

BỘ GIÁO DỤC & ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH
KHOA ĐIỆN – ĐIỆN TỬ
BỘ MÔN TỰ ĐỘNG ĐIỀU KHIỂN

----- ♣ Δ ♣ -----



BÁO CÁO MÔN HỌC
THỰC TẬP KỸ THUẬT ROBOT
ĐỀ TÀI: THIẾT KẾ VÀ ĐIỀU KHIỂN CÁN H TAY
ROBOT 3 BẬC TỰ DO

GVHD: TS. Nguyễn Văn Thái

SVTH:	MSSV:
Nguyễn Đức Mạnh	19151253
Tạ Trần Nhật Minh	19151034

Tp. Hồ Chí Minh tháng 12 năm 2022

Mục lục

MỤC LỤC	
Mục lục	i
Danh sách hình ảnh	ii
Chương 1. THIẾT KẾ VÀ THI CÔNG MÔ HÌNH	3
1.1 Thiết kế và xây dựng mô hình	3
1.2 Các thiết bị được sử dụng trong mô hình	5
1.2.1 Thiết bị thực hiện chức năng truyền động	5
1.2.2 Thiết bị điều khiển	6
Chương 2. TÍNH TOÁN VÀ KIỂM NGHIỆM KẾT QUẢ ĐỘNG HỌC ROBOT 3-DOF	10
2.1 Tính toán động học thuận	10
2.2 Tính toán động học nghịch	12
2.3 Kiểm nghiệm kết quả động học trên Matlab Simulink	14
Chương 3. VẼ KHÔNG GIAN LÀM VIỆC VÀ QUY HOẠCH QUỸ ĐẠO CHO ROBOT	16
3.1 Xác định không gian làm việc của robot	16
3.2 Quy hoạch quỹ đạo cho robot	17
Chương 4. CHƯƠNG TRÌNH VÀ GIAO DIỆN ĐIỀU KHIỂN	19
4.1 Sơ đồ khối điều khiển cánh tay robot 3 bậc tự do	19
4.2 Giao diện điều khiển	19
Chương 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	22
5.1 Kết luận	22
5.2 Hướng phát triển	22

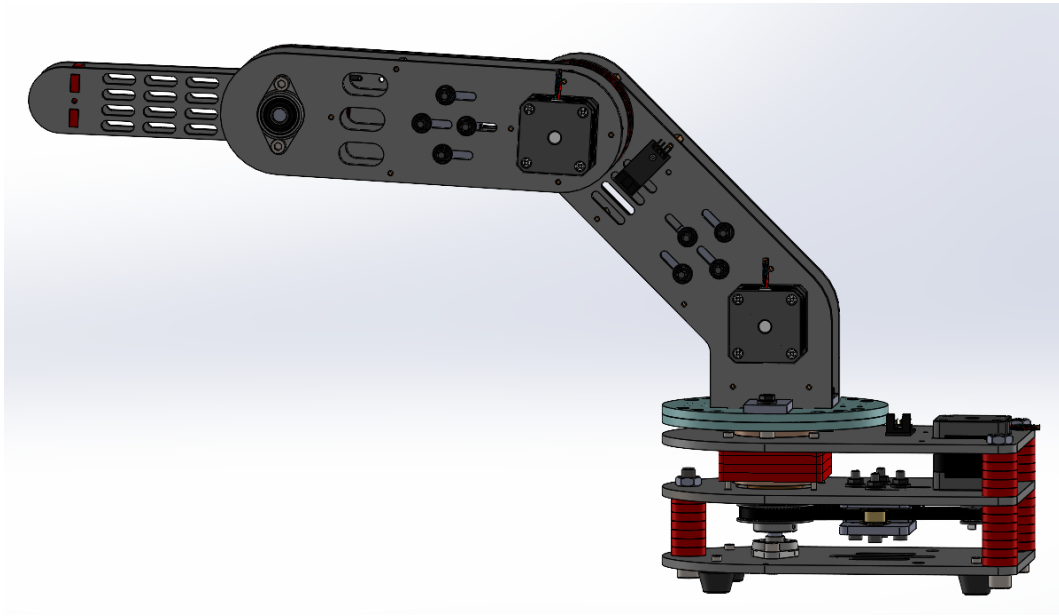
Danh sách hình ảnh

Hình 1.1: Mô hình Robot 3-DOF	1
Hình 1.2: Phần đế của robot	1
Hình 1.3: Link 1 của robot	2
Hình 1.4: Link 2 của robot	2
Hình 1.5: Link 3 của robot	3
Hình 1.6: Puly GT2 16 răng	3
Hình 1.7: Puly GT2 60 răng	3
Hình 1.8: Dây đai GT2 300mm	4
Hình 1.9: Động cơ bước 17HS4401	4
Hình 1.10: Board Arduino Uno	5
Hình 1.11: Board Arduino mở rộng Shield V3	5
Hình 1.12: Driver A4988	6
Hình 1.13: Công tắc hành trình D35	7
Hình 1.14: Nguồn tổ ong 24VDC-10A	7
Hình 2.1: Hệ trục tọa độ của robot	9
Hình 2.2: Mô phỏng giá trị góc khớp với 3 góc $\theta = 0$	13
Hình 2.3: Mô phỏng giá trị góc khớp với $\theta_1 = 45^\circ$, $\theta_2 = 90^\circ$, $\theta_3 = 30^\circ$	13
Hình 3.1: Không gian làm việc với $\theta = 0$	14
Hình 4.1: Sơ đồ khối điều khiển	17
Hình 4.2: Giao diện mở đầu	18
Hình 4.3: Giao diện điều khiển	19

