Dancing and singing

1. Learning objectives

In this course, we mainly learn how to use Python programming to make the Iron Armored Hand "sing" and "dance" at the same time, that is, the motor, servo, buzzer, and RGB light work at the same time.

2. Building blocks

For the building blocks steps, please refer to the installation drawings of [Assembly course]-[clip robot] in the materials or the building blocks installation album.

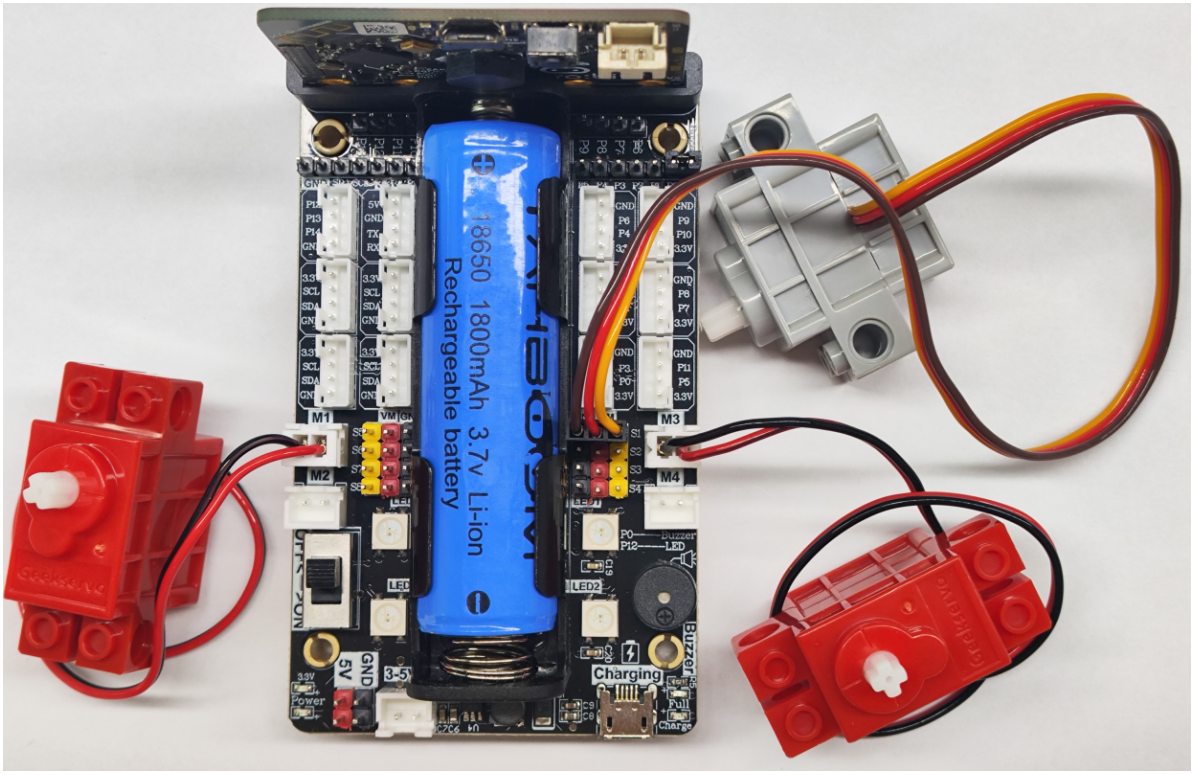
3. Motor wiring

The motor wiring on the left side of the car is inserted into the M1 interface of the Super:bit expansion board, and the black wire is close to the battery side;

The motor wiring on the right side of the car is inserted into the M3 interface of the Super:bit expansion board, and the black wire is close to the battery side;

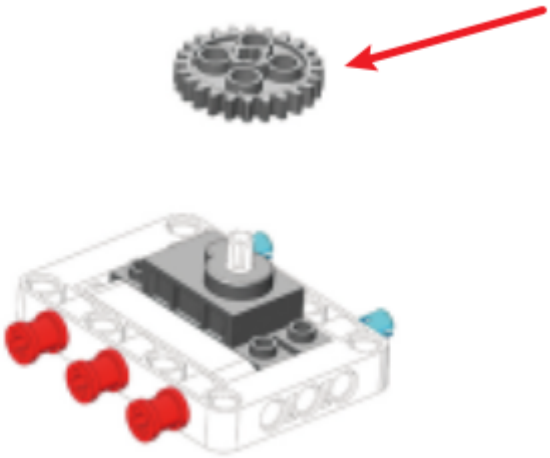
The building blocks servo wiring is inserted into the S1 interface of the Super:bit expansion board, and the orange servo wiring is inserted into the yellow pin of S1.

