

## Throwing machine

### Learning goals

In this lesson, we mainly use the python programming language to use the micro:bit and super:bit expansion boards to control the 270 ° building block servo.

### Code

```
1  from microbit import *
2  import superbit
3
4  display.show(Image.HEART)
5  superbit.servo270(superbit.S1, 0)
6  sleep(1000)
7  while True :
8      if button_a.is_pressed():
9          superbit.servo270(superbit.S5, 60)
10     elif button_b.is_pressed():
11         superbit.servo270(superbit.S5, 0)
12
```

`from microbit import *` is to import everything from the microbit library. Every program that uses microbit must import this library. We need to import microbit library, import superbit library.

`display.show (Image.HAPPY)`: show love;

`superbit.servo270 (superbit.S1, 0)`: Rotate the 270 ° servo on S1 port to 0 °, which is the initial position of the servo.

`sleep (1000)`: 500 milliseconds delay.

`If ... elif`: A conditional statement is a block of code that is determined by the execution result (True or False) of one or more statements.

`button_a.is_pressed ()`: Press the A key on the micro:bit to execute the following statement.

`button_b.is_pressed ()`: Press the B key on the micro:bit to execute the following statement.

### 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.**

The screenshot shows the Yahboom Microbit Editor interface. At the top, there's a toolbar with various icons: Mode, New, Load, Save, Flash, Files, REPL, Plotter, Zoom-in, Zoom-out, Theme, and Check. Below the toolbar is a code editor window titled "microbit-superbit\_270servo\_one.py". The code is as follows:

```

1 from microbit import *
2 import microbit
3 import superbit
4
5 display.show(Image.HEART)
6 superbit.servo270(superbit.S1, 0)
7 microbit.sleep(1000)
8
9 while True:
10     superbit.servo270(superbit.S1, 0)
11     microbit.sleep(1000)
12     superbit.servo270(superbit.S1, 90)
13     microbit.sleep(1000)
14     superbit.servo270(superbit.S1, 180)
15     microbit.sleep(1000)
16     superbit.servo270(superbit.S1, 270)
17     microbit.sleep(1000)
18     superbit.servo270(superbit.S1, 180)
19     microbit.sleep(1000)
20     superbit.servo270(superbit.S1, 90)
21     microbit.sleep(1000)

```

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.

The screenshot shows the Yahboom Microbit Editor interface. The toolbar and code editor are identical to the previous one. However, the status bar at the bottom now displays the message "Hurrah! Checker turned up no problems." This indicates that the code has been successfully checked for errors.

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

The screenshot shows the Yahboom software interface. At the top, there is a toolbar with various icons: Mode, New, Load, Save, Flash, Files, REPL (which is highlighted with a red box), Plotter, Zoom-in, Zoom-out, and Theme. Below the toolbar, a code editor window displays the following Python script:

```

1 from microbit import *
2 import microbit
3 import superbit
4
5 display.show(Image.HEART)
6 superbit.servo270(superbit.S1, 0)
7 microbit.sleep(1000)

```

Below the code editor is a BBC micro:bit REPL window containing the following text:

MicroPython for Super:bit V1.1 modified by Yahboom Team  
Type "help()" for more information.  
>>>  
>>> |

4. After writing the code, please click the 'Flash' button to download the program to the micro:bit board.

The screenshot shows the Yahboom software interface. The toolbar at the top has the Flash icon highlighted with a red box. Below the toolbar, a code editor window displays the same Python script as before:

```

1 from microbit import *
2 import microbit
3 import superbit
4
5 display.show(Image.HEART)
6 superbit.servo270(superbit.S1, 0)
7 microbit.sleep(1000)
8
9 while True:
10     superbit.servo270(superbit.S1, 0)
11     microbit.sleep(1000)
12     superbit.servo270(superbit.S1, 90)
13     microbit.sleep(1000)

```

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】---【How to import Yahboom superbit library】

### Hardware connection

The 270° block servo connect to the S1 interface of the Super:bit expansion board. The orange wire of the 270° block servo is connected to the yellow pin

of S5, the red wire of the 270° block servo is connected to the red pin of S5, and the brown wire of the 270° block servo is connected to the black pin of S5.

### Experimental phenomena

After the program is downloaded successfully, the micro:bit dot matrix will show heart pattern; press the A key it will throw out the object, press the B key it will return to the home position.

If you need to restart, press the reset key on the back of the micro:bit board.