt lệch một góc 90 so với rảnh đầu tiên.



##### Hình 2.x.: Đĩa xoay của động cơ có encoder với độ phân giải 8-Bit

##### Hình 2.x. : Mã hóa dữ liệu bằng PWM

Những với việc hệ thống mã hóa vị trí rotor thì số lượng dây cần rất lớn để điều khiển được động cơ này, động cơ 8-bit mã hóa cần it nhất 12 chân để động cơ hoạt động ( 8 chân cho của vdk cho dữ liệu vị trí của rotor, 2 chân cung cấp nguồn cho bộ mã hóa, 2 chân để cơ động cơ hoạt động ) để tiện cho việc giao tiếp với vi điều khiển giao tiếp thì nhà sản xuất có thế đùng các bộ chuyển đổi dữ liệu sang dạng sóng răng cưa hay dạng xung pwm nhưng người dùng cần đọc datasheet.



##### Hình 2.x : Sơ đồ hoạt động của encoder với nhiều bit dữ liệu

Các ánh sáng cần được cho qua một bộ lọc để chóng lóa để ngăn việc nhiễu từ đó có đảm bảo độ chính xác cho dữ liệu mã hóa vị trí của rotor. Đồng thời mã hóa dữ liệu theo 2 cách:

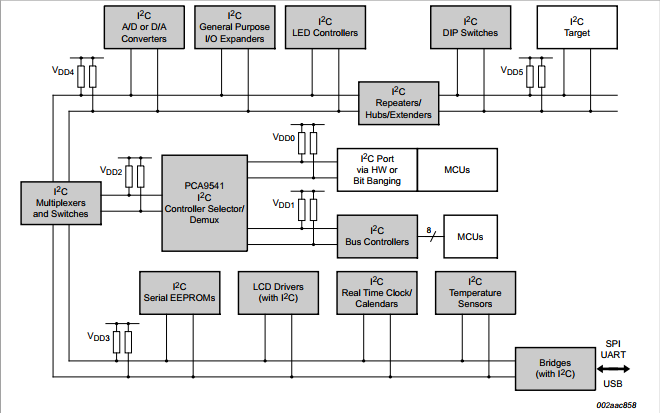
+ Mã hóa theo dữ liệu **DAC** với dữ liệu bit có thể biến đổi thành giá trị hiệu điện thế như một biến trở. Với ưu điểm có thế đưa thông tin vị trí nhanh nhưng số liệu dễ bị nhiễu nên cần vdk có thể xử lý cũng như phần cứng tương ứng cho việc xử lý dữ liệu analog và điều chỉnh cho phù hợp.

+ Mã hóa dữ liệu vị trí thông qua xung **PWM** với việc dữ liệu vị trí sẽ là biến thời gian cho độ rộng của xung pwm với việc mã hóa cho tính hiệu 10-bit thì người dùng chỉ cần một chu kỳ tính hiệu **1024us** ( **210** ) với độ rộng xung tối đa là 1023us. Ưu điểm thiết kế và quá trình xử lý của vi điều khiển đơn giản chỉ cần một bộ đếm thời gian cho vi điều khiển cũng như hỗ trợ interupt, nhưng nhược điểm dữ liệu vị trí cần một khoản thời gian thì vdk mới có thể xác định được, độ phân giải càng cao thì thời gian chờ có thế lâu hơn.

## 2.y. TỔNG QUAN VỀ GIAO THỨC I2C

### 2.y.1. Khái niệm về giao thức I2C

##### Hình 2.y : Sơ đồ kết nối chân cơ bản của giao thức I2C

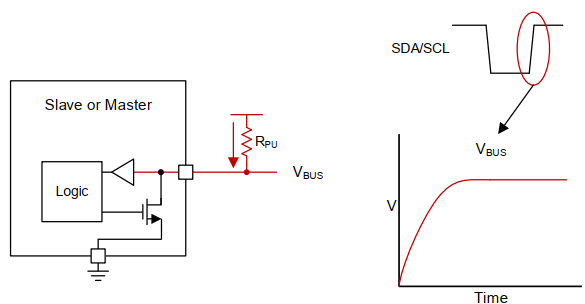
 Giao thức **I2C** ( **Inter-Integrated Circuit** ) một loại giao thức truyền dữ liệu đơn giản được phát triển bởi Philips Semiconductors vào năm 1982 với thiết kế đơn giản của nó chỉ dùng 2 dây cho việc giao tiếp và truyền dữ liệu theo hình thức **Master-Slave** thứ là tại một thời điểm chỉ có một thiết bị thực hiện truyền dữ liệu còn các thiết bị khác sẽ trong chế độ lắng nghe hoặc bỏ qua dữ liệu.

##### Hình 2.y : Sơ đồ kết nối I2C

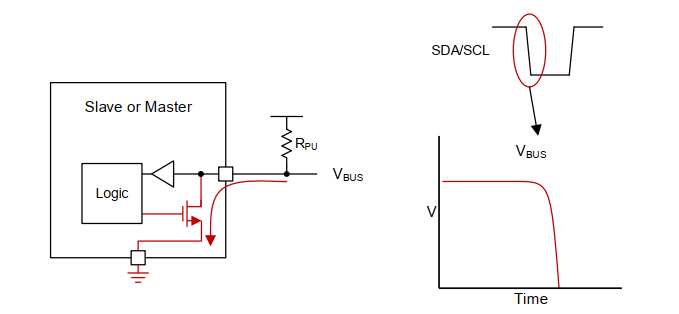
+ Dây truyền dữ liệu chính của giao thức I2c chính là **SDA**, dữ liệu được truyền theo chuỗi 8 bit và khi nhận được dữ liệu thì **Slave** sẽ thực hiện tín hiệu **ACK** để cho master biết rằng là dữ liệu đã nhận đủ và thực hiện tiếp thông số dữ liệu chuỗi tiếp theo. Nhưng dữ liệu truyền lại không có thêm bất kỳ thông tin gì nên người dùng có thể thực hiện chỉnh sửa thêm các tính năng cho chuỗi dữ liệu này. Khi việc truyền dữ liệu bị lỗi do không nhận đủ **1 byte** hay nếu người dùng thêm dữ liệu kiểm tra thì slave có thể gửi tín hiện **NACK** để cho master biết và thực hiện truyền lại dữ liệu.

+ Dây **SCL** có chức năng thực hiện truyền tần số xung cho các thiết bị trên đường truyền từ đó quyết định tốc độ truyền của dữ liệu giữa **Master** và **Slave**. Chuẩn **I2C** hỗ trợ các tốc độ như : ***100Kbits/s*** ( ***Standard-mode*** \_ Sm ), ***400Kbits/s*** ( ***Fast-mode*** \_ Fm ), ***1Mbit/s*** ( ***Fast-mode Plus*** \_ Fm+), ***3.4Mbits/s*** ( ***High-speed mode*** \_ Hs-mode ).

