

MICROBIT PROGRAMMING (BLOCK-BASED)

(Time required 30-45 minutes session)

Introduction

In this activity, we will be reading one of the most common and simple inputs – a push button – by using a digital input. We will use it to cycle through different colors on the RGB.

Parts Needed (included in the kit):

1x micro:bit

1x Micro B USB Cable

1x micro:bit Breakout (with Headers)

1x Breadboard

8x Jumper Wires

1x Momentary Push Button

1x 10k Ω Resistor

1x Common Cathode RGB LED

3x 100 Ω Resistors

Push Button Introduction

A momentary push button closes or completes the circuit only while it is being pressed. The button has four pins, which are broken out into two sets of two pins. When you press down on the button and get a nice "click," the button bridges the two sets of pins and allows current to flow through the circuit.

How do you know which pins are paired up? The buttons included in this kit will only fit across the breadboard ditch in one direction. Once you get the button pressed firmly into the breadboard (across the ditch), the pins are horizontally paired. The pins toward the top of the breadboard are connected, and the pins toward the bottom of the breadboard are connected.



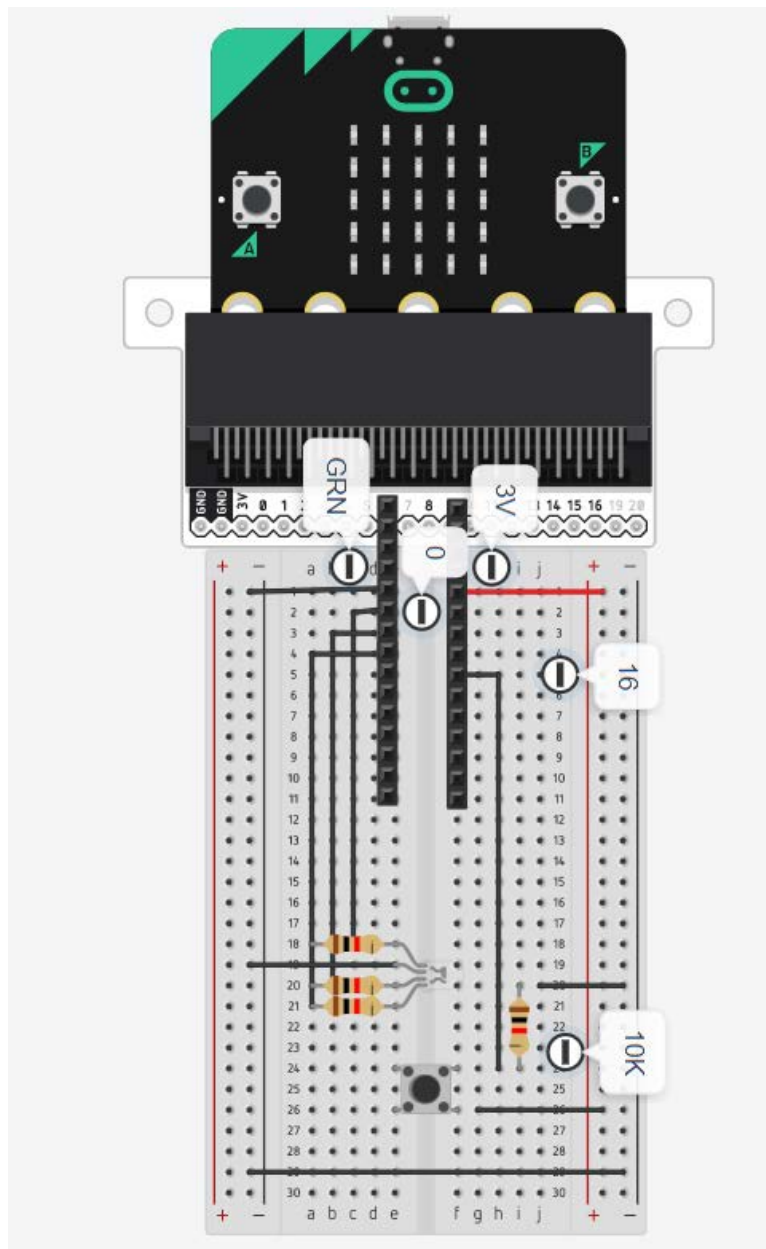
Introduction the Red/Green/Blue (RGB) LED

