Lesson Eight: Introduction to Arduinos
Center for Sensorimotor Neural Engineering
Lesson Plan Author: Phelana Pang

LESSON OVERVIEW

Activity Time: Three+ 50 minute periods.

Lesson Plan Summary:
In this lesson, students will be introduced to the parts of an Arduino, basic Arduino code, upload standard sketches, and set up the correlating breadboard circuits.

STUDENT UNDERSTANDINGS

Big Idea & Enduring Understanding:

- Arduinos are small processors (microcontrollers) that can be programmed to sense and control objects.

Essential Question:

- How can Arduinos be set up and programmed to control output based on input received?

Learning Objectives:
Students will know…

- Arduinos can be programmed to produce desired outputs like blinking LEDs.
- Arduino codes are open-source; they are available online and can be edited as needed.
- Arduinos can be connected to breadboards to create more complex circuits.

Students will be able to…

- Upload sample codes onto the Arduino device and produce desired outputs.
- Read codes to understand their purpose (ex: make an LED blink on for 1 second, off for 1 second) and modify these codes to change their purpose (ex: blink at a faster rate).
- Create conditional statements (if, then, else) to vary output based on input.
- Build circuits which integrate the breadboard and the Arduino and upload programs which correspond to the components and their desired action on the breadboard.

Copyright © 2018, Center for Sensorimotor Neural Engineering, University of Washington

Vocabulary:

- **Sketch**: a program for the Arduino which performs a certain function; a unit of code that is uploaded to and run on an Arduino board.

Standards Alignment: This lesson addresses the following middle school Next Generation Science Standards (NGSS)

NGSS Cross-Cutting Concepts

- Systems and System Models
- Structure and Function
- Cause and Effect

NGSS Science & Engineering Practices

- Developing and Using Models

MATERIALS

Material	Description	Quantity
Student Handout 8.1: Introduction to Arduinos	Students label the parts of the Arduino and parts of simple codes.	1 copy per student
Arduinos	$19.95 from https://www.amazon.com/Arduino-Uno-R3-Microcontroller-A000066/dp/B008GRTSV6	1 per group/pair
Breadboards	Item #64, $5.00 from https://www.adafruit.com/product/64	1 per group/pair
LEDs	Item #12062, $2.95 for 20 pack from https://www.sparkfun.com/products/12062	1+ per group/pair
330 ohm resistors	Item #11507, $0.95 for 20 pack from https://www.sparkfun.com/products/11507	1+ per group/pair
Push buttons	Item #9190, $0.50 from https://www.sparkfun.com/products/9190	1 per group/pair
Potentiometers	Item #9806, $0.95 from https://www.sparkfun.com/products/9806	1 per group/pair
Vibration motors	Item #8449, $4 from https://www.sparkfun.com/products/8449	1 per group/pair
Buzzers	$4.49 from https://www.amazon.com/Electric-Buzzer-DC-Physics-Circuits/dp/B0083LWHDQ	1 per group/pair
Motors	Item #11696, $1.95 from	1 per

Copyright © 2018, Center for Sensorimotor Neural Engineering, University of Washington

	https://www.sparkfun.com/products/11696	group/pair
Tilt sensors	$2 from https://www.sparkfun.com/products/10289	1 per group/pair
Pressure sensors	$7 from https://www.sparkfun.com/products/9375	1 per group/pair
Photoresistors	$1.50 from https://www.sparkfun.com/products/9088	1 per group/pair
Temperature sensor	$1.50 from https://www.sparkfun.com/products/10988	1 per group/pair
Flex sensor	$8 from https://www.sparkfun.com/products/10264	1 per group/pair (or 2-3 per class)
Proximity sensor	$13.95 from https://www.sparkfun.com/products/12728	1 per group/pair (or 2-3 per class)

TEACHER PREPARATION

1. Copy handouts as listed in the Materials section above.
2. If you are not familiar with using Arduinos, try them out yourself. Look up tutorials on YouTube and try sample sketches. See the resources section for some online tutorials.

