Automatic Lift

1. Learning objectives

In this course, we mainly learn how to use Python programming to make the lift platform automatically rise and fall.

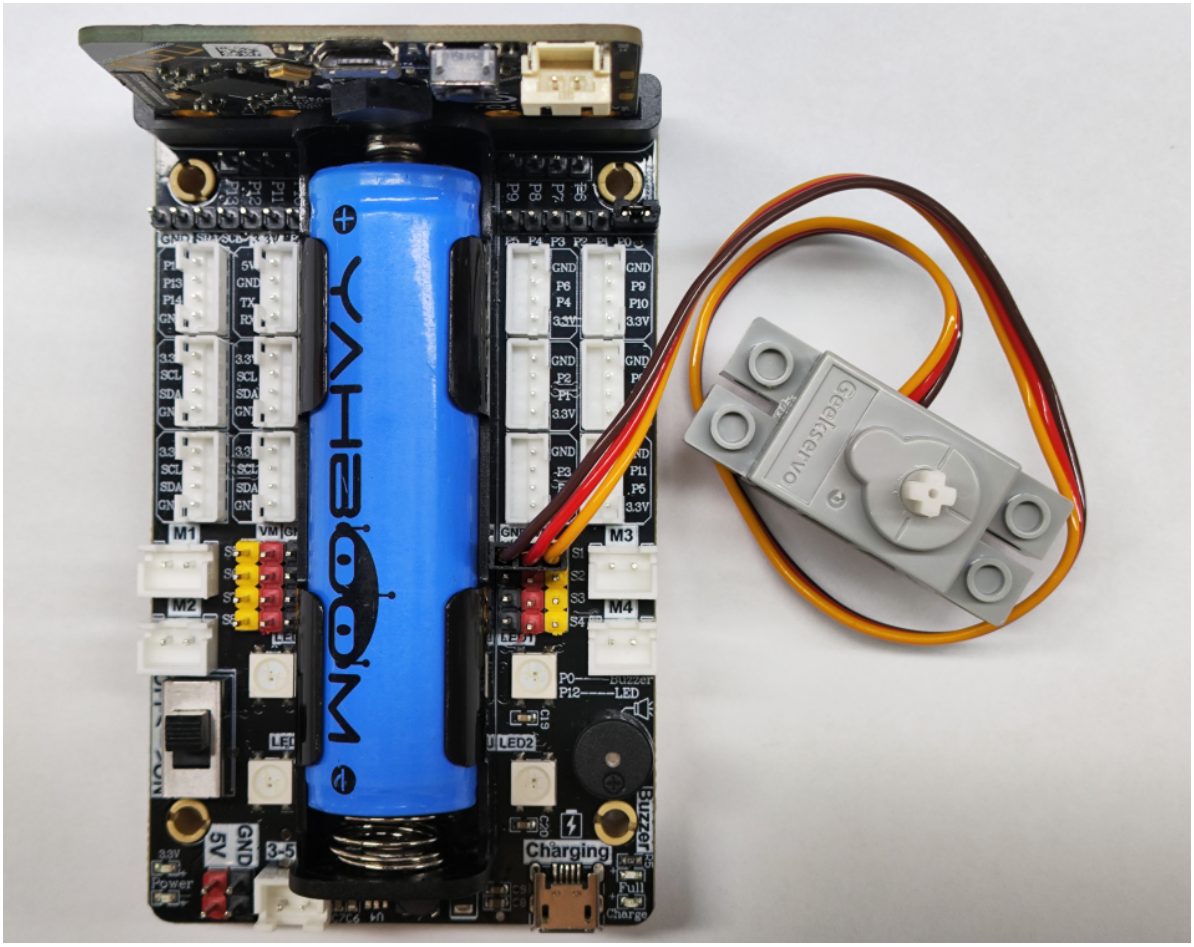
2. Building blocks

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

3. Motor wiring

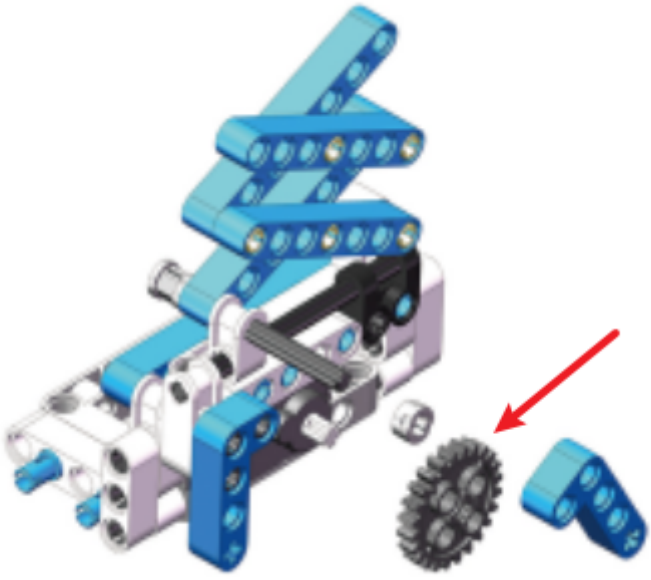
The building blocks servo wiring is inserted into the S1 interface of the Super:bit expansion board, and the orange servo wiring is inserted into the yellow pin of S1.

As shown in the figure below:



! Notes:

When taking 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 lifting platform to the lowest position, as shown in the figure below, and then install the servo gear. (If you have used the lifting platform and servo-related programs before, you can skip this step)



4. Code analysis

For the program of this course, please see the **Automatic Lift.py** file.

```
from microbit import *
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; neopixel is used to control the RGB light.

