Lecture 04
Prototyping & Digital Circuits

CE346 – Microcontroller System Design

Branden Ghena – Spring 2026

Some slides borrowed from:
Josiah Hester (Northwestern), Prabal Dutta (UC Berkeley)

Administrivia

- Labs
  - See schedule of Lab hours available on Canvas for checkoffs
    - Due by end-of-day Thursday (but last office hours ends at 8pm)

- Quiz
  - Today at end of class! (tell your friends who aren't here yet)
  - Someone remind me at ~4:30 if I don't stop

Today's Goals

- Understand the basics of digital circuitry
  - Enough to be able to interact with the Microbit
  - Inputs: Switches and Buttons
  - Ouptuts: LEDs

- Discuss prototyping methods such as breadboarding
  - And where to find sensors

- Introduce final project goals and requirements

Outline

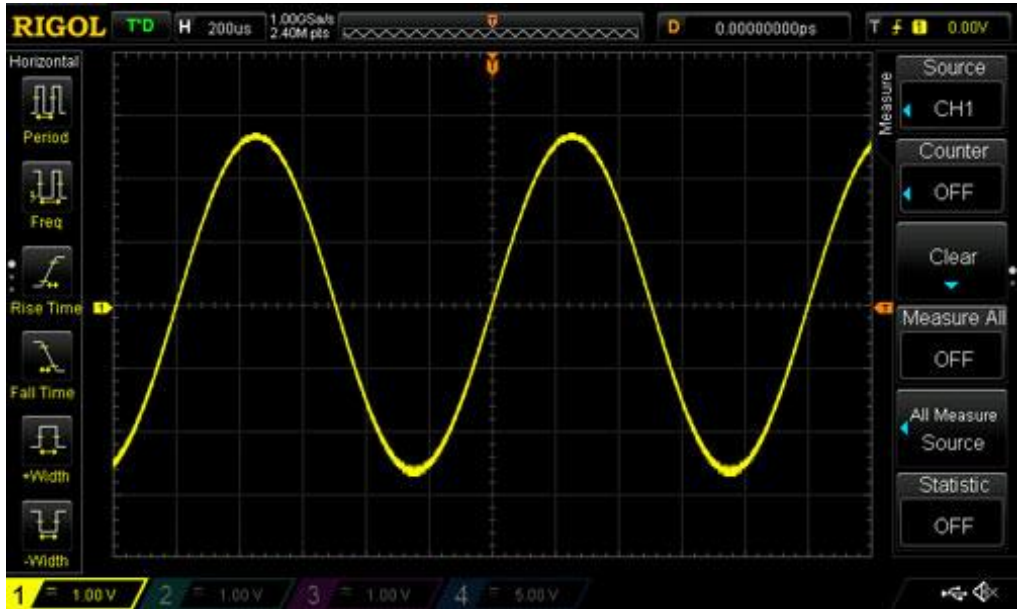
- **Digital Circuits**
  - **Inputs**
  - Outputs
- Prototyping
- Project Overview

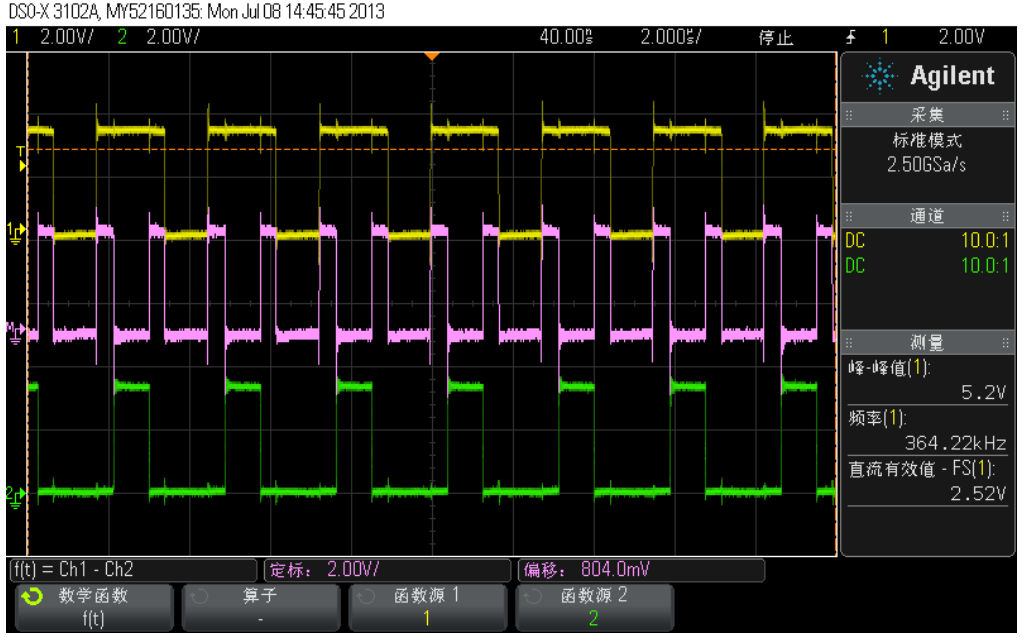
Digital signals

- Exist in two states:
  - High (a.k.a. Set, a.k.a. 1)
  - Low (a.k.a. Clear, a.k.a. 0)

- Simpler to interact with
  - Constrained to two voltages
  - With quick transitions between the two

  - No math for voltage level
    - Either high or low





Digital circuits

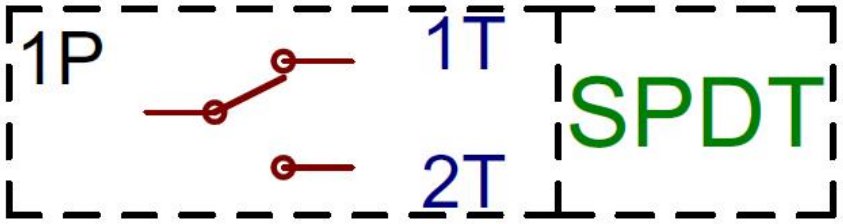
- Connecting components together with digital signals
  - Mostly ICs
  - Also buttons/switches and LEDs

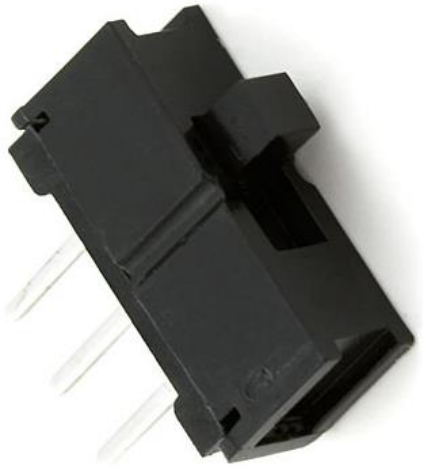
- Way simpler than analog circuits
  - Mostly connecting boxes with wires
  - Plus a few resistors here and there

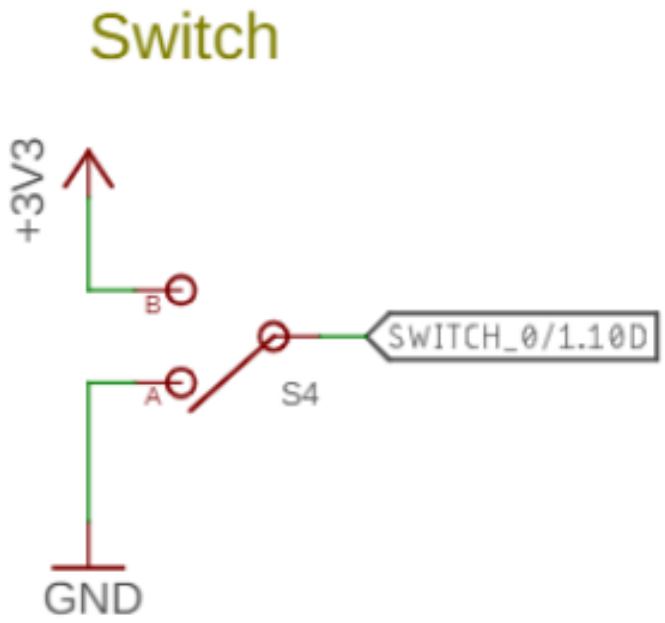
- An abstraction
  - Not sufficient for fully understanding electronics behavior, but close

Switches

- Single Pole, Double Throw switch
  - Middle pin (Pole) connects to one of two outer pins (Throws)





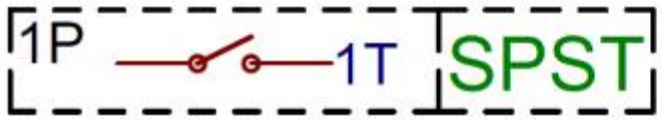


- For controlling microcontrollers
  - Often connect outer pins to VCC and Ground respectively
  - Input then goes High or Low depending on switch state

https://learn.sparkfun.com/tutorials/button-and-switch-basics/

Buttons

- Single Pole, Single Throw switch
  - Pole pin either connects to Throw pin or is disconnected
  - Come in normally-closed (connected) and normally-open (disconnected)



Disconnected circuits

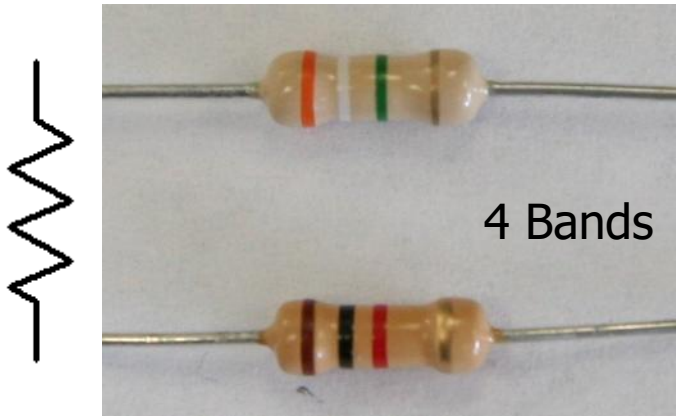
- When button is pushed, input signal is low
- **What is the value of the input when the button is unpressed?**

Disconnected circuits

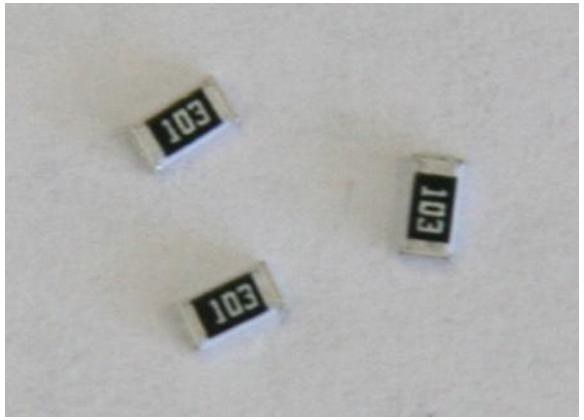
- When button is pushed, input signal is low
- **What is the value of the input when the button is unpressed?**
  - Floating! Could be any voltage
  - Solution: need to connect weakly to either high or low voltage

