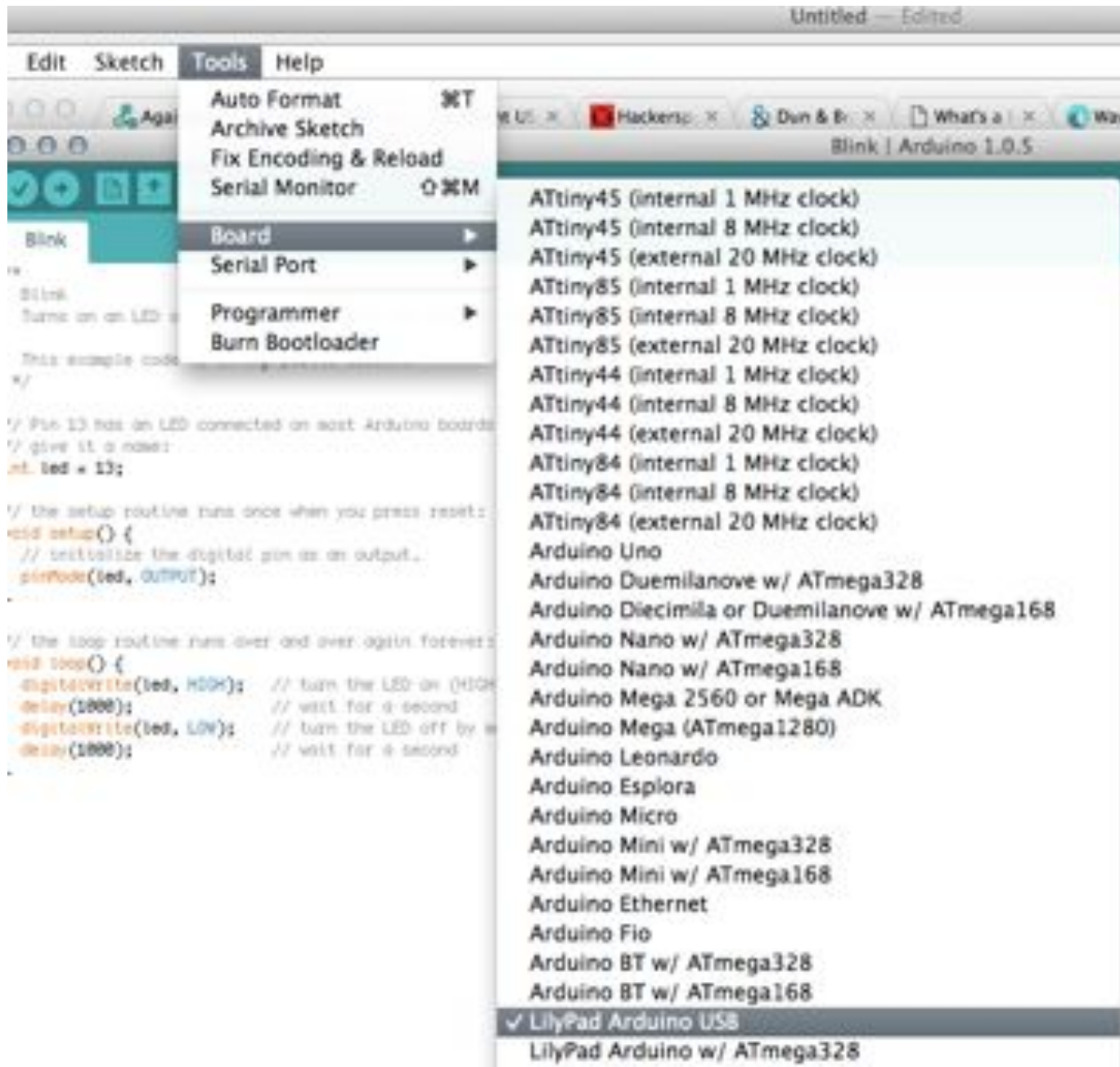


Inputs & Outputs & Sensors

Reminder: Arduino LilyPad



Reminder: Arduino Micro

The screenshot shows the Arduino IDE interface. The 'Tools' menu is open, and the 'Board' submenu is selected. The 'Board' submenu lists various Arduino boards, with 'Arduino Micro' highlighted and marked with a checkmark. The main editor window shows the 'Blink' sketch code.

```
no File Edit Sketch Tools Help
Blink | Arduino 1.0.5

Auto Format ⌘T
Archive Sketch
Fix Encoding & Reload
Serial Monitor ↕ MM

Board ▶
Serial Port ▶
Programmer ▶
Burn Bootloader

ATtiny45 (internal 1 MHz clock)
ATtiny45 (internal 8 MHz clock)
ATtiny45 (external 20 MHz clock)
ATtiny85 (internal 1 MHz clock)
ATtiny85 (internal 8 MHz clock)
ATtiny85 (external 20 MHz clock)
ATtiny44 (internal 1 MHz clock)
ATtiny44 (internal 8 MHz clock)
ATtiny44 (external 20 MHz clock)
ATtiny84 (internal 1 MHz clock)
ATtiny84 (internal 8 MHz clock)
ATtiny84 (external 20 MHz clock)
Arduino Uno
Arduino Duemilanove w/ ATmega328
Arduino Diecimila or Duemilanove w/ ATmega168
Arduino Nano w/ ATmega328
Arduino Nano w/ ATmega168
Arduino Mega 2560 or Mega ADK
Arduino Mega (ATmega1280)
Arduino Leonardo
Arduino Esplora
✓ Arduino Micro
Arduino Mini w/ ATmega328

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
const int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the val
  delay(1000); // wait for a second:
  digitalWrite(led, LOW); // turn the LED off by making the
  delay(1000); // wait for a second:
}
```

