Lab 2: Analog to Digital Circuit Interfaces

EECS 16B Fall 2022

Slides: links.eecs16b.org/lab2-slides

Logistics: Makeups/Extensions

- **Makeup:** you need to attend a different lab section to finish the lab on time
  - Sign up at https://makeup.eecs16b.org
  - Only one group member needs to sign up
  - Labs are due by the end of your next section
- **Extension:** you need additional time to complete the lab
  - Same form as HW Extensions: https://eecs16b.org/extensions
  - Without an extension, late labs are 50% credit

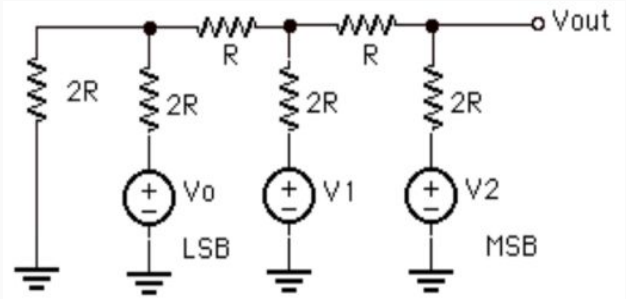
Lab 2 Overview: DAC and ADC

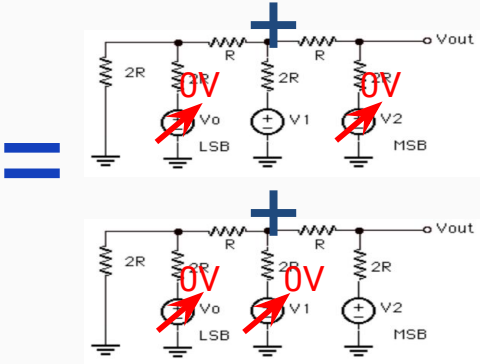
- DAC = Digital to Analog Converter
- ADC = Analog to Digital Converter
- Real world is continuous, but computers need to store data digitally
  - Need to find a way to convert between analog and digital for signals
  - ~~EE 123 discusses consequences of digitally sampling analog signals, EE 140 discusses the design of DACs/ADCs~~
- DAC/ADC in your life:
  - DAC for MP3 players, analog TVs, video on cell phones
  - ADC for sound/video recording
  - VoIP (voice over IP) uses both!