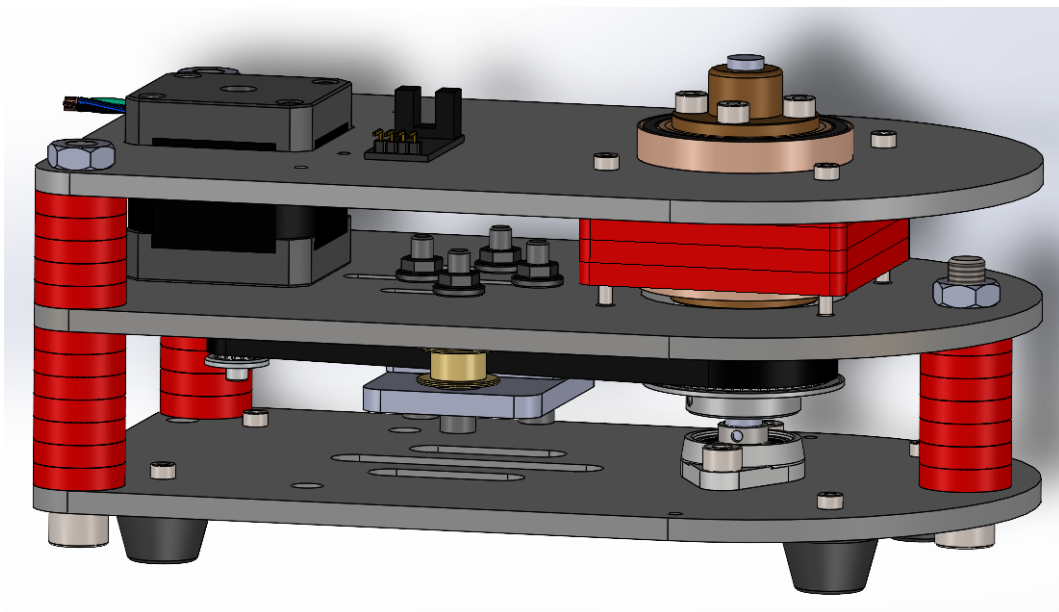
Chương 1. THIẾT KẾ VÀ THI CÔNG MÔ HÌNH

1.1 Thiết kế và xây dựng mô hình

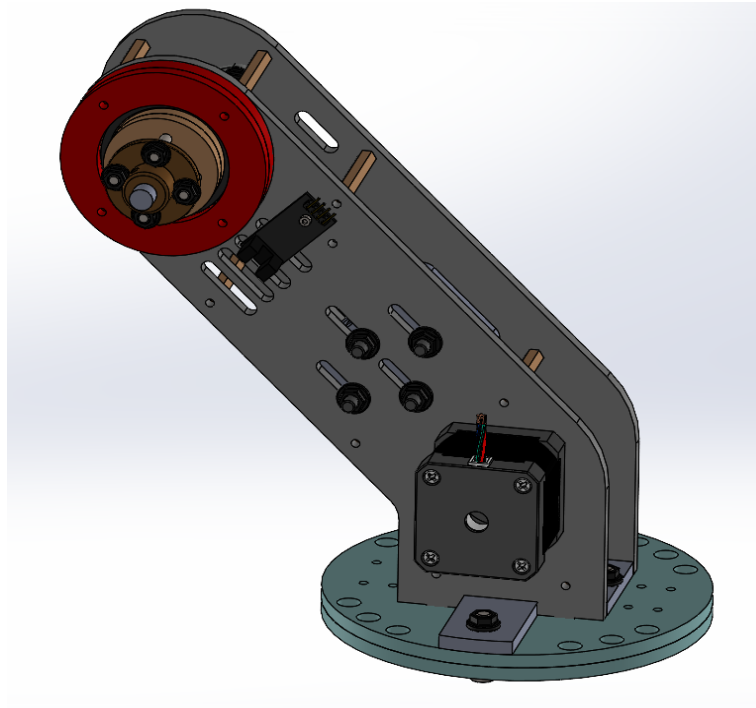
Mô hình cánh tay robot 3 bậc tự do được thiết kế trên phần mềm SolidWorks như hình sau:



Hình 1.1: Mô hình Robot 3-DOF



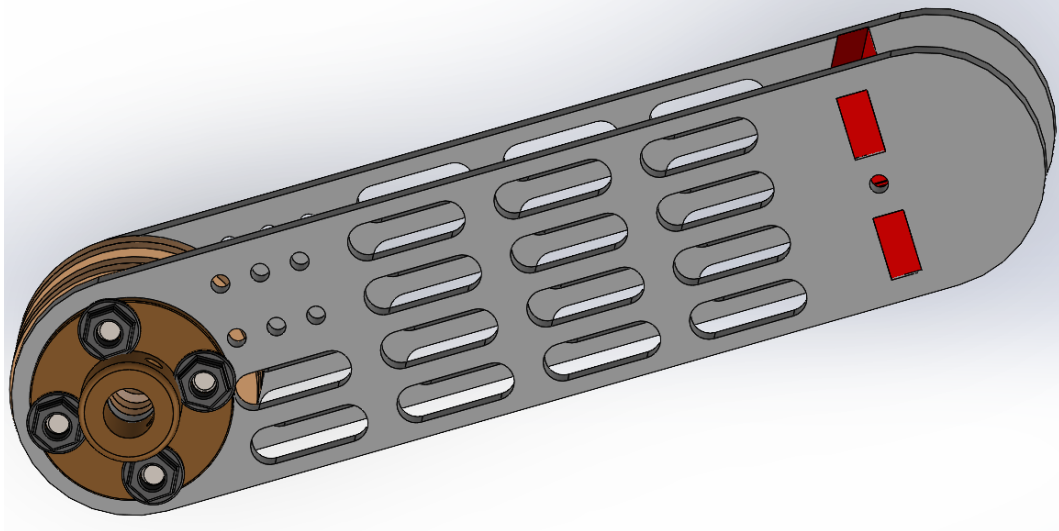
Hình 1.2: Phần đế của robot



Hình 1.3: Link 1 của robot



Hình 1.4: Link 2 của robot



Hình 1.5: Link 3 của robot

1.2 Các thiết bị được sử dụng trong mô hình

1.2.1 Thiết bị thực hiện chức năng truyền động

- *Puly GT2:*



Hình 1.6: Puly GT2 16 răng



Hình 1.7: Puly GT2 60 răng

- *Vòng dây đai GT2:*

CHƯƠNG 2. TÍNH TOÁN VÀ KIỂM NGHIỆM KẾT QUẢ ĐỘNG HỌC ROBOT 3-DOF



Hình 1.8: Dây đai GT2 300mm

- Động cơ bước:



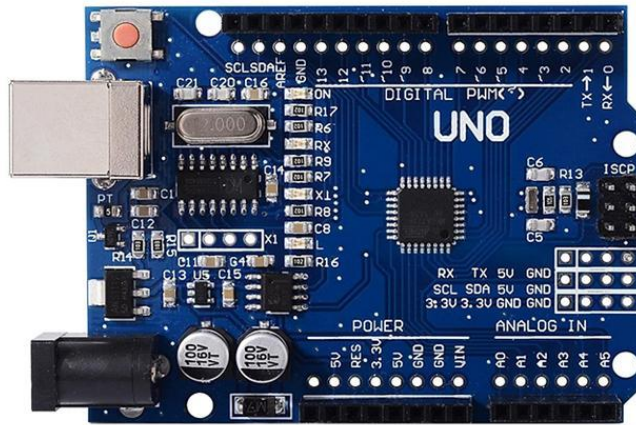
Hình 1.9: Động cơ bước 17HS4401

- Thông số kỹ thuật:
 - + Góc quay mỗi bước: 1.8° (200 bước / vòng quay)
 - + Dòng điện: 1.7A
 - + Mômen Xoắn giữ: 40N.cm
 - + Mômen xoắn: 2.2N.cm
 - + Rotor quán tính: 54g.cm²
 - + Độ chính xác góc bước: $\pm 5\%$ (bước đầy đủ, không tải)
 - + Khối lượng động cơ: 280g
 - + Chiều dài thân: 40 mm
 - + Kích thước khung: 42 x 42mm
 - + Đường kính trục: 5mm
 - + Chiều dài trục: 23mm

1.2.2 Thiết bị điều khiển

- Board Arduino Uno:

CHƯƠNG 2. TÍNH TOÁN VÀ KIỂM NGHIỆM KẾT QUẢ ĐỘNG HỌC ROBOT 3-DOF

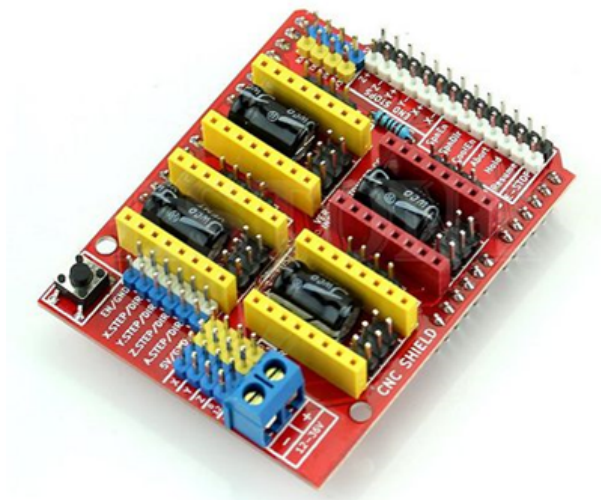


Hình 1.10: Board Arduino Uno

- Thông số kỹ thuật:
 - + Điện áp hoạt động: 5VDC
 - + Tần số hoạt động: 16MHz
 - + Dòng tiêu thụ: khoảng 30mA
 - + Số chân Digital I/O: 14 (6 chân PWM)
 - + Số chân Analog: 6 (độ phân giải 10 bit)
 - + Dòng tối đa trên mỗi chân I/O: 30 mA
 - + Dòng tối đa (5V): 500 mA
 - + Dòng tối đa (3.3V): 50 mA
 - + Bộ nhớ flash: 32KB
 - + SRAM: 2KB
 - + EEPROM: 1KB

- Board CNC Shield V3:

CHƯƠNG 2. TÍNH TOÁN VÀ KIỂM NGHIỆM KẾT QUẢ ĐỘNG HỌC ROBOT 3-DOF

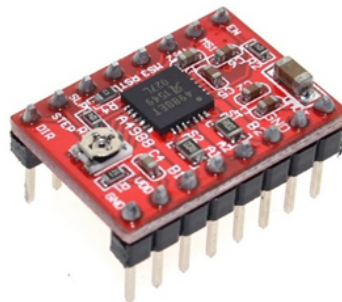


Hình 1.11: Board Arduino mở rộng Shield V3

- Thông số kỹ thuật:

- + Tương thích GRBL (mã nguồn mở chạy trên Arduino UNO R3 để điều khiển CNC mini)
- + Hỗ trợ lên tới 4 trục (trục X, Y, Z và một trục thứ tư tùy chọn)
- + Hỗ trợ tới 2 Endstop (cảm biến đầu cuối) cho mỗi trục
- + Tính năng điều khiển spindle
- + Tính năng điều khiển dung dịch làm mát khi máy hoạt động
- + Sử dụng các mô đun điều khiển động cơ bước, giúp tiết kiệm chi phí khi thay thế, nâng cấp
- + Thiết lập độ phân giải bước động cơ bằng jump đơn giản
- + Thiết kế nhỏ gọn, các đầu nối tiêu chuẩn thông dụng
- + Điện áp nguồn cấp đa dạng từ 12V tới 36V.

- *Driver điều khiển động cơ bước:*



Hình 1.12: Driver A4988

- Thông số kỹ thuật:

CHƯƠNG 2. TÍNH TOÁN VÀ KIỂM NGHIỆM KẾT QUẢ ĐỘNG HỌC ROBOT 3-DOF

- + Điện áp cấp tối thiểu: 8 V
- + Điện áp cấp cực đại: 35 V
- + Dòng cấp liên tục cho mỗi pha: 1 A (không cần tản nhiệt, làm mát)
- + Dòng cấp liên tục cho mỗi pha: 2 A (khi có làm mát, tản nhiệt)
- + Điện áp logic 1 tối thiểu: 3 V
- + Điện áp logic 1 tối đa: 5.5 V
- + Độ phân giải: full, 1/2, 1/4, 1/8, và 1/16

- Công tắc hành trình:



Hình 1.13: Công tắc hành trình D35

- Thông số kỹ thuật:
 - + Dòng định mức: 15A
 - + Điện áp: 250VAC
 - + Cấu hình tiếp điểm: SPDT-NO/NC
 - + Lực tác động: 1.23 N
 - + Tuổi thọ: 3.000.000 lần

- Nguồn tổ ong:



Hình 1.14: Nguồn tổ ong 24VDC-10A

- Thông số kỹ thuật:
 - + Điện áp đầu vào: 110-220VAC chỉnh bằng công tắc gạt.
 - + Công suất: 360W
 - + Dòng đầu ra tối đa: 15A
 - + Nhiệt độ làm việc -10 ~ 60 độ C
 - + Kích thước: 215x115x50mm

Chương 2. TÍNH TOÁN VÀ KIỂM NGHIỆM KẾT QUẢ ĐỘNG HỌC ROBOT 3-DOF

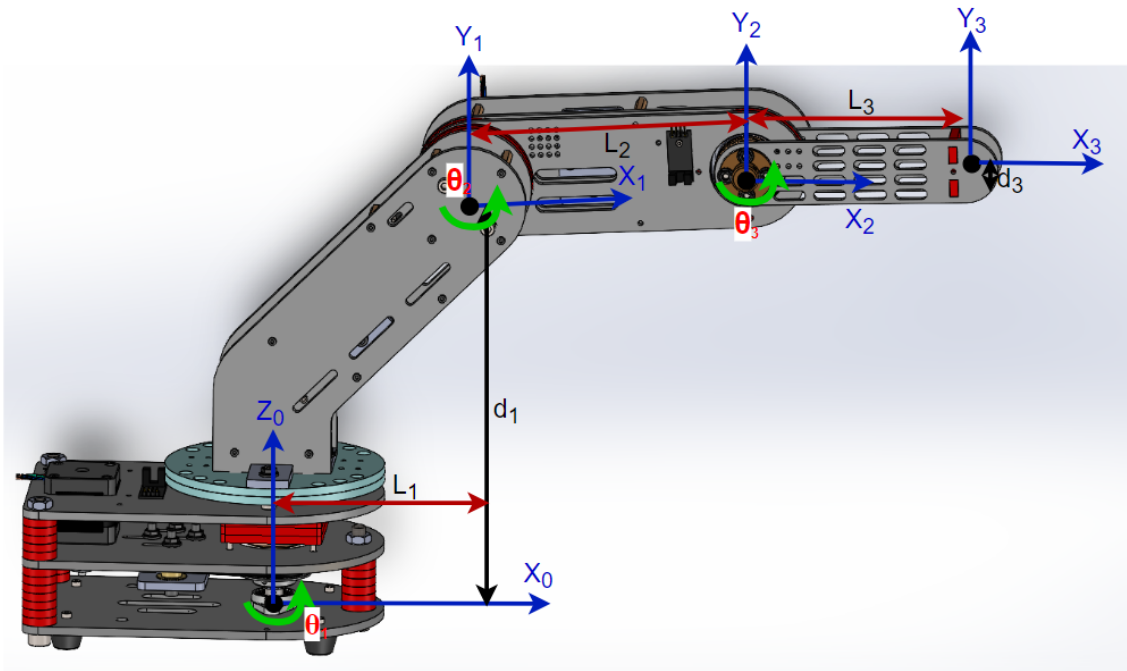
2.1 Tính toán động học thuận

Mục đích của bài toán động học thuận là xác định vị trí của khâu tác động cuối của tay máy khi biết các biến khớp của tay máy.