##### Hình 2.y : Sơ đồ điện trở kéo lên cho giao thức I2C

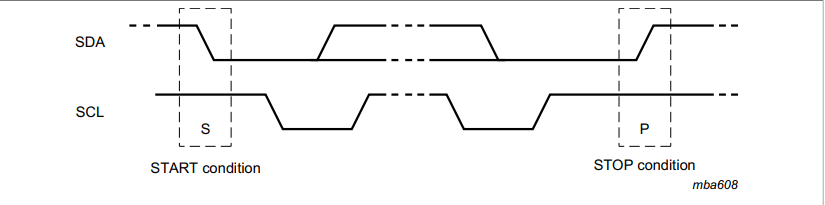
Với sự đơn giản của giao thức truyền tín hiệu thì giao thức **I2C** có thể dẫn đến bị nhiễu tín hiệu đường truyền hoặc do thiết kế đường tín hiệu hẹp do kích thước broad nhỏ. Dẫn đến việc thời gian cho tín hiệu đạt được mức logic 1 sẽ mất thời gian lâu hơn, để khắc phục điều này thì người dùng cần thiết kế hoặc mắc thêm 2 điện trở kéo lên để đó giảm nhiễu và giúp cho **bus** dữ liệu ổn định hơn.

##### Hình 2.y: Sơ đồ điện áp kéo lên cho Bus I2C

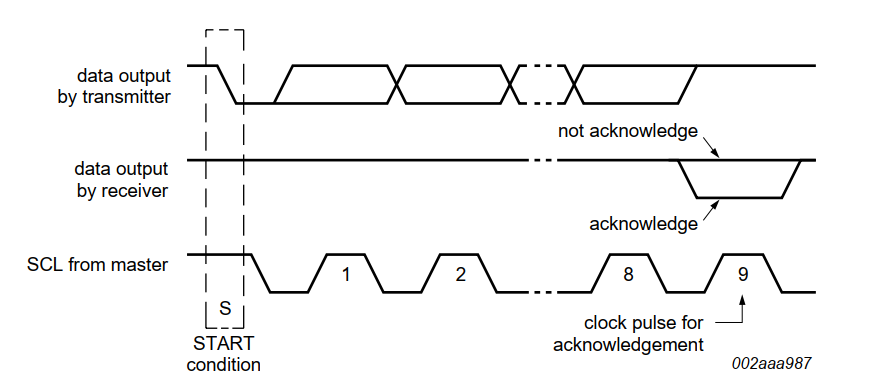
Thông qua việc thiết kế thêm điện trở kéo lên cho đường bus dữ liệu lúc này chúng ra có thế tối giản thiết kế cho giao thức **I2C** bằng việc dùng **transitor** hay **fet npn** ( **N-Chanel** ) từ đó dẫn đến việc xác định bit logic đơn giản chỉ là việc tại thời điểm của tín hiệu xung Clock thì lúc này giá trị trên bus dữ liệu **SDA** có thể được quyết định bởi hiệu điện thế.

##### Hình 2.y : Thực hiện kéo đường Bus dữ liệu xuống mức 0

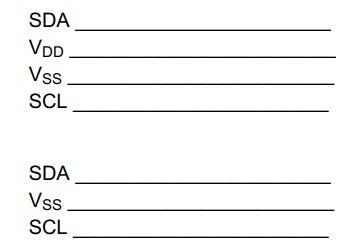
Giá trị điện trở thường là **1.2KΩ** đến **10KΩ** tùy thuộc và số lượng thiết bị đang giao tiếp trên đường **bus** dữ liệu và những với giá trị điện trở càng cao thì tốc độ kéo đường dữ liệu sẽ chậm hơn để cải thiện điều này thì người dùng có thế gắn thêm tụ điện tùy theo giá trị điện trở và thời gian mà người dùng muốn thời gian kéo lên của điện trở mà giá trị tụ điện giao động từ **10pF** đến **400pF**.



##### Hình 2.y: Điều kiện khởi động và kết thúc việc truyền dữ liệu

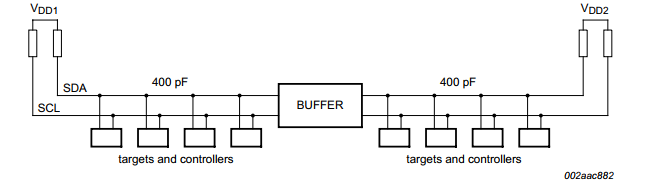
 Để hạn chế các tiến trình cần chạy trong vi điều khiển thì giao thức cho chuẩn giao tiếp **I2C** cần được thiết kế với ngoại vi hỗ trợ **interupt** phần cứng để từ đó khi dây **bus** dữ liệu sẽ luôn được trống và người dùng có thế dùng cổng trống cho nhiều ứng dụng khác. Dấu hiệu nhận biết cho việc có thiết bị cần truyền hay nhận thì lúc bất giờ **Master** bắt đầu khởi tạo tín hiệu xung **Clock** và chờ khi tín hiệu xung đang ở mức cao thực hiện truyền tín hiệu **start** đường **bus** dữ liệu xuống từ đó từ đó **master** này sẽ chiếm toàn bộ quyền xử dụng **bus** dữ liệu và bất kỳ master nào cần xử dụng đường dữ liệu này thì cần phải đợi master đang chiếm dây gửi tín hiệu **stop** thì mới có thể thực hiện truyền dữ liệu.

##### Hình 2.y: Tính hiệu ACK và NACK trong giao thức I2C

 Trong quá trình nhận dữ liệu thì slave sẽ phản hồi bằn tín hiệu ACK hay NACK để cho master biết rằng bản thân có nhận đủ 8-bit dữ liệu không, phần phát hiện lỗi thì người dùng phải tự thiết kế riêng và ứng dụng trong giao thức. Xung ACK là khi BUS SDA đang ở mức cao và có xung clock, xung NACK khi không nhận đủ 8-Bit thì slave sẽ kéo đường tín hiệu xuống và chờ tín hiệu xung clock. Nhưng phần lỗi đường truyền có thế được giảm thiểu thông qua điện trở kéo lên cũng nhưng người dùng có thế thiết kế thêm các dây nguồn đi kèm để giảm nhiễu.

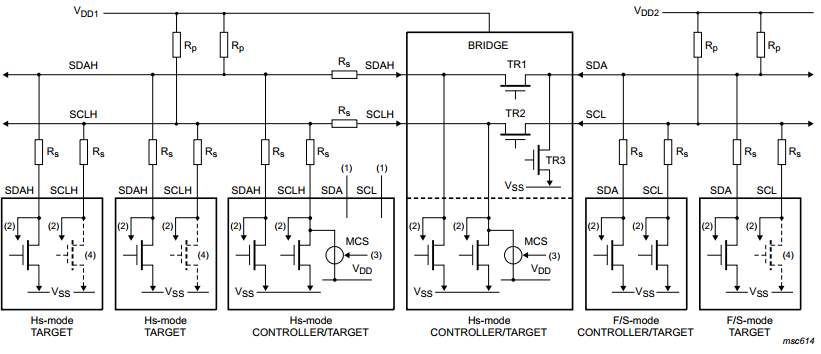
##### Hình 2.y. : Sơ đồ đường tín hiệu giúp giảm nhiễu cho I2C

Vss thường được nối đất với thông toán bài toán từ trường thì với sự thay đổi nhanh của dòng điện của chân **SDA** và **SCL** thì cả hai sẽ tạo ra các giá trị điện áp theo chiều ngược lại, nhưng với việc đặt vùng mặt đồng nối với **GND** thì các giá trị dòng điện phản hồi ngược chiều sẽ được vùng nối đất hấp thụ và giúp cho dây **BUS** dữ liệu ổn định hơn.