```
display.show(Image.HAPPY)
np = neopixel.NeoPixel(pin12, 4)
superbit.servo270(superbit.S1, 90)
```

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, 90): Initialize the building block servo to rotate to about 90°;

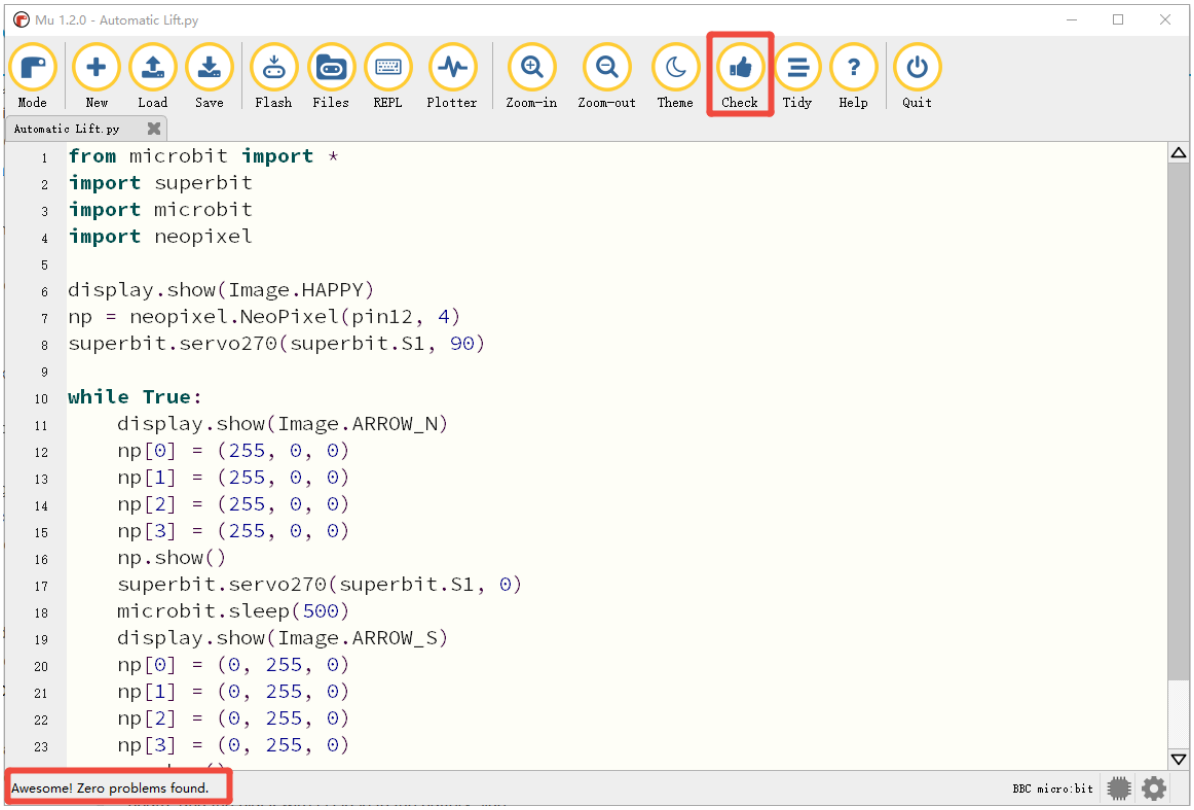
```
while True:
display.show(Image.ARROW_N)
np[0] = (255, 0, 0)
np[1] = (255, 0, 0)
np[2] = (255, 0, 0)
np[3] = (255, 0, 0)
np.show()
superbit.servo270(superbit.S1, 0)
microbit.sleep(500)
display.show(Image.ARROW_S)
np[0] = (0, 255, 0)
np[1] = (0, 255, 0)
np[2] = (0, 255, 0)
np[3] = (0, 255, 0)
np.show()
superbit.servo270(superbit.S1, 90)
microbit.sleep(1000)
```

while True: infinite loop

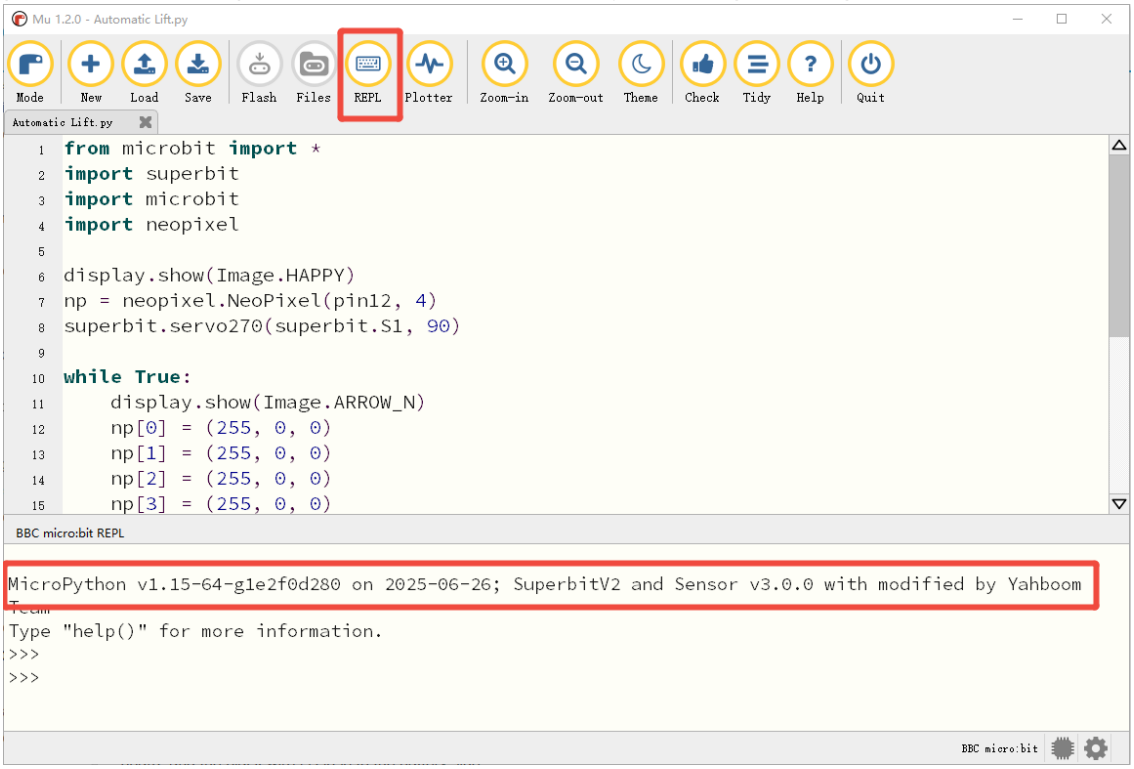