Resistors

Carbon Film Resistors

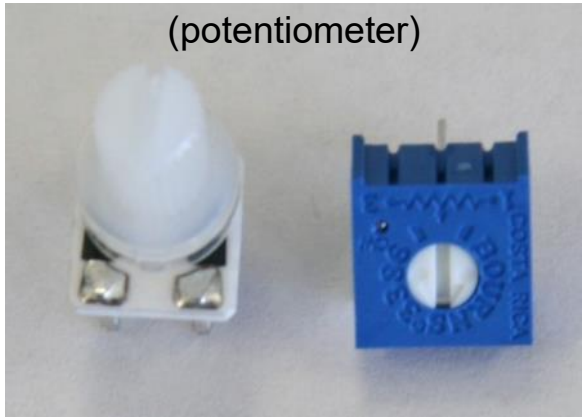


Surface Mount Resistors



Variable Resistors

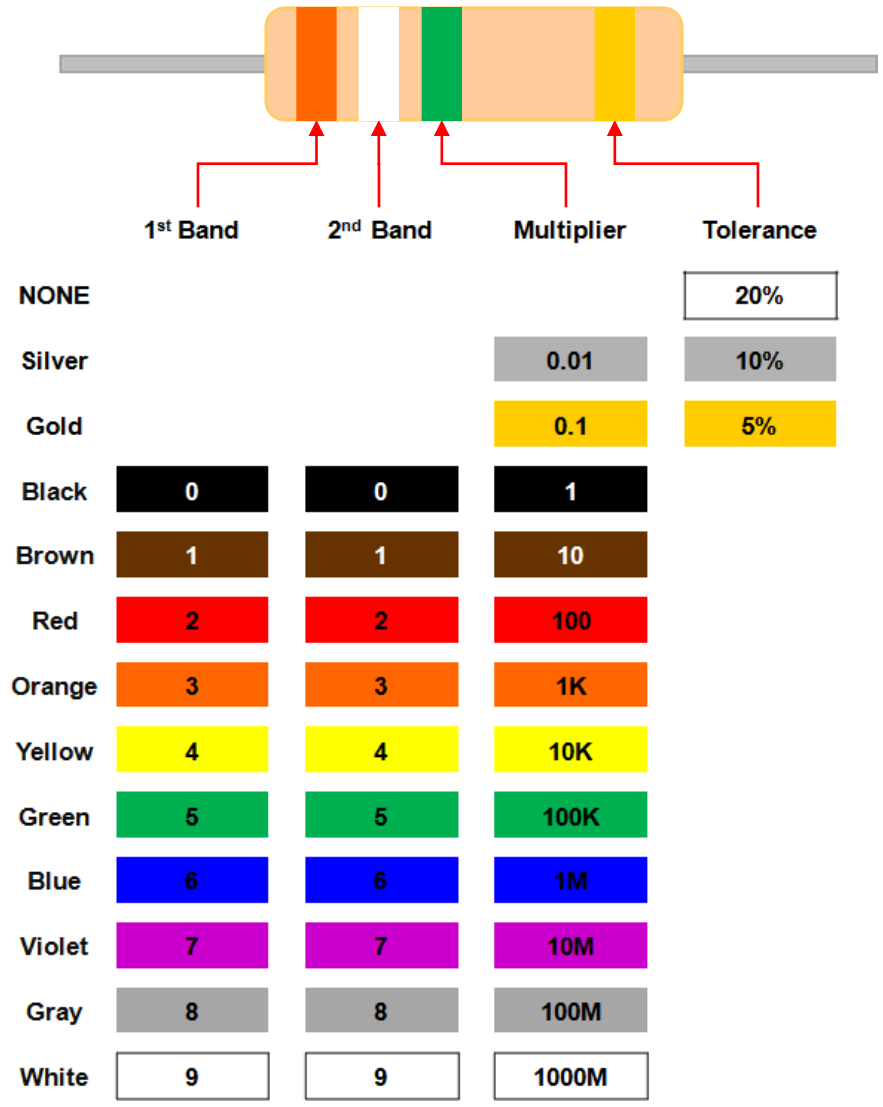
(potentiometer)



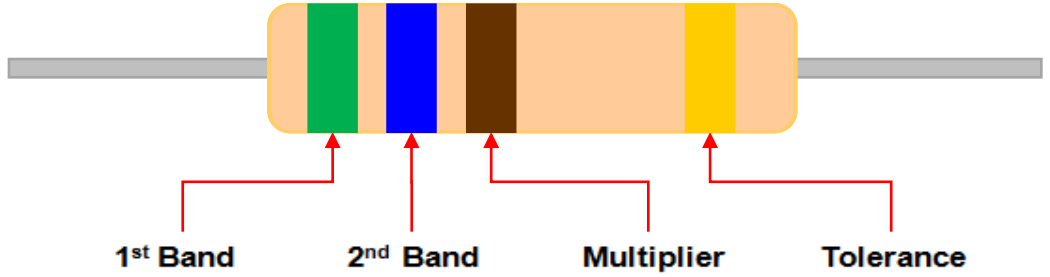
- Resistors are not directional
  - Either orientation is fine and works identically

Resistor color codes

- Colored bands on resistors label the resistance value of the part
- First and second bands are the digits
- Third band is multiplier
- Fourth band is tolerance
  - Usually gold: +/- 5%



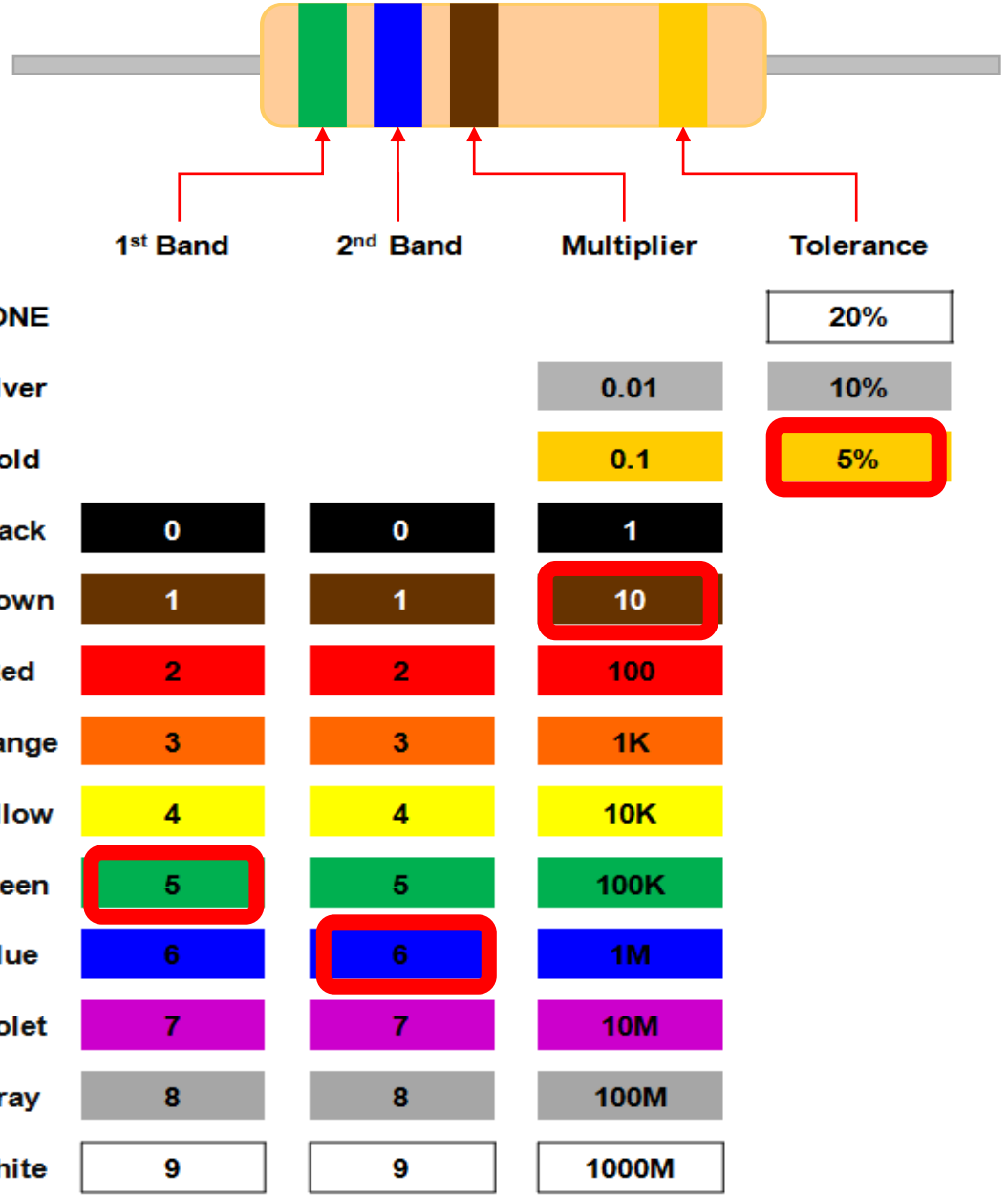
Break + Determine the resistor



	1st Band	2nd Band	Multiplier	Tolerance
NONE				20%
Silver			0.01	10%
Gold			0.1	5%
Black	0	0	1	
Brown	1	1	10	
Red	2	2	100	
Orange	3	3	1K	
Yellow	4	4	10K	
Green	5	5	100K	
Blue	6	6	1M	
Violet	7	7	10M	
Gray	8	8	100M	
White	9	9	1000M	