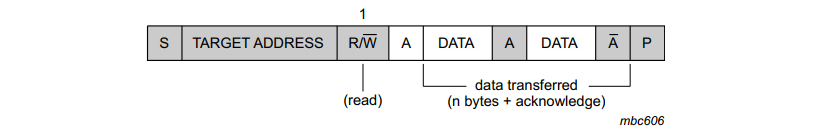
##### Hình 2.y: Sơ đồ kết nối I2C với 2 hiệu điện thế khác nhau

Hiện nay với một lượng lớn vdk trên thị trường với các mức điện áp hoạt động khác nhau **1.8V**, **3.3V**, **5V** nên việc giao tiếp giữa 2 loại vi điều khiển sẽ xảy ra thường xuyên. Để không phải thiết kế các các giao thức hoạt động phức tạp người dùng có thế dùng các **IC buffer** đồng thời cũng có thể đóng vai trò như **MUX** cho phép một **Master** giao tiếp với nhiều kênh **I2C** và mỗi kênh có hiệu điện thế khác nhau cũng như nhiều ngoại vi với cùng địa chỉ.



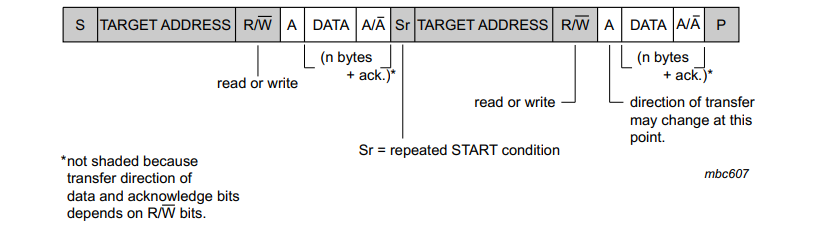
##### Hình 2.y: Sơ đồ kết nối I2C Buffer với điện trở bảo vệ chống quá dòng

Với việc kết nối nhiều kênh với mức điện áp khác nhau sẽ yêu cầu thêm nhiều bit dữ liệu cho đường địa chỉ của buffer ( **8-Bit**) cũng dữ liệu chọn kênh ( **8-Bit** ) cho master nếu còn bất kỳ còn những thông tin gì khác thì người dùng cần thiết kế thêm nhiều dữ liệu hơn, để giảm thời gian ghi và đọc dữ liệu thì người dùng phải tăng tốc độ truyền dữ liệu như cũng từ đó tạo ra dòng đánh ngược, để bảo vệ các thiết bị ngoại vị trên mạch thì người dùng có thể dùng điện trở nối với bus **I2C** có giá trị giao động từ **100Ω** đến **300Ω**, do thiết kế mạch đọc chân dữ liệu đọc mức logic theo xung **clock**.



##### Hình 2.y: Cấu trúc truyền – nhận dữ liệu I2C

Với việc chiếm đường dữ liệu của master thì khi thực hiện thiết lập kết nối thì nếu master không gửi tín hiệu dừng thì nay cả master cũng không thế giao tiếp với các thiết bị khác, để có thế giao tiếp với thiết bị ngoại vị khác hay thay đổi chế độ thì người dùng cần phải truyền tính hiệu stop hoặc tín hiệu start để thay đối địa chỉ Slave hay đổi chế độ đọc ghi.

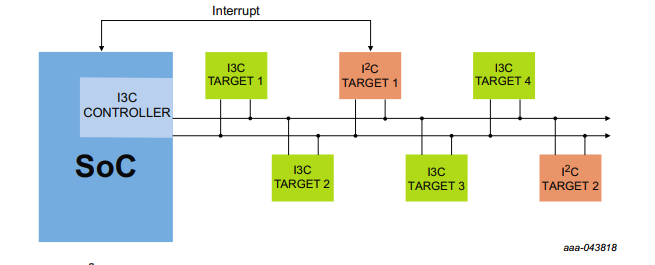


##### Hình 2.y: Sơ đồ chuỗi dữ liệu I2C cho nhiều ứng dụng

Với việc gửi lại tín hiệu Start thì master có thể thay đổi slave hoặc nếu như slave có nhiều chế độ thì master có thể thay đổi giữa chế độ đọc ghi giữa một hoặc nhiều slave thông qua chuỗi dữ liệu.

### 2.y.2. Ứng dụng

##### Hình 2.y: Ứng dụng kết nối của I2C

 Với việc I2C có thể đóng vai trò như một bộ mở rộng ngoại vi với khả năng dọc và ghi dữ liệu trên các chân bất kỳ và thiết kế đơn giản từ dó khiến cho ứng dụng sử dụng giao thức **I2C** được ứng dụng rất nhiều trong trong việc thiết kế cảm biến, mà mình hiển thị với tần số quét thấp cũng như các bộ chuyển đổi tín hiệu **PWM**, **ADC**, đồng thời bằng việc kế thêm bộ **trasitor** nhận biết giá trị dòng điện cho các ngõ vào của IC nhà sản suất có thêm tính năng **interupt** từ đó cho phép người dùng áp dụng các tính năng có thời gian đọc dữ liệu không cố định.

##### Hình 2.y: Sơ đồ kết nối I2C với interupt I3C

# CHƯƠNG 3 THIẾT KẾ XE TỰ HÀNH

## 3.1. Giới thiệu về xe

Xe tự hành một trong những lĩnh vực đang phát triển với tốc độ chóng mặt nhưng hiện nay và một trong những yếu tố giúp cho bộ xử lý bên trong xe tự hành có thế thu thập dữ liệu cho việc học hỏi của model A.I bên trong xe.

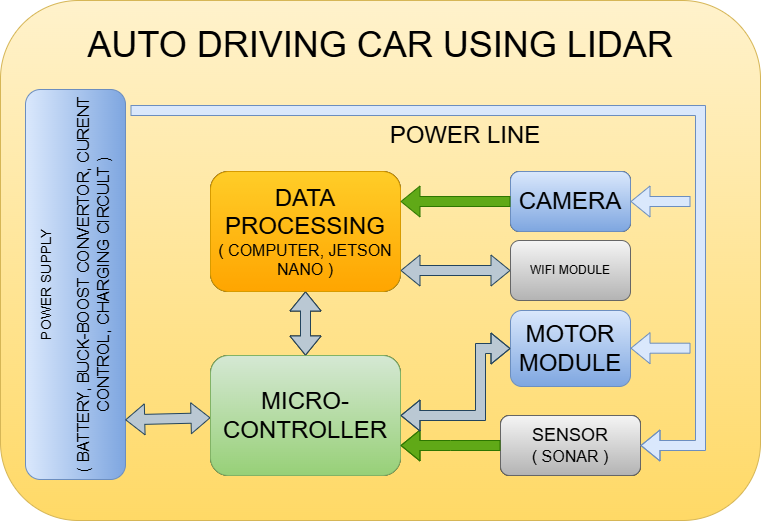
Nhóm em sẽ tập trung vào thiết kế mô hình có khả năng:

+ Vẻ bản đồ 2D và phát họa bản đồ 3D.