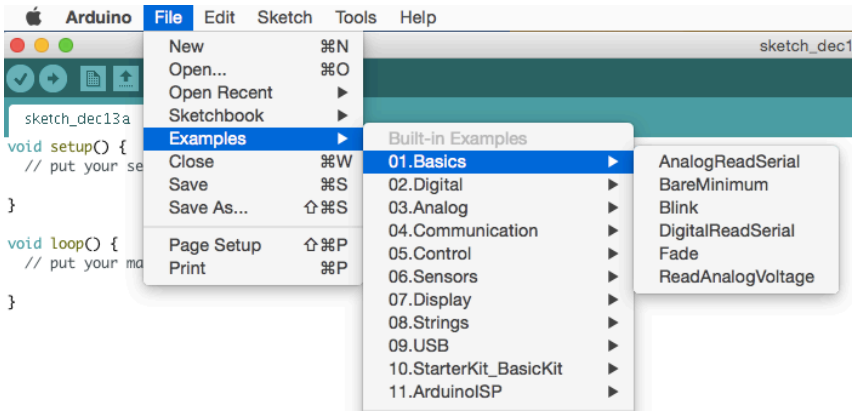
PROCEDURE

Part 1 (60 min)

1. Examine the Arduino and label its parts (page 1 of *Student Handout 8.1*).

2. Examine the basic Arduino code "Blink" and identify parts of the code (page 1 of *Student Handout 8.1).* Main parts to point out:
    a. Comments following "//" describe what the code does
    b. Code is written between { and }
    c. Void Setup command assigns pins
    d. Void Loop command allows action(s) to repeat
    e. Parts that can be modified easily: output pin, time interval

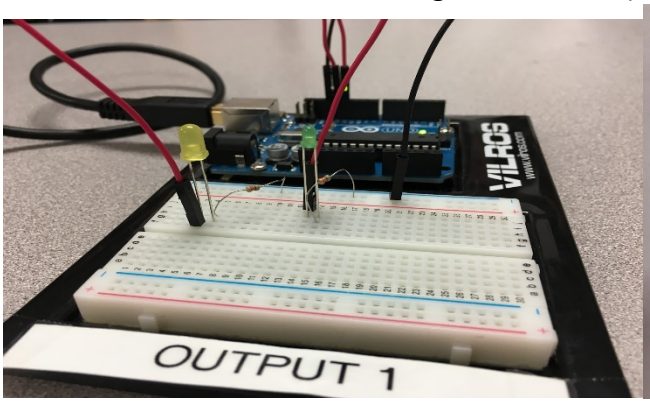
Copyright © 2018, Center for Sensorimotor Neural Engineering, University of Washington

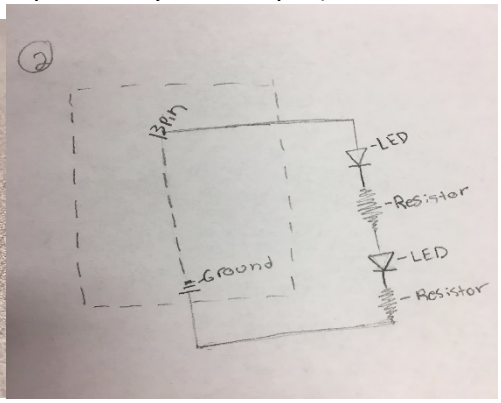


3. Upload Blink code and build associated circuit.

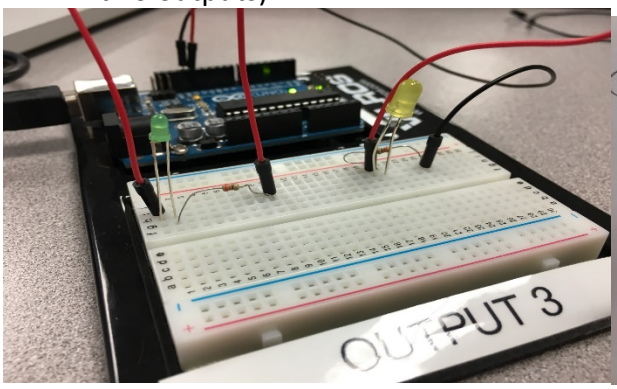
Part 2 (70 min)

