

# Micro:bit Magic

Engaging K-12, CS1/2, and non-majors with IoT & Embedded

**Bill Siever**

**Washington University in St. Louis**

**Michael Rogers**

**Northwest Missouri State University**



# Outline

- Intros: Us, You, the micro:bit
- ★ Setup
- ★ “Hello, World!”: First Program
- ★ Programming: Logic & Action
- ★ Broadcast Basics
- ★ Awesome Audio & Motor Mayhem
- ★ Bluetooth Basics & Phone Phun
- ★ Extensions & Graphing
- ★ Cutting the Cord
- Conclusions



- **Intros: Us & You**



# Intros: Us & You

- 

- Us



# Intros: Us & You

- 
- Us
- You: Roll Call & Intros



# Intros: Us & You

- 
- Us
- You: Roll Call & Intros
  - Who has Chrome? Who has an iOS Device with the App?



# Intros: Us & You

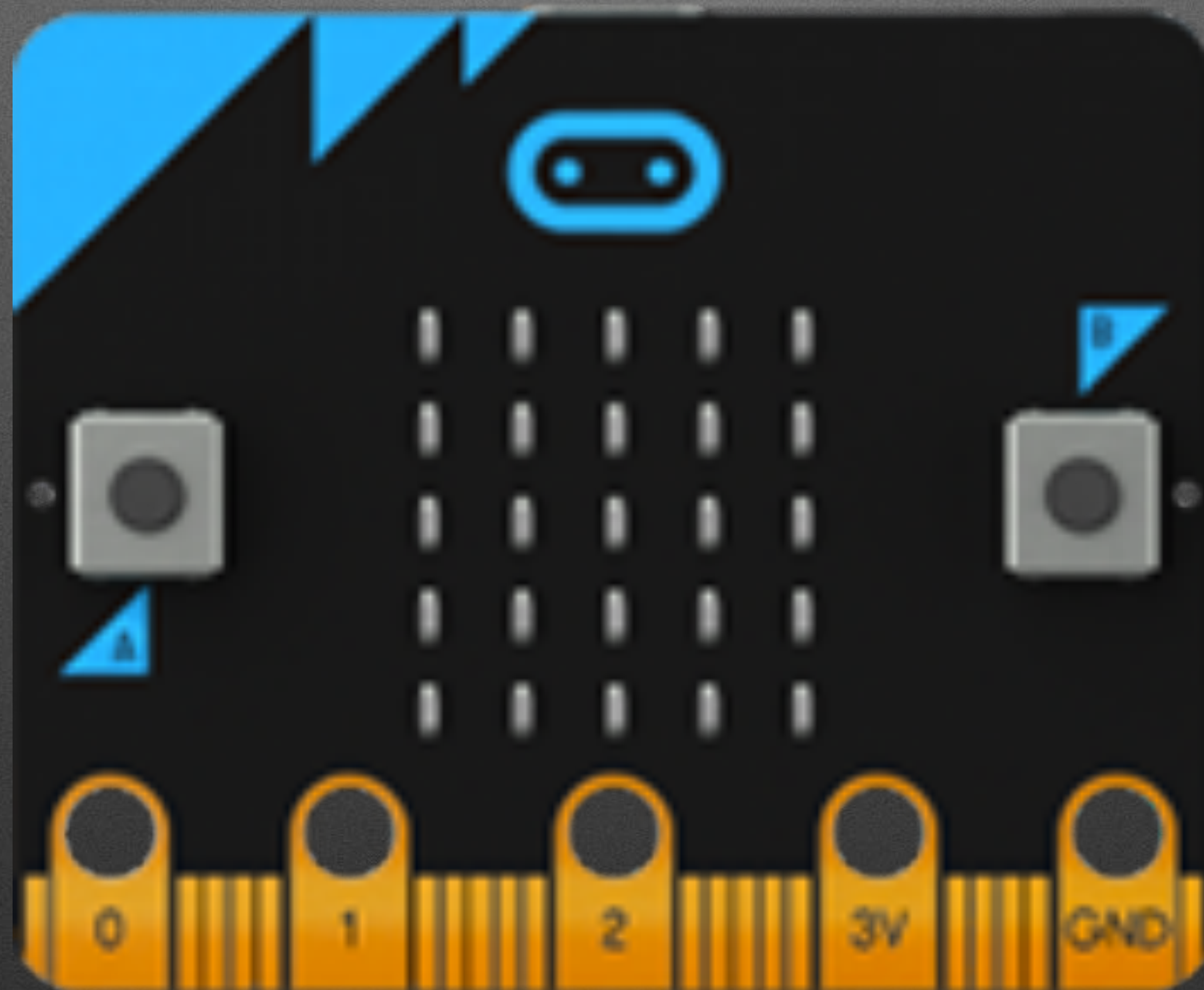
- - Us
  - You: Roll Call & Intros
    - Who has Chrome? Who has an iOS Device with the App?
  - Pair programming —pair up!



# Intros: the micro:bit

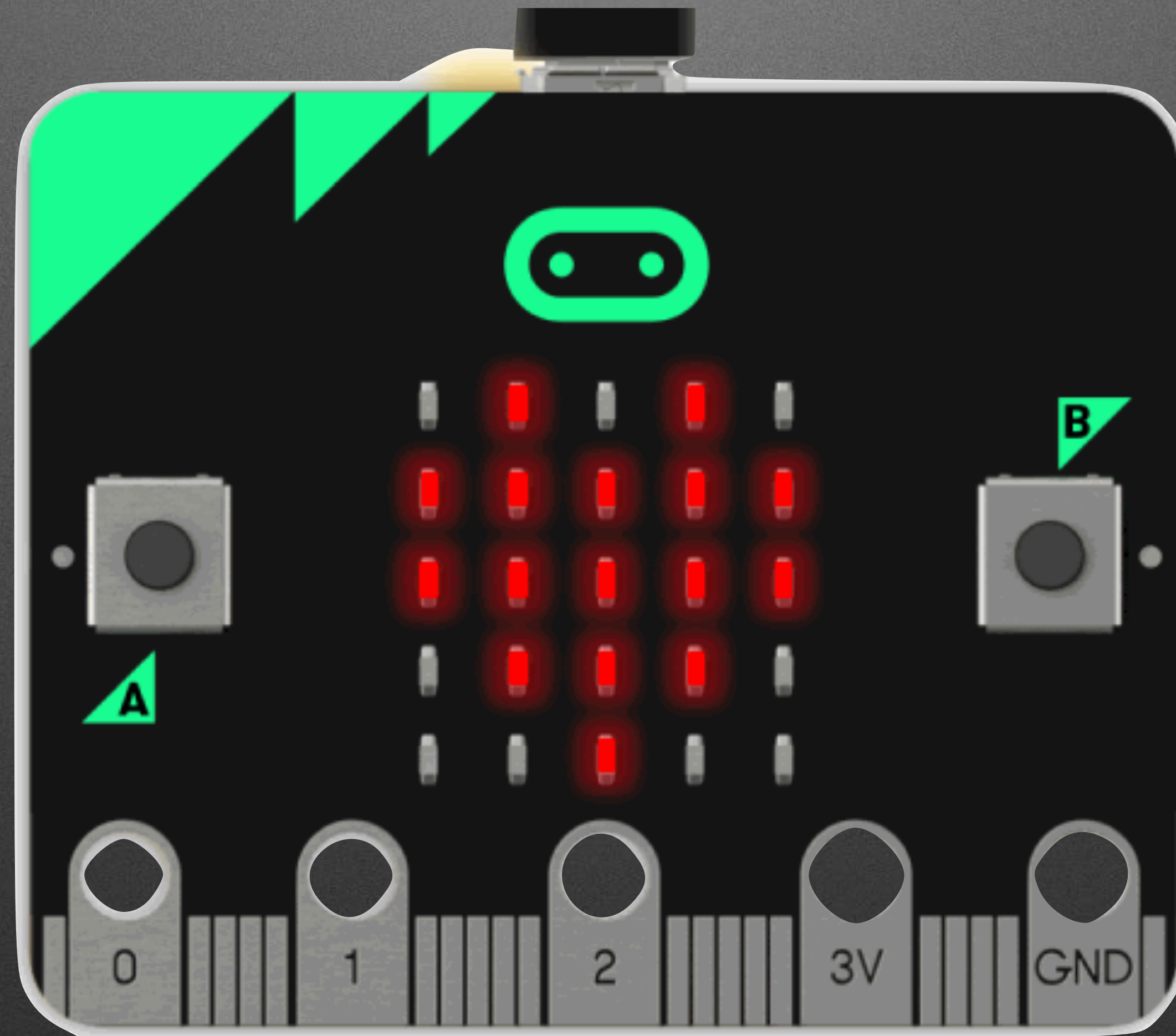


# Small



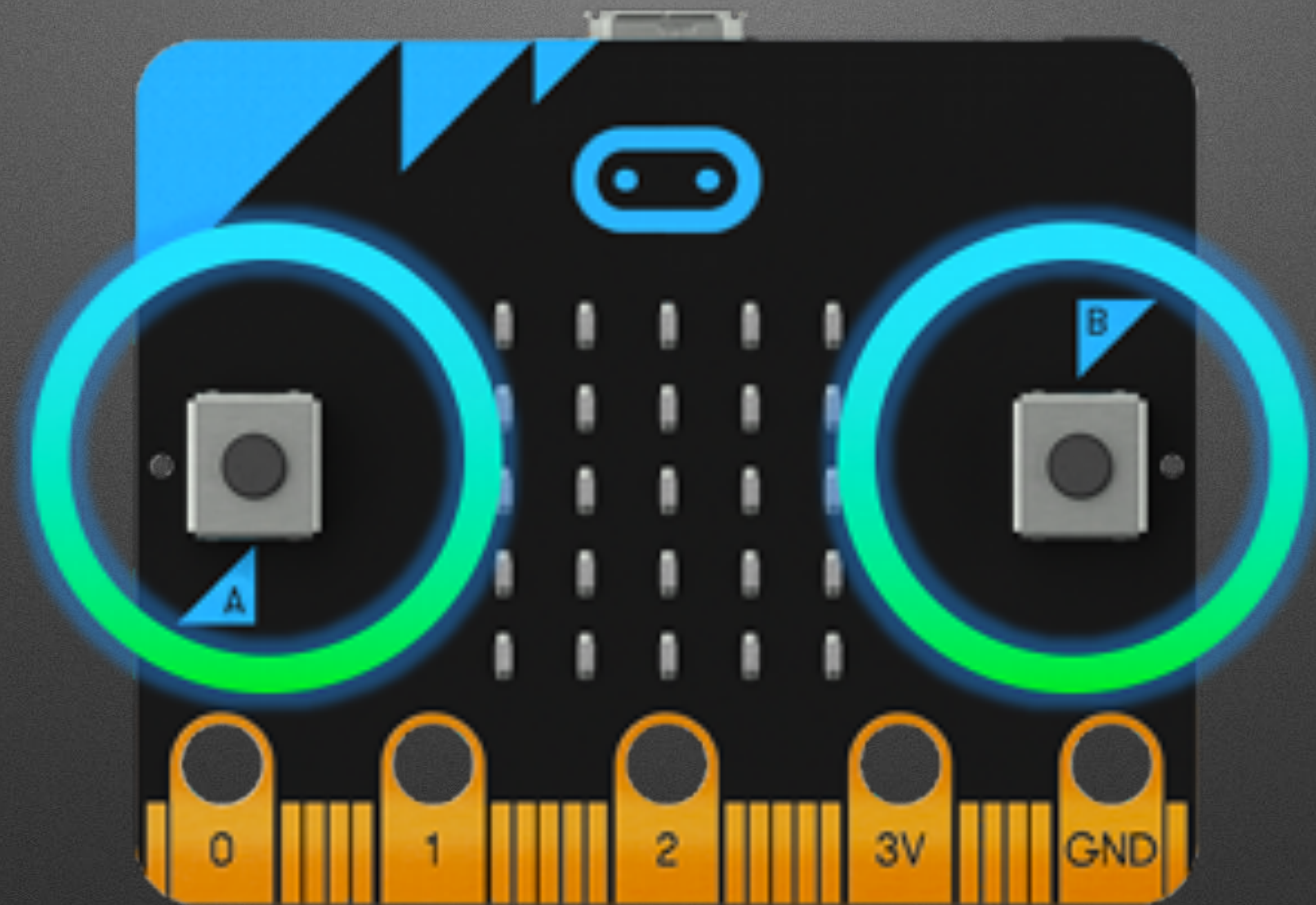


# LED Grid



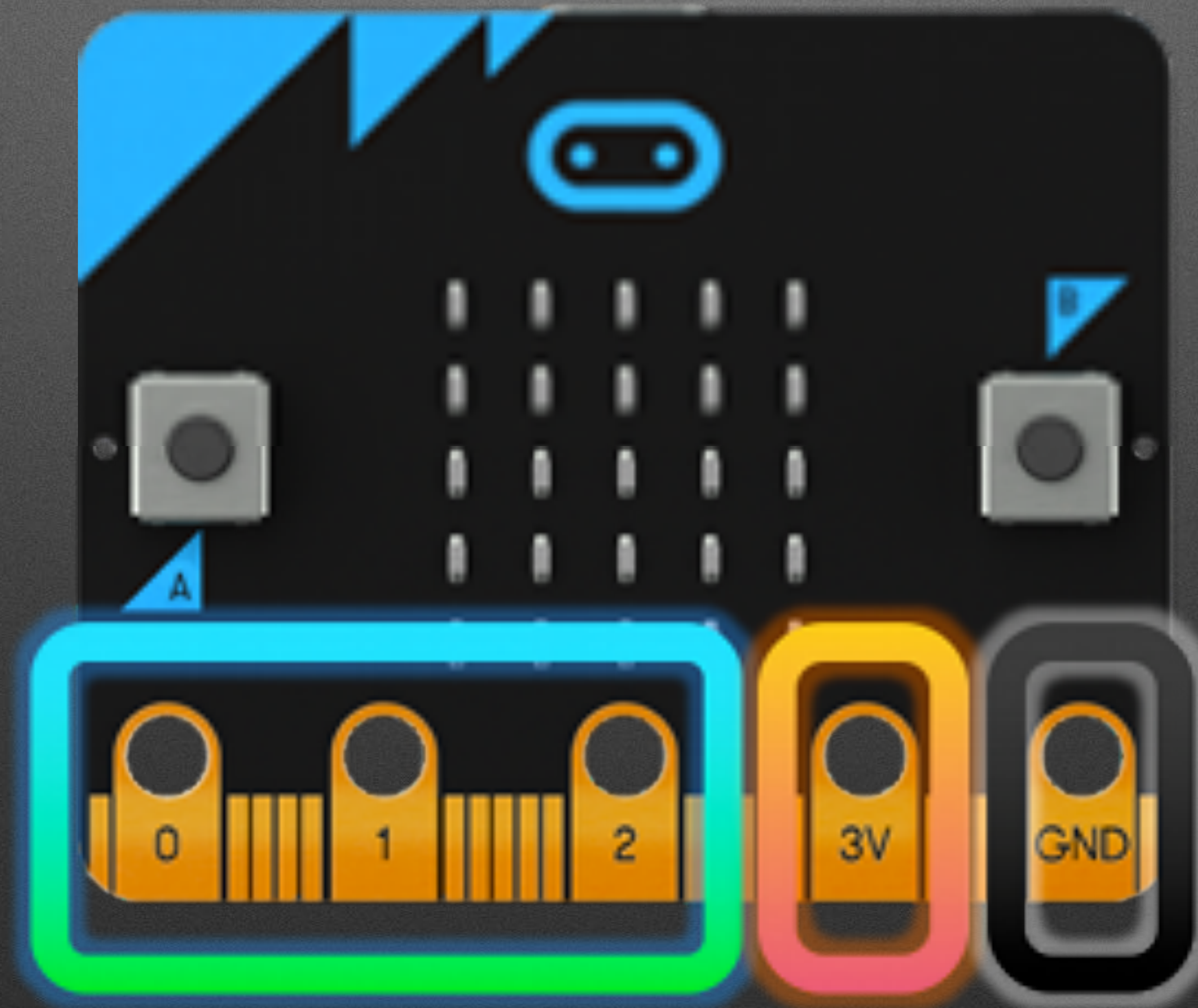


# Buttons



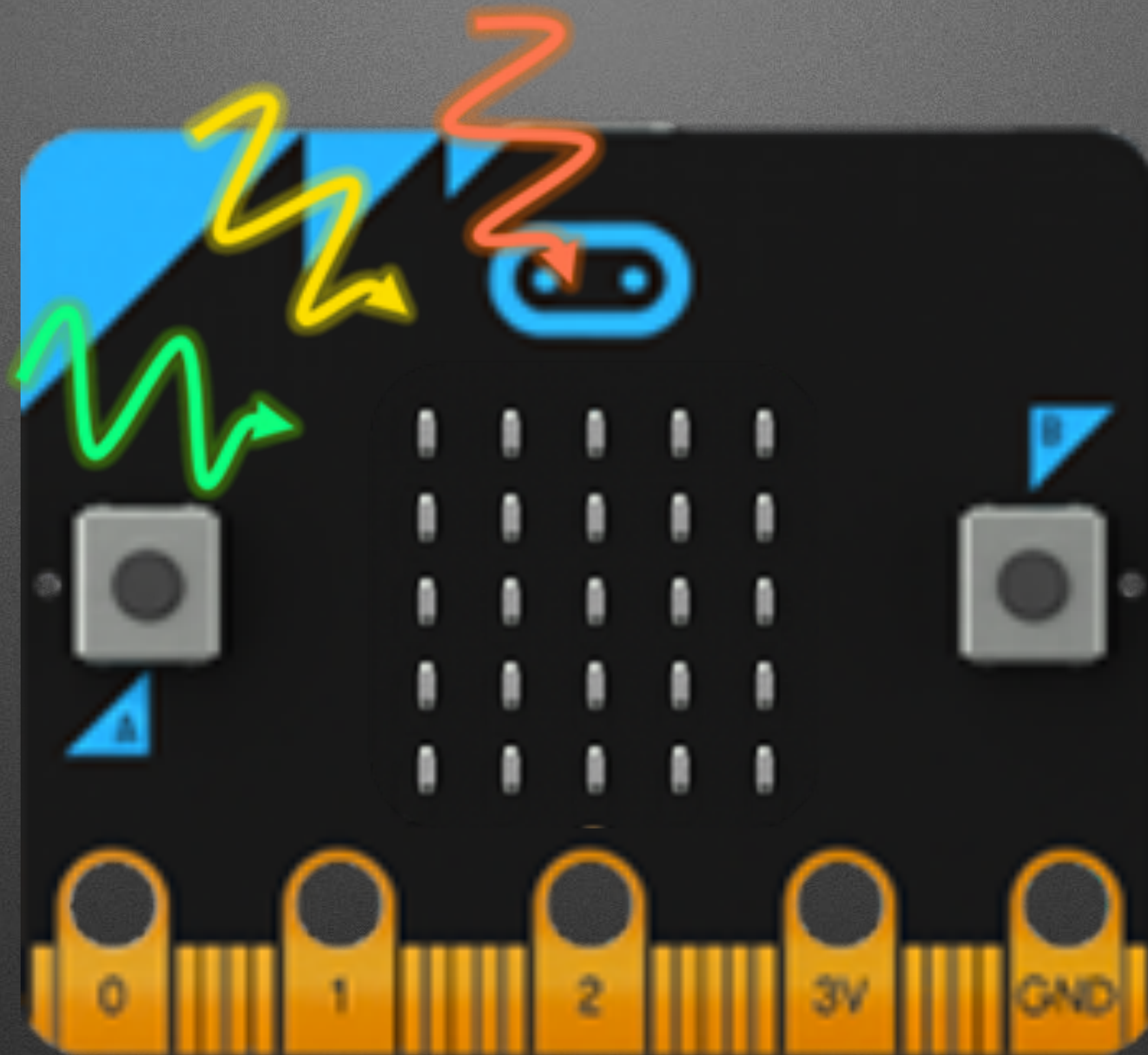


# Connectors



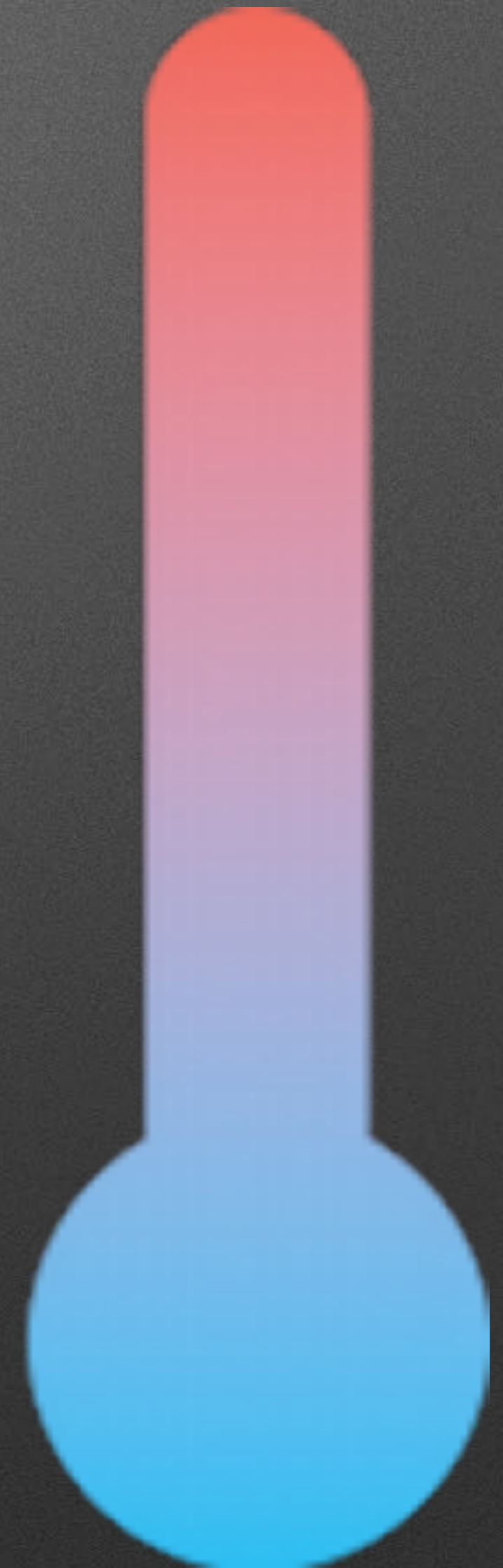
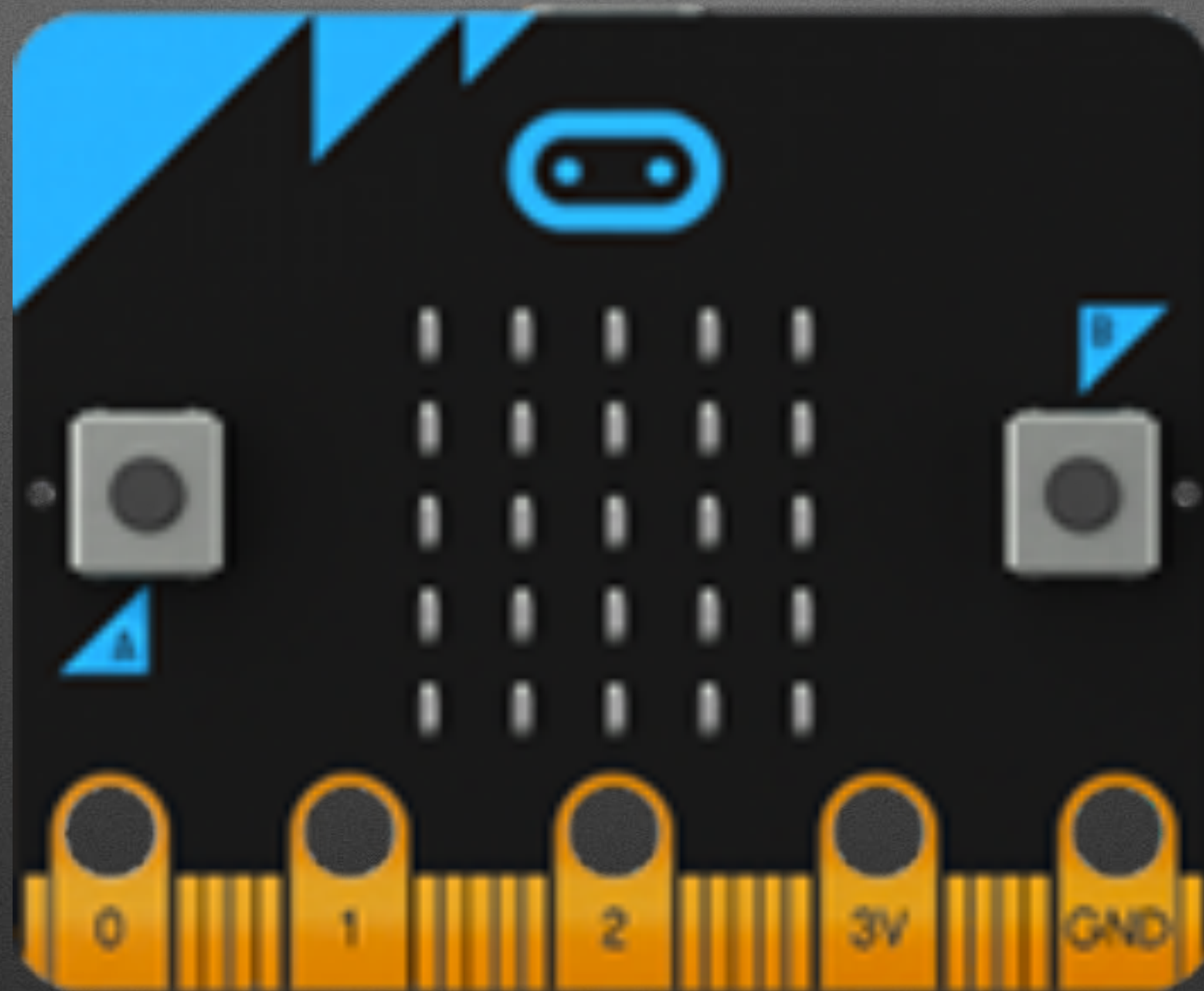


