

bu

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HCM

KHOA ĐÀO TẠO CHẤT LƯỢNG CAO



ĐỒ ÁN HỆ THỐNG CƠ ĐIỆN TỬ

NGHIÊN CỨU, THIẾT KẾ VÀ CHẾ TẠO MÔ HÌNH MOBILE ROBOT

GVHD: Ths. Võ Lâm Chương

SVTH

MSSV

Nguyễn Hoàng Danh

18146090

Nguyễn Tấn Duy

18146092

Đỗ Đình Long

18146157

Tp. Hồ Chí Minh, tháng 1 năm 2022

DANH SÁCH THÀNH VIÊN THAM GIA
HỌC KỲ I NĂM HỌC 2021-2022

| STT | Họ và tên | MSSV | Tỉ lệ % hoàn thành |
|-----|-------------------|----------|--------------------|
| 1 | Nguyễn Hoàng Danh | 18146090 | 100% |
| 2 | Nguyễn Tấn Duy | 18146092 | 100% |
| 3 | Đỗ Đình Long | 18146157 | 100% |

Ghi chú:

- Tỷ lệ % = 100%: Mức độ phần trăm của từng sinh viên tham gia.
- Trưởng nhóm: Nguyễn Hoàng Danh

NHẬN XÉT

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Ký tên

Ths. Võ Lâm Chương

LỜI CẢM ƠN

Để có được kết quả đồ án như ngày hôm nay là một chặng đường đầy khó khăn thử thách đối với nhóm thực hiện. Nhóm chúng em xin cảm ơn đến thầy Võ Lâm Chương đã cho tụi em những hướng đi đúng đắn, tận tình hướng dẫn và chỉ bảo cho chúng em, truyền cảm hứng và động lực để nhóm hoàn thành tốt đồ án môn học này.

Bên cạnh đó chúng em rất biết ơn đến những thầy cô tại trường đại học Sư Phạm Kỹ Thuật, đặc biệt là bộ môn cơ điện tử đã giảng dạy chúng em bằng cả tâm huyết, nhiệt tình hỗ trợ chúng em trong khoảng thời gian lên lớp và giải đáp thắc mắc đồ án của chúng em.

Cuối cùng, dù đã rất cố gắng để có thể hoàn thiện thật tốt đồ án môn học, nhưng với điều kiện thời gian cũng như kinh nghiệm vẫn còn hạn chế, nhóm chúng em vẫn không tránh khỏi những thiếu sót và sai sót. Chúng em rất mong nhận được sự góp ý và những lời nhận xét quý báu từ các thầy cô để nhóm có thể hoàn thiện và nâng cao kiến thức. Nhóm chúng em xin tiếp nhận bằng cả lòng biết ơn.

Xin chân thành cảm ơn.

Sinh viên thực hiện

Nguyễn Hoàng Danh

Nguyễn Tấn Duy

Đỗ Đình Long

LỜI NÓI ĐẦU

Thế giới đã trải qua bốn cuộc cách mạng công nghiệp vĩ đại của lịch sử con người. Và bây giờ là năm 2022 của thế kỉ XXI, con người đang sống và đóng góp từng ngày cho sự phát triển của nhân loại trong cuộc cách mạng công nghiệp lần thứ tư – “Công nghiệp 4.0”. Năm 2013, một từ khóa mới là “Công nghiệp 4.0” bắt đầu nổi lên xuất phát từ một báo cáo của chính phủ Đức đề cập đến cụm từ này nhằm nói tới chiến lược công nghệ cao, điện toán hóa ngành sản xuất mà không cần sự tham gia của con người. Đây được gọi là cuộc cách mạng số, thông qua các công nghệ như Internet vạn vật (IoT), trí tuệ nhân tạo (AI), thực tế ảo (VR), tương tác thực tại ảo (AR), mạng xã hội, điện toán đám mây, di động, phân tích dữ liệu lớn (SMAC)... để chuyển hóa toàn bộ thế giới thực thành thế giới số.

Để phục vụ cho nền công nghiệp đang phát triển vô cùng năng động và đang là xu thế trên toàn thế giới, không thể thiếu đến Robot. Robot là một loại thiết bị có thể thực hiện những công việc một cách tự động bằng sự điều khiển của máy tính hoặc các vi mạch điện tử được lập trình. Robot có một trong các đặc điểm sau đây: do con người sáng tạo ra, có khả năng nhận biết môi trường xung quanh và tương tác với những vật thể trong môi trường, có khả năng đưa ra các lựa chọn dựa trên môi trường và được điều khiển một cách tự động theo những trình tự đã được lập trình trước, có thể điều khiển được bằng các lệnh để có thể thay đổi tùy theo yêu cầu của người sử dụng, có thể di chuyển quay hoặc tịnh tiến theo một hay nhiều chiều và khéo léo trong vận động.

Trong các họ Robot, chúng ta không thể không nhắc tới Mobile Robot với những đặt thù riêng mà những loại Robot khác không có. Với khả năng di chuyển linh hoạt và vùng hoạt động rộng, thu hút nhiều sự đầu tư nghiên cứu hiện nay. Mobile Robot có thể phân chia thành nhiều loại theo cách vận hành (Robot di chuyển bằng chân, bánh xe, xích,...).

Để góp phần vào sự phát triển nền khoa học kỹ thuật của nước nhà, nhóm chúng em đã chọn “ ***Mobile Robot di chuyển đa hướng bằng bánh xe omni*** ”. Từ những suy nghĩ đó chúng em sử dụng những kiến thức còn hạn chế của mình để nghiên cứu chế tạo Mobile Robot trong phạm vi đồ án tốt nghiệp với ước muốn đóng góp vào công nghệ chế tạo Robot của nước nhà trong thời gian tới.

Nội dung cơ bản gồm 7 chương. Có các nội dung cơ bản như sau:

Chương 1: Tổng quan về đề tài

Chương 2: Tổng quan về Mobile Robot

Chương 3: Thiết kế cơ khí cho Mobile Robot

Chương 4: Xây dựng hệ thống Điện – Điều khiển

Chương 5: Xây dựng giải thuật điều khiển

Chương 6: Kết quả và thực nghiệm

Chương 7: Kết luận và hướng phát triển

Xin chân thành gửi lời cảm ơn tới thầy Võ Lâm Chương đã giúp chúng em hoàn thành đồ án này. Do kiến thức còn hạn chế nên đồ án còn nhiều sai sót. Chúng em xin nhận sự giúp đỡ của các thầy và các bạn để đồ án được hoàn thiện hơn. Chúng em xin chân thành cảm ơn.

MỤC LỤC

| | |
|--|-----------|
| LỜI NÓI ĐẦU | i |
| CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI..... | 1 |
| 1.1. Giới thiệu | 1 |
| 1.2. Lý do chọn đề tài | 3 |
| 1.3. Tính cấp thiết của đề tài..... | 4 |
| 1.4. Ý nghĩa của đề tài | 4 |
| 1.5. Mục tiêu của đề tài | 4 |
| 1.6. Phương pháp nghiên cứu | 5 |
| CHƯƠNG 2: TỔNG QUAN VỀ MOBILE ROBOT | 6 |
| 2.1. Lịch sử phát triển..... | 6 |
| 2.2. Phân loại | 8 |
| 2.3. Đặt vấn đề..... | 10 |
| 2.4. Giới hạn đề tài | 12 |
| CHƯƠNG 3: THIẾT KẾ CƠ KHÍ CHO MOBILE ROBOT | 13 |
| 3.1. Một số nghiên cứu về Omni Robot trên thế giới và Việt Nam | 13 |
| 3.2. Thi công mô hình..... | 14 |
| 3.3. Lựa chọn cơ cấu chuyển động cho Mobile Robot..... | 16 |
| 3.4. Thiết kế kỹ thuật cơ khí..... | 17 |
| 3.4.1. Mô hình động học cho robot | 17 |
| 3.4.2. Mô hình động lực học cho robot | 18 |
| 3.4.3. Mô hình cho động cơ..... | 20 |
| 3.4.4. Mô hình động học cho một số chuyển động cơ bản..... | 21 |
| 3.4.5. Tính toán lựa chọn động cơ..... | 23 |
| 3.4.6. Tính toán chọn số cảm biến..... | 24 |
| CHƯƠNG 4: XÂY DỰNG HỆ THỐNG ĐIỆN – ĐIỀU KHIỂN..... | 28 |
| 4.1. Hệ thống điều khiển Mobile Robot | 28 |
| 4.2. Hệ thống điện Mobile Robot | 29 |
| 4.2.1. Các thiết bị được sử dụng trên Mobile Robot | 29 |

| | |
|--|-----------|
| 4.2.2. An toàn điện | 44 |
| CHƯƠNG 5: XÂY DỰNG GIẢI THUẬT ĐIỀU KHIỂN | 45 |
| 5.1. Tổng quan sơ đồ điều khiển | 45 |
| 5.2. Điều khiển động cơ..... | 45 |
| 5.2.1. Lý thuyết về bộ điều khiển PID: | 46 |
| 5.2.2. Mô hình toán và chọn thông số PID..... | 52 |
| 5.3. Phát hiện vật cản..... | 54 |
| 5.4. Xử lý dữ liệu cho cảm biến gia tốc MPU6050 | 55 |
| CHƯƠNG 6: KẾT QUẢ THỰC NGHIỆM | 57 |
| 6.1. Kết quả thu được cảm biến gia tốc MPU6050 | 57 |
| 6.2. Kết quả thu được từ cảm biến khoảng cách US-015 | 57 |
| 6.3. Kiểm tra tính ổn định của bộ PID..... | 58 |
| 6.4. Thực nghiệm:..... | 59 |
| CHƯƠNG 7: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN..... | 60 |
| 7.1. Về lý thuyết | 60 |
| 7.2. Về thực nghiệm | 60 |
| 7.3. Ưu điểm, nhược điểm | 60 |
| 7.4. Hướng phát triển..... | 60 |
| TÀI LIỆU THAM KHẢO..... | 61 |
| PHỤ LỤC | 62 |

DANH MỤC CÁC TỪ VIẾT TẮT

| | |
|-------------|--|
| AGV | Automatic Guided Vehicle |
| ANSI | American National Standards Institute |
| ARM | Advanced RISC Machine |
| AVR | Automatic Voltage Regulator |
| CSS | Cascading Style Sheets |
| DMP | Data Management Platform |
| DOM | Document Object Model |
| DSP | Digital Motion Processor |
| GPS | Global Positioning System |
| HDMI | High-Definition Multimedia Interface |
| HTML | Hyper Text Markup Language |
| I2C | Inter-Integrated Circuit |
| IACS | International Association of Classification Societies |
| IMU | Inertial Measurement Unit |
| PI | Proportional Integral |
| PIC | Programmable Intelligent Computer |
| PID | Proportional Integral Derivative |
| PoE | Power over Ethernet |

MỤC LỤC HÌNH ẢNH

| | |
|--|----|
| Hình 2. 1 Military Robot..... | 9 |
| Hình 2. 2 iRobot 510 PackBot..... | 9 |
| Hình 2. 3 Robot dò line..... | 9 |
| Hình 2. 4 Robot hút bụi Xiaomi Mijia Gen | 10 |
| Hình 2. 5 Robomow RS622..... | 10 |
| Hình 3. 1 Mô hình động học và động lực học của Robot..... | 16 |
| Hình 3. 2 Mô hình động học và động lực học của Robot khi quay một góc θ | 16 |
| Hình 3. 4 Các giá trị của IMU | 30 |
| Hình 3. 5 Cảm biến từ trường..... | 31 |
| Hình 3. 6 Hiệu ứng Hall..... | 31 |
| Hình 4. 1 WowWee Rovio Robot..... | 13 |
| Hình 4. 2 Robot Omni do đại học bách khoa thành phố Hồ Chí Minh thiết kế ... | 13 |
| Hình 4. 3 Robot SPARK IV | 14 |
| Hình 4. 4 Hệ số ma sát khô và hệ số ma sát ướt..... | 19 |
| Hình 4. 5 Chuyển động tịnh tiến..... | 22 |
| Hình 4. 6 Chuyển động quay | 22 |
| Hình 4. 7 Sử dụng cảm biến | 25 |
| Hình 4. 8 Sử dụng xử lý ảnh..... | 25 |
| Hình 4. 9 Mô phỏng cách một cảm biến quét trong không gian | 26 |
| Hình 4. 10 Cách phân bố cảm biến trên Mobile Robot | 27 |
| Hình 5. 1 Sơ đồ khối hệ thống điều khiển Mobile Robot..... | 28 |
| Hình 5. 2 Sơ đồ khối điều khiển động cơ | 46 |
| Hình 5. 3 Giải thuật điều khiển động cơ PID | 46 |
| Hình 5. 4 Sơ đồ khối xác định vị trí của Mobile Robot trong không gian | 52 |
| Hình 5. 5 Các góc quét của cảm biến siêu âm | 54 |
| Hình 5. 8 Đặc tính của aSMD300..... | 29 |

| | |
|---|----|
| Hình 5. 9 Động cơ DC Servo JGB37-520 DC Geared Motor | 29 |
| Hình 5. 10 MPU6050..... | 32 |
| Hình 5. 11 US-015 | 40 |
| | |
| Hình 7. 1 Thực nghiệm US-015..... | 57 |
| Hình 7. 2 Đồ thị hiển thị vận tốc theo thời gian khi Setpoint là 100rpm | 58 |
| Hình 7. 3 Đồ thị hiển thị vận tốc theo thời gian khi Setpoint là 250rpm | 58 |

CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI

1.1. Giới thiệu

Khi nhắc đến Robot thì không thể nào bỏ qua một loại robot đang rất thịnh hành và phổ biến trên toàn thế giới. Đó chính là Robot tự hành (Robot di động) hay còn gọi là Autonomous Mobile Robots (AMRs). Đây là loại robot hội tụ đầy đủ những yếu tố đặc trưng của kỷ nguyên máy tính – tự động hóa với hệ thống liên kết thế giới thực và ảo. Hiện nay, AMRs không chỉ phục vụ cho công nghiệp mà chúng còn đang được con người sử dụng một cách đầy thông minh và tối ưu trong đời sống xã hội: y tế, giáo dục, nhà hàng khách sạn,... thậm chí là phục vụ nhu cầu sinh hoạt tại nhà.

Một robot di động, là một con robot có khả năng di chuyển trong xung quanh (vận động). Chúng có khả năng điều hướng trong môi trường không được kiểm soát mà không cần đến các thiết bị hướng dẫn vật lý hoặc cơ điện. Ngoài ra, rô bốt di động có thể dựa vào các thiết bị hướng dẫn cho phép chúng đi theo một lộ trình điều hướng được xác định trước trong không gian được kiểm soát tương đối (AGV - xe dẫn đường tự hành).

Robot di động là một ngành định hướng giải pháp đang phát triển nhanh chóng, kết hợp khoa học kỹ thuật và công nghệ thông tin với các ngành như khoa học nhận thức, Trí tuệ nhân tạo và nhiều ngành khác. Chính sự tương tác liên ngành này đã giúp chúng ta có thể làm chủ được sự phức tạp vốn có của robot di động. Robot di động có khả năng di chuyển độc lập và thực hiện các hành động nhất định. Về cơ bản, ngoài khả năng di chuyển, chúng có thể hoạt động tự chủ mà không cần sự can thiệp của con người. Có một sự khác biệt chung giữa các thiết bị bán tự trị và hoàn toàn tự động, dựa trên nguồn điện của robot; nguồn điện bên ngoài, chẳng hạn như trạm sạc, có tác dụng hạn chế quyền tự chủ.

Mặc dù lĩnh vực robot di động hiện đang ở giai đoạn phát triển tương đối sớm, các nguyên mẫu và thậm chí một số sản phẩm nổi tiếp đang bắt đầu được tìm thấy trong một loạt các lĩnh vực. Robot di động cũng đã tìm được đường vào các ứng dụng hàng ngày trong nước với việc phát minh ra máy hút bụi di động, có thể hoạt động với khả năng tự chủ gần như hoàn toàn. Robot di động đã trở nên phổ biến hơn trong các môi trường thương mại và công nghiệp. Các bệnh viện đã sử dụng robot di động tự động để di chuyển vật liệu trong nhiều năm. Các nhà kho đã lắp đặt hệ thống rô bốt di động để di chuyển hiệu quả nguyên liệu từ các kệ chứa hàng đến khu vực thực hiện đơn hàng. Robot

đi động cũng là một trọng tâm chính của nghiên cứu hiện nay và hầu hết mọi trường đại học lớn đều có một hoặc nhiều phòng thí nghiệm tập trung vào nghiên cứu robot di động. Robot di động cũng được tìm thấy trong các cơ sở công nghiệp, quân sự và an ninh. Những đổi mới được thực hiện trong lĩnh vực Trí tuệ nhân tạo và các ứng dụng của chúng được thiết lập để nâng cao đáng kể mức độ liên quan và sử dụng các robot di động, độc lập trong tương lai.

Ưu nhược điểm của Robot di động:

- ***Ưu điểm :***

Mobile Robot là loại robot có thể thực hiện các tác vụ ở các địa điểm khác nhau, không ở cố định một vị trí nào. Linh động là đặc tính của robot di động, có thể có được từ các bộ phận chuyển động như bánh xe, chân, tay, cánh quạt... Robot di động “phải biết” định vị và “thu nhận” được thông tin đầy đủ về môi trường xung quanh, sau đó mới có quyết định thực hiện hành động nào cho phù hợp.

Do đó, Mobile Robot thường được tích hợp các cảm biến nhằm giúp cho chúng có thể nhận biết. Ngoài ra, robot di động còn có thể gắn kết với một hệ thống máy tính điều khiển và hệ thống cung cấp điện năng cho các chuyển động cũng như các cảm biến. Tùy vào tính chất công việc, các robot di động có thể phải mang theo nguồn điện, camera, micro, bộ cảm biến và các bộ xử lý. Tuy nhiên, do các robot di động đều có một tải trọng nhất định, nên khi thiết kế, cần tính toán trọng lượng các vật mang theo này ở mức vừa phải. Một đặc điểm quan trọng khác nữa là các robot di động cần phải có tính tự động một cách tương đối, nghĩa là phải có khả năng tự làm một hành động nào đó mà không cần có sự can thiệp của con người.

- ***Nhược điểm:***

Khác với robot cố định, robot di động có những yêu cầu cao hơn, đòi hỏi đầu tư nhiều hơn. Trong khi robot cố định vận hành khá đơn giản, chỉ cần không gian cố định để thực các công việc lặp đi lặp lại, còn hệ thống robot di động hoạt động trong không gian mở, thay đổi liên tục và đôi khi rất phức tạp.

Ứng dụng của Robot di động:

Mobile Robot ứng dụng cho rất nhiều loại công việc khác nhau từ xây dựng đến nông nghiệp, từ đào mìn đến thăm dò dầu khí, xử lý môi trường, y tế, giải trí, vận chuyển,...

- **Ngành lắp ráp tự động**
 - Lắp ráp lớp: di chuyển lớp xe màu xanh lá cây từ lưu trữ trung gian để chữa các trạm báo chí
 - Thiết bị điện tử ô tô: PCBs di chuyển và các thành phần nhỏ cho dòng phụ bổ sung
 - Phụ kiện tự động: vận chuyển totes đến và đi từ các trạm ép phun
- **Kỹ thuật số**
 - Semiconductor wafer fab: vận tải WIP intrabay giữa stockers để xử lý các công cụ, lưới carô vận tải cho quá trình in ảnh litô
 - Đóng gói bán dẫn và kiểm tra: di chuyển khay chip IC qua xe đẩy
 - Mobile Device Sản xuất: di chuyển tote của PCBs trong lắp ráp thiết bị cầm tay
 - Trung tâm dữ liệu: giám sát môi trường và xử lý sự cố trong máy chủ trang trại lớn
- **Đồ ăn - uống**
 - Nhà máy Snack: di chuyển các hộp các tông để đóng gói đường
 - Cơ sở phục vụ: di chuyển totes thực phẩm nướng để phòng chứng khoán
- **Logistics**
 - Kho hàng: thương mại điện tử thực hiện đơn hàng, trung tâm phân phối hàng
 - Chung ánh sáng Sản xuất Sản phẩm / tiêu dùng
- **Sản xuất đồ trang sức**
 - Di chuyển khuôn mẫu trang sức hoàn thành trạm đúc
 - Sản xuất kính mát: băng tải ảo để di chuyển hộp kính mát từ ASRS đài sắp xếp lại và sau đó mang lại cho phần hoàn thiện đến tải
- **Khách sạn:** Dịch vụ phòng giao hàng
- **Bệnh viện:** Dụng cụ phẫu thuật sinh tại bệnh viện

1.2. Lý do chọn đề tài

Để phục vụ cho nền công nghiệp đang phát triển vô cùng năng động và đang là xu thế trên toàn thế giới, không thể thiếu đến Robot. Robot là một loại thiết bị có thể thực hiện những công việc một cách tự động bằng sự điều khiển của máy tính hoặc các vi

mạch điện tử được lập trình. Robot có một trong các đặc điểm sau đây: do con người sáng tạo ra, có khả năng nhận biết môi trường xung quanh và tương tác với những vật thể trong môi trường, có khả năng đưa ra các lựa chọn dựa trên môi trường và được điều khiển một cách tự động theo những trình tự đã được lập trình trước, có thể điều khiển được bằng các lệnh để có thể thay đổi tùy theo yêu cầu của người sử dụng, có thể di chuyển quay hoặc tịnh tiến theo một hay nhiều chiều và khéo léo trong vận động.

Để góp phần vào sự phát triển nền khoa học kỹ thuật của nước nhà, nhóm chúng em đã chọn “ ***Mobile Robot di chuyển đa hướng bằng bánh xe omni*** ”. Từ những suy nghĩ đó chúng em sử dụng những kiến thức mình có để nghiên cứu chế tạo Mobile Robot trong phạm vi đồ án tốt nghiệp với ước muốn đóng góp vào công nghệ chế tạo Robot của nước nhà trong thời gian tới.