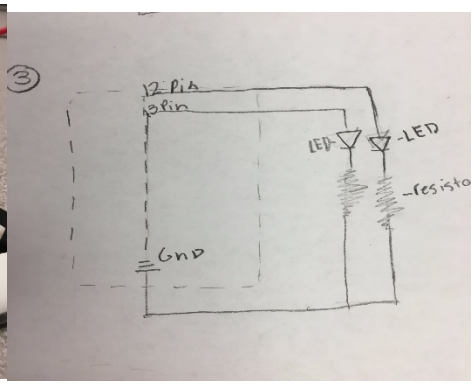
4. Modify the Blink code to do the following:
    a. LED blinks on/off for 0.5 sec, then on/off for 1 sec, then repeat
    b. Two LEDs in series blinking at same rate (requires only one output)





    c. Two LEDs in parallel blinking alternately (one is on while other is off - requires two outputs)





    d. Replace LED with vibration motor (will not work with resistor)

Copyright © 2018, Center for Sensorimotor Neural Engineering, University of Washington

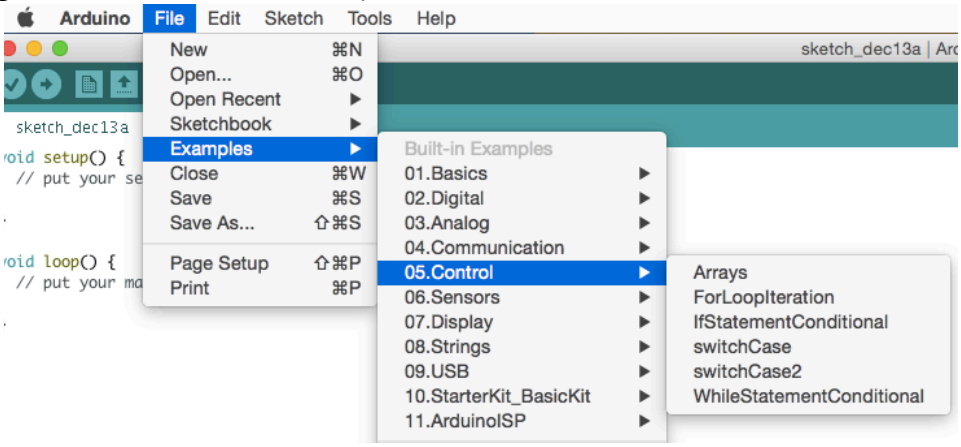
5. For each variation, do the following:
    a. Edit the code
    b. Edit the comments
    c. Save the files (to be uploaded and turned in) following format provided (example: “PP Blink Half Full Sec” to include student initials and description of code)
    d. Draw a circuit diagram

Part 3 (40 min)

6. Review control of output with students presenting code, breadboard, and circuit diagrams for the modifications they worked on previously:
    a. LED blinks on/off for 0.5 sec, then on/off for 1 sec, then repeat
    b. Two LEDs in series blinking at same rate (requires only one output)
    c. Two LEDs in parallel blinking alternately (one is one while other is off - requires two outputs)
    d. Replace LED with vibration motor (will not work with resistor)

Part 4 (30 min)

7. Examine the basic Arduino code “IfStatementConditional” and identify parts of the code (page 2 of *Student Handout 8.1*).