# Light Sensor





# Temperature Sensor



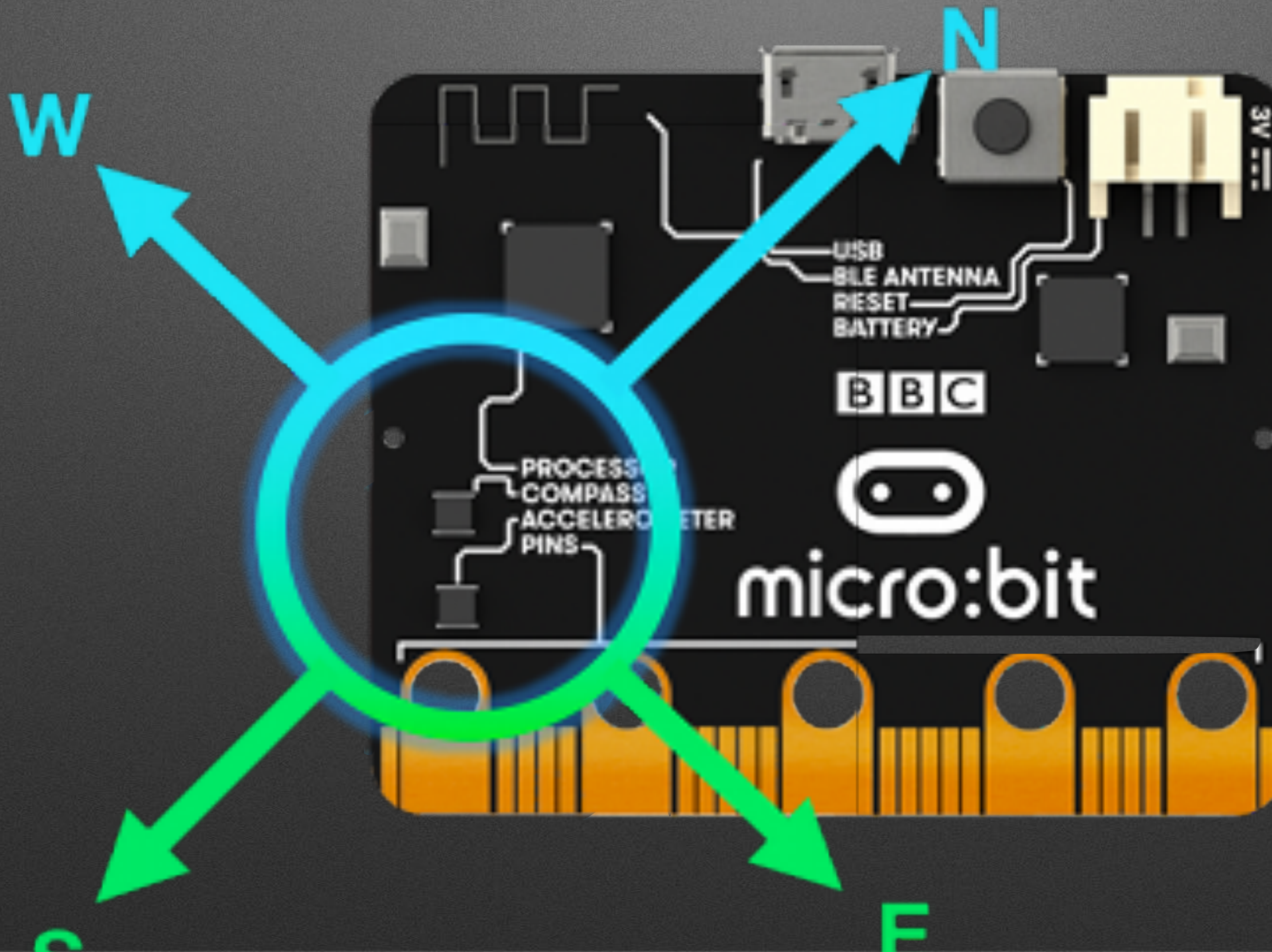


# Accelerometer



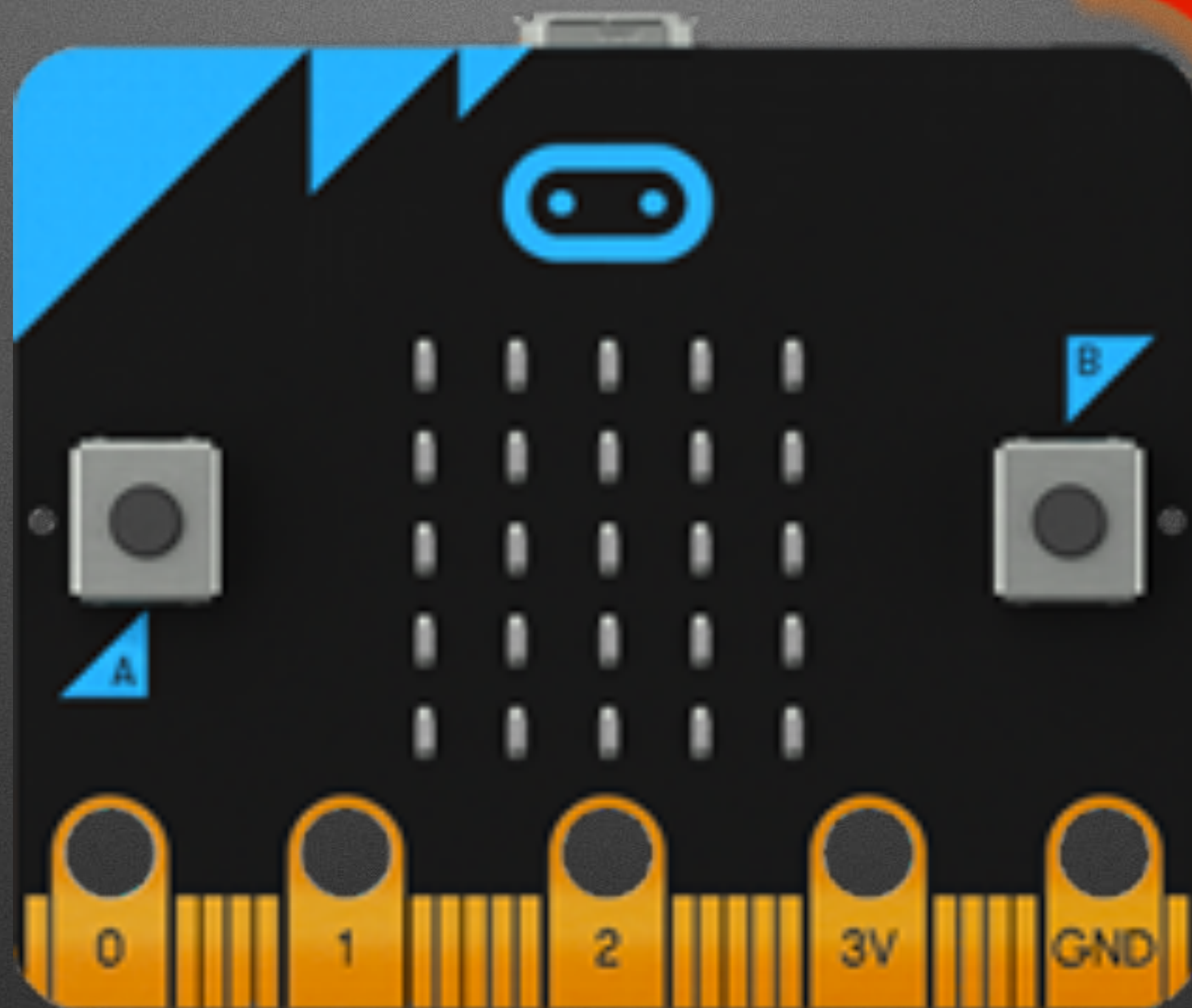


# Compass



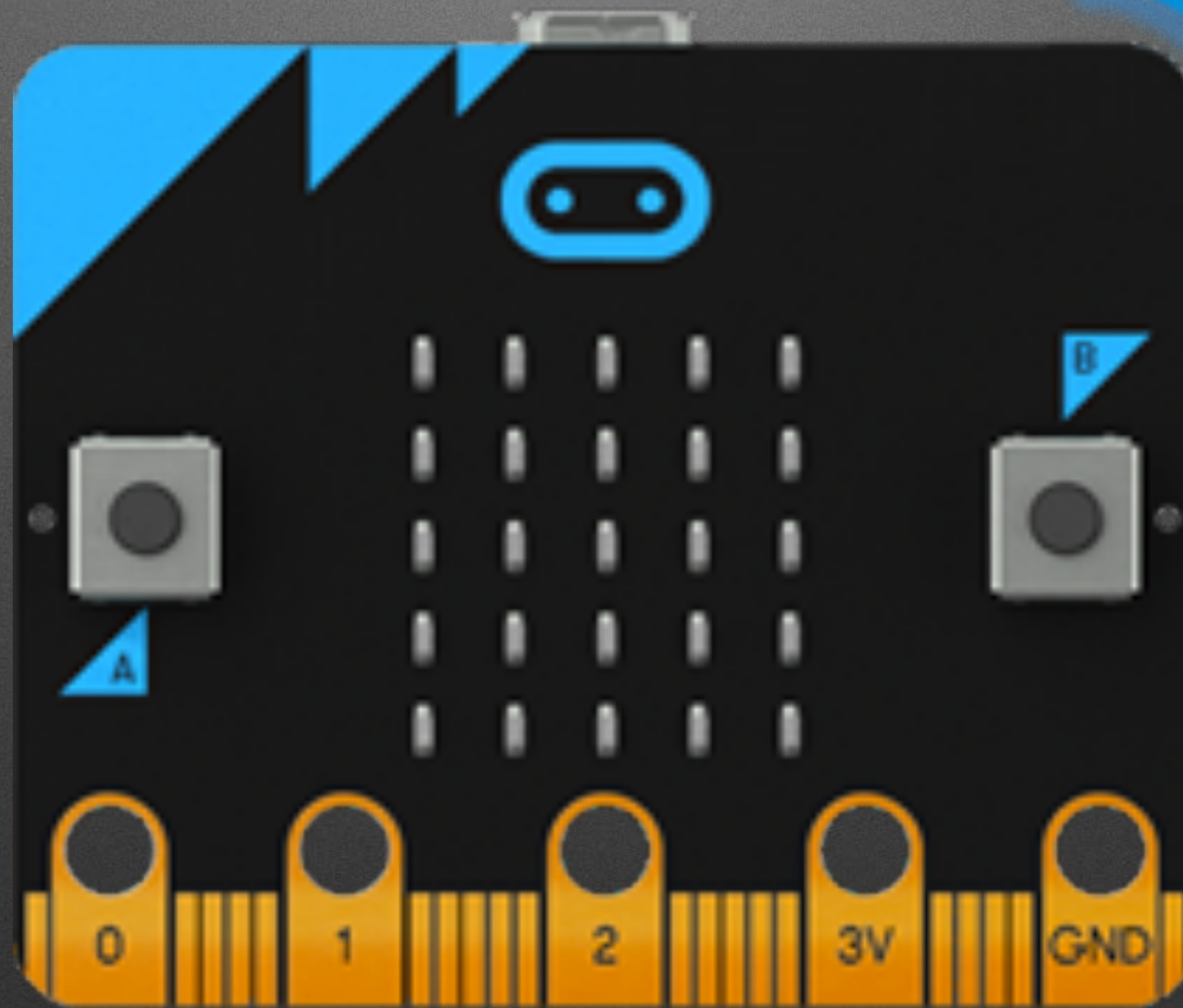


# Radio



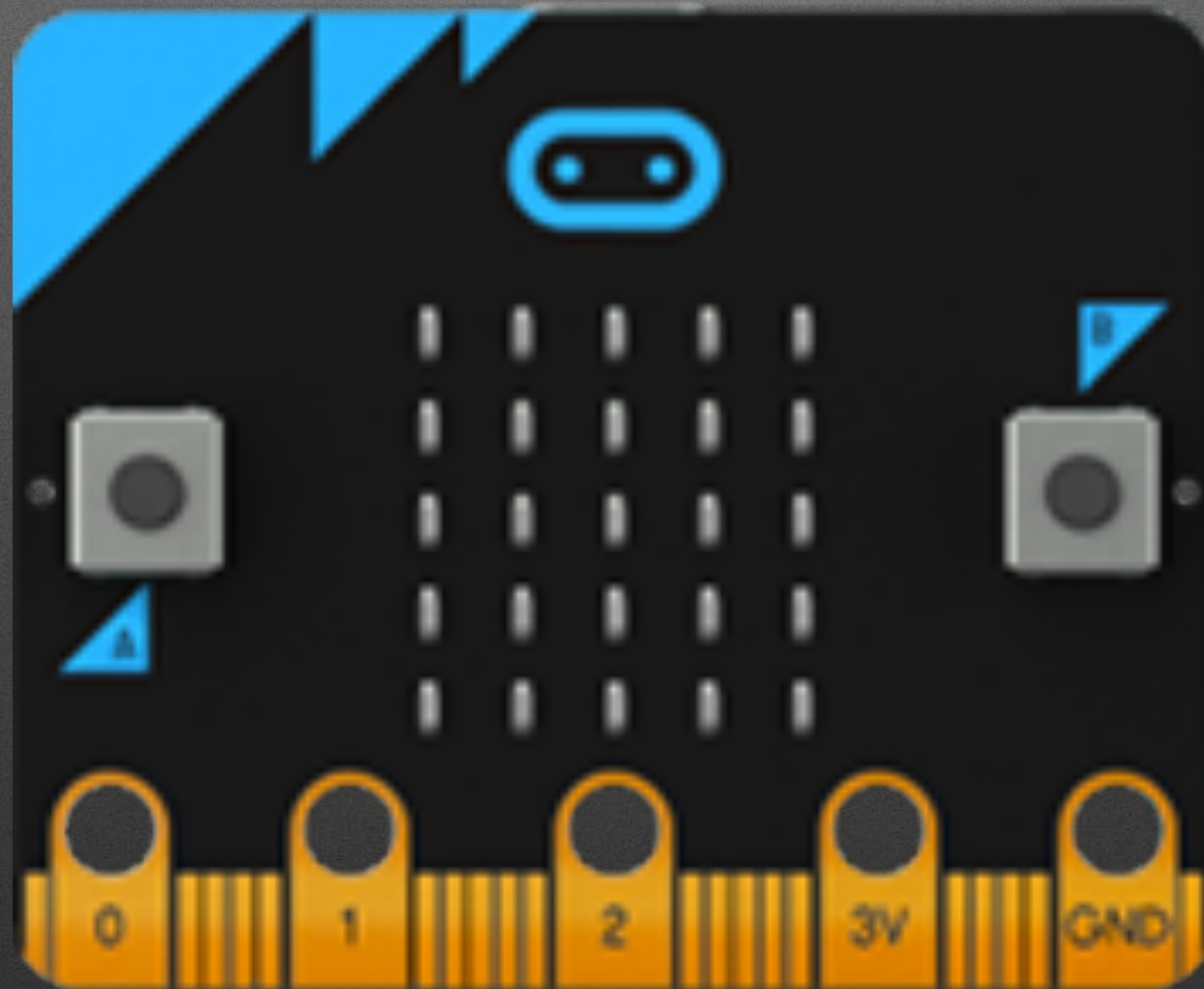


# Bluetooth





**Low Cost: ~\$13 US**



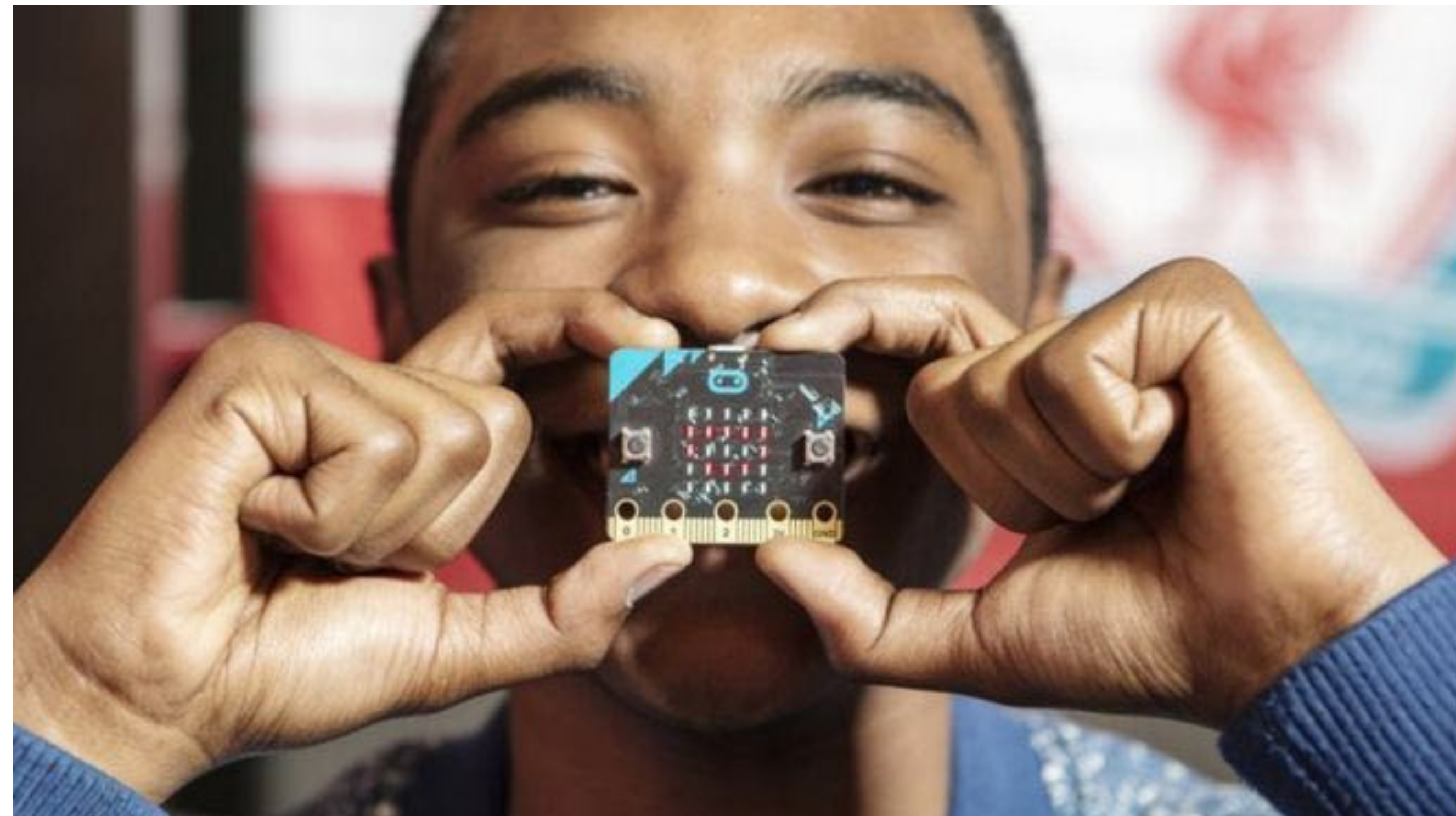


**Thanks: Micro:bit Educational  
Foundation  
and Hal Speed**



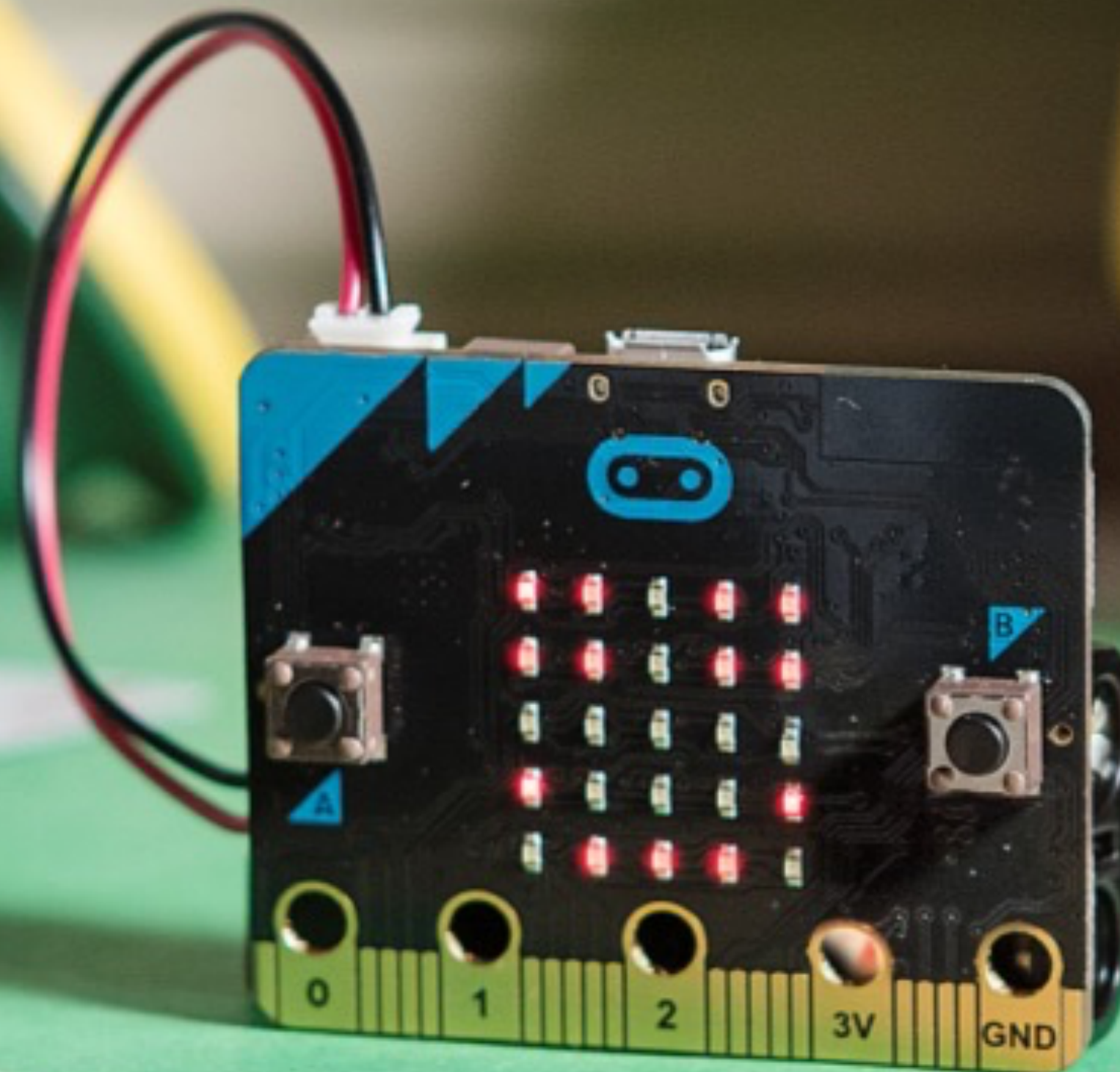
# 2015

- BBC Make It Digital
- 29 partners
- 1 million micro:bit devices
- 11-12 year olds
- Across the U.K.



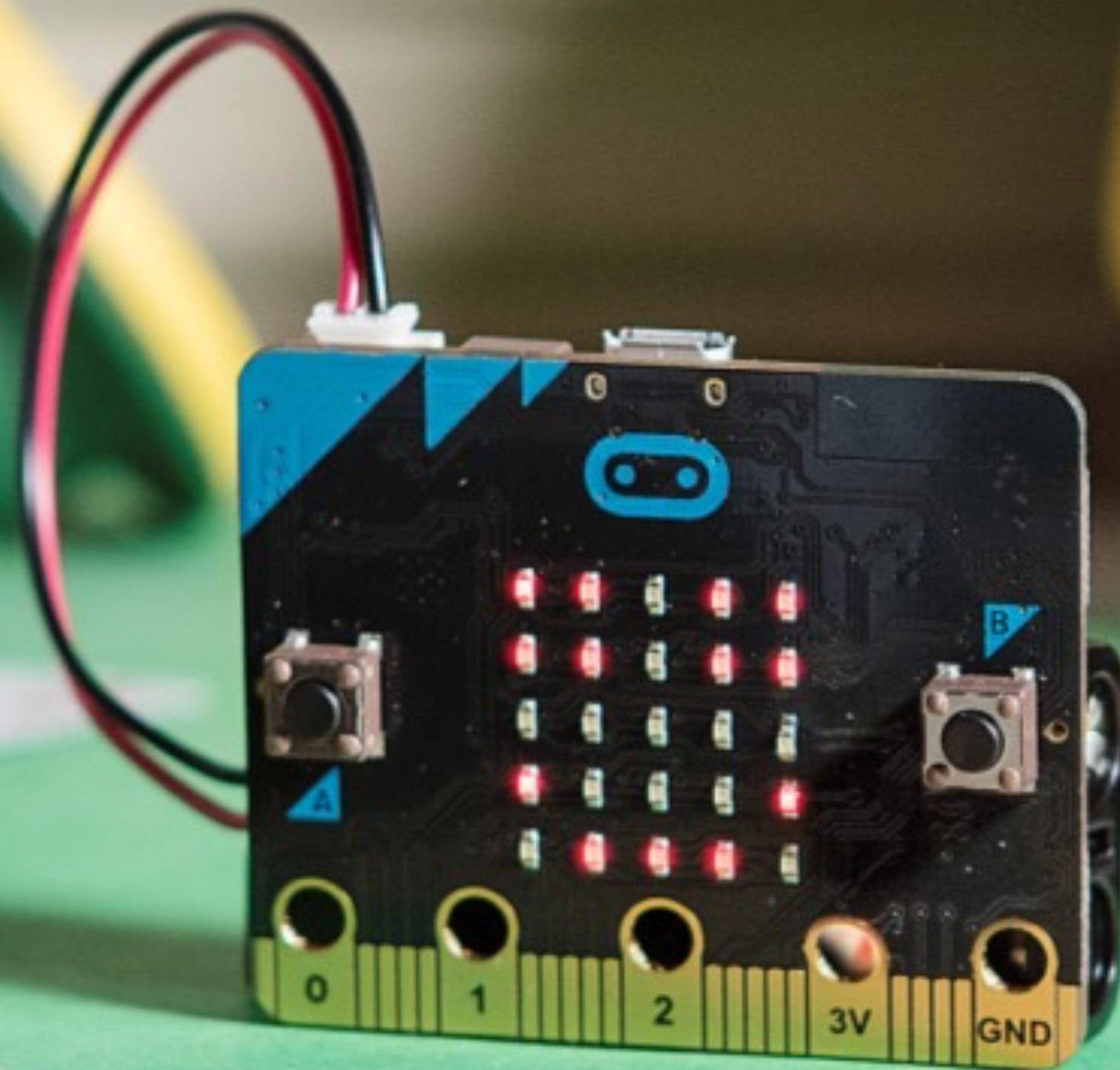


90% of students said it helped  
show that anyone can code





70% increase in the number of girls that said they would definitely choose computing





# 2016 Micro:bit Educational Foundation Formed

To empower children, parents and teachers around the globe to learn and innovate using the micro:bit



2017

micro:bit available in the U.S.



10 New & Innovative EdTech  
Products Announced at ISTE 2017



# Lessons Aligned to Code.org CS Fundamentals

- Lessons extend the concepts taught in the Code.org curriculum by using micro:bit and MakeCode
  - Course E – Loop and Functions
  - Course F – Variables and Conditionals

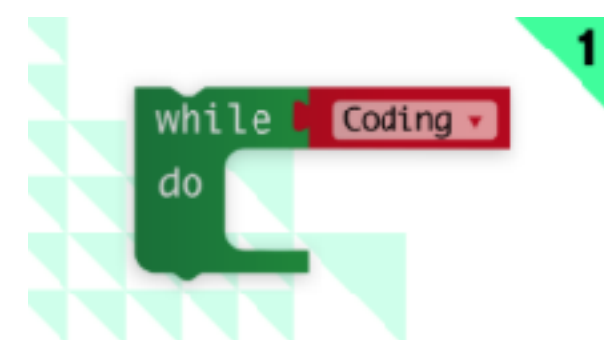


4<sup>th</sup> Grade

Course E

5<sup>th</sup> Grade

Course F



Lesson

**Course E - Loops 1 -  
Loops and  
Animations**



Lesson

**Course E - Loops 2 -  
Nested Loops and  
Scoreboards**



Lesson

**Course E -  
Functions 1 - A  
Simple Function for  
a Superhero**



Lesson

**Course E -  
Functions 2 -  
Functions for a  
Digital Pet**



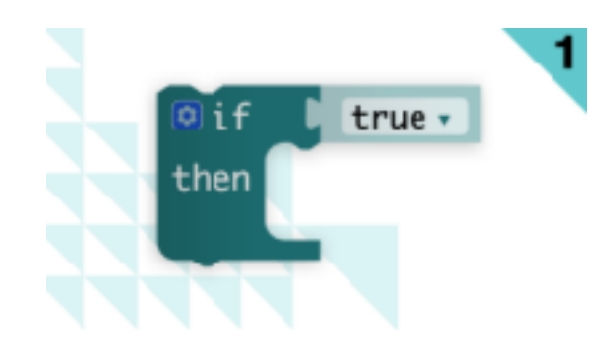
Lesson

**Course F -  
Variables 1 -  
Variables With A  
Counter**



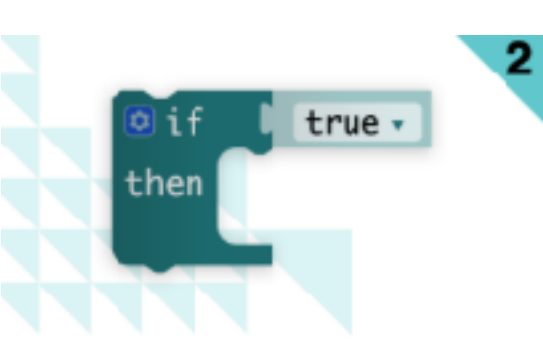
Lesson

**Course F -  
Variables 2 -  
Variables and  
Emotions**



Lesson

**Course F -  
Conditionals 1 -  
Conditionals with  
the Weather  
Predictor**



Lesson

**Course F -  
Conditionals 2 -  
Conditionals with  
Rock Paper  
Scissors Game**



# Third-Party Curricula



Microsoft MakeCode Intro to CS

<https://aka.ms/intro2cs>

- |                           |                               |
|---------------------------|-------------------------------|
| 1. Making                 | 8. Coordinate Grid System     |
| 2. Algorithms             | 9. Booleans                   |
| 3. Variables              | 10. Music and Arrays          |
| 4. Conditionals           | 11. Bits, Bytes, and Binary   |
| 5. Iteration              | 12. Radio                     |
| 6. Review/Mini-Project    | 13. Arrays                    |
| 7. Coordinate Grid System | 14. Independent Final Project |



PLTW Gateway:  
Computer Science for  
Innovators and Makers

<https://www.pltw.org/our-program-pltw-gateway-curriculum#curriculum-4>



**Let's Play**





# **“Hello, World!”: First Program**





# “Hello, World!”: First Program

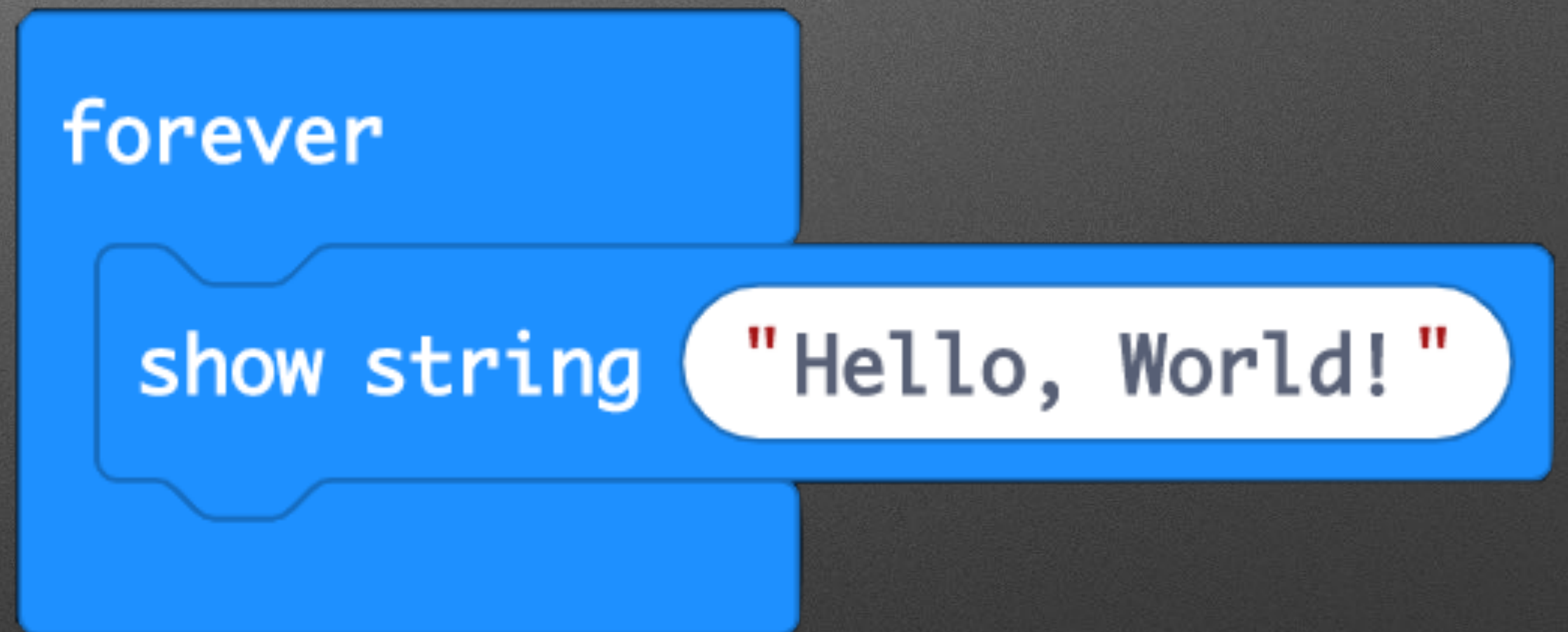
- Block-based editor





# “Hello, World!”: First Program

- Block-based editor

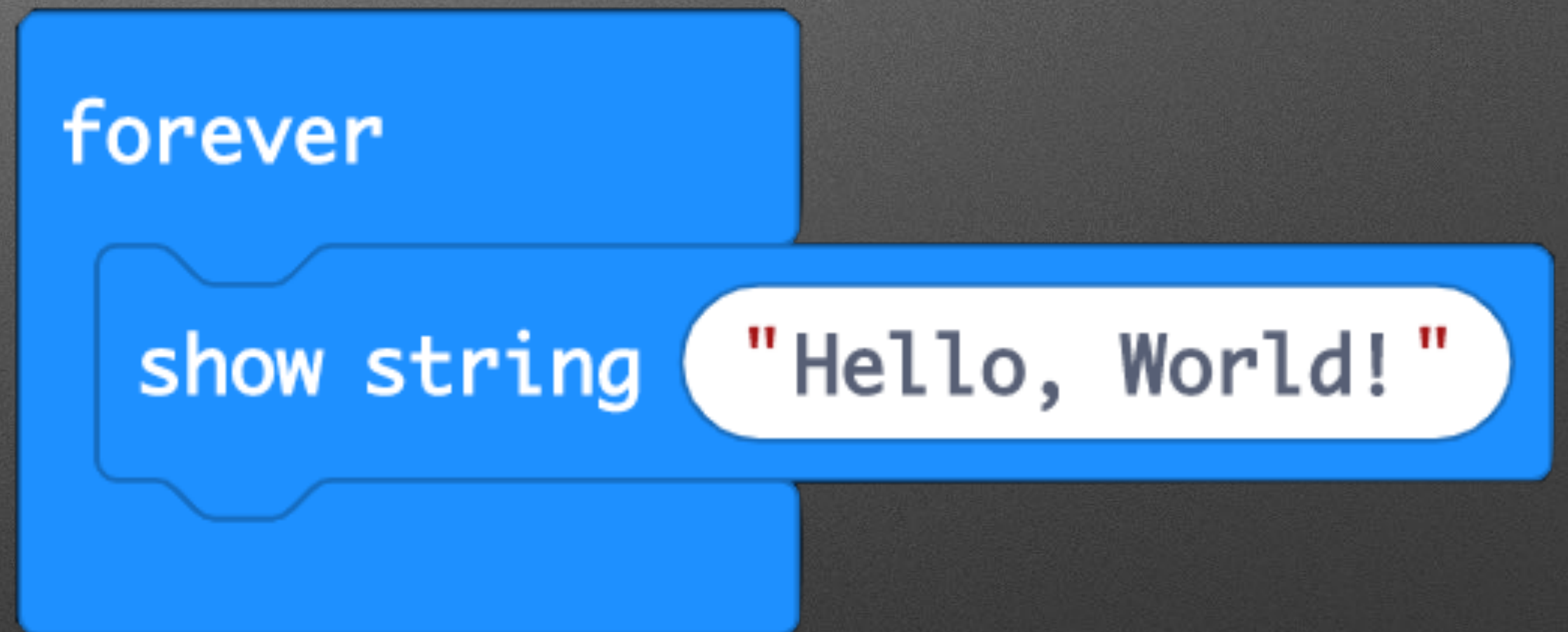






# “Hello, World!”: First Program

- Block-based editor
- Built-in simulator

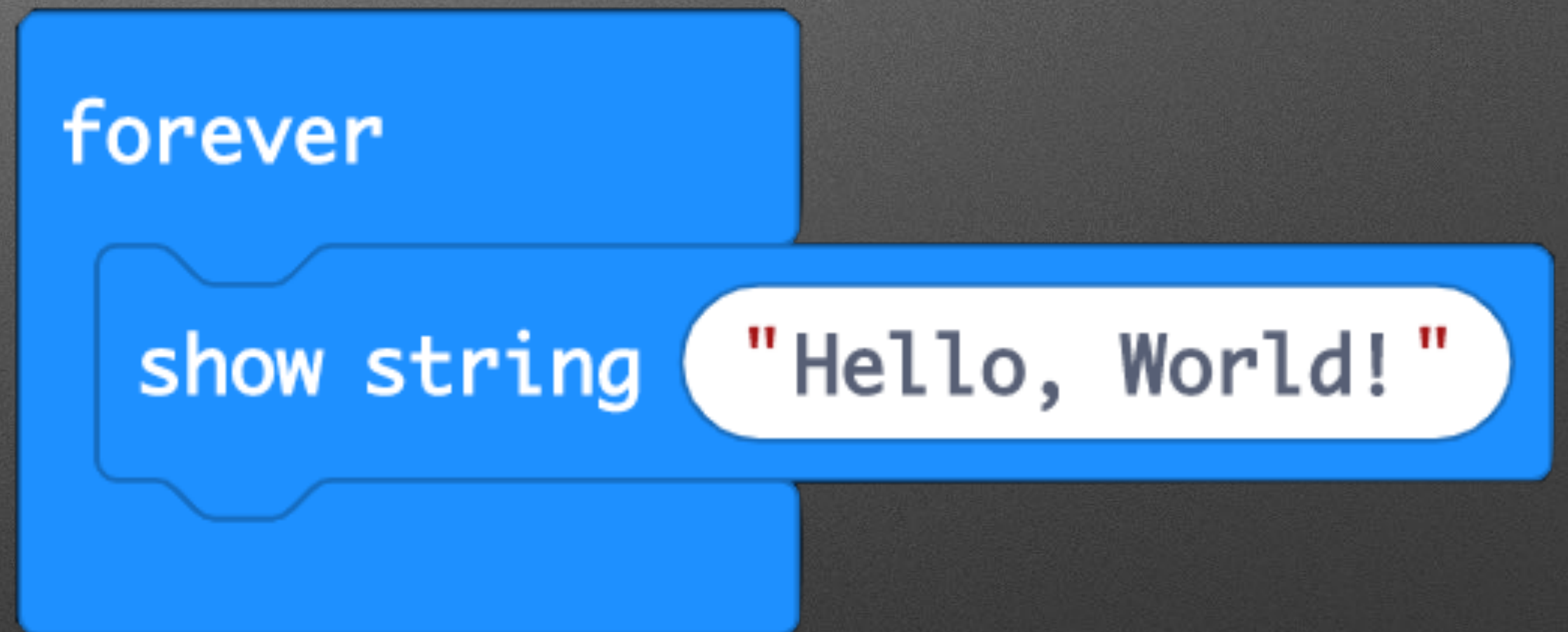






# “Hello, World!”: First Program

- Block-based editor
- Built-in simulator
- Deployment to Micro:bit

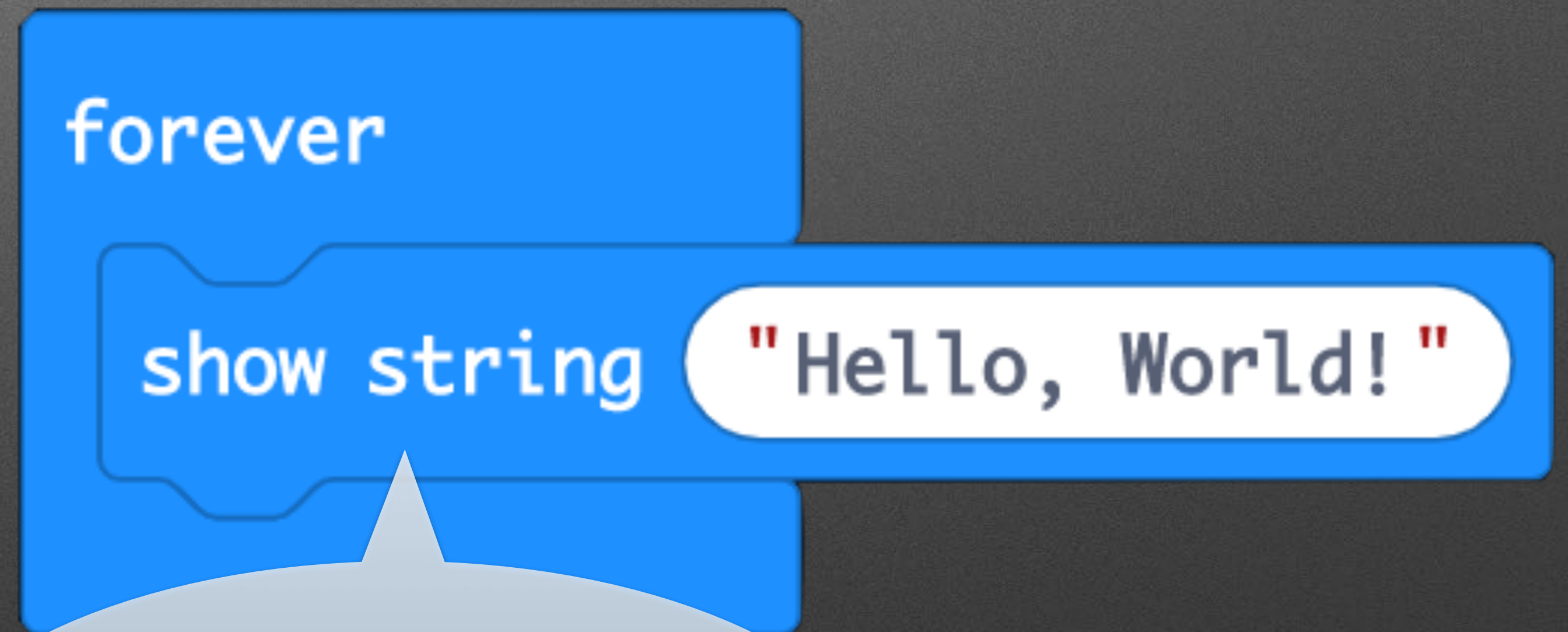






# “Hello, World!”: First Program

- Block-based editor
- Built-in simulator
- Deployment to Micro:bit



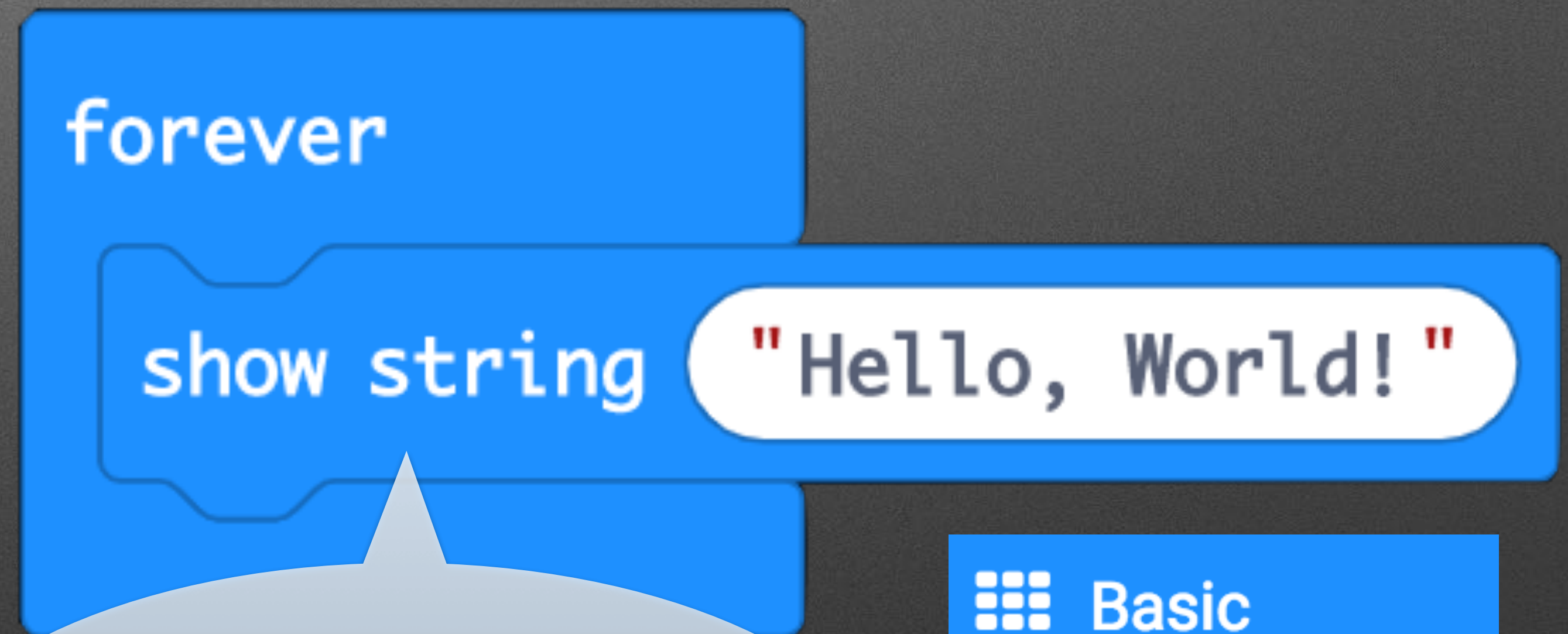
Block Color Indicates Palette



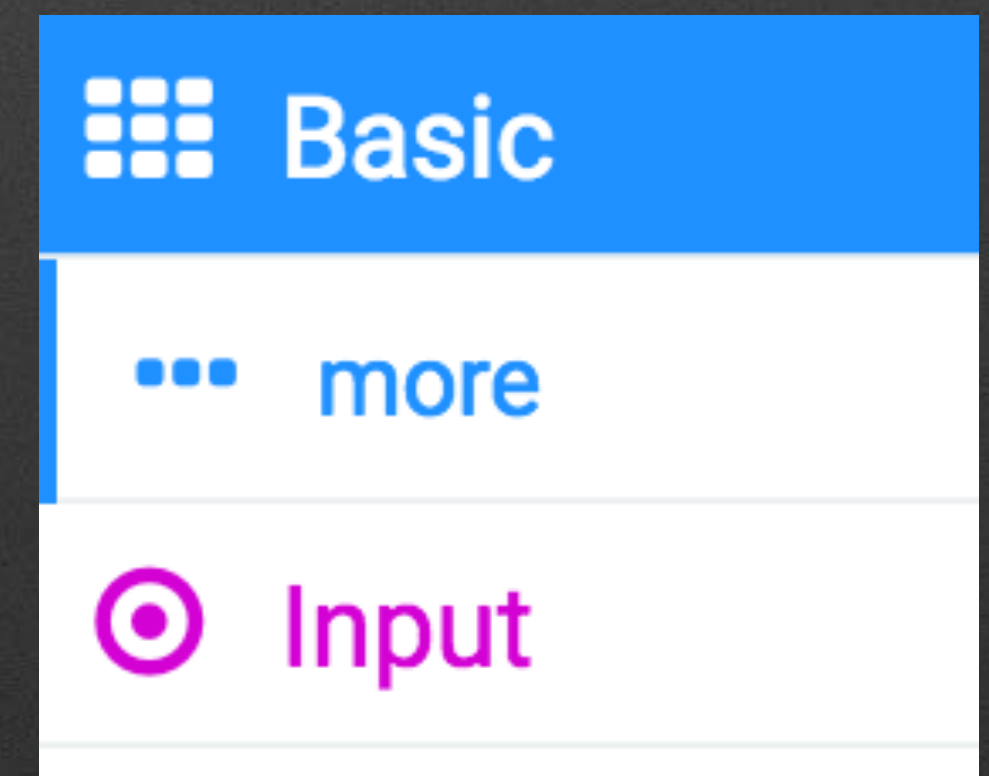


# “Hello, World!”: First Program

- Block-based editor
- Built-in simulator
- Deployment to Micro:bit



Block Color Indicates Palette





**Blocks are just the beginning...**



# Blocks are just the beginning...

- JavaScript



# Blocks are just the beginning...

- JavaScript
- Python w/ REPL



# Blocks are just the beginning...

- JavaScript
- Python w/ REPL
- Arduino / C++



# Blocks are just the beginning...

- JavaScript
- Python w/ REPL
- Arduino / C++
- Commercial IDEs / C++



# Workshop Format



# Workshop Format

- Moderate pace with small examples



# Workshop Format

- Moderate pace with small examples
- Only covering blocks-based approach



# Workshop Format

- Moderate pace with small examples
- Only covering blocks-based approach
- Will cover many “building blocks”, but not much depth



# Workshop Format

- Moderate pace with small examples
- Only covering blocks-based approach
- Will cover many “building blocks”, but not much depth
  - Putting pieces together for awesome projects left as an exercise for you...





# Setup

- Hardware Handout
  1. Open Box
  2. Pull out micro:bit
  3. Pull out micro USB cable (under cardboard)
  4. Connect via USB cable





# Setup

- Browser
  1. Open [microbit.org](https://microbit.org)
  2. Select “Let’s Code”
  3. Click “Let’s Code” button on MakeCode
  4. Select “New Project”

Let's Code

Let's Code

My Projects



New Project



# Personalization!



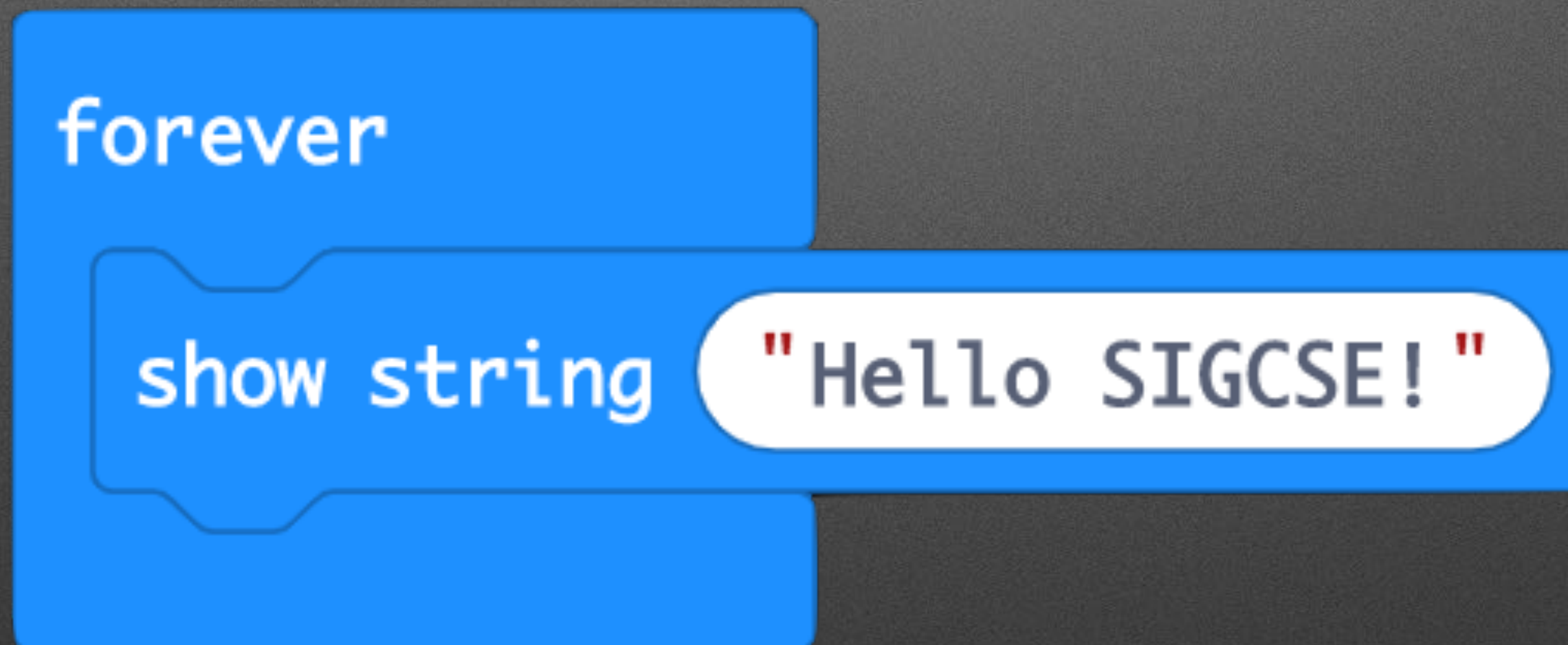
# Personalization!

