

Cơ sở kỹ thuật và lập trình Rô bôt

Giáo viên: Đặng Hà Dũng,

Đánh giá: thi tự luận 75, báo cáo thực hành, điểm danh và bài tập trên lớp.

Nội dung: 2 nội dung chính: Cơ sở kỹ thuật của Robot và Lập trình cơ bản cho Robot.

- Tài liệu tham khảo: Robot Công nghiệp, GS. Nguyễn Thiện Phúc.

Handbook of Robotic.

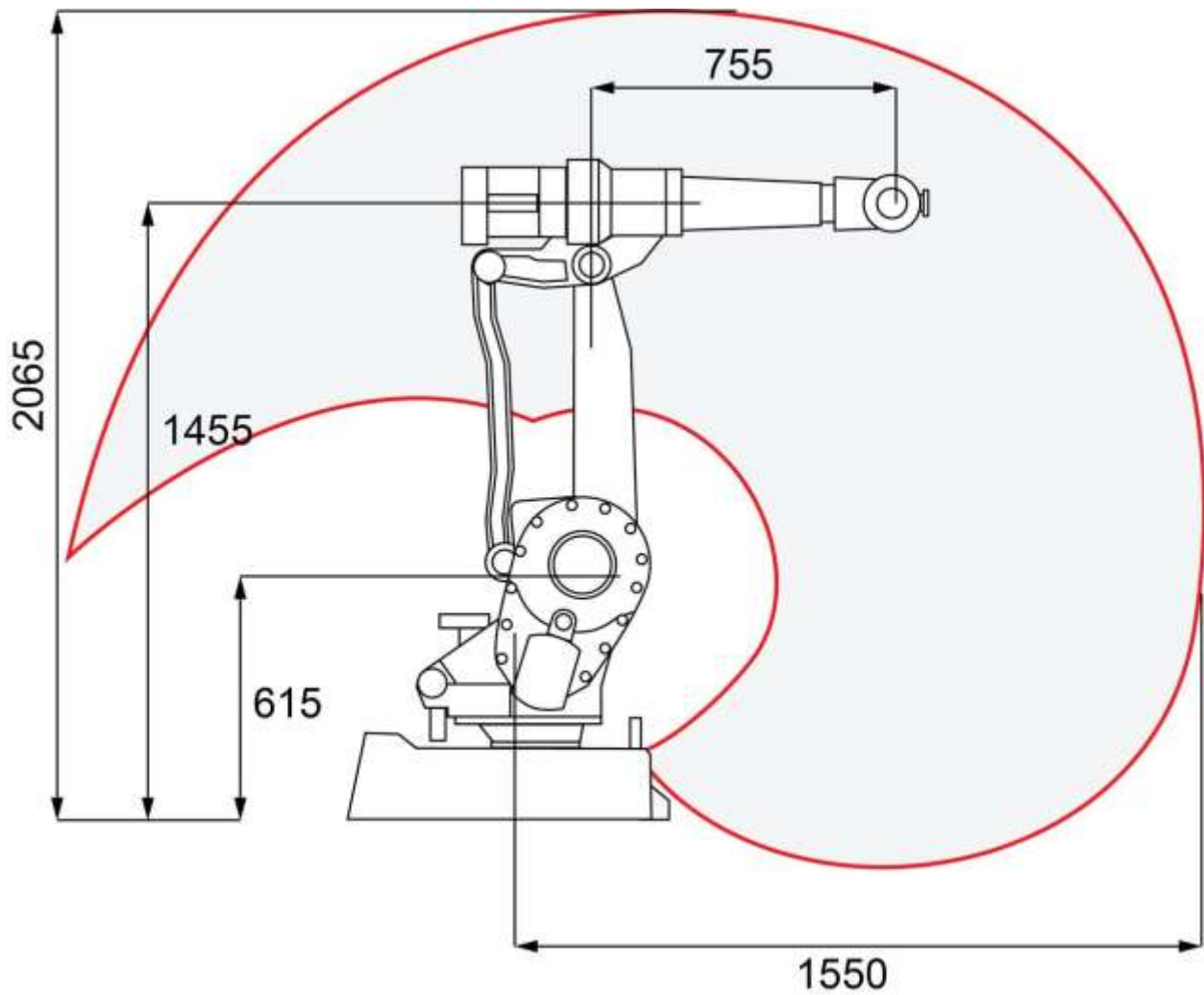
- Lập trình máy tính: Matlab, lập trình vi điều khiển: arduino, stm32.

- Đối tượng Robot là Robot công nghiệp, điều khiển vị trí, tốc độ. Có sử dụng công cụ lý thuyết điều khiển.

Chương 1. Tổng quan về kỹ thuật và lập trình Robot

1. Động học thuận và ngược của Robot (ôn tập)

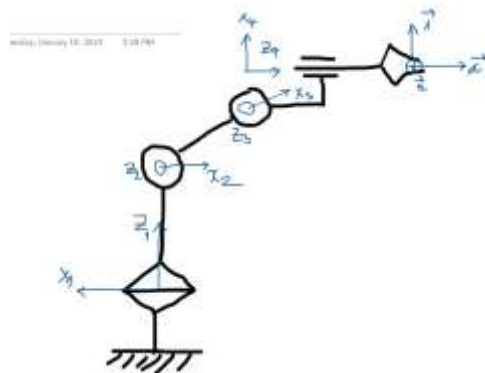




Buổi 2.

Giải bài toán động học thuận và ngược Robot

Với cơ cấu Robot 4 bậc tự do đã xác định được hệ tọa độ:



Từ các hệ tọa độ như trên, ta có bảng tham số DH cho Robot như sau:

	α_i	a_i	d_i	θ_i
Khớp 1				
Khớp 2				
Khớp 3				
Khớp 4				

Bảng thứ 2:

	θ_i	d_i	a_i	α_i
Khớp 1	θ_1^*	d_1	0	90°
Khớp 2	θ_2^*	0	a_2	0
Khớp 3	θ_3^*	0	a_3	90°
Khớp 4	-90°	d_4^*	0	0

- Bài toán động học thuận:

Tìm được vị trí và hướng của cơ cấu chấp hành cuối

Ma trận tổng quát biến đổi cho 1 khớp là:

$$T_1^2 = R(Oz, \theta) * T(0,0,d) * T(a,0,0) * R(Ox, \alpha)$$

$$= \begin{pmatrix} \cos(\theta) & -\sin(\theta)\cos(\alpha) & \sin(\theta)\sin(\alpha) & \cos(\theta) \\ \sin(\theta) & \cos(\theta)\cos(\alpha) & -\cos(\theta)\sin(\alpha) & \sin(\theta) \\ 0 & \sin(\alpha) & \cos(\alpha) & d \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

syms th1 th2 th3 d4

d1 = 0.3, a2 = 0.1, a3 = 0,

L(1) = Link([0 d1 0 pi/2 0])

L(2) = Link([0 0 a2 0 0])

L(3) = Link([0 0 a3 pi/2 0])

L(4) = Link([-pi/2 0 0 0 1])

myRobot_1 = SerialLink(L)

for th1 = 0:0.5:pi/2

for th2 = 0:0.5:pi/2

for th3 = 0:0.5:pi/2

```

    for d4 = 0.05:0.01:0.1
        myRobot_1.plot([th1 th2 th3 d4],'workspace',[-1 1 -1 1 -1 1])
        pause(0.5)
    end
end
end
end
end

```

Buổi 3.

- Phương pháp biến đổi DH có sửa đổi (modified DH)

Phương pháp xác định hệ tọa độ gắn với các khớp không có sự khác biệt.

Khi thực hiện biến đổi hệ tọa độ, thì trình tự thực hiện có sự thay đổi:

- + Quay xung quanh trục Ox góc α để cho 2 trục Oz cùng phương
- + Tịnh tiến dọc trục Ox hiện tại khoảng a để cho 2 trục Oz trùng nhau
- + Quay xung quanh trục Oz hiện tại góc θ để cho 2 trục Ox trùng phương
- + Tịnh tiến dọc trục Oz hiện tại khoảng d để cho 2 gốc tọa độ trùng nhau.

