int shaftAngle=0;

#include <Servo.h>

Servo servo1;

void setup() {

// put your setup code here, to run once:

servo1.attach(9);

Serial.begin(9600);

}

void loop() {

// put your main code here, to run repeatedly:

shaftAngle=map(analogRead(A0),0,1023,0,180);

servo1.write(shaftAngle);

Serial.println((String)"Given command: "+shaftAngle);

delay(100);

}

/\*

Arduino Brushless Motor Control

\*/

#include <Servo.h>

Servo ESC; // create servo object to control the ESC

int potValue; // value from the analog pin

void setup() {

// Attach the ESC on pin 9

ESC.attach(9,1000,2000); // (pin, min pulse width, max pulse width in microseconds)

}

void loop() {

potValue = analogRead(A0); // reads the value of the potentiometer (value between 0 and 1023)

potValue = map(potValue, 0, 1023, 0, 180); // scale it to use it with the servo library (value between 0 and 180)

ESC.write(potValue); // Send the signal to the ESC

}

/\*Example sketch to control a stepper motor with A4988 stepper motor driver and Arduino without a library.\*/

// Define stepper motor connections and steps per revolution:

#define dirPin 2

#define stepPin 3

#define stepsPerRevolution 200

#define Delay 1000 //For a fast motion use 500.

void setup() {

// Declare pins as output:

pinMode(stepPin, OUTPUT);

pinMode(dirPin, OUTPUT);

}

void loop() {

// Set the spinning direction clockwise:

Rotate(200); //Rotate the motor 1 revolution

delay(500);

RotateReverse(200); //Rotate the motor 1 revolution in the reverse direction

delay(500);

}

void Rotate(int numberOfSteps){

digitalWrite(dirPin, HIGH);

for (int i = 0; i <numberOfSteps; i++) {

// These four lines result in 1 step:

digitalWrite(stepPin, HIGH);

delayMicroseconds(Delay);

digitalWrite(stepPin, LOW);

delayMicroseconds(Delay);

}

}

void RotateReverse(int numberOfSteps){

digitalWrite(dirPin, LOW);

for (int i = 0; i <numberOfSteps; i++) {

// These four lines result in 1 step:

digitalWrite(stepPin, HIGH);

delayMicroseconds(Delay);

digitalWrite(stepPin, LOW);

delayMicroseconds(Delay);

}

}

//step motor 28BYJ-48

//Driver ULN2003 module

//https://www.youtube.com/watch?v=avrdDZD7qEQ

#define IN1 8

#define IN2 9

#define IN3 10

#define IN4 11

#define Delay 2

void setup() {

// put your setup code here, to run once:

pinMode(IN1, OUTPUT);

pinMode(IN2, OUTPUT);

pinMode(IN3, OUTPUT);

pinMode(IN4, OUTPUT);

digitalWrite(IN1, LOW);

digitalWrite(IN2, LOW);

digitalWrite(IN3, LOW);

digitalWrite(IN4, LOW);

}

void loop() {

// put your main code here, to run repeatedly:

Rotate(512); // Rotates the shaft 1 full revolution

delay(1000);

RotateReverse(512);

delay(1000);

}

void Rotate(int n){

for (int i=0;i<n;i++){

digitalWrite(IN1,HIGH);

delay(Delay);

digitalWrite(IN1,LOW);

digitalWrite(IN2,HIGH);

delay(Delay);

digitalWrite(IN2,LOW);

digitalWrite(IN3,HIGH);

delay(Delay);

digitalWrite(IN3,LOW);

digitalWrite(IN4,HIGH);

delay(Delay);

digitalWrite(IN4,LOW);

}

}

void RotateReverse(int n){

for (int i=0;i<n;i++){

digitalWrite(IN4,HIGH);

delay(Delay);

digitalWrite(IN4,LOW);

digitalWrite(IN3,HIGH);

delay(Delay);

digitalWrite(IN3,LOW);

digitalWrite(IN2,HIGH);

delay(Delay);

digitalWrite(IN2,LOW);

digitalWrite(IN1,HIGH);

delay(Delay);

digitalWrite(IN1,LOW);

}

}

#define IN1 8

#define IN2 9

#define EN 10

void setup() {

// put your setup code here, to run once:

pinMode(IN1,OUTPUT);

pinMode(IN2,OUTPUT);

pinMode(EN,OUTPUT);

}

void loop() {

// put your main code here, to run repeatedly:

digitalWrite(IN1,HIGH);

digitalWrite(IN2,LOW);

analogWrite(EN,125);

delay(3000);

digitalWrite(IN1,LOW);

digitalWrite(IN2,LOW);

delay(1000);

digitalWrite(IN1,LOW);

digitalWrite(IN2,HIGH);

analogWrite(EN,250);

delay(3000);

}

//visit: http://www.pjrc.com/teensy/td\_libs\_Encoder.html

//Download the library from https://github.com/PaulStoffregen/Encoder

#include <Encoder.h>

// Change these two numbers to the pins connected to your encoder.

// Best Performance: both pins have interrupt capability

// Good Performance: only the first pin has interrupt capability

// Low Performance: neither pin has interrupt capability

Encoder myEnc(5, 6);

// avoid using pins with LEDs attached

void setup() {

Serial.begin(9600);

Serial.println("Basic Encoder Test:");

}

long oldPosition = -999;

void loop() {

long newPosition = myEnc.read();

if (newPosition != oldPosition) {

oldPosition = newPosition;

Serial.println(newPosition);

}

}

int shaftAngle=0;

#include <Encoder.h>

#include <Servo.h>

Encoder myEnc(5, 6);

Servo servo1;

long oldPosition = -999;

void setup() {

// put your setup code here, to run once:

servo1.attach(9);

Serial.begin(9600);

}

void loop() {

// write command is explained in https://www.pjrc.com/teensy/td\_libs\_Encoder.html

long newPosition = myEnc.read();

//following two lines force the encoder to generate a number between 0 and 1800.

if (newPosition<0) myEnc.write(0);

if (newPosition>1800) myEnc.write(1800);

if (newPosition != oldPosition){

oldPosition = newPosition;

Serial.println(newPosition);

}

shaftAngle=(newPosition/10); //decrease the movement of the shaft

servo1.write(shaftAngle);

}