The RGB has four pins with each of the three shorter pins controlling an individual color: red, green or blue. The longer pin of the RGB is the common ground pin. You can create a custom-colored LED by turning different colors on and off to combine them. For example, if you turn on the red pin and green pin, the RGB will light up as yellow.



1 - RED
2 - GROUND
3 - GREEN
4 - BLUE

Push Button Wiring Diagram



Set Pull Pin

When you start your micro:bit, some pins can be set to be naturally on or naturally off. The set pull pin block allows you to set an initial state of a pin by selecting a pin and then its pull state, which is UP, DOWN or NONE.

Set Pin to Emit Event

You use the set pin to emit event block to create a type of event for a specific pin to emit or send out when it reaches that state. As an example, we set P16 to emit an EDGE event, which means that it changed from HIGH to LOW or LOW to HIGH.

On Event

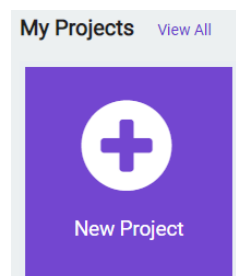
Under the Advanced blocks you can find the Control blocks. These are the blocks that are the most complicated to use, but are the most powerful. The On Event block accepts an event type to watch for and a pin that event should happen on. When that specific event is emitted from that pin, it will trigger whatever code is inside of it. Take a moment to look through the list of different events that you can listen for!

if / Else if / Else

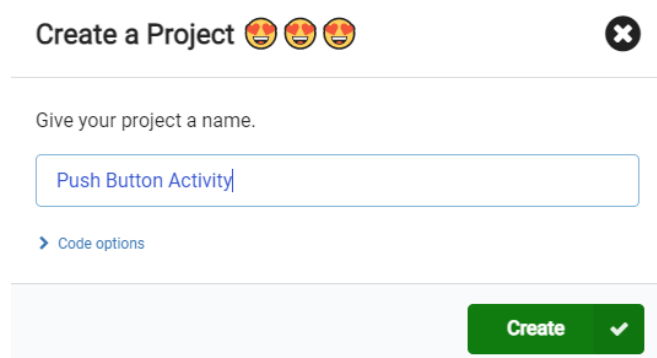
Finally, inside of the forever block is a more complex if block, which is an if / if else / else tree. To build this more complex "if" statement, add a standard if / else block into your program. Then click on this small gear in the upper left-hand corner of the block. This will open a tiny interface with more blocks in it. You can drag more else if blocks into the structure here to build your decision tree.

Building the "Push Button Activity"

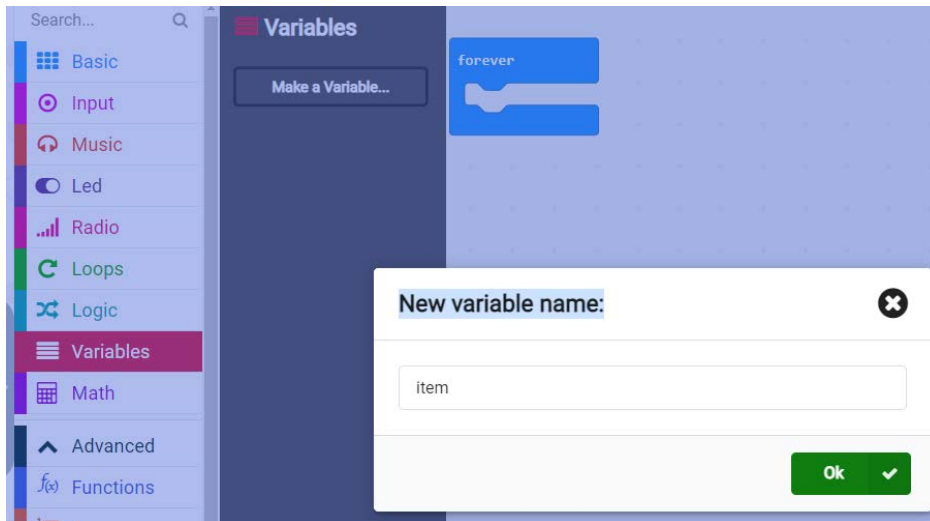
Step 1: Go to <https://makecode.microbit.org/#> and create a New Project



