

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ



BÁO CÁO TỔNG KẾT
ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN
NĂM 2020

THIẾT KẾ, CHẾ TẠO VÀ ĐIỀU KHIỂN CÁNH TAY
ROBOT 4 BẬC TỰ DO

Sinh viên thực hiện:

- | | | |
|-------------------|-------------|----------------|
| 1. Nguyễn Tú Ninh | Lớp: K62-M3 | Khoa: CHKT&TĐH |
| 2. Lê Hoài Nam | Lớp: K62-M3 | Khoa: CHKT&TĐH |
| 3. Đào Thành Lộc | Lớp: K62-M1 | Khoa: CHKT&TĐH |
| 4. Bùi Minh Hiếu | Lớp: K61-M | Khoa: CHKT&TĐH |

Cán bộ hướng dẫn: TS. Nguyễn Ngọc Linh

HÀ NỘI - 2020

LỜI CAM ĐOAN

Nhóm nghiên cứu xin cam đoan công trình này là công trình nghiên cứu của nhóm dưới sự hướng dẫn của TS. Nguyễn Ngọc Linh. Nội dung nghiên cứu không sao chép từ bất kì công trình nghiên cứu nào khác. Những phần tham khảo đã được nhóm trích dẫn đầy đủ trong phần tài liệu tham khảo.

Nếu có bất kỳ sự gian lận nào, nhóm nghiên cứu xin chịu hoàn toàn trách nhiệm trước hội đồng về kết quả của công trình nghiên cứu này.

Nhóm sinh viên thực hiện

Giảng viên hướng dẫn

TÓM TẮT CÔNG TRÌNH NGHIÊN CỨU

Sau một quá trình tìm tòi nghiên cứu các vấn đề về Robot công nghiệp nhất là về cánh tay robot(Arm Robot), nhóm sinh viên nghiên cứu chúng em dưới sự hướng dẫn của Ts. Nguyễn Ngọc Linh đã lựa chọn đề tài “Thiết kế, mô phỏng và điều khiển cánh tay robot 4 bậc tự do” bao gồm:

- Xây dựng phương pháp giải bài toán động học tay máy(thuận và nghịch).
- Tính toán, thiết kế mô hình cánh tay robot 4 bậc tự do
- Mô phỏng các chuyển động cơ học của cánh tay robot 4 bậc tự do trên máy tính sử dụng Matlab/Simulink.
- Xây dựng mô hình cánh tay robot thực tế và giao tiếp với máy tính thông qua cổng USB của Arduino. Đồng thời kết nối các phần mềm thiết kế với phần mềm mô phỏng và mô hình thực tế.
- Thiết kế giao diện người dùng tích hợp trong Matlab(GUI Matlab) điều khiển song song đồng thời mô phỏng trên máy tính và mô hình thực nghiệm của cánh tay robot.

MỤC LỤC

I. ĐẶT VẤN ĐỀ.....	2
1. Đối tượng và mục đích nghiên cứu	2
2. Những vấn đề cần được nghiên cứu.....	2
II. GIẢI QUYẾT VẤN ĐỀ.....	2
1. Cách tiếp cận và phương pháp nghiên cứu	3
1.1. Cách tiếp cận	3
1.2. Phương pháp nghiên cứu	3
2. Quá trình thực hiện	3
2.1. Tổng quan về cánh tay robot	3
2.1.1. Giới thiệu.....	3
2.1.2. Phân loại.....	4
2.1.3. Cấu trúc cơ bản tay máy robot	5
2.1.4. Ứng dụng.....	6
2.2. Xây dựng bài toán động học tay máy.....	7
2.2.1. Các khái niệm ban đầu	7
2.2.2. Đặt vấn đề	16
2.2.3. Xác định trạng thái của robot tại “điểm tác động cuối”	16
2.2.4. Mô hình động học.....	16
2.2.5. Trình tự thiết lập hệ phương trình động học của robot.....	19
2.2.6. Động học thuận tay máy	19
2.2.7. Bài toán động học nghịch(động học ngược) tay máy.....	22
2.3. Thiết kế cánh tay robot 4 bậc tự do	25
2.3.1. Giới thiệu phần mềm Autodesk Inventor	25
2.3.2. Thiết kế cánh tay robot 4 bậc tự do.....	26
2.4. Mô hình hóa và mô phỏng.....	27
2.4.1. Giới thiệu về Simulink và Simscape Multibody	27
2.4.2. Mô phỏng	28
2.5. Xây dựng mô hình thực tế cánh tay robot 4 bậc tự do.....	30

2.5.1. Tính toán lựa chọn động cơ servo phù hợp cho các khớp quay	30
2.5.2. Phần cứng sử dụng.....	31
2.5.3. Lắp ráp mô hình.....	34
2.6. Điều khiển	35
2.6.1. Tổng quan mô hình điều khiển	35
2.6.2. Điều khiển mô hình trong mô phỏng với thời gian thực	37
2.6.3. Điều khiển mô hình thực nghiệm cánh tay robot.....	39
2.6.4. Điều khiển đồng thời mô phỏng và mô hình thực nghiệm	40
III. KẾT LUẬN.....	46
1. Kết luận	46
2. Hạn chế	46
3. Hướng phát triển	46
TÀI LIỆU THAM KHẢO	

DANH MỤC HÌNH VẼ

Hình 2.1: Robot Descartes.....	4
Hình 2.2: Robot hệ tọa độ trụ	4
Hình 2.3: Robot hệ tọa cầu	5
Hình 2.4: Robot Scara	5
Hình 2.5: Robot khớp bản lề	5
Hình 2.6: Sơ đồ khối cấu trúc cơ bản của robot công nghiệp	6
Hình 2.7: Cấu trúc cơ bản của robot công nghiệp.....	6
Hình 2.8: Ứng dụng của robot trong sản xuất	7
Hình 2.9: Các hệ tọa độ đối với 2 khâu động liên tiếp.....	18
Hình 2.10: Các hệ tọa độ đối với 2 khâu động liên tiếp có khớp tịnh tiến.....	18
Hình 2.11: Sơ đồ mô tả tổng quan bài toán động học thuận tay máy.	20
Hình 2.12: Hệ tọa độ của cánh tay robot 4 bậc tự do	20
Hình 2.13: Sơ đồ mô tả tổng quan bài toán động học nghịch tay máy	22
Hình 2.14: Phần mềm Autodesk Inventor Professional 2018	25
Hình 2.15: Các khâu của robot	26
Hình 2.16: Các khâu của gripper	27
Hình 2.17: Hình ảnh gripper hoàn chỉnh	27
Hình 2.18: Các khâu của robot	27
Hình 2.19: Ví dụ ứng dụng của SimMechanics	28
Hình 2.20: Sơ đồ mô phỏng trong Simulink	28
Hình 2.21: Cấu tạo khối “Control system”	29
Hình 2.22: Kết quả mô phỏng được hiện thị trong Mechanics Explorers.....	30
Hình 2.23: Mô hình thể hiện các lực	30
Hình 2.24: Động cơ servo sử dụng trong mô hình	31
Hình 2.25: Arduino Mega 2560.....	31
Hình 2.26: Cấu tạo cơ bản của Arduino Mega2560	32
Hình 2.27: Động cơ MG996R	33
Hình 2.28: Động cơ Servo SG90	33
Hình 2.29: Lắp ráp mô hình	34
Hình 2.30: Mô hình thực nghiệm cánh tay robot	34
Hình 2.31: Không gian hoạt động của robot	35
Hình 2.32: Sơ đồ khối hệ thống điều khiển vị trí	37
Hình 2.33: Sơ đồ khối tổng quan hệ thống điều khiển.	37
Hình 2.34: Sơ đồ khối điều khiển độc lập từng khớp trong mô phỏng	38
Hình 2.35: Khối “PID Controller” trong Simulink	38
Hình 2.36: Cấu trúc cơ bản của khối PID Controller	38
Hình 2.37: Tín hiệu vào và ra của các khớp trong mô phỏng ban đầu.....	39
Hình 2.38: Sơ đồ kết nối các động cơ Servo với Arduino	39
Hình 2.39: Sơ đồ khối điều khiển động cơ servo	40

Hình 2.40: Sơ đồ trình tự thực hiện với GUI.....	41
Hình 2.41: Giao diện bảng điều khiển hoàn chỉnh	41
Hình 2.42: Lưu đồ thuật toán hệ thống điều khiển.....	43
Hình 2.43: Bảng điều khiển robot ở các chế độ khác nhau	44
Hình 2.44: Mô phỏng robot trên máy tính.....	44
Hình 2. 45: Sơ đồ tín hiệu tại các khớp xoay trong mô phỏng ở chế độ tự động.....	44
Hình 2.46: Sơ đồ tín hiệu của các động cơ servo	45

DANH MỤC BẢNG

Bảng 1: Tham số động học của Robot.....	21
Bảng 2: Các khối simulink sử dụng mô phỏng	29
Bảng 3: Thông số kỹ thuật Arduino Mega 2560	32
Bảng 4: Thông số kỹ thuật của động cơ servo MG996R	33
Bảng 5: Thông số kỹ thuật của động cơ servo SG90.	34
Bảng 6: Giới hạn của các khớp xoay	35
Bảng 7: Các khối simulink điều khiển mô hình	40
Bảng 8: Các chức năng của bảng điều khiển cánh tay robot.....	43

DANH MỤC CÁC TỪ VIẾT TẮT

GUI	Graphical User Interface (<i>Giao diện đồ họa người dùng</i>)
DH	Denavit – Hartenberg(<i>Tên riêng – người đầu tiên nghiên cứu ra phương pháp động học DH</i>)
DOF	Degrees Of Freedom (<i>Bậc tự do</i>)
PID	Proportional Integral Derivative (<i>Bộ điều khiển vi tích phân tỉ lệ</i>)
PC	Personal Computer (<i>Máy tính cá nhân</i>)
CAD	Computer Aided Design (<i>Máy tính hỗ trợ thiết kế</i>)
3D	3 - Dimension (<i>3 chiều</i>)
2D	2 - Dimension (<i>2 chiều</i>)
HIL	Hardware in the loop(<i>vòng lặp phần cứng</i>)
PWM	Pulse-width modulation (<i>Điều chế độ rộng xung</i>)
USB	Universal Serial Bus (<i>Chuẩn kết nối có dây</i>)

I. ĐẶT VẤN ĐỀ

1. Đối tượng và mục đích nghiên cứu

Hiện nay, cách mạng công nghiệp 4.0 đang mang đến cho nhân loại cơ hội để thay đổi bộ mặt của các nền kinh tế. Tự động hóa trong công nghiệp bằng Robot chính là chìa khóa chiến lược để nâng cao hơn nữa khả năng cạnh tranh cho tất cả các doanh nghiệp và phát triển kinh tế. Công nghiệp robot phát triển mạnh, mang lại những lợi ích to lớn cho con người và xã hội... Ở nước ta hiện nay cũng đang trong thời kì công nghiệp hóa, hiện đại hóa, vì vậy việc đẩy mạnh sản xuất, tăng năng suất lao động bằng việc ứng dụng công nghệ tự động hóa và robot. Ngành công nghiệp robot hiện nay đang phát triển rất mạnh mẽ với rất nhiều chủng loại robot làm việc trong nhiều môi trường khác nhau như trong công xưởng, ngoài vũ trụ, dưới biển hay thậm chí là trong các đường ống nước thải. Phổ biến nhất trong số đó phải kể tới robot công nghiệp. Cùng với đó, vấn đề sản xuất robot để phục vụ sản xuất đang là một vấn đề cấp bách mà phải trải qua rất nhiều công đoạn phức tạp khác nhau.

Chính vì vậy, việc nghiên cứu và phát triển robot công nghiệp cũng cần được đẩy mạnh để có thể đáp ứng được nhu cầu của thị trường và những yêu cầu cụ thể trong từng môi trường làm việc khác nhau. Việc liên kết, kết nối từng công đoạn nhỏ như thiết kế, mô phỏng chuyển động, điều khiển,... cũng sẽ làm việc nghiên cứu và phát triển trở nên đơn giản hơn mà vẫn đảm bảo được các vấn đề cơ bản về kĩ thuật. Nhận thức được vấn đề đó, chúng em đã lựa chọn đề tài “Thiết kế, mô phỏng và điều khiển cánh tay robot 4 bậc tự do” với hi vọng có thể vận dụng những kiến thức đã được học tập tại giảng đường đại học để giải quyết vấn đề đã đặt ra.

2. Những vấn đề cần được nghiên cứu

- Tìm hiểu về cấu trúc, hoạt động cánh tay robot 4 bậc tự do.
- Tìm hiểu phương pháp giải bài toán động học tay máy.
- Thiết kế mô hình robot công nghiệp 4 bậc tự do.
- Xây dựng mô phỏng chuyển động cánh tay robot.
- Xây dựng mô hình cánh tay robot.
- Xây dựng phương pháp, thuật toán điều khiển đồng thời mô hình và mô phỏng.
- Xây dựng giao diện trên máy tính để điều khiển, hiển thị các thông số quan trọng của cánh tay robot.

II. GIẢI QUYẾT VẤN ĐỀ

1. Cách tiếp cận và phương pháp nghiên cứu

1.1. Cách tiếp cận

- Dựa vào các kiến thức đã được học và tích lũy tại trường lớp: Ở môi trường giáo dục của trường Đại học Công nghệ, chúng em đã được học các kiến thức thiết kế, kiến thức về lập trình lập trình C, các lý thuyết chuyên ngành về điện tử, đo lường, điều khiển tự động,... Đó là công cụ cho chúng em tìm tòi và phát triển các hướng giải quyết các vấn đề đã đặt ra.
- Tiếp cận qua Internet, các thông tin từ các nguồn khác nhau: Qua các bài báo khoa học, các thông tin tìm hiểu được qua các trang công nghệ và các kênh thông tin thời sự, chúng em đã tích lũy được một lượng kiến thức nhất định về robot công nghiệp, hướng nghiên cứu và phát triển robot.

1.2. Phương pháp nghiên cứu

- Tham khảo các tài liệu giới thiệu về robot, các cơ cấu robot và đặc biệt là các mô hình cấu cánh tay robot 4 bậc tự do. Dựa trên đó để thiết kế được hệ thống robot 4 bậc tự do làm cơ sở để mô hình hóa và mô phỏng.
- Nghiên cứu và áp dụng các giải thuật điều khiển đã được học trên giảng đường để điều khiển mô phỏng trên máy tính và mô hình thực tế.
- Nghiên cứu xây dựng giao diện người dùng đồ họa GUI để thiết kế bảng điều khiển, điều khiển song song cả mô hình và mô phỏng, đồng thời hiển thị các thông số quan trọng của robot như tọa độ, vị trí, góc quay của các khớp,...

2. Quá trình thực hiện

2.1. Tổng quan về cánh tay robot

2.1.1. Giới thiệu

Cánh tay robot là một cỗ máy phục vụ trong quy trình sản xuất công nghiệp được vận hành bởi con người. Nổi bật nhất là thiết kế linh hoạt, thao tác nhanh nhẹn, tỉ mỉ và có khả năng hoàn thiện cả những phân chi tiết sản phẩm nhỏ nhất.

