Music Ferris wheel

Learning goals

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

Code

```
from microbit import *
import music
import superbit
import microbit
import neopixel

display.show(Image.HAPPY)
np = neopixel.NeoPixel(pin12, 4)

    
while True:
    music.play('E4:4')
    superbit.motor_control(superbit.M1, 255, 0)
    np[0] = (255, 0, 0)
    np.show()
    music.play('E4:4')
    superbit.motor_control(superbit.M1, 255, 0)
    np[1] = (0, 255, 0)
    np.show()
    music.play('F4:4')
    superbit.motor_control(superbit.M1, 255, 0)
    music.play('G4:4')
    np[2] = (0, 0, 255)
    np.show()
    superbit.motor_control(superbit.M1, -255, 0)
    music.play('G4:4')
    np[3] = (255, 255, 0)
    np.show()
    superbit.motor_control(superbit.M1, -255, 0)
    music.play('F4:4')
    np[0] = (255, 255, 255)
    np.show()
    superbit.motor_control(superbit.M1, -255, 0)
    music.play('E4:4')
    np[1] = (255, 0, 255)
    np.show()
    superbit.motor_control(superbit.M1, -255, 0)
    music.play('D4:4')
    np[2] = (0, 255, 255)
```

```
    np.show()
    superbit.motor_control(superbit.M1, 255, 0)
    music.play('C4:4')
    np[3] = (255, 0, 0)
    np.show()
    superbit.motor_control(superbit.M1, 255, 0)
    music.play('C4:4')
    np[0] = (0, 255, 0)
    np.show()
    superbit.motor_control(superbit.M1, -255, 0)
    music.play('D4:4')
    np[1] = (0, 0, 255)
    np.show()
    superbit.motor_control(superbit.M1, -255, 0)
    music.play('E4:4')
    np[2] = (255, 255, 0)
    np.show()
    superbit.motor_control(superbit.M1, 255, 0)
    music.play('E4:6')
    np[3] = (0, 255, 255)
    np.show()
    superbit.motor_control(superbit.M1, 255, 0)
    music.play('D4:2')
    np[0] = (255, 0, 0)
    np.show()
    superbit.motor_control(superbit.M1, -255, 0)
    music.play('D4:2')
    np[1] = (0, 255, 0)
    np.show()
    superbit.motor_control(superbit.M1, -255, 0)
    microbit.sleep(500)
    music.play('E4:4')
    superbit.motor_control(superbit.M1, 255, 0)
    music.play('E4:4')
    np[2] = (255, 255, 255)
    np.show()
    superbit.motor_control(superbit.M1, 255, 0)
    music.play('F4:4')
    np[3] = (255, 255, 0)
```

```
    np.show()
    superbit.motor_control(superbit.M1, -255, 0)
    music.play('G4:4')
    np[0] = (0, 0, 255)
    np.show()
    superbit.motor_control(superbit.M1, -255, 0)
    music.play('G4:4')
    np[1] = (0, 255, 0)
    np.show()
    superbit.motor_control(superbit.M1, 255, 0)
    music.play('F4:4')
    np[2] = (255, 0, 0)
    np.show()
    superbit.motor_control(superbit.M1, 255, 0)
    music.play('E4:4')
    np[3] = (255, 0, 255)
    np.show()
    superbit.motor_control(superbit.M1, -255, 0)
    music.play('D4:4')
    np[0] = (0, 255, 255)
    np.show()
    superbit.motor_control(superbit.M1, -255, 0)
    music.play('C4:4')
    np[1] = (255, 255, 0)
    np.show()
    superbit.motor_control(superbit.M1, 255, 0)
    music.play('C4:4')
    np[2] = (0, 255, 0)
    np.show()
    superbit.motor_control(superbit.M1, 255, 0)
    music.play('D4:4')
    np[3] = (255, 0, 0)
    np.show()
    superbit.motor_control(superbit.M1, -255, 0)
    music.play('E4:4')
```

```
    np[0] = (255, 255, 255)
    np.show()
    superbit.motor_control(superbit.M1, -255, 0)
    music.play('D4:6')
    np[1] = (255, 0, 255)
    np.show()
    superbit.motor_control(superbit.M1, 255, 0)
    music.play('C4:2')
    np[2] = (0, 255, 255)
    np.show()
    superbit.motor_control(superbit.M1, 255, 0)
    music.play('C4:2')
    np[3] = (0, 255, 0)
    np.show()
    superbit.motor_control(superbit.M1, -255, 0)
    microbit.sleep(1000)

```

import superbit, neopixel, music, microbit library;

display.show(Image.HAPPY): show smile pattern;
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;

np[0] = (255, 0, 0) : Set the first pixel to full brightness red;

