Integration Project Week 2

2020/11/12

instructor:Mr. Terry Sturtevant

Dekai Meng 186800570

Hongsha Shang 163042150

We do this project together, because she does not have kits yet and I do not have a windows device to support putty. We had told our professor and he allowed us to do that.

1, Introduction:

The report will cover integration projects week1 and week2, we had used four devices: Servo motor ultrasonic sensor, tube and potentiometer. We let them hooked up and work in the pi, connect both of them and run the code for four devices working together and working well with intersection. In the below report, there are following parts:

connect the ultrasonic sensor.

connect the servo motor.

connect the tube.

connect the potentiometer.

We had connected all of them at the same time.

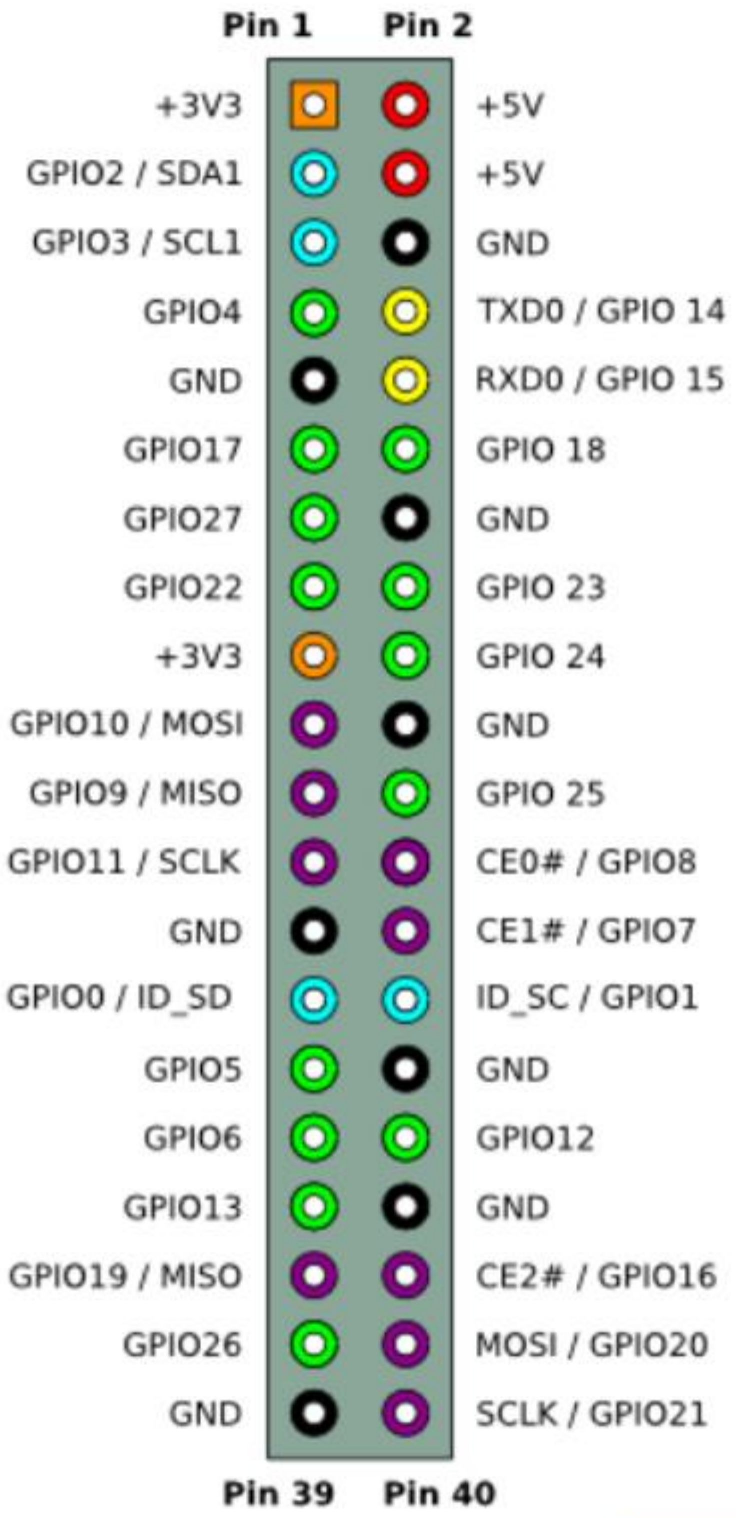
For each of the parts, we had the connected picture, testing the code result, our observation, proper shutdown, and our conclusion and reflection for this part of the integration project. The main things for week 2 are in part 5, which is shown below. We had done a large part of our project and all the devices working well!

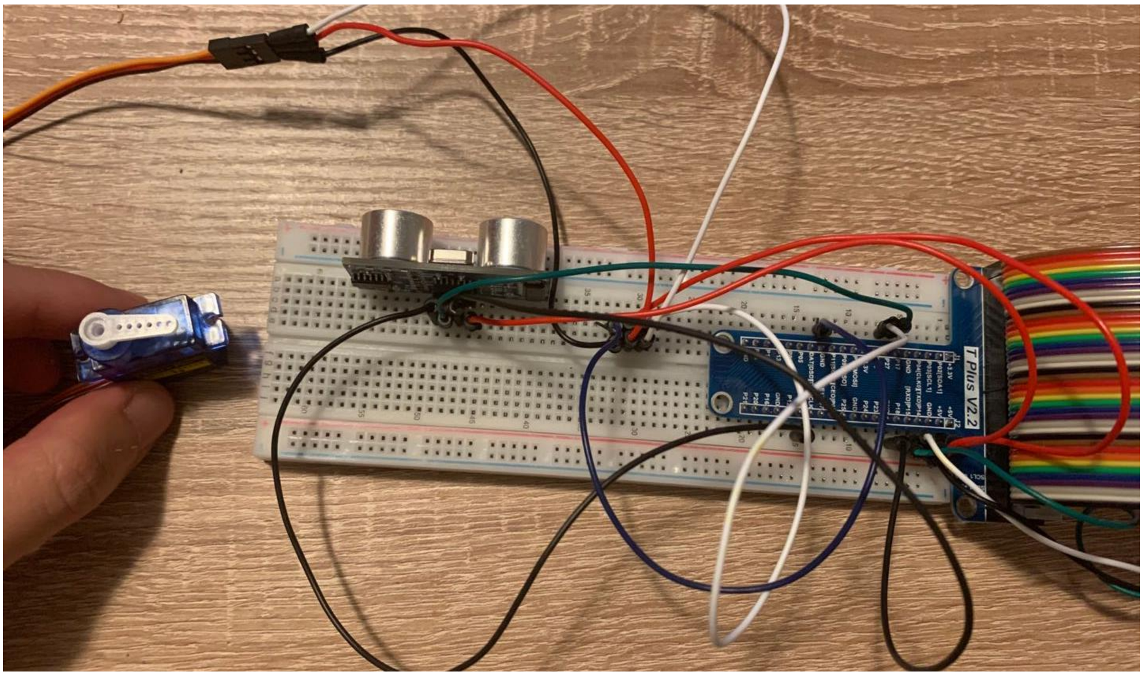
2, The connection for ultrasonic sensor individually:

2.1: connected picture:

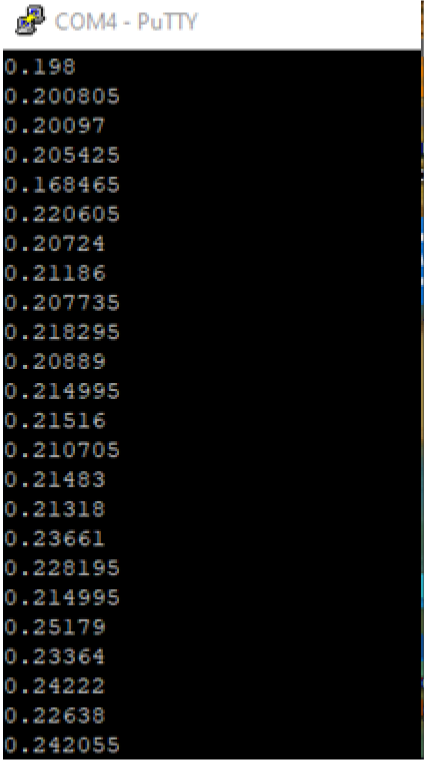
We place the sensor on the board as shown. The sensor parallel to the breadboard, r ather than perpendicular

Since the ultrasonic sensor requires 5V for Vcc, connect a red wire from the 5V GPIO pin to the Vcc pin on the sensor.