Cánh tay robot hoạt động rất linh hoạt với cơ chế khớp ngón hết như một con người. Ngày nay, nhờ những ưu điểm vượt trội, các cỗ máy cơ khí này đang ngày càng được ứng dụng nhiều vào nhiều lĩnh vực công nghiệp khác nhau.

Theo đó, những cỗ máy này sẽ thay thế bàn tay của con người trong các công đoạn lắp ráp, sửa chữa, thay thế linh kiện siêu nhỏ trong nhà máy. Cơ chế hoạt động của chúng dựa trên liên kết giữa bộ điều khiển và các khớp nối chuyển

động. Theo đó, bộ phận điều khiển chủ động truyền hiệu lệnh đến các khớp nối cho phép cánh tay chuyển động quay hoặc chuyển dịch(tuyến tính).

Các liên kết được thao túng theo nhu cầu sản xuất để tạo thành một chuỗi động học. Điểm cuối của chuỗi hành động này được gọi là hiệu ứng kết thúc và quy luật hoạt động của nó mô phỏng hoàn hảo chuyển động của bàn tay con người.

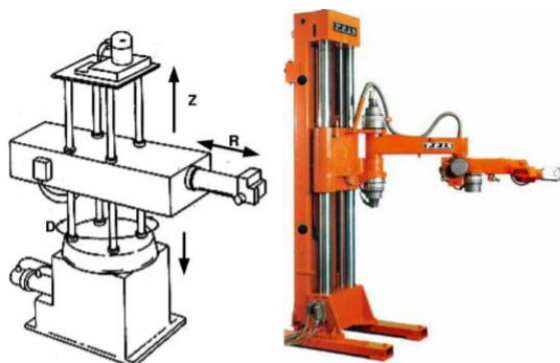
2.1.2. Phân loại

- **Robot trục tọa độ Descartes(còn gọi là robot tuyến tính):** là một robot công nghiệp có ba trục điều khiển chính là tuyến tính (tức là chúng di chuyển theo đường thẳng chứ không phải xoay) và vuông góc với nhau. Ba khớp trượt tương ứng với việc di chuyển cổ tay lên xuống, vào-ra, tới-lùi. Trong số các ưu điểm khác, cách bố trí cơ khí này đơn giản hóa giải pháp cánh tay điều khiển Robot.



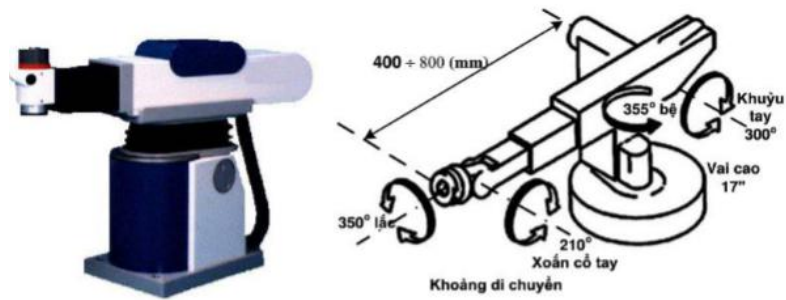
Hình 2.1: Robot Descartes

- **Robot hình trụ :** Được sử dụng cho các hoạt động lắp ráp, xử lý tại các máy công cụ, hàn điểm và xử lý tại các máy diecasting. Đó là một robot có trục tọa độ hình trụ.



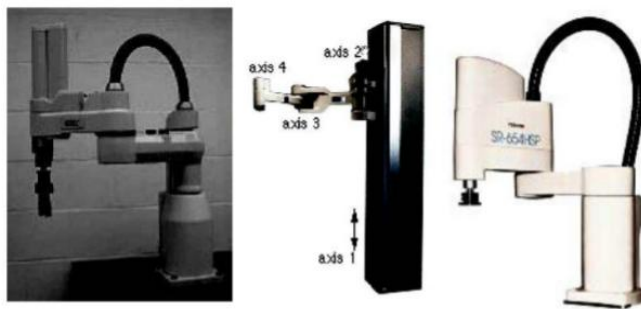
Hình 2.2: Robot hệ tọa độ trụ

- **Robot hình cầu / Robot cực** Được sử dụng để xử lý các công cụ máy móc, hàn điểm, diecasting, máy làm ướt, hàn khí và hàn hồ quang. Đó là một robot có trục tọa độ cực.



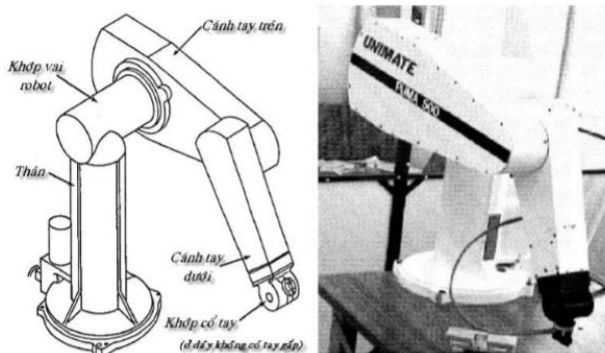
Hình 2.3: Robot hệ tọa cầu

- **Robot SCARA:** Được sử dụng để chọn và đặt công việc, ứng dụng chất trám, hoạt động lắp ráp và xử lý các công cụ máy móc. Robot này có hai khớp quay song song để cung cấp sự tuân thủ trong một mặt phẳng.



Hình 2.4: Robot Scara

- **Robot khớp bản lề :** Được sử dụng cho các hoạt động lắp ráp, diecasting, máy làm ướt, hàn khí, hàn hồ quang và phun sơn. Đó là một robot có cánh tay có ít nhất ba khớp quay.



Hình 2.5: Robot khớp bản lề

2.1.3. Cấu trúc cơ bản tay máy robot

a. Các thành phần cơ bản

- Tay máy (Manipulator) là cơ cấu cơ khí gồm các khâu, khớp. Chúng hình thành cánh tay để tạo các chuyển động cơ bản, cổ tay tạo nên sự khéo léo, linh hoạt và bàn tay để trực tiếp hoàn thành các thao tác trên đối tượng.

- Ứng dụng của Arm robot trong công nghiệp rất đa dạng, tùy vào những ngành nghề, công việc khác nhau mà ta có thể áp dụng những robot công nghiệp riêng biệt. Dưới đây là một số ngành trong sản xuất áp dụng robot công nghiệp.
- Công nghiệp đúc: Arm robot làm nhiệm vụ rót kim loại nóng chảy vào khuôn, cắt mép thừa, làm sạch vật đúc hoặc làm tăng bền vật đúc bằng cách phun cát.
- Ngành gia công và lắp ráp: Arm robot thường được sử dụng vào những việc như tháo lắp phôi và sản phẩm cho các máy ra công bánh răng, máy khoan, máy tiện.



Hình 2 8: Ứng dụng của robot trong sản xuất

2.2. Xây dựng bài toán động học tay máy

2.2.1. Các khái niệm ban đầu

a. Quỹ đạo

- Tọa độ suy rộng: có thể là chuyển vị góc ở các khớp quay hoặc chuyển vị dài ở các khớp tịnh tiến của các khâu thành viên: q_1, q_2, \dots, q_n

$$q = q(t)$$

$$x_M = (q_1, q_2, \dots, q_n) = x_M(t) \quad (2-1)$$

- Quỹ đạo điểm M: $\begin{cases} y_M = (q_1, q_2, \dots, q_n) = y_M(t) \\ z_M = z_M(q_1, q_2, \dots, q_n) = z_M(t) \end{cases}$

b. Bậc tự do của robot

- Bậc tự do (DOF – Degree Of Freedom): là thông số độc lập cần thiết để xác định hoàn toàn vị trí của cơ cấu, nó cũng là số khả năng chuyển động tương đối độc lập của cơ cấu đó.
- Công thức tính bậc tự do:

$$F = \lambda(n-1) - \sum_{i=1}^j c_i \quad \leftrightarrow \quad F = \lambda(n-j-1) + \sum_{i=1}^j f_i$$

- c_i số chuyển động bị ràng buộc khớp i
- f_i số chuyển động tương đối cho phép bởi khớp i

- j : Tổng số khớp trong cơ cấu
- n : Tổng số khâu trong cơ cấu (kể cả khâu cố định)
- λ : Số bậc tự do của không gian mà cơ cấu hoạt động

2.2.2. Đặt vấn đề

Cơ cấu chấp hành của robot thường là một cơ cấu hở, gồm một chuỗi các khâu (links) nối với nhau bằng các khớp (joint). Các khớp động này là khớp quay (R) hoặc khớp tịnh tiến (T). Để robot có thể thao tác linh hoạt, cơ cấu chấp hành của nó phải cấu tạo sao cho điểm mút của khâu cuối cùng đảm bảo dễ dàng di chuyển theo một quỹ đạo nào đó, đồng thời khâu này có một định hướng nhất định. Khâu cuối thường là bàn kẹp (gripper) hoặc là khâu gắn liền với dụng cụ làm việc (tools). Điểm mút của khâu cuối cùng là điểm đáng quan tâm nhất vì đó là điểm tác động của robot lên đối tác và được gọi là “điểm tác động cuối”.

Gắn vào “điểm tác động cuối” này một hệ tọa độ thứ n và gắn mỗi khâu động một hệ tọa độ động khác, còn gắn liền giá đỡ với một hệ tọa độ cố định. Đánh số ký hiệu các hệ này từ 0 đến n bắt đầu. Khi khảo sát chuyển động của robot cần biết “định vị và định hướng” tại “điểm tác động cuối” trong mọi thời điểm.

2.2.3. Xác định trạng thái của robot tại “điểm tác động cuối”

Như trên đã nhấn mạnh tại các “điểm tác động cuối” của robot không những cần biết vị trí của điểm mút của khâu cuối cùng, tức là điểm tác động của robot lên đối tác, mà còn phải xác định hướng tác động của khâu cuối cùng đó.

Như vậy trạng thái của robot tại “điểm tác động cuối” hoàn toàn xác định bằng sự định vị và định hướng tại điểm tác động cuối đó.

Sự biểu thị định vị và định hướng đó bằng ma trận cuối A . Ma trận này được viết lại như sau:

$$A = \begin{bmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Trong đó tọa độ p_x, p_y, p_z là tọa độ của điểm cuối.

a : vector có hướng tiếp cận (approach) với đối tác

s : vector có hướng đường trượt (sliding) đóng mở bàn kẹp

n : vector pháp tuyến (normal)

2.2.4. Mô hình động học

a. Ma trận quan hệ

Chọn hệ tọa độ cố định gắn liền với giá đỡ các hệ tọa độ động gắn với từng khâu động. Ký hiệu các hệ tọa độ này từ 0 đến n, kể từ giá cố định trở đi.

Một điểm bất kỳ nào đó trong không gian được xác định trong hệ tọa độ thứ i bằng bán kính vectơ r_i và trong hệ tọa độ cố định x_0, y_0, z_0 được xác định bằng bán kính vectơ r_0 :

$$\begin{aligned} r_0 &= A_1 \times A_2 \times \dots \times A_i \times r_i \\ \text{hoặc} \quad r_0 &= T_i \times r_i \\ \text{với} \quad T_i &= A_1 \times A_2 \times \dots \times A_i \quad (4.7) \\ i &= 1, \dots, n \end{aligned}$$

Trong đó ma trận A_1 mô tả vị trí hướng của khâu đầu tiên; ma trận A_2 mô tả vị trí và hướng của khâu thứ 2 so vs khâu đầu; ma trận A_i mô tả vị trí và hướng của khâu thứ i so với khâu thứ i-1.

Như vậy, tích của các ma trận A_i là ma trận T_i mô tả vị trí và hướng khâu của khâu thứ i so với giá cố định. Thường ký hiệu ma trận T với 2 chỉ số trên và dưới. Chỉ số dưới để chỉ khâu đang xét còn chỉ số trên để chỉ tọa độ được dùng để đối chiếu. Ví dụ, biểu thức (4.7) có thể viết lại là:

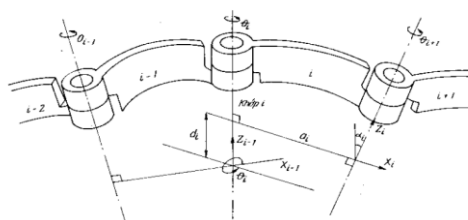
$$\begin{aligned} T_i &= {}^0T_i = A_1 \times T_i \\ \text{với} \quad {}^1T_i &= A_1 \times A_2 \times A_3 \times \dots \times A_i \end{aligned}$$

là ma trận mô tả vị trí và hướng của khâu thứ i so với khâu thứ nhất. Trong ký hiệu thường bỏ qua chỉ số trên nếu chỉ số đó bằng 0.

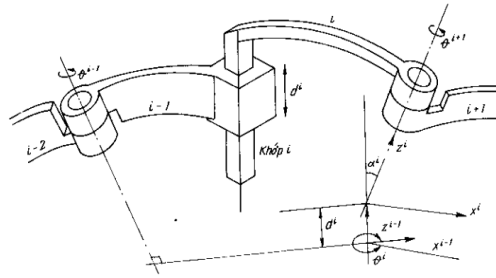
Denavit L.& Hartenbeg R.S. đã đề xuất dùng ma trận thuần nhất 4x4 để mô tả quan hệ giữa 2 khâu liên tiếp trong cơ cấu không gian. Pieper D.L. đã đầu tiên áp dụng ma trận thuần nhất 4x4 trong nghiên cứu robot. Litvin F.L đã dùng ma trận thuần nhất 4x4 trong nghiên cứu lý thuyết ăn khớp với bánh răng.

b. Bộ thông số DH

Dưới đây trình bày cách xây dựng các hệ tọa độ đối với 2 khâu động liên tiếp i và i+1. Hình 4.4 là trường hợp 2 khớp động liên tiếp là khớp quay. Hình 4.5 khớp i là khớp tịnh tiến.



Hình 2.9: Các hệ tọa độ đối với 2 khâu động liên tiếp



Hình 2.10: Các hệ tọa độ đối với 2 khâu động liên tiếp có khớp tịnh tiến

Xác định bộ thông số cơ bản giữa 2 trục quay của khớp động $i+1$ và i :

- a_i là độ dài đường vuông góc chung giữa 2 trục khớp động $i+1$ và i .
- α_i là góc chéo giữa 2 trục khớp động $i+1$ và i .
- d_i là khoảng cách đo dọc trục khớp động i từ đường vuông góc chung giữa trục khớp động i và trục khớp động $i-1$.
- θ_i là góc giữa 2 đường vuông góc chung nói trên.

Bộ thông số này được gọi là bộ thông số Denavit – Hartenberg, hoặc viết tắt là *bộ thông số DH*.

- *Biến khớp* (joint variable):
 - + Nếu khớp động i là khớp quay thì θ_i là biến khớp.
 - + Nếu khớp động i là tịnh tiến thì d_i là biến khớp.

