Lab 2: Analog to Digital Circuit Interfaces

EECS 16B Spring 2024

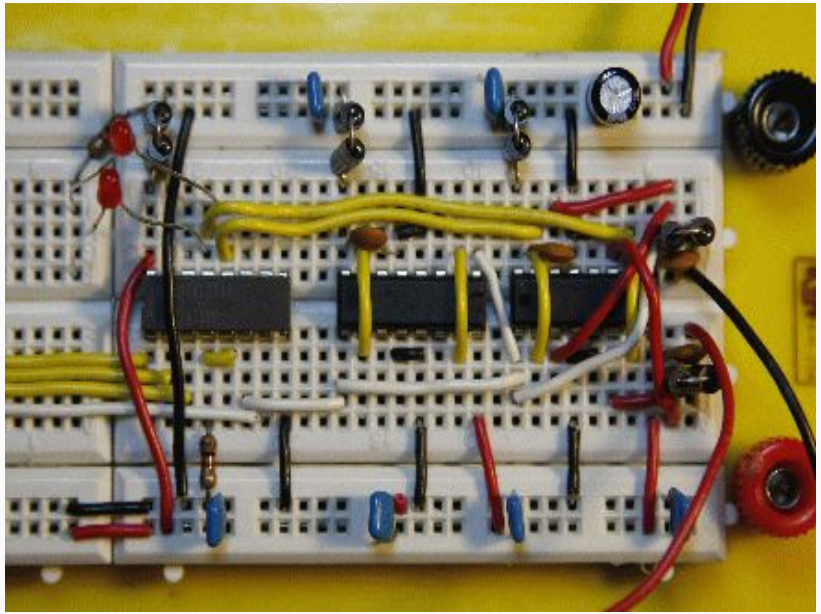
Slides: links.eecs16b.org/lab2-slides

Logistics: Groups

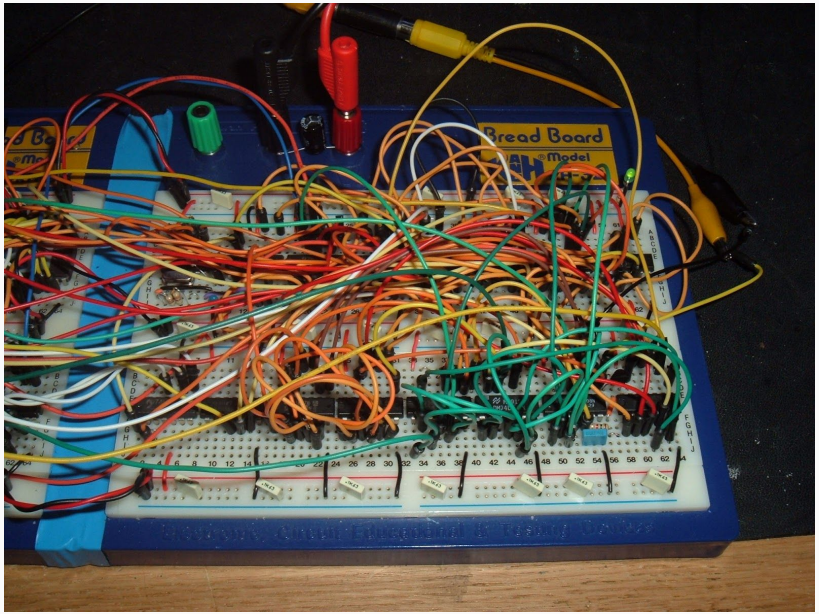
- **Fill out the Lab Group Form** (necessary to receive an Arduino): https://eecs16b.org/lab-groups