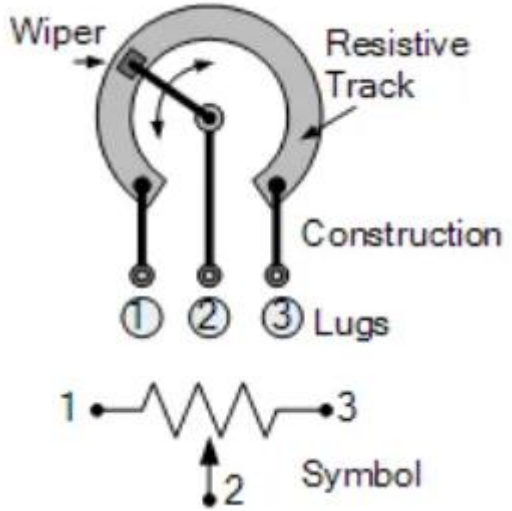
Break + Determine the resistor

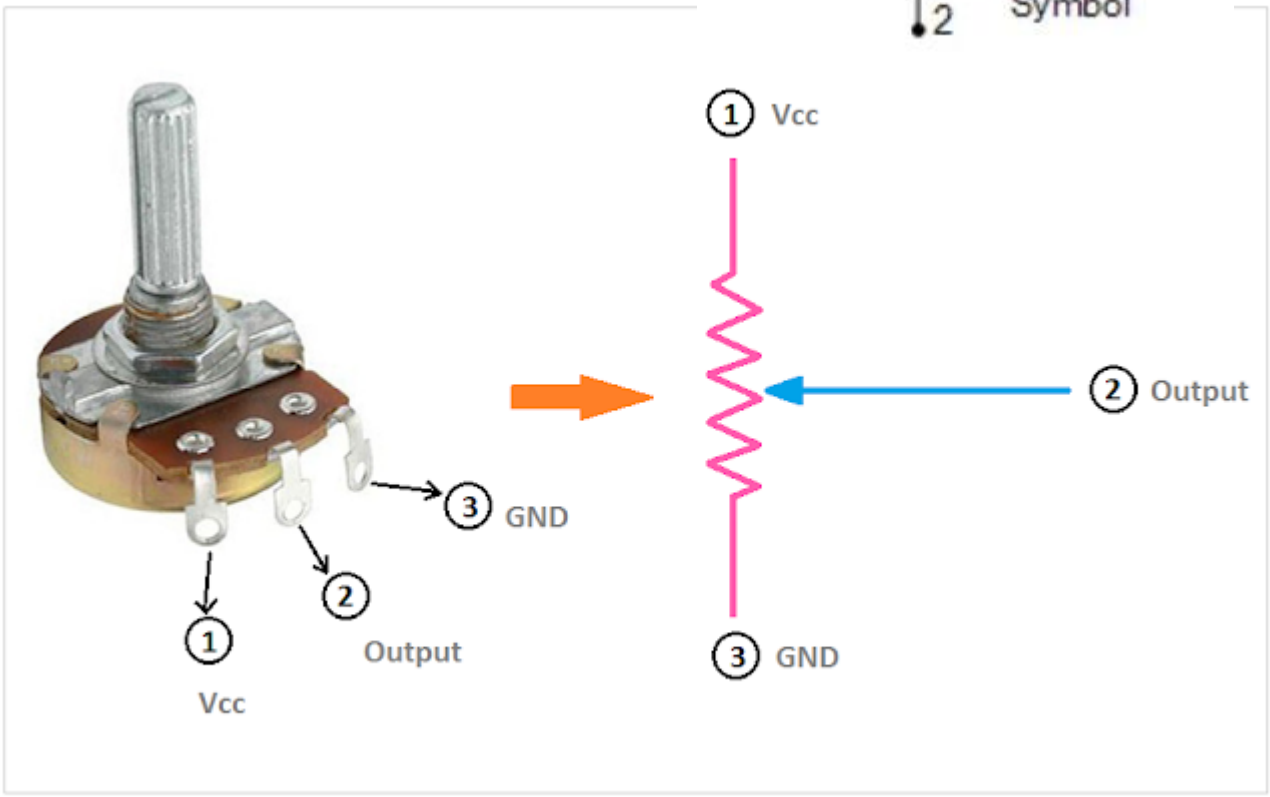
- 56 x 10 Ω = 560 Ω (±5%)



Potentiometers

- Vary resistance between zero and some maximum
  - 1 kΩ, 10 kΩ, 100 kΩ common
- Connect middle and an edge for just a changeable resistor
- Middle terminal is a movable resistor divider
  - Knob changes middle output if outer pins are VCC and Ground

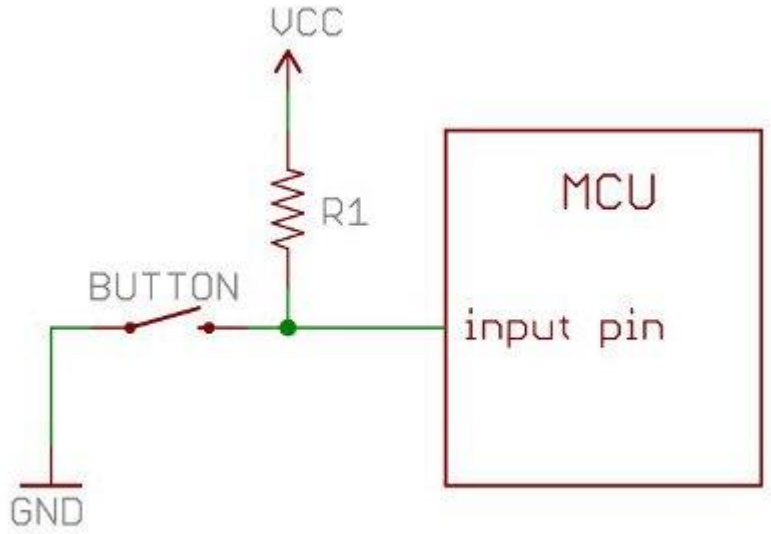




Current flows through the "path of least resistance"

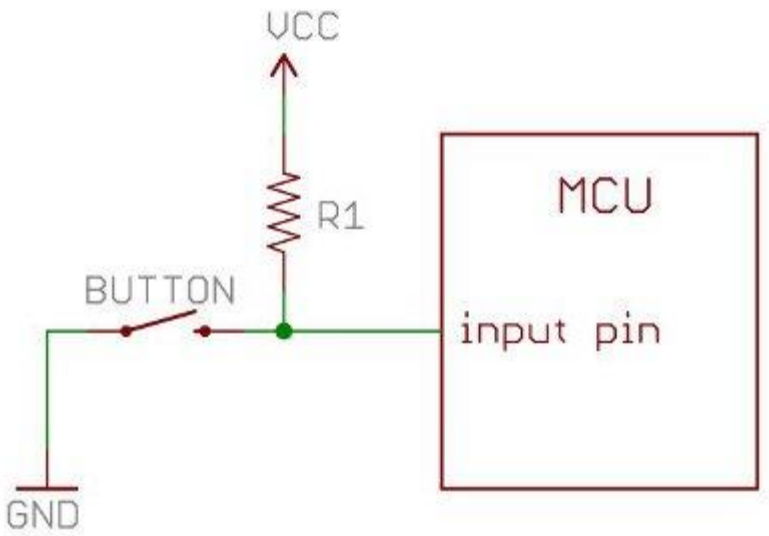
- Simplification
  - Works well for the types of circuits we use

- Pull-up resistor
  - When button is open (disconnected), the only path is through the resistor
  - When button is closed (connected) the least resistance path is through the button to Ground



Pull-up resistors and pull-down resistors

- Resistor sets the "default" value of a wire
  - Pull-up connects to VCC
  - Pull-down connects to Ground
  - Usually 10-100 kΩ

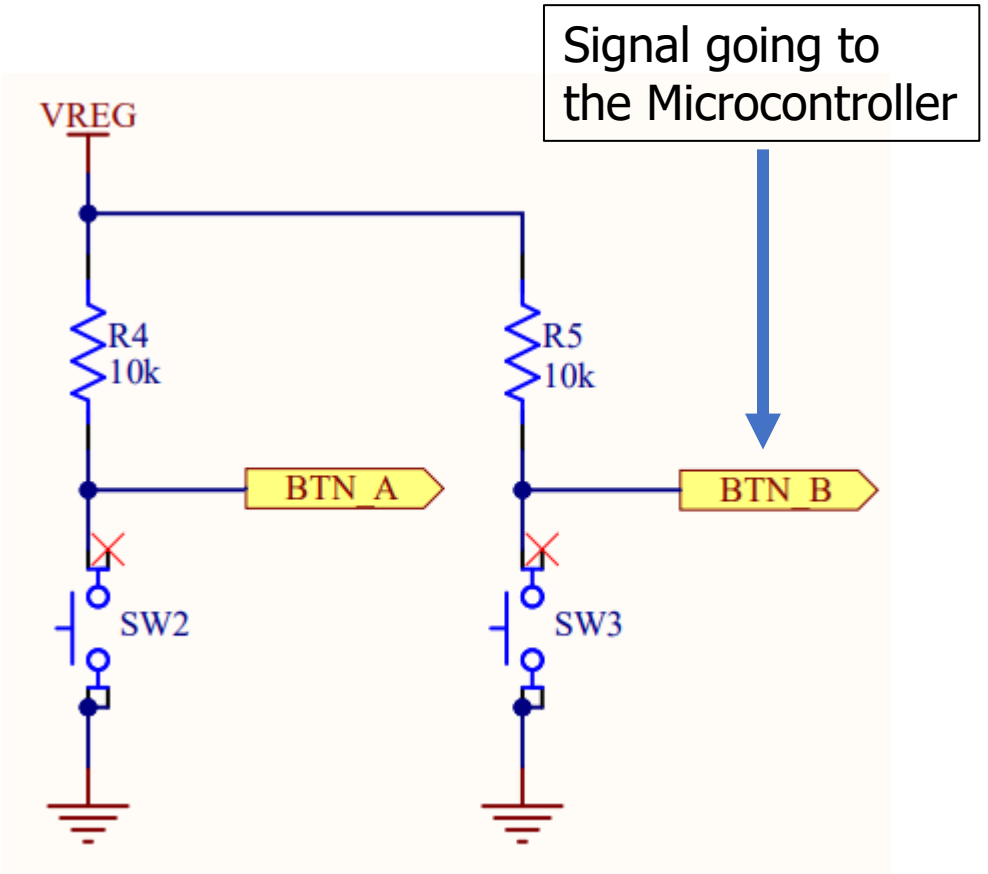


- When button is open (disconnected)
  - Connection through the resistor sets signal

- When button is closed (connected)
  - Signal is directly connected to a voltage source
  - Much lower resistance means that signal dominates

Buttons on the Microbit

- Normally open buttons
  - Disconnected by default
- Active low signal
  - Activating (pushing) button creates a low signal
- Pull-up resistors
  - Set button signal high by default

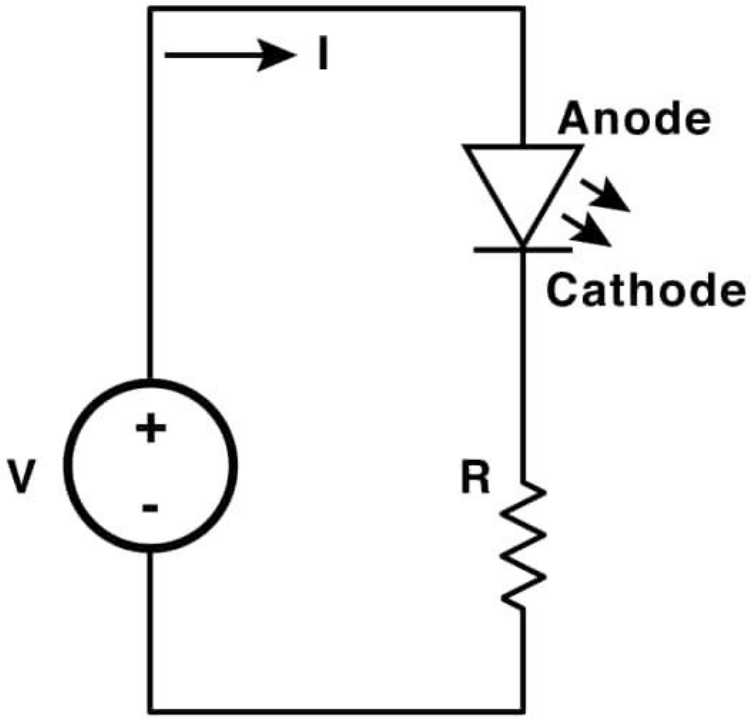


Outline

- **Digital Circuits**
  - Inputs
  - **Outputs**
- Prototyping
- Project Overview

LEDs

- Light Emitting Diodes
  - Generate light as current passes through them
  - Various colors available
- Diodes
  - Only allow current to go through one way
  - Not particularly relevant for LEDs
    - Treat as a digital component



