

Giải thuật điều khiển robot omni 4 bánh



AutoRC1- Robot tự động của đội BKRC1- 2013

Mục lục

I-	Nội dung, mục tiêu	3
II-	Robot Omni	3
III-	Bánh Omni.....	3
IV-	So sánh giữa omni 3 bánh và 4 bánh	4
V-	Sơ đồ cấu trúc của robot omni 4 bánh	5
VI-	Điều khiển robot 4 bánh omni	6
1.	Đầu vào	6
2.	Đầu ra.....	6
3.	Kiểm nghiệm công thức.....	7
4.	Chương trình	10
5.	Hiệu chỉnh.....	12
a)	Hiệu chỉnh chiều quay động cơ.....	12
b)	Hiệu chỉnh giá trị lớn nhất	12
VII-	Kết quả.....	13
VIII-	Tài liệu tham khảo	14



ROBOT CLUB

I- Nội dung, mục tiêu

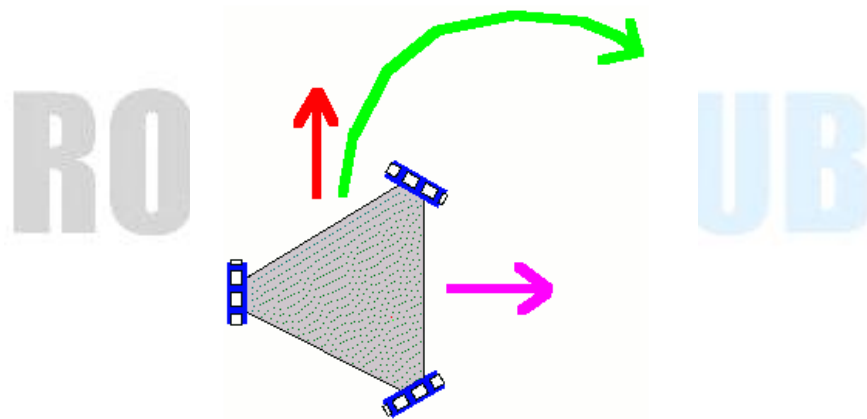
Bài viết cung cấp kiến thức sơ bộ về robot 4 bánh omni, cấu trúc cơ bản của một robot omni, phương pháp điều khiển cơ bản.

Ngoài những kiến thức được trình bày trong bài, tối thiểu cần có các kiến thức về điều khiển động cơ để có thể hiện thực được một robot đa hướng 4 bánh. Ngoài ra, có thể kết hợp thêm một số kỹ thuật khác như giao tiếp gamepad PS2, hoặc dò line, đọc cảm biến siêu âm, giao tiếp SPI... để thực hiện các yêu cầu phức tạp hơn.

II- Robot Omni

Robot sử dụng bánh omni có thể di chuyển theo bất kì hướng nào, ở bất kì góc nào mà không cần quay trước khi di chuyển.

Nhìn chung, để thực hiện được khả năng di chuyển đó, đồng nghĩa với việc có 3 bậc tự do trong mặt phẳng di chuyển (tạm gọi là mặt phẳng Oxy) gồm: di chuyển độc lập theo các trục x, y và xoay theo trục z, cần có tối thiểu là 3 bánh Omni với các động cơ điều khiển độc lập vì mỗi động cơ cho khả năng điều khiển một bậc tự do (trường hợp nhiều hơn 3 bánh, thì vector vận tốc của các bánh đó không phải là các vector độc lập tuyến tính, hay nói cách khác, các bánh sẽ phải ràng buộc nhau để giới hạn số bậc tự do là 3)