An upward arrow pattern is displayed on the micro:bit dot matrix, the RGB light is red, and the lifting platform rises; after 500 milliseconds, we can see a downward arrow pattern on the micro:bit dot matrix, the RGB light is green, and the lifting platform descends.

5. Write and download the program

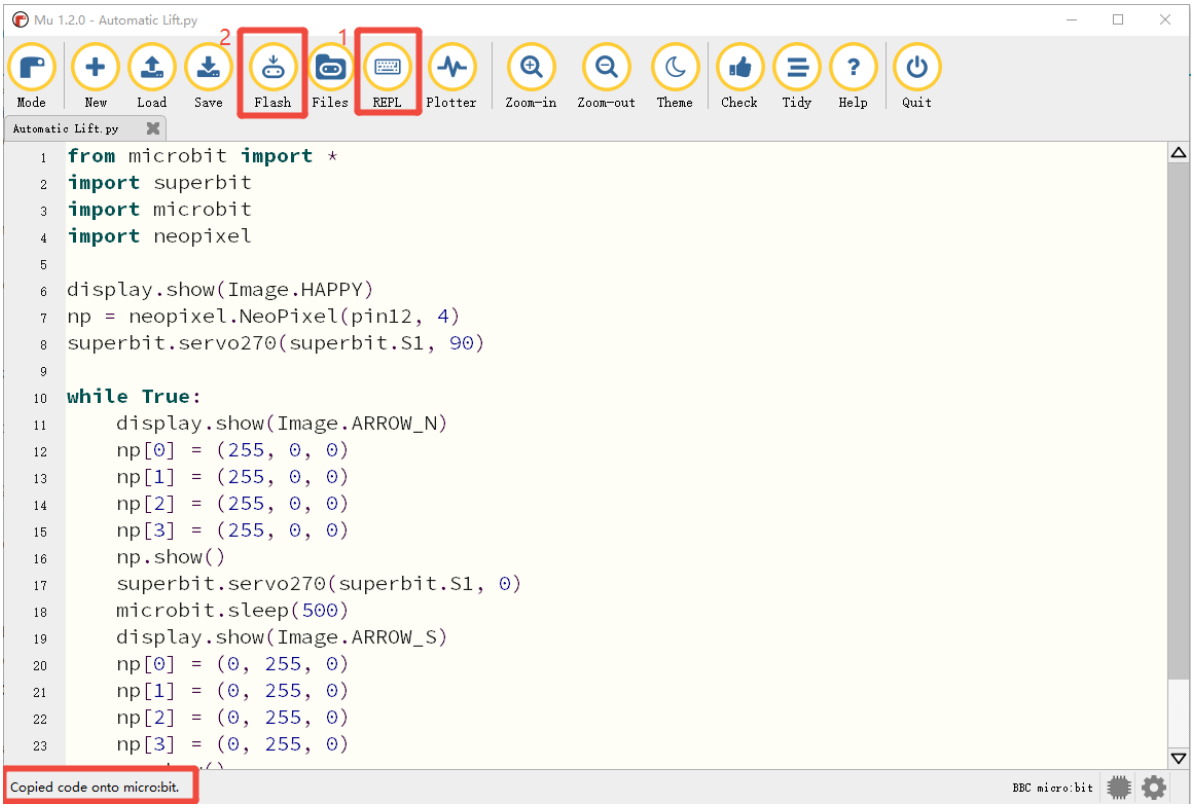
1. Open the Mu software and enter the code in the editing window. **Note! All English and symbols should be entered in English mode, use the Tab key (tab key) for indentation, and the last line ends with a blank program.**
2. Click the thumb 'Check' button to check if there are any errors in our code. 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 connected to the computer normally via the microUSB data cable and the Python library of Superbit has been imported.

6. Experimental phenomenon

After the program is successfully downloaded, the micro:bit dot matrix will display a heart. Turn on the power switch, the servo will initialize to 0° (the lifting platform descends). After one second, we can see an upward arrow pattern on the micro:bit dot matrix, the RGB light turns red, and the lifting platform rises; after 500 milliseconds, we can see a downward arrow pattern on the micro:bit dot matrix, the RGB light turns green, and the lifting platform descends. And keep looping in this state.

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