Từ trình tự biến đổi ta có thể xác định được ma trận biểu diễn trong trường hợp này:

$$T_1^2 = R(Ox, \alpha) * T(a, 0, 0) * R(Oz, \theta) * T(0, 0, d)$$

Chương trình trên Matlab để thực hiện tính ma trận biểu diễn:

% Khai báo các biến ký tự:

```
>>syms alpha a theta d
```

% Xây dựng các hàm để tính các ma trận: Rox(alpha), T(a,b,c), Roz(theta)

% Tạo function: [R] = Rox(a)

```
function [R] = Roz(c)
```

```
c_rad = (c*pi)/180;
```

```
R = [cos(c_rad),-sin(c_rad),0;sin(c_rad),cos(c_rad),0;0,0,1];
```

End

Sau khi có các hàm biến đổi cơ bản thuần nhất, thử tính tính ma trận biểu diễn trên Matlab:

```
>>syms alpha, a, gamma, d
```

```
>>T21 = Rox(alpha)*T(a,0,0)*Roz(gamma)*T(0,0,d)
```

Kết quả:

```
[      cos(ga),      -sin(ga),      0,      a]
[cos(al)*sin(ga), cos((pi*al)/180)*cos((pi*ga)/180), -sin((pi*al)/180), -d*sin(al)]
[sin((pi*al)/180)*sin((pi*ga)/180),      cos((pi*ga)/180)*sin((pi*al)/180),
cos((pi*al)/180), d*cos((pi*al)/180)]
[      0,      0,      0,      1]
```

- Sau khi đã có biểu diễn DH dưới dạng tiêu chuẩn hoặc biến đổi, thì ta có thể sử dụng công cụ là thư viện Robotics của Matlab để thực hiện mô phỏng vận hành của Robot. (Yêu cầu: sinh viên cài thư viện Robotics của Peter Corke vào Matlab)

Một số lệnh thực hiện để xây dựng mô hình Robot:

%Tạo ra khớp quay: sử dụng lệnh Resolute, ví dụ với cơ cấu Robot RRRP

```
>>L(1) = Resolute('d', 0.3, 'a', 0, 'alpha',pi/2);
```

```
>>L(2) = Resolute('d',0,'a',0.1,'alpha',0);
```

```
>>L(3) = Resolute('d',0,'a',0,'alpha'pi/2);
```

%Khớp tịnh tiến, ta dùng lệnh Prismatic

```
>>L(4) = Prismatic('theta',-pi/2,'a',0,'alpha',0);
```

%Để tạo ra Robot ta dùng lệnh SerialLink(L);

```
>>my_RRRP = SerialLink(L); % Phương thức để tạo ra đối tượng Robot nối tiếp
```

% Sau khi đã tạo ra đối tượng Robot, ta có thể sử dụng các phương thức như plot để vẽ hình nổi robot, phương thức tính động học thuận fkine, phương thức tính động học ngược ikine.

Cách 2: Sử dụng phương thức Link. Để sử dụng phương thức tạo của Link thì ta cần chú ý: có thể tạo ra từng khớp của Robot, hoặc có thể tạo ra toàn bộ Robot.

Ví dụ trường hợp để tạo ra từng khớp Robot:

```
>>L(1) = Link([0 0.3 0 pi/2]);
```

```
>>L(2)= Link([0,0,0.1,0,0]);
```

```
>>L(3)= Link([0,0,0,pi/2]);
```

```
>>L(4) = Link([-pi/2,0,0,0,1]);
```

```
>>my_RRRP = SerialLink(L);
```

%Để vẽ hình robot ra một cách trực quan ta có thể sử dụng phương thức plot, tham số của plot là tập các giá trị của các biến khớp.

```
>>my_RRRP.plot([0,0,0,0.1], 'space', [-1 1 -1 1 -1 1]);
```

%Lập trình để tạo ra chuyển động trực quan của robot:

Khớp 1 biến đổi trong khoảng:

```
th1= linspace(0,pi/2,20);
```

```
th2 = linspace(-pi/3,pi/3,20);
```

```
th3= linspace(-pi/4,pi/4,20);
```

```
d4 = linspace(0.1,0.2,20);
```

```
for i=1:20
```

```
    my_RRRP.plot([th1(i),th2(i),th3(i),d4(i)]);
```

```
    pause(0.2);
```

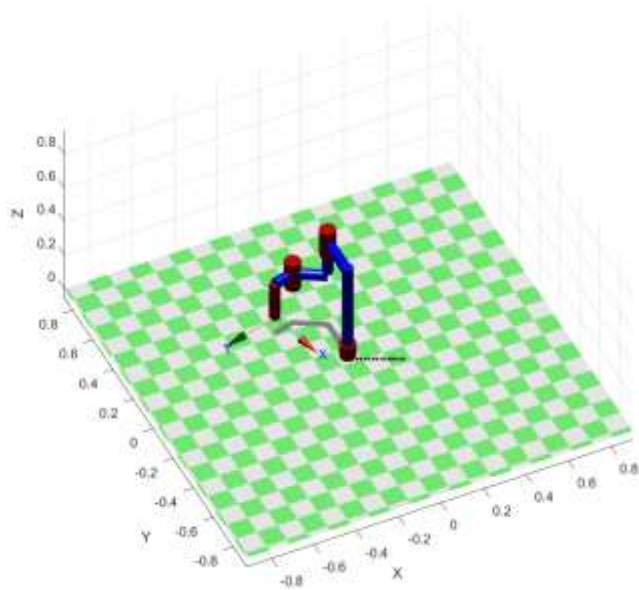
```
end
```

Buổi 4 (24/1)

Lập trình mô phỏng quỹ đạo chuyển động của cơ cấu chấp hành cuối Robot

- Sử dụng mô hình Robot SCARA 4 bậc tự do:

- Đưa được bàn kẹp di chuyển theo quỹ đạo định trước: đường thẳng, đường tròn, parabol,... Để thực hiện ta phải giải quyết bài toán động học ngược thông qua bài toán động học thuận:



Bài toán động học thuận của cơ cấu robot SCARA:

$$E = T_1^2(\theta_1^*) * T_2^3(\theta_2^*) * T_3^4(\theta_3^*) * T_4^E(d_4^*)$$

Trong đó:

$$T_1^2(\theta_1^*) = \begin{pmatrix} \cos(\theta_1^*) & -\sin(\theta_1^*) & 0 & a_1 \cos(\theta_1^*) \\ \sin(\theta_1^*) & \cos(\theta_1^*) & 0 & a_1 \sin(\theta_1^*) \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$T_2^3(\theta_2^*) = \begin{pmatrix} \cos(\theta_2^*) & -\sin(\theta_2^*) & 0 & a_2 \cos(\theta_2^*) \\ \sin(\theta_2^*) & \cos(\theta_2^*) & 0 & a_2 \sin(\theta_2^*) \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$T_3^4(\theta_3^*) = \begin{pmatrix} \cos(\theta_3^*) & \sin(\theta_3^*) & 0 & a_3 \cos(\theta_3^*) \\ \sin(\theta_3^*) & -\cos(\theta_3^*) & 0 & a_3 \sin(\theta_3^*) \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$T_4^E(d_4^*) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & d_4^* \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Vậy: (Lấy: $d_1 = 0.5$; $a_1 = 0.2$; $d_2 = 0.1$; $a_2 = 0.1$; $d_3 = 0.1$; $a_3 = 0.05$;))

$$E = [-\cos(\text{te3}) * (\cos(\text{te1}) * \sin(\text{te2}) + \cos(\text{te2}) * \sin(\text{te1})) - \sin(\text{te3}) * (\cos(\text{te1}) * \cos(\text{te2}) - \sin(\text{te1}) * \sin(\text{te2})), \cos(\text{te3}) * (\cos(\text{te1}) * \cos(\text{te2}) - \sin(\text{te1}) * \sin(\text{te2})) - \sin(\text{te3}) * (\cos(\text{te1}) * \sin(\text{te2}) + \cos(\text{te2}) * \sin(\text{te1})), 0, \cos(\text{te1})/5 + (\cos(\text{te1}) * \cos(\text{te2}))/10 - (\sin(\text{te1}) * \sin(\text{te2}))/10 + (\cos(\text{te3}) * (\cos(\text{te1}) * \cos(\text{te2}) - \sin(\text{te1}) * \sin(\text{te2}))) / 20 - (\sin(\text{te3}) * (\cos(\text{te1}) * \sin(\text{te2}) + \cos(\text{te2}) * \sin(\text{te1}))) / 20]$$

