Using the BBC micro:bit to Teach STEM

History & Future of the micro:bit project

Jonathan Austin, Chief Technology Officer
What we’re going to cover
...for a mixed audience

• Some history
• What is the BBC micro:bit?
• Who are the Micro:bit Educational Foundation?
• How are people teaching with the micro:bit?
• Introduction to BBC micro:bit V2
• What’s next?
The Story
1981
BBC Micro
**BBC micro:bit**

Versatile, programmable IoT device that is designed for use in the classroom

Available in 70 countries for ~£10 in volume

- Easy to program, no installation or drivers
- Simple wireless communication
- Battery powered
- Built-in motion sensors, instantly interactive
- Hundreds of sensors and accessories

*Low floor, high ceiling*

*Designed by a coalition of 29 partners!*
Push buttons
External Power supply
3.3V in or battery V out

Display
5x5 LED Matrix

Touch and Input/Output Pins
(for connecting other components and sensors)

Holes for banana plugs
Pads for crocodile clips

“Edge Connector”
Micro USB
Power and Programming

Antenna (Bluetooth & RADIO)

Main Processor
Nordic nRF51822

Magnetometer

Accelerometer

Battery connector
(2xAAA)

Reset Button

Interface Processor
Provides drag-and-drop
Edge Connector

- Innovative
- Robust
- Adaptable
- Easy for kids to use
- Easy to adopt in hardware
- No fragile pins to bend
- No soldering

Enables a huge ecosystem of accessories and extensions around the core micro:bit platform
Edge Connector

- Innovative
- Robust
- Adaptable
- Easy for kids to use
- Easy to adopt in hardware
- No fragile pins to bend
- No soldering

Enables a huge ecosystem of accessories and extensions around the core micro:bit platform
micro:bit ‘is a USB Drive’
micro:bit is a USB Drive
**MakeCode Editor**

The MakeCode editor provided by Microsoft makes it easy to program your micro:bit with blocks and JavaScript.

We have recently updated the editor, and the previous version is still available for anyone that needs it. If you have any issues accessing the editor, check that it isn't blocked in your school.

- Let's Code
- Reference
- Lessons

**Python Editor**

Our Python editor is perfect for those who want to push their coding skills further. A selection of snippets and a range of premade images and music give you a helping hand with your code. Powered by the global Python Community.

- Let's Code
- Reference
200 accessories
200 accessories

16 Third party editors
200 accessories

16 Third party editors

1000 Testers on the mailing list
200 accessories

16 Third party editors

1000 Testers on the mailing list

800 Developers on Slack
200 accessories

16 Third party editors

1000 Testers on the mailing list

800 Developers on Slack

15 (?) Student projects
🎁 200 accessories
📝 16 Third party editors
🔬 1000 Testers on the mailing list
🛠 800 Developers on Slack
🎓 15 (?) Student projects
💬 571 translators
🎁 200 accessories
📝 16 Third party editors
딥런 1000 Testers on the mailing list
🛠 800 Developers on Slack
🎓 15 (?) Student projects
💬 571 translators

✅ 150 approved Microsoft MakeCode extensions
The Story
Demonstration
The Story
2015

BBC

micro:bit
Inspiring every child to create their best digital future
Mission

We enable and inspire all children to participate in the digital world, with particular focus on girls and those from disadvantaged groups.

We make micro:bit the easiest and most effective learning tool for digital skills and creativity.

We work in collaboration with educators to create and curate exceptional curriculum materials, training programmes and resources.

We build and support communities of educators and partners to remove the barriers to learning digital skills.
Make it: code it

Quick projects to suit all ages, searchable by computing topic, level, coding language and micro:bit feature

Search

Programming language

- MakeCode (71)
- Python (65)
- Scratch (6)

Subjects & topics

- Computational thinking
- Computer systems
- Digital arts

Heart
Light up your micro:bit with love...
Beginner

Beating heart
Make your micro:bit’s heart...
Beginner

Animated animals
Animate your own animals on the...
Beginner

Emotion badge
Use your micro:bit...

Get silly
Shake your...

