[This document was copy-pasted from the Introduction to Arduino Slides… it may be stale]

**Arduino Microcontrollers**

"Open Source" means that all of the electronic specifications and source code are publicly available. You are welcome to use, improve, and modify an open source project for any (legal) use. With respect to Arduino, the hardware is documented and posted for all to see. All of the source code for all of the libraries that come with it are publically available. The Arduino software is Free Open Source (FOSS) which of course means that you don't have to pay to use it. It's important realize that at its heart, "open source" is really nothing more than a legal license that the copyright holder distributes a product with. But the copyright holder often puts some restrictions on the user; for instance, releasing the copyright holder from liability due to damages or prohibiting someone from taking the open source project and creating a closed source modification. Examples of Open Source Licenses include the MIT License and GNU General Public Use (GPL) license. Arduino hardware is distributed under under the CC BY-SA license whereas the software is licenced under GNU Lesser GPL and GNU GPL.

Microcontrollers are a lower-end CPUs with slower clock speeds, small memories, and limited instruction sets. Unlike a regular computer, they do not run a full operating system. Instead, they run one program (usually) that is created for a very specific purpose. Because of their limited functionality, they are usually require less power and are cheaper than microcomputers. In contrast, a Raspberry Pi is a microcomputer. It is physically about the same size as an Arduino but it is faster, with more memory, and runs a full operating system.

Microcontrollers are often used in embedded devices such as toasters, automobiles, thermostats, etc. etc. If you have an electronic device that has a specific purpose, it probably has a microcontroller for a brain.

Arduino devices range from $20 to $50 depending on the size and capabilities. This is the price of the microcontroller board itself, but Arduino's have many hardware add-on components that can be upwards of hundreds of dollars. Some examples of expensive add-ons might be a high-quality thermal camera or a fancy display. Really, you can connect them to anything… as long as the Arduino microcontroller has enough processing power to drive the device.

**References**

[Show these references in Canvas]

**Mega 2560 Hardware**

5V/3.3V/GND pins

Analog pins

Digital pins

PWM pins

TX/RX pins

Talk about how some pins have dual use. For example, the builtin LED is pin 13, which is also a digital/PWM. This is helpful if you want a quick visual debug of the voltage being output on pin 13. But it might also be a distraction. The only bad thing is if you're trying to write one thing to the LED and output another thing to the pin… the two components will be fighting each other.

If anyone asks about specs: 16 MHz processor, 256k program memory on the flash chip (8k used), 8k memory,...

**The Arduino IDE**

Demonstrate on Chromebook if possible

**Connecting with Chromebooks**

Show how to install the Linux Subsystem, bring up a terminal window, and download/run the Arduino IDE.

Reset the default Arduino IDE directory to be lower-case arduino (or whatever)

Download the sample code and user guide from Canvas

Navigate the file system using basic Linux commands

Connect the USB device and show how to select in the Arduino IDE

**Arduino Sketches**

Show this slide and let the students look at the code on their own computer.

**Blink Example**

The example sketches are read-only, they'll have to save any changes to a new sketch.

**Physics of Electricity**

Show some free YouTube videos

**Sample Project**

#include <SimpleDHT.h>

#define DHT11\_PIN 2

SimpleDHT11 dht11(DHT11\_PIN);

void setup()

{

Serial.begin(57600);

Serial.println("=========================");

}

void loop()

{

Serial.println("Querying DHT11…");

byte temperature = 0;

byte humidity = 0;

int err = dht11.read(&temperature, &humidity, NULL);

if (err == SimpleDHTErrSuccess)

{

Serial.print((int)temperature); Serial.print(" \*C, ");

Serial.print((int)humidity); Serial.println(" %H");

}

delay(5000);

}

**Power Sources**

Point out how the board can be powered. Mention that it can be powered through the pins but there's you have to be careful because you can easily damage the board. Please don't.

**Summary of Arduino Microcontrollers**

Open-source

Power, slow, limited

Single, embedded

Mega

Analog, digital, PWM

C++

USB serial, IDE