

# 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?

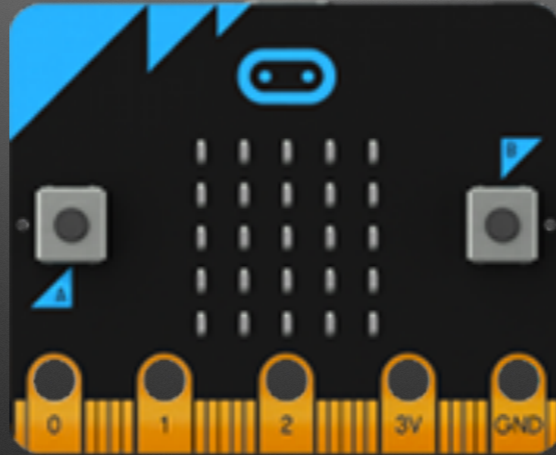
- ## Intros: Us & You

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

## **Intros: the micro:bit**



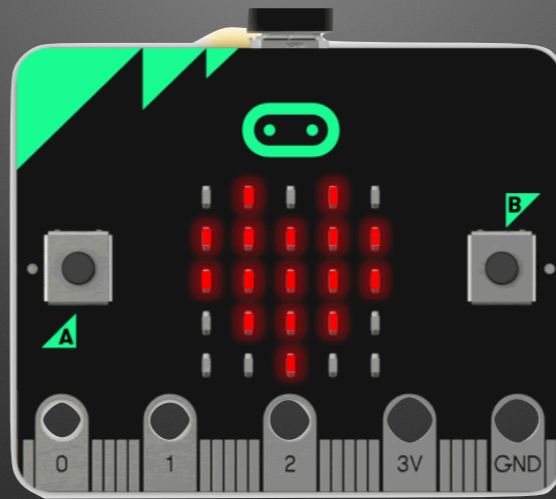
## Small



5cm x 4cm

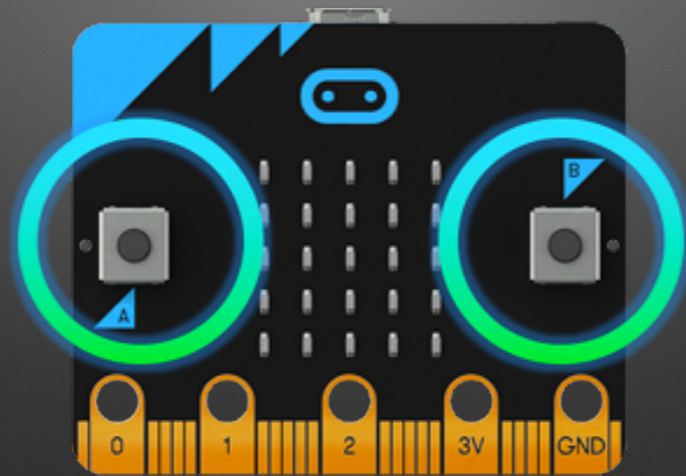
Artwork source: <http://microbit.org/images/microbit-features-temp.png>

# LED Grid



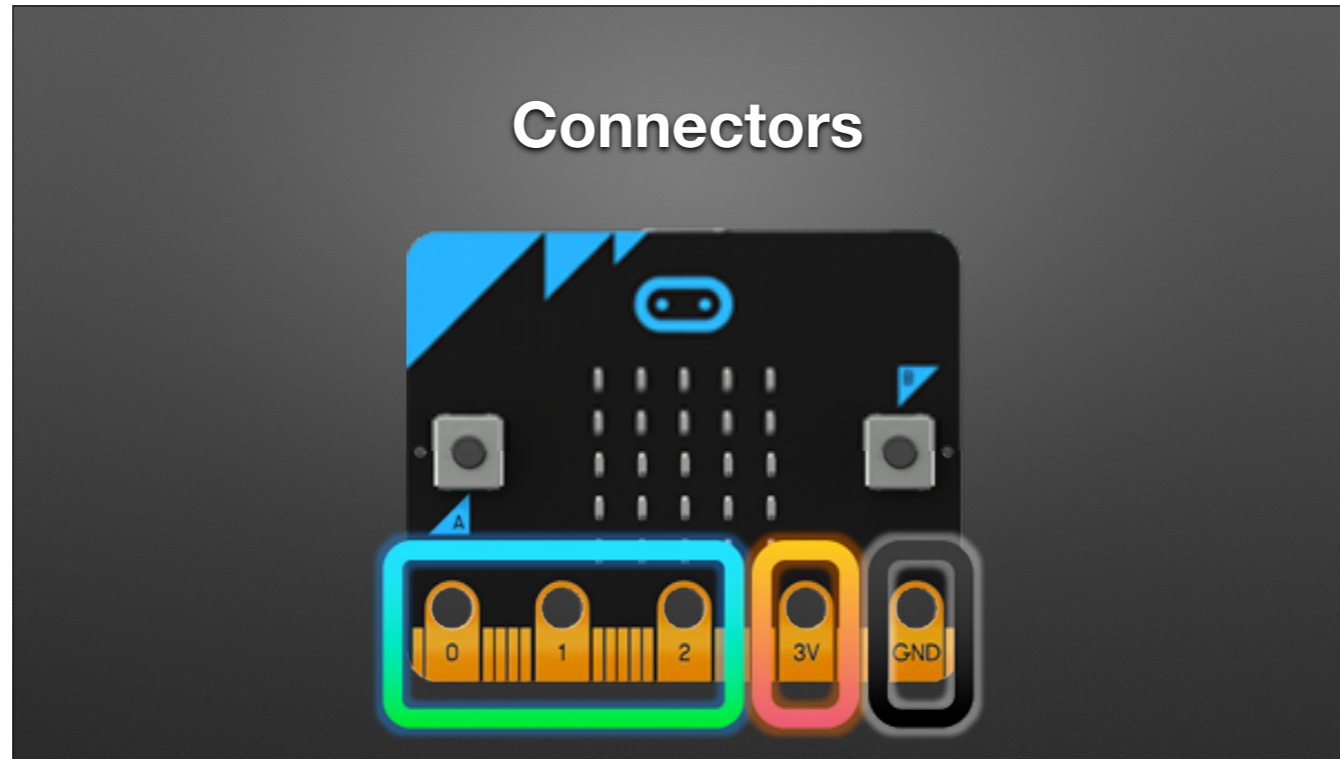
5xm x 4cm

# Buttons



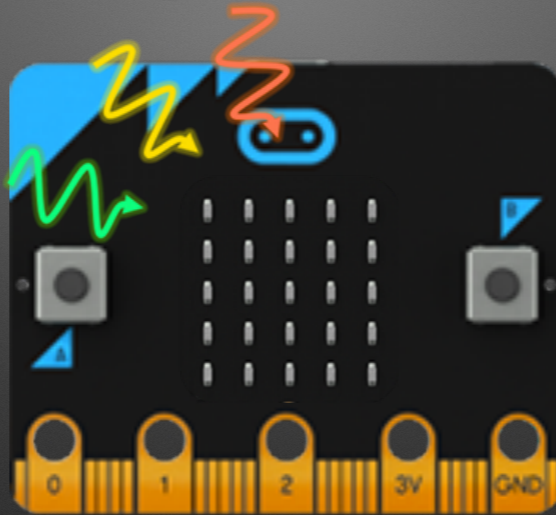
Artwork source: <http://microbit.org/images/microbit-features-buttons.png>

# Connectors



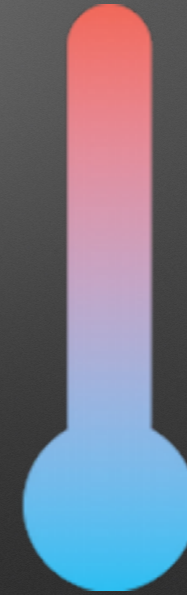
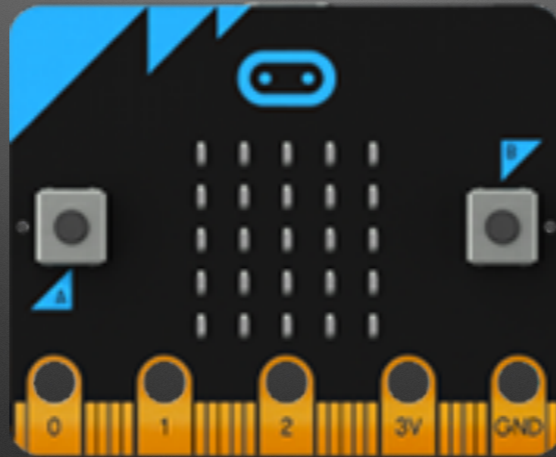
Artwork source: <http://microbit.org/images/microbit-features-pins.png>

## Light Sensor



Artwork: <http://microbit.org/images/microbit-features-light.png>

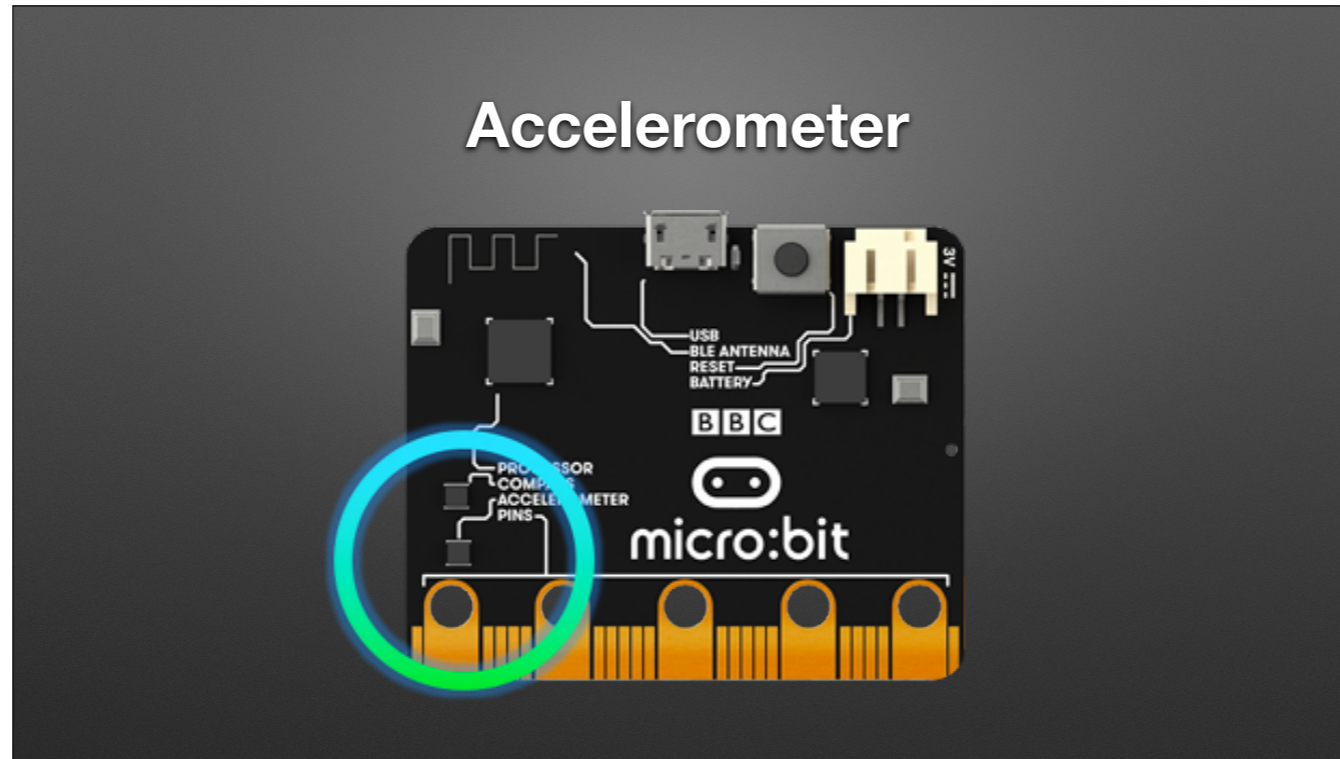
## Temperature Sensor



Within about 2 degrees C (die temperature)

