Micro:bit Magic

Engaging Computer Science Activities for Grades 5-12 Sponsored by Washington University's Institute for School Partnership

> Bill Siever Principal Lecturer Washington University in St. Louis



Me: Background (degrees & career training) Computer Science & Computer Engineering. I'm a Lecturer here, so my main roll is teaching (but I don't have a formal background in education nor do I have deep experience with K-12)

Lunch!



- Us - You		Intros: Us & You	
- You	• Us		
	• You		



You: Stand If You Teach	
AP C.S. Principles	
Arduino	
 Raspberry Pi 	









5cm x 4cm Artwork source: <u>http://microbit.org/images/microbit-features-temp.png</u>



5xm x 4cm



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



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



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



Within about 2 degrees C (die temperature)/3.6 degrees F. Artwork source: <u>http://microbit.org/images/microbit-features-temp.png</u>



Detect/respond to tilt/tip/shake/etc. Artwork source: http://microbit.org/images/microbit-featuresaccelerometer.png



Artwork source:<u>http://microbit.org/images/microbit-features-compass.png</u> (ugh. A little awkward to use)



"Broadcast"



Bluetooth: It can talk to mobile devices!!! (Mutually exclusive with broadcast) Artwork source: <u>http://microbit.org/images/microbit-features-bluetooth.png</u>



Currently \$9 at micro:center here in town. Artwork source: <u>http://microbit.org/images/microbit-features-temp.png</u>



Thanks to The Micro:bit Educational Foundation and Hal Speed for the following 9 slides. (Hal is Chief of Global Engagement; Micro:bit foundation is a non-profit)





Empowering students / enabling diversity in STEM (hopefully)



2016 Micro:bit Educational Foundation Formed	
To empower children, par- teachers around the globe and innovate using the m	ents and to learn hicro:bit
26 © Monthi Educational Foundation 2018 @innorchet_edu @Heldipaud	micro:bit









Third-Party C	urricula	
{	}	
Microsoft MakeC https://aka.	Code Intro to CS ms/intro2cs	PLTW Gateway: Computer Science for
1. Making 2. Algorithms 3. Variables 4. Conditionals 5. Iteration 6. Review/Mini-Project	 Coordinate Grid System Booleans Music and Arrays Iits, Bytes, and Binary Radio Arrays 	https://www.pltw.org/our-programs/ https://www.pltw.org/our-programs/ pltw-gateway- curriculum#curriculum-4
7. Coordinate Grid System	14. Independent Final Project	
@microbit_edu @HalSpeed		micro:bit



I was working with the processor used by the micro:bit well before the micro:bit existed. It's a prominent processor used in many Bluetooth gizmos, like some FitBits. I have a lot of experience developing bluetooth devices/ software/apps.

Campus Network SSID: wustl-guest-2.0

"Hello, W	/orld!": First Pro	gram
	Icon Indicates Palett	e
Block-based editor Built-in simulator Deployment to Micro-bit	<pre>forever forever find the show string the</pre>	ello, World! "
	Block Color Indicates Palette	Search Q III Basic III More O Input



Hardware Setup & Programming

Hardware Handout - Thanks to Wash U's Institute for School Partnership!

- Pull out the micro:bit box / open
- 2. Connect via USB cable
- Find browser Download Location
- Micro:bit acts like a flash drive. It's programmed by dropping files on it

Chrome Setup	
 chrome://settings/downloads OR Show advanced settings. 	
Downloads	
Location CHANGE C\Users\Hai\Downloads CHANGE	
Ask where to save each file before downloading	
Downloads	-
Download location: C:\Users\Hal\Downloads Change	
Ask were to save each file before downloading	
37 © Microbit Educational Foundation 2018	••
@microbit_edu @HalSpeed	micro:bit



Build a Marquee / Name Badge / Etc.!

Re-program the micro:bit to scroll your name

Changing code in simulator does not change real micro:bit

Need to drag/drop each update

Real micro:bit retains program until re-programmed

Try the battery pack!



Name it "PickOne" (or something else)





https://openclipart.org/detail/17370/a-die https://openclipart.org/detail/19632/scissors





1. Test the "shake" construct (in the simulator)



Notice the simulator now has a "shake" button to simulate shaking



2. Try to understand variable / random number



2. Try to understand variable / random number



Use "Turtle mode" to better follow execution. Notice how blocks are highlighted as they execute. In particular, notice the behavior of the "body" of the if-statement (when/if the "show LEDs" block is highlighted)



2. Try to understand variable / random number



2. Try to understand variable / random number









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







Notice the history. They are tied to the browser.