- Hello Bill / Hello Michael / Hello ....



# Personalization!

- Hello Bill / Hello Michael / Hello ....





# Aside: Text-based Languages



 Home



 Blocks

 JavaScript



# Aside: Storage



# Aside: Storage

- Projects are stored in the cloud



# Aside: Storage

- Projects are stored in the cloud
  - No accounts (by default, but GitHub repositories can be used)



# Aside: Storage

- Projects are stored in the cloud
  - No accounts (by default, but GitHub repositories can be used)
  - Based on *machine you're on!*



# Aside: Storage

- Projects are stored in the cloud
  - No accounts (by default, but GitHub repositories can be used)
  - Based on *machine you're on!*
- But...Downloaded files can be restored via Drag & Drop!



# WebUSB



# WebUSB

- Why: Get rid of Files!



# WebUSB

- Why: Get rid of Files!
- Faster programming



# WebUSB

- Why: Get rid of Files!
  - Faster programming
  - Additional Features: a Console!



# WebUSB

- Why: Get rid of Files!
  - Faster programming
  - Additional Features: a Console!
- How: Chrome 65+ & Setup

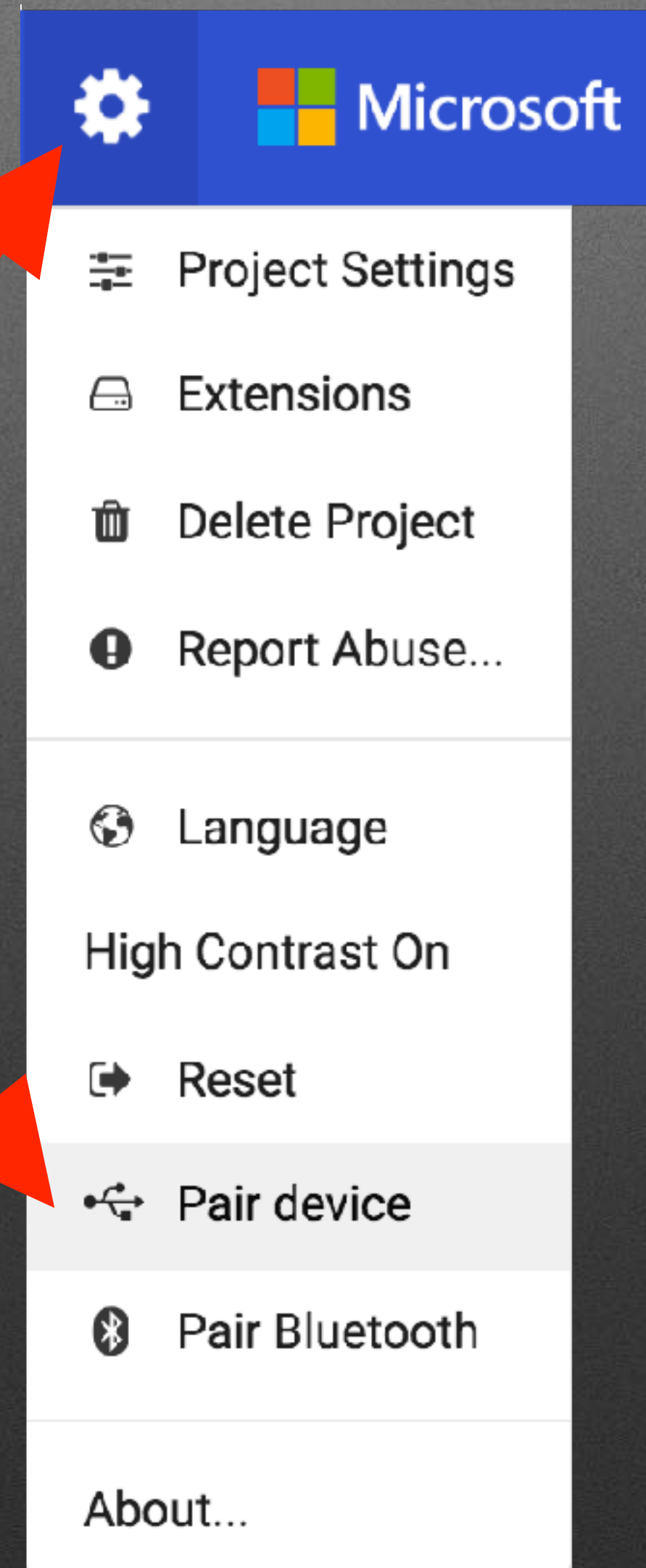




# WebUSB Setup

1. Go to Gear Menu

2. Select Pair Device








# Setup: Chrome v65+

## 3. Select Pair Device


Pair device for one-click downloads

First time here?


You must have version 0249 or above of the firmware



Check your firmware version [here](#) and update if needed




**1** Connect the micro:bit to your computer with a USB cable  
Use the microUSB port on the top of the micro:bit



**2** Pair your micro:bit  
Click 'Pair device' below and select BBC micro:bit CMSIS-DAP or DAPLink CMSIS-DAP from the list

Help ?

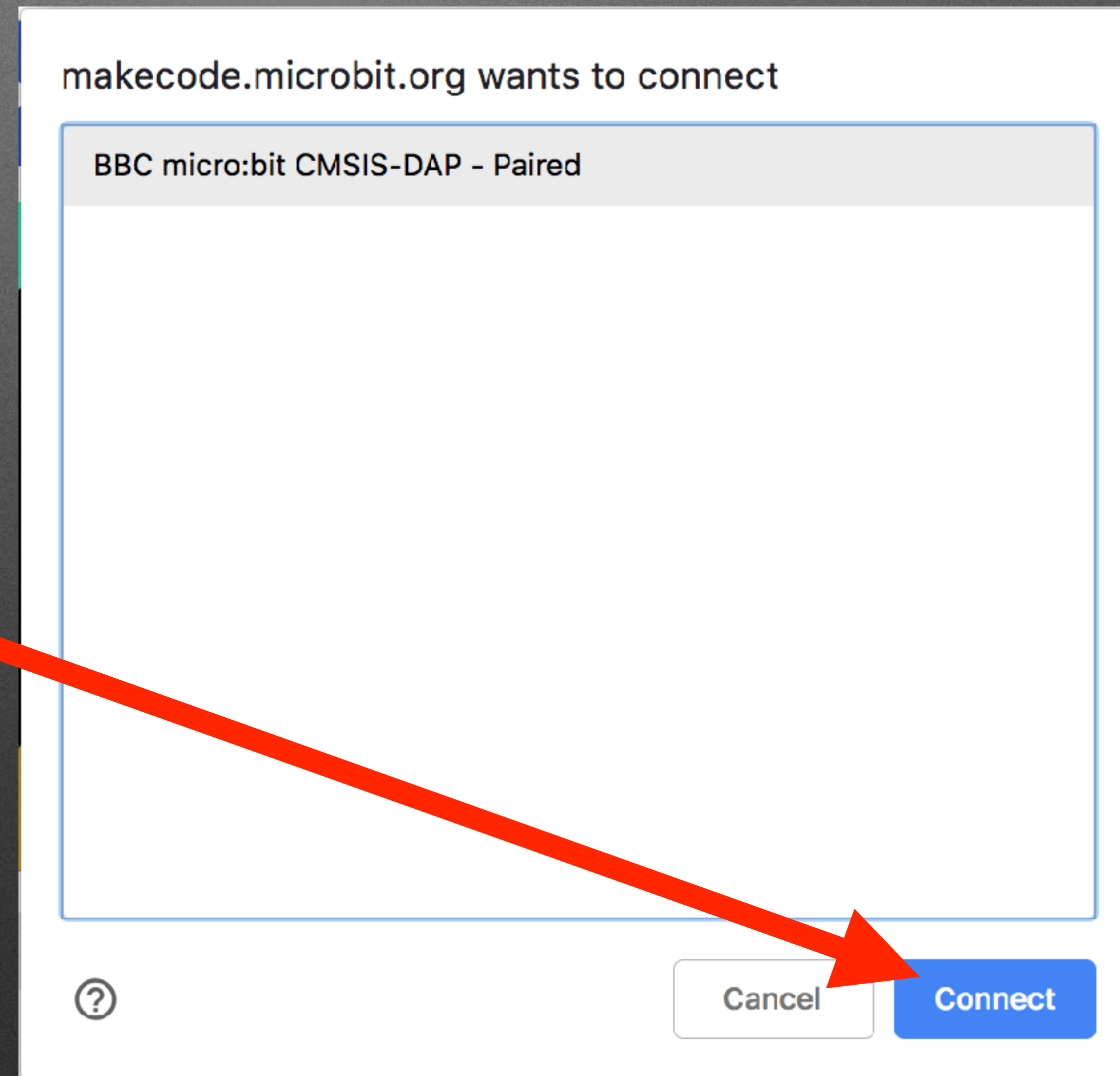
Pair device 





# Setup: Chrome v65+

4. Connect





# Try It!

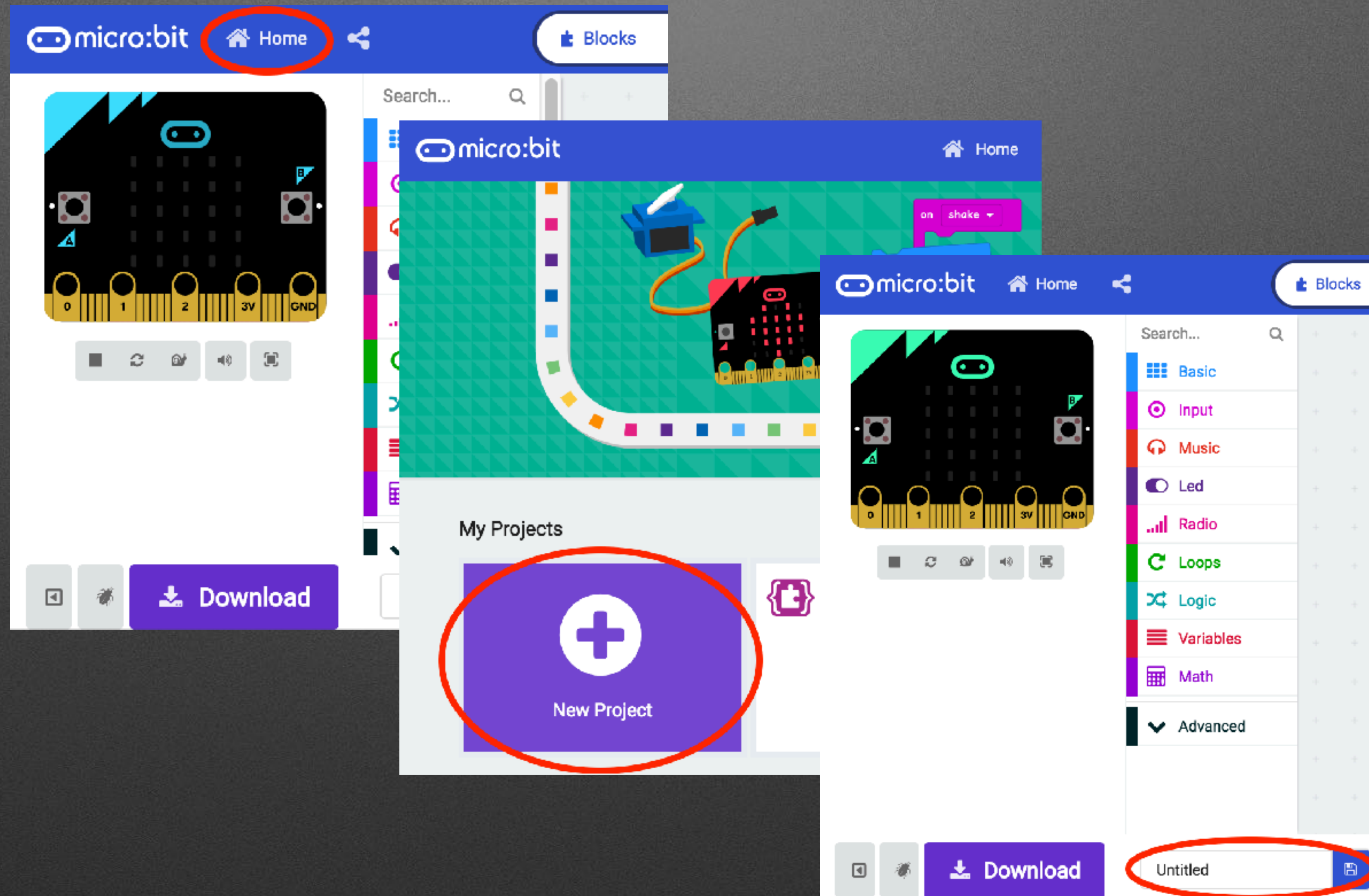
 [Download](#)



**New Project: Home > New Project...**



# New Project: Home > New Project...







# Programming: Logic & Action

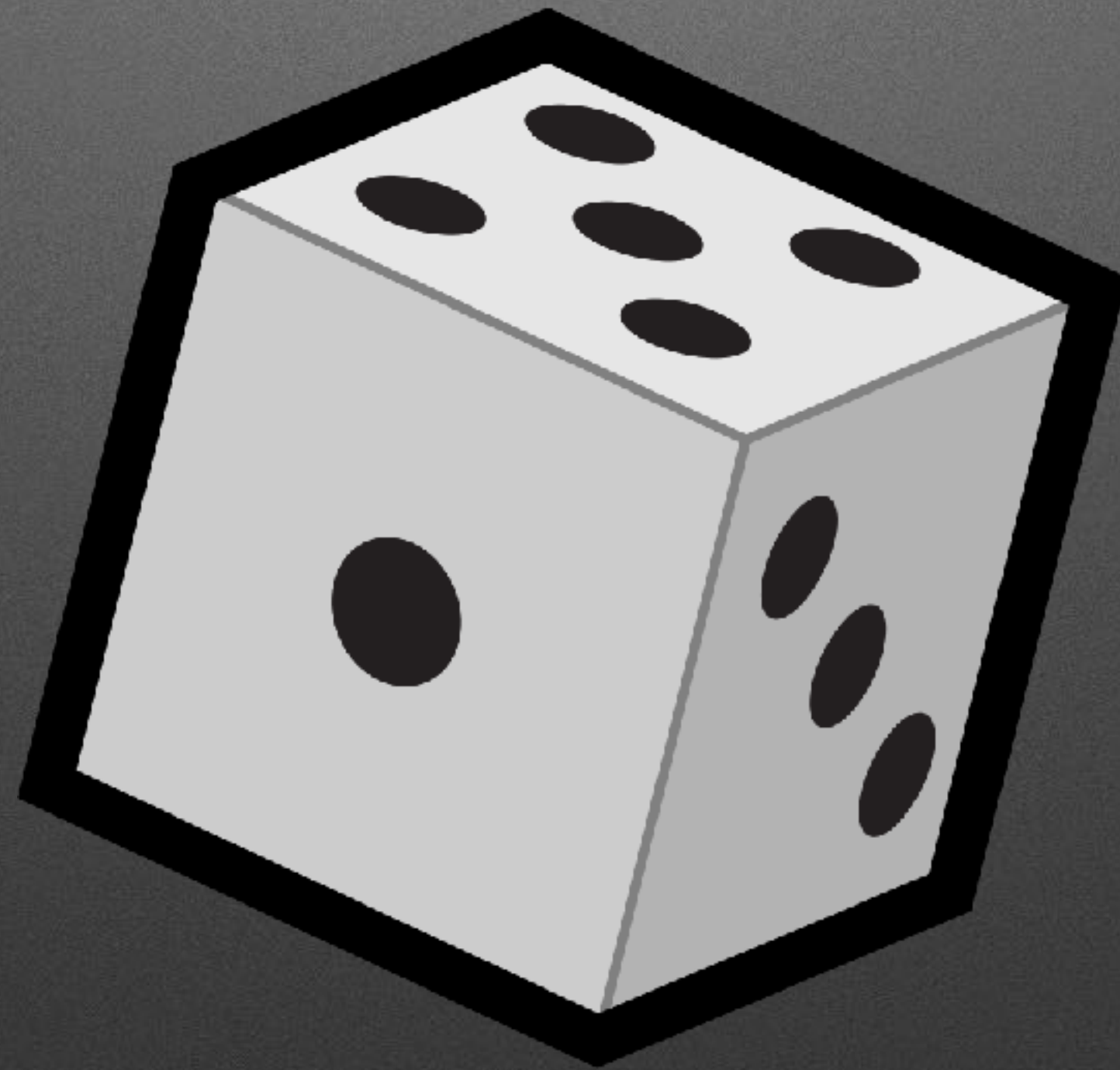
- Picking between *three* tough choices
  - Cookie, Cake, Pie
  - Super Strength, Invisibility, Telekinesis
  - ...



# Obvious Solution...

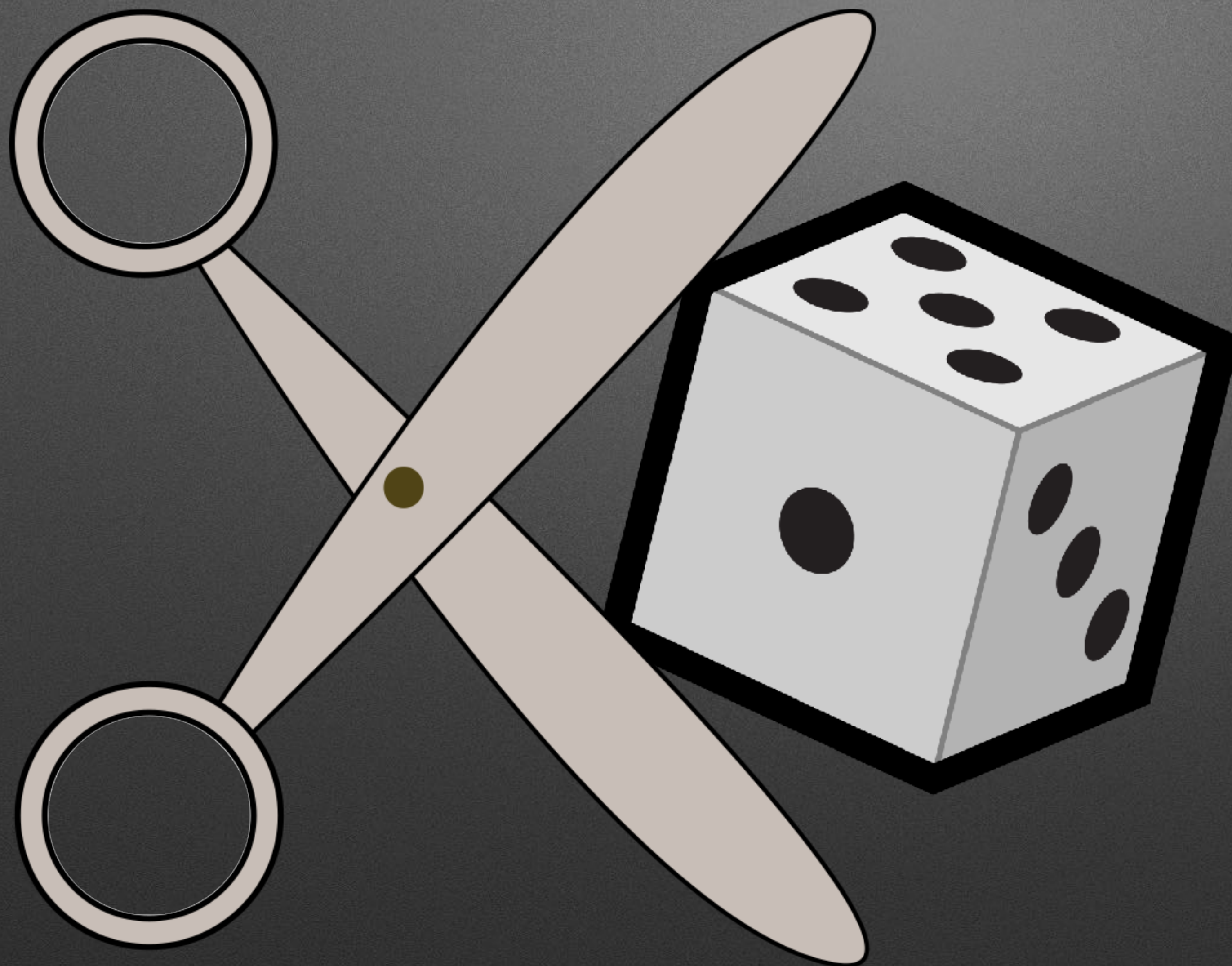


# Obvious Solution...



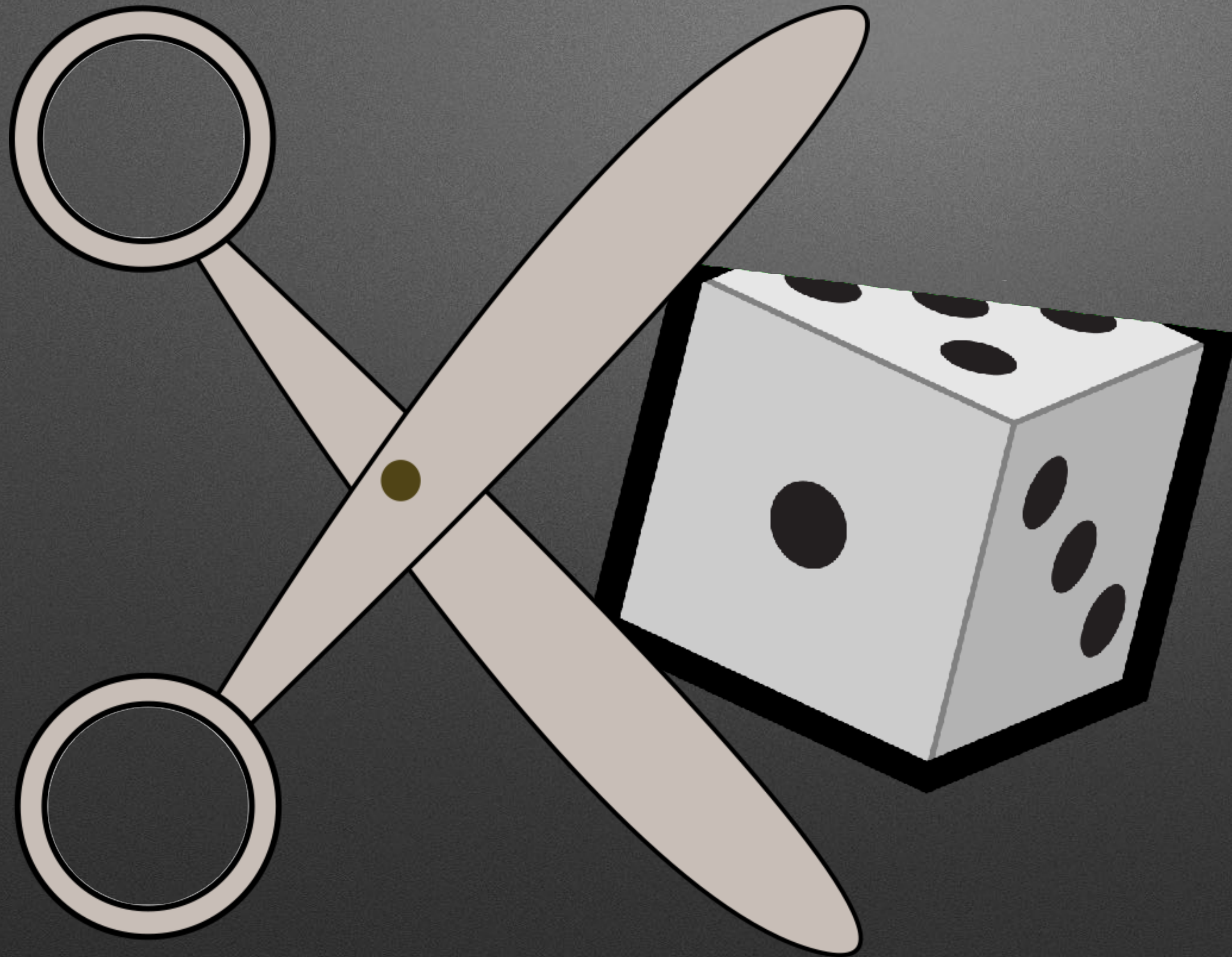


# Obvious Solution...



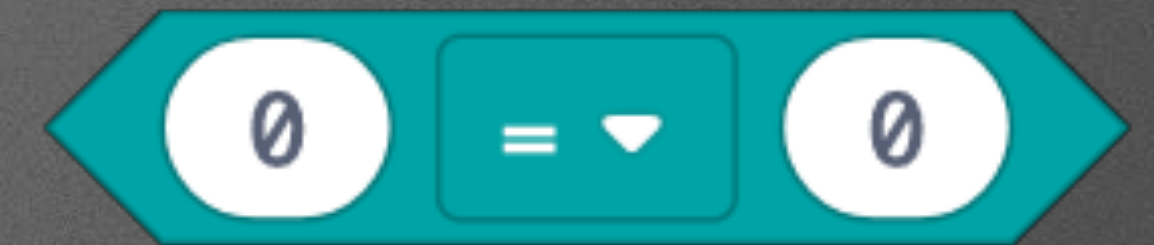
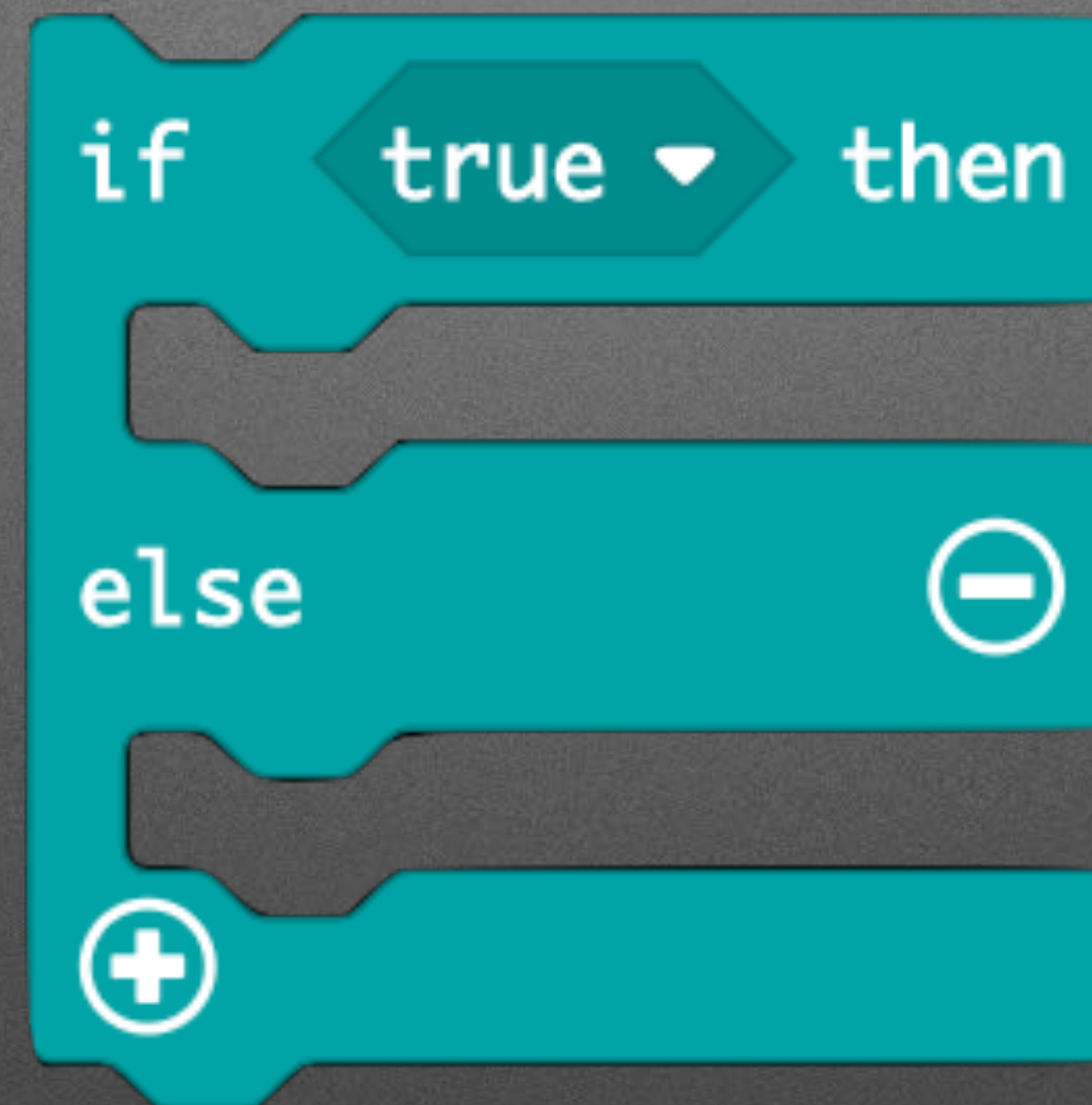
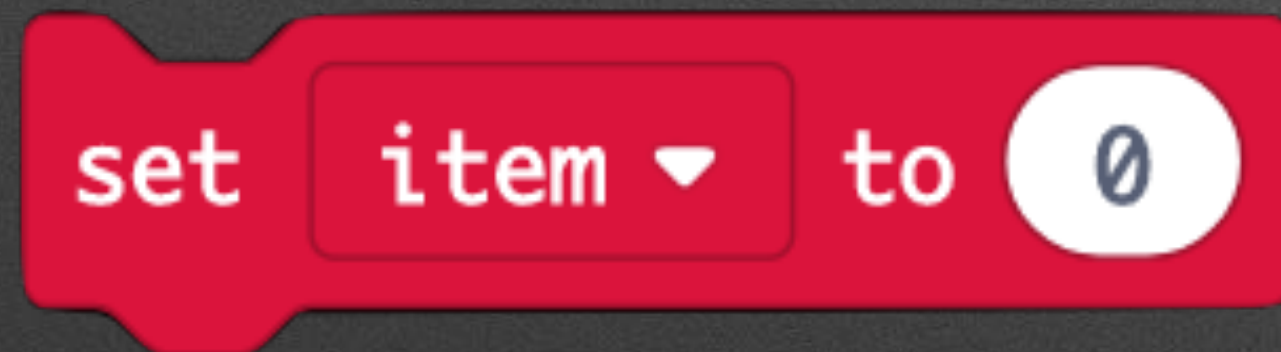
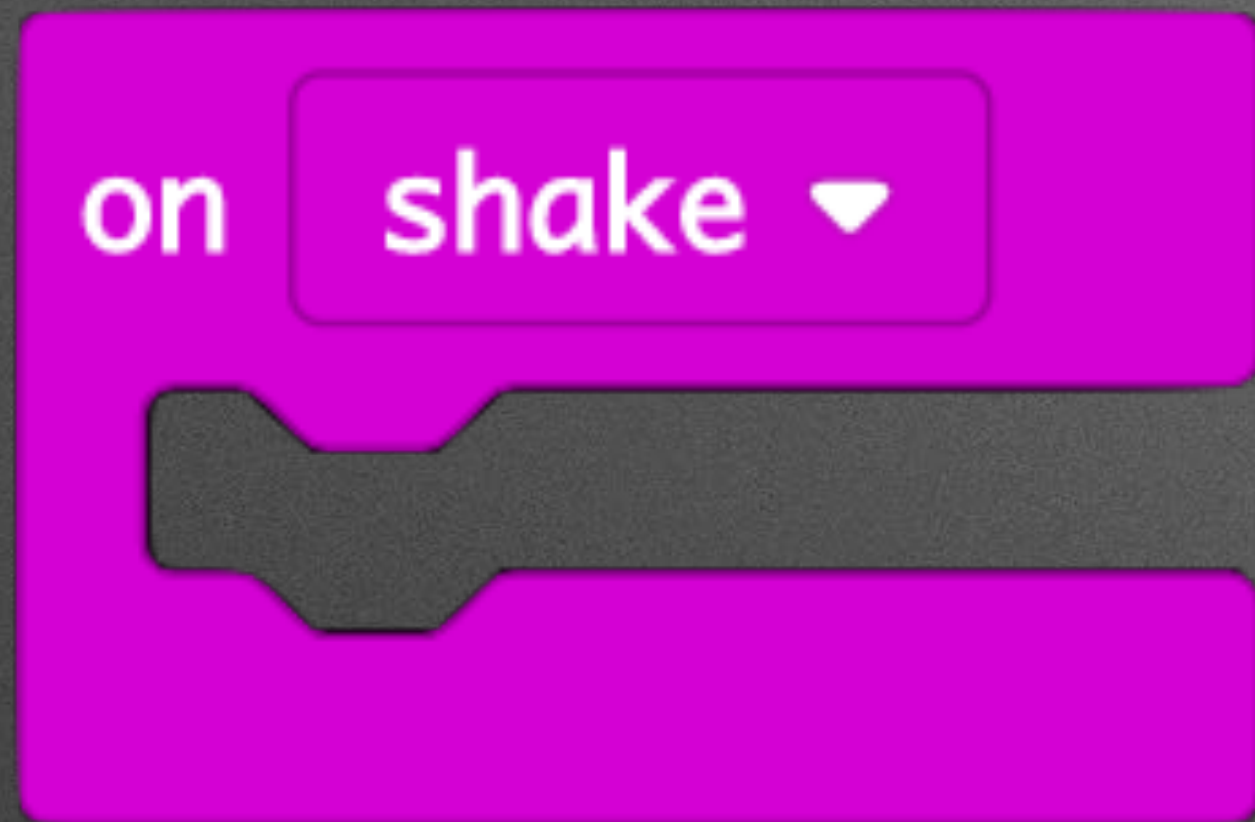


# Obvious Solution...





# Parts



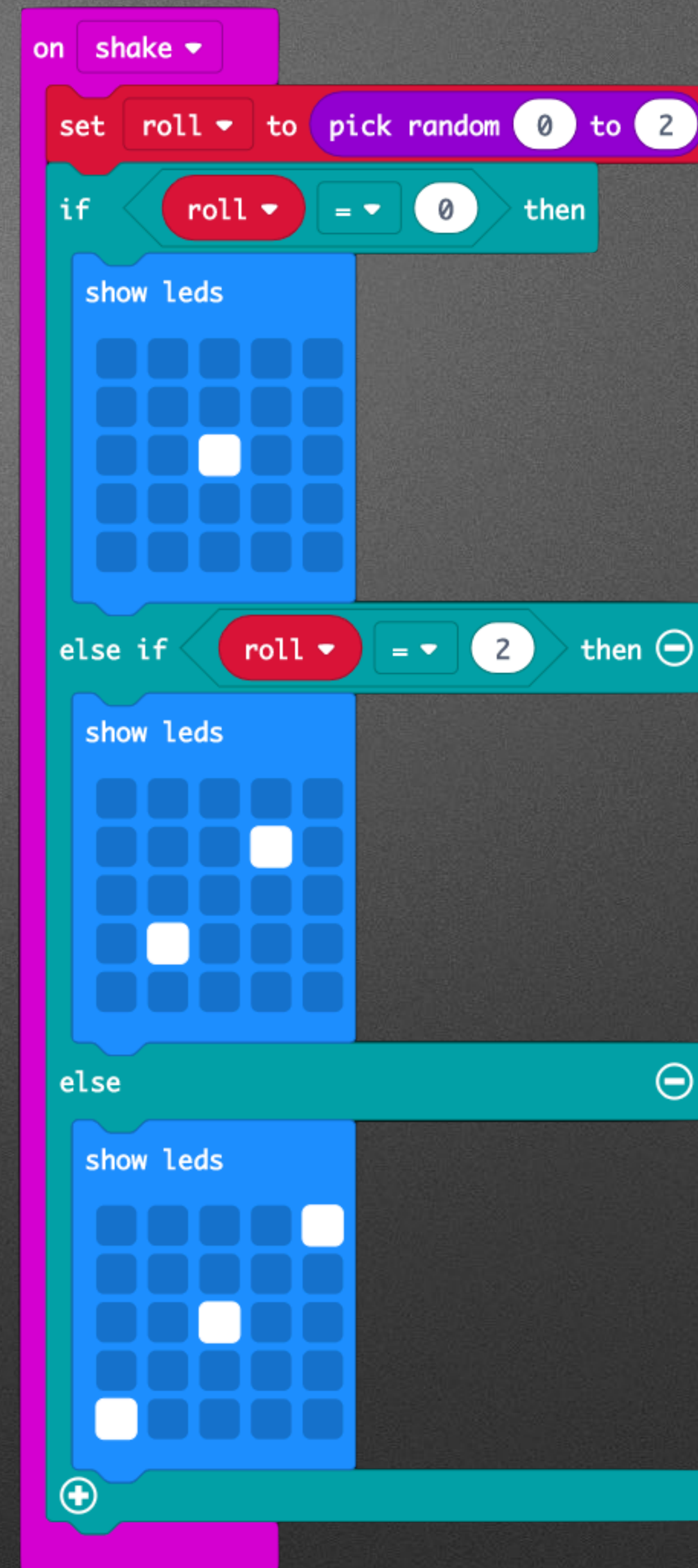
1. Color indicates Palette
2. Incremental Development:  
Try parts in Simulator
3. Play...Start with showing 0/1



**Let's play...**



# A solution





# Concepts



# Concepts

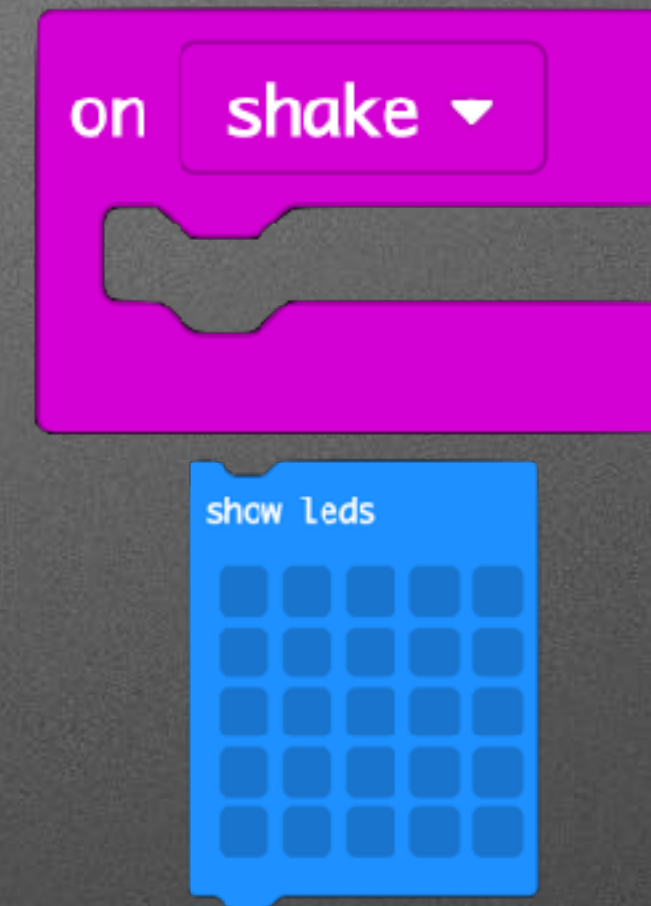
- Event driven programming





# Concepts

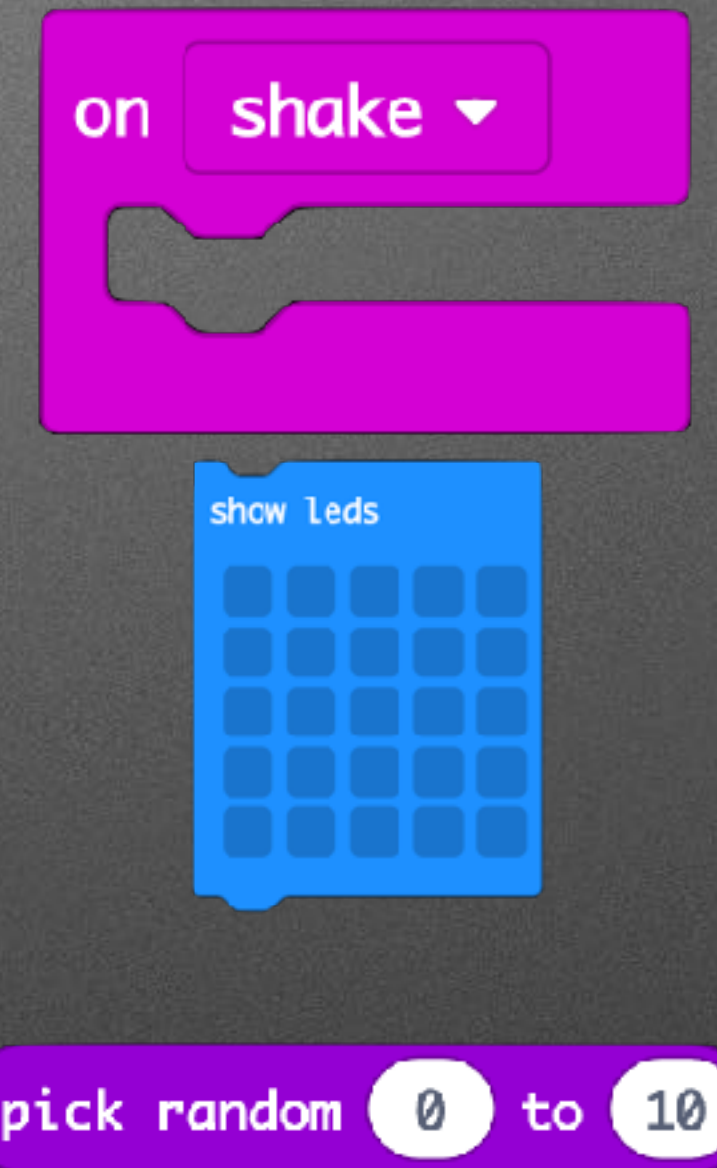
- Event driven programming
- Bitmapped Graphics