~~Can you think of a few more examples?~~

Review: Superposition

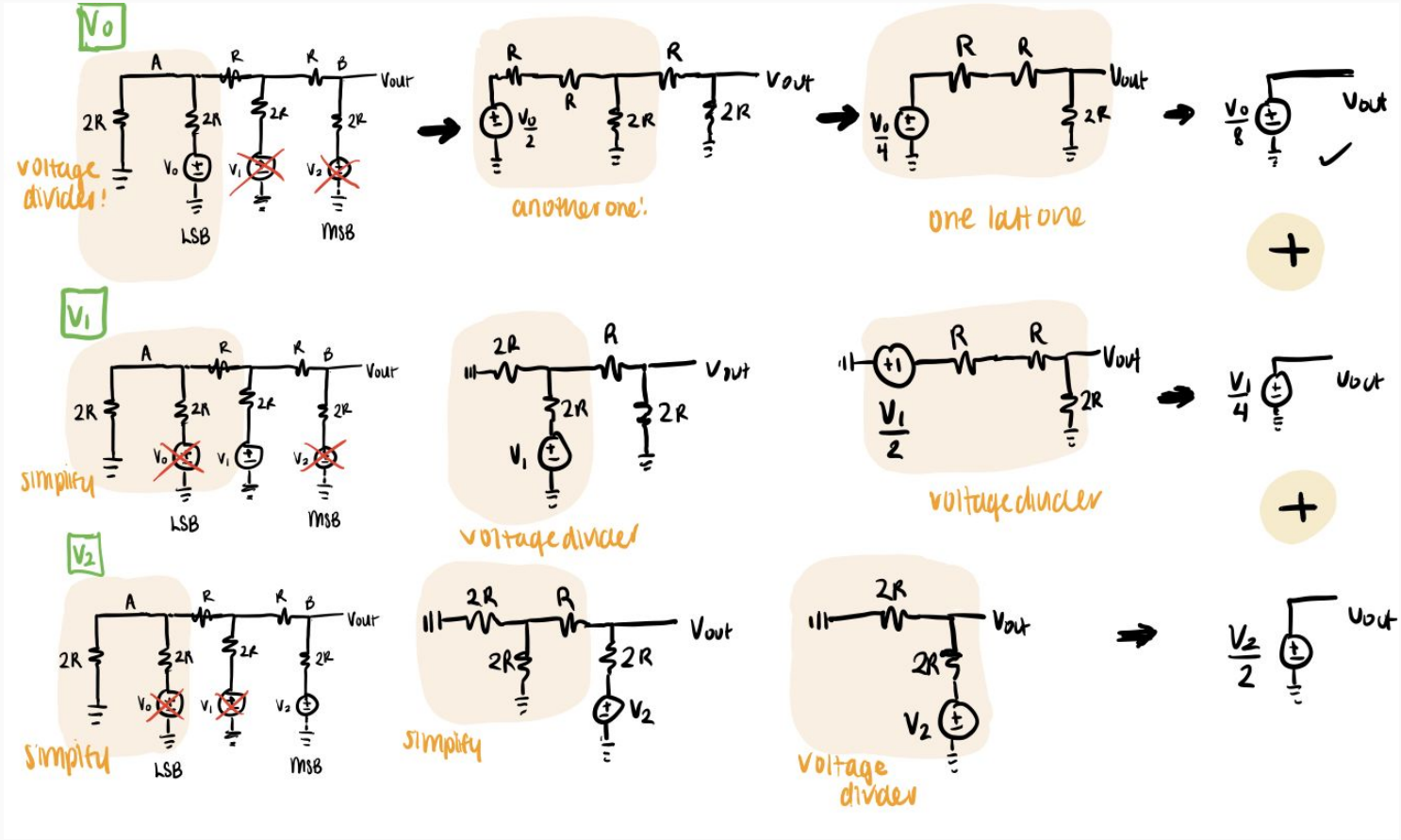
- Since resistive circuits are linear, we can apply the principle of superposition:
  - Treat each source independently – zero out all but one
  - The total effect is the sum of the effects of each source
- Example:





$$Vout = Vout_1 + Vout_2 + Vout_3$$

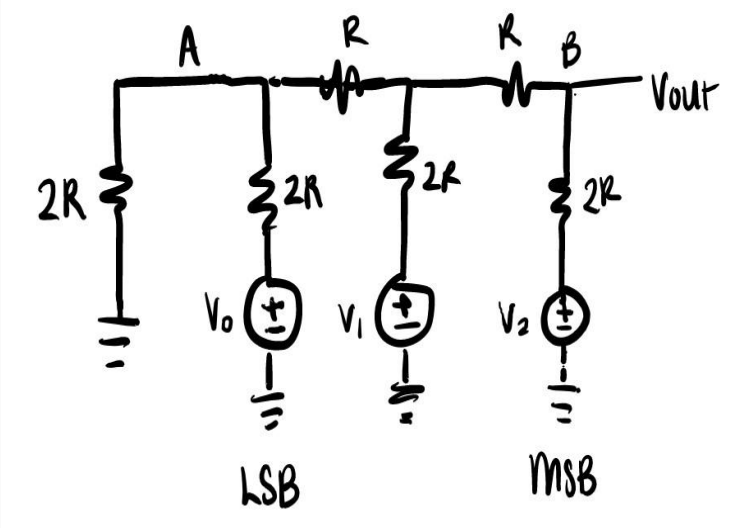
Review: Superposition Example



Review: Superposition Example

$$V_{out} = \frac{V_0}{8} + \frac{V_1}{4} + \frac{V_2}{2}$$

V_0: least significant bit; V_2: most significant bit



Treat each input as a bit. 3 bits represents 0 to $2^3 - 1 = 7$, [0, 7]

