

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT
THÀNH PHỐ HỒ CHÍ MINH



CÔNG TRÌNH NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN

**HỆ THỐNG CÁNH TAY ROBOT 4 BẬC TỰ DO
ĐIỀU KHIỂN BẰNG MATLAB - SIMULINK**

MÃ SỐ: SV2021 - 116

CHỦ NHIỆM ĐỀ TÀI: NGUYỄN BÁ HUY



Tp. Hồ Chí Minh, tháng 10/2021

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐH SƯ PHẠM KỸ THUẬT TPHCM

BÁO CÁO TỔNG KẾT

ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN

**HỆ THỐNG CÁNH TAY ROBOT 4 BẬC TỰ ĐỘNG ĐIỀU KHIỂN BẰNG
MATLAB - SIMULINK**

SV2021- 116

Thuộc nhóm ngành khoa học : Kỹ thuật

SV thực hiện : Nguyễn Bá Huy

Nam, Nữ : Nam

Dân tộc : Kinh

Lớp, khoa : 17151CLA, Đào tạo chất lượng cao ; Năm thứ : 4/Số năm đào tạo : 4

Ngành học : Control and Automation Engineering Technology

Người hướng dẫn : Lê Thị Hồng Lam

Chức danh khoa học : Thạc sĩ

Học vị : Thạc sĩ

MỤC LỤC

MỞ ĐẦU.....	1
Tổng quan tình hình nghiên cứu thuộc lĩnh vực của đề tài.....	1
Lý do chọn đề tài.....	2
Mục tiêu đề tài.....	2
Cách tiếp cận, phương pháp nghiên cứu, phạm vi nghiên cứu.....	2
CHƯƠNG 1: TỔNG QUAN.....	3
1.1 Giới thiệu	3
1.2 Mục tiêu đề ra	4
1.3 Giới hạn đề tài.....	4
1.4 Nội dung đề tài	4
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT.....	6
2.1 Giới thiệu chương 2	6
2.2 Cánh tay robot	6
2.2.1 Khái niệm về cánh tay robot.....	6
2.2.2 Cấu trúc của cánh tay robot.....	6
2.2.3 Các loại cánh tay robot	7
2.3 Thuật toán điều khiển	9
2.3.1 Khái niệm về thuật toán điều khiển.....	9
2.3.2 Các loại thuật toán điều khiển	10
2.4 Phần mềm vận hành	13
2.4.1 Khái niệm	13
2.4.2 Các loại phần mềm vận hành.....	13
2.5 Lựa chọn cho đề tài	15
CHƯƠNG 3: THIẾT KẾ HỆ THỐNG	16
3.1 Giới thiệu chương 3	16
3.2 Phần cứng	16
3.2.1 Vị điều khiển	16
3.2.2 Động cơ DC tích hợp với bộ mã hóa.....	17
3.2.3 Bộ điều khiển động cơ.....	18
3.2.4 Nguồn cung cấp	19
3.2.5 Phần cánh tay robot in 3D	20
3.2.6 Ròng rọc	23
3.2.7 Sơ đồ nối dây phần cứng	24

3.3 Thiết kế phần mềm.....	24
3.3.1 Thiết kế bộ điều khiển PID.....	24
3.3.2 Phương pháp điều chỉnh bộ điều khiển PID.....	25
3.3.3 Ứng dụng bộ điều khiển PID để điều khiển hệ thống	26
CHƯƠNG 4: ĐỘNG HỌC VÀ QUY HOẠCH QUỸ ĐẠO.....	29
4.1 Tính toán động học của cánh tay robot.....	29
4.2 Quy hoạch quỹ đạo của cánh tay robot.....	32
CHƯƠNG 5: KẾT QUẢ ĐẠT ĐƯỢC.....	33
5.1 Phần cứng.....	33
5.2 Phần mềm.....	33
CHƯƠNG 6: ĐÁNH GIÁ KẾT QUẢ.....	36
KẾT LUẬN VÀ KIẾN NGHỊ.....	38
Kết Luận.....	38
Động học.....	38
Kiến nghị:	43
TÀI LIỆU THAM KHẢO.....	44
PHỤ LỤC	45
Code	45
Quy hoạch quỹ đạo	45
Động học nghịch.....	45
Động học thuận.....	45
Điều khiển L298	45
Điều khiển encoder L298	46
Mô phỏng hệ thống trên Matlab – Simulink.....	51

DANH MỤC HÌNH ẢNH

<i>Hình 1. 1 Kỳ vọng tăng trưởng trung hạn đối với robot công nghiệp trong năm 2019-2022 [1]</i>	3
<i>Hình 2. 1 Cánh tay robot Epson G1 Mini SCARA [2]</i>	7
<i>Hình 2. 2 Phó giáo sư Denny Oetomo đang thử nghiệm một cánh tay nhân tạo. Hình: David Caird / Herald Sun [3]</i>	8
<i>Hình 2. 3 Denso - cánh tay robot 4-DOF [3]</i>	9
<i>Hình 2. 4 Sơ đồ của bộ điều khiển PID</i>	11
<i>Hình 2. 5 Ví dụ về bộ điều khiển logic mờ [8]</i>	12
<i>Hình 2. 6 Sơ đồ của bộ điều khiển Mờ - PID</i>	13
<i>Hình 3. 1 Arduino Mega 2560 R3 Atmega [10]</i>	17
<i>Hình 3. 2 DC- động cơ mã hóa JGA25 [11]</i>	17
<i>Hình 3. 3 Bảng dữ liệu của nhiều loại động cơ mã hóa DC-JGA25 [11]</i>	18
<i>Hình 3. 4 Bộ điều khiển động cơ L298N trong thực tế [12]</i>	19
<i>Hình 3. 5 Nguồn điện 12V-DC 5A trong thực tế [13]</i>	19
<i>Hình 3. 6 Khớp nối thứ nhất của cánh tay robot trong Solidwork</i>	20
<i>Hình 3. 7 Khớp nối thứ nhất của cánh tay robot sau khi in ra</i>	20
<i>Hình 3. 8 Khớp nối thứ hai của cánh tay robot trong Solidwork</i>	21
<i>Hình 3. 9 Khớp nối thứ hai của cánh tay robot sau khi in ra</i>	21
<i>Hình 3. 10 Khớp nền của cánh tay robot trong Solidwork</i>	22
<i>Hình 3. 11 Khớp nền của cánh tay robot sau khi in ra</i>	22
<i>Hình 3. 12 Ròng rọc thực tế [13]</i>	23
<i>Hình 3. 13 Sơ đồ nối dây thực tế</i>	24
<i>Hình 3. 14 Bộ điều khiển PID cho hệ thống</i>	25
<i>Bảng 3. 1 Đặc điểm thông số bộ điều khiển PID [14]</i>	25
<i>Hình 3. 15 Thông số PID cho động cơ 1</i>	26
<i>Hình 3. 16 Sơ đồ bộ điều khiển PID động cơ 1 trong Matlab – Simulink</i>	26
<i>Hình 3. 17 Thông số PID cho động cơ 2</i>	27
<i>Hình 3. 18 Sơ đồ bộ điều khiển PID động cơ 2 trong Matlab – Simulink</i>	27
<i>Hình 3. 19 Lưu đồ giải thích hoạt động của hệ thống</i>	28
<i>Hình 4. 1 Động học ngược dựa vào hình học</i>	29
<i>Hình 4. 2 Trình điều khiển phẳng</i>	29
<i>Hình 5. 1 Cánh tay robot 2 bậc SCARA</i>	33
<i>Hình 5. 2 Góc 1, tín hiệu xung vuông và đáp ứng thực tế</i>	34
<i>Hình 5. 3 Góc 1, tín hiệu sóng sin và đóng ứng thực tế</i>	34

<i>Hình 5. 5 Góc 2, tín hiệu sóng sin và đóng ứng thực tế</i>	35
<i>Hình 5. 4 Góc 2, tín hiệu xung vuông và đáp ứng thực tế</i>	35
<i>Hình 6. 1 Ảnh quy hoạch quỹ đạo hình tròn theo lý thuyết, ảnh mục tiêu</i>	36
<i>Hình 6. 2 Ảnh đạt được của hệ thống khi hoạt động thực tế</i>	36

DANH MỤC BẢNG BIỂU

<i>Bảng 3. 1 Đặc điểm thông số bộ điều khiển PID [14]</i>	25
---	----

DANH MỤC CÁC TỪ VIẾT TẮT

DC	Direct Current (Dòng một chiều)
IDE	Integrated Development Environment (Môi trường phát triển tích hợp)
I/O	Input/Output (Vào/Ra)
ICSP	In-Circuit Serial Programming (Lập trình nối tiếp trong mạch)
MV	Measured value (giá trị đo được)
PWM	Pulse Width Modulation (Điều chế độ rộng xung)
PID	Proportional Integral Derivative (Đạo hàm - tích phân - tỷ lệ)
SCARA	Selective Compliance Assembly Robot Arm (Cánh tay robot lắp ráp tuân thủ có chọn lọc)
SP	Set point (điểm đặt)
SSR	Solid state relay (rơ le trạng thái rắn)
UART	Universal Asynchronous Receiver / Transmitter (Máy thu / Máy phát không đồng bộ đa năng)

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐH SƯ PHẠM KỸ THUẬT TPHCM

THÔNG TIN KẾT QUẢ NGHIÊN CỨU CỦA ĐỀ TÀI

1. Thông tin chung :

- Tên đề tài : Hệ thống cánh tay robot 4 bậc tự do điều khiển bằng Matlab - Simulink
- Chủ nhiệm đề tài : Nguyễn Bá Huy Mã số SV : 17151013
- Lớp : 17151CLA Khoa : Khoa đào tạo chất lượng cao
- Thành viên đề tài:

Số	Họ và tên	MSSV	Lớp	Khoa
1	Nguyễn Tuấn Thanh	17151028	17151CLA	Khoa đào tạo chất lượng cao
2	Nguyễn Lê Đức Duy	17151178	17151CLA	Khoa đào tạo chất lượng cao
3	Trương Nguyễn Quang Duy	17151007	17151CLA	Khoa đào tạo chất lượng cao
4	Lê Thành Nam	17142122	17151CLA	Khoa đào tạo chất lượng cao

- Người hướng dẫn: Lê Thị Hồng Lam

2. Mục tiêu đề tài:

- Thiết kế thi công và chế tạo cánh tay robot 4 bậc tự do đồng thời điều khiển chúng bằng phần mềm Matlab – Simulink.

3. Tính mới và sáng tạo:

- Điều khiển chuyển động robot bằng phần mềm Matlab - Simulink

4. Kết quả nghiên cứu:

- Mô hình cánh tay robot 2 bậc SCARA

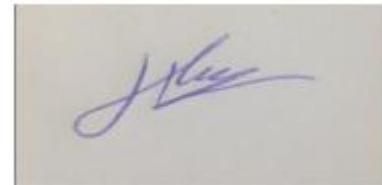
- Điều khiển chuyển động bằng Matlab - Simulink

5. Đóng góp về mặt giáo dục và đào tạo, kinh tế - xã hội, an ninh, quốc phòng và khả năng áp dụng của đề tài:

- Những công ty có thể ứng dụng cánh tay robot vào việc lắp ráp thiết bị trong hệ thống như VinFast, Samsung....

6. Công bố khoa học của SV từ kết quả nghiên cứu của đề tài (ghi rõ tên tạp chí nếu có) hoặc nhận xét, đánh giá của cơ sở đã áp dụng các kết quả nghiên cứu (nếu có):

Ngày tháng năm
SV chịu trách nhiệm chính
thực hiện đề tài
(kí, họ và tên)



Nguyễn Bá Huy

Nhận xét của người hướng dẫn về những đóng góp khoa học của SV thực hiện đề tài (phần này do người hướng dẫn ghi):

Điểm mạnh: Có kỹ năng phát triển hệ thống điều khiển tự động từ thiết kế, mô phỏng, chế tạo, cảm biến đến xây dựng phần mềm chất lượng cao.

Điểm yếu: Do diễn biến phức tạp của đại dịch COVID nên quá trình đánh giá chỉ được thực hiện với mô hình đơn giản. Tuy nhiên, kết quả tốt đạt được có thể khẳng định tư duy đầy hứa hẹn cũng như khả năng kỹ thuật của học sinh.

Ngày tháng năm
Người hướng dẫn
(kí, họ và tên)



Lê Thị Hồng Lam

MỞ ĐẦU

Tổng quan tình hình nghiên cứu thuộc lĩnh vực của đề tài

- Trong nước: hiện tại đã có những nghiên cứu về việc chế tạo và sử dụng cánh tay robot trong một số đề tài nghiên cứu khoa học.
 - Tại trường đại học sư phạm kỹ thuật khóa k16 đã có nhóm sinh viên Võ Nguyên Thông và Nguyễn Đặng Hùng Phú thuộc ngành tự động hóa khoa đạo tạo chất lượng cao chương trình đào tạo chất lượng cao tiếng việt nghiên cứ đề tài này dưới cái tên “Hệ thống phân loại liên tục sử dụng cánh tay rô bốt 5 bậc tự do và camera” và thực hiện nó thành đồ án tốt nghiệp dưới sự hướng dẫn của tiến sĩ Đặng Xuân Ba. Sử dụng công nghệ in 3D để chế tạo rô bốt 5 bậc tự do cũng với kỹ thuật điều khiển robot cho ra quy đạo chuyển động của rô bốt tối ưu và mượt mà và sử dụng xử lý ảnh xác định tính toán vị trí và phân loại màu sắc vật cần phân loại.
 - Tại trường đại học kỹ thuật - công nghệ Càm Thơ cũng thực hiện đề tài dưới cái tên “Điều khiển và phân loại vật thể dựa trên màu sắc sử dụng cánh tay robot 3 bậc tự do của Fischertechnik” do tác giả Phạm Thế Thịnh và Đỗ Vinh Quang.
- Ngoài nước:
 - Tại Sandip University, cánh tay rô bốt được nghiên cứu với mục đích phát hiện và phân loại đối tượng, một cánh tay robot được sử dụng trong dự án được điều khiển để tự động phát hiện và phân loại các đối tượng khác nhau cụ thể là trái cây. Trong dự án này, máy ảnh sẽ ghi lại hình ảnh trái cây để xử lý thêm trong mô hình dựa trên mạng nơ-ron phức hợp (CNN). Khi mô hình được học sẽ phát hiện đối tượng trong hình ảnh, một tín hiệu cụ thể sẽ được gửi đến cánh tay robot sử dụng Arduino uno, sẽ đặt đối tượng được phát hiện vào một cái rổ. Bằng cách này, dự án của chúng tôi sẽ nhận ra và phân loại hai loại trái cây khác nhau và sẽ đặt chúng vào các giỏ khác nhau. Dự án này là một minh chứng về sự kết hợp giữa khái niệm deep learning cùng với lập trình Arduino.
 - Tại đại học the South Pacific Suva, người ta dùng cánh tay robot xác định và gấp chính xác vật thể. Đầu tiên các đối tượng được phát hiện và điều

này được thực hiện bằng thuật toán trích xuất tính năng. Tiếp theo, hình ảnh được trích xuất được gửi đến trình phân loại để nhận ra đối tượng đó là gì và khi điều này được hoàn thiện, đầu ra sẽ là loại đối tượng cùng với tọa độ của nó để sẵn sàng cho cánh tay robot để thực hiện nhiệm vụ chọn và đặt.

- Tại đại học Faculty of Engineering & Technology International University for Science and Technology, người ta tăng cường tác động của thị giác máy tính đối với việc định vị và nắm bắt của robot trong các dây chuyền lắp ráp công nghiệp. Vấn đề phát hiện và bản địa hóa đối tượng trong thời gian thực được giải quyết cho hoạt động nắm bắt và định vị của robot bằng cách sử dụng Cánh tay robot lắp ráp tuân thủ có chọn lọc (SCARA). Mô hình Mạng nơ ron sâu (CNN), được gọi là KSSnet, được phát triển để phát hiện đối tượng dựa trên CNN Alexnet bằng cách sử dụng phương pháp học truyền. Bộ dữ liệu đào tạo SCARA với 4000 hình ảnh của hai loại đối tượng liên kết với 20 vị trí khác nhau được tạo và gắn nhãn để đào tạo mô hình KSSnet. Vị trí của đối tượng được phát hiện được đưa vào kết quả dự đoán ở lớp phân loại đầu ra.

Lý do chọn đề tài

- Đề tài nghiên cứu về cánh tay robot hiện là công nghệ đã được phát triển trên rất nhiều nơi ở nước ngoài nhưng chúng chỉ mới bắt đầu phát triển ở nước ta trong một khoảng thời gian không lâu. Để bắt kịp với xu hướng công nghệ với thế giới phục vụ cho ngành công nghiệp của nước nhà việc có những nghiên cứu về đề tài này là rất cấp bách và cần thiết.

Mục tiêu đề tài

- Thiết kế thi công và chế tạo cánh tay robot 4 bậc tự do đồng thời điều khiển chúng bằng phần mềm Matlab – Simulink.

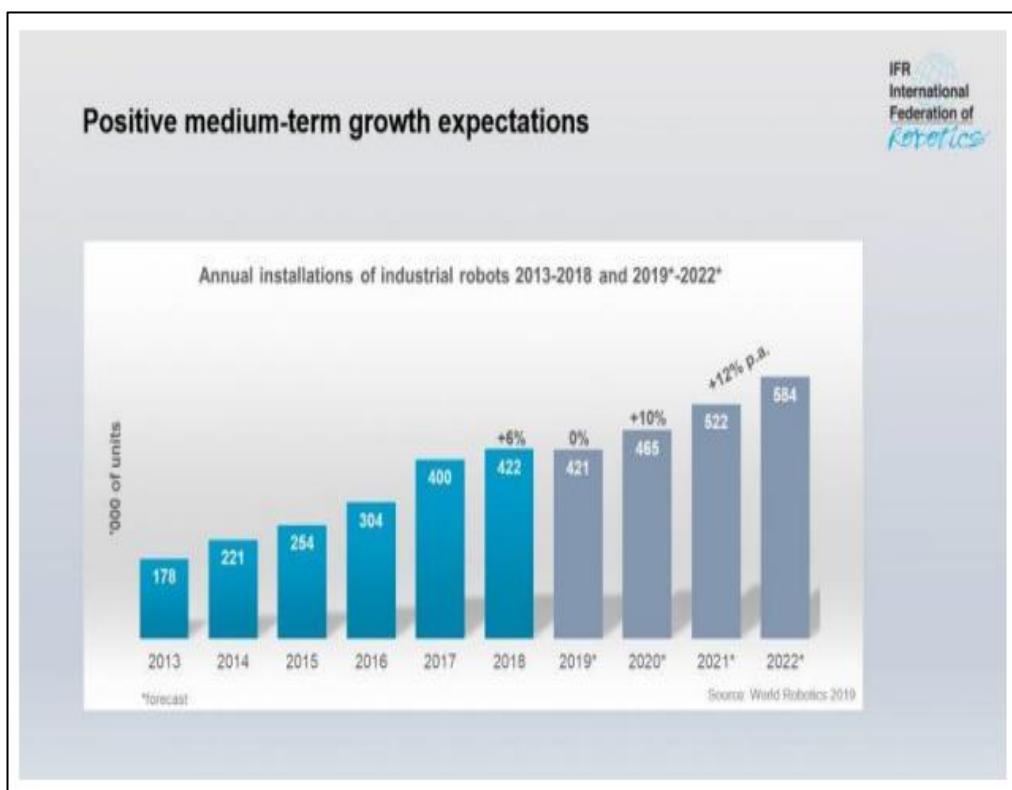
Cách tiếp cận, phương pháp nghiên cứu, phạm vi nghiên cứu

- Tiếp cận và phân tích các nghiên cứu các phương pháp chế tạo và điều khiển robot.
- Nghiên cứu cách điều khiển robot thông qua phần mềm.