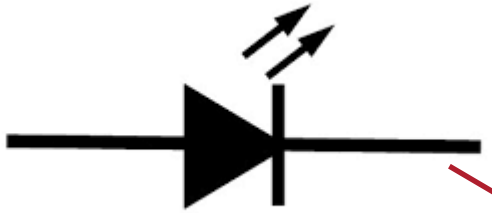
- Connect anode to high voltage and cathode to ground
  - Plus a resistor to limit the total amount of current

https://learn.sparkfun.com/tutorials/light-emitting-diodes-leds

LEDs

- Directional component: only allows current to flow one way

- Shorter side is the negative one
  - i.e. where current flows to



Schematic Symbol

Negative (-) lead

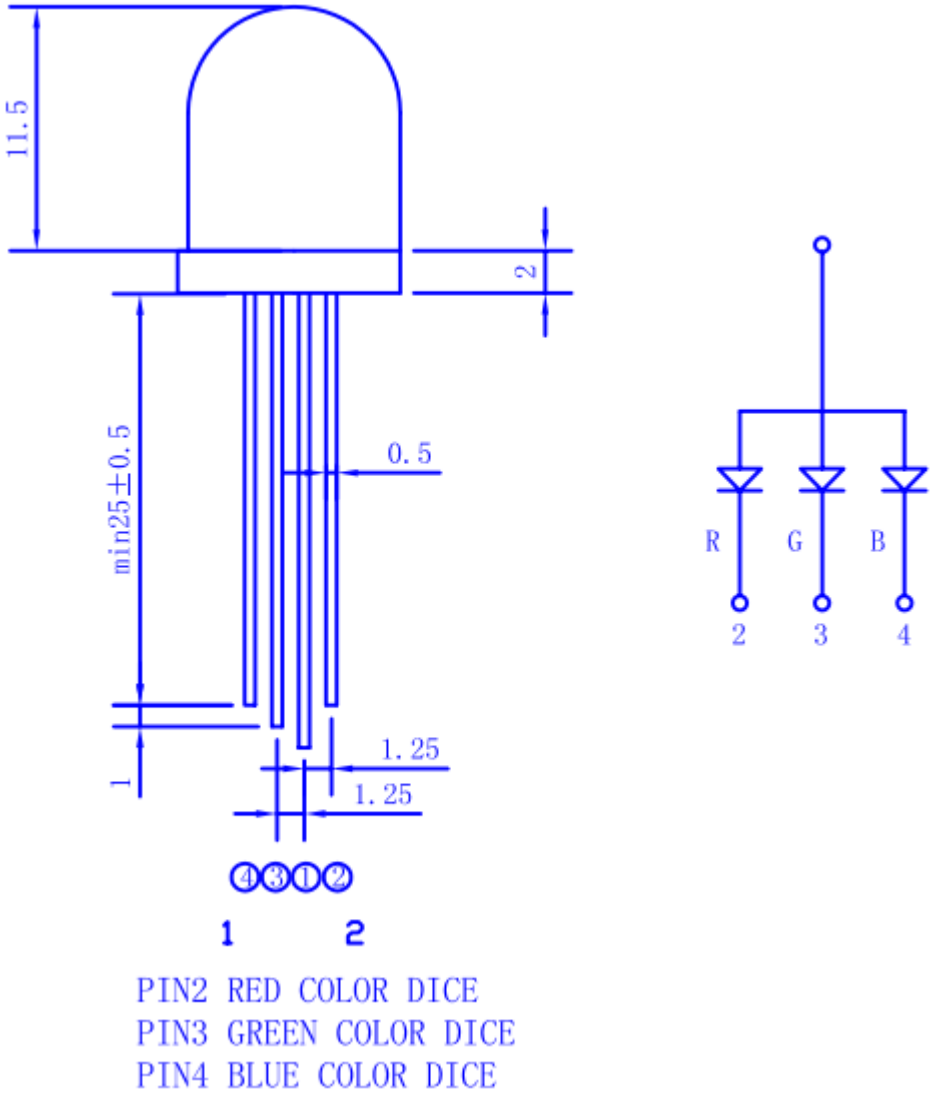
Surface-mount LED

Larger metal component inside of case or case flat spot is cathode or negative (-) lead

Shorter wire is cathode or negative (-) lead

RGB LED

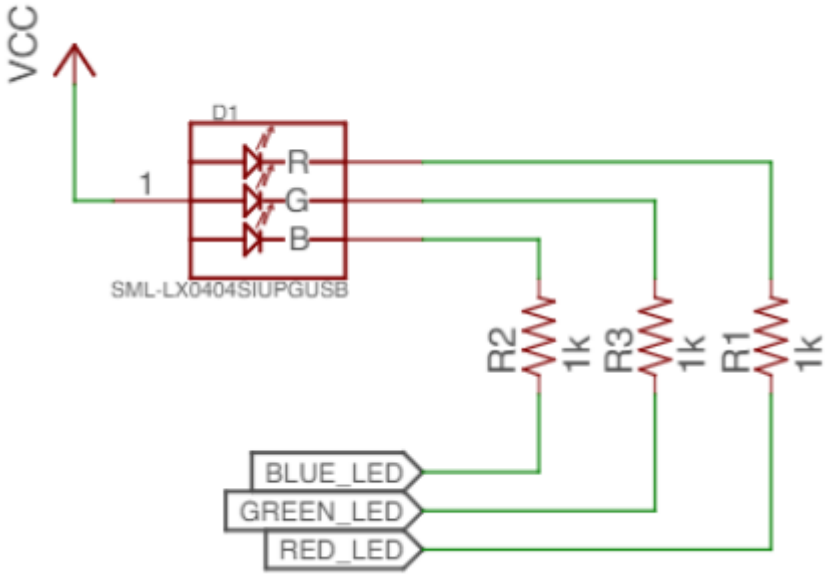
- Three different colors of LED in a single large diffuser
- Short leads are negative ends
  - One for each color
- Long lead is common power
  - Common anode
- Combinations of LEDs give other colors
  - Cyan, Yellow, Violet, White

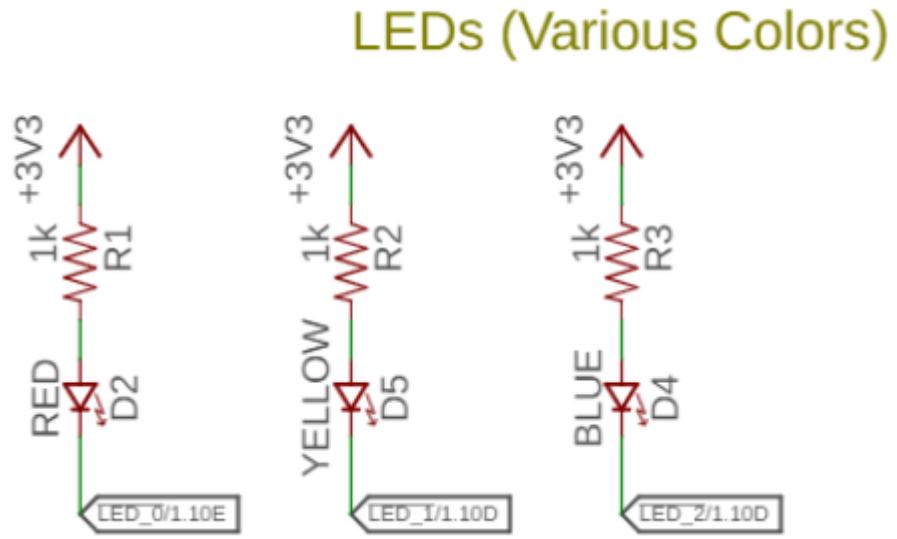


https://cdn-shop.adafruit.com/datasheets/FLR-100WAS-RGB.pdf

Active state for LEDs

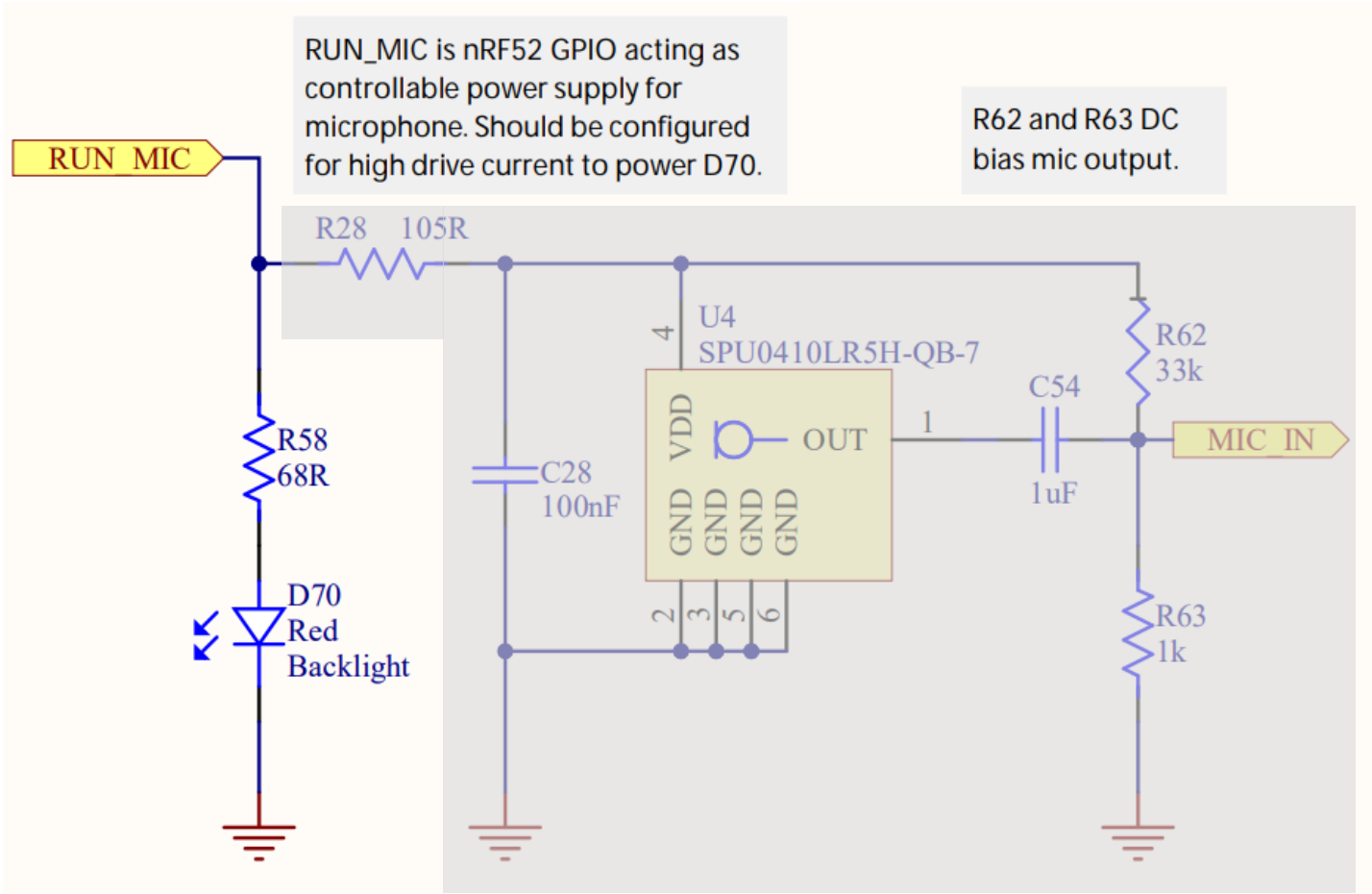
- LEDs can be active high or active low depending on configuration
  - Active high is how people assume they work
  - Active low is often used instead
    - GPIO pins can usually sink more current than they can source





LEDs on the Microbit

- Microphone LED
  - Active high
- Simple to use, just set the GPIO high to enable it

