```
/* Description:  This example
opens/closes a robot arm according to
EMG Signal
 * Last Change:  16.02.2018
 * Version:      1.2
 */

#include <Servo.h>

#define NUM_LED 6   //sets the numbers
of LEDs
#define MAX 800     //maximum posible
analogArr. TWEAK THIS VALUE!!
#define ARR_LEN 10  //number of
measurement
#define THRESHOLD 500 //threshold for
movement of the arm
Servo ArmServo;

byte ledPins[] = {8, 9, 10, 11, 12, 13};
byte litLeds = 0;
int analogArr[ARR_LEN] = {0};
int finalReading = 0;
bool openFlag = false;

void setup() {
  for (int i = 0; i < NUM_LED; i++) {
//initialize LEDs as outputs and set
them to low
    pinMode(ledPins[i], OUTPUT);
    digitalWrite(ledPins[i], LOW);
  }

  ArmServo.attach(2);
  for (int pos = 0; pos <= 180; pos +=
1) { //open the arm. Initial movement
    ArmServo.write(pos);
    delay(15);
  }
  for (int pos = 180; pos >= 0; pos -=
1) { //close the arm. Initial movement
    ArmServo.write(pos);
    delay(15);
  }
}

void loop() {

//---------------------------- EMG
Measurement
----------------------------------------
-----
  for (int i = 0; i < ARR_LEN; i++) {
```

```
//take a set number of measurements
    analogArr[i] = analogRead(A0);
    delay(2);
  }
  for (int i = 0; i < ARR_LEN; i++) {
//average the ten measurements
    finalReading += analogArr[i];
  }
  finalReading /= ARR_LEN;

  litLeds = map(finalReading, 0, MAX, 0,
NUM_LED);  //define number of lit LEDs

//---------------------------- LED
Control
----------------------------------------
-----
  for (int i = 0; i < NUM_LED; i++) {
//write all LEDs low
    digitalWrite(ledPins[i], LOW);
  }
  for (int i = 0; i < litLeds; i++) {
//set the defined number of LEDs
    digitalWrite(ledPins[i], HIGH);
  }

//---------------------------- Arm
```

```
Movement
------------------------------------------
-----
  if(finalReading > THRESHOLD){ //check
if threshold is reached
    if(openFlag == false){  //check if
arm is open or closed
      ArmServo.write(180);
      delay(150);
    }else{
      ArmServo.write(0);
      delay(150);
    }
    openFlag = !openFlag;
  }

  finalReading = 0;
}
```