

L1: Hello Again, Arduino! Resistive sensing playground

Wednesday lab: due 11:59 PM, 2/27/13

Friday lab: due 11:59 PM, 3/1/13

Monday lab: due 11:59 PM, 3/4/13

Work in your official group for this lab.

Submission procedure and academic integrity guidelines are the same as outlined in the Lab 0 instructions.

Part 0: Background reading & helpful tutorials

1. Required reading in textbook: Before lab, read: chapters 1–3 (assigned earlier), skim chapter 4 (get familiar with pins; see debugging p. 62–63), read through chapter 6 (analog and digital input and output; schematics) and chapter 11 (touch).

2. Optional tutorials, recommended if you get stuck or want inspiration:

Adafruit tutorial on FSRs: <http://learn.adafruit.com/force-sensitive-resistor-fsr>

Adafruit tutorial on analog inputs, in general, and pots more specifically:

<http://learn.adafruit.com/adafruit-arduino-lesson-8-analog-inputs/variable-resistors-pots>

Adafruit photocell tutorial: <http://learn.adafruit.com/photocells>

Arduino tutorial: using analog input pins: <http://arduino.cc/en/Tutorial/AnalogInput>

Part 1: Acquire your lab kit.

1. Acquire a lab kit for your group from the locker. Your kit equipment includes:

- White breadboard
- Battery holder
- Diagonal cutters
- Wire stripper
- Sparkfun Arduino Flex Kit (in the red cardboard box)
- Sparkfun Beginner Parts Kit (in the red plastic box)

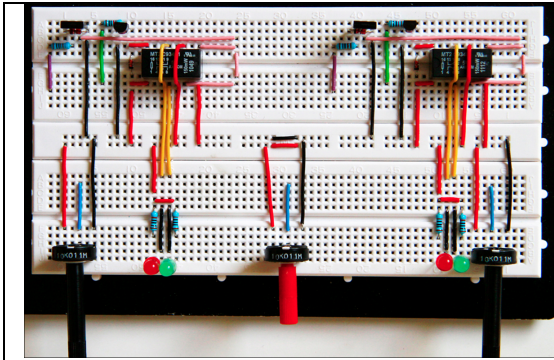
Please pick a Flex Kit and Parts Kit with matching numbers (written in marker on the Flex box, and in marker on tape on the Parts Kit). Learn the number, as **it is now your group number**.

Congratulations! You are now responsible for the care and maintenance of your lab kit. You may remove it from the lab (e.g., to work from your dorm), but you are responsible for returning it at the end of the semester, with all components in working condition and not permanently altered (glued, soldered).

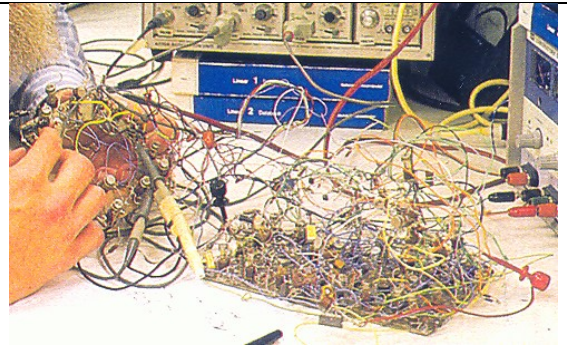
The one exception to the “don’t permanently alter your lab equipment” rule: You are permitted (and encouraged, within reason) to trim leads of resistors, photocell, LEDs, etc. to make breadboarding easier and cleaner. (But DON’T cut your jumper wires! Use the spools of wire for custom wire lengths.)

Any items not listed in the set above (e.g., FSRs, motors, etc.) are to be shared by all members of the class. Do not remove shared items from the lab space, and always store shared items in the tall class cabinet.

If you do plan on taking your group lab kit home with you, we strongly suggest buying a weatherproof box for carrying it around for the rest of the semester.



A neat breadboard, with legs trimmed so that components fit snugly and legs don't get bent or shorted easily. Aim for this.



The alternative. Don't do this.

2. Pick a location in the lab where your group you will work (i.e., 4 chairs in a row, in one of the bays dedicated to your lab day). Select a spot in the cupboard below the lab bench near your work location. This will be the place you keep your equipment and parts. Make a paper sign with your group member names and group number, and place it in the cabinet to claim your space. Congratulations! The space is now yours.

3. Review the lab kit agreement at

<http://www.cs.princeton.edu/courses/archive/spring13/cos436/handouts/LabKitAgreement.pdf>

Please verify that all parts are present (especially make sure that your LEDs are all there, and not stolen for someone else's Lab 0 project). Let us know immediately if anything is missing. If you discover extra parts (e.g., extra LEDs), put them in the big plastic bin in the class cabinet.