Để ký hiệu biến khớp dùng thêm dấu * và trong trường hợp khớp tịnh tiến thì a_i được xem là bằng 0.

c. Thiết lập hệ tọa độ

Gốc của hệ tọa độ gắn liền với khâu thứ i (gọi là hệ tọa độ thứ i) đặt tại giao điểm giữa đường vuông góc chung (a_i) và trục khớp động $i+1$. Trường hợp hai trục giao nhau thì gốc hệ tọa độ lấy trùng với giao điểm đó. Nếu hai trục song song với nhau thì chọn gốc tọa độ là điểm bất kỳ trên trục khớp động $i+1$.

Trục z_i của hệ tọa độ thứ i nằm dọc theo trục khớp động $i+1$.

Trục x_i của hệ tọa độ thứ i nằm dọc theo đường vuông góc chung hướng từ khớp động i đến khớp $i+1$. Trường hợp hai trục giao nhau, hướng trục x_i trùng với hướng vector tích $z_i * z_{i-1}$, tức là vuông góc với mặt phẳng chứa z_i và z_{i-1} .

d. Mô hình biến đổi

Trên cơ sở đã xây dựng các hệ tọa độ với 2 khâu động liên tiếp như trên, có thể thiết lập mối quan hệ giữa 2 hệ tọa độ liên tiếp theo 4 bước sau:

1. Quay quanh trục z_{i-1} một góc θ_i .
2. Tịnh tiến dọc trục z_{i-1} một quãng d_i .
3. Tịnh tiến dọc trục x_{i-1} (đã trùng với x_i) một đoạn a_i .
4. Quay quanh trục x_i một góc α_i .

Bốn bước biến đổi được biểu hiện này bằng tích các ma trận thuận nhất sau:

$$A_i = R(z, \theta_i) \times T(o, o, d_i) \times T_p(a_i, o, o) \times R(x, \alpha_i)$$

Biểu thức trên là quan hệ giữa hệ tọa độ i so với hệ tọa độ $i-1$ và được gọi là mô hình DH (DH - model). Cách thiết lập mô hình động học theo kiểu mô hình DH tỏ ra thuận tiện trong khi giải quyết các vấn đề cơ học robot, vì thế được dùng khá rộng rãi. Ngoài ra còn tồn tại một vài cách khác để thiết lập mô hình động học như Universal – model, S- model. Ta có:

$$T = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R(x, \alpha_i) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R(z, \theta) = \begin{bmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\text{Qua đó } A_i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \times \cos \alpha_i & \sin \theta_i \times \sin \alpha_i & a_i \times \cos \theta_i \\ \sin \theta_i & \cos \theta_i \times \cos \alpha_i & -\cos \theta_i \times \sin \alpha_i & a_i \times \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2.2.5. Trình tự thiết lập hệ phương trình động học của robot

Để thiết lập phương trình động học của robot, có thể tiến hành các bước sau:

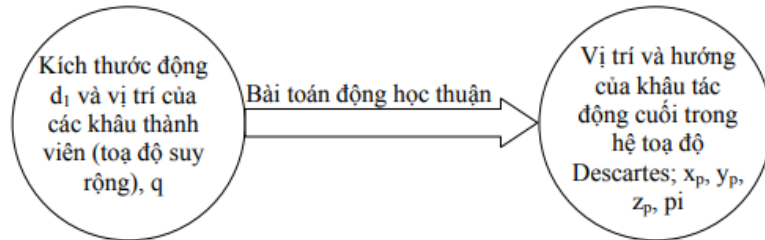
1. Xác định các hệ tọa độ
2. Lập bảng thông số DH.
3. Xác định các ma trận A_i theo các thông số DH.
4. Lập phương trình động học cơ bản $A_i = {}^0A_1 \times {}^1A_2 \times {}^2A_3 \times \dots \times {}^{i-1}A_i$

2.2.6. Động học thuận tay máy

a. Tổng quan về bài toán động học tay máy

Cho trước cơ cấu và quy luật của các yếu tố chuyển động thể hiện bằng các tọa độ suy rộng ta phải xác định quy luật chuyển động của điểm trên khâu tác

động cuối nói riêng hoặc của điểm bất kỳ trên một khâu nào đó của tay máy nói chung trong hệ trục tọa độ vuông góc (hệ trục tọa độ Descartes). Hay nói cách khác là xác định vị trí trong hệ tọa độ Descartes(X, Y, Z) khi biết được vị trí của các góc khớp.

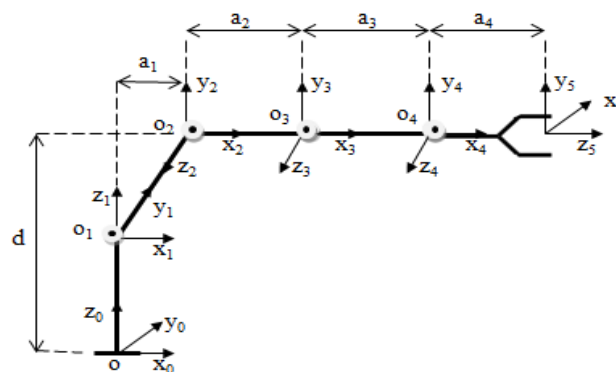


Hình 2. 11: Sơ đồ mô tả tổng quan bài toán động học thuận tay máy.

b. Giải bài toán động học thuận sử dụng phương pháp Denavit – Hartenberg

- Đặt hệ trục tọa độ: Theo Denavit – Hartenberg (1955) đã quy ước hệ tọa độ Decard gắn vào mỗi khâu của một tay máy robot như sau:
 - Chọn trục Z_i : trục Z_i nằm dọc theo trục khớp thứ $i+1$. Hướng của phép quay và phép tịnh tiến được chọn tùy ý.
 - Chọn trục X_i : được xác định dọc theo đường vuông góc chung giữa trục khớp động thứ i và $(i+1)$, hướng từ khớp động thứ i tới trục $(i+1)$.
 - Trục Y_i được xác định theo qui tắc bàn tay phải.

Từ quy tắc trên ta xây dựng các tọa độ khảo sát: $O_0X_0Y_0Z_0$, $O_1X_1Y_1Z_1$, $O_2X_2Y_2Z_2$, $O_3X_3Y_3Z_3$, $O_4X_4Y_4Z_4$.



Hình 2.12: Hệ tọa độ của cánh tay robot 4 bậc tự do

- Xác định các thông số động học của bảng Denavit – Hartenberg (DH)

Vị trí của hệ tọa độ khớp $(Oxyz)_i$ đối với hệ tọa độ khớp $(Oxyz)_{i-1}$ được xác định bởi 4 tham số $\theta_i, d_i, \alpha_i, a_i$ như sau:

- a_i : khoảng cách giữa hai trục z
- α_i : z_{i-1} xoay quanh x_i một góc α_i để sao cho z_{i-1} trùng với z_i .

- d_i : khoảng cách giữa hai trục x
 - θ_i : góc xoay quanh trục z
- Thiết lập bộ thông số động học DH
- Với cách thành lập hệ tọa độ như trên ta có thể xác định các tham số động học của robot như sau:

Khâu	a_i	α_i	d_i	θ_i
1	a_1	90°	d	θ_1
2	a_2	0°	0	θ_2
3	a_3	0°	0	θ_3
4	a_4	0°	0	θ_4

Bảng 1: Tham số động học của Robot

- Dạng tổng quát của ma trận Denavit – Hartenberg cho các khâu

$${}^{i-1}A_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \cdot \cos\alpha_i & \sin\theta_i \cdot \sin\alpha_i & a_i \cdot \cos\theta_i \\ \sin\theta_i & \cos\theta_i \cdot \cos\alpha_i & -\cos\theta_i \cdot \sin\alpha_i & a_i \cdot \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Ma trận DH cho từng khâu

- Ma trận Denavit- Hartenberg của khâu 1:

$${}^0A_1 = \begin{bmatrix} \cos\theta_1 & 0 & \sin\theta_1 & a_1 \cdot \cos\theta_1 \\ \sin\theta_1 & 0 & -\cos\theta_1 & a_1 \cdot \sin\theta_1 \\ 0 & 1 & 0 & d \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Ma trận Denavit- Hartenberg của khâu 2:

$${}^1A_2 = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & a_2 \cdot \cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & a_2 \cdot \sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Ma trận Denavit- Hartenberg của khâu 3:

$${}^2A_3 = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 & 0 & a_3 \cdot \cos\theta_3 \\ \sin\theta_3 & \cos\theta_3 & 0 & a_3 \cdot \sin\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Ma trận Denavit- Hartenberg của khâu 4:

$${}^3A_4 = \begin{bmatrix} \cos\theta_4 & 0 & -\sin\theta_4 & a_4 \cdot \cos\theta_4 \\ \sin\theta_4 & 0 & \cos\theta_4 & a_4 \cdot \sin\theta_4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Từ đó ta có thể tính được ma trận chuyển đổi từ tọa độ ban đầu tới tọa độ điểm cuối là:

$${}^0A_4 = {}^0A_1 * {}^1A_2 * {}^2A_3 * {}^3A_4$$

$${}^0A_4 = \begin{matrix} & \text{position} & \text{orientation} \\ \begin{bmatrix} nx & mx & ax & px \\ ny & my & ay & py \\ nz & mz & az & pz \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{matrix}$$

Ma trận chuyển đổi 0A_4 giúp ta có thể tính được tọa độ các điểm px, py, pz. Từ phần mềm Matlab ta có thể tính được giá trị px, py, pz cụ thể như sau:

$$px = \cos(\theta_1) * (a_3 * \cos(\theta_2 + \theta_3) + a_2 * \cos(\theta_2) + a_4 * \cos(\theta_2 + \theta_3 + \theta_4))$$

$$py = \sin(\theta_1) * (a_3 * \cos(\theta_2 + \theta_3) + a_2 * \cos(\theta_2) + a_4 * \cos(\theta_2 + \theta_3 + \theta_4))$$

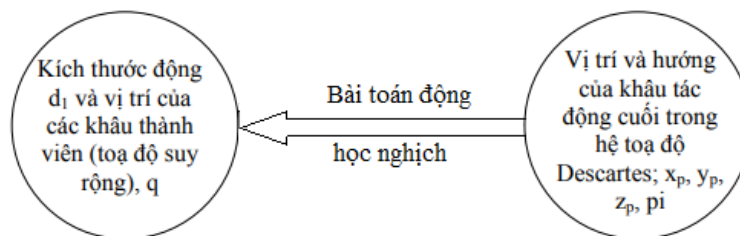
$$pz = a_1 + a_3 * \sin(\theta_2 + \theta_3) + a_2 * \sin(\theta_2) + a_4 * \sin(\theta_2 + \theta_3 + \theta_4)$$

Có quỹ đạo điểm P(P_x, P_y, P_z) là tọa độ điểm cuối của tay gấp tương ứng với tọa độ $Ox_5y_5z_5$

2.2.7. Bài toán động học nghịch (động học ngược) tay máy.

a. Tổng quan về bài toán động học ngược tay máy

Cho trước cơ cấu và quy luật chuyển động của điểm trong khâu tác động cuối (hoặc quy luật chuyển động của khâu cuối bao gồm vị trí và hướng của nó) được biểu diễn trong hệ tọa độ vuông góc, ta phải xác định quy luật chuyển động của các khâu thành viên thể hiện qua các hệ tọa độ suy rộng. Hay nói một cách đơn giản là cho tọa độ điểm P(P_x, P_y, P_z) bất kì trong không gian hoạt động của tay máy, ta phải tính toán chiều, và góc quay của từng khớp sao cho khâu cuối đạt đến tọa độ của điểm P.



Hình 2.13: Sơ đồ mô tả tổng quan bài toán động học ngược tay máy

b. Giải bài toán động học nghịch

- Bài toán động học vị trí nghịch là xác định giá trị các biến khớp (θ_i) khi biết được vị trí và hướng của dụng cụ (so với hệ tọa độ gốc)

$$A_4 = {}^0A_1 \times {}^1A_2 \times {}^2A_3 \times {}^3A_4$$

$$\Leftrightarrow {}^0A_1^{-1} \times A_4 = {}^0A_1^{-1} \times {}^0A_1 \times {}^1A_2 \times {}^2A_3 \times {}^3A_4$$

$$\Leftrightarrow \begin{bmatrix} \dots & \dots & \dots & n_1 \\ \dots & \dots & \dots & n_2 \\ \dots & \dots & \dots & n_3 \\ \dots & \dots & \dots & 1 \end{bmatrix} = \begin{bmatrix} \dots & \dots & \dots & m_1 \\ \dots & \dots & \dots & m_2 \\ \dots & \dots & \dots & m_3 \\ \dots & \dots & \dots & 1 \end{bmatrix} \Leftrightarrow \begin{cases} n_1 = m_1 \\ n_2 = m_2 \\ n_3 = m_3 \end{cases}$$

$$\text{Ký hiệu: } S_1 = \sin\theta_1, S_2 = \sin\theta_2, S_3 = \sin\theta_3, S_4 = \sin\theta_4$$

$$C_1 = \cos\theta_1, C_2 = \cos\theta_2, C_3 = \cos\theta_3, C_4 = \cos\theta_4$$

$$S_{23} = \sin(\theta_2 + \theta_3), C_{23} = \cos(\theta_2 + \theta_3)$$

$$S_{234} = \sin(\theta_2 + \theta_3 + \theta_4), C_{234} = \cos(\theta_2 + \theta_3 + \theta_4)$$

$$\text{Có: } \begin{cases} n_1 = m_1 \\ n_2 = m_2 \\ n_3 = m_3 \end{cases}$$

$$n_1 = p_x \times C_1 - L_1 + p_y \times S_1$$

$$n_2 = d_1 - p_z$$

$$n_3 = p_y \times C_1 - p_x \times S_1$$

$$m_1 = L_3 \times C_{23} + L_2 \times C_2 + L_4 \times C_{234}$$

$$m_2 = L_3 \times S_{23} + L_2 \times S_2 + L_4 \times S_{234}$$

$$m_3 = 0$$

$$*n_3 = m_3$$

$$\Leftrightarrow p_x \times S_1 = p_y \times C_1 \Leftrightarrow \frac{S_1}{C_1} = \frac{p_y}{p_x} \Leftrightarrow \tan\theta_1 = \frac{S_1}{C_1} \Rightarrow \theta_1 = \arctan2(p_y, p_x)$$