CHƯƠNG 1: TỔNG QUAN

1.1 Giới thiệu

Ngày nay, với sự phát triển của khoa học kỹ thuật điện tử và công nghệ điều khiển và khoa học tự động hóa đã có những bước phát triển vượt bậc trong công nghệ tự động hóa và robot. Liên đoàn Người máy Quốc tế (IFR) từng công bố Báo cáo Người máy Thế giới hàng năm của mình, trong đó cho thấy giá trị bán hàng toàn cầu hàng năm là 16,5 tỷ USD vào năm 2018. IFR cho biết 422.000 chiếc đã được xuất xưởng trong năm 2018, tăng 6% so với năm 2017. Tuy nhiên, tập đoàn cho biết các lô hàng trong năm 2019 sẽ giảm so với mức kỷ lục trong năm 2018, nhưng cũng dự kiến tăng trưởng trung bình 12% mỗi năm từ năm 2020 đến năm 2022.



Hình 1. 1 Kỳ vọng tăng trưởng trung hạn đối với robot công nghiệp trong năm 2019-2022 [1]

Với những dữ liệu chúng ta có thể thấy để hòa nhập với sự phát triển của thế giới, xây dựng một mô hình robot có khả năng dễ dàng đổi mới phần mềm và phần cứng, cố định trong không gian, không cầu tạo quá phức tạp dễ dàng tính toán và thi công, có thể làm việc trong không gian ba chiều, lập trình bộ điều khiển dễ dàng và đặc biệt phải điều khiển theo vòng kín chi phí thấp, đáp ứng được các đặc tính cơ học. Nó rất

cần thiết cho việc nghiên cứu, học tập hay thậm chí là ứng dụng vào các công việc đơn giản trong sản xuất.

Trong dự án này, chúng tôi sẽ chế tạo một “cánh tay robot SCARA 2 bậc”, nó được coi là loại robot cơ bản, cổ điển và thông dụng nhất. Nếu hoàn thành tốt việc thiết kế và điều khiển robot này, chúng ta sẽ có nền tảng để chế tạo và điều khiển bất kỳ robot nào khác.

Đối với điều khiển robot, có nhiều cách thực hiện điều khiển với nhiều thuật toán điều khiển như điều khiển PID, điều khiển thần kinh, điều khiển mờ và bạn có thể tính toán động học của robot để điều khiển vị trí bằng nhiều phương pháp khác nhau như hệ thống điều khiển chế độ trượt, Hệ thống điều khiển LQR, ...

Nhưng trong dự án này, chúng tôi quyết định sử dụng phương pháp hình học để điều khiển động lực học của robot, PID để điều khiển động cơ và Matlab - Simulink để lập trình nó vì đây là những phương pháp, thuật toán và phần mềm hiệu quả đơn giản để xây dựng và điều khiển robot mà chúng tôi đã được dạy trong trường đại học.

1.2 Mục tiêu đề ra

Dự án của chúng tôi nhằm đến mục tiêu thiết kế, mô phỏng, xây dựng và điều khiển cánh tay robot 2-DOF di chuyển thiết bị đầu cuối theo quỹ đạo mong muốn (mặt phẳng hình tròn) trong thời gian thực.

Khoảng vượt kiểm soát sẽ không quá 5% và thời gian chuyển tiếp sẽ từ 2 giây - 5 giây.

1.3 Giới hạn đề tài

Trong dự án này, cánh tay robot không di chuyển. Các tham số PID được lựa chọn để hệ thống có các chỉ tiêu chất lượng thỏa mãn trong miền thời gian (độ vọt lố, thời gian chuyển tiếp). Tuy nhiên, việc lựa chọn tham số PID để giảm thiểu tham số tích phân cũng như các kỹ thuật điều khiển khác nằm ngoài phạm vi của đề tài. Cánh tay robot chỉ di chuyển theo quỹ đạo mong muốn mà không phải nắm lấy hoặc nâng bất kỳ vật thể nào.

1.4 Nội dung đề tài

Chương 2: Cơ sở lý thuyết

Chương 3: Thiết kế hệ thống

Chương 4: Cấu trúc sản phẩm

Chương 5: Kết quả

Chương 6: Kết luận

Chương 7: Tài liệu tham khảo

Chương 8: Phụ lục

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1 Giới thiệu chương 2

Trong chương này, chúng ta sẽ thảo luận về tất cả các yếu tố cho một dự án cánh tay robot như cấu trúc cánh tay robot, thuật toán điều khiển và phần mềm vận hành, ưu nhược điểm của chúng tùy thuộc vào đó chúng ta sẽ lựa chọn những yếu tố phù hợp nhất cho dự án của mình.

2.2 Cánh tay robot

2.2.1 Khái niệm về cánh tay robot

Cánh tay robot là một loại máy chuyên phục vụ quá trình sản xuất công nghiệp và được vận hành bởi con người. Chúng có ưu điểm là thiết kế linh hoạt, dễ vận hành với tác dụng hoàn thiện từng chi tiết nhỏ nhất cho sản phẩm.

Cánh tay robot hoạt động rất linh hoạt với cơ chế khớp ngón tay như tay người. Đó là một ưu điểm vượt trội mà các loại máy cơ này ngày càng được sử dụng rộng rãi trong các lĩnh vực công nghiệp khác nhau. Máy móc có thể thay thế bàn tay con người trong việc lắp ráp, sửa chữa, thay thế các linh kiện nhỏ trong nhà máy.

Cơ chế hoạt động dựa trên các liên kết của bộ điều khiển và khớp nối chuyên động. Nhờ đó, bộ điều khiển chủ động truyền lệnh đến các bộ phận kết nối để cánh tay đòn chuyển động hoặc di chuyển tuyến tính. Liên kết này sẽ được vận dụng dựa trên nhu cầu sản xuất để tạo thành một chuỗi động lực. Trong đó, sự kết thúc của chuỗi hành động được gọi là hiệu ứng kết thúc và chúng hoạt động dựa trên sự mô tả của bàn tay con người.

2.2.2 Cấu trúc của cánh tay robot

Tay máy: Tay máy là một bộ phận cơ khí bao gồm các đường khâu và khớp nối. Chúng được chế tạo giống như những cánh tay với khả năng vận động cơ bản, bao gồm các chi tiết: khéo léo ở cổ tay, di chuyển dễ dàng; tay thực hiện các thao tác, trực tiếp hoàn thành công việc. Các bộ phận được kết nối với nhau bằng các khớp trượt và xoay, giúp cho cánh tay robot vừa chắc chắn vừa linh hoạt.

Hệ thống điều khiển: Đây là bộ phận chịu trách nhiệm thực hiện các thao tác sau khi các tín hiệu đã được nhận và xử lý. Tùy theo yêu cầu đối với từng hoạt động, các chức năng của hệ thống điều khiển cánh tay robot sẽ được phân thành nhiều cấp

độ khác nhau, từ giải quyết đơn giản phương thức hành động chung, xác định hệ tọa độ, vị trí thao tác, các điểm phải đi qua, ... cho đến phức tạp hơn như thiết lập quỹ đạo, tính toán động năng, giải mã lệnh, nội suy, xử lý lỗi, ...

Phần mềm vận hành: Là môi trường lập trình, là phương tiện để người vận hành truyền lệnh cho robot. Phần mềm tiêu chuẩn phải đi kèm với ngôn ngữ lập trình phù hợp, thân thiện và dễ sử dụng.

2.2.3 Các loại cánh tay robot

Cánh tay robot SCARA: Thiết kế hai khớp quay song song hoạt động trên một mặt phẳng. Chúng được sử dụng trong các hoạt động lắp ráp hoặc gia công các bộ phận máy móc.

Ưu điểm của cánh tay robot SCARA là dễ chế tạo, cấu tạo, sửa chữa cũng như điều khiển, nhưng đồng thời do cấu tạo đơn giản nên nhược điểm chính là cánh tay thiếu tính linh hoạt. một số góc của môi trường và không gian đặc biệt.

Sau đây là hình ảnh của cánh tay robot SCARA:

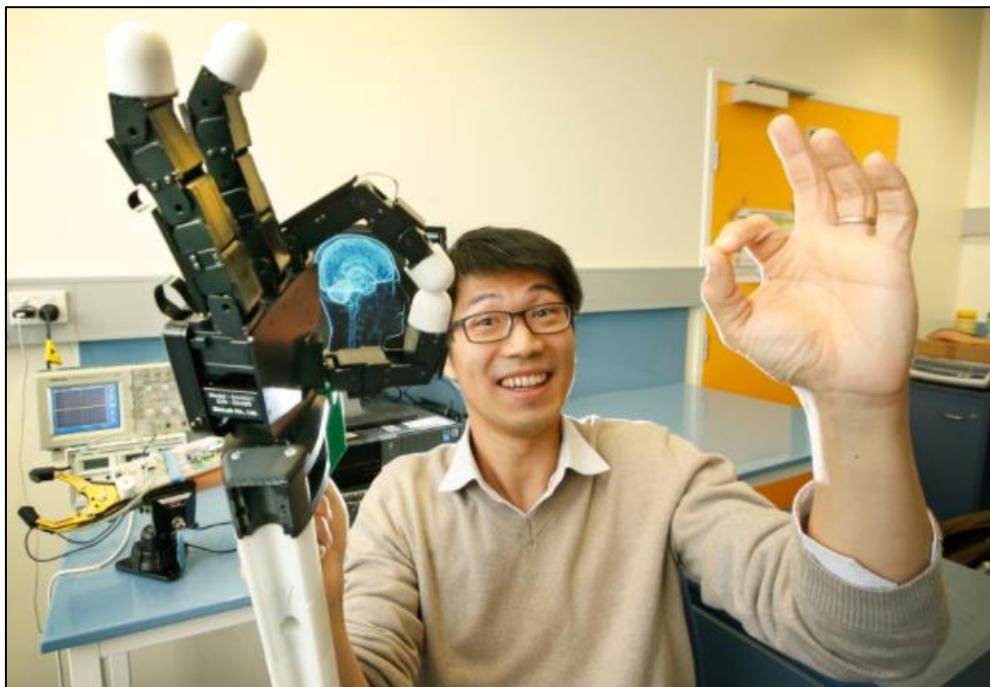


Hình 2. 1 Cánh tay robot Epson G1 Mini SCARA [2]

Cánh tay robot nhân tạo: Được mô phỏng theo chuyển động của con người, bàn tay và ngón cái di chuyển độc lập với nhau. Chính sự linh hoạt cho phép họ tham gia vào việc lắp ráp và xử lý tất cả các loại linh kiện điện tử hay công nghệ microcar.

Các cánh tay robot nhân tạo có ưu điểm là thực hiện các công việc tỉ mỉ, phức tạp và phân tích, nhưng điều này cũng dẫn đến nhược điểm của chúng: giá thành cao, khó chế tạo và điều khiển.

Sau đây là hình ảnh của cánh tay robot nhân tạo:



Hình 2. 2 Phó giáo sư Denny Oetomo đang thử nghiệm một cánh tay nhân tạo. Hình:
David Caird / Herald Sun [3]

Cánh tay rô bốt có khớp nối: Thường được thiết kế với ba khớp quay. Máy móc thông dụng là cánh tay rô bốt 3 tầng, 4 tầng... Theo đó, cấu tạo gồm hai nhánh đơn giản đến hệ thống có 10 khớp tương tác trở lên.

Ưu điểm của cánh tay rô bốt khớp nối là tính linh hoạt khá hữu dụng trong hầu hết các công việc cũng như việc chế tạo không quá phức tạp nhưng nhược điểm của nó là mặc dù chế tạo không phức tạp đối với người vận hành, và lập trình nó cũng cần có kiến thức chuyên môn nhất định để thực hiện

Đây là hình ảnh sau đây về một mô hình cỗ điển của cánh tay rô bốt có khớp nối - cánh tay rô bốt 4-DOF:



Hình 2. 3 Denso - cánh tay robot 4-DOF [3]

2.3 Thuật toán điều khiển

2.3.1 Khái niệm về thuật toán điều khiển

Khi giải thích các thuật toán điều khiển, chúng ta cần giải thích ý nghĩa của chúng trong hai thuật toán và các thành phần điều khiển:

Thuật toán: Trong toán học và khoa học máy tính, thuật toán là một chuỗi hữu hạn các lệnh được xác định rõ ràng, có thể thực hiện được bằng máy tính, thường để giải quyết một lớp vấn đề hoặc để thực hiện một phép tính [4].

Không phải tất cả các thủ tục đều có thể được gọi là một thuật toán. Một thuật toán phải có các đặc điểm sau:

- **Xác định:** Thuật toán phải rõ ràng và không mơ hồ. Mỗi giai đoạn (hoặc bước) nên rõ ràng và chỉ có một mục đích cụ thể.
- **Dữ liệu đầu vào được chỉ định:** Một thuật toán phải có 0 hoặc nhiều dữ liệu đầu vào được xác định.
- **Đầu ra:** Một thuật toán phải có một hoặc nhiều dữ liệu đầu ra được xác định và phải kết nối với loại đầu ra mong muốn.
- **Thuật toán:** Phải kết thúc sau một số bước hữu hạn.

- **Tính hiệu quả:** Một thuật toán phải khả thi với các nguồn lực sẵn có và có thể giải quyết vấn đề một cách hiệu quả trong thời gian và nguồn lực cho phép.
- **Phổ biến:** Một thuật toán phổ biến nếu nó có thể giải quyết một lớp các vấn đề tương tự.
- **Độc lập:** Một thuật toán nên có các lệnh độc lập với bất kỳ mảng lập trình nào.

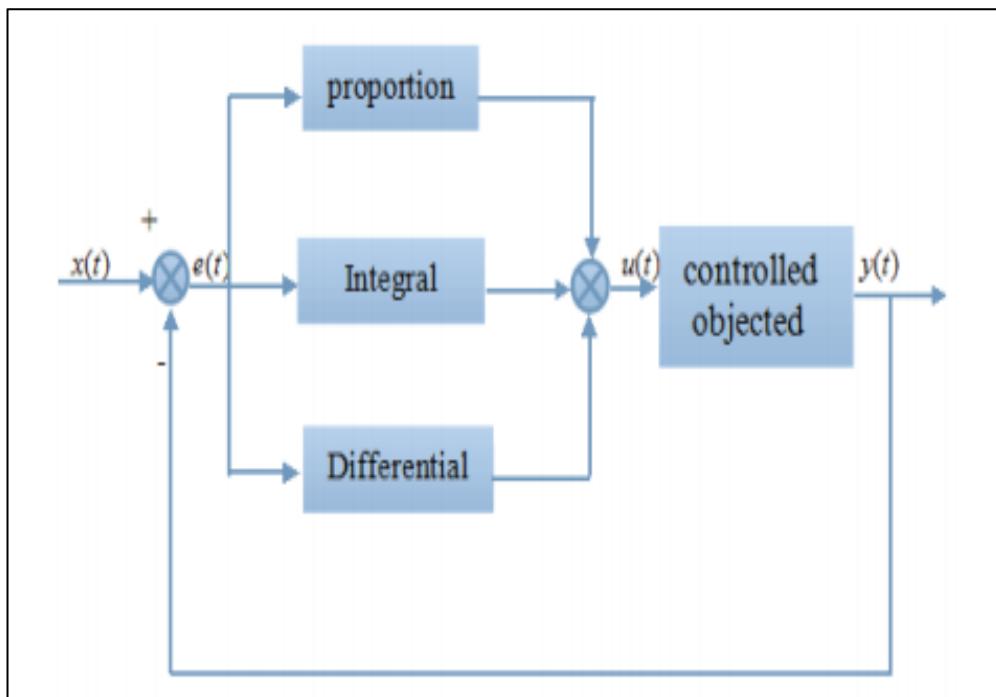
Điều khiển: Điều khiển là một chức năng của quản lý giúp kiểm tra các sai sót để đưa ra các hành động sửa chữa. Điều này được thực hiện để giảm thiểu sự sai lệch so với các tiêu chuẩn và đảm bảo rằng các mục tiêu đã nêu của tổ chức đạt được theo cách thức mong muốn. Theo quan niệm hiện đại, điều khiển là một hành động thấy trước; các khái niệm trước đó về điều khiển chỉ được sử dụng khi các lỗi được phát hiện. Điều khiển trong quản lý bao gồm thiết lập các tiêu chuẩn, đo lường hiệu suất thực tế và thực hiện hành động khắc phục trong quá trình ra quyết định. [5]

Thuật toán điều khiển: Thuật toán điều khiển là một loại logic được lập trình trong bộ điều khiển để phân tích sai số giữa giá trị đo được (MV) và điểm đặt (SP). Logic này giúp bộ điều khiển xử lý dữ liệu trường và quyết định phải làm gì. [6]

2.3.2 Các loại thuật toán điều khiển

Điều khiển PID: Bộ điều khiển dẫn xuất tỷ lệ - tích phân (bộ điều khiển PID hoặc bộ điều khiển ba kỳ hạn) là một cơ chế vòng điều khiển sử dụng phản hồi được sử dụng rộng rãi trong các hệ thống điều khiển công nghiệp và nhiều ứng dụng khác yêu cầu điều khiển được điều chỉnh liên tục. Bộ điều khiển PID liên tục tính toán giá trị lỗi $e(t)$ là hiệu số giữa điểm đặt mong muốn (SP) và biến quá trình đo được (PV) và áp dụng hiệu chỉnh dựa trên các số hạng tỷ lệ, tích phân và đạo hàm (ký hiệu là P, I và D tương ứng), do đó có tên [7].

Hình ảnh dưới đây cho thấy sơ đồ của bộ điều khiển PID:



Hình 2. 4 Sơ đồ của bộ điều khiển PID

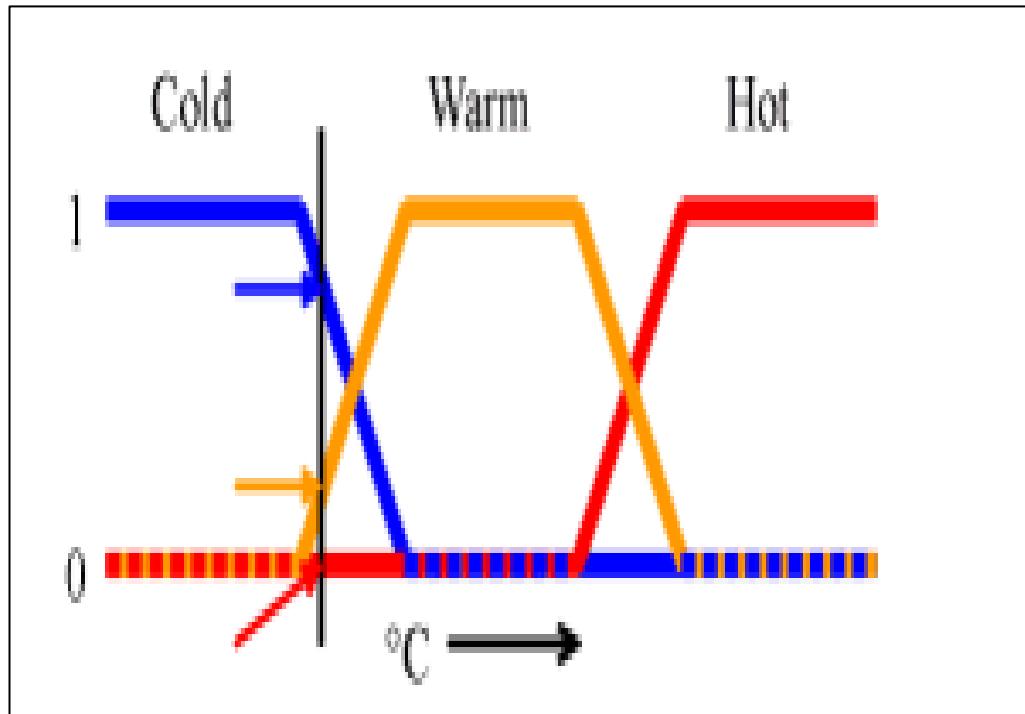
Ưu điểm: điều khiển chính xác cao, tiết kiệm năng lượng tối đa, đảm bảo hệ thống hoạt động ổn định.