Planar Circuits

Planar

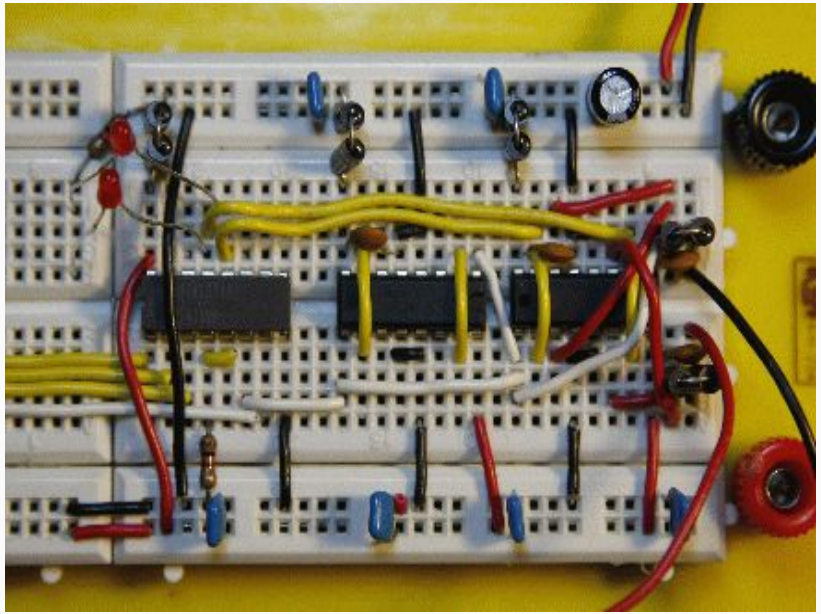


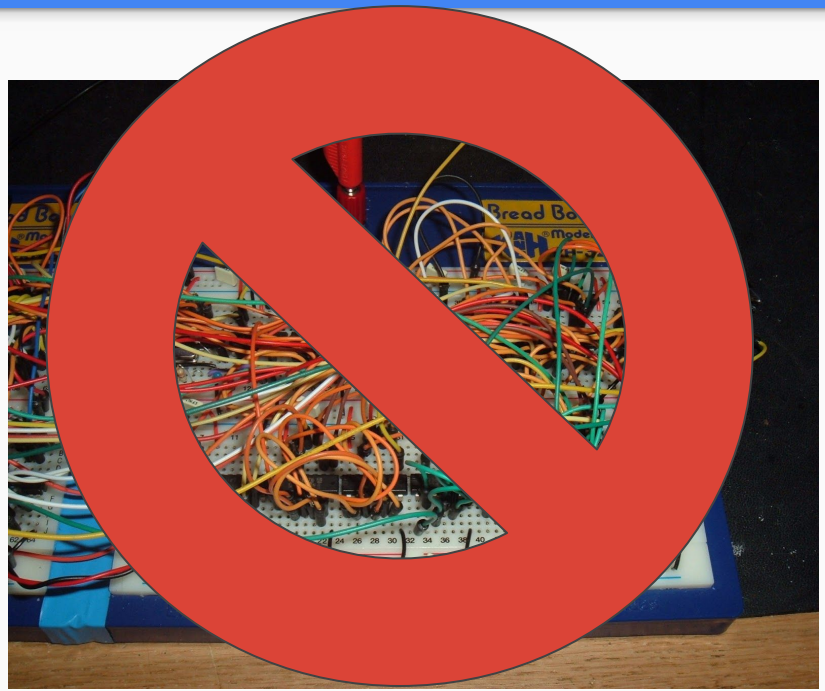
Non-planar



Planar Circuits

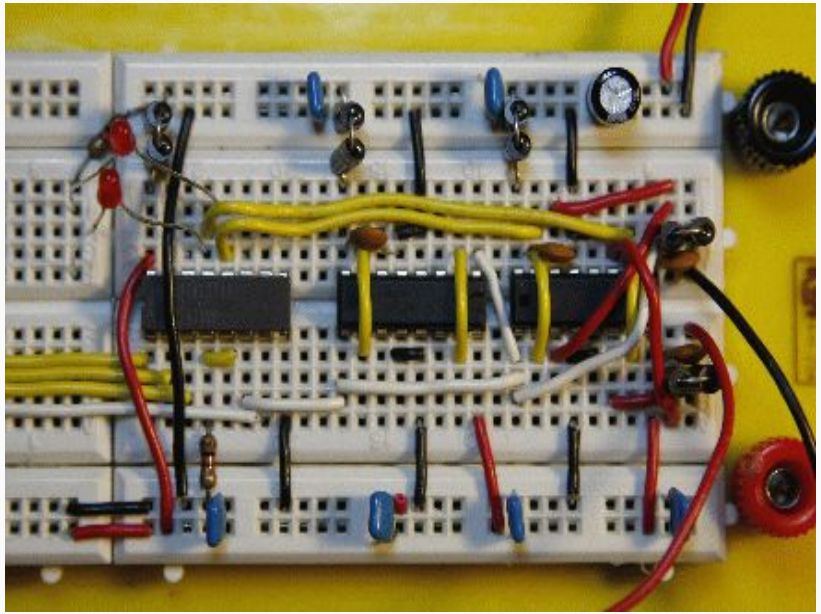
Planar





Planar Circuits

Planar



- Planar Circuits are now **a requirement** to:
  - Get help from staff
  - Get checked off
- Tips to create planar circuits:
  - Plan your circuit ahead of time
  - Trim wires to length
  - Cut components to length

