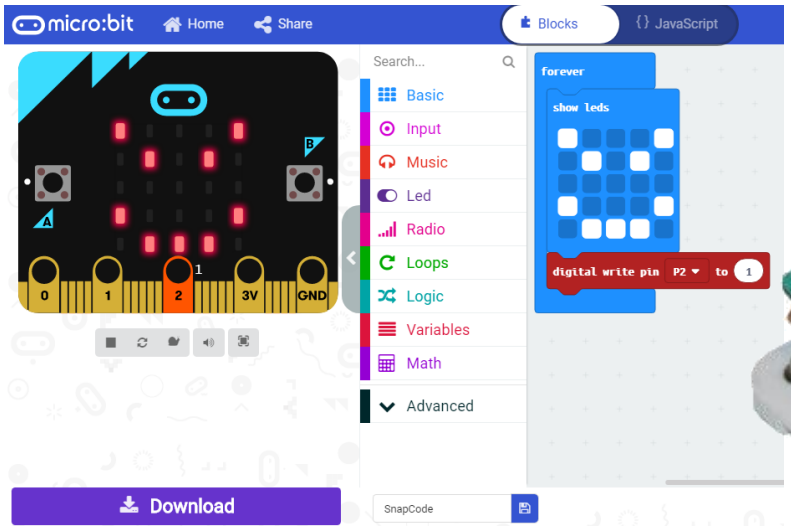
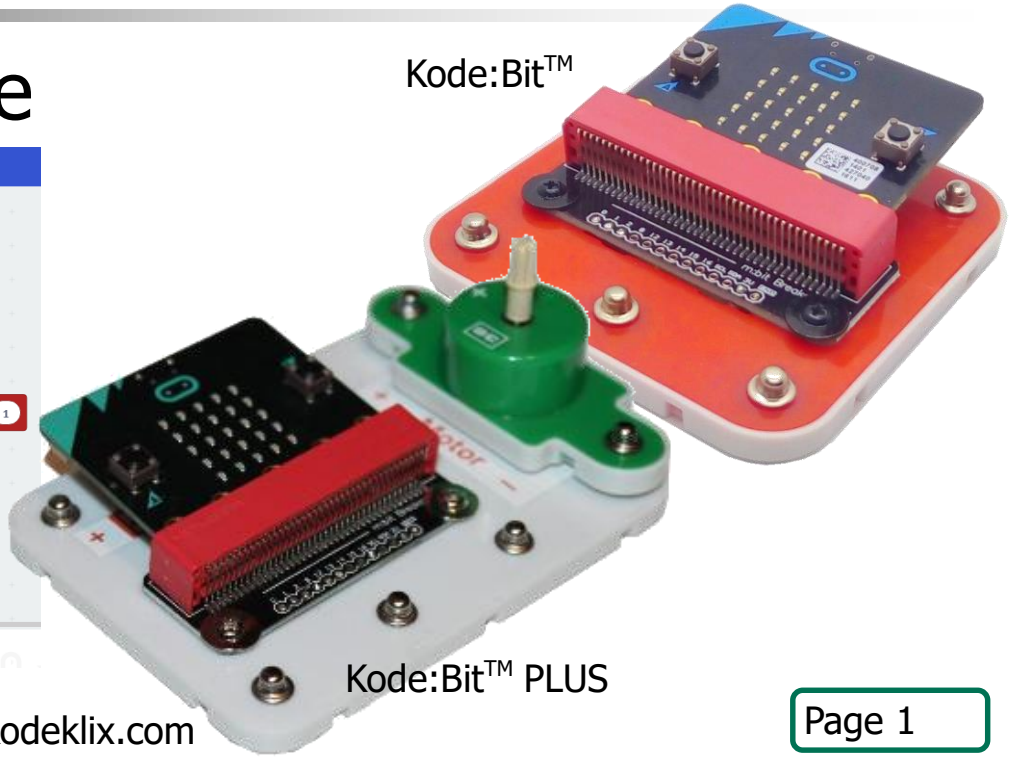
KODEKLIX®