Hình 1. Khả năng di chuyển của robot omni

III- Bánh Omni

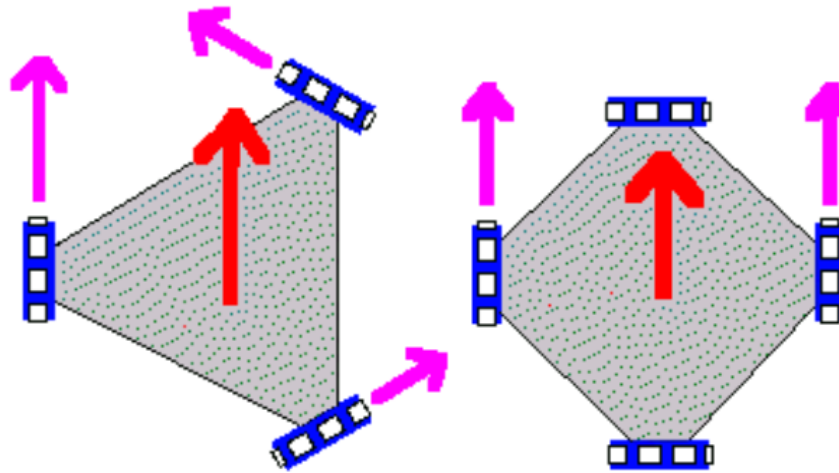
Để robot di chuyển đa hướng, cần sử dụng một loại bánh đặc biệt. Gồm một bánh lớn và các bánh con có trục theo phương tiếp tuyến với bánh lớn và vuông góc với trục bánh lớn. Nhờ đó bánh omni có thể di chuyển theo phương song song với trục.



