

Lightning 'Bug'

A micro:bit makecode project (mentor guide)

In the real world, no code is perfect - especially when it's first written.

This workshop project is designed to follow on from the Firefly coding workshop, where hopefully several micro:bits were programmed to create an awesome light display for a local public event.

Firefly Code

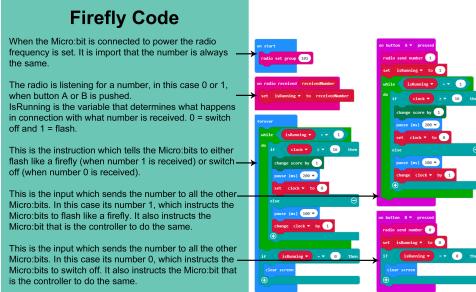
When the Micro:bit is connected to power the radio frequency is set. It is important that the number is always the same.

The radio is listening for a number, in this case 0 or 1, when button A or B is pressed.

isRadio0 is the variable that determines what happens in connection with what number is received. 0 = switch off and 1 = flash

This is the instruction which tells the Micro:bit to either flash like a firefly (when number 1 is received) or switch off (when number 0 is received).

This is the input which sends the number to all the other Micro:bits. In this case its number 1, which instructs the Micro:bits to flash like a firefly. It also instructs the Micro:bit that is the controller to do the same.



Leeds Libraries code club CoderDojo

Young coders are told that a problem has been found with the code that would allow 'Mischievous Mallory' to take control of the artistic light display.

Their challenges are:

1. To program a micro:bit with Mallory's code.
2. To rewrite the Firefly code so that Mallory's code no longer affects it.



Firefly – Coding Project

Take part in our community project to make 100s of interactive electronic fireflies on BBC micro:bits for an installation in Central Library during Leeds Light Night.

Discover the illuminating installation created by you and inspired by botanical collections from Leeds Libraries on Friday 11th October as part of the Light Night Festival.

Leeds Libraries code club CoderDojo

Mallory designed his program to use button A to start a Denial of Service (DoS) attack on the Firefly display, and button B to stop the attack.

Variables and functions should always be given meaningful names.

Even before understanding the programme can you guess what Mallory meant for the loadedSwat variable to do?

(to 'load' and 'unload' the swat)?

can you ...

```
on start
  radio set group [101]
  set [loadedSwat v] to [false v]
```

```
on button [A v] pressed
  set [loadedSwat v] to [true v]
  call startSwat [2]
```

```
function [startSwat v num v]
  while [loadedSwat v] = [true v]
    do
      radio send number [num v]
      pause (ms) [1000 v]
```

```
on button [B v] pressed
  if [loadedSwat v] = [true v] then
    set [loadedSwat v] to [false v]
    call stopSwat [1]
  +
```

```
function [stopSwat v num v]
  radio send number [num v]
```