Pin 1
Pin 2
+3V3
+5V
GPIO2 / SDA1
+5V
GPIO3 / SCL1
GND
GPIO4
TXD0 / GPIO 14
GND
RXD0 / GPIO 15
GPIO17
GPIO 18
GPIO27
GND
GPIO22
GPIO 23
+3V3
GPIO 24
GPIO10 / MOSI
GND
GPIO9 / MISO
GPIO 25
GPIO11 / SCLK
CE0# / GPIO8
GND
CE1# / GPIO7
GPIO0 / ID_SD
ID_SC / GPIO1
GPIO5
GND
GPIO6
GPIO12
GPIO13
GND
GPIO19 / MISO
CE2# / GPIO16
GPIO26
MOSI / GPIO20
GND
SCLK / GPIO21
Pin 39
Pin 40

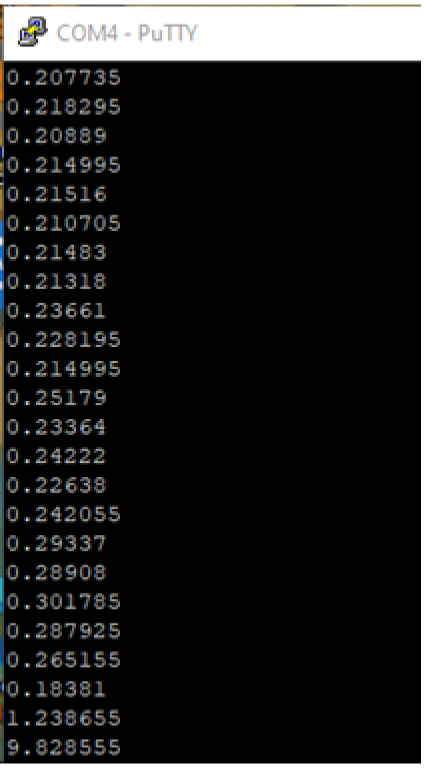


2.2: testing the code result:



COM4 - PuTTY

```
0.198
0.200805
0.20097
0.205425
0.168465
0.220605
0.20724
0.21186
0.207735
0.218295
0.20889
0.214995
0.21516
0.210705
0.21483
0.21318
0.23661
0.228195
0.214995
0.25179
0.23364
0.24222
0.22638
0.242055
```



COM4 - PuTTY

```
0.207735
0.218295
0.20889
0.214995
0.21516
0.210705
0.21483
0.21318
0.23661
0.228195
0.214995
0.25179
0.23364
0.24222
0.22638
0.242055
0.29337
0.28908
0.301785
0.287925
0.265155
0.18381
1.238655
9.828555
```

2.3: our observation:

First I put a book parallel to the sensor , and next we change the direction of the boo k to not parallel, the result increase to 9.8. At first, book is parallel to the sensor, the o utput value is small. But as we change the direction of book, the output value increase largely.

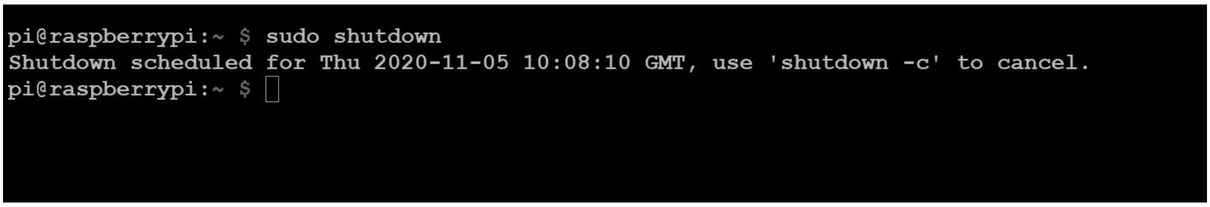
According to the datasheet ,the range is 0.02-4m.

The object is parallel or not. If the object is not parallel, then the sound will touch the wire or other things.

Because the max distance is in the range from 0.2m to 4m, then if the distance is not in the valid range, the value will not be returned correctly.

2. 4： proper shutdown：

We type in sudo shutdown and shutdown successfully.

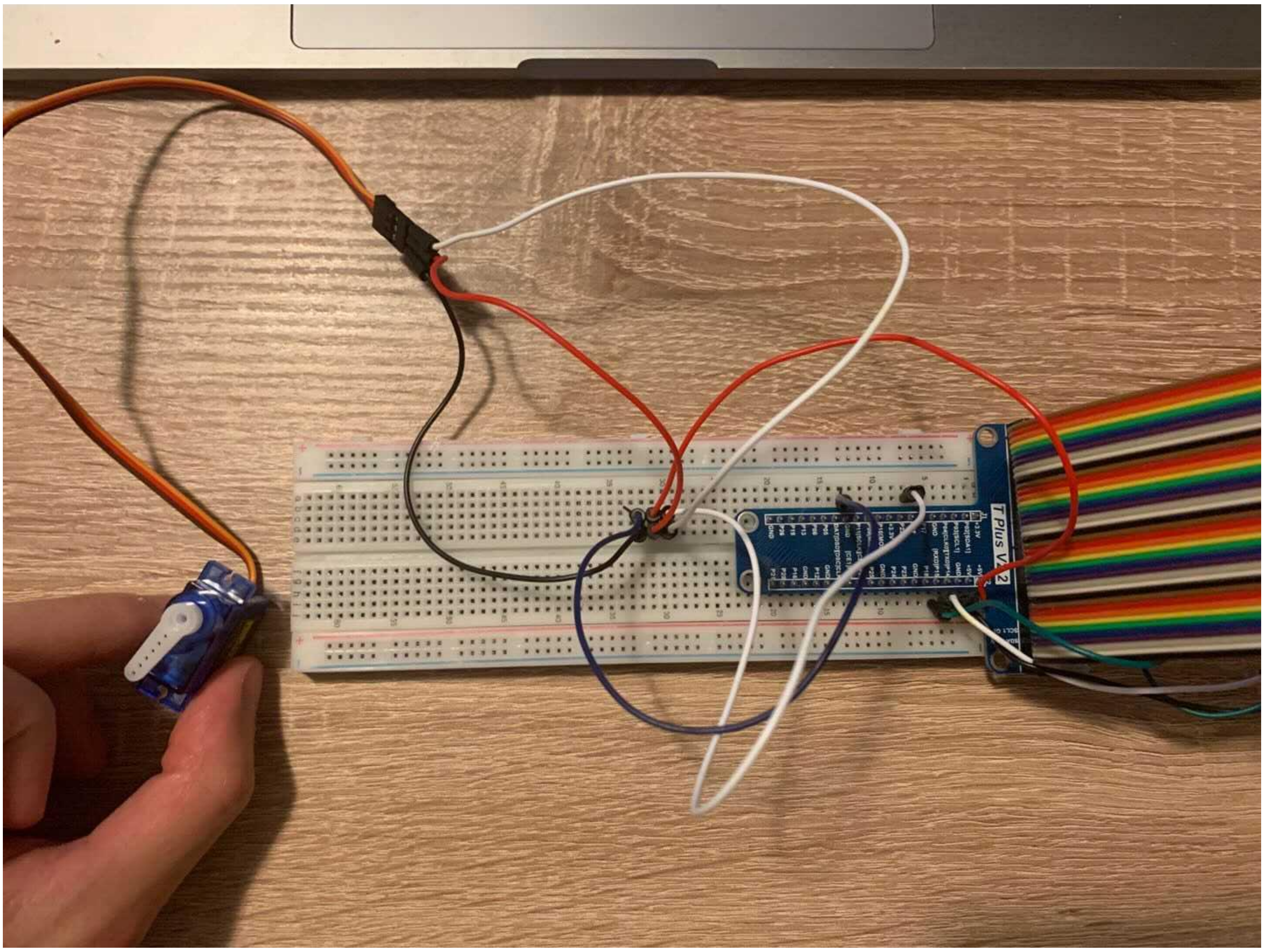


```
pi@raspberrypi:~ $ sudo shutdown
Shutdown scheduled for Thu 2020-11-05 10:08:10 GMT, use 'shutdown -c' to cancel.
pi@raspberrypi:~ $ 
```

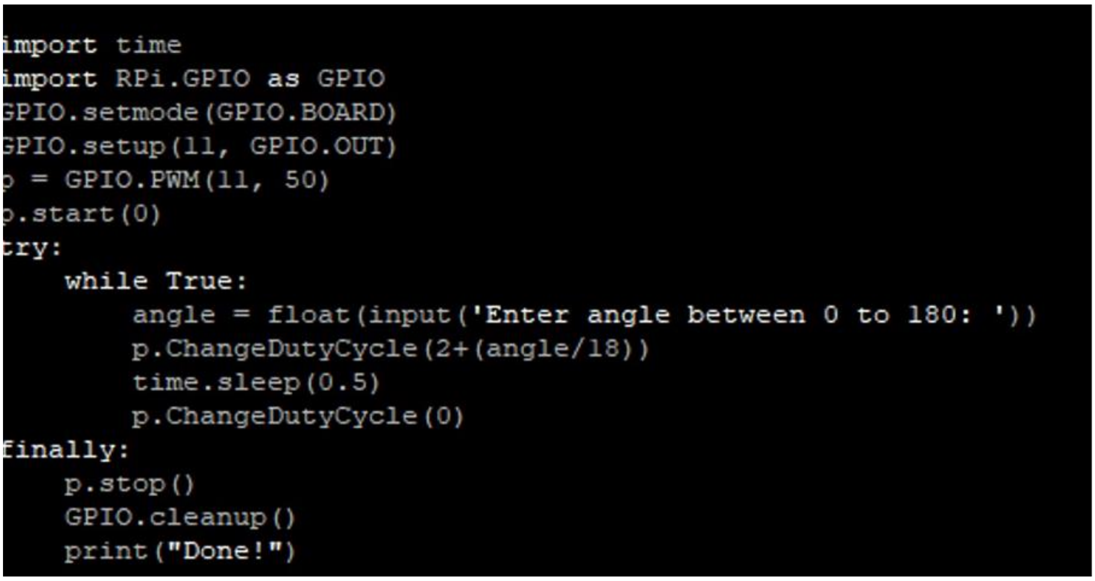
Then, I started to type the third connection.