Hình 2. Một số loại bánh omni

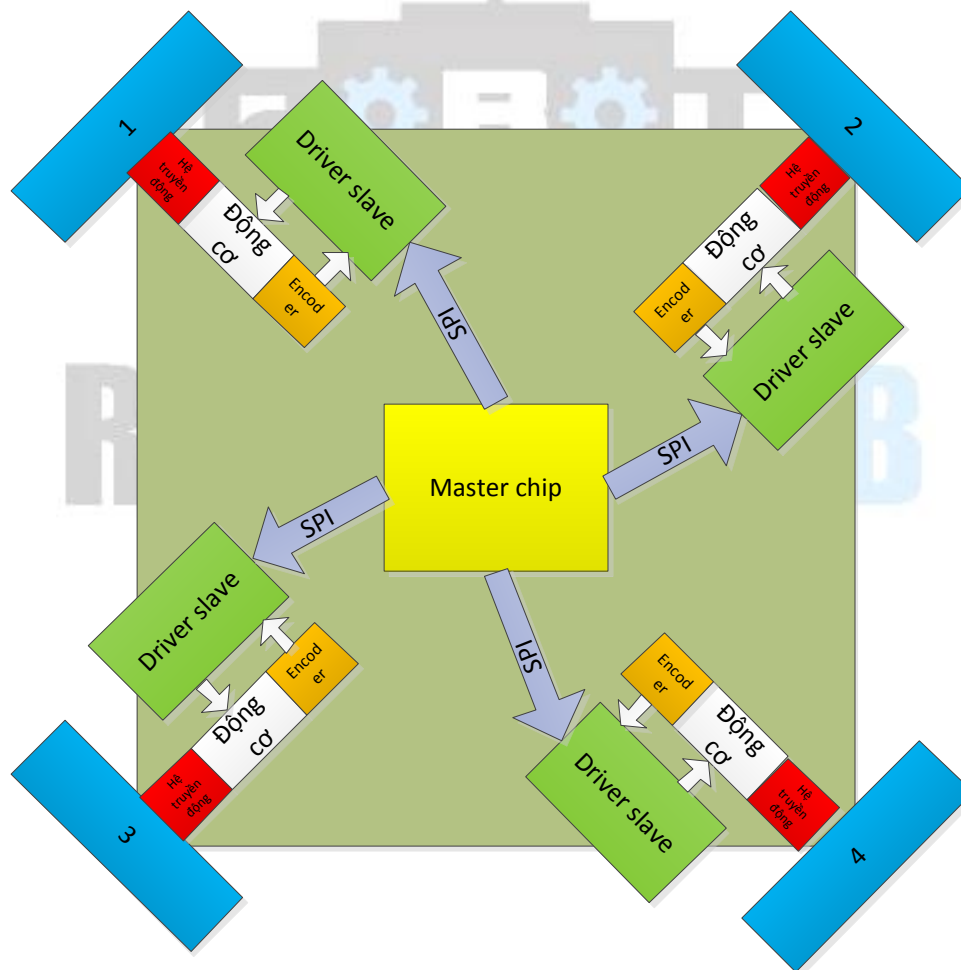
IV- So sánh giữa omni 3 bánh và 4 bánh

	3 bánh	4 bánh
Ưu điểm	3 bánh đồng phẳng mà không cần cơ cấu đặc biệt nào, dễ tương thích với các địa hình gồ ghề 3 bánh- 3 động cơ- 3 encoder- 3 driver-> rẻ tiền hơn	Hiệu suất cao do có 2 bánh có thể chạy ở 100% hiệu suất
Nhược điểm	Hiệu suất không cao do không thể có nhiều hơn một bánh nằm cùng phương với chuyển động	Cần cơ cấu tự lựa để giữ 4 bánh đồng phẳng, làm cấu trúc robot thêm phức tạp Đắt tiền hơn 3 bánh



Hình 3. So sánh về hiệu suất của robot 3 bánh và 4 bánh

V- Sơ đồ cấu trúc của robot omni 4 bánh



Hình 4. Cấu trúc của robot omni

Master chip: thực hiện đọc cảm biến, xử lý, tính toán hướng và vận tốc di chuyển, tính vận tốc của mỗi bánh, truyền vận tốc về cho mỗi Driver slave.

Driver slave: nhận giá trị vận tốc yêu cầu từ vi điều khiển master, thực hiện PID vận tốc để điều khiển vận tốc động cơ cho phù hợp.

VI- Điều khiển robot 4 bánh omni

1. Đầu vào

yêu cầu của người điều khiển gồm:

- Góc yêu cầu θ_d : góc của chuyển động tịnh tiến của robot so với hệ tọa độ gắn với thân
- Vận tốc yêu cầu V_d : robot di chuyển nhanh hay chậm
- Vận tốc xoay V_θ : robot xoay nhanh hay chậm

2. Đầu ra

vận tốc cho mỗi bánh của robot để có thể đáp ứng những yêu cầu đầu vào

$$V_1 = V_d \sin\left(\theta_d + \frac{\pi}{4}\right) + V_\theta$$

$$V_2 = V_d \cos\left(\theta_d + \frac{\pi}{4}\right) - V_\theta$$

$$V_3 = V_d \cos\left(\theta_d + \frac{\pi}{4}\right) + V_\theta$$

$$V_4 = V_d \sin\left(\theta_d + \frac{\pi}{4}\right) - V_\theta$$

Trong đó

V_x : Vận tốc của bánh thứ x, , giá trị trong khoảng $[-1; 1]$

V_d : Vận tốc tịnh tiến của robot, giá trị trong khoảng $[-1; 1]$

θ_d : góc của chuyển động tịnh tiến, giá trị trong khoảng $[0; 2\pi]$; ($[0; 360^\circ]$)

V_θ : Vận tốc xoay của robot, giá trị trong khoảng $[-1; 1]$

Mỗi công thức trên gồm 2 phần chính:

- Phần tác động đến vận tốc tịnh tiến $V_d \sin\left(\theta_d + \frac{\pi}{4}\right)$

- Phân tác động đến vận tốc xoay V_θ

Những giá trị này có thể được hiệu chỉnh bằng một hệ số cho phù hợp với yêu cầu điều khiển hoặc dễ hình dung. CLB Robot sử dụng khoảng giá trị cho các vận tốc là $[-10000; 10000]$

3. Kiểm nghiệm công thức

Thay vì chứng minh công thức này, tác giả trình bày một số trường hợp để kiểm nghiệm tính đúng đắn của công thức

TH1: $\theta_d = 0$; $V_d = 1$; $V_\theta = 0$: robot chạy thẳng

Kết quả đầu ra:

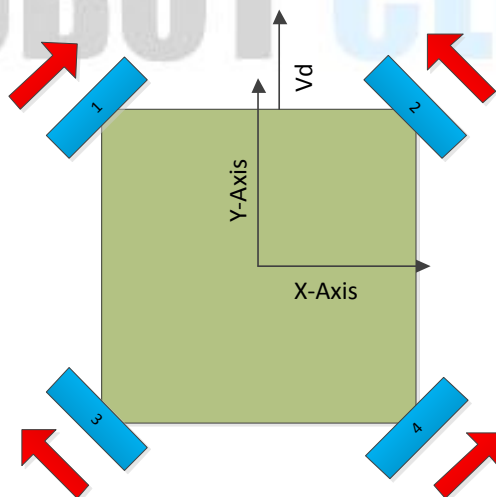
$$V_1 = \frac{\sqrt{2}}{2}$$

$$V_2 = \frac{\sqrt{2}}{2}$$

$$V_3 = \frac{\sqrt{2}}{2}$$

$$V_4 = \frac{\sqrt{2}}{2}$$

Các bánh xe quay như hình vẽ, để thấy rằng robot sẽ chuyển động thẳng về phía trước



Hình 5. Robot chuyển động thẳng về phía trước

TH2: $\theta_d = 45^\circ(\frac{\pi}{4})$; $V_d = 1$; $V_\theta = 0$: robot chạy chéo góc 45 độ

Kết quả đầu ra:

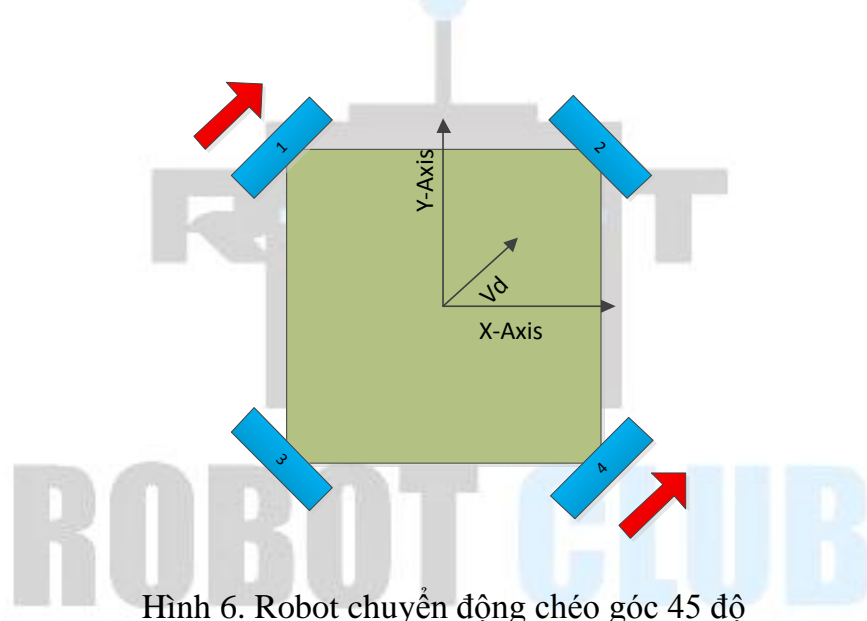
$$V_1 = 1$$

$$V_2 = 0$$

$$V_3 = 0$$

$$V_4 = 1$$

Các bánh xe quay như hình vẽ, dễ thấy rằng robot sẽ chuyển động chéo góc 45 độ



Hình 6. Robot chuyển động chéo góc 45 độ

TH3: $\theta_d = 90^\circ(\frac{\pi}{2})$; $V_d = 1$; $V_\theta = 0$: robot chạy sang ngang góc 90 độ

Kết quả đầu ra:

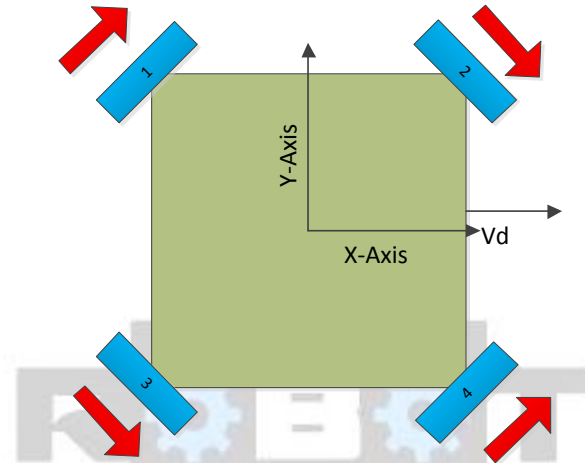
$$V_1 = \frac{\sqrt{2}}{2}$$

$$V_2 = -\frac{\sqrt{2}}{2}$$

$$V_3 = -\frac{\sqrt{2}}{2}$$

$$V_4 = \frac{\sqrt{2}}{2}$$

Các bánh xe quay như hình vẽ, dễ thấy rằng robot sẽ chuyển động sang ngang



Hình 7. Robot chuyển động thẳng sang ngang

TH4: $\theta_d = 0^\circ(0)$; $V_d = 0$; $V_\theta = 1$: robot xoay tại chỗ

Kết quả đầu ra:

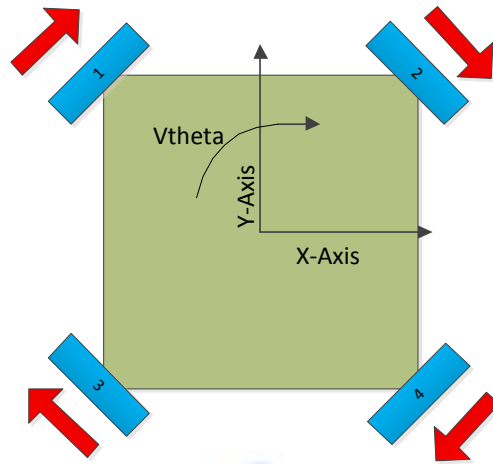
$$V_1 = 1$$

$$V_2 = -1$$

$$V_3 = 1$$

$$V_4 = -1$$

Các bánh xe quay như hình vẽ, dễ thấy rằng robot sẽ chuyển động xoay tại chỗ



Hình 8. Robot chuyển động xoay tại chỗ

4. Chương trình

Dưới đây trình bày chương trình con thực hiện việc điều khiển robot omni. Chú ý là hàm `SPI_SendVelocity(1,(int16_t)V1)` thực hiện việc truyền yêu cầu cho động cơ số 1 giá trị vận tốc là `V1`, slave điều khiển động cơ số 1 nhận dữ liệu này qua giao tiếp SPI và thực hiện việc điều khiển vận tốc động cơ cho đúng. Ở đây chúng ta có thể thấy ưu điểm của hệ thống master- multi slaves: chip vi điều khiển master chỉ việc tính toán, xử lí, truyền yêu cầu về slave, slave sẽ tự thực thi yêu cầu, master không cần “nhúng tay” vào :D

```

/*****
*****
Function name: Move
Decription: Control the movement of robot by giving velocity to each
motor slave
Input: Vd: Transition velocity
          Theta: angle of velocity
          Vtheta: Orientation velocity

Output: None
*****
*****/
void Robot_Move_PID(float Vd, float Theta, float Vtheta)
{
    //float Vd, Theta;           // Desired robot speed [-10000,10000];

    // Desired robot angle [0,360]

