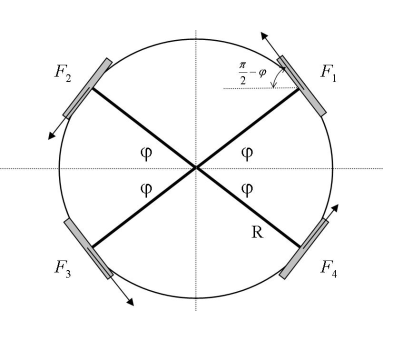
HƯỚNG DẪN LẬP TRÌNH ĐẾ OMNI 4 BÁNH

1. Phân tích lực

*Hình1: biểu diễn vector của đế omni trên*

*hệ trục tọa độ Oxy*

Mục tiêu là ta đi tính toán các lực Fx cho từng bánh.

* Phân tích lực tác động lên từng bánh xe: tịnh tiến các lực F về gốc tọa độ(ví dụ cho F1).

45

ox

oy

F1

F1 = xcos(135)

* Tương tự cho những Fx cong lại ta được hệ phương trình sau.

F1 = xcos(135)

F2 = xcos(-135)

F3 = xcos(-45)

F4 = xcos(45)

* Nếu đế di chuyển trên hệ tọa độ với góc θ,thì ta được:

F1 = xcos(θ + 135) + fx

F2 = xcos(θ - 135) + fx

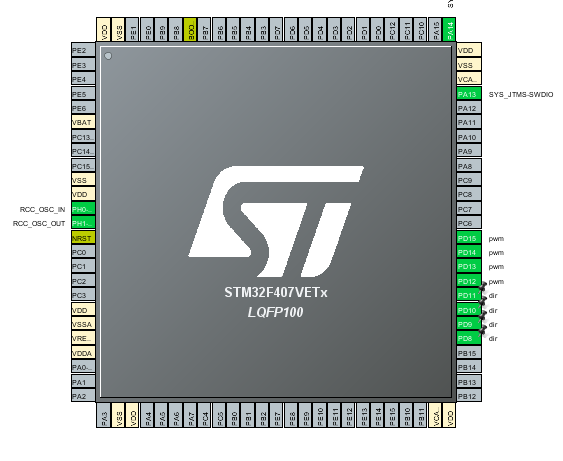
F3 = xcos(θ - 45) + fx

F4 = xcos(θ + 45) + fx

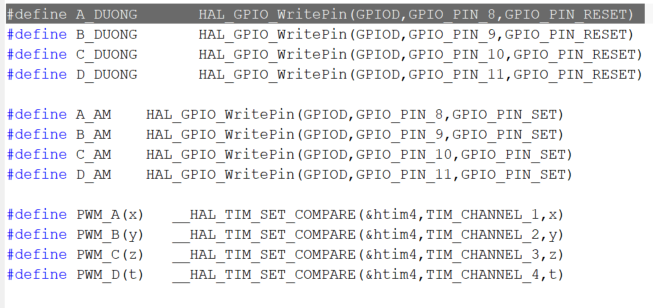
* Ngoài việc di chuyển tịnh tiến, đế còn có thể xoay. Nên ta sẽ cộng thêm 1 lực xoay cho từng bánh(fx).

1. Cách lập trình
   1. Cấu hình cubemx.

Để điều khiển được động cơ dc với cầu h thì cần 2 thông số đó là chiều quay và tốc độ(pwm).

Với đế omni sử dụng 4 động cơ dc thì ta tiến hành cấu hình thông số điều khiển cho 4 động cơ.

* Cấu hình các chân điều khiển như hình trên. Chân pwm sử dụng timer4 với 4 channel.
* Cách tính xung pwm:
  + Tần số của vdk:84mHz.
  + Tần số pwm:1kHz(tự chọn).
  + Giá trị pwm\_max(counter period):999(0->999=1000).
  + Giá trị scaler = 84000000/(10000\*1000).
* Cần cấu hình thêm 1 timer để thực hiên băm xung(pwm) liên tục(1-10ms).
  1. Các hàm cần thiết.
  + Ban đầu ta sẽ đi #define các chân chiều và băm xung để dễ dàng sử dụng.



* + 1 số biến toàn cục:
    - * Vận tốc cài đặt:v,w(vận tốc tịnh tiến và vận tốc xoay mong muốn
      * Vận tốc di chuyển:vt,wt(vận tốc hiện tại).
      * Góc di chuyển: theta.
      * Giá trị gia tốc: at,ag(tăng tốc và giảm tốc).
      * Giá trị pwm cho từng bánh:Va.Vb,Vc,Vd.
  + Hàm cài đặt thông số cho từng bánh.

Int16\_t Wheel\_A(\_vt,\_wt,theta)

{

Static int16\_t \_va;

//tính toán giá trị pwm.

\_Va = \_vt\*cos(theta+phi) +\_wt

//xét chiều cho động cơ

If(\_Va>0) A\_dương.

If(\_Va<0) A\_âm.

Return((int16\_t)fabs(\_va))

}

->viết tương tự cho các bánh còn lại

* + Hàm băm xung và gia tốc(hàm xử lí chính,đặt trong timer).

Void process()

{

//gia tốc

If(vt<v) {

vt+=at;

if(vt>v) vt=v

}

if(vt>v) {

vt-=ag

if(vt<v) vt=v

}

\*tương tự cho vận tốc xoay

//pwm

Pwm\_A(va);

Pwm\_B(vb);

Pwm\_C(vc);

Pwm\_D(vd);

}

* + Hàm di chuyển.

Void move(\_v,\_w,\_theta,\_at,\_ag)

{

V = \_v

W = \_w

Theta = \_theta

At = \_at

Ag = \_ag

Va = wheel\_a(vt,wt,theta)

Vb = …

Vc = …

Vd = …

}

1. Kết hợp với gamepad

Về cơ bản để lập trình được đế omni thì cần các hàm như trong mục II. Nhưng để có thể điều khiển được nó một cần cách dễ dàng thông qua game pad thì cần thêm 1 số hàm di chuyển cố định như: tiến, lùi, trái, phải, xoay trái, xoáy phải,… thì ta sẽ viết các hàm này dựa trên hàm move() với việc điều chỉnh theta cố định(hàm xoay thì chỉ cần w).

Trước hết, cần có thể kết nối được game pad với vdk.(phần này sẽ nói trực tiếp).

Sau khi kết nối được rồi thì sẽ kiểm tra giá trị gamepad gửi về và có thể viết được hàm điều khiển theo y muốn.

1. Kết hợp với la bàn
2. Chạy vị trí
3. Stop DMA  
   Neu data 0 # 1 thi ngat DMA truyen lai tu dau