1.3. Tính cấp thiết của đề tài

Tính ứng dụng của Mobile Robot hiện nay là rất nhiều. Vì thế, nhờ sự phát triển mạnh mẽ của công nghệ, các khu công nghiệp hay các tổ ấm gia đình đều có xuất hiện Mobile Robot, việc này giúp cho năng suất làm việc và cuộc sống của các gia đình trở nên dễ dàng hơn tiết kiệm thời gian, nhanh chóng, hiệu quả và sức lao động của con người. Tuy nhiên, do điều kiện các doanh nghiệp hay các hộ gia đình nhà máy ở Việt Nam còn có sự hạn chế của công nghệ này nên họ cần phải có giải pháp nhanh chóng để tiết kiệm thời gian

1.4. Ý nghĩa của đề tài

Với đề tài này, nhóm mong muốn tạo ra một robot có khả năng tự hành trong một khu vực nhất định được xác định từ trước, có thể thay thế con người để đảm bảo an toàn, tránh những nguy hiểm có thể xảy ra. Ngoài ra có thể tiết kiệm được thời gian cũng như chi phí. Với đề tài này, nhóm nghiên cứu mong muốn góp phần nâng cao năng suất lao động, mang lại lợi ích cho doanh nghiệp.

1.5. Mục tiêu của đề tài

- Nghiên cứu, thiết kế Mobile Robot có ngoại hình gần gũi phù hợp với các ứng dụng của doanh nghiệp, dễ dàng tương tác.
- Xác định được vị trí của Robot trong khu vực hoạt động.
- Phát hiện và dừng lại để tránh các vật cản.

1.6. Phương pháp nghiên cứu

Để thực hiện đề tài này, phương pháp sử dụng cho nghiên cứu này chủ yếu là kết hợp giữa nghiên cứu lý thuyết và thực nghiệm. Về nghiên cứu lý thuyết, nhóm đã tham khảo các tài liệu về cơ khí, điện - điện tử, các tài liệu nghiên cứu trong và ngoài nước, các đề tài chế tạo Mobile Robot đã được làm. Về thực nghiệm, nhóm đã tiến hành thiết kế, lắp ráp mô hình Mobile Robot, Ngoài ra, nhóm cũng kết hợp các kiến thức chuyên ngành cơ điện tử, kiến thức đa ngành, sử dụng các phần mềm máy tính để tính toán, thiết kế, mô phỏng, xây dựng mô hình thực nghiệm, qua đó có thể đáp ứng các đặc điểm của Mobile Robot.

CHƯƠNG 2: TỔNG QUAN VỀ MOBILE ROBOT

2.1. Lịch sử phát triển

Năm 1939 – 1945, Trong cuộc chiến thế giới lần thứ II, những con Robot di động đầu tiên xuất hiện. Nó là kết quả của những thành tựu công nghệ trong những lĩnh vực nghiên cứu mới có liên quan như khoa học máy tính và điều khiển học, hầu hết chúng là những quả bom bay, ví dụ như những quả bom chỉ nổ trong những dãy mục tiêu nhất định, sử dụng trong hệ thống hướng dẫn vad rada điều khiển. Tên lửa V1 và V2 có “phi công tự động” và hệ thống phát nổ, chúng là tiền thân của đầu đạn hạt nhân tự điều khiển hiện đại.

Năm 1961 đến 1963: Trường đại học Johns Hopkins phát triển “Beast”. Beast sử dụng hệ thống định vị di chuyển xung quanh. Khi pin yếu nó sẽ tự tìm ổ cắm điện và cắm vào.

Năm 1969: Mowbot là con Robot đầu tiên cất cả bãi cỏ một cách tự động. The Stanford Cart line follower là một con Robot di động có thể di chuyển thông qua nhận dạng đường kẻ trắng, sử dụng một camera để nhìn. Nó bao gồm một “kênh truyền thanh” gắn với hệ thống máy tính lớn để tạo ra những tính toán.

Năm 1980: Thị hiếu của người tiêu dùng về Robot tăng, robot được bày bán và mua về để sử dụng trong nhà. Ví dụ RB5X vẫn tồn tại tới ngày nay và một loạt mẫu robot HERO. Robot The Stanford Cart được phát triển mạnh, nó đã có thể lái tàu biển vượt qua những trở ngại và tạo lên bản đồ những nơi nó đi qua.

Năm 1989: Mark Tinden phát minh ra BEAM robotics.

Năm 1990: Cha đẻ của nền rôbốt công nghiệp Joseph Engelberger làm việc với các đồng nghiệp và đã phát minh ra những con rôbốt tự động trong ngành y tế và được bán bởi Helpmate.

Năm 1996-1997: NASA phóng con tàu Mars Pathfinder có 2 Robot Rover và Sojourner lên sao Hoả. The Rover thám hiểm bề mặt sao hoả được điều khiển từ mặt đất. Sojourner được trang bị với một hệ thống tránh rủi ro cao. Hệ thống này làm cho Sojourner có thể tìm thấy đường đi của nó một cách độc lập trên địa hình của sao Hoả.

Năm 1999: Sony giới thiệu Aibo, một con rôbốt có khả năng đi lại, quan sát và tác động qua lại tới môi trường. Robot điều khiển từ xa dùng cho quân sự PackBot cũng được giới thiệu.

Năm 2001: Dự án Swaim-Bots, Swaim-Bots giống những bầy côn trùng được khởi động. Chúng bao gồm một số lượng lớn các con Robot đơn lẻ, có thể tác động lẫn nhau và cùng nhau thực hiện những nhiệm vụ phức tạp.

Năm 2002: Roomaba, một con robot di động dung trong gia đình, thực hiện công việc lau nhà xuất hiện. Tiếp tục phát triển hiện nay có rất nhiều loại Robot phục vụ cho con người dần xuất hiện ngày càng thân thiện hơn.

Năm 2004: Robosapien, một con rôbốt đồ chơi, thiết kế bởi Mark Tilden được bán sẵn. Trong dự án “The Centibots Project” 100 con Robot cùng làm việc với nhau để tạo lên một bản đồ cho một vùng không xác định và tìm những vật thể trong môi trường đó. Trong cuộc thi đầu tiên DARPA Grand Challenge, các con Robot tự động đã cùng nhau tranh tài cùng nhau trên sa mạc.

Năm 2006: Sony dừng việc sản xuất Aibo và Helpmate. PatrolBot trở lên phổ biến khi các Robot di động vẫn tiếp tục cạnh tranh nhau để trở thành mặt hàng độc quyền. Sở an ninh Mỹ đã bỏ dự án MDARS-I, nhưng lại gây quỹ cho dự án MDARS-E một loại Robot an ninh tự động khác. TALON-Sword, một loại Robot tự động dùng để bán sẵn với dàn phóng lựu đạn và những sự lựa chọn về vũ khí hợp thành khác đã ra đời. Asimo của Honda biết cách chạy và leo cầu thang chỉ với hai chân như con người.

Năm 2008, Boston Dynamics phát hành video quay toàn cảnh một BigDog thể hệ mới có thể đi trên địa hình bằng giá và phục hồi thăng bằng khi bị tác động từ bên hông.

Năm 2010, Thử thách quốc tế robot tự động đa năng có đội xe tự điều khiển ánh xạ một môi trường đô thị năng động lớn, xác định, theo dõi con người và các đối tượng tránh thù địch.

Năm 2016, các hướng dẫn của robot di động sử dụng các thẻ RFID là một phương pháp mới để đi theo con đường sử dụng các thẻ RFID. Người ta chứng minh rằng robot luôn đến đích bằng sai số đo khoảng cách ngay cả khi các phép đo khoảng cách ngay cả khi các phép đo khoảng cách và góc không chính xác. Nó cũng có khả năng chọn đúng con đường trong nhiều con đường khác nhau.

Năm 2017, trong cuộc thi ARGOS Challenge, các robot được phát triển để hoạt động trong điều kiện khắc nghiệt tại các trạm dầu khí ngoài khơi.

2.2. Phân loại

Robot di động có thể được phân loại theo:

- Môi trường mà chúng phục vụ:

Robot trên đất liền hoặc robot gia đình thường được gọi là Phương tiện trên mặt đất không người lái (Unmanned Ground Vehicles -UGVs). Chúng phổ biến nhất là có bánh xe hoặc bánh xích, nhưng cũng bao gồm các robot có chân có hai hoặc nhiều chân (hình người, hoặc giống động vật hoặc côn trùng).

Robot Giao hàng & Vận chuyển có thể di chuyển vật liệu và vật tư trong môi trường làm việc.

Robot trên không thường được gọi là Máy bay không người lái (Unmanned Aerial Vehicles - UAVs)..

Robot dưới nước thường được gọi là phương tiện tự động dưới nước (Autonomous Underwater Vehicles - AUVs).

Các robot vùng cực, được thiết kế để điều hướng các môi trường đầy băng giá như ở các trạm nghiên cứu tại Bắc Cực, Nam Cực.

- Thiết bị họ sử dụng để di chuyển, chủ yếu là:

Robot có chân : chân giống người (tức là android) hoặc chân giống động vật.

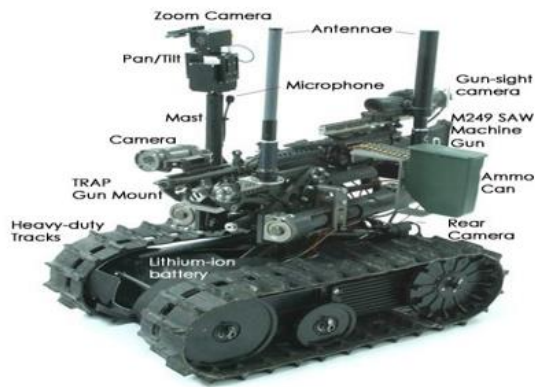
Robot có bánh xe.

Robot có bánh ray.

- Các dạng điều khiển

Điều khiển từ xa bằng tay

Robot điều khiển từ xa bằng tay được điều khiển thông qua sóng RF, wifi, hồng ngoại hay Bluetooth... Robot điều khiển từ xa giúp con người tránh khỏi những nguy hiểm. Ví dụ Robot điều khiển từ xa bằng tay gồm có: Military-Robot và iRobot's PackBot...



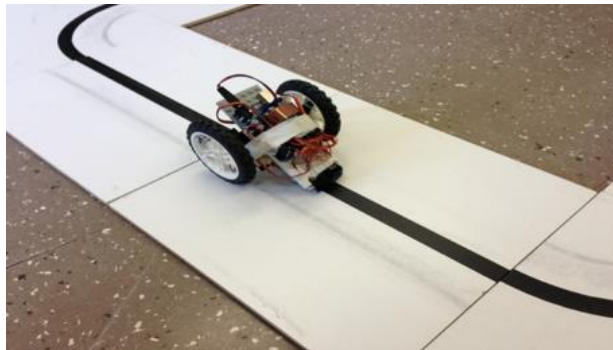
Hình 2. 1 Military Robot



Hình 2. 2 iRobot 510 PackBot

Di chuyển theo lộ trình

Một vài Robot tự động đầu tiên là những con Robot theo lộ trình. Chúng có thể theo những đường được sơn khắc trên sàn trên trần nhà hay một dây điện. Đa số những Robots này hoạt động theo một thuật toán đơn giản là giữ lộ trình trong bộ cảm biến trung tâm, chúng không thể vòng qua các chướng ngại vật, chúng chỉ dừng lại khi có vật nào đó cản đường chúng. Rất nhiều mẫu của loại Robot này vẫn được bán bởi FMC, Egemin, HK sytem và một vài công ty khác Egemin, HK sytem và một vài công ty khác.



Hình 2. 3 Robot dò line

Di chuyển ngẫu nhiên:

Robot hoạt động độc lập với những chuyển động ngẫu nhiên, về cơ bản đó là những chuyển động nhảy bật lên tường, những bức tường được cảm nhận do sự cản trở về mặt vật lý như máy hút bụi Roomba hoặc với bộ cảm biến điện tử của máy cắt cỏ Friendly Robotics.

*Hình 2. 4 Robot hút bụi Xiaomi Mijia Gen**Hình 2. 5 Robomow RS622*

2.3. Đặt vấn đề

Trong thế kỷ XXI, cuộc sống con người trở nên bận rộn để làm những công việc hỗ trợ cuộc sống con người ngày càng tiện nghi và dễ dàng hơn. Robot dịch vụ có thể cung cấp sự tiện nghi cho người sử dụng bằng việc thực hiện các công việc nhàm chán, lặp đi lặp lại, nguy hiểm,...

Tuy nhiên, việc đầu tư và kết quả nghiên cứu về Robot di động đặc biệt các robot phục vụ chuyên nghiệp các nhiệm vụ như: quét dọn, phục vụ nhà cửa, nội trợ, quản lý không gian, trợ lý cho con người,... còn khá khiêm tốn. Đây là do chúng ta mới bắt đầu nghiên cứu robot dịch vụ vì lực lượng nghiên cứu về lĩnh vực này hiện tại còn rất mỏng tại Việt Nam.

Vì vậy, bên cạnh những nghiên cứu ban đầu này, vẫn chưa có một nghiên cứu mang tính hệ thống hay là một sản phẩm Robot di động hoàn thiện đáp ứng được yêu cầu trong khi nhu cầu thực tế đặt ra đối với Robot dịch vụ và an ninh là cấp bách.

Robot gia đình là loại robot thân thiện với người dùng, rẻ tiền, an toàn và hỗ trợ người dùng trong một số công việc gia đình cụ thể. Trong khi mục tiêu cuối cùng của robot gia đình là một robot thông minh giúp việc trong gia đình thay thế công việc đa năng của một “ô sin” thì thực tế phát triển của robot gia đình hiện nay lại là những robot nhỏ và chỉ làm một việc nhất định như hút bụi, lau nhà, lau kính hay cắt cỏ. Đây là một thị trường đã bị quên lãng ở Mỹ nhưng lại được Nhật rất chú ý. Trong khi chính phủ Mỹ hỗ trợ kinh phí lớn cho phát triển các robot quân sự và không chú ý đến phát triển robot gia đình thì ngược lại, Nhật Bản lại tập trung nguồn lực cho nghiên cứu phát triển robot dịch vụ dân sự hơn là quân sự. Nếu tình trạng này tiếp tục kéo dài thì Nhật Bản sẽ là nước chiếm lĩnh thị trường robot gia đình trên thế giới và bỏ xa Mỹ tụt hậu phía sau. Ở

Mỹ, công nghệ robot được coi là tài sản quốc gia với nhiều bí mật quân sự và rất lâu sau đó mới chuyển giao cho các lợi ích dân sự. Trong khi đó, Nhật coi công nghệ robot là nền tảng cạnh tranh quốc tế của Nhật trong tương lai. Chính phủ Nhật ước tính công nghiệp robot sẽ tăng trưởng mạnh và nước Nhật sẽ đi lên với thành công của robot gia đình. Chỉ có một loại robot gia dụng của Mỹ thành công trên thị trường đó là robot hút bụi Roomba của hãng iRobot. Giá thành của Roomba chỉ vài trăm USD nhưng làm việc tốt và thân thiện với người dùng. Các loại robot gia dụng trước Roomba hoặc là quá phức tạp như robot công nghiệp hoặc là quá đơn giản như loại robot đồ chơi. Thực tế nhiều robot công nghiệp được coi là phức tạp thực chất lại kém phức tạp hơn Roomba.

Nhật Bản có động lực cho phát triển ngành công nghiệp robot. Với hơn 1/5 dân số có tuổi thọ trên 65 tuổi, nước Nhật phụ thuộc vào robot để thay thế nguồn nhân lực và chăm sóc người già. Do vậy, đối với nước Nhật, một cuộc cách mạng robot là bắt buộc. Hàn Quốc cũng là quốc gia đặt mục tiêu phát triển mạnh robot gia đình với giá thành rẻ để mỗi gia đình Hàn Quốc có một robot gia đình trong thập niên này.

Tuy thị trường của robot gia đình còn bé hiện nay nhưng khi ngành công nghiệp này khởi sắc nó sẽ có sức tăng trưởng mạnh và trở thành một lĩnh vực rất hữu ích và rất sinh lợi.

Khả năng định hướng trong không gian:

Bài toán định hướng hiện nay được phân làm 2 loại: định hướng trong nhà (indoor) và định hướng ngoài địa hình (outdoor). Quá trình định hướng của robot đi động gồm 4 bước: thu nhận cảnh quan môi trường, xác định vị trí, thiết kế quỹ đạo và tạo chuyển động. Với môi trường có cấu trúc, quá trình nhận biết cho phép tạo ra bản đồ hay mô hình không gian phục vụ cho bài toán định vị và thiết kế quỹ đạo robot. Đối với môi trường phi cấu trúc hay thay đổi, robot cần có khả năng tự học quan sát môi trường để xác định được hướng đi của mình. Do đó lĩnh vực xác định hướng đi cho robot di động là một lĩnh vực mà các phương pháp trí khôn nhân tạo như quá trình nhận biết môi trường, suy diễn và tìm hướng đi tối ưu có thể được áp dụng. Vấn đề định vị và tạo bản đồ là những vấn đề nghiên cứu trọng tâm ở robot di động thời gian qua.

Quá trình định vị là quá trình robot xác định được hiện nó ở đâu trong không gian hoạt động. Để đạt được mục tiêu này, cần sử dụng nhiều cảm biến thu nhận các dữ liệu liên quan đến trạng thái của robot và môi trường xung quanh. Các dữ liệu này thường bị

hiều và có sai số tích lũy nên cần có các phương pháp lọc động và sử dụng các phương pháp phối hợp cảm biến để có được số liệu đo chính xác hơn. Phương pháp định vị có thể là cục bộ hay toàn cục. Giải pháp đơn giản nhất là định vị cục bộ khi robo thường xuyên cập nhật vị trí của nó so với điểm xuất phát. Ngược lại các phương pháp định vị toàn cục không đòi hỏi biết vị trí của điểm xuất phát. Để khắc phục độ bất định của các thông tin đo được từ các cảm biến ta cần sử dụng các phương pháp xác xuất. Các phương pháp định vị được sử dụng thường dựa trên nguyên lý lọc Bayes kết hợp với một thuật toán đệ quy để ước lượng được vị trí và hướng từ phương trình mô tả chuyển động của robot. Thời gian tính toán của lọc Bayes lâu nên nhiều nghiên cứu gần đây tập trung vào tìm các phương pháp đơn giản hóa để giảm khối lượng tính toán. Quá trình đơn giản hóa này dẫn đến nhiều thuật toán định vị khác nhau phân làm 2 loại tùy thuộc vào cách mô tả độ tin cậy của dữ liệu. Nếu dữ liệu được mô tả bằng các hàm phân bố Gauss, ta có thể sử dụng phương pháp lọc Kalman. Nếu dữ liệu được mô tả bằng nhiều hàm phân bố xác xuất khác nhau, ta có thể sử dụng các thuật toán định vị dựa trên quá trình Markov. Phương pháp định vị dựa trên phân bố Gauss và lọc Kalman chỉ ứng dụng hiệu quả cho bài toán định vị cục bộ. Các phương pháp định vị Markov có thể là phương pháp tô pô, phương pháp lưới và phương pháp sử dụng các mẫu rời rạc của các giá trị trạng thái. Khi các dữ liệu robot được mô tả bằng các mẫu rời rạc ngẫu nhiên ta có thể sử dụng các phương pháp lọc phần tử (particle filter) để xác định được vị trí của rô bốt tốt hơn.

2.4. Giới hạn đề tài

Mục tiêu của nhóm là làm một chiếc **Mobile Robot** trong nhà (indoor) có khả năng phục vụ con người về các công việc đơn giản trong nhà: lau nhà, hút bụi, quản lý không gian, an ninh trong nhà,...

Điều khiển được động cơ và thu được giá trị từ các cảm biến hỗ trợ cho Mobile Robot.

Vì lý do dịch bệnh mọi người không thể làm việc trực tiếp với nhau vì vậy việc làm đồ án môn học đối với chúng em rất khó khăn mong thầy cô có thể thông cảm cho tụi em.

CHƯƠNG 3: THIẾT KẾ CƠ KHÍ CHO MOBILE ROBOT

3.1. Một số nghiên cứu về Omni Robot trên thế giới và Việt Nam

Trên thế giới, Omni robot đã được phát triển từ lâu và ngày càng có nhiều ứng dụng đa dạng do được tích hợp các công nghệ mới nhất.

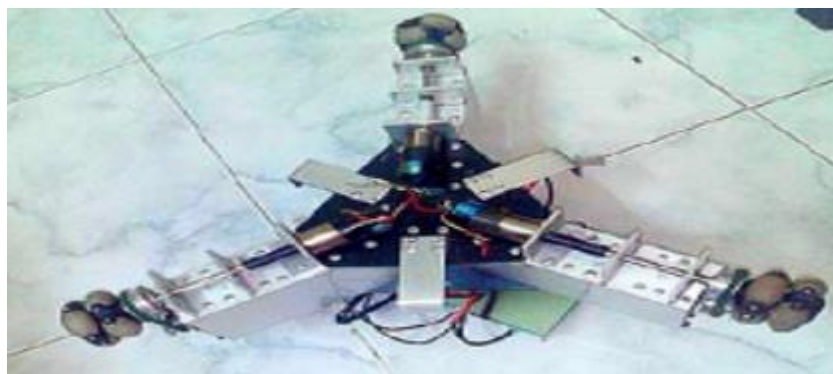


Hình 3. 1 WowWee Rovio Robot

Tính năng:

- Có thể điều khiển qua wifi.
- Có thể điều khiển qua mạng internet.
- Có thể tự di chuyển xung quanh nhà do có nguồn dự trữ.

Những nghiên cứu về Omni Robot ở Việt Nam đã được thực hiện trong thời gian gần đây bởi các sinh viên và giáo viên trong các trường đại học. Những nghiên cứu này đã đạt được những kết quả bước đầu, tạo ra được sản phẩm thực nghiệm.