```

```

//float Vtheta = 0;           // Desired speed for changing direction [-
10000,10000]
float V1, V2, V3, V4; // Velocity given to motor slave
float V1_abs, V2_abs, V3_abs, V4_abs, Vmax; //using to orientate
smoothly
float Temp;
// Calculate angle and module received from gamepad
//   Theta = -Analog_Angle_Left();
//   Vd = abs(Analog_Module_Left());
//   Vtheta = Analog_Module_Right();

// Calculate velocity for each mechanism driver
V1 = Vd*sin((Theta+45)*0.01745)+Vtheta;           //
Vd*sin((Theta+90)*3.1416/180)+Vtheta
V2 = Vd*cos((Theta+45)*0.01745)-Vtheta;           //
Vd*cos((Theta+90)*3.1416/180)-Vtheta
V3 = Vd*cos((Theta+45)*0.01745)+Vtheta;           //
Vd*cos((Theta+90)*3.1416/180)+Vtheta
V4 = Vd*sin((Theta+45)*0.01745)-Vtheta;           //
Vd*sin((Theta+90)*3.1416/180)-Vtheta

//Calculate the ratio
V1_abs=abs(V1);
V2_abs=abs(V2);
V3_abs=abs(V3);
V4_abs=abs(V4);
////////////////////
//Calculate max value of V
Vmax=V1_abs;
if (V2_abs>Vmax) Vmax=V2_abs;
if (V3_abs>Vmax) Vmax=V3_abs;
if (V4_abs>Vmax) Vmax=V3_abs;
////////////////////
//reduce V to reach the required range
if (Vmax>10000)
{
Temp=10000/Vmax;
V1= V1*Temp;
V2= V2*Temp;
V3= V3*Temp;
V4= V4*Temp;

```

```

    }
    else if (Vmax<10000)
    {
        if (abs(Vd)>abs(Vtheta)) Temp = abs(Vd);
        else Temp = abs(Vtheta);
        Temp=Temp/Vmax;
        V1= V1*Temp;
        V2= V2*Temp;
        V3= V3*Temp;
        V4= V4*Temp;
    }
    // Put the velocity value into the funtion controlling the speed of
    each mechanum driver
    V1 = V_MAX*V1/10000;
    V2 = V_MAX*V2/10000;
    V3 = V_MAX*V3/10000;
    V4 = V_MAX*V4/10000;
    //num=1111;
    SPI_SendVelocity(1,(int16_t)V1);
    //num=2222;
    SPI_SendVelocity(2,(int16_t)V2);
    //num=3333;
    SPI_SendVelocity(3,(int16_t)V3);
    //num=4444;
    SPI_SendVelocity(4,(int16_t)V4);
    //num=9999;
}

```

5. Hiệu chỉnh

a) Hiệu chỉnh chiều quay động cơ

Chiều quay mỗi động cơ có thể bị sai trong quá trình lắp ráp phần cứng, đi dây điện nên cần hiệu chỉnh chiều quay động cơ cho đúng. Để thực hiện việc hiệu chỉnh, gọi hàm Robot_Move_PID(3000, 0, 0) để robot đi thẳng và kiểm tra xem các bánh có quay đúng chiều chưa, nếu sai cần đổi lại dây động cơ và encoder.

b) Hiệu chỉnh giá trị lớn nhất

Trong biểu thức tính vận tốc của mỗi bánh V_x , sẽ có trường hợp tổng của thành phần vận tốc tịnh tiến và vận tốc xoay vượt quá dải giá trị của V_x . Giả sử cả 4

bánh đều vượt quá dải giá trị này thì lúc đó mỗi bánh sẽ đều chạy ở vận tốc lớn nhất, như vậy sẽ bị sai hướng cần điều khiển.

Cách giải quyết trong trường hợp này là chia tỉ lệ cho mỗi bánh sao cho bánh có vận tốc lớn nhất sẽ được ép về thành giá trị vận tốc động cơ lớn nhất, các bánh còn lại sẽ được giảm tương ứng theo tỉ lệ.

```

//////////
//Calculate max value of V
Vmax=V1_abs;
if (V2_abs>Vmax) Vmax=V2_abs;
if (V3_abs>Vmax) Vmax=V3_abs;
if (V4_abs>Vmax) Vmax=V3_abs;
//////////
//reduce V to reach the required range
if (Vmax>10000)
{
Temp=10000/Vmax;
V1= V1*Temp;
V2= V2*Temp;
V3= V3*Temp;
V4= V4*Temp;
}
else if (Vmax<10000)
{
if (abs(Vd)>abs(Vtheta)) Temp = abs(Vd);
else Temp = abs(Vtheta);
Temp=Temp/Vmax;
V1= V1*Temp;
V2= V2*Temp;
V3= V3*Temp;
V4= V4*Temp;
}
}