Các bước tính toán động học thuận của cánh tay robot 3 bậc tự do được trình bày như sau:

Bước 1: Đặt trục tọa độ cho hệ cánh tay robot

CHƯƠNG 2. TÍNH TOÁN VÀ KIỂM NGHIỆM KẾT QUẢ ĐỘNG HỌC ROBOT 3-DOF



Hình 2.1: Hệ trục tọa độ của robot

Bảng thông số vật lý của mô hình:

Thông số	Ý nghĩa
$L_1 = 105 \text{ mm}$	Chiều dài của khâu 1
$L_2 = 162 \text{ mm}$	Chiều dài của khâu 2
$L_3 = 130 \text{ mm}$	Chiều dài của khâu 3
$d_1 = 270 \text{ mm}$	Độ lệch của khớp
$d_3 = 55 \text{ mm}$	

Bước 2: Lập bảng D-H (Denavit – Hartenberg).

i	a	α	d	θ
1	L_1	90°	d_1	θ_1
2	L_2	0	0	θ_2
3	L_3	0	d_3	θ_3

- Trong đó:

CHƯƠNG 2. TÍNH TOÁN VÀ KIỂM NGHIỆM KẾT QUẢ ĐỘNG HỌC ROBOT 3-DOF

- + a là chiều dài của mỗi khâu được xác định bằng đường vuông chung giữa hai trục Z.
- + α là độ xoắn của mỗi khâu được xác định bằng góc lệch giữa hai trục Z.
- + d là độ lệch của khớp được xác định bằng đường vuông góc chung hay khoảng cách giữa hai mặt phẳng chứa trục tọa độ x_i và x_{i+1} .
- + θ là góc khớp được xác định bằng góc lệch giữa hai khâu liên kề.

Bước 3: Ta tiến hành thay lần lượt các giá trị trong bảng D-H vào ma trận chuyển đổi sau:

Ta có ma trận chuyển đổi tổng quát từ hệ thứ i sang hệ thứ $i+1$:

$$T_{i+1}^i = \begin{bmatrix} \cos(\theta) & -\sin(\theta)\cos(\alpha) & \sin(\theta)\sin(\alpha) & a\cos(\theta) \\ \sin(\theta) & \cos(\theta)\cos(\alpha) & -\cos(\theta)\sin(\alpha) & a\sin(\theta) \\ 0 & \sin(\alpha) & \cos(\alpha) & d \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ma trận chuyển đổi từ hệ 0 sang hệ 1:

$$T_1^0 = \begin{bmatrix} \cos(\theta_1) & 0 & \sin(\theta_1) & L_1 \cos(\theta_1) \\ \sin(\theta_1) & 0 & -\cos(\theta_1) & L_1 \sin(\theta_1) \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ma trận chuyển đổi từ hệ 1 sang hệ 2:

$$T_2^1 = \begin{bmatrix} \cos(\theta_2) & -\sin(\theta_2) & 0 & L_2 \cos(\theta_2) \\ \sin(\theta_2) & \cos(\theta_2) & 0 & L_2 \sin(\theta_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ma trận chuyển đổi từ hệ 2 sang hệ 3:

$$T_3^2 = \begin{bmatrix} \cos(\theta_3) & -\sin(\theta_3) & 0 & L_3 \cos(\theta_3) \\ \sin(\theta_3) & \cos(\theta_3) & 0 & L_3 \sin(\theta_3) \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ma trận chuyển đổi tổng quát từ hệ thế 0 sang hệ thứ 3 như sau:

$$T_3^0 = T_1^0 \times T_2^1 \times T_3^2 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & P_x \\ r_{21} & r_{22} & r_{23} & P_y \\ r_{31} & r_{32} & r_{33} & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Trong ma trận này chứa hai thành phần vị trí và hướng của cơ cấu cuối nên ta suy ra tọa độ vị trí cơ cấu cuối:

$$\begin{aligned} P_x &= L_1 \cos(\theta_1) + d_3 \sin(\theta_1) + L_2 \cos(\theta_1) \cos(\theta_2) + L_3 \cos(\theta_1) \cos(\theta_2) \cos(\theta_3) - L_3 \cos(\theta_1) \sin(\theta_2) \sin(\theta_3) \\ P_y &= L_1 \sin(\theta_1) - d_3 \cos(\theta_1) + L_2 \cos(\theta_2) \sin(\theta_1) + L_3 \cos(\theta_2) \cos(\theta_3) \sin(\theta_1) - L_3 \sin(\theta_1) \sin(\theta_2) \sin(\theta_3) \\ P_z &= d_1 + L_3 \sin(\theta_2 + \theta_3) + L_2 \sin(\theta_2) \end{aligned}$$

2.2 Tính toán động học nghịch

Mục đích của bài toán động học nghịch là tìm các biến khớp của tay máy khi biết vị trí khâu tác động cuối của tay máy. Có 2 phương pháp để giải bài toán động học nghịch là phương pháp hình học và phương pháp đại số. Ở đây nhóm đã áp dụng phương pháp đại số để giải. Các bước thực hiện được trình bày dưới đây:

Bước 1: Từ ta nhân hai vế của đẳng thức cho ma trận T_1^0 nghịch đảo.

$${}^1T^{-1} {}^3T = {}^1T^{-1} ({}^1T {}^2T {}^3T)$$

Kết quả cho các thành phần (*) giống nhau và các thành phần khác nhau như n_1, m_1, m_2, \dots .

CHƯƠNG 2. TÍNH TOÁN VÀ KIỂM NGHIỆM KẾT QUẢ ĐỘNG HỌC ROBOT 3-DOF

$$\begin{bmatrix} * & * & * & n_1 \\ * & * & * & n_2 \\ * & * & * & n_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} * & * & * & m_1 \\ * & * & * & m_2 \\ * & * & * & m_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Trong đó:

$$\begin{cases} n_1 = P_x \cos(\theta_1) - L_1 + P_y \sin(\theta_1) \\ n_2 = P_z - d_1 \\ n_3 = P_x \sin(\theta_1) - P_y \cos(\theta_1) \end{cases} \quad \text{và} \quad \begin{cases} m_1 = L_3 \cos(\theta_2 + \theta_3) + L_2 \cos(\theta_2) \\ m_2 = L_3 \sin(\theta_2 + \theta_3) + L_2 \sin(\theta_2) \\ m_3 = d_3 \end{cases}$$

Bước 2: Tính θ_1 .

Ta có: $n_3 = m_3 \Rightarrow P_x \sin(\theta_1) - P_y \cos(\theta_1) = d_3$

$$\Rightarrow \frac{P_x}{\sqrt{(P_x)^2 + (P_y)^2}} \sin(\theta_1) - \frac{P_y}{\sqrt{(P_x)^2 + (P_y)^2}} \cos(\theta_1) = \frac{d_3}{\sqrt{(P_x)^2 + (P_y)^2}}$$

Đặt: $\cos(\alpha) = \frac{P_x}{\sqrt{(P_x)^2 + (P_y)^2}}; \sin(\alpha) = -\frac{P_y}{\sqrt{(P_x)^2 + (P_y)^2}}; \alpha \in [0; 2\pi]$

$$\cos(\alpha) \sin(\theta_1) + \sin(\alpha) \cos(\theta_1) = \frac{d_3}{\sqrt{(P_x)^2 + (P_y)^2}}$$

Từ ta có:

$$\Rightarrow \sin(\theta_1 + \alpha) = \frac{d_3}{\sqrt{(P_x)^2 + (P_y)^2}} \Leftrightarrow (\theta_1 + \alpha) = \arcsin\left(\frac{d_3}{\sqrt{(P_x)^2 + (P_y)^2}}\right)$$

$$\Rightarrow \theta_1 = \arcsin\left(\frac{d_3}{\sqrt{(P_x)^2 + (P_y)^2}}\right) - \alpha$$

Với:

$$\begin{cases} \alpha = \arcsin\left(\frac{-P_y}{\sqrt{(P_x)^2 + (P_y)^2}}\right), (P_x > 0, P_y > 0) \\ \alpha = \arccos\left(\frac{P_x}{\sqrt{(P_x)^2 + (P_y)^2}}\right), (P_x < 0, P_y < 0) \end{cases}$$

Bước 3: Tính θ_2 .