3, The connection for servo motor individually:

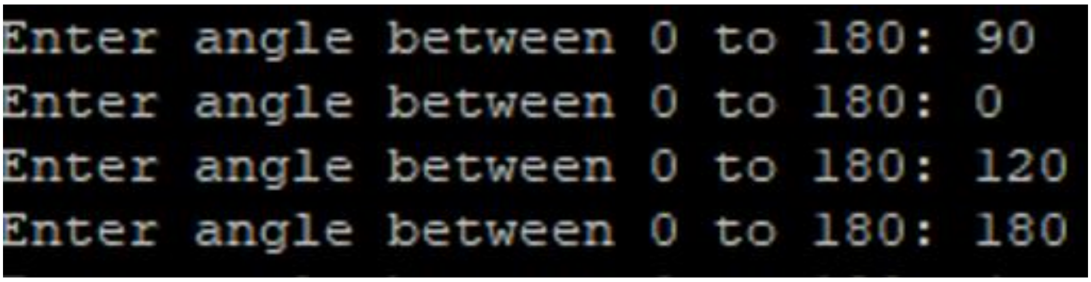
3.1: connected picture:



3.2: testing the code result:



```
import time
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BOARD)
GPIO.setup(11, GPIO.OUT)
p = GPIO.PWM(11, 50)
p.start(0)
try:
    while True:
        angle = float(input('Enter angle between 0 to 180: '))
        p.ChangeDutyCycle(2+(angle/18))
        time.sleep(0.5)
        p.ChangeDutyCycle(0)
finally:
    p.stop()
    GPIO.cleanup()
    print("Done!")
```



```
Enter angle between 0 to 180: 90
Enter angle between 0 to 180: 0
Enter angle between 0 to 180: 120
Enter angle between 0 to 180: 180
```

The original:

The 30 degree:

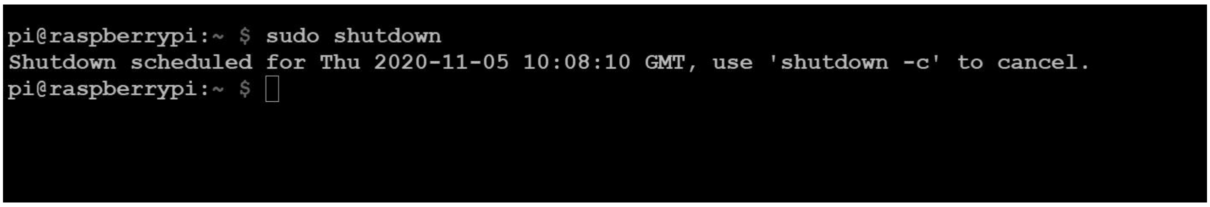
The 120 degree:

3.3: our observation:

We observed that the motor’s angle will change when we change the dutycycle in our code, l ike the previous lab. And we calculated the cycle as 20ms, when the duty cycle is highest, whi ch is about 12.5ms, that is about 180 degrees. And the 2.5ms is 0 degrees. My testing result i s similarly to our calculated result.

3.4: proper shutdown:

We type in sudo shutdown and shutdown successfully.

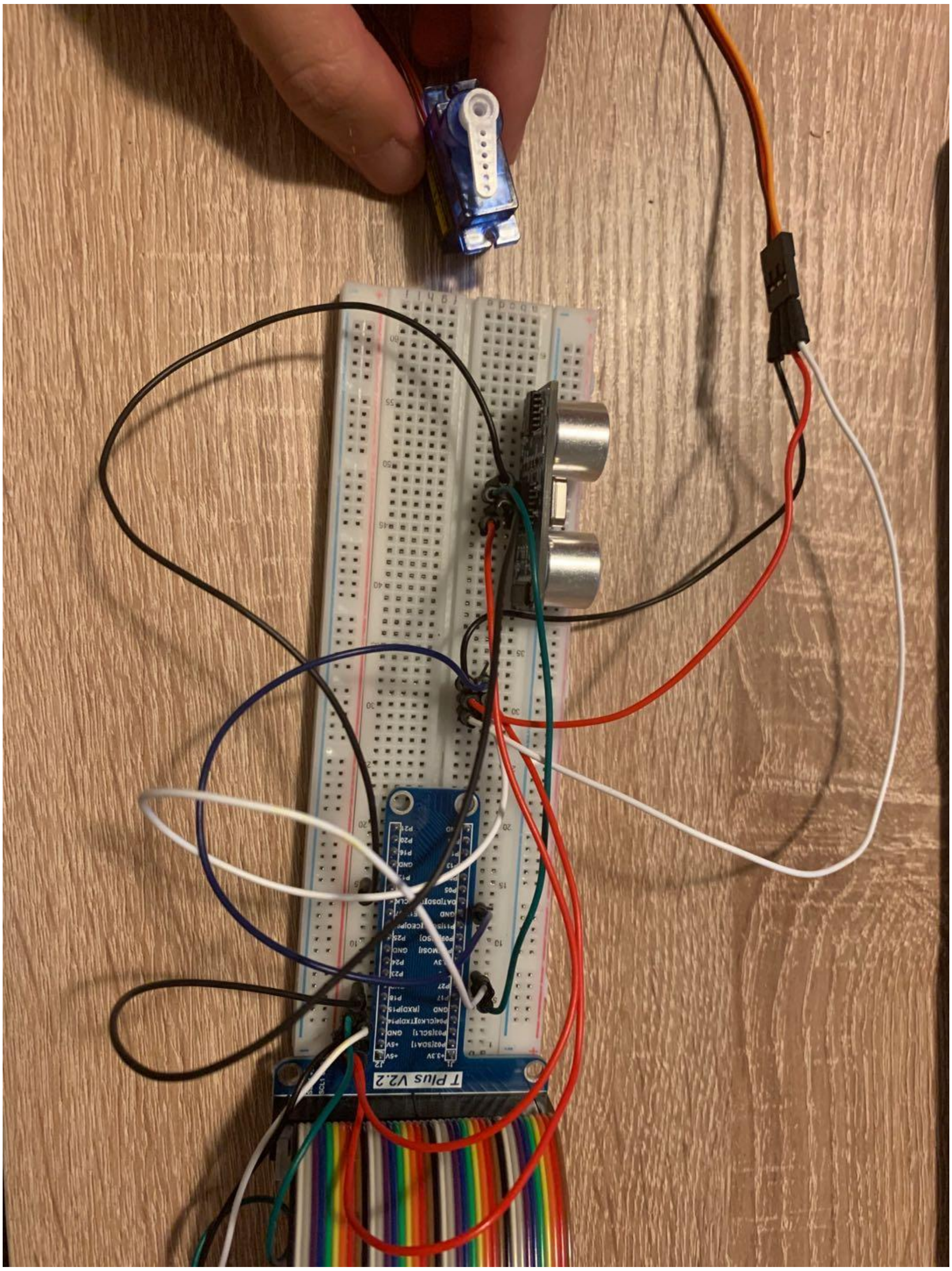


```
pi@raspberrypi:~ $ sudo shutdown
Shutdown scheduled for Thu 2020-11-05 10:08:10 GMT, use 'shutdown -c' to cancel.
pi@raspberrypi:~ $ 
```

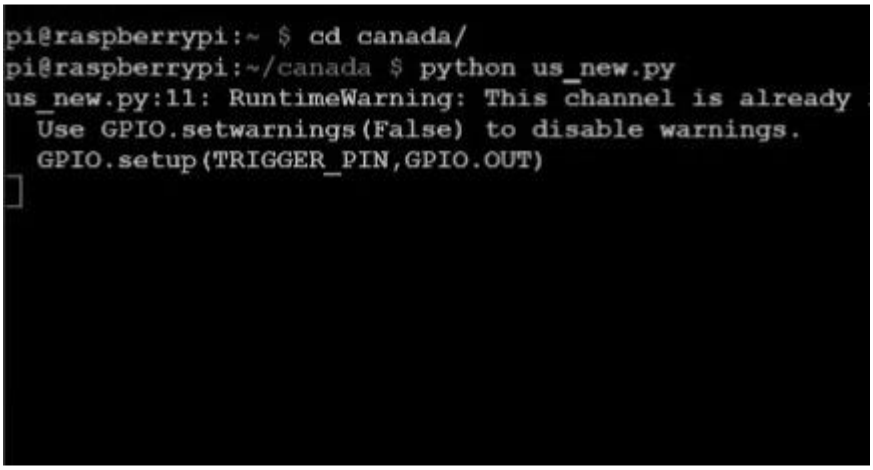
Then, I started to connect the forth connection.