Kode:Bit for the BBC micro:bit

Quick Start Guide





Kode:Bit™

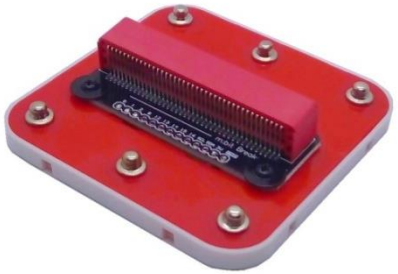
Kode:Bit™ PLUS

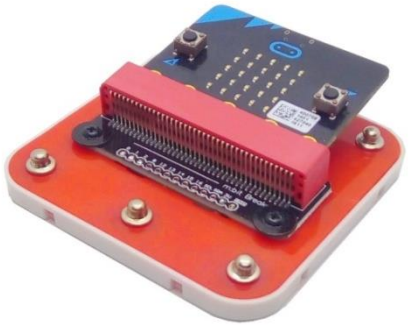
Introduction

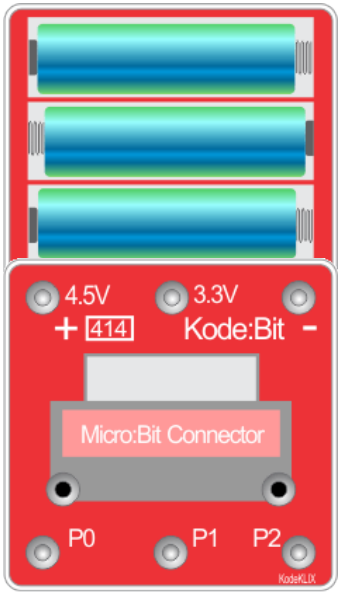
- Kode:Bit™ allows you to connector your BBC micro:bit board to:
  - Other snap controllers such as the SnapCPU™ for use as a co-process for display or other functionality
  - Build interfacing circuits using your KodeKLIX® or other compatible snap electronic components

Power Configuration

- Your BBC micro:bit simply inserts into the connector on the Kode:Bit, but before doing so, consider how you will be using your micro:bit…

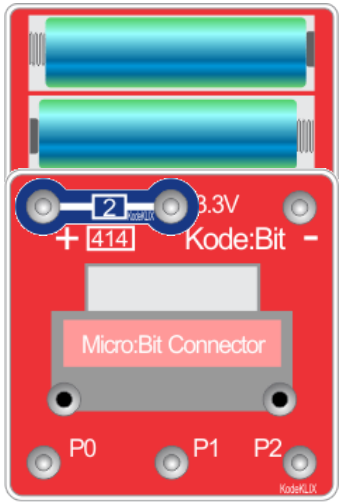






4.5V Battery Pack

In this configuration **no link** is required. Internally, the voltage from battery pack [70] is reduced to the safe level needed for the micro:bit



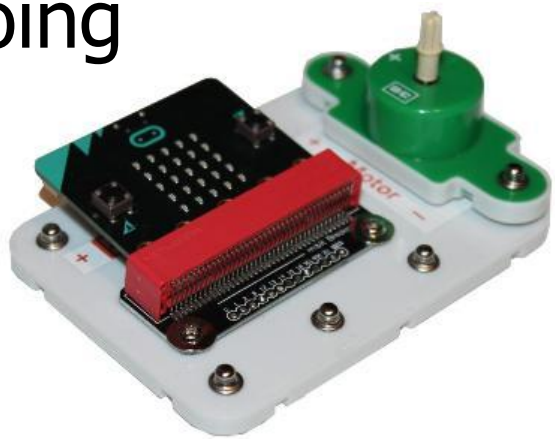
3.0V Battery Pack

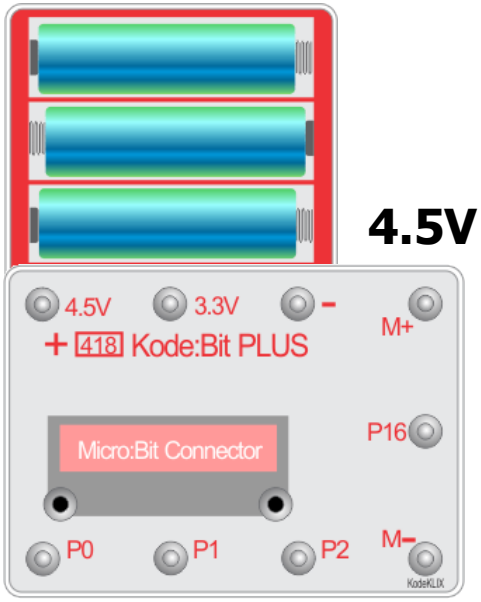
In this configuration a link may be required if your display is very dim or some of the micro:bit's sensors do not work reliably.

This configuration is also used if you power your Bit with the USB cable.

Power Configuration - PLUS

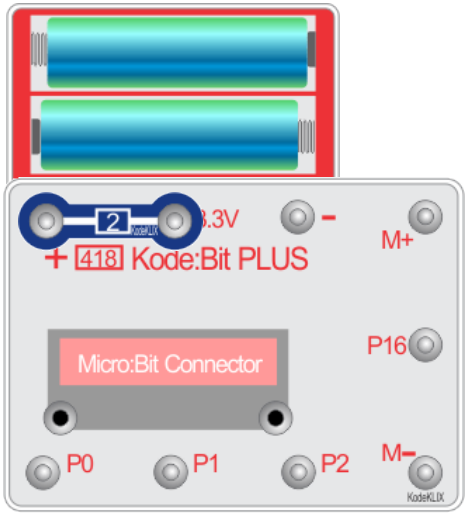
- Your BBC micro:bit simply inserts into the connector on the Kode:Bit PLUS, but before doing so, consider how you will be using your micro:bit…





4.5V Battery Pack

In this configuration **no link** is required. Internally, the voltage from battery pack [70] is reduced to the safe level needed for the micro:bit



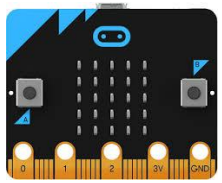
3.0V Battery Pack

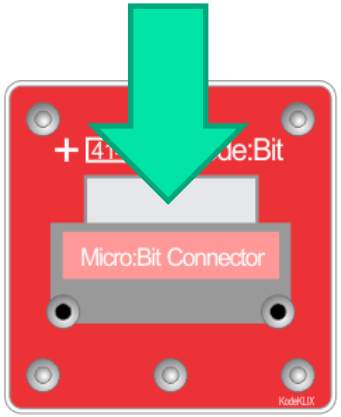
In this configuration a link may be required if your display is very dim or some of the micro:bit's sensors do not work reliably.