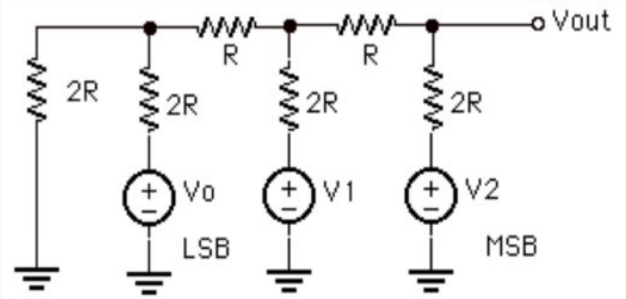
We will ask you to redo your circuit if it is nonplanar

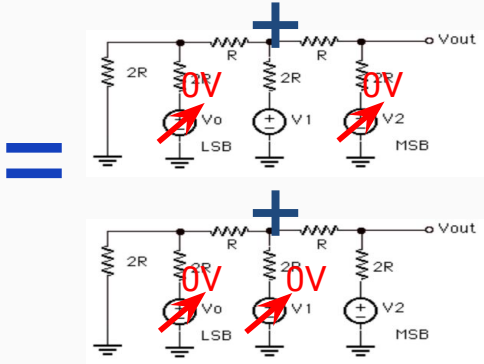
Lab 2 Overview: DAC and ADC

- 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!

DAC Review: Superposition

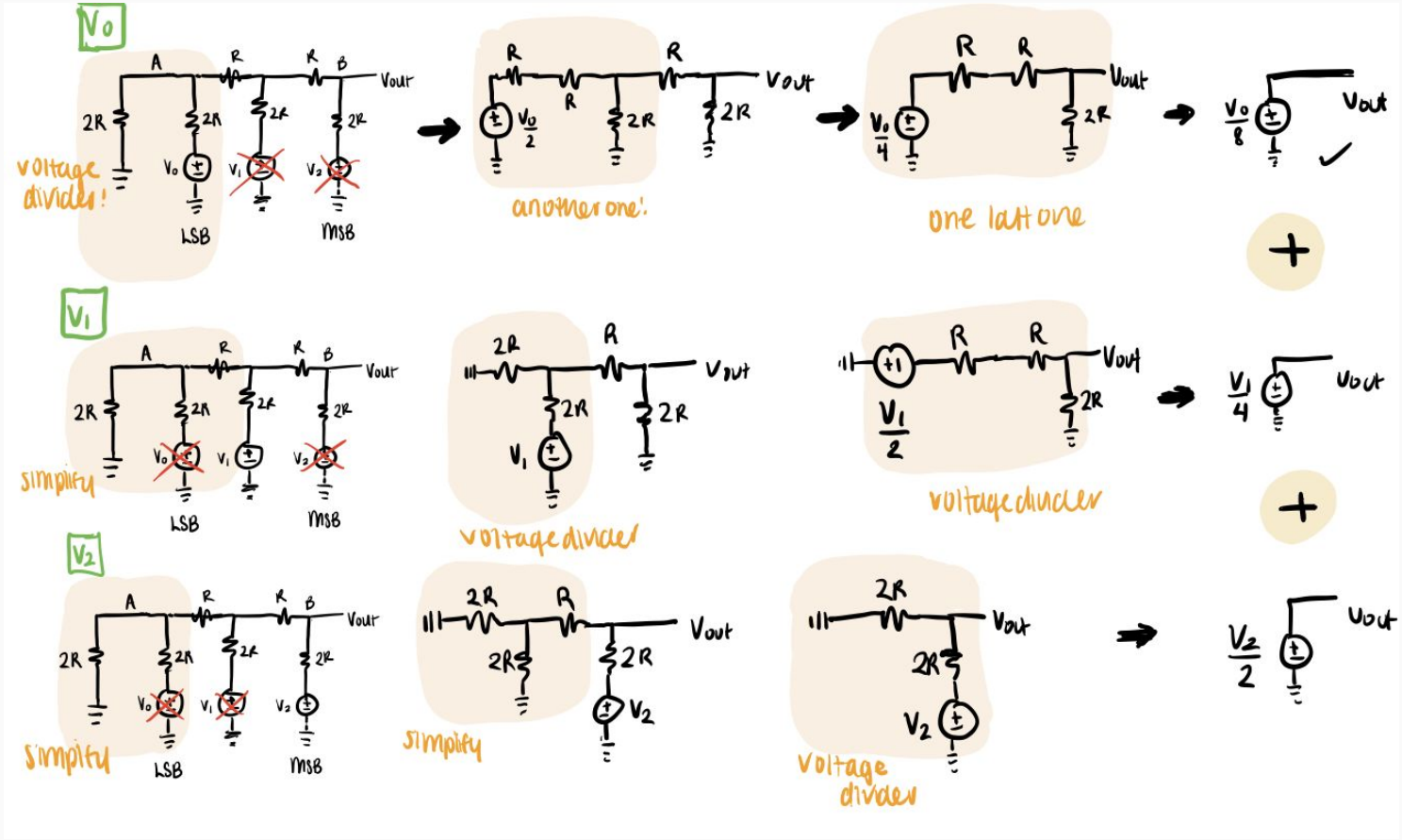
- Since resistive circuits are linear, we can apply the principle of superposition:
  - Treat each source independently – zero out all but one
  - The total effect is the sum of the effects of each source
- Example:





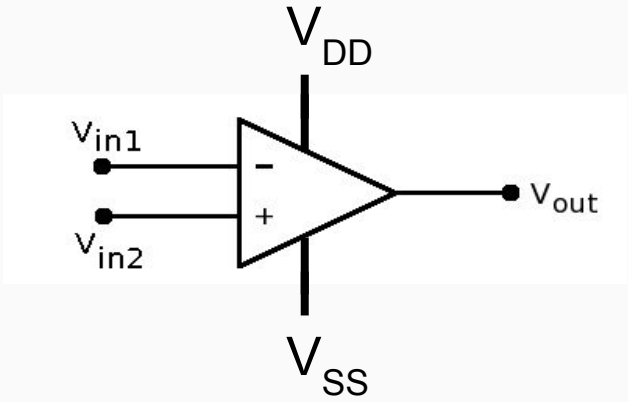
$$Vout = Vout_1 + Vout_2 + Vout_3$$

DAC Review: Superposition Example



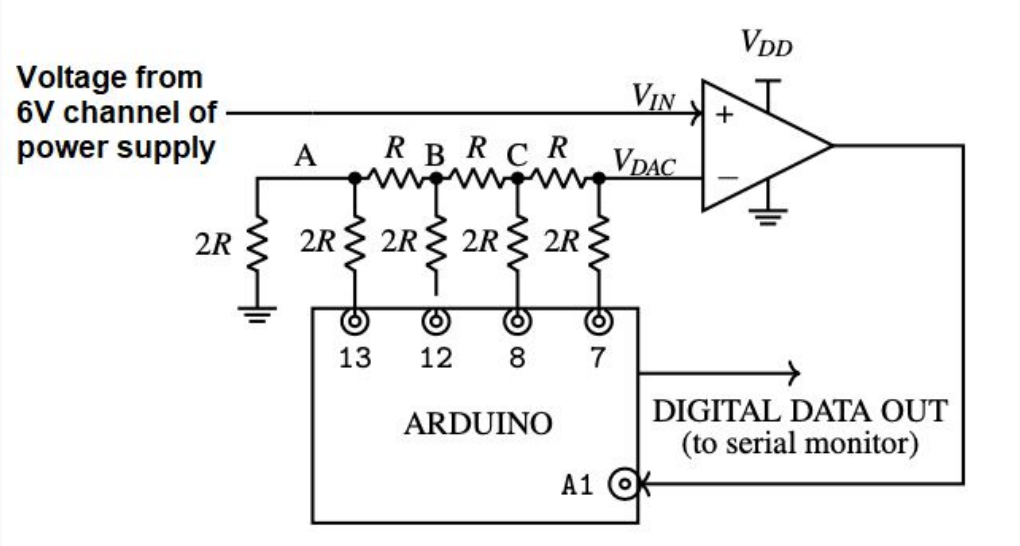
ADC Review: Comparators

- 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



