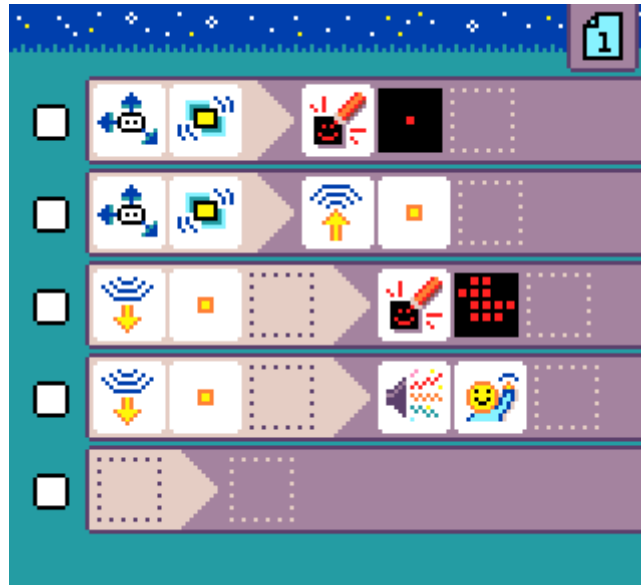




MicroCode for micro:bit

Created by Peli de Halleux



<https://learn.adafruit.com/microcode-for-microbit>

Last updated on 2024-06-03 03:43:06 PM EDT

Table of Contents

Overview	3
• Parts	
First Program	4
Web Editor	6
• Automatic download to micro:bit V2	
• Accessibility	
Accessories	7

Overview

[MicroCode](#) is a icon-based (text-free), keyboard friendly editor for the micro:bit V2. It is suitable for younger learners and users with variable accessibility needs.



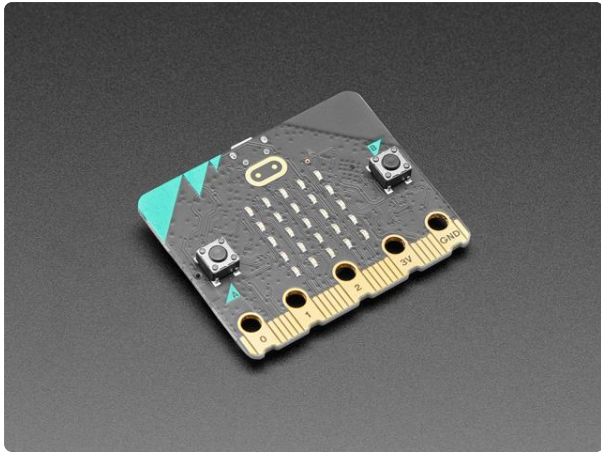
- Kid-friendly, icon-based, structured [web-based](#) code editor
- Live, the code is downloaded to the micro:bit on every edit
- Cursor based navigation with keyboard (switch access compatible), mouse, touch, screen reader support
- Accessories (LEDs, servos, ...) supported via [Jacdac](#)

MicroCode is open source at <https://github.com/microsoft/microcode>.

MicroCode requires micro:bit V2. The micro:bit V1 is **not** supported.

Parts

Please be sure the micro:bit is a V2 (Version 2). This project doesn't work with the original / V1 board.

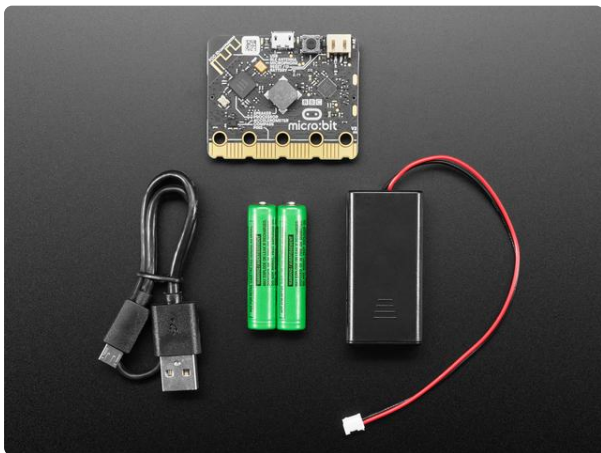


BBC micro:bit v2

Meet the new BBC micro:bit v2 - Upgraded with a powerful new processor that has tons more capability and also adds more sensing with a new speaker and...

<https://www.adafruit.com/product/4781>

or



micro:bit v2 Go Bundle - Batteries and USB Cable Included

Meet the new BBC micro:bit v2 - Upgraded with a powerful new processor that has tons more capability and also adds more sensing with a new speaker and...