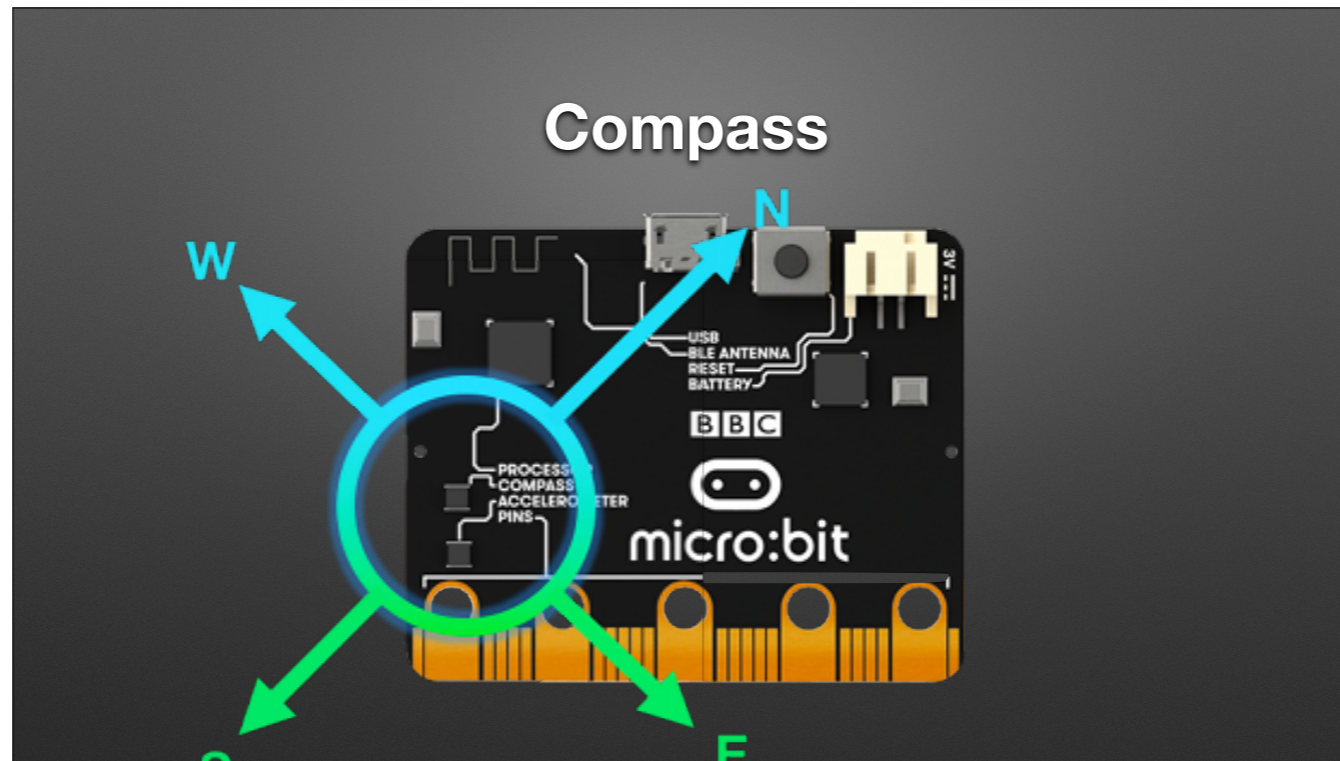
Artwork source: <http://microbit.org/images/microbit-features-temp.png>

# Accelerometer



Detect/respond to tilt/tip/shake/etc.

Artwork source: <http://microbit.org/images/microbit-features-accelerometer.png>



5cm x 4cm

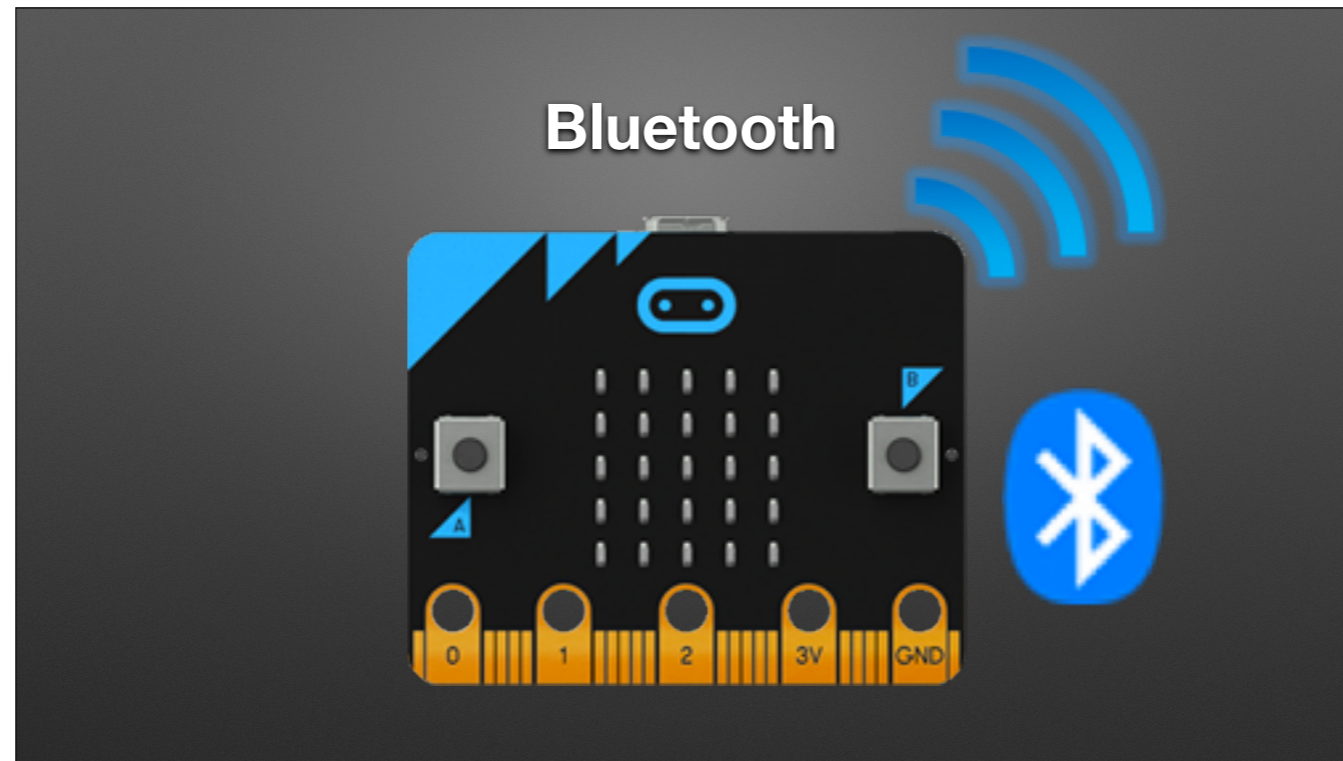
Artwork source:<http://microbit.org/images/microbit-features-compass.png>





5xm x 4cm

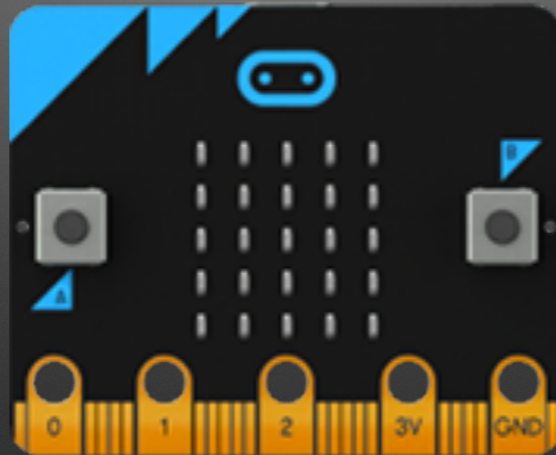
Artwork source: <http://microbit.org/images/microbit-features-radio.png>



Bluetooth: It can talk to mobile devices!!!

Artwork source: <http://microbit.org/images/microbit-features-bluetooth.png>

Low Cost: ~\$13 US



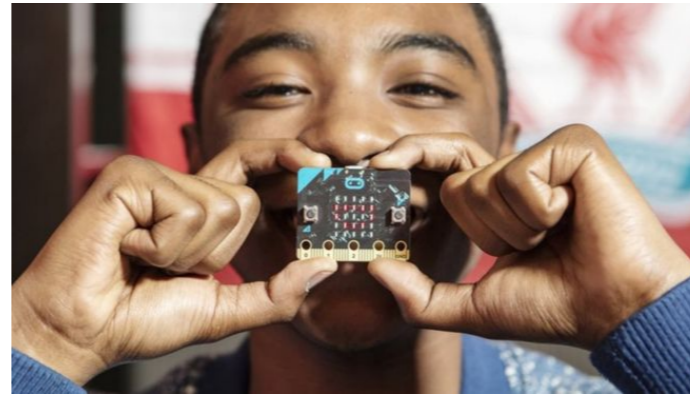
Artwork source: <http://microbit.org/images/microbit-features-temp.png>

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

Thanks to The Micro:bit Educational Foundation and Hal Speed for the following slides. (Hal is Chief of Global Engagement; Micro:bit foundation is a non-profit)

## 2015

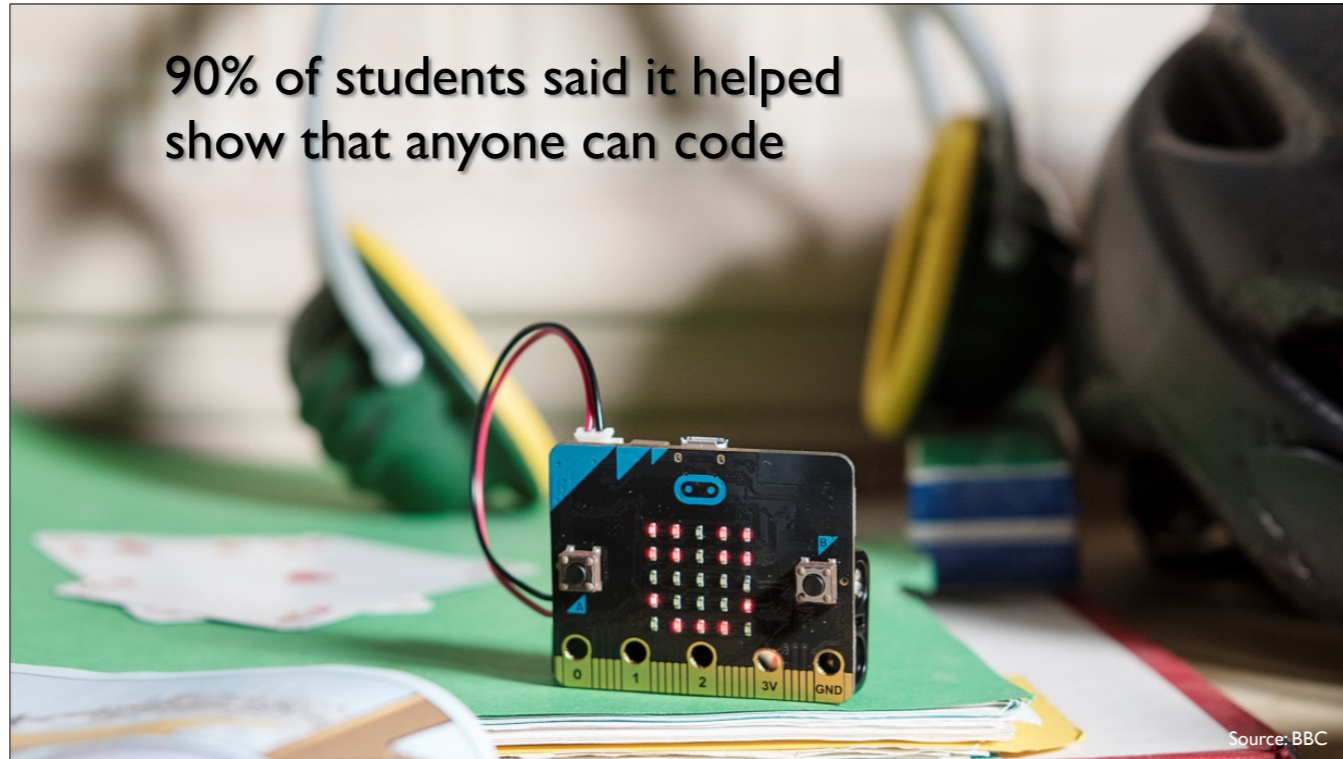
- BBC Make It Digital
- 29 partners
- 1 million micro:bit devices
- 11-12 year olds
- Across the U.K.



