

Dance Battle

Learning goals

In this lesson, we mainly learn how to control building block motor and RGB lights by micro:bit and Super:bit expansion board.

Code

```
1 from microbit import *
2 import music
3 import superbit
4 import microbit
5
6 display.show(Image.HAPPY)
7
8 while True:
9     music.play('E4:4')
10    superbit.motor_control(superbit.M1, 255, 0)
11    superbit.motor_control(superbit.M3, 255, 0)
12    music.play('E4:4')
13    superbit.motor_control(superbit.M1, 255, 0)
14    superbit.motor_control(superbit.M3, 255, 0)
15    music.play('F4:4')
16    superbit.motor_control(superbit.M1, -255, 0)
17    superbit.motor_control(superbit.M3, -255, 0)
18    music.play('G4:4')
19    superbit.motor_control(superbit.M1, -255, 0)
20    superbit.motor_control(superbit.M3, -255, 0)
21    music.play('G4:4')
22    superbit.motor_control(superbit.M1, 255, 0)
23    superbit.motor_control(superbit.M2, -255, 0)
24    music.play('F4:4')
25    superbit.motor_control(superbit.M1, 255, 0)
26    superbit.motor_control(superbit.M2, -255, 0)
27    music.play('E4:4')
28    superbit.motor_control(superbit.M1, -255, 0)
29    superbit.motor_control(superbit.M2, 255, 0)
30    music.play('D4:4')
31    superbit.motor_control(superbit.M1, -255, 0)
32    superbit.motor_control(superbit.M2, 255, 0)
33    music.play('C4:4')
```

```
34     superbit.motor_control(superbit.M1, -255, 0)
35     superbit.motor_control(superbit.M2, -255, 0)
36     music.play('C4:4')
37     superbit.motor_control(superbit.M1, -255, 0)
38     superbit.motor_control(superbit.M2, -255, 0)
39     music.play('D4:4')
40     superbit.motor_control(superbit.M1, 255, 0)
41     superbit.motor_control(superbit.M2, 255, 0)
42     music.play('E4:4')
43     superbit.motor_control(superbit.M1, 255, 0)
44     superbit.motor_control(superbit.M2, 255, 0)
45     music.play('E4:6')
46     superbit.motor_control(superbit.M1, 255, 0)
47     superbit.motor_control(superbit.M2, -255, 0)
48     music.play('D4:2')
49     superbit.motor_control(superbit.M1, 255, 0)
50     superbit.motor_control(superbit.M2, -255, 0)
51     music.play('D4:2')
52     superbit.motor_control(superbit.M1, -255, 0)
53     superbit.motor_control(superbit.M2, 255, 0)
54     microbit.sleep(500)
55     music.play('E4:4')
56     superbit.motor_control(superbit.M1, -255, 0)
57     superbit.motor_control(superbit.M2, 255, 0)
58     music.play('E4:4')
59     superbit.motor_control(superbit.M1, 255, 0)
60     superbit.motor_control(superbit.M2, 255, 0)
61     music.play('F4:4')
62     superbit.motor_control(superbit.M1, 255, 0)
63     superbit.motor_control(superbit.M2, 255, 0)
64     music.play('G4:4')
65     superbit.motor_control(superbit.M1, 255, 0)
66     superbit.motor_control(superbit.M2, 255, 0)
```

```
67     music.play('G4:4')
68     superbit.motor_control(superbit.M1, -255, 0)
69     superbit.motor_control(superbit.M2, -255, 0)
70     music.play('F4:4')
71     superbit.motor_control(superbit.M1, -255, 0)
72     superbit.motor_control(superbit.M2, -255, 0)
73     music.play('E4:4')
74     superbit.motor_control(superbit.M1, -255, 0)
75     superbit.motor_control(superbit.M2, -255, 0)
76     music.play('D4:4')
77     superbit.motor_control(superbit.M1, 255, 0)
78     superbit.motor_control(superbit.M2, -255, 0)
79     music.play('C4:4')
80     superbit.motor_control(superbit.M1, 255, 0)
81     superbit.motor_control(superbit.M2, -255, 0)
82     music.play('C4:4')
83     superbit.motor_control(superbit.M1, 255, 0)
84     superbit.motor_control(superbit.M2, -255, 0)
85     music.play('D4:4')
86     superbit.motor_control(superbit.M1, -255, 0)
87     superbit.motor_control(superbit.M2, 255, 0)
88     music.play('E4:4')
89     superbit.motor_control(superbit.M1, -255, 0)
90     superbit.motor_control(superbit.M2, 255, 0)
91     music.play('D4:6')
92     superbit.motor_control(superbit.M1, -255, 0)
93     superbit.motor_control(superbit.M2, 255, 0)
94     music.play('C4:2')
95     superbit.motor_control(superbit.M1, 255, 0)
96     superbit.motor_control(superbit.M2, 255, 0)
97     music.play('C4:2')
98     superbit.motor_control(superbit.M1, -255, 0)
99     superbit.motor_control(superbit.M2, -255, 0)
```