Part 2: Force-Sensing Resistor

So what exactly is a Force-Sensing Resistor (FSR)? Just as its name suggests, it is a sensor whose resistance can be changed by applying force/pressure to the surface. In this part of the lab, you will hook an FSR up to your Arduino and watch the resistance change as you apply a force.

1. Get a loaner FSR from the equipment cabinet, in the cardboard box marked "FSRs". This will not be your FSR to keep, so do not remove it from the lab, and replace it in the box when you leave the lab space.

2. Stick the FSR in your breadboard as in the image below. Be careful not to bend or break the delicate leads!

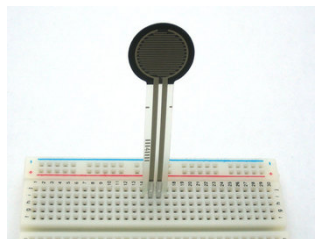


Photo from <http://learn.adafruit.com/force-sensitive-resistor-fsr/connecting-to-an-fsr>

3. Wire your FSR into a circuit, using a **10k Ω pull-down resistor** between your FSR and ground. Connect the point between the FSR resistor and the pull-down resistor to an analog input pin on the Arduino, as shown below. (Again, do this all on your breadboard.)

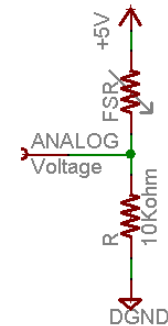
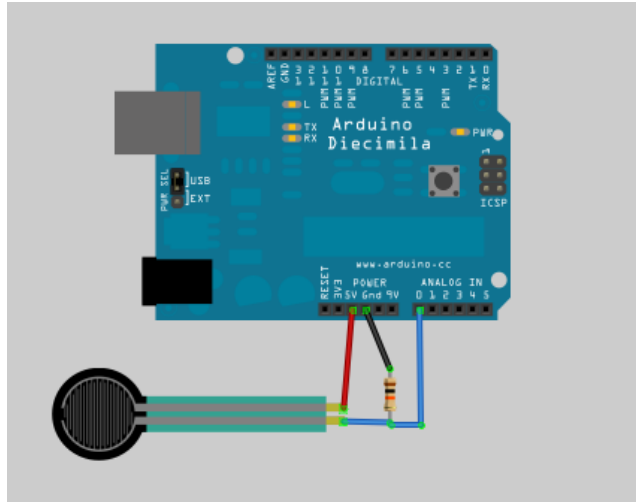


Illustration of FSR with 10kΩ pull-down resistor

Equivalent circuit diagram

Images from <http://learn.adafruit.com/force-sensitive-resistor-fsr/using-an-fsr>

3. Once you have the hardware set up, try out this sample code from <http://learn.adafruit.com/force-sensitive-resistor-fsr/using-an-fsr>. (This code assumes you're measuring voltage using analog input 0.)

```
/* FSR simple testing sketch.

Connect one end of FSR to power, the other end to Analog 0.
Then connect one end of a 10K resistor from Analog 0 to ground

For more information see www.ladyada.net/learn/sensors/fsr.html */

int fsrPin = 0;      // the FSR and 10K pulldown are connected to a0
int fsrReading;     // the analog reading from the FSR resistor divider

void setup(void) {
  // We'll send debugging information via the Serial monitor
  Serial.begin(9600);
}

void loop(void) {
  fsrReading = analogRead(fsrPin);

  Serial.print("Analog reading = ");
  Serial.println(fsrReading);    // the raw analog reading

  delay(1000);
}
```

5. Once you have your FSR set up and responding to your fingers squeezing it, play around with it and answer the following questions.

Q1: Press the FSR in various locations. Is the output a function of where you press it? Explain.

Q2: Is the output a function of how hard you press, keeping the surface area constant? (For example, to keep area constant, you could use a pencil eraser. Don't use anything sharp!)

Q3: What are the minimum and maximum values observed (from the println command output)?

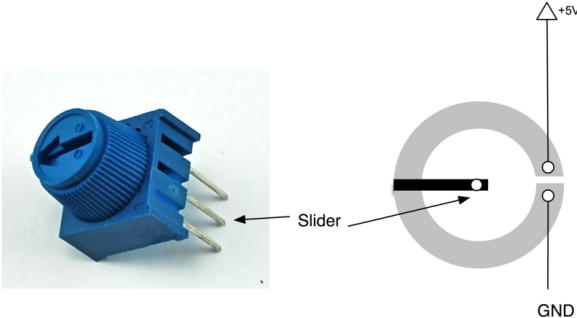



Record your answers on paper. At the end of the lab, use the blackboard form for Lab1 to upload them.