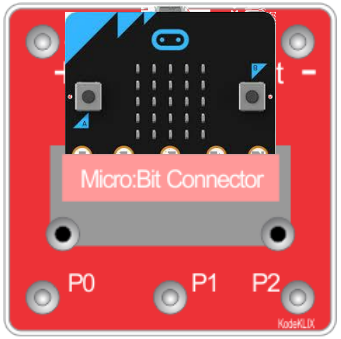
This configuration is also used if you power your Bit with the USB cable.

Inserting the micro:bit

- Once you have decided on whether to use the power link or not, the micro:bit can be easily inserted into the Kode:Bit adaptor
- Power and ground (0V) are supplied to the micro:bit through the top snaps
- Input/Output pins P0, P1 and P2 are routed as shown
  - These pins are yours to use, but remember all the connection rules for the micro:bit still apply

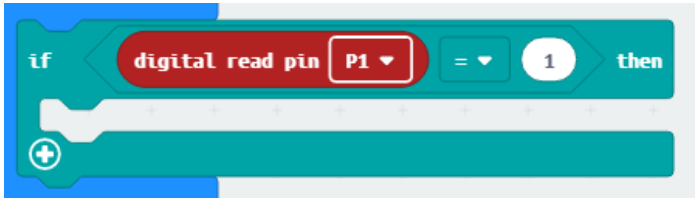


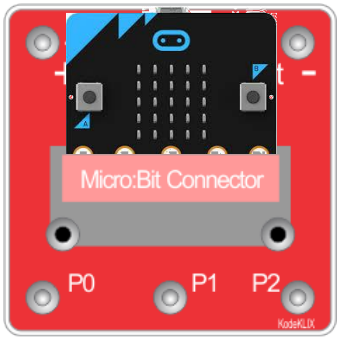




micro:bit Coding

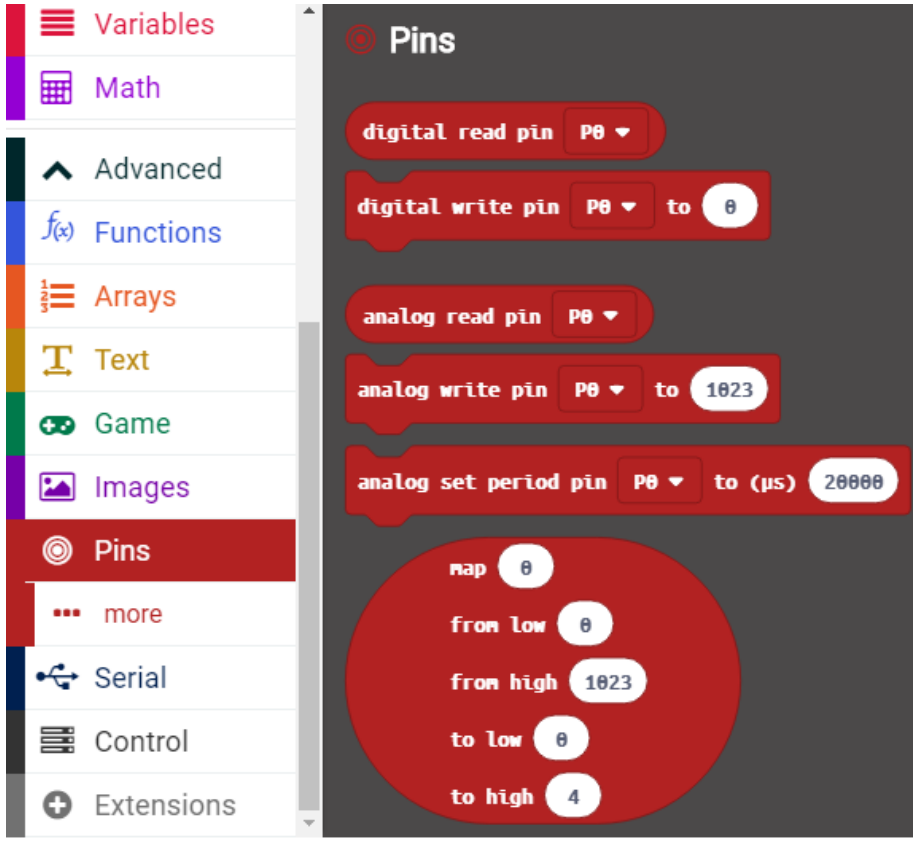
- Making code for the micro:bit is the same as usual, using the on-line MakeCode utilities at http://makecode.microbit.org/#editor
- Downloading code for the micro:bit is the same as usual, using the USB cable and drag-and-drop method
- Basic pin I/O can be controlled with these MakeCode Blocks