4, The connection for both of them at the same time:

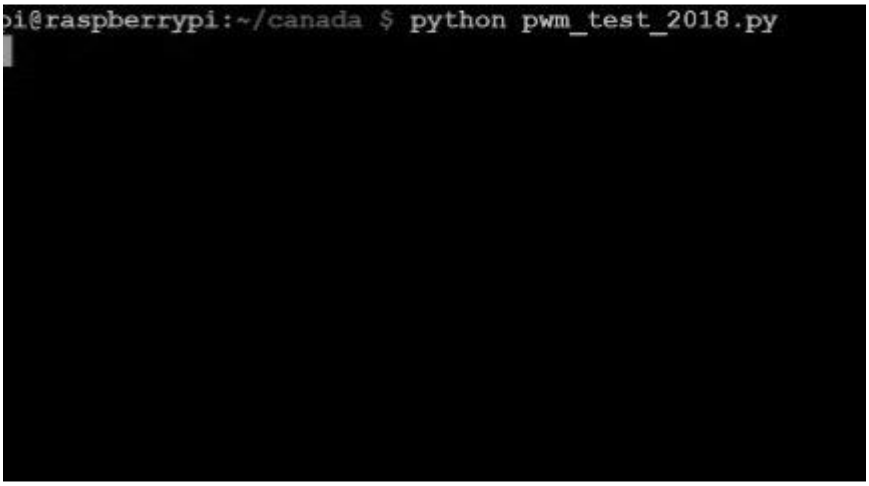
4.1: connected picture:



4.2: testing the code result:



```
pi@raspberrypi:~ $ cd canada/
pi@raspberrypi:~/canada $ python us_new.py
us_new.py:11: RuntimeWarning: This channel is already
  Use GPIO.setwarnings(False) to disable warnings.
  GPIO.setup(TRIGGER_PIN,GPIO.OUT)
```



```
pi@raspberrypi:~/canada $ python pwm_test_2018.py
```

4.3: our observation:

When we test both the servo motor and sensor at the same time, that should not work. Because the pins in the current code are duplicated. We need to change the program for servo Motor, which is changing 12 to 13, then we get the same testing result.

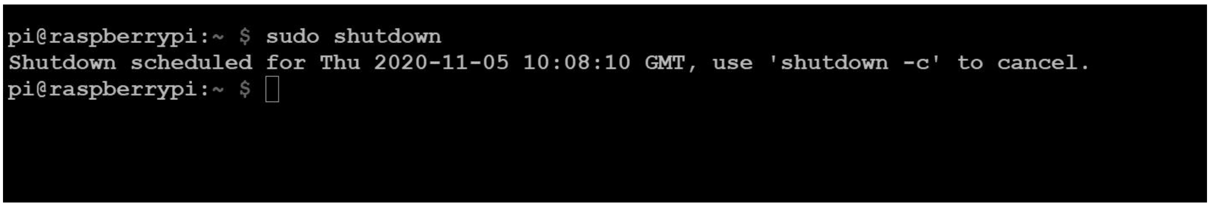
pwm_test_2018.py change to:

```
import time
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BOARD)
GPIO.setup(13, GPIO.OUT)

p = GPIO.PWM(13, 60)  # channel=12 frequency=60Hz
p.start(50)
```

```
time.sleep(5)
try:
  while True:
        p.ChangeDutyCycle(20)
        time.sleep(5)
        p.ChangeDutyCycle(100)
        time.sleep(5)
except KeyboardInterrupt:
  pass
p.stop()
GPIO.cleanup()
```

4. 4： proper shutdown：

We type in sudo shutdown and shutdown successfully.

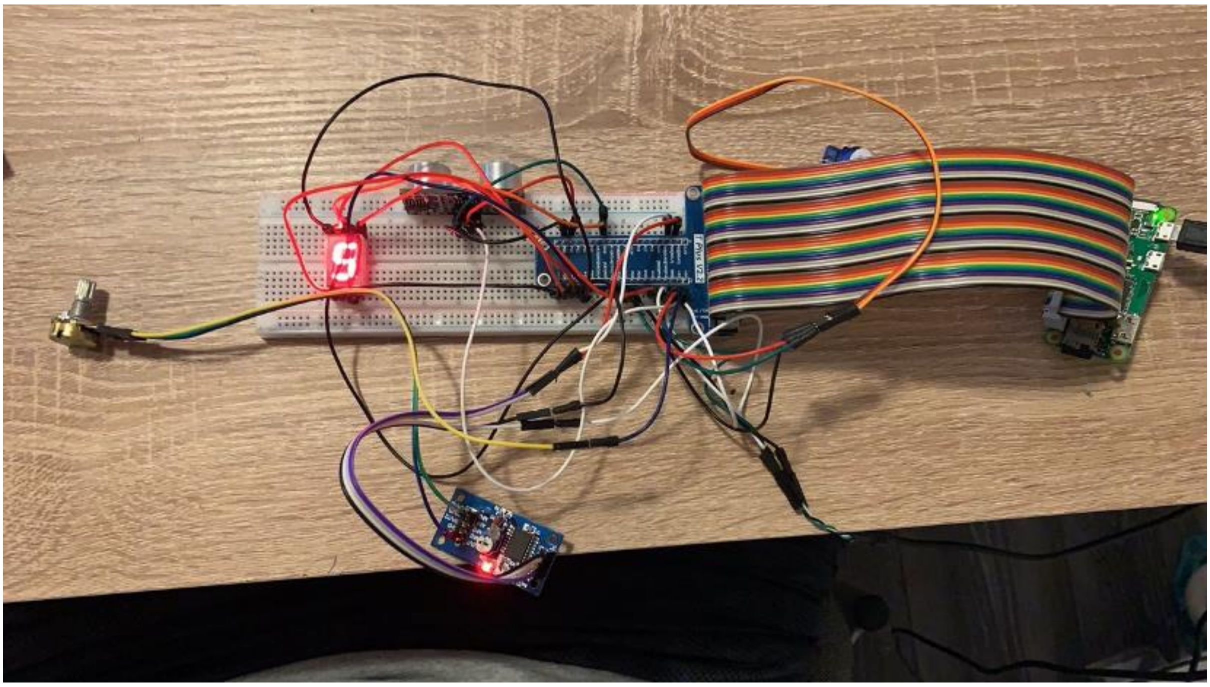


```
pi@raspberrypi:~ $ sudo shutdown
Shutdown scheduled for Thu 2020-11-05 10:08:10 GMT, use 'shutdown -c' to cancel.
pi@raspberrypi:~ $ 
```

Then, I disconnect the connection.

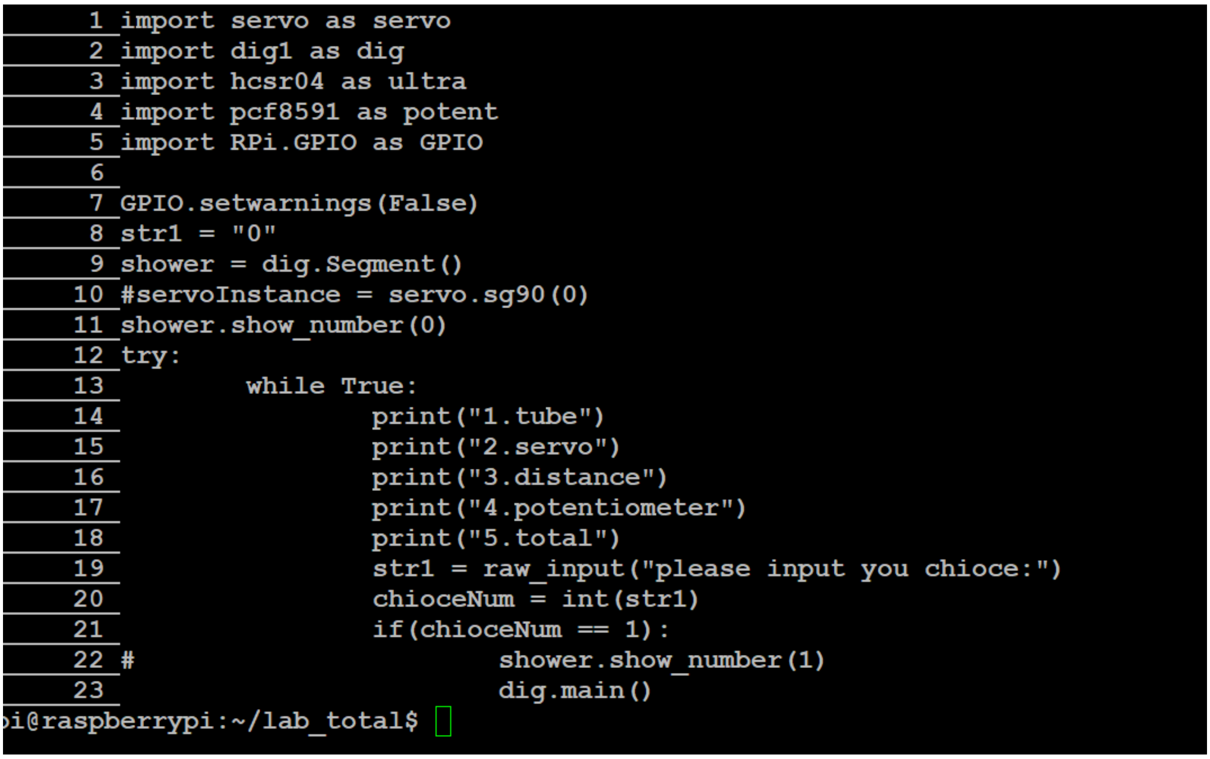
5: connect four device together and testing our code:

5.1: connected pricture :



5.2: code and testing result:

We had create a python program for 4 devices, which allowed users to enter the number to select the following devics:

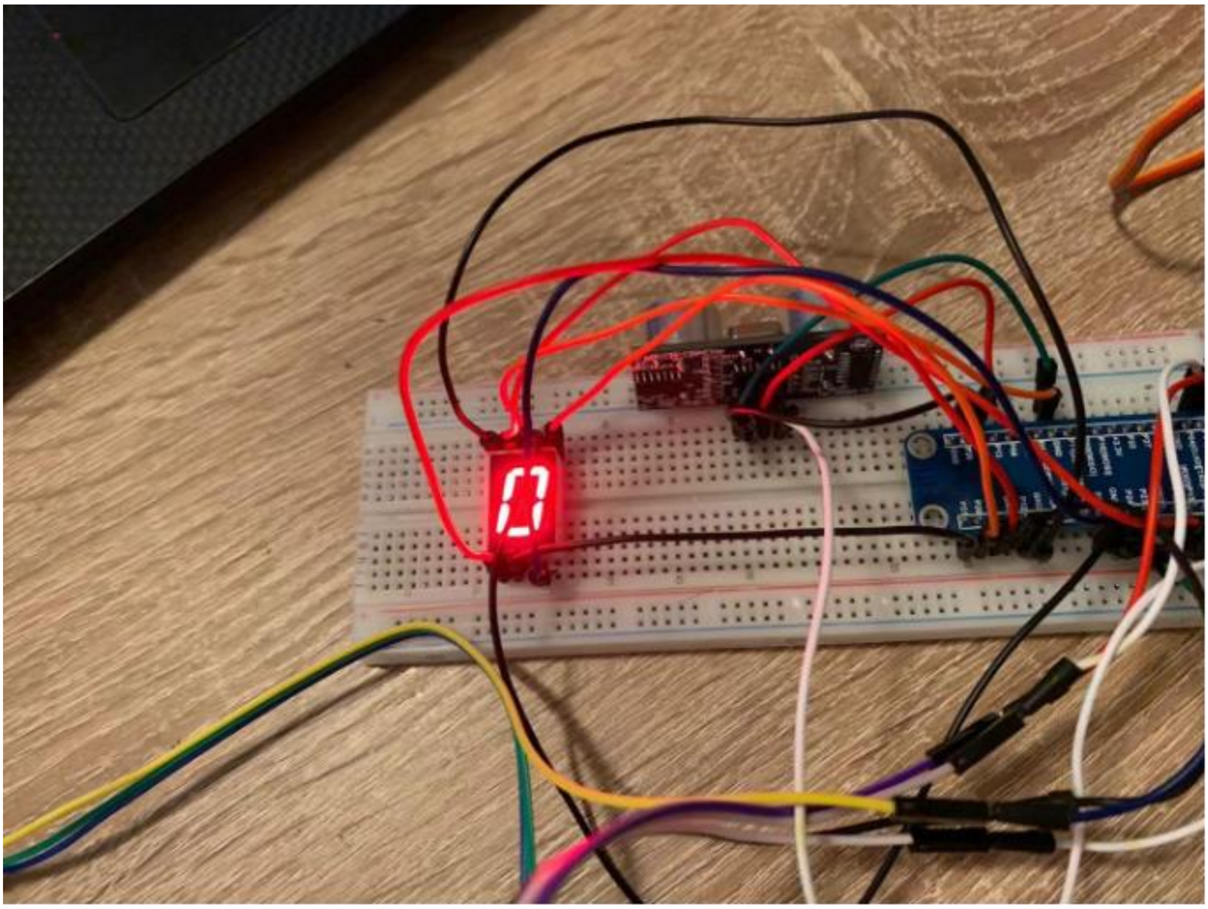


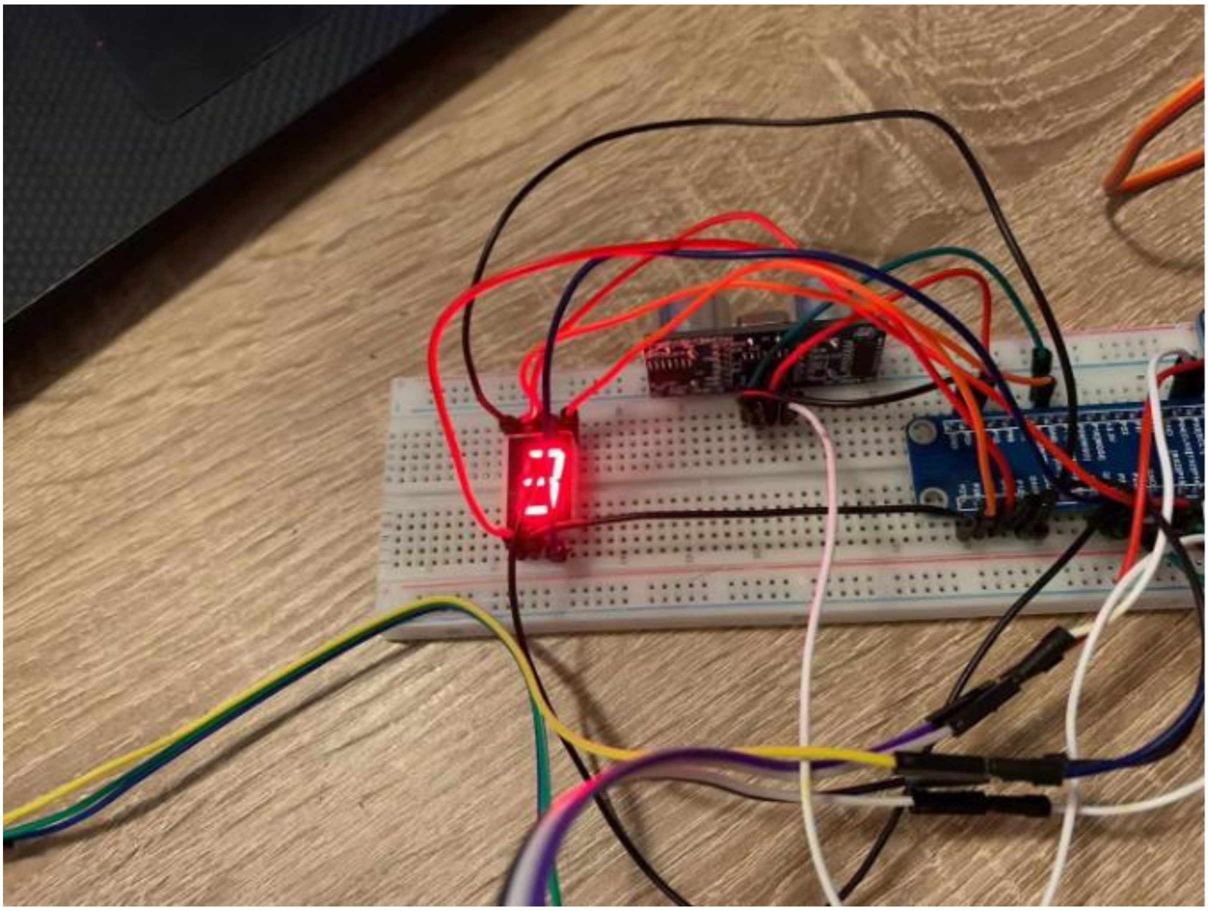
```
import servo as servo
import dig1 as dig
import hcsr04 as ultra
import pcf8591 as potent
import RPi.GPIO as GPIO

GPIO.setwarnings(False)
str1 = "0"
shower = dig.Segment()
#servoInstance = servo.sg90(0)
shower.show_number(0)
try:
        while True:
                print("1.tube")
                print("2.servo")
                print("3.distance")
                print("4.potentiometer")
                print("5.total")
                str1 = raw_input("please input you chioce:")
                chioceNum = int(str1)
                if(chioceNum == 1):
#                       shower.show_number(1)
                        dig.main()
pi@raspberrypi:~/lab_total$
```