Lại có:

$$\begin{cases} n_1 = m_1 \\ n_2 = m_2 \end{cases} \Leftrightarrow \begin{cases} p_x \times C_1 + p_y \times S_1 - L_1 = L_3 \times C_{23} + L_2 \times C_2 + L_4 \times C_{234} \\ d_1 - z = L_3 \times S_{23} + L_2 \times S_2 + L_4 \times S_{234} \end{cases}$$

$$\Leftrightarrow \begin{cases} n_1 = L_3 \times C_{23} + L_2 \times C_2 + L_4 \times C_{234} \\ n_2 = L_3 \times S_{23} + L_2 \times S_2 + L_4 \times S_{234} \end{cases}$$

$$\Leftrightarrow \begin{cases} n_1 - L_4 \times C_{234} = L_3 \times C_{23} + L_2 \times C_2 \\ n_2 - L_4 \times S_{234} = L_3 \times S_{23} + L_2 \times S_2 \end{cases}$$

$$\text{Đặt } n_1 - L_4 \times C_{234} = n_x, n_2 - L_4 \times S_{234} = n_y$$

$$\Rightarrow n_x^2 + n_y^2 = L_2^2 + 2\cos\theta_3 \times L_2 \times L_3 + L_3^2 \Leftrightarrow \cos\theta_3 = \frac{n_x^2 + n_y^2 - L_2^2 - L_3^2}{2 \times L_2 \times L_3} \quad (1)$$

Tính n_x, n_y :

$$\begin{cases} n_x = n_1 - L_4 \times C_{234} \\ n_y = n_2 - L_4 \times S_{234} \end{cases}$$

$$\text{mà } \theta_2 + \theta_3 + \theta_4 = 0$$

$$\Rightarrow \sin(\theta_2 + \theta_3 + \theta_4) = S_{234} = 0$$

$$\cos(\theta_2 + \theta_3 + \theta_4) = C_{234} = 1$$

$$\Rightarrow \cos\theta_3 = \frac{(n_1-L_4)^2+n_2^2-L_2^2-L_3^2}{2 \times L_2 \times L_3} = \frac{(p_x \times C_1 + p_y \times S_1 - L_1 - L_4)^2 + (p_z - d_1)^2 - L_2^2 - L_3^2}{2 \times L_2 \times L_3}$$

$$\sin\theta_3 = \sqrt{1 - \cos^2\theta_3} \Leftrightarrow \tan\theta_3 = \frac{\sin\theta_3}{\cos\theta_3} \Leftrightarrow \theta_3 = \arctan2(\sin\theta_3, \cos\theta_3)$$

Tính θ_3 :

$$\begin{cases} n_x = L_3 \times C_{23} + L_2 \times C_2 \\ n_y = L_3 \times S_{23} + L_2 \times S_2 \end{cases}$$

$$\Leftrightarrow \begin{cases} n_x = L_3 \times (C_2 \times C_3 - S_2 \times S_3) + L_2 \times C_2 \\ n_y = L_3 \times (S_3 \times C_2 + S_2 \times C_3) + L_2 \times S_2 \end{cases}$$

$$\Leftrightarrow \begin{cases} n_x = C_2 \times (L_3 \times C_3 + L_2) - L_3 \times S_2 \times S_3 \\ n_y = S_2 \times (L_3 \times C_3 + L_2) + L_3 \times C_2 \times S_3 \end{cases}$$

$$\Leftrightarrow \begin{cases} C_2 = n_x + L_3 \times S_3 \times S_2 & (2) \\ S_2 = \frac{n_y - L_3 \times C_2 \times S_3}{L_3 \times C_3 + L_2} & (3) \end{cases} (*)$$

Thay (3) vào (2):

$$C_2 = \frac{n_x + L_3 \times S_3 \times \frac{n_y - L_3 \times C_2 \times S_3}{L_3 \times C_3 + L_2}}{L_3 \times C_3 + L_2}$$

$$\Leftrightarrow C_2 = \frac{n_x \times (L_3 \times C_3 + L_2) + L_3 \times S_3 \times (n_y - L_3 \times C_2 \times S_3)}{(L_3 \times C_3 + L_2)^2}$$

$$\Leftrightarrow C_2 = \frac{n_x \times (L_3 \times C_3 + L_2) + L_3 \times S_3 \times n_y - L_3^2 \times S_3^2 \times S_3}{(L_3 \times C_3 + L_2)^2}$$

$$\Leftrightarrow C_2 = \frac{n_x \times (L_3 \times C_3 + L_2) + L_3 \times S_3 \times n_y}{(L_3 \times C_3 + L_2)^2} - \frac{L_3^2 \times S_3^2 \times S_3}{(L_3 \times C_3 + L_2)^2}$$

$$\Leftrightarrow C_2 \times \left[1 + \frac{L_3^2 \times S_3^2}{(L_3 \times C_3 + L_2)^2} \right] = \frac{n_x \times (L_3 \times C_3 + L_2) + L_3 \times S_3 \times n_y}{(L_3 \times C_3 + L_2)^2}$$

$$\Leftrightarrow C_2 \times \left[\frac{(L_3 \times C_3 + L_2)^2 + L_3^2 \times S_3^2}{(L_3 \times C_3 + L_2)^2} \right] = \frac{n_x \times (L_3 \times C_3 + L_2) + L_3 \times C_3 \times n_y}{(L_3 \times C_3 + L_2)^2}$$

$$\Leftrightarrow C_2 = \frac{n_x \times (L_3 \times C_3 + L_2) + L_3 \times S_3 \times n_y}{(L_3 \times C_3 + L_2)^2 + L_3^2 \times S_3^2}$$

Trong đó $n_x = p_x \times C_1 + p_y \times S_1 - L_1 - L_4$

$$n_y = p_z - d_1$$

Thay C_2 vào (*):

$$S_2 = \frac{n_y \times (L_3 \times C_3 + L_2) - L_3 \times S_3 \times n_x}{(L_3 \times C_3 + L_2)^2 + L_3^2 \times S_3^2}$$

$$\tan\theta_2 = \frac{S_2}{C_2}$$

$$\Rightarrow \theta_2 = \arctan2(S_2, C_2)$$

$$\text{Lại có: } \theta_2 + \theta_3 + \theta_4 = 0$$

$$\Rightarrow \theta_4 = -\theta_2 - \theta_3$$

Kết luận: Từ tọa độ điểm P(x,y,z) tính toán được giá trị các góc $\theta_1, \theta_2, \theta_3, \theta_4$.

2.3. Thiết kế cánh tay robot 4 bậc tự do

2.3.1. Giới thiệu phần mềm Autodesk Inventor

a. Inventor là gì?

Autodesk Inventor là phần mềm xây dựng mô hình 3D, thiết kế, hình mẫu và kiểm tra ý tưởng các sản phẩm. Inventor tạo ra các nguyên mẫu mô phỏng chuẩn xác khối lượng, áp lực, độ ma sát, tải trọng,... của các đối tượng sản phẩm trong môi trường 3D. Các công cụ mô phỏng, phân tích được tích hợp trong Inventor cho phép người dùng thiết kế từ khuôn đúc cơ bản đến nâng cao như thiết kế chi tiết máy, trực quan hóa sản phẩm. Inventor còn được tích hợp CAD và các công cụ giao tiếp thiết kế nhằm nâng cao năng suất làm việc của CAD và giảm thiểu phát sinh lỗi, tiết kiệm thời gian.



Hình 2.14: Phần mềm Autodesk Inventor Professional 2018

b. Lợi ích của Inventor

Những người đang hoặc sẽ sử dụng AutoCAD sẽ thừa hưởng được nhiều lợi ích của Inventor. Inventor cũng cấp một môi trường thiết kế và phím tắt tương tự với AutoCAD, hỗ trợ tập tin DWG, cho phép người dùng chuyển từ vẽ 2D hiện hành sang xây dựng mô hình 3D. Inventor được sử dụng phổ biến trong tạo nguyên mẫu kỹ thuật số, các mẫu được tạo ra từ bản vẽ 2D AutoCAD được tích hợp và các dữ liệu 3D, hình thành nên sản phẩm ảo. Bằng cách này, các kỹ sư có thể thiết kế, mô phỏng sản phẩm mà không phải tạo ra các mẫu vật lý.

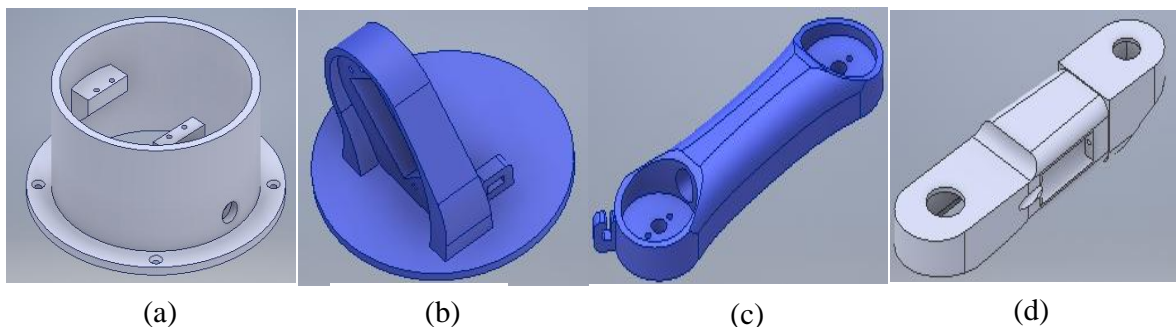
Người dùng có thể sử dụng các công cụ thiết kế 3D cơ khí trong Inventor để nghiên cứu và đánh giá mô hình thuận tiện và hiệu quả hơn. AutoDesk Inventor còn đưa ra các công cụ và tính năng khác nhằm nâng cao năng suất làm việc như: Integrated Data Management, Design Automation, Automatic Drawing Updates and Views, Automatic Bill of Materials,...

c. Ứng dụng của Inventor

Autodesk Inventor được sử dụng để rút ngắn khoảng cách giữa thiết kế, kỹ thuật và sản xuất. Ví dụ: ngành sản xuất xe đạp leo núi sử dụng Inventor tạo ra các mẫu kỹ thuật số và thành phẩm để tối ưu các tác động giữa các bộ phận lắp ráp, đảm bảo chính xác dung sai và các khoảng hở. Trong sản xuất thuyền buồm, Inventor được sử dụng để xây dựng mô hình và tạo mẫu đột phá, chạy các thử nghiệm áp lực nhằm xác định các bộ phận cần cắt giảm khối lượng và cải thiện hiệu suất. Một ứng dụng khác của Inventor là trong công nghiệp khai khoáng, nhằm thực hiện các phân tích ứng suất, mô phỏng chuyển động máy tìm ra các va chạm ngoài ý muốn và các lỗi phát sinh. Inventor giúp cắt giảm chi phí sản xuất bằng việc tạo ra các nguyên mẫu và thử nghiệm ảo, hạn chế lỗi và lao động thủ công, tăng chu kỳ sản xuất và đưa sản phẩm đến thị trường nhanh hơn.

2.3.2. Thiết kế cánh tay robot 4 bậc tự do

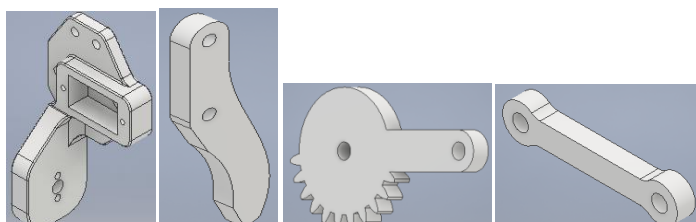
a. Phần thân cánh tay robot 4 bậc tự do



Hình 2.15: Các khâu của robot

(a) Đế; (b) Eo; (c) Khâu 1 ;(d) Khâu 2

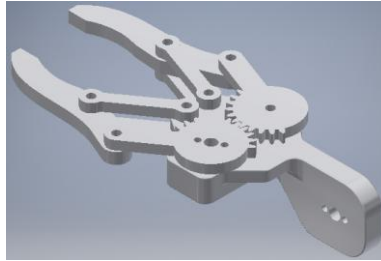
b. Phần tay gấp



Hình 2.16: Các khâu của gripper

(a) Đế tay gấp; (b) Tay gấp; (c) Bánh răng; (d) Khâu gấp

Để lắp ghép thành 1 tay gấp hoàn chỉnh cần 01 Gripper base, 02 Gripper, 02 Gear, 04 Gripper link. Động cơ servo sẽ được gắn ở Gripper base sau đó chuyển động tới các gear, thông qua các grip link tới Gripper.



Hình 2.17: Hình ảnh gripper hoàn chỉnh

2.4. Mô hình hóa và mô phỏng.

2.4.1. Giới thiệu về Simulink và Simscape Multibody

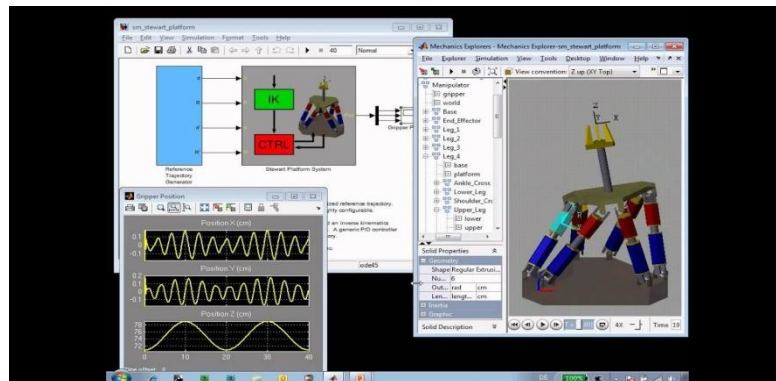
- Simulink là phần chương trình mở rộng của Matlab nhằm mục đích mô hình hóa, mô phỏng và khảo sát các hệ thống động học. Giao diện đồ họa trên màn hình của Simulink cho phép thể hiện hệ thống dưới dạng sơ đồ tín hiệu với các khối chức năng quen thuộc. Simulink cung cấp cho người sử dụng một thư viện rất phong phú, có sẵn với số lượng lớn các khối chức năng cho các hệ tuyến tính, phi tuyến, gián đoạn. Hơn thế, người sử dụng cũng có thể tạo nên các khối riêng mình.