Hình 3. 2 Robot Omni do đại học bách khoa thành phố Hồ Chí Minh thiết kế



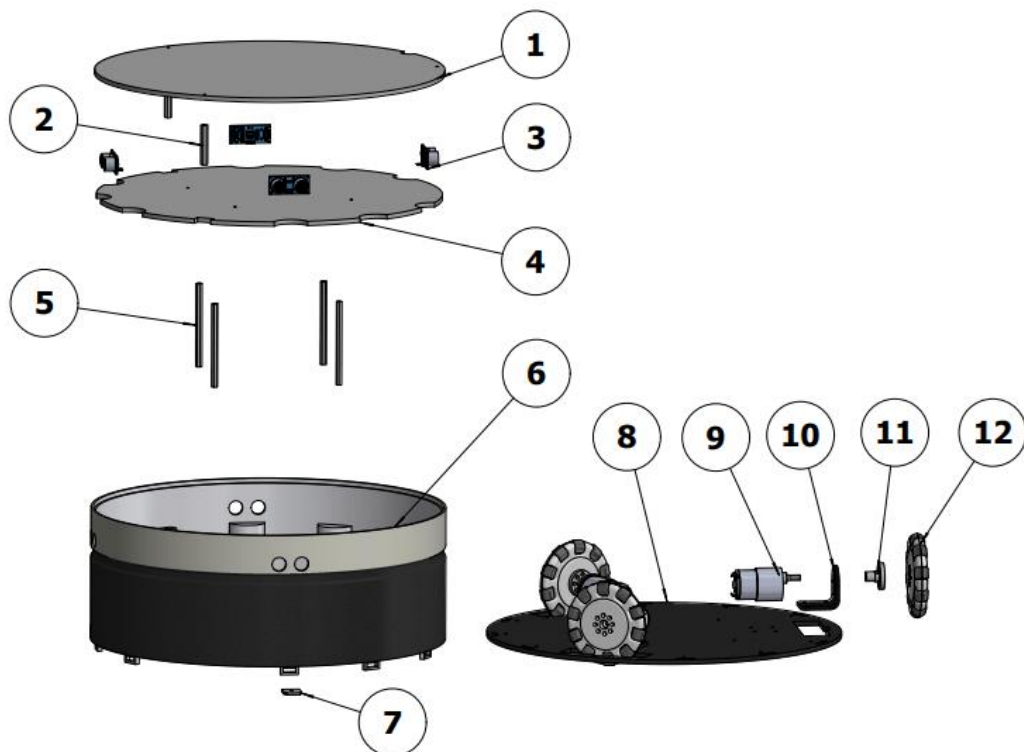
Hình 3. 3 Robot SPARK IV

Omni Robot tránh vật cản sử dụng 3 cảm biến hồng ngoại do nhóm sinh viên đại học bách khoa Hà Nội chế tạo.

Mục đích của hệ thống:

Trên cơ sở đồ cơ cấu Mobile Robot của nhóm nghiên cứu trước đã thực hiện được bám đối tượng và tránh vật cản trong quá trình hoạt động, đồ án tính toán động học và quỹ đạo cho robot để xác định được thuật toán điều khiển cho từng quỹ đạo.

3.2. Thi công mô hình



Hình 3. 4 Bản vẽ phân rã của Robot

| Chi tiết | Tên | Chức năng |
|----------|---------------|---|
| 1 | Nắp | Che phần bên trong robot đồng thời bảo vệ robot |
| 2 | Trụ đỡ 45mm | Đỡ nắp trên |
| 3 | Cảm biến | Cảm biến siêu âm SR-HC04 giúp robot tránh vật cản |
| 4 | Nắp ngăn | Tăng độ thẩm mỹ cho robot |
| 5 | Trụ đỡ 100mm | Đỡ nắp che |
| 6 | Thân | Chứa các bộ phận hoạt động của robot |
| 7 | Then | Liên kết cho đế và thân robot |
| 8 | Đế | Giữ bánh xe và đỡ các phần điện của robot |
| 9 | Động cơ | Giúp robot chuyển động |
| 10 | Gá đỡ động cơ | Đỡ động cơ để gá trên đế |
| 11 | Chốt | Liên kết động cơ và bánh xe |
| 12 | Bánh xe | Giúp robot chuyển động |

Hình 3. 5 Bảng các chi tiết và chức năng

Mô hình sản phẩm hoàn thành:

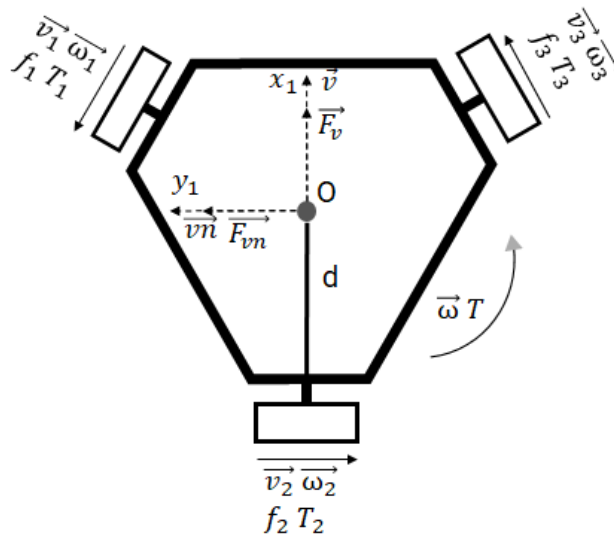


Hình 3.6 Mô hình sản phẩm hoàn thành Mobile Robot

3.3. Lựa chọn cơ cấu chuyển động cho Mobile Robot

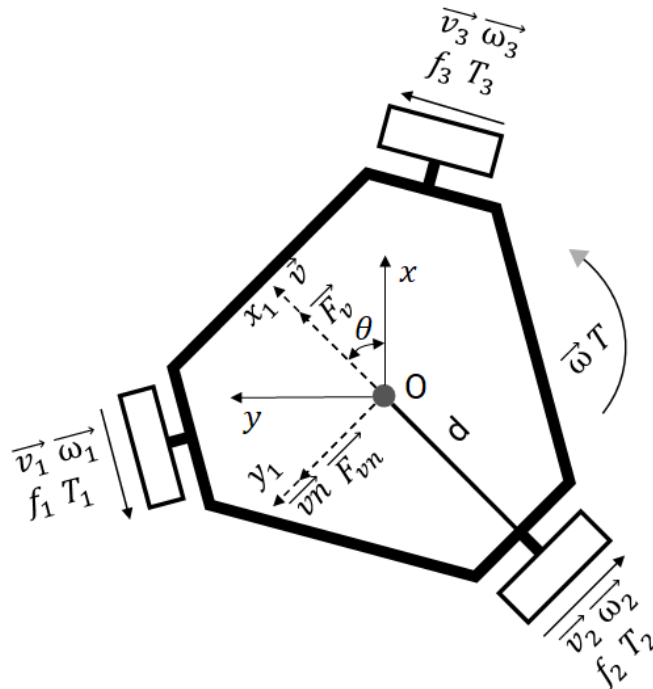
Chuyển động của robot là do chuyển động của các bánh xe. Các bánh xe chuyển động do động cơ lắp ở mỗi trục bánh quay và sinh ra lực kéo. Vậy ta xét đến mô hình robot, trong đó các lực kéo đặt ở các bánh xe. Vận tốc của robot được tính dựa vào vận tốc của các bánh xe. Từ đó ta có mô hình robot như sau:

Lúc đầu robot chưa quay:



Hình 3. 7 Mô hình động học và động lực học của Robot

Sau khi robot quay một góc θ :



Hình 3. 8 Mô hình động học và động lực học của Robot khi quay một góc θ

Robot gồm 3 bánh đặt lệch nhau 120° , tiếp tuyến với một đường tròn đường kính 350mm. Gọi các bánh xe lần lượt là 1, 2, 3. Chọn 2 hệ trục tọa độ như sau: hệ trục tọa độ Oxy gắn cố định với mặt phẳng sàn, hệ trục tọa độ $O_{x_1y_1}$ gắn với robot như trên hình vẽ. Tại thời điểm ban đầu giả thiết rằng hai trục tọa độ trùng với nhau. Tại thời điểm t, hai hệ trục tọa độ lệch nhau 1 góc θ chính là góc quay của robot so với vị trí ban đầu. Sau đây là các kí hiệu dùng trong hình vẽ.

Vị trí của robot trong hệ trục tọa độ Oxy theo phương Ox: x.

Vị trí của robot trong hệ trục tọa độ Oxy theo phương Oy: y.

Khoảng cách giữa các bánh và tâm robot: d[m] (Hình 2.4).

Vận tốc dài của các bánh xe: v_1, v_2, v_3 [m/s].

Vận tốc góc của bánh: $\omega_1, \omega_2, \omega_3$ [rad/s].

Lực do động cơ sinh ra trên các bánh: f_1, f_2, f_3 [N].

Mô men kéo do động cơ sinh ra đặt lên các bánh: T_1, T_2, T_3 [Nm].

Vận tốc theo 2 phương trên hệ $O_{x_1y_1z_1}$ của robot: v, vn, ω_1 [m/s] (z_1 trùng z).

Vận tốc góc của robot: ω [rad/s].

3.4. Thiết kế kỹ thuật cơ khí

3.4.1. Mô hình động học cho robot

Gọi $v_x(t), v_y(t), \omega(t)$ là vận tốc tức thời của robot trên hệ trục tọa độ Oxy.

$$\begin{aligned} v_x(t) &= \frac{dx(t)}{dt} \\ v_y(t) &= \frac{dy(t)}{dt} \\ \omega(t) &= \frac{d\theta(t)}{dt} \end{aligned} \quad (3.1)$$

Chiếu các vận tốc trên lên hệ trục tọa độ $O_{x_1y_1}$ gắn với robot, ta có:

$$\begin{aligned} v(t) &= v_x(t) \cos(\theta(t)) + v_y(t) \sin(\theta(t)) \\ vn(t) &= -v_x(t) \sin(\theta(t)) + v_y(t) \cos(\theta(t)) \\ \omega_1(t) &= \omega(t) \end{aligned} \quad (3.2)$$

Viết lại các công thức trên dưới dạng ma trận, ta được công thức, ta nhận được công thức liên hệ giữa vận tốc theo 2 hệ trục tọa độ khác nhau:

$$\begin{bmatrix} v(t) \\ vn(t) \\ \omega_1(t) \end{bmatrix} = \begin{bmatrix} \cos(\theta(t)) & \sin(\theta(t)) & 0 \\ -\sin(\theta(t)) & \cos(\theta(t)) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_x(t) \\ v_y(t) \\ \omega(t) \end{bmatrix} \quad (3.3)$$

Xét trong hệ trục gắn với robot, $O_{x_1y_1}$, giả sử robot chuyển động với các vận tốc là $v(t)$, $vn(t)$ và $\omega(t)$, ta có công thức tính vận tốc của các bánh là:

$$\begin{aligned} v_1(t) &= -v(t)\sin(\pi/3) + vn(t)\cos(\pi/3) + \omega_1(t)d \\ v_2 &= -vn(t) + \omega(t)d \end{aligned} \quad (3.4)$$

$$v_3(t) = v(t)\sin(\pi/3) + vn(t)\cos(\pi/3) + \omega_1(t)d$$

Viết lại hệ phương trình (4.4) ta thu được dạng ma trận:

$$\begin{bmatrix} v_1(t) \\ v_2(t) \\ v_3(t) \end{bmatrix} = \begin{bmatrix} -\sin(\pi/3) & \cos(\pi/3) & d \\ 0 & -1 & d \\ \sin(\pi/3) & \cos(\pi/3) & d \end{bmatrix} \begin{bmatrix} v(t) \\ vn(t) \\ \omega_1(t) \end{bmatrix} \quad (3.5)$$

Giải hệ phương trình trên ta có:

$$\begin{aligned} v(t) &= \frac{\sqrt{3}}{3}(v_3(t) - v_1(t)) \\ vn(t) &= \frac{1}{3}(v_3(t) + v_1(t)) - \frac{2}{3}v_2(t) \\ \omega_1(t) &= \frac{1}{4d}(v_1(t) + v_2(t) + v_3(t)) \end{aligned} \quad (3.6)$$

Các công thức trên cho phép ta tính vận tốc của các robot(vận tốc theo các phương của trục tọa độ và vận tốc robot) khi đã biết vận tốc các bánh xe.

3.4.2. Mô hình động lực học cho robot

Theo phương trình các phương trình tính gia tốc và gia tốc góc cho một vật

$$M\dot{v} = F$$

$$J\dot{\omega} = T$$

Ta có:

$$\begin{aligned} m \frac{dv(t)}{dt} &= F_v(t) - F_{Bv}(t) - F_{Cv}(t) \\ m \frac{dvn(t)}{dt} &= F_{vn}(t) - F_{Bvn}(t) - F_{Cvn}(t) \end{aligned} \quad (3.7)$$

$$J \cdot \frac{d\omega(t)}{dt} = T(t) - T_{B\omega}(t) - T_{C\omega}(t)$$

Trong đó:

$m[\text{kg}]$: Khối lượng robot

$J[\text{kgm}^2]$: Mô men quán tính của robot

$F_{Bv}, F_{Bvn}[N]$: Lực ma sát nhớt dọc theo phương Ox_1 và Oy_1 tác dụng lên robot

$T_{B\omega}[Nm]$: Mô men ma sát nhớt tác dụng lên robot

$F_{Cv}, F_{Cvn}[N]$: Lực ma sát khô dọc theo phương Ox_1 và Oy_1 tác dụng lên robot

$T_{C\omega}[Nm]$: Mô men ma sát khô tác dụng lên robot

Lực ma sát nhớt tỉ lệ với vận tốc của robot, xem hình 3.9 :

$$F_{Bv}(t) = B_v v(t)$$

$$F_{Bvn}(t) = B_{vn} vn(t)$$

$$F_{Bvn}(t) = B_{\omega} \omega(t)$$

$B_v, B_{vn}[N/(m/s)]$: Hệ số ma sát nhớt

$B_{\omega}[N/(rad/s)]$: Hệ số ma sát nhớt do vận tốc quay

Độ lớn của lực ma sát khô là hằng số, xem hình 3.9:

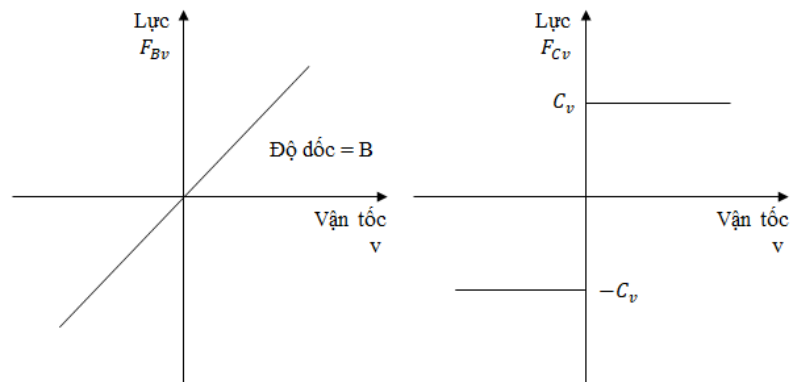
$$F_{Cv}(t) = C_v \text{sign}(v(t))$$

$$F_{Cvn}(t) = C_{vn} \text{sign}(vn(t)) \quad (3.8)$$

$$T_{C\omega}(t) = C_{\omega} \text{sign}(\omega(t))$$

$C_v, C_{vn}[N]$: Hệ số ma sát khô dọc theo phương Ox_1 và Oy_1

$C_{\omega}[Nm]$: Hệ số ma sát khô theo ω



Hình 3. 9 Hệ số ma sát khô và hệ số ma sát ướt

Mối quan hệ giữ lực kéo của robot, mô men quay và lực quay trên mỗi bánh được đưa vào công thức sau:

$$\begin{aligned}F_v(t) &= (f_3(t) - f_1(t)) \sin \frac{\pi}{3} \\F_{vn}(t) &= -f_2(t) + (f_3(t) + f_1(t)) \cos \frac{\pi}{3} \\T(t) &= (f_1(t) + f_2(t) + f_3(t))d\end{aligned}$$

Lực tác dụng lên robot dọc theo phương O_{x_1} và O_{y_1} : F_v, F_{vn} [N]

Mô men robot (theo ω): T [Nm]

Lực kéo trên mỗi bánh được ước lượng theo mô men kéo có thể xác định được bằng cách sử dụng dòng điện của động cơ, được miêu tả bằng công thức dưới đây :

$$f(t) = \frac{T(t)}{r} \quad (3.9)$$

$$T(t) = K \cdot i(t)$$

r: bán kính bánh xe

l : Hệ số của hộp tốc độ

r[m]: Bán kính bánh xe

$K_t [\frac{Nm}{A}]$: Hằng số mô men động cơ

i_j [A] : Dòng động cơ

3.4.3. Mô hình cho động cơ

$$u_j(t) = L \frac{di_j(t)}{dt} + Ri_j(t) + K_v \omega_{mj}(t) \quad (3.10)$$

$$T_j(t) = K_t i_j$$

L[H]: Độ tự cảm của động cơ

R[Ω]: Điện trở của động cơ

K_v [V/(rad/s)]: Hằng số suất điện động ngược

u_j [V]: Điện áp động cơ

ω_{mj} [rad/s]: Vận tốc quay của động cơ

T_{mj} : Mô men động cơ thứ j

Trong trạng thái ổn định, độ tự cảm L bằng 0, nên ta có thể viết lại công thức:

$$\begin{aligned} u_j(t) &= R i_j(t) + K_v \omega_{mj}(t) \\ \frac{u_j(t)}{i_j(t)} &= R + \frac{K_v \omega_{mj}(t)}{i_j(t)} \end{aligned} \quad (3.11)$$

Kết hợp công thức ở trên ta có công thức:

$$\frac{dx(t)}{dt} = Ax(t) + Bu(t) + K \text{sign}(x) \quad (3.12)$$

$$x(t) = [v(t) \quad v_n(t) \quad \omega(t)] \quad (3.13)$$

Công thức này trình bày cách tính gia tốc của robot theo phương v , v_n và gia tốc góc cho robot.

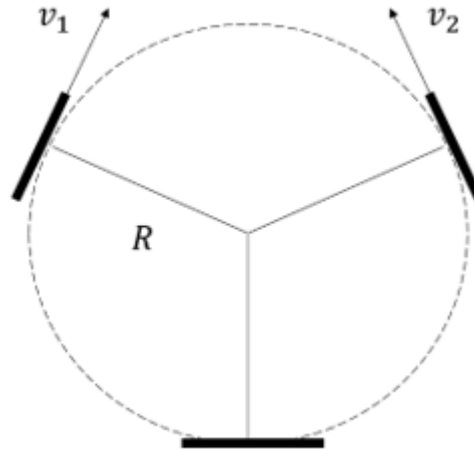
Trong đó ta có:

$$\begin{aligned} A &= \text{diag}(A_{11}, A_{22}, A_{33}) \\ A_{11} &= -\frac{3K_t^2 l^2}{2r^2 RM} - \frac{B_v}{M} \\ A_{22} &= -\frac{3K_t^2 l^2}{2r^2 RM} - \frac{B_{v_n}}{M} \\ A_{33} &= -\frac{3dK_t^2 l^2}{r^2 RJ} - \frac{B_\omega}{J} \\ B &= \frac{lK_t}{rR} \begin{bmatrix} -\frac{\sqrt{3}}{2M} & 0 & \frac{\sqrt{3}}{2M} \\ \frac{1}{2M} & \frac{1}{M} & \frac{1}{2M} \\ \frac{d}{J} & \frac{d}{J} & \frac{d}{J} \end{bmatrix} \\ K &= \text{diag}\left(-\frac{C_v}{M}, -\frac{C_{v_n}}{M}, -\frac{C_\omega}{M}\right) \end{aligned}$$

3.4.4. Mô hình động học cho một số chuyển động cơ bản

Các chuyển động dùng để lập trình cho robot gồm có:

Chuyển động tịnh tiến: Khi cho 2 bánh bất kỳ trong 3 bánh quay cùng lúc theo 1 phương nào đó (2 bánh đó quay ngược chiều nhau) thì sẽ làm cho robot chuyển động tịnh tiến.



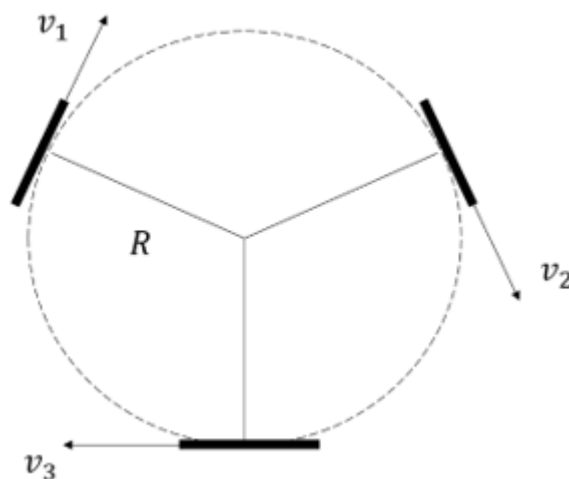
Hình 3.10 Chuyển động tịnh tiến

Ví dụ: Khi cho 2 bánh xe V1 và V2 theo phương như hình vẽ 3.10 với r là bán kính của bánh xe Omni. Giả sử 2 bánh xe cùng quay với vận tốc ω (quay ngược chiều nhau) có vector vận tốc dài là V1 và V2.

Tổng hợp 2 chuyển động trên ta có vector chuyển động tịnh tiến của robot:

$$v_{tt} = 2v_1 \cos \varphi = 2v_2 \cos \varphi = 2\omega r \cos 30^\circ = \sqrt{3}\omega r$$

Chuyển động quay: Khi cho 3 bánh xe quay cùng chiều thì sẽ làm cho robot quay quanh tâm của nó (hình 3.11). Tùy vào góc quay mà độ lớn vận tốc và hướng quay của các bánh sẽ khác nhau. Căn cứ vào vận tốc các bánh mà ta có thể điều khiển góc quay và vận tốc của robot.



Hình 3.11 Chuyển động quay

3.4.5. Tính toán lựa chọn động cơ

Khối lượng ước tính $m = 10\text{kg}$, vận tốc $v = 0.25\text{m/s}$, bán kính bánh xe chính $r = 50\text{mm}$, hệ số ma sát của bánh xe $\mu = 0.3$, $g = 9.8\text{m/s}^2$.