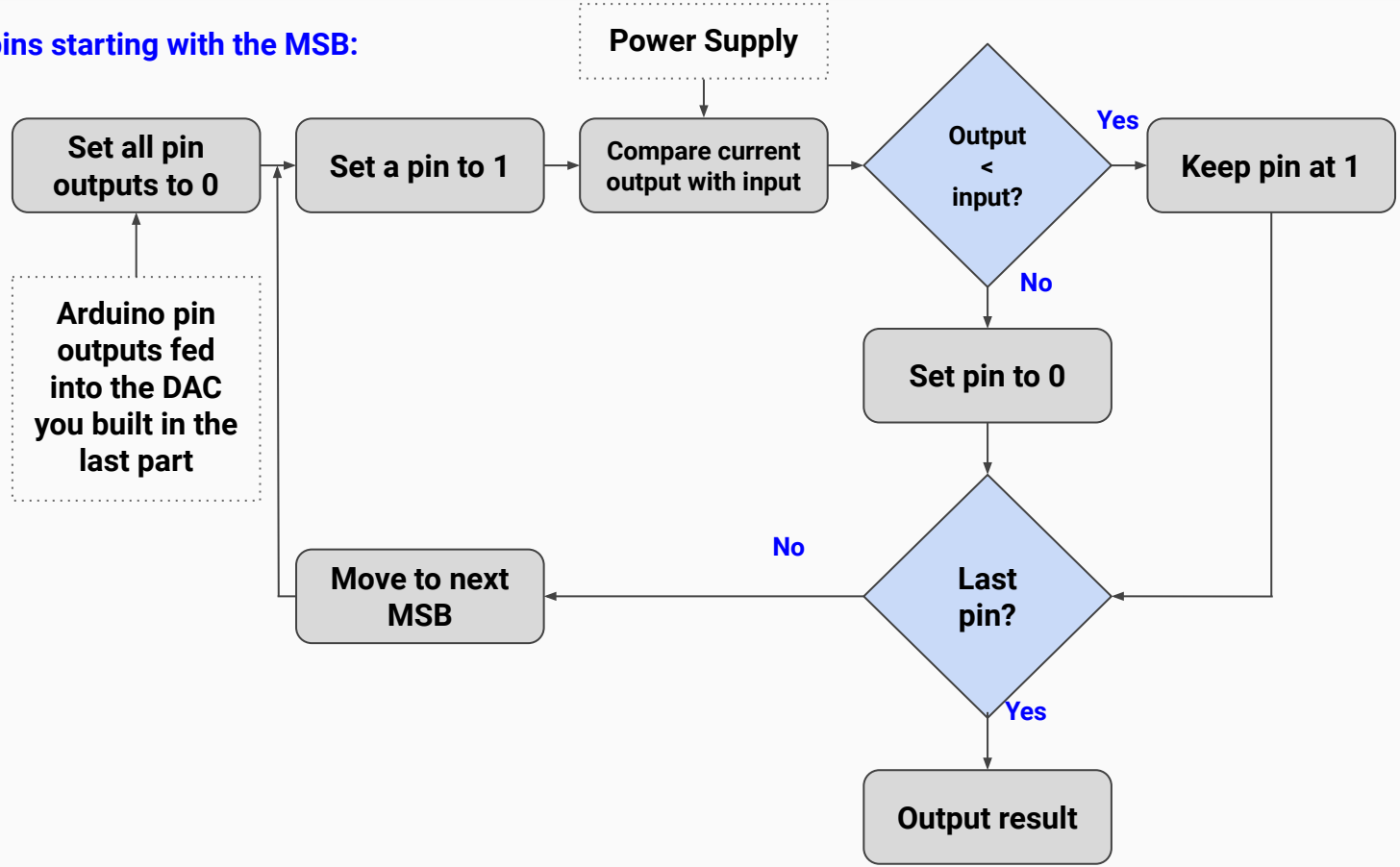
Review: ADC

- 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}

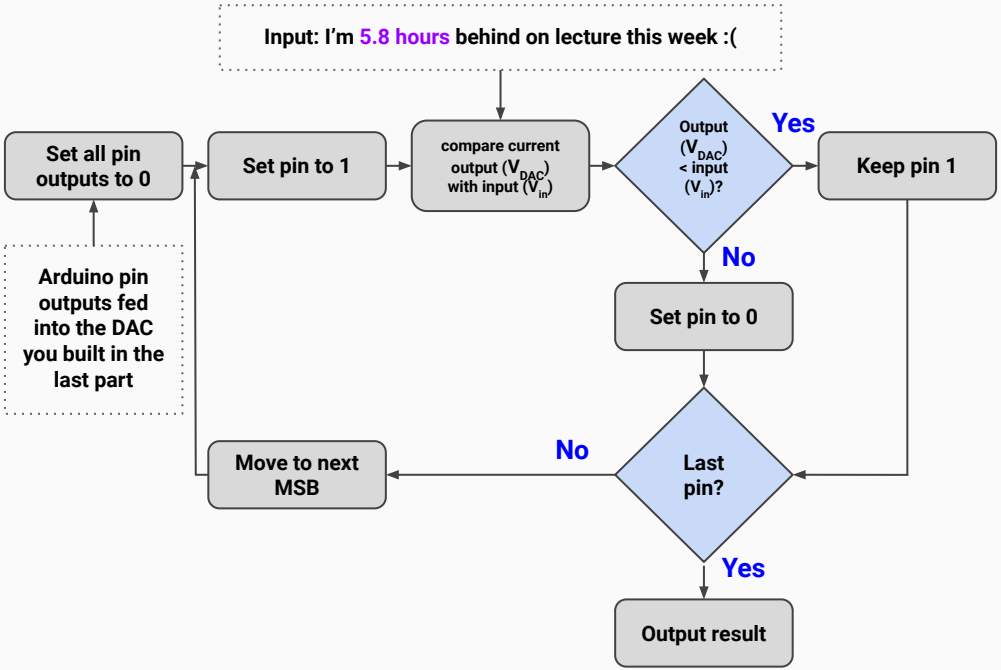


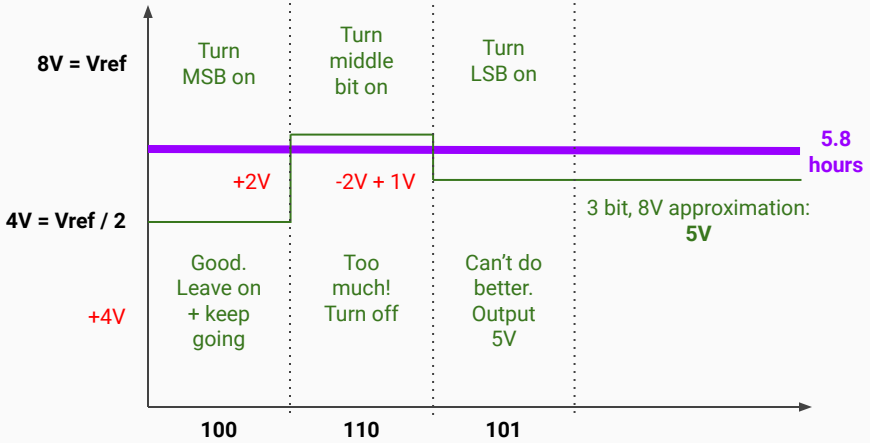
Successive Approx. Register ADC

Loop over all pins starting with the MSB:



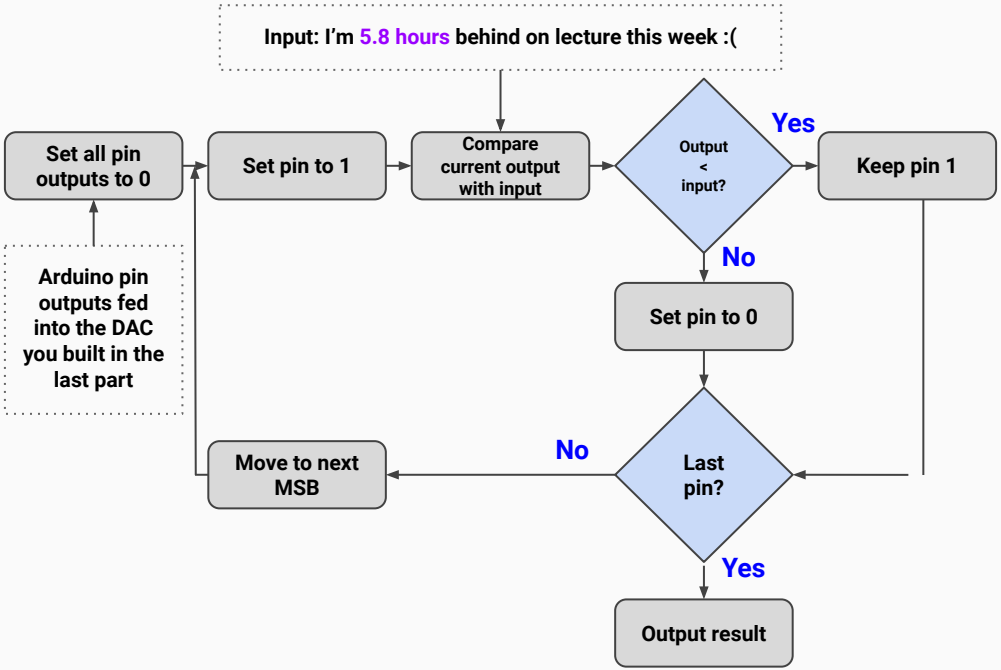
Successive Approx. Register ADC





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

Successive Approx. Register ADC



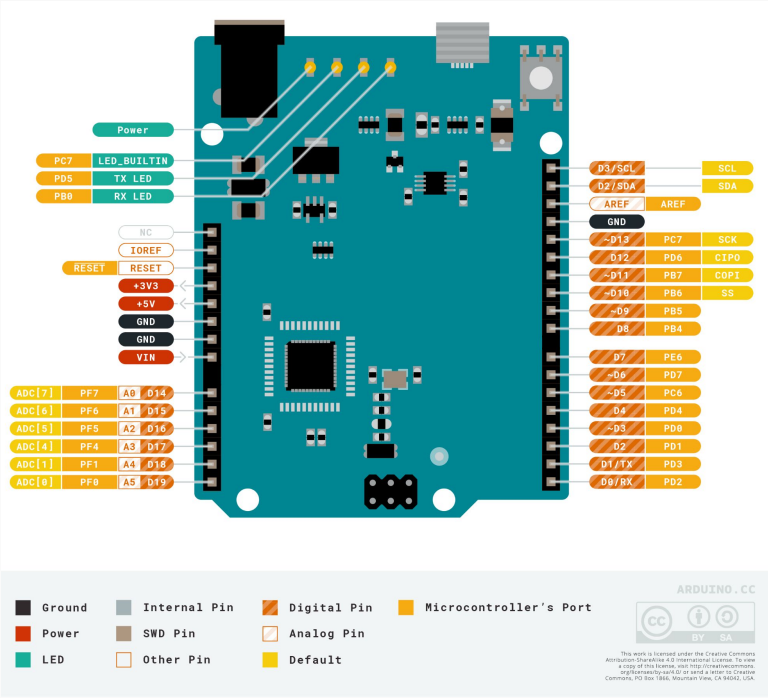
We're trying to match our 5.8V input:

- 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

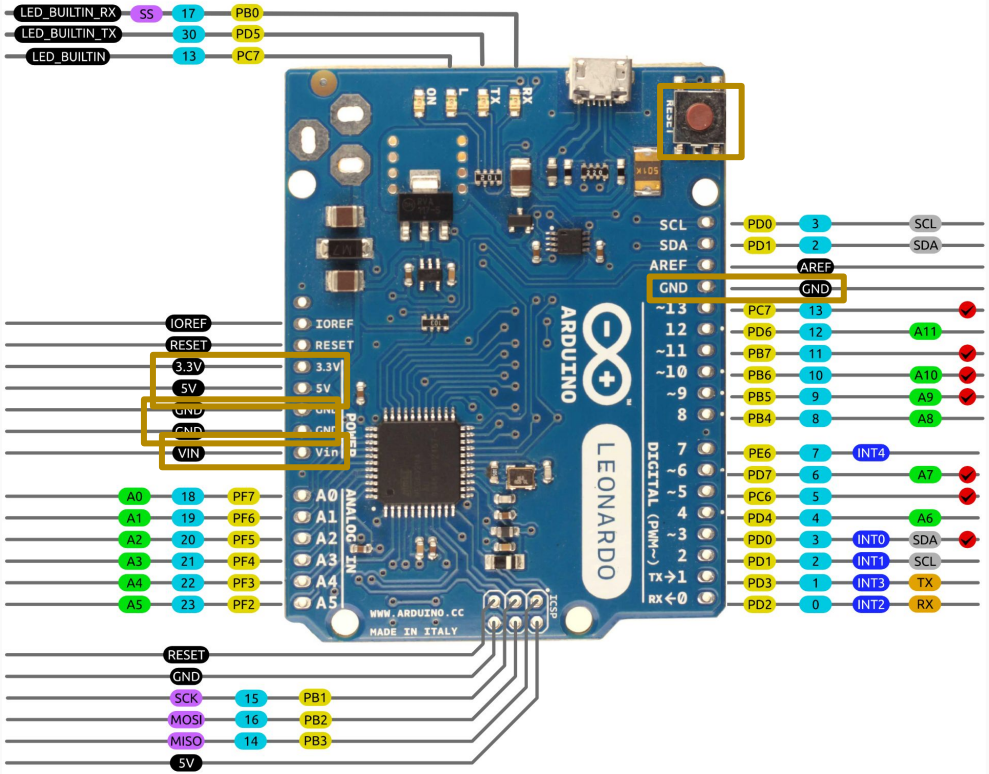