Hình 2.18: Các khâu của robot

- Simscape Multibody TM (trước đây là SimMechanics TM) cung cấp môi trường mô phỏng đa khung cho các hệ thống cơ khí 3D, chẳng hạn như robot, hệ thống treo xe, thiết bị xây dựng và thiết bị hạ cánh của máy bay. Người dùng có thể mô hình hóa các hệ thống đa khung bằng cách sử dụng các khối đại diện cho các khớp, các ràng buộc, các yếu tố lực và cảm biến. Simscape Multibody xây dựng và giải các phương trình chuyển động cho hệ thống cơ hoàn chỉnh. Người dùng có thể tạo hệ

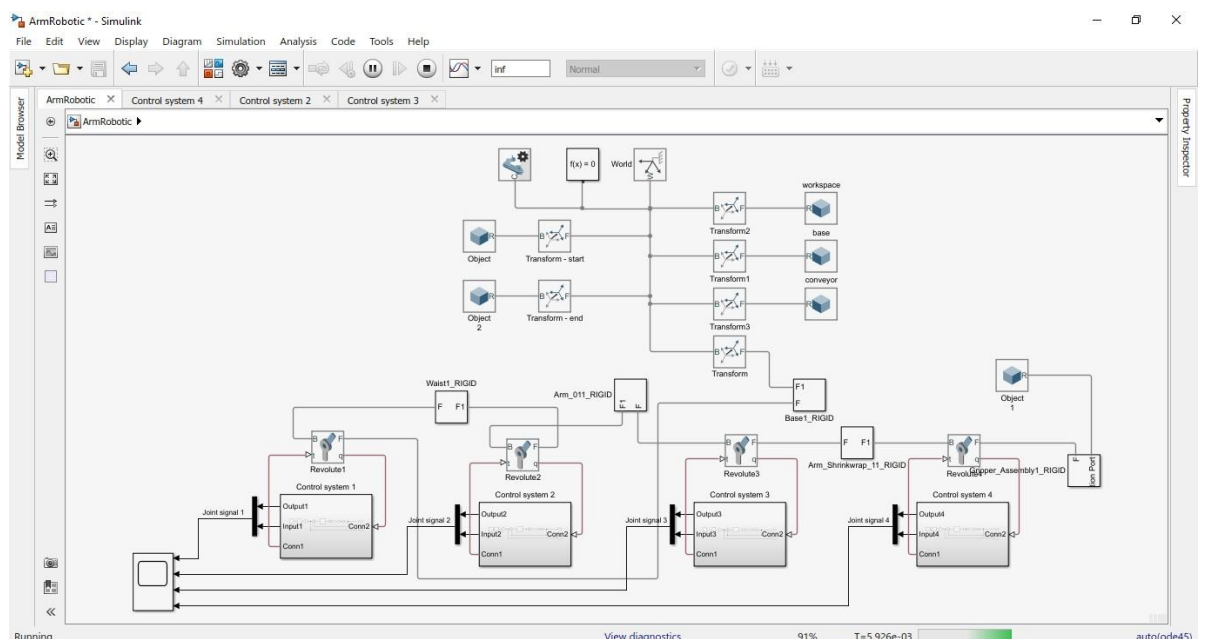
CAD hoàn chỉnh, bao gồm tất cả khối lượng, quán tính, khớp, ràng buộc và hình học 3D, vào mô hình của bạn. Hoạt hình 3D được tạo tự động cho phép người hình dung động lực học của hệ thống.



Hình 2.19: Ví dụ ứng dụng của SimMechanics

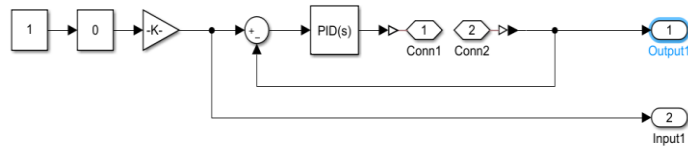
- Simscape Multibody giúp người phát triển các hệ thống điều khiển và kiểm tra hiệu năng cấp hệ thống. Người dùng có thể tham số hóa các mô hình của mình bằng các biến và biểu thức MATLAB[®] và hệ thống kiểm soát thiết kế cho hệ thống đa khung của người dùng trong Simulink[®]. Người dùng có thể tích hợp hệ thống thủy lực, điện, khí nén và các hệ thống vật lý khác vào mô hình của mình bằng các bộ phận từ dòng sản phẩm Simscape. Để triển khai các mô hình của người dùng đến các môi trường mô phỏng khác, bao gồm các hệ thống vòng lặp phản cứng (HIL), bên cạnh đó Simscape Multibody còn hỗ trợ lập trình C – ngôn ngữ lập trình cơ bản, để người dùng dễ thao tác.

2.4.2. Mô phỏng



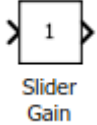
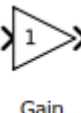
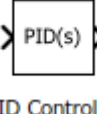



Hình 2.20: Sơ đồ mô phỏng trong Simulink

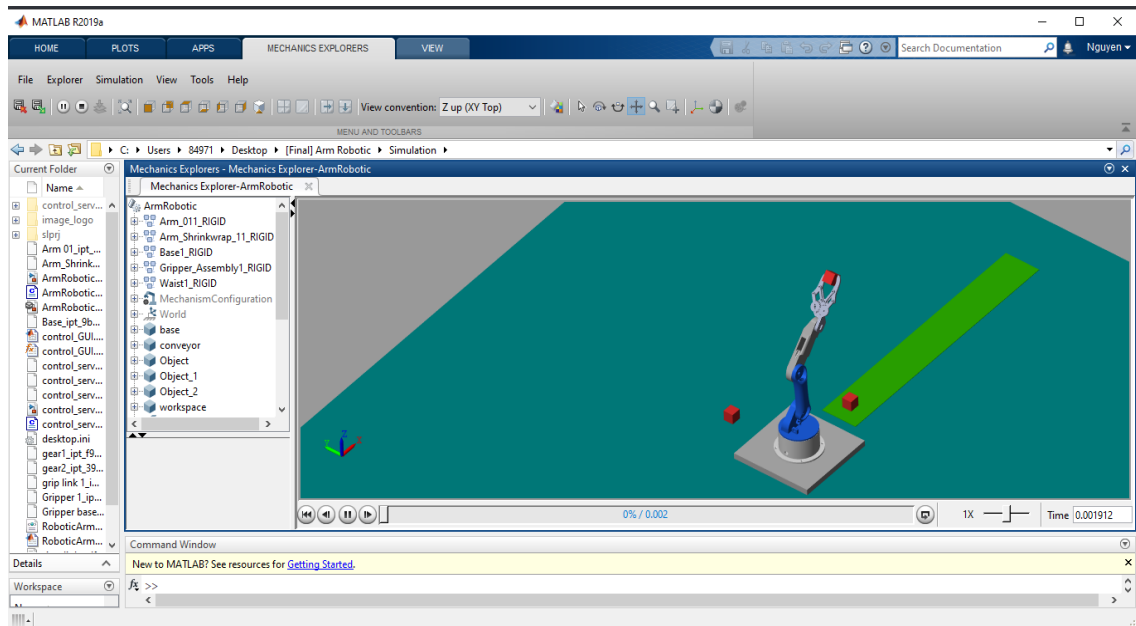
4 khối “Control system” được tạo ra để điều khiển liên tục 4 khớp xoay của robot sử dụng thuật toán điều khiển PID. Sơ đồ khối cấu thành nên các khối “Control system” được thiết lập từ các khối:



Hình 2.21: Cấu tạo khối “Control system”

STT	Khối	Chức năng
1	 Revolute Joint	<ul style="list-style-type: none"> - Khối khớp xoay được tạo ra trong quá trình chuyển đổi sang Matlab Simulink - Khối này tích hợp sẵn các cảm biến để trả về các giá trị như tọa độ, vận tốc, gia tốc,...
2	 Constant	Khối hằng số đầu vào
3	 Slider Gain	Khối thay đổi hệ số khuếch đại vô hướng
4	 Gain	Khuếch đại tín hiệu đầu vào
5	 PID Controller	Bộ điều khiển PID
6	 Scope	Hiển thị tín hiệu của quá trình mô phỏng

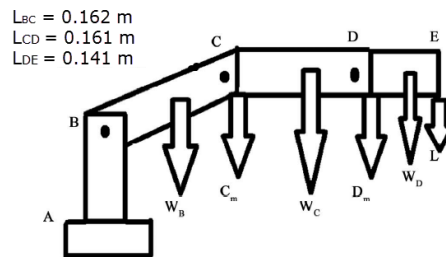
Bảng 2: Các khối simulink sử dụng mô phỏng



Hình 2.22: Kết quả mô phỏng được hiển thị trong Mechanics Explorers

2.5. Xây dựng mô hình thực tế cánh tay robot 4 bậc tự do

2.5.1. Tính toán lựa chọn động cơ servo phù hợp cho các khớp quay



Hình 2.23: Mô hình thể hiện các lực

Để chuẩn bị loại servo thích hợp, ta cần tính toán các thông số mômen dựa từ mô hình lực phía trên. Các thông số đã biết :

$$W_C = 0.55 \text{ N (Trọng lượng của khớp BC)}$$

$$W_B = 0.53 \text{ N (Trọng lượng của khớp CD)}$$

$$W_D = 0.61 \text{ N (Trọng lượng của khớp DE)}$$

$$L = 0.5 \text{ N (Tải)}$$

$$C_m = 0.55 \text{ N (Trọng lượng của servo tại C)}$$

$$D_m = 0.14 \text{ N (Trọng lượng của servo tại D)}$$

Ta có

$$\begin{aligned} \sum M_C &= -W_C \times \frac{L_{CD}}{2} - W_D \times \left(L_{CD} + \frac{L_{DE}}{2} \right) - L \times (L_{CD} + L_{DE}) - D_m \times L_{CD} \\ &+ M_C = 0 \end{aligned}$$

$$\begin{aligned}
\sum M_B &= -W_C \times \left(\frac{L_{CD}}{2} + L_{BC} \right) - W_D \times \left(L_{BC} + L_{CD} + \frac{L_{DE}}{2} \right) - L \\
&\quad \times (L_{CD} + L_{DE} + L_{BC}) - D_m \times (L_{CD} + L_{BC}) - C_m \times L_{BC} - W_B \times \frac{L_{BC}}{2} \\
&\quad + M_B = 0 \\
\sum M_D &= -W_D \times \left(\frac{L_{DE}}{2} \right) - L \times (L_{DE}) + M_D = 0
\end{aligned}$$

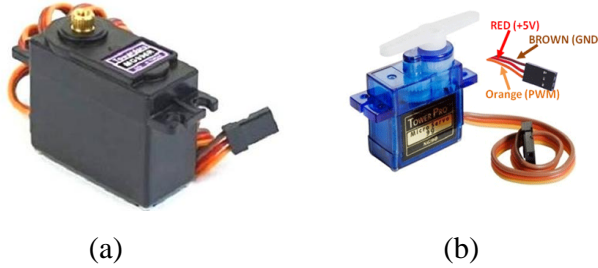
Vậy ta có

$$M_C = 0.359 \text{ Nm} = 3.66 \text{ kg.cm}$$

$$M_B = 0.783 \text{ Nm} = 8 \text{ kg.cm}$$

$$M_D = 0.1153 \text{ Nm} = 1.18 \text{ kg.cm}$$

Dựa trên thông số tính toán được, nhóm đã lựa chọn các loại servo sau là servo MG996R và SG90. Với servo MG996R có momen giữ là 9.4 kg.cm (tại 4V) phù hợp với yêu cầu của khớp B và C. Còn servo SG90 có momen giữ là 2.5 kg.cm (tại 4V) phù hợp với khớp D để quay tay gấp. Cả hai loại servo đều là hai loại phổ biến nên có dễ dàng trong việc sử dụng.



Hình 2.24: Động cơ servo sử dụng trong mô hình

(a) Động cơ servo MG996R; (b) Động cơ servo SG90

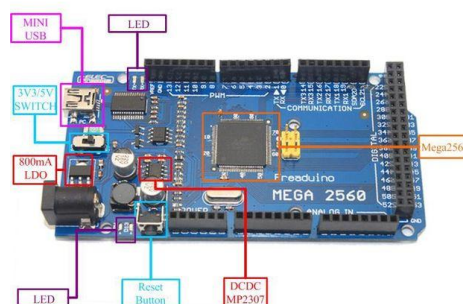
2.5.2. Phần cứng sử dụng

a. Arduino Mega 2560



Hình 2.25: Arduino Mega 2560

Arduino Mega 2560 là phiên bản hiện đang được sử dụng rộng rãi. Arduino Mega 2560 sử dụng chip Atmega2560 có bộ nhớ flash memory 256 KB, 8KB cho bộ nhớ SRAM, 4 KB cho bộ nhớ EEPROM. Giúp cho người dùng thêm khả năng viết những chương trình phức tạp và điều khiển các thiết bị lớn hơn như robot, ...



Hình 2.26: Cấu tạo cơ bản của Arduino Mega2560

Vi điều khiển	Atmega2560
Điện áp hoạt động	5V
Điện áp đầu vào (được đề nghị)	7 – 12V
Điện áp đầu vào (giới hạn)	6 – 20V
Số lượng chân I / O	54 (14 chân có thể cấp xung PWM)
Số lượng chân Input Analog	16
Dòng điện DC mỗi I / O	40 mA
Dòng điện DC với chân 3.3V	50mA
Bộ nhớ flash	256 KB
SRAM	8 KB
EEPROM	4 KB
Tốc độ đồng hồ	16 MHz
Chiều dài	101,52 mm
Bề rộng	53,3 mm
Cân nặng	37 g

Bảng 3: Thông số kỹ thuật Arduino Mega 2560

b. Động cơ servo MG996R

Động cơ Servo MG966R là động cơ có mô men xoắn lớn, chạy mượt mà, phù hợp với những mô hình điều khiển có trọng tải lớn. Động cơ Servo MG966R được nâng cấp từ MG995 6V/11kg, có tốc độ và độ chính xác cao.



Hình 2.27: Động cơ MG996R

Khối lượng	9 gram
Kích thước	40.7 x 19.7 x 42.9 mm
Moomen xoắn	9.4 kg/cm (4.8 V), 11 kg/cm (6 V)
Tốc độ vận hành	0.17 s/60° (4.8 V), 0.14 s/60° (6 V)
Điện áp làm việc	4.8 V – 7.2 V
Nhiệt độ làm việc	0 °C -55 °C
Góc quay	0 – 180°

Bảng 4: Thông số kỹ thuật của động cơ servo MG996R

c. Động cơ servo SG90



Hình 2.28: Động cơ Servo SG90

Động cơ RC Servo 9G là động phổ biến dùng trong các mô hình điều khiển nhỏ và đơn giản như cánh tay robot. Động cơ có tốc độ phản ứng nhanh, được tích hợp sẵn Driver điều khiển động cơ, dễ dàng điều khiển góc quay bằng phương pháp điều độ rộng xung PWM.