Theo định luật 2 Newton:

$$\Sigma F = F_w - F_f = ma$$

$$F_w = ma + F_f$$

$$F_w = ma + \mu mg$$

$$F_w = 31.9 \text{ N}$$

Momen của động cơ

$$T = F_w r = 1.595 \text{ N.m}$$

Số vòng quay của trục động cơ (vòng/phút)

$$N = \frac{v \cdot 1000}{\pi \cdot D} \text{ (vòng/phút)}$$

$$N = \frac{15 \cdot 1000}{\pi \cdot 100} = 48 \text{ (vòng/phút)}$$

$$\omega = \frac{N \cdot 2\pi}{60} = 5.03 \text{ (rad/s)}$$

Công suất cần thiết:

$$P = T\omega = 10.2 \text{ W}$$

Hiệu suất của hộp số: $\eta_1 = 0.97$

Hiệu suất của động cơ: $\eta_2 = 0.85$

Hiệu suất của hệ thống: $\eta = \eta_1 \eta_2 = 0.8$

$$P_t = \frac{P}{\eta} = 12.75 \text{ W}$$

Với $T = 1.585 \text{ N.m}$, $P_t = 12.75 \text{ W}$, $N = 48 \text{ v/p}$.

Ta chọn động cơ DC Servo JGB37-520 DC Geared Motor.

Thông tin sản phẩm: Động cơ DC Servo JGB37-520 DC Geared Motor



Hình 3.12 Động cơ DC Servo JGB37-520 DC Geared Motor

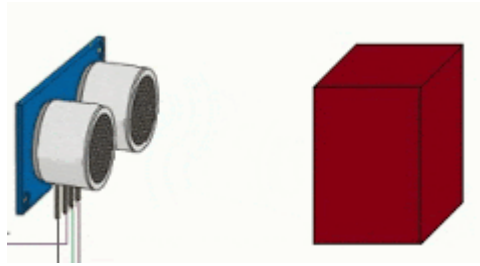
Thông số kỹ thuật:

- Điện áp sử dụng: 12VDC
- Đường kính: 37mm
- Encoder: Cảm biến từ trường Hall, có 2 kênh AB lệch nhau giúp xác định chiều quay và vận tốc của động cơ, đĩa Encoder trả ra 11 xung/1 kênh/ 1 vòng (nếu đo tín hiệu đồng thời của cả hai kênh sẽ thu được tổng 22 xung / 1 vòng quay của Encoder).
 - Cách tính số xung của mỗi kênh trên 1 vòng quay của trục chính động cơ = Tỷ số truyền x số xung của Encoder, ví dụ tỷ số 150:1 thì số xung Encoder trả ra cho 1 vòng quay của trục chính động cơ sẽ là $11 \times 150 = 1650$ xung / 1 kênh.
 - Điện áp cấp cho Encoder hoạt động: 3.3~5VDC, lưu ý cấp quá áp hoặc ngược chiều sẽ làm cháy Encoder ngay lập tức
 - Công suất : từ 7W – 15W.

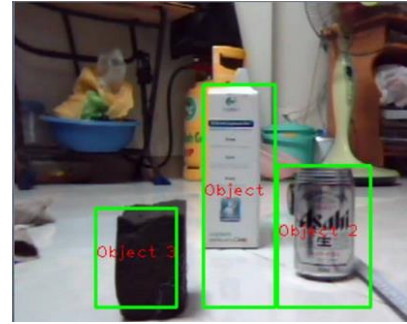
3.4.6. Tính toán chọn số cảm biến

Cách thức di chuyển và ta cũng đã chọn được động cơ cho Mobile Robot vậy giờ làm sao để robot có thể di chuyển một cách tự do trong phạm vi khu vực bạn cho robot di chuyển?

Có rất nhiều cách để cho Mobile robot nhận diện được vật cản trước mắt để cử như sử dụng cảm biến được gắn xung quanh để robot nhận diện được khoảng cách tới vật cản mà tránh né, nếu như cần chính xác hơn khu vực di chuyển phức tạp hơn thì có thể sử dụng xử lý ảnh để có thể nhận diện đối tượng trước mặt mà di chuyển thông minh hơn trong hệ thống công nghiệp.



Hình 3.13 Sử dụng cảm biến



Hình 3.14 Sử dụng xử lý ảnh

Vì giới hạn của đề tài này sử dụng cảm biến gắn xung quanh để robot nhận biết là cách hiệu quả và ít tốn kém nhất. Chọn cảm biến US015 để sử dụng cho đề tài này.

Cảm biến siêu âm US015 là một dạng cảm biến module. Cảm biến này thường chỉ là một bản mạch, hoạt động theo nguyên lý thu phát sóng siêu âm bởi 2 chiếc loa cao tần.

Nguyên lý hoạt động:

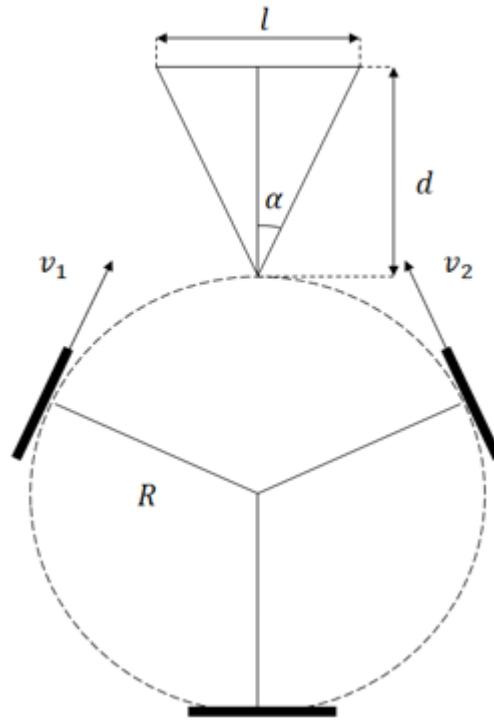
Để đo khoảng cách bằng cảm biến siêu âm US-015, ta sẽ phát 1 xung rất ngắn (5 microSeconds) từ chân Trig. Tiếp theo, 1 xung HIGH ở chân Echo sẽ được cảm biến tạo ra và phát đi cho đến khi nhận lại được sóng phản xạ ở chân này. Lúc này, độ rộng của xung sẽ bằng với thời gian sóng siêu âm được phát từ cảm biến và phản xạ lại.

Trong không khí, tốc độ âm thanh đạt mức 340 m/s (hằng số), tương đương với 29,412 cm/ μ s ($106 / (340 * 100)$).

Khi đã tính được thời gian, ta sẽ chia cho 29,412 để ra giá trị khoảng cách.

Tính toán chọn số lượng cảm biến:

Di chuyển cơ bản của robot là di chuyển tịnh tiến vì vậy ta gắn cảm biến vào hướng di chuyển thẳng của cảm biến như hình vẽ.



Hình 4. 4 Mô phỏng cách một cảm biến quét trong không gian

d (mm) khoảng cách phát hiện vật của US-015.

α ($^{\circ}$) góc cảm biến.

l (mm) khoảng quét vật của cảm biến.

R (mm) bán kính của robot.

v_1, v_2 (m/s) vận tốc của hai bánh trước.

Theo thông số của cảm biến US-015 ta có:

$$d_{max} = 4m = 4000mm$$

$$\alpha = 15^{\circ}$$

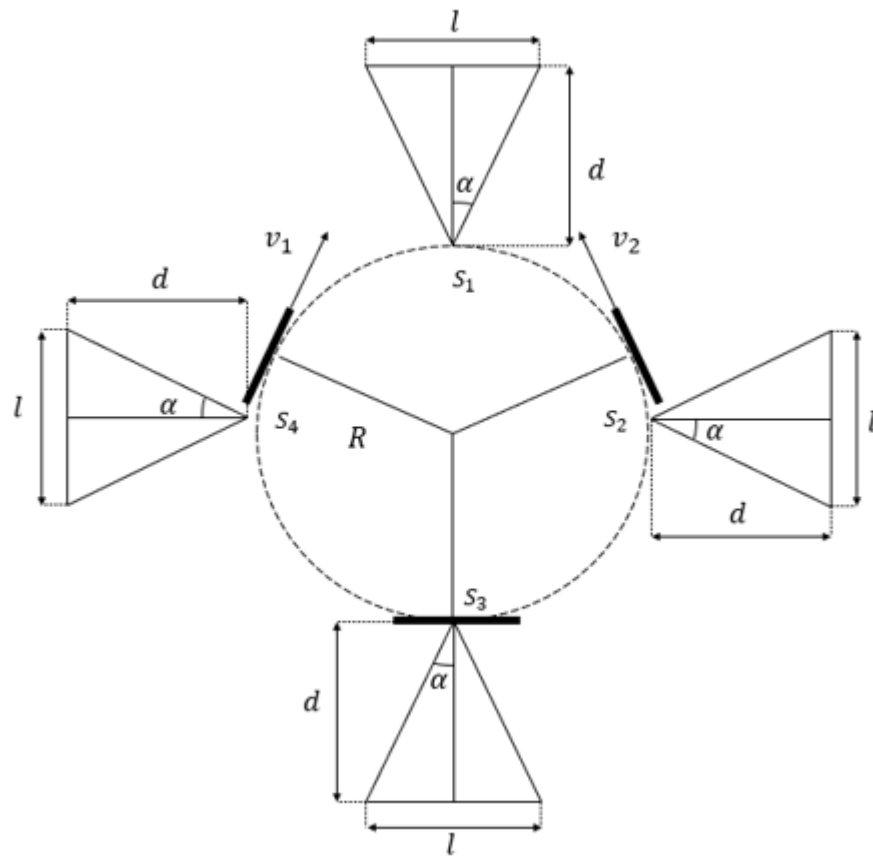
$$\frac{l_{max}}{2} = \sin(\alpha) \times d_{max} = \sin(15^{\circ}) \times 4000 = 1035.3 \text{ (mm)}$$

$$\Rightarrow l_{max} = 2070.55 \text{ (mm)}$$

Ví dụ: Chọn $d = 100mm$ ta tìm được khoảng quét :

$$\frac{l}{2} = \sin(\alpha) \times d = \sin(15^{\circ}) \times 100 = 26 \text{ mm} \Rightarrow l = 52 \text{ mm}$$

Tránh trường hợp 2 cảm biến quét cùng một khoảng quét ta chọn 4 cảm biến phân bố như hình:

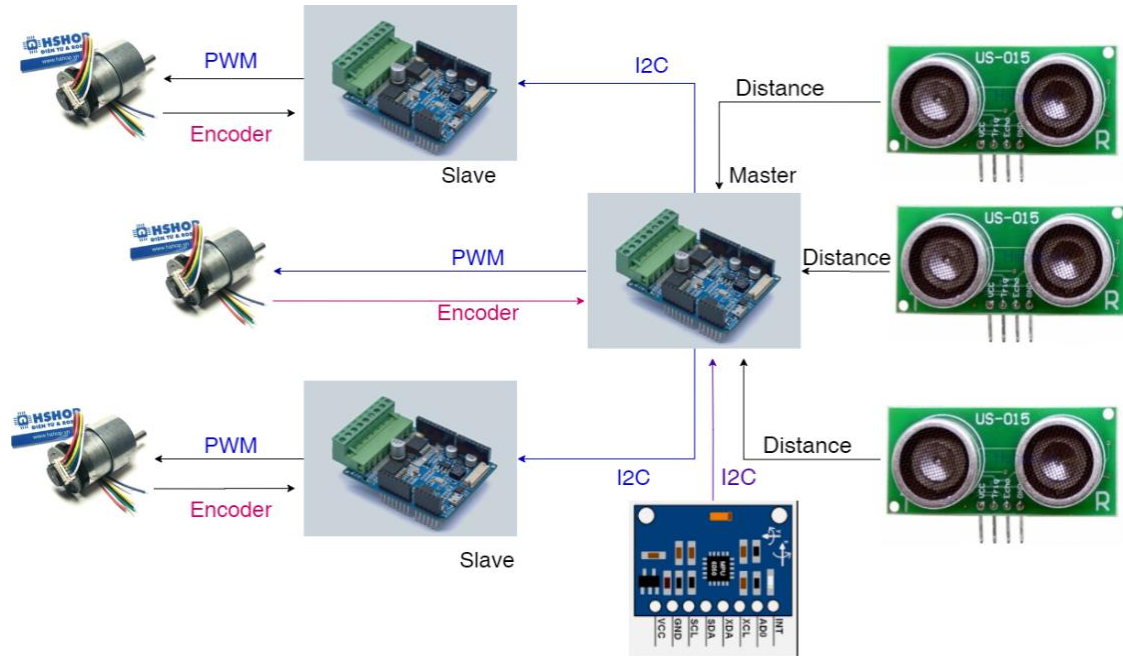


Hình 4. 5 Cách phân bố cảm biến trên Mobile Robot

CHƯƠNG 4: XÂY DỰNG HỆ THỐNG ĐIỆN – ĐIỀU KHIỂN

4.1. Hệ thống điều khiển Mobile Robot

Mạch điều khiển robot có nhiệm vụ điều khiển động cơ, đọc tín hiệu từ encoder và MPU6050 để xác định vị trí Robot, đọc tín hiệu cảm biến siêu âm để phát hiện vật cản.



Hình 4. 1 Sơ đồ khối hệ thống điều khiển Mobile Robot

Trong đó:

- aSMD350 Driver là khối xử lý trung tâm H-Bridge và STM32F103C8T6, giúp điều khiển động cơ cũng như xử lý các tín hiệu cảm biến.
- MPU6050(GY-521) xác định góc xoay.
- Encoder xác định vận tốc xe.
- Cảm biến siêu âm US-015.

4.2. Hệ thống điện Mobile Robot

4.2.1. Các thiết bị được sử dụng trên Mobile Robot

4.2.1.1. Automotive 300W Single Motor Driver



aSMD300

automotive 300W Single Motor Driver

Specifications

| | |
|---------------|-------------------------------|
| Dimension | 64x53x12 (mm) |
| Voltage range | 12-30VDC |
| Rated Current | 12A* |
| Peak Current | 20A* |
| MCU | STM32 open source PID control |
| Protection | Fast fuse/over current/ over |

Hình 4.2 Đặc tính của aSMD300

aSMD300 là một bộ Driver được tích hợp MCU STM32. Dễ dàng sử dụng lắp ráp để điều khiển nhiều động cơ một lúc. Gồm nhiều cách giao tiếp như: I2C, Uart, CAN.

4.2.1.2. Động cơ

Động cơ DC Servo JGB37-520 DC Geared Motor



Hình 4.3 Động cơ DC Servo JGB37-520 DC Geared Motor

Thông số kỹ thuật:

- Điện áp sử dụng: 12VDC
- Đường kính: 37mm

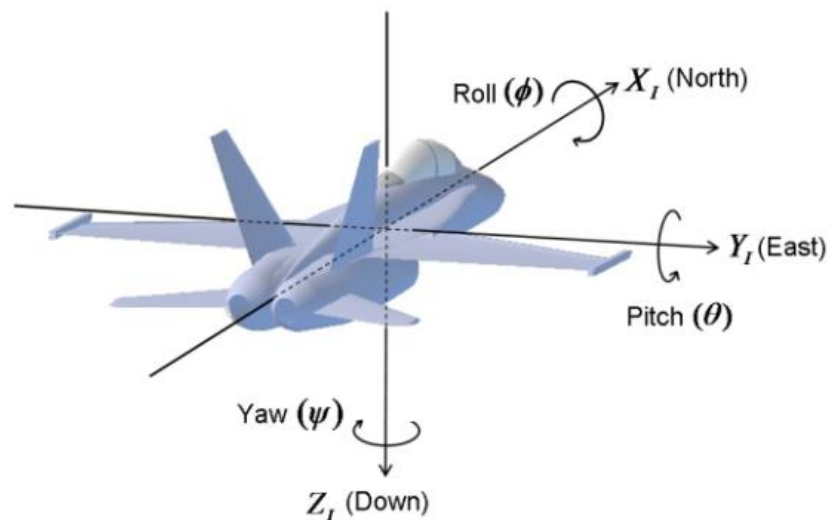
4.2.1.3. Cảm biến gia tốc MPU6050

Giới thiệu về IMU:

Inertial Measurement Unit - Đơn vị đo lường quán tính, có chức năng cung cấp thông tin về tốc độ góc và góc nghiêng của hệ thống. Dựa trên nguyên lý hoạt động của

2 cảm biến accelerometer (gia tốc kế) và gyroscope (con quay hồi chuyển). Tuy nhiên khái niệm này được sử dụng khá lỏng lẻo, do vậy IMU có thể hiểu là đơn vị đo lường quán tính của hệ thống, cung cấp các giá trị cảm biến sau khi đã xử lý giúp cho hệ thống hoạt động tốt nhất.

Inertial Measurement Unit = Đơn vị đo lường quán tính, có chức năng cung cấp thông tin về tốc độ góc và góc nghiêng của hệ thống. Dựa trên nguyên lý hoạt động của 2 cảm biến accelerometer (gia tốc kế) và gyroscope (con quay hồi chuyển). Tuy nhiên khái niệm này được sử dụng khá lỏng lẻo: do vậy IMU có thể hiểu là đơn vị đo lường quán tính của hệ thống, cung cấp các giá trị cảm biến sau khi đã xử lý giúp cho hệ thống hoạt động tốt nhất.



Hình 4.4 Các giá trị của IMU

Ứng dụng của IMU

Các cảm biến IMU được sử dụng trong các ứng dụng như: robot tự cân bằng, quadcopter, điện thoại thông minh.... Cảm biến IMU giúp chúng ta có được vị trí của vật thể gắn với cảm biến trong không gian ba. Chúng được sử dụng để phát hiện hướng của điện thoại thông minh hoặc trong các tiện ích như Fitbit, sử dụng cảm biến IMU để theo dõi chuyển động.

Module cảm biến IMU có thể được tích hợp gồm hai hoặc nhiều cảm biến như:

- Cảm biến gia tốc (accelerometer)
- Cảm biến góc quay (gyroscope)
- Cảm biến từ trường (magnetometer)

- Áp suất kế để đo độ cao (altimeter)

Từ các giá trị đọc được từ cảm biến, thông thường sẽ cần phải xử lý qua các thuật toán calib, lọc, fuse dữ liệu để cho ra các giá trị cảm biến có thể sử dụng cho yêu cầu ứng dụng tùy trường hợp.

Nguyên lý hoạt động của từng loại:

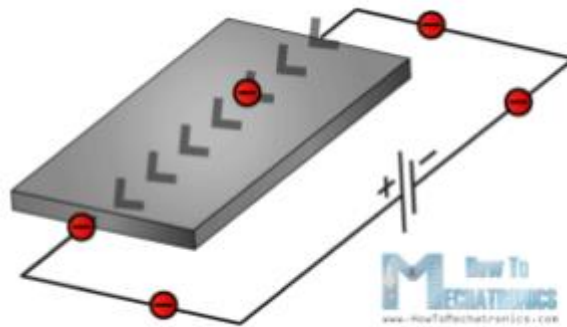
- Cảm biến gyroscope:

Khi bạn đặt một con chip IMU để im không chuyển động, giá trị trả về gyro = [0.0, 0.0, 0.0] do không có bất cứ chuyển động quay nào cả. Gyro chỉ đo tốc độ quay chứ không đo trực tiếp góc quay, nên khi bạn quay module một góc nào đó rồi dừng, giá trị của gyro sẽ tăng lên rồi hạ xuống về 0.

- Cảm biến gia tốc kế:

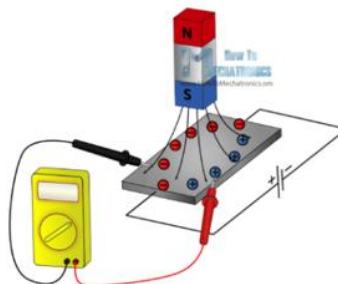
Nguyên tắc để đo góc dùng gia tốc kế là phân tích sự tác dụng của trọng lực lên các trục ta có thể dùng nó để tính các góc lệch roll và pitch.

- Cảm biến từ trường:



Hình 4.5 Cảm biến từ trường

Cảm biến từ trường đa phần hoạt động dựa trên hiệu ứng Hall. Ta cấp nguồn cho dòng điện chạy qua mạch, có một tấm dẫn điện như trên hình.



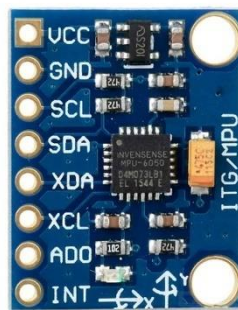
Hình 4.6 Hiệu ứng Hall

Khi xung quanh nó có từ trường, lực Lorent sẽ làm dòng electron chạy trong mạch nó di chuyển lệch đi về 2 phía. Nếu ta đo điện áp thì sẽ biết được độ lớn của nguồn từ trường này.

Nếu chọn trục z là trục vuông góc với mặt đất thì góc yaw là góc khi xoay trục z. Nó thường được ứng dụng để đo góc yaw bằng công thức đơn giản. Trên thực tế ta có thể tính toán góc yaw bằng gyroscope, nhưng nếu ta kết hợp thêm cảm biến từ trường nó sẽ cho kết quả chính xác hơn, lấp đi nhược điểm khi dùng gyroscope.

- Cảm biến áp suất kế (Barometer):

Barometer tính toán độ cao bằng cách sử dụng tính chất độ cao thay đổi theo áp suất, càng lên cao áp suất càng giảm. Ta chỉ cần áp dụng công thức để chuyển đổi.



Hình 4.7 MPU6050

Thông tin kỹ thuật:

- Điện áp sử dụng: 3~5VDC
- Điện áp giao tiếp: 3~5VDC
- Chuẩn giao tiếp: I2C
- Giá trị Gyroscopes trong khoảng: +/- 250 500 1000 2000 degree/sec
- Giá trị Acceleration trong khoảng: +/- 2g, +/- 4g, +/- 8g, +/- 16g
- Board mạch mạ vàng, linh kiện hàn tự động bằng máy chất lượng tốt nhất.