$$[\cos(\text{te3}) * (\cos(\text{te1}) * \cos(\text{te2}) - \sin(\text{te1}) * \sin(\text{te2})) - \sin(\text{te3}) * (\cos(\text{te1}) * \sin(\text{te2}) + \cos(\text{te2}) * \sin(\text{te1})), \cos(\text{te3}) * (\cos(\text{te1}) * \sin(\text{te2}) + \cos(\text{te2}) * \sin(\text{te1})) + \sin(\text{te3}) * (\cos(\text{te1}) * \cos(\text{te2}) - \sin(\text{te1}) * \sin(\text{te2})), 0, \sin(\text{te1})/5 + (\cos(\text{te1}) * \sin(\text{te2}))/10 + (\cos(\text{te2}) * \sin(\text{te1}))/10 + (\cos(\text{te3}) * (\cos(\text{te1}) * \sin(\text{te2}) + \cos(\text{te2}) * \sin(\text{te1}))) / 20 + (\sin(\text{te3}) * (\cos(\text{te1}) * \cos(\text{te2}) - \sin(\text{te1}) * \sin(\text{te2}))) / 20]$$

$$[\begin{matrix} 0, \\ 0, \\ 3/5 - d4 \end{matrix} \begin{matrix} 0, \\ -1, \\ \end{matrix}]$$

$$[\begin{matrix} 0, \\ 0, \\ 1 \end{matrix} \begin{matrix} 0, \\ 0, \\ \end{matrix}]$$

Ta có:

$$E(1,1) = -\sin(\theta_1^* + \theta_2^* + \theta_3^*) = -s_{123} = n_x;$$

$$E(1,2) = c_{123} = s_x;$$

$$E(1,3) = 0;$$

$$E(1,4) = \frac{c_1}{5} + \frac{c_{12}}{10} + \frac{c_{123}}{20} = p_x;$$

$$E(2,1) = c_{123} = n_y;$$

$$E(2,2) = s_{123} = s_y;$$

$$E(2,3) = 0;$$

$$E(2,4) = \frac{s_1}{5} + \frac{s_{12}}{10} + \frac{s_{123}}{20} = p_y;$$

$$E(3,1) = 0;$$

$$E(3,2) = 0;$$

$$E(3,3) = -1;$$

$$E(3,4) = \frac{3}{5} - d_4^* = p_z$$

Giả sử ma trận biểu diễn của cơ cấu chấp hành cuối là:

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

Ta có:

$$\theta_1 + \theta_2 + \theta_3 = \text{atan2}(s_{123}, c_{123}) = \text{atan2}(s_y, n_y)$$

$$\frac{c_1}{5} + \frac{c_{12}}{10} + \frac{c_{123}}{20} = p_x \rightarrow \frac{c_1}{5} + \frac{c_{12}}{10} = p_x - \frac{s_x}{20}$$

$$\frac{s_1}{5} + \frac{s_{12}}{10} + \frac{s_{123}}{20} = p_y \rightarrow \frac{s_1}{5} + \frac{s_{12}}{10} = p_y - \frac{s_y}{20}$$

Tùy chọn góc $\theta_1 = \alpha \rightarrow$

$$\theta_2 = \text{atan2}\left(p_y - \frac{s_y}{20} - \left(\frac{\sin(\alpha)}{5}\right), p_x - \frac{s_x}{20} - \left(\frac{\cos(\alpha)}{5}\right)\right) - \alpha$$

Phương trình đường thẳng theo tham số thời gian t:

$$p_x =$$

$$p_y =$$

Phương trình đường tròn theo thời gian t:

$$p_x = A * \sin(b * t)$$

$$p_y = A * \cos(b * t)$$

Buổi 5(21/2)

Mục tiêu: xây dựng chương trình mô phỏng quá trình thực hiện chuyển động của Robot SCARA để đưa điểm chấp hành cuối theo một số quỹ đạo:

a. Đường tròn:

Phương trình tham số của đường tròn:

$$p_x = a + r * \sin(t)$$

$$p_y = b + r * \cos(t)$$

Tọa độ p_z được chọn tùy vào cao độ của mặt phẳng làm việc.

Chọn hướng của cơ cấu chấp hành cuối thỏa mãn:

$$\vec{s} = [1,0,0]; \vec{n} = [0,1,0].$$

Mặt phẳng làm việc (quyết định bởi kích thước của Robot):

Chiều cao làm việc: $z = 0 - 0.4(\text{m})$, tầm với: $0 - 0.35(\text{m})$

Chọn $p_z = 0.1$ là chiều cao của mặt phẳng làm việc

Chọn $p_z = 0.3$ là chiều cao của điểm chuẩn.

Vậy ma trận biểu diễn của cơ cấu chấp hành cuối:

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

p_z nhận giá trị 0.1 hoặc 0.3

p_x, p_y được tính theo công thức đường tròn: tâm (0,0), $r = 0.3$

Vị trí ban đầu tương ứng với: $\theta_1 = \theta_2 = \theta_3 = 0, z = 0.3$

Dựa trên kết quả tính toán động học ngược, ta có:

$$\theta_1 + \theta_2 + \theta_3 = \text{atan2}(s_{123}, c_{123}) = \text{atan2}(0,1) = 0$$

$$\text{Chọn: } \theta_1 = 30^\circ \rightarrow \theta_2 + \theta_3 = -30^\circ$$

$$\theta_2 = \text{atan2}\left(p_y - \left(\frac{\sin(30^\circ)}{5}\right), p_x - \frac{1}{20} - \left(\frac{\cos(30^\circ)}{5}\right)\right) - 30^\circ$$

$$\theta_3 = -30^\circ - \theta_2$$

$$d_4 = 0.6 - p_z$$

Chương trình Matlab

% Xây dựng mô hình Robot:

```
L(1)=Link([0,d1,a1,0,0]);
L(2)=Link([0,-d2,a2,0,0]);
L(3)=Link([0,0,a3,pi,0]);
```

```

L(4)=Link([-pi/2,0,0,0,1]);
my_SCARA1 = SerialLink(L);

%Đặt Robot ở vị trí ban đầu:

My_SCARA1.plot([0,0,0,0.3], 'workspace', [-1 1 -1 1 0 1]
Pause(0.5);
%Đưa Robot đến vị trí bắt đầu vẽ đường tròn:
px = 0.3*sin(0); py = 0.3*cos(0);
Pz = 0.3;
The1 = deg2rad(30);
The2 = atan2(0.3*cos(0)-sin(The1)/5,0.3*sin(0)-1/20 - cos(The1)/5)-The1;
The3 = -The1 - The2;
D4 = 0.6 - pz;
My_SCARA1.plot([The1,The2,The3,d4], 'workspace', [-1 1 -1 1 0 1])
%Đánh dấu vị trí điểm làm việc trên quỹ đạo:
Plot3(px,py,pz,'*');hold on;
Pause(0.5);
%Đưa Robot đến điểm bắt đầu gia công:
Pz = 0.1;
D4 = 0.6 - pz;
My_SCARA1.plot([The1,The2,The3,d4], 'workspace', [-1 1 -1 1 0 1])
Plot3(px,py,pz,'*');
Pause(0.5);
%Thực hiện vẽ hình tròn:
%Khai báo biến thời gian t= 0 - 2pi, với khoảng chia 0.1
For t = 0:0.1:2*pi
    Px = 0.3*sin(t);py = 0.3*cos(t);
    The2 = atan2(py -sin(The1)/5,px-1/20 - cos(The1)/5)-The1;
    The3 = -The1 - The2;
    My_SCARA1.plot([The1,The2,The3,d4], 'workspace', [-1 1 -1 1 0 1])
    Plot3(px,py,pz,'*');
    Pause(0.5);
End
%Đặt Robot ở vị trí ban đầu:

My_SCARA1.plot([0,0,0,0.3], 'workspace', [-1 1 -1 1 0 1]
Pause(0.5);hold off;

```

Bài tập về nhà:

Xây dựng mô hình Robot Scara 3 bậc tự do, thực hiện viết chương trình điều khiển Robot vẽ: đường thẳng, tròn, elip tùy ý trong không gian làm việc. Robot RRP.