5. 3： our observation

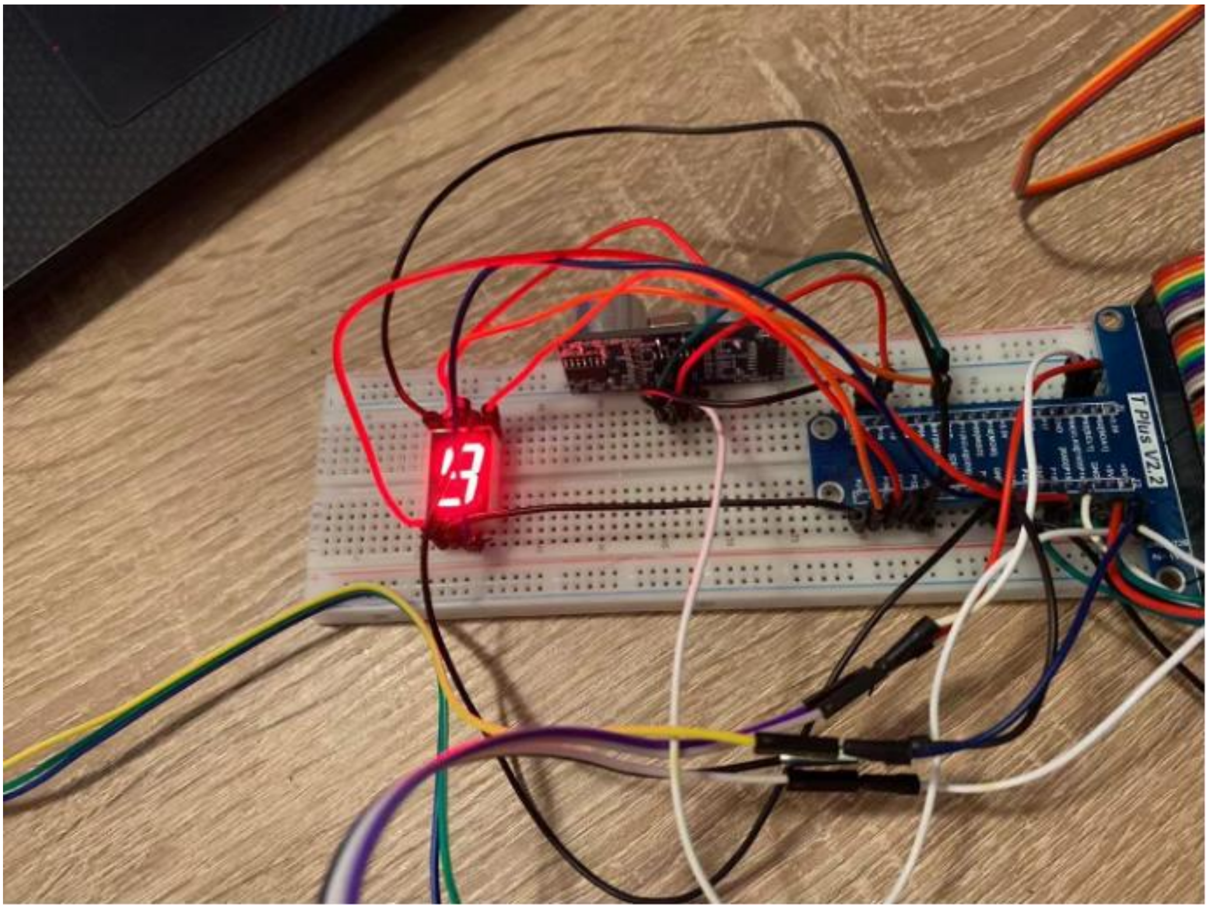
5.3.1:

If I enter 1, the tube will show from 0 to 9:





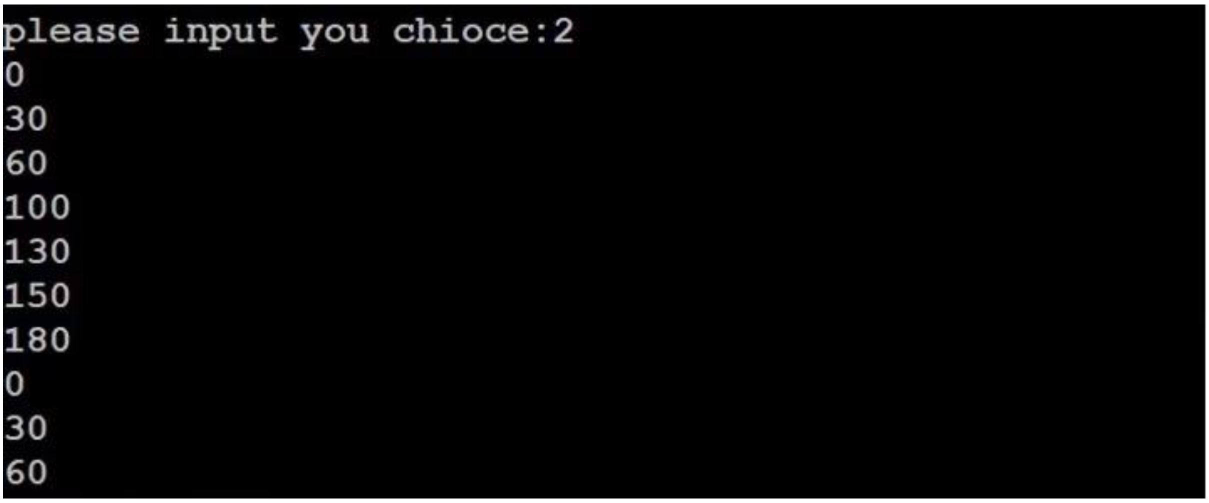
....



It is obvious that that is so quick to change the number, so we did not photo all of the digits from 0 to 9.

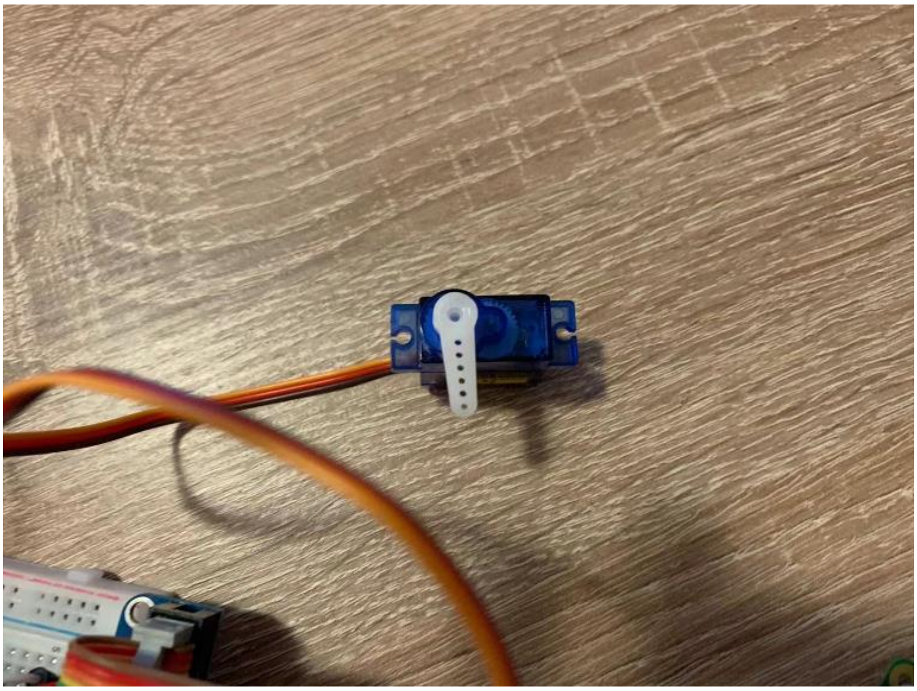
5.3.2:

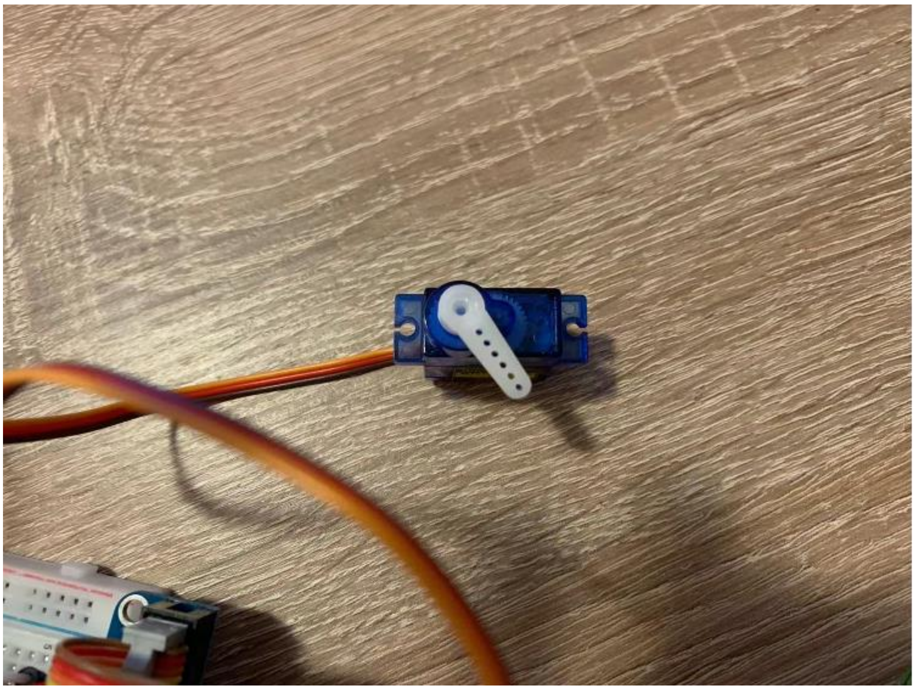
If I enter 2, the servo will work like this:

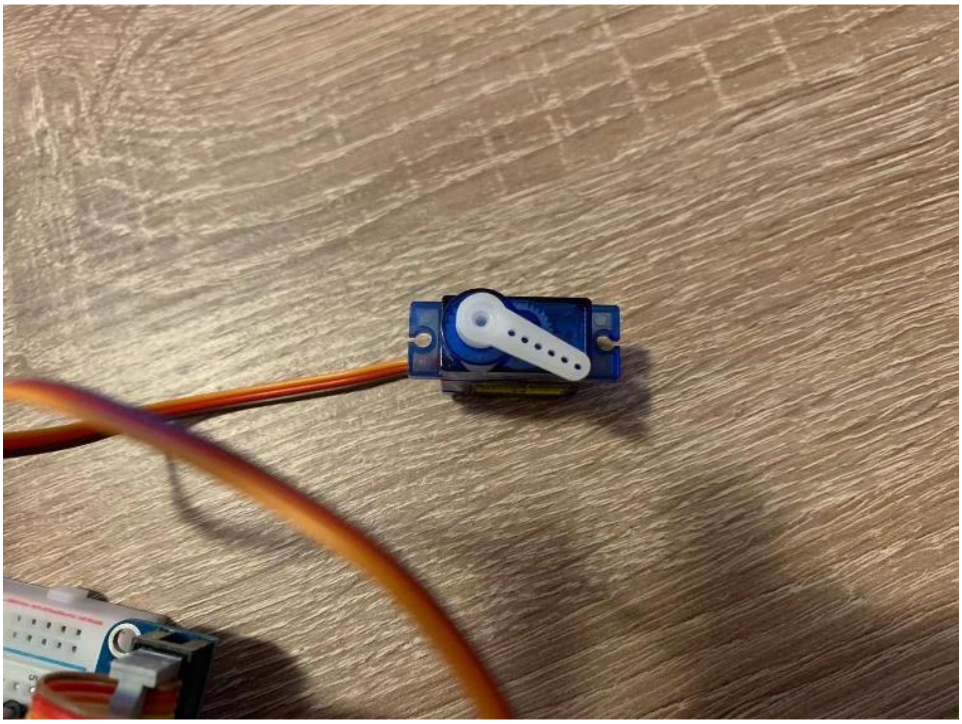


```
please input you chioce:2
0
30
60
100
130
150
180
0
30
60
```

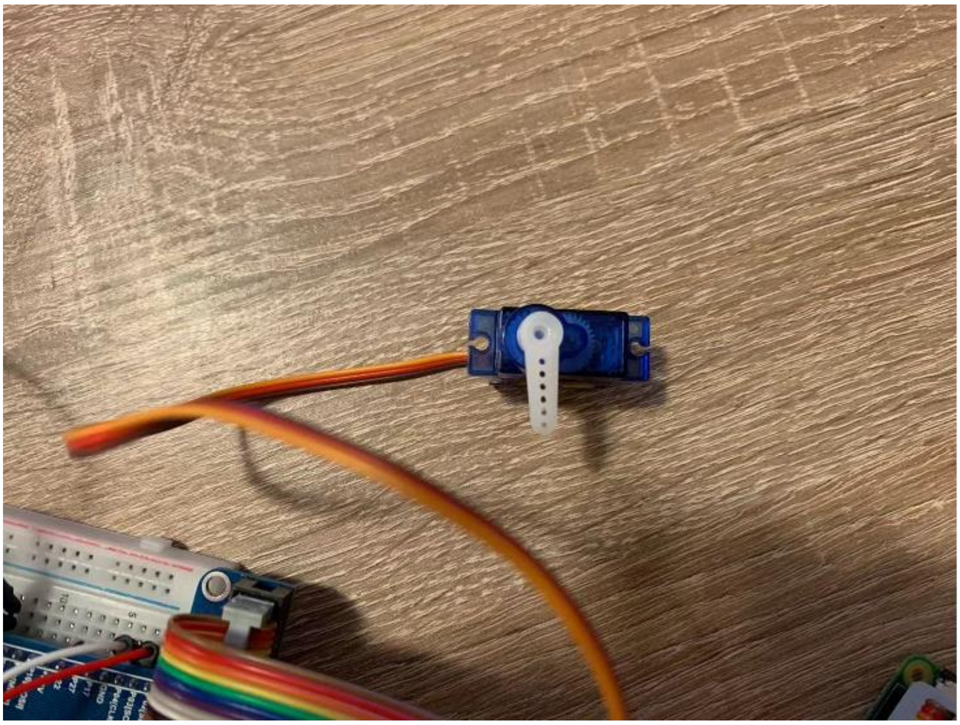
And the degree of the server will be changed ass the code running result.







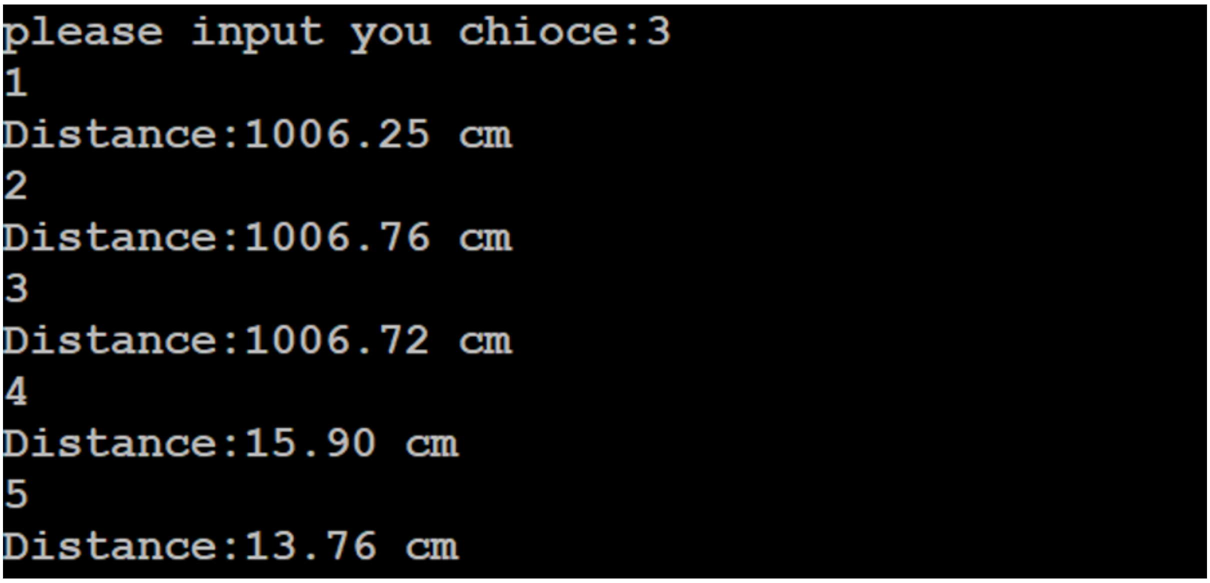
…



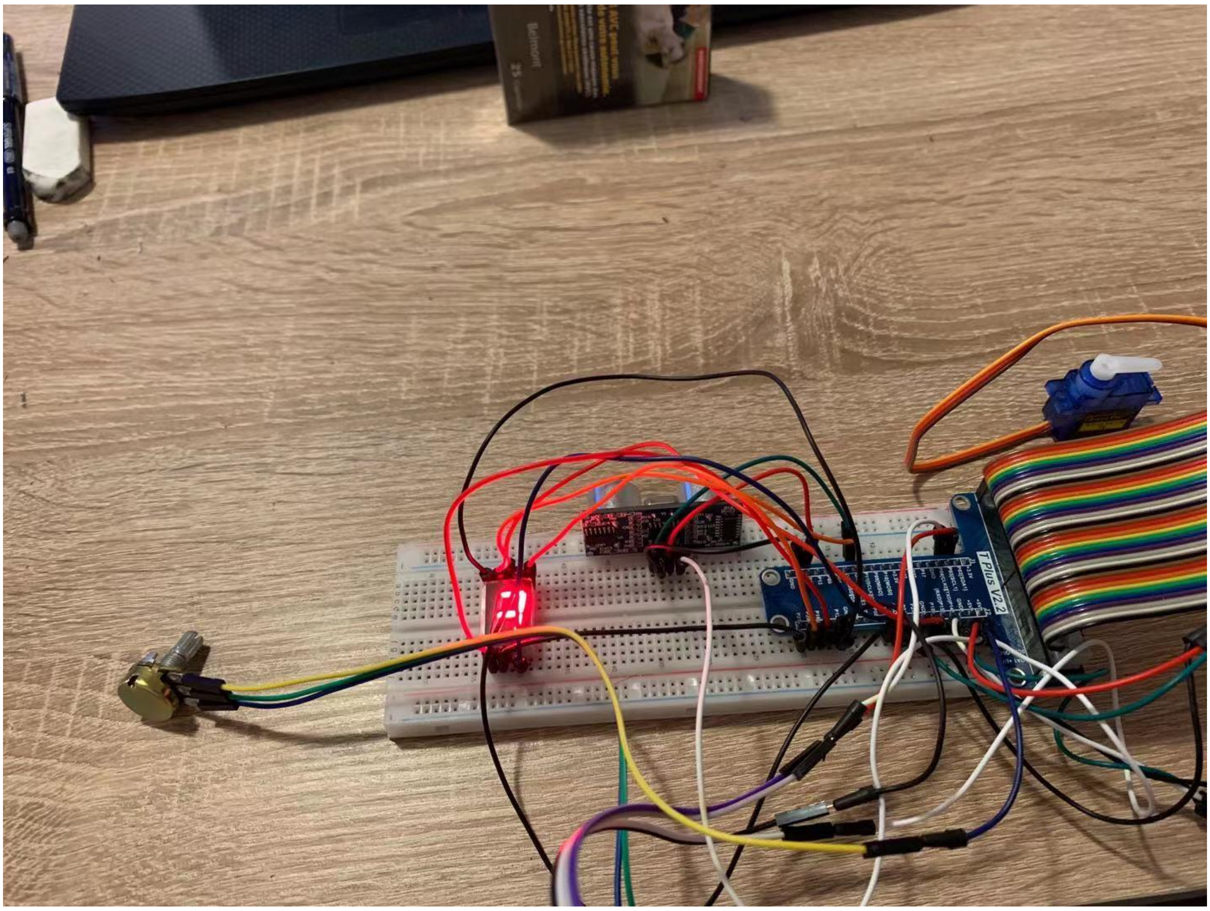
The degree will be changed and it will return to 0 sometimes. That is working well as our exception.