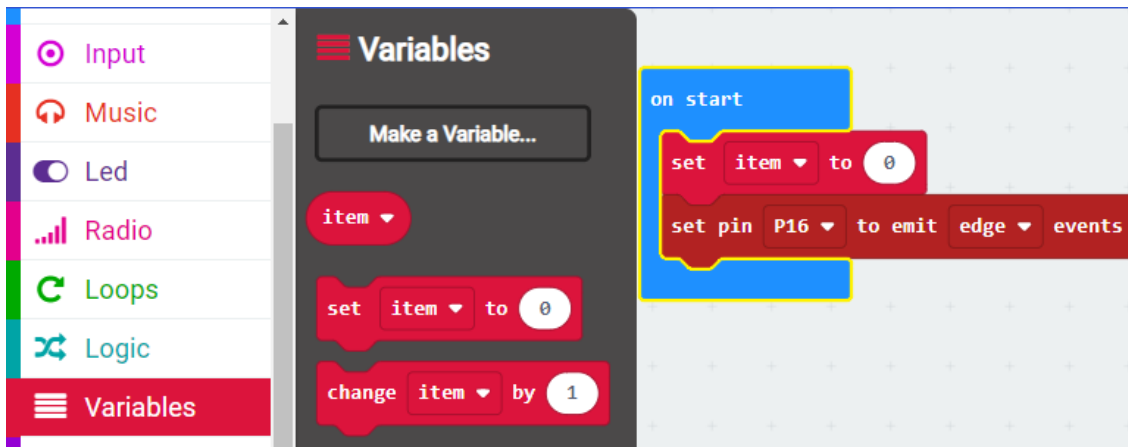
Step 2: Click on New Project and give it a project name **Push Button Activity** and click Create



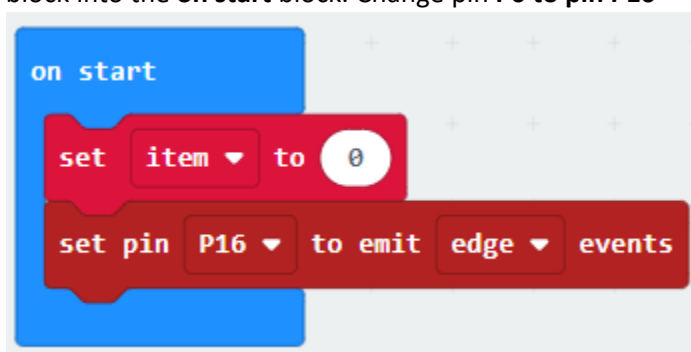
Step 3: Once the MakeCode is launched, click on the **Variables** category and then click on **Make a Variable**. In the New Variable name window type **item** then click **Ok**



Step 4: Once the **item** variable has been created select and drag the **set item to 0** block into the **on start** block



Step 5: Click on the **Advance** | **...more** category and select and drag the **set pin P0 to emit edge events** block into the **on start** block. Change pin **P0** to **pin P16**



Step 6: Click on the **Logic** | **Conditionals** category then select and drag the **if <true> then** block into the **forever** block

Step 6: Click on the **Logic | Comparison** category then select and drag the **<0> = <0>** block into the **if <true> then** block

Step 7: Click on the **Variables** category then select and drag the **item** block into the **if <true> then** block

Step 8: Click on the **Advance** category then **Pins** then select and drag the **analog write pin P0 to 1023** block into the **if <true> then** block.