+ Tự di chuyển đến khi bao gọn không gian kép kín.

+ Tự động nén vật cản.

+ Có thể tự điều chỉnh tốc độ của motor.

+ Dùng camera nhận diện vật cản, thực hiện né vật cản.

##### Hình 3.1: Sơ đồ khối cấu tạo của xe

Xe sẽ được thiết kế theo các khối module để khi thực hiện thay đổi bất kỳ bộ phận nào thì nhóm có chỉ cần cập nhật lại phần mềm cho hệ thống là xe có thể hoạt động bình thường. Xe được cấu tạo từ 4 khối chính:

+ Khối nguồn: được thiết kế bằng pin litium với mạch buck-boost giúp cố định dòng điện ở một giá trị hiệu điện thế ổn định trong suốt quá trình hoạt động. Cung cấp 2 giá trị điện áp riêng biệt +5V cho vi điều khiển và khối xử lý (máy tính nhúng raserry hoặc jetson nano) đến 6-12V cho động cơ DC hoạt động

+ Khối vi điều khiển (vi điều khiển ATMEGA328P) có chức năng thực hiện thu thập số liệu, từ khối xử lý vận tốc của động cơ, thực hiện đo đạt giá trị dung lượng pin còn sót lại và thực hiện điều chỉnh cần thiết để đảm bảo tốc độ di chuyển của xe không thay đổi trong suất quá trình hoạt động.

+ Khối xử lý dữ liệu: thu thập dữ liệu từ cảm biến lidar với hình ảnh từ camera để từ đó có thể vẻ một bản đồ 2D, cũng như thực hiện phát họa bản đồ 3D của những khung vực đã đi qua, xác định vị trí của mô hình xe trong bản đồ, tính toán ra được tốc độ và hướng đi của động cơ.

+ Module: động cơ xe lúc này được đóng gói tích hợp với diver điều khiển chiều xoay của động cơ, được tích hợp với IC mở rộng GPIO và PWM giao tiếp I2C từ đó cho phép người dùng có thể điều khiển thông qua BUS dữ liệu.

## 3.2. Sơ đồ khối điều khiển động cơ

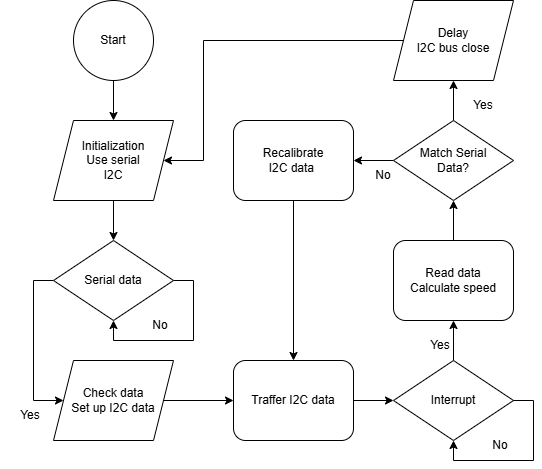
##### Hình 3.2: Sơ đồ truyền nhận dữ liệu khối xử lý và khối điều khiển

Để tiện cho việc phát triển lập trình phần cứng và thực hiện nâng cấp sửa chữa thì nhóm em đã thực hiện tách bộ phận xử lý thành 2 khối:

+ Khối xử lý: tập chung vào việc xử lý các tác vụ phức tạp như render hình ảnh điểm tọa độ vẽ bản đồ, tính toán tọa độ của xe tại một thời điểm, tính toán tốc độ vị trí hướng đi của xe để né vật cản.

+ Khối điều khiển: thực hiện điều khiển động cơ, điều chỉnh chiều xoay của động cơ theo hướng xoay, thực hiện thu thập dữ liệu của động cơ để xác định tốc độ, thu thập thêm bất kỳ dữ liệu cảm biến đơn giản như siêu âm, cảm biến hồng ngại…

+ Khối điều khiển cũng đóng vai trờ như một bộ tiền xử lý giúp tính toán xử lý dữ liệu thô thông số tốc độ của motor và thực hiện các điều chỉnh để cho động cơ chạy đúng theo như yêu cầu của khối xử lý.



##### Hình 3.3: Sơ đồ khối truyền và xử lý dữ liệu motor của khối điều khiển

## 3.3. Thiết kế phần cứng

### 3.3.1. Nguồn

##### Hình 3.4: Sơ đồ khối các tính năng bên trong khối pin

Khối pin sẽ bao gồm 4 chức năng chính:

+ Chức năng quản lý pin: dòng IC với với khả năng thực hiện xử lý dữ liệu của pin ( điện thế, cường độ dòng điện ) từ đó sẽ kích hoạt các tính năng cần thiết như bảo vệ pin khi tải hoặc điện áp đầu vào quán lớn thì mạch sẽ kích hoạt tính năng bảo vệ pin thực hiện ngắt nguồn pin.

+ Thực hiện giám sát pin: sử dụng IC hay mạch có chức năng thực thu thập dữ liệu dung dung lượng điện áp ra của pin chuẩn đổi những dữ liệu này sang dạng số rồi thực hiện truyền đến vi điều khiển thông qua các giao thức như I2C, SPI, UART, … cũng như cho phép vdk thực hiện điều chỉnh khi cần.

+ Cân bằng pin: đây là mạch thụ động chủ yếu cân bằng pin trong suất quá trình sử dụng sạc vì pin sẽ bị chai không đồng đều dẫn đến hiện tượng ngắn mạch dẫn đến pin không thế dùng được nữa nếu như không được cân bằng đúng.

### 3.3.2. Khối điều khiển động cơ

##### Hình 3.5: Sơ đồ các lớp điều khiển cho động cơ

Việc điều khiển động cơ có thể được chia thành các chức năng chính:

+ ***Mạch quản lý nguồn động cơ***: có chức năng cung cấp một điện áp ổn định với động cơ hoạt động lâu dài.

+ ***Motor driver***: với việc động cơ thường yêu cầu công suất tải rất lớn khiến cho vi điều khiển khi tiếp xúc với động cơ thường bị dòng đánh ngược do quá trình quay của động cơ làm cháy mạch, IC driver dùng để tạo một lớp bảo vệ đồng thời dùng tín hiệu từ vi điều khiển để thực hiện thay đổi chiều quay và tốc độ cho động cơ.

+ ***Kiểm soát động cơ vòng mở***: bằng cách sử dụng động cơ với một encoder một bộ giải mã vị trí hay tốc độ quay của động cơ.

+ ***Kiểm soát động cơ vòng đóng***: vi điều khiển có thể dùng số liệu tốc độ của động cơ ( số xung trong 1s ) từ dó có thế thay đổi thông tin tốc độ của động cơ nhanh hơn hay chậm hơn theo nhu cầu của người dùng.

+ ***Thiết lập một kênh giao tiếp***: Cho phép vi điều khiển có thế điều khiển động cơ như một module với chuẩn giao tiếp như SPI hoặc I2C.