<https://www.adafruit.com/product/4834>

First Program

Here is a screenshot of one the first examples in MicroCode: **smiley buttons**.



The goal of this program is to show a smiley and plays a happy sound when pressing A, and a frowney animation and unhappy sound when B is pressed.

The [MicroCode language](#), is defined in terms of **pages**, where a page has a list of **rules**, and each rule consists of a **When** section and a **Do** section, each with a list of programming **tiles**. For this example, we have 4 rules.

The first rule tells MicroCode to show an image when 'A' is pressed on the micro:bit.

You will find more [samples](#) in the MicroCode documentation.



• when



press



button A, do



show image smiley.

The next rule tells MicroCode to play a sound when 'A' is pressed as well. All rules run at the same time.



• when



press



button A, do



play sound emoji



happy.

The next two rules are similar but for button 'B'.

Web Editor

The MicroCode web editor is at aka.ms/m9. It requires a browser with WebUSB support.

- **Left**, **Right**, **Up**, **Down** moves the cursor
- the **Left** and **Right** keys will wrap around in the editor on the **same** page
- the **Up** key on top of the screen will act as **back**. It will go back the pages until page 1 is in focus, then pressing up will go back to the start screen.
- the **Down** key will wrap around the pages
- **Enter** or **Space** for **A** button
- **Backspace** for **B** button
- **]** or **Page Up** to go to the next page, **[** or **Page Down** to go to the previous page
- Copy the URL to share your program.

Automatic download to micro:bit V2

When the editor does not detect a micro:bit connected to the computer, it will automatically prompt you with instructions on how to connect (requires WebUSB).

Once your micro:bit V2 is paired, the web editor will be able to download your MicroCode program to your micro:bit. This happens on every edit, so your program is always up-to-date (and running)!

If your micro:bit disconnects, you will see a micro:bit icon show up on the upper left corner of the editor. Click on that button to reconnect again.

Accessibility

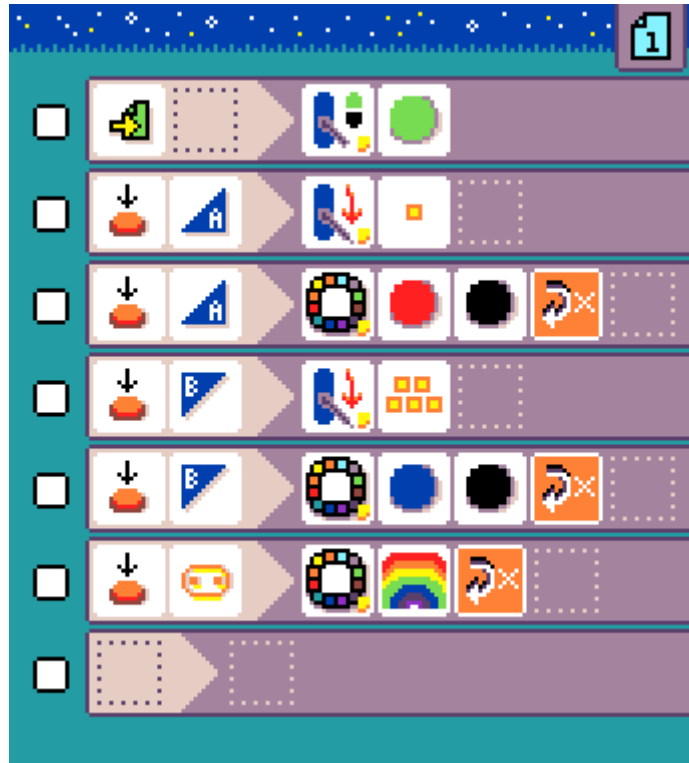
MicroCode has a lot of accessibility goodness:

- fully keyboard enabled, and switch panel friendly. Only 3 keys are need to use MicroCode.
- tooltip read aloud and screen reader support
- game console support (XBox, ...)

Accessories

Additional sensors and actuators can be added as [Jacdac](#) modules. MicroCode automatically detects additional hardware and will show the additional tiles if supported.

The [railroad crossing sample](#) shows a program using a servo and a LED ring Jacdac module.



The



servo uses a [servo motor](#) move to move a physical arm. The



servo arm orientation is mapped to the wall clock hours: **0** (or **12**) is rotated 90 degrees from the resting position to the left, **6** is rotated 90 degrees right from the rest position. There is also a secret animation mode when you press the micro:bit logo button.



- when

press



button A, do



servo set arm to 1 o'clock.



The



LED uses a [programmable LED ring](#) module to display blue and red colors.



• when

press



button A, do



LED set all color to



red, black and repeat.