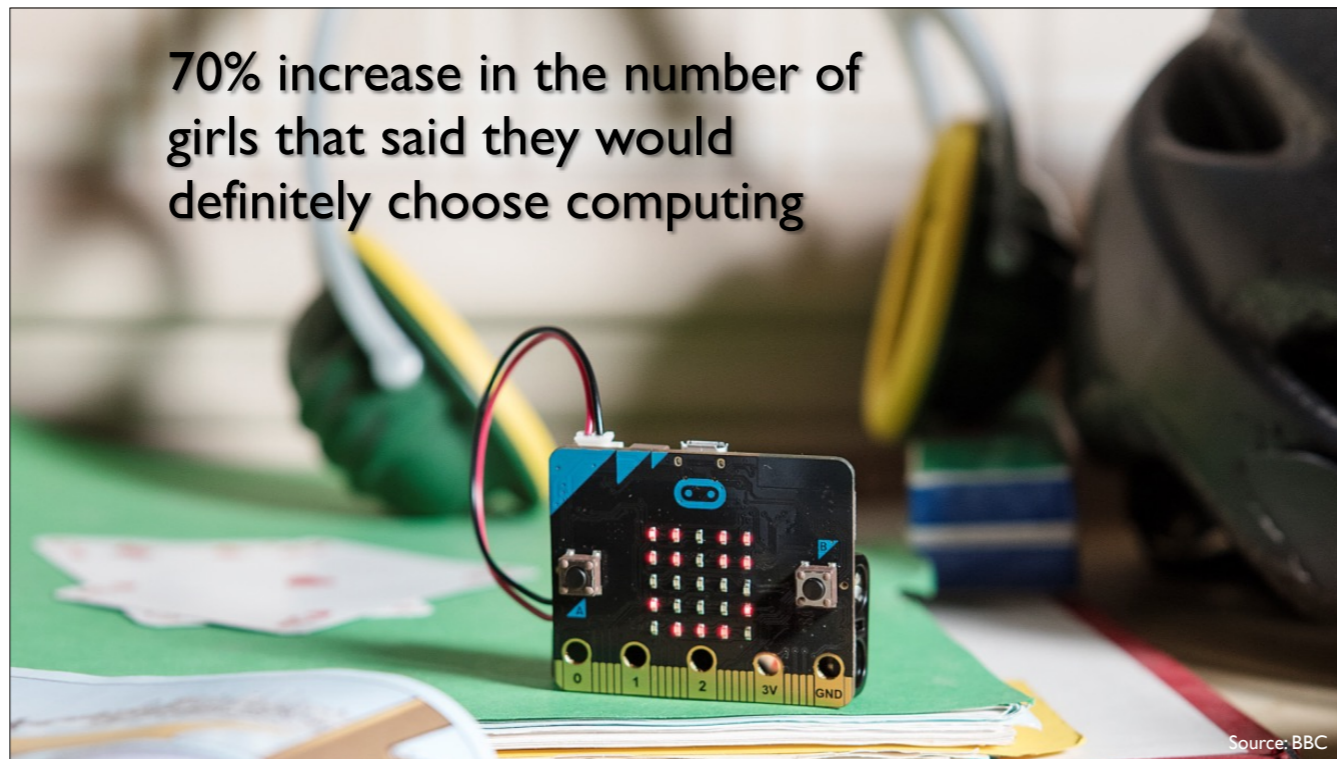
17 © Micro:bit Educational Foundation 2018  
@microbit\_edu @HalSpeed



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				5 <sup>th</sup> Grade			
Course E				Course F			
							
Lesson <b>Course E - Loops 1 - Loops and Animations</b>	Lesson <b>Course E - Loops 2 - Nested Loops and Scoreboards</b>	Lesson <b>Course E - Functions 1 - A Simple Function for a Superhero</b>	Lesson <b>Course E - Functions 2 - Functions for a Digital Pet</b>	Lesson <b>Course F - Variables 1 - Variables With A Counter</b>	Lesson <b>Course F - Variables 2 - Variables and Emotions</b>	Lesson <b>Course F - Conditionals 1 - Conditionals with the Weather Predictor</b>	Lesson <b>Course F - Conditionals 2 - Conditionals with Rock Paper Scissors Game</b>

22 © Micro:bit Educational Foundation 2018  
 @microbit\_edu @HalSpeed <http://microbit.org/teach/code-org-fundamentals/> 

## 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

Environment: Palette, Color & Icon coded; Most used features are prominent, others are on “...more” menu.

Simulator

Block area



## “Hello, World!”: First Program

- Block-based editor

Environment: Palette, Color & Icon coded; Most used features are prominent, others are on “...more” menu.

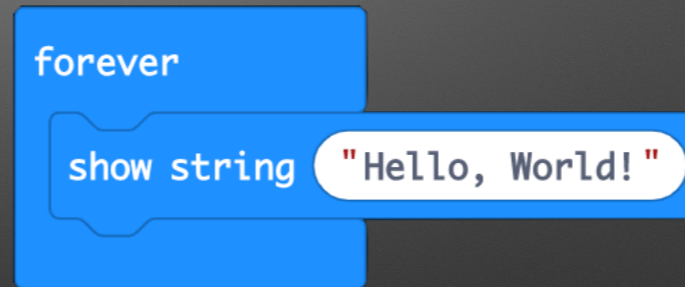
Simulator

Block area



# “Hello, World!”: First Program

- Block-based editor



Environment: Palette, Color & Icon coded; Most used features are prominent, others are on “...more” menu.

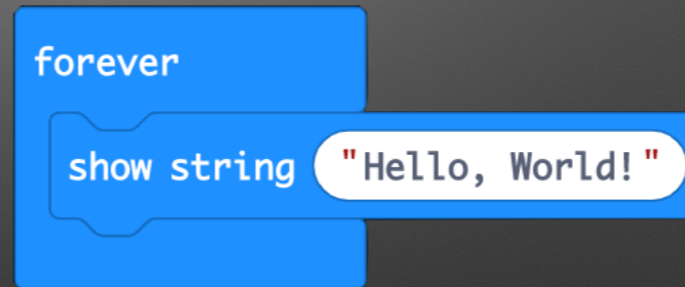
Simulator

Block area



## “Hello, World!”: First Program

- Block-based editor
- Built-in simulator



Environment: Palette, Color & Icon coded; Most used features are prominent, others are on “...more” menu.

Simulator

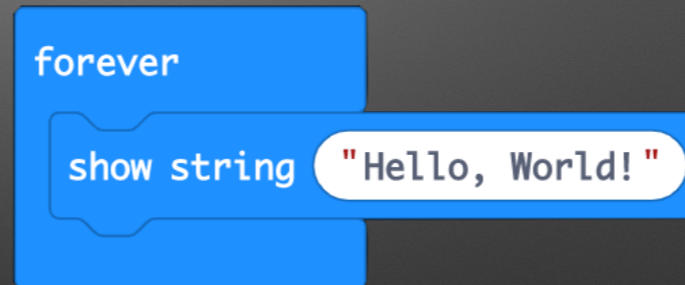
Block area





## “Hello, World!”: First Program

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



Environment: Palette, Color & Icon coded; Most used features are prominent, others are on “...more” menu.

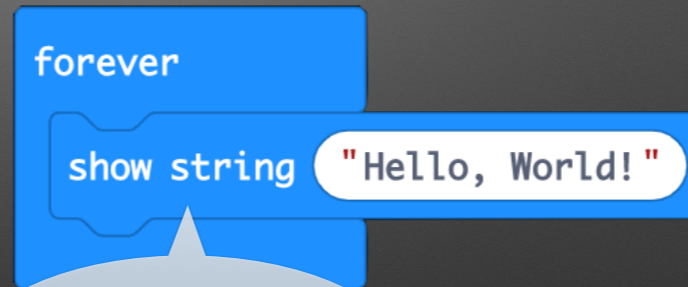
Simulator

Block area



## “Hello, World!”: First Program

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



Block Color Indicates Palette

Environment: Palette, Color & Icon coded; Most used features are prominent, others are on “...more” menu.

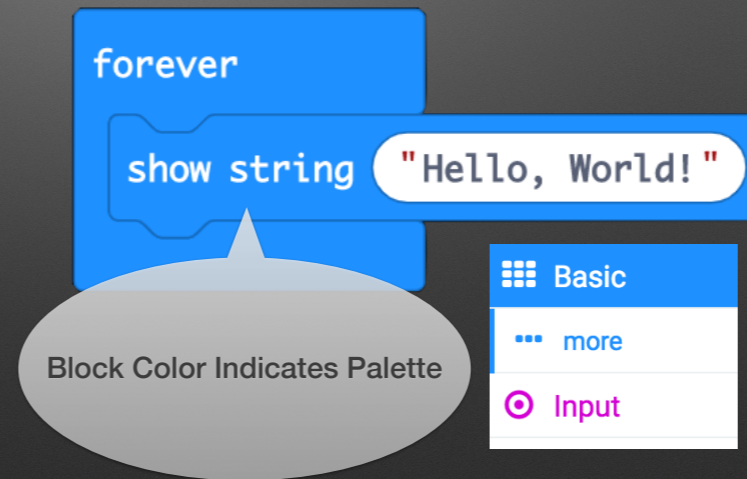
Simulator

Block area



# “Hello, World!”: First Program

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



Environment: Palette, Color & Icon coded; Most used features are prominent, others are on “...more” menu.

Simulator

Block area

**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

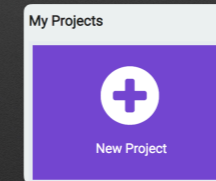
- Hardware Handout
  1. Open Box
  2. Pull out micro:bit
  3. Pull out micro USB cable (under cardboard)
  4. Connect via USB cable



# 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”

A rectangular button with a black border and white text that reads "Let's Code".A rectangular button with an orange background and white text that reads "Let's Code".

**Personalization!**



# Personalization!

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

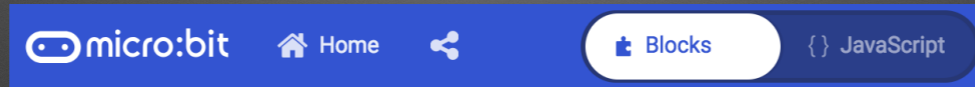
# Personalization!

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

```
forever
```

```
  show string "Hello SIGCSE!"
```

# Aside: Text-based Languages



## 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

A vertical settings menu with a blue header containing a gear icon and the Microsoft logo. The menu items are: Project Settings, Extensions, Delete Project, Report Abuse..., Language, High Contrast On, Reset, Pair device (highlighted), Pair Bluetooth, and About...

