

Clumsy reptile

1. Learning goals

In this course, we mainly learn how to use Python programming to make the unicycle movement.

The principle of unicycle walking:

We can change the friction of the front wheels by adjusting the 1# bolt connection snap ratchet to control the direction of the unicycle.

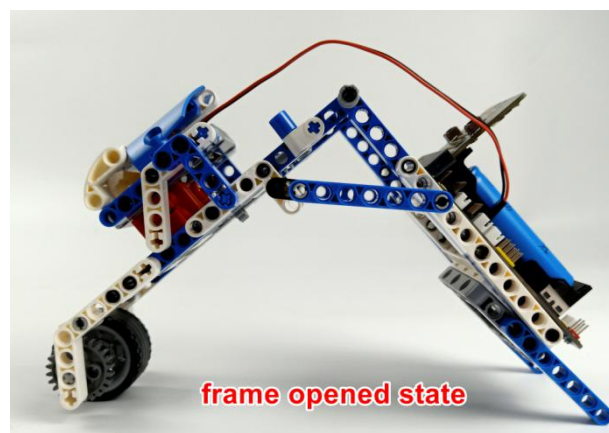
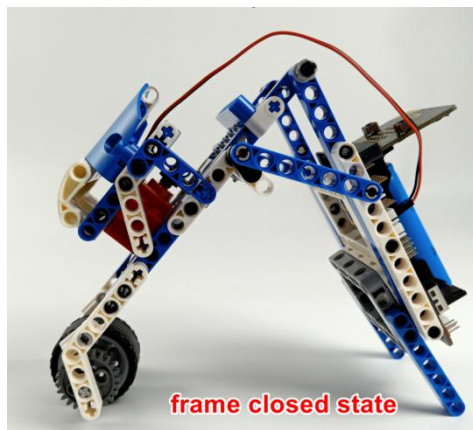
When the 1# bolt connector is located on the front side of the 24-tooth gear, the front wheel can only move forward, so the unicycle forward;

When the 1# bolt connector is located on the rear side of the 24-tooth gear, the front wheels can only move backwards, and the unicycle backwards.

2. Building block assembly steps

For the building block construction steps, please refer to the installation manual or building block installation picture of [Assembly course]-[Unicycle].

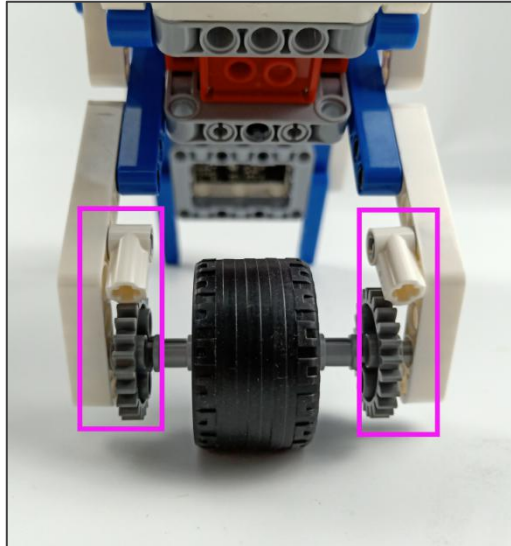
After the assembly is completed, the frame of the unicycle needs to be adjusted to a closed state.



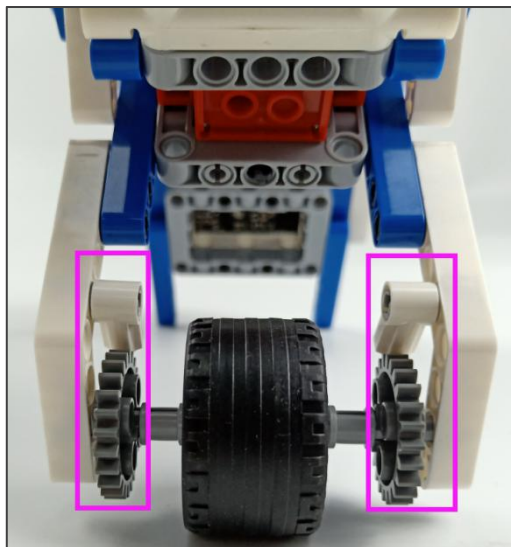
! Note

When 1# bolt connector are placed in front of the 24-tooth gear, the unicycle can move forward.

When 1# bolt connector are placed behind the 24-tooth gear, the unicycle can move backwards.