you can set $V_0 = V_1 = V_2 = 8V$ and it'll give you → 0b111

$$V_{out} = \frac{V_0}{8} + \frac{V_1}{4} + \frac{V_2}{2} = 1 + 2 + 4 = 7$$

"turn off" $V_0 = 0V$ → 0b110

$$V_{out} = \frac{V_0}{8}^{0} + \frac{V_1}{4} + \frac{V_2}{2} = 0 + 2 + 4 = 6$$

"turn on" $V_0 = 8V$, "turn off" $V_1 = 0V$ → 0b101

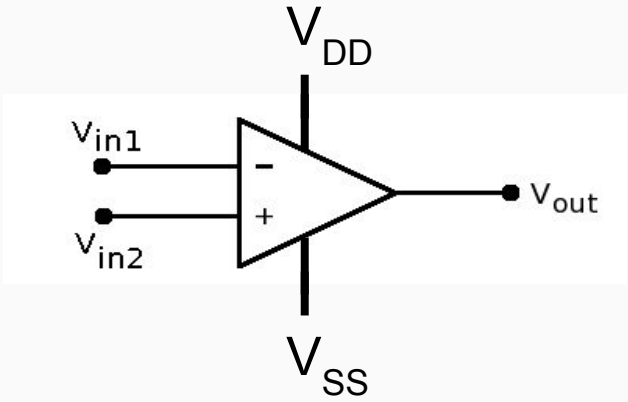
$$V_{out} = \frac{V_0}{8} + \frac{V_1}{4}^{0} + \frac{V_2}{2} = 1 + 0 + 4 = 5$$

we can keep going, counting down to 0.

$0b100 = 4_{10}$, $0b011 = 3_{10}$, $0b010 = 2_{10}$, $0b001 = 1_{10}$, $0b000 = 0_{10}$

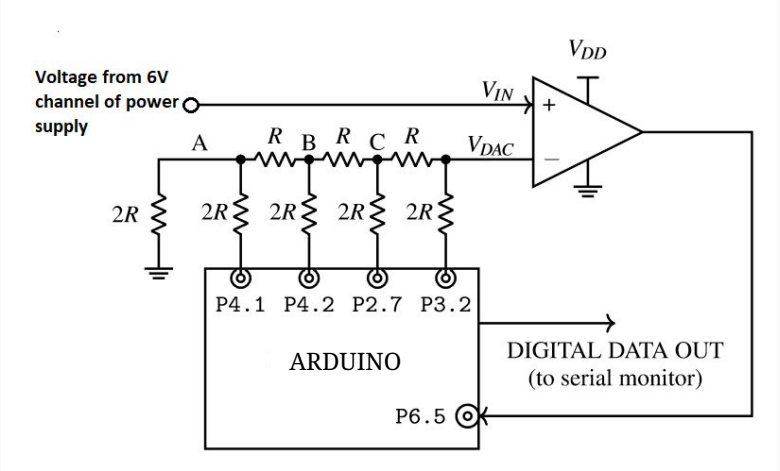
Review: Comparators

- A device that compares two voltages (or currents) and outputs a digital signal to indicate which is larger
- Op-amp Implementation:
  - If $V_{in2} > V_{in1}$, V_{out} goes to VDD
  - If $V_{in1} > V_{in2}$, V_{out} goes to VSS
  - (think: if V_{out} is connected to V^-, its value will bring V^- closer to V^+)
- NOTE: Arduinos use 5V pin logic
  - VDD = 5 V
  - VSS = 0 V