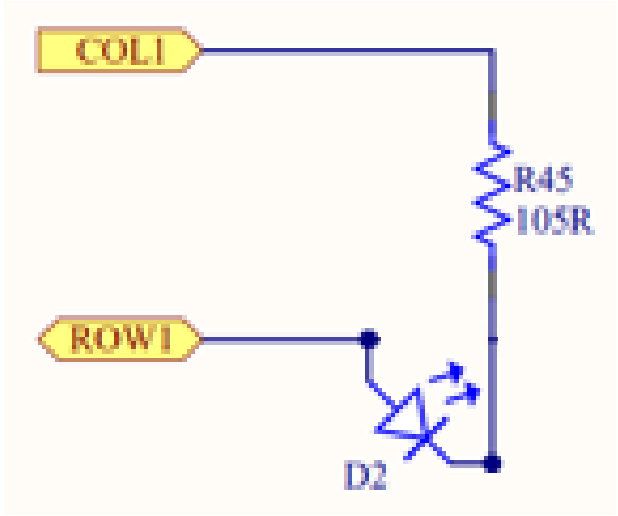


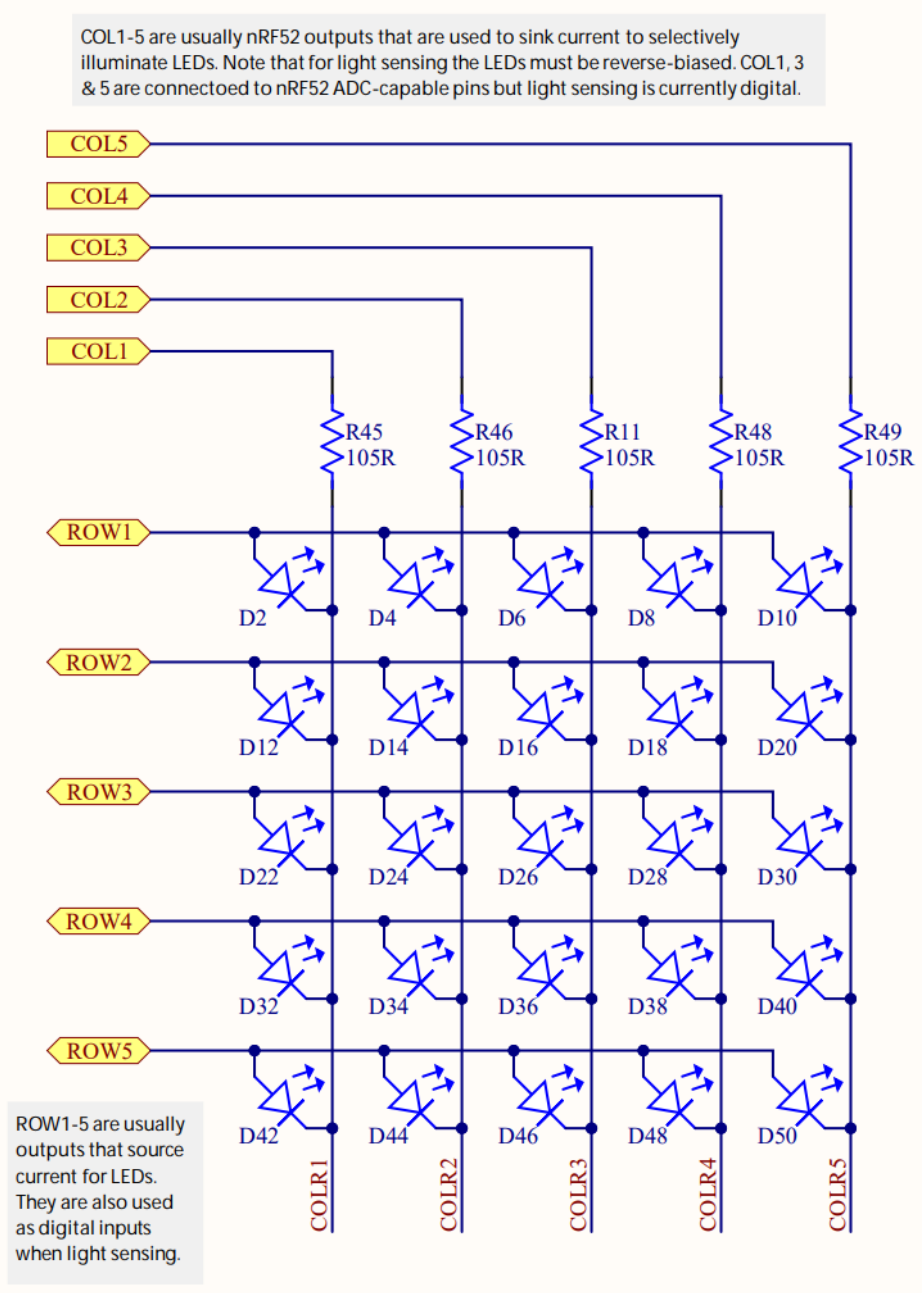
Ignore this other part for now

LEDs on the Microbit

- Use two GPIO pins to control each LED
  - Row high as VDD
  - Column low as Ground

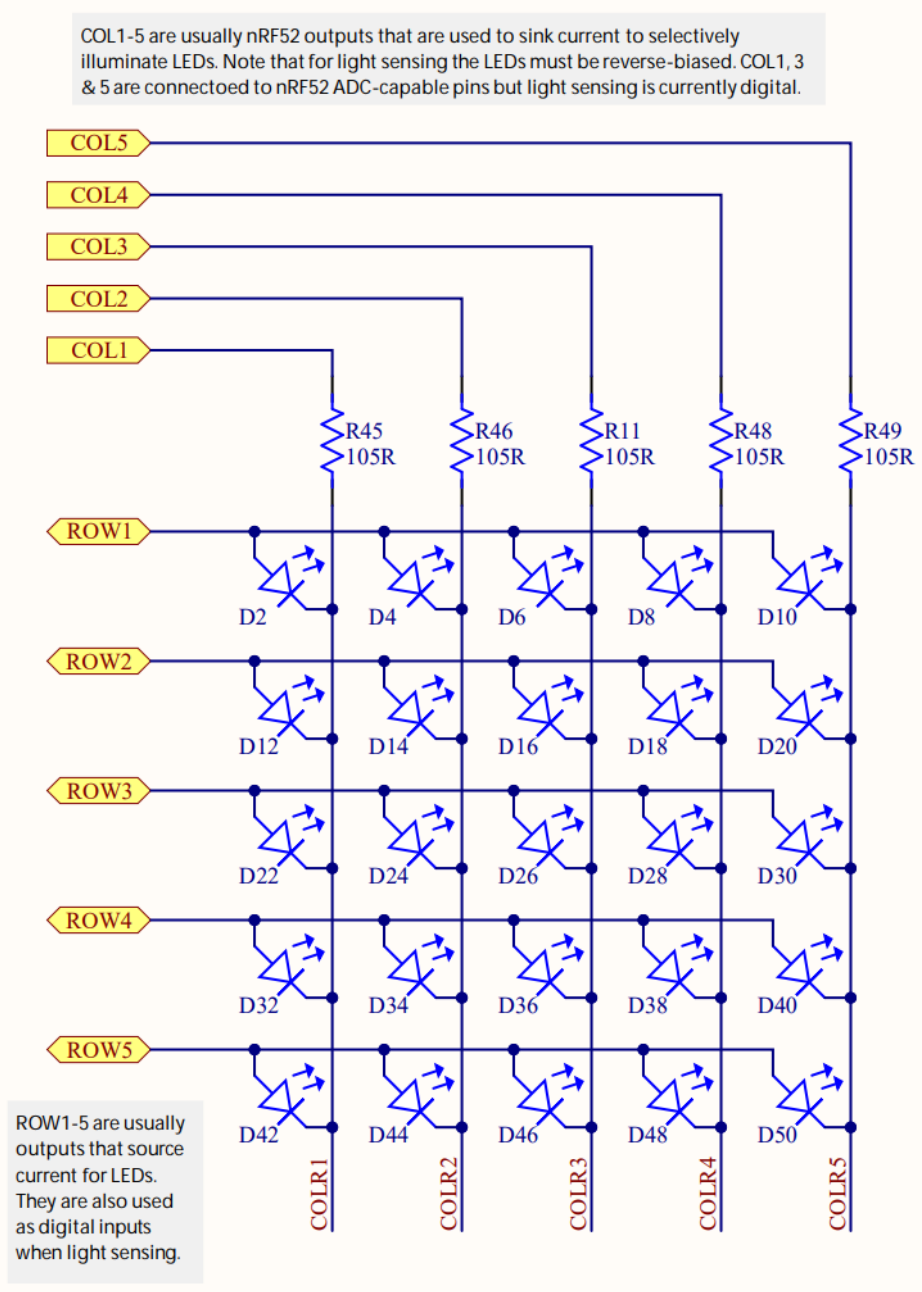
- Connections on circuit schematics only exist where there are dots





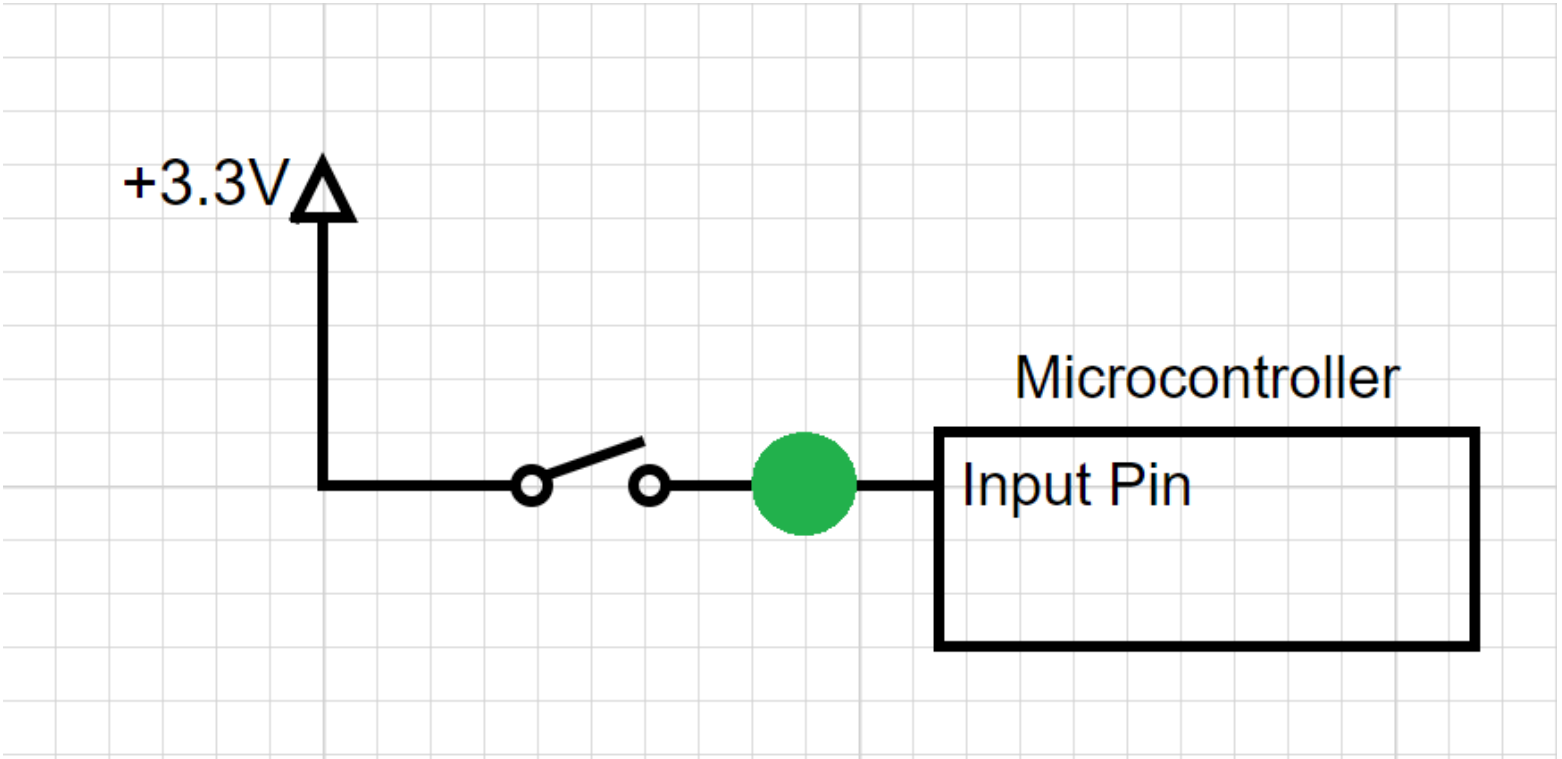
Controlling the LED matrix

- Cannot individually control all LEDs simultaneously
  - Need to light one row at a time
  - Iterate rows quickly to make them appear on all the time
- We'll have a lab on these later
  - Combines GPIO and timers