# Concepts

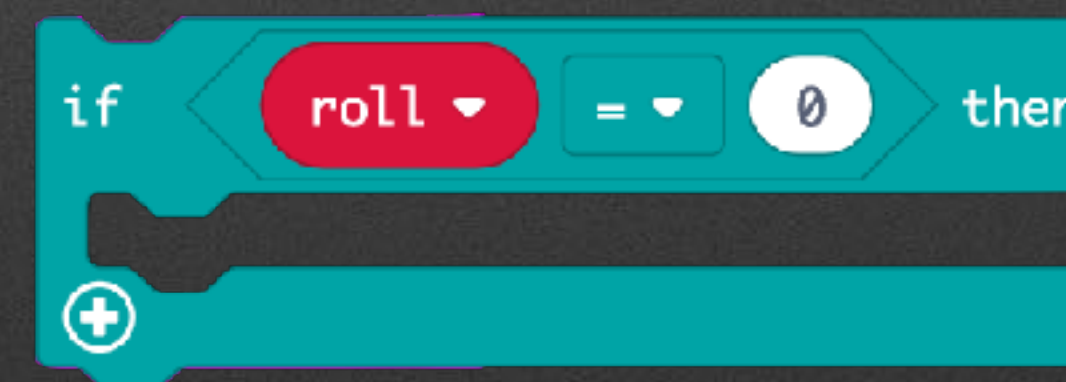
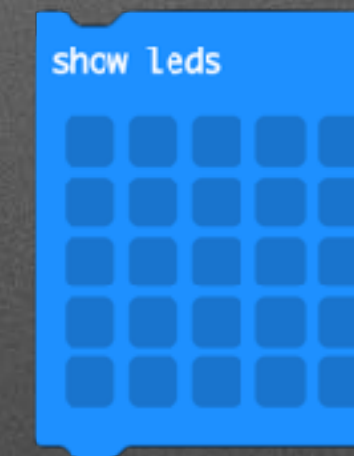
- Event driven programming
- Bitmapped Graphics
- Ranges & Representations





# Concepts

- Event driven programming
- Bitmapped Graphics
- Ranges & Representations
- Boolean Logic





# Pedagogy

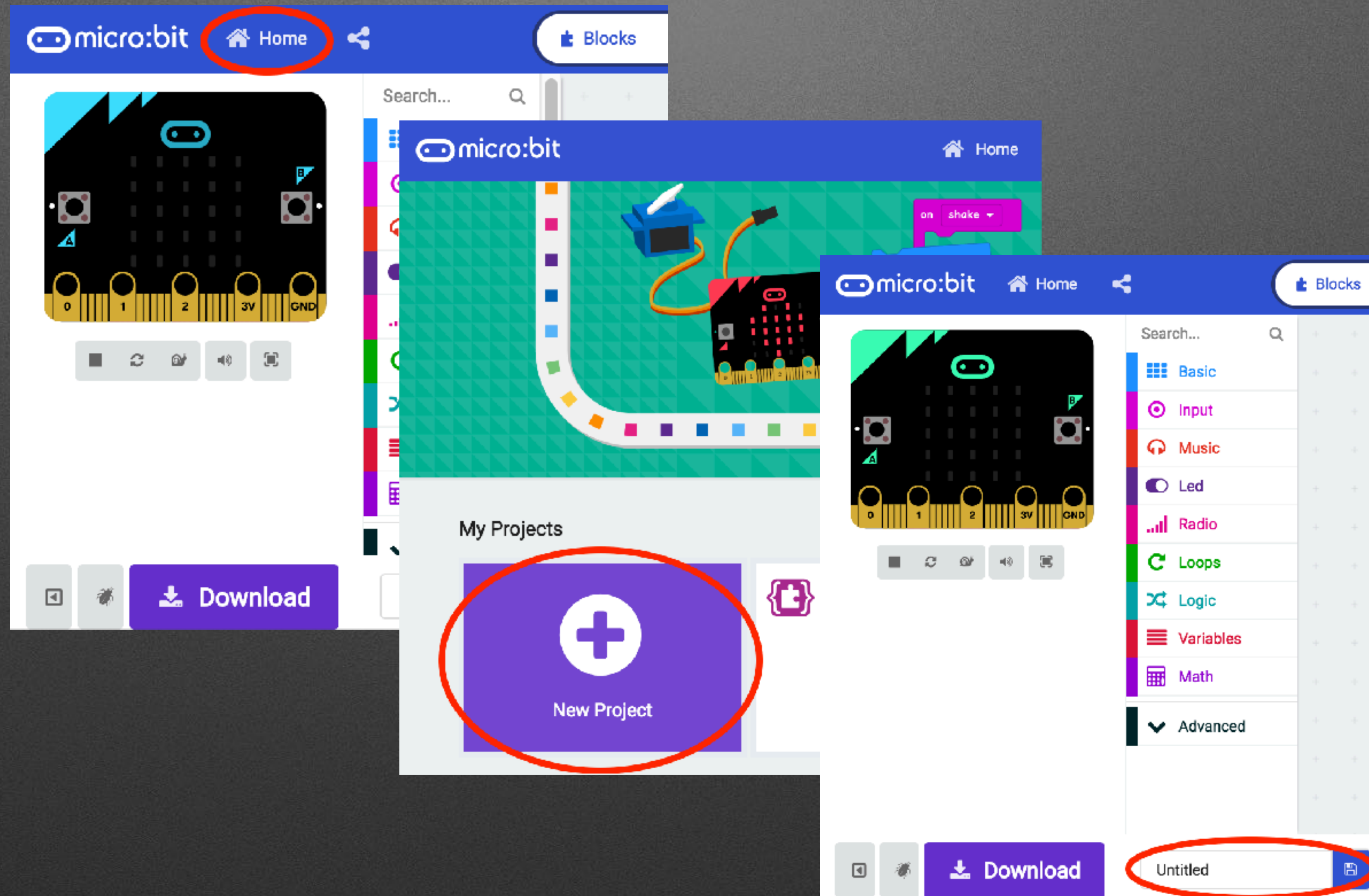
- Active Learning
- Discovery Based
- Constructionist



**New Project: Home > New Project...**



# New Project: Home > New Project...







# Broadcast Basics

- Radio Palette: Broadcast Based Radio Transmissions
  - String, Number, Key/Value Pairs, ...



# Receiver

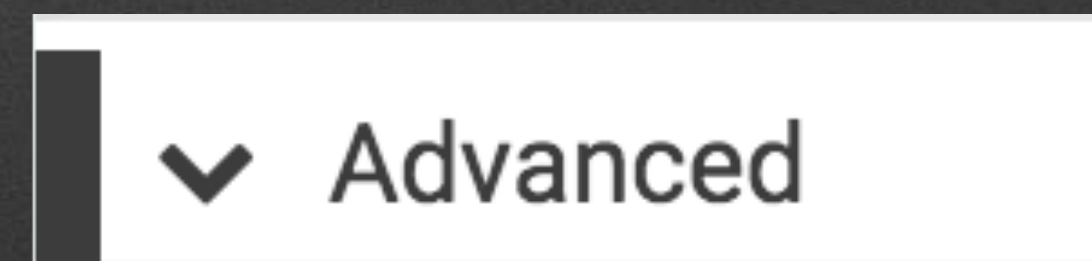




# Receiver



“Game” blocks in “Advanced”  
Section of Palette





# Full Boadcaster

on radio received

change score by

on button  pressed

radio send number

on start

radio set group

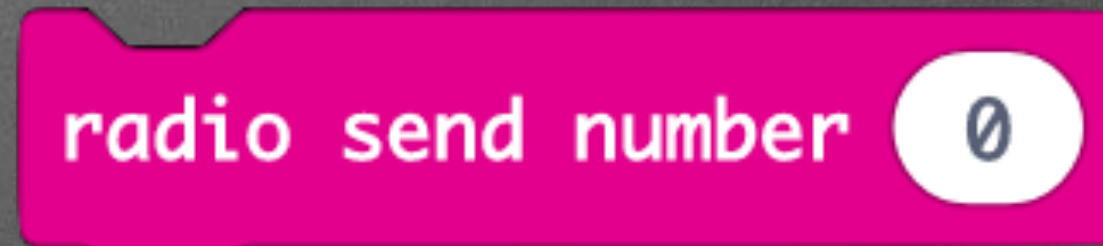


# Concepts



# Concepts

- Broadcasting





# Concepts

- Broadcasting
- Network Addresses

radio send number 0

radio set group 1



# Concepts

- Broadcasting
- Network Addresses
- Asynchronous clocks / Sync problems

radio send number 0

radio set group 1

on radio received receivedNumber ▼



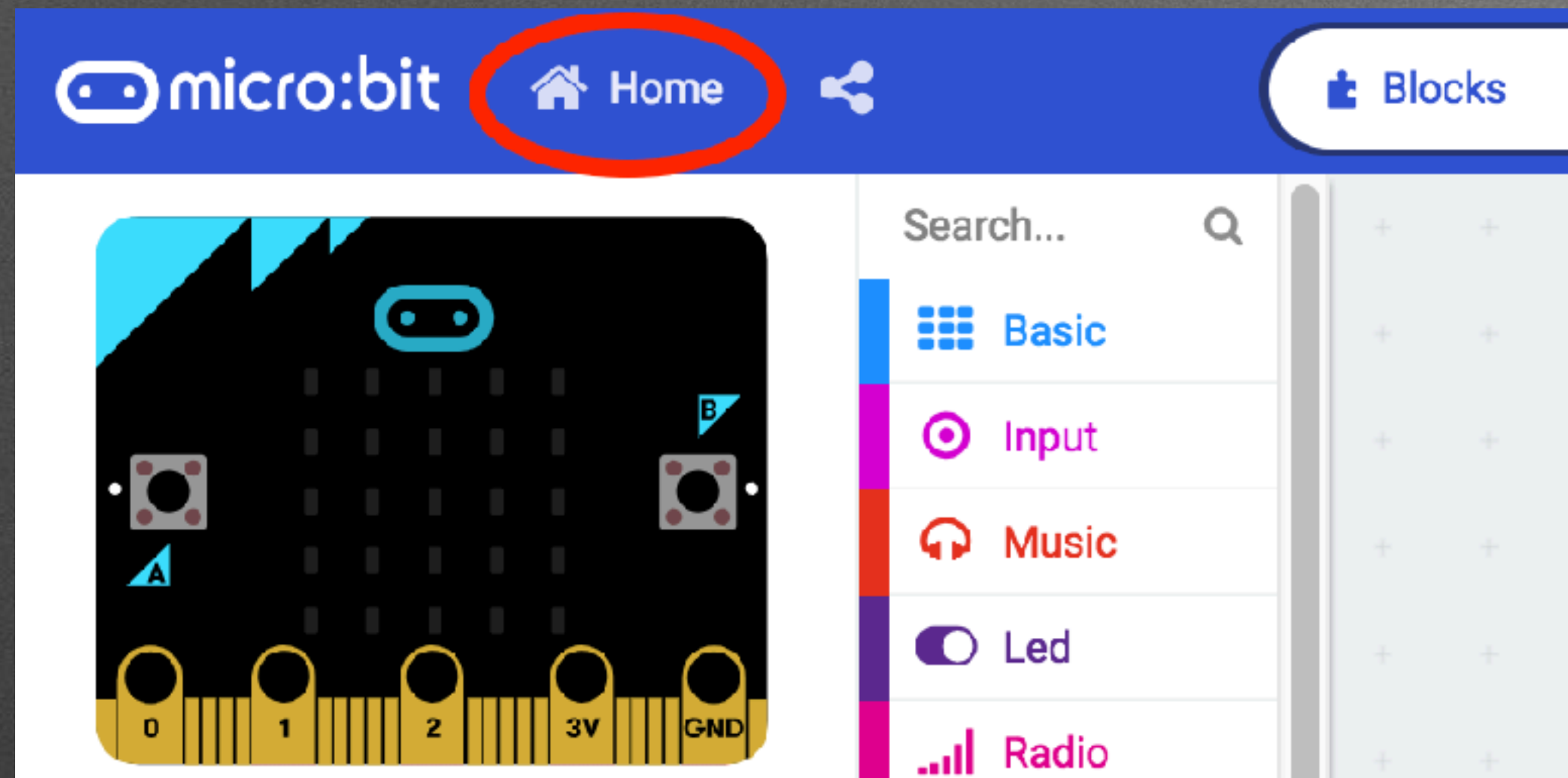


**& Firefly Fun**





# & Firefly Fun



## Radio Games



Multi Dice



Mood Radio



Tele-potato



Fireflies

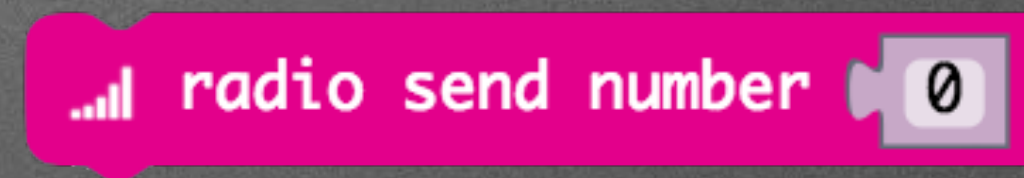


# Concepts



# Concepts

- Broadcasting





# Concepts

- Broadcasting
- Network Addresses

radio send number 0

radio set group 1



# Concepts

- Broadcasting
- Network Addresses
- Asynchronous clocks / Sync problems

radio send number 0

radio set group 1

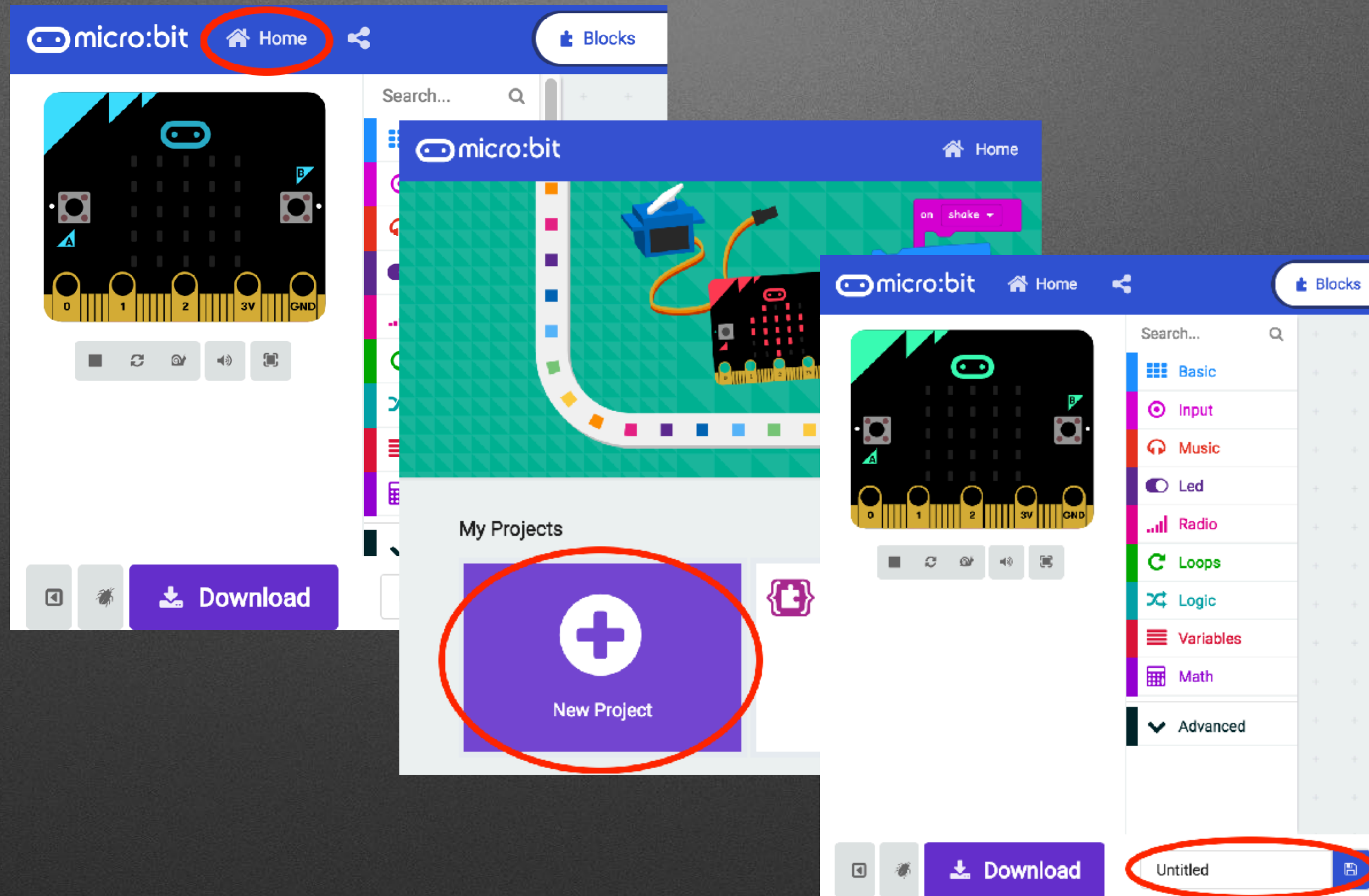
on radio received receivedNumber



**New Project: Home > New Project...**



# New Project: Home > New Project...





# Goody Bag: Hardware





**Awesome (?) Audio**



# Parts: 2 Clips + Headphone





# Concepts



# Concepts

- I/O



# Concepts

- I/O
- Basic Electric Circuits/Electronics





# Motor Mayhem

An Intro to Servos





# Motor Mayhem

An Intro to Servos

on button  pressed

servo write pin  to

on button  pressed

servo write pin  to





# Motor Mayhem

An Intro to Servos

on button  pressed

servo write pin  to

on button  pressed

servo write pin  to

These servos are  
limited to 0-120°

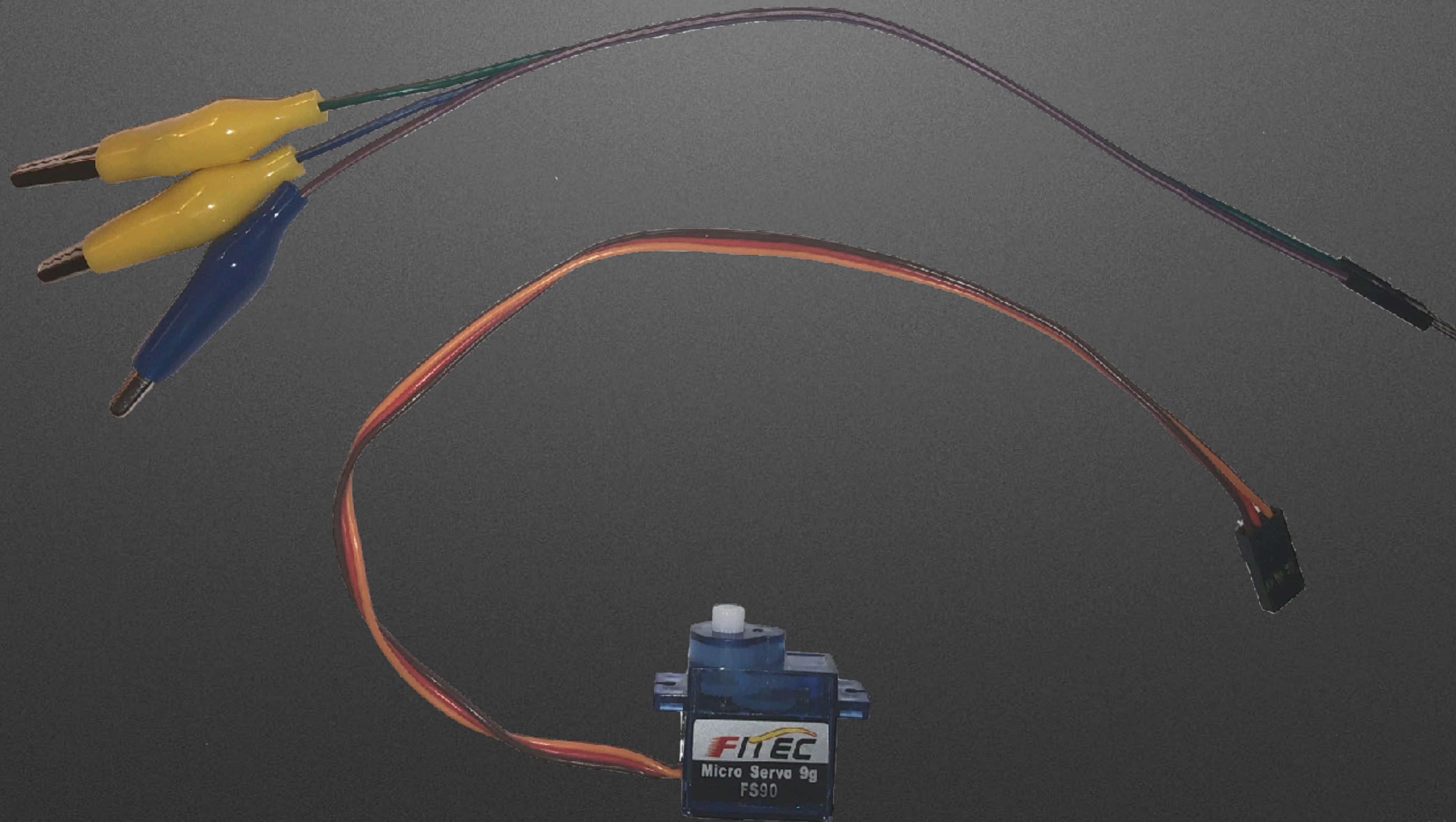


# Testing...

- Test in Simulator

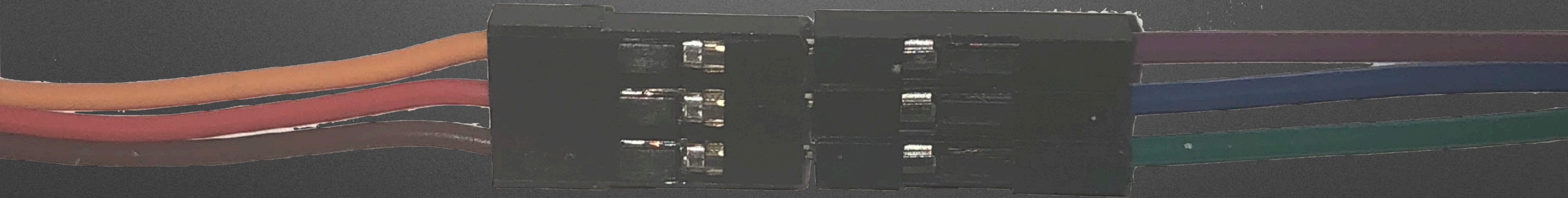


# Parts



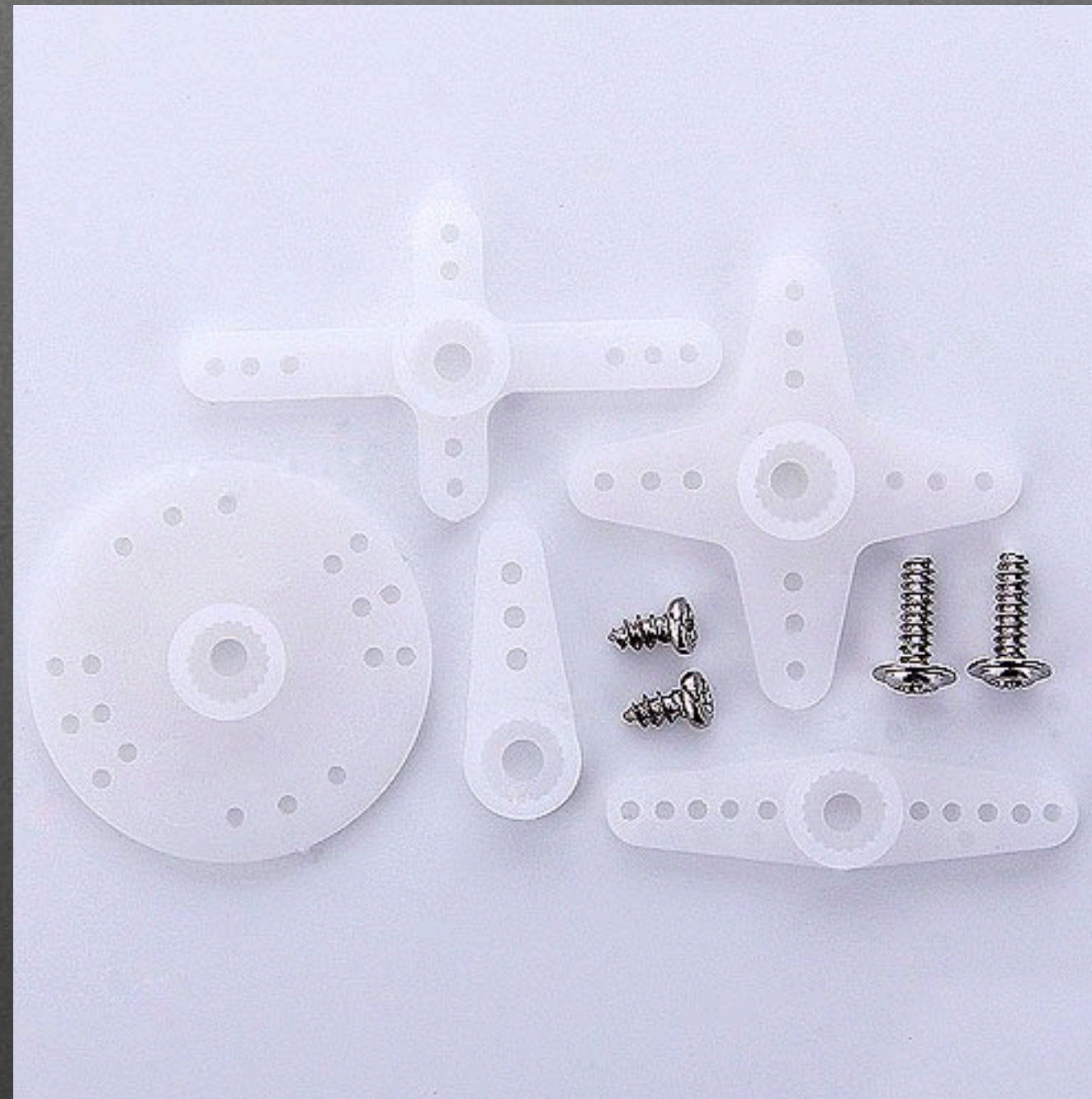


# Connect them...





# Add a Horn





# Clip to micro:bit

- Match color on Servo to pad name on micro:bit (clip colors don't matter)
  - Brown on Servo to GND on micro:bit
  - Red on Servo to 3V on micro:bit
  - Orange on Servo to 0 on micro:bit



# Inchworm Insanity

<https://makecode.microbit.org/projects/inchworm>



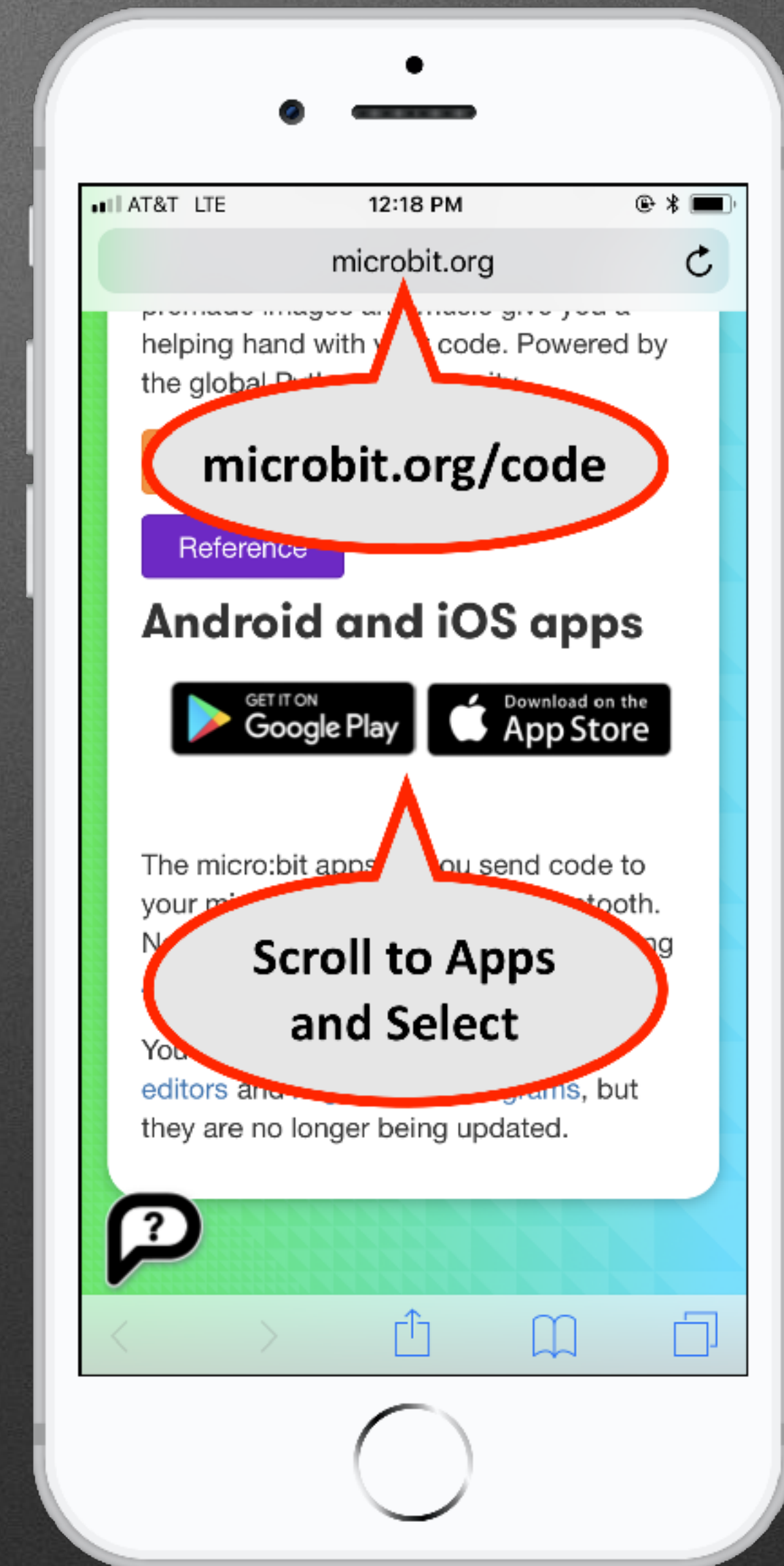
# Break

## 1. Firmware Update

- A. Go to <https://tinyurl.com/uBitUpdate>
- B. Follow Instructions to Upgrade

## 2. App Install

- A. Open Browser on phone to <http://microbit.org/code>
- B. Scroll to Apps and Select





# Bluetooth Background

- Uses different protocol than  Radio
- Not a group broadcast



# Bluetooth Background



# Bluetooth Background

Central



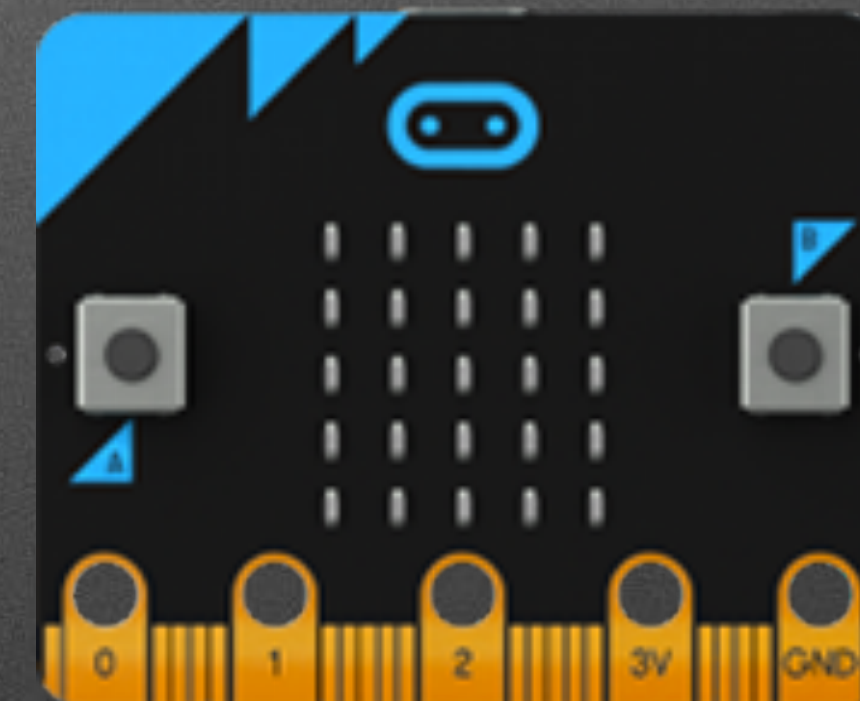


# Bluetooth Background

Central



Peripheral





# Bluetooth Background



# Bluetooth Background

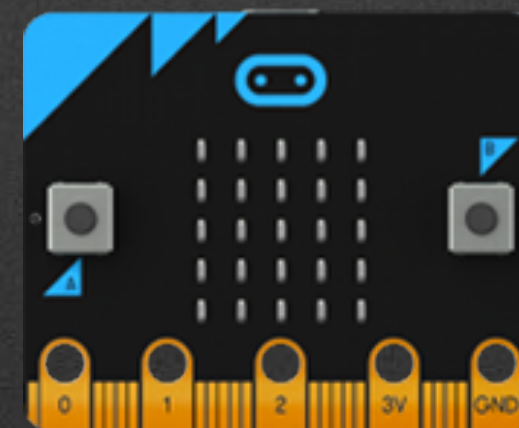
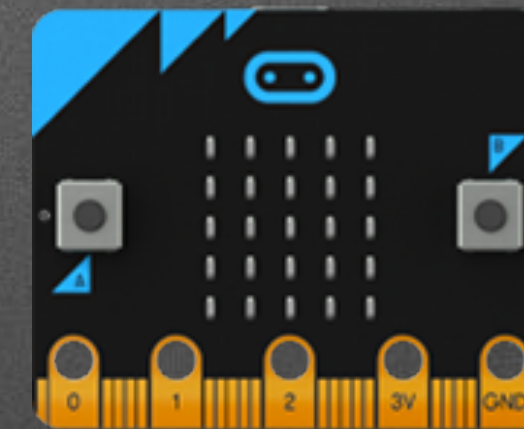
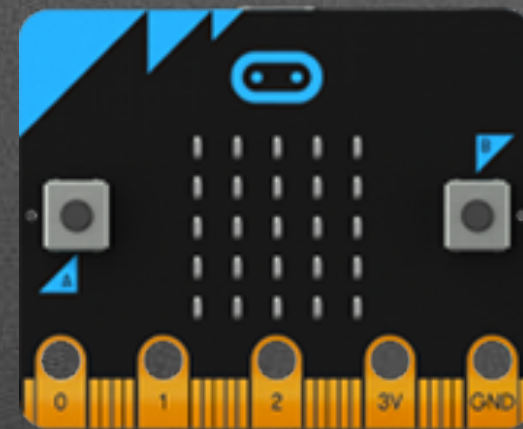
Central