Nhược điểm: thuật toán điều khiển phức tạp, yêu cầu người dùng có trình độ và kinh nghiệm.

Khi sử dụng chế độ điều khiển PID trong công nghiệp, loại đầu ra điều khiển tối ưu là SSR bán dẫn. Không nên sử dụng role thường vì dễ xảy ra các sự cố ngoài ý muốn như đánh lửa, kẹt tiếp điểm, giảm tuổi thọ thiết bị ...

Điều khiển logic mờ: mô tả mối quan hệ dựa trên quy tắc if-then, ví dụ, chẳng hạn như "nếu van nóng được mở, nhiệt độ tăng". Tính mơ hồ (không xác định) trong nồng độ, ý nghĩa của các yếu tố ngôn ngữ (ví dụ, nhiệt độ cao) được thể hiện dưới dạng các tập mờ, là tập có ranh giới chồng chéo. Theo nghĩa của tập mờ, thì một miền phần tử có thể nằm trong nhiều tập đồng thời (với các mức tham gia khác nhau cùng nhau) hơn là sử dụng quy tắc của logic mờ để điều khiển trạng thái của hệ thống.

Hình dưới đây là một ví dụ về bộ điều khiển logic mờ:



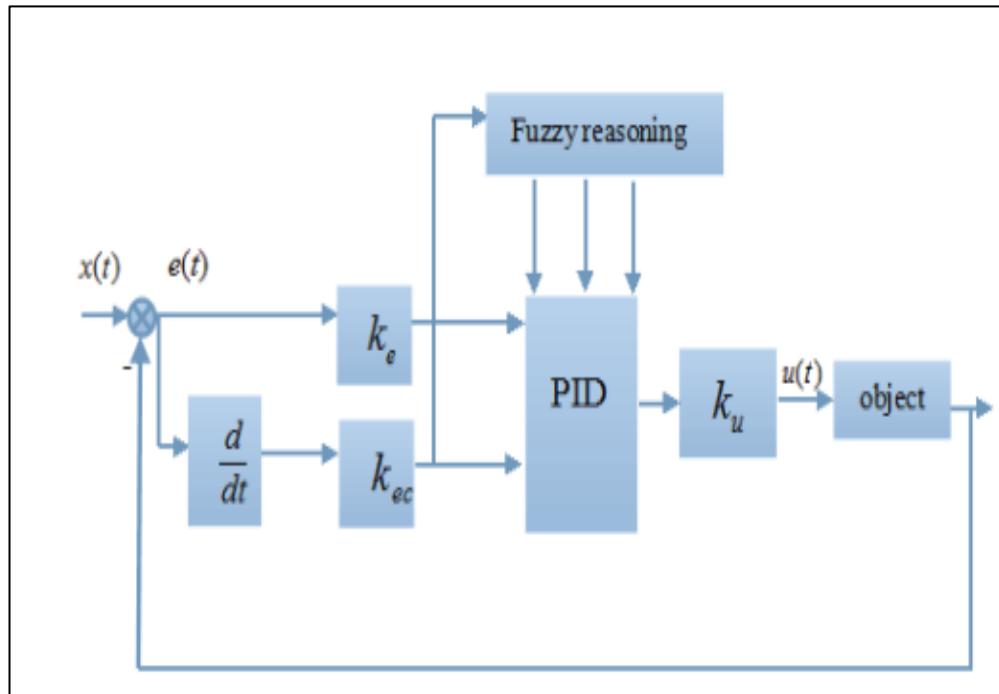
Hình 2. 5 Ví dụ về bộ điều khiển logic mờ [8]

Ưu điểm của bộ điều khiển logic mờ là tính linh hoạt trong hệ thống cho phép chúng ta điều khiển bất kỳ loại hệ thống nào mà không cần biết người xây dựng và cấu trúc phần cứng.

Nhược điểm của hệ thống điều khiển này là không có độ chính xác cao. Tất cả các kết quả kiểm soát đều cho kết quả với xác suất tương đối và việc xây dựng bộ điều khiển hoàn toàn dựa trên kinh nghiệm và không tuân thủ cấu trúc. Kiến trúc của thiết bị hoặc hệ thống nên yêu cầu người thiết kế thử nghiệm nhiều lần để có được bộ điều khiển gần đúng nhất.

PID mờ: Bộ điều khiển đạo hàm tích phân tỷ lệ mờ (PID) được đề xuất có thể được điều chỉnh bằng cách thực hiện các quy tắc điều chỉnh từ miền PID sang miền mờ. Vì bộ điều khiển phi tuyến có thể điều khiển quá trình phi tuyến hiệu quả hơn, bộ điều khiển mờ có thể cung cấp hiệu suất tốt hơn về thời gian tăng và độ vọt lô nhỏ hơn. Bộ điều khiển được đề xuất được đánh giá bằng cách sử dụng một số mô phỏng. Kết quả mô phỏng cho thấy cả hiệu suất nhất thời và hiệu suất trạng thái ổn định đều tốt hơn so với các bộ điều khiển PID thông thường [9].

Hình ảnh dưới đây cho thấy sơ đồ của một bộ điều khiển Fuzzy – PID:



Hình 2. 6 Sơ đồ của bộ điều khiển Mờ - PID

Ưu điểm của bộ điều khiển là phản ứng của các thiết bị với thời gian tăng và thời gian cài đặt hợp lý, độ vọt lô không đáng kể và loại bỏ các lỗi cài đặt. Ngoài ra, bộ điều khiển còn đáp ứng yêu cầu về thời gian thực và độ ổn định trước ảnh hưởng của nhiều đối với hệ thống.

Nhược điểm của bộ điều khiển này là cần có kinh nghiệm thiết kế luật mờ cũng như kiến thức chuyên sâu về bộ điều khiển PID và phải thực nghiệm nhiều mới có kết quả tốt nhất.

2.4 Phần mềm vận hành

2.4.1 Khái niệm

Như đã nói ở trên phần mềm vận hành là môi trường lập trình, là phương tiện để người vận hành truyền lệnh cho robot. Phần mềm tiêu chuẩn phải đi kèm với ngôn ngữ lập trình phù hợp, thân thiện và dễ sử dụng.

2.4.2 Các loại phần mềm vận hành

Arduino IDE được viết tắt (Arduino Integrated Development Environment) là một trình soạn thảo văn bản giúp bạn viết mã để tải lên board Arduino. Chúng ta có thể sử dụng Arduino Ide để điều khiển trực tiếp nó thông qua bảng điều khiển Arduino.

Ưu điểm của phần mềm này là chúng có tất cả các thư viện và công cụ cần thiết cho người mới bắt đầu và chúng tương thích 100% với tất cả các loại vi điều khiển Arduino.

Nhược điểm là chúng chỉ tương thích và sử dụng lập trình được cho vi điều khiển Arduino.

MATLAB (Phòng thí nghiệm Ma trận) là phần mềm khoa học được thiết kế để cung cấp tính toán số và hiển thị đồ họa bằng ngôn ngữ lập trình cấp cao. Cung cấp bởi MATLAB.

Các tính năng tương tác tuyệt vời cho phép người dùng linh hoạt thao tác dữ liệu dưới dạng mảng ma trận để tính toán và quan sát. Đầu vào MATLAB có thể nhập từ "Command line" hoặc từ "mfiles", trong đó tập lệnh được cung cấp bởi MATLAB.

MATLAB cung cấp cho người dùng các hộp công cụ tiêu chuẩn tùy chọn. Người dùng, bạn cũng có thể tạo hộp công cụ "mfiles" của riêng mình được viết cho các công cụ dành riêng cho ứng dụng. Chúng tôi có thể sử dụng tệp trợ giúp của MATLAB cho các chức năng và lệnh này được liên kết với các hộp công cụ có sẵn (sử dụng lệnh trợ giúp).

Simulink là một công cụ trong MATLAB để lập mô hình, mô phỏng và phân tích các hệ thống với giao diện người dùng đồ họa. Việc xây dựng mô hình được đơn giản hóa với các thao tác nhập và kéo. Simulink bao gồm một tập hợp các vien khói chũ cái với các hộp công cụ toàn diện cho cả phân tích tuyến tính và phi tuyến tính. Simulink là một phần quan trọng của MATLAB và có thể dễ dàng chuyển đổi qua lại trong quá trình phân tích và do đó người dùng có thể tận dụng lợi thế của cả hai môi trường. Chúng tôi có thể sử dụng phần mềm này để mô phỏng, xây dựng, lập trình và điều khiển cánh tay robot bằng bất kỳ bộ vi xử lý nào trong gói hỗ trợ phần cứng.

Ưu điểm của phần mềm này là chúng có đầy đủ các thư viện, công cụ cần thiết và các gói hỗ trợ cho người mới bắt đầu dễ dàng lập trình các thuật toán điều khiển khó và chúng tương thích 100% với các loại vi điều khiển.

Nhược điểm là chúng chỉ tương thích và chỉ sử dụng lập trình cho phần mềm này. Ngôn ngữ lập trình của nó được thiết kế rất đơn giản nhưng nó chỉ dùng được cho phần mềm này vì phần mềm khác không thể hiểu hoặc đọc được ngôn ngữ lập trình của chúng.

2.5 Lựa chọn cho đề tài

Thông qua việc tìm hiểu về các loại cánh tay rô bốt, các thuật toán điều khiển, và phần mềm điều khiển, nhóm chúng tôi đã quyết định chọn loại cánh tay rô bốt mà chúng tôi sẽ chế tạo là rô bốt SCARA - 2 bậc tự do vì đây là mẫu cổ điển phù hợp với những người mới bắt đầu chế tạo và có thể phát triển thành những cánh tay robot phức tạp hơn.

Thuật toán bộ điều khiển mà chúng tôi chọn sử dụng là bộ điều khiển PID vì tính dễ điều khiển và tính linh hoạt của nó hoàn toàn phù hợp với yêu cầu không quá khắt khe của phần cứng.

Phần mềm điều hành được lựa chọn là MATLAB - Simulink vì chúng dễ dàng tính toán các thuật toán phức tạp cũng như điều khiển các mô hình khó mà đồng thời không cần kiến thức lập trình chuyên sâu. với các yêu cầu về phần cứng cũng như đối tượng.

CHƯƠNG 3: THIẾT KẾ HỆ THỐNG

3.1 Giới thiệu chương 3

Yêu cầu của đề tài là thiết kế, xây dựng, lập trình và điều khiển SCARA - 2 bậc của cánh tay robot tự do, dựa trên những ưu điểm của phần mềm và cả thuật toán điều khiển nhóm được chọn lập trình và điều khiển bởi MATLAB – Simulink và sử dụng thuật toán PID để điều khiển, dựa vào các yếu tố trên, nhóm sẽ lựa chọn phần cứng, cũng như thiết kế phần cứng và phần mềm như sau cho phù hợp với dự án.

3.2 Phần cứng

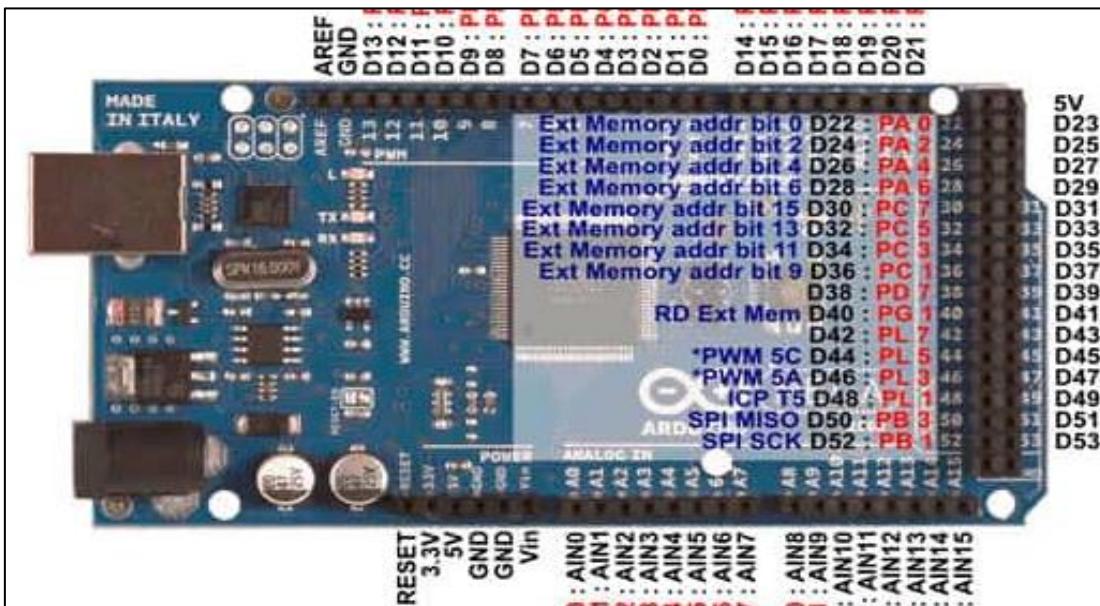
3.2.1 Vi điều khiển

3.2.1.1 Arduino Mega 2560 R3

Mạch Arduino Mega 2560 R3 Atmega là phiên bản nâng cấp của Arduino Uno R3 với nhiều chân giao tiếp, thiết bị ngoại vi và bộ nhớ hơn, mạch được thiết kế và sử dụng linh kiện tương đương với phiên bản gốc trên Arduino. cc, phù hợp với các ứng dụng cần nhiều bộ nhớ hoặc nhiều chân, cổng giao tiếp hơn Arduino Uno, bo mạch này có 54 chân I / O (14 chân PWM), 16 đầu vào analog, 4 UART (phần cứng cổng nối tiếp), sử dụng thạch anh 16 MHz, Kết nối cổng USB, giắc cắm nguồn, chân ICSP và nút đặt lại.

Lý do để chọn bo mạch Arduino mega thay vì các vi điều khiển khác là vì Arduino mega 2560 R3 có khả năng tương thích cao với phần mềm lập trình Simulink của MATLAB so với Arduino UNO R3 mà giá thành lại không quá đắt như các bo mạch máy tính mini khác như Raspberry Pi, chúng vẫn đáp ứng các yêu cầu phần cứng tốt nhất có thể.

Sau đây là hình ảnh thực tế board điều khiển Arduino Mega 2560 R3 Atmega:



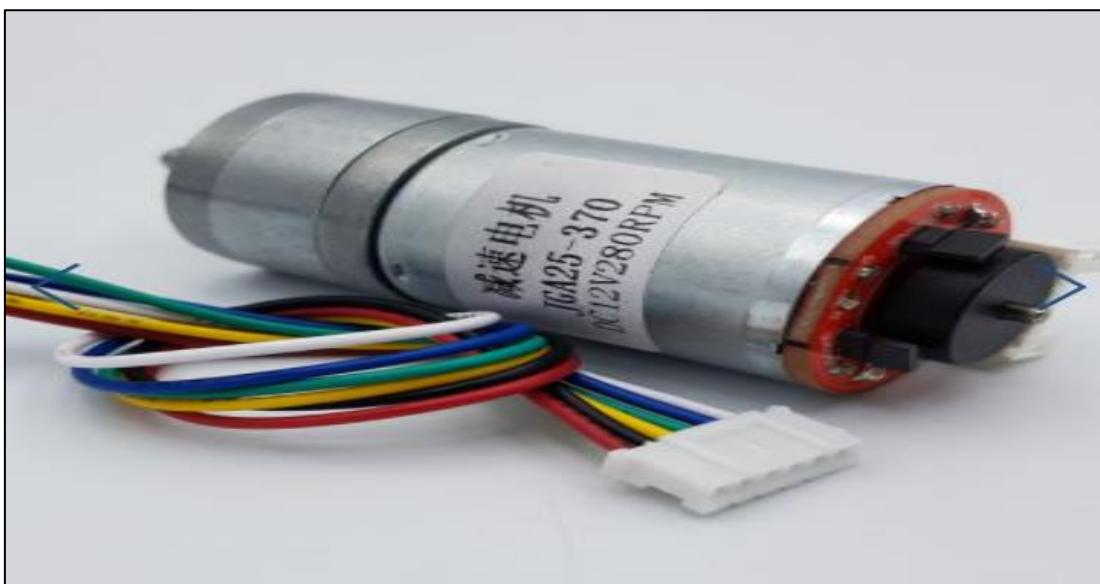
Hình 3. 1 Arduino Mega 2560 R3 Atmega [10]

3.2.2 Động cơ DC tích hợp với bộ mã hóa

Động cơ được chọn cho dự án này là động cơ DC có bộ mã hóa tích hợp. Việc lựa chọn động cơ DC so với các loại động cơ khác là do tính linh hoạt nhỏ gọn của chúng giúp khung cánh tay đỡ nặng hơn mà vẫn đảm bảo độ linh hoạt cần thiết, việc tích hợp thêm bộ mã hóa sẽ giúp chúng ta lấy được phản hồi cấp xung điều khiển đến bộ điều khiển.

3.2.2.1 DC- động cơ mã hóa JGA25

Sau đây là hình ảnh của động cơ mã hóa DC JGA25 trong thực tế:



Hình 3. 2 DC- động cơ mã hóa JGA25 [11]

Động cơ dc encoder jga25 là loại motor dẽ kiém trên thị trường, giá cả hợp lý cho sinh viên, kích thước nhỏ gọn tối ưu cho việc sử dụng trong thiết kế phần cứng.

Trong đờ án ta sử dụng 2 loại động cơ, 1 loại 60 vòng / phút và một loại 130 vòng / phút phải sử dụng một loại 60 vòng / phút vì ở khớp thứ nhất động cơ có momen xoắn cần kéo 1,9kg tải và dùng động cơ quay chậm chắc chắn đáp ứng được yêu cầu đó.

Sau đây là hình ảnh là bảng dữ liệu của nhiều loại động cơ mã hóa DC JGA25 trong thực tế:

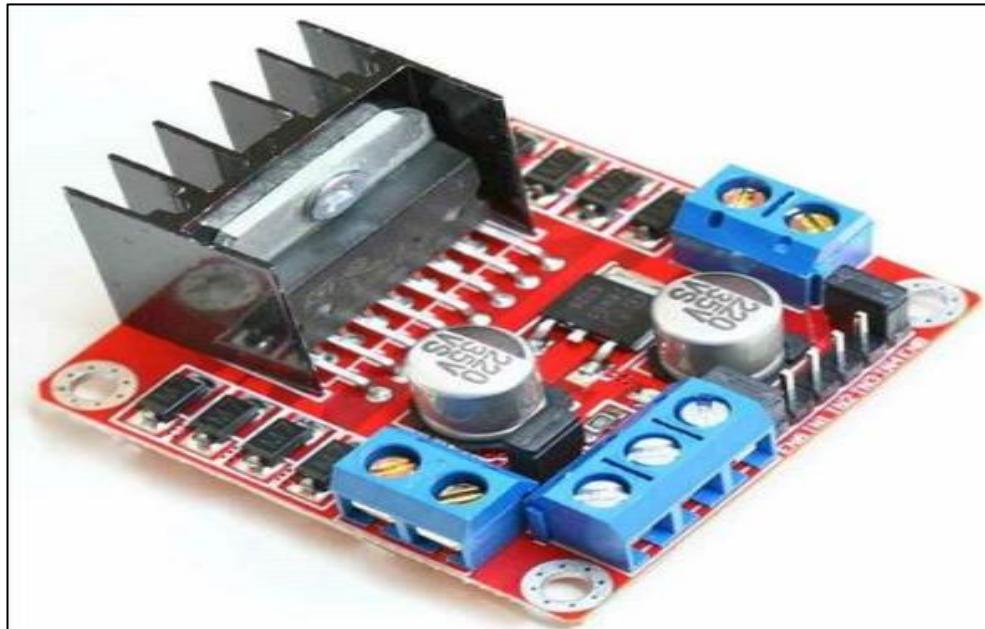
CGM25-370-18250 4W DC12.0V											
Tỷ Lệ hộp số	4.4	9.28	21.3	34	45	75	100	165	217	362	478
L	15.0	17.0	19.0	21.0	21.0	23.0	23.0	25.0	25.0	27.0	27.0
rpm	1930	915	400	250	190	110	85	50	39	23	18
MA	≤150	≤150	≤150	≤150	≤150	≤150	≤150	≤150	≤150	≤150	≤150
kg.cm	0.5	1.2	2.7	4.3	5.0	8.0	8.0	8.0	8.0	8.0	8.0
rpm	105	500	230	140	100	65	55	40	32	20	17
MA	≤750	≤750	≤750	≤750	≤750	≤750	≤750	≤750	≤750	≤750	≤750
kg.cm	1.0	2.0	4.5	7.5	9.0	9.0kg,cm					
	48.4	102.1	234.3	374	495	825	1100	1815	2387	3982	5258

Hình 3. 3 Bảng dữ liệu của nhiều loại động cơ mã hóa DC-JGA25 [11]

3.2.3 Bộ điều khiển động cơ

Vì đề tài chỉ cần điều khiển 2 động cơ nên việc lựa chọn sử dụng bộ điều khiển động cơ L298N là rất hợp lý, chưa kể chúng có giá thành rất rẻ và dẽ kiém trên thị trường.