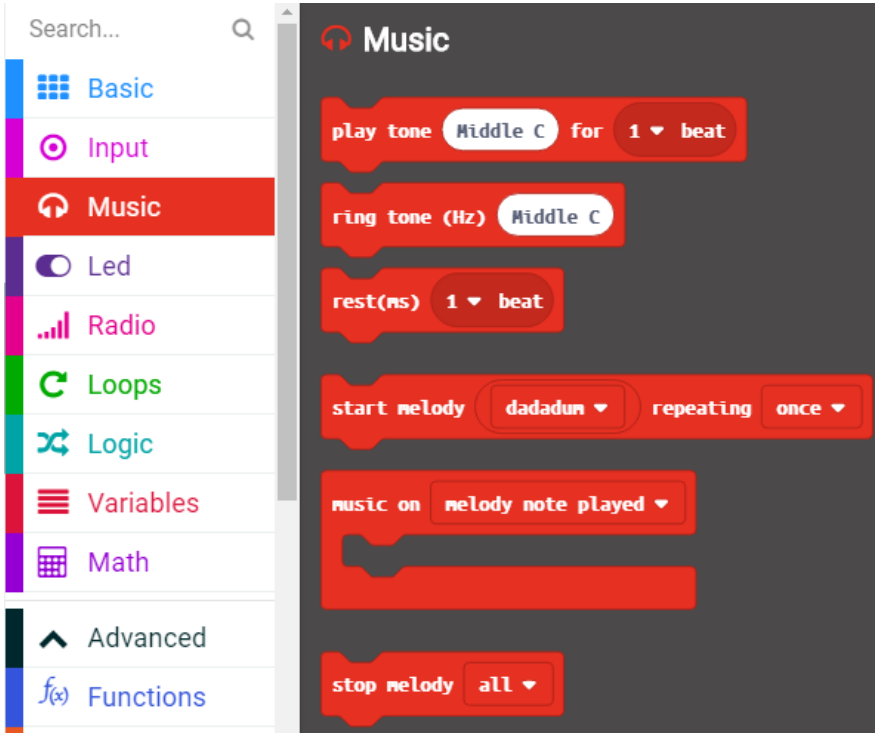
micro:bit I/O Coding Blocks

- Various coding blocks can be found under the Pins menu in the advanced section
- Digital I/O is where signals are either ON (1) or OFF (0)
- Analog inputs are signals which have a value between 0-255



micro:bit Music Coding

- By default, the micro:bit plays all sounds and music through P0
- Since the micro:bit has no built-in speaker so an external speaker [20] or piezo [11] will need to be connected

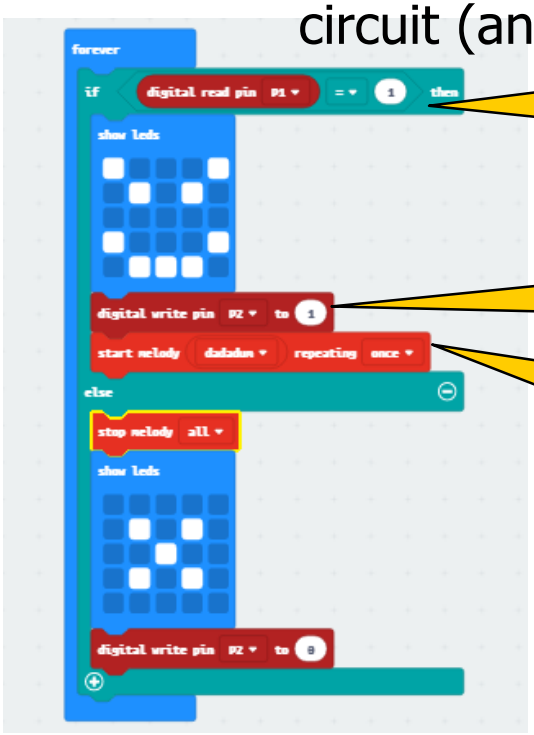


Completed ☐

Example Project 1.0

Push button input to the Kode:Bit™

- Assemble the snap components as shown in circuit 1.0; Construct the MakeCode below and download to the micro:bit
- The battery pack [70] is for powering the circuit (and the micro:bit when not on USB)