Inputs

- Buttons
- Switches
- Potentiometers
- Physical world

Outputs

- LEDs
- Sensors
- Motors
- Sound
- Data



START CROSSING

Watch For
Vehicles



FLASHING

DON'T START

Finish Crossing
If Started

STEADY

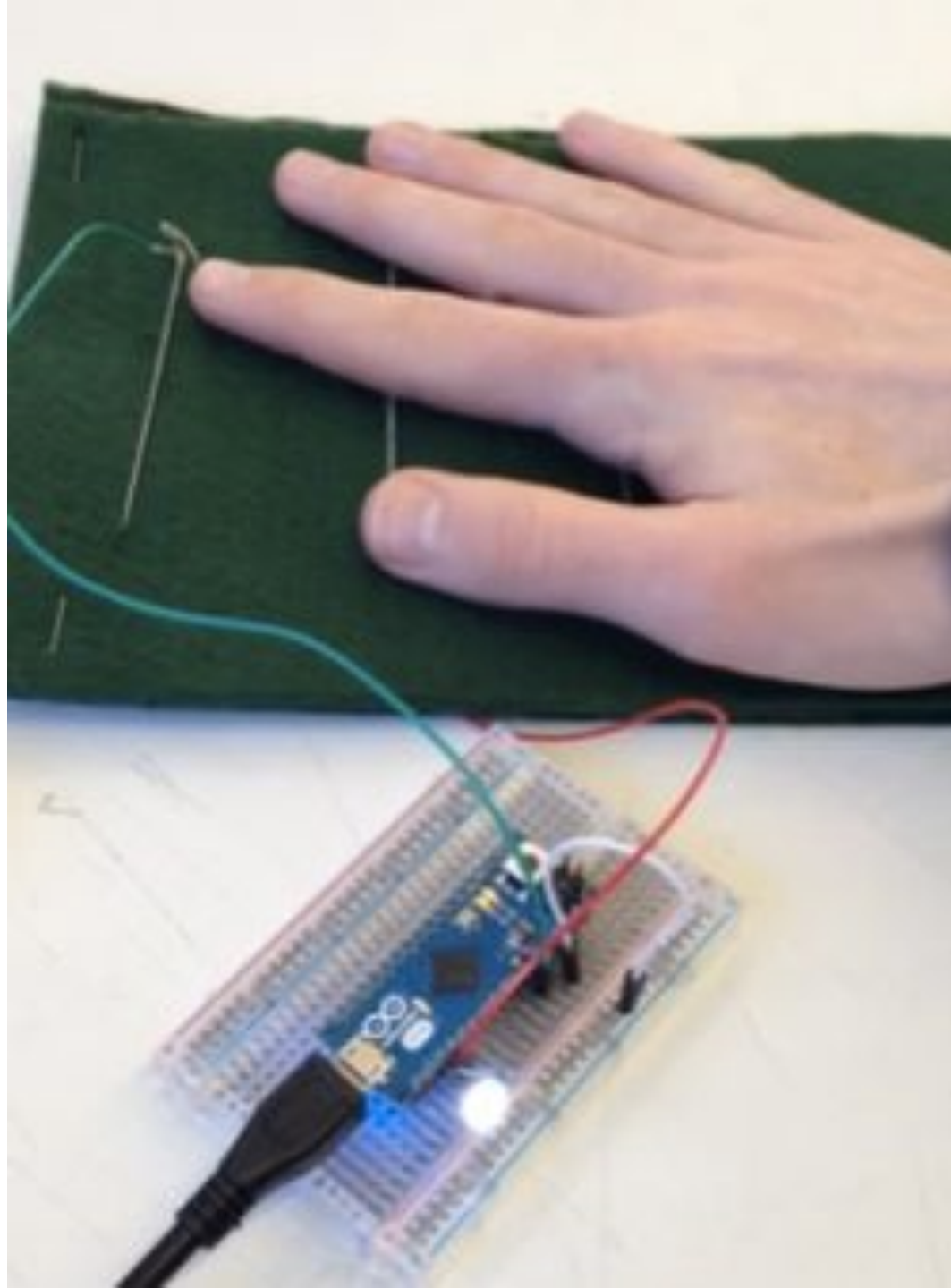


DON'T CROSS







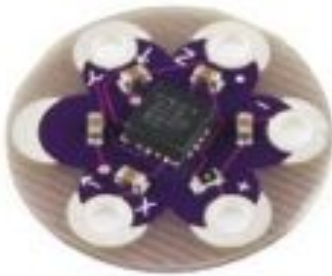






Sensors

- Some are sewable, some you will have to engineer a solution.



Datasheets, Webpages & Tutorials

Begin at the beginning: Where did you get your sensor from?

- Pro tip – buy from a store that has tech support
- Pro tip – get a breakout board, they are much easier to use!

Example: Flex Sensor

- Webpage: <https://www.sparkfun.com/products/10264>
- Datasheet: <https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/FLEX%20SENSOR%20DATA%20SHEET%202014.pdf>
- Tutorial: <https://www.sparkfun.com/tutorials/270>
- Another Tutorial: <http://bildr.org/2012/11/flex-sensor-arduino/>