# Bluetooth Background

Central





# Bluetooth Basics

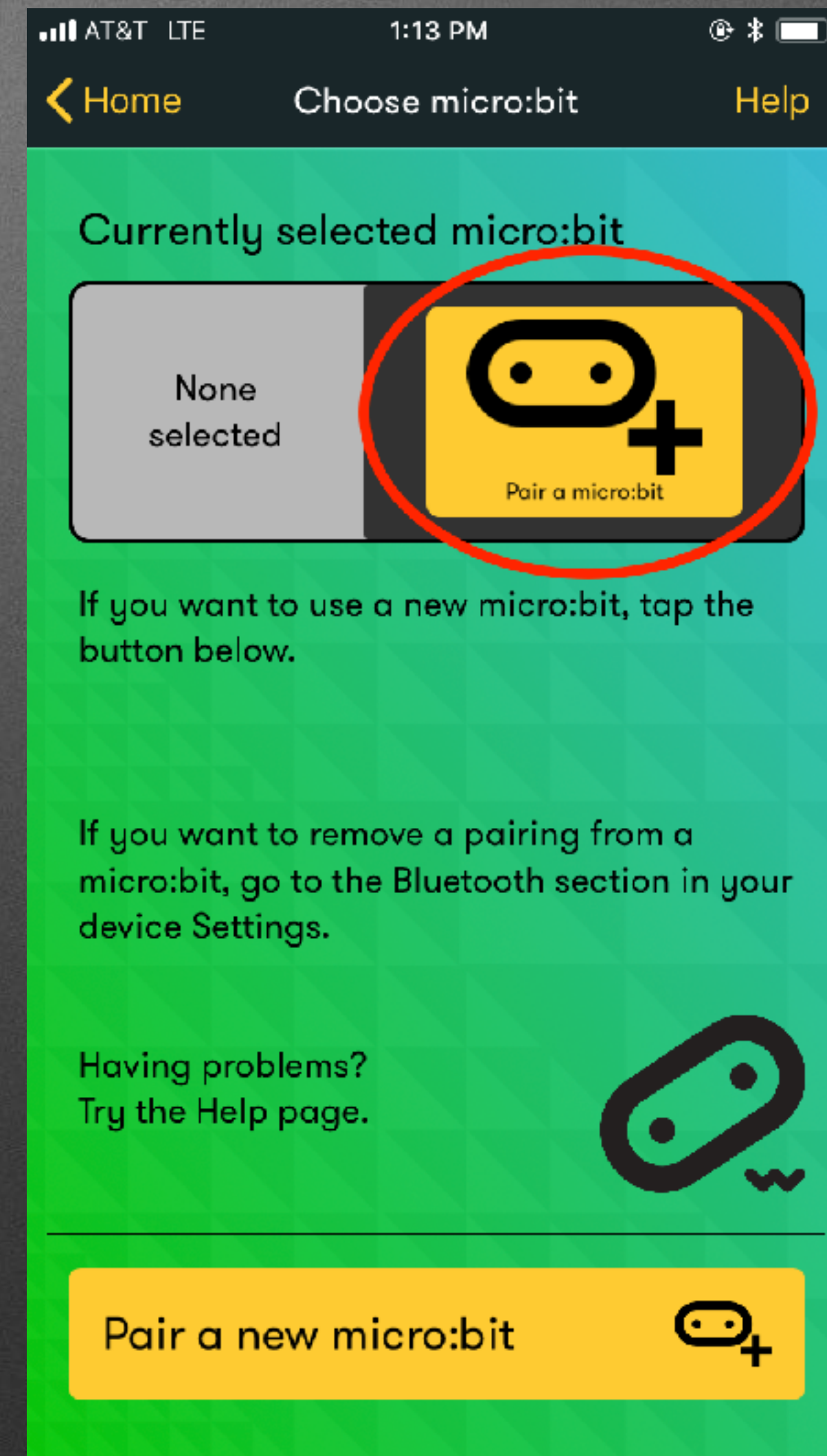
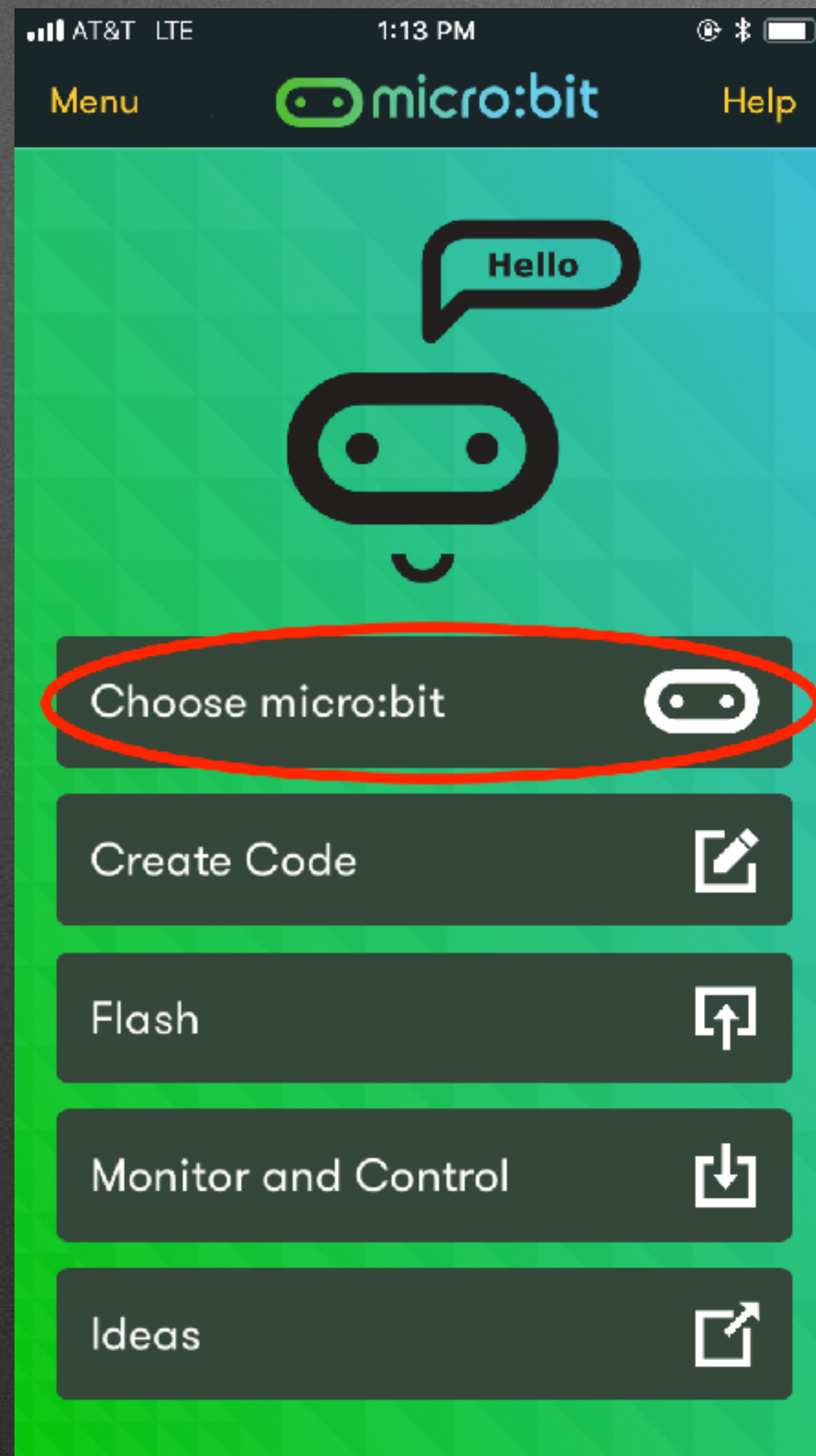
- Bluetooth has various levels of security
  - “Pairing” — Forming a “permanent” bond  
(Exchanging security info. once and storing it)
- Block editor supports three types
  - No pairing (“insecure” - we’ll use this)
  - Just Works (default; pretty safe)
  - Passkey Pairing (more secure)



# Pairing



# Pairing

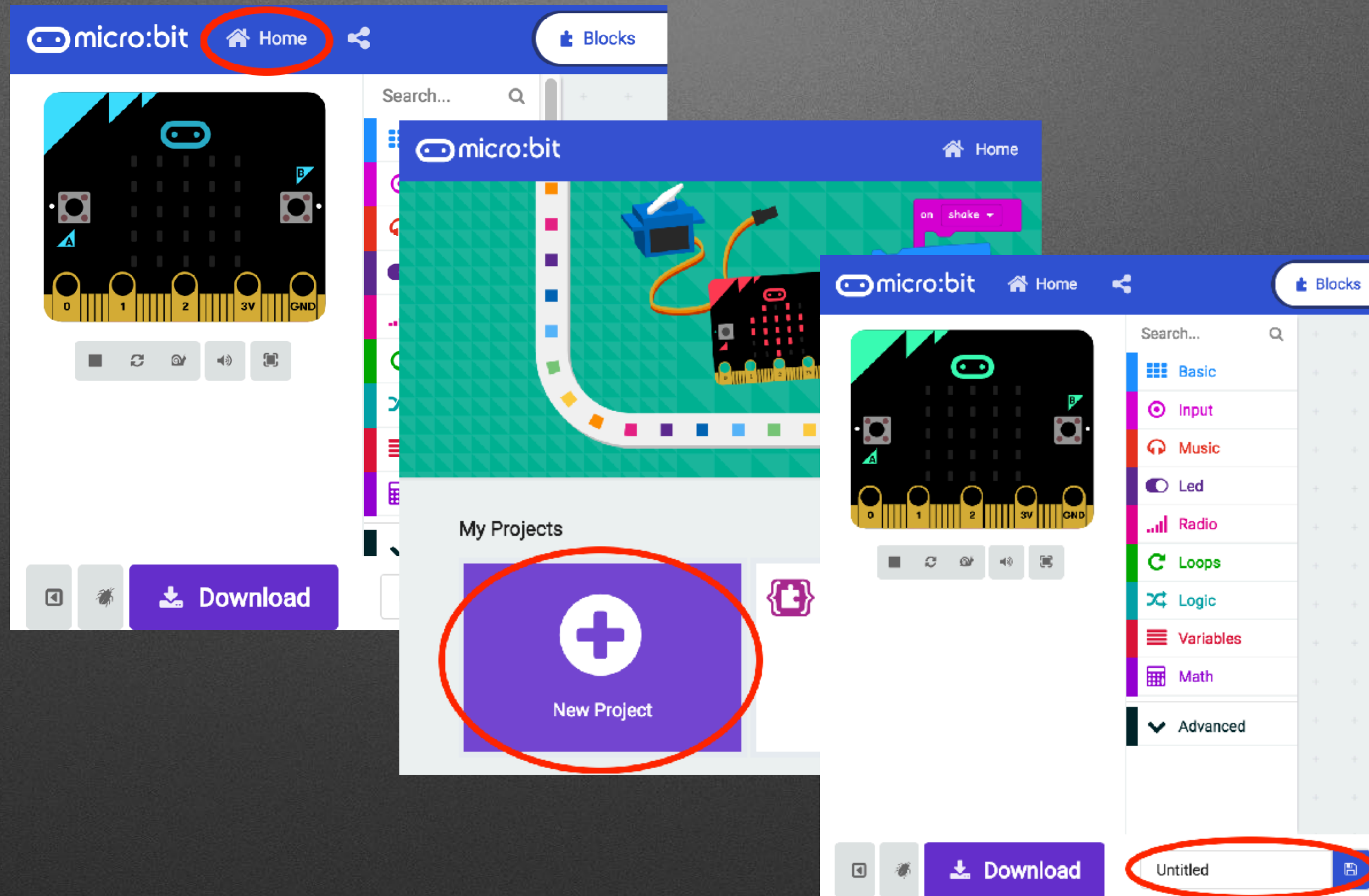




**New Project: Home > New Project...**



# New Project: Home > New Project...





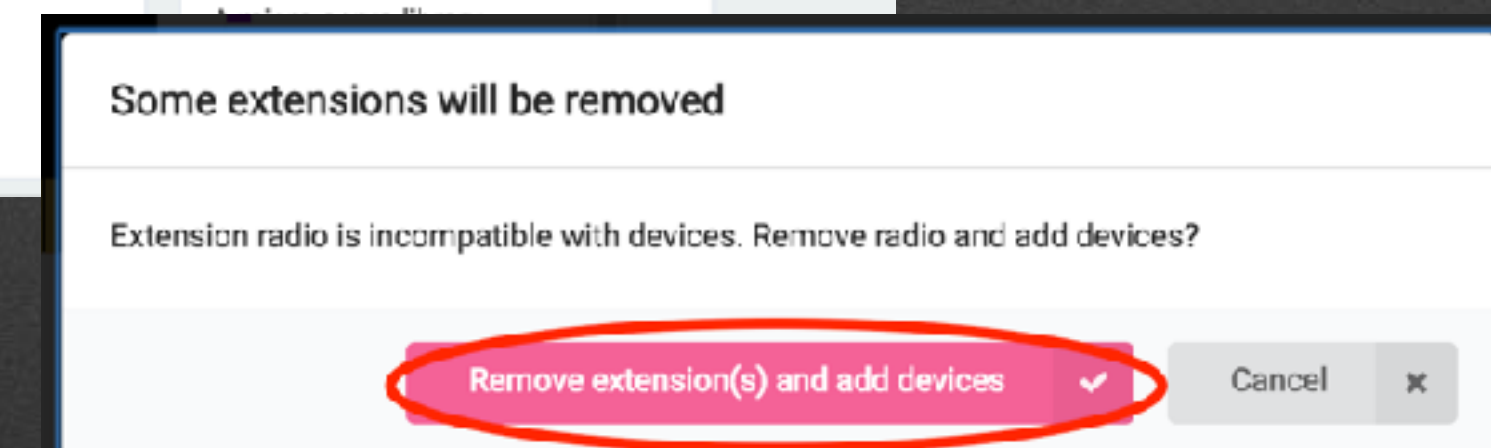
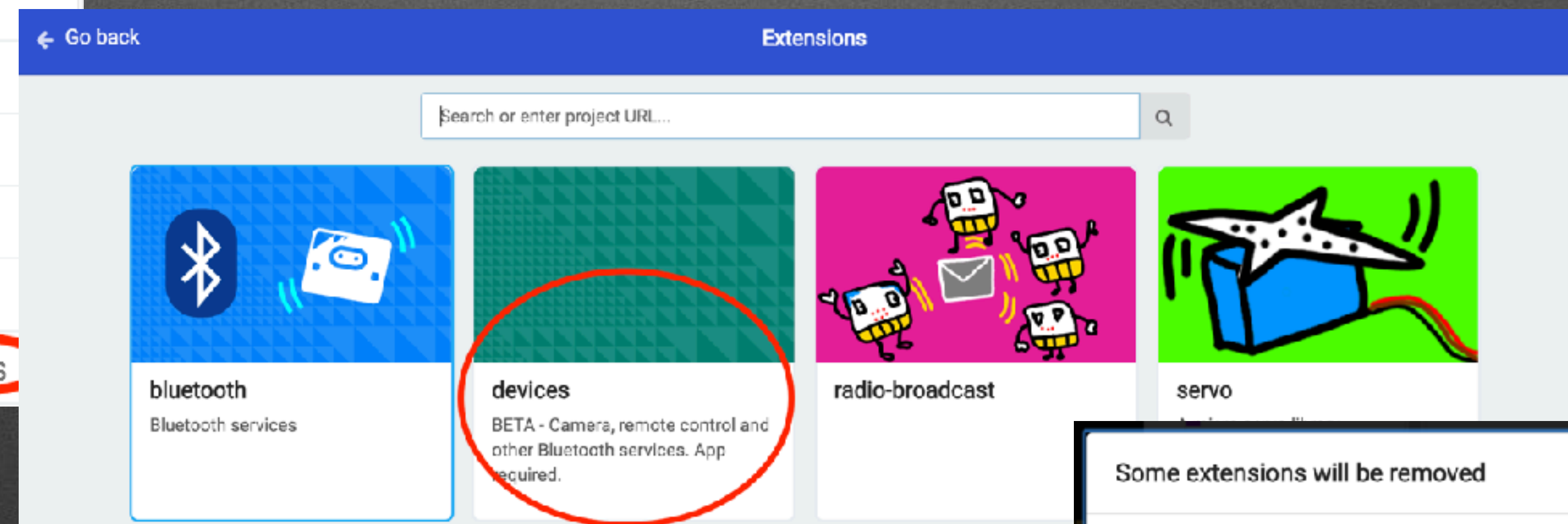
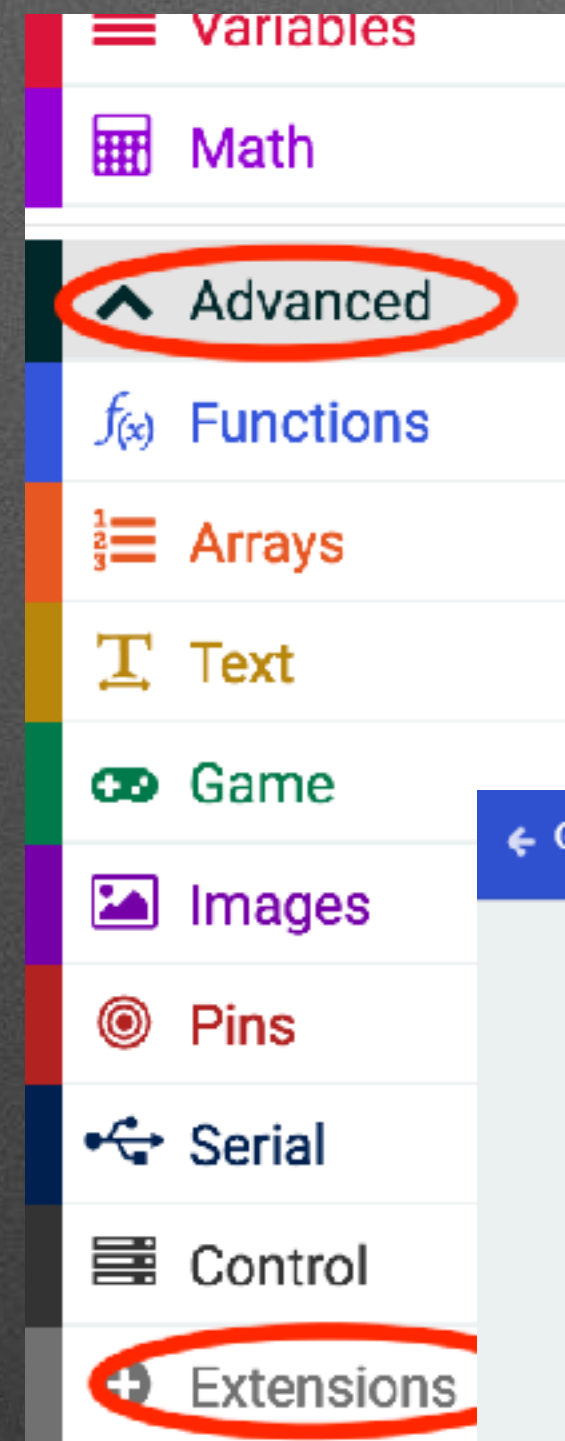
# Add Bluetooth

(& remove Radio)



# Add Bluetooth

(& remove Radio)

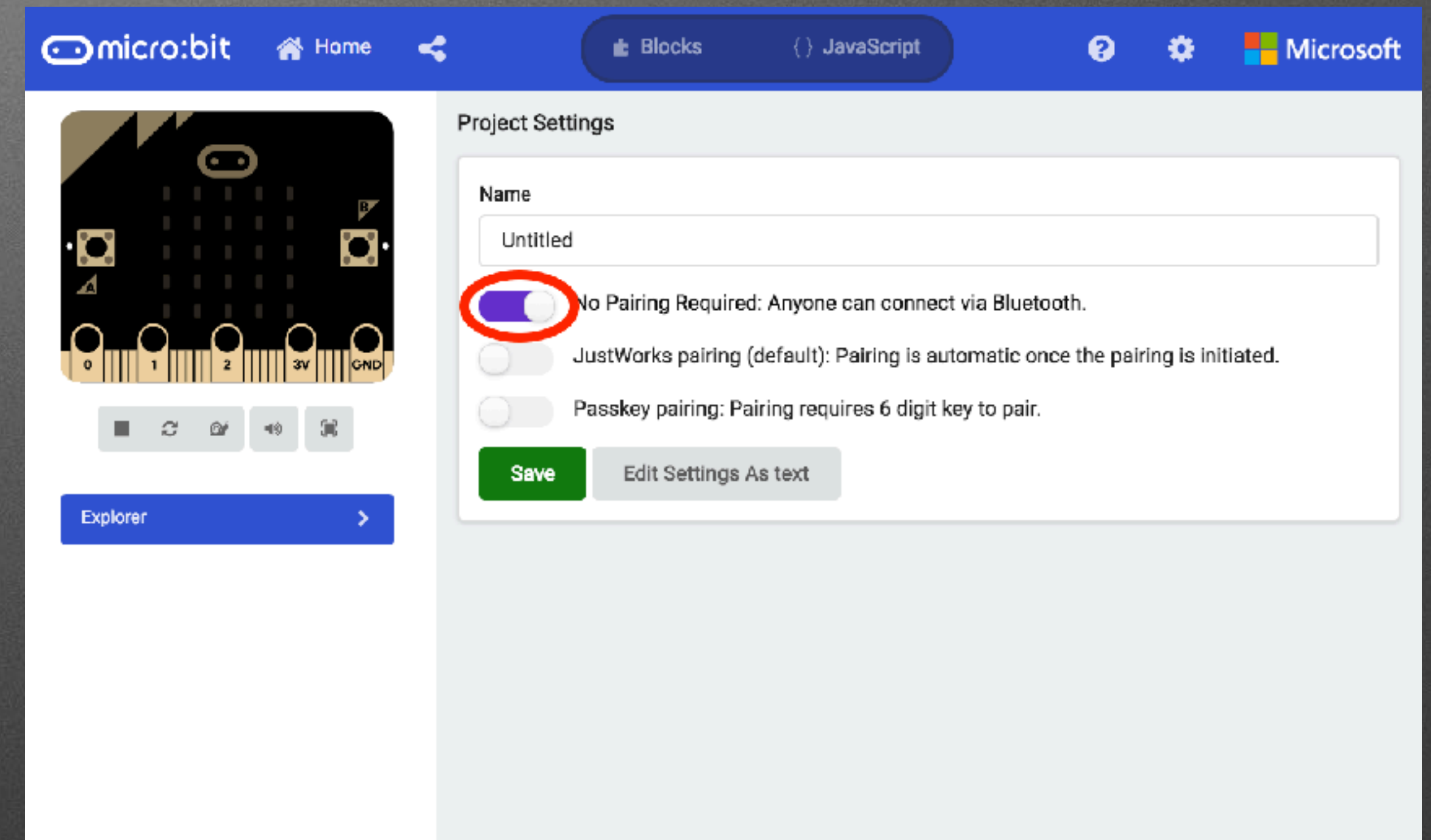
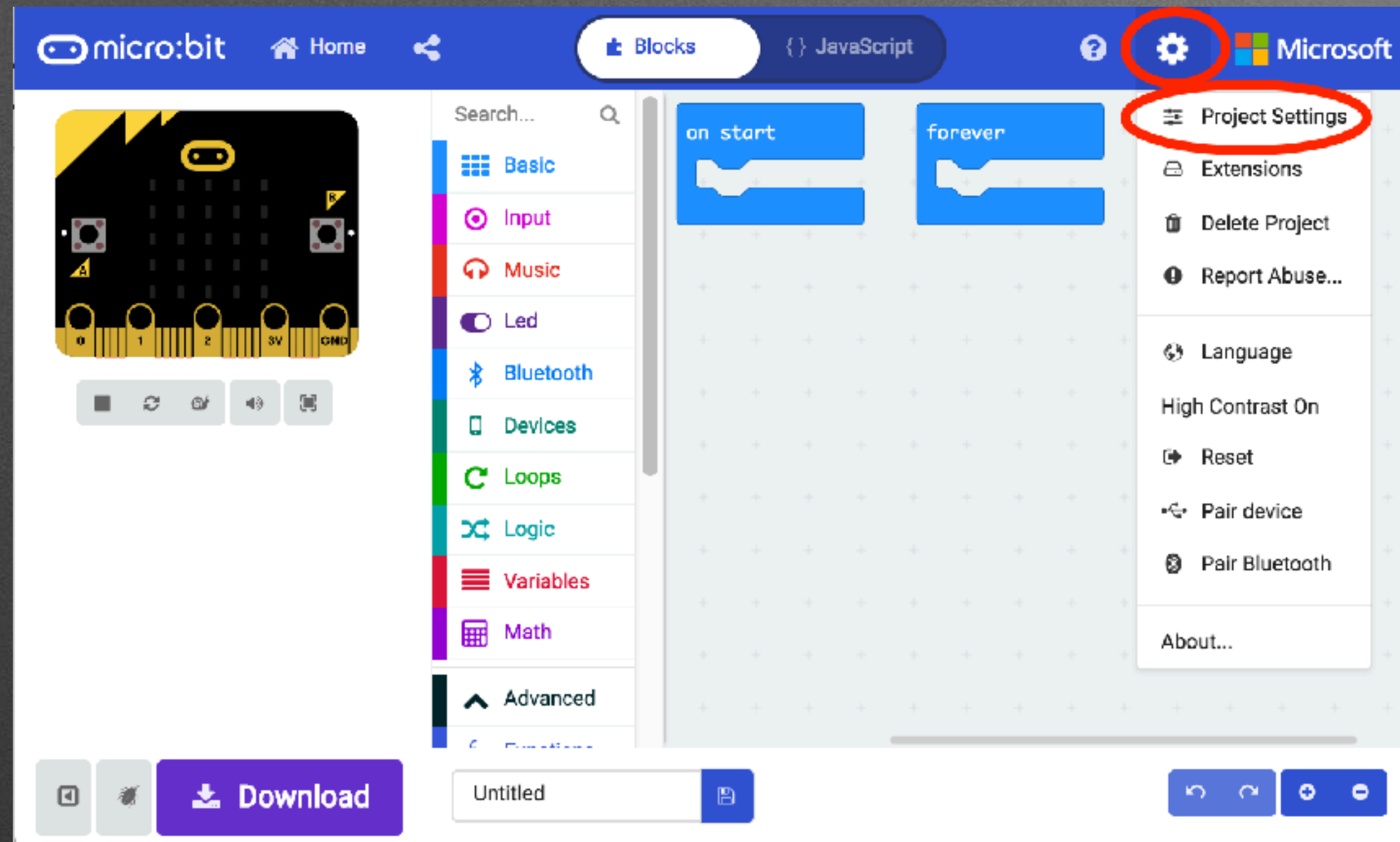




# Project Settings

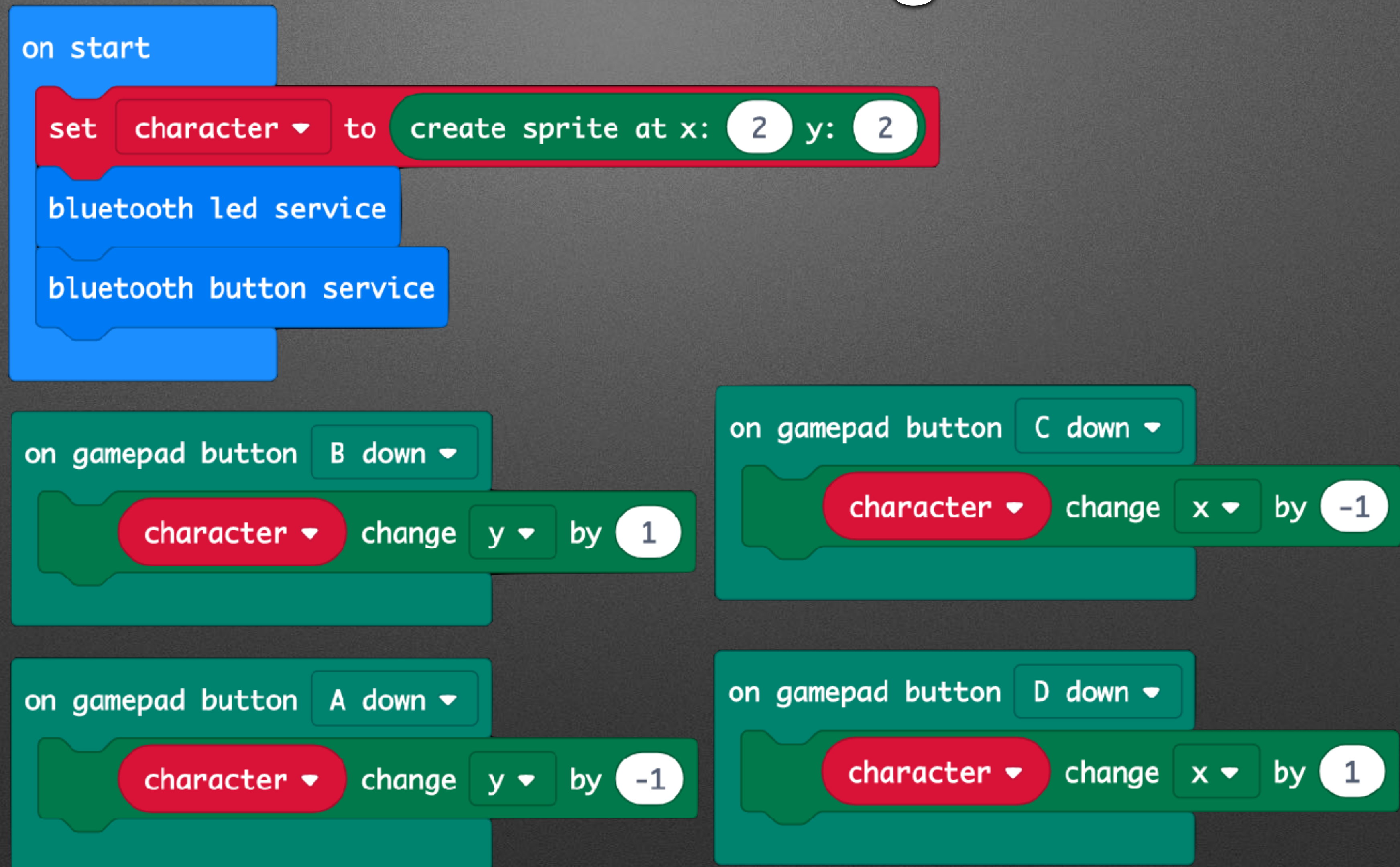


# Project Settings





# Phone Phun: Program





# Phone Phun: Program

Devices Palette

on start

set character ▼ to create sprite at x: 2 y: 2

bluetooth led service

bluetooth button service

on gamepad button B down ▼

character ▼ change y ▼ by 1

on gamepad button C down ▼

character ▼ change x ▼ by -1

on gamepad button A down ▼

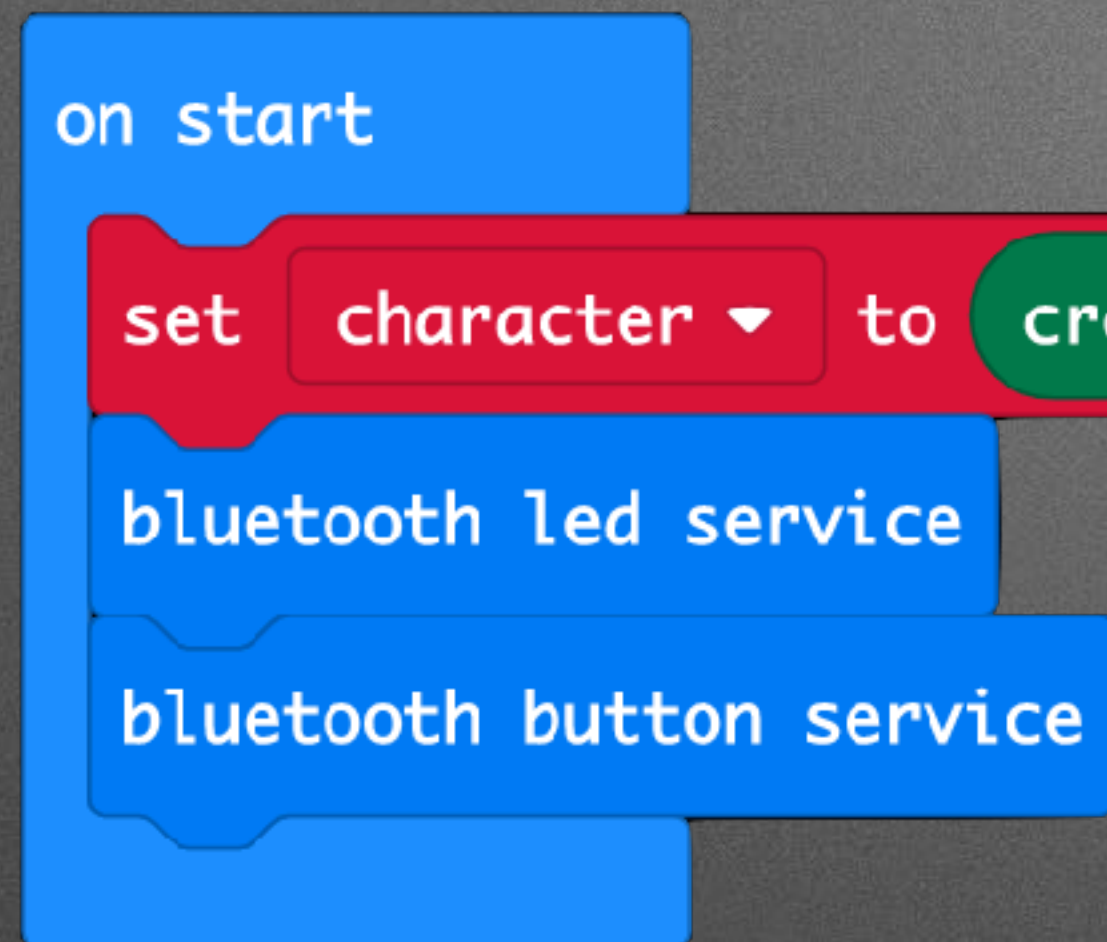
character ▼ change y ▼ by -1

on gamepad button D down ▼

character ▼ change x ▼ by 1

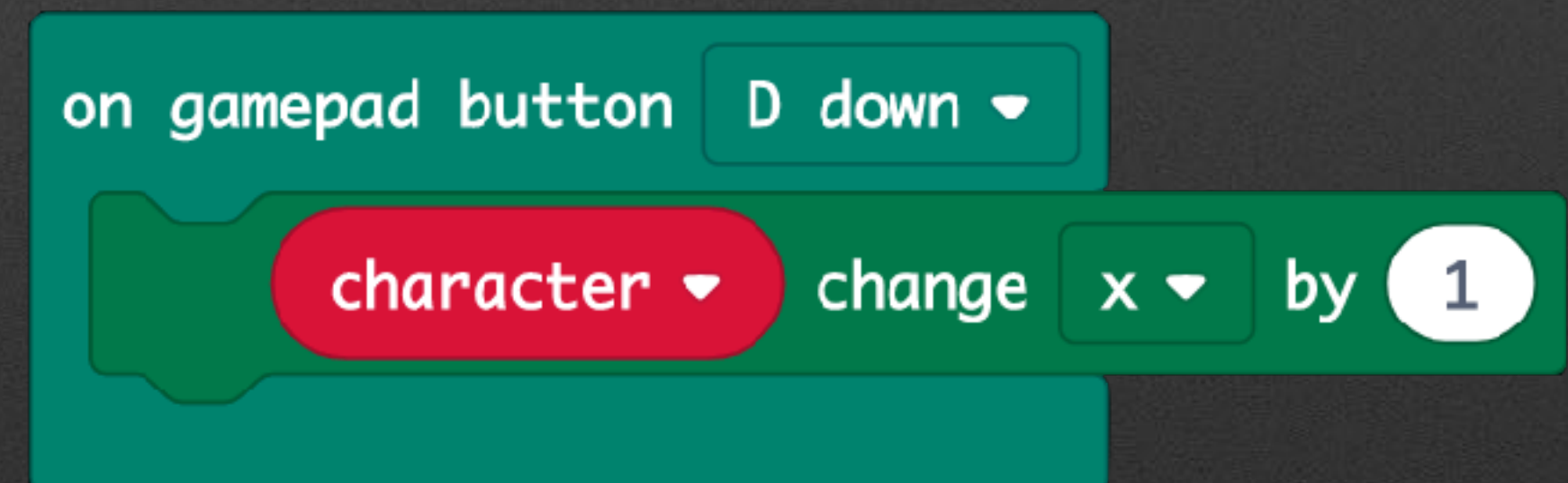
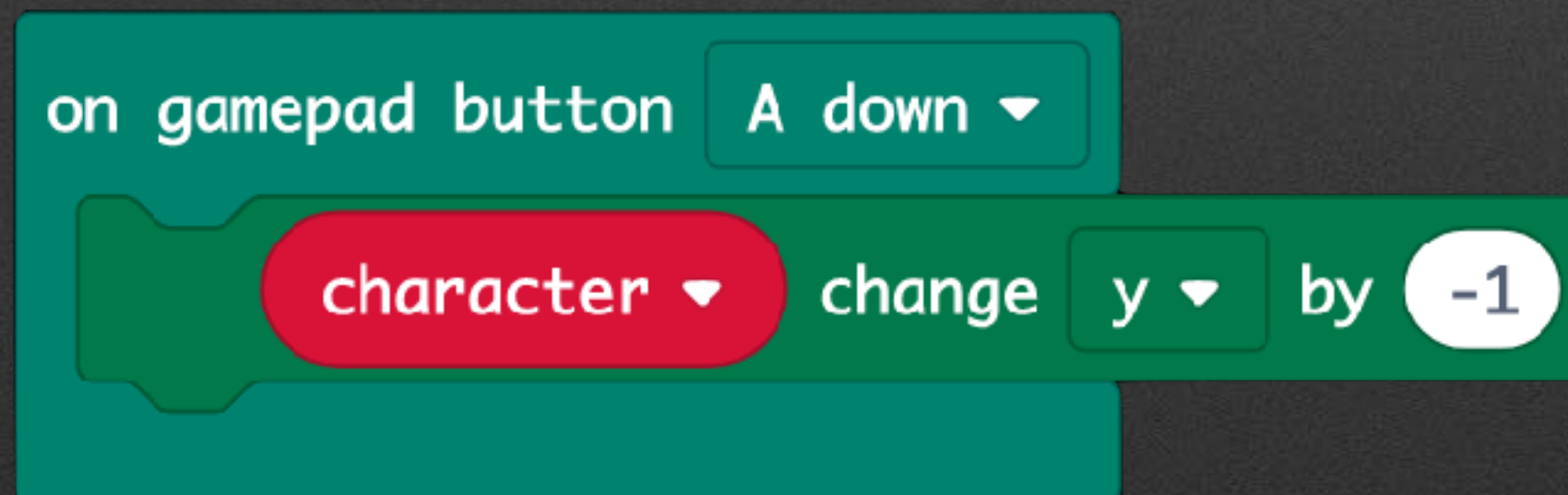
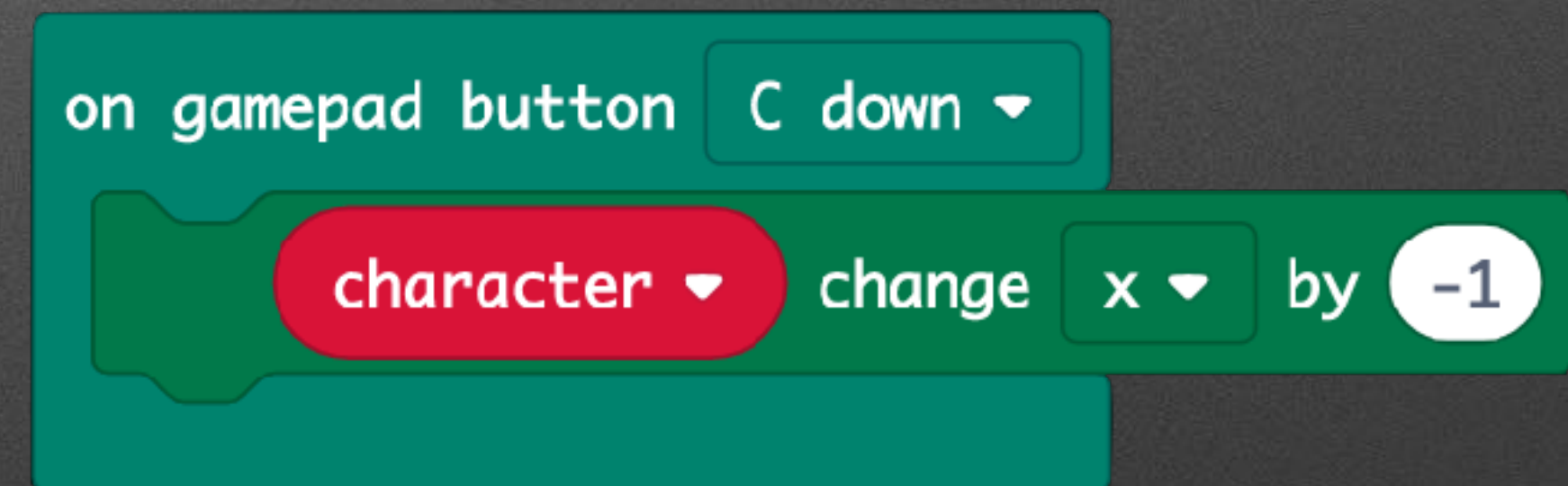
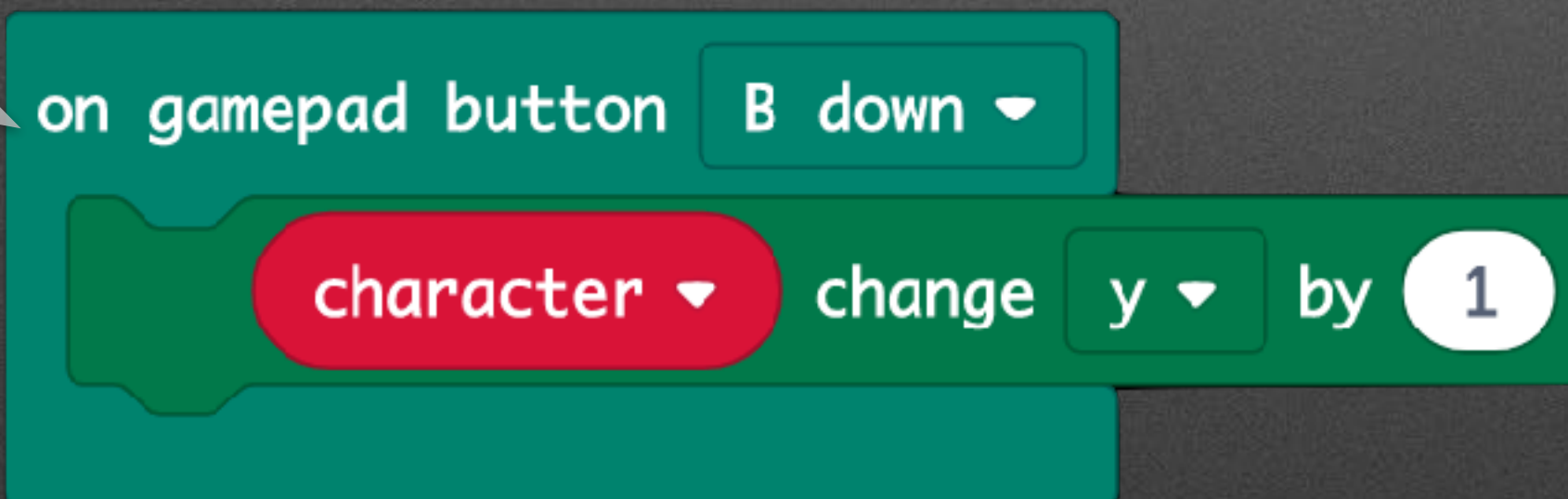


