

MultiWingSpan

Home Programming Web Design Computer Science Twisting Puzzles Arduino BBC micro:bit

BBC micro:bit Bit:Commander - The Buzzer

Introduction

The buzzer on the Bit:Commander is connected to pin 0. This is convenient because that is the default pin. The minor inconvenience is that the potentiometer shares the same pin. Although you can use both in the same project, you need to think carefully about what is happening at any point in the program for that to make sense.



If you are using the Bit:Commander as a controller for a micro:bit game, then you have the buzzer for the game sound effects. If you are using the Bit:Commander as a controller for a project, then it's good as an alert. You can also use it to play a different tone for each button press.

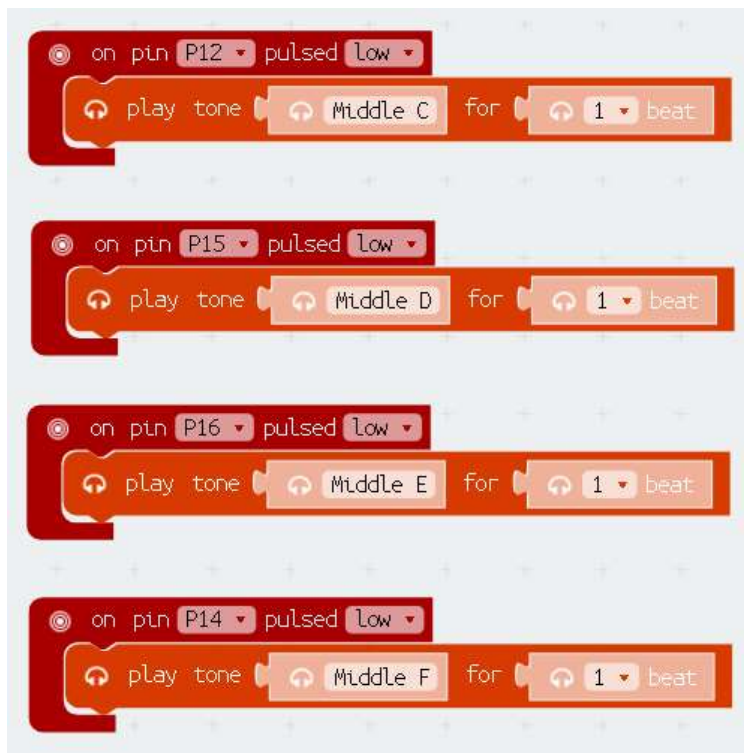
Programming

There is no need for any special treatment for the buzzer, it's just there, connected up for you on pin 0. This program plays a different tone when the buttons are pressed.

BBC Microbit

- + [Block Editor - The Basics](#)
- + [Block Editor - Components](#)
- + [Kodu - micro:bit Worlds](#)
- + [JavaScript Blocks](#)
- + [JavaScript Blocks - Exercises](#)
- + [Blocks - Bit:Bot](#)
- [Blocks - Bit:Commander](#)
 - ✧ [Bit:Commander](#)
 - ✧ [The Joystick](#)
 - ✧ [The Neopixels](#)
 - ✧ [The Potentiometer](#)
 - ✧ [The Pushbuttons](#)
 - ✧ [The Buzzer](#)
 - ✧ [Simon Game](#)
- + [MicroPython - Starting Off](#)
- + [MicroPython - Examples](#)
- + [MicroPython - Components](#)
- + [MicroPython - Breakout Boards](#)
- + [MicroPython - Exercises](#)
- + [MicroPython - Pi Accessories](#)
- + [MicroPython - Bit:Bot](#)
- + [MicroPython - Bit:Commander](#)
- + [MicroPython - Projects](#)
- + [MicroPython - Visual Basic](#)
- + [Other - Odds & Ends](#)

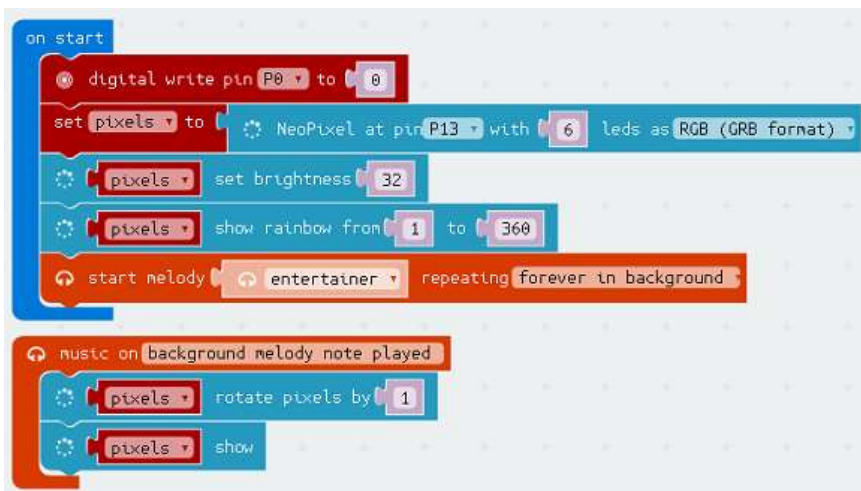




JavaScript

```
pins.onPulsed(DigitalPin.P12, PulseValue.Low, () => {
  music.playTone(262, music.beat(BeatFraction.Whole))
})
pins.onPulsed(DigitalPin.P15, PulseValue.Low, () => {
  music.playTone(294, music.beat(BeatFraction.Whole))
})
pins.onPulsed(DigitalPin.P16, PulseValue.Low, () => {
  music.playTone(330, music.beat(BeatFraction.Whole))
})
pins.onPulsed(DigitalPin.P14, PulseValue.Low, () => {
  music.playTone(349, music.beat(BeatFraction.Whole))
})
```

There are a few nice events that you can use when a melody is playing. In the following program, a rainbow pattern is lit up on the Neopixels. Every time a note plays on one of the built-in tunes, the rainbow pattern is rotated by a pixel. This creates quite a nice musical light show.



JavaScript

```
let pixels: neopixel.Strip = null
music.onEvent(MusicEvent.BackgroundMelodyNotePlayed, () => {
  pixels.rotate(1)
  pixels.show()
})
pins.digitalWritePin(DigitalPin.P0, 0)
pixels = neopixel.create(DigitalPin.P13, 6, NeopixelMode.RGB)
pixels.setBrightness(32)
pixels.showRainbow(1, 360)
music.beginMelody(music.builtInMelody(Melodies.Entertainer),
  MelodyOptions.ForeverInBackground)
```