+ ***Cổng giao tiếp cho vi điều khiển***: thực hiện viết một chương trình cho phép người dùng hay các chương trình khác thực hiện giao tiếp với vi điều khiển cho phép vi điều khiển thực hiện thay đổi tốc độ chiều quay của từng bánh xe.

### 3.3.3. Cơ chế điều khiển

##### Hình 3.6: Schematic của motor driver L293D

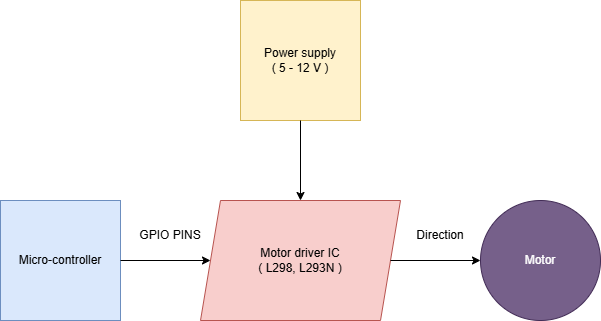
IC L293D được thiết kế như một bộ buffer cho các tín hiệu của vi điều khiển bằng việc sửa dụng CMOS từ đó cho phép IC có thể cung cấp nguồn 0.6A ổn định cho bất kỳ động cơ nào với giá trị hiệu điện thế từ 4.5V đến 36V. Với thiết kế CMOS có cực máng mở cho phép IC không chỉ cung nguồn cũng như thực hiện điều hướng dòng ra của động cơ DC.

Chức năng của các chân IC:

+ Chân IN1-IN4 có tính năng là ngõ vào cho phép người dùng đều khiển động cơ thông qua mức logic, với mức giá trị mức 0 thì ngõ ra tương ứng sẽ có tự động thành một công tắc nối đất từ đó giúp cho điện áp ngõ ra là 0V.

+ Chân EN1-EN2 cho phép driver thực hiện nhận dữ liệu ngõ vào theo cặp EN1 điều khiển chân IN1 – IN2 và EN2 điều khiển IN3 – IN4, với mức logic 0 thì ngõ nào chân IN1 và IN2 sẽ không còn có khả năng điều khiển được động cơ nứa lúc này động cơ có chiều quay giống như chiều quay lúc chân EN1 có mức logic 1 trước đó.

##### Hình 3.7: Sơ đồ logic của IC 293D

a. Điều khiển trực tiếp

##### Hình 3.8: Sơ đồ khối điều khiển động cơ trực tiếp

Để thực hiện điều khiển trực tiếp người dùng phải dùng dùng jumper để nối chân EN1 với EN2 với một nguồn 5V. Từ đó giúp cho IC hoạt động bằng cách nhận ngõ vào thông qua chân IN1-IN4.

Lúc này vi điều khiển có thế dùng các cặp IN1-IN2 và IN3-IN4 để thực hiện điều khiển 2 động cơ, thông qua việc nối chân dương (+) của động cơ vào cổng OUT1 hoặc OUT3, chân âm (-) được nối với ngõ ra OUT3 hoặc OUT4 với 4 chế độ khác nhau thông qua 2 bit điều khiển:

+ 0x0: Động cơ ngắt hoàn toàn do không có bất kỳ gí trị điện áp nào vào động cơ.

+ 0x1: Động cơ quay ngược chiều kim đồng hồ.

+ 0x2: Động cơ quay theo chiều kim đồng hồ.

+ 0x3: Lúc này với cùng một giá trị điện áp khiến cho động cơ và phần lõi mất đi hết từ tính từ đay giúp cho động cơ dừng hòa toàn.

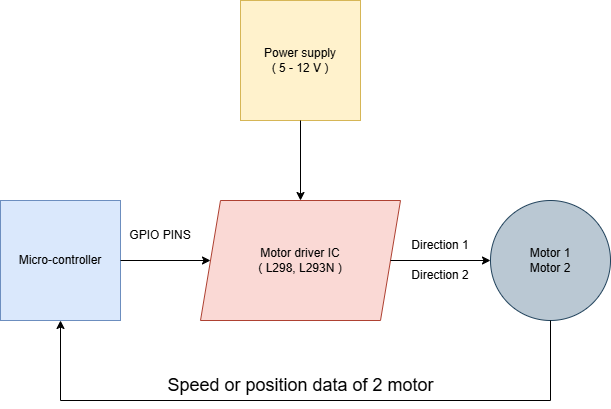
Ưu điểm:

+ Thiết kế đơn giản.

+ Có thế thay đổi chế độ quay với tốc độ cao.

+ Một driver có thể điều khiển 2 động cơ

Nhược điểm: Người dùng không thể biết được tốc độ động cơ, nên phụ thuộc nhiều chế độ dừng khẩn cấp rất nhiều, làm giảm độ bền của động cơ.



##### Hình 3.10: Sơ đồ khối cho điều khiển 2 động cơ trực tiếp

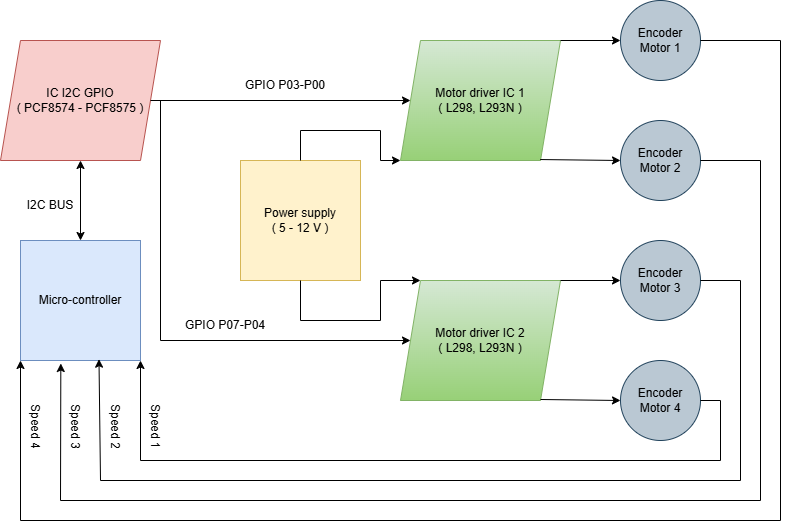
Với việc thêm chân mã hóa kênh A và kênh B người dúng có thể xác định được chiều quay cả động cơ thông qua việc xét xung interrupt tại thời điểm t. Interrupt bị kích hoạt thông qua xung đi lên của tín hiệu A, từ dó mỗi lần vi điều khiển phát hiện interrupt thì sẽ kiểm tra coi xung tín hiệu B đang ở mức cao (quay theo chiều kim đồng hồ) hay mức thấp (quay ngược chiều kim đồng hồ).

Ưu điểm: Người dùng điều chỉnh được chiều quay trong quá trình chạy.

Nhược điểm:

+ Tuy có thế điều chỉnh được tốc độ quay của rotor bằng pwm nhưng yêu cầu vi điều khiển phải có thiết kế thêm timer interrupt tăng độ phức tạp cho phần mềm.