Buổi 6 (23/2)

Thực hiện bài tập:

1. Hệ tọa độ gắn với Robot
2. Bảng tham số DH

3. Mô phỏng mô hình Robot RRP

4. Tìm bài toán động học thuận

5. Giải bài toán động học ngược

6. Chương trình để tạo các chuyển động của Robot: đường thẳng, đường tròn, đường elip.

- Chia thành 2 trường hợp (theo ảnh)

Bảng DH của trường hợp 1:

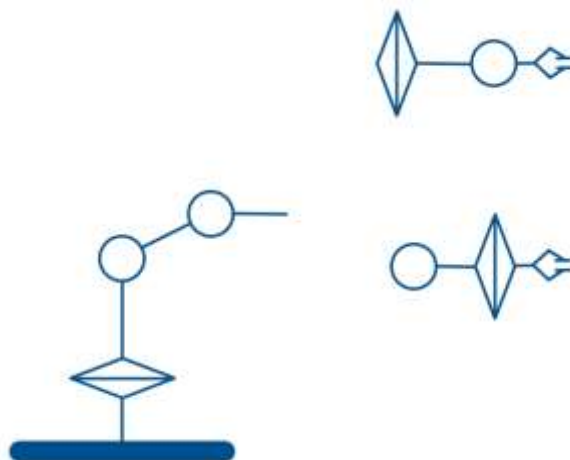
Khớp	θ	d	a	α
0	0	$d_0 = 0.5$	0	0°
1	θ_1	$d_1 = 0.1$	$a_1 = 0.3$	0°
2	θ_2	$-d_2 = -0.2$	$a_2 = 0.1$	180°
3	-90°	$d_3 = [0, 0.2]$	0	0°

Bảng DH của trường hợp 2:

Khớp	θ	d	a	α
0	0	$d_0 = 0.5$	0	0°
1	θ_1	$d_1 = 0.5$	$a_1 = 0.4$	0°
2	θ_2	$d_2 = 0.3$	$a_2 = 0.35$	180°
3	-90°	$d_3 = [0, 1]$	0	0°

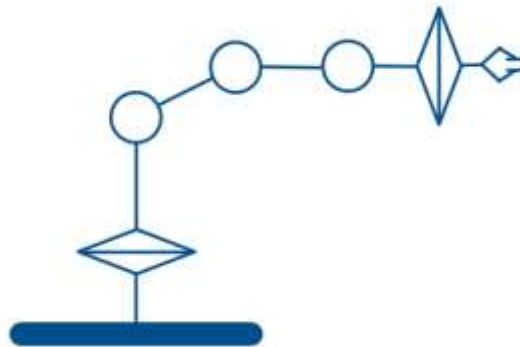
Bài tập 2

Viết chương trình xây dựng mô hình Robot và khảo sát không gian làm việc của cơ cấu Robot có cấu trúc như sau:



Buổi 7 (28/2)

Trường hợp khớp xoay tròn đặt phía trước:

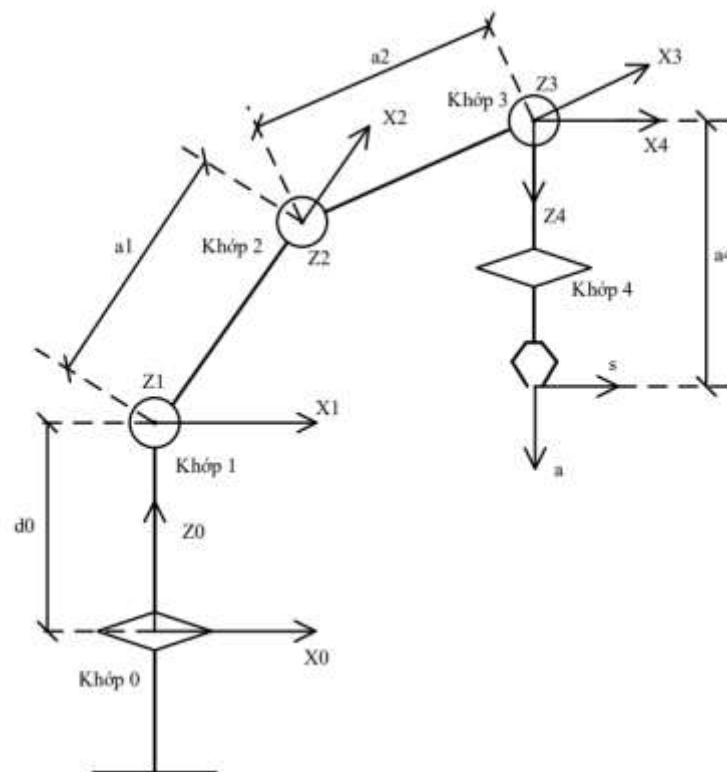


Bước 1: Xây dựng hệ tọa độ gắn với khớp robot:

Buổi 8 (3/1)

Có 2 trường hợp với cơ cấu cổ tay Robot

Trường hợp 1:



Bảng tham số DH: (đơn vị dài cm, góc là độ)

Khớp	θ	d	a	α
0-1	θ_0^*	18	0	90
1-2	θ_1^*	0	16	0
2-3	θ_2^*	0	15	0
3-4	θ_3^*	0	0	90
4-E	θ_4^*	12	0	0

Trường hợp 2:

(Phần bảng)

Xây dựng mô hình Robot trên Matlab

Dùng lệnh transl để lấy ra tọa độ của cơ cấu chấp hành cuối.

Bài tập:

1. Xây dựng mô hình RBTN_2 và 3
2. Giải các bài toán động học thuận và ngược cho cả 3 trường hợp.
3. Khảo sát khả năng làm việc của Robot trong không gian làm việc.

Buổi 9 (6/3)