# Phone Phun: Program



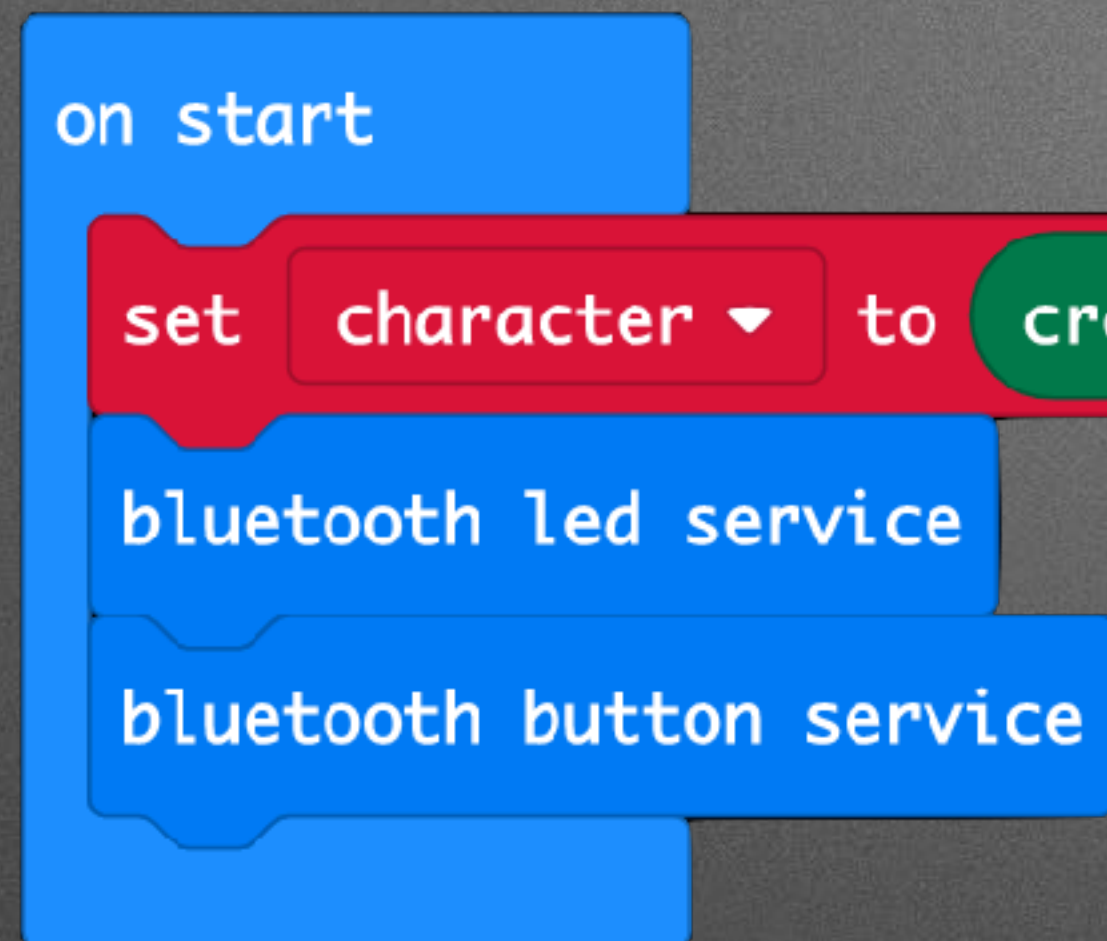
Devices Palette

Game Palette  
(Under Advanced)



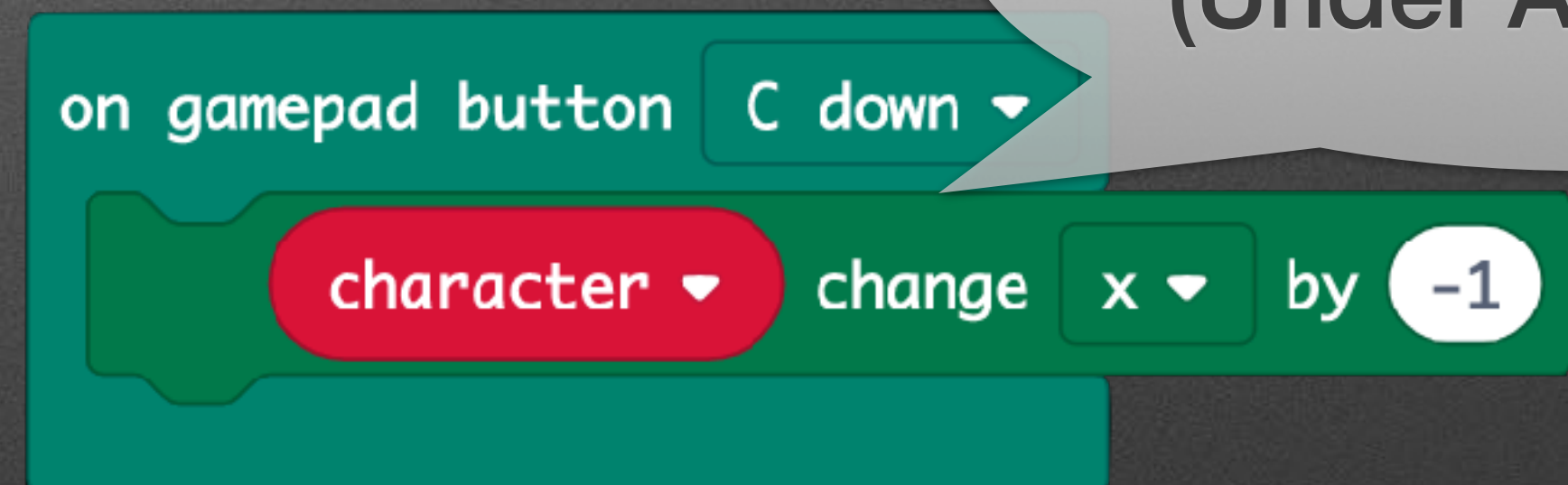
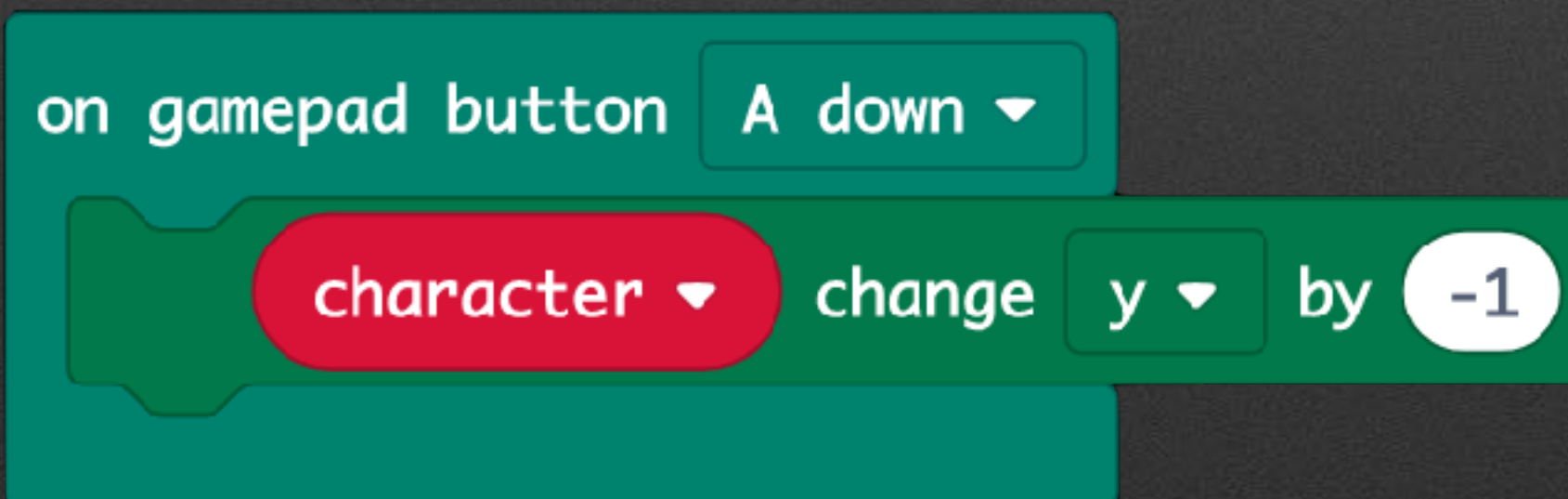
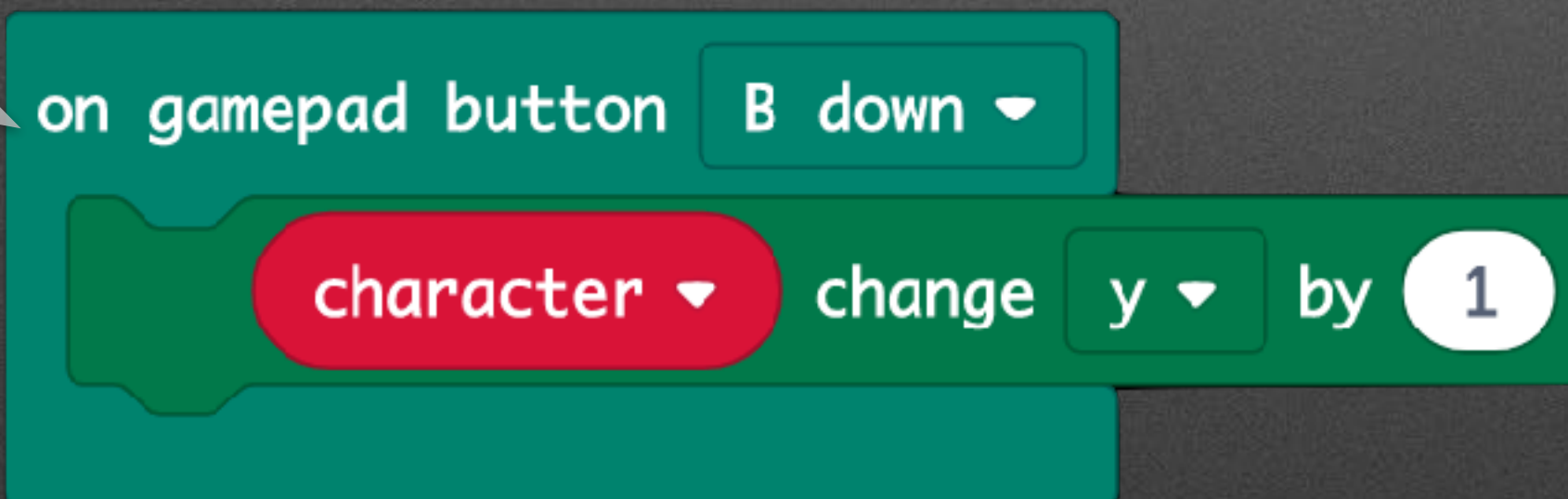


# Phone Phun: Program

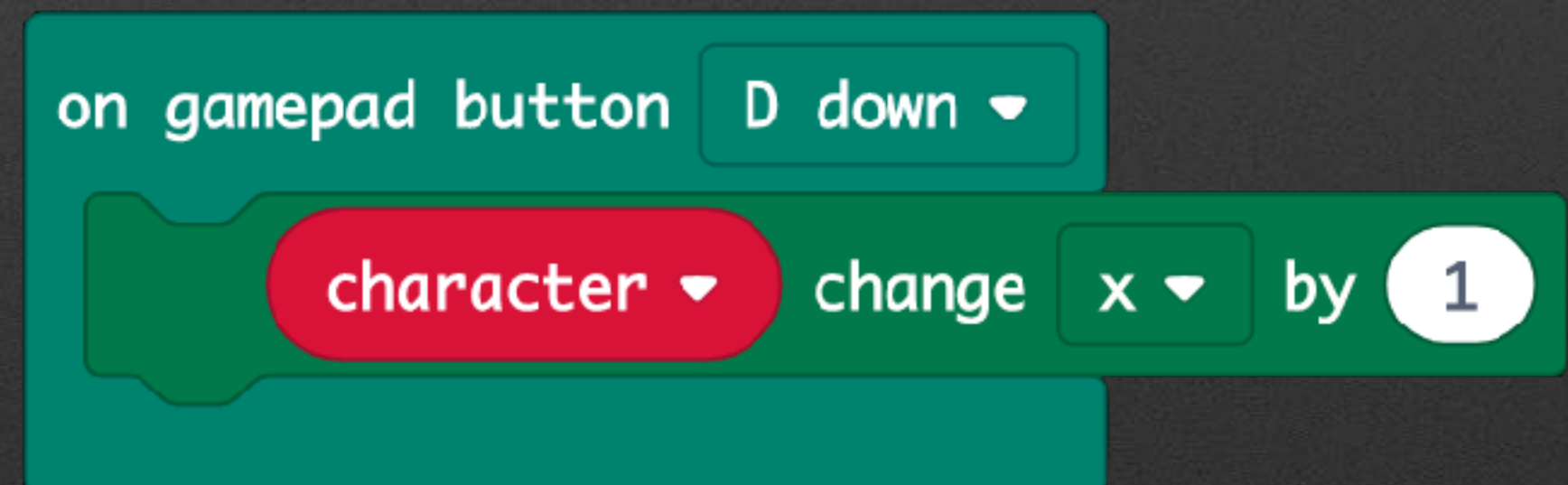


Devices Palette

Game Palette  
(Under Advanced)



Game Palette  
(Under Advanced)





# Phone Phun: Program

on start

set character ▼ to create sprite at x: 2 y: 2

bluetooth led service

bluetooth button service

Game Palette  
(Under Advanced)

Devices Palette

on gamepad button B down ▼

character ▼ change y ▼ by 1

on gamepad button C down ▼

character ▼ change x ▼ by -1

Game Palette  
(Under Advanced)

on gamepad button A down ▼

character ▼ change y ▼ by -1

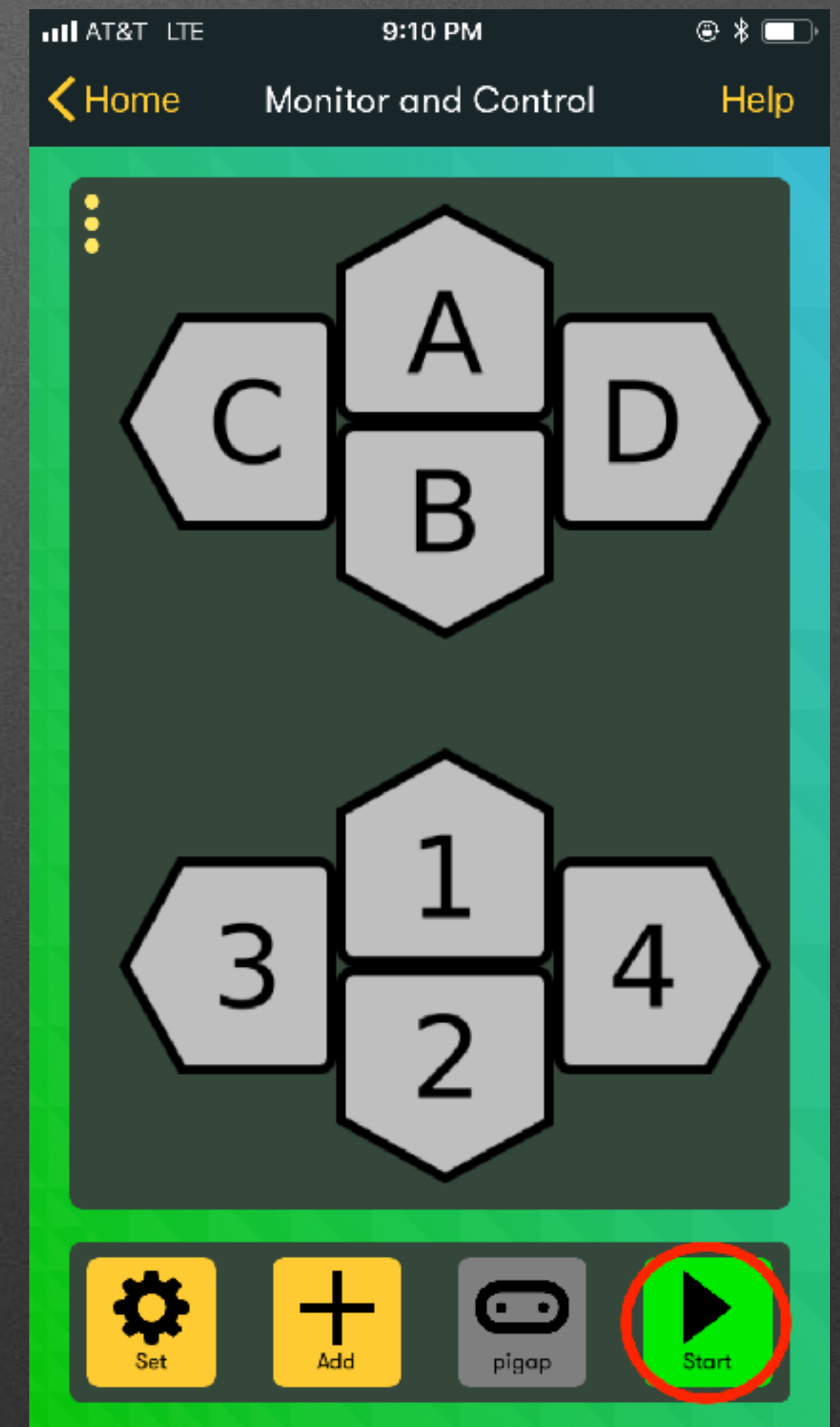
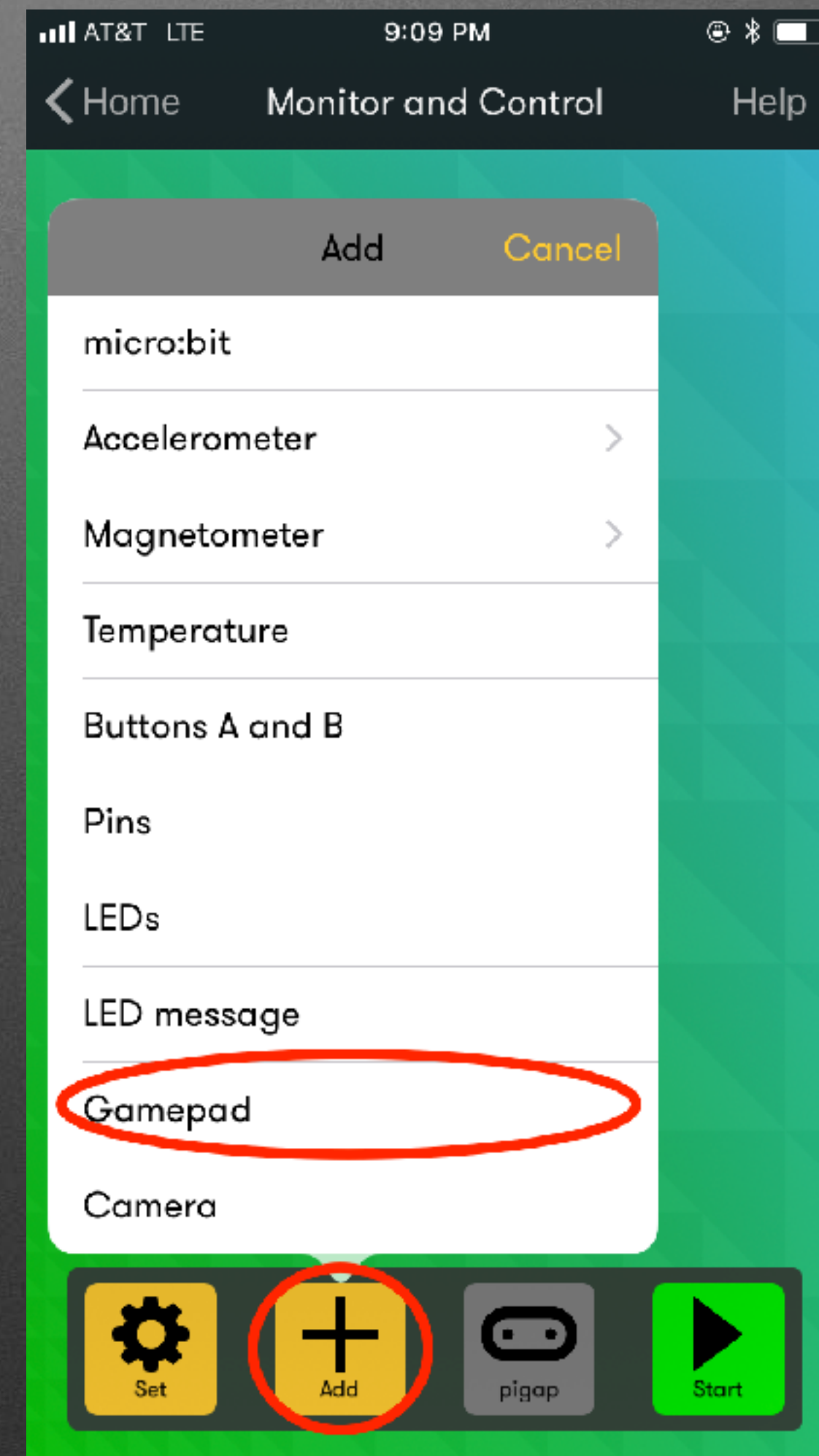
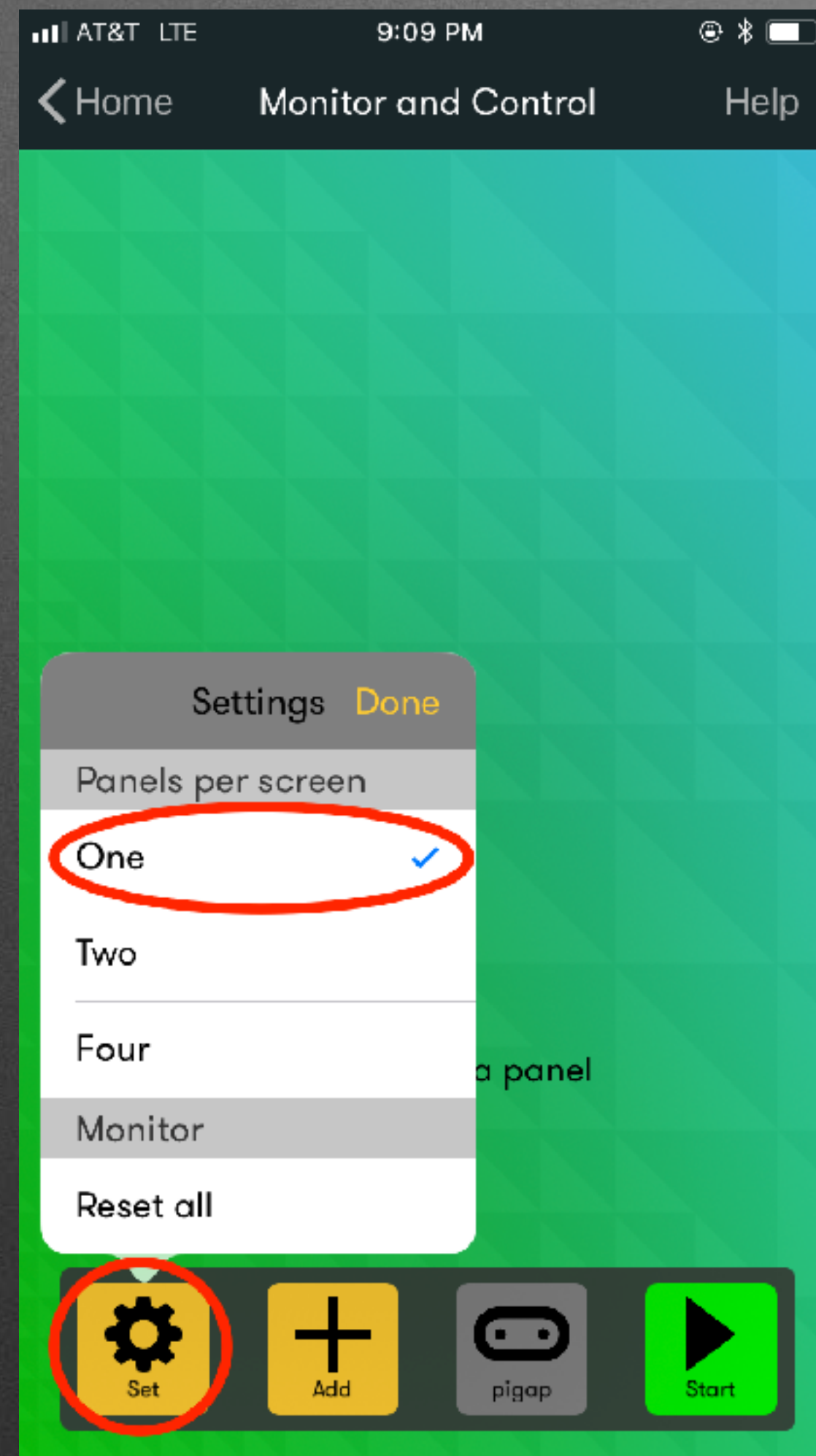
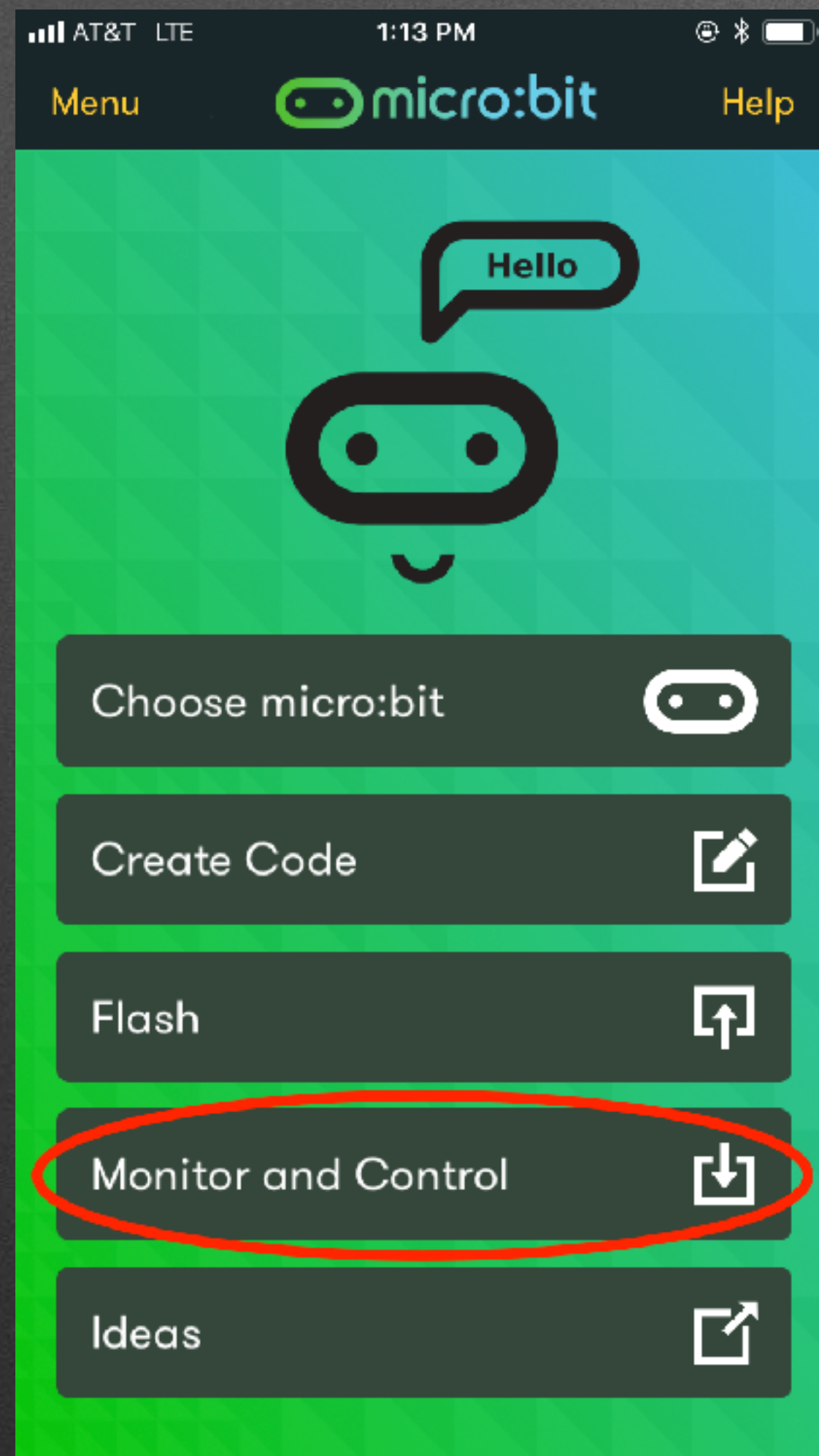
on gamepad button D down ▼

character ▼ change x ▼ by 1

Pro tip:  
Create one of  
these, then right-  
click and  
“duplicate” 3x,  
then modify