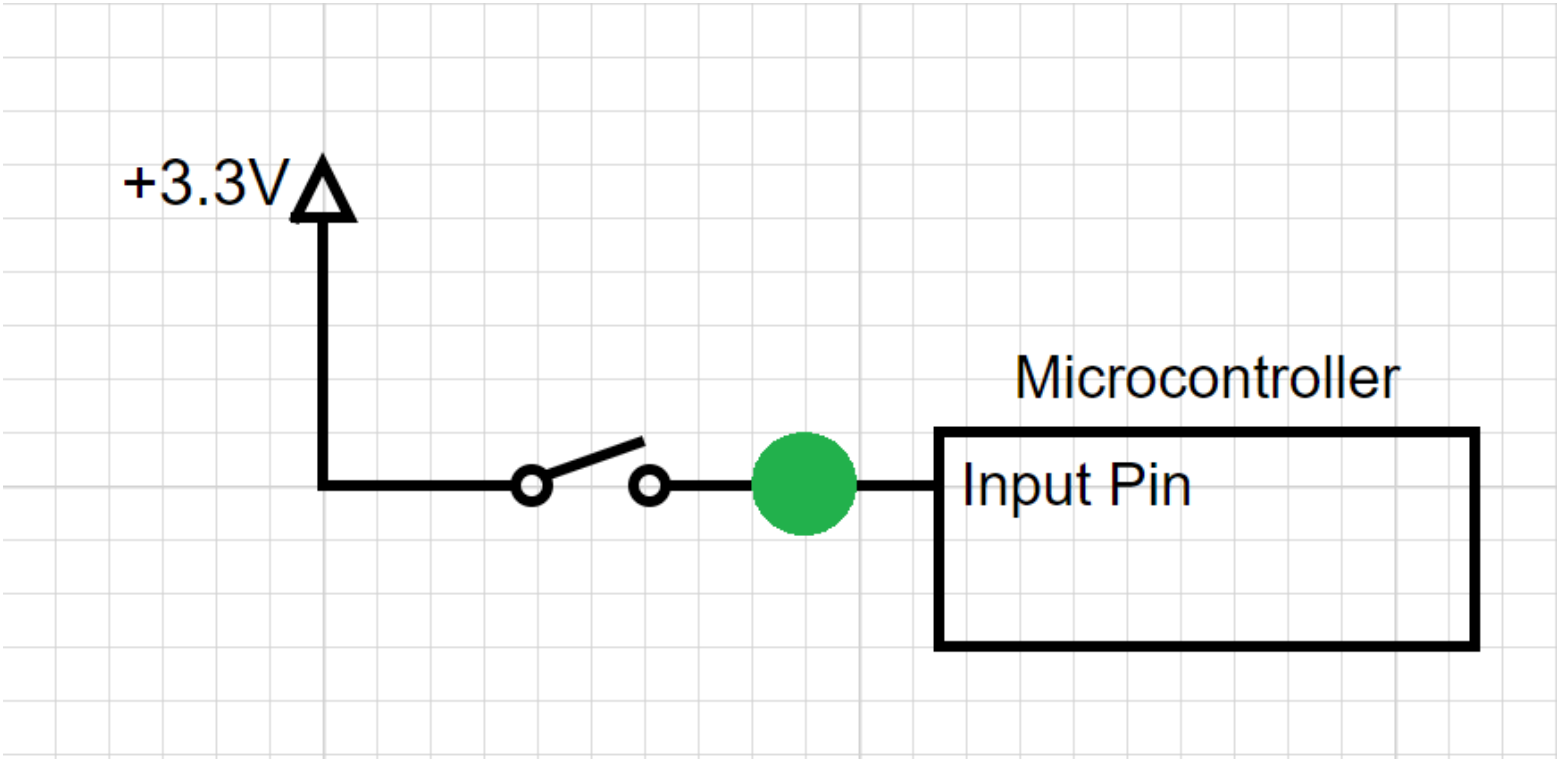
Break + Question

- Should the spot in green have?
  A. Pull-up Resistor
  B. Pull-down Resistor
  C. Either
  D. Neither



Break + Question

- Should the spot in green have?
  A. Pull-up Resistor
  B. Pull-down Resistor (needs to pull input low by default)
  C. Either
  D. Neither



Outline

- Digital Circuits
  - Inputs
  - Outputs
- **Prototyping**
- Project Overview

Prototyping goals

- Does this thing work at all?
  - Particular IC
  - Circuit layout
  - Software design
  - etc.

- Sometimes before doing something more serious with it
  - Design a PCB, Make a product, etc.

  - Not uncommon that the prototype is as far as you'll get

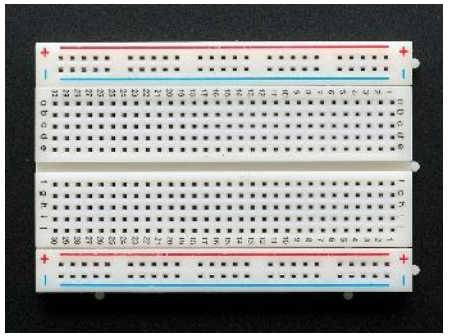
Isolating tests

- The goal when prototyping is to isolate the question at hand

- Do consider
  - New sensor/IC/component/whatever

- Do not consider
  - Power
  - Interference
  - Enclosure
  - Stable microcontroller
  - Soldering skills

Prototyping methods

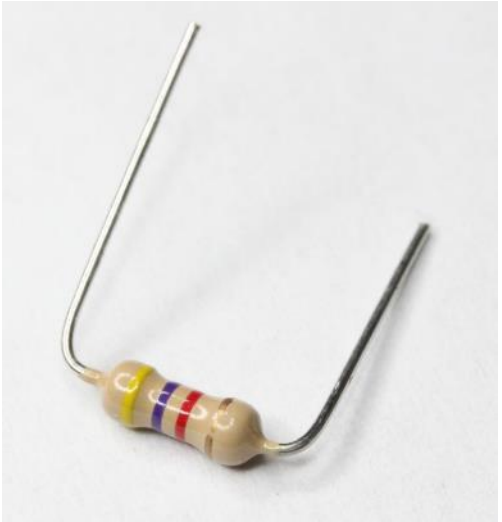


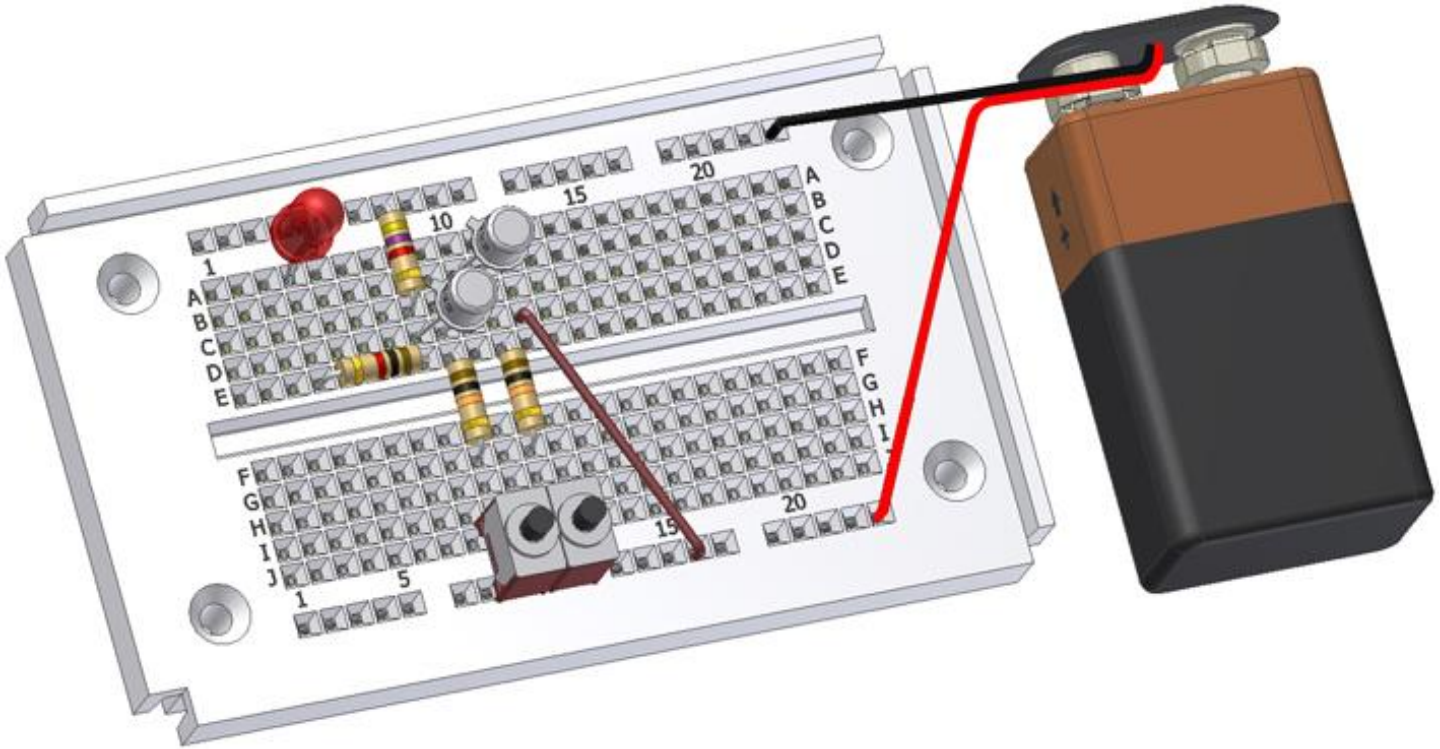
- Breadboarding
  - Plug and connect components as needed
  - Build up arbitrarily complex designs from nothing
- Development kits
  - Pre-fabricated systems design for testing components
- Small-scale test PCBs
  - Design a PCB that demonstrates the thing you're interested in
    - Making a PCB is less hard than some might think (Eagle, Fritzing, etc.)
    - $20-30 for small, low-speed PCBs from batch services like OSHPark