Xây dựng mô hình động cơ

- Khảo sát không gian làm việc của Robot

```
L(1)= Link([0 18 0 pi/2]);
L(2) = Link([0 0 16 0]);
L(3) = Link([0 0 15 0]);
L(4) = Link([0 0 0 pi/2]);
L(5) = Link([0 12 0 0]);
my_RBTN1 = SerialLink(L,'name','RBTN1');
delta = [0 deg2rad(60) deg2rad(-30) 0 0];
q=[0 0 0 0 0];
qreal = q + delta;
w = [-60 60 -60 60 -10 60];
my_RBTN1.plot(qreal,'workspace',w);
hold on;
for theta1 = 0:5:270
    q = [deg2rad(theta1) 0 0 0 0];
    qreal = q + delta;

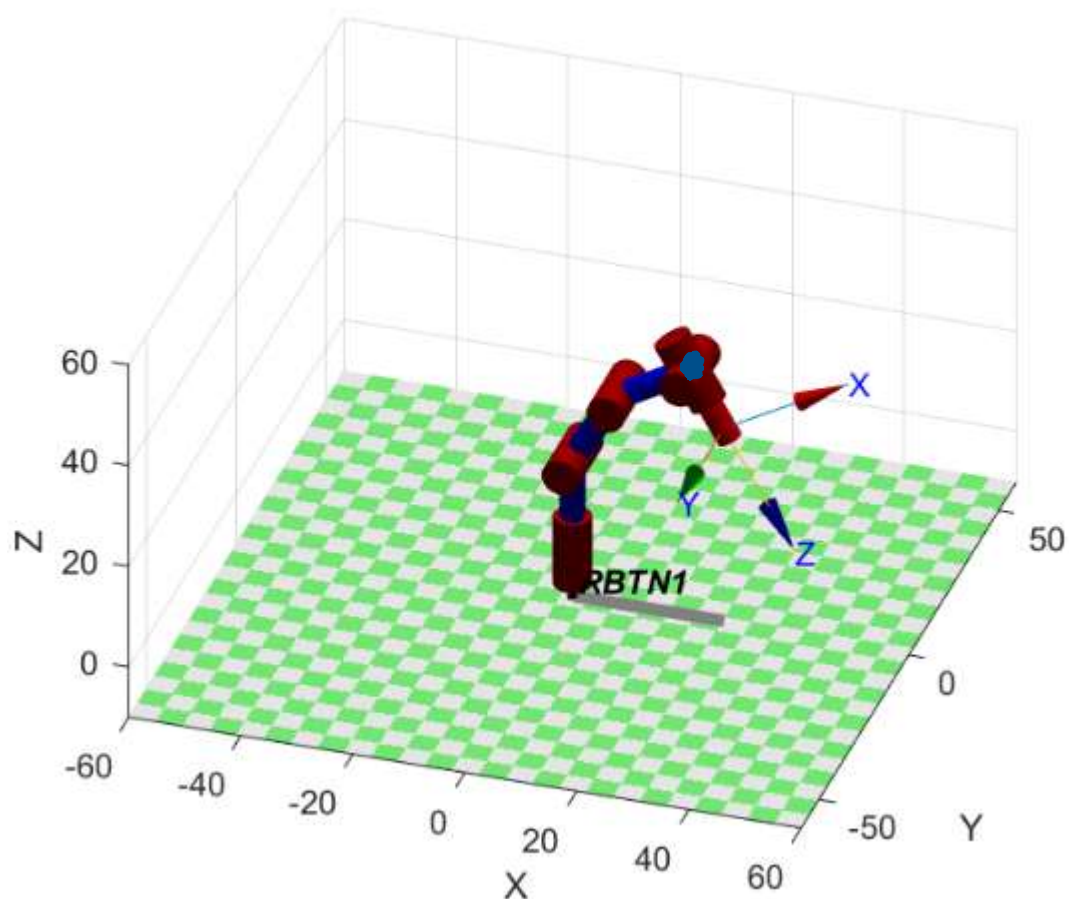
my_RBTN1.plot(qreal,'workspace',w,'delay',0.05,'zoom',1.5,'workspace',w,'view','x');
E = my_RBTN1.fkine(qreal);
p = E.transl;
```

```

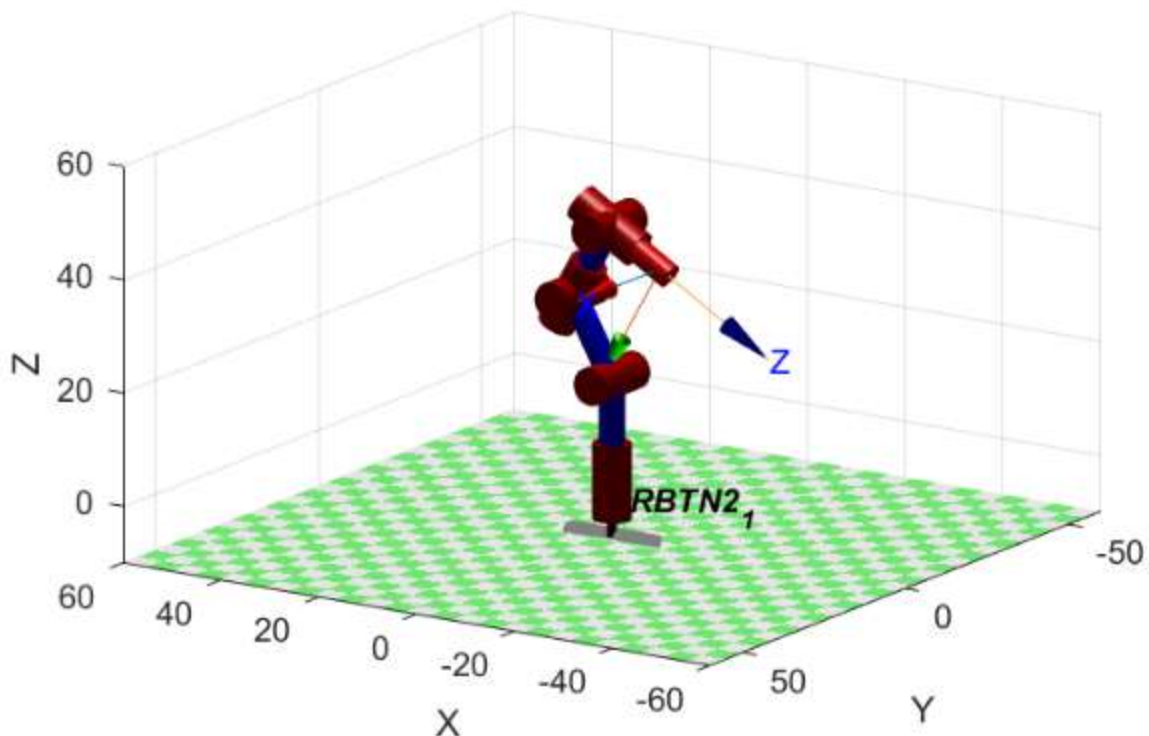
    plot3(p(1),p(2),p(3), '*');
%     hold on;
end
for theta2 = -80:5:80
    q = [deg2rad(theta1) deg2rad(theta2) 0 0 0];
    qreal = q + delta;
    my_RBTN1.plot(qreal, 'workspace', w, 'delay', 0.05, 'zoom', 1.5, 'workspace', w);
    E = my_RBTN1.fkine(qreal);
    p = E.transl;
    plot3(p(1),p(2),p(3), '*');
%     hold on;
end

```

Trường hợp 1:



Trường hợp 2.1:



Chú ý: lệnh Tên Robot.teach để khảo sát Robot.

Bài toán động học ngược

- Có thể sử dụng lệnh ikine để xác định bài toán động học ngược, tuy nhiên hàm ikine không phải luôn trả về kết quả phù hợp.

- Vận dụng vào cơ Robot thực hành:

+ Cách thứ 1: Thực hiện bài toán động học thuận cho cả 5 khớp Robot, sau đó giải bài toán động học ngược.

+ Cách thứ 2: Tách Robot thành 2 cơ cấu riêng biệt: thành phần cánh tay gồm 3 khớp đầu tiên và thành phần cổ tay gồm 2 khớp gắn với cổ tay.

$$E = T_1^2(\theta_1^*) * T_2^3(\theta_2^*) * T_3^4(\theta_3^*)$$

Ma trận biến đổi tổng quát:

$$\begin{pmatrix} \cos(\theta) & -\sin(\theta) \cos(\alpha) & \sin(\theta) \sin(\alpha) & a \cos(\theta) \\ \sin(\theta) & \cos(\theta) \cos(\alpha) & -\cos(\theta) \sin(\alpha) & a \sin(\theta) \\ 0 & \sin(\alpha) & \cos(\alpha) & d \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$${}^0R_1 = \begin{bmatrix} C\theta_0^* & 0 & S\theta_0^* & 0 \\ S\theta_0^* & 0 & -C\theta_0^* & 0 \\ 0 & 1 & 0 & d_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad {}^1R_2 = \begin{bmatrix} C\theta_1^* & -S\theta_1^* & 0 & a_1 S\theta_1^* \\ S\theta_1^* & C\theta_1^* & 0 & a_1 C\theta_1^* \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix};$$

$${}^2R_3 = \begin{bmatrix} C\theta_2^* & -S\theta_2^* & 0 & a_2 S\theta_2^* \\ S\theta_2^* & C\theta_2^* & 0 & a_2 C\theta_2^* \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad {}^3R_4 = \begin{bmatrix} C\theta_3^* & 0 & -S\theta_3^* & 0 \\ S\theta_3^* & 0 & C\theta_3^* & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix};$$

Bài toán động học thuận:

$$n_z = \sin(\theta_2 + \theta_3)$$

$$s_z = \cos(\theta_2 + \theta_3)$$

$$\rightarrow \theta_2 + \theta_3 = \text{atan2}(n_z, s_z)$$

$$n_x = \cos(\theta_1) * \cos(\theta_2 + \theta_3) = s_z * \cos(\theta_1)$$

$$n_y = \sin(\theta_1) * \cos(\theta_2 + \theta_3) = s_z * \sin(\theta_1)$$

$$\rightarrow \theta_1 = \text{atan2}(n_y, n_x)$$

Bài tập VN:

Giải bài toán động học ngược

Lập quỹ đạo của cánh tay robot.