Ta có:

$$P_z = d_1 + L_3 \sin(\theta_2 + \theta_3) + L_2 \sin(\theta_2)$$

Đặt: $\theta = \theta_2 + \theta_3 \Rightarrow \theta_3 = \theta - \theta_2$. Góc θ ta tự cho trước.

Khi đó phương trình viết lại như sau:

$$P_z = d_1 + L_3 \sin(\theta) + L_2 \sin(\theta_2)$$

$$\Rightarrow \sin(\theta_2) = \frac{P_z - d_1 - L_3 \sin(\theta)}{L_2}$$

$$\Rightarrow \theta_2 = \arcsin\left(\frac{P_z - d_1 - L_3 \sin(\theta)}{L_2}\right)$$

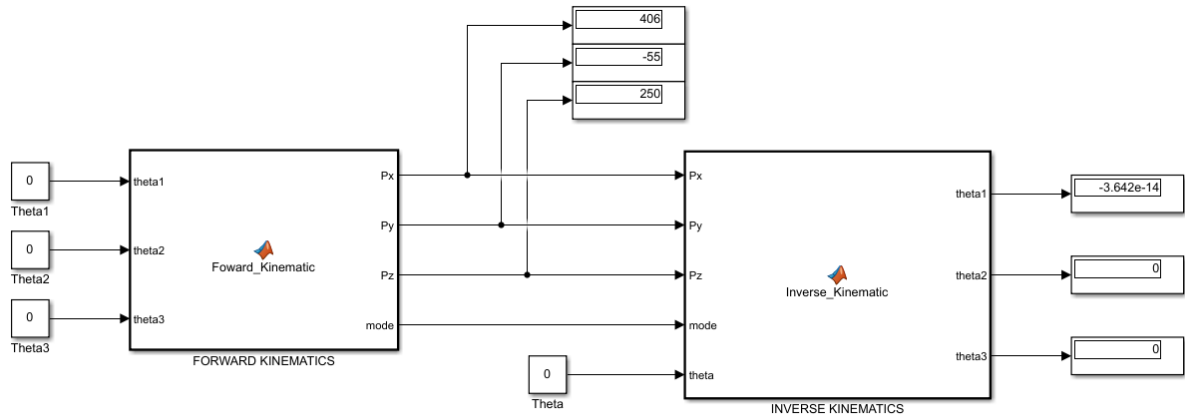
Bước 4: Tính θ_3 .

Từ $\theta = \theta_2 + \theta_3 \Rightarrow \theta_3 = \theta - \theta_2$.

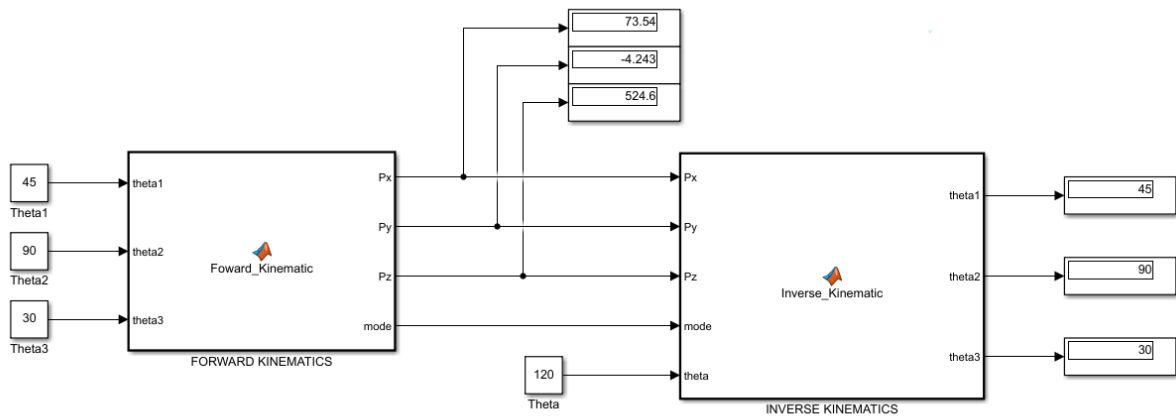
2.3 Kiểm nghiệm kết quả động học trên Matlab Simulink

Nhóm tiến hành kiểm nghiệm kết quả động học nghịch đã tính toán trên Matlab Simulink. Công thức tính toán động học nghịch đã được lập trình bên trong khối Inverse Kinematic. Với thông số đầu vào là tọa độ điểm cuối (x, y, z) và đầu ra là tọa độ 3 góc khớp của cánh tay.

CHƯƠNG 2. TÍNH TOÁN VÀ KIỂM NGHIỆM KẾT QUẢ ĐỘNG HỌC ROBOT 3-DOF



Hình 2.2: Mô phỏng giá trị góc khớp với 3 góc $\theta = 0$



Hình 2.3: Mô phỏng giá trị góc khớp với $\theta_1 = 45$, $\theta_2 = 90$, $\theta_3 = 30$

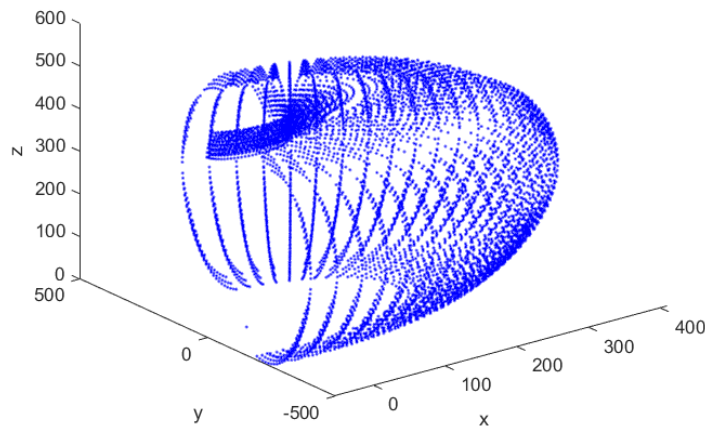
Các thông số vật lý của mô hình được sử dụng giống với kích thước thật của robot được thiết kế nhằm mang lại tính chính xác về thực tế. Kết quả kiểm nghiệm ra đúng.

Chương 3. VẼ KHÔNG GIAN LÀM VIỆC VÀ QUY HOẠCH QUỸ ĐẠO CHO ROBOT

3.1 Xác định không gian làm việc của robot

Không gian làm việc của robot là vùng chứa các điểm, vị trí mà cơ cấu cuối của robot có thể đạt tới. Việc xác định được không gian làm việc của robot, ta có thể đáp ứng được các yêu cầu, nhiệm vụ làm việc trong thực tế của robot, cũng như nâng cấp và cải tiến robot sau này.