As shown below:



! Notes:

When conducting a course related to the building block servo for the first time, we need to remove the gear on the servo and upload the program of this course to the micro:bit; then turn on the power switch of the Super:bit expansion board and wait for the building block servo to turn to the initial position; then, we can turn off the power, adjust the clamp to open to the widest point, and then install the servo gear. (If you have used the Iron Armored Saint Hand and the servo-related program before, you can skip this step)



4. Code Analysis

For the program of this course, please see the **Dancing and singing.py** file.

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

First, import the libraries needed for this lesson from microbit: the superbit library is dedicated to the superbit expansion board; the music library is used to play music; and the neopixel is used to control the RGB light.

```
display.show(Image.HAPPY)
np = neopixel.NeoPixel(pin12, 4)
superbit.servo270(superbit.S1, 60)
```

display.show(Image.HAPPY): Display a smiley face pattern on the microbit dot matrix;

np = neopixel.NeoPixel(pin12, 4): Initialize the RGB light. There are 4 RGB lights in total, connected to the P12 pin of the microbit motherboard (you can check the hardware interface manual);

superbit.servo270(superbit.S1, 120): Initialize the building block servo to rotate to about 120°;

```
while True:
music.play('E4:4')
superbit.servo270(superbit.S1, 0)
superbit.motor_control(superbit.M1, 255, 0)
superbit.motor_control(superbit.M3, 255, 0)
np[0] = (255, 0, 0)
np.show()
...
```

while True: infinite loop

music.play('E4:4'): The buzzer plays a tone. Parameter 1 E4 represents the tone, and parameter 2 4 represents the beat.

superbit.servo270(superbit.S1, 0): The building block servo rotates to 60°;

superbit.motor_control(superbit.M1, 255, 0): The motor connected to the M1 interface rotates forward at a speed of 255;

superbit.motor_control(superbit.M3, 255, 0): The motor connected to the M3 interface rotates forward at a speed of 255;

np[0] = (255, 0, 0)

np.show(): The first RGB light lights up red.

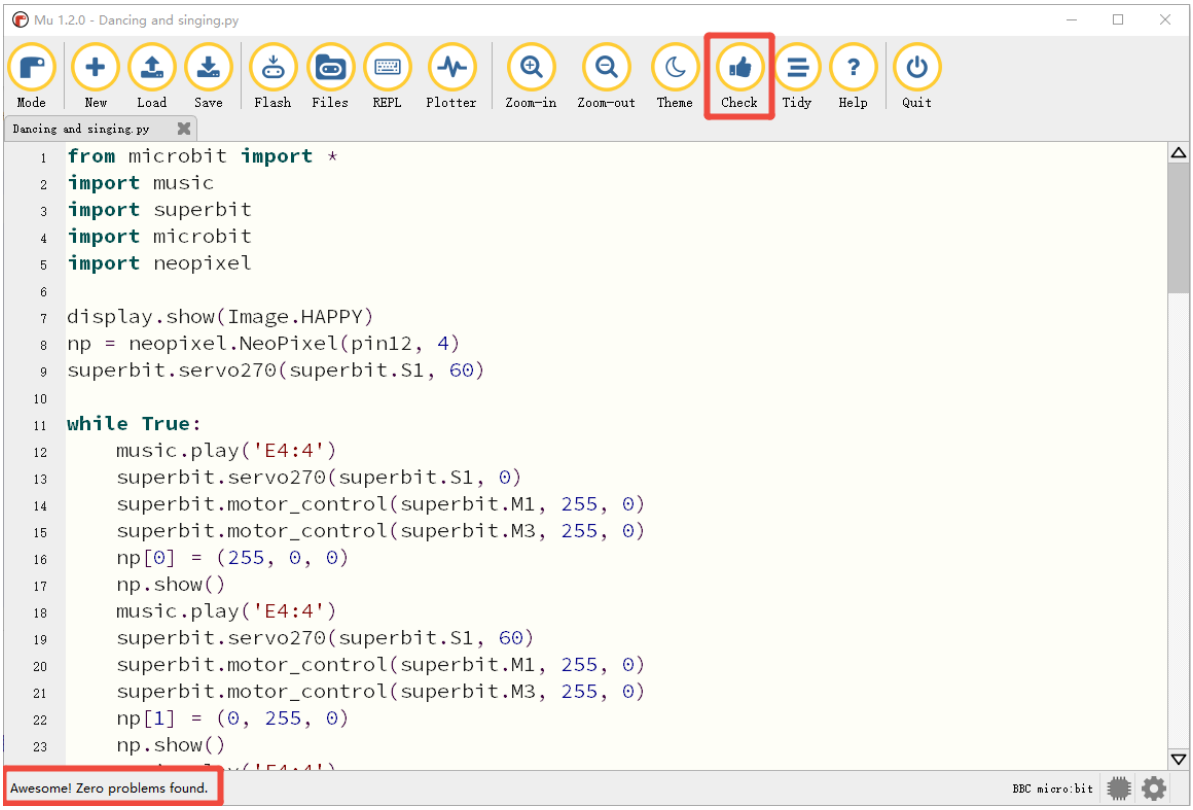
This cycle repeats.