+ Thiết này xử dụng nhiều chân của vi điều khiển, một motor dùng đến 4 chân của vdk.

b. Điều khiến gián tiếp thông qua PCF8574

##### Hình 3.11: Sơ đồ khối điều khiển động cơ bằng PCF8574 và I2C

Bằng việc xử dụng IC pcf8574 thì người dùng có thế điều khiển chân IN của ic l2893 thông qua giao thức I2C.

Ưu điểm:

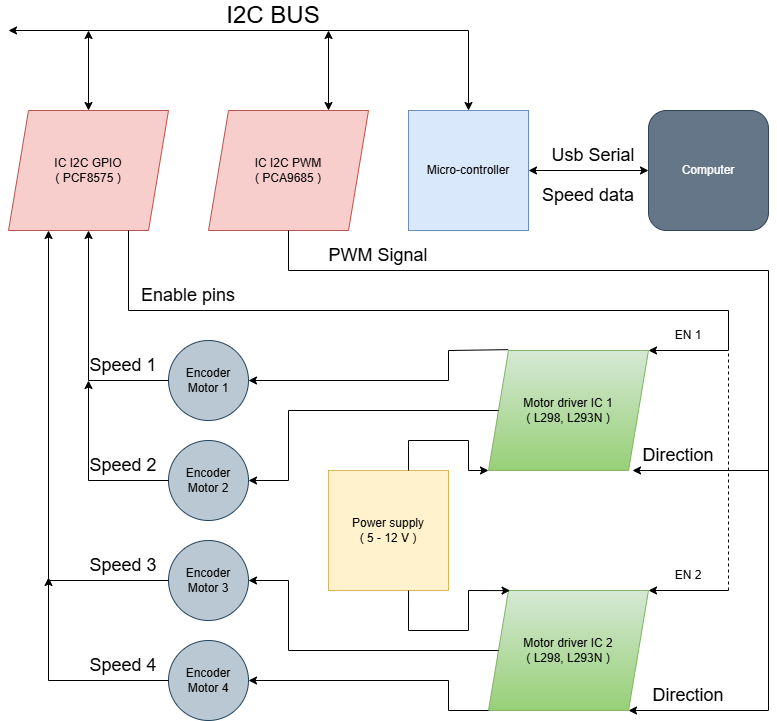
+ Giảm thiểu được số lượng chân IO cần dùng xuống 1 nữa.

+ Giúp tinh gọn mà module khóa động cơ, cho phép điều khiển qua I2C.

Nhược điểm:

+ Làm tăng độ phức tạp của thiết kế với nhiều IC.

+ Làm cho việc điều khiển bằng PWM khó khăn năng hơn do phải tín toán chung với thời gian truyền I2C.

c. Điều khiển gián tiếp thông qua PCA9685 và thu thập dữ liệu bằng PCF8575

##### Hình 3.12: Sơ đồ khối động cơ encoder bằng IC PWM PCA9685 và PCF8574

Thông qua IC PCA9685 người dùng có thế điều khiển động cơ với mức công suất chạy tùy ý thông qua việc bật tắc động cơ nhanh trong 1 giây.

Ưu điểm:

+ Tất cả động cơ mà xe cần có thế được điều khiển thông qua I2C.

+ Giảm thiểu số chân cần giao tiếp xuống 2 chân.

Nhược điểm:

+ Thiết kế rất phức tạp phần cứng lần phần mềm.

+ IC PCF8575 không hỗ trợ đọc mức logic của ngõ nào.