Step 9: Right click on the **analog write pin P0 to 1023** and select duplicate. Drag the copy block into the **if <true> then** block. Change pin P0 to pin P1 and **1023** to 0

Step 10: Right click on the **analog write pin P1 to 0** and select duplicate. Drag it to the **if <true> then** block. Change pin P1 to pin P2



Step 11: Click on the plus (+) to create an additional **else if** statement block

Step 12: Click on the **Logic | Comparison** category then select and drag the **<0> = <0>** block into the **else if <true> then** block. Change the value of 0 to 1

Step 13: Click on the **Variables** category then select and drag the **item** block into the **else if <true> then** block

Step 14: Click on the **Advance** category then **Pins** then select and drag the **analog write pin P0 to 1023** block into the **else if <true> then** block. Change 1023 to 0

Step 15: Right click on the **analog write pin P0 to 0** and select duplicate. Drag it to the **else if <true> then** block. Change pin P0 to pin P1 and **0** to 1023

Step 16: Right click on the **analog write pin P1 to 1023** and select duplicate. Drag it to the **else if <true> then** block. Change pin P1 to pin P2 and 1023 to 0

```

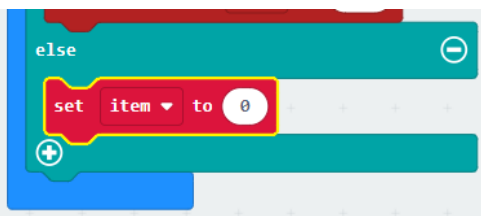
forever
  if item = 0 then
    analog write pin P0 to 1023
    analog write pin P1 to 0
    analog write pin P2 to 0
  else if item = 1 then
    analog write pin P0 to 0
    analog write pin P1 to 1023
    analog write pin P2 to 0
  else
  
```

Step 17: Repeat steps 11 – 16, but with the following changes as the figure below

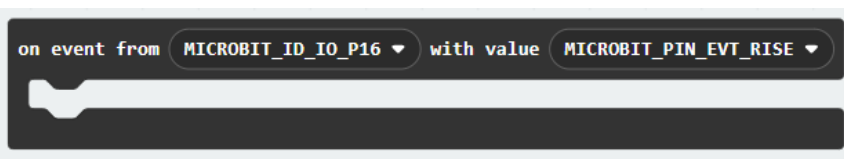
```

forever
  if item = 0 then
    analog write pin P0 to 1023
    analog write pin P1 to 0
    analog write pin P2 to 0
  else if item = 1 then
    analog write pin P0 to 0
    analog write pin P1 to 1023
    analog write pin P2 to 0
  else if item = 2 then
    analog write pin P0 to 0
    analog write pin P1 to 0
    analog write pin P2 to 1023
  else
  
```

Step 18: Click on the **Variables** category then select and drag the **set item to 0** block into the **else block**

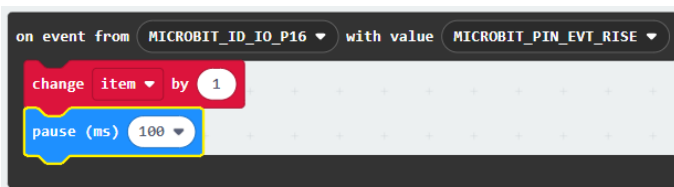


Step 19: Click on the **Control** category then select and drag the **on event from** <MICROBIT_ID_BUTTON_A> with value <MICROBIT_EVT_ANY> block into the **program space**. Then change <MICROBIT_ID_BUTTON_A> to <MICROBIT_ID_IO_P16> and <MICROBIT_EVT_ANY> to <MICROBIT_PIN_EVT_RISE>