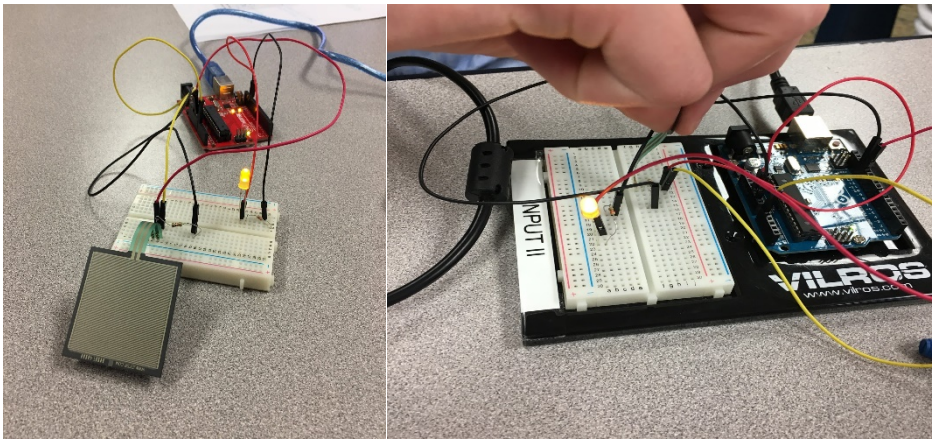
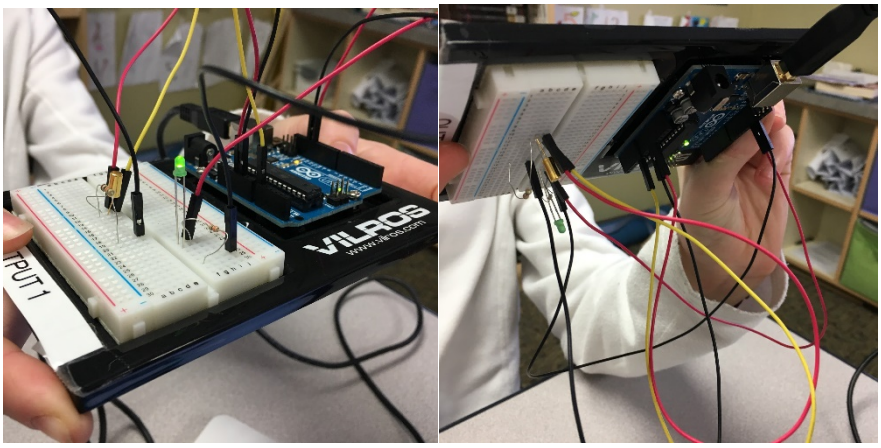
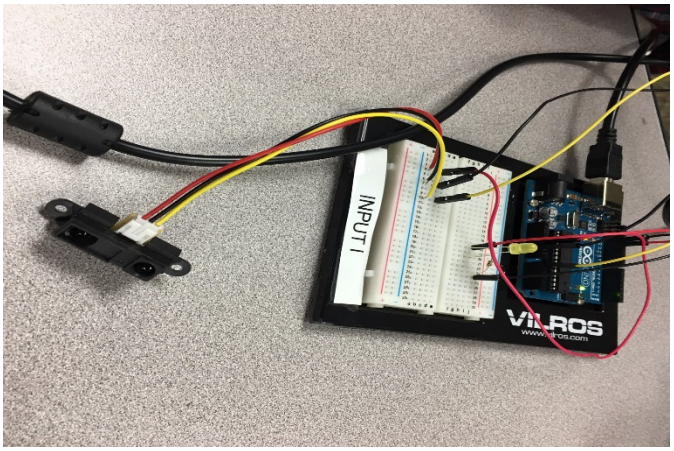


    a. Comments following “//” describe what the code does
    b. Code is written between { and }
    c. Constants: output, analog input, threshold
    d. Void Setup command assigns pins
    e. If, then, else command
    f. Parts that can be modified easily: threshold value

Copyright © 2018, Center for Sensorimotor Neural Engineering, University of Washington

Part 5 (60 min)

8. Modify the "IfStatementConditional" code and/or breadboard to do the following:
    a. Change the threshold value
    b. Change the input: force sensor, tilt sensor, proximity sensor (use 10 ohm resistor with each sensor)
    c. Control multiple LEDs

Force Sensor	
Tilt Sensor	
Proximity Sensor	

Copyright © 2018, Center for Sensorimotor Neural Engineering, University of Washington

9. For each variation, do the following:
    a. Edit the code
    b. Edit the comments
    c. Save the files (to be uploaded and turned in) following format provided (example: “PP Blink Half Full Sec” to include student initials and description of code)

STUDENT ASSESSMENT

Assessment Opportunities: Student knowledge, skills, and concepts for this lesson will be assessed in a number of ways.