Broadcast Basics

Radio Palette: Broadcast Based Radio Transmissions

String, Number, Key/Value Pairs, ...





Program micro:bit









Broadcasting can be done with Scratch











Concepts 1/0 Basic Electric Circuits/Electronics



Artwork: <u>https://mockuphone.com</u> and <u>microbit.org</u> site Prep the Pick One Application

Beyond Blocks: Bootstrapping Text-based languages

The editor translates all the code to static TypeScript

TypeScript is superset of JavaScript

Gets translated/compiled in the browser to machine code

Easy to show the correlation from blocks to structured text!













Bluetooth CAN do a type of broadcast, but that's not how the micro:bit uses bluetooth.







The data being exchanged is usually "low risk"



Follow instructions to pair.

But you said we wouldn't use pairing...

- Pairing stores info. on *both* the micro:bit and the phone
- This info. is created when they first pair and is unique
- It must match for the devices to be able to connect securely
- Reprogramming the micro:bit erases its pairing information
- But the Phone still retains its pairing information
- The pairing details will be out-of-sync & they will be unable to connect

Micro:bit app (iOS)

- The only way for the app to identify a micro:bit by going through the pairing process
- The app retains the identify of the micro:bit even if the micro:bit no longer needs to be paired
- I.e., the "pairing" was just done to let the App know about the micro:bit

But you said we wouldn't use pairing...

- Every time you re-program the micro:bit it erases its pairing information <u>but</u> your phone does <u>not</u> forget the pair.
- Option A: Re-pair every time you re-program:
- Open bluetooth settings on phone and "Forget" the micro:bit
- 2. Use the App to pair again

But you said we wouldn't use pairing...

- Every time you re-program the micro:bit it erases its pairing information <u>but</u> your phone does <u>not</u> forget the pair.
- Option B (today):
 - Pair once so the app identifies the micro:bit
- 2. Disable pairing on the micro:bit

Misc. on Pairing & Security

Opportunity to discuss security issues & concepts

The previous example was done on the official micro:bit app.

Other apps may or may not require pairing (many don't need to be "tricked" by pairing once)









Pairing only works prior to installing a bluetooth sketch. May need to re-load a blank sketch and then start pairing process.





Apps for Data Logging: bittydatalogger

(This app doesn't require the "pairing once" trick)



Demo App / Collecting Data Applications: Citizen Science (measuring vibration on a bridge...Frisbee being thrown...etc.

Concepts
Data Formats (CSV vs. JSON) Data Analysis

Can also be used with Mobile Development

Simple data logger program (on the micro:bit)

Provides a standard "service"

Very easy to write iOS and Android apps that access the data

Play Time!

Peer-up

- Form groups of 2-4 based on a common bond. Either:
- Students are near the same ages
- Teaching the same topics
- Same level of comfort with everything today

Challenge: Lesson-plan-athon!

- Inspiration: Hackathons & Programming Competitions
- Review and try resources
- I'll walk around to help where/how I can
- Time will be pseudo-structured / timed
- You'll (informally) report-out at the end to share your findings

Part 1: Finding and Prioritizing Potential Activities Part 2-3: Play! Try them out! (and/or Create) Try 2+ (and then, optionally, brainstorm and create new activity)

> Part 4: Reprioritize & <u>Report</u> <u>http://bit.ly/microbitreport</u>

http://bit.ly/washumicrobit (microbit report)

Report Out!

Question: What about more advanced topics?

IoT Example



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



Demo https://io.adafruit.com/bsiever/dashboards/sigcse

Other Hardware











Odds & Ends: Weirdness & Weaknesses

Currently no floating point numbers

Great for practicing Integer concepts

Not a limitation of the board; Expected to be included in next major update

Support for Functions is weak using blocks

Motivates JavaScript (or Python)

Odds & Ends: Micro:bit vs. ...

- Scratch: Probably a better choice for block-based intro to programming
- But micro:bit can extend that to hardware & the physical world
- Raspberry Pi is a much more complex computer w/ lots of software choices; They are good for things like webcams
- But micro:bit is good for somesimple motors/motion/communication examples
- Lego: Expensive, but great at "mechanical" things

Questions / Discussion

Thanks 1: Thanks for attending!

Thanks 2: Thanks ISP! Special thanks to Washington University's Institute for School Partnership (https://schoolpartnership.wustl.edu/) Jaime Gilligan & Chris Mohr All arrangements

Supplies (Food! Swag bags!)

Bill's Blog Post http://bit.ly/washumicrobit (WashU Micro:bit)

Eval form (to compete certificate)