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](#)

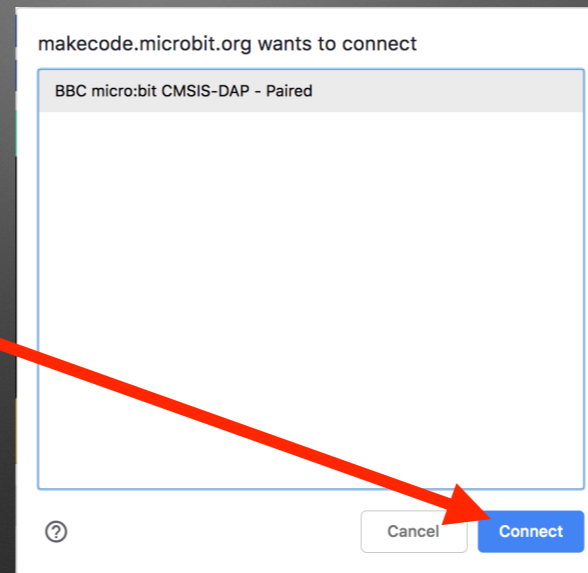
- 1** Connect the micro:bit to your computer with a USB cable  
Use the microUSB port on the top of the micro:bit
- 2** Pair your micro:bit  
Click 'Pair device' below and select BBC micro:bit CMSIS-DAP or DAPLink CMSIS-DAP from the list

Help [Pair device](#)



# Setup: Chrome v65+

4. Connect



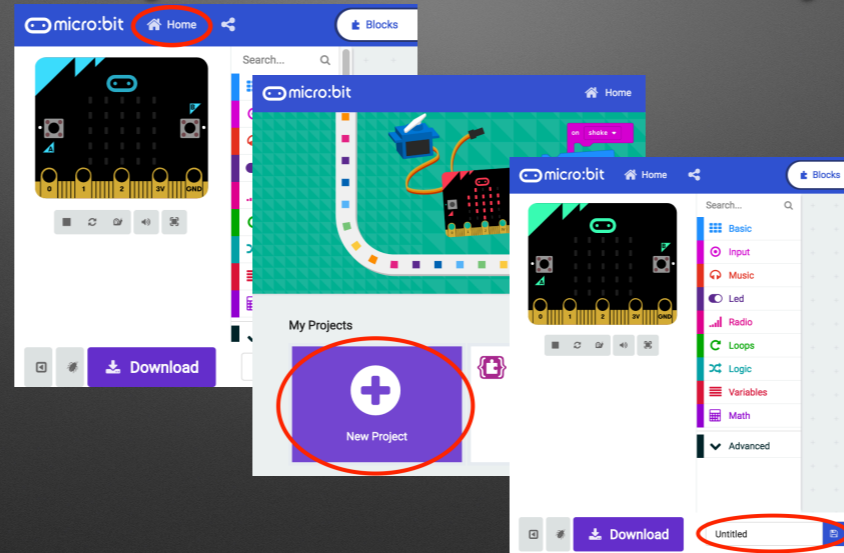


**Try It!**

 [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
  - ...

CS...Int division; Mod; Etc.

**Obvious Solution...**

<https://openclipart.org/detail/17370/a-die>

<https://openclipart.org/detail/19632/scissors>

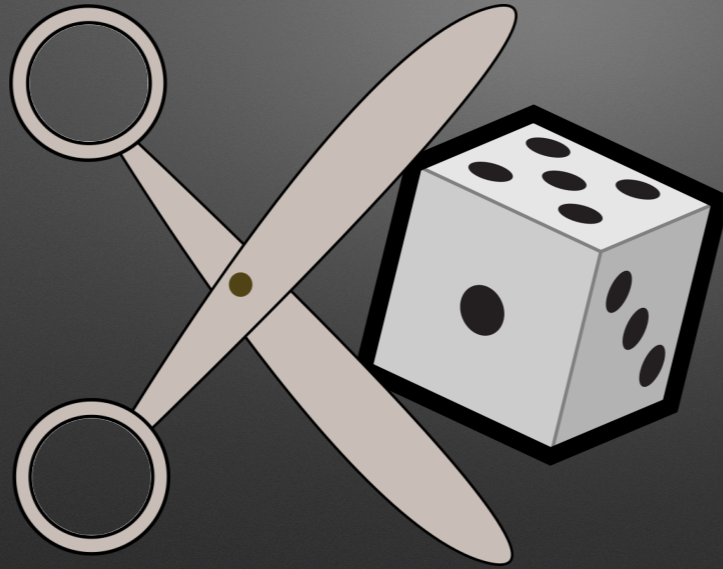
## Obvious Solution...



<https://openclipart.org/detail/17370/a-die>

<https://openclipart.org/detail/19632/scissors>

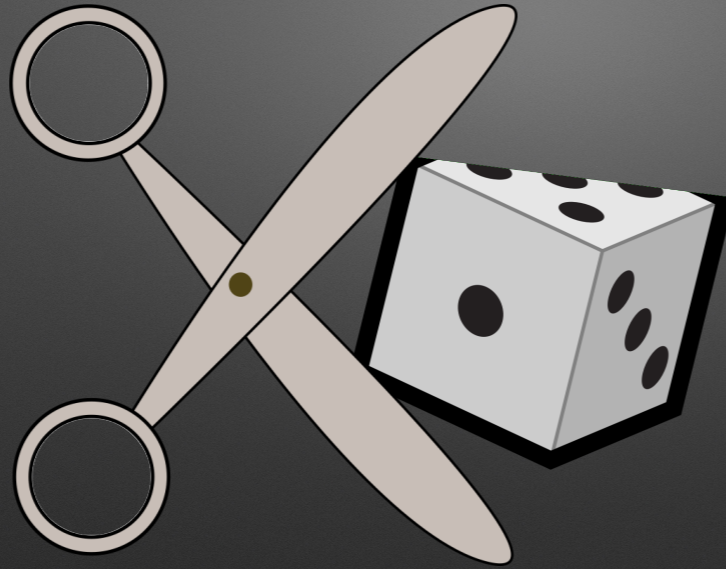
Obvious Solution...



<https://openclipart.org/detail/17370/a-die>

<https://openclipart.org/detail/19632/scissors>

Obvious Solution...

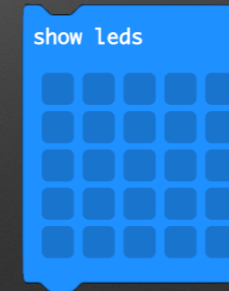
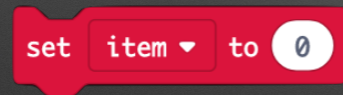


<https://openclipart.org/detail/17370/a-die>

<https://openclipart.org/detail/19632/scissors>



# Parts



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

**Let's play...**

Get started w/ Shake & Show Random Number

# A solution

```
on shake
  set roll to pick random 0 to 2
  if roll = 0 then
    show leds
  else if roll = 2 then
    show leds
  else
    show leds
```

Full Program: 03-Roll.hex

# Concepts

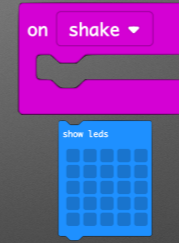
# Concepts

- Event driven programming



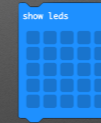
# Concepts

- Event driven programming
- Bitmapped Graphics



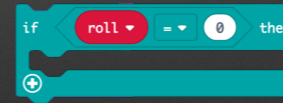
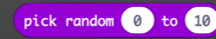
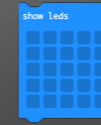
# Concepts

- Event driven programming
- Bitmapped Graphics
- Ranges & Representations



# Concepts

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





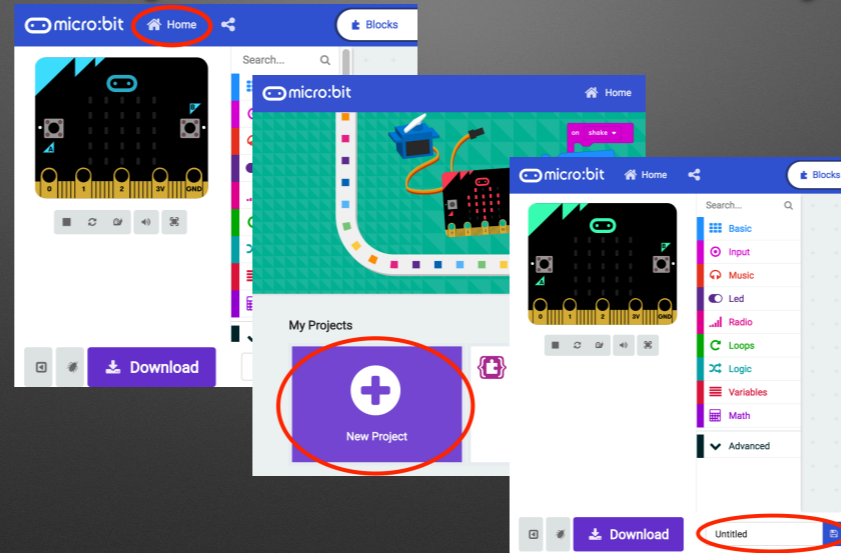
# Pedagogy

- Active Learning
- Discovery Based
- Constructionist

Great...but all concepts can be done with scratch.

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

# New Project: Home > New Project...

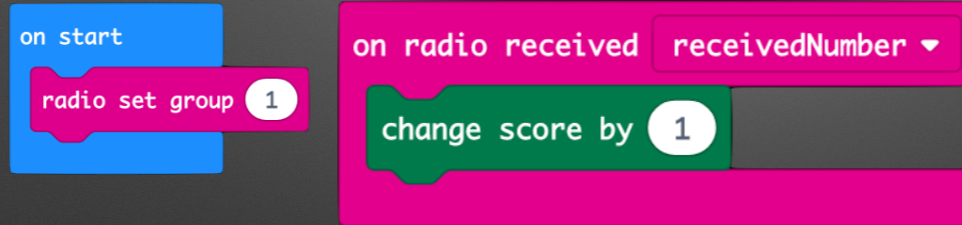




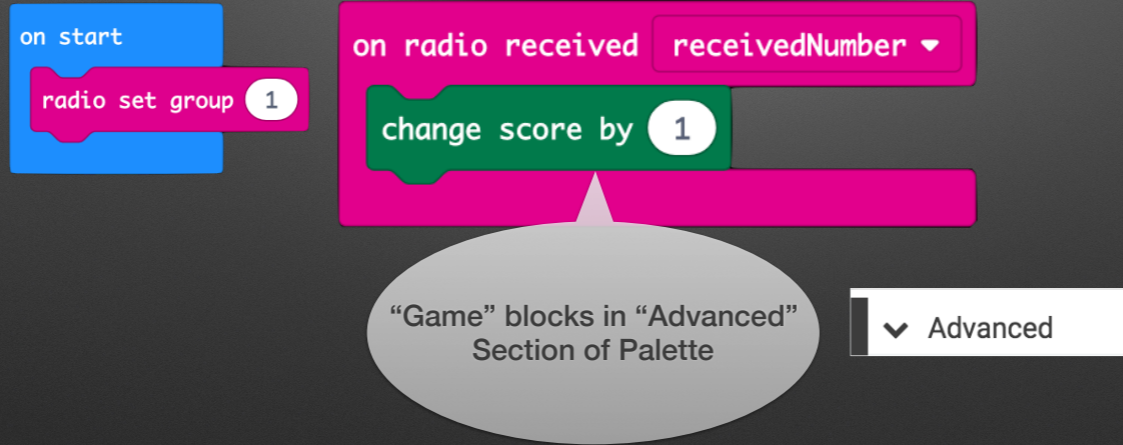
# Broadcast Basics

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

# Receiver



# Receiver



# Full Boadcaster

```
on radio received receivedNumber  
  change score by 1
```

```
on button A pressed  
  radio send number 0
```

```
on start  
  radio set group 1
```

Full Program: 04-FullAutoBroadcaster.hex

# Concepts

Broadcasting can be done with Scratch



# Concepts

- Broadcasting

radio send number 0

Broadcasting can be done with Scratch

# Concepts

- Broadcasting
- Network Addresses

radio send number 0

radio set group 1

Broadcasting can be done with Scratch

# Concepts

- Broadcasting
- Network Addresses
- Asynchronous clocks / Sync problems

radio send number 0

radio set group 1

on radio received receivedNumber ▾

Broadcasting can be done with Scratch



**& Firefly Fun**



# & Firefly Fun

The screenshot shows the micro:bit Home interface. At the top, there is a blue header with the 'micro:bit' logo, a 'Home' button (circled in red), and a 'Blocks' button. Below the header is a search bar and a list of categories: Basic, Input, Music, Led, and Radio. A 'Radio Games' category is highlighted with a red circle. Below this category, four game tiles are displayed: 'Multi Dice', 'Mood Radio', 'Tele-potato', and 'Fireflies'. The 'Fireflies' tile is highlighted with a yellow border and a red circle.