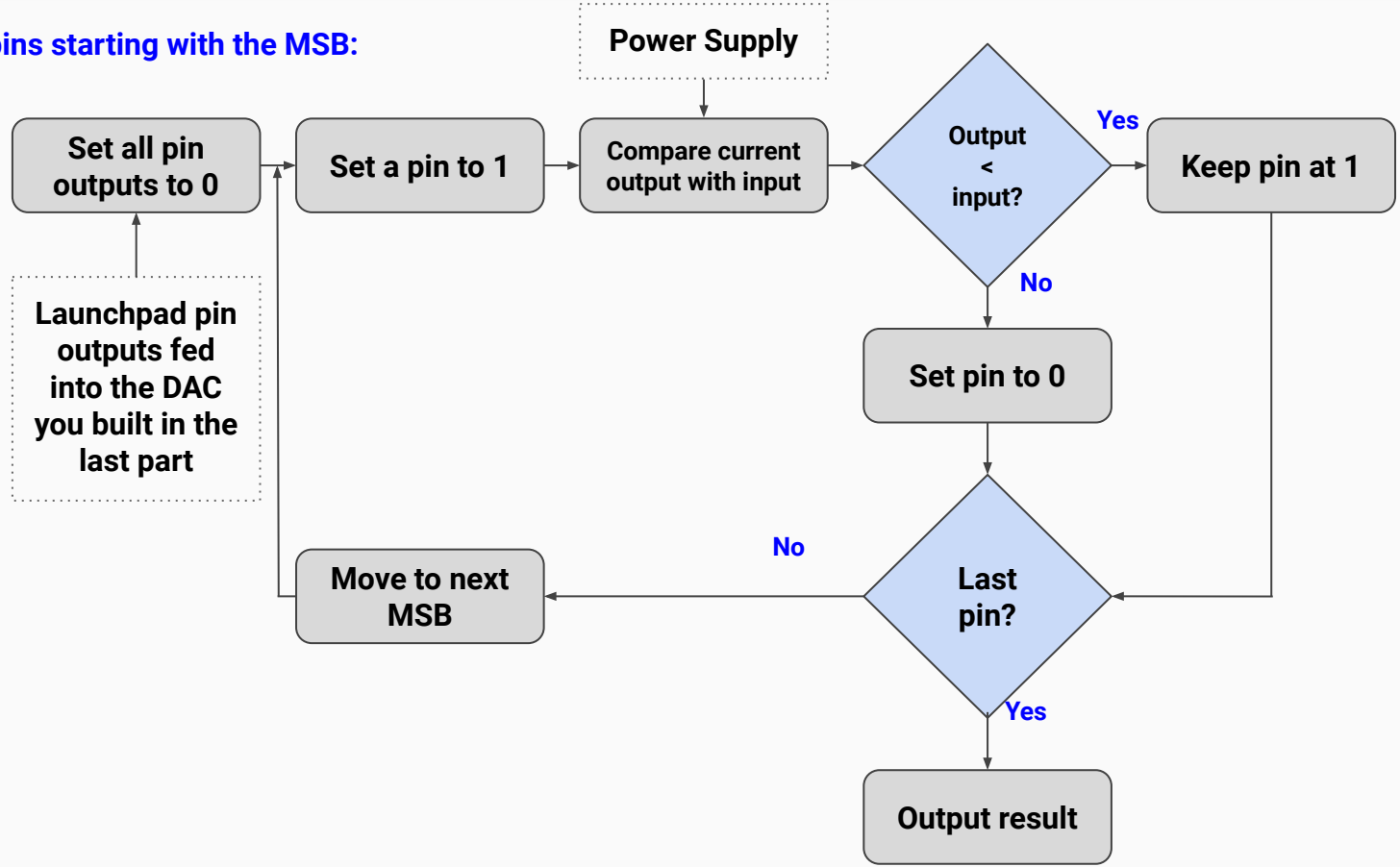
Review: ADC

- ADC - The Arduino uses binary search when turning on MSB (most significant bit) to LSB (least significant bit) and comparing the resulting V_{DAC} with V_{in}