5.3.3:

If I chose 3, the motor will perform like this(because the information about distance is so huge, so I only select five of them):

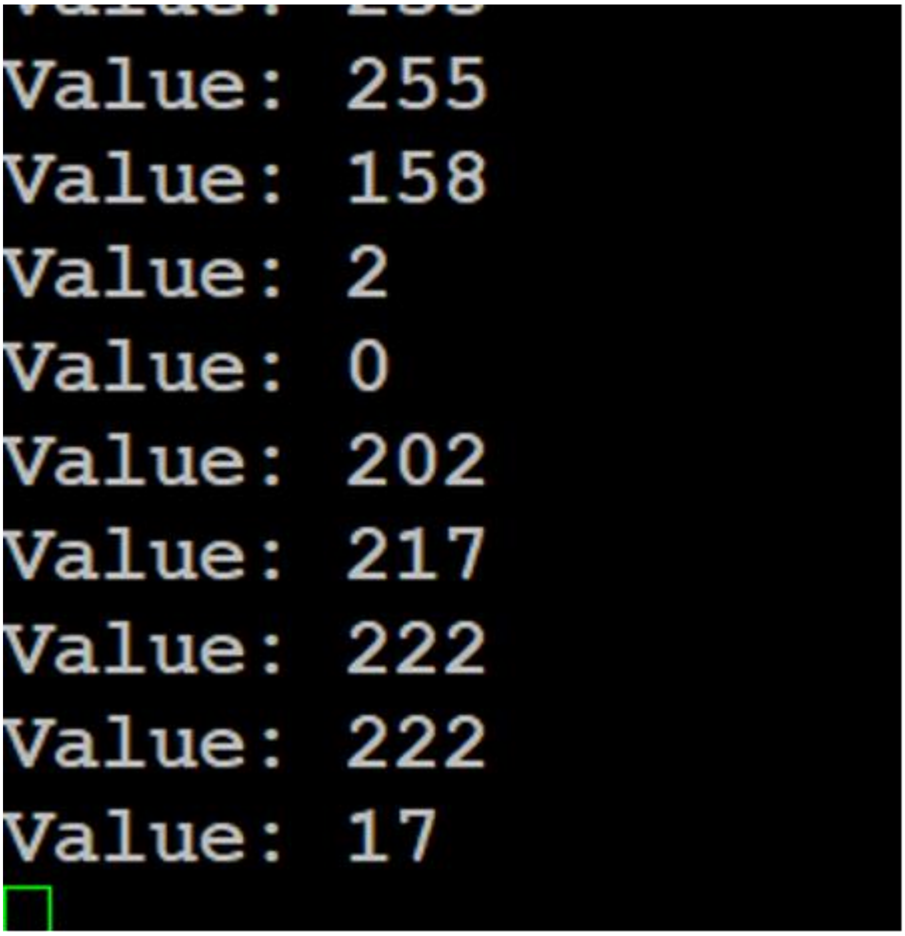


```
please input you chioce:3
1
Distance:1006.25 cm
2
Distance:1006.76 cm
3
Distance:1006.72 cm
4
Distance:15.90 cm
5
Distance:13.76 cm
```



The distance will clearly change as we put something near the device, and that works well !

5.3.4: potentiometer:

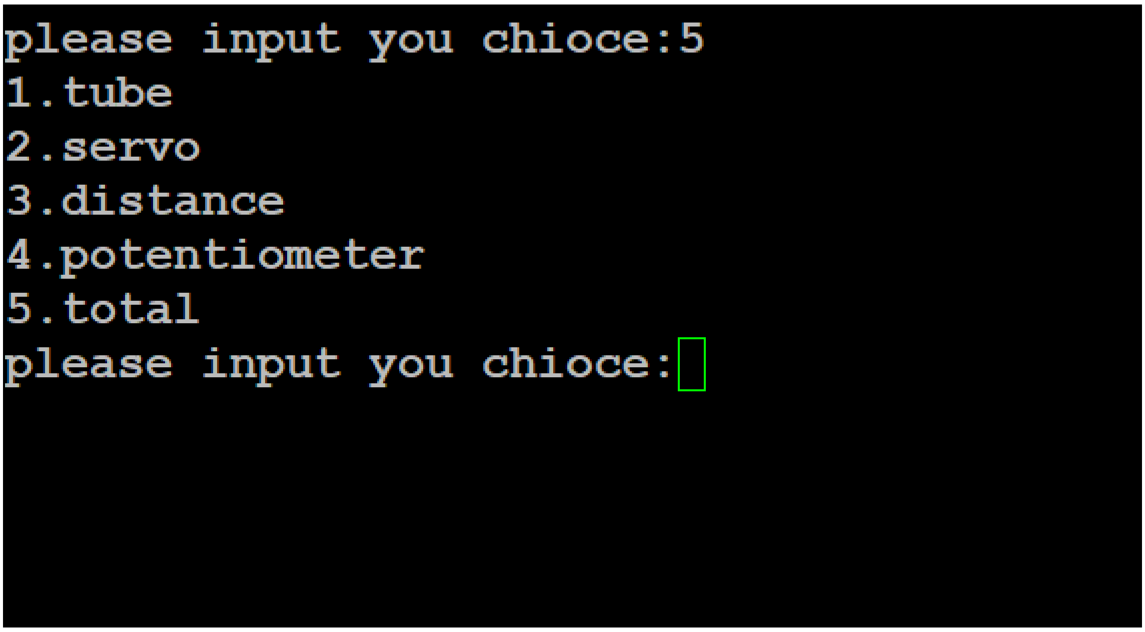


```
Value: 255
Value: 158
Value: 2
Value: 0
Value: 202
Value: 217
Value: 222
Value: 222
Value: 17
```

When I rotate the device, the number in the putty will be changed between 0 and 255 . Like the lab what we had done, the device is working well.

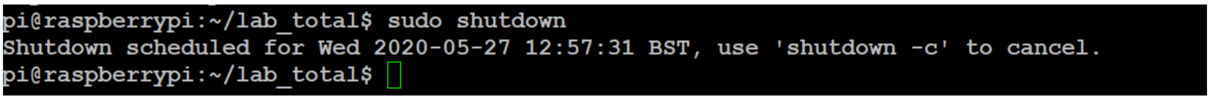
5.3.5:

If we choose number 5, the device layout will be shown in the putty, and the tube will show number 5 as well!



```
please input you chioce:5
1.tube
2.servo
3.distance
4.potentiometer
5.total
please input you chioce:
```

5.4 : sudo shutdown :



```
pi@raspberrypi:~/lab_total$ sudo shutdown
Shutdown scheduled for Wed 2020-05-27 12:57:31 BST, use 'shutdown -c' to cancel.
pi@raspberrypi:~/lab_total$
```

shutdown successfully.

6, The conclusion and reflection:

In the integration project of week2, we had already done nearly everything, except the fix of code final code. We had created a code in order to let all of the devices which in the integration project work well. We had Try to let them work at the situation of intersection.

To be specific, we try to use the tube to show the user’s choice and let the other device which the tube number corresponds to the specific device and let them work well.

After week 2 part, we had found that every device can intersect together and work well.

That is not like the lab which we studied, which put some of the devices on the pi and worked individually.

All in all, we had paid more time to connect the device and write the code, following the professor’s order to let them work together. And we will fix the final code for the integration project as soon as possible.

Thank you for your reading! Look forward to getting your feedback and suggestions.