Arduino is a community



Arduino is open source hardware – lots of people build things and share things – and you will too! Remember not everyone builds the circuit you need or keeps their code up to date. You might have to search for several examples when troubleshooting.

Tutorials are your friend

- Search for multiple tutorials – don't stop at one
- Some tutorials will work for you, some will not, some are old, some are new, but even if they don't work they might trigger what you need to do next.

Searching for tutorials

- Add Arduino or LillyPad into your search
- Look up sensor name and your interaction
- Look up the chip name
- Search multiple forums

Resistance is important

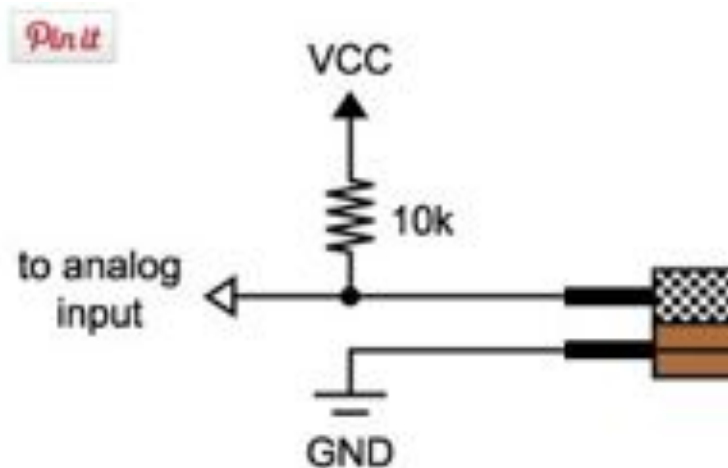
- Many sensors will not work without the proper resistor.
- Many sensors will give you strange values without the right resistor.