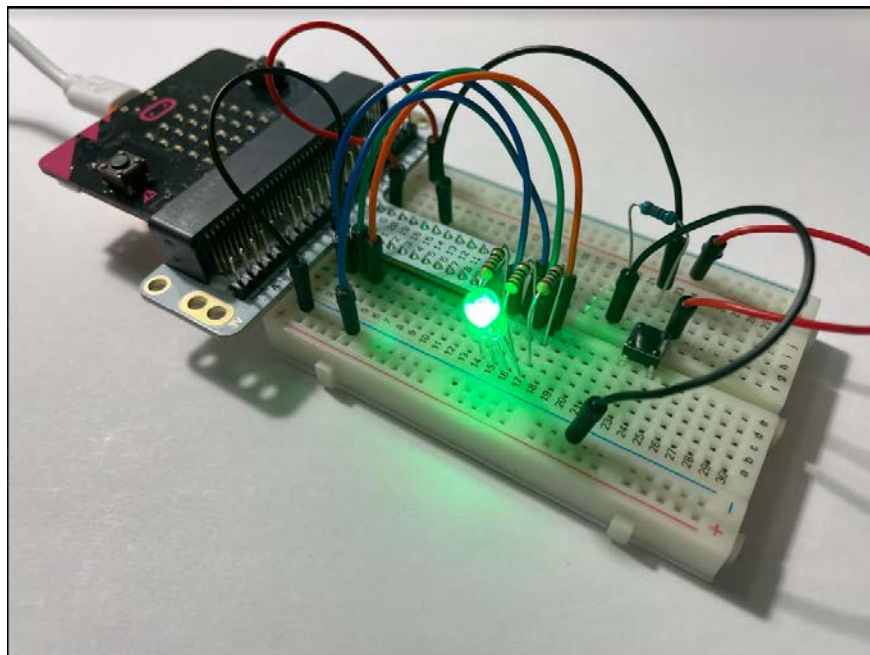


Step 20: Click on the **Variables** category then select and drag the **change item to 1** block into the **on event from block**

Step 21: Click on the **Basic** category then select and drag the **pause (ms)** into the **on event from block**



Final Prototype

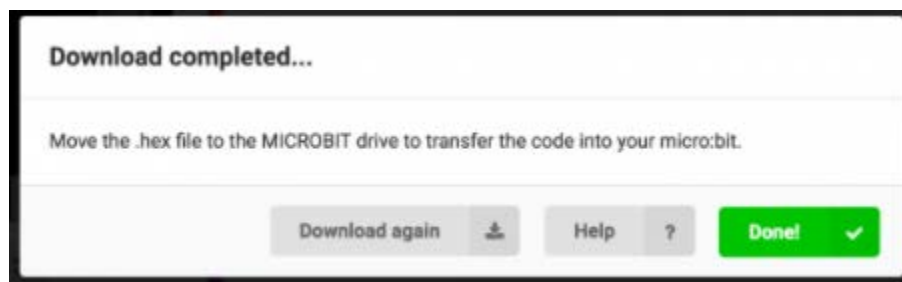


Plug the USB cable to the micro:bit

Step 22: Click the download button in the lower lefthand corner of the code window. It will downloaded most probably in the Downloads folder



Step 23: Simply click and drag your program file from its download location to your micro:bit drive, which shows up as an external device.



Step 24: When you press the button, the RGB will turn on to a color. When you press it again, the color will change and another press will change the color once again. Press it one more time, and it will turn off. Every time you press the button, it increments a variable, and then we check against it to set the color. If the variable goes over the value of 2, we reset it to 0, which is off.

Step 25: Make changes to your program by modifying the values in the analog write pins.

Congratulations! You have successfully completed this activity.

Reference: Sparkfun Inventor's Kit for micro:bit Experiment Guide

<https://learn.sparkfun.com/tutorials/sparkfun-inventors-kit-for-microbit-experiment-guide>