Để xác định không gian làm việc của robot, nhóm đã sử dụng kết quả động học thuận và dựa vào các bộ nghiệm của động học nghịch để xây dựng chương trình vẽ không gian làm việc trên phần mềm Matlab. Hình dưới thể hiện kết quả không gian làm việc sau khi chạy chương trình.



Hình 3.1: Không gian làm việc với $\theta_3=0$

Giới hạn của các góc quay:

$$\theta_1 = \pm 90^\circ$$

$$\theta_2 = -45^\circ \div 90^\circ$$

$$\theta_3 = \pm 90^\circ$$

3.2 Quy hoạch quỹ đạo cho robot

Hoạch định quỹ đạo của robot là xác định quỹ đạo chuyển động các biến khớp để điều khiển chuyển động từng khớp và tổng hợp chung thành chuyển động của cả robot theo một quỹ đạo xác định.

Quỹ đạo cần thiết kế đảm bảo phải đi qua những điểm nút chính như điểm đầu và điểm cuối của quỹ đạo. Ngoài các điểm nút chính, thông thường còn có thêm những nút trung gian (khi môi trường có vật cản, hoặc tránh các điểm kì dị ràng buộc về kết cấu cơ khí của robot...).

Quỹ đạo được thiết kế phải tạo nên những chuyển động mượt mà, nhẹ nhàng và hạn chế những chuyển động rung lắc. Do đó, bài toán quy hoạch quỹ đạo chính là việc xây dựng các hàm đa thức theo thời gian liên quan đến vị trí, vận tốc và gia tốc cho robot từ điểm bắt đầu đến điểm đích. Có nhiều phương pháp để xây dựng hàm đa thức quỹ đạo như phương pháp hình thang, đa thức bậc 3, đa thức bậc 5... Ở đây nhóm sử dụng phương pháp đa thức bậc 5 để tạo quỹ đạo chuyển động cho robot. Dưới đây là các bước hình thành hàm quỹ đạo đa thức bậc 5.

Ta có hàm đa thức bậc 5 như sau:

$$x(t) = at^5 + bt^4 + ct^3 + dt^2 + et + f$$

Các điều kiện ban đầu:

$$\begin{cases} x(t_0) = x_0, x(t_f) = x_1 \\ x'(t_0) = 0, x'(t_f) = 0 \\ x''(t_0) = 0, x''(t_f) = 0 \end{cases}$$

$$\begin{cases} x(t_0) = x(0) = f = x_0 \\ x(t_f) = at_f^5 + bt_f^4 + ct_f^3 + dt_f^2 + et_f = x_1 - x_0 \end{cases}$$

Đạo hàm bậc nhất phương trình ta được:

$$\Rightarrow \begin{cases} x'(t_0) = e = 0 \\ x'(t_f) = 5at_f^4 + 4bt_f^3 + 3ct_f^2 + 2dt_f = 0 \end{cases}$$

Đạo hàm bậc hai phương trình ta được:

$$\Rightarrow \begin{cases} x''(t_0) = x''(0) = 2d = 0 \\ x''(t_f) = 20at_f^3 + 12bt_f^2 + 6ct_f = 0 \end{cases}$$

Từ , , ta có hệ phương trình:

CHƯƠNG 3. VỀ KHÔNG GIAN LÀM VIỆC VÀ QUY HOẠCH QUỸ ĐẠO

$$\Rightarrow \begin{cases} at_f^5 + bt_f^4 + ct_f^3 + dt_f^2 + et_f = x_1 \\ 5at_f^4 + 4bt_f^3 + 3ct_f^2 + 2dt_f = 0 \\ 20at_f^3 + 12bt_f^2 + 6ct_f = 0 \end{cases}$$

Giải phương trình ta được:

$$\Rightarrow \begin{cases} a = \frac{6(x_1 - x_0)}{t_f^5} \\ b = \frac{-15(x_1 - x_0)}{t_f^4} \\ c = \frac{10(x_1 - x_0)}{t_f^3} \end{cases}$$

Thay vào ta suy ra phương trình tổng quát hàm đa thức quỹ đạo bậc 5 là:

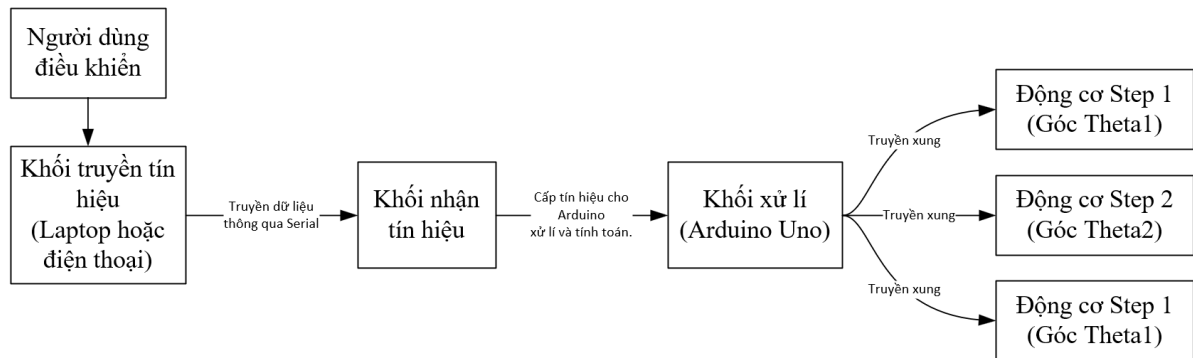
$$x(t) = \frac{6(x_1 - x_0)}{t_f^5} t^5 - \frac{15(x_1 - x_0)}{t_f^4} t^4 + \frac{10(x_1 - x_0)}{t_f^3} t^3$$

Chương 4. CHƯƠNG TRÌNH VÀ GIAO DIỆN ĐIỀU KHIỂN

KHIỂN

4.1 Sơ đồ khối điều khiển cánh tay robot 3 bậc tự do

Điều khiển cánh tay robot 3 bậc tự do dựa trên sơ đồ Hình 4.1. Trong đó khối giao diện có chức năng tùy chỉnh các thông số robot và các tùy chọn điều khiển robot từ đó gửi bộ góc qua Arduino thông giao tiếp serial (UART). Khối vi điều khiển có chức năng nhận bộ góc, xử lý, tính toán sau đó tạo ra các xung để cung cấp điều khiển từng step motor. Từng động cơ step sẽ gắn vào mạch Arduino CNC Shield V3 nhận tín hiệu trả về từ Arduino và thực hiện chuyển động trong robot.



Hình 4.1: Sơ đồ khối điều khiển

4.2 Giao diện điều khiển

Để thuận tiện cho việc điều khiển cũng như theo dõi hoạt động của robot, các giá trị biến khớp, tọa độ công tác. Nhóm đã tạo giao diện điều khiển trên phần mềm Sublime Text bằng ngôn ngữ Python.

CHƯƠNG 4. CHƯƠNG TRÌNH VÀ GIAO DIỆN ĐIỀU KHIỂN



Hình 4.2: Giao diện mở đầu

Giao diện giới thiệu gồm các thông tin: logo trường, tên ngành, khoa, tên môn học, tên đề tài, tên gvhd, tên svth và nút bấm để chuyển đến giao diện điều khiển

The screenshot shows a software window titled "CONTROL INTERFACE". It is divided into two main sections: "FORWARD KINEMATIC" and "INVERSE KINEMATIC".