Flashin emotions
Make flashing...
## micro:bit units (by age)

<table>
<thead>
<tr>
<th></th>
<th>7-8</th>
<th>8-9</th>
<th>9-10</th>
<th>10-11</th>
<th>11-12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nature art</td>
<td>Nature art</td>
<td>Volcano animations</td>
<td>Data handling</td>
<td>Getting active</td>
<td>Computing fundamentals</td>
</tr>
<tr>
<td>Digital flashcards</td>
<td>Digital flashcards</td>
<td>Electrical conductors</td>
<td>Musical micro:bit</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11-14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Health tech</td>
<td>Health tech</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sensory classroom</td>
<td>Sensory classroom</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Introduction to cyber security</td>
<td>Introduction to cyber security</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cryptography</td>
<td>Cryptography</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Source: microbit.org

→ Select: Lessons
Education with the micro:bit is about kids inventing things and expressing themselves, with technology as a tool.

The idea comes first!
The idea comes first!

• This approach is one of the key things that sets micro:bit apart
  ....And one of our key tools in making micro:bit relevant and engaging for children that might not currently think computer science/engineering is for them

• We want children to feel confidence and ownership over their technology
Demonstration
How it Works
Device Abstraction
Ensuring a consistent experience across multiple editors & enabling ecosystem

- Courtesy of Lancaster University
- “Device Abstraction Layer” – DAL
- Now used well beyond the micro:bit project
- C++
  - Strong inheritance model for components
- EventBus for synchronisation
- Lightweight ‘fibers’ not threads
- Resource efficient
Avoid compilation – there are a lot of micro:bits (!) and building the project takes time and computing resources

Combine a pre-compiled blob with the user’s script

In the case of MicroPython: Interpreter+filesystem generated in browser

In the case of MakeCode, compile script to machine code in browser

Reduces compile time

Works offline
Offline, every time there is a new micro:bit runtime release

- `pxt/touchdevelop` wrapper
  - C++

- micro:bit runtime
  - C++

- ARM mbed & nRF SDK
  - C++

- Compile with yotta/gcc

- Partial/"Shell" hex file
  - ARM binary

In the browser (javascript) every time the user clicks ‘compile’

- Blocks Script
  - Block compiler

- Javascript (pxt) or TouchDevelop
  - In-browser compiler

- Users’s program hex
  - ARM binary

- Partial/"Shell" hex file
  - ARM binary

Combine and download

x1

x millions
Dedicated USB chip
Ensures robustness and ease of programming. Also a debugger.
Simulation

- Kids don’t even need physical device, and the test cycle is massively reduced

- This was possible because there was a compiler team working in a large collaborative project
  - Bringing in the heritage of TouchDevelop
  - But also building on .Net Gadgeteer

- Ongoing project with Arm to extend simulation capability
Analysis
Analyzing micro:bit’s Impact
…and why is it important? http://microbit.org/impact

• We look at impact micro:bit has through case studies, academic research, reporting and awards

• Combined, we offers insight into:
  • how micro:bit is being used with evidence as to why it works
  • Illuminates and recognises the work of the global community
  • Helps build accountability to our mission & celebrates Foundation achievements
  • Demonstrates the variety in education programmes and initiatives
  • Offers insight and context for new and potential programmes
• 12 case studies from UK, Uruguay, Lithuania, Denmark, Ireland, Taiwan, Singapore, Canada
• Impact statements from children, teachers and partners

“Students (especially many girls) become more enthusiastic about digital learning when you add social purpose…”
Allison Bellwood, World’s Largest Lesson

“This is a completely new experience – it is the first time I’ve done programming. I had great fun!”

“We would like children to think that they are a part of a larger global community.”
Wilson Shum, CEO SCALE InnoTech, Hong Kong

“They get super inspired, and really move on. They don’t have any barriers to learning how to program as we adults have. They just do it.”
Research
Academic studies into the micro:bit and its effectiveness

- Research papers from the UK, Denmark, Finland, Hungary, Western Balkans
- Academic investigations into teacher confidence, student response and engagement
- Tools such as baselining to assess confidence in computing before and after micro:bit activities
- Longitudinal studies monitor groups of teachers over time to assess changing teaching practice

As more M&E takes place worldwide, scope to engage with more research bodies and university partners to grow this body of evidence
Research methodology

Tools and questions used for evaluation

• Academic research into micro:bit uses a variety of methods:
  • Teacher surveys / interviews
  • Student surveys
  • Anecdotal evidence: teacher stories, examples
  • Supporting data e.g. from Microsoft Makecode / school learning portals

• Questions vary, but themes include:
  • teacher evaluation of programming competence **before** and **after** the programme if teachers would recommend micro:bit as a tool for their peers
  • if they intend to continue using micro:bit in their teaching
  • how easy and fun the students found programming with the micro:bit
  • if the students had done any programming before, if they found programming interesting and
  • if the students were nervous about programming
• Research studies into micro:bit can serve two purposes:

1. Evaluating teacher experience, confidence (self-efficacy) in programming & using micro:bit

   - 99% of teachers agreed that micro:bit made lessons and activities enjoyable for students
   - 97% find the device perfect for improving algorithmic thinking
   - 86% felt more confident as a teacher in their subject by using micro:bits

2. Indicates student attitudes towards learning programming with the micro:bit

   - 95% of students liked that the micro:bit was hands on
   - 93% of students said they learnt something new
   - 88% of students said they found it fun and exciting
   - 87% of students said the micro:bit helped them learn more about coding

Source: BBC research 2017; Data was gathered from 300 teachers who participated in telephone interviews.
70% more girls said they would choose Computing as a school subject after using the micro:bit

Source: BBC
An academic study explored the introduction of a micro:bit pilot in Finland

- 100 teachers and nearly 2,000 students

Research showed:

80% of students said that anyone could learn programming

90% of teachers said they would use the micro:bit again

85% of teachers felt the micro:bit could be spread as a national tool for programming

- Indicating a positive response to the device as a teaching & learning tool
- Positive attitudes and engagement towards the device from young learners
- Use of device supports building teacher confidence
- Use of device supports implementing new curriculum requirements
“Every school district in New Jersey will engage families, district boards of education, and community stakeholders to establish an equitable approach to computer science instruction...”
Katie Henry
Katie@microbit.org
Head of Partner Engagement, North America
The micro:bit runtime architecture

The micro:bit community encourages many languages...

University Collaboration

“Creating cool stuff”
– Pupils’ experience of the BBC micro:bit

Sue Sentance
King’s College London
London, UK
sue.sentance@kcl.ac.uk

Jane Waite
King’s College London
London, UK
jane.waite@kcl.ac.uk

Steve Hodges
Microsoft Research
Cambridge, UK
steve.hodges@microsoft.com

Emily MacLeod
King’s College London
London, UK
emily.macleod@kcl.ac.uk

Lucy Yeomans
King’s College London
London, UK
lucy.yeomans@kcl.ac.uk

ABSTRACT

The recent introduction of computer science (CS) education into schools in many countries has led to a surge in interest in programming tools and approaches which make CS concepts and tasks engaging, motivating and accessible to all. There is renewed interest in supporting learning through physical computing, which has been shown to be motivational whilst offering opportunities for collaboration and creativity. Within this context the BBC recently led a collaborative venture in the UK to develop a portable and low-cost programmable device. The consortium funded and produced one million devices, enough for every 11-12 year-old in the UK. In this paper, we report on what we believe

Figure 1: The BBC micro:bit
micro:bit labs

Current, historic and work in progress research and development projects relating to technical aspects of the micro:bit.

This page collates technical projects, research and development that have been done on/with the BBC micro:bit in universities, labs, clubs and schools around the world. It has been created by the micro:bit community, and you are welcome to submit any of your own work to it by submitting a project template.

If you’d like to see research about the impact of micro:bit, refer to our main site.

Add your research/project to the lab

What children's imagined use of the BBC micro:bit tells us about designing for their IoT privacy, security and safety

In this paper we describe initial results from the PETRAS project IoT4Kids, exploring the privacy and security implications of children programming the BBC micro:bit, an IoT-ready device designed for children
Analysis 🤔
Reflections on Tools
Blocks versus text
Increase the chance of success first time

- Blocks simplify the introductory programming experience
- Reduce the range of possible errors
  - Typos much harder to make
  - No syntax errors
  - Harder to make type errors
  - Structure of a programme is visible
- Make it easy to explore the available API
- MakeCode use “scratch-blocks”
• We want to support learners in transitioning to text
• Certain tasks are much easier when typing
  • EG mathematical and logic compound operations
• At the moment only about 10% of user switch
• Two way switching is crucial to limiting fear but is not common.
Orthogonality of features
Influence from Arduino (in that it’s not common there)

• Composibility
• Avoids needing to understand oddities of hardware
• Don’t break expectations
• Unusual in the embedded world
• Places where we do break this are painful
Concurrency & Events
Influence from Scratch