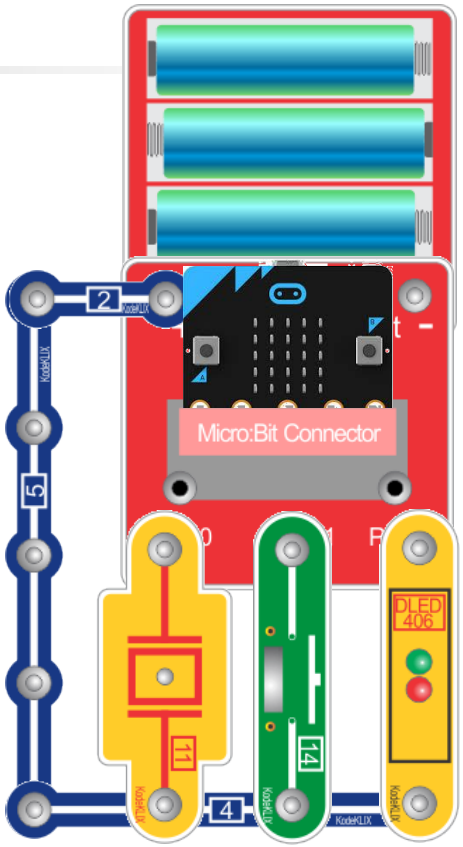
**code set to read from inputP1

**illuminates the DLED on P2

**plays tune through piezo on P0

Connecting to "high" is the equivalent of connecting to the +ve terminal of the battery box.

A push-button switch is momentary (only valid when being pressed) so needs to be held until the code "sees" it.

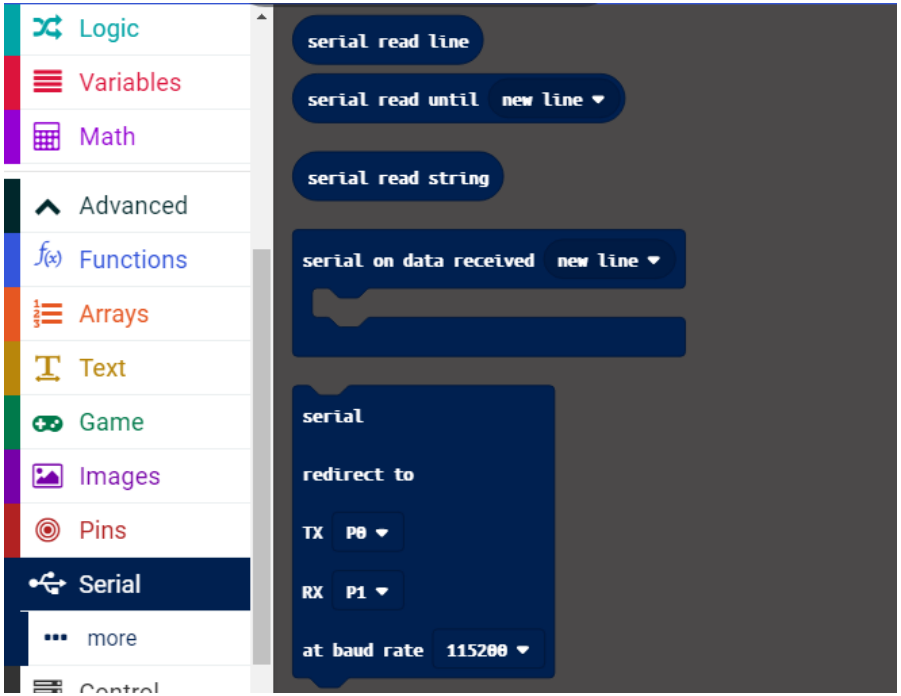


Circuit 1.0

Challenge: modify the code to make P2 stay lit for a fixed time after the button P1 is pressed.

Advanced micro:bit I/O Coding

- Various coding blocks can be found under the Pins menu in the advanced section
- Digital I/O is where signals are either ON (1) or OFF (0)
- Analog inputs are signals which have a value between 0-255