Import the microbit library and the superbit library;

`display.show(Image.HAPPY)`: show smile faces;

`superbit.motor_control(superbit.M1, 255, 0)`: M1 is the interface on the super:bit board, speed is 255;

`music.play('C4:4')`: Refers to the note named 'C' in octave number 4 to be played for a duration of 4.

Assembly steps

Please refer to the **Pretty car assembly steps** in the Assembly instructions folder for building blocks assembly steps.

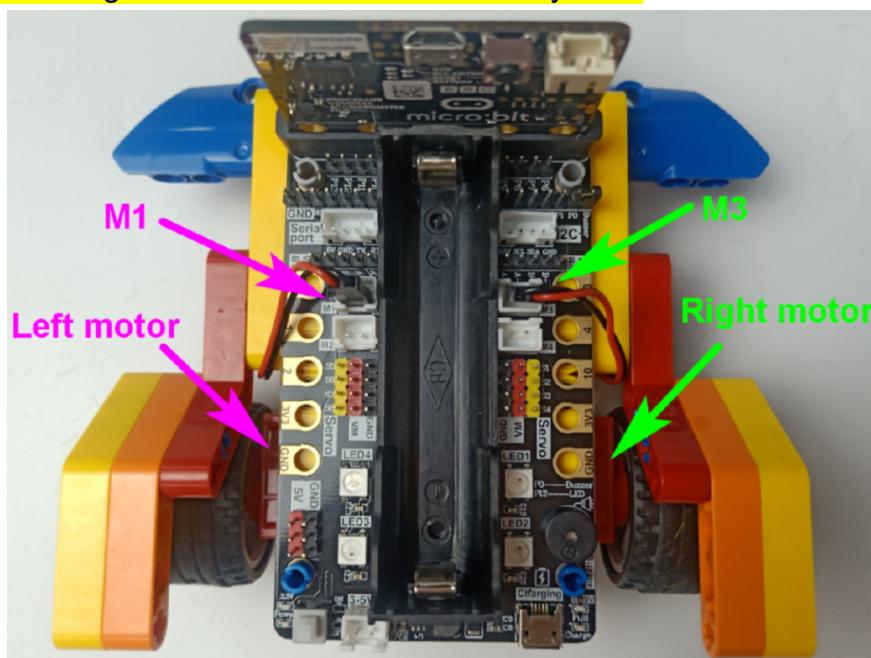
About wiring

As shown below,

Left motor connect to M1 interface of super:bit.

Right motor connect to M3 interface of super:bit.

The black wiring of the motor is near the battery side.



Programming and downloading :

1. You should open the Mu software, and enter the code in the edit window, , as shown below.

Note! All English and symbols should be entered in English, and the last line must be a space.

```

6
7 np = neopixel.NeoPixel(pin12, 2)
8 np.clear()
9 tinybit.car_HeadRGB(0, 0, 0)
10 display.show(Image.HAPPY)
11
12 item = 0

```

2. You can click the “**Check**” button to check if our code has an error. If a line appears with a cursor or an underscore, the program indicating this line is wrong.

```

6
7 np = neopixel.NeoPixel(pin12, 2)
8 np.clear()
9 tinybit.car_HeadRGB(0, 0, 0)
10 display.show(Image.HAPPY)
11
12 item = 0
13
14
15 while True:
16     voice = tinybit.getVoicedata()
17     if voice > 100:

```

3.Click “REPL” button,check whether the tinybit library has been downloaded.
If not, please refer to the [preparation before class]---> [Python programming]

```

# Write your code here :-)

BBC micro:bit REPL

MicroPython for Tinybit V1.1 Modified by Yahboom Team
Type "help()" for more information.
>>>
>>> |

```

4.Click the “Flash” button to download the program to micro:bit board.



If the program is wrong or the experimental phenomenon is wrong after downloading, please confirm whether you have downloaded the Superbit library hex file we provided to the micro:bit board.

For the specific method of adding library files, please refer to 【1.Preparation before class】---【Python programming】

Experimental phenomena

After the program is successfully downloaded, the micro:bit dot matrix will display the smile pattern and play “Happy Birthday”. The car will advance--> back-->spin left--> spin right--> . At the same time, RGB lights will change different color.

If you need to start over, press the reset button on the back of the micro:bit board.