- As students work with their Arduinos, ask them to explain to you what each component or each part of the code does.
- Check students’ comments on the code they modify. This helps to see whether they understand the purpose of each line of code.

Student Metacognition:

- Ask students to think about where they can seek answers/help without asking the teacher.
- As students troubleshoot their circuits and code, they can reflect on the strategies they used to solve their problems. They can write a reflection about this in their lab notebooks at the end of each day.

EXTENSION ACTIVITIES

Extension Activities:

- Students can use more components or try sketches beyond the ones that are outlined in this lesson.
- Students who are more familiar with the Arduino can create tutorials for others.
- Students can try different circuits, such as wiring an RGB LED and using the Arduino code to control its color. The Vilros Ultimate Starter Kit Guide or the SparkFun Inventor’s Kit Guide provides some helpful tutorials.
    - https://cdn.sparkfun.com/datasheets/Kits/SFE03-0012-SIK.Guide-300dpi-01.pdf

Copyright © 2018, Center for Sensorimotor Neural Engineering, University of Washington

TEACHER BACKGROUND & RESOURCES

Background Information:

General Arduino troubleshooting note: if board and port are set correctly but code won't upload to the board, go to Tools > Board > Board Manager to check for updates on board packages. Update and restart the Arduino IDE.

Teacher Resource 8.1 - Sample sketches for varying outputs

Resources:

- Arduino Board Anatomy:
  - https://www.arduino.cc/en/Guide/BoardAnatomy
  - http://arduinoarts.com/wp-content/uploads/2011/08/Arduino-callouts1.jpg
- Arduino Tutorials:
  - https://www.arduino.cc/en/Tutorial/HomePage
  - http://www.ladyada.net/learn/arduino/
  - http://forefront.io/a/beginners-guide-to-arduino/
- Videos by Jeremy Blum:
  - https://www.youtube.com/watch?v=fCxzA9_kg6s

Arduino Codes:

- *Teacher Resource 8.1: Sample Sketches for Varying Outputs*
- Force Sensor Resistor Arduino Code:
  - https://learn.adafruit.com/force-sensitive-resistor-fsr/using-an-fsr
- Analog Input Varying Length of LED Blink:
  - https://www.arduino.cc/en/Tutorial/AnalogInput
- Analog Input/Output Dimming LED:
  - https://www.arduino.cc/en/Tutorial/AnalogInOutSerial
- Boolean Operators (and, or, not):
  - https://www.arduino.cc/en/Reference/Boolean
- Flex Sensor Tutorial (SparkFun):
  - https://learn.sparkfun.com/tutorials/flex-sensor-hookup-guide

Citation:

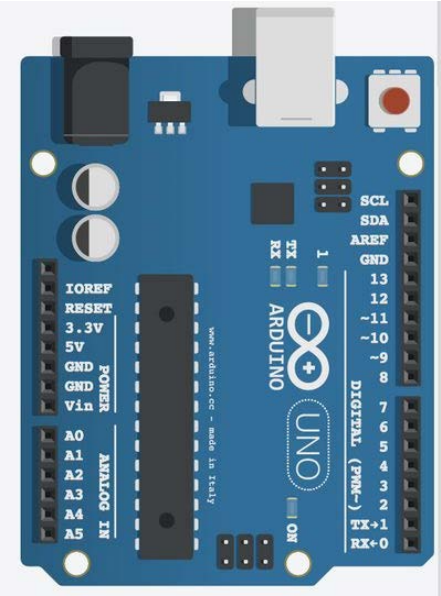
Photographs by Phelana Pang.