# App Configuration

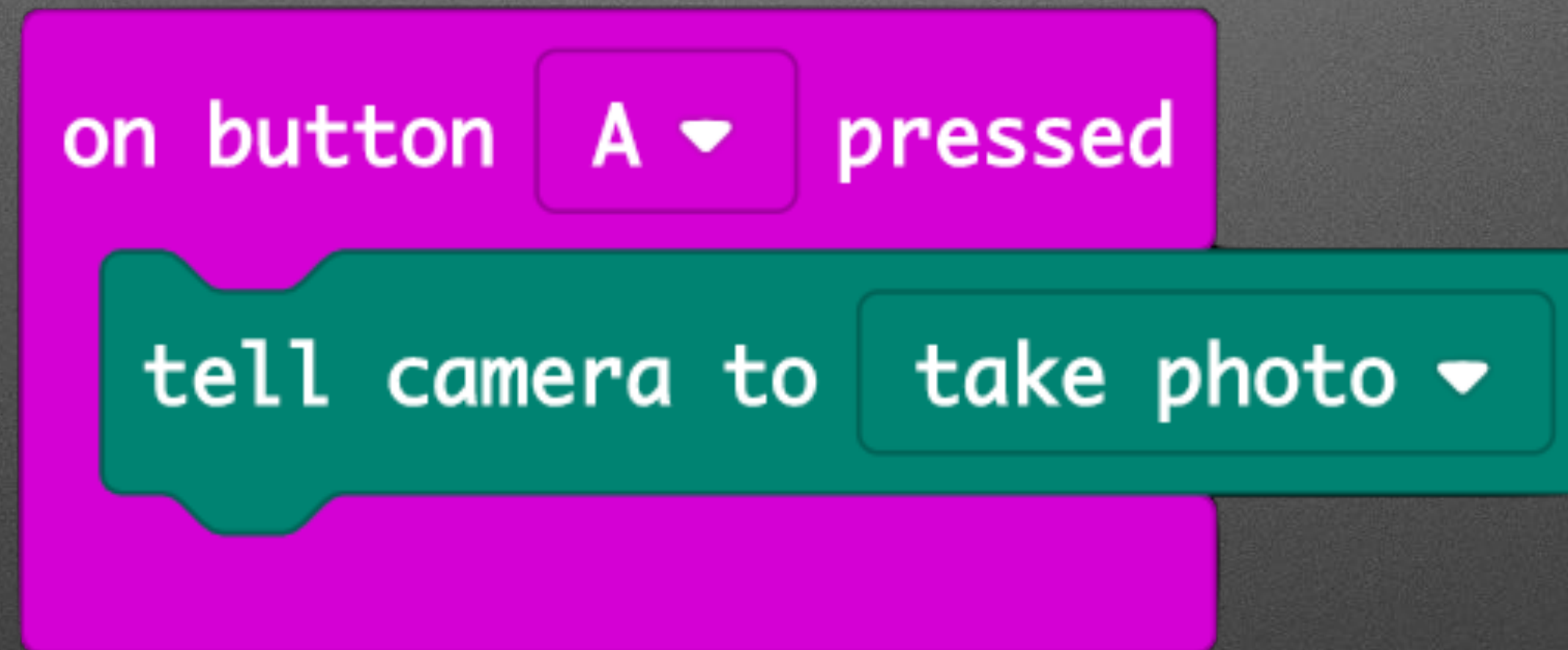




# Micro:bit Shutter Release



# Program

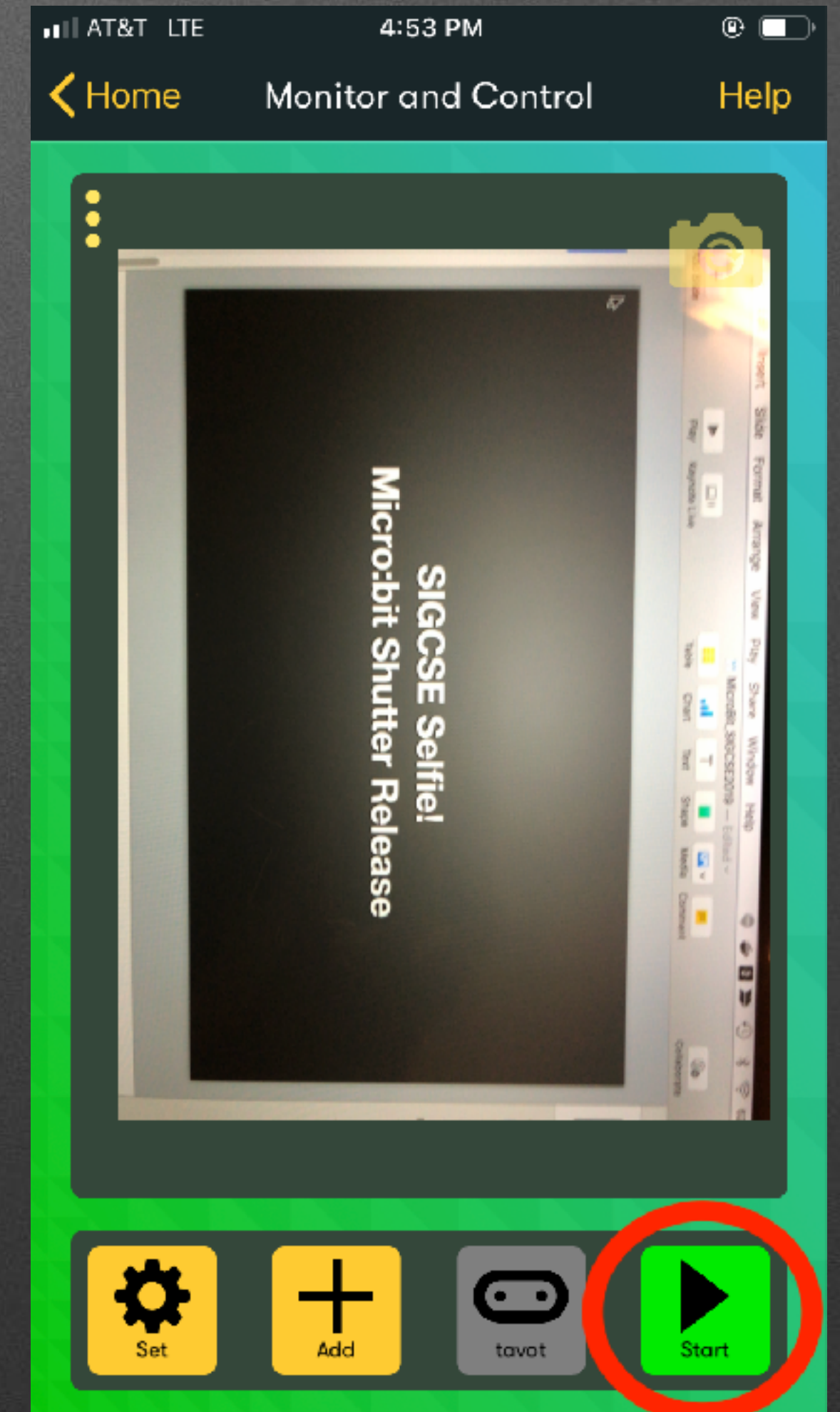
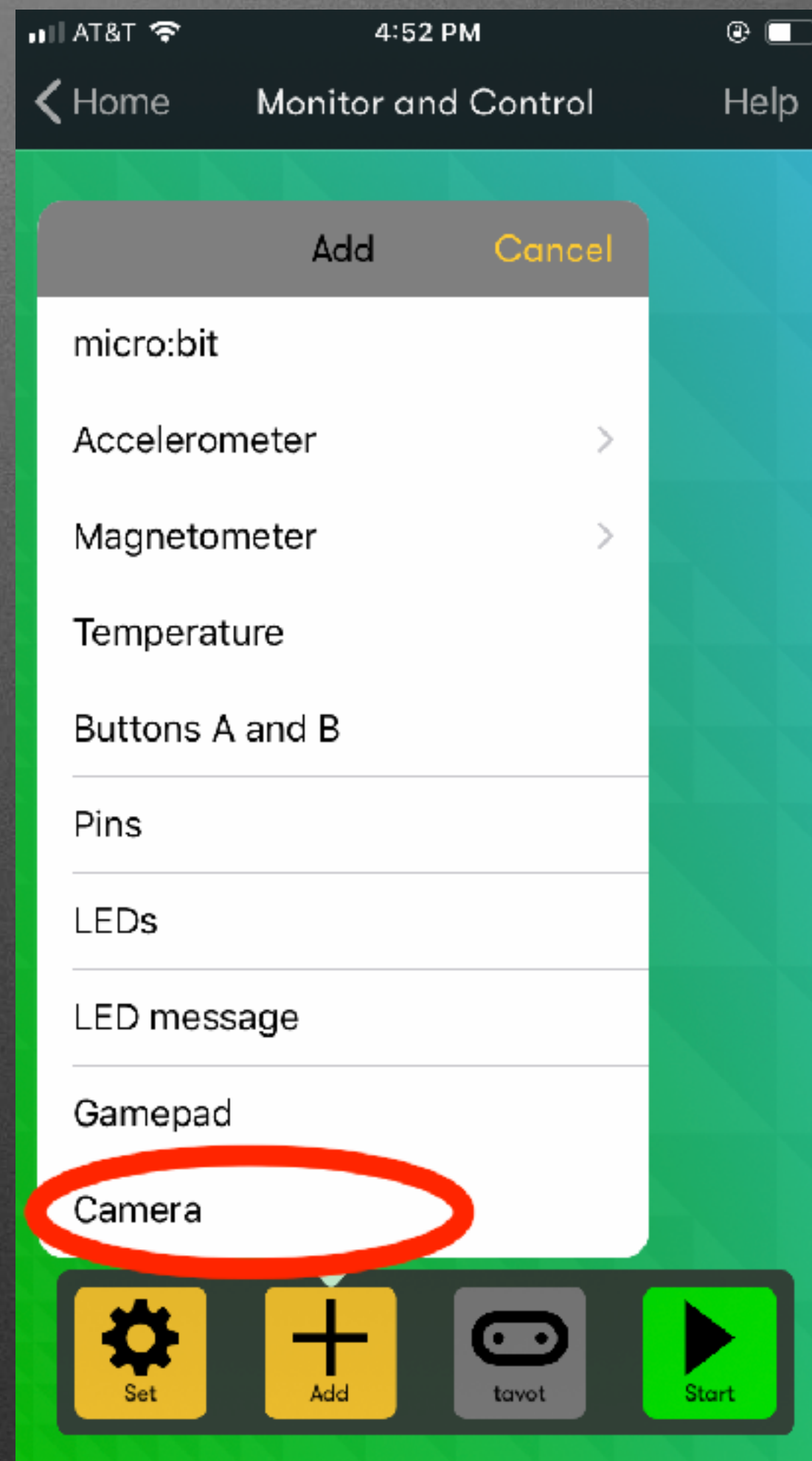
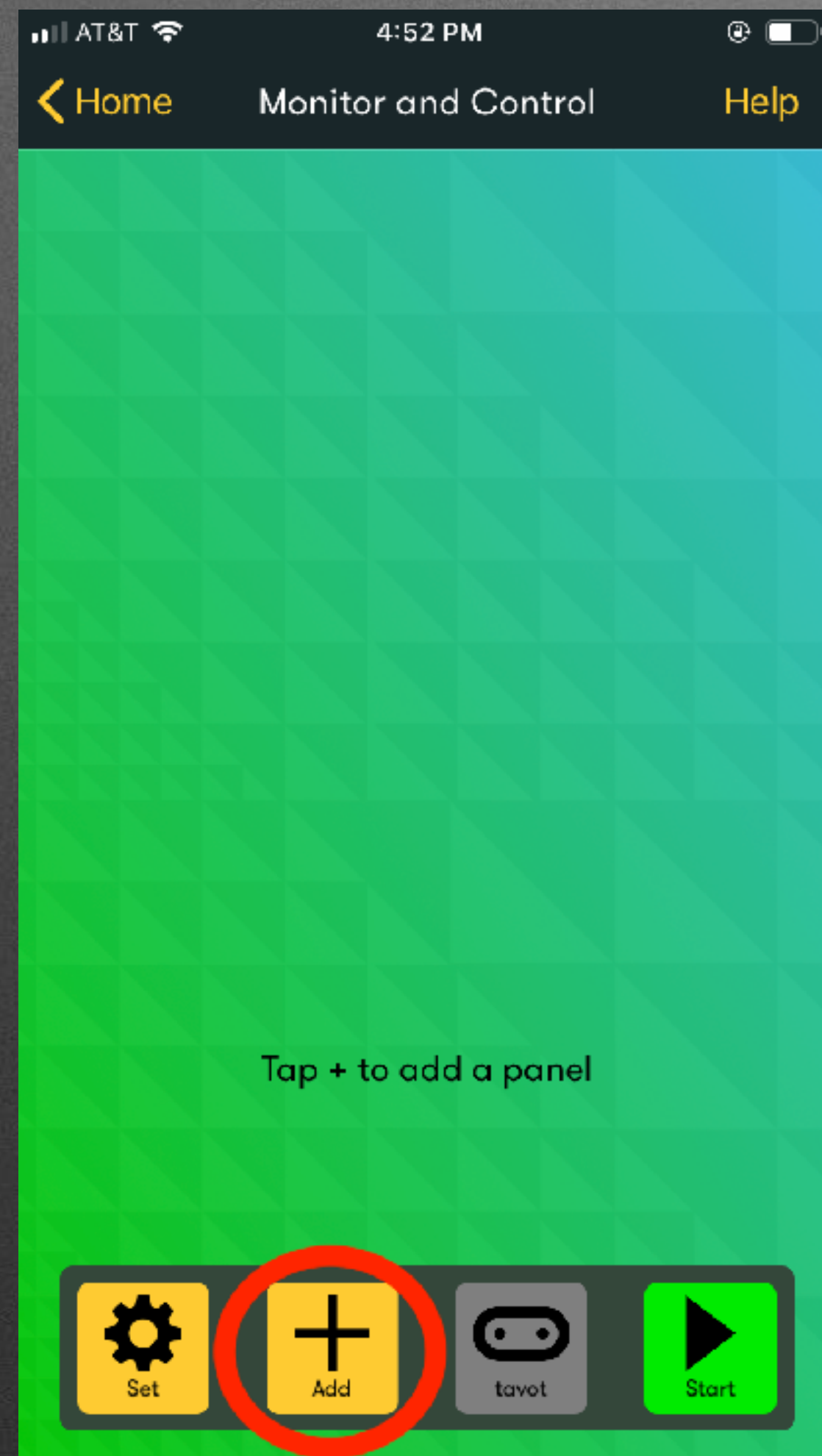
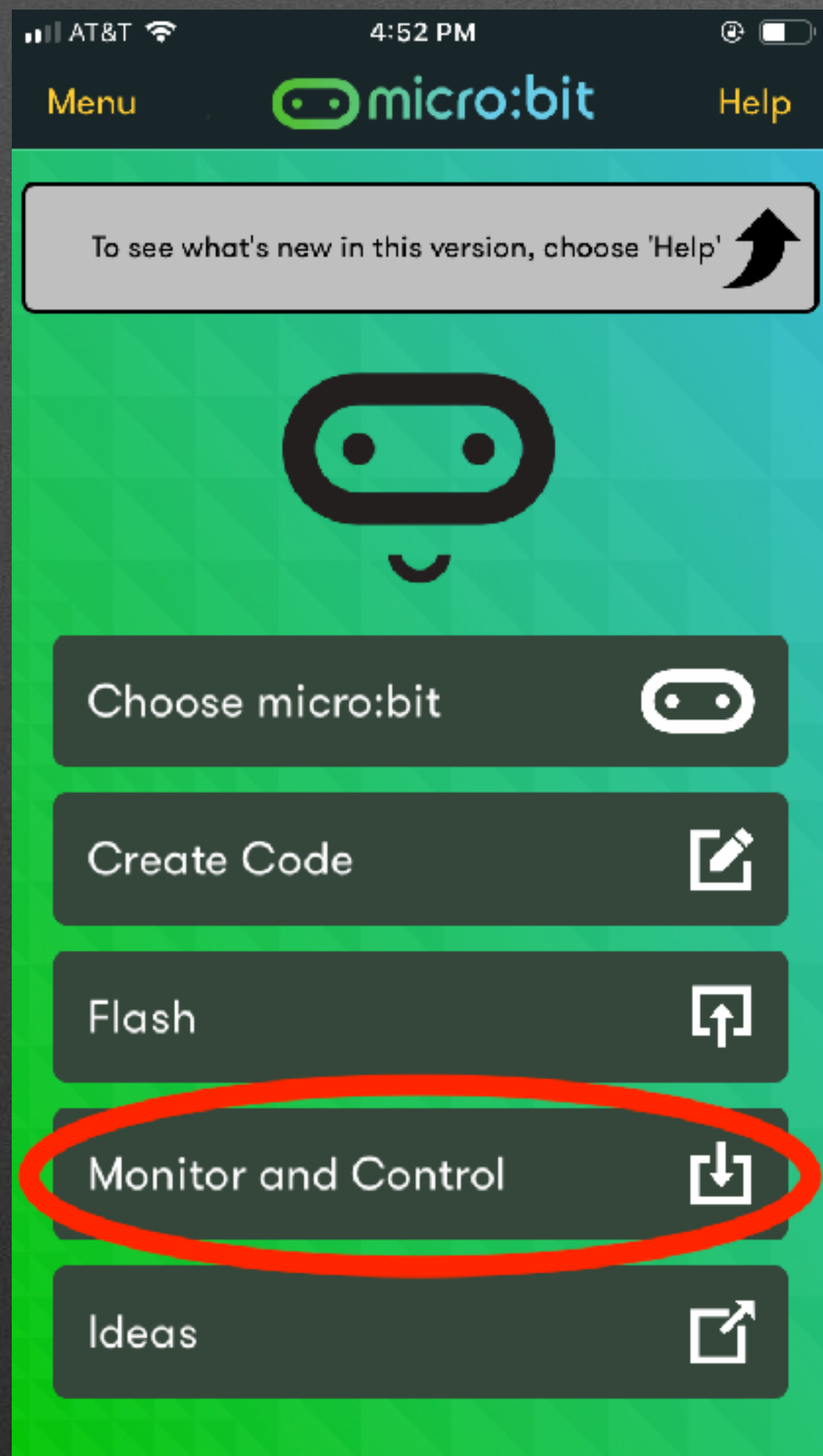




# App Config



# App Config





**SIGCSE Selfie!**



# Extra Hardware: Extensions

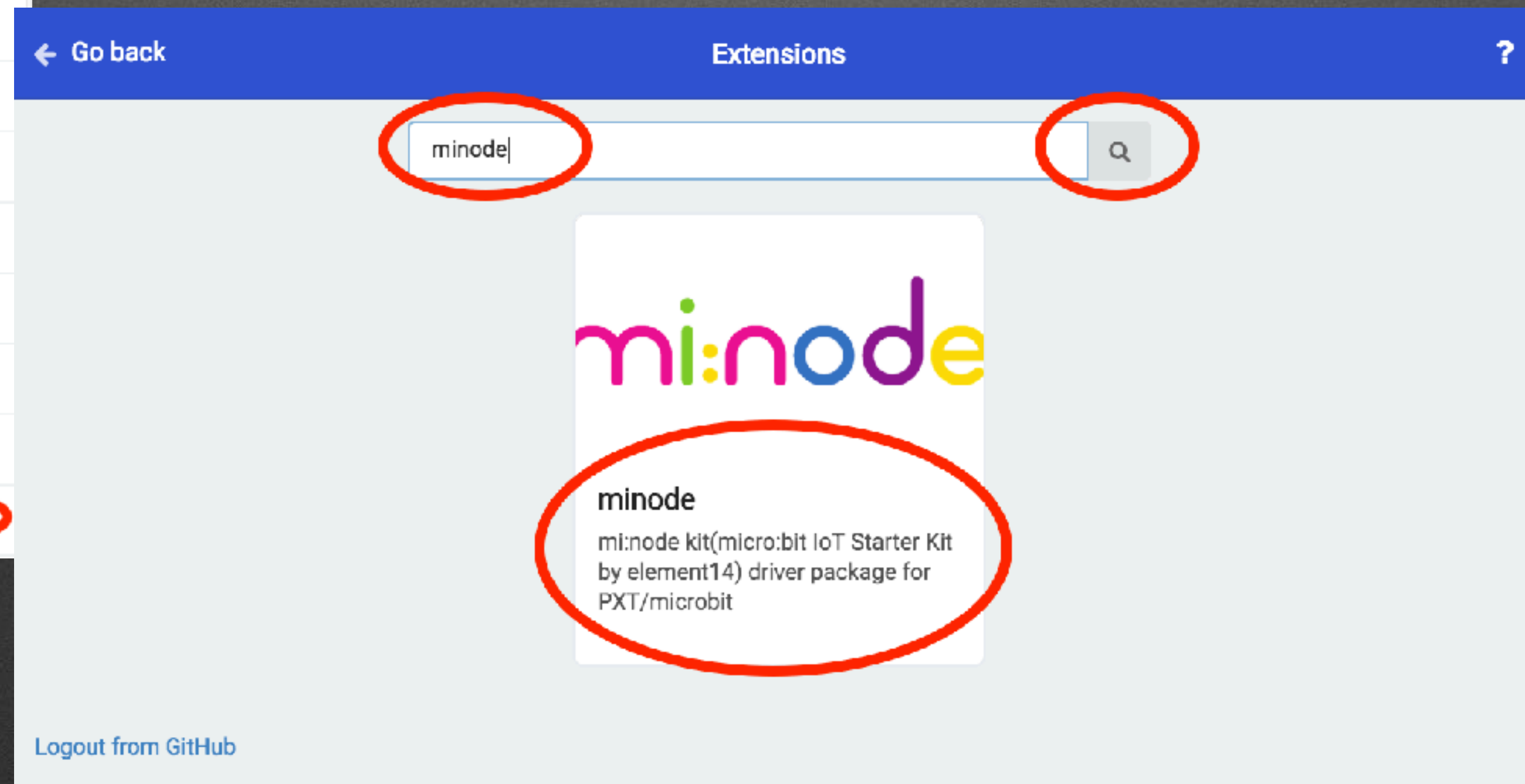
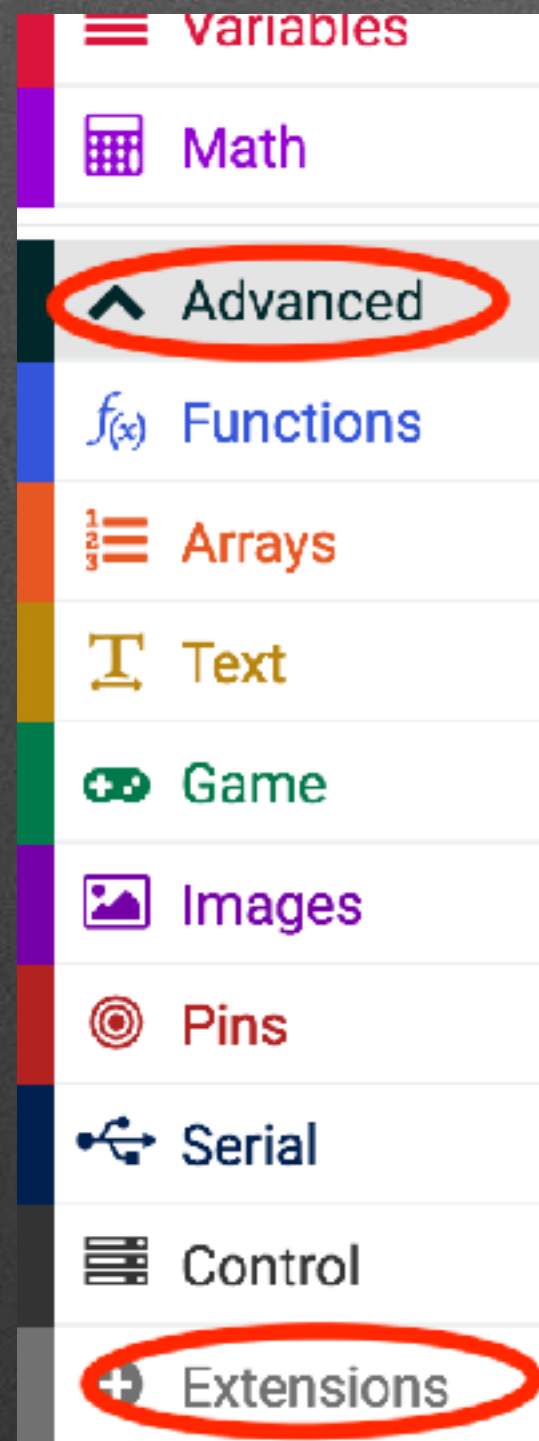
- Extensions...extend
  - Additional hardware support (today)
  - Additional simulator features



# Extra Hardware: Extensions

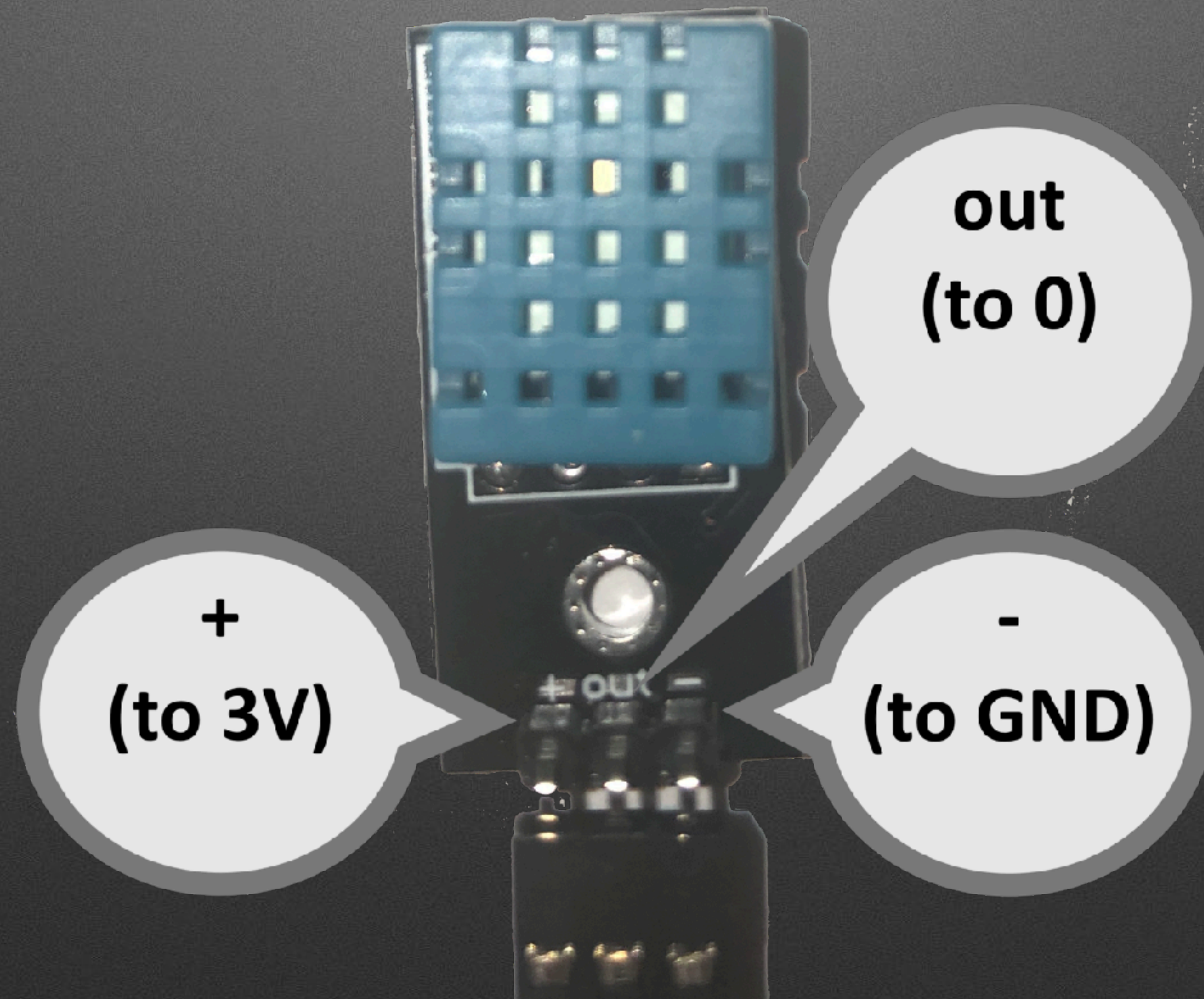


# Extra Hardware: Extensions





# Wiring Sensor





# Collecting Data

forever

serial write value "Temp" = dht11 A0 ▼ tempreature Celsius ▼

serial write value "Humidity" = dht11 A0 ▼ humidity



# Collecting Data

Minode Palette

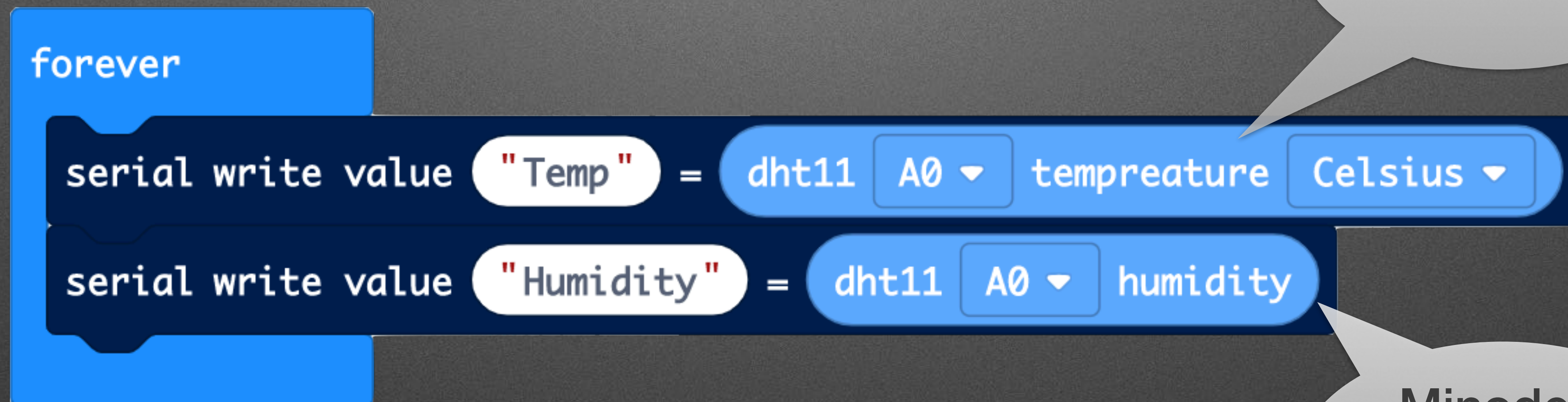
forever

serial write value "Temp" = dht11 A0 ▼ tempreature Celsius ▼

serial write value "Humidity" = dht11 A0 ▼ humidity



# Collecting Data



Minode Palette

Minode's  
"...more" Palette



# Collecting Data



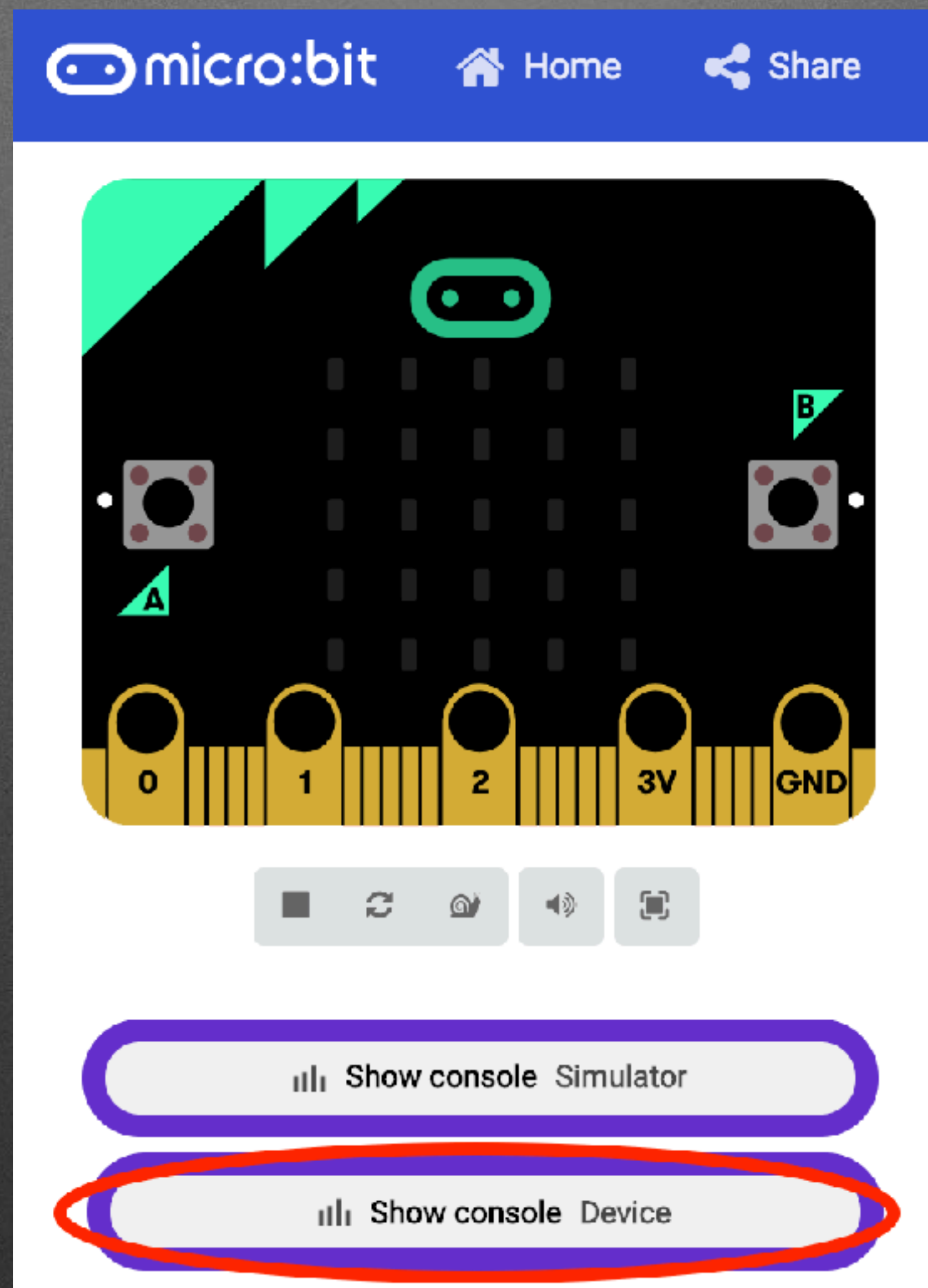
Minode Palette

Advanced  
Serial Palette

Minode's  
"...more" Palette



# Graphing

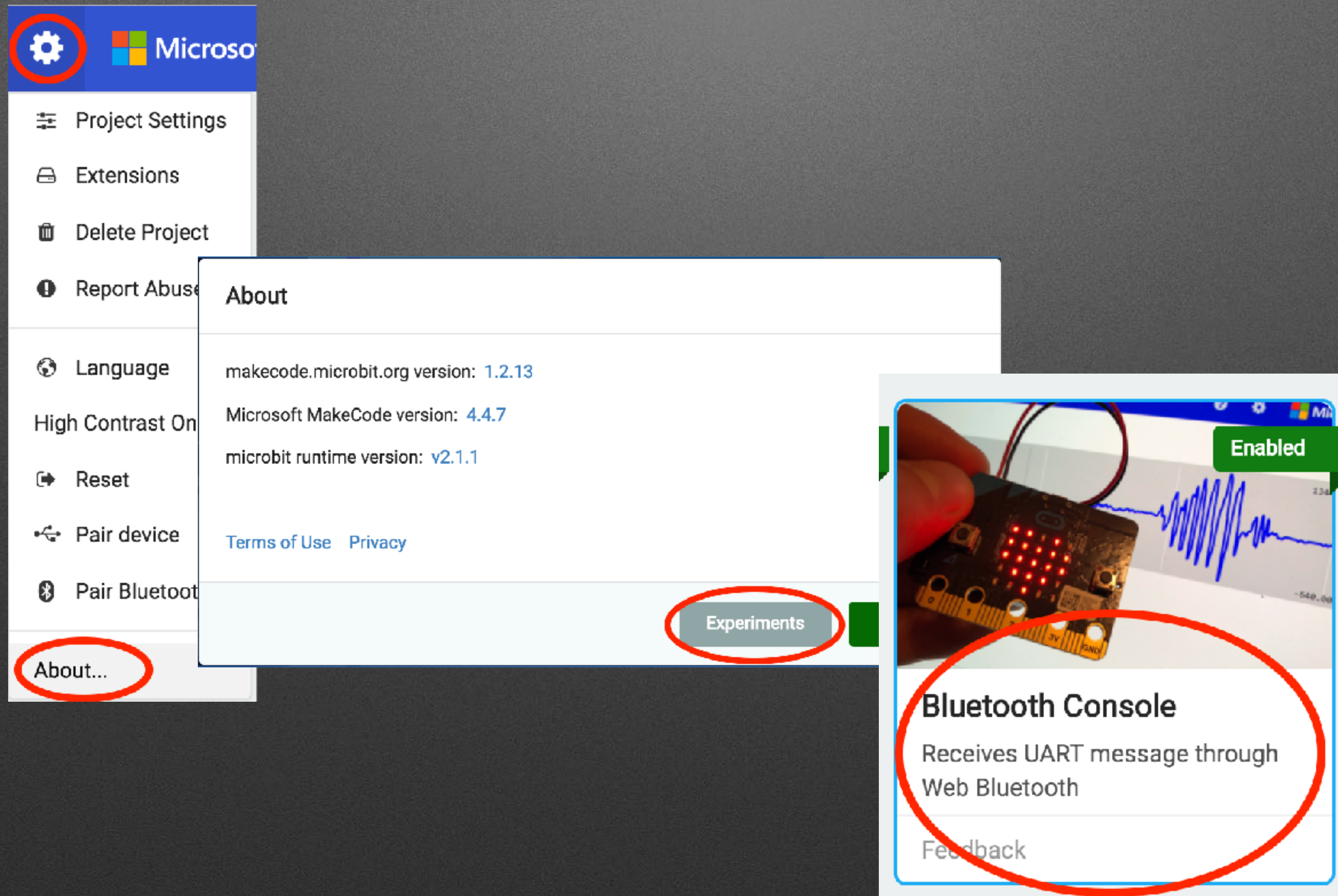




# Bluetooth Streaming: Setup



# Bluetooth Streaming: Setup





# Bluetooth Streaming: Program

on start

bluetooth uart service

forever

bluetooth uart write value "a.x" = acceleration (mg) x ▼

bluetooth uart write value "a.y" = acceleration (mg) y ▼

bluetooth uart write value "a.z" = acceleration (mg) z ▼

bluetooth uart write value "str" = acceleration (mg) strength ▼



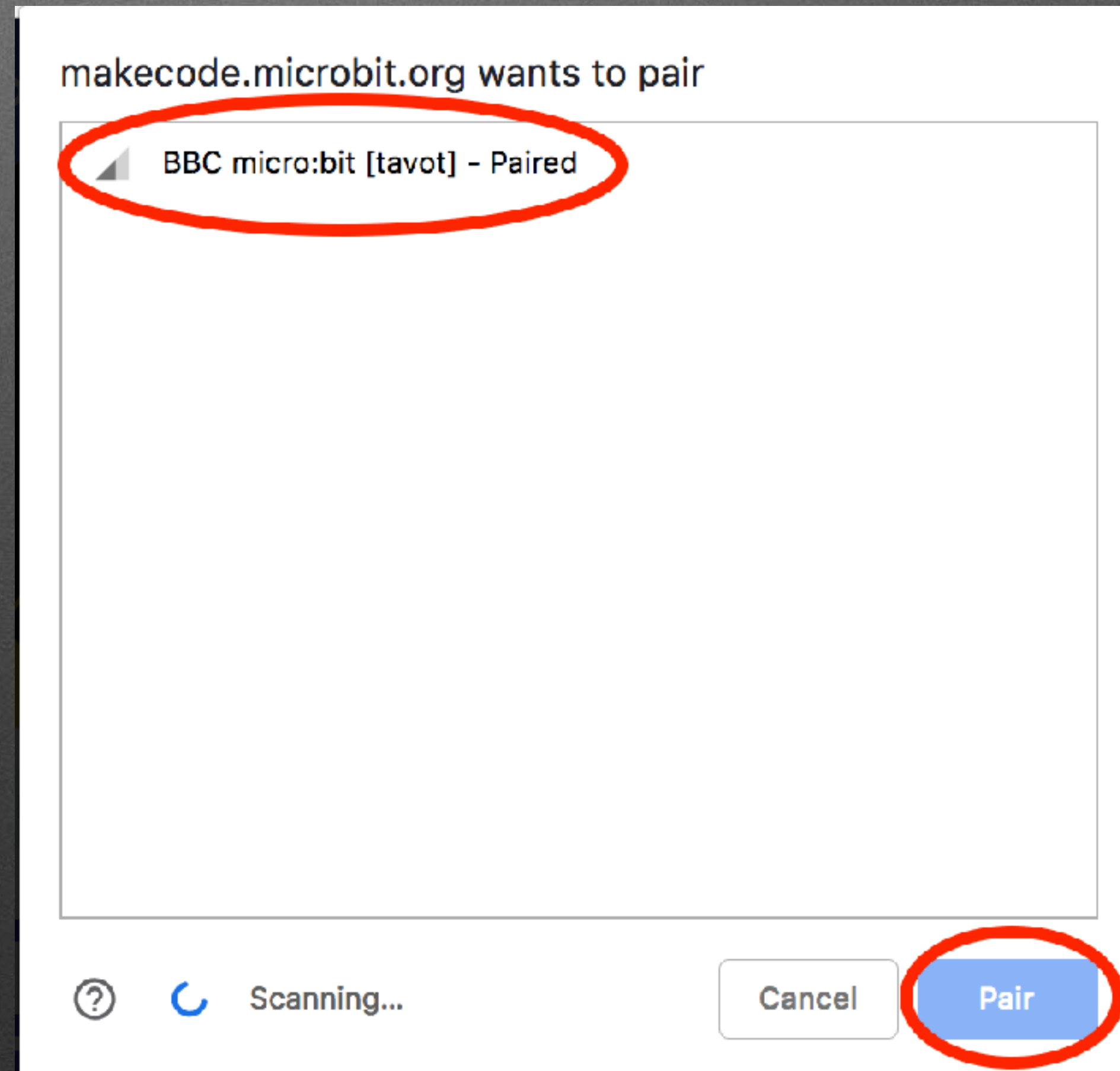
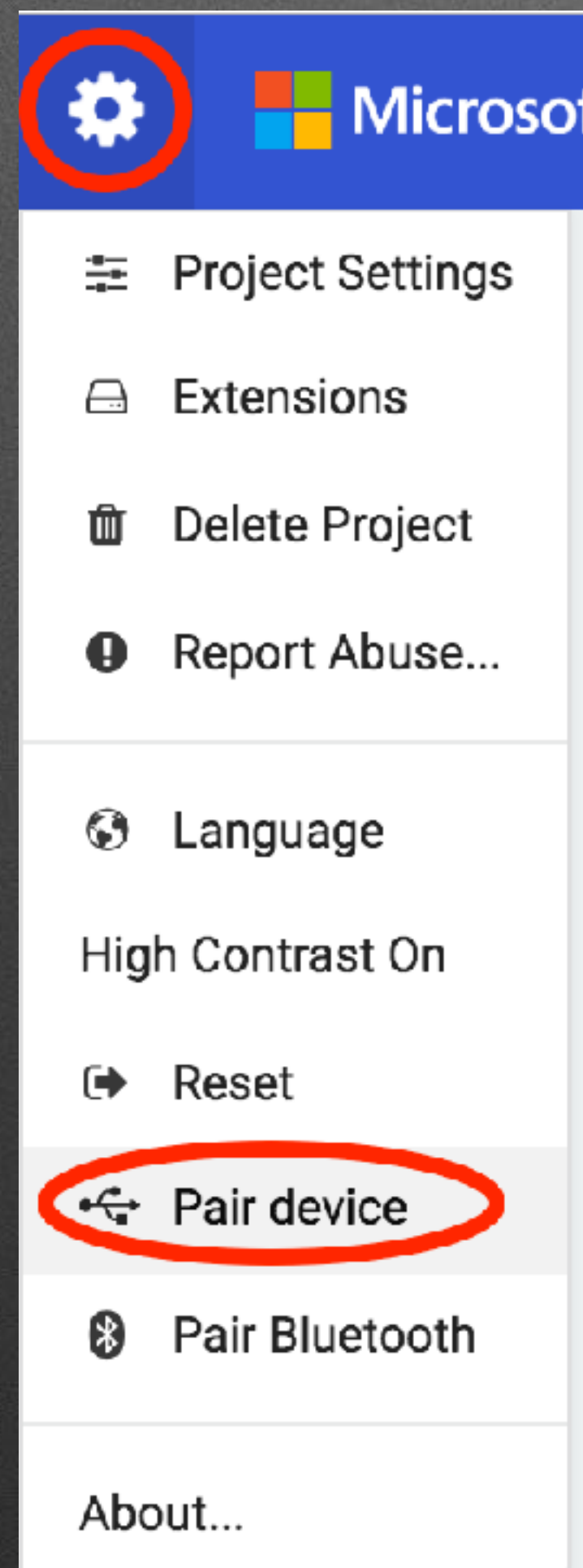
# Pairing Process

(Settings>No Pairing Required;  
but need to connect to micro:bit)



# Pairing Process

(Settings>No Pairing Required;  
but need to connect to micro:bit)





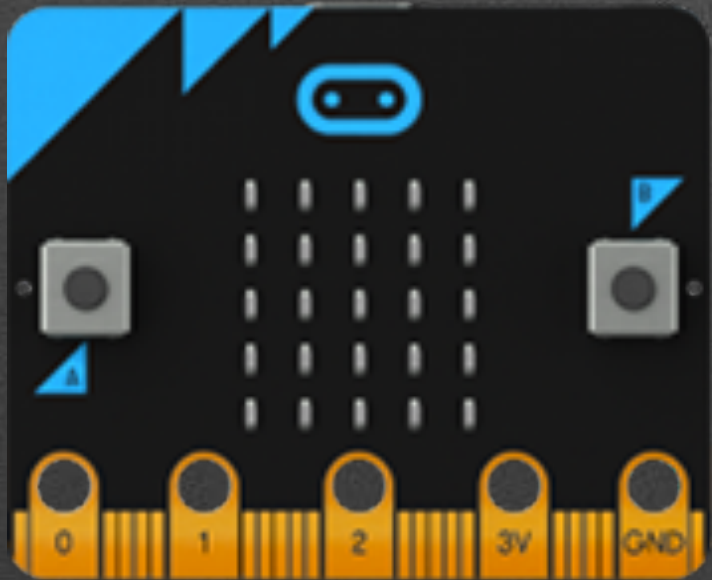
# IoT Example Overview



AdaFruit.io

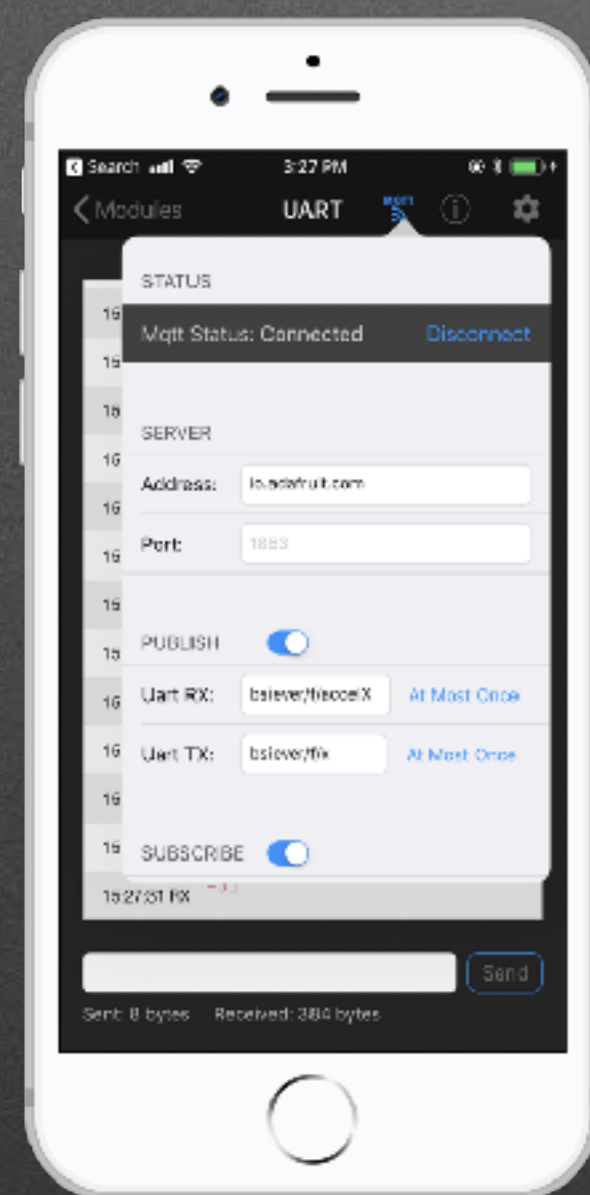
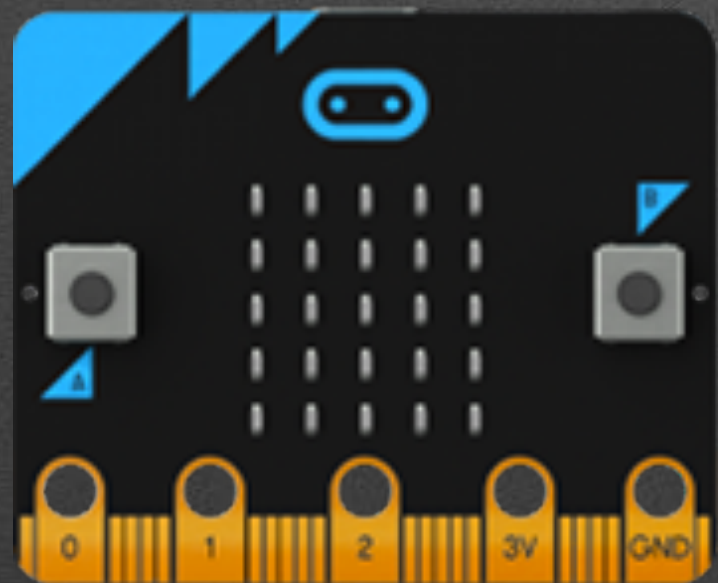