- Very natural for kids for ‘UI’ style interactions
- Smaller code size
- Visual representation of a programme
- Can become more complex for some games
- In micro:bit there’s a split between paradigms – MicroPython is ‘main loop’ driven, MakeCode is event-based
  - Quite different representations of similar programs can be confusing
  - But teaching both approaches is important
- Concurrency:
  https://makecode.microbit.org/device/reactive

```python
from microbit import *

while True:
    if button_a.is_pressed():
        display.show(Image.HAPPY)
    if button_b.is_pressed():
        display.show(Image.SAD)
```
This is a BBC micro:bit
This is a BBC micro:bit

This is a BBC micro:bit (V2)
This is a BBC micro:bit (V2)
This is a BBC micro:bit (V2)
It does everything the original micro:bit does

And also...
This is a BBC micro:bit (V2)
It does everything the original micro:bit does
And also...
It can make and sense sound
This is a BBC micro:bit (V2)
It can make and sense sound
It can detect touches on the logo
And also...
It can make and sense sound
It can detect touches on the logo
This is a BBC micro:bit (V2)
It does everything the original micro:bit does

And also...
It can make and sense sound
It can detect touches on the logo
It has enough processing power for AI and ML experiments
Power indicator

Microphone

Speaker
micro:bit voice print

https://www.edgeimpulse.com/blog/voice-activated-microbit
MICROBIT
PROJECT: MACHINE LEARNING DANCE MOVE DETECTOR

a dance detecting micro:bit
Demonstration
Influences...
CMT-1203, MLT-8530, SMT-0440-S-R and 1.4

![Graph showing frequency response for different components]
Birdsong generation project

Generating birdsongs with Wavenet!

- Listen to natural song at soundcloud
- Listen to generated song at soundcloud

Table of Contents

- Quick execution
  - Requirements
  - Command
  - Generated song
- Overview
  - Abstract
  - Literature review
  - Methodology
  - Results
  - Discussion
  - Conclusion
We’re right at the beginning
micro:bit Classroom

• Supported by Nominet
• Researched-backed, education focus

Issues
• Starting with a blank slate
• Collecting and assessing students’ work
• Resuming lessons is hard

People liked
• No sign in, use immediately
• Physical computing!
• Using micro:bit for group work
micro:bit Classroom

- Create a classroom session
- Start students on the same code
- View & share code during lessons
- Collect & download students work
  - As word doc for editing
  - As HTML to resume lessons

- No logins, dynamic creation of sessions
- Designed to support teachers in a real classroom during activities, not replace them!
Classroom Demo
Bonus Material

• V2 Audio Pipeline
• Some interesting sound blocks
Micro:bit v2 Audio Pipeline (Default)

- Speaker Disable
- Volume blocks
- "Play sound" blocks
- V1 sound blocks
- Sound Expressions Synth
- Virtual Audio Pin (behaves like P0)
- Memory Source

Audio Data Flow:
- "Play sound" blocks -> Sound Expressions Synth
- V1 sound blocks -> Virtual Audio Pin (behaves like P0)
- Memory Source -> Mixer
- Mixer -> NRF52PWM
- NRF52PWM -> PO / P1 / P2

Audio Control Flow:
- Speaker Disable
- Volume blocks
- "Play sound" blocks
- V1 sound blocks

© Micro:bit Educational Foundation 2020
What can we do at the moment?
As with everything today, this is a work in progress

Making Sound 🎵 expiresIn

- All of the v1 ‘music’ blocks, which come out the speaker
- Volume control
- Combine sound with all of the other sensors
- Use pre-built micro:bit ‘sounds’
- Create sounds using C/C++

Sensing Sound 🎧👂

- Events for loud, quiet
- Getting the sound level
- Change the level at which the ‘loud’ event triggers
Make a sound

https://github.com/microsoft/pxt-microbit/blob/master/libs/core/music.ts (and .cpp)

- “Ring tone” block – nothing to do with 90s mobile phones
  - Makes a sound of a particular frequency
  - The input can be numbers you choose
  - Or any of the inputs on the micro:bit

- The sounds keeps playing until you stop it!
  - There’s a bug in MakeCode so we’ve made a special ‘stopRinging’ block

- If you don’t have a ‘delay’ after your ‘ring tone’ block, the next tone will start immediately. The ‘pause’ will choose how long the tone sounds for!
Make a sound (music blocks)

- All the existing blocks work!
- You can have a lot of fun with the sound blocks and the new ‘play melody’ block
- Sound will also come out of the edge connector.
• We can use the ‘acceleration (mg)’ block to give us values
  • This gives us values between -1024->1024
• We can’t have a ‘negative’ frequency so we use ‘absolute of’ which ensure the value is the ‘positive’ bit only
  • But we could use other maths – for example adding 1024!