Breadboards for prototyping

- Reusable platform for temporary circuits

- Plug in jumper wires and through-hold components

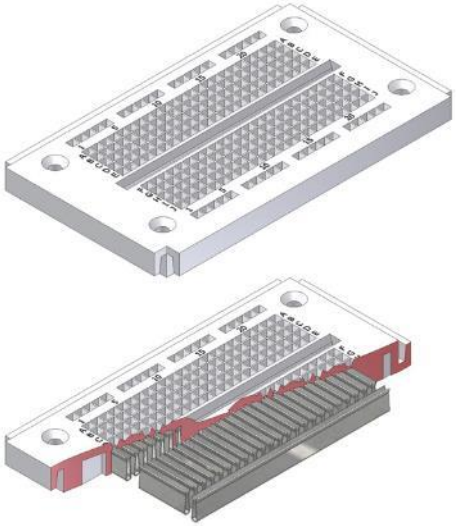


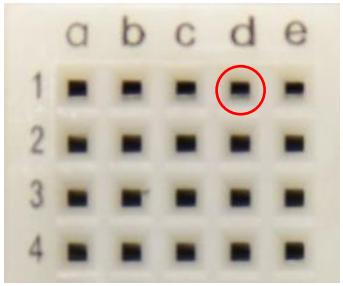


How a breadboard works

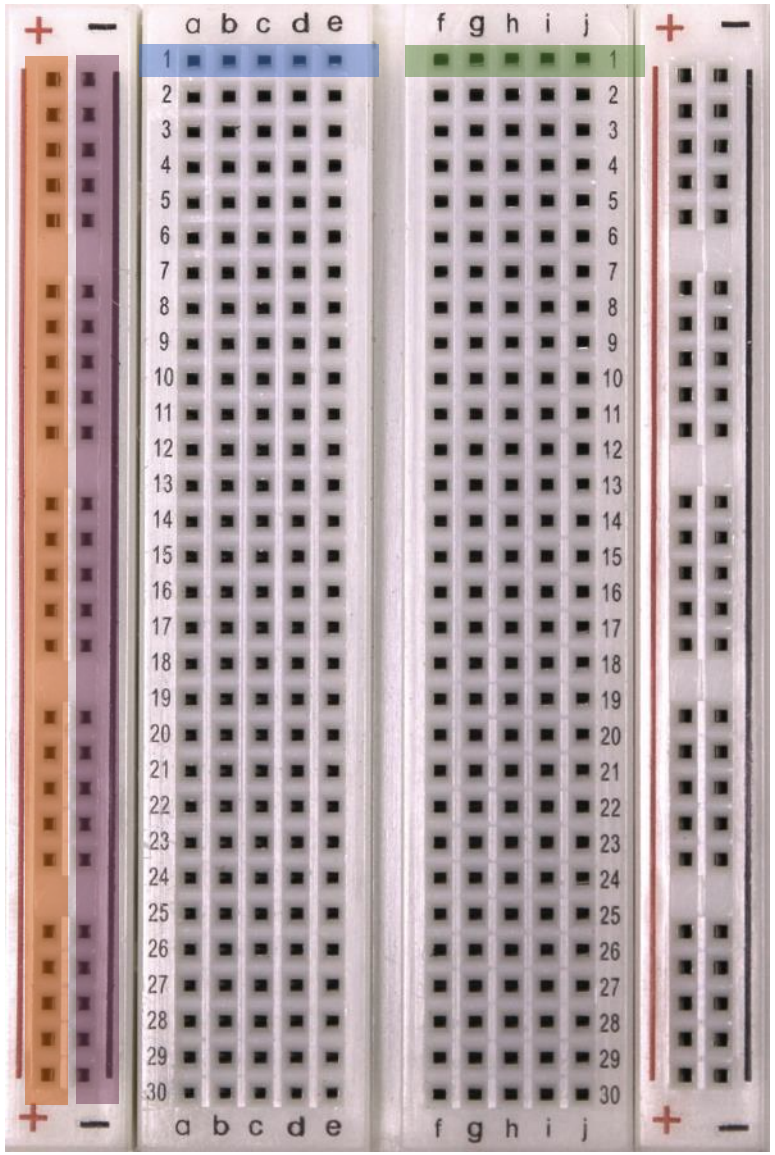
- Component leads and wires are inserted into holes in the breadboard

- Half-rows of five holes are connected
- Vertical columns are connected for power/ground



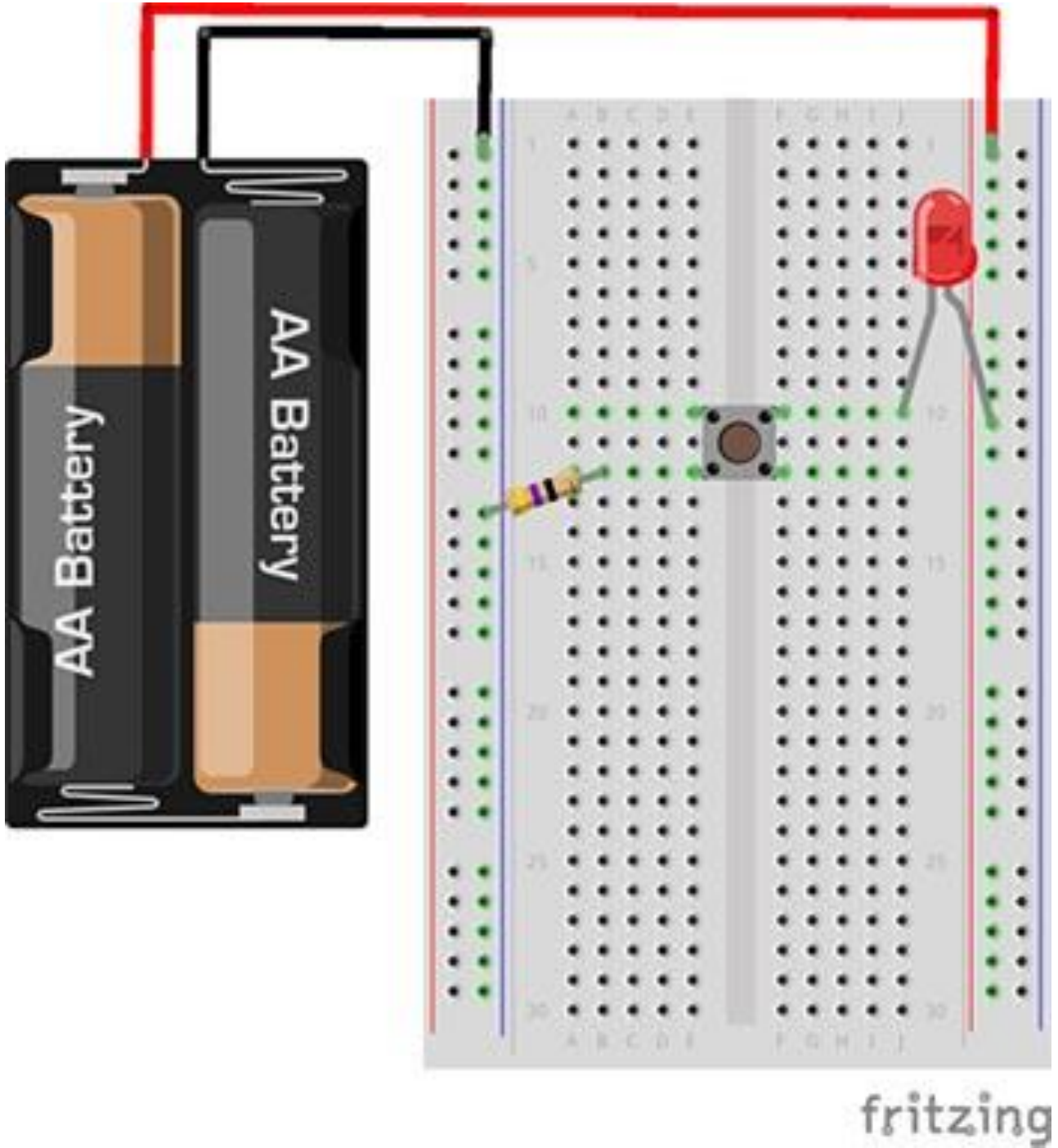


Holes to insert wires



Breadboard LED example

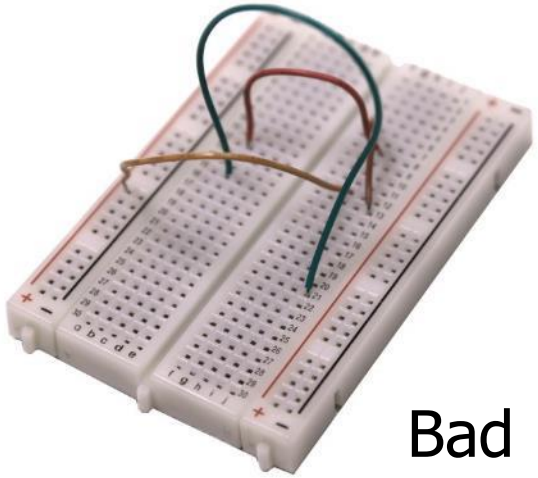
- Uses button to control LED



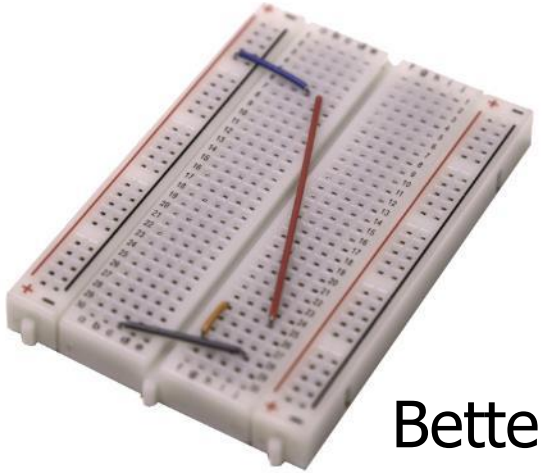
Breadboard guidelines

- Long wires in large bird nests makes debugging very difficult
  - Shorter, constrained wires are easier to understand
  - In this class, we'll only have large jumper wires though…