Sau đây là hình ảnh thực tế của bộ điều khiển động cơ L298N:



Hình 3. 4 Bộ điều khiển động cơ L298N trong thực tế [12]

3.2.4 Nguồn cung cấp

Đè tài sử dụng 2 nguồn cấp là nguồn cấp cho vi điều khiển lấy trực tiếp từ cổng USB của máy tính để truyền đồng thời dữ liệu điều khiển phần cứng, 2 là nguồn cấp điều khiển động cơ kết nối trực tiếp với L298N, do L298N cần nguồn DC 12v để hoạt động, ở đây nhóm mình quyết định sử dụng nguồn 12V-DC 5A để duy trì sự ổn định cũng như cung cấp đủ điện áp để điều khiển động cơ.

Sau đây là hình ảnh là bộ nguồn 12V-DC 5A trong thực tế:

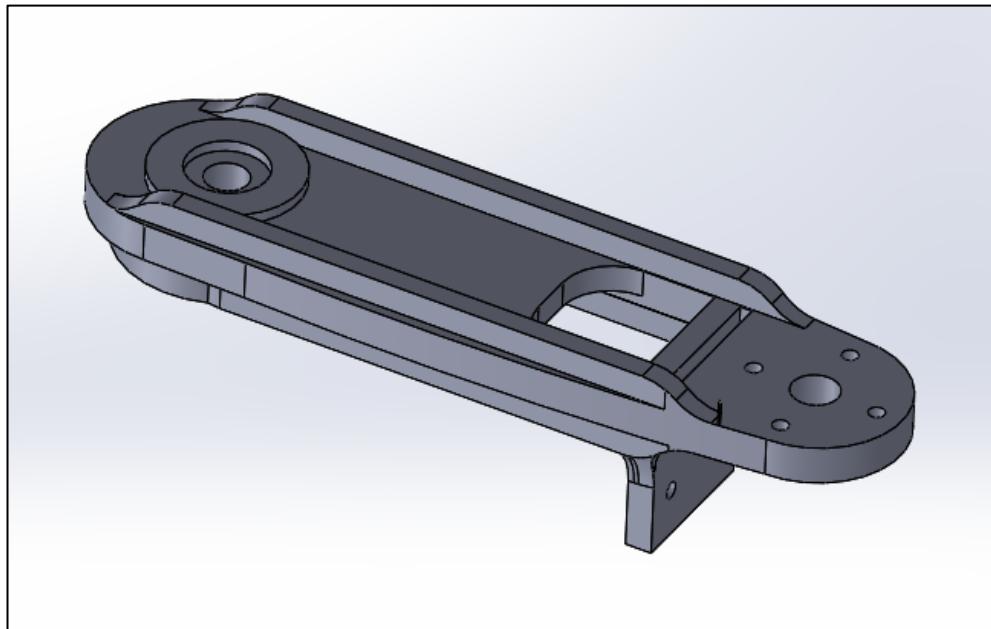


Hình 3. 5 Nguồn điện 12V-DC 5A trong thực tế [13]

3.2.5 Phần cánh tay robot in 3D

Hiện nay công nghệ in 3D rất phát triển khiến chúng ta có nhiều thuận lợi trong việc chế tạo cơ khí các loại tay máy, chỉ cần thiết kế phần cứng trên Solid gia công rồi đem đi in là chúng ta đã có phần cơ khí. Độ chính xác cao không thua kém gì công nghệ CNC mà nhanh hơn và dễ dàng hơn.

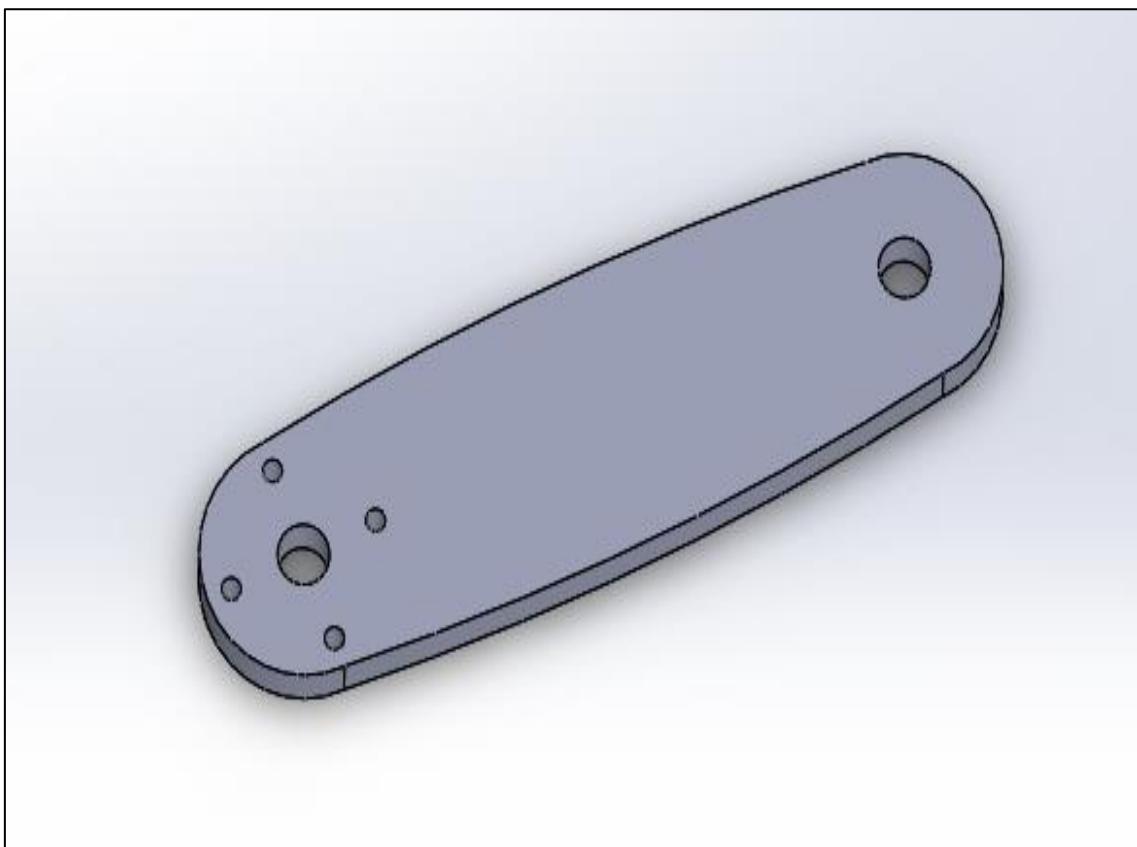
Sau đây là những hình ảnh về 3 phần cánh tay robot khi thiết kế trên Solid work và khi in ra:



Hình 3. 6 Khớp nối thứ nhất của cánh tay robot trong Solidwork



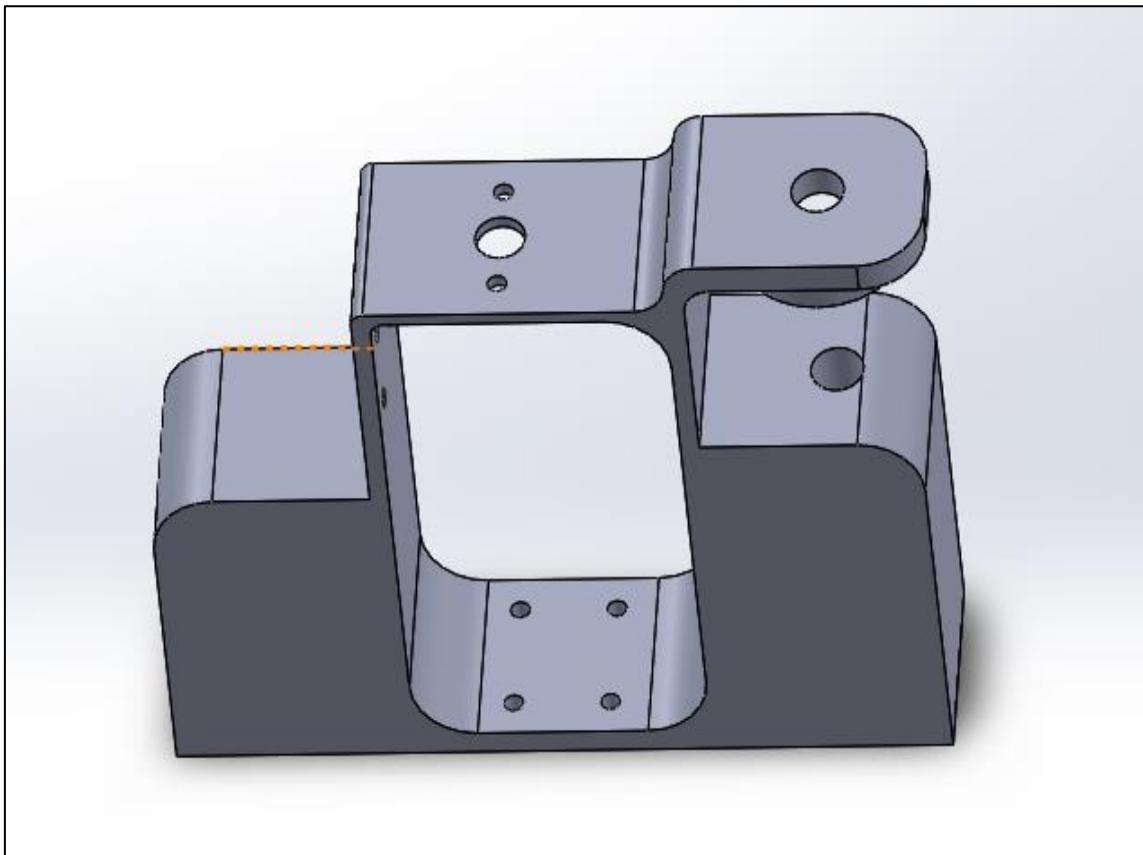
Hình 3. 7 Khớp nối thứ nhất của cánh tay robot sau khi in ra



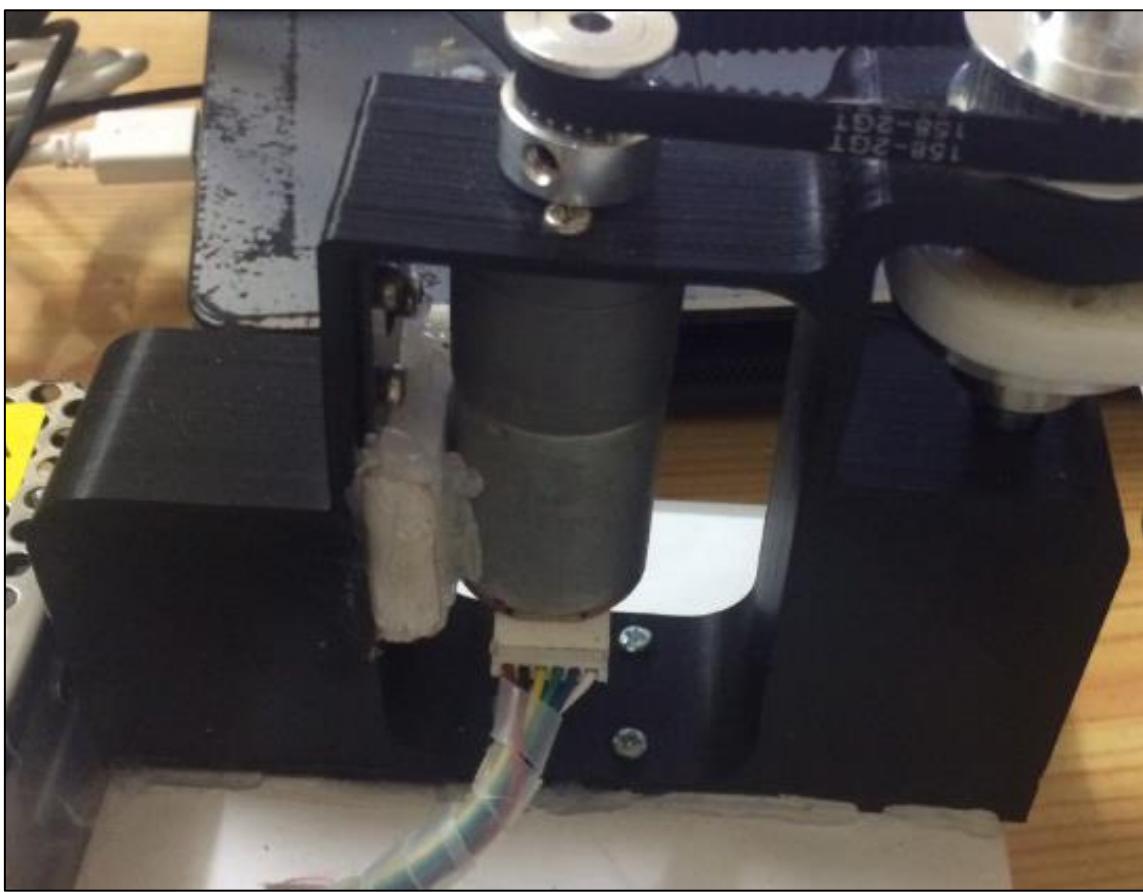
Hình 3. 8 Khớp nối thứ hai của cánh tay robot trong Solidwork



Hình 3. 9 Khớp nối thứ hai của cánh tay robot sau khi in ra



Hình 3. 10 Khớp nền của cánh tay robot trong Solidwork



Hình 3. 11 Khớp nền của cánh tay robot sau khi in ra

3.2.6 Bánh răng

Vì để tăng lực kéo cho động cơ nên trong đề tài này chúng tôi đã sử dụng hai loại bánh răng và truyền động giữa chúng bằng đai răng 2mm. Nhóm chúng tôi chọn 20-răng gt-2 hộp số gắn vào trục động cơ và bánh răng 40-răng gt-2 trên trục muôn truyền động điều này sẽ làm giảm vận tốc góc ở trục và giảm tốc tức là mômen quay của trục. tăng theo lý thuyết vật lý.

Điều này ngoài việc tăng mômen kéo, chúng còn kéo căng dây đai để có góc quay chính xác hơn và giảm độ nhám của động cơ.

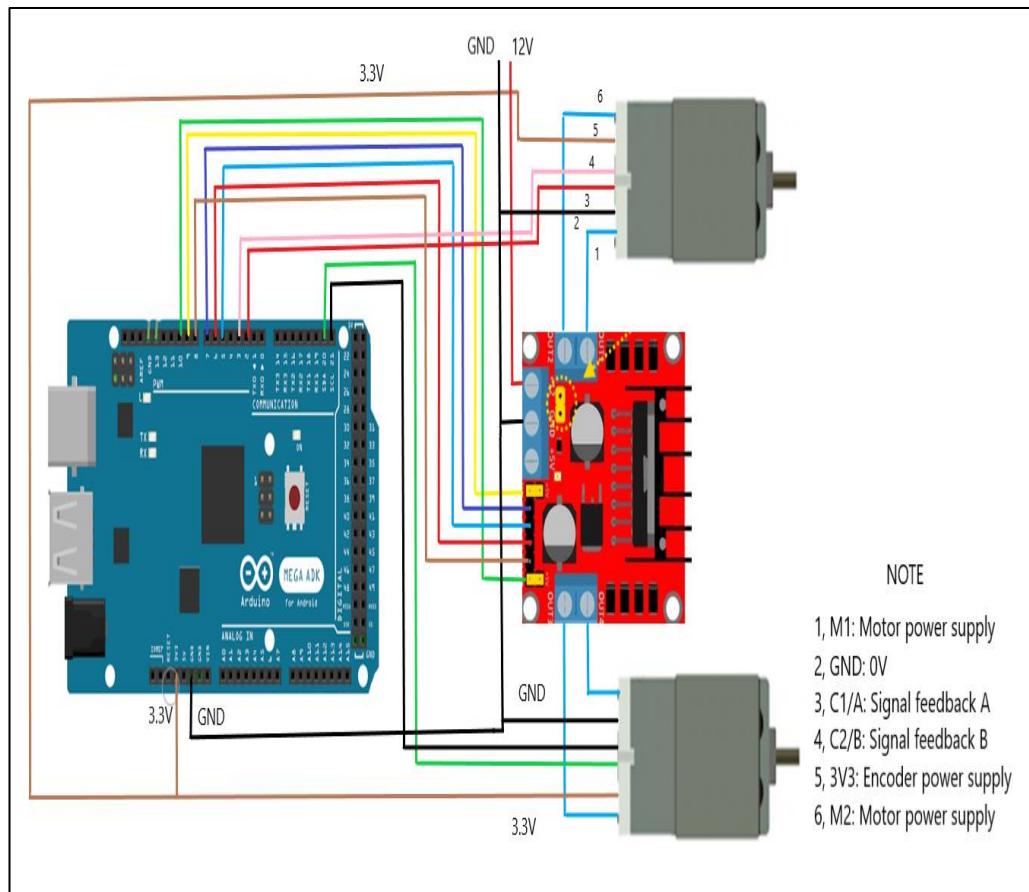
Sau đây là hình ảnh là ròng rọc trong thực tế:



Hình 3. 12 Ròng rọc thực tế [13]

3.2.7 Sơ đồ nối dây phần cứng

Như tất cả các thiết bị liệt kê trong chương 3, chúng tôi sẽ có sơ đồ dây phần cứng như hình sau:



Hình 3. 13 Sơ đồ nối dây thực tế

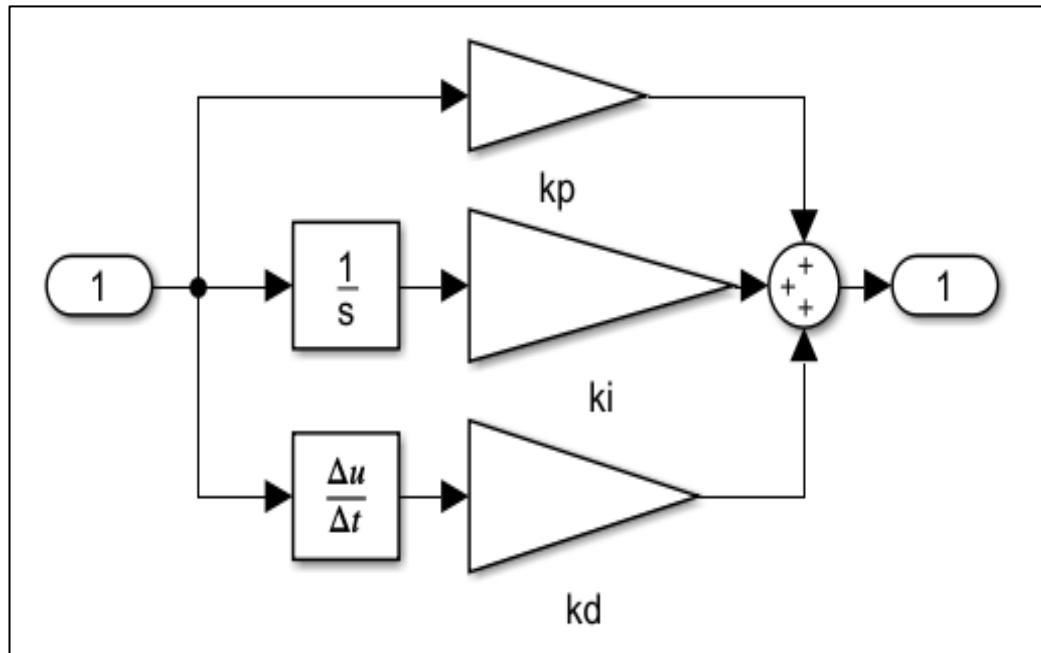
3.3 Thiết kế phần mềm

3.3.1 Thiết kế bộ điều khiển PID

Bộ điều khiển PID tính toán đầu ra của nó dựa trên sai số đo được và ba lợi ích của bộ điều khiển; độ lợi tỷ lệ K_p , độ lợi tích phân K_i và độ lợi đạo hàm K_d . Độ lợi tỷ lệ chỉ đơn giản là nhân sai số với hệ số. Phản hồi này dựa trên mức độ nghiêm trọng của lỗi. Thuật ngữ tích phân là phép nhân của tích phân và tổng các lỗi gần đây. Thuật ngữ tích phân loại bỏ các lỗi ở trạng thái ổn định và làm cho hệ thống bắt kịp với điểm đặt mong muốn [14]. Bộ điều khiển phái sinh phản ứng với tốc độ thay đổi của lỗi. Công thức của nó là:

$$U = K_p e(t) + K_i \int e dt + K_d \frac{de}{dt} \quad (1)$$

Trong dự án này, bộ điều khiển PID được thiết kế như trong hình dưới đây:



Hình 3. 14 Bộ điều khiển PID cho hệ thống

3.3.2 Phương pháp điều chỉnh bộ điều khiển PID

Các giá trị bộ điều khiển PID K_p , K_d , và K_i có thể được xác định bằng phương pháp thử và sai. Chúng tôi bắt đầu bằng cách đưa ra phỏng đoán ban đầu về giá trị K_p bằng cách gán $K_i=K_d=0$. Chúng tôi nhận thấy rằng khi K_p có một giá trị lớn, chúng tôi có thời gian tăng nhanh hơn (rising time), nhưng lại tăng độ vọt lố (overshoot). Để cải thiện hiệu ứng của K_p , một giá trị bổ sung K_i cũng được đặt thành một giá trị nhỏ. Việc bổ sung độ lợi tích phân có tác dụng lớn trong việc giảm sai số xác lập (steady-state error) giữa đầu ra mong muốn và đầu ra thực tế. Bảng dưới đây sẽ cho chúng tôi một ý tưởng sơ bộ về việc thay đổi từng tham số PID có thể thay đổi phản ứng hệ thống của chúng tôi như thế nào [14].

Bảng 3. 1 Đặc điểm thông số bộ điều khiển PID [14]

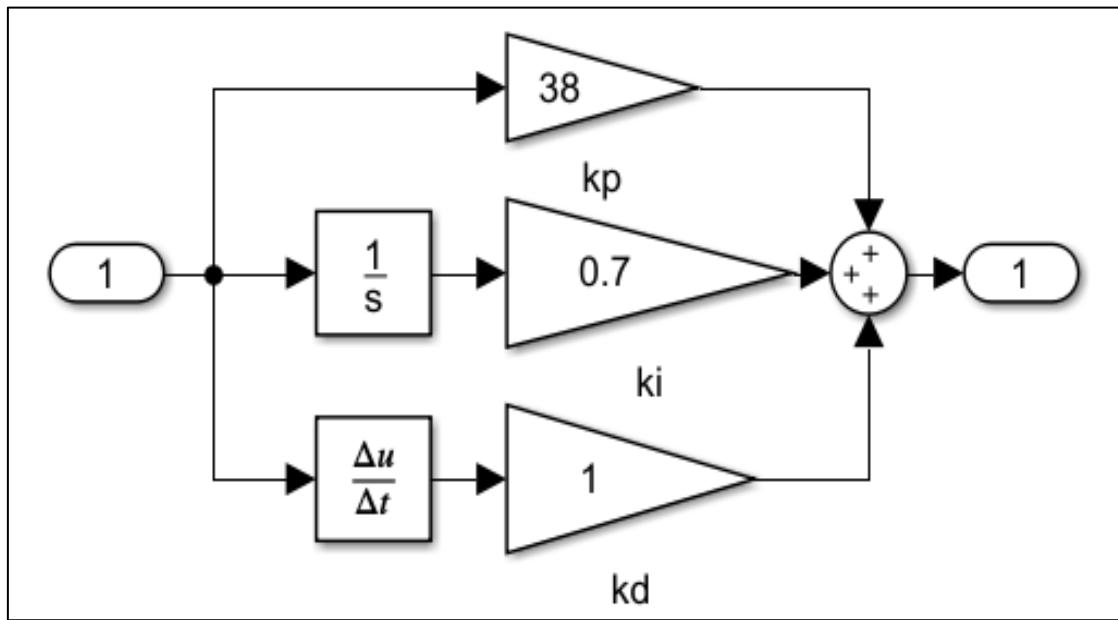
Thông số	Rise time (thời gian tăng)	Overshoot (độ vọt lố)	Settling Time (thời gian xác lập)	Steady-state Error (sai số xác lập)
K_p	↓	↑	Thay đổi nhỏ	↓
K_i	↓	↑	↑	Giảm mạnh
K_d	Thay đổi nhỏ	↓	↓	Thay đổi nhỏ

3.3.3 Ứng dụng bộ điều khiển PID để điều khiển hệ thống

Trong hệ thống này, điều khiển PID được áp dụng cho cả hai động cơ. Vì vậy, tham số bộ điều khiển PID là khác nhau cho hai trường hợp.

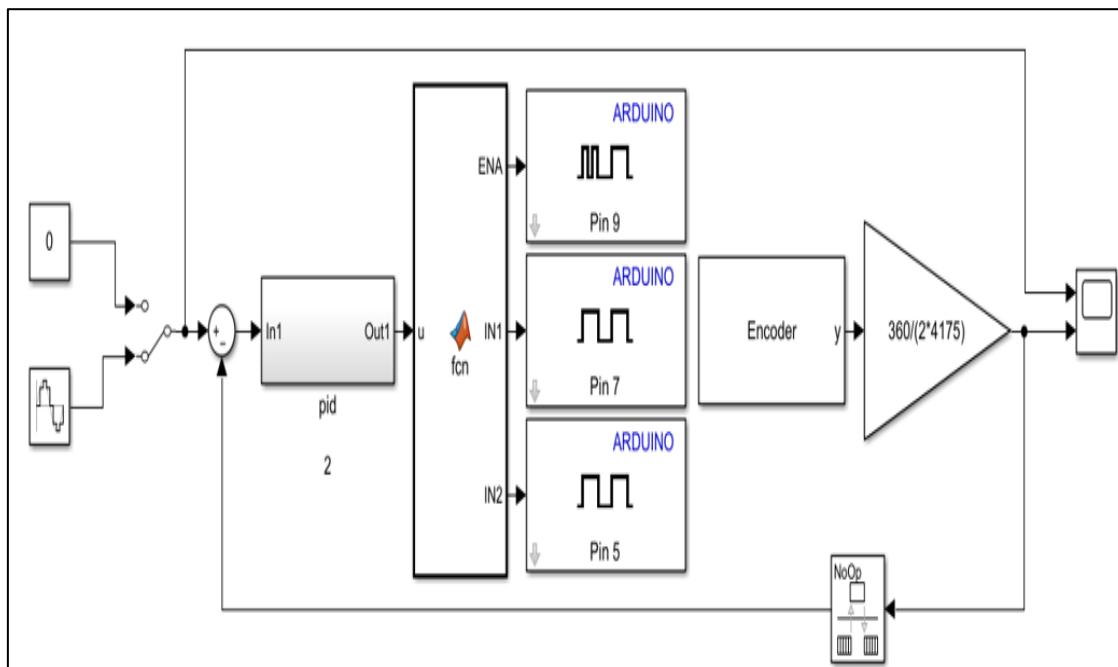
Đối với động cơ 1:

- Sau khi sử dụng phương pháp điều chỉnh bộ điều khiển PID, chúng ta có các thông số tốt của bộ điều khiển PID trong động cơ 1:



Hình 3. 15 Thông số PID cho động cơ 1

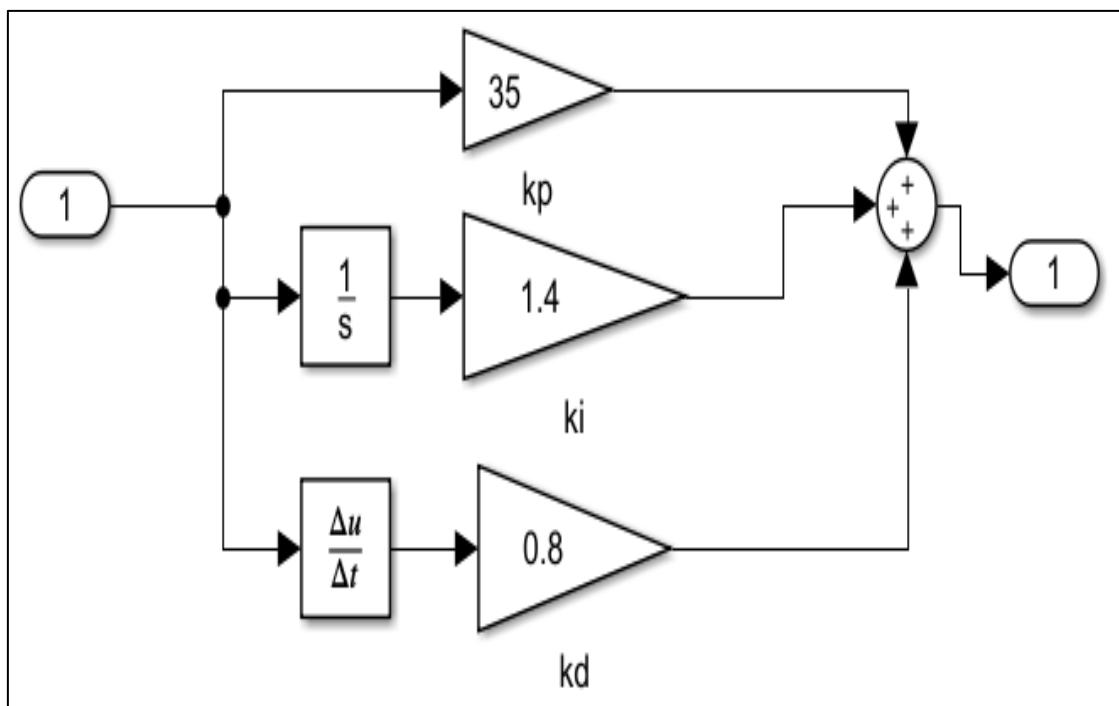
- Chu trình hoàn chỉnh của điều khiển PID động cơ 1:



Hình 3. 16 Sơ đồ bộ điều khiển PID động cơ 1 trong Matlab – Simulink

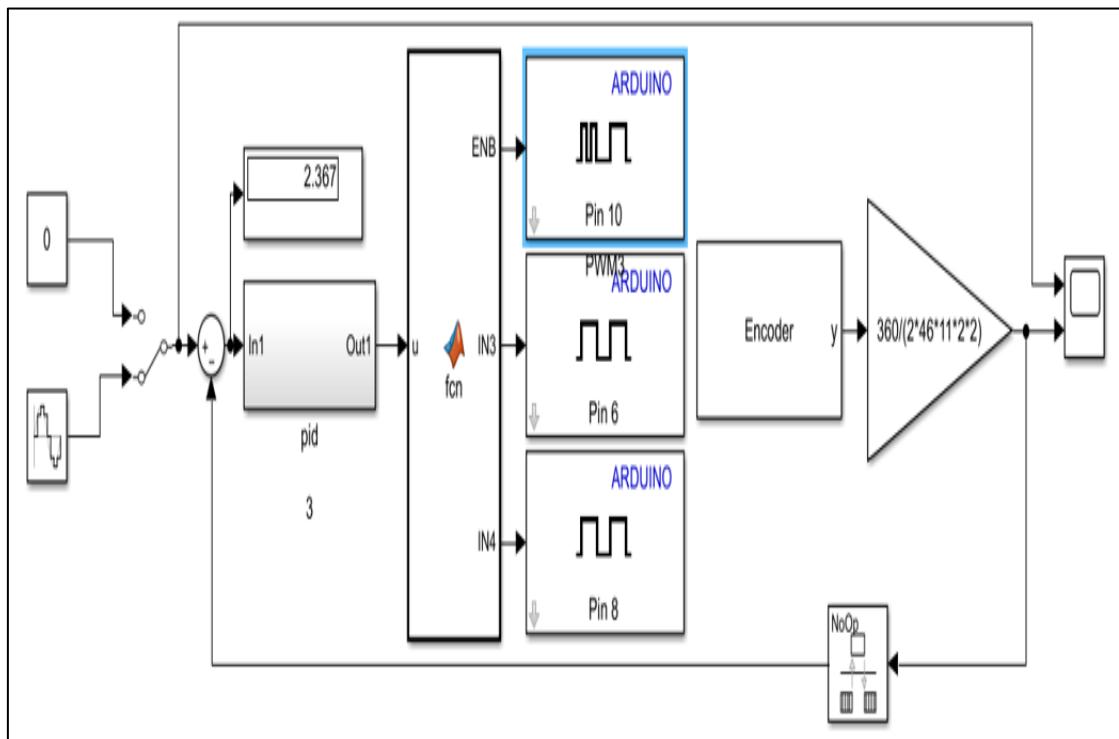
Đối với động cơ 2:

- Sau khi sử dụng phương pháp điều chỉnh bộ điều khiển PID, chúng ta có các thông số tốt của bộ điều khiển PID trong động cơ 2:



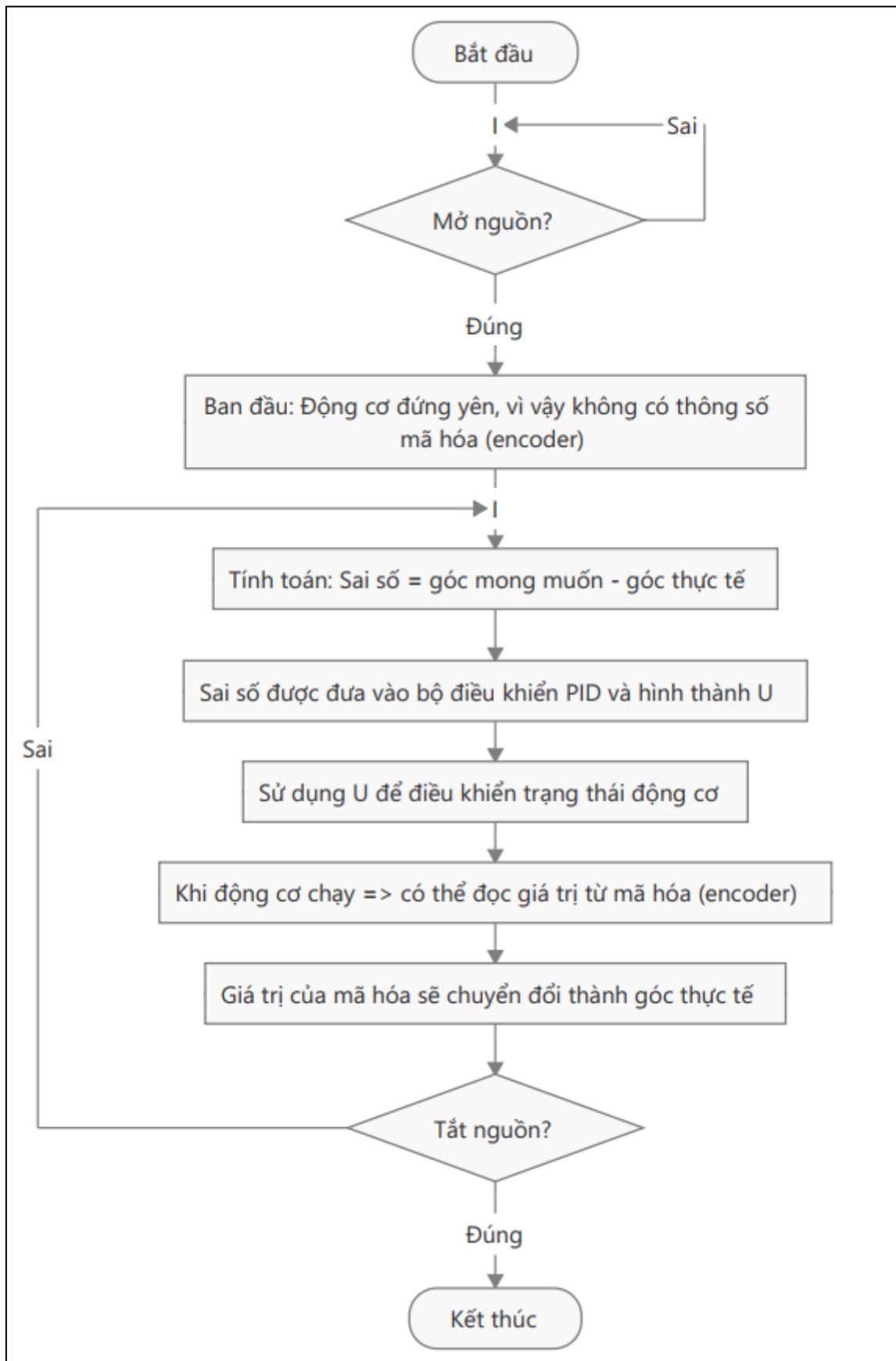
Hình 3. 17 Thông số PID cho động cơ 2

- Chu trình hoàn chỉnh của điều khiển PID động cơ 2:



Hình 3. 18 Sơ đồ bộ điều khiển PID động cơ 2 trong Matlab – Simulink

Giải thích về quy trình hoàn chỉnh của điều khiển PID cho động cơ:

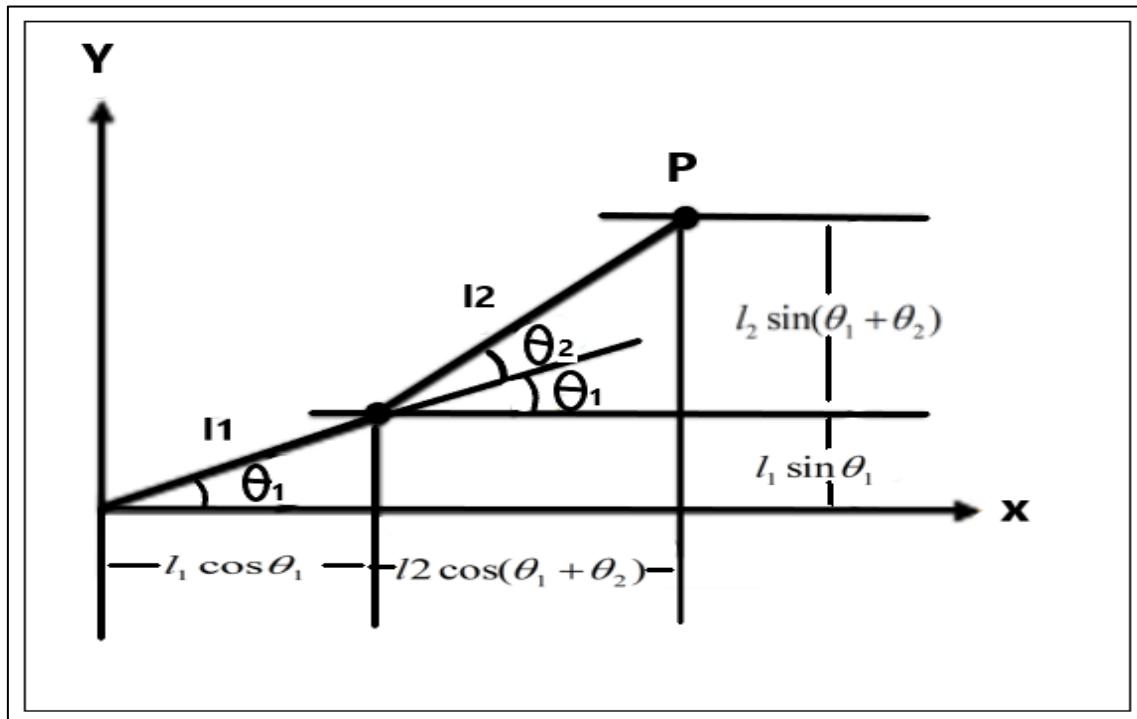


Hình 3. 19 Lưu đồ giải thích hoạt động của hệ thống

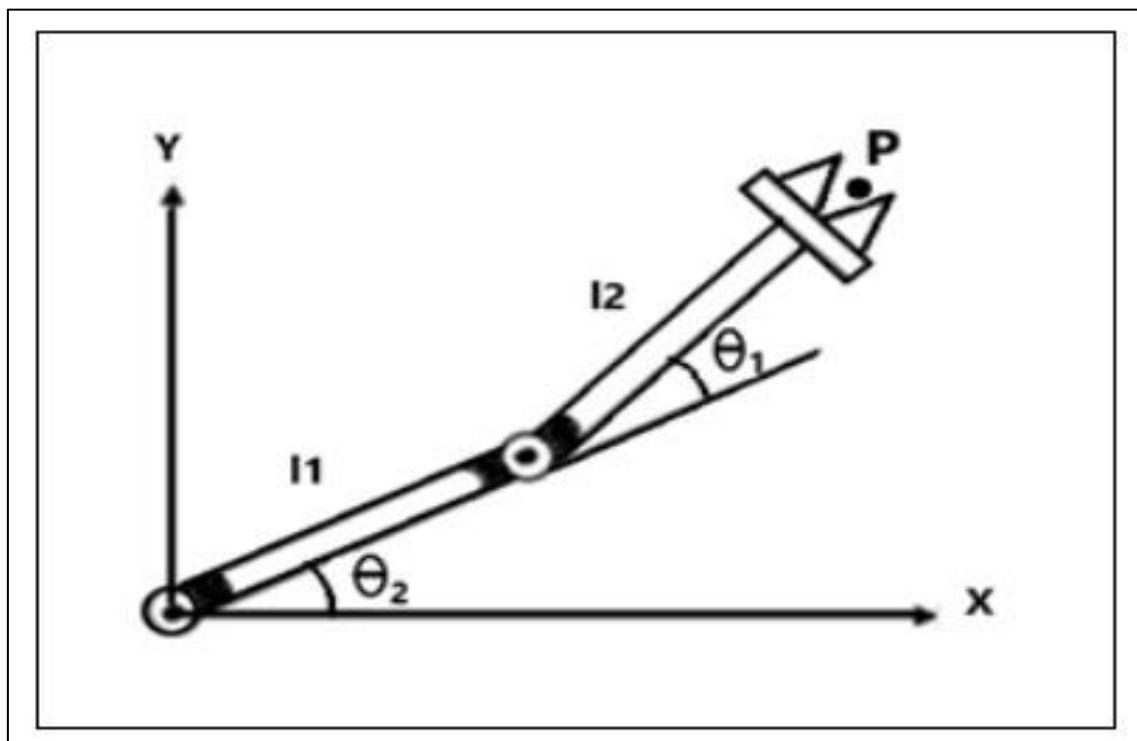
CHƯƠNG 4: ĐỘNG HỌC VÀ QUY HOẠCH QUÝ ĐẠO

4.1 Tính toán động học của cánh tay robot

Trong đề tài này do 2 DOF là mô hình cỗ điện nên chúng tôi sử dụng phương pháp hình học để tính toán động học của cánh tay robot.



Hình 4. 1 Động học ngược dựa vào hình học



Hình 4. 2 Trình điều khiển phẳng

$$p_x = l_1 c \theta_1 + l_2 c \theta_{12} \quad (2)$$

$$p_y = l_1 s \theta_1 + l_2 s \theta_{12} \quad (3)$$

Ở đó:

$$c \theta_{12} = c \theta_1 c \theta_2 - s \theta_1 s \theta_2 \quad (4)$$

Và:

$$s \theta_{12} = s \theta_1 c \theta_2 - c \theta_1 s \theta_2 \quad (5)$$

Nghiệm của θ_2 có thể được tính từ tổng bình phương của cả hai phương trình:

$$\begin{aligned} l_1 \cos \theta_1 \\ l_1 \sin \theta_1 \\ l_2 \cos(\theta_1 + \theta_2) \end{aligned} \quad (6)$$

Do đó phương trình $c^2 \theta_1 + s^2 \theta_1 = 1$ cho ở trên được đơn giản hóa như sau:

$$\begin{aligned} p_x^2 + p_y^2 &= l_1^2 + l_2^2 + 2l_1 l_2 (c \theta_1 [c \theta_1 c \theta_2 - s \theta_1 s \theta_2] + s \theta_1 [s \theta_1 c \theta_2 + c \theta_1 s \theta_2]) \\ p_x^2 + p_y^2 &= l_1^2 + l_2^2 + 2l_1 l_2 (c^2 \theta_1 c \theta_2 - c \theta_1 s \theta_1 s \theta_2 + s^2 \theta_1 c \theta_2 + c \theta_1 s \theta_1 s \theta_2) \\ p_x^2 + p_y^2 &= l_1^2 + l_2^2 + 2l_1 l_2 (c \theta_2 [c^2 \theta_1 + s^2 \theta_1]) \\ p_x^2 + p_y^2 &= l_1^2 + l_2^2 + 2l_1 l_2 c \theta_2 \end{aligned} \quad (7)$$

Và vì vậy:

$$\begin{aligned} U &= K_p e(t) + K_i \int e dt + K_d \frac{de}{dt} \\ x &= R \sin \left(\frac{2\pi t}{T} \right) \end{aligned} \quad (8)$$

Từ phương trình $c^2 \theta_i + s^2 \theta_i = 1$ ($i = 1, 2, 3, \dots$), $s \theta_2$ thu được là:

$$s \theta_2 = \pm \sqrt{1 - \left(\frac{p_x^2 + p_y^2 - l_1^2 - l_2^2}{2l_1 l_2} \right)^2} \quad (9)$$

Cuối cùng, hai nghiệm của θ_2 có thể được viết như sau:

$$\theta_2 = A \tan 2 \left(\pm \sqrt{1 - \left(\frac{p_x^2 + p_y^2 - l_1^2 - l_2^2}{2l_1 l_2} \right)^2}, \frac{p_x^2 + p_y^2 - l_1^2 - l_2^2}{2l_1 l_2} \right) \quad (10)$$

Đầu tiên, chúng ta hãy nhân mỗi vế của phương trình với và phương trình với và cộng các phương trình kết quả để tìm nghiệm của θ_1 theo tham số liên kết và biến đổi biệt θ_2 .

$$\begin{aligned} c\theta_1 p_x &= l_1 c^2 \theta_1 + l_2 c^2 \theta_1 c \theta_2 - l_2 c \theta_1 s \theta_1 s \theta_2 \\ s\theta_1 p_y &= l_1 s^2 \theta_1 + l_2 s^2 \theta_1 c \theta_2 + l_2 s \theta_1 c \theta_1 s \theta_2 \\ c\theta_1 p_x + s\theta_1 p_y &= l_1 (c^2 \theta_1 + s^2 \theta_1) + l_2 c \theta_2 (c^2 \theta_1 + s^2 \theta_1) \end{aligned} \quad (11)$$

Phương trình đơn giản thu được như sau:

$$c\theta_1 p_x + s\theta_1 p_y = l_1 + l_2 c \theta_2 \quad (12)$$

Trong bước này, nhân cả hai vế của phương trình p_x với $-s\theta_1$ và phương trình p_y với $c\theta_1$ và sau đó cộng các phương trình kết quả sẽ tạo ra:

$$\begin{aligned} -s\theta_1 p_x &= -l_1 s \theta_1 c \theta_1 - l_2 s \theta_1 c \theta_1 c \theta_2 + l_2 s^2 \theta_1 s \theta_2 \\ c\theta_1 p_x &= l_1 s \theta_1 c \theta_1 + l_2 c \theta_1 s \theta_1 c \theta_2 + l_2 c^2 \theta_1 s \theta_2 \\ -s\theta_1 p_x + c\theta_1 p_y &= l_2 s \theta_2 (c^2 \theta_1 + s^2 \theta_1) \end{aligned} \quad (13)$$

Và vì vậy:

$$c\theta_1 = \frac{p_x (l_1 + l_2 c \theta_2) + p_y l_2 s \theta_2}{p_x^2 + p_y^2} \quad (14)$$

$s\theta_1$ thu được là:

$$\theta_1 = \pm \sqrt{1 - \left(\frac{p_x (l_1 + l_2 c \theta_2) + p_y l_2 s \theta_2}{p_x^2 + p_y^2} \right)^2} \quad (15)$$

Do đó, có thể viết hai nghiệm khả thi cho $s\theta_1$:

$$\theta_1 = A \tan 2 \left(\pm \sqrt{1 - \left(\frac{p_x (l_1 + l_2 c \theta_2) + p_y l_2 s \theta_2}{p_x^2 + p_y^2} \right)^2}, \frac{p_x (l_1 + l_2 c \theta_2) + p_y l_2 s \theta_2}{p_x^2 + p_y^2} \right) \quad (16)$$

Cuối cùng, chúng ta có phương trình của θ_1 và θ_2 :

$$\theta_1 = A \tan 2 \left(\pm \sqrt{1 - \left(\frac{p_x (l_1 + l_2 c \theta_2) + p_y l_2 s \theta_2}{p_x^2 + p_y^2} \right)^2}, \frac{p_x (l_1 + l_2 c \theta_2) + p_y l_2 s \theta_2}{p_x^2 + p_y^2} \right) \quad (17)$$

$$\theta_2 = A \tan 2 \left(\pm \sqrt{1 - \left(\frac{p_x^2 + p_y^2 - l_1^2 - l_2^2}{2l_1 l_2} \right)^2}, \frac{p_x^2 + p_y^2 - l_1^2 - l_2^2}{2l_1 l_2} \right) \quad (18)$$

4.2 Quy hoạch quỹ đạo của cánh tay robot

Đối với việc lập kế hoạch quỹ đạo, chúng ta sử dụng phương trình đường tròn để tính tọa độ x và y của mọi điểm của đường tròn vẽ trong một khoảng thời gian quyết định, nghĩa là 1s nó sẽ vẽ 1 điểm nếu khoảng thời gian dài hơn thì đường tròn sẽ hoàn hảo nhất có thể.

$$x = R \sin \left(\frac{2\pi t}{T} \right) \quad (19)$$

$$y = R \cos \left(\frac{2\pi t}{T} \right) \quad (20)$$

Trong đó:

R : is the radius of the circle

t : is time (s)

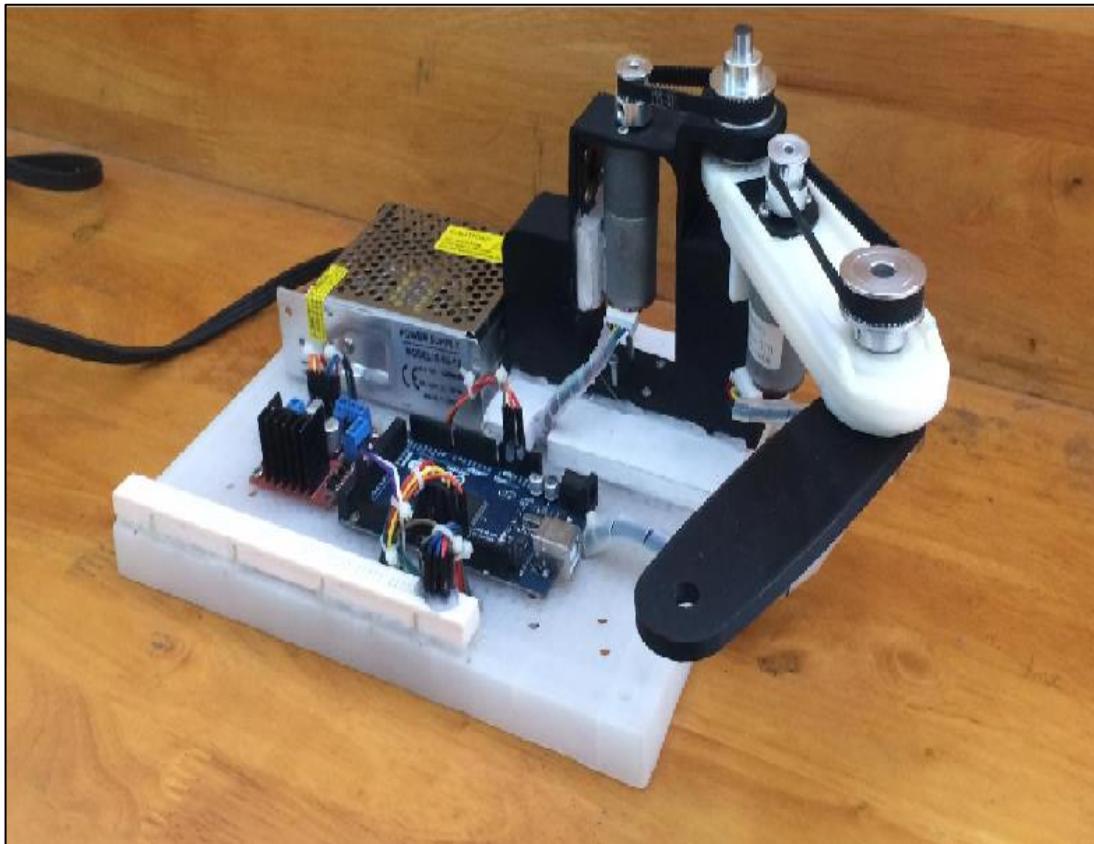
T : is period of time (s)

Sau khi có tọa độ của quy hoạch quỹ đạo, chúng tôi áp dụng tọa độ để nghịch đảo động học để điều khiển robot.

CHƯƠNG 5: KẾT QUẢ ĐẠT ĐƯỢC

5.1 Phần cứng

Đây là mô hình hoàn chỉnh của cánh tay robot 2 bậc:



Hình 5. 1 Cánh tay robot 2 bậc SCARA

Nguồn đầu vào: 220V-AC.

Giới hạn θ1: -90 độ đến 90 độ.

Giới hạn θ2: -160 độ đến 160 độ.

Chiều dài khớp nối thứ nhất: 11,5cm.

Chiều dài khớp nối thứ 2: 10 cm.

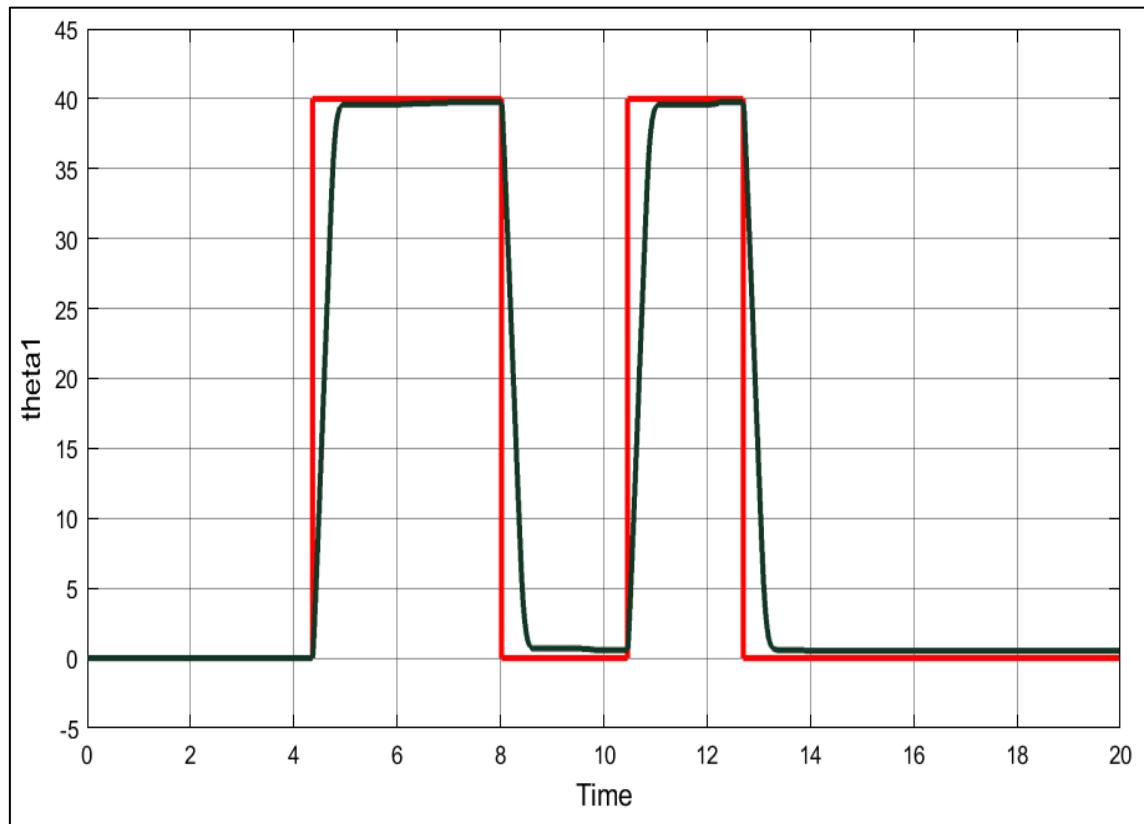
5.2 Phần mềm

Trong hình sau, nó sẽ hiển thị kết quả của θ1 với điểm đặt và chuyển động thực tế của cánh tay robot (90 độ):

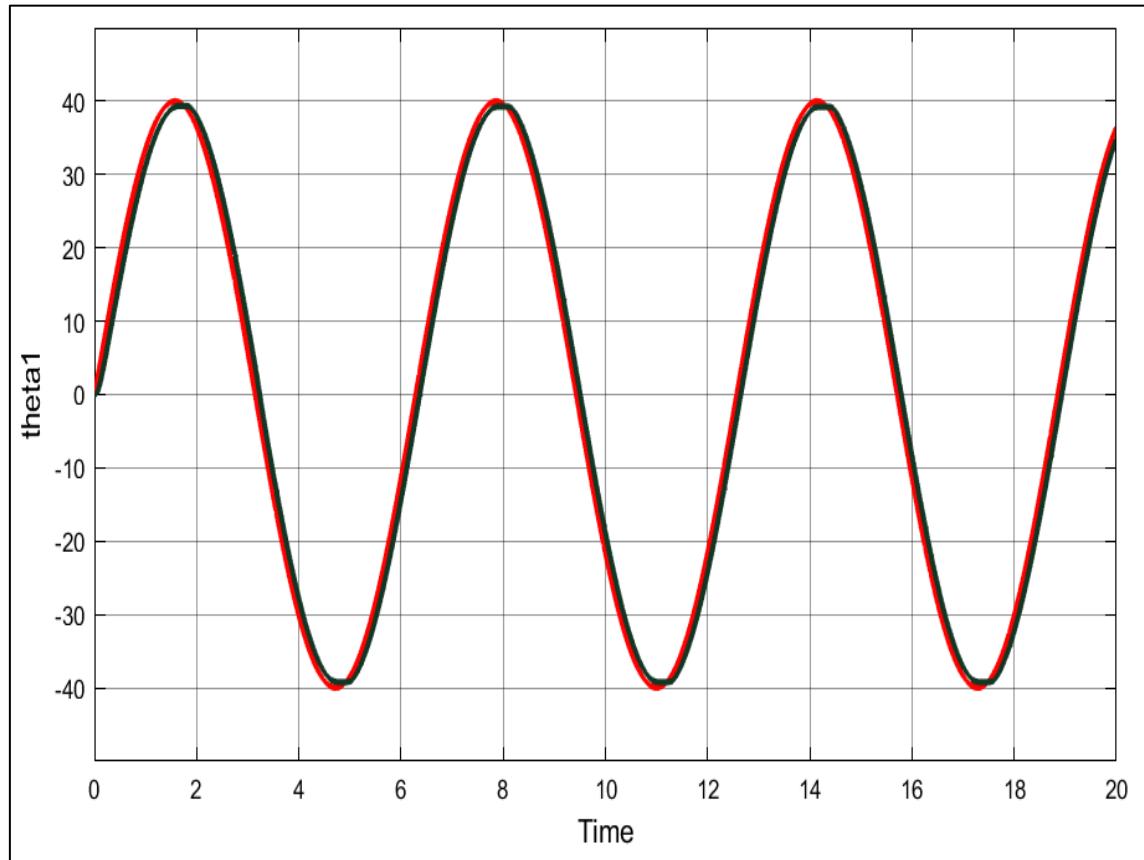
(Trong tất cả các hình này, đường màu đỏ là điểm đặt và đường màu đen là tín hiệu phản hồi của bộ mã hóa)

Trong hình sau, nó sẽ hiển thị kết quả của θ1 với điểm đặt và chuyển động thực tế của cánh tay robot (-90 độ).