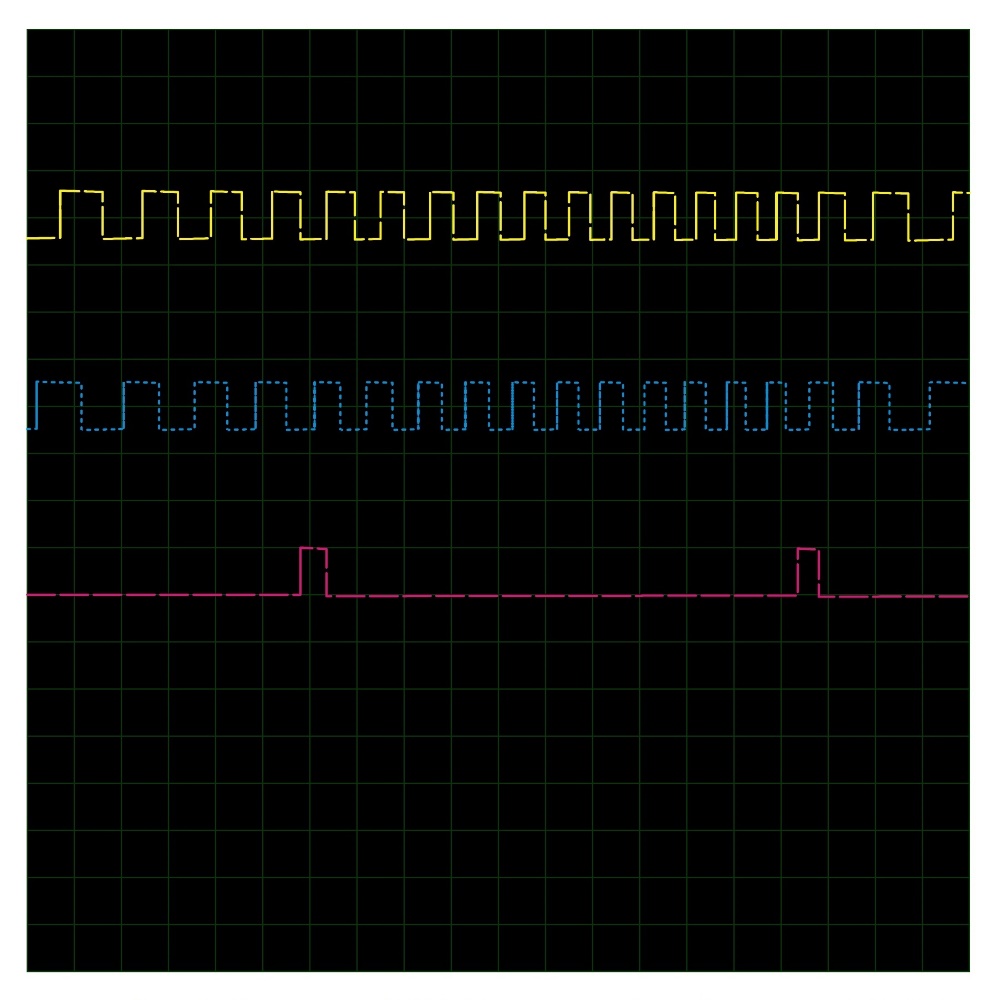
# CHƯƠNG 4 THUẬT TOÁN QUÉT 2D

# CHƯƠNG 5 KIỂM TRA VÀ ĐÁNH GIÁ

## 5.1. THỬ NGHIỆM ĐIỀU KHIỂN ĐỘNG CƠ

### 5.1.1. Mô hình điều khiển trực tiếp.

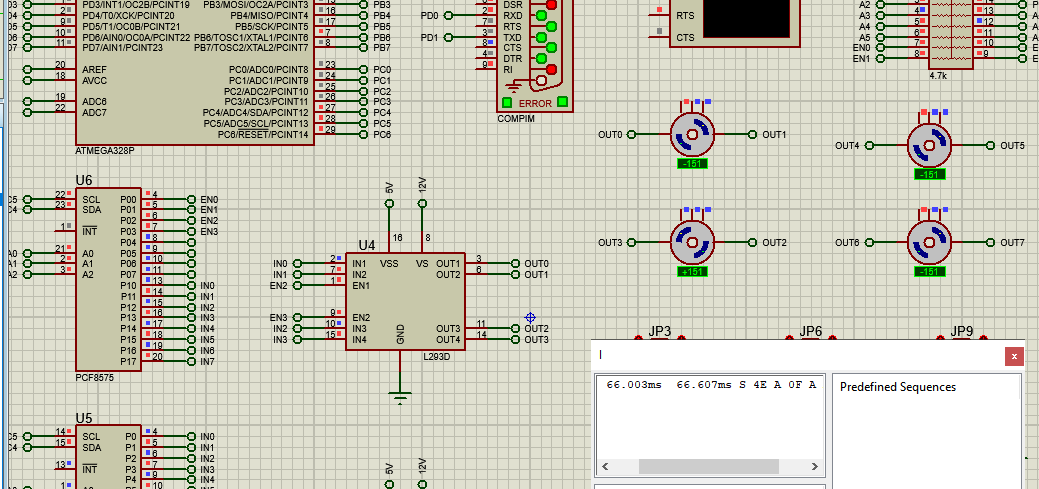
##### Hình 5.1: Sơ đồ mô phòng của motor encoder



##### Hình 5.2: Sóng xung thu được của động cơ encoder

##### Hình 5.3: Xung tín hiệu của động cơ khi quay theo chiều kim đồng hồ

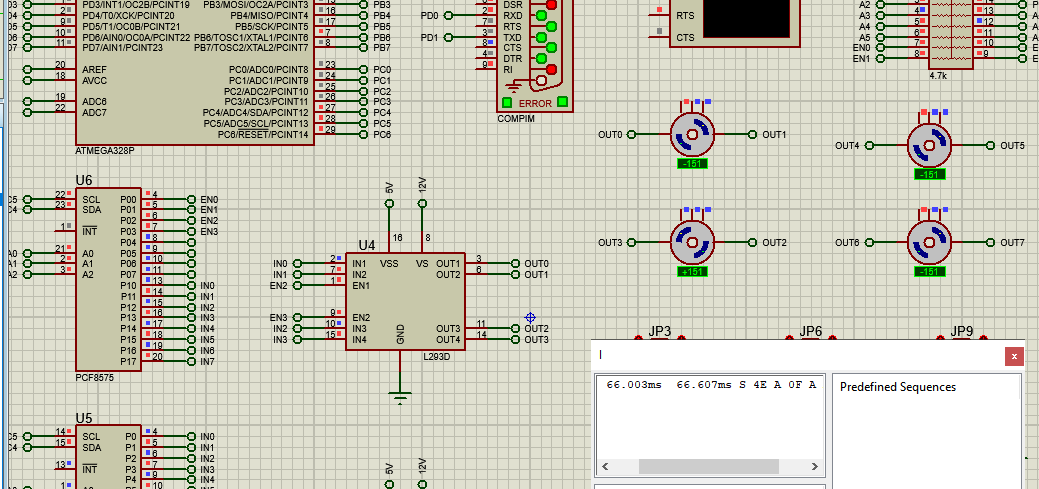
##### Hình 5.4: Xung tín hiệu của động cơ quay ngược chiều kim đồng hồ



##### Hình 5.5: Tốc độ mô phòng của motor

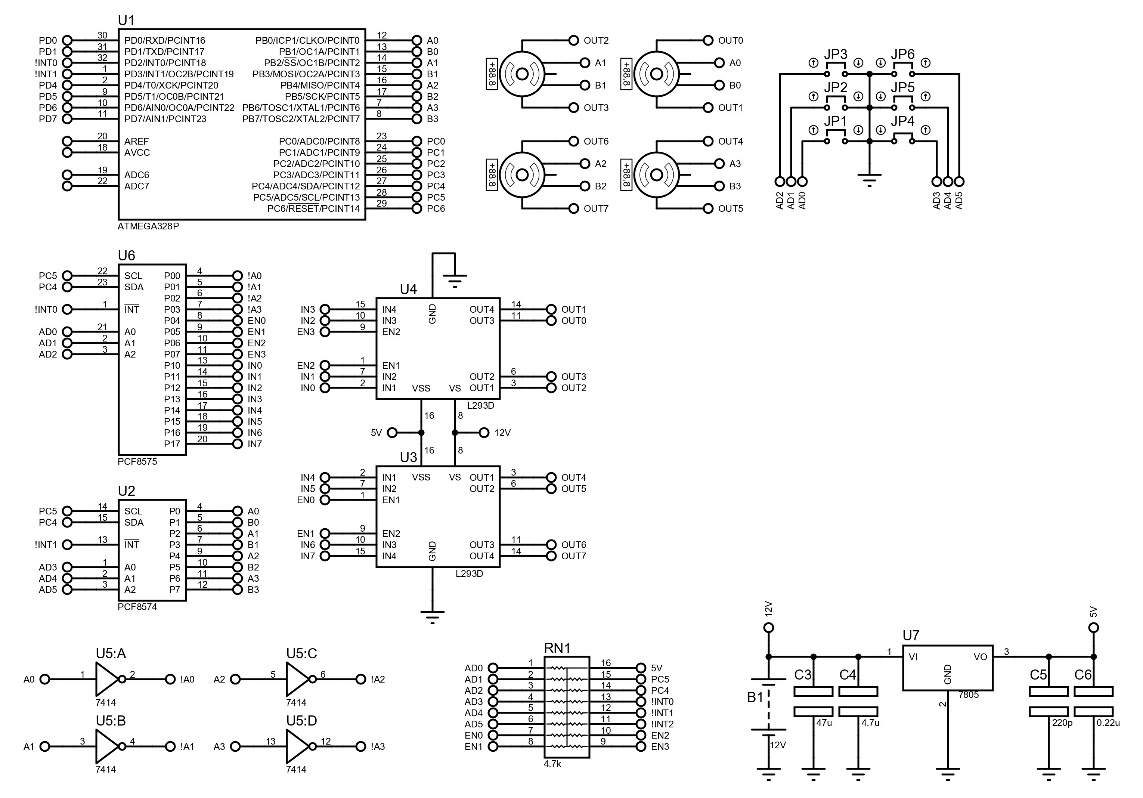
### 5.1.2. Mô hình điều khiển qua I2C.