6. **Return your FSR** to the box in the locker.

Part 3: Potentiometer

In this part of the lab, you will set up a rotary potentiometer (or “pot”). You’ve seen these sensors everywhere, although they’re usually covered with nicer-looking knobs. A rotary potentiometer converts rotational motion into a change in resistance. It is a resistor with three terminals (points for connecting to the rest of the circuit). As you physically rotate the knob, a contact inside the pot slides along a resistive strip, increasing or decreasing the resistance between the middle terminal and the two outside terminals (i.e., it’s a changing voltage divider!).

	
<p>The rotary pot in your lab kit. Blue and cute! The right side shows what’s going on inside your pot. Turning the knob varies the length of resistive material between the wiper (attached to the middle pin) and the other two pins. Image from http://learn.adafruit.com/assets/2250</p>	<p>A more common, less cute version of a rotary pot.</p>

1. Disconnect anything hooked up to your Arduino. Connect the potentiometer to the breadboard so that one outer pin is connected to +5V, and the other to ground. Connect the middle pin to an analog input 2 on the Arduino.

Note that no pull-up/pull-down resistor is necessary. Before you proceed, make sure you understand why!

2. Adapt your code from part 1 (if necessary) and examine the behavior of the raw analog readings as you change the pot wiper position.

3. Answer the following questions:

Q4: Why is a pull-down (or pull-up) resistor necessary for the FSR, but not for the rotary pot?

Q5: What is the numerical range of the voltage values you observed in step 2 above, as you changed the pot wiper position? (min, max)

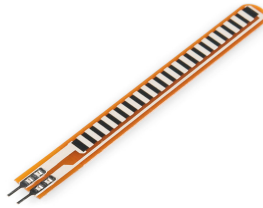
Q6: True or false? The analog voltage level is linear in the rotary position. (i.e., the positions 0.0, 0.1, 0.2, ... 0.9, 1.0 reflect similar integer outputs in the A/D readings)



Record your answers on paper. At the end of the lab, use the Blackboard form to upload them.

Part 4: Bend-Sensitive (Flex) Sensor

From now on, we’re going to give you a little more freedom and independence in hooking up these devices to your Arduino. For this part, use the flex sensor included in your lab kit.



Your flex sensor looks like this.

1. Using electrical tape, affix a piece of cardboard to the bottom to stabilize the bottom part of the sensor, near the leads, so they will not be damaged.
2. Connect it to your breadboard, similar to how you connected the FSR. (Use a $10\text{k}\Omega$ pull-down resistor, and connect to analog pin 0.) Write your own code that prints the readings from analog input 0.
3. Answer the following questions:

Q7: What is the numerical range of the readings you get on pin 0? (min, max)

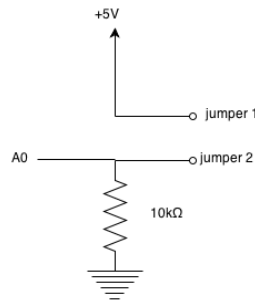
Q8: True or false? The output is symmetric: it doesn't really matter which way you bend the sensor.



Record your answers on paper. At the end of the lab, use the Blackboard form to upload them.

Part 5: A Variable You-sistor

1. Remove the flex sensor from the breadboard, keeping the rest of the circuit intact. Connect one end of one jumper wire to the breadboard port where the left terminal of the flex sensor used to be. Connect one end of **another** jumper wire to the breadboard port where the right terminal of the flex sensor used to be. Let the other ends of the jumper wires fly free, forming the circuit below:



Circuit diagram for you-sistor. Put yourself between jumper 1 and jumper 2.

2. Now do the following:
 - Hold the wires by the insulation on the table, and press your thumb and index finger of one hand gently onto the bare ends. Observe and note the output values.
 - Now press harder, then as hard as you comfortably can. Observe and note the output values.
 - Now lick (or dip in water) your thumb and index finger (I know it's gross, but this is science), and press down hard on the bare wires. Observe and note the output values.

Questions:

- Q9. What is the range of readings (min, max) you get on pin 0 for gentle pressing on the wires?
Q10. For firm pressing on the wires? (min, max)

Q11. For wet fingers on the wires? (min, max)



Record your answers on paper. At the end of the lab, use the Blackboard form to upload them.