# AdaFruit.io



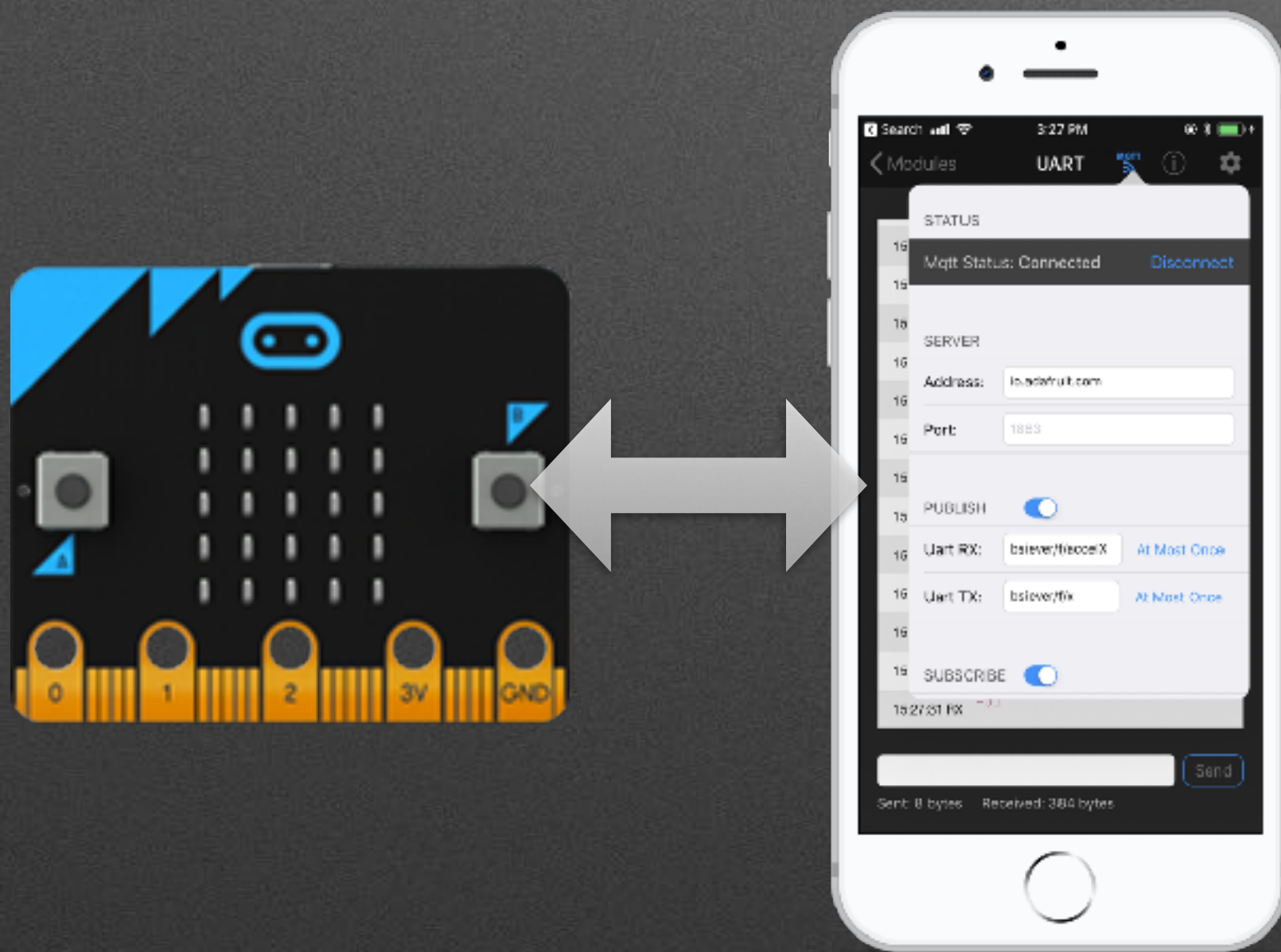


# AdaFruit.io



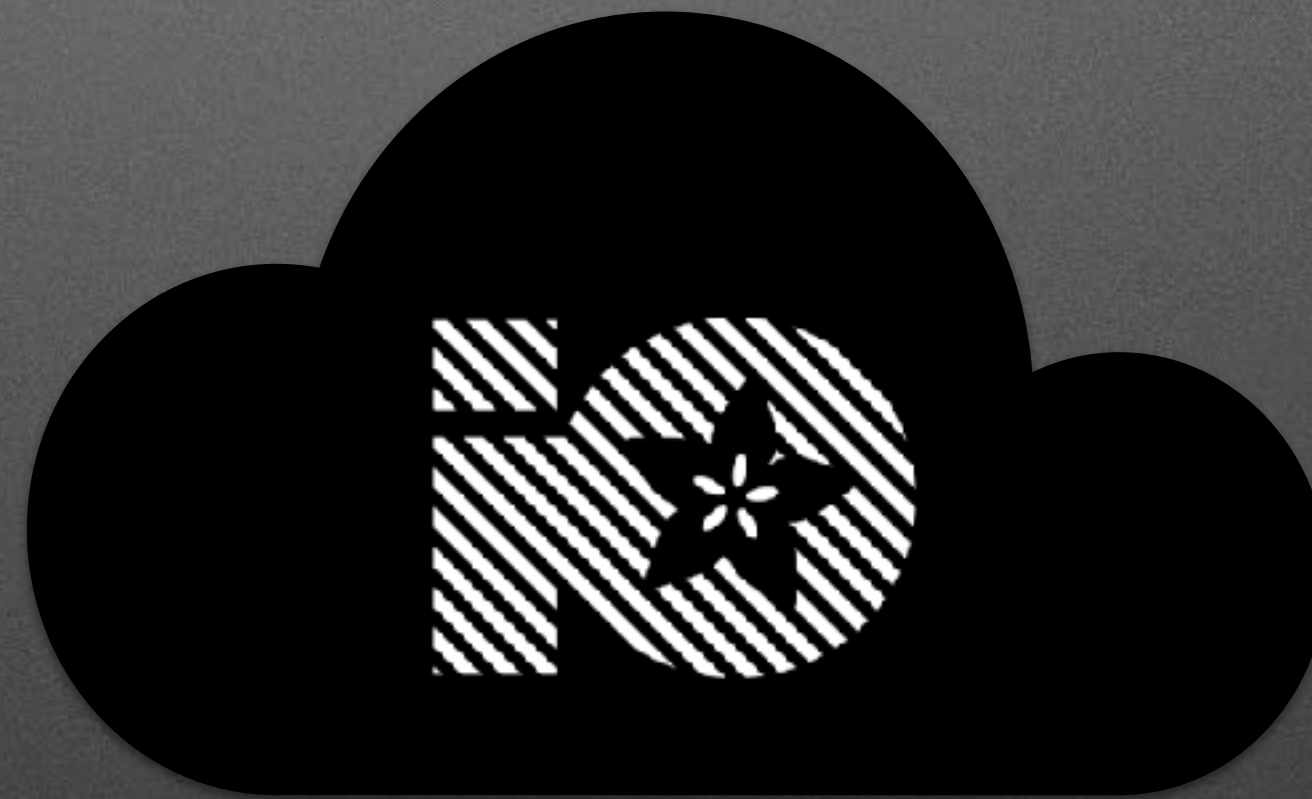
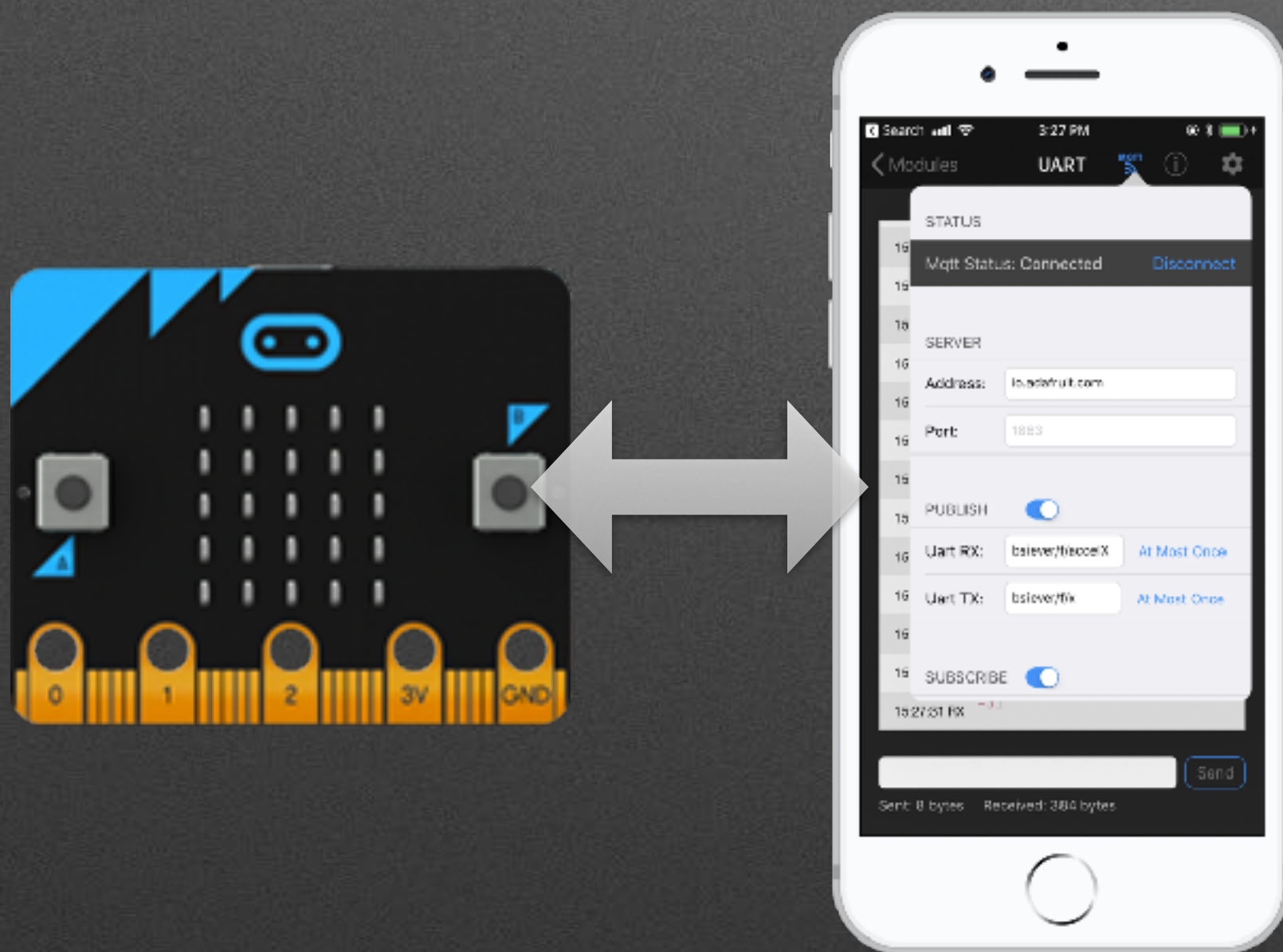


# AdaFruit.io





# AdaFruit.io



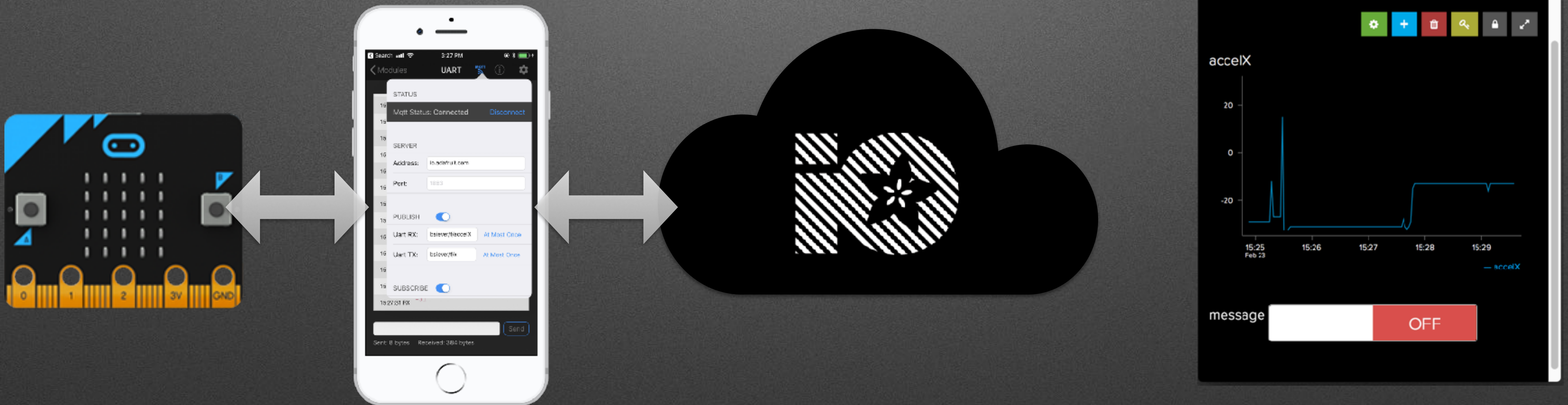


# AdaFruit.io



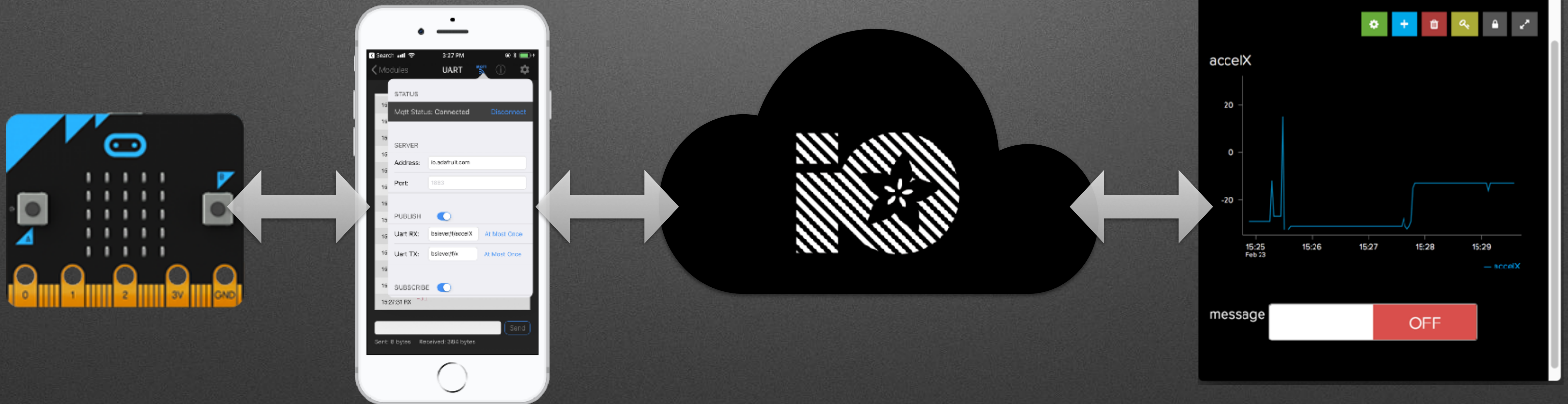


# AdaFruit.io





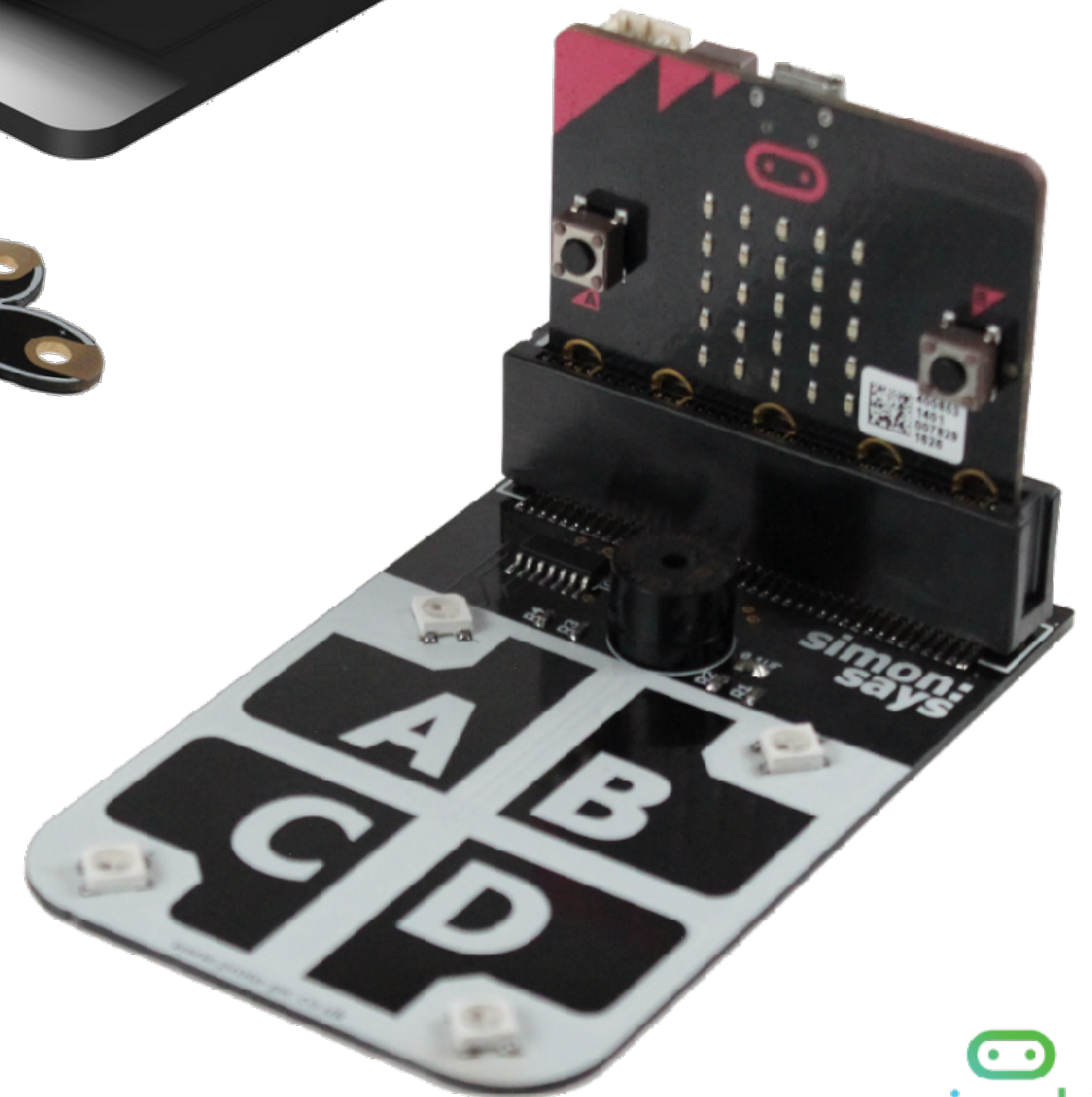
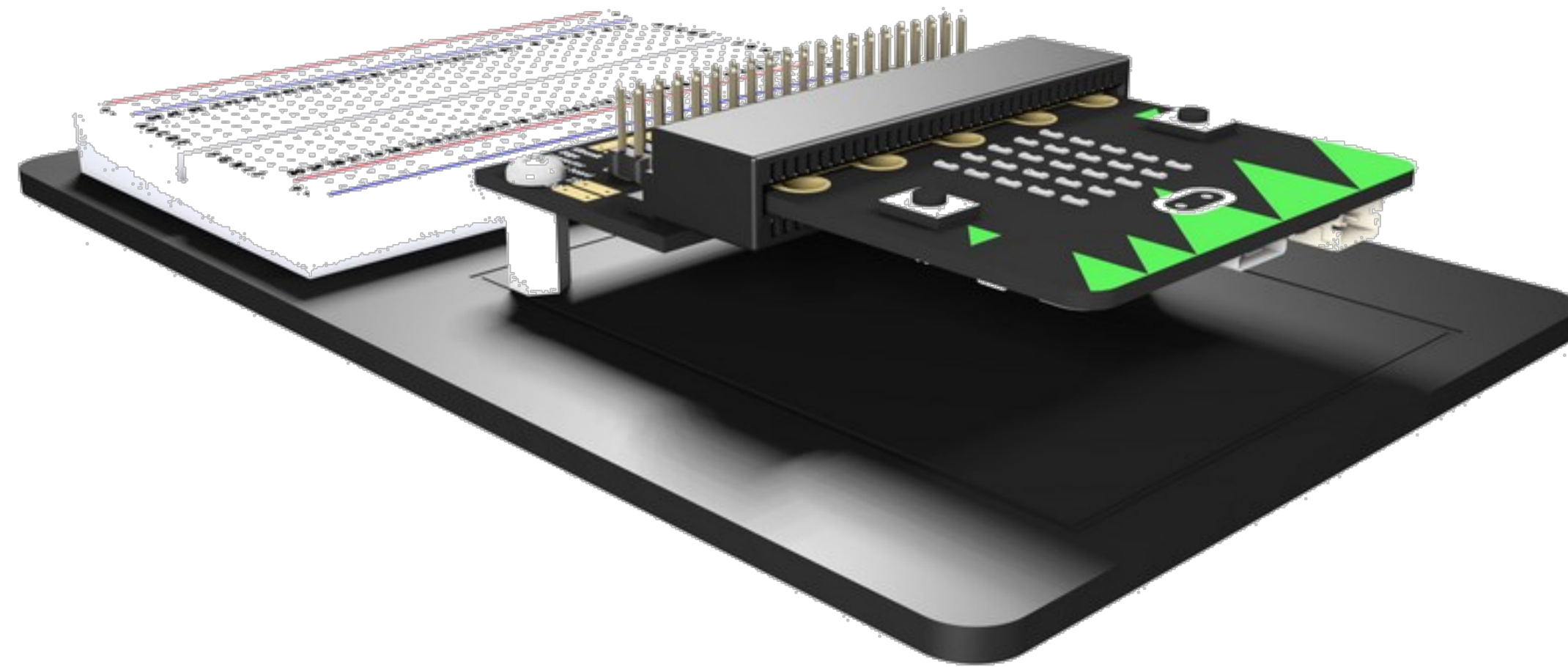
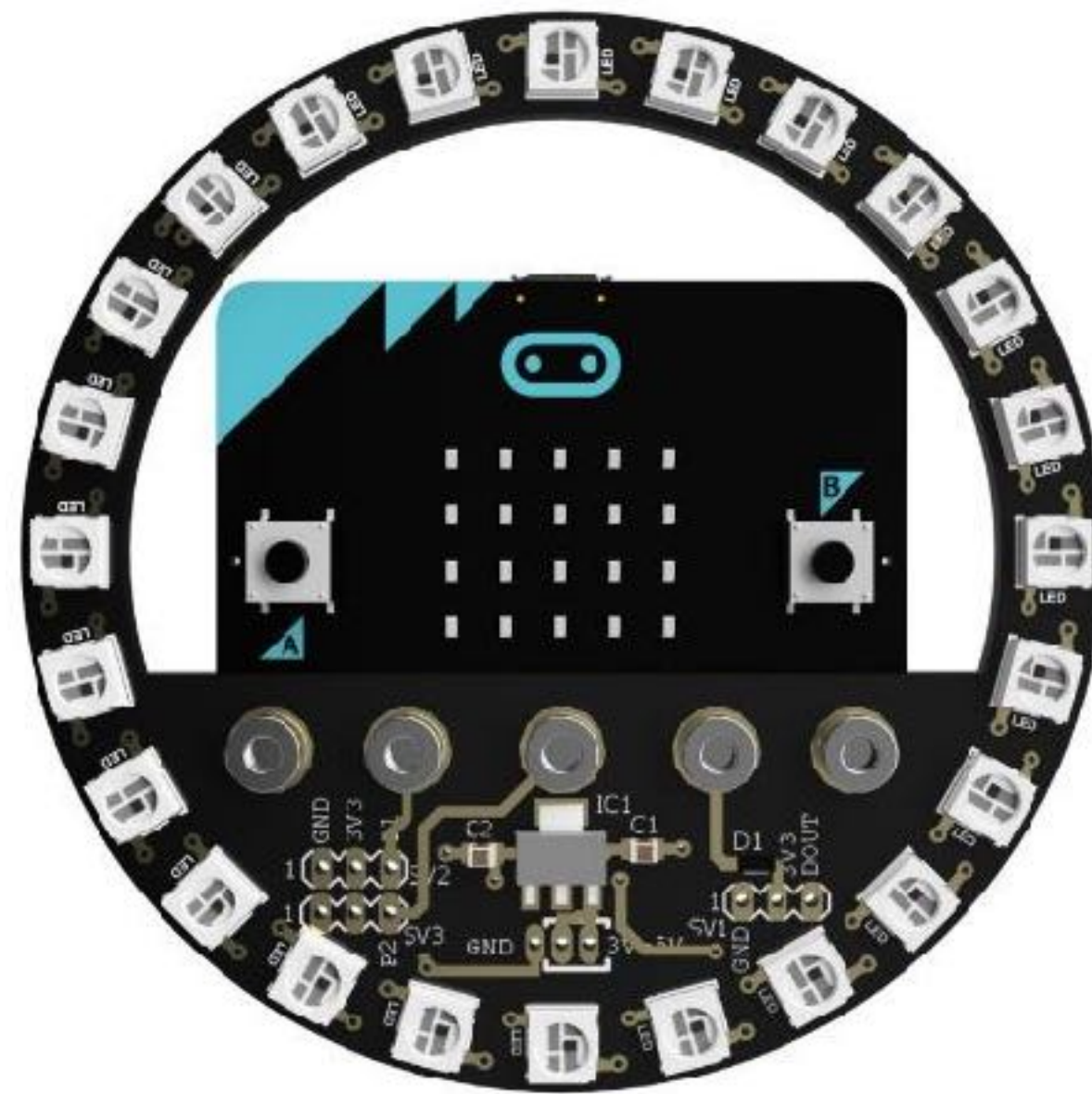
# AdaFruit.io





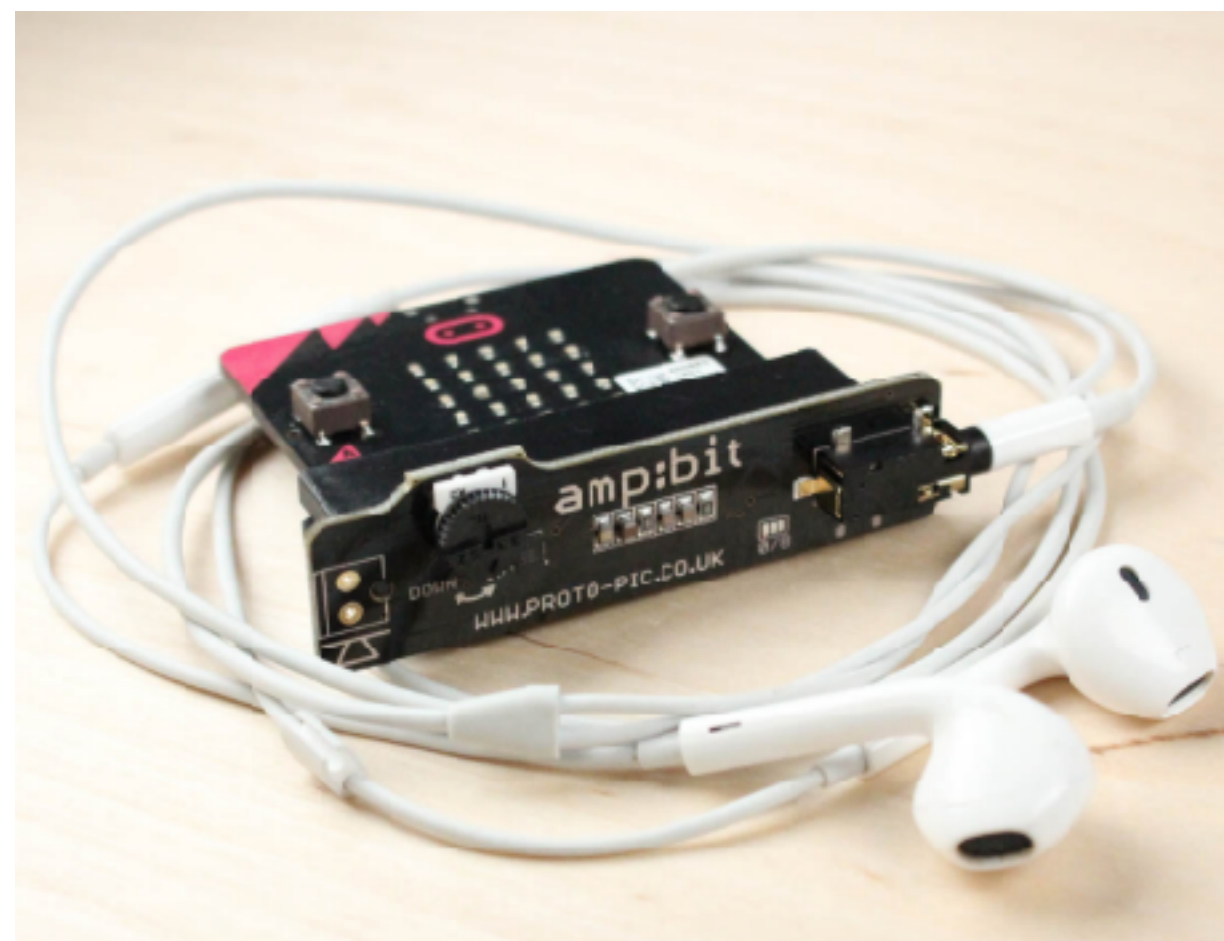
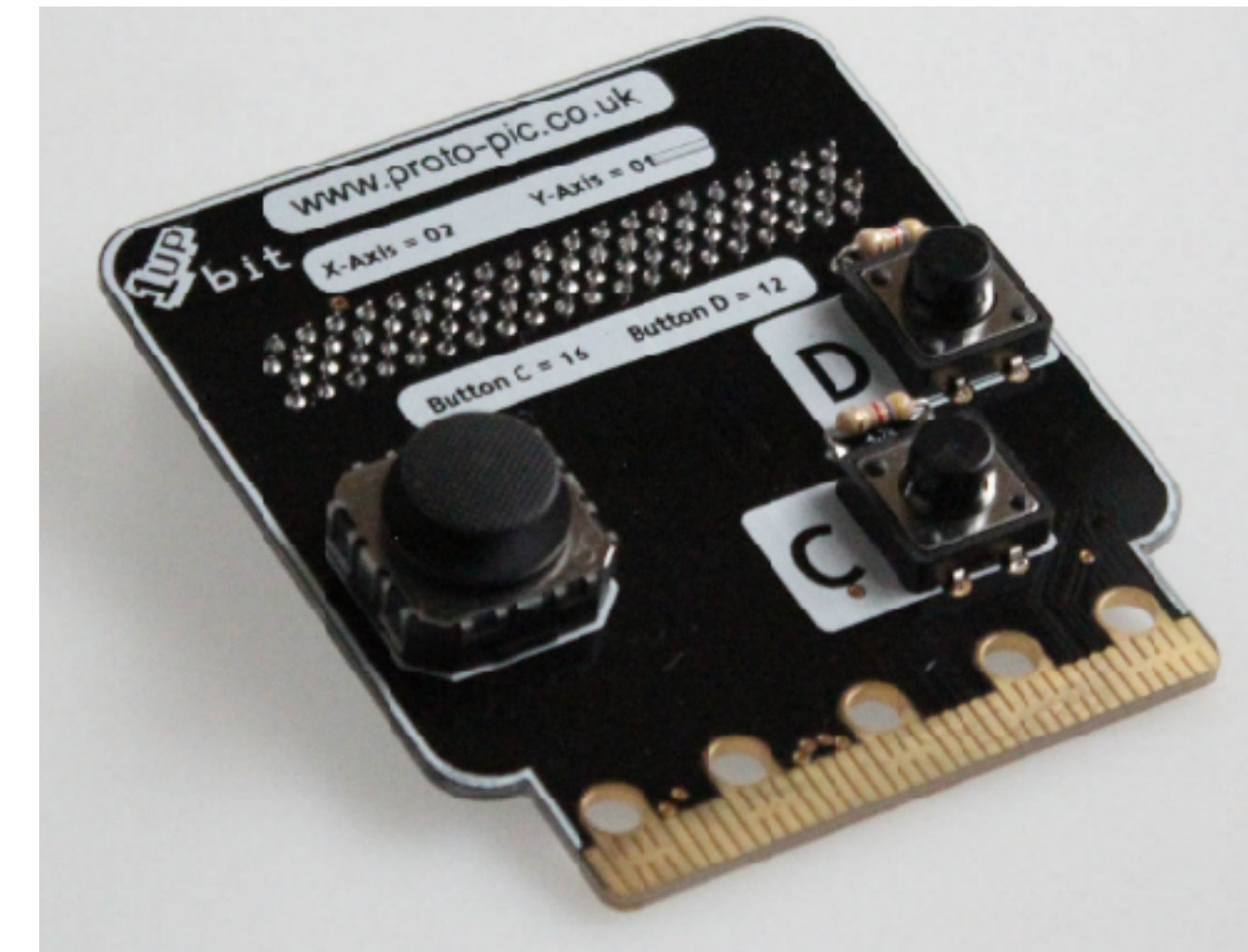
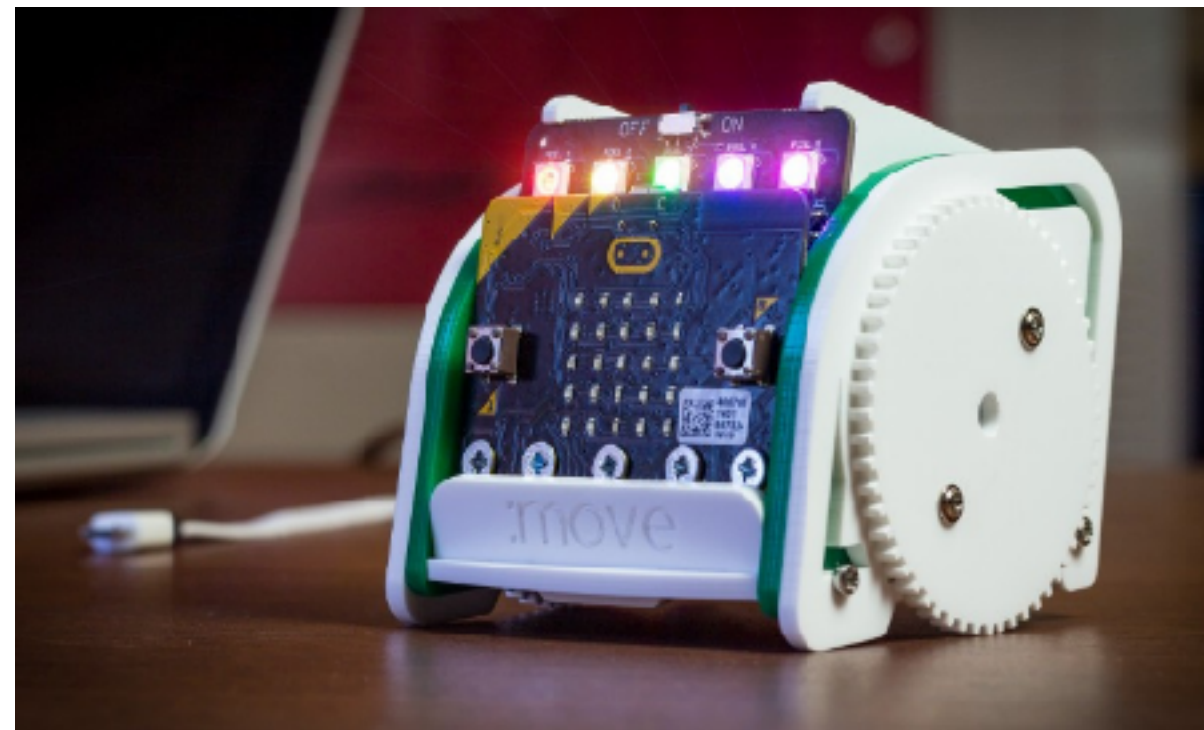
# Hardware Ecosystem





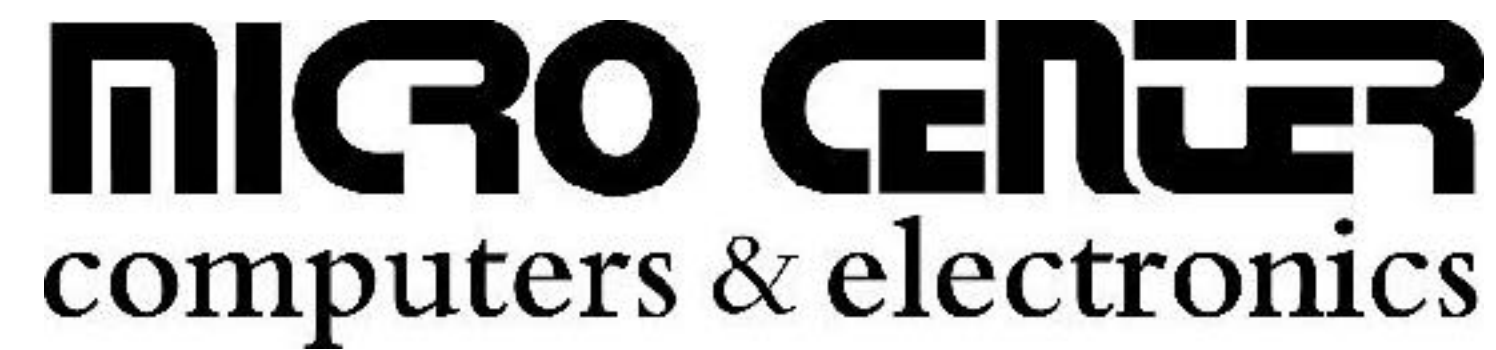
<http://microbit.org/assets/documents/microbit-accessories.pdf>







# U.S. Resellers





# Available via DonorsChoose.org



- AKJ Education is an approved DonorsChoose.org vendor and micro:bit reseller
- Teachers enter projects and request classroom materials
- Individuals and companies can donate money towards the purchase of those materials



# Misc.

- Address Safety!
  - Low voltage / low current vs. Mains power



**Bill's SIGCSE Blog Post**

**<https://tinyurl.com/SIGCSE19uBit>**



# Questions / Discussion







# Remove Add Bluetooth

(& remove Radio)



# Remove Add Bluetooth

(& remove Radio)