Copyright © 2018, Center for Sensorimotor Neural Engineering, University of Washington

Student Handout 8.1: Introduction to Arduinos

Name:_____ Date:_____ Period:_____

Label the parts in the diagram below.



https://cdn.instructables.com/F6R/IPAP/HQF9H5IO/F6RIPAPHQF9H5IO.MEDIUM.jpg

Label the parts in the basic Arduino sketch below.

```
/*
  Blink
  Turns on an LED on for one second, then off for one second, repeatedly.

  Most Arduinos have an on-board LED you can control. On the Uno and
  Leonardo, it is attached to digital pin 13. If you're unsure what
  pin the on-board LED is connected to on your Arduino model, check
  the documentation at http://www.arduino.cc

  This example code is in the public domain.

  modified 8 May 2014
  by Scott Fitzgerald
 */


// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(13, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(13, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);              // wait for a second
}
```

ngineering, University of Washington

What can you conclude about what the following code does?

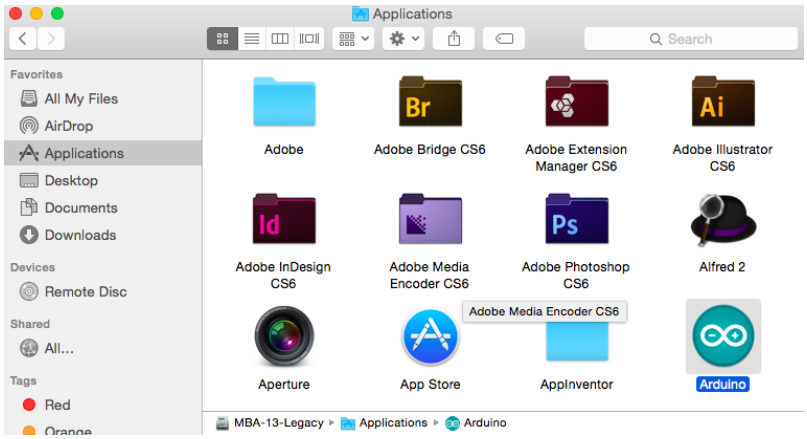
```
/*
 Conditionals - If statement

 This example demonstrates the use of if() statements.
 It reads the state of a potentiometer (an analog input) and turns on an LED
 only if the potentiometer goes above a certain threshold level.

 The circuit:
 * potentiometer connected to analog pin 0.
 Center pin of the potentiometer goes to the analog pin.
 side pins of the potentiometer go to +5V and ground
 * LED connected from digital pin 13 to ground

 * Note: On most Arduino boards, there is already an LED on the board
 connected to pin 13, so you don't need any extra components for this example.

 created 17 Jan 2009
 modified 9 Apr 2012
 by Tom Igoe

This example code is in the public domain.
http://www.arduino.cc/en/Tutorial/IfStatement

 */

// These constants won't change:
const int analogPin = A0;    // pin that the sensor is attached to
const int ledPin = 13;      // pin that the LED is attached to
const int threshold = 400;   // an arbitrary threshold level that's in the range of the analog input

void setup() {
  // initialize the LED pin as an output:
```

Copyright © 2018, Center for Sensorimotor Neural Engineering, University of Washington

```
  pinMode(ledPin, OUTPUT);

}

void loop() {
  // read the value of the potentiometer:
  int analogValue = analogRead(analogPin);

  // if the analog value is high enough, turn on the LED:
  if (analogValue > threshold) {
    digitalWrite(ledPin, HIGH);
  } else {
    digitalWrite(ledPin, LOW);
  }

}
```

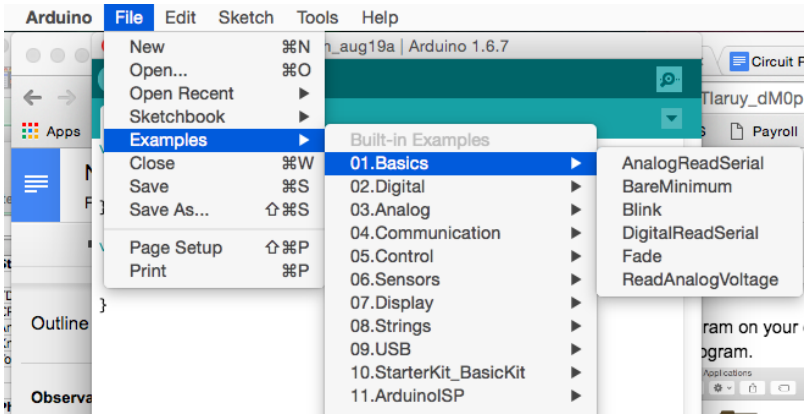
How to use the Arduino program on your computer