Cách sử dụng MPU6050 bằng STM32:

Khởi tạo:

- Thanh ghi 25: Xác định tần số lấy mẫu (Sample Rate Divider):

| Register (Hex) | Register (Decimal) | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|----------------|--------------------|-----------------|------|------|------|------|------|------|------|
| 19 | 25 | SMPLRT_DIV[7:0] | | | | | | | |

$$Ts \text{ lấy mẫu} = \text{Gyroscope Output Rate} / (1 + \text{SMPLRT_DIV})$$

Với:

- Ts lấy mẫu là do chúng ta chọn:
- Gyroscope Output Rate:
 - = 1KHz nếu Digital Low Pass Filter (DLPF) is enable (được chọn ở thanh ghi 26: Configuration: DLPF_CFG = 1 -> 6)
 - = 8KHz nếu Digital Low Pass Filter (DLPF) is disable (được chọn ở thanh ghi 26: Configuration: DLPF_CFG = 0 or 7)
- SMPLRT_DIV: bộ chia Sample Rate => được tính khi đã có hết các thông số trên

Ví dụ: 1s đo 50 lần => 50Hz

- $0.05\text{KHz} = 1 / (1 + \text{SMPLRT_DIV}) \Rightarrow \text{SMPLRT_DIV} = 19 \text{ (decimal)} \Rightarrow 0x13 \text{ (Hex)}$

| Register (Hex) | Register (Decimal) | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|----------------|--------------------|------|------|-------------------|------|------|---------------|------|------|
| 1A | 26 | - | - | EXT_SYNC_SET[2:0] | | | DLPF_CFG[2:0] | | |

- Thanh ghi 26:

Thanh ghi này cấu hình cho nguồn xung ngoại (The External Frame Synchronization -FSYNC) và bộ lọc số (The Digital Low Pass Filter – DLPF, có nhắc đến trong thanh ghi 25) để cài đặt cho cả con quay hồi chuyển (gyroscope) và gia tốc kế (accelerometers).

Giá trị mẫu được ghi lại ở bit đáng kể tối thiểu (LSB – least significant bit) trong thanh ghi dữ liệu của cảm biến, nó được xác định bởi giá trị của EXT_SYNC_SET theo bảng sau.

| EXT_SYNC_SET | FSYNC Bit Location |
|--------------|--------------------|
| 0 | Input disabled |
| 1 | TEMP_OUT_L[0] |
| 2 | GYRO_XOUT_L[0] |
| 3 | GYRO_YOUT_L[0] |
| 4 | GYRO_ZOUT_L[0] |
| 5 | ACCEL_XOUT_L[0] |
| 6 | ACCEL_YOUT_L[0] |
| 7 | ACCEL_ZOUT_L[0] |

DLPF được cấu hình bởi DLPF_CFG. Gia tốc kế và con quay hồi chuyển được sàng lọc theo các cách khác nhau mà đại diện mỗi cách được đại diện bởi giá trị của DLPF_CFG được nêu trong bảng sau.

| DLPF_CFG | Accelerometer ($F_s = 1\text{kHz}$) | | Gyroscope | | |
|----------|--|---------------|-------------------|---------------|----------|
| | Bandwidth (Hz) | Delay (ms) | Bandwidth (Hz) | Delay (ms) | Fs (kHz) |
| 0 | 260 | 0 | 256 | 0.98 | 8 |
| 1 | 184 | 2.0 | 188 | 1.9 | 1 |
| 2 | 94 | 3.0 | 98 | 2.8 | 1 |
| 3 | 44 | 4.9 | 42 | 4.8 | 1 |
| 4 | 21 | 8.5 | 20 | 8.3 | 1 |
| 5 | 10 | 13.8 | 10 | 13.4 | 1 |
| 6 | 5 | 19.0 | 5 | 18.6 | 1 |
| 7 | RESERVED | | RESERVED | | 8 |

- Thanh ghi 27: Cấu hình cho bộ đo Góc (Gyroscope)

| Register (Hex) | Register (Decimal) | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------------------|-----------------------|-------|-------|-------|-------------|------|------|------|------|
| 1B | 27 | XG_ST | YG_ST | ZG_ST | FS_SEL[1:0] | | - | - | - |

XG_ST, YG_ST, ZG_ST được chọn thì nó sẽ tự kiểm tra (self test).

Chọn FS_SEL cho khoảng đo (full scale range) cho các đầu ra bộ đo Góc (the Gyroscope outputs):

| FS_SEL | Full Scale Range |
|--------|---------------------------|
| 0 | $\pm 250^\circ/\text{s}$ |
| 1 | $\pm 500^\circ/\text{s}$ |
| 2 | $\pm 1000^\circ/\text{s}$ |
| 3 | $\pm 2000^\circ/\text{s}$ |

- Thanh ghi 28: Cấu hình cho bộ đo Gia tốc (Accelerometer – Acceleration: gia tốc)

| Register (Hex) | Register (Decimal) | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|-------------------|-----------------------|-------|-------|-------|--------------|------|------|------|------|
| 1C | 28 | XA_ST | YA_ST | ZA_ST | AFS_SEL[1:0] | | - | | |

XA_ST, YA_ST, ZA_ST được set = 1 thì nó sẽ tự kiểm tra giống thanh ghi 27 (self test)

Chọn AFS_SEL cho full scale range cho các đầu ra bộ đo Gia tốc (the Accelerometer outputs).

| AFS_SEL | Full Scale Range |
|---------|------------------|
| 0 | $\pm 2g$ |
| 1 | $\pm 4g$ |
| 2 | $\pm 8g$ |
| 3 | $\pm 16g$ |

- Thanh ghi 56: Cấu hình cho Interrupt

| Register (Hex) | Register (Decimal) | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|----------------|--------------------|------|------|------|---------------|----------------|------|------|-------------|
| 38 | 56 | | - | | FIFO_OFLOW_EN | I2C_MST_INT_EN | - | - | DATA_RDY_EN |

FIFO_OFLOW_EN: khi bit này được set lên 1, bit này cho phép tràn bộ đệm FIFO để tạo ra 1 sự kiện ngắt.

I2C_MST_INT_EN: khi bit này được set lên 1, bit này kích hoạt bất kỳ nguồn ngắt từ Master I2C nào để tạo ra 1 sự kiện ngắt.

DATA_RDY_EN: khi bit này được set lên 1, bit này kích hoạt sự kiện ngắt Data Ready, thứ mà xảy ra mỗi khi hoàn thành thao tác ghi vào các thanh ghi cảm biến

- Thanh ghi 107: Quản lý năng lượng (Power Manegement)

| Register (Hex) | Register (Decimal) | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|----------------|--------------------|--------------|-------|-------|------|----------|-------------|------|------|
| 6B | 107 | DEVICE_RESET | SLEEP | CYCLE | - | TEMP_DIS | CLKSEL[2:0] | | |

Thanh ghi này cho phép người dùng cấu hình cho mode năng lượng và nguồn clock. Nó cũng cung cấp 1 bit để reset cho toàn bộ thiết bị và 1 bit để vô hiệu hóa cảm biến nhiệt độ.

| CLKSEL | Clock Source |
|--------|---|
| 0 | Internal 8MHz oscillator |
| 1 | PLL with X axis gyroscope reference |
| 2 | PLL with Y axis gyroscope reference |
| 3 | PLL with Z axis gyroscope reference |
| 4 | PLL with external 32.768kHz reference |
| 5 | PLL with external 19.2MHz reference |
| 6 | Reserved |
| 7 | Stops the clock and keeps the timing generator in reset |

| | | |
|--------------|---|---|
| DEVICE_RESET | 1 | Reset toàn bộ thanh ghi bên trong về giá trị khởi tạo của nó. Sau khi việc reset hoàn tất, DEVICE_RESET tự động chuyển về 0. |
| | 0 | Tắt chế độ Reset |
| SLEEP | 1 | MPU-6050 chuyển về ngủ |
| | 0 | MPU-6050 hoạt động bình thường |
| CYCLE | 1 | Cùng với chế độ SLEEP vô hiệu hóa, MPU-60X0 sẽ chuyển đổi giữa chế độ ngủ và thức dậy để lấy một mẫu dữ liệu từ các cảm biến đang hoạt động với tốc độ được xác định bởi LP_WAKE_CTRL |
| | 0 | Hoạt động bình thường |
| TEMP_DIS | 1 | Vô hiệu hóa cảm biến nhiệt độ |
| | 0 | Cảm biến nhiệt độ hoạt động bình thường |
| CLKSEL | Giá trị 3 bit. Được mô tả chi tiết qua bảng trên. | |

Khi bật nguồn, nguồn clock của MPU-6050 mặc định lấy từ bộ dao động nội. Tuy nhiên, nó rất được khuyến khích rằng thiết bị được cấu hình bằng cách sử dụng một trong các con quay hồi chuyển (hoặc một nguồn xung ngoại) như là bộ tham chiếu xung để cải thiện khả năng hoạt động.

- Thanh ghi 59 – 64: Bộ

| Register (Hex) | Register (Decimal) | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|----------------|--------------------|------------------|------|------|------|------|------|------|------|
| 3B | 59 | ACCEL_XOUT[15:8] | | | | | | | |
| 3C | 60 | ACCEL_XOUT[7:0] | | | | | | | |
| 3D | 61 | ACCEL_YOUT[15:8] | | | | | | | |
| 3E | 62 | ACCEL_YOUT[7:0] | | | | | | | |
| 3F | 63 | ACCEL_ZOUT[15:8] | | | | | | | |
| 40 | 64 | ACCEL_ZOUT[7:0] | | | | | | | |

Các thanh ghi này ghi lại các giá trị của phép đo gia tốc trước đó.

Mỗi phép đo gia tốc kể 16 bit có một thang đo đầy đủ được xác định trong ACCEL_FS (Thanh ghi 28). Đối với mỗi cài đặt thang đo, độ nhạy của máy đo gia tốc trên mỗi LSB trong ACCEL_xOUT được hiển thị trong bảng bên dưới.

| AFS_SEL | Full Scale Range | LSB Sensitivity |
|---------|------------------|-----------------|
| 0 | $\pm 2g$ | 16384 LSB/g |
| 1 | $\pm 4g$ | 8192 LSB/g |
| 2 | $\pm 8g$ | 4096 LSB/g |
| 3 | $\pm 16g$ | 2048 LSB/g |

- Thanh ghi 67 - 72: Bộ đo vận tốc góc

| Register (Hex) | Register (Decimal) | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|----------------|--------------------|-----------------|------|------|------|------|------|------|------|
| 43 | 67 | GYRO_XOUT[15:8] | | | | | | | |
| 44 | 68 | GYRO_XOUT[7:0] | | | | | | | |
| 45 | 69 | GYRO_YOUT[15:8] | | | | | | | |
| 46 | 70 | GYRO_YOUT[7:0] | | | | | | | |
| 47 | 71 | GYRO_ZOUT[15:8] | | | | | | | |
| 48 | 72 | GYRO_ZOUT[7:0] | | | | | | | |

Các thanh ghi này ghi lại các giá trị của phép đo vận tốc góc trước đó.

Mỗi phép đo vận tốc góc kể 16 bit có một thang đo đầy đủ được xác định trong FS_SEL (Thanh ghi 27). Đối với mỗi cài đặt thang đo, độ nhạy của máy đo vận tốc góc trên mỗi LSB trong GYRO_xOUT được hiển thị trong bảng bên dưới.

| FS_SEL | Full Scale Range | LSB Sensitivity |
|--------|------------------------------------|-----------------------------|
| 0 | $\pm 250 \text{ }^\circ/\text{s}$ | 131 LSB/ $^\circ/\text{s}$ |
| 1 | $\pm 500 \text{ }^\circ/\text{s}$ | 65.5 LSB/ $^\circ/\text{s}$ |
| 2 | $\pm 1000 \text{ }^\circ/\text{s}$ | 32.8 LSB/ $^\circ/\text{s}$ |
| 3 | $\pm 2000 \text{ }^\circ/\text{s}$ | 16.4 LSB/ $^\circ/\text{s}$ |

- Thanh ghi 65 – 66: Đo nhiệt độ

| Register (Hex) | Register (Decimal) | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|----------------|--------------------|----------------|------|------|------|------|------|------|------|
| 41 | 65 | TEMP_OUT[15:8] | | | | | | | |
| 42 | 66 | TEMP_OUT[7:0] | | | | | | | |

Thanh ghi này ghi lại các giá trị của phép đo nhiệt độ trước đó.

Giá trị nhiệt độ (oC) được lấy ra từ thanh ghi có thể tính như sau:

$$\text{Giá trị nhiệt độ (oC)} = \frac{\text{Giá trị thô lấy ra từ thanh ghi}}{340} + 36.53$$

Các lệnh cần dùng với cảm biến MPU6050:

- Lệnh Write (Configuration)

```
HAL_I2C_Mem_Write    ( I2C_HandleTypeDef * hi2c,
                        uint16_t DevAddress,
                        uint16_t MemAddress,
                        uint16_t MemAddSize,
                        uint8_t * pData,
                        uint16_t Size,
                        uint32_t Timeout
                        )
```

- Lệnh Read (Read Data từ cảm biến)

```
HAL_I2C_Mem_Read    ( I2C_HandleTypeDef * hi2c,
                        uint16_t DevAddress,
                        uint16_t MemAddress,
                        uint16_t MemAddSize,
                        uint8_t * pData,
                        uint16_t Size,
                        uint32_t Timeout
                        )
```

- Định nghĩa thành phần trong lệnh

Với:

- hi2c (hi2c1/hi2c2): cấu hình i2c được chỉ định
- DevAddress: địa chỉ thiết bị đích
- MemAddress: địa chỉ thanh ghi mong muốn trên thiết bị đích
- MemAddSize: kích thước của địa chỉ thanh ghi nội (thường là 1)
- pData: dữ liệu đọc được
- Size: số lượng dữ liệu được gửi
- Timeout : thời gian chờ

Các bước thực hiện

- Cấu hình:

Chọn tần số lấy mẫu và bộ chia của nó dựa trên công thức:

$$Ts_{\text{lấy mẫu}} = \text{Gyroscope Output Rate} / (1 + \text{SMPLRT_DIV})$$

Với:

- Ts lấy mẫu là do chúng ta chọn:

- Gyroscope Output Rate:
 - = 1 kHz nếu Digital Low Pass Filter (DLPF) is enable (được chọn ở thanh ghi 26: Configuration: DLPF_CFG = 1 -> 6)
 - = 8 kHz nếu Digital Low Pass Filter (DLPF) is disable (được chọn ở thanh ghi 26: Configuration: DLPF_CFG = 0 or 7)
- SMPLRT_DIV: bộ chia Sample Rate => được tính khi đã có hết các thông số trên

Ví dụ: 1s đo 1000 lần => 1 kHz và Gyroscope Output Rate = 8 kHz

$1\text{kHz} = 8 / (1 + \text{SMPLRT_DIV}) \Rightarrow \text{SMPLRT_DIV} = 7 \text{ (decimal)} \Rightarrow \text{pData} = 0x07 \text{ (Hex)}.$

- Tắt bộ lọc số và cấu hình cho nguồn xung ngoại cung cấp cho MPU6050:

Dựa vào phần 2.1.2. Thanh ghi 26, tắt bộ lọc số DLPF_CFG = 0 và chọn xung nội làm nguồn cung cấp xung cho MPU6050 EXT_SYNC_SET = 0 => pData = 0x00.

- Cấu hình cho con quay hồi chuyển (Gyroscope):

Chọn thang đo ± 250 o/s => FS_SEL = 0 được set ở bit 3 - 4 => 0000 0000

=> pData = 0x00 (Hex).

- Cấu hình cho gia tốc kế (Accelerometer):

Chọn thang đo $\pm 2g$ => AFS_SEL = 0 được set ở bit 3 - 4 => 0000 0000

=> pData = 0x00 (Hex).

- Cấu hình cho lệnh ngắt Interrput:

Chỉ bật duy nhất chế độ ngắt Data Ready, bit 1 được set lên 1 => 0000 0001

=> pData = 0x01 (Hex).

- Cấu hình quản lý năng lượng:

Chọn con quay hồi chuyển trực X làm bộ nguồn xung clock cho MPU6050 (được khuyến cáo) => 0000 0001 => pData = 0x01 (hex).

- Đọc giá trị đầu ra:

- Đọc giá trị từ thanh ghi:

HAL_I2C_Mem_Read(&hi2c2, MPU6050_ADDR, ACCEL_XOUT_H_REG, 1, Rec_Data, 6, 1000);

Với Rec_Data là dãy 6 giá trị (Rec_Data[6]). Khi đọc từ thanh ghi ACCEL_XOUT_H_REG, lệnh Read sẽ tự động các thanh ghi liên kề với từng giá trị được gán vào 1 biến trong dãy Rec_Data[6].

- Xử lý dữ liệu thô

Vì mỗi gia tốc của mỗi trục đo được 1 giá trị 16 bit mà lệnh HAL_I2C_Mem_Read chỉ đọc được giá trị 8 bit từ các thanh ghi, cho nên cần ghép 8 bit đầu và 8 bit cuối từ mỗi gia tốc để của một trục nào đó thành 1 giá trị 16 bit hoàn chỉnh

Accel_X_RAW = (int16_t)(Rec_Data[0]<<8 | Rec_Data[1]);

Accel_Y_RAW = (int16_t)(Rec_Data[2]<<8 | Rec_Data[3]);

Accel_Z_RAW = (int16_t)(Rec_Data[4]<<8 | Rec_Data[5]);

Sau đó cần chia với thang đo tương ứng để được giá trị thực:

Ax = (float)Accel_X_RAW/2048.0;

Ay = (float)Accel_Y_RAW/2048.0;

Az = (float)Accel_Z_RAW/2048.0;

- Tính góc quay quanh trục X (roll), góc quay quanh trục Y (pitch)

$$rollX = \text{atan2}\left(\frac{A_y}{\sqrt{A_x^2 + A_z^2}}\right) \times \frac{180}{\pi}$$

$$pitchY = \text{atan2}\left(\frac{A_x}{\sqrt{A_y^2 + A_z^2}}\right) \times \frac{180}{\pi}$$

4.2.1.4. Cảm biến siêu âm đo khoảng cách vật cản

Giới thiệu về cảm biến siêu âm

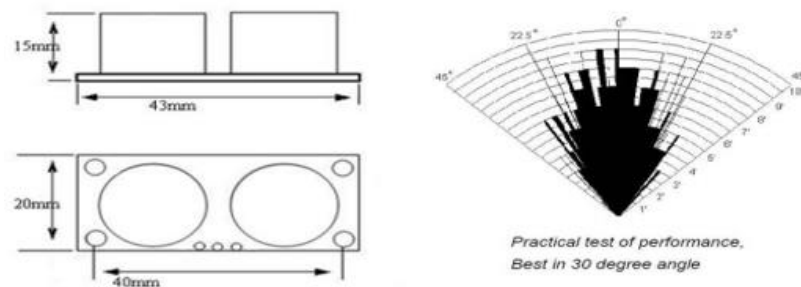
Thiết bị cảm biến siêu âm là thiết bị đo nhỏ gọn chứa máy phát siêu âm và mô-đun điện tử dùng để đo liên tục hoặc cảm nhận mức giới hạn của chất lỏng, khoảng cách vật cản, nước thải, bùn, chất kết dính, nhựa trong các mạch đóng và mở khác nhau,...



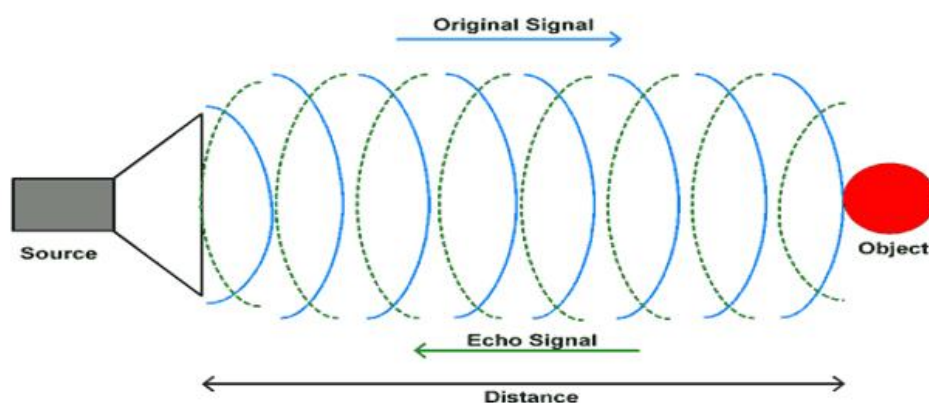
Hình 4.8 US-015

Thông số kỹ thuật:

- Điện thế hoạt động: DC 5V
- Dòng tiêu thụ: 2.2mA
- Khoảng nhiệt độ hoạt động: 0 - 70°C
- Chuẩn tín hiệu: TTL
- Góc quét: < 15 độ
- Khoảng cách nhận: 2 ~ 400cm
- Độ chính xác: 0.3cm + 1%.

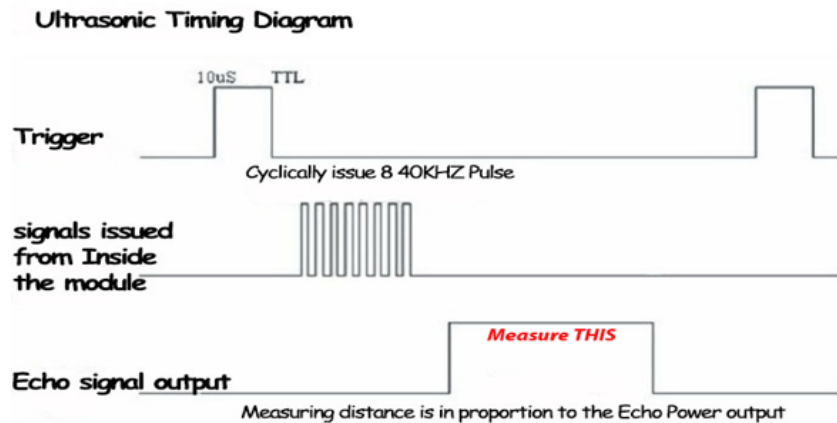
**Nguyên lý hoạt động của cảm biến:**

Hoạt động theo nguyên tắc sonar hoặc radar. Cảm biến siêu âm phát ra siêu âm, siêu âm sau khi chạm vào vật cản sẽ phản xạ lại và được thu hồi bởi cảm biến. Ta có thể đo khoảng cách bằng cách xác định thời gian từ lúc phát đến lúc nhận được phản xạ.



