Lesson 4 The Movement on XYZ Axes

1. Preparation

1.1 Software

Please connect MaxArm to Python editor according to the tutorial in folder "4. MaxArm Underlying Program Learning/Python Development/Lesson 1 Set Development Environment".

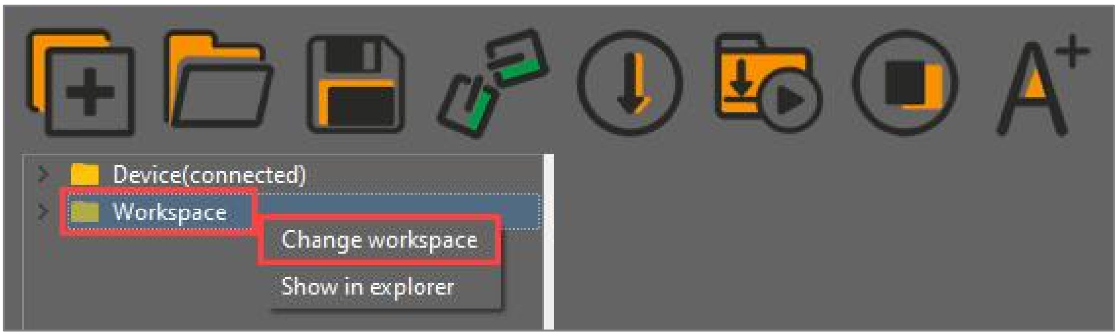
2. Working Principle

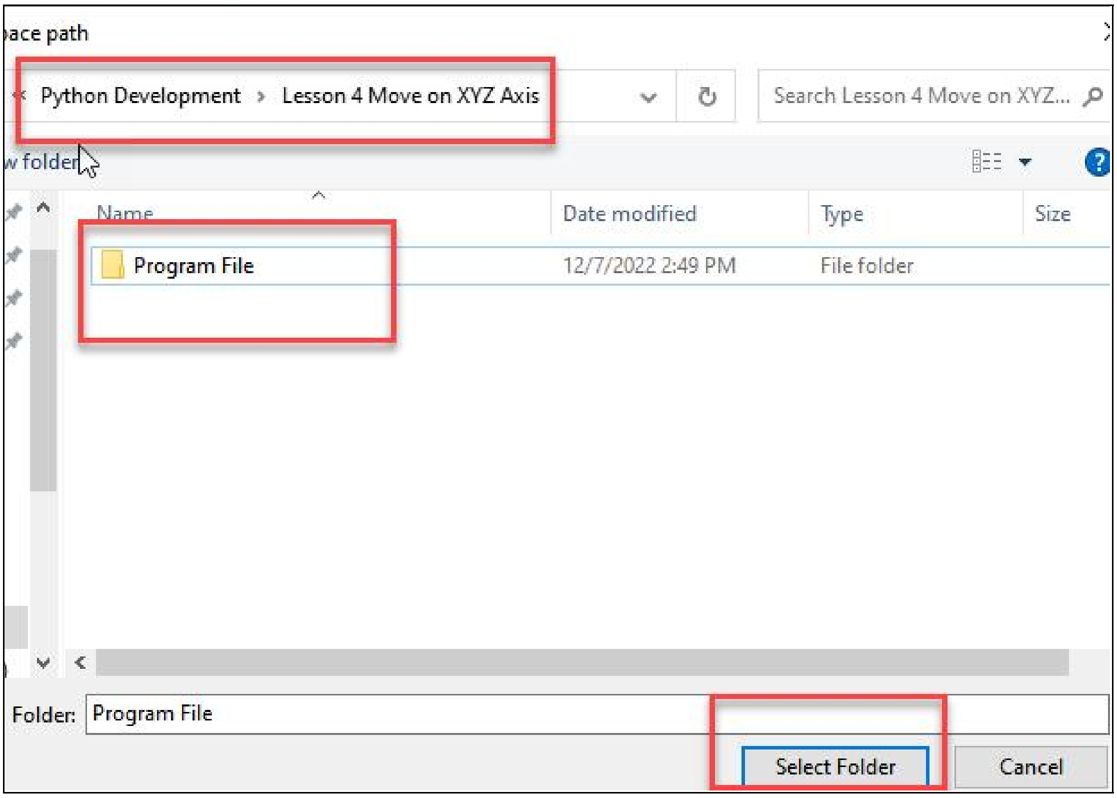
Call the function related to the kinematics by calling "espmax" kinematics encapsulation library to realize the movement of robotic arm on xyz axes.

The path to the kinematics encapsulation library is "Appendix/8.Underlying File/Python Development/espmax.py"

3. Program Download

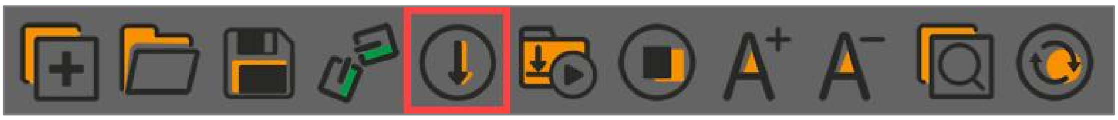
After connecting, change the path of Workspace to " 8. Inverse Kinematics Basic and Application/ Lesson 4 The Movement on XYZ Axes/Program File".