About the tone:

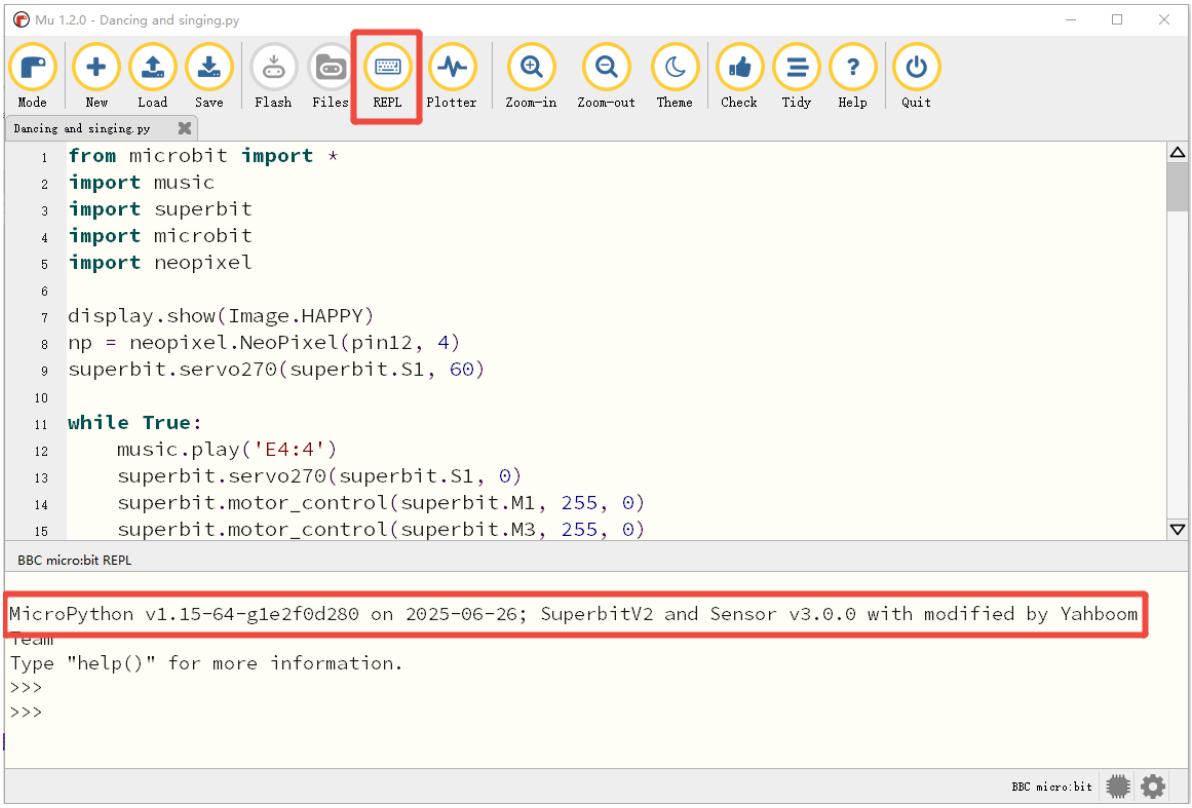
The song played here is "Ode to Joy". You can check the music score of the song online and then write the corresponding program according to the music score. For example: music.play('E4:4'), music.play('F4:8') ...

5. Write and download the program

1. Open the Mu software and enter the code in the editing window. **Note! All English and symbols should be entered in English mode, use the Tab key for indentation, and the last line ends with a blank program.**
2. Click the thumb 'Check' button to check if there are any errors in our code. If a cursor or underline appears in a line, it means a syntax error. Please check and modify it. If there is no error, the lower left corner will prompt that there is no problem with the detection.