- Trên chân Trig ta sẽ phát 1 xung rất ngắn độ rộng khoảng 10μs.
- Tiếp theo đó từ bộ phát của cảm biến sẽ phát ra chuỗi xung 8 chu kỳ có tần số 40KHZ đồng thời 1 xung ở chân Echo sẽ nhảy từ mức thấp lên cao và sẽ tiếp tục ở mức cao cho đến khi chạm vật cản và trở về thì nó sẽ nhảy xuống mức thấp.

- Lúc này động rộng của xung sẽ bằng với thời gian song siêu âm được phát từ cảm biến và phản xạ lại.



Trong không khí, tốc độ âm thanh đạt mức 340 m/s (hằng số), tương đương với 29,412 cm/ μ s (106 / (340*100)).

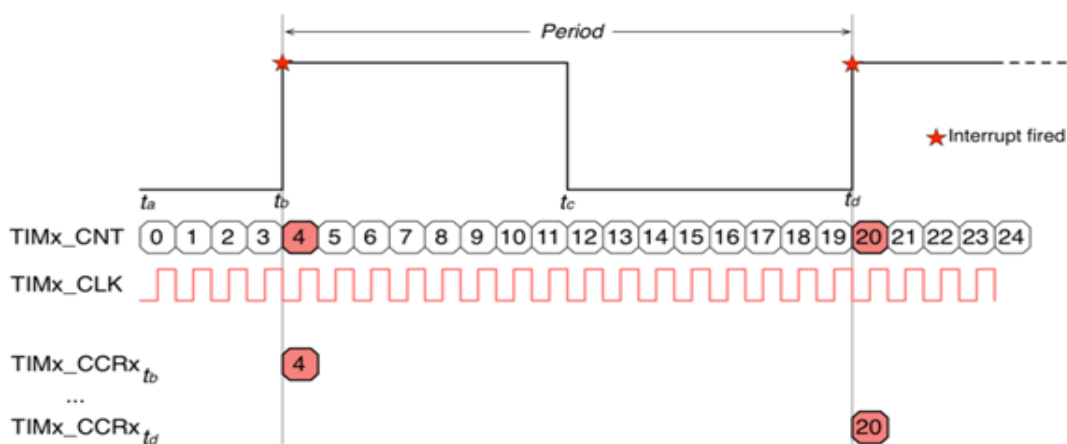
Khi đã tính được thời gian, ta sẽ chia cho 29,412 để ra giá trị khoảng cách.

Đo khoảng cách cảm biến bằng chế độ Input Capture

Như đã đề cập ở trên module sẽ gửi tín hiệu ở mức cao tỉ lệ với khoảng cách đo được. Tín hiệu này sẽ duy trì ở mức cao trong vòng vài micro giây, vì vậy ta sẽ sử dụng chế độ InputCapture với Interrupts trong Timer2 để đo khoảng cách chính xác hơn. Phương pháp này sẽ tránh các lỗi thường gặp đối với các phương pháp cũ.

Input Capture là gì?

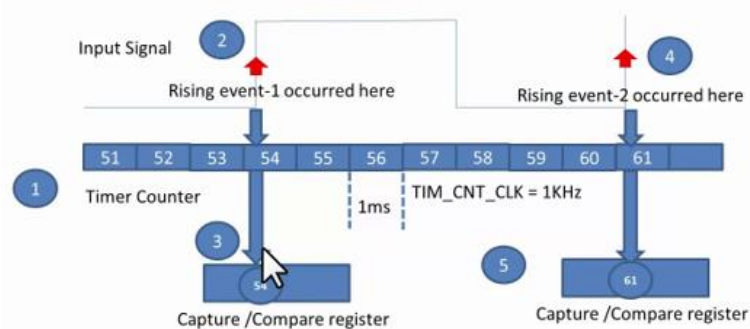
Input Capture là chế độ bắt sườn đầu vào ở chế độ này, timer sẽ bắt sự kiện xung từ bên ngoài vào, cụ thể là bắt sườn lên hoặc sườn xuống của xung, thường ứng dụng trong các hoạt động đo tần số hoặc độ rộng xung.



STM32 Timer chế độ Input Capture

Input Capture có 2 chế độ là bắt xung sườn lên (Rising) hoặc sườn xuống (Falling)
Nguyên lý hoạt động như sau:

- Sau khi khởi động Timer, mỗi khi có xung sườn lên (hoặc xuống) tại đầu vào Channelx của Timer. Thanh ghi TIMx_CCRx (x là Timer số 1,2,3,4...) sẽ được nạp giá trị của thanh ghi Counter TIMx_CNT.
- Bit CC1IF được đặt lên 1 (Cờ ngắt), Bit CC1OF sẽ được set lên 1 nếu 2 lần cờ liên tiếp cờ CC1IF được đặt lên 1 mà không xóa. Nghĩa là mỗi khi có sự kiện xung lên sẽ có ngắt xảy ra, lập trình viên phải đọc giá trị lưu vào CCR1 sau đó xóa cờ CC1IF, nếu không cờ CC1OF sẽ được bật lên báo tràn và dữ liệu sẽ không được ghi vào nữa.
- Một sự kiện ngắt được sinh ra nếu Bit CC1IE được đặt lên 1.
- Một sự kiện DMA sinh ra nếu Bit CC1DE được đặt lên 1.



Thuật toán

Sử dụng chế độ Input Capture để chụp giá trị cạnh lên ở mức cao (High) và giá trị cạnh xuống ở mức thấp (Low)

- Khoảng thời gian đầu tiên khi phát hiện vật sẽ được ghi lại khi chân Echo bật lên mức cao giá trị Timer sẽ được ghi lại vào thanh ghi CCR1 và chúng ta sẽ có khoảng thời gian T1
- Sau đó sẽ kích hoạt một tín hiệu ngắt để chụp ở trên cạnh rơi
- Khi chân Echo ở mức thấp giá trị Timer cũng sẽ được ghi lại ở trên mép cạnh rơi ở mức thấp và chúng ta sẽ có khoảng thời gian T2
- Độ rộng xung được tính bằng khoảng thời gian chênh lệch $T = T2 - T1$
- Khi đã tính được thời gian, ta sẽ nhân cho $29,412 \text{ cm}/\mu\text{s}$ để ra giá trị khoảng cách.

4.2.1.5. Acquy

- Bình Acquy GS GTZ5S được sản xuất tại tập đoàn sản xuất ắc quy GS Nhật Bản nổi tiếng thế giới.

- Loại bình: VRLA, siêu kín siêu bền, miễn bảo dưỡng.

- Ưu điểm Acquy GS: tuổi thọ lâu dài, chất lượng ổn định, thiết kế phù hợp với điều kiện khí hậu tại Việt Nam.

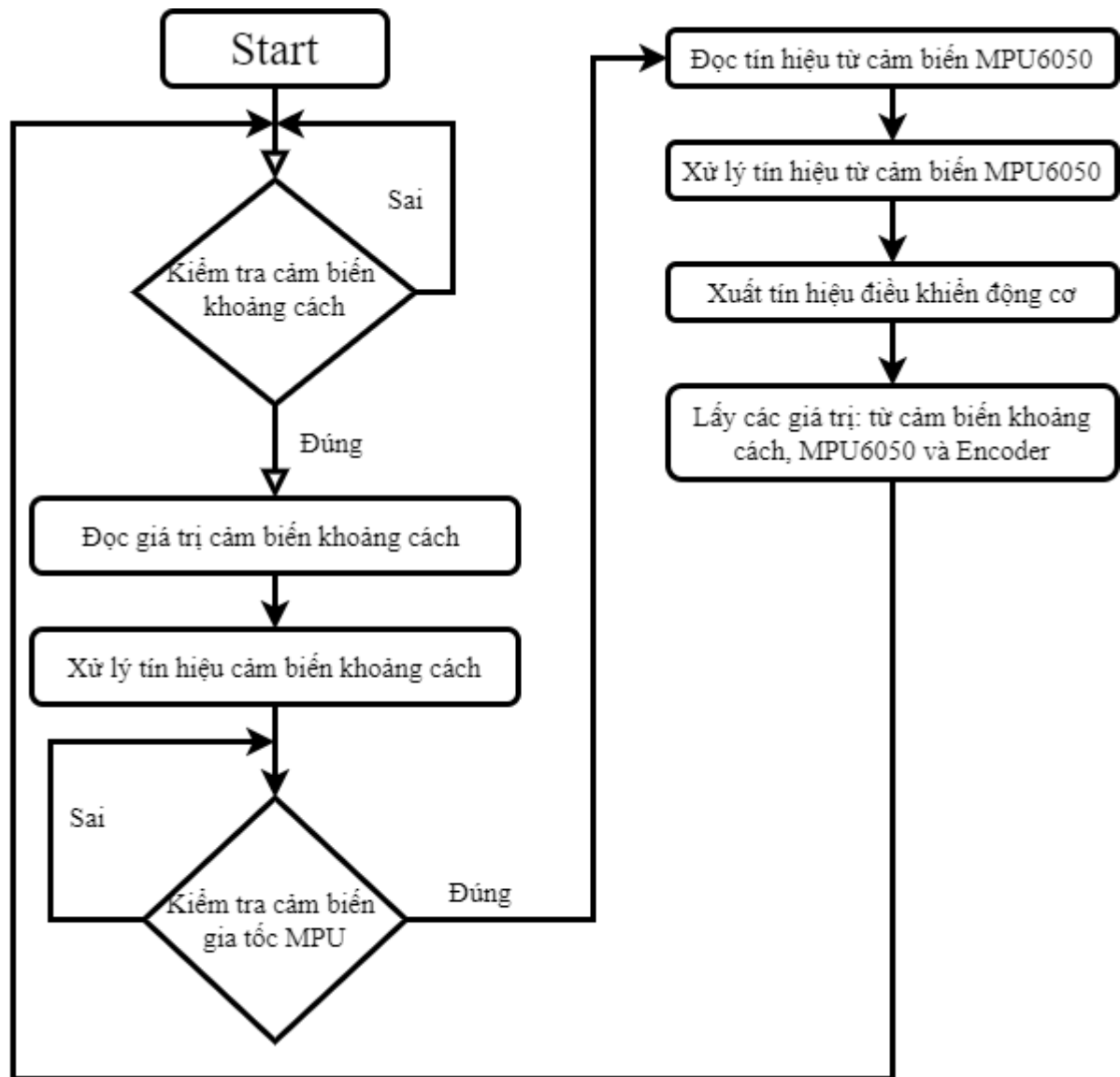
- Dung lượng: 12V – 3.5Ah (10 hours). - Kích thước (mm): 112 x 70 x 85 (dài x rộng x cao).

4.2.2. An toàn điện

Để đảm bảo an toàn điện cho hệ thống và bảo vệ cho các thiết bị điện tránh hư hỏng, nhóm bố trí các mạch ổn áp ở đầu vào của thiết bị điện. Khi gặp sự cố, mạch sẽ tự ngắt. Nhờ vậy đảm bảo an toàn cho hệ thống robot.

CHƯƠNG 5: XÂY DỰNG GIẢI THUẬT ĐIỀU KHIỂN

5.1. Tổng quan sơ đồ điều khiển

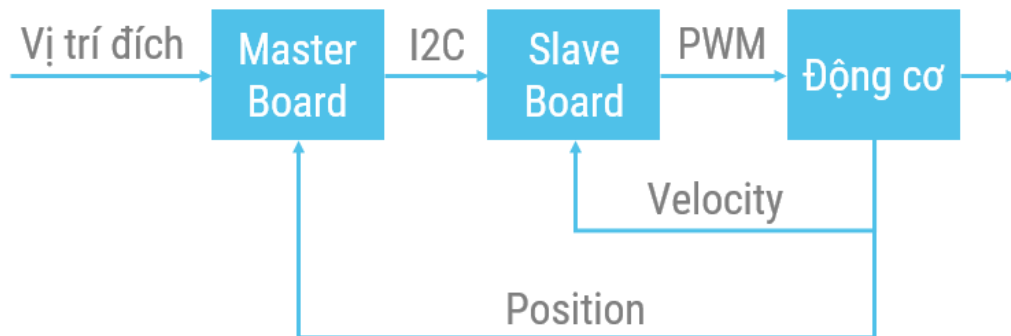


Theo như sơ đồ khối hệ thống sẽ lấy được các thông số cần thiết để có thể xác định được đúng vị trí di chuyển của Mobile Robot gồm: Khoảng cách tránh vật cản, toàn độ X, Y và góc quay của Robot, Quãng đường Robot di chuyển được từ encoder.

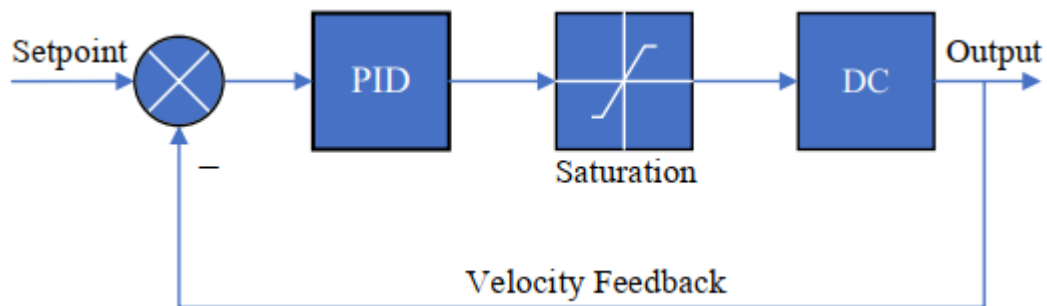
5.2. Điều khiển động cơ

Để điều khiển động cơ ổn định ở tốc độ mong muốn, nhóm sử dụng một aSMD350 cho mỗi động cơ. Các Driver giao tiếp với nhau qua giao thức I2C. Bộ điều khiển sẽ tính toán ra các giá trị vận tốc. Mạch điều khiển gửi giá trị của vận tốc về bộ Master. Driver nhận được dữ liệu vận tốc tiến hành điều khiển tốc độ động cơ và đọc giá trị

encoder để phản hồi vận tốc. Driver sử dụng bộ điều khiển PID để điều khiển động cơ. Giá trị vận tốc tính được từ encoder sẽ được đưa về mạch Master để tính toán vị trí của Robot và Driver đọc để điều khiển PID cho động cơ.



Hình 5. 2 Sơ đồ khối điều khiển động cơ

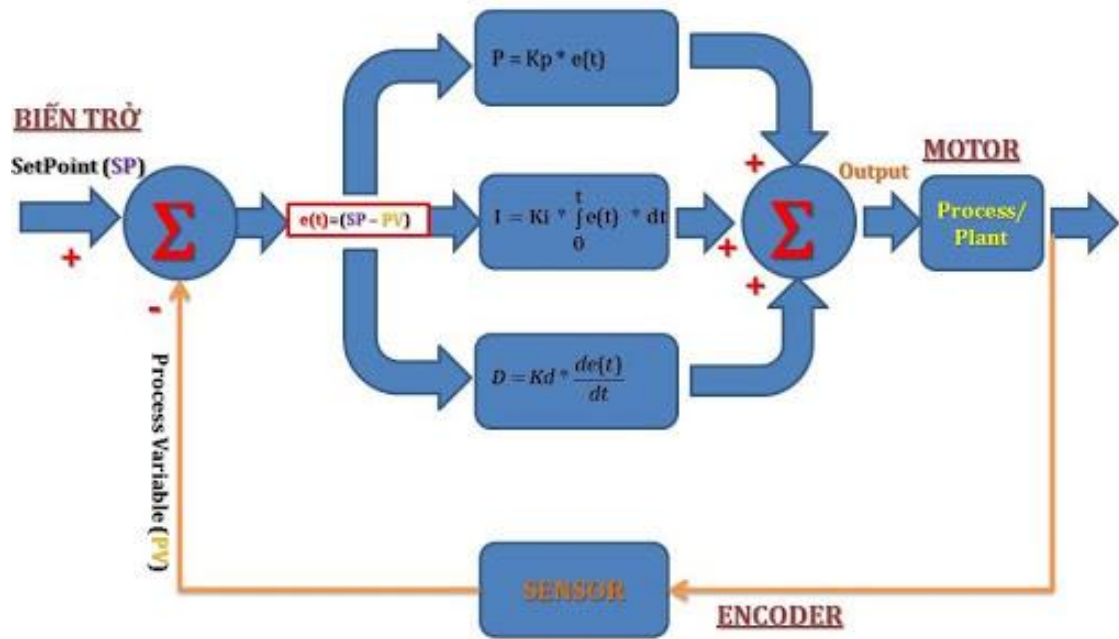


Hình 5. 3 Giải thuật điều khiển động cơ PID

5.2.1. Lý thuyết về bộ điều khiển PID:

Định nghĩa:

PID là cách viết tắt của các từ Proportional (tỉ lệ), Integral (tích phân) và Derivative (đạo hàm). Tuy xuất hiện rất lâu nhưng đến nay PID vẫn là giải thuật điều khiển được dùng nhiều nhất trong các ứng dụng điều khiển tự động. Bộ điều khiển PID là một bộ điều khiển vòng kín được sử dụng rộng rãi trong hệ thống điện, hệ thống tự động, điện tử. Mục tiêu của bộ điều khiển PID là điều chỉnh giá trị điều khiển ở ngõ ra Output sao cho sai lệch Error $e(t) = (SP - PV)$ giữa giá trị đo được của hệ thống PV (Process Variable) với giá trị cài đặt SP (SetPoint) nhỏ nhất có thể (~ 0), đạt được sự ổn định và có đáp ứng nhanh.



Giải thuật PID số:

$$u_k = u_k^P + u_k^D + u_k^I$$

Khâu P : $u_k^P = K_p e_k$

Khâu D : $u_k^D = K_D \left(\frac{e_k - e_{k-1}}{T} \right)$ với T : Thời gian lấy mẫu.

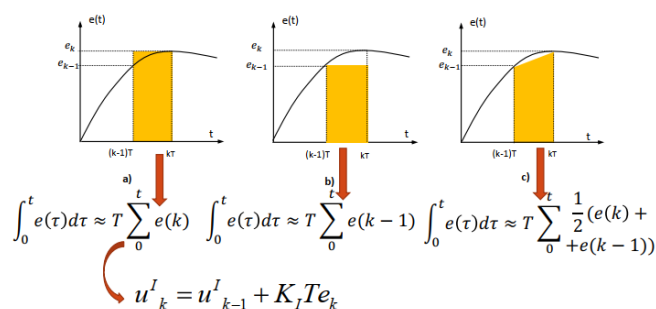
Khâu I:

Có 3 cách:

Tích phân lùi: Dùng e_k kéo về tính hình chữ nhật.

Tích phân tiến: Dùng e_{k-1} kéo về tính hình chữ nhật.

Tích phân hình thang: Nối e_k và e_{k-1} lại với nhau và tính hình thang.



Giải thích thư viện `pid_controller.c` và `pid_controller.h`:

Vì là một thư viện đã được các chuyên gia tạo ra nên thuật toán ở đây chúng em chỉ giải thích trên phương diện những gì mình biết không được đầy đủ chỉ giải thích những hàm mà chúng em sử dụng.

Trong tệp pid.controller.h các biến được định nghĩa và tạo các kiểu dữ liệu để tệp pid_controller.c xử lý. Một số dạng chính được định nghĩa:

```
typedef enum
{
    MANUAL,      // PID tắt đi cho người dùng tùy chỉnh output
    AUTOMATIC    // PID mở lên tính và ra output
}
PIDMode;

typedef enum
{
    DIRECT,      // Quay thuận (CW)
    REVERSE     // Quay nghịch (CCW)
}
PIDDirection;

typedef struct
{
    //
    // Input to the PID Controller
    //
    float input; // Input vào PID

    //
    // Previous input to the PID Controller
    //
    float lastInput; // Input trước đó của PID

    //
    // Output of the PID Controller
    //
    float output; // Output của PID

    // Gain constant values that were passed by the user
    // These are for display purposes
    //
    float dispKp;
    float dispKi;
    float dispKd;

    //
    // Gain constant values that the controller alters for
    // its own use
    //
    float alteredKp;
    float alteredKi;
    float alteredKd;

    //
    // The Integral Term
    //
    float iTerm;
```

```
//
// The interval (in seconds) on which the PID controller
// will be called
//
float sampleTime;

//
// The values that the output will be constrained to
//
float outMin;
float outMax;

//
// The user chosen operating point
//
float setpoint;

//
// The sense of direction of the controller
// DIRECT: A positive setpoint gives a positive output
// REVERSE: A positive setpoint gives a negative output
//
PIDDirection controllerDirection;

//
// Tells how the controller should respond if the user has
// taken over manual control or not
// MANUAL: PID controller is off.
// AUTOMATIC: PID controller is on.
//
PIDMode mode;
```

Một số dạng biến trên được định nghĩa bằng C++ và tệp pid_controller.c sẽ xử lý một số vấn đề có liên quan đến PID như sau.

Hàm này dùng để giới hạn giá trị lại.

```
//
#define CONSTRAIN(x,lower,upper) ((x)<(lower)?(lower):((x)>(upper)?(upper):(x)))
```

PIDInit sẽ chọn các giá trị cho PID

PIDControl *pid : Chọn điều khiển vị trí hay là điều khiển tốc độ.

float kp, **float** ki, **float** kd: Chọn thông số cho bộ điều khiển.

float sampleTimeSeconds: Chọn thời gian lấy mẫu.

float minOutput, **float** maxOutput: Giới hạn đầu ra và đầu vào.

PIDMode mode: Chọn Manual hay là Automatic.