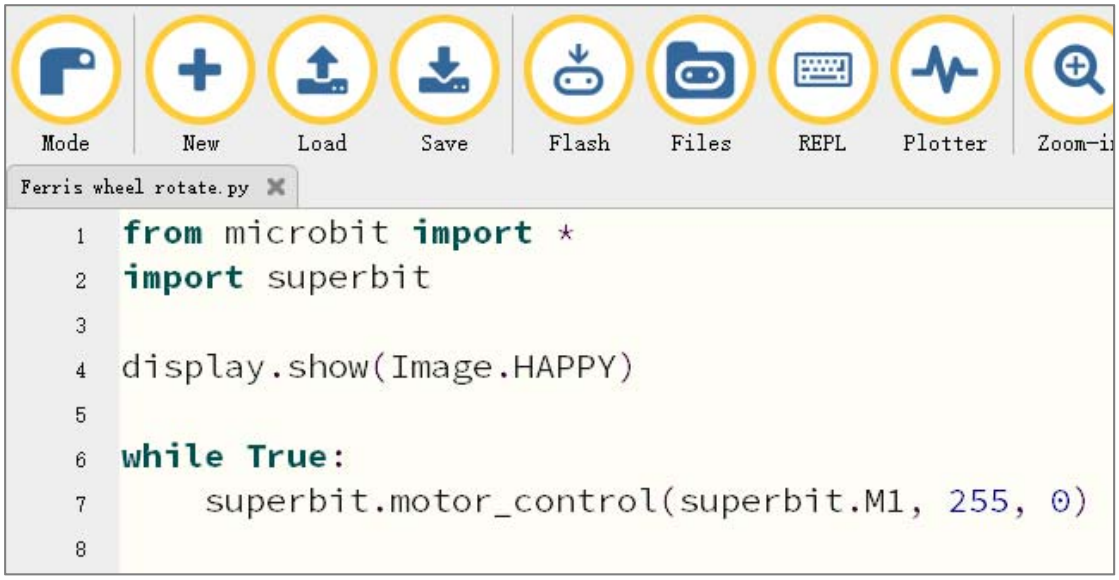
np.show(): Push the new colour data to your Neopixel strip.

About wiring

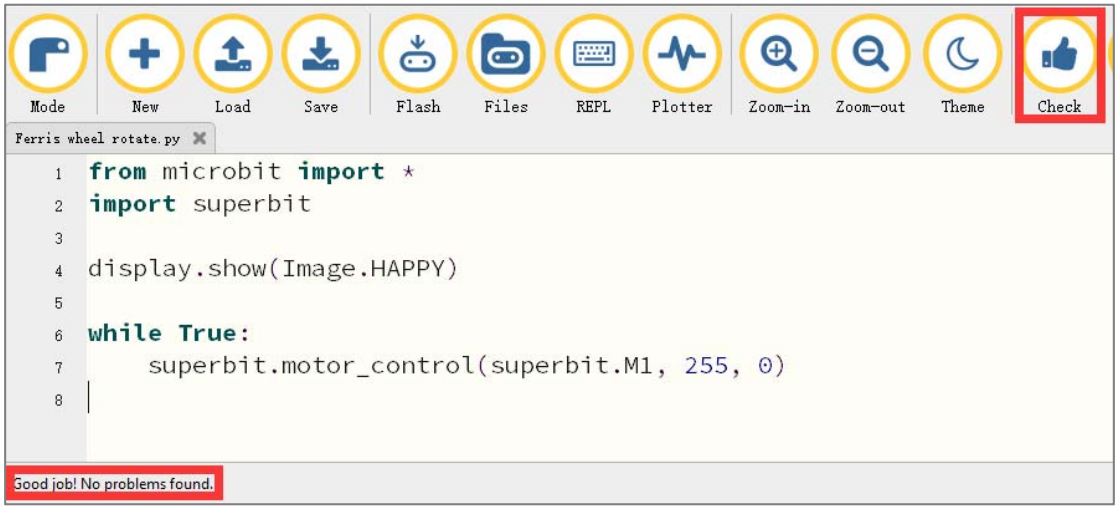
We need to connect two building block motors to the **M1** interfaces of the Super:bit expansion board.
The black wiring of the motor is near the battery side.