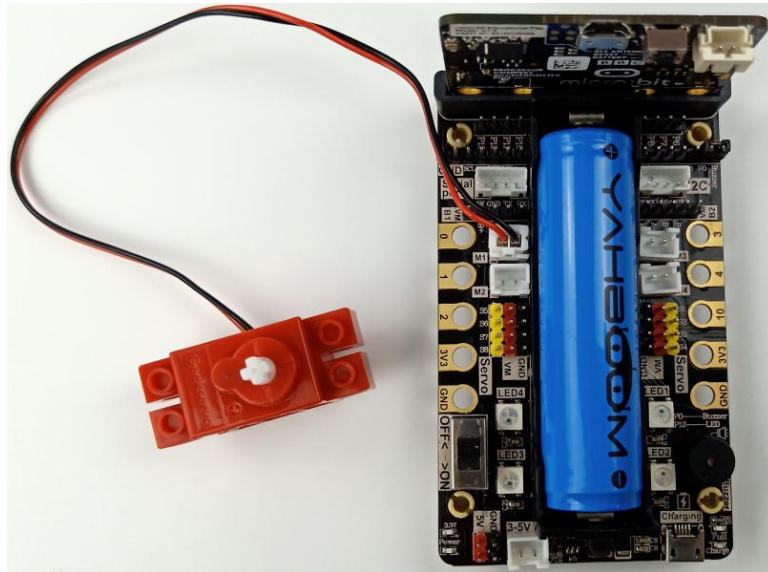
【1# bolt connector are placed in front of the 24-tooth gear】



【1# bolt connector are placed behind the 24-tooth gear】

3.Wiring of motor

The motor wiring is inserted into the M1 interface of the Super:bit expansion board, and the black wire is close to the battery side;
As shown below.



4.Code and analysis

The program for this course, please view .py file.

```
from microbit import *
import superbit
```

First, we need to import the library needed for this lesson from micro:bit, superbit library is dedicated to super:bit expansion board.

```
5 display.show(Image.HAPPY)
```

display.show(Image.HAPPY): Display the smile pattern on the micro:bit matrix;

```
while True:
    superbit.motor_control(superbit.M1, 255, 0)
```

while True: loop

superbit.motor_control(superbit.M1, -255, 0): The motor of the M1 interface is rotate with 255 speed;

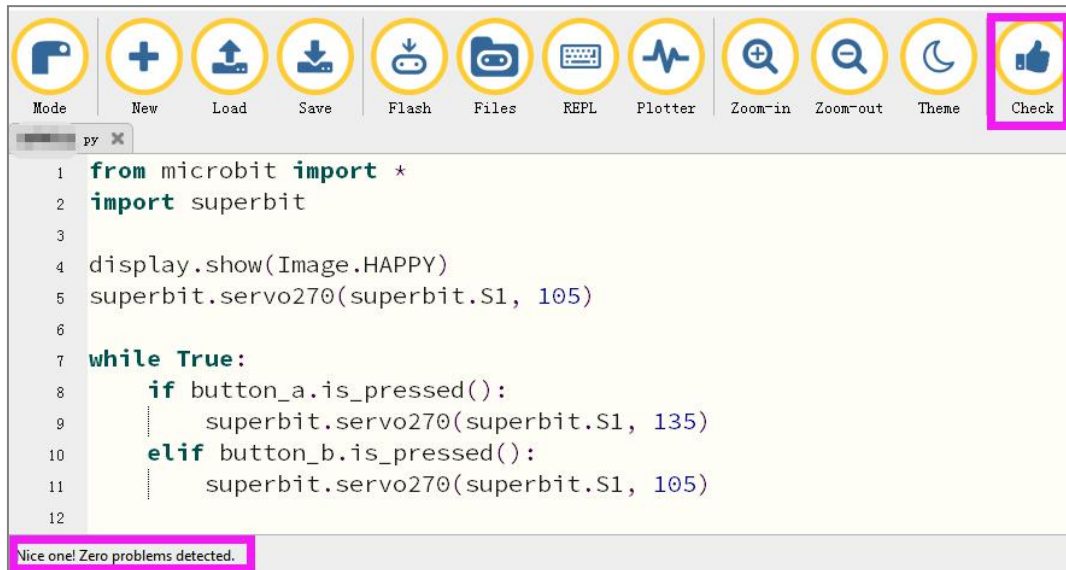
5.Writing and download code

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, use the Tab key (tab key) to indent and the last line must be a space.