syms t1 t2 t3

$$T12 = [\cos(t1) \ 0 \ \sin(t1) \ 0; \sin(t1) \ 0 \ -\cos(t1) \ 0; 0 \ 1 \ 0 \ 18; 0 \ 0 \ 0 \ 1]$$

$$T23 = [\cos(t2) \ -\sin(t2) \ 0 \ 16*\cos(t2); \sin(t2) \ \cos(t2) \ 0 \ 16*\sin(t2); 0 \ 0 \ 1 \ 0; 0 \ 0 \ 0 \ 1]$$

$$T_{34} = [\cos(t_3) \ -\sin(t_3) \ 0 \ 15*\cos(t_3); \sin(t_3) \ \cos(t_3) \ 0 \ 15*\sin(t_3); 0 \ 0 \ 1 \ 0; 0 \ 0 \ 0 \ 1]$$

$$E = T_{12}*T_{23}*T_{34}$$

$$E =$$

$$[\cos(t_1)*\cos(t_2)*\cos(t_3) \ - \ \cos(t_1)*\sin(t_2)*\sin(t_3), \ - \ \cos(t_1)*\cos(t_2)*\sin(t_3) \ - \\ \cos(t_1)*\cos(t_3)*\sin(t_2), \ \sin(t_1), \ 16*\cos(t_1)*\cos(t_2) \ - \ 15*\cos(t_1)*\sin(t_2)*\sin(t_3) \ + \\ 15*\cos(t_1)*\cos(t_2)*\cos(t_3)]$$

$$[\cos(t_2)*\cos(t_3)*\sin(t_1) \ - \ \sin(t_1)*\sin(t_2)*\sin(t_3), \ - \ \cos(t_2)*\sin(t_1)*\sin(t_3) \ - \\ \cos(t_3)*\sin(t_1)*\sin(t_2), \ -\cos(t_1), \ 16*\cos(t_2)*\sin(t_1) \ - \ 15*\sin(t_1)*\sin(t_2)*\sin(t_3) \ + \\ 15*\cos(t_2)*\cos(t_3)*\sin(t_1)]$$

$$[\cos(t_2)*\sin(t_3) \ + \ \cos(t_3)*\sin(t_2), \ \cos(t_2)*\cos(t_3) \ - \\ \sin(t_2)*\sin(t_3), \ 0, \ 16*\sin(t_2) \ + \ 15*\cos(t_2)*\sin(t_3) \ + \\ 15*\cos(t_3)*\sin(t_2) \ + \ 18]$$

$$[\ 0, \ 0, \ 0, \\ 1]$$

Buổi 10 (8/3)

- Diễn giải công thức của E:

$$n_x = \cos(t_1)*\cos(t_2)*\cos(t_3) - \cos(t_1)*\sin(t_2)*\sin(t_3) = c_1*c_{23}$$

$$n_y = \cos(t_2)*\cos(t_3)*\sin(t_1) - \sin(t_1)*\sin(t_2)*\sin(t_3) = s_1*c_{23}$$

$$n_z = \cos(t_2)*\sin(t_3) + \cos(t_3)*\sin(t_2) = s_{23}$$

$$s_x = -\cos(t_1)*\cos(t_2)*\sin(t_3) - \cos(t_1)*\cos(t_3)*\sin(t_2) = -c_1*s_{23}$$

$$s_y = -\cos(t_2)*\sin(t_1)*\sin(t_3) - \cos(t_3)*\sin(t_1)*\sin(t_2) = -s_1*s_{23}$$

$$s_z = \cos(t_2)*\cos(t_3) - \sin(t_2)*\sin(t_3) = c_{23}$$

$$a_x = \sin(t_1) = s_1$$

$$a_y = -\cos(t_1) = -c_1$$

$$a_z = 0$$

$$p_x = 16 \cdot \cos(t_1) \cdot \cos(t_2) - 15 \cdot \cos(t_1) \cdot \sin(t_2) \cdot \sin(t_3) + 15 \cdot \cos(t_1) \cdot \cos(t_2) \cdot \cos(t_3) \\ = 16c_1c_2 + 15c_1c_2c_3$$

$$p_y = 16 \cdot \cos(t_2) \cdot \sin(t_1) - 15 \cdot \sin(t_1) \cdot \sin(t_2) \cdot \sin(t_3) + 15 \cdot \cos(t_2) \cdot \cos(t_3) \cdot \sin(t_1) = \\ 16s_1c_2 + 15s_1c_2c_3$$

$$p_z = 16 \cdot \sin(t_2) + 15 \cdot \cos(t_2) \cdot \sin(t_3) + 15 \cdot \cos(t_3) \cdot \sin(t_2) + 18 = 16s_2 + 15s_2c_3 + 18$$

- Bài toán động học ngược:

$$\theta_1 = \text{atan2}(a_x, -a_y) = \text{atan2}(n_y, n_x) = \text{atan2}(p_y, p_x) = \text{atan2}(s_y, s_x)$$

$$\theta_2 + \theta_3 = \text{atan2}(n_z, s_z) = 0$$

$$\cos(\theta_2) = c_2 = \frac{(p_x - 15c_1c_2c_3)}{16c_1} = \frac{p_y - 15s_1c_2c_3}{16s_1}$$

$$\sin(\theta_2) = s_2 = \frac{p_z - (18 + 15s_2c_3)}{16}$$

Vậy:

$$\theta_2 = (\theta_1 \neq 90^\circ): \frac{(p_x - 15c_1c_2c_3)}{16c_1}; (\theta_1 = 90^\circ): \frac{p_y - 15s_1c_2c_3}{16s_1}$$

$$\theta_3 = \text{atan2}(n_z, s_z) - \theta_2$$

Chọn $\theta_4 = \theta_5 = 0 = \text{const}$

Tạo quỹ đạo chuyển động theo đường thẳng.

$$x = a \cdot t + x_0$$

$$y = b \cdot t + y_0$$

$$z = c \cdot t + z_0$$

Đi từ điểm A(5,5,5) đến điểm B(12,15,5)

Đưa Robot từ điểm đầu đến điểm A, tiếp đó dịch chuyển từ A đến B.

Tọa độ gốc O: $t_1 = t_2 = t_3 = 0$. Tọa độ điểm gốc O(31,0,18).

Phương trình OA:

Tại thời điểm đầu $t = 0$, ta có: $x_0 = 31, y_0 = 0, z_0 = 18$; $t = 30(s)$, ta có: $5 = a \cdot 30 + 31, 5 = b \cdot 30, 5 = c \cdot 30 + 18 \rightarrow a = -\frac{26}{30}, b = \frac{1}{6}, c = -\frac{13}{30}$

Phương trình AB: (thời gian thực hiện $30s - 30 + 20s$)

$$\begin{aligned} 5 &= a_2 30 + x_{02} \\ 5 &= b_2 30 + y_{02} \\ 5 &= c_2 30 + z_{02} \\ 12 &= a_2 50 + x_{02} \\ 15 &= b_2 50 + y_{02} \\ 5 &= c_2 50 + z_{02} \end{aligned}$$

Ta có: $c_2 = 0, z_{02} = 5, b_2 = 0.5, y_{02} = -10, a_2 = \frac{7}{20}, x_{02} = -\frac{11}{2}$

Với thời gian dịch chuyển chọn 1(s):

$$\begin{aligned} \theta_2 &= (\theta_1 \neq 90^\circ): \text{atan2}\left(\frac{p_z - (18 + 15s23)}{16}, \frac{(p_x - 15c1)}{16c1}\right); (\theta_1 \\ &= 90^\circ): \text{atan2}\left(\frac{p_z - (18 + 15s23)}{16}, \frac{p_y - 15s1}{16s1}\right) \\ \theta_3 &= -\theta_2 \end{aligned}$$

Lệnh Matlab:

T4 =

T5 =

For t = 0:1:30

T1 = atan2(b*t+y0, a*t+x0)

If T1 == pi/2

T2 = atan2((c*t+z0 - 18)/16, (b*t-15)/(16))

T3 = -T2

Else ...

End

TenRB.plot([T1,T2,T3, T4, T5], 'w',w, 'delay',1);

//Vẽ minh chứng cho chuyển động đúng của Robot

End

For t = 30:1:50

End

Buổi 11(13/3)

Chương 2. Lập trình điều khiển chuyển động khớp Robot

Mối liên hệ giữa quỹ đạo chuyển động của cơ cấu chấp hành cuối và các khớp Robot là mật thiết. Nội dung chương đề cập tới việc thực hiện điều khiển chuyển động của khớp Robot.

Nguồn tạo chuyển động cho Robot là động cơ điện.

Thực hiện điều khiển mạch hở và điều khiển mạch kín sử dụng bộ điều khiển PID.