1. Open the Arduino Program.

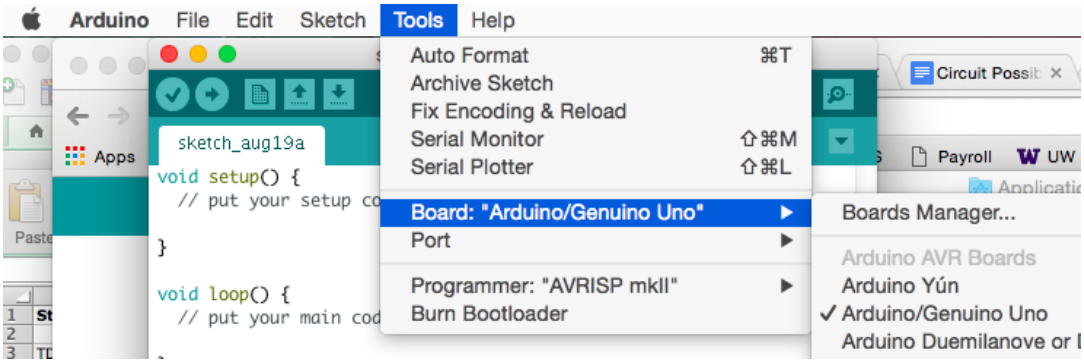


Copyright © 2018, Center for Sensorimotor Neural Engineering, University of Washington

2. Open up an existing sketch.

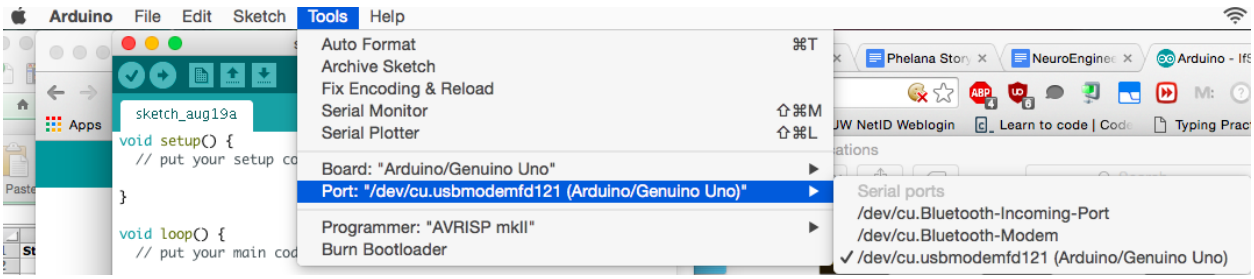


3. Select the correct Arduino Device. We are using the Arduino Uno.



4. Plug in the USB cord to the Arduino and to your computer.

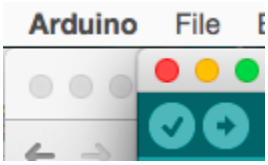
5. Make sure the correct port is connected.



Copyright © 2018, Center for Sensorimotor Neural Engineering, University of Washington

6. Verify your code by clicking on the check button. A message will occur at the bottom of the window to show whether there are any errors in the code.

