```
/*****************************************************************************
 * Steli – solenoid code
 * Reads and processes a value from a flex sensor
 * Activates a solenoid to play a sound
 *****************************************************************************/

//////////////////////////////////////////////////////////////////////////////
// Declarations
//////////////////////////////////////////////////////////////////////////////

// Libraries
import processing.serial.*;
import cc.arduino.*;

Arduino arduino;
Serial port;

// FLEX VARIABLES  ///////////////////////////////////////////////////////////
int flexDefault = 767;      // flex defaul positions, ordered by pin number
int variazioneFlex = 40;     // amount of bending variation to still calculate
the flex sensor as not bended
//int maxFlex = 700;          // max bending value of the flex
int millisSolenoid;
// booleans to know if im playing / bending / soundDelay
boolean stoSuonando;
boolean stoPiegando;
boolean stoContando;

float bend;     // bending value from the flex sensor
float newBend;
float volume;
int conta;      // delay counter
float   millisDelay = 3;              // delay of sound in millis

//////////////////////////////////////////////////////////////////////////////
// Set up
//////////////////////////////////////////////////////////////////////////////

void setup()
{
  // GRAPHICS   //////////////////////////////////////////////////////////////
  size(550, 380);
  background(0);
  smooth();
  noFill();
  strokeWeight(3);
  stroke(0,255,50);

  frameRate(30);

  // ARDUINO   ///////////////////////////////////////////////////////////////
  arduino = new Arduino(this, Arduino.list()[0], 115200);

  //flex sensor
  arduino.pinMode(0, Arduino.INPUT);

  //solenoide
  arduino.pinMode(12, Arduino.OUTPUT);
}

//////////////////////////////////////////////////////////////////////////////
// Draw
//////////////////////////////////////////////////////////////////////////////
```

```
void draw() {
  println(arduino.analogRead(0));
  activeFlex();
}

////////////////////////////////////////////////////////////////////////////
// Active Flex
////////////////////////////////////////////////////////////////////////////

void activeFlex(){

  // FLEX IN DEFAULT POSITION
  // reset sounds, bending value and delay counter
  if(arduino.analogRead(0) > flexDefault-variazioneFlex && arduino.analogRead(0)
< flexDefault+variazioneFlex) {
    stoSuonando = false;
    stoPiegando = false;
    stoContando = false;
    conta = 0;
    bend = 0;
    //println(0+": SONO FERMO --- " +arduino.analogRead(0));
  }

  // DELAY COUNTER
  // counts till millisDelay value, then stops
  if(stoContando = true) {
    if(conta < millisDelay) conta++;
    else {
      stoContando = false;
    }
  }

  // FLEX BENDED
  // detects that it's active, and starts counting
  if(arduino.analogRead(0) <= flexDefault-variazioneFlex ||
arduino.analogRead(0) >= flexDefault+variazioneFlex) {
    stoPiegando = true;
    if(conta < millisDelay) stoContando = true;
    //println(0+": MI HAI PIEGATOOO -> " +arduino.analogRead(0)+" - "+bend+" ---
c="+conta);
  }

  // Calculates max bending achieved before the delay counter stops
  if(stoPiegando == true && stoContando == true) {
    newBend = arduino.analogRead(0);
    if(newBend > bend) bend = newBend;
  }

  // println(bend);

  // PLAY SOUND
  if(stoSuonando == false && stoContando == false) {
    if(bend > 0) {
      stoSuonando = true;
      arduino.digitalWrite(12, Arduino.HIGH);
      millisSolenoid=millis();
      //println(i+": PLAY! :D -- volume = "+bend[i]);
      conta = 0;
    }
  }
  if ((millis()-millisSolenoid) > 100){
    arduino.digitalWrite(12, Arduino.LOW);
  }
}
```