Programming and downloading:

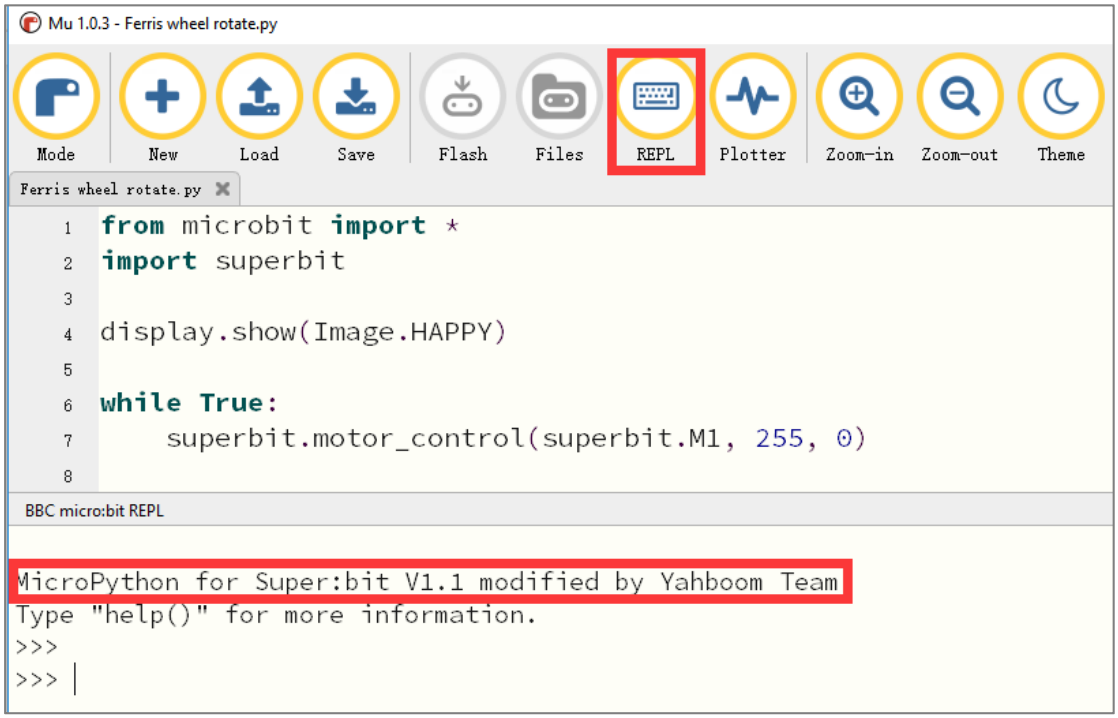
1.You should open the Mu software, and enter the code in the edit window, , as shown below.
Note! All English and symbols should be entered in English, and the last line must be a space.



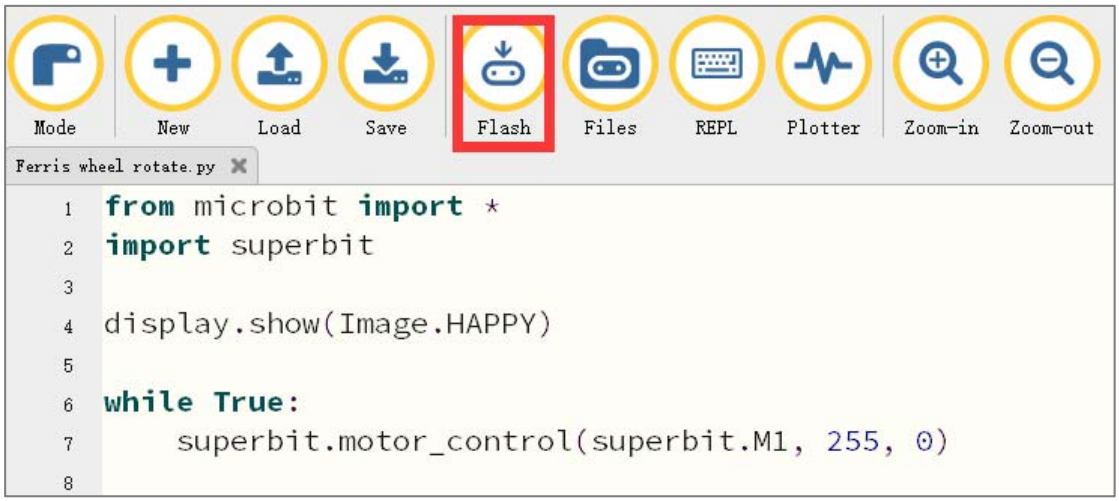
2.You can click the **"Check"** button to check if our code has an error. If a line appears with a cursor or an underscore, the program indicating this line is wrong.



3.Click the 'REPL' button to check whether the super:bit library has been downloaded. If not, please refer to the [1.preparation before class] ---> [2.How to import Yahboom superbit library] import super:bit library tutorial.



4.After writing the code, please click the 'Flash' button to download the program to the 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 【Preparation before class】---【How to import Yahboom superbit library】

Experimental phenomena

After the program is successfully downloaded, open the power, the micro:bit dot matrix will display the "smile" pattern, and the music will be played in a loop. Motor forward ---> back and RGB light switch different color.
If you need to restart, please press the reset button on the micro:bit board.