Voltage is important

- Your Arduino is spec'ed to a specific voltage and so are sensors
- For this class you will want to work with 3.3 v sensors

Sometimes you need to divide the voltage

- If you see the term “voltage divider” do something like this:

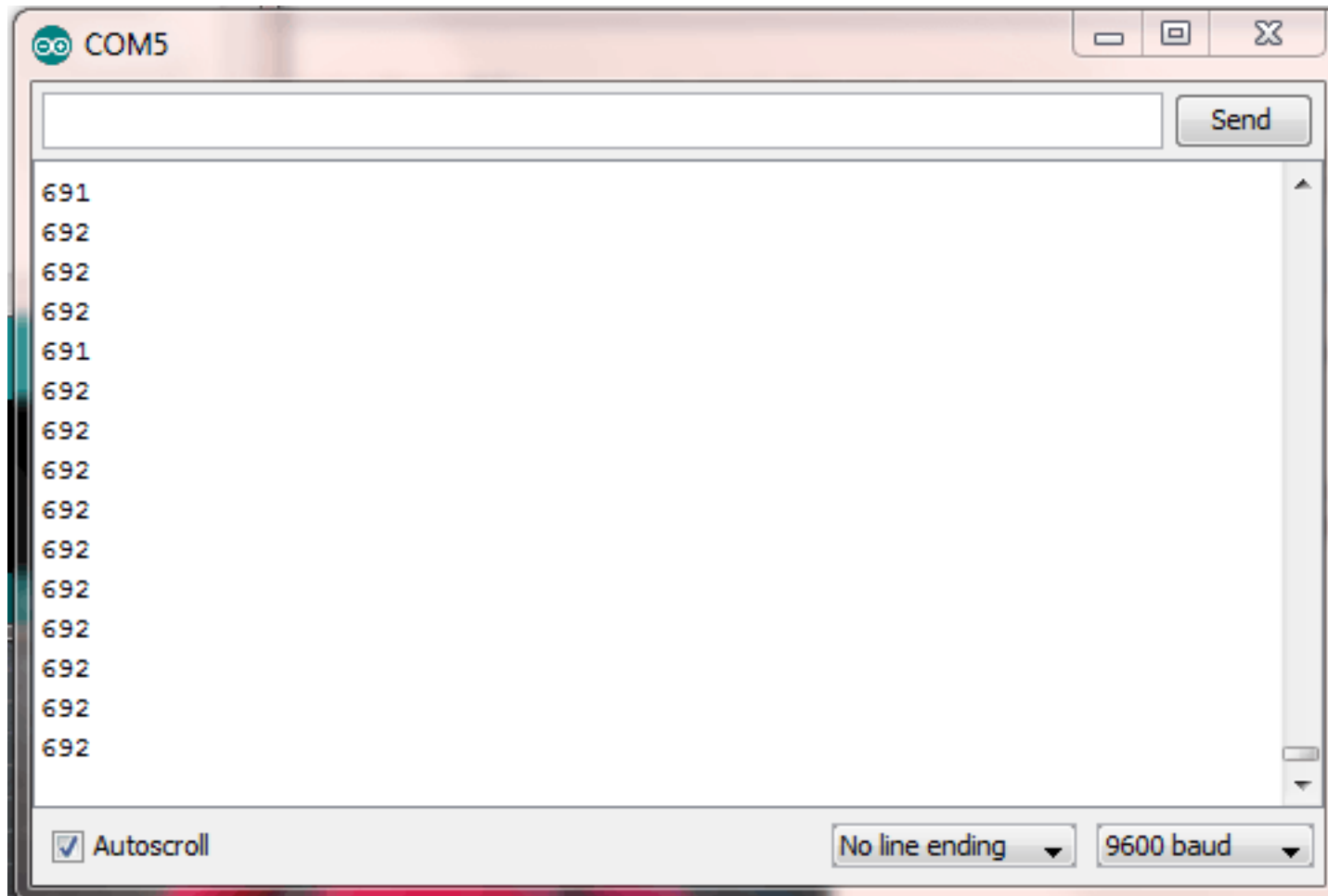


Pin type is important

- Pay attention to whether you have an analog sensor or digital sensor. Analog sensors need to be in the “a” pins.