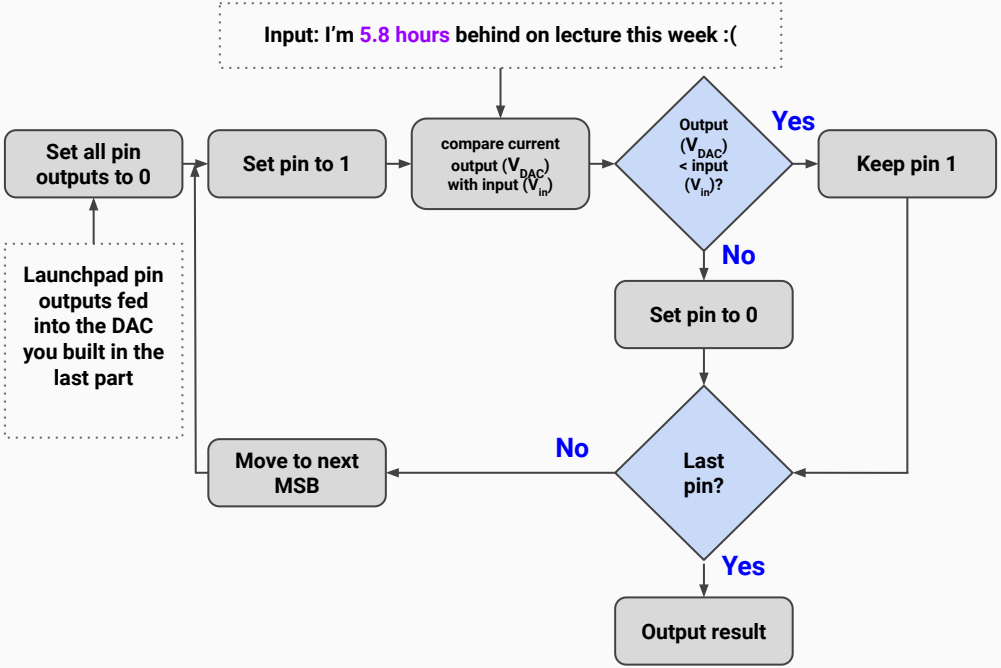


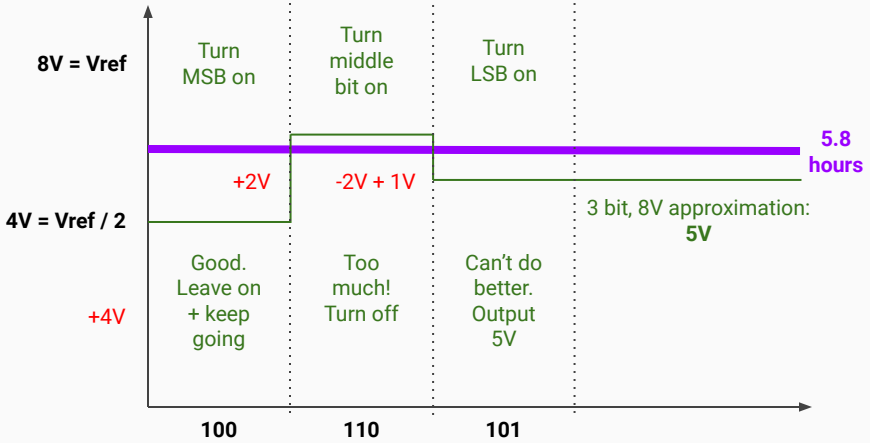
Successive Approx. Register ADC

Loop over all pins starting with the MSB:



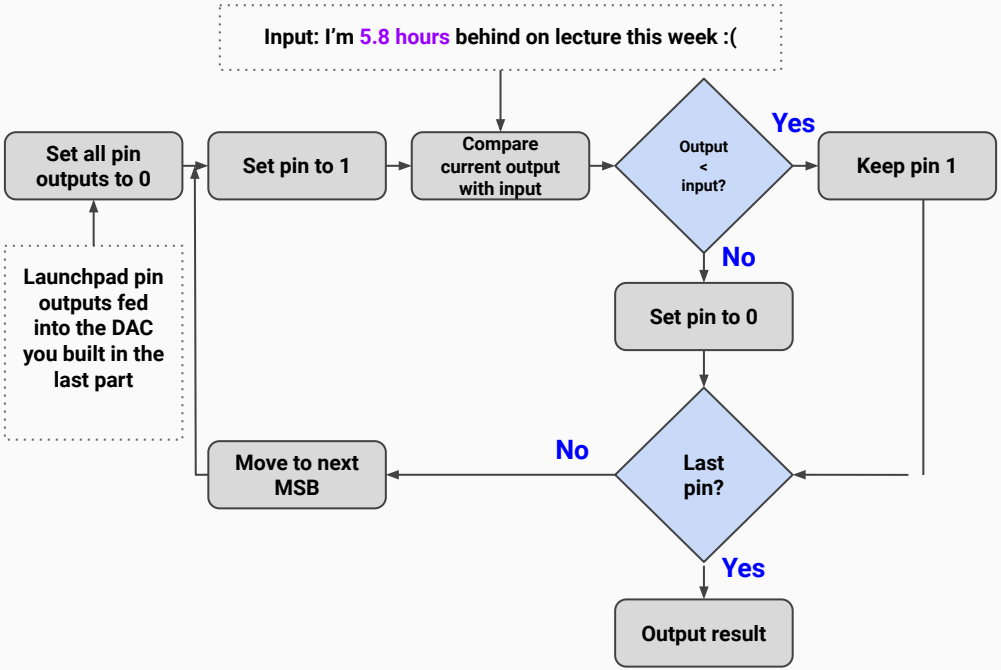
Successive Approx. Register ADC





Result: We forget to account for the last 0.8 hour of lecture because our 3 bits cannot represent it. :(