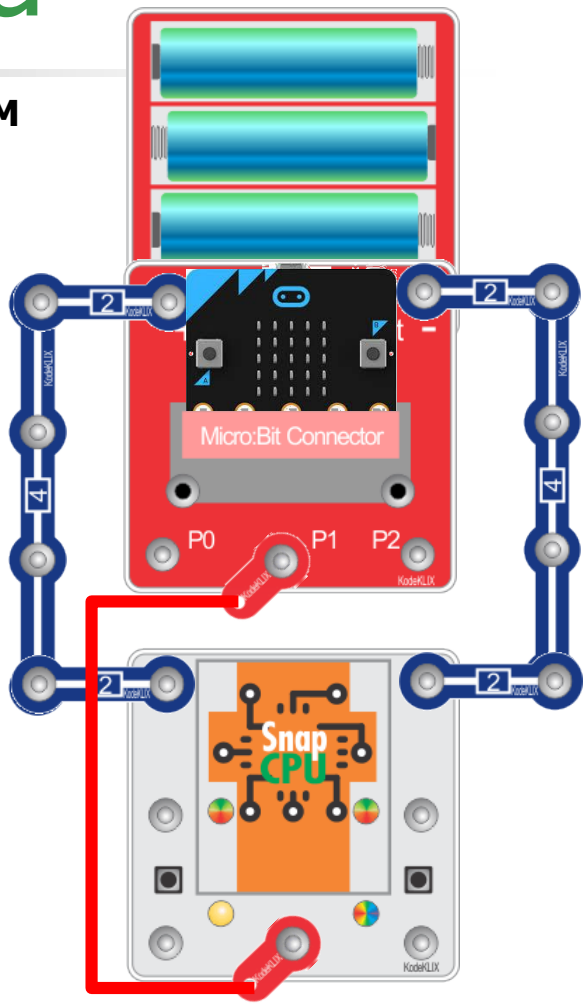


Completed ☐

Example Project 2.0a

Using the micro:bit with the SnapCPU™

- This project demonstrates how to computers, the SnapCPU and the micro:bit can be made to communicate
- In this project, the micro:bit is a display for the SnapCPU™
- Assemble the snap components as shown in circuit 2.0
- Construct the MakeCode on the next page and download to the micro:bit
- Construct the Blockly code on the next page and download it to the SnapCPU™
- If all goes to plan, the micro:bit display show A, B, C, …

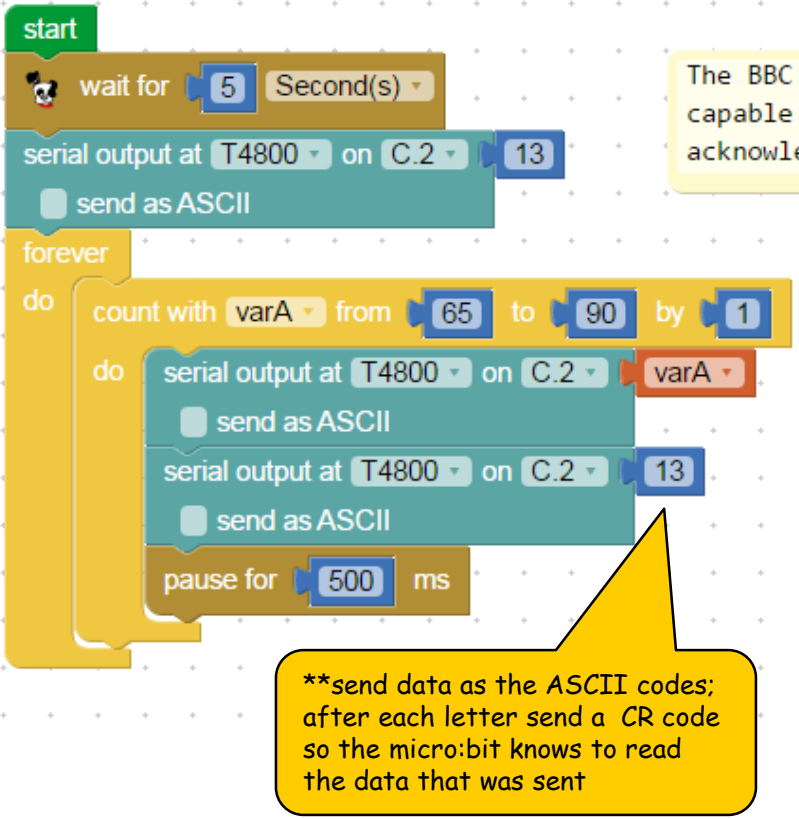


Circuit 2.0

Completed ☐

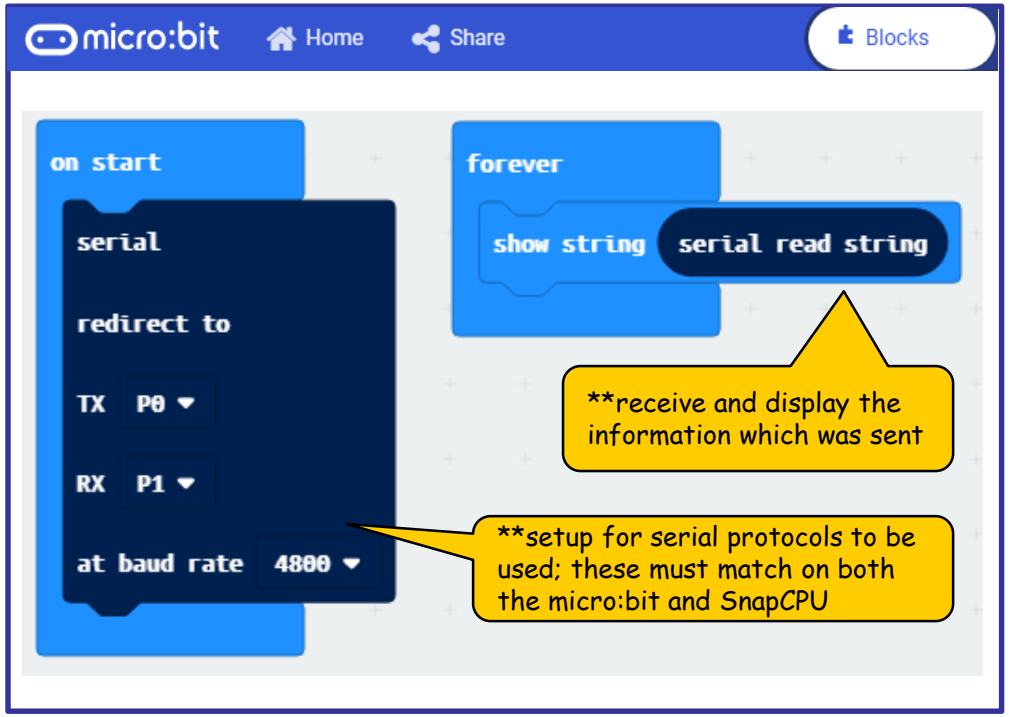
Example Project 2.0b

Using the micro:bit with the SnapCPU™



The BBC micro:bit is very slow to wake up, so allow 5seconds. It is only capable of True Signal Level RS232, and needs a CR (code 13) to reliably acknowledge data is available in its RS232 input buffer.