##### Hình 5.6: Schematic thực hiện mô phòng module động cơ I2C

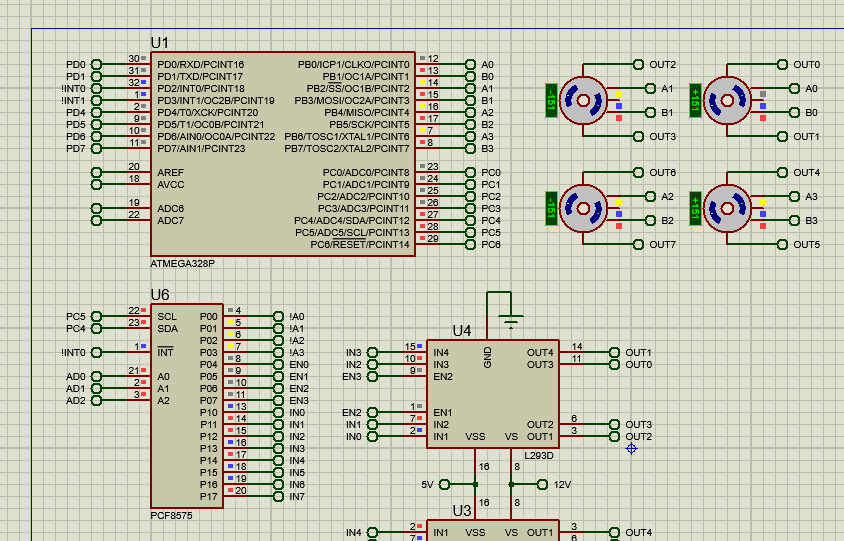


##### Hình 5.7: Ảnh điều khiển động cơ chiều bằng I2C

### 5.1.3. Mô hình điều khiển PID qua I2C.

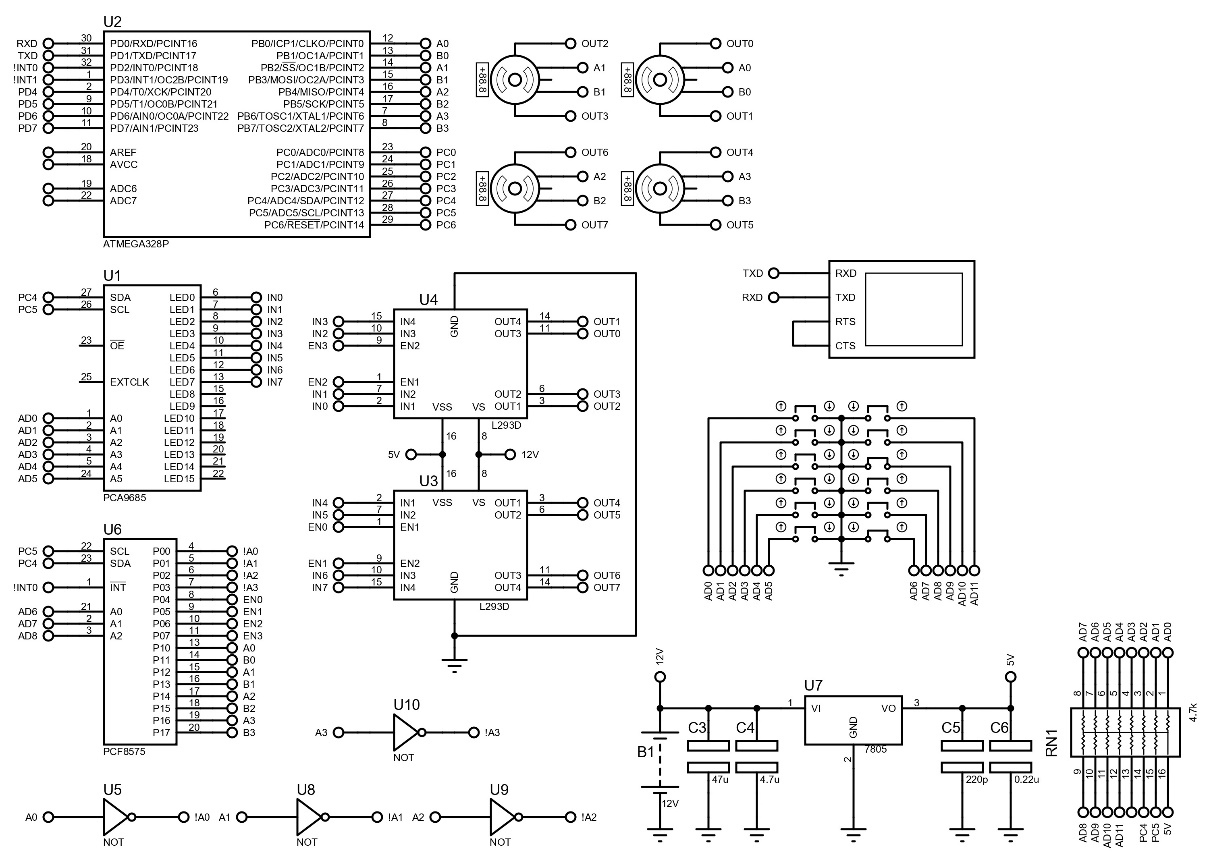


##### Hình 5.8: Schematic đọc dữ liệu xung encoder thông qua PCF8575



##### Hình 5.9: Lỗi quá trình đọc ngõ ra encoder motor

Với việc sóng tín hiệu ngõ của encoder dùng cạnh lên của xung tính hiệu để ghi dữ liệu từ đó khi tín hiệu đi qua IC cổng đảo làm delay tín hiệu ban đầu dẫn đến việc đọc ghi tín hiệu của encoder bị khó khăn hơn.



##### Hình 5.10: Schematic điều khiển động cơ bằng PCA9685 và đọc ngõ vào PCF8575

# CHƯƠNG 6 HƯỚNG PHÁT TRIỂN

# TÀI LIỆU KHAM KHẢO

1. [Chuỗi video về xe tự định hướng của MathLab](https://youtube.com/playlist?list=PLn8PRpmsu08rLRGrnF-S6TyGrmcA2X7kg&si=d0ht1fj6rm5R8wlb)
2. [Video giải thích về encoder motor](https://youtu.be/j8rG3bjqriI?si=LKc2Nz0BQLN2DarY)
3. [Wikipedia về động cơ có mã hóa](https://en.wikipedia.org/wiki/Rotary_encoder)
4. [Encoder types and mechanism by AshakiKASEI](https://www.akm.com/global/en/products/rotation-angle-sensor/tutorial/type-mechanism-1/)
5. [Absolute Encoder Interfaces by U.S Digital](https://www.usdigital.com/blog/absolute-encoder-interface-types/).
6. [Basic Knowledge of Encoder by AshakiKASEI](https://www.akm.com/global/en/products/rotation-angle-sensor/tutorial/type-mechanism-1/)
7. [Pulse Width Modulation (PWM) Background by Microchip](https://skills.microchip.com/programming-an-8-bit-pic-mcu-pulse-width-modulation-pwm-output-using-mcc/726696)
8. [Analog To Digital Conversion – Performance Criteria](https://www.electronics-lab.com/article/analog-to-digital-conversion-performance-criteria/)
9. [What Is ADC, Working Of ADC, Analog To Digital converter](https://cselectricalandelectronics.com/what-is-adc-working-of-adc-analog-to-digital-converter/)
10. [Flash ADC](https://www.allaboutcircuits.com/textbook/digital/chpt-13/flash-adc/)

# TRÍCH DẪN