PIDDirection controllerDirection chọn chiều cho động cơ.

```

void PIDInit(PIDControl *pid, float kp, float ki, float kd,
             float sampleTimeSeconds, float minOutput, float maxOutput,
             PIDMode mode, PIDDirection controllerDirection)
{
    pid->controllerDirection = controllerDirection;
    pid->mode = mode;
    pid->iTerm = 0.0f;
    pid->input = 0.0f;
    pid->lastInput = 0.0f;
    pid->output = 0.0f;
    pid->setpoint = 0.0f;

    if(sampleTimeSeconds > 0.0f)
    {
        pid->sampleTime = sampleTimeSeconds;
    }
    else
    {
        // If the passed parameter was incorrect, set to 1 second
        pid->sampleTime = 1.0f;
    }

    PIDOutputLimitsSet(pid, minOutput, maxOutput);
    PIDTuningsSet(pid, kp, ki, kd);
}

```

Các giá trị sẽ được trả vào các giá trị của chúng. Trong đó có hàm PIDtuningsSet nhiệm vụ của hàm này là sẽ lấy các giá trị đã được set:

```

pid->dispKp = kp;
pid->dispKi = ki;
pid->dispKd = kd;

```

Sau xử lý theo PID số:

```

pid->alteredKp = kp;
pid->alteredKi = ki * pid->sampleTime;
pid->alteredKd = kd / pid->sampleTime;

```

Và xử lý hướng cho động cơ.

```

void
PIDTuningsSet(PIDControl *pid, float kp, float ki, float kd)
{
    // Check if the parameters are valid
    if(kp < 0.0f || ki < 0.0f || kd < 0.0f)
    {
        return;
    }

    // Save the parameters for displaying purposes
    pid->dispKp = kp;
    pid->dispKi = ki;
    pid->dispKd = kd;

    // Alter the parameters for PID
    pid->alteredKp = kp;
    pid->alteredKi = ki * pid->sampleTime;
    pid->alteredKd = kd / pid->sampleTime;

    // Apply reverse direction to the altered values if necessary
    if(pid->controllerDirection == REVERSE)
    {
        pid->alteredKp = -(pid->alteredKp);
        pid->alteredKi = -(pid->alteredKi);
        pid->alteredKd = -(pid->alteredKd);
    }
}

```

Hàm PIDCompute là hàm quan trọng nhất trong thư viện hàm này sẽ tính toán và cho ra output.

`error = (pid->setpoint) - (pid->input);` : Tính sai số giữa giá trị đo của hệ thống và giá trị đặt.

`pid->iTerm += (pid->alteredKi) * error;` : xử lý khâu I bằng cách cộng dồn giá trị như lý thuyết của PID số ở trên.

`pid->iTerm = CONSTRAIN((pid->iTerm), (pid->outMin), (pid->outMax));` : Giải quyết vấn đề giới hạn giá trị của khâu I.

`dInput = (pid->input) - (pid->lastInput);` : Ở bộ điều khiển của hệ thống này thư viện sử dụng đạo hàm của input thay vì sử dụng đạo hàm của sai số (error) như PID số.

`pid->output = (pid->alteredKp) * error + (pid->iTerm) - (pid->alteredKd) * dInput;` : Tính ra OUTPUT bằng các giá trị trên công cả ba khâu P,D,I dạng rõ ràng hơn của phương trình trên như sau:

Khâu P: `(pid->alteredKp) * error = kp * error`

Khâu I: `(pid->iTerm) = (pid->iTerm) + ki * pid->sampleTime * error`

Khâu D: `(pid->alteredKd) * dInput = kd / pid->sampleTime * dInput`

Chỉ khác phương trình trên lý thuyết là khâu D thay vì dùng đạo hàm của sai số ta sử dụng đạo hàm của Input.

```
bool
PIDCompute(PIDControl *pid)
{
    float error, dInput;

    if(pid->mode == MANUAL)
    {
        return false;
    }

    // The classic PID error term
    error = (pid->setpoint) - (pid->input);

    // Compute the integral term separately ahead of time
    pid->iTerm += (pid->alteredKi) * error;

    // Constrain the integrator to make sure it does not exceed output bounds
    pid->iTerm = CONSTRAIN( (pid->iTerm), (pid->outMin), (pid->outMax) );

    // Take the "derivative on measurement" instead of "derivative on error"
    dInput = (pid->input) - (pid->lastInput);

    // Run all the terms together to get the overall output
    pid->output = (pid->alteredKp) * error + (pid->iTerm) - (pid->alteredKd) * dInput;

    // Bound the output
    pid->output = CONSTRAIN( (pid->output), (pid->outMin), (pid->outMax) );

    // Make the current input the former input
    pid->lastInput = pid->input;

    return true;
}
```

Xác định tọa độ và góc quay của Mobile Robot:

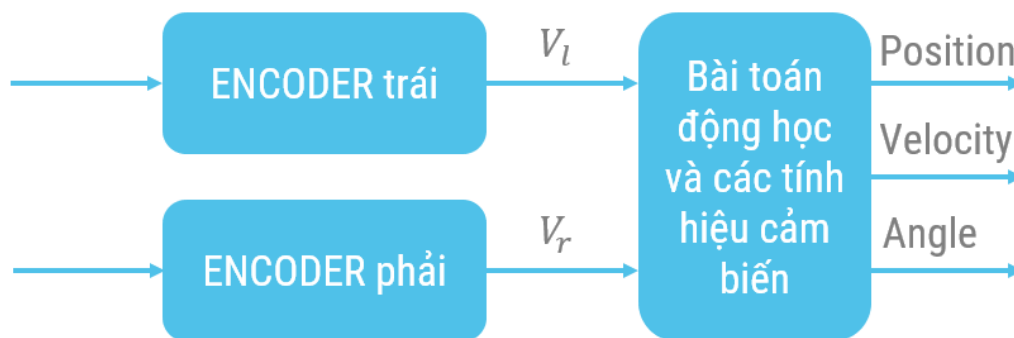
Định vị robot là vấn đề quan trọng trong các nghiên cứu về mobile robot. Ngày nay có nhiều phần cứng hiện đại và việc phát triển giải thuật tính toán dẫn đến tăng độ chính xác và tin cậy trong việc định vị robot. Nhìn chung định vị robot được chia làm ba phương pháp.

- Phương pháp thứ nhất là sử dụng các cảm biến gắn trên robot như: GPS, cảm biến gia tốc... Đặc điểm của phương pháp này có sai số lớn thích hợp với các ứng dụng ngoài trời.

- Phương pháp thứ hai là sử dụng các tia Laser (RP lidar), camera... Đặc điểm của phương pháp này là có độ chính xác cao phù hợp với các ứng dụng trong nhà nhưng chi phí cao và quá trình tính toán phức tạp.

- Phương pháp thứ ba là phương pháp được dùng trong hầu hết trên robot di động vì đơn giản và dễ thực hiện. Đó là sử dụng bộ mã hóa số vòng quay bánh xe (encoder).

Trong phạm vi đề tài và thời gian nghiên cứu có hạn nên nhóm em sẽ kết phương pháp thứ nhất và thứ ba để định vị Mobile Robot.



Hình 5. 4 Sơ đồ khối khác định vị trí của Mobile Robot trong không gian

5.2.2. Mô hình toán và chọn thông số PID

- **Mô hình toán:**

$$G(s) = \frac{\omega(x)}{U(s)} = \frac{K}{\tau_s + 1}$$

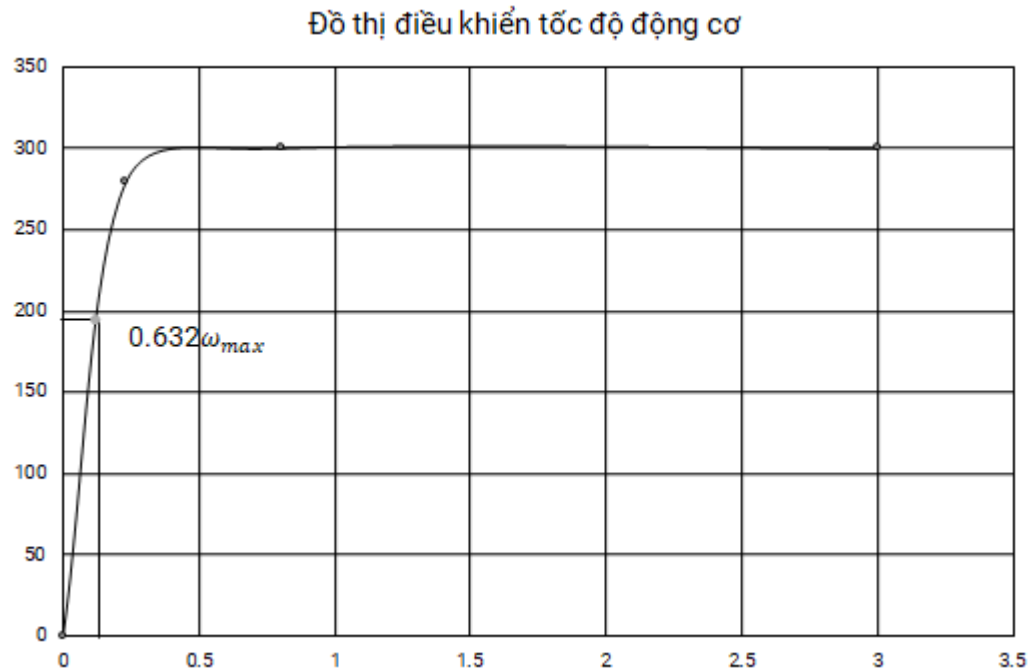
Với $\omega(x)$: vận tốc góc của Motor.

$U(s)$: Giá trị PWM xuất ra và $U_{\max} = 100$.

K : Độ lợi.

τ_s : Thời hằng.

Cần cung cấp Input tương đương PWM = 100% $\Leftrightarrow U = 100$ với $K = \frac{\omega_{max}}{U}$



Từ đồ thị lấy điểm $0.632\omega_{max} \Rightarrow \tau_s$ Từ đó ta tìm được hàm truyền của động cơ

Để điều khiển động cơ ta chỉ cần tính theo bộ điều khiển PI Controller theo hai công thức để tìm ra hệ số K_p và K_I .

$$G_c = K_p + \frac{K_I}{s}$$

$$G_c = K_c \left(1 + \frac{1}{\tau_I s}\right)$$

Áp dụng vào mô hình Mobile Robot:

- Theo đồ thị ta đo được $\omega_{max} = 299 \text{ m/s}$ khi tốc độ đặt là 300m/s

- Với $U = 100 \Rightarrow K = 2.99$

- $0.632\omega_{max} = 189 \text{ m/s} \Rightarrow \tau_s = 0.0688\text{s}$

- Theo công thức PI controller :

$$G_c = K_c \left(1 + \frac{1}{\tau_I s}\right) \Rightarrow K_c = \frac{\tau_s}{K\tau_c}$$

Với τ_c là thời gian đáp ứng mong muốn. Chọn $\tau_c = 0.1$

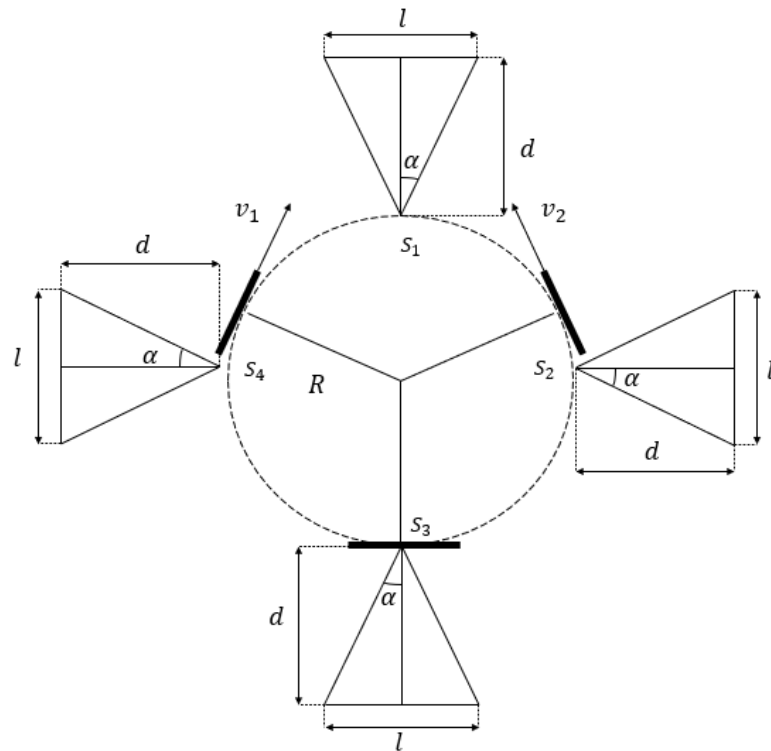
$$K_c = \frac{0.0688}{2.99 \cdot 0.1} = 0.23$$

Công thức PID :

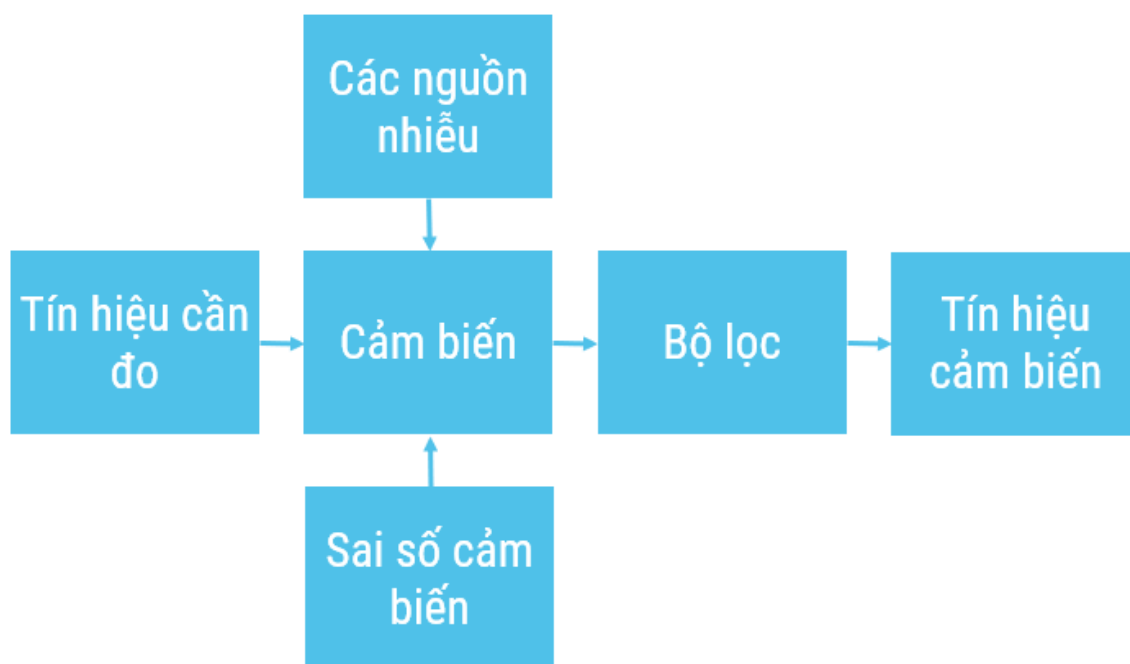
$$G_c = K_p + \frac{K_I}{s} \Rightarrow K_p = K_c = 0.23; K_i = \frac{K_c}{\tau_I} = 3.34$$

5.3. Phát hiện vật cản

Robot cần phát hiện vật cản trong quá trình di chuyển để tránh va chạm gây hư hỏng robot hoặc va chạm với con người làm tổn thương con người. Ngoài các vật cản đã biết trước khi tạo bản đồ, robot cũng có thể gặp phải các vật cản xuất hiện bất ngờ. Để phát hiện các vật cản này có nhiều phương pháp có thể sử dụng như lidar, camera, cảm biến radar, cảm biến siêu âm... Trong đề tài này nhóm sử dụng cảm biến siêu âm để phát hiện vật cản. Cảm biến siêu âm có đặc điểm ít bị nhiễu bởi môi trường, độ chính xác ở mức độ vừa phải, dễ lập trình điều khiển.



Hình 5. 5 Các góc quét của cảm biến siêu âm

5.4. Xử lý dữ liệu cho cảm biến gia tốc MPU6050

Như sơ đồ khối thì xử lý tín hiệu này được sử dụng cho cảm biến gia tốc MPU6050 để đảm bảo được giá trị chính xác nhất ta thu được.

Thuật toán bộ lọc Kalman rời rạc

Bộ lọc Kalman ước lượng một quá trình bằng cách sử dụng một dạng điều khiển phản hồi (Feedback Control): bộ lọc ước lượng trạng thái của quá trình tại một thời điểm sau đó có được phản hồi từ các số liệu đo đạc (có nhiễu). Như vậy, quá trình lọc của bộ lọc Kalman được chia thành hai bước:

Cập nhật theo thời gian (time update): Để dự đoán (Projecting Forward) trạng thái hiện tại và vector hiệp phương sai lỗi nhằm ước lượng trạng thái tiên nhiệm cho bước tiếp theo. Các phương trình cập nhật theo giá trị đo lường dùng để cung cấp 65 phản hồi. Ví dụ như kết hợp một giá trị đo lường mới với ước lượng tiên nhiệm để có được ước lượng trạng thái hậu nhiệm.

Cập nhật giá trị đo lường (Measurement Update): Các phương trình cập nhật theo giá trị đo lường dùng để cung cấp phản hồi. Chúng ta có thể coi các phương trình cập nhật theo thời gian là các phương trình dự đoán (Predictor Equations), trong khi đó các phương trình cập nhật theo giá trị đo lường là các phương trình sửa sai (Corrector Equations). Vậy thuật toán ước lượng cuối cùng tương tự như một thuật toán dự đoán - sửa sai để giải quyết các vấn đề về số học.

CHƯƠNG 6: KẾT QUẢ THỰC NGHIỆM

6.1. Kết quả thu được cảm biến gia tốc MPU6050

Kết quả góc X:

| Watch 1 | | |
|--------------------|-------|------|
| Name | Value | Type |
| Angle_X | 45 | uint |
| Angle_Y | 0 | uint |
| <Enter expression> | | |

Kết quả góc Y:

| Watch 1 | | |
|--------------------|-------|------|
| Name | Value | Type |
| Angle_X | 0 | uint |
| Angle_Y | 35 | uint |
| <Enter expression> | | |

Kết quả này thu được khi nghiêng góc $X = 45^\circ - Y = 0$ và $X = 0 - Y = 35^\circ$.

6.2. Kết quả thu được từ cảm biến khoảng cách US-015



Hình 6.1 Thực nghiệm US-015

- Kết quả cảm biến 1

| Name | Value | Type |
|-----------|-----------|-------|
| Distance | 8.4829998 | float |
| Distance1 | 0 | float |
| Distance2 | 0 | float |

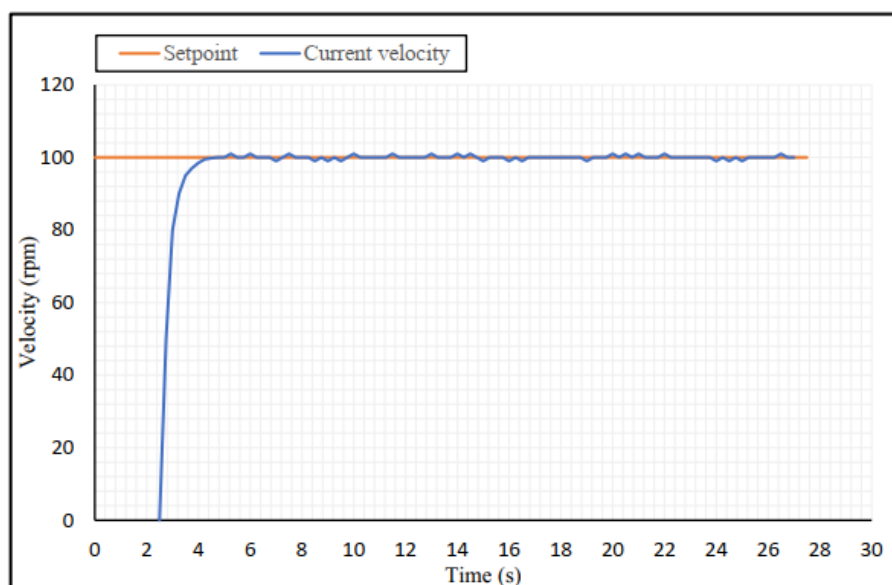
- Kết quả cảm biến 2

| | | |
|-----------|------------|-------|
| Distance | 0 | float |
| Distance1 | 8.15999985 | float |
| Distance2 | 0 | float |

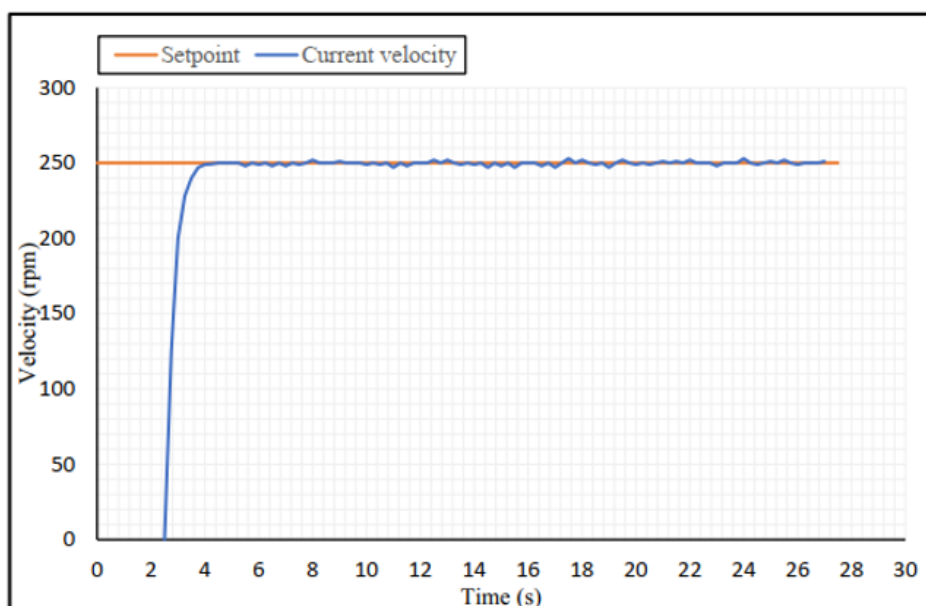
- Kết quả cảm biến 3

| Watch 1 | | |
|-----------|------------|-------|
| Name | Value | Type |
| Distance | 0 | float |
| Distance1 | 0 | float |
| Distance2 | 8.14299965 | float |

6.3. Kiểm tra tính ổn định của bộ PID



Hình 6.2 Đồ thị hiển thị vận tốc theo thời gian khi Setpoint là 100rpm



Hình 6.3 Đồ thị hiển thị vận tốc theo thời gian khi Setpoint là 250rpm

6.4. Thực nghiệm:

Giới hạn của việc thực nghiệm:

- Robot chỉ thực hiện công việc giám sát một khu vực hẹp.
- Robot còn nhiều xảy ra ở các cảm biến
- Chưa thực hiện được trên phần cứng

CHƯƠNG 7: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

7.1. Về lý thuyết

Về cơ bản, chúng em đã hoàn thành việc lựa chọn cấu hình robot, cơ cấu truyền động, cơ cấu tác động. Thiết kế được robot trên phần mềm Inventer.