7. If your code is good to go, you can upload your code by clicking on the right arrow button.



8. Save your code by renaming it if you modified it.

Copyright © 2018, Center for Sensorimotor Neural Engineering, University of Washington

Teacher Resource 1.1: Sample Arduino Sketches

The following can be used to change the LED output based on an input which acts like a switch (on or off based on whether the input is below or above a certain threshold).

```
/*
 Conditionals - If statement
 This example demonstrates the use of if() statements.
 It reads the state of a potentiometer (an analog input) and turns on an LED
 only if the potentiometer goes above a certain threshold level

 The circuit:
 * potentiometer connected to analog pin 0.
 Center pin of the potentiometer goes to the analog pin.
 side pins of the potentiometer go to +5V and ground
 * LED connected from digital pin 13 to ground

 created 17 Jan 2009
 by Tom Igoe
 modified 12 Aug 2016
 by Phelana Pang, with the help of Lefteris Kampianakis
 */

// These constants won't change:
const int analogPin = A0;    // pin that the sensor is attached to
const int ledPin = 13;       // pin that the green LED is attached to
const int redPin = 12;       // pin that the red LED is attached to
const int threshold = 500;   // threshold value

//this function will run first
void setup() {
  // initialize the LED pin as an output:
  pinMode(ledPin, OUTPUT);
  pinMode(redPin, OUTPUT);
}

//then this function will run all the time looping
void loop() {
  // read the value of the potentiometer:
  int analogValue = analogRead(analogPin);

  // if the analog value is high enough, turn on the LED:
  if (analogValue > threshold) {
    digitalWrite(ledPin, HIGH);
  } else {
    digitalWrite(ledPin, LOW);
  }
  if (analogValue < threshold) {
    digitalWrite(redPin, HIGH);
  } else {
```

Copyright © 2018, Center for Sensorimotor Neural Engineering, University of Washington

```
   digitalWrite(redPin, LOW);
 }
}
```

The following has 3 LEDs as output, and the output depends on whether the input is detected in a low, medium, or high range:

```
/*
 Conditionals - If statement
 This example demonstrates the use of if() statements.
 It reads the state of a potentiometer (an analog input) and turns on a certain color LED
 depending on the range in which the potentiometer reads.
The potentiometer can be substituted with different input sensors (pressure sensor, proximity sensor, photoresistor).

 The circuit:
 * potentiometer connected to analog pin 0.
 Center pin of the potentiometer goes to the analog pin.
 side pins of the potentiometer go to +5V and ground
 * LED connected from digital pin 13 to ground

 created 17 Jan 2009
 by Tom Igoe
 modified 12 Aug 2016
 by Phelana Pang, with the help of Lefteris Kampianakis
 */


// These constants won't change:
const int analogPin = A0;    // pin that the pressure sensor is attached to
const int redPin = 13;       // pin that the red LED is attached to
const int yedPin = 12;       // pin that the yellow LED is attached to
const int gedPin = 11;       // pin that the green LED is attached to
const int threshold1 = 200;   // low threshold level that's in the range of the analog input
const int threshold2 = 400;   // med threshold level that's in the range of the analog input

//this function will run first
void setup() {
  // initialize the LED pin as an output:
  pinMode(redPin, OUTPUT);
  pinMode(yedPin, OUTPUT);
  pinMode(gedPin, OUTPUT);
}
```

Copyright © 2018, Center for Sensorimotor Neural Engineering, University of Washington

```
//then this function will run all the time looping
void loop() {
 // read the value of the pressure sensor:
 int analogValue = analogRead(analogPin);

 // depending on how hard pressure sensor is pressed, green, yellow, or red LED will turn on:
 if (analogValue < threshold1) {
  digitalWrite(redPin, HIGH); // if pressure is low, red LED is on
  digitalWrite(yedPin, LOW); // if pressure is low, yellow LED is off
  digitalWrite(gedPin, LOW); // if pressure is low, green LED is off
 }
 else if (analogValue >= threshold1 && analogValue <= threshold2) {
  digitalWrite(redPin, LOW); // if pressure is medium, red LED is off
  digitalWrite(yedPin, HIGH); // if pressure is medium, yellow LED is on
  digitalWrite(gedPin, LOW); // if pressure is medium, green LED is off

 } else if (analogValue > threshold2) {
  digitalWrite(redPin, LOW); // if pressure is high, red LED is off
  digitalWrite(yedPin, LOW); // if pressure is high, yellow LED is off
  digitalWrite(gedPin, HIGH); // if pressure is high, green LED is on
 }

 }
```

Copyright © 2018, Center for Sensorimotor Neural Engineering, University of Washington