# Concepts

Broadcasting can be done with Scratch

# Concepts

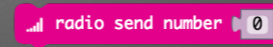
- Broadcasting



Broadcasting can be done with Scratch

# Concepts

- Broadcasting



- Network Addresses

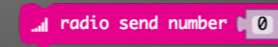


Broadcasting can be done with Scratch



# Concepts

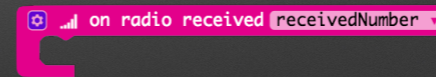
- Broadcasting



- Network Addresses



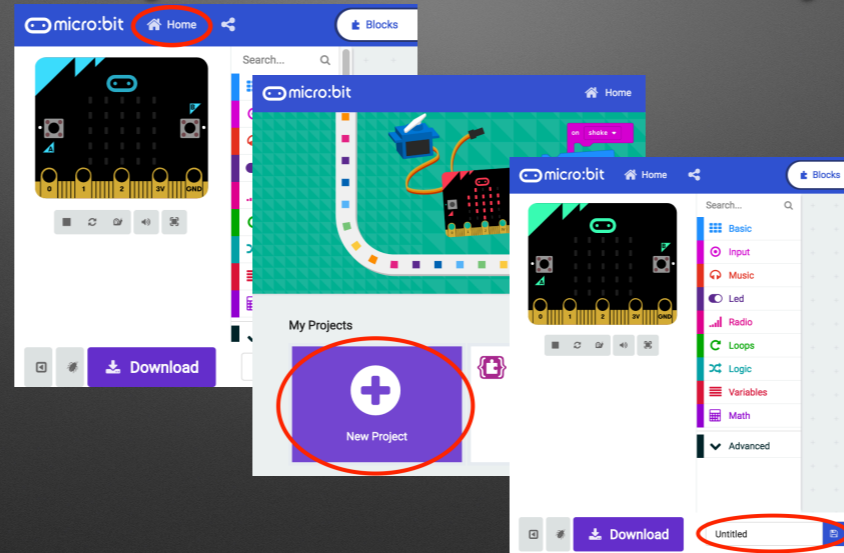
- Asynchronous clocks / Sync problems



Broadcasting can be done with Scratch

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

# New Project: Home > New Project...



# Goody Bag: Hardware



**Awesome (?) Audio**

Show an example of playing a note / Using Clips to connect to headphones

**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 **A** pressed

servo write pin **P0** to **0**

on button **B** pressed

servo write pin **P0** to **120**

Program



# Motor Mayhem

An Intro to Servos

on button **A** pressed

servo write pin **P0** to **0**

on button **B** pressed

servo write pin **P0** to **120**

These servos are limited to 0-120°

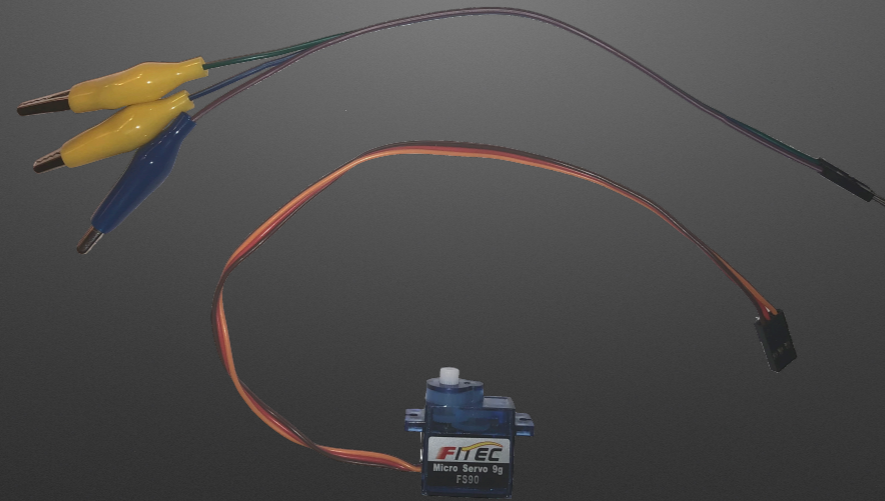
Program

## Testing...

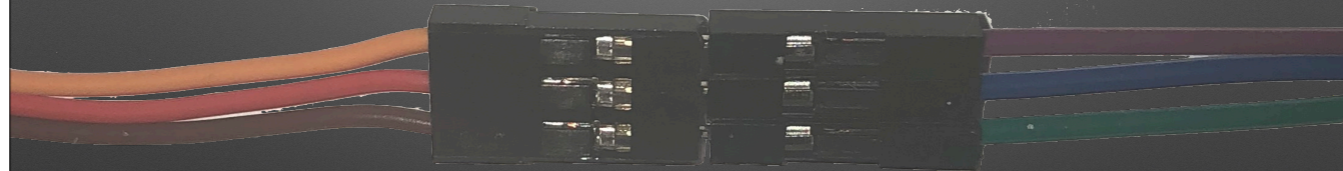
- Test in Simulator

TODO: Add picture

# Parts



Connect them...







## Clip to micro:bit

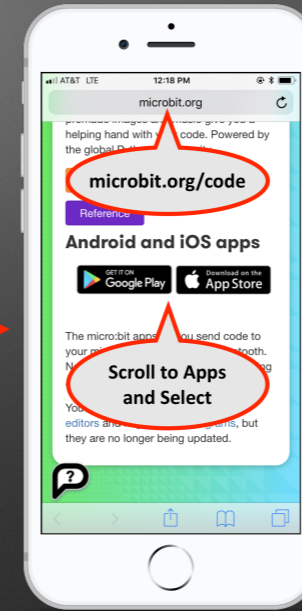
- Match *color on Servo* to pad *name on micro:bit* (clip colors don't matter)
  - Brown on Servo to GND on micro:bit
  - Red on Servo to 3V on micro:bit
  - Orange on Servo to 0 on micro:bit

# 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

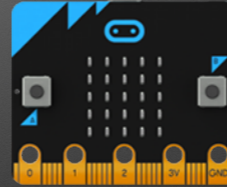


# Bluetooth Background

Central



Peripheral



# Bluetooth Background



# Bluetooth Background

Central



# Bluetooth Background



# 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

Follow instructions to pair. NOTE THE NAME of your Micro:bit!!!! Will need it later

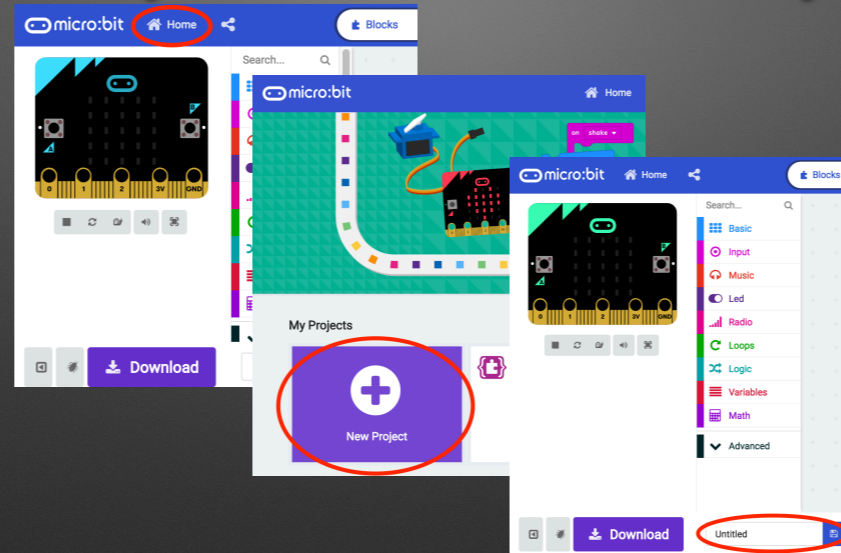
# Pairing



Follow instructions to pair. NOTE THE NAME of your Micro:bit!!!! Will need it later

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

# New Project: Home > New Project...



# Add Bluetooth

(& remove Radio)

FIXME



# Add Bluetooth (& remove Radio)

The screenshot shows the Arduino IDE interface. On the left, the 'Extensions' menu is open, with 'Advanced' and 'Extensions' highlighted by red circles. The main workspace displays the 'Extensions' window, which lists several extensions: 'bluetooth', 'devices', 'radio-broadcast', and 'servo'. The 'devices' extension is highlighted with a red circle. A dialog box titled 'Some extensions will be removed' is open, displaying the message: 'Extension radio is incompatible with devices. Remove radio and add devices?'. The 'Remove extension(s) and add devices' button is highlighted with a red circle.

FIXME

# Project Settings

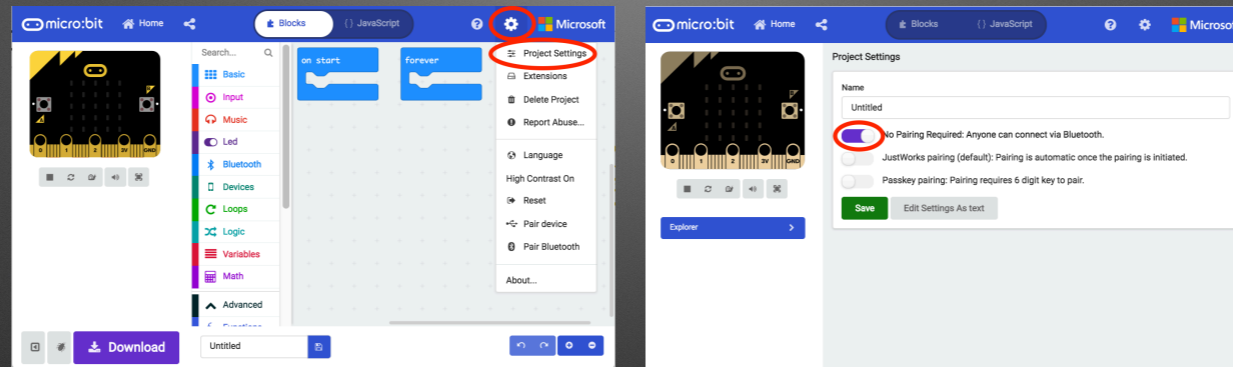
Pairing only works prior to installing a bluetooth sketch.

May need to re-load a blank sketch and then start pairing process.

Each sketch will need this setting.

Need to know name of YOUR microbic

# Project Settings



Pairing only works prior to installing a bluetooth sketch.

May need to re-load a blank sketch and then start pairing process.

Each sketch will need this setting.