Result: We forget to account for the last 0.8 hour of lecture because our 3 bits cannot represent it. :(

Introduction to Arduinos



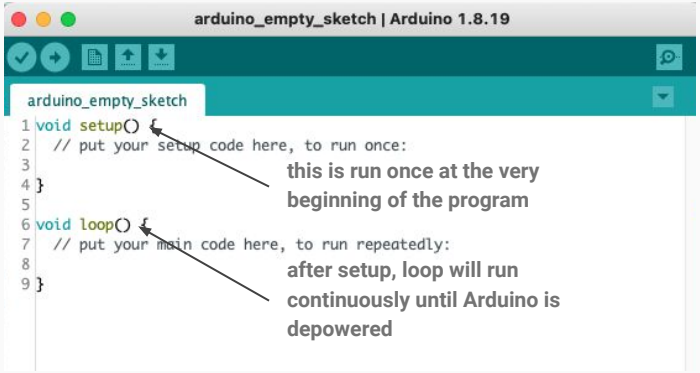
- There are 4 main “Pin Modes”
- Digital: High (5V) or Low (0V) [1s and 0s]
  1. Digital Output
  2. Digital Input
- Analog: range from 0-5V [numerical values]
  1. Analog Output: mapped from 0 - 255
  2. Analog Input: mapped to 0 - 1023

Arduino Leonardo Pinout



AVR DIGITAL ANALOG POWER SERIAL SPI I2C PWM INTERRUPT

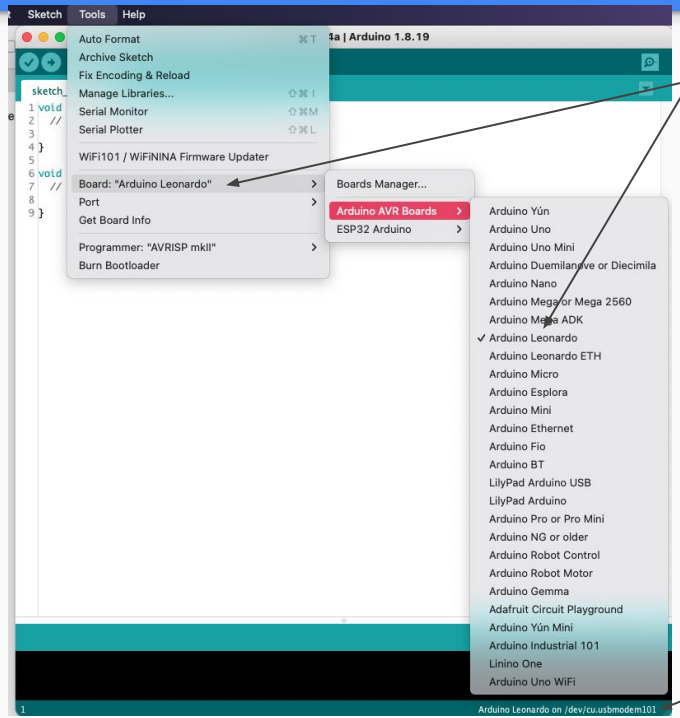
Introduction to Arduinos



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

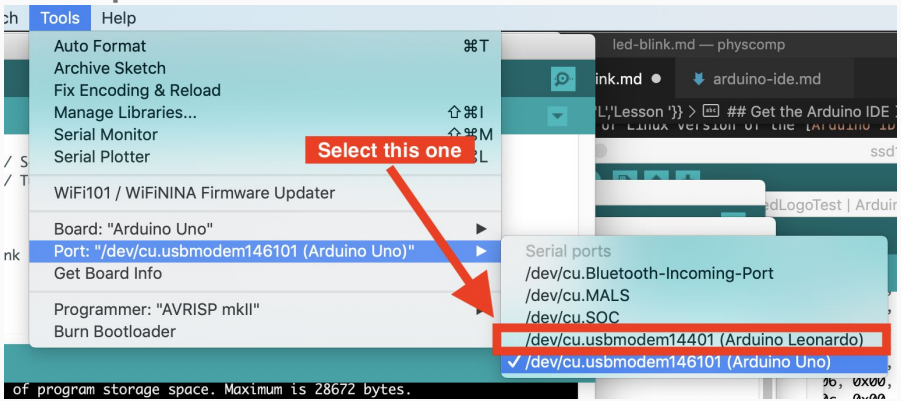
- Code uploaded from computer to Arduino via micro-USB port
- If powered, code is ALWAYS running
  - RST -> restart
  - Unpowering and powering Arduino -> begins re-running whatever was last uploaded
- If you find this to be an issue, the easiest solution is to upload a blank program

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 (pt2)

- 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**
- Fill out https://eecs16b.org/lab-groups to receive your Arduino

General Reminders/Habits

- Return resistors to the brown RETURN RESISTORS HERE box.
- 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
  - i.e. 5V/3.3V/GND from power supply
- PLEASE CLEAN UP AFTER YOURSELF!! Put probes back, pack up kits, throw away stripped wires etc.
- Don't unplug computers
- Work on the lab report :), the deadline will creep up on you

Important Forms/Links

- Help request form: https://eecs16b.org/lab-help
- Checkoff request form: https://eecs16b.org/lab-checkoff
- Slides: links.eecs16b.org/lab2-slides
- Lab Groups: https://eecs16b.org/lab-groups
- Anon Feedback: https://eecs16b.org/lab-anon-feedback