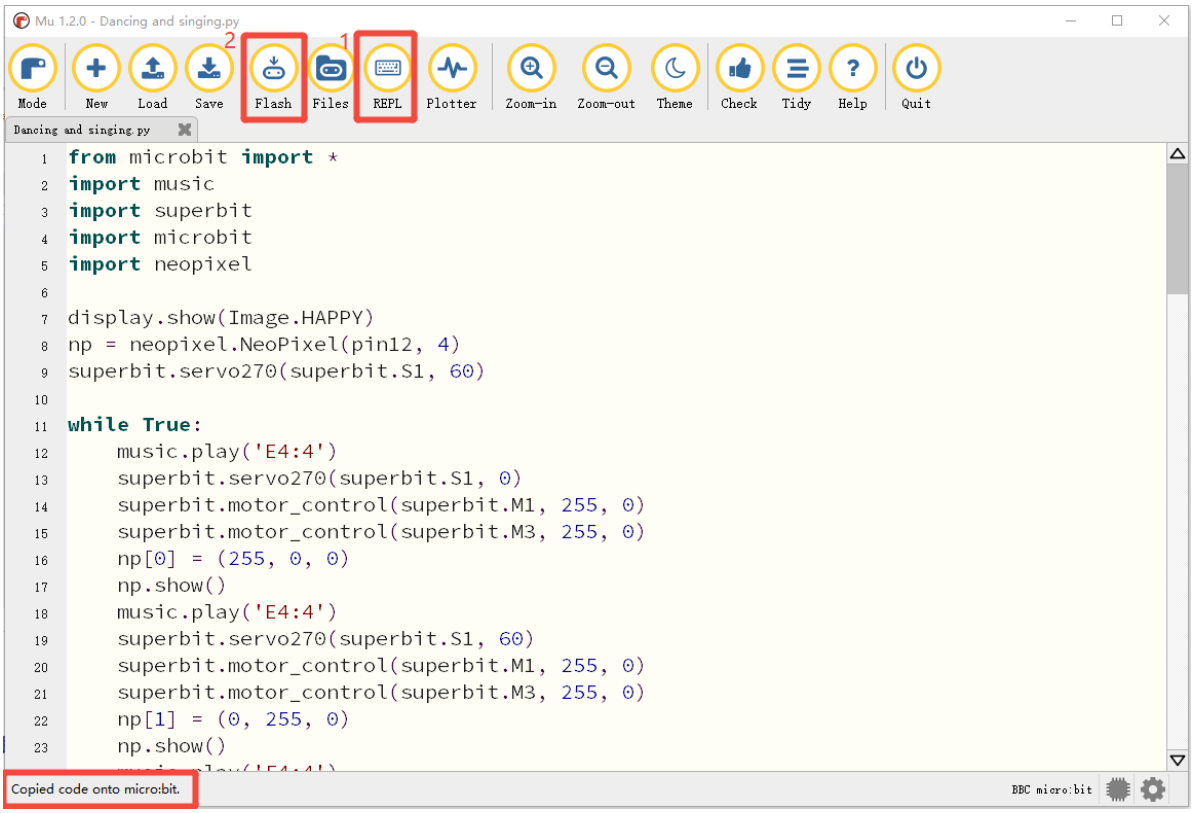


3. Click the 'REPL' button to check whether the Superbit library has been downloaded. If not, please refer to [Preparation before class] --> [2.4 Python Programming Guide].



4. After the program is written, connect the computer and the microbit mainboard with a microUSB data cable, and click the 'Flash' button to download the program to the micro:bit

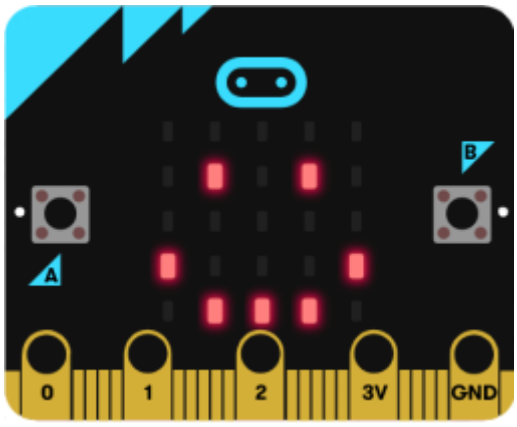
mainboard. **(You need to click the 'REPL' button again to turn off the import library file function before you can download the program normally).**



5. If the download fails, please confirm whether the microbit is properly connected to the computer via the microUSB data cable and the Superbit Python library has been imported.

6. Experimental phenomenon

After the program is successfully downloaded, the micro:bit dot matrix will display a smiley face, as shown in the figure below. Turn on the power switch, the Iron Armored Saint will play the music "Ode to Joy", and will move forward-->backward-->left-->right-->left-->right, the RGB light will switch to different colors, and the clamp will continue to tighten and loosen, and keep the cycle in this state.



If you need to restart, please press the reset button on the back of the micro:bit motherboard.