Successive Approx. Register ADC



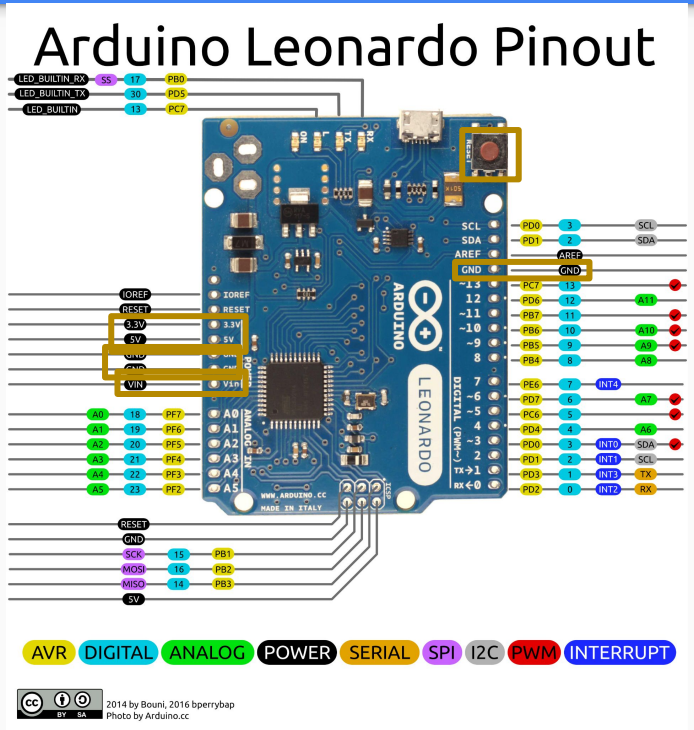
We're trying to match our 5.8V input:

- Vref = 8V, we have 3 bits

1. **Turn on MSB: 1 0 0**
   4V < 5.8V, keep going, keep bit on
1. **We can do better: 1 1 0**
   (4V + 2V) > 5.8V, too much, turn off
1. **Try the next pin (LSB): 1 0 1**
   (4V + 1V) < 5.8V
1. **That's all folks, we're out of bits**
   Output: 5V

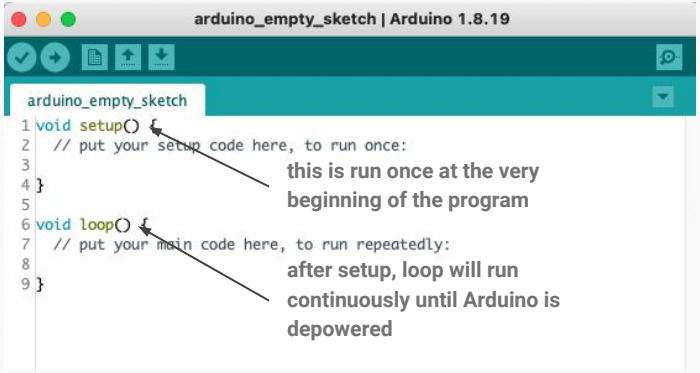
Result: We forget to account for the last 0.8 hour of lecture because our 3 bits cannot represent it. :(

Introduction to Arduinos



- There are 4 main “Pin Modes”
- Digital: High (5V) or Low (0V)
  1. Digital Output
  2. Digital Input
- Analog: range from 0-5V
  3. Analog Output: mapped from 0 - 255
  4. Analog Input: mapped to 0 - 1023