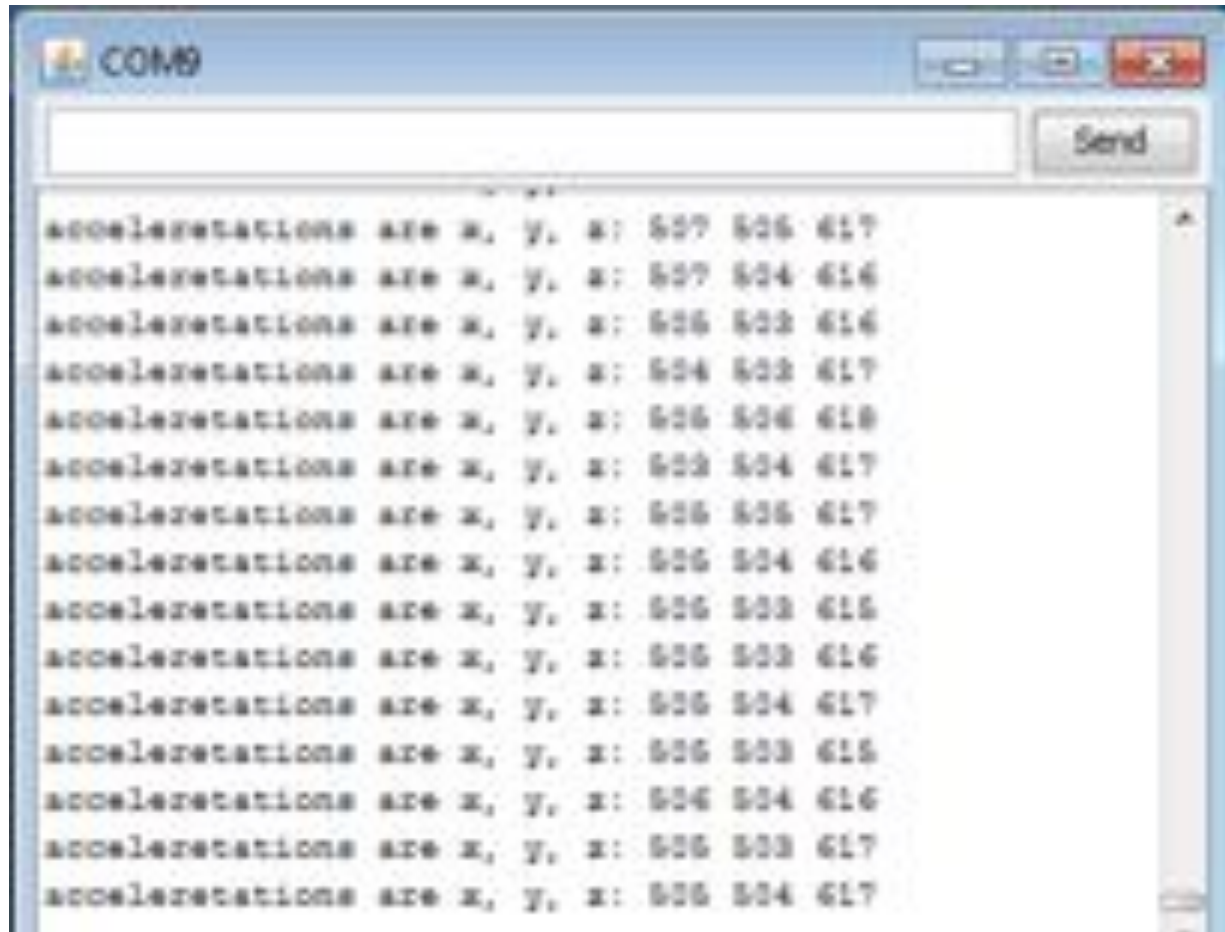
Sensors spit out values

- Sometimes the values don't make sense.



The serial monitor is your friend

- You can add clarification to your values so they are easier to read:



Sensor Data

- Filtering / smoothing data (take an average)
- 1. Take a reading 8 (or x) times
- 2. Divide that reading by 8 (or x) to get an average

Sensor Mapping

- Sometime you will have to map your sensor data to another component.
- This means you will need to understand the range of both components and map the data from one component to the other.

Still having trouble? Think it's the sensor?*

- If all else fails, you can always call tech support.
- But remember, if you've put too much voltage into it, they probably won't replace it.

* Spoiler Alert: The sensor is usually just fine