Tín hiệu của động cơ thứ nhất:

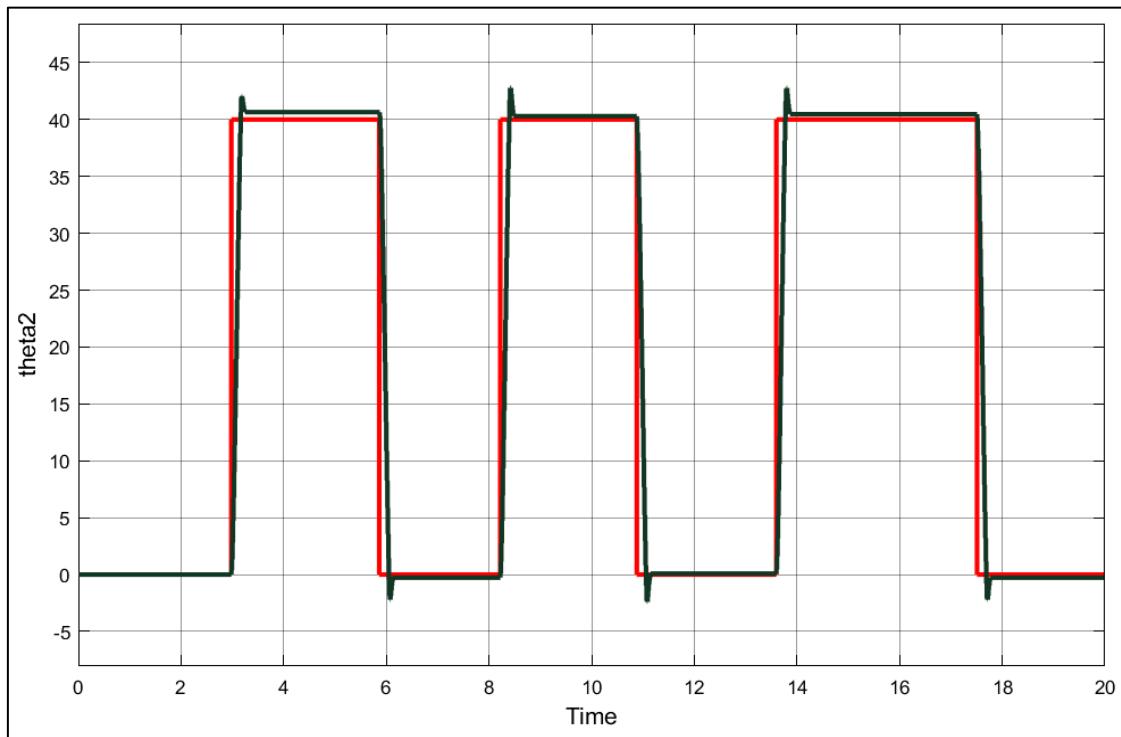


Hình 5. 2 Góc 1, tín hiệu xung vuông và đáp ứng thực tế

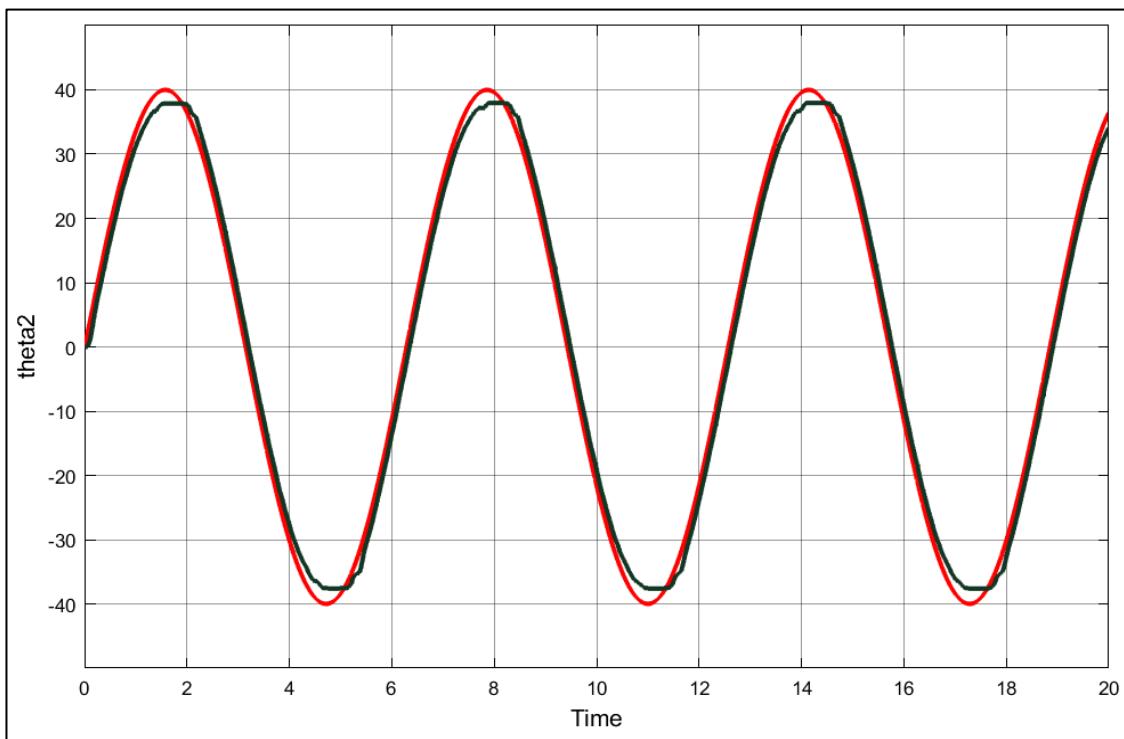


Hình 5. 3 Góc 1, tín hiệu sóng sin và đóng ứng thực tế

Tín hiệu của động cơ thứ 2:



Hình 5. 5 Góc 2, tín hiệu xung vuông và đáp ứng thực tế

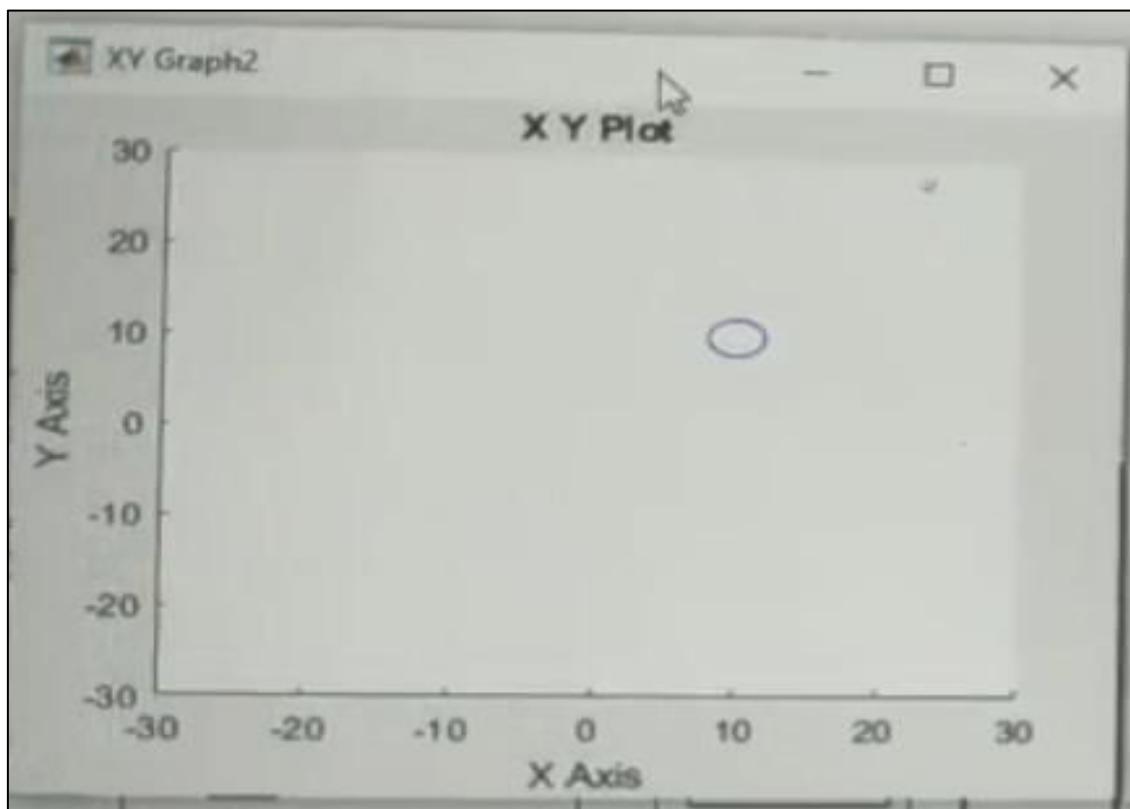


Hình 5. 4 Góc 2, tín hiệu sóng sin và đáp ứng thực tế

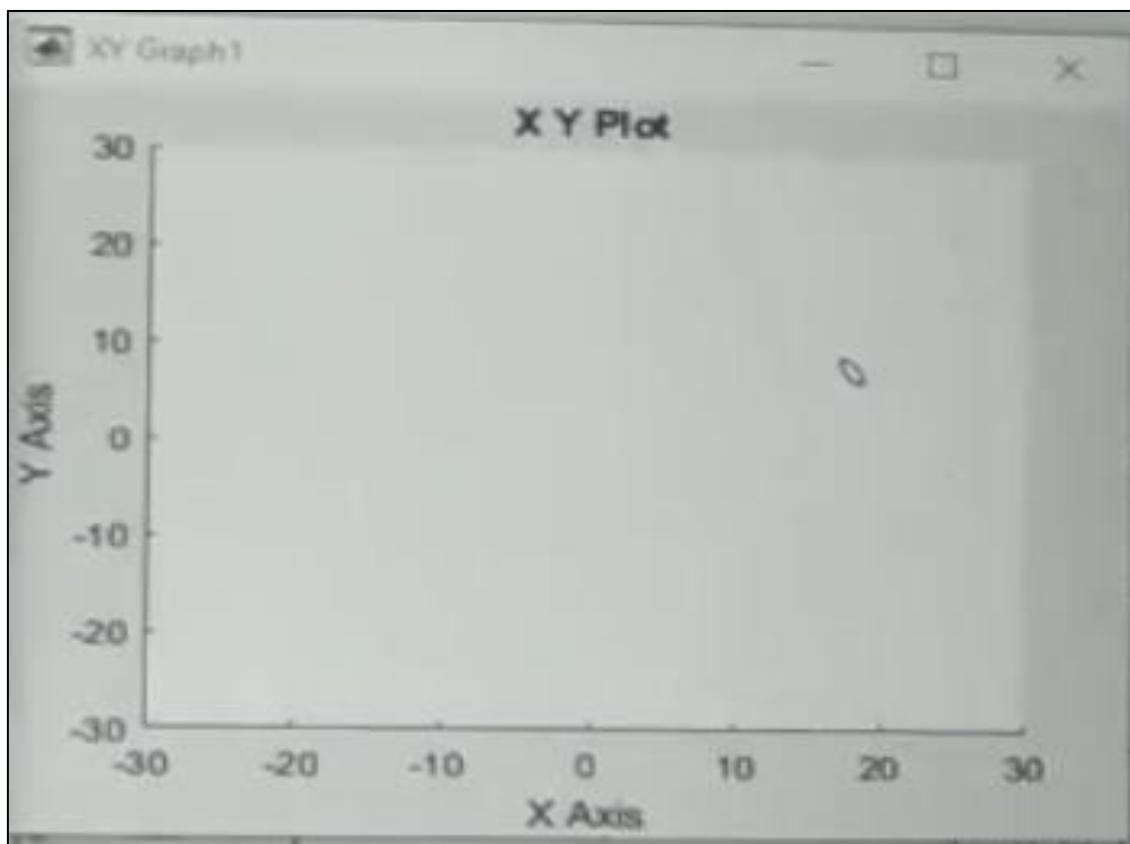
Đối với kết quả, các đối tượng hoạt động hiệu quả với cả tín hiệu sóng xung và sóng sin, sai số chỉ nằm trong khoảng 0 đến 0,2% có nghĩa là sự khác biệt giữa tín hiệu điểm đặt và tín hiệu đọc bộ mã hóa thực tế là 0 đến 0,2 độ.

CHƯƠNG 6: ĐÁNH GIÁ KẾT QUẢ

Sau đây là hình ảnh khi mô phỏng trên Matlab – Simulink:



Hình 6. 1 Ảnh quy hoạch quỹ đạo hình tròn theo lý thuyết, ảnh mục tiêu



Hình 6. 2 Ảnh đạt được của hệ thống khi hoạt động thực tế

Dự án đã sử dụng thuật toán điều khiển PID để điều khiển khớp của cánh tay robot hoạt động với sai số góc từ -2 đến 2 độ và hệ số xác lập đạt trong thời gian ngắn nhất từ 0 đến 3 giây theo tiêu chuẩn đặt ra trước khi xây dựng và chế tạo hệ thống.

Theo như kết quả mô phỏng thì sai số tồn tại giữa hình ảnh mong muốn và thực tế là không thể tránh khỏi do nhiều tính chất như: cấu tạo phần cứng, thông số bộ điều khiển PID, thời gian để đáp ứng của hệ thống. Trong quá trình mô phỏng nhóm chúng tôi nhận thấy rằng thời gian mô phỏng càng lâu thì hệ thống đáp ứng và vẽ càng chính xác hình ảnh mong muốn.

KẾT LUẬN VÀ KIẾN NGHỊ

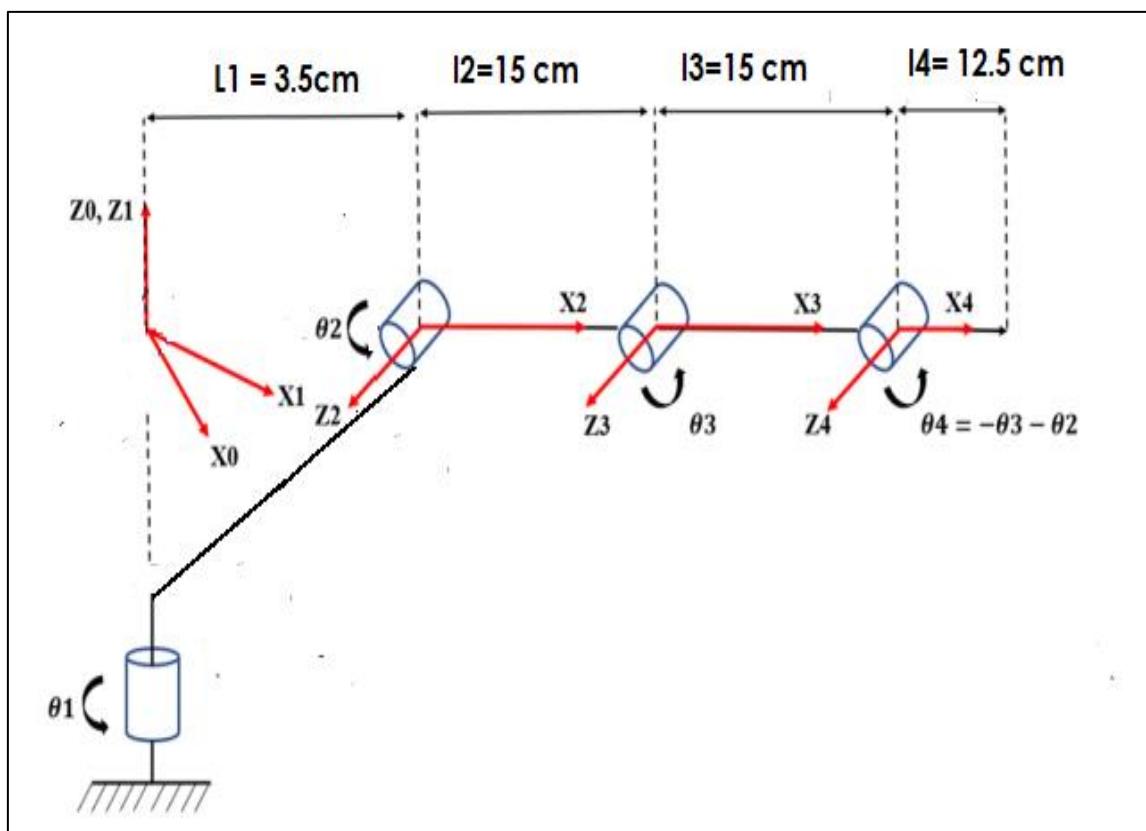
Kết Luận

Nhóm chúng tôi đã hoàn tất được mô hình cánh tay robot 2 bậc điều khiển bằng Matlab – Simulink với quỹ đạo hình tròn. Do tình hình phức tạp của dịch bệnh Covid-19 nên nhóm đã về quê và không có trang thiết bị để tiếp tục làm hoàn tất mô hình cánh tay robot 4 bậc, do vậy nhóm chúng tôi sẽ đảm bảo hoàn thành mô hình đầy đủ sau khi nhà trường cho phép đi học trở lại và tình hình dịch bệnh ổn định.