Giải được bài toán động học của robot.

Viết được code điều khiển chuyển động robot, đọc các cảm biến hoạt động trên Robot.

7.2. Về thực nghiệm

Hoàn thành việc gia công các chi tiết và lắp ráp hoàn thiện robot.

Thu được các giá trị mong muốn từ các ngoại vi.

Chưa thực nghiệm điều khiển trên robot.

7.3. Ưu điểm, nhược điểm

Ưu điểm:

- Cơ cấu cứng vững, di chuyển tốt, đổi hướng ổn định.
- Hoạt động êm.
- Robot nhỏ gọn.

Nhược điểm:

- Độ chính xác chưa cao, trong quá trình di chuyển còn rung lắc
- Độ thẩm mỹ chưa cao.

7.4. Hướng phát triển

Đề tài có tính mở, có thể tiếp tục nghiên cứu phát triển để áp dụng vào thực tế, ứng dụng trong các môn khác khó hơn như phục vụ, chuyển hàng, ...

Có thể thiết kế gọn và đẹp hơn nữa.

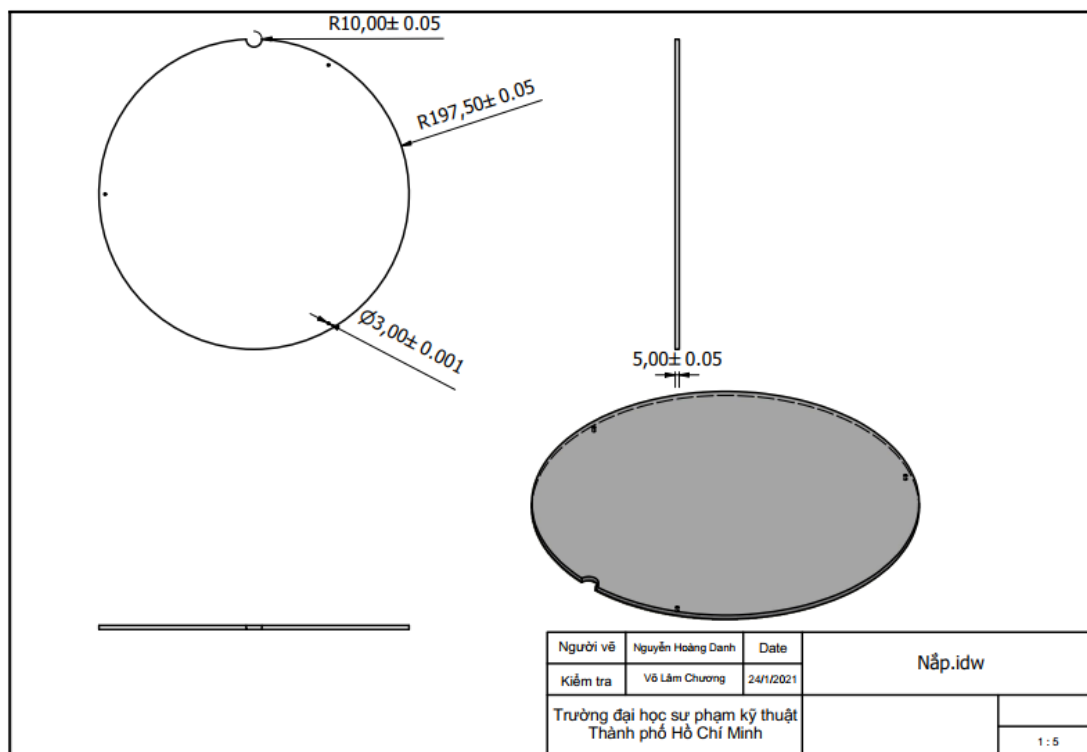
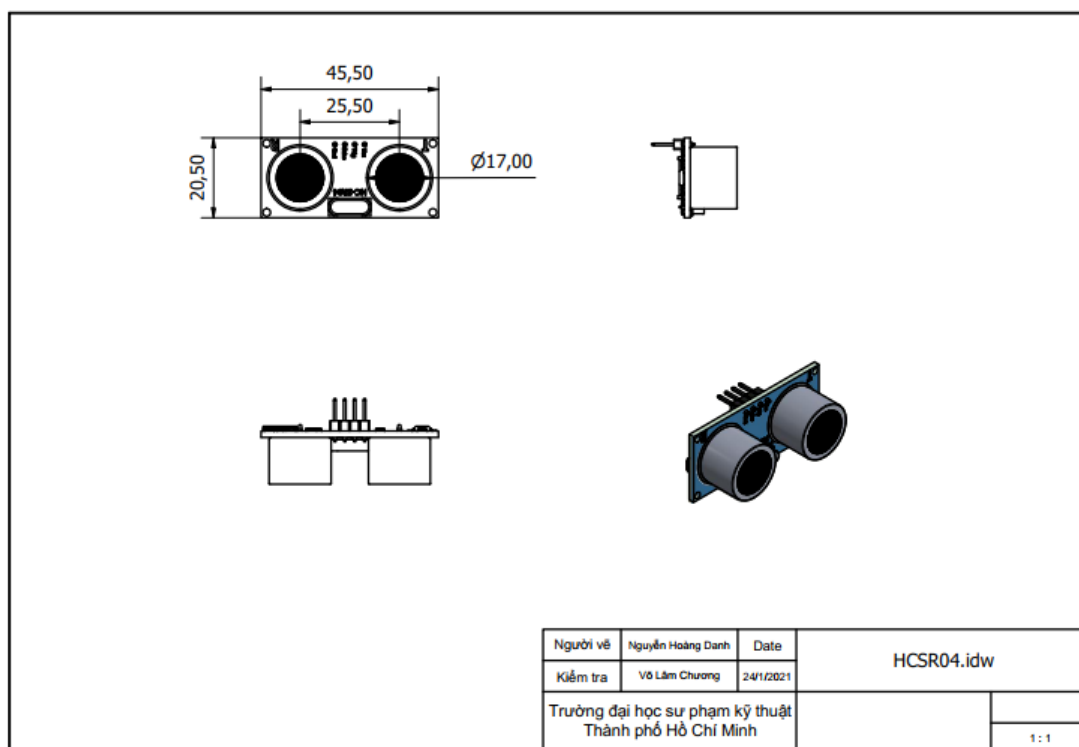
Có thể trang bị thêm cho robot các cảm biến khác, hệ thống quan sát,.. để robot chạy tự động. Tối ưu hóa code điều khiển hoặc bộ vi xử lý, để robot hoạt động nhanh nhạy và linh hoạt hơn.

TÀI LIỆU THAM KHẢO

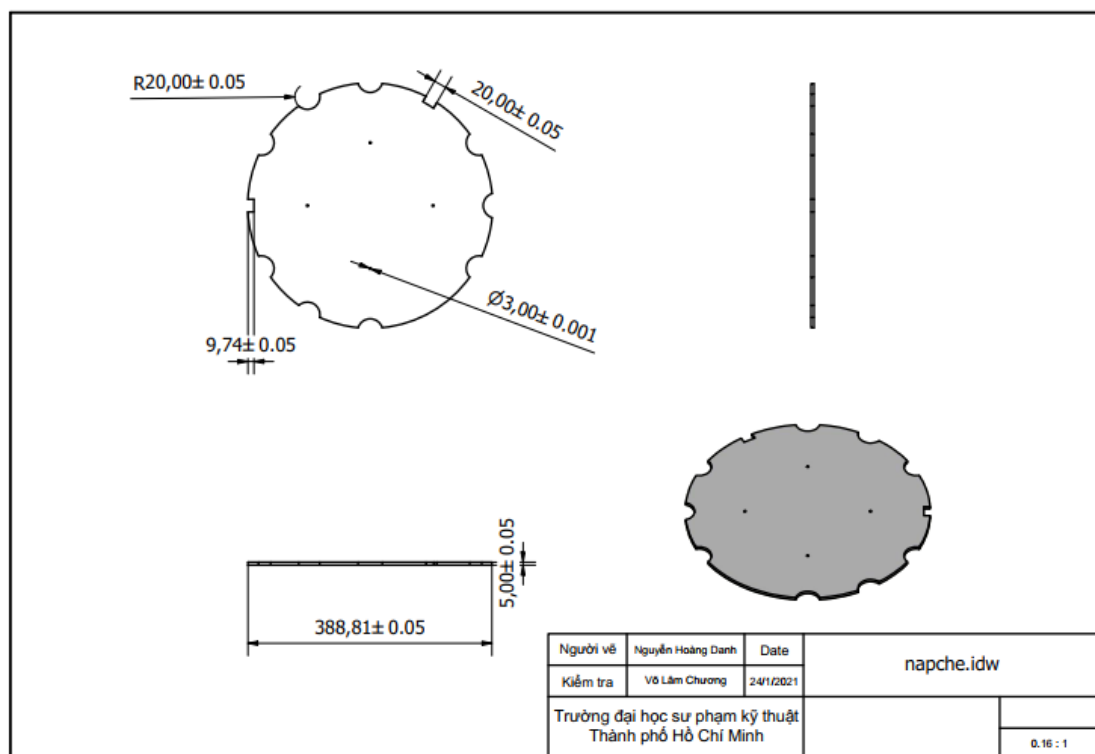
1. Mobile robot
https://en.wikipedia.org/wiki/Mobile_robot
2. Giới thiệu Mobile robot
https://baoanjsc.com.vn/tin-hang/mobile-robots-lagi_2_1_31453_vn.aspx
3. Bài tập lớn Tự động hóa: Mobile robot ba bánh - ba motor - CĐ Kỹ thuật Cao Thắng
<https://tailieu.vn/doc/bai-tap-lon-tu-dong-hoa-mobile-robot-ba-banh-ba-motor-cd-ky-thuat-cao-thang-1683320.html>
4. Đồ án tốt nghiệp: Mobile robot
<https://luanvan123.info/threads/mobile-robot-phuc-vu-ban.122472/>
5. Types and Applications of Autonomous Mobile Robots (AMRs)
<https://www.conveyco.com/types-and-applications-of-amrs/>
6. Tracked mobile robot:
<https://123doc.net/document/2320101-do-an-tot-nghiep-tracked-mobile-robot.htm>
7. The future of Smart Logistics Automation:
<https://www.roboticstomorrow.com/article/2020/10/the-future-of-smart-logistics-automation/15857/>
8. Analysis and Control of Mobile Robots in Various Environmental Conditions
<https://core.ac.uk/download/pdf/53189138.pdf>
9. A simulation and experimental study on wheeled mobile robot path control in road roundabout environment
<https://journals.sagepub.com/doi/pdf/10.1177/1729881419834778>
10. Sách Wheeled Mobile Robot 1st Edition

PHỤ LỤC

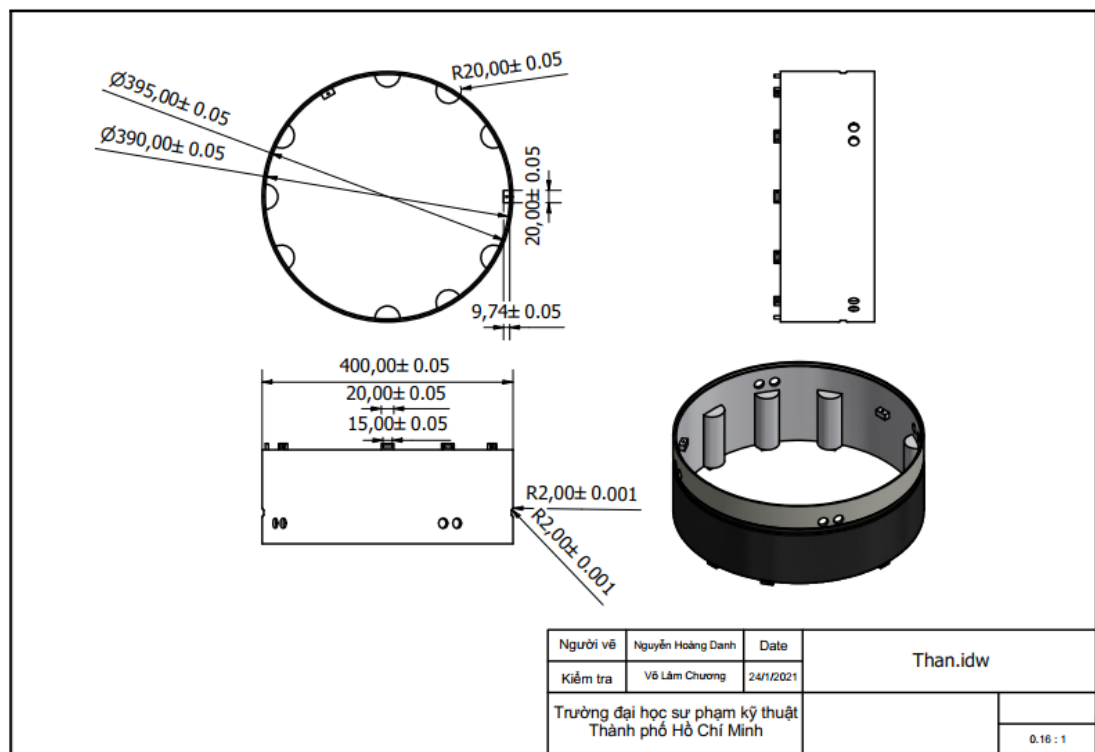
CHI TIẾT CƠ KHÍ MOBILE ROBOT

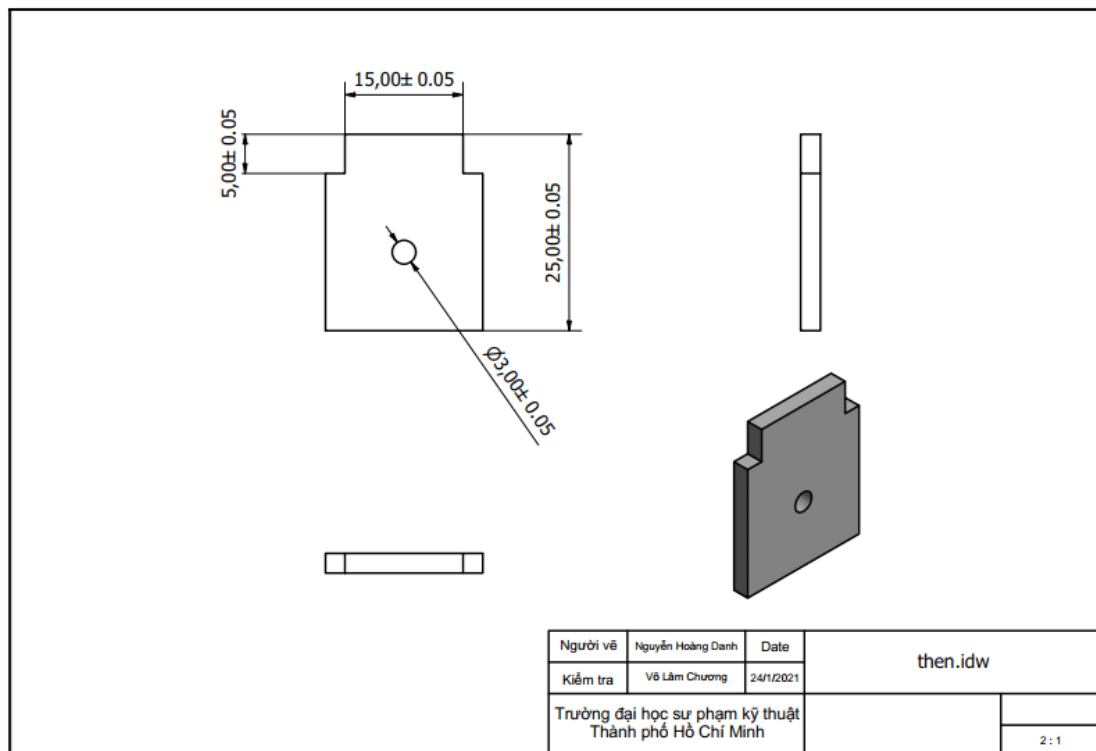
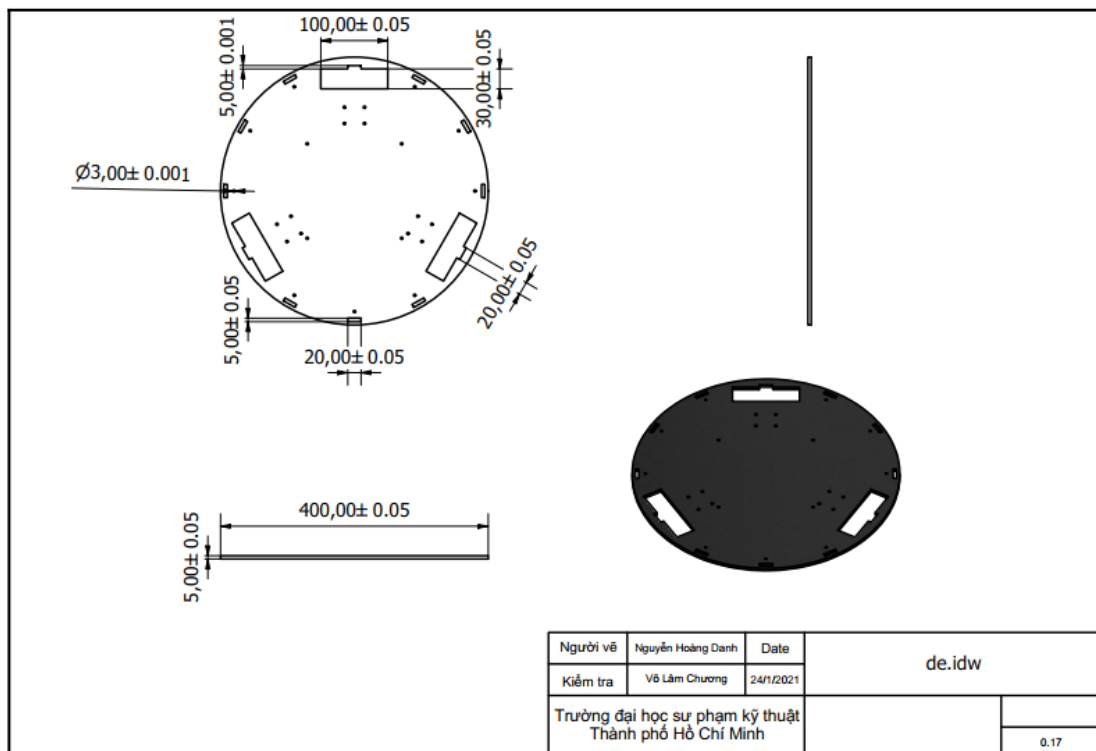
Chi tiết 1:**Chi tiết 3**

Chi tiết 4:

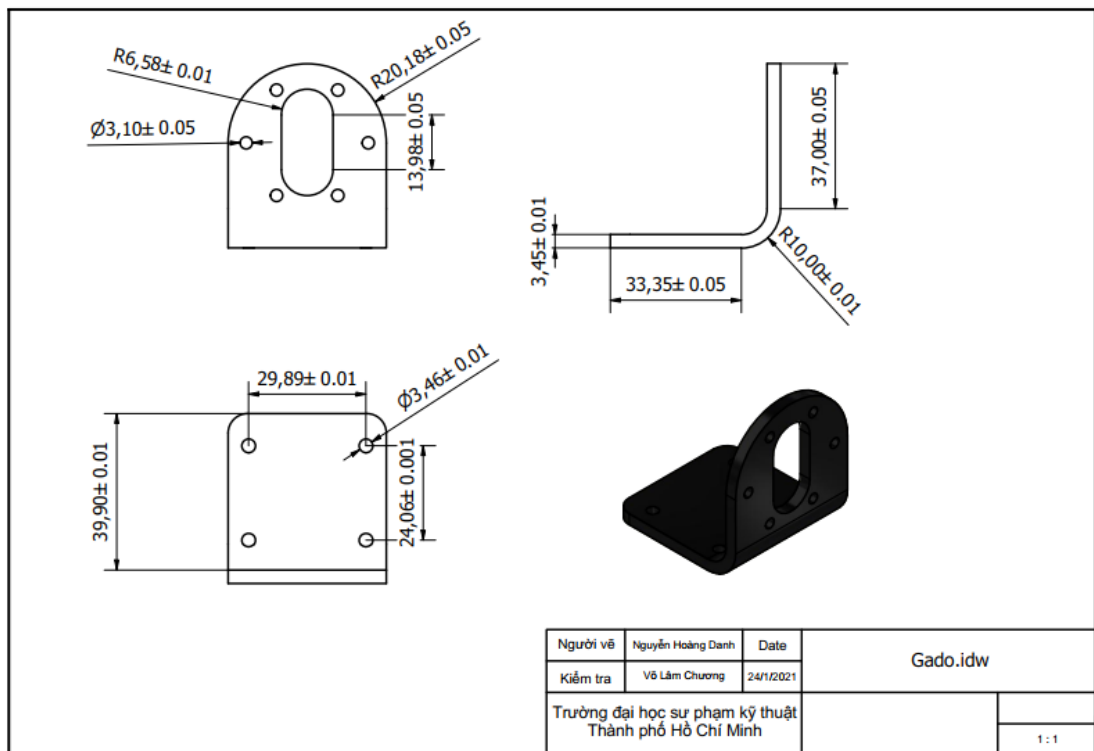


Chi tiết 6:



Chi tiết 7:**Chi tiết 8:**

Chi tiết 10:



Chi tiết 12:

