Lesson2 Testing MG996R Servo

Table

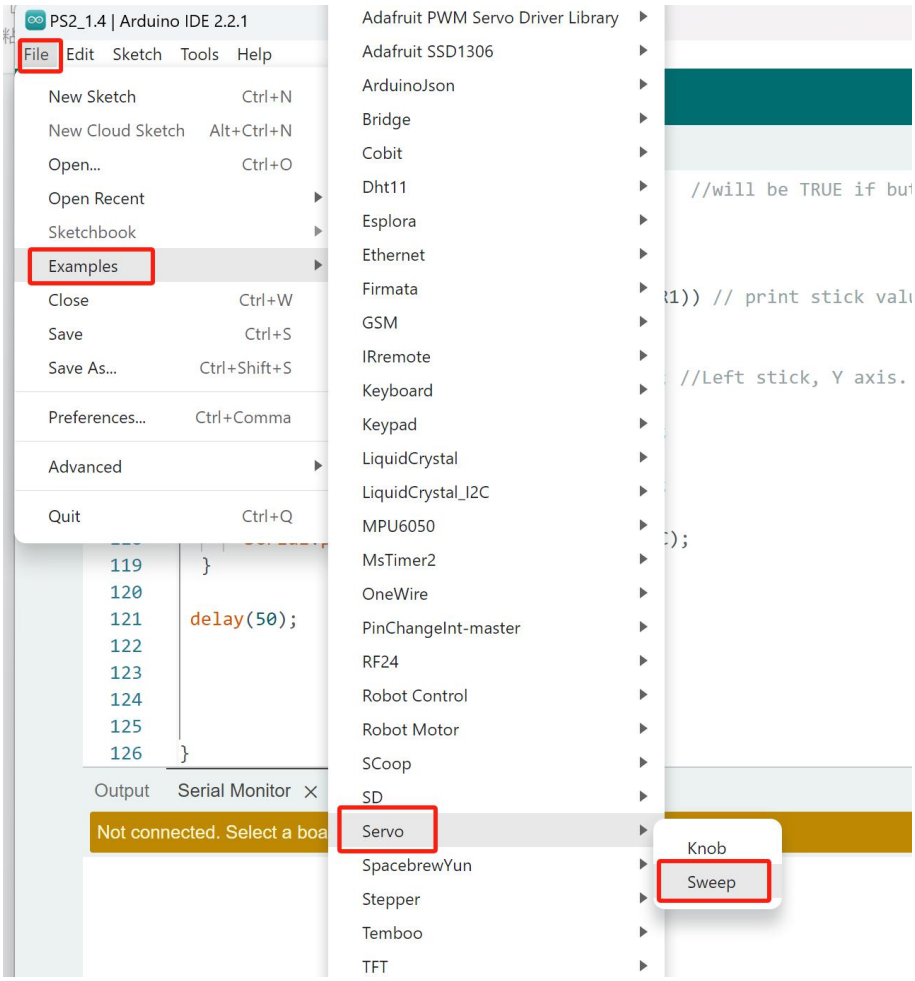
1.Testing MG996R Servo by Arduino

1.1 Set up the environment for Arduino to run

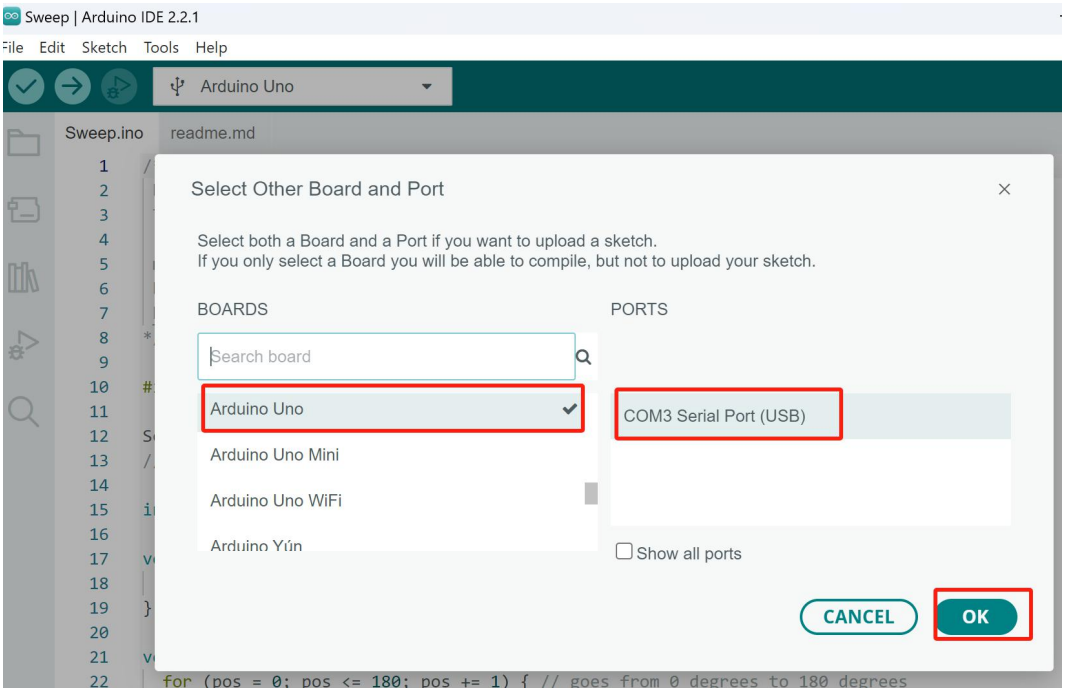
Please refer to the documentation
“CKK0016-main\Tutorial\Configuring the Operating Environment For Arduino”

1.2 Upload the servo test program to the UNO board

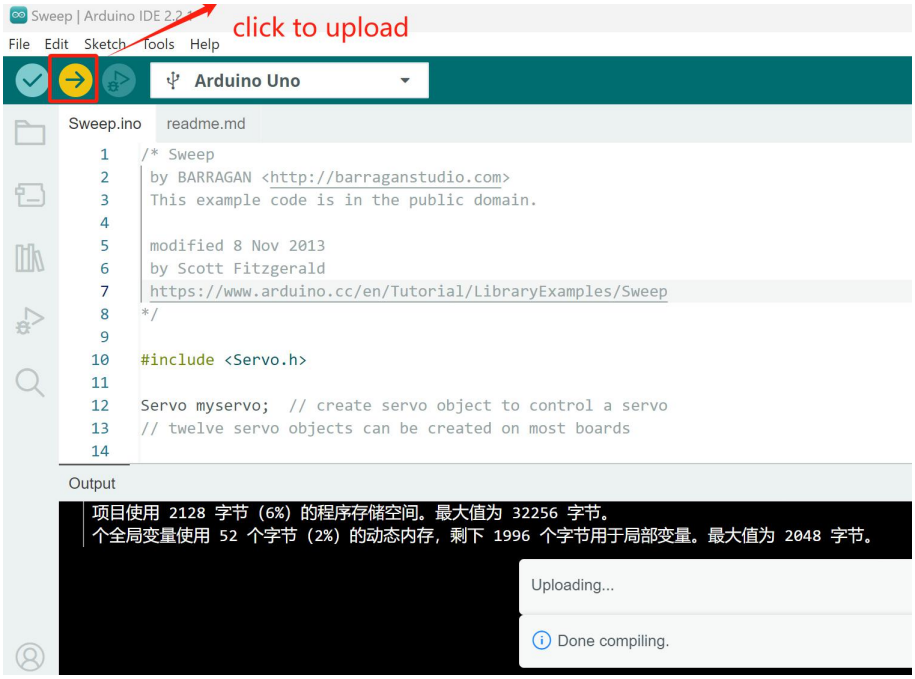
After the Arduino IDE is installed, double-click the Arduino IDE shortcut on the desktop to start the IDE interface, and then click "File--Examples--Servo--Sweep" to enable the sample program



Connect the Arduino UNO board to the computer through a USB cable, click the Select Other Board and Port menu to select the Arduino Uno and Serial ports corresponding to the UNO board. This example here is com3.

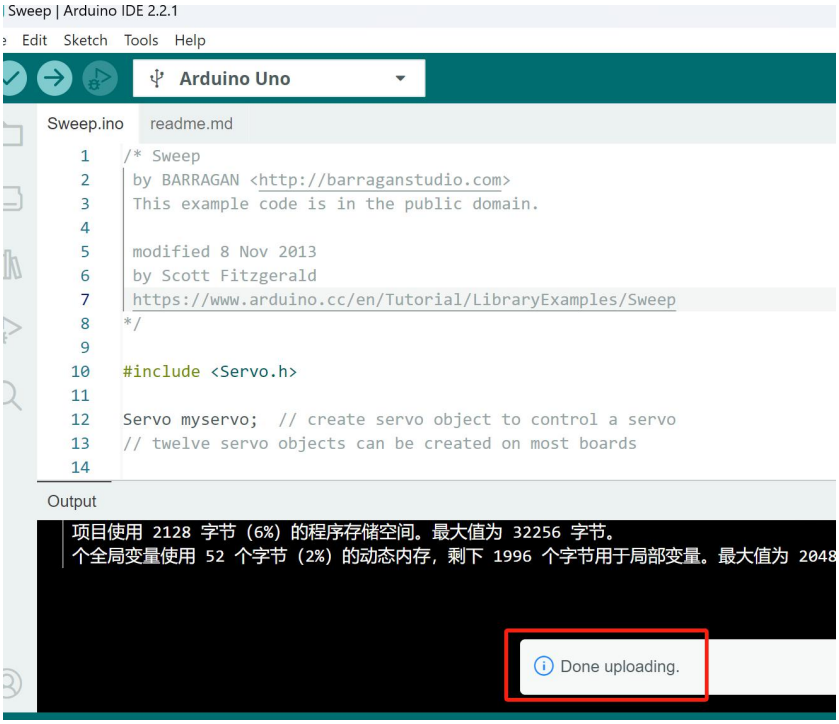


Click the "upload" button on the IDE interface to start uploading the Sweep program to the UNO board.



Support : cokoino@outlook.com

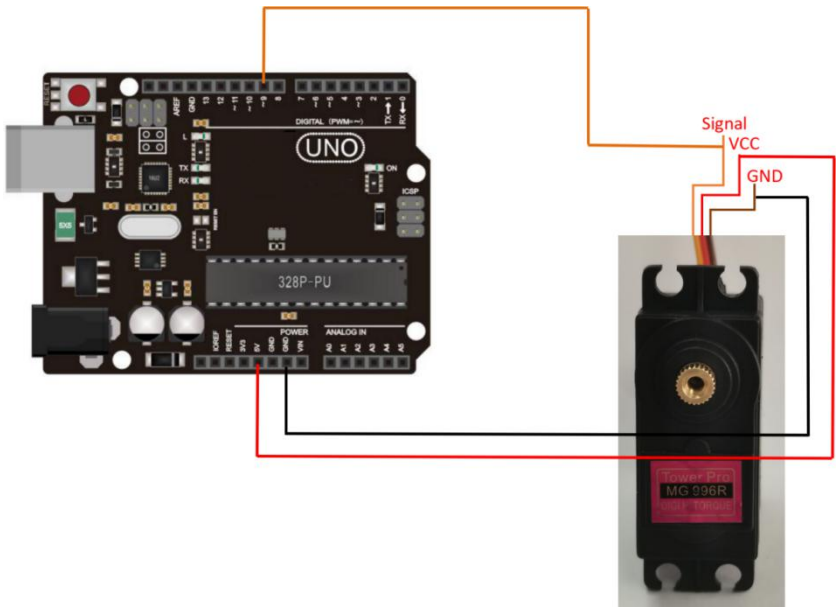
"done uploading" will be displayed when the program upload is completed.



1.3 Test the Servo with the Uno R3

1.3.1 Connect the yellow signal pin of the MG996R Servo to pin9 of the UNO board through a DuPont cable, connect the red VCC pin of the MG996R Servo to the 5V of the UNO board, and connect the brown GND pin of the MG996R Servo to the GND of the UNO board.

Circuit connection diagram:



Support : cokoino@outlook.com

The corresponding connection relationship is shown in the table below:

UNO R3 PIN	MG996R PIN
9	Signal
5v	VCC
GND	GND

1.3.2 You can directly power the UNO board through the USB cable connection, or you can use two 18650 batteries to power it. After the code is powered on, you can see that the MG996R Servo rotates 180 degrees in a loop.

1.3.3 Code

```
#include <Servo.h>
Servo myservo;
int pos = 0;
void setup() {
  myservo.attach(9);
}
void loop() {
  for (pos = 0; pos <= 180; pos += 1) {
    myservo.write(pos);
    delay(15);
  }
  for (pos = 180; pos >= 0; pos -= 1) {
    myservo.write(pos);
    delay(15);
  }
}
```

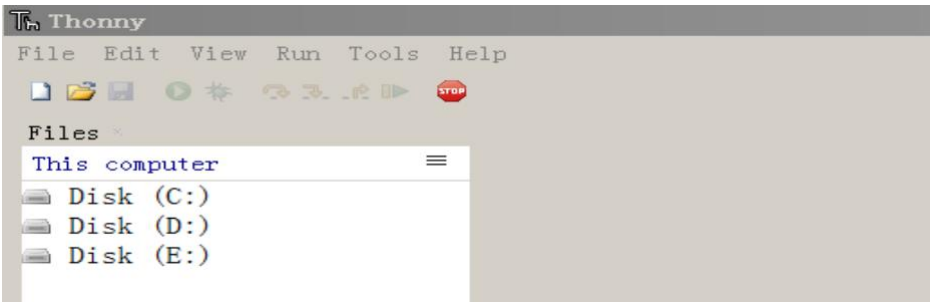
2. Testing MG996R Servo by Pico

2.1 Set up an environment for the Pico program to run

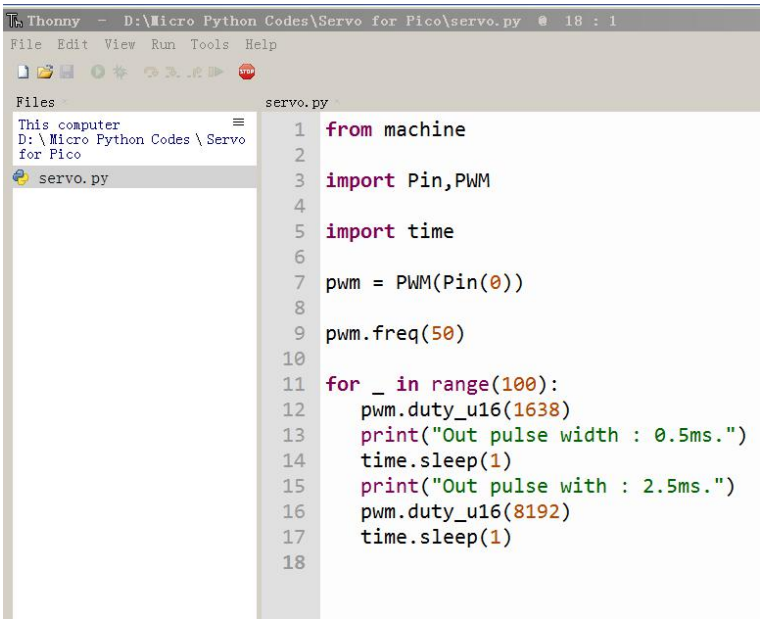
Please refer to the documentation “CKK0016-main\Tutorial\Configuring the Operating Environment For Micro Python”

2.2 Upload the “Servo for Pico” code to the Pico board

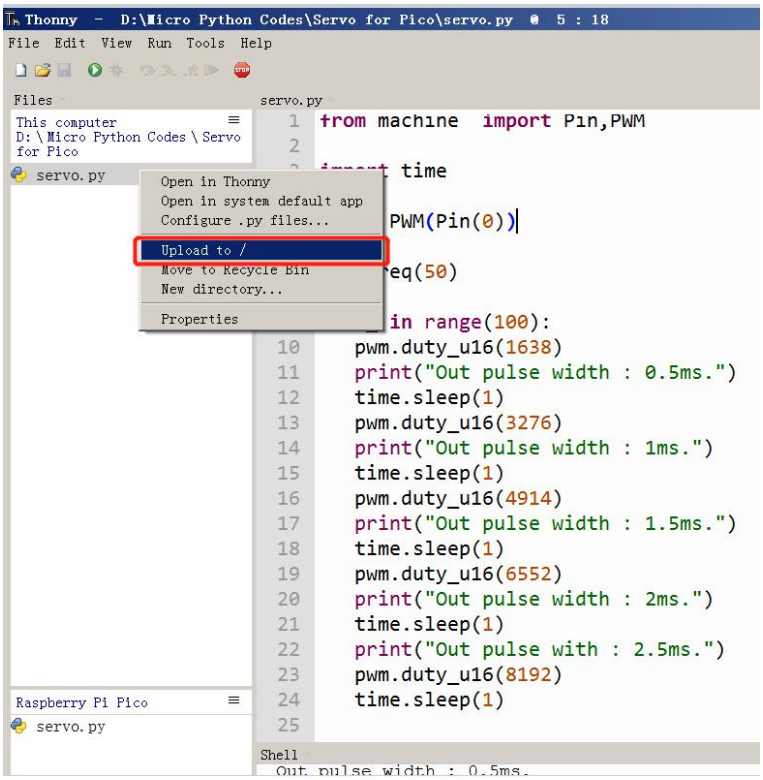
2.2.1 We provide a servo.py file for running offline. The code added tomain.py is a bootstrap that executes the user's code file. All you need to do is upload the offline project's code file (.py) to the Raspberry PiPico device.Move the program folder “CKK0016-main\Tutorial\Python\Python_Codes” to disk(D) in advance with the path of “D:/Micro Python Codes”. Open “Thonny”。



2.2.2 Expand “servo for pico” in the “Micro Python Codes” in the directory of disk(D), and double-click main.py, which is provided by us to enable programs in “MicroPython device” to run offline.

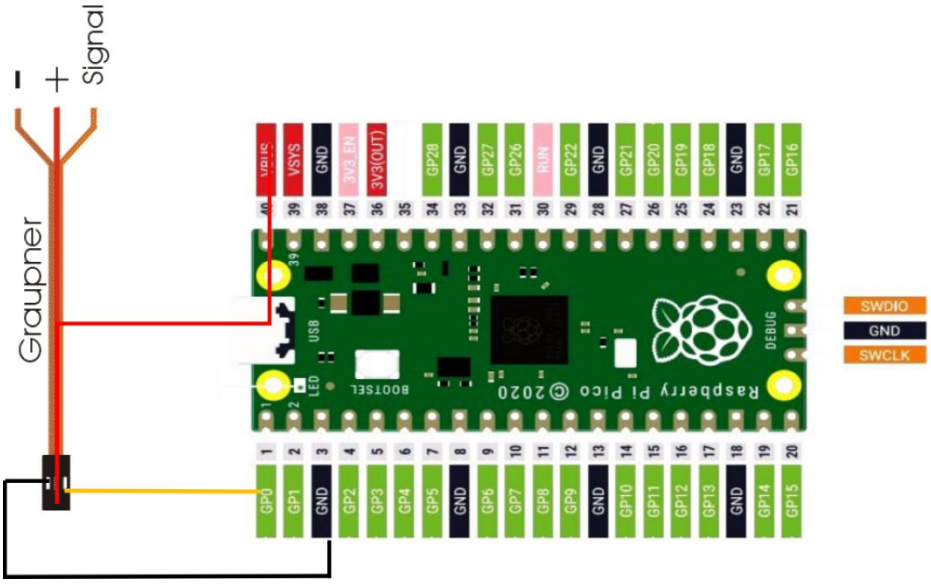


2.2.3 As shown in the following illustration, right-click the file servo.py and select “Upload to /” to upload code to Raspberry Pi Pico.



2.3 Connect the MG996R Servo to the Pico board

2.3.1 As shown in the following picture, the signal pin of MG90S Micro Servo is connected to GP0 of Pico, the VCC(+) pin of MG90S Micro Servo is connected to VBUS of Pico, and the GND(-) pin of MG90S Micro Servo is connected to GND of Pico.



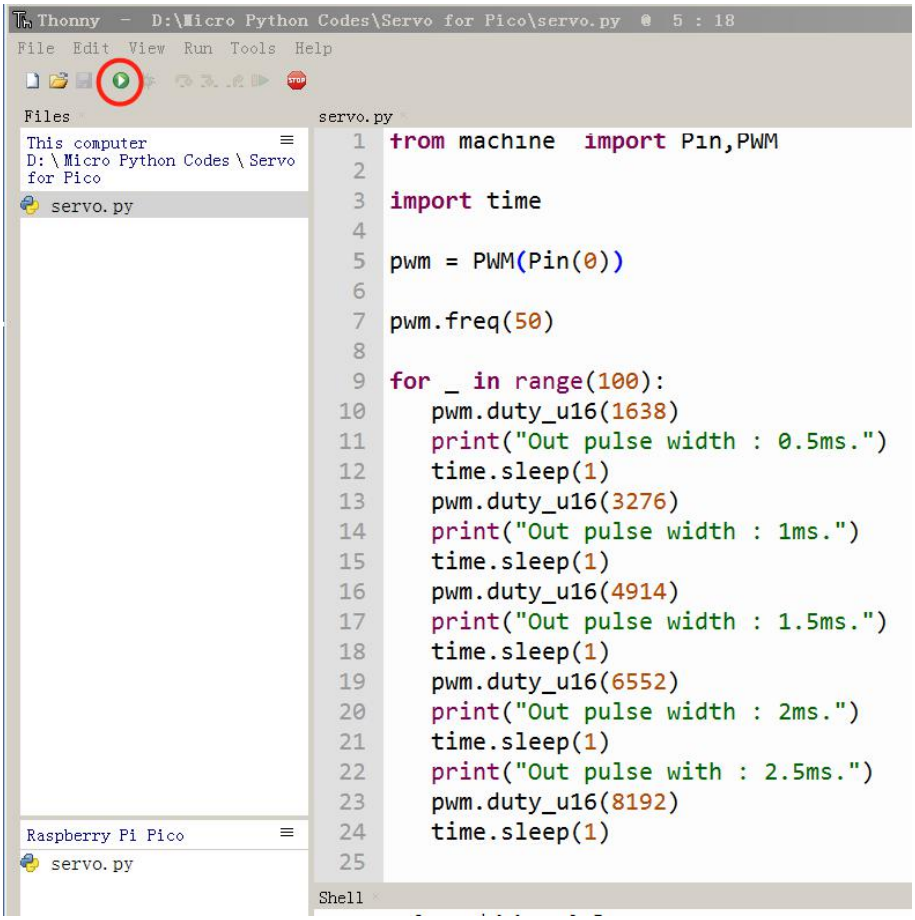
The corresponding connection relationship is shown in the table below:

Raspberry Pi Pico Pin	MG996R Servo Pin
GP0	Signal

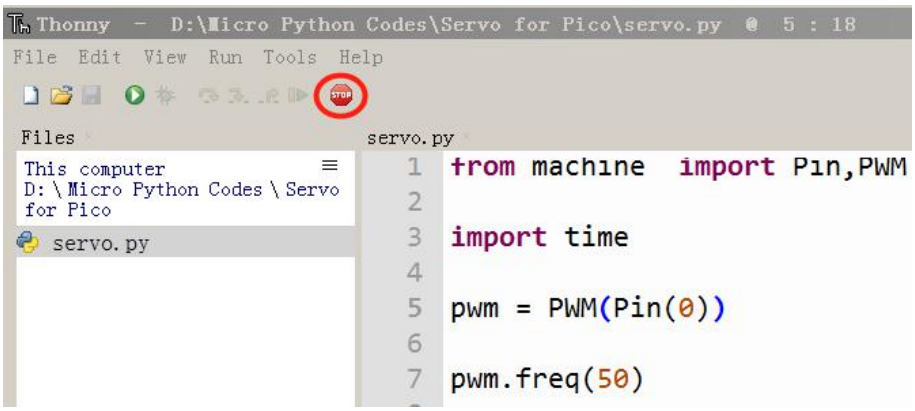
VBUS	VCC(+)
GND	GND(-)

2.4 MG996R Servo testing by Pico

2.4.1 As shown in the following picture, click the “Run” to start the program and the MG996R Servo will rotate 0~180 degrees periodically.



2.4.2 As shown in the following picture, click the "Stop" to stop the program and the MG996R Servo stops spinning.



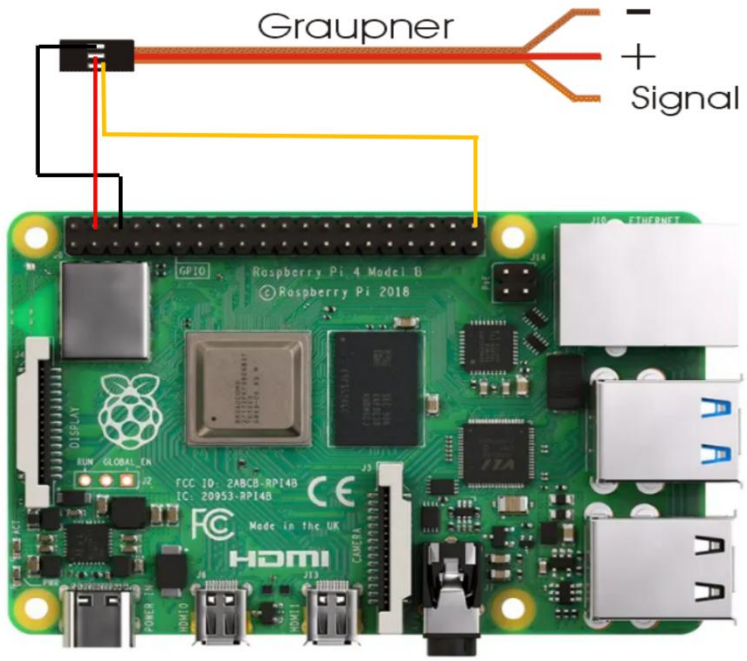
3. Testing MG996R Servo by Raspberry Pi

3.1 Set up an environment for Raspberry Pi programs to run

Please refer to the documentation "CKK0016-main\Tutorial\ Installing and Configuring Raspberry Pi System"

3.2 Connect the MG996R Servo to the Raspberry Pi 4B board

3.2.1 As shown in the following picture, the signal pin of the MG996R Servo is connected to the GPI0.29 pin of the Raspberry Pi 4B board， the VCC(+) pin of the MG996R Servo is connected the +5V pin of the Raspberry Pi 4B board, the GND(-) pin of the MG996R Servo is connected the GND pin of the Raspberry Pi 4B board.



The corresponding connection relationship is shown in the table below:

Raspberry Pi 4B Board Pin	MG996R Servo Pin
21 (BCM)	signal
5V	VCC
GND	GND

3.3 Upload the code to Raspberry Pi 4B and testing

3.3.1 The Servo for Raspberry Pi code is placed in the

"CKK0016-main\Tutorial\Python\Python_Codes" folder.

3.3.2 Please refer to the document "CKK0016-main\Tutorial\ Installing and Configuring Raspberry Pi System" to learn how to upload code to Raspberry Pi

3.3.3 When the code is upload to the Raspberry Pi 4B and run,you can see that the MG996R Servo rotates 180 degrees in a loop.

Note: The pin number used in the code is the "pin number corresponding to BCM"

3.3.4 Python code

Servo for Raspberry Pi.py

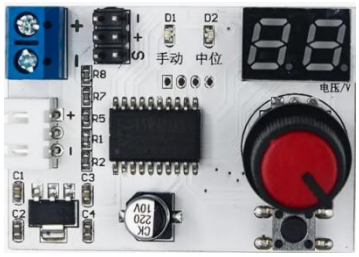
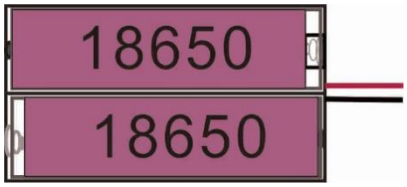
code as below:

```
1.  # -*- coding: utf-8 -*-
2.  #!/usr/bin/env python
3.
4.  import RPi.GPIO as GPIO
5.  import time
6.  import signal
7.  import atexit
8.
9.  atexit.register(GPIO.cleanup)
10.
11. servopin = 21
12. GPIO.setmode(GPIO.BCM)
13. GPIO.setup(servopin, GPIO.OUT, initial=False)
14. p = GPIO.PWM(servopin,50) #50HZ
15. p.start(0)
16. time.sleep(2)
17.
18. while(True):
19.   for i in range(0,181,10):
20.     p.ChangeDutyCycle(2.5 + 10 * i / 180) #set rotation angle
21.     time.sleep(0.02)                #wait 20ms for the cycle time
22.     p.ChangeDutyCycle(0)            #Initialize
23.     time.sleep(0.2)
24.
25.   for i in range(181,0,-10):
26.     p.ChangeDutyCycle(2.5 + 10 * i / 180)
27.     time.sleep(0.02)
28.     p.ChangeDutyCycle(0)
29.     time.sleep(0.2)
```

4. Testing MG996R Servo by RC Digital Servo Tester

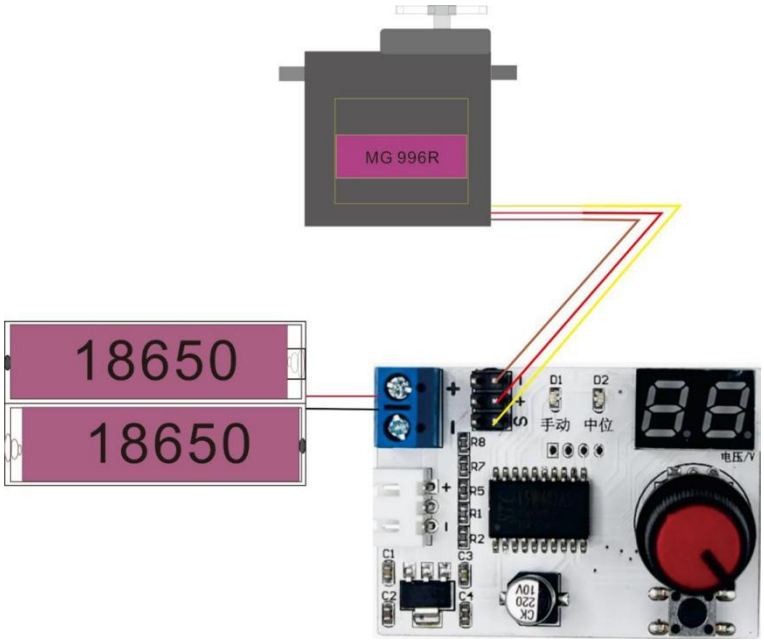
It is very convenient to use the RC digital servo tester to test the servo because it has a built-in control program. Simply rotate the potentiometer knob on top to change the pulse width of the pulse signal. Different pulse width signals make the servo rotate to different angles.
When testing the MG996R Servo function and for simple use, it is recommended to use this solution without the need for additional programming control, which is more convenient.

4.1 What do you need to prepare

Components	Quantity	Picture	Remark
MG996R Servo	1		
RC digital servo tester	1		Not included in the Kit,you should prepared by yourself.This is a sample for your reference.And there are many kinds of RC digital servo tester on Amazon,you can choose what you want.
Battery box with 2pcs 18650 batteries	1		Not included in the Kit,you should prepared by yourself

4.2 Connection Circuit

Connect the battery box containing two 18650 batteries to the RC digital servo tester, ensuring that the positive and negative poles are connected to the corresponding positions and not reversed. The RC digital servo tester in the example has two servo interfaces. Insert the MG996 servo into either interface, and note that the "signal vcc gnd" of the MG996 servo pin corresponds to the "**s + -**" on the RC digital servo tester.



4.3 Test

After the circuit is connected and checked for no issues, manually rotate the potentiometer knob on the RC digital servo tester. When rotating clockwise, the corresponding rotation angle of the MG996R Servo gradually increases. When rotating clockwise to the limit position, the MG996R Servo rotates to 180 degrees. When rotating counterclockwise, the corresponding rotation angle of MG996R Servo gradually decreases. When rotating counterclockwise to the limit position, MG996R Servo rotates to 0 degrees.

5. Make your suggestion and get support

THANK YOU for participating in this learning experience!

If you find any errors, omissions or you have suggestions and/or questions about this lesson, please feel free to contact us: **cokoino@outlook.com**

We will make every effort to make changes and correct errors as soon as feasibly possible and publish a revised version.

If you want to learn more about Arduino, Raspberry Pi, Smart Cars, Robotics and other interesting products in science and technology, please continue to visit our Amazon Store by search for "**LK COKOINO**" on Amazon. We will continue to launch fun, cost-effective, innovative and exciting products.

Thank you again for choosing Cokoino products.

LK COKOINO