Sau đây là những lý thuyết về cánh tay robot 4 bậc tự do mà chúng tôi sẽ hoàn thành:

Động học

Động học thuận



Bảng D-H:

i	a_{i-1}	α_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	L_1	90	0	θ_2
3	L_2	0	0	θ_3
4	L_3	0	0	$-\theta_2 - \theta_3$

Từ bảng D-H ta có 4 ma trận như sau:

$$T^0_1 = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad T^1_2 = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & L1 \\ 0 & 0 & 1 & 0 \\ \sin\theta_2 & \cos\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$T^2_3 = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 & 0 & L2 \\ \sin\theta_3 & \cos\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad T^3_4 = \begin{bmatrix} \cos(\theta_2 + \theta_3) & \sin(\theta_2 + \theta_3) & 0 & L3 \\ -\sin(\theta_2 + \theta_3) & \cos(\theta_2 + \theta_3) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

Nói mà:

$$T^i_{i-1} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & a_{i-1} \\ \sin\theta_i * \cos\alpha_{i-1} & \cos\theta_i * \cos\alpha_{i-1} & -\sin\alpha_{i-1} & -d_i * \sin\alpha_{i-1} \\ \sin\theta_i * \sin\alpha_{i-1} & \cos\theta_i * \sin\alpha_{i-1} & \cos\alpha_{i-1} & d_i * \cos\alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

Và tọa độ điểm đầu cuối:

$$^4Pee = \begin{bmatrix} L4 \\ 0 \\ 0 \\ 1 \end{bmatrix},$$

Với 4 ma trận biến đổi trên ta tính được ma trận biến đổi của 3 khớp rô bốt theo công thức $T04 = T01 * T12 * T23 * T34 * ^4Pee$, ta nhận được điểm cuối của rô bốt là:

$$x = \cos\theta_1 * (L3 * \cos(\theta_2 + \theta_3) + L4 + L1 + L2 * \cos\theta_2) \quad (1)$$

$$y = \sin\theta_1 * (L3 * \cos(\theta_2 + \theta_3) + L4 + L1 + L2 * \cos\theta_2) \quad (2)$$

$$z = L2 * \sin\theta_2 + L3 * \sin(\theta_2 + \theta_3) \quad (3)$$

Động học nghịch

Lấy (2)/ (1):

$$\frac{\sin(\theta_1)}{\cos(\theta_1)} = \frac{y}{x} \Rightarrow \theta_{1a} = \text{atan2}(y, x) \text{ or } \theta_{1b} = \text{atan2}(-y, -x) \quad (4)$$

Với $\theta_{1a} = \text{atan2}(y, x)$

Lấy: (1) $x \cos(\theta_{1a}) + (2) x \sin(\theta_{1a})$, sẽ có được:

$$L3 * \cos(\theta_2 + \theta_3) + L4 + L1 + L2 * \cos(\theta_2) = x * \cos(\theta_{1a}) + y * \sin(\theta_{1a})$$

$$\Leftrightarrow L3 * \cos(\theta_2 + \theta_3) + L2 * \cos(\theta_2) = x * \cos(\theta_{1a}) + y * \sin(\theta_{1a}) - (L4 + L1) \quad (5)$$

Lấy: $(5)^2 + (3)^2$, sẽ có được:

$$L2^2 + L3^2 + 2*L2*L3*cos(\theta3) = (x*cos(\theta1a) + y*sin(\theta1a) - (L4+L1))^2 + Z^2 \quad (6)$$

$$\text{Đặt: } Da = (x*cos(\theta1a) + y*sin(\theta1a) - (L4+L1))^2 + Z^2 \quad (7)$$

$$L2^2 + L3^2 + 2*L2*L3*cos(\theta3) = Da \quad (8)$$

$$\cos(\theta3a) = (Da - L2^2 - L3^2) / (2*L2*L3) \quad (9)$$

$$\sin(\theta3a) = +\sqrt{1 - \cos(\theta3a)^2} \text{ or } \sin(\theta3a) = -\sqrt{1 - \cos(\theta3a)^2} \quad (10)$$

Có được:

$$\theta3a = \text{atan2}(+\sqrt{1 - \cos(\theta3a)^2}, \cos(\theta3a)) \quad (11)$$

$$\theta3b = \text{atan2}(-\sqrt{1 - \cos(\theta3a)^2}, \cos(\theta3a)) \quad (12)$$

Lấy $\theta1a = \text{atan2}(y, x)$ và $\theta3a = \text{atan2}(+\sqrt{1 - \cos(\theta3a)^2}, \cos(\theta3a))$ thay vào:

$$(I) \begin{cases} L3 * \cos(\theta2 + \theta3) + L2 * \cos(\theta2) = x * \cos(\theta1a) + y * \sin(\theta1a) - (L4 + L1) \\ L2 * \sin(\theta2) + L3 * \sin(\theta2 + \theta3) = Z \end{cases}$$

$$(I) \begin{cases} (L3 * \cos(\theta3a) + L2) * \cos(\theta2) - L3 * \sin(\theta3a) * \sin(\theta2) \\ = x * \cos(\theta1a) + y * \sin(\theta1a) - (L4 + L1) \\ L3 * \sin(\theta3a) * \cos(\theta2) + (L3 * \cos(\theta3a) + L2) * \sin(\theta2) = Z \end{cases} \quad (13)$$

Đặt :

$$a1 = L3 * \cos(\theta3a) + L2 \quad (14)$$

$$b1 = -L3 * \sin(\theta3a) \quad (15)$$

$$c1 = x * \cos(\theta1a) + y * \sin(\theta1a) - (L4 + L1) \quad (16)$$

$$d1 = L3 * \sin(\theta3a) \quad (17)$$

$$e1 = L3 * \cos(\theta3a) + L2 \quad (18)$$

(I):

$$a1 * \cos(\theta2) + b1 * \sin(\theta2) = c$$

$$d1 * \cos(\theta2) + e1 * \sin(\theta2) = Z$$

$\Rightarrow \sin(\theta_{2a}) = \sin(\theta_2) = (c1*d1 - a1*z) / (b1*d1 - a1*e1)$ and $\cos(\theta_{2a}) = \cos(\theta_2) = (c1*e1 - z*b1) / (a1*e1 - b1*d1)$

$$\Rightarrow \theta_{2a} = \text{atan2}(\sin(\theta_{2a}), \cos(\theta_{2a})) \quad (19)$$

Tương tự vậy: Nếu $\theta_{1a} = \text{atan2}(y, x)$; $\theta_{3b} = \text{atan2}(-\sqrt{1 - \cos(\theta_{3a})^2}, \cos(\theta_{3b}))$ vào (I) sẽ có được θ_{2b} :

$$a2 = L3 * \cos(\theta_{3b}) + L2 \quad (20)$$

$$b2 = -L3 * \sin(\theta_{3b}) \quad (21)$$

$$c2 = x * \cos(\theta_{1a}) + y * \sin(\theta_{1a}) - (L4 + L1) \quad (22)$$

$$d2 = L3 * \sin(\theta_{3b}) \quad (23)$$

$$e2 = L3 * \cos(\theta_{3b}) + L2 \quad (24)$$

$\Rightarrow \sin(\theta_{2b}) = (c2*d2 - a2*z) / (b2*d2 - a2*e2)$ and $\cos(\theta_{2b}) = (c2*e2 - z*b2) / (a2*e2 - b2*d2)$

$$\Rightarrow \theta_{2b} = \text{atan2}(\sin(\theta_{2b}), \cos(\theta_{2b})) \quad (25)$$

Với $\theta_{1b} = \text{atan2}(-y, -x)$

Lấy: (1)*cos(θ_{1b}) + (2) *sin(θ_{1b}), sẽ có được:

$$L3 * \cos(\theta_{2+3}) + L4 + L1 + L2 * \cos(\theta_2) = x * \cos(\theta_{1b}) + y * \sin(\theta_{1b})$$

$$\Leftrightarrow L3 * \cos(\theta_{2+3}) + L2 * \cos(\theta_2) = x * \cos(\theta_{1b}) + y * \sin(\theta_{1b}) - (L4 + L1) \quad (26)$$

Lấy: (26) $\wedge 2 + (3) \wedge 2$, sẽ có được:

$$L2^2 + L3^2 + 2 * L2 * L3 * \cos(\theta_3) = (x * \cos(\theta_{1b}) + y * \sin(\theta_{1b}) - (L4 + L1))^2 + Z^2 \quad (27)$$

$$\text{Đặt: } D_b = (x * \cos(\theta_{1b}) + y * \sin(\theta_{1b}) - (L4 + L1))^2 + Z^2 \quad (28)$$

$$L2^2 + L3^2 + 2 * L2 * L3 * \cos(\theta_3) = D_b \quad (29)$$

$$\cos(\theta_{3b}) = (D_b - L2^2 - L3^2) / (2 * L2 * L3) \quad (30)$$

$$\sin(\theta_{3b}) = +\sqrt{1 - \cos(\theta_{3b})^2} \text{ or } \sin(\theta_{3b}) = -\sqrt{1 - \cos(\theta_{3b})^2} \quad (31)$$

Có được:

$$\theta_{3c} = \text{atan2}(+\sqrt{1 - \cos(\theta_{3b})^2}, \cos(\theta_{3b})) \quad (32)$$

$$\theta 3d = \text{atan2}(-\sqrt{1 - \cos(\theta 3b)^2}, \cos(\theta 3b)) \quad (33)$$

Lấy $\theta 1c = \text{atan2}(-y, -x)$ và $\theta 3c = \text{atan2}(\sqrt{1 - \cos(\theta 3c)^2}, \cos(\theta 3c))$ thay vào:

$$(I) \begin{cases} L3 * \cos(\theta 2 + \theta 3) + L2 * \cos(\theta 2) = x * \cos(\theta 1b) + y * \sin(\theta 1b) - (L4 + L1) \\ L2 * \sin(\theta 2) + L3 * \sin(\theta 2 + \theta 3) = Z \end{cases}$$

$$(I) \begin{cases} (L3 * \cos(\theta 3c) + L2) * \cos(\theta 2) - L3 * \sin(\theta 3c) * \sin(\theta 2) \\ = x * \cos(\theta 1b) + y * \sin(\theta 1b) - (L4 + L1) \\ L3 * \sin(\theta 3c) * \cos(\theta 2) + (L3 * \cos(\theta 3c) + L2) * \sin(\theta 2) = Z \end{cases} \quad (34)$$

Đặt :

$$a3 = L3 * \cos(\theta 3c) + L2 \quad (35)$$

$$b3 = -L3 * \sin(\theta 3c) \quad (36)$$

$$c3 = x * \cos(\theta 1b) + y * \sin(\theta 1b) - (L4 + L1) \quad (37)$$

$$d3 = L3 * \sin(\theta 3c) \quad (38)$$

$$e3 = L3 * \cos(\theta 3c) + L2 \quad (39)$$

(I):

$$a3 * \cos(\theta 2) + b3 * \sin(\theta 2) = c$$

$$d3 * \cos(\theta 2) + e3 * \sin(\theta 2) = Z$$

$$\Rightarrow \sin(\theta 2c) = \sin(\theta 2) = (c1 * d1 - a1 * z) / (b1 * d1 - a1 * e1) \text{ and } \cos(\theta 2c) = \cos(\theta 2) = (c3 * e3 - z * b3) / (a3 * e3 - b3 * d3)$$

$$\Rightarrow \theta 2c = \text{atan2}(\sin(\theta 2c), \cos(\theta 2c))$$

(40)

Tương tự vậy: Nếu $\theta 1b = \text{atan2}(-y, -x)$; $\theta 3d = \text{atan2}(-\sqrt{1 - \cos(\theta 3d)^2}, \cos(\theta 3d))$ thay vào (I) sẽ có được $\theta 2d$:

$$a4 = L3 * \cos(\theta 3d) + L2 \quad (41)$$

$$b4 = -L3 * \sin(\theta 3d) \quad (42)$$

$$c4 = x * \cos(\theta 1b) + y * \sin(\theta 1b) - (L4 + L1) \quad (43)$$

$$d4=L3 * \sin(\theta3d) \quad (44)$$

$$e4=L3 * \cos(\theta3d) + L2 \quad (45)$$

$$\Rightarrow \sin(\theta2d) = (c4 * d4 - a4 * z) / (b4 * d4 - a4 * e4) \text{ and } \cos(\theta2d) = (c4 * e4 - z * b4) / (a4 * e4 - b4 * d4)$$

$$\Rightarrow \theta2d = \text{atan2}(\sin(\theta2d), \cos(\theta2d)) \quad (46)$$

Tôi đã thử nghiệm 4 bộ giải pháp trên bằng cách nhập chúng vào mã động học nghịch đảo vào matlab, nhập bất kỳ giải pháp nào, sau đó đưa đầu ra vào động học chuyển tiếp và trả về giống như giải pháp ban đầu cho cả 4 bộ giải pháp.

Kiến nghị:

Với mô hình cánh tay robot 4 bậc điều khiển bằng Matlab – Simulink có thể ứng dụng và các nhà máy sản xuất lắp ráp để thay thế lao động chân tay như các dây chuyền sản xuất tự động về thực phẩm, xe hơi, ...

TÀI LIỆU THAM KHẢO

- [1] Shaw, K., World Robotics Report: Global Sales of Robots Hit \$16.5B in 2018, in robotics business review 2019.
- [2] <http://vietpowertech.com/canh-tay-robot-epson-g1-mini-scara-371.html#>
- [3] http://techbyviv.blogspot.com/p/robotics_28.html
- [4]"Algorithm,"Wikipedia.[Online].Available:<https://en.wikipedia.org/wiki/Algorithm#:~:text=In%20mathematics%20and%20computer%20science,or%20to%20performance%20a%20computation.>
- [5]"Control(management),"Wikipedia,[Online].Available:[https://en.wikipedia.org/wiki/Control_\(management\)](https://en.wikipedia.org/wiki/Control_(management)).
- [6] "Control theory: types of control algorithm," Visaya, [Online]. Available: <https://visaya.solutions/en/article/control-theory-types-of-control-algorithm/>.
- [7] "PID controller," Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/PID_controller.
- [8] https://vi.wikipedia.org/wiki/Logic_m%E1%BB%9D
- [9] A.A. Khan , N. Rapal, "Fuzzy PID Controller: Design, Tuning and Comparison with Conventional PID Controller," IEEE Xplore, [Online]. Available: <https://ieeexplore.ieee.org/document/1703213>.
- [10] <https://nshopvn.com/product/arduino-mega2560-r3-atmega16u2/>
- [11] <https://nshopvn.com/product/dong-co-dc-giam-toc-ga25-encoder/>
- [12] <https://icdayroi.com/driver-motor-l298n>
- [13] <https://nshopvn.com/product/puly-gt2-36-rang-duong-kinh-22-4mm/>
- [14]<https://people.ece.cornell.edu/land/courses/ece4760/FinalProjects/s2012/fas57nyp7/Site/pidcontroller.html>

PHỤ LỤC

Code

Quy hoạch quỹ đạo

```
function [x, y] = fcn(t ,T,r)
x =10+r*sin((2*pi*t)/T);
y =10+r*cos((2*pi*t)/T);
```

Động học nghịch

```
function [t1, t2, t1_do, t2_do] = fcn(x, y)
%INVERSEKINEMATIC
l1=11.5;
l2=10;
c2=(x^2+y^2-l1^2-l2^2)/(2*l1*l2);
s2=sqrt(abs(1-c2^2));
t2=atan2(s2,c2);
t2_do=atan2(s2,c2)*180/3.14;
c1=(x*(l1+l2*c2)+y*l2*s2)/((l1+l2*c2)^2+(l2*s2)^2);
s1=(y.* (l1+l2*c2)-x.* l2*s2)/((l1+l2*c2)^2+(l2*s2)^2);
t1=atan2(s1,c1);
t1_do=atan2(s1,c1)*180/3.14;
```

Động học thuận

```
function [px ,py] = fcn(t1, t2)
l1=11.5;
l2=10;
px=l2*cos(t1 + t2) + l1*cos(t1);
py=l2*sin(t1 + t2) + l1*sin(t1);
```

Điều khiển L298

```
function [ENA,IN1,IN2] = fcn(u)
```

if $u > 0$

```

IN1=1;IN2=0;
elseif u<0
    IN1=0;IN2=1;
else
    IN1=0;IN2=0;
end
ENA=abs(u);
End

```

Điều khiển encoder L298

```

classdef Encoder_arduino < realtime.internal.SourceSampleTime ...
    & coder.ExternalDependency ...
    & matlab.system.mixin.Propagates ...
    & matlab.system.mixin.CustomIcon
    %Read the position of a quadrature encoder.
    % Copyright 2014 The MathWorks, Inc.
    %#codegen
    %#ok<*EMCA>
    properties (Nontunable)
        Encoder = 0
        PinA = 2
        PinB = 3
    end
    properties (Constant, Hidden)
        % AvailablePin specifies the range of values allowed for Pin. You
        % can customize the AvailablePin for a particular board. For
        % example, use AvailablePin = 2:13 for Arduino Uno.
        AvailablePin = 0:53;
        MaxNumEncoder = 3
    end

```

methods

```
% Constructor

function obj = Encoder_arduino(varargin)
    coder.allowpcode('plain');

    % Support name-value pair arguments when constructing the object.
    setProperties(obj,nargin varargin{:});

end

function set.PinA(obj,value)
    coder.extrinsic('sprintf') % Do not generate code for sprintf
    validateattributes(value,...

        {'numeric'},...
        {'real','nonnegative','integer','scalar'},...
        ", ...
        'PinA');

    assert(any(value == obj.AvailablePin), ...

        'Invalid value for Pin. Pin must be one of the following: %s', ...
        sprintf('%d ', obj.AvailablePin));

    obj.PinA = value;

end

function set.PinB(obj,value)
    coder.extrinsic('sprintf') % Do not generate code for sprintf
    validateattributes(value,...

        {'numeric'},...
        {'real','nonnegative','integer','scalar'},...
        ", ...
        'PinB');

    assert(any(value == obj.AvailablePin), ...

        'Invalid value for Pin. Pin must be one of the following: %s', ...
        sprintf('%d ', obj.AvailablePin));

    obj.PinB = value;
```

```

end

function set.Encoder(obj,value)
    validateattributes(value,...)
    {'numeric'},...
    {'real','nonnegative','integer','scalar','>='0,'<='obj.MaxNumEncoder},...
    ", ...
    'Encoder');

    obj.Encoder = value;
end

end

methods (Access=protected)

function setupImpl(obj)
    if coder.target('Rtw')
        % Call: void enc_init(int enc, int pinA, int pinB)
        coder.cinclude('encoder_arduino.h');
        coder.ceval('enc_init', obj.Encoder, obj.PinA, obj.PinB);
    end
end

function y = stepImpl(obj)
    y = int32(0);
    if coder.target('Rtw')
        % Call: int enc_output(int enc)
        y = coder.ceval('enc_output', obj.Encoder);
    end
end

function releaseImpl(obj) %#ok<MANU>
end

methods (Access=protected)

%% Define output properties

```

```

function num = getNumInputsImpl(~)
    num = 0;
end

function num = getNumOutputsImpl(~)
    num = 1;
end

function flag = isOutputSizeLockedImpl(~,~)
    flag = true;
end

function varargout = isOutputFixedSizeImpl(~,~)
    varargout{1} = true;
end

function flag = isOutputComplexityLockedImpl(~,~)
    flag = true;
end

function varargout = isOutputComplexImpl(~)
    varargout{1} = false;
end

function varargout = getOutputSizeImpl(~)
    varargout{1} = [1,1];
end

function varargout = getOutputDataTypeImpl(~)
    varargout{1} = 'int32';
end

function icon = getIconImpl(~)
    % Define a string as the icon for the System block in Simulink.
    icon = 'Encoder';
end

```

```

methods (Static, Access=protected)

    function simMode = getSimulateUsingImpl(~)
        simMode = 'Interpreted execution';
    end

    function isVisible = showSimulateUsingImpl
        isVisible = false;
    end

end

methods (Static)

    function name = getDescriptiveName()
        name = 'Encoder';
    end

    function b = isSupportedContext(context)
        b = context.isCodeGenTarget('rtw');
    end

function updateBuildInfo(buildInfo, context)
    if context.isCodeGenTarget('rtw')
        % Update buildInfo
        rootDir = fullfile(fileparts(mfilename('fullpath')), '..', 'src');
        buildInfo.addIncludePaths(rootDir);
        buildInfo.addIncludeFiles('encoder_arduino.h');
        buildInfo.addSourceFiles('encoder_arduino.cpp', rootDir);
    end
end
end

```

Mô phỏng hệ thống trên Matlab – Simulink