Need to know name of YOUR microbic

# Phone Phun: Program

```
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

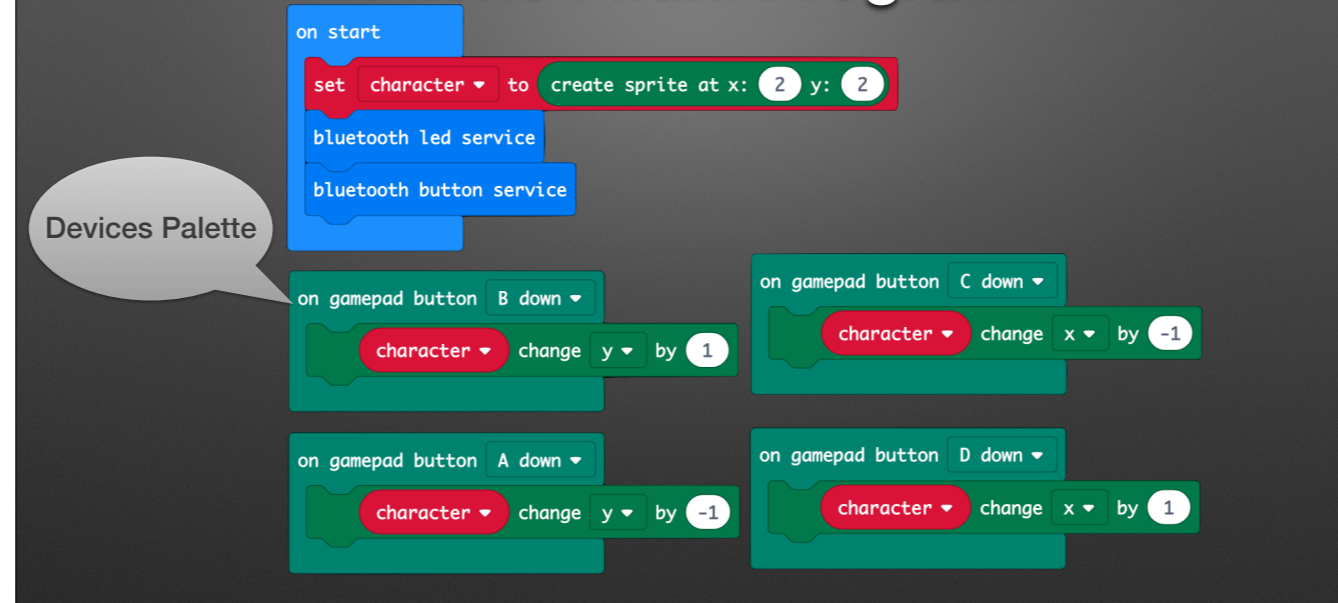
on gamepad button A down
  character change y by -1

on gamepad button D down
  character change x by 1
```

Use right-click "duplicate"

Full Program: 07-BluetoothControl.hex

# Phone Phun: Program



The image shows a Scratch script for a program titled "Phone Phun: Program". The script is organized into several blocks:

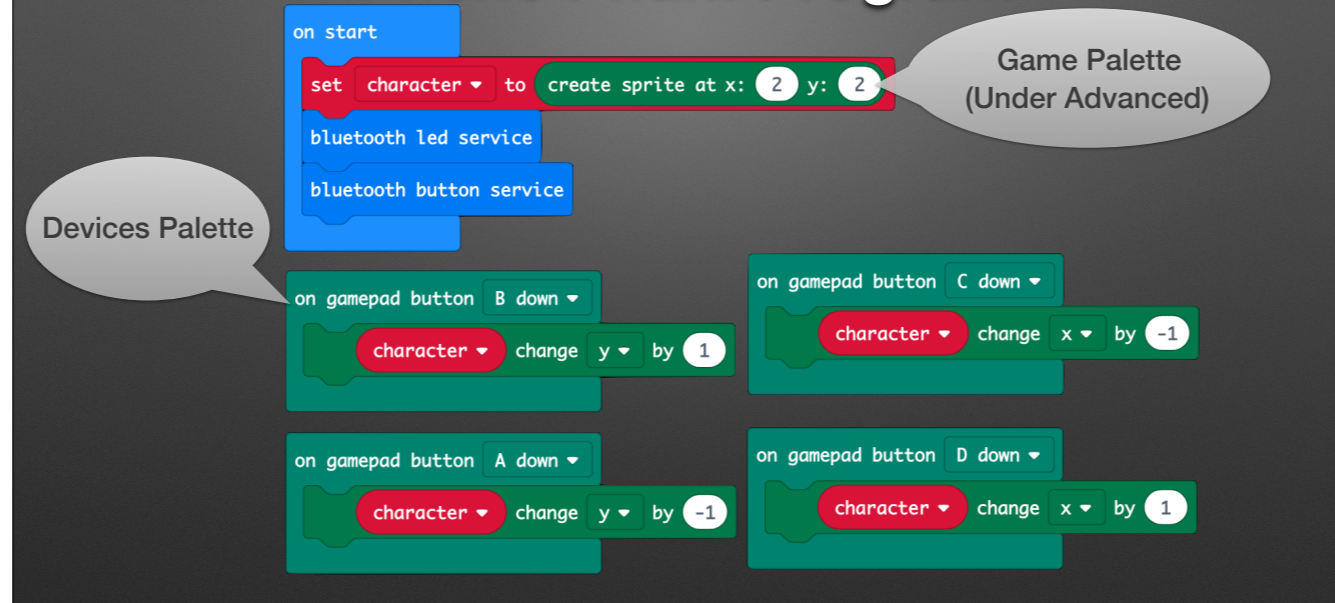
- on start** (blue block):
  - set character to create sprite at x: 2 y: 2** (red block)
  - bluetooth led service** (blue block)
  - bluetooth button service** (blue block)
- on gamepad button B down** (green block):
  - character change y by 1** (green block)
- on gamepad button C down** (green block):
  - character change x by -1** (green block)
- on gamepad button A down** (green block):
  - character change y by -1** (green block)
- on gamepad button D down** (green block):
  - character change x by 1** (green block)

A speech bubble labeled "Devices Palette" points to the "bluetooth led service" and "bluetooth button service" blocks.

Use right-click "duplicate"

Full Program: 07-BluetoothControl.hex

# Phone Phun: Program



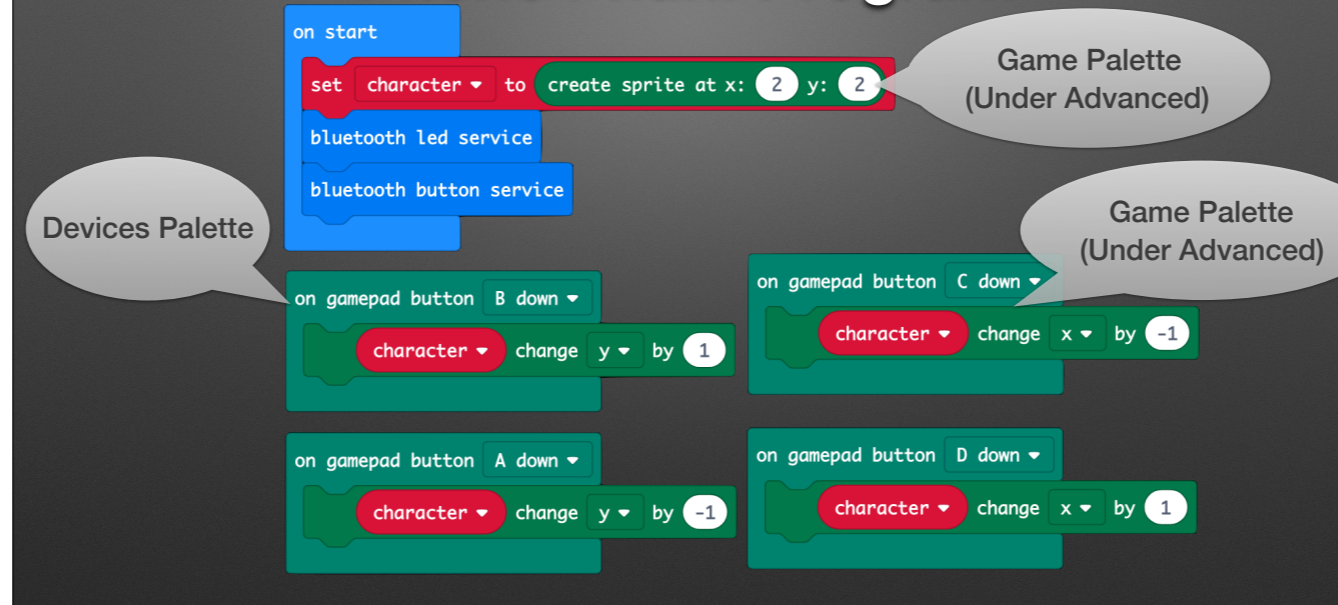
The image shows a Scratch script for a game titled "Phone Phun". The script is organized into several sections:

- on start** (blue block):
  - set **character** to create sprite at x: 2 y: 2 (green block, highlighted with a red border and a callout bubble labeled "Game Palette (Under Advanced)")
  - bluetooth led service (blue block)
  - bluetooth button service (blue block)
- on gamepad button B down** (green block):
  - character change y by 1 (green block, highlighted with a red border and a callout bubble labeled "Devices Palette")
- on gamepad button C down** (green block):
  - character change x by -1 (green block)
- on gamepad button A down** (green block):
  - character change y by -1 (green block)
- on gamepad button D down** (green block):
  - character change x by 1 (green block)

Use right-click "duplicate"

Full Program: 07-BluetoothControl.hex

# Phone Phun: Program



Use right-click "duplicate"

Full Program: 07-BluetoothControl.hex

# Phone Phun: Program

The image shows a Scratch script for a game called "Phone Phun". The script is as follows:

- on start** (blue block):
  - set character to create sprite at x: 2 y: 2 (red block, highlighted with a red border and a callout bubble: "Game Palette (Under Advanced)")
  - bluetooth led service (blue block)
  - bluetooth button service (blue block)
- on gamepad button B down** (green block):
  - character change y by 1 (red block)
- on gamepad button C down** (green block):
  - character change x by -1 (red block)
- on gamepad button A down** (green block):
  - character change y by -1 (red block)
- on gamepad button D down** (green block):
  - character change x by 1 (red block)

Annotations:

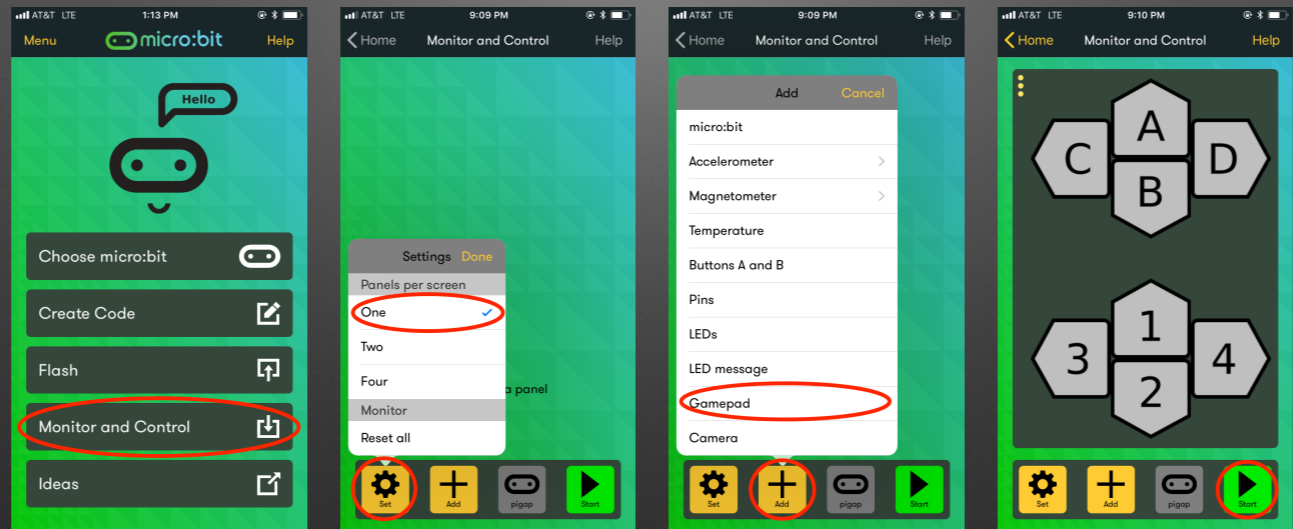
- "Devices Palette" points to the "bluetooth led service" and "bluetooth button service" blocks.
- "Game Palette (Under Advanced)" points to the "set character to create sprite at x: 2 y: 2" block.
- "Game Palette (Under Advanced)" points to the "character change x by -1" block.
- "Pro tip: Create one of these, then right-click and 'duplicate' 3x, then modify" points to the "on gamepad button B down" block.

