Beyond “Computers”
Prevalence of Electronics

- We interact with electronics on a daily basis
- Low cost manufacturing
- Wide range of specialty (embedded) hardware
- Electronics require programming
- Worldwide communities of developers
- Open source libraries
Hardware Perspective

- Electrical current moves through material
  - Conductors facilitate the flow of electrons
  - Resistors oppose the flow of electrons
- Voltage is the difference in electric potential between two points
- Electrical charge over time is current
Hardware Components

❖ Resistors
  ❖ Limits flow of electricity (ohms)
  ❖ Can change resistance based on sliders or sensors

❖ Capacitors
  ❖ Stores electrical charge (farads)
  ❖ Can smooth dips and spikes in current signal

❖ Diodes
  ❖ Allows current to flow in one direction
  ❖ Can block or invert signal

❖ Transistors
  ❖ Acts as electrical switch or amplifier
Sensors

- Acquire data from physical world
  - Touch
  - Force
  - Proximity
  - Many more
- Signal converted to digital value
- Digital value influences hardware’s behavior
Circuits and Micro Controllers

- Circuits are configurations of hardware components to perform specific tasks
  - Based on the physical properties of electricity
- Micro controllers are small, simple computers
  - Allow for programmable control of circuitry
- Wiring and Arduino use development environment built on Processing
Hardware/Software Layer

- Programmable I/O boards are micro controllers that accept a variety of components
- Easy to add or remove pieces
- Translate Processing-based code into native, embedded language
- Code uploaded onto board using Serial library
Serial Library

- Reads and writes data from external devices
- Serial port is nine pin I/O port that is emulated through USB
- Add library to code:
  - `import processing.serial.*;
- `Serial()` constructor takes parent, portName and “baud rate”
Serial port;
void setup() {
    //this is program to serialize
    //Serial.list() finds all available serial ports
    //9600 is default baud rate
    port = new Serial(this, Serial.list()[0], 9600);
}
Basic Functions

- `buffer(int)` sets the number of bytes in buffer to serialize
- `read()` reads number between 0 and 255 for next byte in buffer (-1 if no byte)
- `write(src)` writes bytes, ints, chars, bytes[] or Strings to serial port
- `stop()` shuts down data communication on port
Processing and Micro Controllers

https://vimeo.com/74377028
http://www.fh-potsdam.de/studieren/design/studiengaenge/interfacedesign/
Hands-on: Electronics

Today’s activities:

1. Conceptualize a project that uses microelectronics
2. Explain how a user would interact with it
3. Write out a list of parts it would require
4. Describe the data that would be transferred using the read() and write() functions