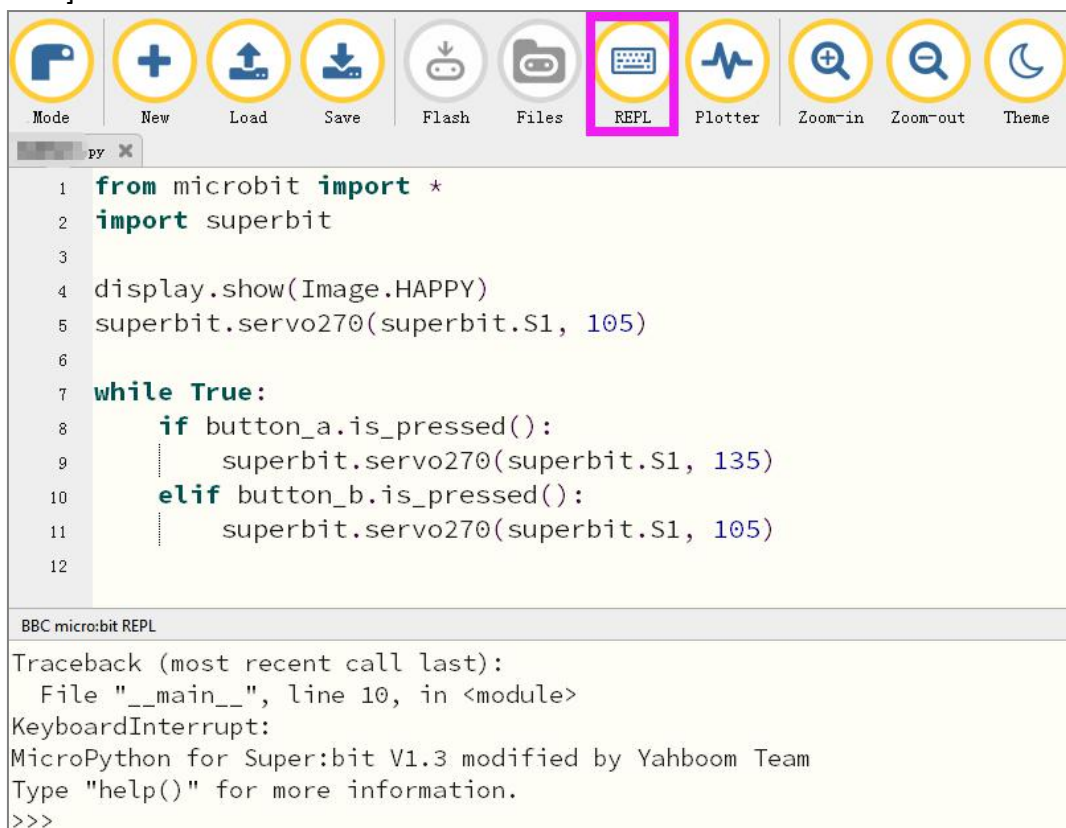
2.You can click the “Check” button to check if our code has an error.

If a cursor or underline appears on a line, it indicates a syntax error, please check and modify. If there is no error in the program, the bottom left of the interface will prompt that there is no problem in detection.

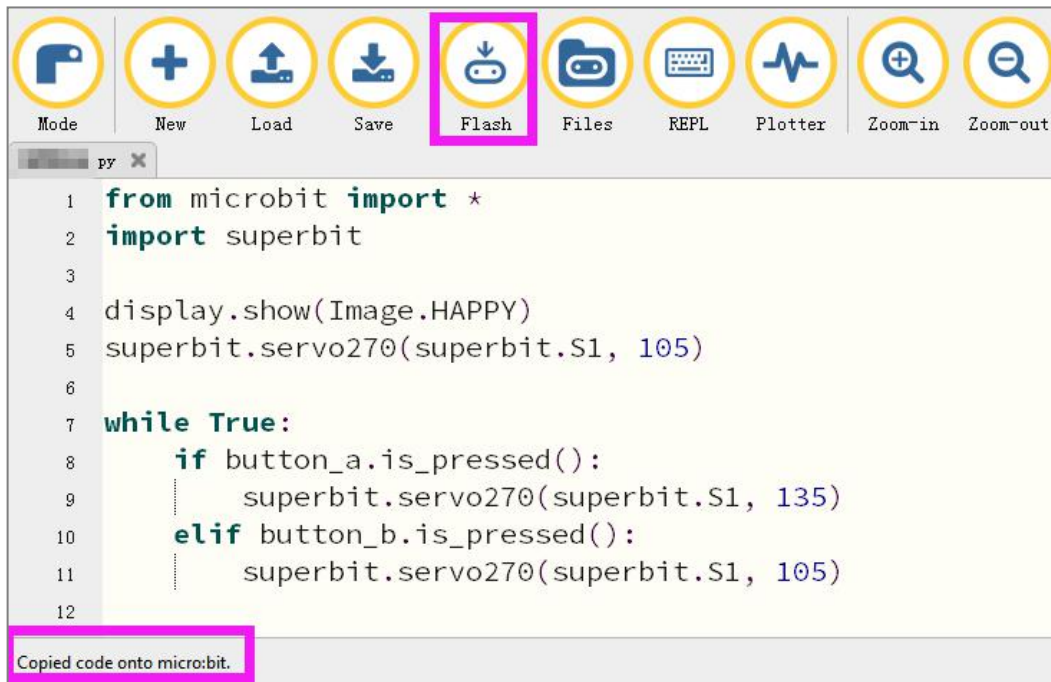


3. Click the 'REPL' button to check whether the Superbit library has been downloaded.

If not, please refer to [Preparation before class] --> [2.4 Python Programming Guide] .



4. After the program is written, use a micro USB cable to connect the computer and the micro:bit board. Please click the 'Flash' button to download the program to the micro:bit motherboard (You need to click the 'REPL' button again to close the function of importing library files before you download the program).



5.If the download failed, please confirm whether the micro:bit is connected to the computer through the micro USB data cable, and confirm whether the Super:bit Python library has been imported.

6.Experimental phenomena

After the program is successfully downloaded, the micro:bit dot matrix will display the smile pattern.

Case 1: If we put two 1# bolt connector in front of the 24-tooth gear, the car will move forward.

Case 2: If we put two 1# bolt connections behind the 24-tooth gear, the car will retreat.