Part 6: Photo Sensor

1. For this part, connect your lab kit photoresistor to the breadboard, wiring as you did the FSR, with a 10K pulldown resistor. Use the same code as the FSR to print out raw readings from the analog input pin, with any modifications you believe are necessary.



The photocell in your lab kit looks like this.

Now do the following:

- Cover the photoresistor entirely with your hand. Don't touch any bare wires. Observe the output.
- Aim the photoresistor directly at light. Don't touch any bare wires. Observe the output.
- Wave your hand, fingers, etc. over the device. Observe the effect.
- Place the photoresistor on the table, pointed at the light source in the room. Move about the room some and see if it affects the output.

Questions:

Q12. Approximately what reading do you get when the sensor is fully covered?

Q13. Approximately what reading do you get for a sensor pointing directly at the light?





Record your answers on paper. At the end of the lab, use the Blackboard form to upload them.

Part 7: Control with Resistors

For this part of the lab, you will design an interactive system using **one or more resistive sensors**. To build this system, you may **only** use parts in the following list. (You are not obligated to use all, or even most, of these parts!)

- The following parts in your lab kit:
 - Arduino, wires, USB cord, breadboard, etc. (The obvious stuff)
 - Resistors
 - LEDs
 - Buttons
 - Rotary pot
 - Photocell
 - Flex sensor
 - Softpot (a variable resistor that detects changes in position, like a touch-sensitive slider)
 - Thermistor (a variable resistor that detects temperature changes)
 - Buzzer
- 1 FSR (must not leave the lab, must be returned to box before you leave)
- Processing (e.g., for drawing on the screen)

- Any common materials (paper, cardboard, tape, etc.)
1. Before you start building or coding, complete at least three sketches of alternative designs, on paper. Plan out the materials you will use, and how you will assemble the system. Take pictures (or scans) of these sketches. Cell phone photos are fine as long as the sketches are clearly visible in the photos.
 2. Choose one of your designs. Draw a simple (~ 4 frame) storyboard communicating how and why you might see your proposed system being useful.
 3. Build it!
 4.  Document your system in a WordPress blog. Create a new blog post, then:
 - a. Add it to the “Lab1” category.
 - b. Include the following in your post:
 - i. The names of everyone in your group
 - ii. Your group number
 - iii. A short description of what you built, why, and your assessment of its success. You should mention what you liked about the final result, and you should indicate your opinion on what didn’t work, or what you might like to do differently. All this text should be **no more than one paragraph**.
 - iv. Photos of your sketches, and a short caption explaining what you did in each one
 - v. A photo (or series of photos, or a digital scan) of your storyboard, with resolution good enough to understand it.
 - vi. Photos and/or video showing your final system in action, with short caption(s)
 - vii. A list of parts used in your final system
 - viii. Instructions that would be sufficient for someone else to recreate your design exactly. (This should be no more than one paragraph.)
 - ix. Your entire source code for Part 7. If the code isn’t too long, copy-paste directly into your writeup, and format the code using either the <pre> tag (if you’re in the HTML editing interface) or using the “Preformatted” style, which you can access using the “kitchen sink” button, shown below. Or, if you have lots of code, you can upload it somewhere (e.g., Princeton WebSpace) and include a link to it.
 5. Hit “Publish” to publish your blog post.
 6.  Fill out the Blackboard form for Lab 1 with answers to all the above questions. Also provide your group member NetIDs and the URL to your blog.
 7. Return any communal equipment, tools, etc. to the cabinet. Place your lab kit components in your cupboard and/or take them with you.

Grading Rubric:

Part 1–6: Graded on correct responses to questions. **Worth 2 points per question, total of 26 points.**

Part 7: **Worth 30 points total.** Graded on:

- a) Completeness and quality of documentation: 10 points
 - 0 points: missing/incomplete
 - 6 points: documentation is there but missing key information, missing photos/video, or unclear, or an inappropriate length
 - 10 points: thorough enough for a novice to follow your instructions and re-create easily
- b) Documented design process and critical self-reflection: 10 points
 - 0 points: missing/incomplete
 - 6 points: documentation is there but missing some information, or not presented clearly, or an inappropriate length

- 10 points: includes at least 2 sketches; includes thoughtful reflection on what worked and what didn't; describes motivation or inspiration for your system
- c) Creativity & coolness: 10 points
 - 0 points: bare minimum, looks just like existing tutorials or previous parts of the lab
 - 6 points: result differs substantially from existing tutorials, shows independent thought
 - 10 points: Mind-blowingly awesome; better than anything the TAs would have come up with