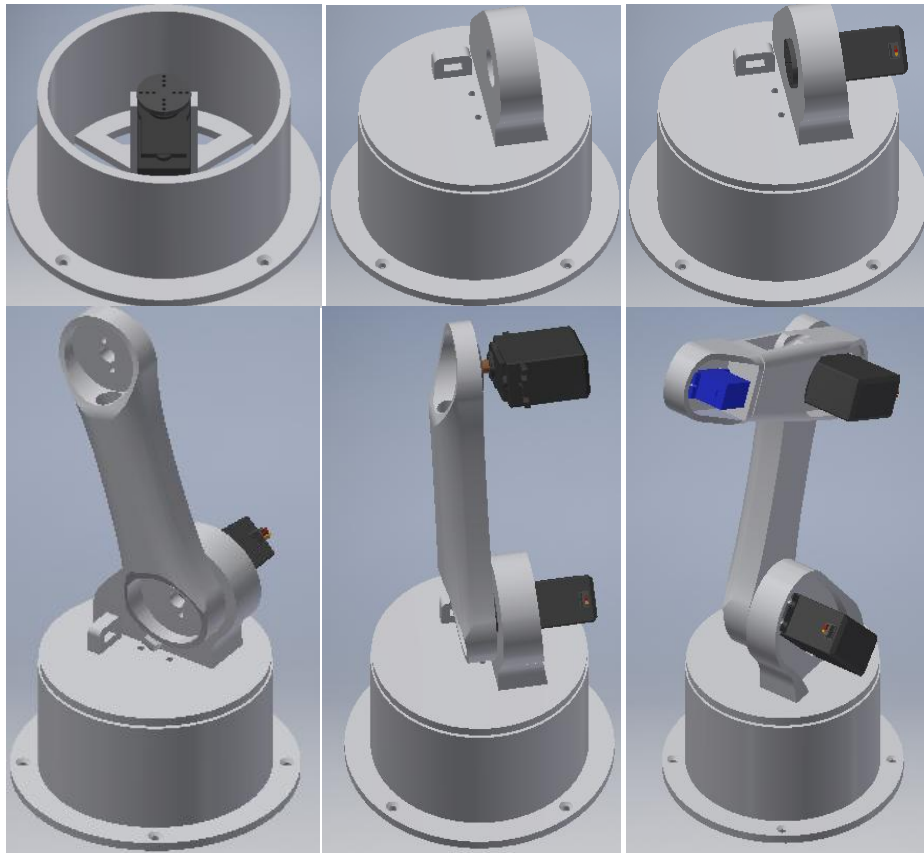
Khối lượng	9 gram
Kích thước	23 x 12.2 x 29 mm
Moomen xoắn	2.5kg/cm
Tốc độ	0.3s/60°
Điện áp làm việc	4.2 V – 6 V
Nhiệt độ làm việc	0 °C - 55 °C
Góc quay	0 – 180°

Bảng 5: Thông số kỹ thuật của động cơ servo SG90.

d. Nguồn cấp

Nguồn cấp cho Arduino: Vì điện áp hoạt động của Arduino là 5V, mặt khác Arduino Mega 2560 sẽ được kết nối và điều khiển Servo thông qua phần mềm Matlab trên PC/laptop nên nhóm sử dụng nguồn cấp từ PC/Laptop có điện áp ổn định khoảng 5V. Đồng thời nguồn cấp cho các động cơ servo cũng sẽ là nguồn DC 5V từ lap top.

2.5.3. Lắp ráp mô hình



Hình 2.29: Lắp ráp mô hình



Hình 2.30: Mô hình thực nghiệm cánh tay robot

2.5.4. Giới hạn chuyển động của khớp và không gian hoạt động

a. Giới hạn chuyển động của các khớp

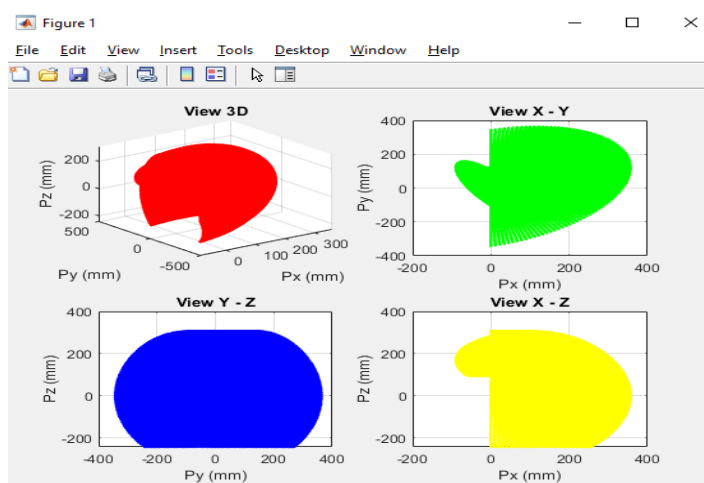
Dựa vào phần mềm thiết kế cơ khí Autodesk Inventor và giới hạn quay của từng khớp trong mô hình thực tế khi có tải. Từ đó có thể thống nhất các giới hạn chuyển động của cả mô phỏng và mô hình thực tế để điều khiển thống nhất, song song đồng thời cả mô phỏng trên máy tính và mô hình. Giới hạn từng khớp của Arm Robot được thể hiện ở bảng dưới đây:

Khớp	Giới hạn trong mô phỏng (độ)	Giới hạn trong mô hình thực tế (độ)	Giới hạn thống nhất (độ)
1	$-180 \leq \theta_1 \leq +180$	$-90 \leq \theta_1 \leq +90$	$-90 \leq \theta_1 \leq +90$
2	$-180 \leq \theta_2 \leq +180$	$-45 \leq \theta_2 \leq +45$	$-45 \leq \theta_2 \leq +45$
3	$-180 \leq \theta_3 \leq +180$	$0 \leq \theta_3 \leq 90$	$0 \leq \theta_3 \leq 90$
4	$-180 \leq \theta_4 \leq +180$	$0 \leq \theta_4 \leq 90$	$0 \leq \theta_4 \leq 90$
Tay gấp	$0 \leq \theta_5 \leq 60$	$0 \leq \theta_5 \leq 60$	$0 \leq \theta_5 \leq 60$

Bảng 6: Giới hạn của các khớp xoay

b. Không gian hoạt động

Không gian hoạt động được xác định là thể tích không gian làm việc của cơ cấu chấp hành cuối mà nó có thể hướng tới. Không gian làm việc bị khống chế bởi kích thước hình học của tay máy cũng như khống chế bởi kích thước hình học của tay máy cũng như khống chế bởi cấu trúc cơ khí của khớp.



Hình 2.31: Không gian hoạt động của robot

2.6. Điều khiển

2.6.1. Tổng quan mô hình điều khiển

a. Bài toán điều khiển chuyển động

Chức năng của hệ thống điều khiển chuyển động là đảm bảo tay robot (End effector) chuyển động bám theo quỹ đạo đặt trước trong môi trường làm việc (không gian làm việc). Chuyển động của tay robot được thực hiện nhờ các hệ thống truyền động khớp robot. Trên cơ sở đó, có hai dạng hệ thống điều khiển chuyển động: hệ thống điều khiển ở không gian khớp và hệ thống điều khiển ở không gian làm việc.

Ở hệ thống điều khiển khớp, đại lượng điều khiển là vị trí của khớp robot: góc quay đối với khớp quay, độ dịch chuyển thẳng đối với khớp tịnh tiến. Bộ điều khiển được thiết kế đảm bảo vị trí khớp luôn bám theo vị trí đặt, tức là sai lệch vị trí khớp hội tụ về không với thời gian nhỏ nhất. Vị trí đặt của khớp được tính toán từ lượng đặt vị trí của tay robot trong không gian làm việc thông qua khâu tính toán động học ngược. Ưu điểm của phương pháp điều khiển ở không gian khớp là bộ điều khiển tác động trực tiếp đến hệ thống truyền động của khớp. Tuy nhiên, hệ thống điều khiển này khó đảm bảo độ chính xác vị trí của tay khi tồn tại các sai lệch trong cơ cấu cơ khí (khe hở trong hộp số,...) hoặc thiếu thông tin quan hệ vị trí giữa tay robot và đối tượng.

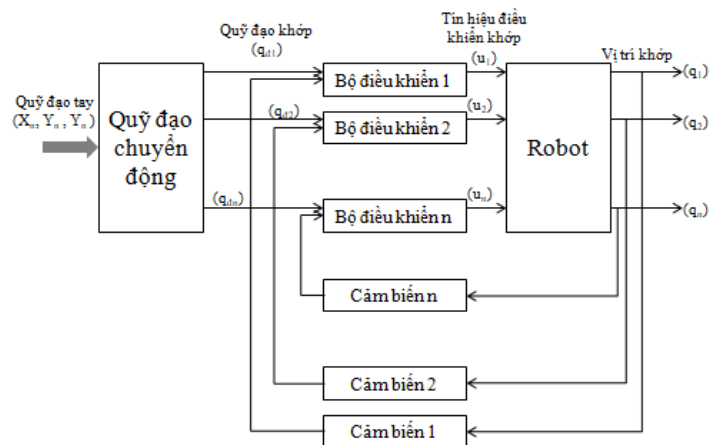
Hệ thống điều khiển không gian làm việc có chức năng duy trì trực tiếp sai lệch vị trí của tay robot trong không gian làm việc bằng không. Lượng đặt của hệ thống điều khiển là vị trí tay trong không gian làm việc bằng không. Khâu tính toán động học ngược sẽ thuộc mạch vòng điều khiển phản hồi. Ưu điểm của phương pháp điều khiển này là tác động trực tiếp các biến không gian làm việc. Nhược điểm là khối tính toán lớn do tồn tại khâu tính toán động học ngược vòng điều khiển.

b. Hệ thống điều khiển độc lập các khớp

Đối với các robot có tỉ số truyền của bộ truyền lớn, có thể coi hệ thống robot n bậc tự do sẽ gồm n hệ thống động lập và là hệ thống 1 đầu vào/1 đầu ra (SISO) và sự ràng buộc giữa các khớp được coi là thành phần nhiễu. Trong trường hợp đơn giản, nếu bỏ qua thành phần mô men nhiễu, sơ đồ cấu trúc mô tả một khớp robot có dạng như hình dưới. Sơ đồ dưới mô tả một khớp bất kỳ và không dùng ký hiệu chỉ số i trong các đại lượng của khớp.

c. Hệ thống điều khiển vị trí

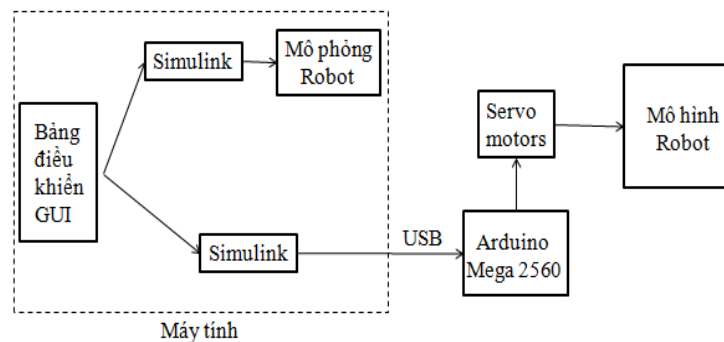
Bộ điều khiển vị trí (Bộ ĐK_i) có chức năng điều khiển chuyển động của robot. Khâu tạo quỹ đạo chuyển động sẽ tính toán các quỹ đạo chuyển động mong muốn của từng khớp ($q_{đi}$) từ quỹ đạo mong muốn của tay robot (x_d, y_d, z_d). Quỹ đạo chuyển động mong muốn khớp ($q_{đi}$) là tín hiệu đặt vị trí của các bộ điều khiển vị trí của từng khớp. Tín hiệu hiệu ra của các bộ điều khiển (u_i) là tín hiệu điều khiển hệ truyền động của khớp tương ứng. Mỗi bộ điều khiển vị trí của từng khớp là bộ điều khiển servo với tín hiệu phản hồi là vị trí của khớp tương ứng được đo bởi các cảm biến vị trí (CB_i).



Hình 2.32: Sơ đồ khối hệ thống điều khiển vị trí

d. Sơ đồ tổng quan hệ thống

Sơ đồ khối của máy tính hỗ trợ điều khiển cánh tay robot được hiển thị trong hình dưới. Tất cả các đơn vị điều khiển chỉ được tập hợp trên một bảng bao gồm bộ giải mã địa chỉ, động cơ bước và ngắt mạch, lập trình. Động cơ bước được gắn trên các khớp xoay của cánh tay robot. Chương trình điều khiển và mô phỏng được lập trình bằng ngôn ngữ C trên phần mềm Matlab với các công cụ hỗ trợ được tích hợp sẵn trong Matlab.

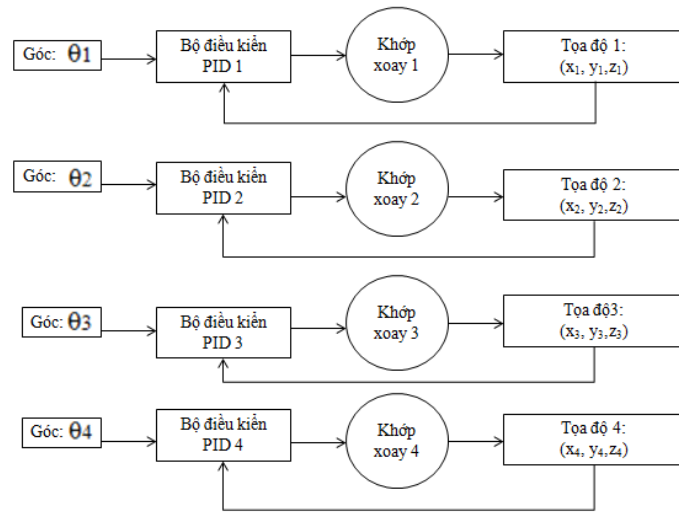


Hình 2.33: Sơ đồ khối tổng quan hệ thống điều khiển.

2.6.2. Điều khiển mô hình trong mô phỏng với thời gian thực

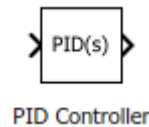
Khi không có điều khiển PID cho từng khớp, mỗi khớp của robot trong mô phỏng sẽ chuyển động 360° xung quay trục quay và hệ tọa độ đã được đặt sẵn. Trong mô phỏng động học cánh tay robot sẽ thực hiện điều khiển với 4 góc xoay độc lập, do đó cần dùng 4 bộ điều khiển PID cho từng khớp xoay:

- Nếu Giá trị đầu vào là các 4 góc với 4 bậc tự do của tay robot thì đầu ra sẽ là tọa độ điểm cuối của robot(động học thuận tay máy).
- Nếu giá trị đầu vào là tọa độ điểm cuối của tay máy thì giá trị đầu ra sẽ là giá trị của các góc xoay(động học nghịch tay máy).

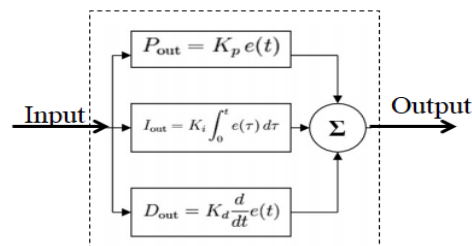


Hình 2.34: Sơ đồ khối điều khiển độc lập từng khớp trong mô phỏng

Trong Simulink để tinh giảm số lượng các khối, tránh dôi mắt mà vẫn đảm bảo được đúng thuật toán điều khiển nhóm đã sử dụng khối “PID Controller” trong Simulink để tính toán song song các giá trị P, I, D.