```

VII- Kết quả

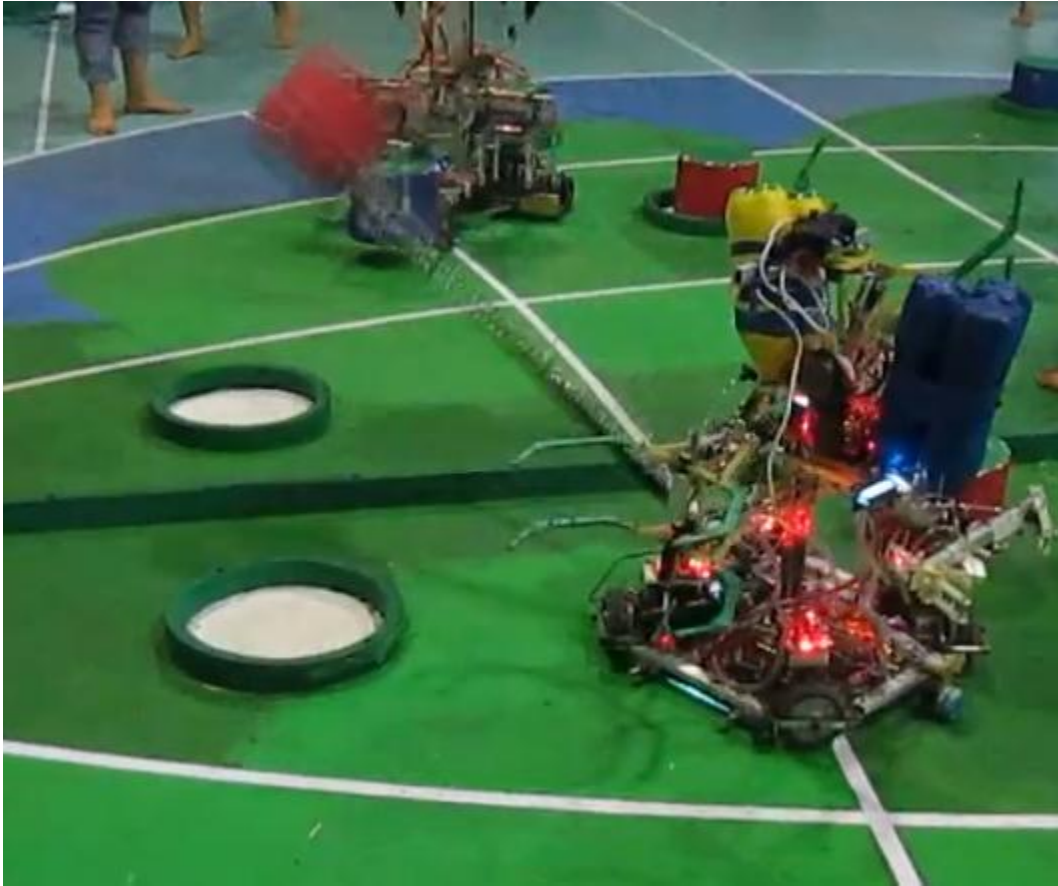
Robot omni được CLB Robot sử dụng trong mùa robocon 2013, với giải thuật điều khiển trên, bước đầu thực hiện được việc điều khiển đa hướng.

Tuy nhiên, cần phát triển thêm:

- Di chuyển theo quỹ đạo định trước.
- Định vị trong không gian

Link tham khảo về hoạt động của robot 4 bánh omni- clb Robot

<http://www.youtube.com/watch?v=3OHvhE1zgNE>



VIII- Tài liệu tham khảo

- [1] Ian McInerney, “Simplistic Control of Mecanum Drive”, FRC Team 2022
- [2] http://www.societyofrobots.com/robot_omni_wheel.shtml