- Use the minimum jumpers necessary, mostly use breadboard for connections



Bad



Better

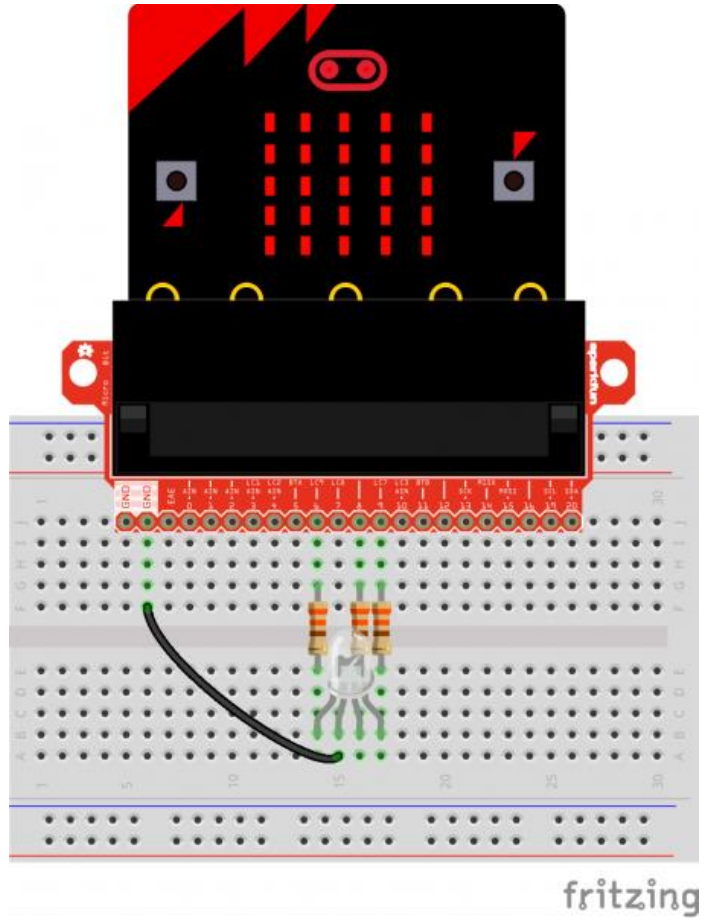
Microbit breakout

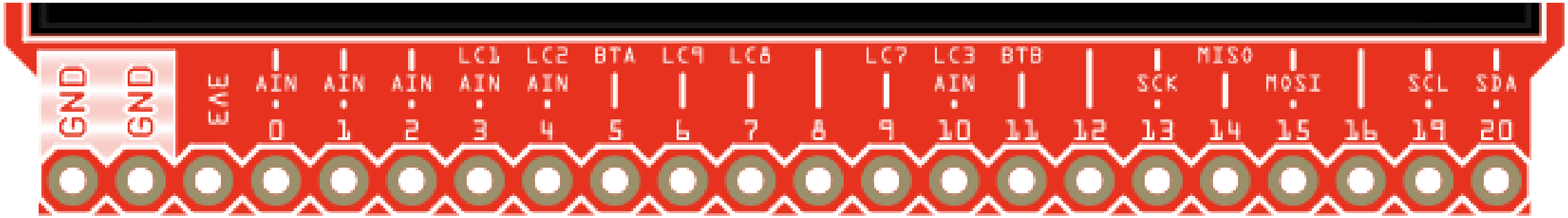
- Connects Microbit pins to breadboard
  - Need to consult table to know which pins
  - https://tech.microbit.org/hardware/schematic/

- **Always connect LED matrix side up**

https://www.sparkfun.com/products/13989

https://learn.sparkfun.com/tutorials/microbit-breakout-board-hookup-guide

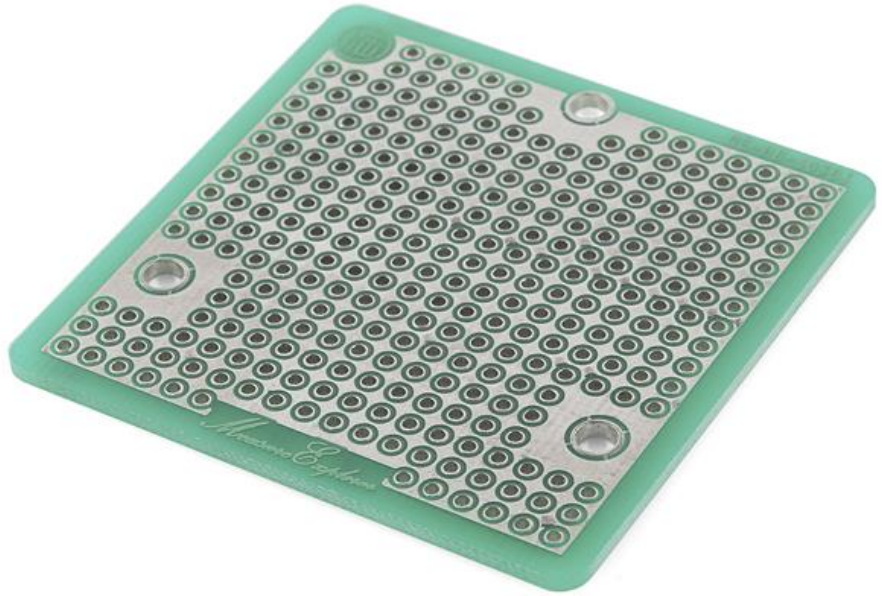




More permanent breadboards

- Breadboards are also known as “Solderless Breadboards”

- Protoboard allows configurable circuits
  - Solder jumper wires between locations
  - Solder adjacent pads to form connection



- Usually not worth it (just make a PCB)
  - Does solve core problem of breadboards: things getting unintentionally unplugged
  - Might be useful for some projects!

When to not use breadboards

- Breadboards work great for digital circuits and simple analog!

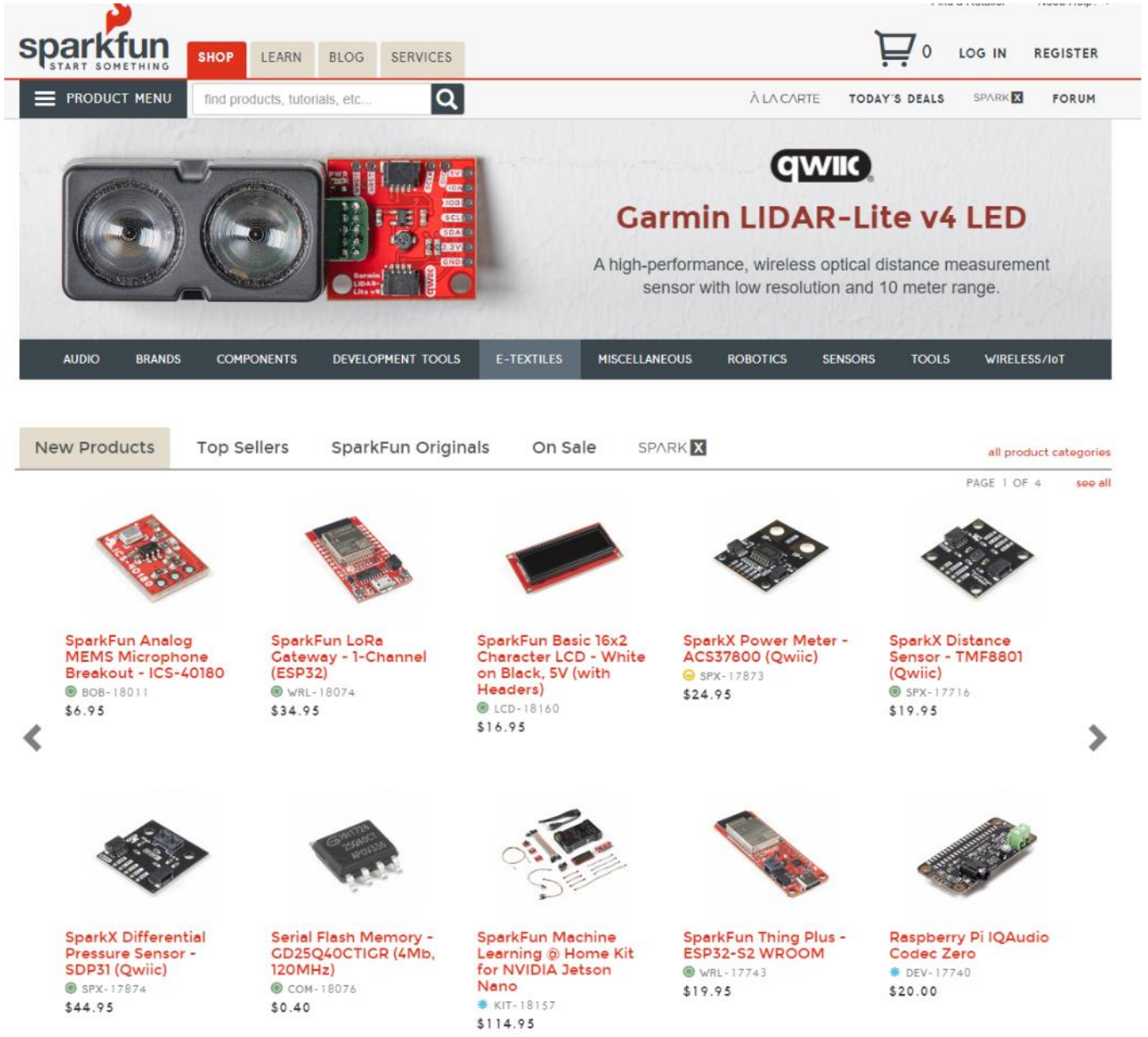
- High voltage/current are bad for breadboards
  - Honestly, anything above 12 volts DC shouldn’t be in a breadboard
  - Also avoid high-power applications above a few Watts
  - Never put AC in a breadboard!

- Sensitive analog circuits
  - Particularly anything sensitive to capacitance may not work right
  - Sets of metal holes with strips connecting them function as capacitors

- Anything in long term use

Buying Parts

- Prototyping vendors
  - Where you look for cool stuff to buy
  - Sparkfun
  - Adafruit
  - NOT Amazon for prototyping parts if possible
- Electronics vendors
  - Where you buy raw parts when you know what you need
  - Digikey
  - Mouser



QWIIC sensors

- https://www.sparkfun.com/qwiic
  - Adafruit has compatible parts it calls “STEMMA” https://www.adafruit.com/category/620

- Sensors have a standard, wired interface
  - Power, Ground, I2C (wired communication protocol)

- Removes electrical/wiring issues
  - Transforms into a software issue of what the correct commands to send to the device are
  - Our usual plan: use an existing driver to guide creation of our own

Outline

- Digital Circuits
  - Inputs
  - Outputs
- Prototyping
- Project Overview

Project Proposals

- It is time to start forming teams and working on Proposals
  - Due next week Friday! (04/24)

  - Project proposal details are posted now
    - 1-2 pages, with some specific items you MUST include

- Project teams are 2-3 students (4 under rare occasions)
  - You may NOT work alone
  - There is a partnership survey if you want us to match you with someone
    - Due by end-of-day Sunday (04/19)

What are the rules for a final project?

- Minimum: anything with at least two inputs and one output
  - Where none of those three were developed for lab
  - Rule of thumb, not a precise limit

- Goals:
  1. Pick something interesting to you

  2. Applies course knowledge to creating the project
     - Often also includes some knowledge beyond this class

  3. Takes some serious effort to complete and is a well-polished demo
     - 40% of your course grade is from the project
     - Proportional expectations to size of group

Specific rules

- 2-3 group members
  - Or 4 if you come talk to me first and it’s justified for a big project
- Uses the Microbit and C code
  - Although it could use additional systems if needed
- Banned projects:
  - Plant Monitoring/Care
  - Home Security System

Judging projects

- Two key metrics: difficulty and quality
  - Grade is a combination of the two
  - Essentially we're considering:
    1. All the work that went into the parts of the project
    2. Making all the parts work together as a cohesive demo

- Either extreme can still be a good project:
  - A project that's really hard to get working, and the demo is kinda jank
  - A project that's simple to get working, and the demo is fantastic

Project collaboration

- Challenge: working on hardware/software projects as a team

- Make sure you communicate with each other about plans and expectations
  - Proposal will ask you to put some small consideration in how best to do this

- Remember that tasks can be different difficulties, so help each other

Navigating proposals

1. Find some people to work with

2. Brainstorm ideas that might be interesting to all of you

3. Do just a little bit of googling to figure out if that seems like a plausible project

4. Write up the proposal
   - Looking for like 70% baked ideas
   - I'll give you feedback if I think the project would be a bad fit (too easy OR too hard)

Outline

- Digital Circuits
  - Inputs
  - Outputs
- Prototyping
- Project Overview