FORWARD KINEMATIC: This section contains three rows for joint angles. Each row has a slider, a numerical input field, and a "Set" button. The first row is for "Theta1" with a value of 0.0 and a "Set Theta1" button. The second row is for "Theta2" with a value of 0.0 and a "Set Theta2" button. The third row is for "Theta3" with a value of 0.0 and a "Set Theta3" button. Below these, there are three input fields for Cartesian coordinates: "Px" (406.0), "Py" (-55.0), and "Pz" (250.0). To the right of these fields is a green "Reset" button.

INVERSE KINEMATIC: This section contains input fields for the Cartesian coordinates and the joint angles. There are three rows for "Px", "Py", and "Pz", each followed by a "Theta" input field. To the right of these fields is a green "Solve" button. At the bottom left is a pink "Set Home" button, and at the bottom right is a red "Stop" button.

Hình 4.3: Giao diện điều khiển

Chức năng giao diện:

- Nút Set Home: chạy mô hình robot ở vị trí Set Home.
- Nút STOP: dừng chạy mô hình robot.
- Nút RESET: đặt lại giá trị của các góc theta về 0.
- Nút SOLVE: tính toán giá trị góc theta 1,2,3 từ vị trí Px, Py, Pz cho trước.
- Nhập góc mô phỏng bằng 2 cách: kéo thanh trượt hoặc nhập trực tiếp.
- Động học nghịch: điều khiển đồng bộ các khớp.

Chương 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 Kết luận

Đề tài “Thiết kế và điều khiển cánh tay robot 3 bậc tự do” là đề tài mang tính ứng dụng thực tế. Mô hình đáp ứng được các yêu cầu đề ra và hoạt động tốt trong một môi trường nhất định.

Mô hình là tiền đề để ứng dụng các công nghệ thông minh vào thực tiễn, đáp ứng yêu cầu năng suất chất lượng và độ chính xác trong sản xuất. Giảm chi phí thuê nhân công và những rủi ro ảnh hưởng đến tính mạng con người trong quá trình sản xuất.

Thông qua quá trình nghiên cứu cơ sở lý thuyết, mô phỏng và xây dựng mô hình, đồ án đạt được kết quả như sau:

- + Thiết kế mô hình sử dụng phần mềm Solidworks
- + Tính toán, mô phỏng kiểm nghiệm kết quả động học của robot trên phần mềm Matlab Simulink.
- + Xây dựng mô hình thực tế với tỉ lệ 1:1 so với thiết kế.
- + Lập trình điều khiển cấp xung động cơ bước để robot hoạt động trên phần mềm Arduino.
- + Thiết kế giao diện điều khiển với nhiều tính năng trên phần mềm Sublime Text bằng ngôn ngữ Python.

5.2 Hướng phát triển

Cải tiến mô hình và động cơ để có thể đáp ứng được các nhiệm vụ phức tạp với độ chính xác cao.

Tăng số bậc tự do cho robot để tăng không gian làm việc và robot sẽ hoạt động linh hoạt hơn.

Áp dụng các thuật toán điều khiển để tăng tốc độ di chuyển của robot trong ứng dụng gấp và đặt vật. Và hạn chế các tác động ngoại vi như nhiễu, rung lắc gây ra trong quá trình hoạt động.

Lắp đặt cảm biến ở các vị trí khớp và encoder để phản hồi các tín hiệu đáp ứng từ động cơ từ đó hiệu chỉnh tín hiệu điều khiển hợp lý.

PHỤ LỤC

Ứng dụng xử lý ảnh để thực hiện các nhiệm vụ phức tạp và nâng cao sự thông minh của robot.

Chương trình động học thuận:

```
function [Px, Py, Pz, mode] =  
Forward_Kinematic(theta1,theta2,theta3)  
    %Program By Nguyen Duc Manh  
    %% Main Parametter  
    L1 = 114;  
    L2 = 162;  
    L3 = 130;  
    d1 = 250;  
    d3 = 55;  
    %%main program  
    Px = L1*cosd(theta1) + d3*sind(theta1) +  
    L2*cosd(theta1)*cosd(theta2) +  
    L3*cosd(theta1)*cosd(theta2)*cosd(theta3) -  
    L3*cosd(theta1)*sind(theta2)*sind(theta3);  
    Py = L1*sind(theta1) - d3*cosd(theta1) +  
    L2*cosd(theta2)*sind(theta1) +  
    L3*cosd(theta2)*cosd(theta3)*sind(theta1) -  
    L3*sind(theta1)*sind(theta2)*sind(theta3);  
    Pz = d1 + L3*sind(theta2 + theta3) + L2*sind(theta2);  
    if (theta1 > 0)  
        mode = 0;  
    else  
        mode = 1;  
    end  
end
```

Chương trình động học nghịch:

```
function [theta1, theta2, theta3] = Inverse_Kinematic(Px,  
Py, Pz, mode,theta)  
    %Program By Nguyen Duc Manh  
    %% Main Parametter  
    L1 = 114;  
    L2 = 162;  
    L3 = 130;  
    d1 = 250;  
    d3 = 55;  
  
    %% Main program  
    if (mode == 0)  
        t = max(min(-Py/sqrt((Px)^2+(Py)^2),1),-1);  
        a1 = asind(t);  
    else  
        t = max(min(Px/sqrt((Px)^2+(Py)^2),1),-1);  
        a1 = acosd(t);  
    end
```

PHỤ LỤC

```
end
% Solve Theta_1
t1 = max(min(d3/sqrt((Px)^2+(Py)^2),1),-1);
theta1 = asind(t1)-a1;

t2 = max(min((Pz-d1-L3*sind(theta))/L2,1),-1);
theta2 = asind(t2);

theta3 = theta - theta2;
end
```

Chương trình vẽ không gian làm việc cho cánh tay robot 3 bậc tự do:

```
clc;
clear all;
close all;
%% Thông số robot
L1 = 114;
L2 = 162;
L3 = 130;
d1 = 250;
d3 = 55;
theta=0;
%% Không gian làm việc
i=0;
i1=0;
i2=0;
i=i+1;
i1=i1+1;
i2=i2+1;
for theta1 =-90:5:90
    for theta2=-45:5:90
        if -45<= theta2&& theta2< -30
            for theta3 = -60:5: 30
                i=i+1;
                Px = L1*cosd(theta1) + d3*sind(theta1) +
L2*cosd(theta1)*cosd(theta2) +
L3*cosd(theta1)*cosd(theta2)*cosd(theta3) -
L3*cosd(theta1)*sind(theta2)*sind(theta3);
                Py = L1*sind(theta1) - d3*cosd(theta1) +
L2*cosd(theta2)*sind(theta1) +
L3*cosd(theta2)*cosd(theta3)*sind(theta1) -
L3*sind(theta1)*sind(theta2)*sind(theta3);
                Pz = d1 + L3*sind(theta2 + theta3) +
L2*sind(theta2);
                workspace_full(:,i)=[Px;Py;Pz];
            end
        elseif -30<= theta2&& theta2< 0
            for theta3 = -30:5: 0
```