Introduction to Arduinos



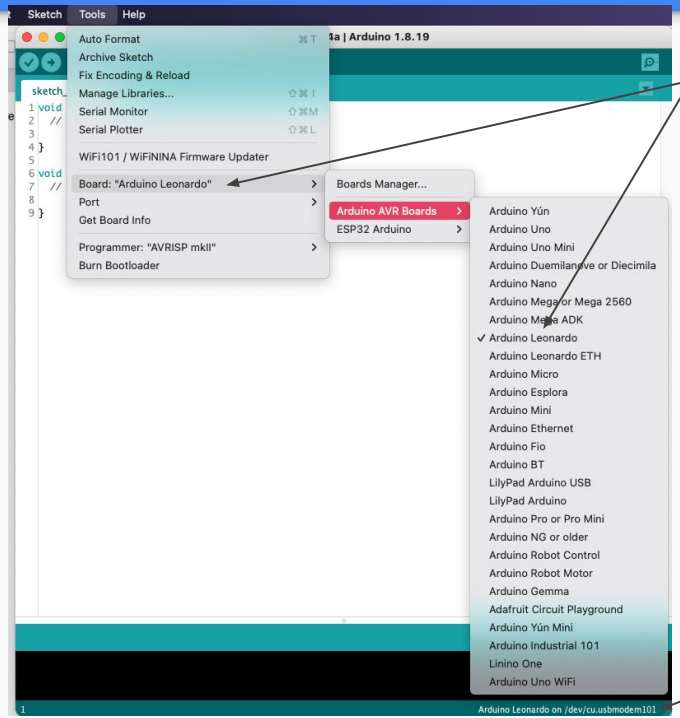
Note: Arduino is programmed in **C** via the [Arduino IDE]
(pre-installed on lab computers)

- Code is uploaded from computer to Arduino via microUSB port
- If powered, code is ALWAYS running
  - RST -> restart
  - Unpowering and powering Arduino -> begin re-running whatever was last uploaded
- If you find this to be an issue, the easiest solution is to upload a blank program

Quick Rundown: Arduino vs Launchpads

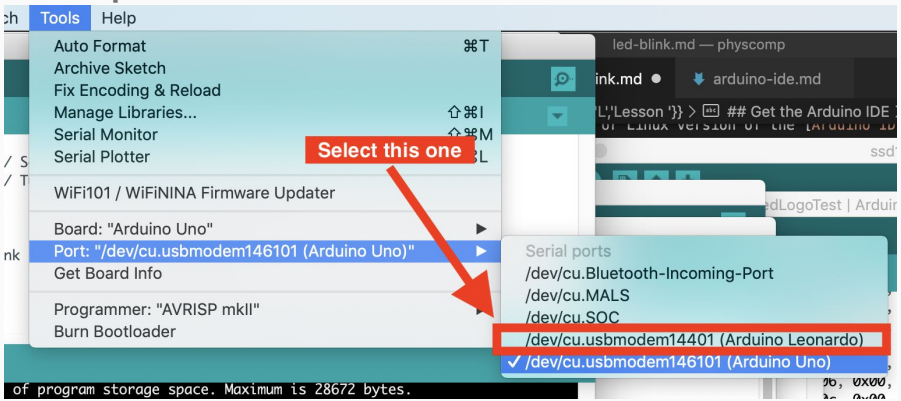
- Launchpads operate on 3.3V logic while Arduinos operate on 5V logic
  - However, for most labs other than this one, we will be keeping our circuits operating at 3.3V for stability reasons.
- External Power: the Launchpad can take 5V as an input to its *5V* pin, while Arduino requires 7-12V as an input to its *V_in* pin.
  - safe to power the Arduino via both the micro-USB and V_in at the same time
  - Launchpads… however… go bakoom
- You actually see the word Arduino outside of 16B, when did you ever see the word “Launchpad” other than complaints about 16B

Uploading Code to Arduino



Ensure this says *Arduino Leonardo*, otherwise select it

- PORT selection
- Upload button



Arduino **should** auto-detect your port

(works 100% of the time 25% of the time)

Arduino Logistics

- Arduinos will be passed out during lab today
- **Arduinos are property of 16B and have to be returned to us by the end of the semester**
- Each Arduino will have a unique ID
  - Fill out https://eecs16b.org/lab-groups once you've received your Arduino

General Reminders/Habits

- connect all grounds together, including the Arduino GND pin (any works)
- In general, avoid having voltage/currents going into your Arduino if your Arduino isn't already powered
- Check that your probes are working by probing a known voltage value
  - Ie 5V/3.3V/GND from power supply
- PLEASE CLEAN UP AFTER YOURSELF!! Put probes back, pack up kits, etc.
- Don't unplug computers

Let’s get into it!

Important Forms/Links

- Help request form: https://eecs16b.org/lab-help
- Checkoff request form: https://eecs16b.org/lab-checkoff
- Extension Requests: https://eecs16b.org/extensions
- Slides: links.eecs16b.org/lab2-slides
- Anon Feedback: https://eecs16b.org/lab-anon-feedback
- https://eecs16c.org