Thực hiện điều khiển mạch hở sử dụng đối tượng động cơ bước (step motor).

Thực hiện điều khiển mạch kín sử dụng đối tượng động cơ một chiều kích từ độ lập với cảm biến vị trí sử dụng biến trở.

Công cụ lập trình điều khiển là vi điều khiển: arduino và/hoặc stm32.

1. Điều khiển mạch hở

- Động cơ được sử dụng là động cơ bước, sơ đồ điều khiển mạch hở góc quay của động cơ bước:



- Việc đảm bảo góc chuyển động của động cơ bước được quyết định bởi cấu tạo và nguyên lý làm việc của động cơ bước.

Về nguyên lý làm việc động cơ bước chia thành 2 loại: động cơ bước lưỡng cực và động cơ bước đơn cực. Trên stator của động cơ bước có 2 pha điện, động cơ bước **lưỡng cực** thì nguồn điện cấp cho các pha có thể đảo dấu, động cơ bước **đơn cực** thì các pha điện có điểm trung gian nối đất, nguồn dương sẽ được lần lượt cấp cho các đầu dây để tạo ra chuyển động của động cơ.

Trong nội dung chương, ta chỉ sử dụng động cơ bước ở chế độ lưỡng cực để tận dụng momen của động cơ. Động cơ bước đơn cực có thể được sử dụng ở chế độ lưỡng cực.

Thông số vỏ máy là 1.8 deg/bước: nghĩa là ở chế độ điều khiển đủ bước mỗi xung sẽ tạo ra dịch chuyển 1.8 deg trong trường hợp momen tải trên trục động cơ không lớn hơn momen danh định của động cơ.

Nhược điểm của chế độ điều khiển đủ bước là chuyển động của động cơ bước giật cục. Ưu điểm của chế độ đủ bước là momen lớn nhất.

Trường hợp momen của động cơ bước đủ lớn, ta có thể chọn sử dụng các chế độ điều khiển chuyển động vi bước: nửa bước, $\frac{1}{4}$ bước, $\frac{1}{16}$ bước, $\frac{1}{32}$ bước. Ở chế độ bước càng nhỏ thì chuyển động của động cơ càng mượt mà. Nhược điểm là làm giảm momen.

Ví dụ: Sử dụng động cơ bước lưỡng cực, cấu hình điều khiển $\frac{1}{16}$ bước, tìm số xung để tạo ra dịch chuyển 30deg trong 2 trường hợp: (động cơ 1.8 độ/bước)

- a. Đo góc trực tiếp tại trục động cơ
- b. Đo góc tại vị trí giảm tốc với tỉ số giảm tốc là 3

Đáp án:

a. 267 xung

b. 800 xung

- Bộ Driver điều khiển động cơ bước

Sử dụng các bộ Driver có sẵn, ví dụ: BT6560 là bộ điều khiển chuyên dụng cho động cơ bước lưỡng cực. Với dòng điện lên tới 3A, và các chế độ làm việc khác.

- Sử dụng vi điều khiển kết nối với Driver thực hiện điều khiển mạch hở động cơ bước:

Tốc độ của động cơ bước? Quyết định bởi tần số của xung: tần số cao tạo ra tốc độ cao và ngược lại, tương ứng với chu kỳ xung T càng nhỏ tạo ra tốc độ cao.

Lập trình tạo chuỗi xung từ vi điều khiển

- **Sử dụng lệnh trễ đơn giản**

- Sử dụng các hàm delay: ưu điểm đơn giản, nhược điểm không điều khiển được nhiều động cơ.

- **Sử dụng timer (các lệnh đợi theo bộ đếm timer)**

- Trong lập trình Arduino sử dụng hàm `millis()`, `micros()` để kiểm tra thời gian thực hiện trạng thái của chân vi điều khiển.

- **Sử dụng các modul tạo xung đặc thù của timer** (khác nhau giữa các loại vi điều khiển khác nhau).

Nhiệm vụ: Sinh viên cài đặt phần mềm Arduino IDE lên máy. Nguồn 12V.

Buổi 12(20/3)

Lập trình điều khiển động cơ bước

- Sử dụng công cụ Arduino: ví dụ sử dụng vi điều khiển atmega 2560

Để xây dựng chương trình ta cần cấu hình phần cứng thực hiện:

+ Sử dụng phương pháp sử dụng hàm trễ: tạo ra chuỗi xung vuông trên chân 13, tạo ra góc quay 30 độ.

Thực hiện: nối chân định hướng của module TB6560 lên VCC hoặc GND

Số xung cần thực hiện $N = 267$ xung.

```
uint32_t N = 267*2;
uint32_t time_ms = 0;
uint64_t time_us = 0;
uint64_t speed_1 = 500;
bool high = true;
void setup() {
    // put your setup code here, to run once:
    pinMode(13,OUTPUT);
}
// time_ms = millis();
time_us = micros();
digitalWrite(13,HIGH);
void loop() {
    // put your main code here, to run repeatedly:
    if((micros()-time_us >= speed_1)&&(N)){
        // time_ms = millis();
        time_us = micros();
        if(high){
            high = false;
            digitalWrite(13,LOW);
        }else{
```

```

    high = true;
    digitalWrite(13,HIGH);
  }
  N--;
}
}

```

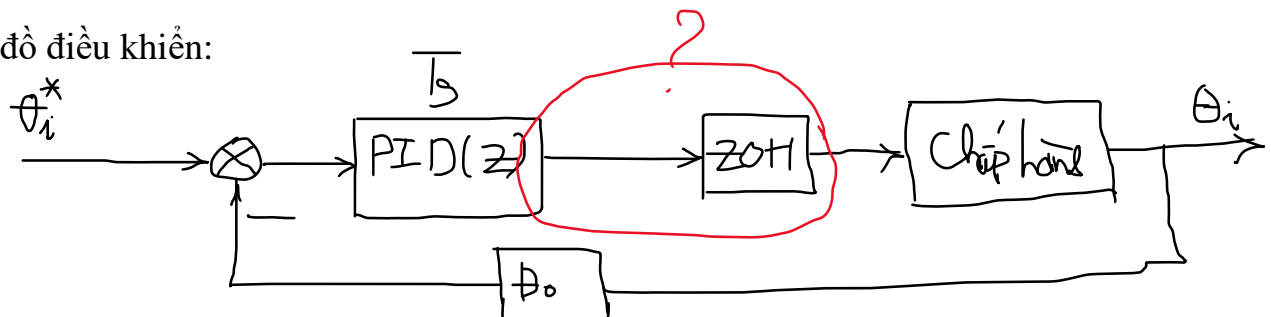
Bài tập: viết chương trình Arduino để điều khiển tốc độ và vị trí cho 2 động cơ bước đồng thời.

Buổi 13(27/3)

Sơ đồ kết nối của mô hình điều khiển động cơ bước (sử dụng arduino)

Mạch điều khiển kín cho truyền động khớp Robot

Sơ đồ điều khiển:



+ Thành phần đo lường:

Đo vị trí có một số phương pháp: một số mô hình robot không yêu cầu độ chính xác chuyển động cao thì có thể sử dụng biến trở. Để đọc giá trị cảm biến trên thiết bị điều khiển, ta dùng ADC.

Với Arduino có thể dùng lệnh `analogRead(số chân)`

Sử dụng ADC của vi điều khiển họ ARM có thể lập trình đọc ADC theo ngắt và ADC theo DMA.

Ví dụ: sử dụng stm32f103 tính năng DMA là tính năng dữ liệu từ ADC sẽ được lấy trực tiếp từ ngoại vi tới vùng nhớ của vi điều khiển.

Ví dụ đo vị trí sử dụng adc

Đo vị trí sử dụng encoder: encoder tương đối đo vị trí, trong stm32 có tính năng encoder mode của timer cho phép tính vị trí từ encoder tương đối.

+ Chấp hành: sử dụng động cơ DC: kết hợp với driver gì và tại sao?

Sử dụng động cơ BLDC: kết hợp với driver gì và tại sao?

