## **Viết chương trình trong Arduino IDE**

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

#include <PID\_v1.h>

#include <Stepper.h>

LiquidCrystal\_I2C lcd(0x27, 16, 2);

const int stepsPerRevolution = 3200;

Stepper myStepper(stepsPerRevolution, 8, 9);

const int pinA = 2;

const int pinB = 5;

const int nutNhanPin = 10;

volatile long viTriEncoder = 0;

volatile bool daOnDinh = false;

volatile bool pidActive = true ;

double Setpoint, Input, Output;

PID myPID(&Input, &Output, &Setpoint, 4, 0.5, 0.1, DIRECT);

int a ;

void setup() {

Serial.begin(9600);

pinMode(pinA, INPUT);

pinMode(pinB, INPUT);

attachInterrupt(digitalPinToInterrupt(pinA), docEncoder, CHANGE);

//attachInterrupt(digitalPinToInterrupt(pinB), docEncoder, CHANGE);

lcd.init();

lcd.backlight();

Setpoint = 845;

myPID.SetMode(AUTOMATIC);

myPID.SetOutputLimits(-1024, 1024);

myStepper.setSpeed(40);

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("dang can chinh can");

do {

myStepper.setSpeed(150);

Input = analogRead(A0);

myStepper.step(1000);

//delay(10);

} while ( Input < 850 );

myStepper.setSpeed(40);

// Đợi cho đến khi đạt trạng thái cân bằng

do {

Input = analogRead(A0);

myPID.Compute();

myStepper.step((int)Output);

delay(150);

Serial.println(Input) ;

Serial.println("SETUP") ;

viTriEncoder = 0 ;

} while (abs(Setpoint - Input) >= 2);

pidActive = false; // Dừng PID sau khi đạt trạng thái cân bằng

viTriEncoder = 0 ;

}

void loop() {

int giaTriAnalog = analogRead(A0);

Input = giaTriAnalog;

Serial.println(Input) ;

Serial.println("loop") ;

myPID.Compute();

int trangThaiNut = digitalRead(nutNhanPin); // Đọc trạng thái của nút nhấn

Serial.print("dau ra ") ;

Serial.println(trangThaiNut) ;

// Kiểm tra điều kiện kích hoạt

if (trangThaiNut == 0 && abs(Setpoint - Input) <= 1 ) {

daOnDinh = false; // Bắt đầu quá trình ổn định PID

Setpoint = 845; // Đặt lại điểm đặt cho PID

viTriEncoder = 0 ;

}

// Phát xung cho động cơ bước và đếm xung encoder nếu hệ thống chưa ổn định

if (!daOnDinh) {

int stepsToMove = (int)Output;

myStepper.step(stepsToMove);

//int b = digitalRead(pinB);

// Serial.println(stepsToMove) ;

//docEncoder(); // Đếm xung encoder

viTriEncoder = viTriEncoder + stepsToMove;

Serial.println(viTriEncoder) ;

}

// Kiểm tra xem hệ thống đã ổn định chưa

if (abs(Setpoint - Input) <= 1) {

daOnDinh = true;

Output = 0;

} else {

daOnDinh = false;

}

double chia = 3978.2872 ;

double socan = (double)viTriEncoder/chia ;

// Cập nhật thông tin lên màn hình LCD

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Input: ");

lcd.print(Input);

lcd.setCursor(0, 1);

lcd.print("so can : ");

lcd.print(abs(socan),5);

Serial.print("dau ra encoder ") ;

Serial.println(socan) ;

delay(150);

}

void docEncoder() {

int b = digitalRead(pinB);

//Serial.print("b ") ;

//Serial.println(b) ;

if(b > 0){

viTriEncoder = viTriEncoder + 2 ;

}

else{

viTriEncoder = viTriEncoder - 2;

}

}