

# Micro:Bit Christmas Worksheet

## Challenge:

Let's make a Christmas decoration that we can use at home that is so irritating it will drive your family bonkers!!

## Success criteria

- #1 plays Jingle Bells over and over and over
- #2 flashes garish lights forever
- #3 displays a Christmas message
- #4 only stops when you take out the battery!!

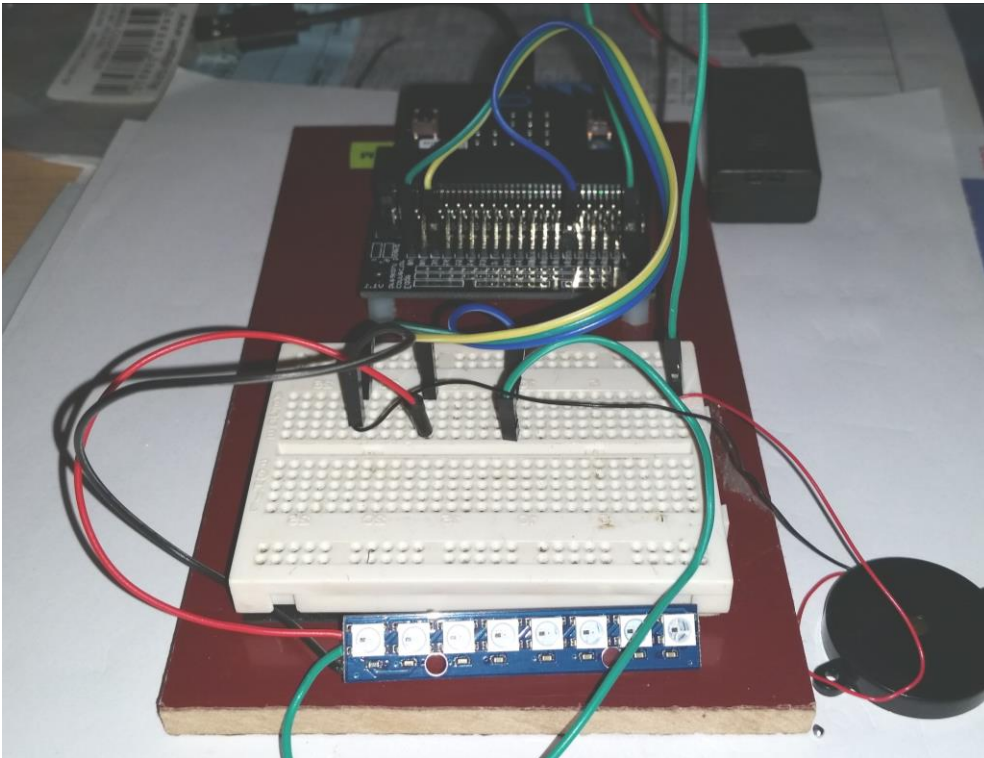
*For this program I have used a breakout board from Kitronic (£5) for simplicity of demonstration, a cheap piezo sounder and a 8 segment programmable LED from Gearbest (£2).*

*You could connect directly to the Micro:bit and power the board from a Kitronic MI:Power board (£4) for super neatness or use the 3v power pack that came with your Micro:bit*

## Wiring

In this example I have connected the micro: bit in the following way:

- 0V (Ground) to Ground of speaker and ground of programmable LED
- 3V to positive of programmable LED
- Pin 1 to LED signal pin
- Pin 0 to positive pin of speaker



## Code – This can be downloaded for the project page on the website

```
#Demonstration of playing sound, displaying text and  
#neopixel LED as three processes running together  
#D Burrin 3/10/2016
```

```
from microbit import *  
import music  
import neopixel  
from random import randint
```

```
music.set_tempo(ticks=4, bpm=120)  
# play Jingle Bells - (c) James Peirpont 1857.
```

```
jingle = [  
    'e:2','e:2','e:4',  
    'e:2','e:2','e:4',  
    'e:2','g:2','c','d',  
    'e:8',  
    'f:2','f:2','f:3',  
    'f:1','f:2','e:2',  
    'e:2','e:1','e:1',  
    'e:2','d:2','d:2',  
    'e:2','d:4','g:4',  
    'e:2','e:2','e:4',  
    'e:2','e:2','e:4',  
    'e:2','g:2','c','d',  
    'e:8','f:2','f:2','f:2',  
    'f:2','f:2','e:2',  
    'e:2','e:1','e:1',  
    'g:2','g:2','f:2',  
    'd:2','c:4'  
]
```

```
music.play(jingle,wait=False,loop=True)  
display.scroll("merry christmas",wait=False,loop=True)
```

```
#this is a direct copy of the NEOPIXEL demo on the MicroPython Documentation page
```

```
np = neopixel.NeoPixel(pin1, 8)
```

```
while True:
```

```
    for pixel_id in range(0, len(np)):
```

```
        red = randint(0, 255)
```

```
        green = randint(0, 255)
```

```
        blue = randint(0, 255)
```

```
        # Assign the current LED a random red, green and blue value between 0 and 60
```

```
        np[pixel_id] = (red, green, blue)
```

```
        # Display the current pixel data on the Neopixel strip
```

```
        np.show()
```

```
        sleep(5)
```

## New Learning code breakdown

- Create your song as a list of notes and timings.
- In the example e:2 is note E for ¼ beat and e:4 is E for ½ beat

```
jingle = [  
  'e:2','e:2','e:4',  
  'e:2','e:2','e:4',  
  'e:2','g:2','c','d',  
  'e:8',  
  'f:2','f:2','f:3',  
  'f:1','f:2','e:2',  
  'e:2','e:1','e:1',  
  'e:2','d:2','d:2',  
  'e:2','d:4','g:4',  
  'e:2','e:2','e:4',  
  'e:2','e:2','e:4',  
  'e:2','g:2','c','d',  
  'e:8','f:2','f:2','f:2',  
  'f:2','f:2','e:2',  
  'e:2','e:1','e:1',  
  'g:2','g:2','f:2',  
  'd:2','c:4'  
]
```

## Playing in the background

```
music.play(jingle,wait=False,loop=True)  
display.scroll("merry christmas",wait=False,loop=True)
```

- Wait is True by default, if this is set to False then the next line won't wait for the display or song to finish before running
- Loop is False by default, setting this to True forces the line to be repeated.

## Equivalent PXT BLOCK Code example of running music in the background

- <https://codethemicrobit.com/#pub:zvzjarrbgw>

The screenshot shows a MicroPython IDE with a project named 'Jingle'. The code is as follows:

```
set leds to Neopixel at pin P3 with 8 leds as RGB  
leds clear  
leds show  
forever  
  set r to pick random 0 to 50  
  set g to pick random 0 to 100  
  set b to pick random 0 to 255  
  leds set pixel color at 0 to red r green g blue b  
  leds show  
  leds shift pixels by 1  
run in background  
  set tempo to (bpm) 75  
  forever  
    play tone E for 1/4 beat  
    play tone E for 1/4 beat  
    play tone E for 1/2 beat  
    play tone E for 1/4 beat  
    play tone E for 1/4 beat  
    play tone E for 1/2 beat  
    play tone G for 1/4 beat  
    play tone C for 1/4 beat  
    play tone D for 1/8 beat  
    play tone E for 1 beat  
    play tone F for 1/4 beat  
    play tone F for 1/4 beat
```