Use right-click "duplicate"

Full Program: 07-BluetoothControl.hex



# App Configuration



# Micro:bit Shutter Release

# Program

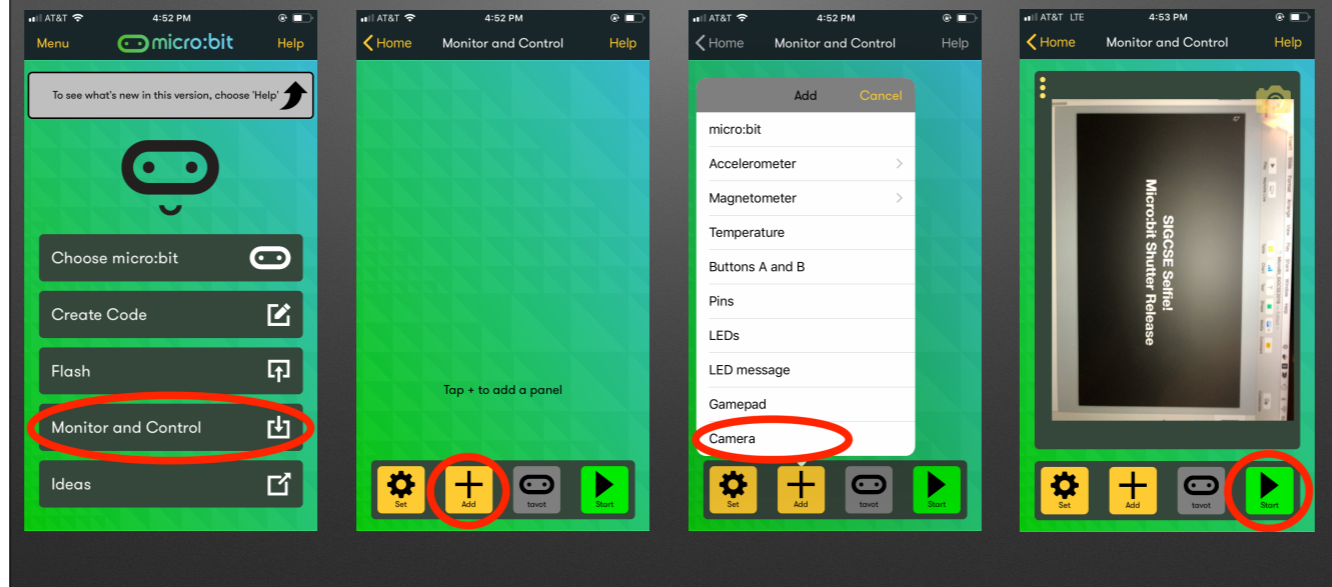
on button  pressed

tell camera to

Full program: 08-Selfie.hex

# App Config

# App Config



**SIGCSE Selfie!**

## Extra Hardware: Extensions

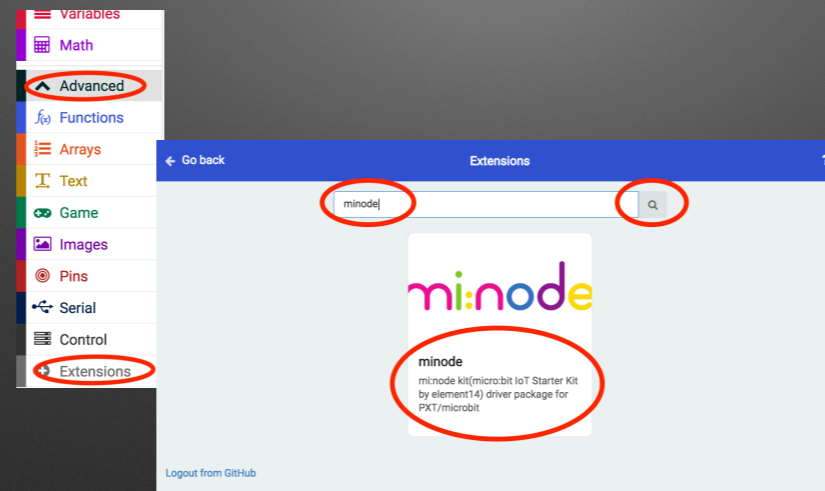
- Extensions...extend
  - Additional hardware support (today)
  - Additional simulator features

## Extra Hardware: Extensions

FIXME

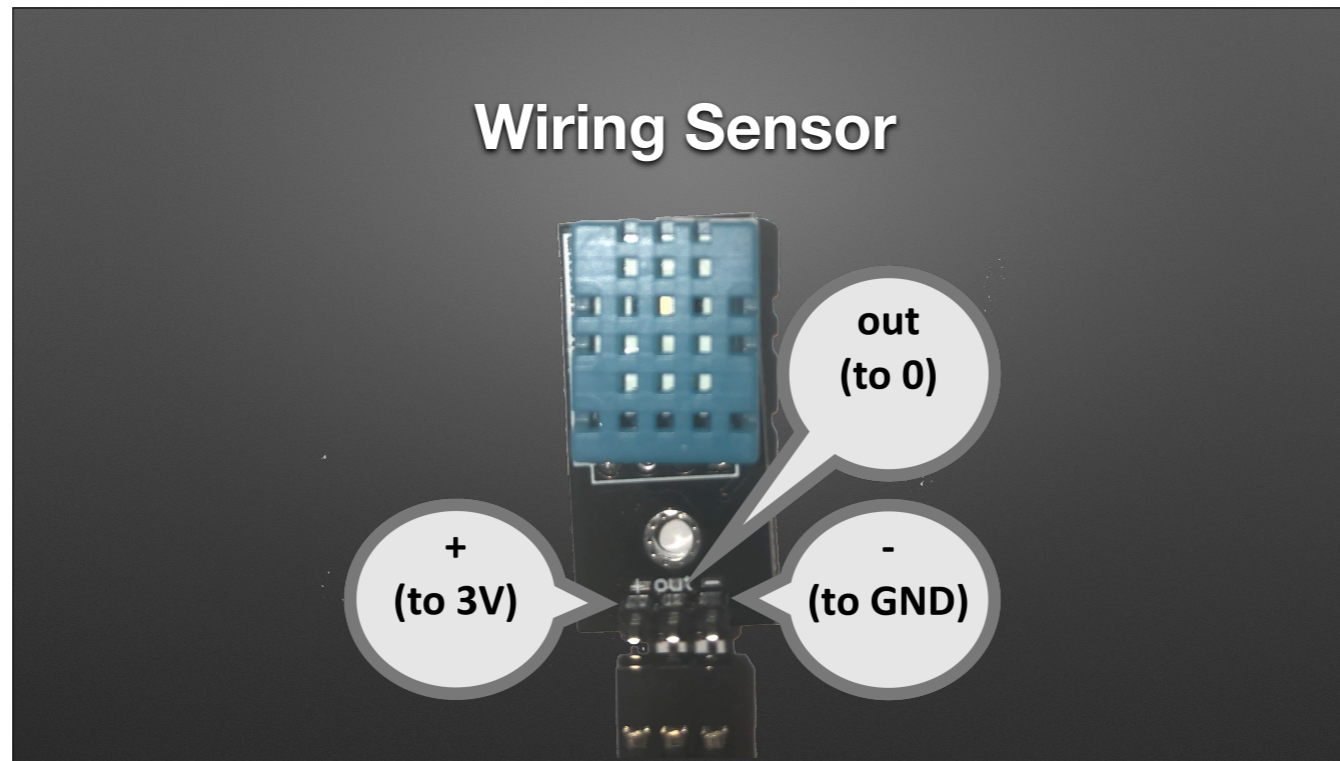


# Extra Hardware: Extensions



FIXME

## Wiring Sensor



TODO

# Collecting Data

```
forever
  serial write value "Temp" = dht11 A0 temperature Celsius
  serial write value "Humidity" = dht11 A0 humidity
```

Full Program: 09-TempHumidity.hex

# Collecting Data

forever

serial write value "Temp" = dht11 A0 temperature Celsius

serial write value "Humidity" = dht11 A0 humidity

Minode Palette

Full Program: 09-TempHumidity.hex

# Collecting Data

forever

serial write value "Temp" = dht11 A0 temperature Celsius

serial write value "Humidity" = dht11 A0 humidity

Minode Palette

Minode's  
"...more" Palette

Full Program: 09-TempHumidity.hex

# Collecting Data

forever

serial write value "Temp" = dht11 A0 temperature Celsius

serial write value "Humidity" = dht11 A0 humidity

Minode Palette

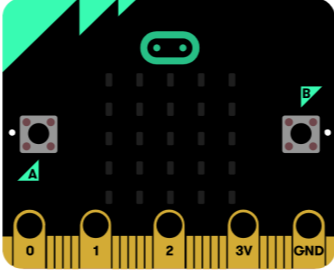
Advanced  
Serial Palette

Minode's  
"...more" Palette

Full Program: 09-TempHumidity.hex

# Graphing

micro:bit Home Share



A simulator of a micro:bit board. The board is black with a green top-left corner. It features a grid of pins, two sensors (A and B), and a central LED display. Below the board, there are five pins labeled 0, 1, 2, 3V, and GND. The board is connected to a graphing interface.

0 1 2 3V GND

Show console Simulator

Show console Device

## Bluetooth Streaming: Setup



# Bluetooth Streaming: Setup

The screenshot displays the Microsoft MakeCode IDE interface. On the left, a settings menu is open with the 'About...' option circled in red. The 'About' dialog box is open, showing version information for makecode.microbit.org (1.2.13), Microsoft MakeCode (4.4.7), and microbit runtime (v2.1.1). The 'Experiments' button at the bottom of the dialog is also circled in red. In the background, the 'Bluetooth Console' extension is visible, with its 'Enabled' toggle and a description: 'Receives UART message through Web Bluetooth'. The extension's interface includes a waveform visualization and a 'Feedback' link.