PHỤ LỤC

```
        i=i+1;
        Px = L1*cosd(theta1) + d3*sind(theta1) +
L2*cosd(theta1)*cosd(theta2) +
L3*cosd(theta1)*cosd(theta2)*cosd(theta3) -
L3*cosd(theta1)*sind(theta2)*sind(theta3);
        Py = L1*sind(theta1) - d3*cosd(theta1) +
L2*cosd(theta2)*sind(theta1) +
L3*cosd(theta2)*cosd(theta3)*sind(theta1) -
L3*sind(theta1)*sind(theta2)*sind(theta3);
        Pz = d1 + L3*sind(theta2 + theta3) +
L2*sind(theta2);
        workspace_full(:,i)=[Px;Py;Pz];
    end
elseif 0<= theta2&& theta2< 30
    for theta3 = 0:5: 30
        i=i+1;
        Px = L1*cosd(theta1) + d3*sind(theta1) +
L2*cosd(theta1)*cosd(theta2) +
L3*cosd(theta1)*cosd(theta2)*cosd(theta3) -
L3*cosd(theta1)*sind(theta2)*sind(theta3);
        Py = L1*sind(theta1) - d3*cosd(theta1) +
L2*cosd(theta2)*sind(theta1) +
L3*cosd(theta2)*cosd(theta3)*sind(theta1) -
L3*sind(theta1)*sind(theta2)*sind(theta3);
        Pz = d1 + L3*sind(theta2 + theta3) +
L2*sind(theta2);
        workspace_full(:,i)=[Px;Py;Pz];
    end
elseif 30<= theta2&& theta2< 60
    for theta3 = 30:5: 60
        i=i+1;
        Px = L1*cosd(theta1) + d3*sind(theta1) +
L2*cosd(theta1)*cosd(theta2) +
L3*cosd(theta1)*cosd(theta2)*cosd(theta3) -
L3*cosd(theta1)*sind(theta2)*sind(theta3);
        Py = L1*sind(theta1) - d3*cosd(theta1) +
L2*cosd(theta2)*sind(theta1) +
L3*cosd(theta2)*cosd(theta3)*sind(theta1) -
L3*sind(theta1)*sind(theta2)*sind(theta3);
        Pz = d1 + L3*sind(theta2 + theta3) +
L2*sind(theta2);
        workspace_full(:,i)=[Px;Py;Pz];
    end
elseif 60<= theta2&& theta2< 90
    for theta3 = 60:5: 90
        i=i+1;
        Px = L1*cosd(theta1) + d3*sind(theta1) +
L2*cosd(theta1)*cosd(theta2) +
```

PHỤ LỤC

```
L3*cosd(theta1)*cosd(theta2)*cosd(theta3) -
L3*cosd(theta1)*sind(theta2)*sind(theta3);
    Py = L1*sind(theta1) - d3*cosd(theta1) +
L2*cosd(theta2)*sind(theta1) +
L3*cosd(theta2)*cosd(theta3)*sind(theta1) -
L3*sind(theta1)*sind(theta2)*sind(theta3);
    Pz = d1 + L3*sind(theta2 + theta3) +
L2*sind(theta2);
    workspace_full(:,i)=[Px;Py;Pz];
    end

    end

    end

end
for theta1 =0:5:90
    for theta2=-45:5:90
        if -45<= theta2&& theta2< -30
            for theta3 = -60:5: 30
                i=i+1;
                i1=i1+1;
                i2=i2+1;

                Px = L1*cosd(theta1) + d3*sind(theta1) +
L2*cosd(theta1)*cosd(theta2) +
L3*cosd(theta1)*cosd(theta2)*cosd(theta3) -
L3*cosd(theta1)*sind(theta2)*sind(theta3);
                Py = L1*sind(theta1) - d3*cosd(theta1) +
L2*cosd(theta2)*sind(theta1) +
L3*cosd(theta2)*cosd(theta3)*sind(theta1) -
L3*sind(theta1)*sind(theta2)*sind(theta3);
                Pz = d1 + L3*sind(theta2 + theta3) +
L2*sind(theta2);
                workspace_full(:,i)=[Px;Py;Pz];
            end
        elseif -30<= theta2&& theta2< 0
            for theta3 = -30:5: 0
                i=i+1;
                Px = L1*cosd(theta1) + d3*sind(theta1) +
L2*cosd(theta1)*cosd(theta2) +
L3*cosd(theta1)*cosd(theta2)*cosd(theta3) -
L3*cosd(theta1)*sind(theta2)*sind(theta3);
                Py = L1*sind(theta1) - d3*cosd(theta1) +
L2*cosd(theta2)*sind(theta1) +
L3*cosd(theta2)*cosd(theta3)*sind(theta1) -
L3*sind(theta1)*sind(theta2)*sind(theta3);
```

PHỤ LỤC

```
Pz = d1 + L3*sind(theta2 + theta3) +
L2*sind(theta2);
workspace_full(:,i)=[Px;Py;Pz];
end
elseif 0<= theta2&& theta2< 30
for theta3 = 0:5: 30
i=i+1;
Px = L1*cosd(theta1) + d3*sind(theta1) +
L2*cosd(theta1)*cosd(theta2) +
L3*cosd(theta1)*cosd(theta2)*cosd(theta3) -
L3*cosd(theta1)*sind(theta2)*sind(theta3);
Py = L1*sind(theta1) - d3*cosd(theta1) +
L2*cosd(theta2)*sind(theta1) +
L3*cosd(theta2)*cosd(theta3)*sind(theta1) -
L3*sind(theta1)*sind(theta2)*sind(theta3);
Pz = d1 + L3*sind(theta2 + theta3) +
L2*sind(theta2);
workspace_full(:,i)=[Px;Py;Pz];
end
elseif 30<= theta2&& theta2< 60
for theta3 = 30:5: 60
i=i+1;
Px = L1*cosd(theta1) + d3*sind(theta1) +
L2*cosd(theta1)*cosd(theta2) +
L3*cosd(theta1)*cosd(theta2)*cosd(theta3) -
L3*cosd(theta1)*sind(theta2)*sind(theta3);
Py = L1*sind(theta1) - d3*cosd(theta1) +
L2*cosd(theta2)*sind(theta1) +
L3*cosd(theta2)*cosd(theta3)*sind(theta1) -
L3*sind(theta1)*sind(theta2)*sind(theta3);
Pz = d1 + L3*sind(theta2 + theta3) +
L2*sind(theta2);
workspace_full(:,i)=[Px;Py;Pz];
end
elseif 60<= theta2&& theta2< 90
for theta3 = 60:5: 90
i=i+1;
Px = L1*cosd(theta1) + d3*sind(theta1) +
L2*cosd(theta1)*cosd(theta2) +
L3*cosd(theta1)*cosd(theta2)*cosd(theta3) -
L3*cosd(theta1)*sind(theta2)*sind(theta3);
Py = L1*sind(theta1) - d3*cosd(theta1) +
L2*cosd(theta2)*sind(theta1) +
L3*cosd(theta2)*cosd(theta3)*sind(theta1) -
L3*sind(theta1)*sind(theta2)*sind(theta3);
Pz = d1 + L3*sind(theta2 + theta3) +
L2*sind(theta2);
workspace_full(:,i)=[Px;Py;Pz];
```

PHỤ LỤC

```
                end
            end
        end
    end

    figure(1);
    grid on
    plot3(workspace_full(1,:),workspace_full(2,:),workspace_full(3,:),'.b','MarkerSize',3);

    xlabel('x');
    ylabel('y') ;
    zlabel('z')
```