Hình 2.35: Khối “PID Controller” trong Simulink



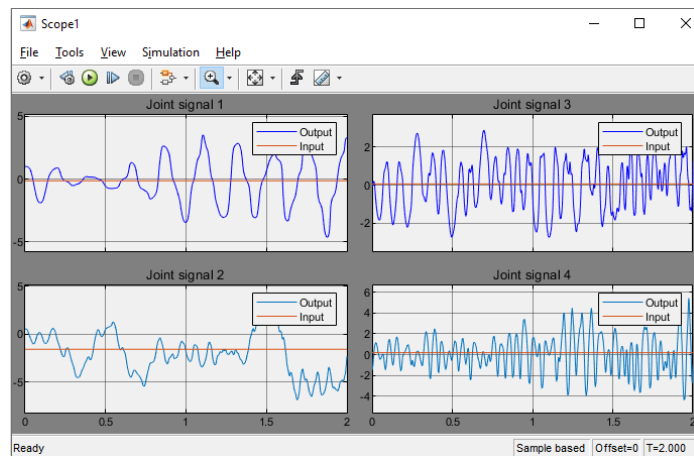
Hình 2.36: Cấu trúc cơ bản của khối PID Controller

Khối PID Controller trong Simulink thực hiện bộ điều khiển PID với tham số miền thời gian liên tục. Đầu ra của khối là tổng, tích phân và đạo hàm của tín hiệu đầu vào. Với các trọng số là các tham số khếch đại tỉ lệ, tích phân và vi phân. Biểu thức của giải thuật PID trong khối PID controller:

$$u(t) = P + I \frac{1}{s} + D \frac{N}{1+N \frac{1}{s}}$$

Trong đó:

- P, I, D lần lượt là các khâu tỉ lệ, tích phân, vi phân
- s là thời gian.
- N là hệ số lọc phái sinh.

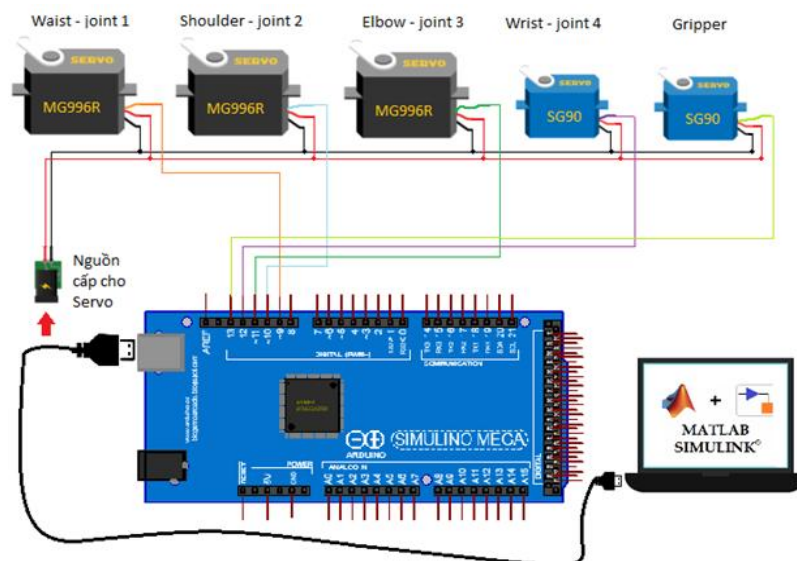


Hình 2.37: Tín hiệu vào và ra của các khớp trong mô phỏng ban đầu

Để tính toán giá trị P, I, D, N ta sử dụng chức năng Tune được tích hợp trong khối “PID Controller” để tự động tính toán.

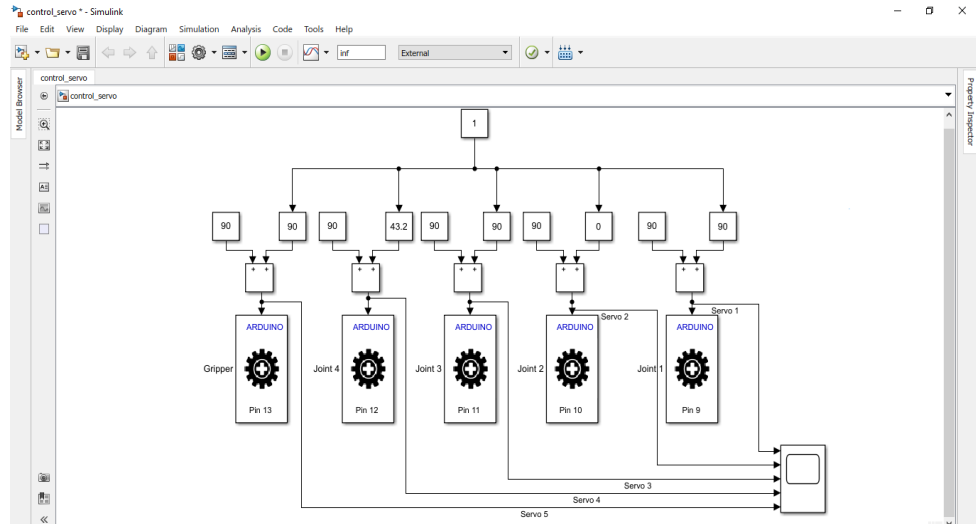
2.6.3. Điều khiển mô hình thực nghiệm cánh tay robot.

a. Kết nối các động cơ Servo với Arduino




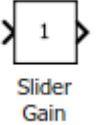

Hình 2.38: Sơ đồ kết nối các động cơ Servo với Arduino

b. Sơ đồ khối Simulink điều khiển động cơ servo



Hình 2.39: Sơ đồ khối điều khiển động cơ servo

Sơ đồ trên sử dụng khối 3 loại khối với chức năng cụ thể của từng khối được mô tả như bảng dưới đây

	Khối hằng số đầu vào
	Khối thay đổi hệ số khuếch đại vô hướng
	Khối đặt góc quay của động cơ servo

Bảng 7: Các khối simulink điều khiển mô hình

Khối “Slider Gain” thay đổi hệ số khuếch đại với giá trị góc mong muốn từ 0 - 180° sau đó truyền dữ liệu vào khối “Standard Servo Write”. Khối “Standard Servo Write” sẽ nhận giá trị đầu vào là các góc quay từ 0 - 180° và tự động truyền xung PWM với độ rộng xung tương ứng với các góc sang động cơ servo.

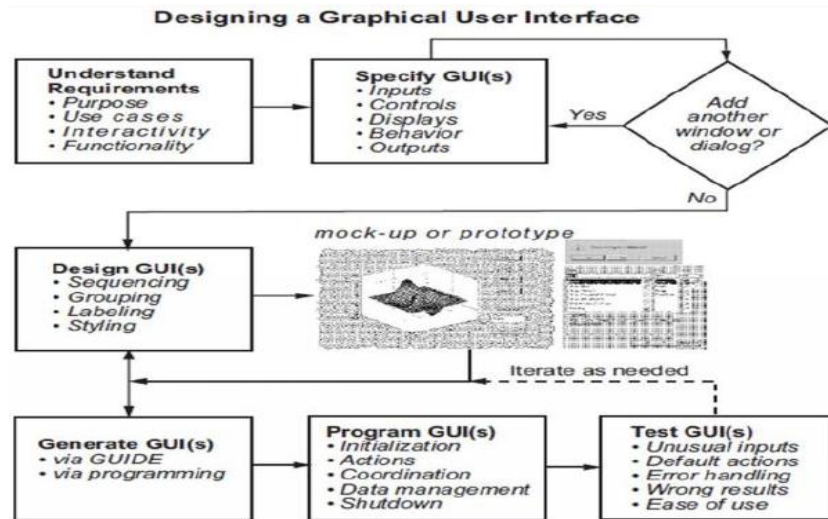
2.6.4. Điều khiển đồng thời mô phỏng và mô hình thực nghiệm

a. Giới thiệu sơ lược về GUI Matlab.

- Tổng quan về GUI

GUI được viết tắt của từ “**Graphical User Interface**” dịch theo tiếng việt nôm la là **Giao diện người dùng đồ họa**. Là ta sẽ tiến hành sử dụng chuột để thao tác các thanh công cụ đã được con người lập trình sẵn và tích hợp vào phần

mền Matlab. Trong Matlab thì GUI hỗ trợ khá là đầy đủ các chương trình để thực hiện như là tính toán với phép toán LOGIC, lập trình không gian 2D, 3D, đọc dữ liệu từ Excel, xử lý hình ảnh, đọc hiển thị dữ liệu, liên kết đa phương tiện. Giao tiếp với người dùng thông qua hình ảnh, các nút nhấn thực thi ...

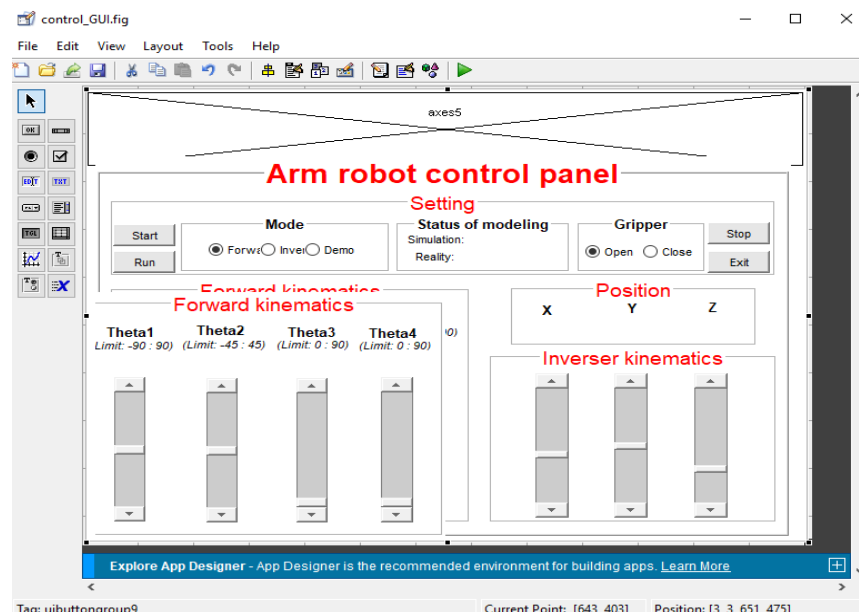


Hình 2.40: Sơ đồ trình tự thực hiện với GUI

b. Thiết kế bảng điều khiển cánh tay robot

- Thiết kế bảng điều khiển cánh tay robot thực nghiệm

Yêu cầu đặt ra của công việc thiết kế bảng điều khiển cánh tay robot là bảng điều khiển đó phải có chức năng là tính toán động học thuận, động học nghịch của tay máy và hiển thị các thông số như góc quay của từng khớp hay tọa độ của điểm cuối. Bên cạnh đó, người dùng cũng có thể tùy chọn chế độ tự động(auto) cho robot.



Hình 2.41: Giao diện bảng điều khiển hoàn chỉnh

Mô tả các chức năng cơ bản của bảng điều khiển cánh tay robot 4 bậc tự do:

Đối tượng	Chức năng
Nút “Start”	Mở sơ đồ simulink mô phỏng và sơ đồ simulink điều khiển mô hình robot cánh tay robot.
Nút “Run”	Bắt đầu quá trình mô phỏng và điều khiển mô hình thực tế
Nút “Stop”	Dừng mô phỏng và điều khiển mô hình
Nút “Exit”	Lưu và thoát khỏi các cửa sổ simulink
Tùy chọn “Mode”	Người dùng có thể tùy chọn chế độ làm việc của robot ở đây. Có 3 chế độ làm việc: <ul style="list-style-type: none"> - Forward: chế độ động học thuận. - Inverse: chế độ động học nghịch. - Auto: chế độ tự động.
Hộp “Status of modeling”	Hộp này hiển thị tình trạng kết nối của sơ đồ simulink mô phỏng và mô hình: <ul style="list-style-type: none"> - Simulation: mô phỏng. - Reality: mô hình thực tế.
Tùy chọn “Gripper”	Tùy chọn này giúp người dùng điều khiển phần tay gắp của robot, có 2 tùy chọn: <ul style="list-style-type: none"> - Open: mở - Close: đóng
Chức năng “Forward kinematics” – động học thuận	Sau khi chọn chế độ “Forward”, chức năng động học thuận này sẽ tự động hiện lên. Nhóm này cho phép người dùng thay đổi các giá trị góc xoay của từng khớp thông qua các thanh trượt và hiển thị chúng.
Chức năng “Inverse kinematics” – động học nghịch	Chức năng này sẽ được hiển thị sau khi người dùng chọn chế độ “Inverse”. Nhóm này có chức năng là thay đổi các giá trị của tọa độ điểm cuối x, y, z . Để từ các giá trị này thông qua phương pháp động học nghịch có thể tính toán được giá trị góc xoay của từng khớp.

Chức năng “Posittion”	Hiển thị giá trị tọa độ x, y, z.
-----------------------	----------------------------------

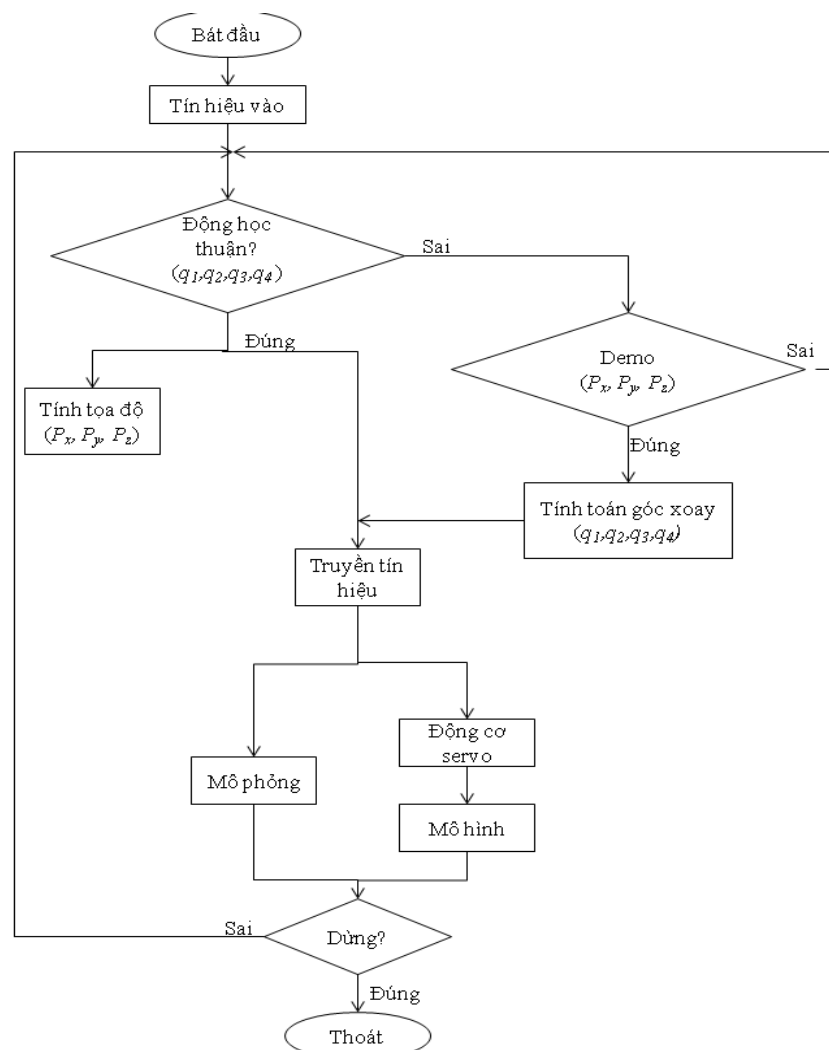
Bảng 8: Các chức năng của bảng điều khiển cánh tay robot

c. Lập trình điều khiển song song mô hình thực nghiệm và mô phỏng

Hầu hết GUI chỉ thực hiện (trả lời) lệnh người dùng thông qua các tác động của người dùng lên giao diện, người sử dụng không cần biết cấu trúc chương trình vẫn có thể thực hiện được. GUI được thực hiện thông qua các hàm CALLBACK, khi người dùng tác động lên giao diện bằng các cách khác nhau, hàm CALLBACK sẽ được gọi để thực thi.