# 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
```

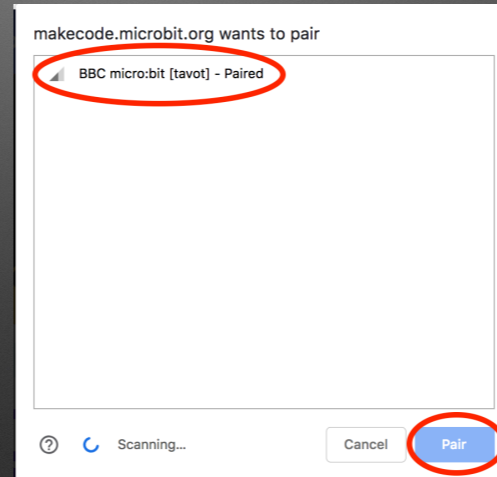
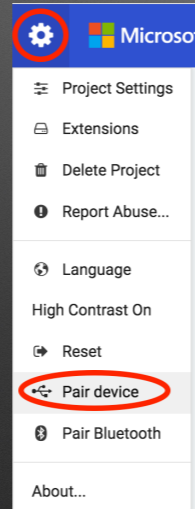
Full Program: 12-WirelessAccel.hex

# 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)

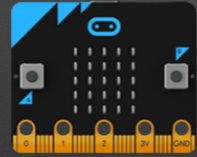


# IoT Example Overview

AdaFruit.io

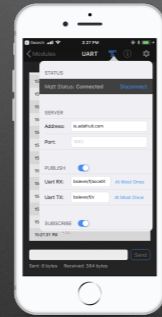
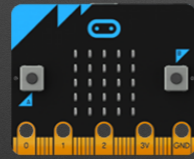
<https://cdn-learn.adafruit.com/guides/images/000/001/691/medium800/Adafruit-IO-Logo.png>

AdaFruit.io



<https://cdn-learn.adafruit.com/guides/images/000/001/691/medium800/Adafruit-IO-Logo.png>

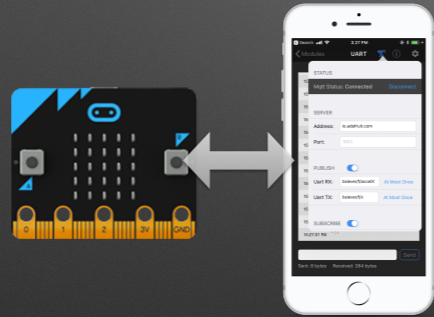
# AdaFruit.io



<https://cdn-learn.adafruit.com/guides/images/000/001/691/medium800/Adafruit-IO-Logo.png>

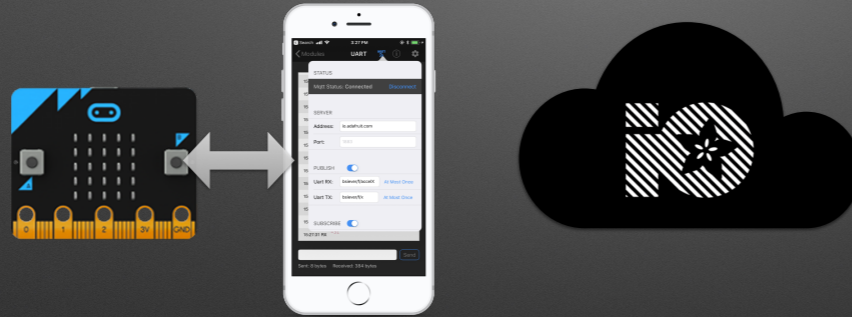


# AdaFruit.io



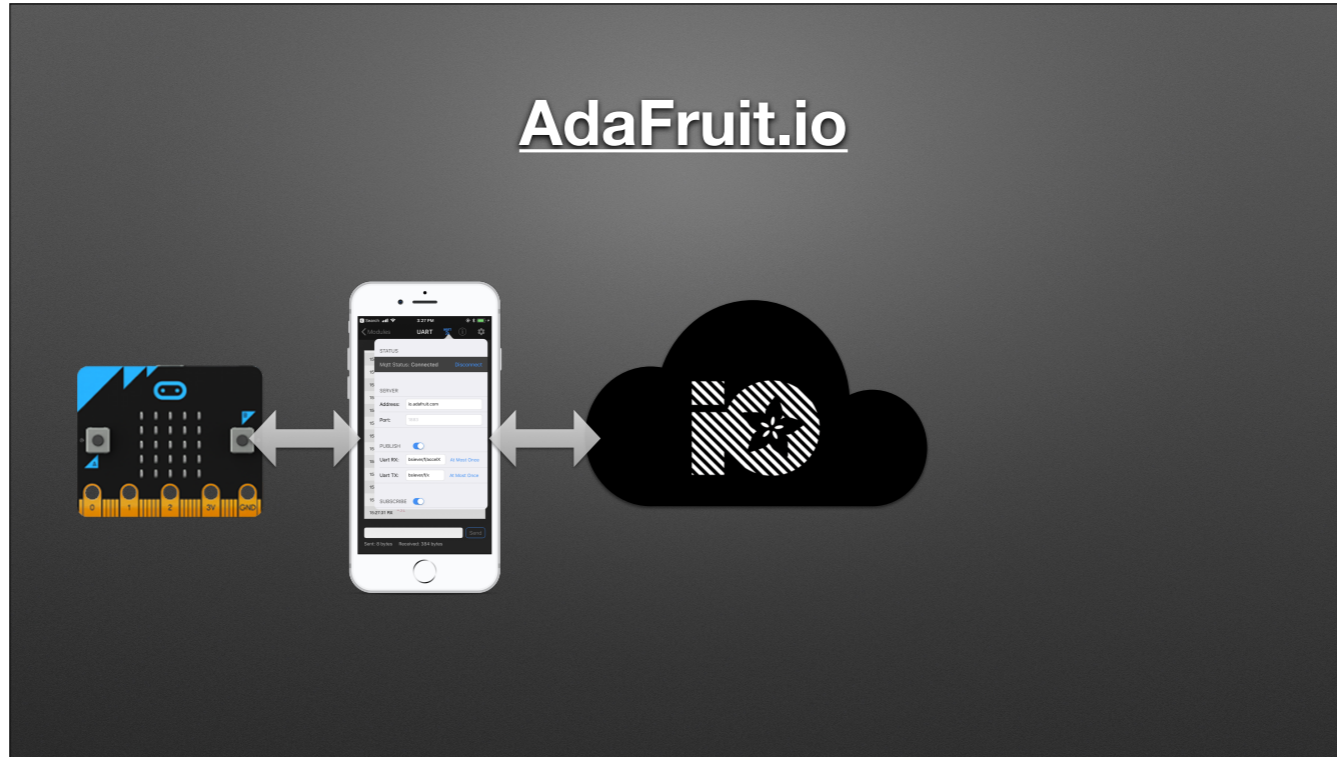
<https://cdn-learn.adafruit.com/guides/images/000/001/691/medium800/Adafruit-IO-Logo.png>

AdaFruit.io

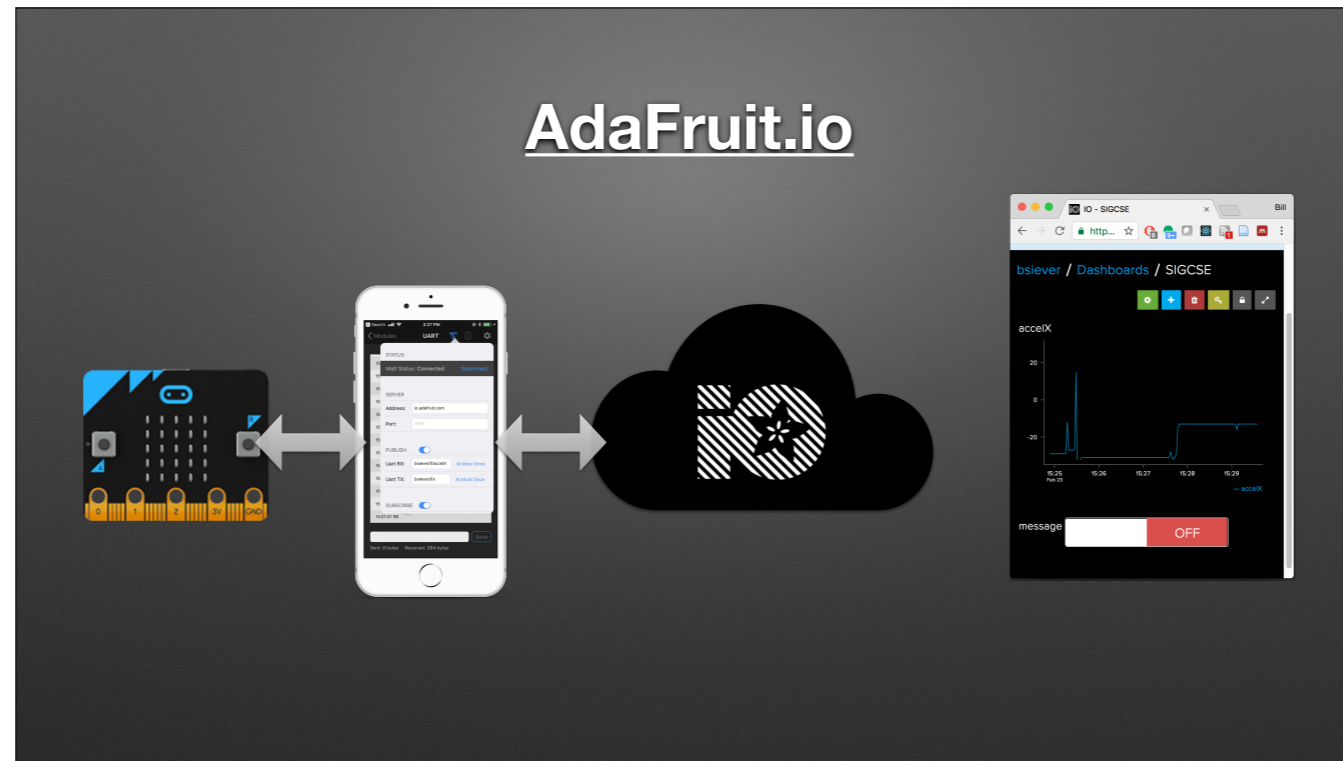


<https://cdn-learn.adafruit.com/guides/images/000/001/691/medium800/Adafruit-IO-Logo.png>

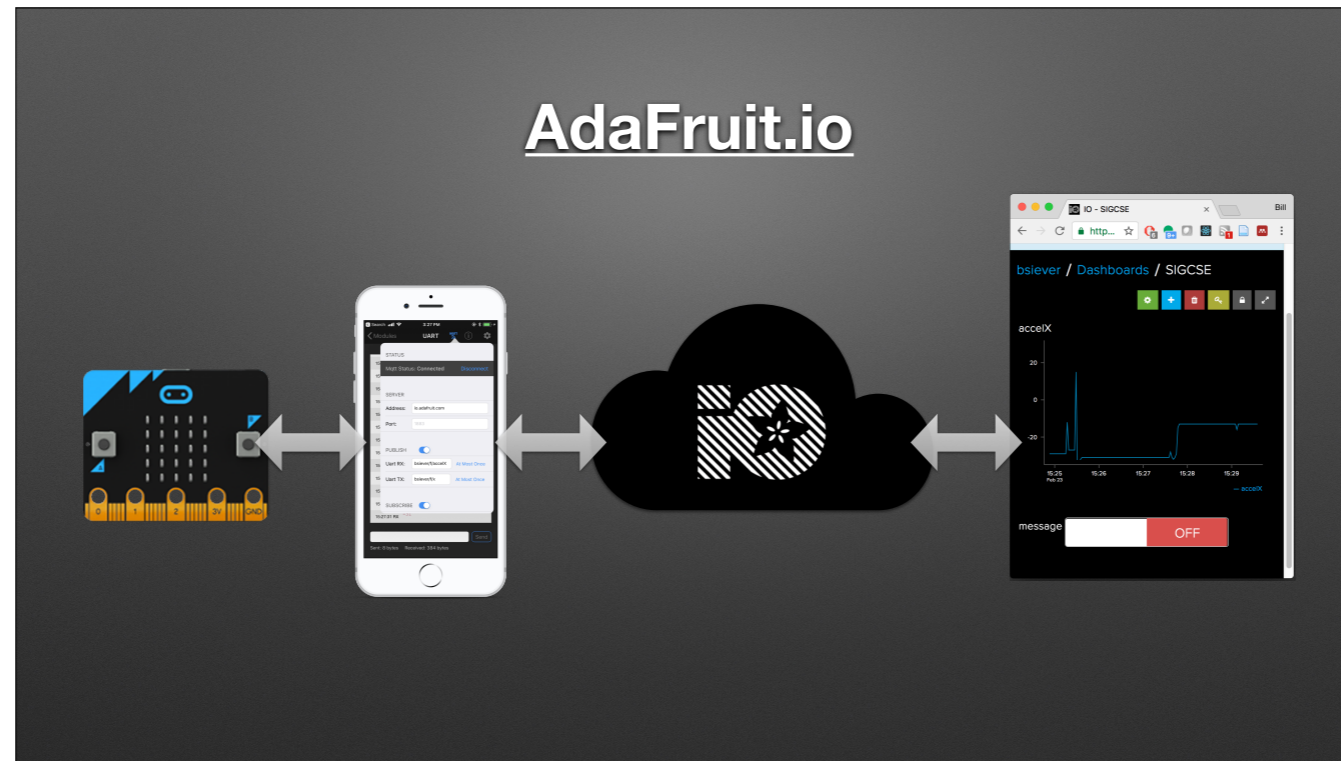
AdaFruit.io



<https://cdn-learn.adafruit.com/guides/images/000/001/691/medium800/Adafruit-IO-Logo.png>



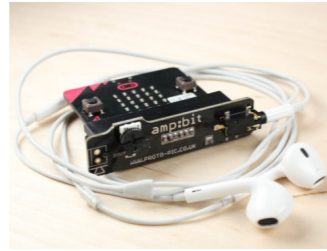
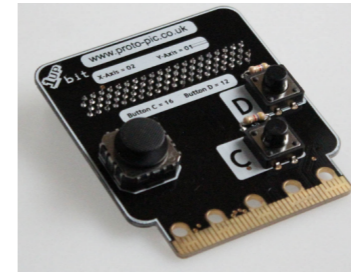
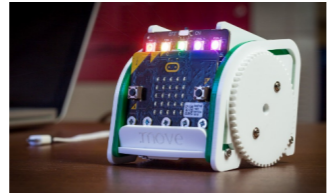
<https://cdn-learn.adafruit.com/guides/images/000/001/691/medium800/Adafruit-IO-Logo.png>



<https://cdn-learn.adafruit.com/guides/images/000/001/691/medium800/Adafruit-IO-Logo.png>

# Hardware Ecosystem







## 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)