**send data as the ASCII codes; after each letter send a CR code so the micro:bit knows to read the data that was sent



**receive and display the information which was sent

**setup for serial protocols to be used; these must match on both the micro:bit and SnapCPU

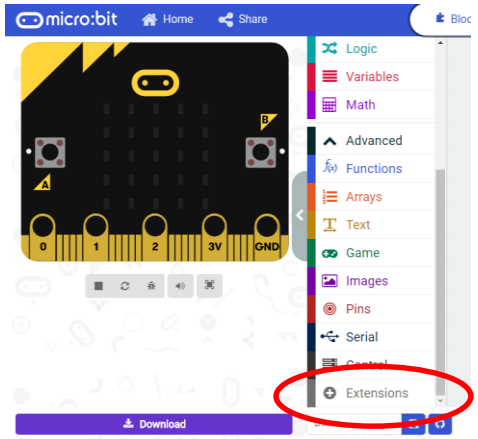
Challenge: modify the code in the SnapCPU to send a message letter by letter, rather than the ABCDE sequence of characters… you can do it by codes or characters!

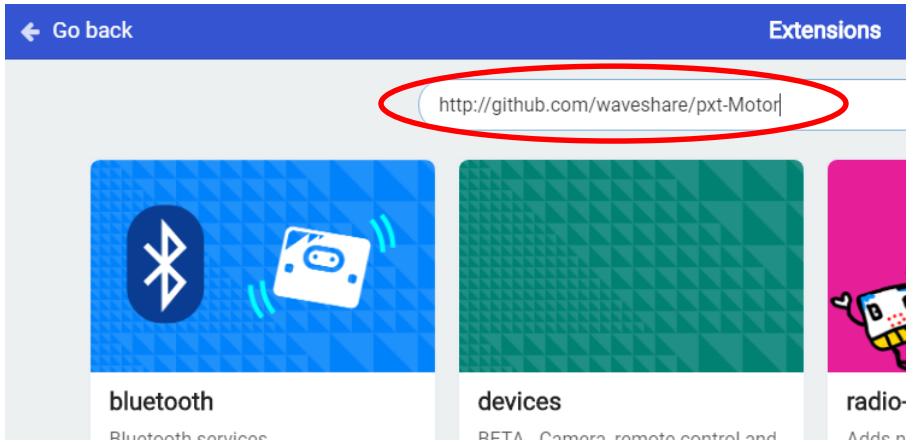
Kode:Bit™ Motor Driver Blocks

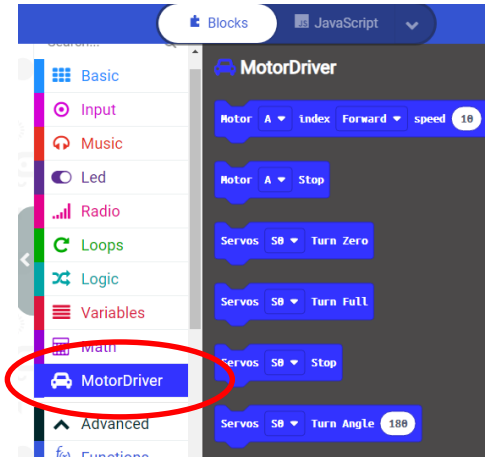
- To simplify the control of motors, install this driver as an extension to MakeCode:

  http://github.com/waveshare/pxt-Motor

  1. Open extensions menu
  2. Enter the web link details
  3. A new block menu is added!



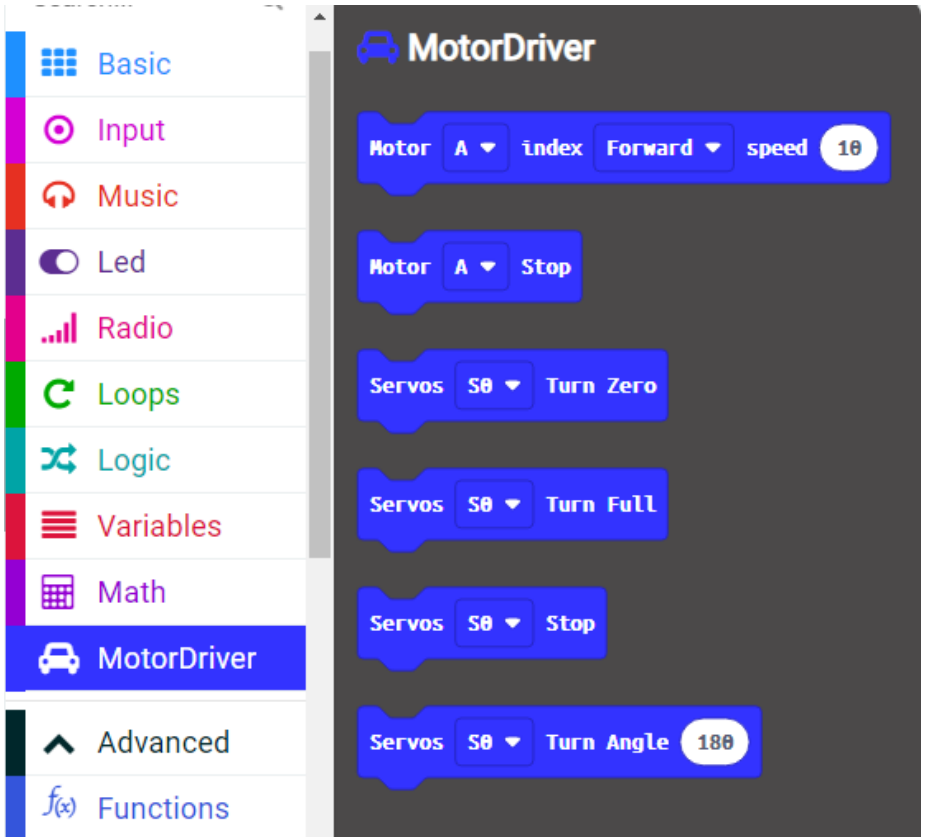




micro:bit Motor Coding Blocks

- Various coding blocks can be found under the new MotorDriver menu
- Motor 'A' is used
- The first block controls motor direction and speed
- The second block stops the motor

Note: driver may need to be installed for each new project



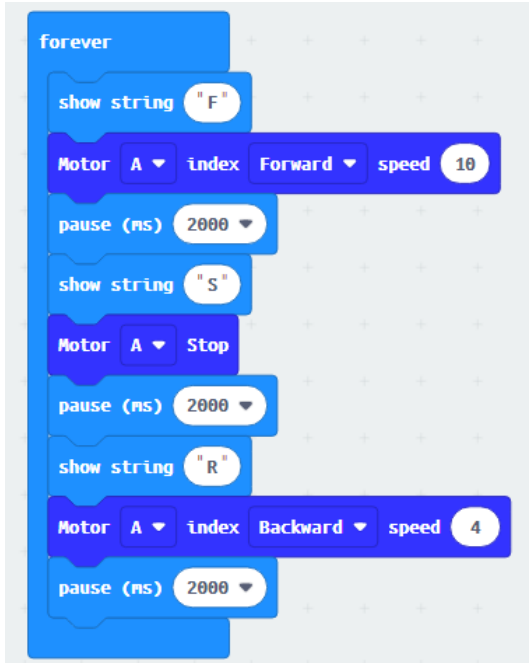
Completed ☐

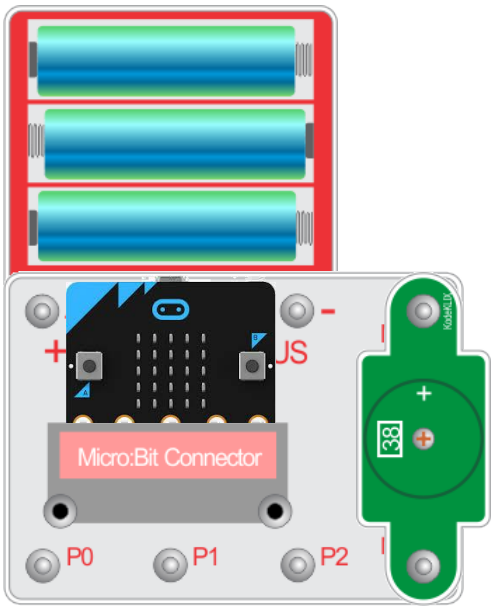
Example Project 3.0

Motor control with the Kode:Bit™ PLUS

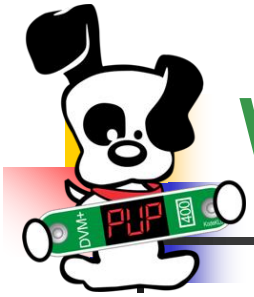
- This project demonstrates how to control a motor from the micro:bit
- Assemble the snap components as shown in circuit 3.0
- Construct and download the MakeCode shown below
- If all goes to plan, the micro:bit display show the direction F, S, R as the motor spins forward, stops and then backwards

Note: driver may need to be installed for each new project





Circuit 3.0



What Next…?

- The internet is has many example projects for the micro:bit
  - Try building these using snap parts rather than tiny components and clips
  - Think of other creative ways to use the power of two CPUs connected at the same time
    - Send / receive data
    - Send / receive signals