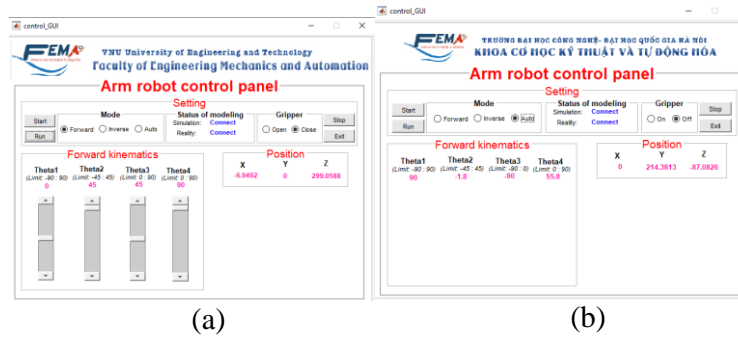
Chương trình điều khiển có chức năng phân tích các chế độ được người dùng tương tác và lựa chọn bao gồm:

- Chế độ demo: Chế độ này được lập trình để robot trong mô hình và mô phỏng thực thi một công việc nào đó trên cơ sở của động học nghịch.
- Chế độ động học thuận: Chế độ này nhận các giá trị đầu vào là các góc xoay (q_1, q_2, q_3, q_4) từ đó tính toán tọa độ điểm cuối (P_x, P_y, P_z)



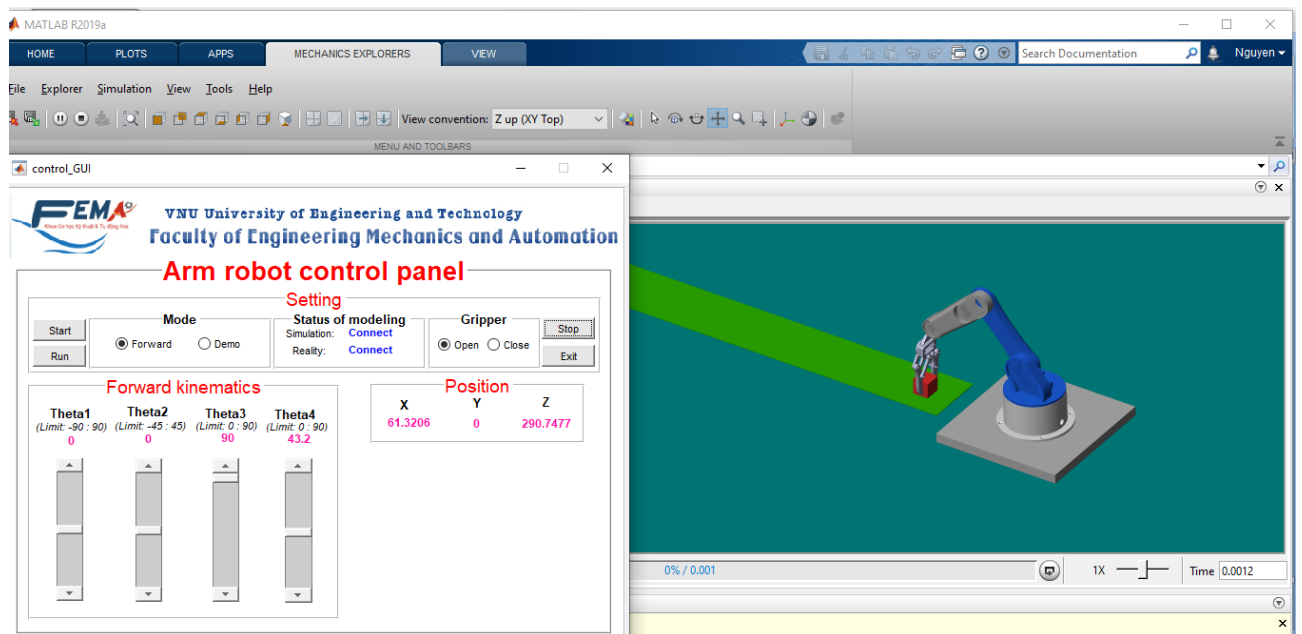
Hình 2.42: Lưu đồ thuật toán hệ thống điều khiển

d. Kết quả

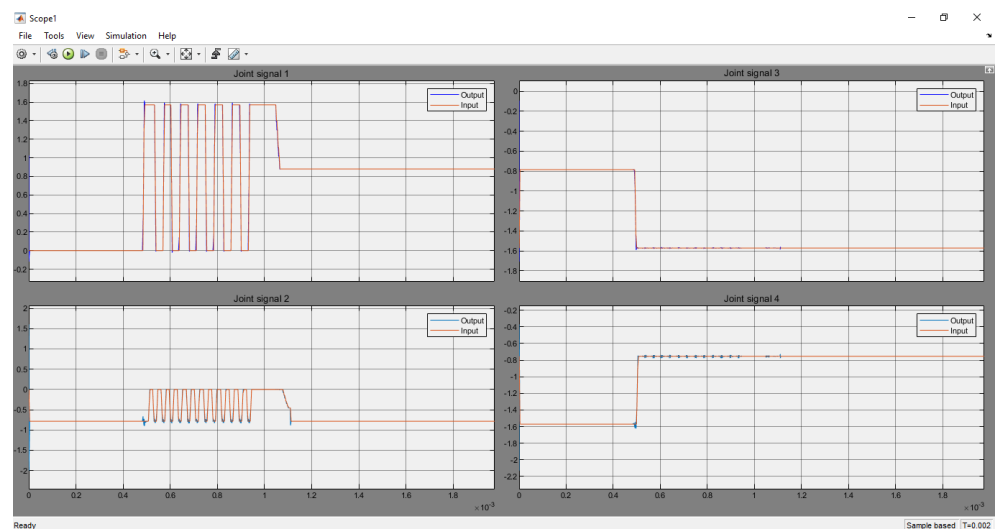


Hình 2.43: Bảng điều khiển robot ở các chế độ khác nhau

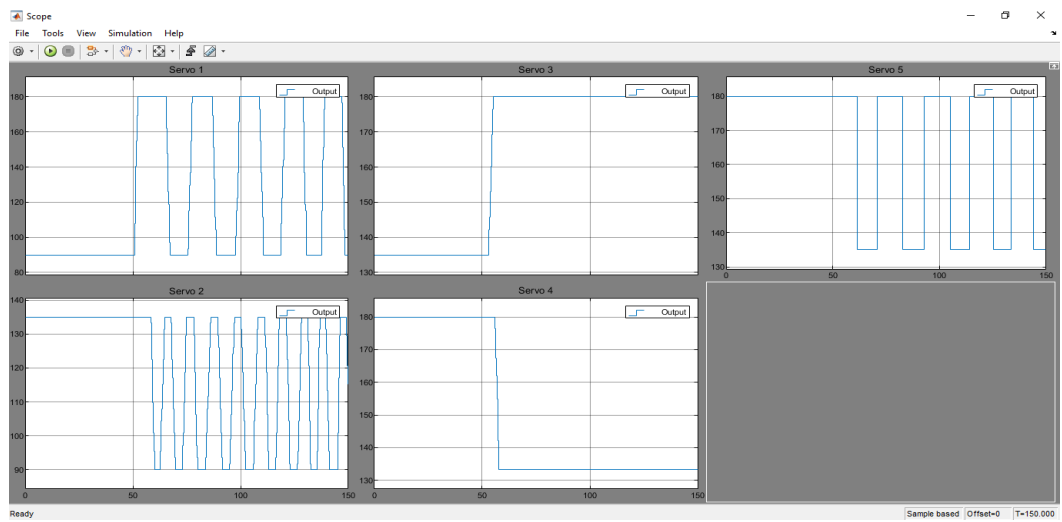
(a) Chế độ động học thuận; (b) Chế độ demo



Hình 2.44: Mô phỏng robot trên máy tính



Hình 2. 45: Sơ đồ tín hiệu tại các khớp xoay trong mô phỏng ở chế độ tự động



Hình 2.46: Sơ đồ tín hiệu của các động cơ servo

III. KẾT LUẬN

1. Kết luận

Qua quá trình nghiên cứu, thiết kế, xây dựng mô hình, mô phỏng và điều khiển đề tài nghiên cứu của chúng em đã thực hiện được một số công việc sau:

- Nghiên cứu được tổng quan lý thuyết về robot, các khái niệm, các cơ cấu, nguyên lý hoạt động và cụ thể hơn là cánh tay robot 4 bậc tự do. Từ cơ sở trên để thiết kế ra cánh tay robot 4 bậc tự do. Vận dụng các kiến thức về cơ khí, chế tạo máy, in 3D để thiết kế cánh tay robot 4 bậc tự do từ phần mềm thiết kế Autodesk Inventor.
- Nghiên cứu phương pháp giải bài toán động học của tay máy. Từ đó làm cơ sở để điều khiển mô hình và mô phỏng các chuyển động của cánh tay robot. Xây dựng mô hình hoàn chỉnh cánh tay robot với các chuyển động thông qua chuyển động quay của động cơ servo và được điều khiển bằng Arduino thông qua công cụ Matlab/Simulink.
- Xây dựng giao diện trên PC/laptop từ giao diện người dùng đồ họa (GUI) được tích hợp trên Matlab để điều khiển song song đồng thời cả mô hình thực tế và mô phỏng các chuyển động của cánh tay robot.

2. Hạn chế

- Chưa mô phỏng được cơ cấu gấp vật trong mô hình mô phỏng giống mô hình thực nghiệm.
- Cơ cấu cơ khí của mô hình thực nghiệm còn đơn giản do gắn thẳng trực tiếp động cơ vào các khớp không thông qua bộ giảm tốc nên tốc độ quay lớn dẫn đến hiện tượng mô hình bị rung, lắc.
- Tay gấp của mô hình thực nghiệm chưa có biện pháp để kiểm soát lực gấp dẫn đến nếu lực gấp vật vượt quá giới hạn hay nói cách khác là động cơ servo không quay được tới giá trị góc mong muốn thì động cơ servo sẽ phản hồi tín hiệu nhiễu về chương trình điều khiển dẫn đến việc thoát khỏi chương trình điều khiển và mô hình thực nghiệm không còn hoạt động.

3. Hướng phát triển

- Tính toán bộ điều khiển khác đáp ứng nhanh hơn và ổn định tốt hơn.
- Thiết kế mô hình thực nghiệm kết hợp các bộ giảm tốc để, bánh răng để giảm thiểu hiện tượng rung lắc của hệ thống.
- Lắp các cảm biến tại các khớp xoay của mô hình robot thực nghiệm để kết hợp thuật toán điều khiển tại các khớp xoay của mô hình thực nghiệm.

TÀI LIỆU THAM KHẢO

Tiếng việt

1. Nguyễn Trường Thịnh (2014), *Giáo trình kỹ thuật robot*, Nhà xuất bản đại học Quốc gia TP. Hồ Chí Minh, TP. Hồ Chí Minh.
2. Nguyễn Thiện Phúc (2006), *Robot công nghiệp*, Nhà xuất bản Khoa học và Kỹ thuật.
3. Nguyễn Mạnh Tiến (2007), *Điều khiển robot công nghiệp*, Nhà xuất bản Khoa học và Kỹ thuật, Hà Nội.
4. Nguyễn Phùng Quang (2004), *Matlab & Simulink*, Nhà xuất bản Khoa học và Kỹ thuật, Hà Nội.
5. Nguyễn Thái Phong(2015), *Nghiên cứu xây dựng mô hình cánh tay robot 5 bậc tự do*, Đồ án tốt nghiệp chuyên ngành điện tự động công nghiệp, Trường Đại học Dân Lập Hải Phòng.
6. Vũ Minh Hùng, Trịnh Quang Trung và Võ Quốc Thắng (2016), *Nghiên cứu cấu trúc động học của một tay máy công nghiệp dư dẫn động dạng chuỗi*, Trường Đại học Dầu khí Việt Nam (PVU).

Tiếng Anh

7. The MathWorks, Inc (2015), *Matlab[®] Creating Graphical User Interfaces*, Mathworks,.
8. Moe Thu Zar and Wai Phyo Ei (November 2017), *Point to Point Trajectory Control of 3R Planar Robot Arm*, International Journal of Science - Engineering and Technology Research (IJSETR), 2278 -7798.
9. Sam Cubero (January 2007), *Industrial Robotics Theory, Modelling and Control*, Advanced Robotic Systems International, Croatia.
10. John J. Craig (2005), *Introduction to Robotics Mechanics and Control - Third Edition*, Pearson Education International, United States of America.
11. Zhiqiang Chao, Fei Wang, Chuanqing Zhang et al (2017), *Inverse Kinematics Solution and Verification of 4-DOF Hydraulic Manipulator*, IOP Conf, 916 012022.
12. Chun Htoo Aung, Khin Thandar Lwin, and Yin Mon Myint (2008), *Modeling Motion Control System for Motorized Robot Arm using MATLAB*, World Academy of Science, Engineering and Technology.
13. Amin A. Mohammed and M. Sunar (2015), *Kinematics Modeling of a 4-DOF Robotic Arm*, International Conference on Control, Automation and Robotics,

Saudi Arabia, 09-ELE786-04.

14. Mohammed Reyad AbuQassem (June 2010), *Simulation and Interfacing of 5 DOF Educational Robot Arm*, Master Thesis for major Engineering Electrical, Islamic University of Gaza, Gaza - Palestine.
15. Osman Gurdal, Mehmet Albayrak and Tuncay Aydogan, *Computer Aided control and simualtion of Arm moving in there dimension*, Gazi University, Turkey.
16. Nourdine Aliane(May 2010), *A Matlab/Simulink-Based Interactive Module for Servo Systems Learning*, IEE Transactions on Education.
17. Das, M.T. and L.C.Dulger, 2005. "Mathematical modelling, simulation and experimental verification of a scara robot", *Simulation Modelling Practice and Theory* 13, pp.257–271
18. Kinematic Modeling and Simulation of a SCARA Robot by Using Solid Dynamics and Verification by MATLAB/Simulink- Mohanraj Selvakumar

Nguồn trên Internet

19. https://vi.wikipedia.org/wiki/Bộ_điều_khiển_PID
20. <https://vi.wikipedia.org/wiki/Robot>
21. <https://www.mathworks.com/>
22. https://www.youtube.com/watch?v=R_U_2K6ii-8
23. <https://www.youtube.com/watch?v=p1wIJut1bTs&t=581s>
24. https://www.youtube.com/watch?v=rJx_XZ8xz6w&t=407s
25. <https://www.youtube.com/watch?v=8MzC8Nmr2rg>