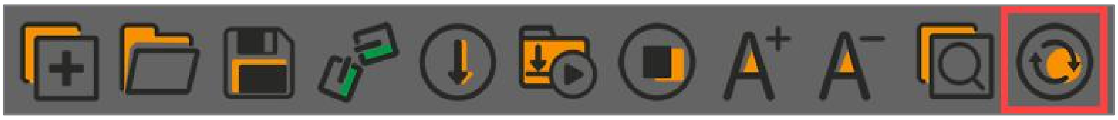
1) Double click “Workshop” folder, and then double click “main.py” to open the program.

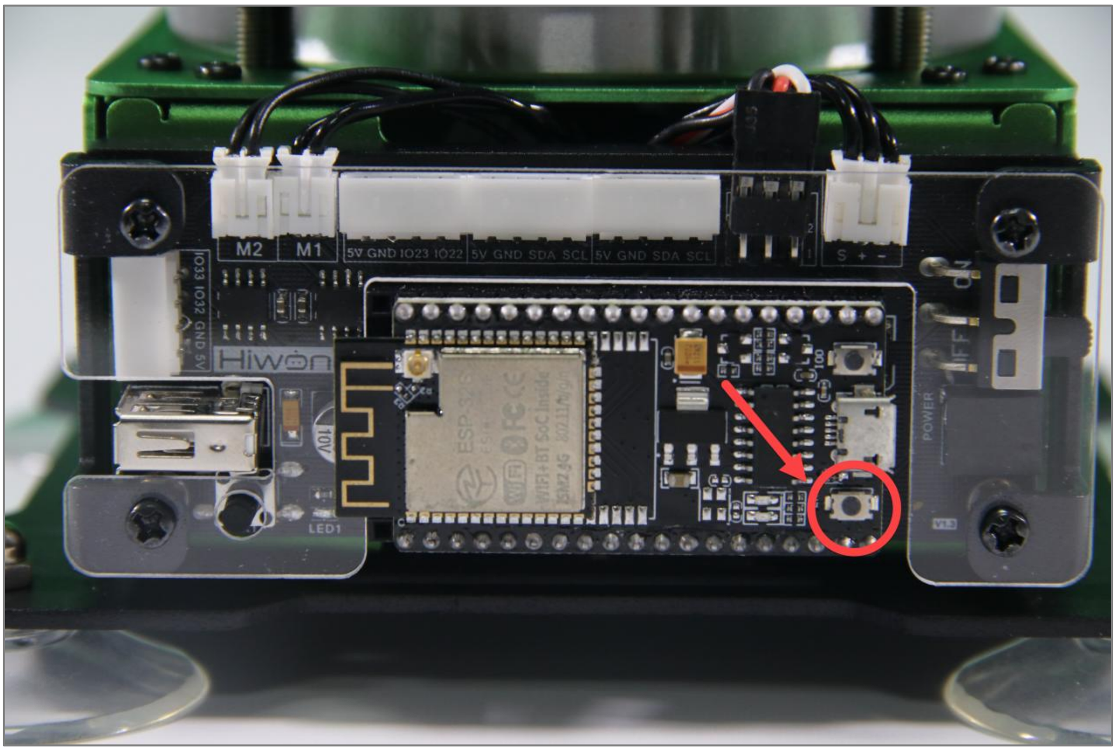
2) Then right click to download the program files to the controller.



3) When the terminal prints the prompt as shown in the image below, it means download completed.

4) After downloading, click on the reset icon or press the reset button on ESP32 controller to run program.





4. Project Outcome

When running the program, the robotic arm will move up and down along z axis, then move to the left and the right along x axis, and then move forwards and backwards along y axis. After the program stops, exit automatically.

5. Program Instruction

5.1 Import Library File

Before the robotic arm starts to move, the Python function libraries related to the kinematics and bus servo need to be imported.

```
import time
from espmax import ESPMax
from BusServo import BusServo
```

5.2 Read Position

Use arm.ORIGIN function of the bus servo to read the xyz position corresponding to the initial position of the robotic arm.

```
(x,y,z) = arm.ORIGIN
print(x,y,z)
```

5.3 Control Robotic Arm

Start to move on xyz axes.

```
arm.set_position((x,y,z-100),2000) #
time.sleep_ms(2000)
arm.set_position((x,y,z-50),1000)  #
time.sleep_ms(1000)

arm.set_position((x-50,y,z-50),1000)
time.sleep_ms(1000)
arm.set_position((x+50,y,z-50),2000)
time.sleep_ms(2000)
arm.set_position((x,y,z-50),1000)
time.sleep_ms(1000)

arm.set_position((x,y-50,z-50),1000)
time.sleep_ms(1000)
arm.set_position((x,y+50,z-50),2000)
time.sleep_ms(2000)
arm.set_position((x,y,z-50),1000)
time.sleep_ms(1000)
```

Take the code “arm.set_position((x,y,z-100),2000)” as example:

The first parameter “(x,y,z-100)” represents the position of the suction nozzle on xyz axes.

The second parameter “200” represents the running time and the unit is ms.