+ Bộ điều khiển PID số và



I. Đo vị trí		IV. Hướng:		V. Nối robot			
A0 - M4		4 - INC3		Modul I:	M1 A M2	Modul B:	M5 A M6
A1 - M5		9 - INA3		Modul II:	M3 A M4	Modul C:	M7 A M8
A2 - M2		10 - INC2		Moton	Đo Vị trí	Enable	Thủy
A3 - M6		11 - INA2		M1	A5	2	13
A4 - M3		12 - INC1		M2	A2	3	12
A5 - M1		13 - INA1		M3	A4	5	11
				M4	A0	6	10
				M5	A1	7	9
				M6	A3	8	4
II. Đặt vị trí		- Nguồn nuôi đo vị trí và đặt vị trí lấy từ 5V của VĐK					
A6 - BT3		- Nguồn nuôi NOT (7414) lấy từ 5V ngoài					
A7 - BT4							
A8 - BT6							
A9 - BT2							
A10 - BT5							
A11 - BT1							
III. PWM (Enable)							
2 - EN1A (M1)							
3 - EN1B (M2)							
5 - EN2A (M3)							
6 - EN2B (M4)							
7 - EN3A (M5)							
8 - EN3B (M6)							

Buổi 14(3/4)

- Đo lường: đo dòng điện: trong bài toán điều khiển Momen của động cơ. Để thực hiện đo cường độ dòng điện có một số phương pháp: sử dụng điện trở shunt ví dụ thông số điện trở: 10A, 7.5 mV

Đặc điểm của bài toán đo lường và điều khiển cường độ dòng điện là đại lượng dòng điện có đặc điểm biến đổi nhanh dẫn tới xử lý đọc ADC cần tốc độ cao.

+ Đo tốc độ: sử dụng encoder tương đối hoặc encoder tuyệt đối.

Đo tốc độ sử dụng encoder tương đối: sử dụng timer để đo tốc độ: đếm số xung trong 1 đơn vị thời gian. Tính chu kỳ xung.

Ví dụ: Sử dụng encoder 360 xung/ vòng để đo tốc độ.

Phương án đếm xung: sử dụng timer 1 chạy ở chế độ đếm xung từ nguồn xung chuẩn, và timer 2 ở chế độ bộ đếm với nguồn xung là 1 trong 2 pha A hoặc B.

Giả sử chu kỳ điều khiển/đo lường là 100ms: timer 1 sau 100ms sẽ ngắt để thực hiện kiểm tra giá trị bộ đếm của timer 2.

Giả sử giá trị đếm của timer 2 là 50 thì tốc độ quay trên trục là: $\omega = 8.72(\text{rad/s})$

Phương án đo chu kỳ: trong vi điều khiển timer thường được tích hợp module Input Capture: ta có thể cấu hình để tính thời gian giữa 2 sườn xung liên tiếp.

Ví dụ trường hợp trên: giả sử thời gian giữa 2 sườn xung là 100 ns thì tốc độ quay là:

- Đo vị trí: để đo vị trí chính xác có thể sử dụng encoder tuyệt đối. Một số dòng vi điều khiển (STM32) hỗ trợ tính năng đo vị trí từ encoder tương đối: từ 2 pha A và B kết hợp với module encoder của các timer của vi điều khiển thì có thể tính chính xác được vị trí. Nên sử dụng kết hợp với encoder tương đối có xung Z.

Một số mô hình Robot: vị trí được đo từ biến trở, có thể đọc trực tiếp từ ADC. Sử dụng Arduino thì lệnh:

```
uint16_t Khop1 = analogRead(0);
```

Nhược điểm của sử dụng biến trở là để tính được góc quay thì phải thực hiện chuẩn hóa giá trị điện trở.

- Bộ điều khiển PID

Bộ điều khiển PID liên tục:

$$\begin{aligned} u(t) &= K_p \left(e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right) \\ &= K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt} \\ &= u_p(t) + u_i(t) + u_d(t) \end{aligned}$$

Bộ điều khiển PID số (rời rạc):

$$\begin{aligned} u &= K_p \left(\overbrace{b^*r - y}^e \right) + K_i T_p(z) \left(\overbrace{r - y}^e \right) + \frac{K_d}{T_f + T_p(z)} \left(\overbrace{c^*r - y}^e \right) \\ &= u_p + u_i + u_d \end{aligned}$$

Thường ta có thể chọn $b = 1$, $c = 1$, khi đó có sai số e .

$T_p(z)$: là thành phần tích phân số.

T_f : là hằng số thời gian lọc nhiễu của thành phần vi phân.

Để tính thành phần tích phân số thì ta có các khả năng sau đây:

+ Tích phân tiến: $Tp(z) = \frac{T_s}{z-1}$

+ Tích phân lùi: $Tp(z) = \frac{T_s z}{z-1}$

+ Tích phân hình thang (xấp xỉ Tustin): $Tp(z) = \frac{T_s}{2} * \frac{z+1}{z-1}$

Vận dụng vào các công thức của các thành phần điều khiển:

Ta có:

Bài tập: Tìm phương trình sai phân trong các trường hợp điều khiển PID với các lựa chọn khác nhau của thực hiện xấp xỉ tích phân. Đưa ra phương án thực hiện trên ngôn ngữ lập trình C.

Buổi 15(10/4)

- Function của bộ điều khiển PI:

```
double my_PI(double setpoint, double measure)
{
    ....
    return u;
}
```

Nhận xét:

Chương trình xử lý điều khiển có đảo chiều chuyển động của khớp Robot:

```
Uint16_t chuKiDieuKhien = 100;
Uint32_t thoiGian = 0;
Double u_truoc;
Uint16_t setpoint = 300;
Bool huong = true;
thoiGian = millis();
if(millis() - thoiGian > chuKiDieuKhien)
{
```

```

thoiGian = millis();
measure = analogRead(khop1);
u = my_PI(setpoint, measure);
if(((u>0)&&(u_truoc <0)) || ((u<0)&&(u_truoc > 0)))
{
    analogWrite(khop1_Motor, 0);
    delay_us(500);
    huong = daodau();
    analogWrite(khop1_Motor ,abs(u));
    u_truoc = u;
}else{
    analogWrite(khop1_Motor ,abs(u));
    u_truoc = u;
}

}

Bool daodau(void){
    If(huong) {
        //Huong = false;
        digitalWrite(Chan_Huong, LOW);
        return false;
    }else{
        //Huong = true;
        digitalWrite(Chan_Huong, HIGH);
        return true;
    }
}

```

}

}

Buổi 16(17/4)

Nội dung ôn tập

1. Xây dựng bảng hệ tọa độ và bảng DH cho một số cơ cấu rô bốt thông dụng: Cơ cấu CNC, Robot Scara 3 và 4 bậc tự do, Robot Puma ...

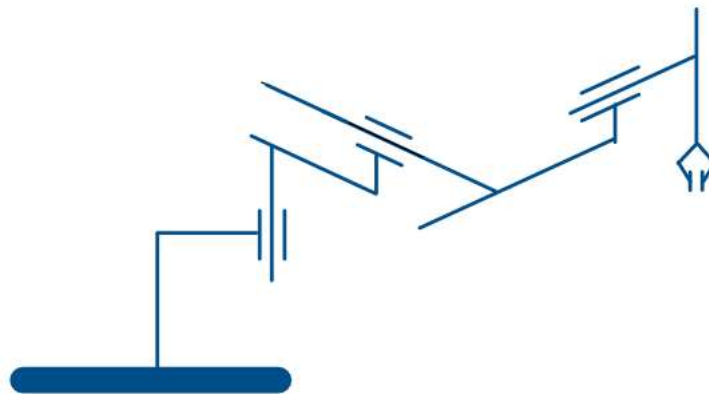
2. Viết chương trình mô phỏng Robot và chuyển động của Robot trên Matlab. Lập trình chuyển động của cơ cấu chấp hành cuối theo quỹ đạo định trước: đường thẳng và cung tròn.

Nhận xét về chuyển động của từng khớp tác động tới quá trình hình thành quỹ đạo Robot từ kết quả mô phỏng.

3. Viết chương trình tạo chuyển động trên động cơ bước.

4. Viết chương trình tạo chuyển động trên động cơ DC trong 2 trường hợp: không sử dụng PID và có sử dụng PID.

Hình ảnh của cơ cấu CNC:



Hình ảnh của cơ cấu Robot Scara 4 bậc tự do:



Hình ảnh của cơ cấu Robot puma 5 bậc tự do:

