

```
// Include the Arduino Stepper.h library:
#include <Stepper.h>

// Servo Control - BioAmp EXG Pill
// https://github.com/upsidedownlabs/BioAmp-EXG-Pill

// Upside Down Labs invests time and resources providing this open source code,
// please support Upside Down Labs and open-source hardware by purchasing
// products from Upside Down Labs!

// Copyright (c) 2021 Upside Down Labs - contact@upsidedownlabs.tech

// Permission is hereby granted, free of charge, to any person obtaining a copy
// of this software and associated documentation files (the "Software"), to deal
// in the Software without restriction, including without limitation the rights
// to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
// copies of the Software, and to permit persons to whom the Software is
// furnished to do so, subject to the following conditions:

// The above copyright notice and this permission notice shall be included in all
// copies or substantial portions of the Software.

// THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
// IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
// FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
// AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
// LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
// OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
// SOFTWARE.

#if defined(ESP32)
  #include <ESP32Servo.h>
#else
  #include <Servo.h>
#endif

// Define number of steps per rotation:
const int stepsPerRevolution = 1024; // 2048;

int muscle_trigger = 1;

// Wiring:
// Pin 8 to IN1 on the ULN2003 driver
// Pin 9 to IN2 on the ULN2003 driver
// Pin 10 to IN3 on the ULN2003 driver
// Pin 11 to IN4 on the ULN2003 driver

// Create stepper object called 'myStepper', note the pin order:
Stepper myStepper = Stepper(stepsPerRevolution, 8, 10, 9, 11);
```

```

#define SAMPLE_RATE 500
#define BAUD_RATE 115200
#define INPUT_PIN A0
#define BUFFER_SIZE 500
#define SERVO_PIN 10
#define EMG_MIN 2
#define EMG_MAX 10

int circular_buffer[BUFFER_SIZE];
int data_index, sum;

Servo servo;

void setup() {
  // Set the speed to 5 rpm:
  myStepper.setSpeed(35);

  // Serial connection begin
  Serial.begin(BAUD_RATE);
  // Attach servo
  servo.attach(SERVO_PIN);
}

////////////////////
void move_cw(int steps){
  myStepper.step(steps);
}

////////////////////
void move_ccw(int steps){
  myStepper.step(-steps);
}

////////////////////
void check_muscle(){
  if (muscle_trigger > 0){
    move_ccw(10);
  }
}

void loop() {
  // Step one revolution in one direction:
  Serial.println("clockwise");
  myStepper.step(stepsPerRevolution);
  delay(500);

  // Step one revolution in the other direction:
  Serial.println("counterclockwise");
  myStepper.step(-stepsPerRevolution);
  delay(500);
}

```

```
// Calculate elapsed time
static unsigned long past = 0;
unsigned long present = micro
```