

2021年第10届广东省创意机器人大赛培训

编程型机器人 传感器模块编程

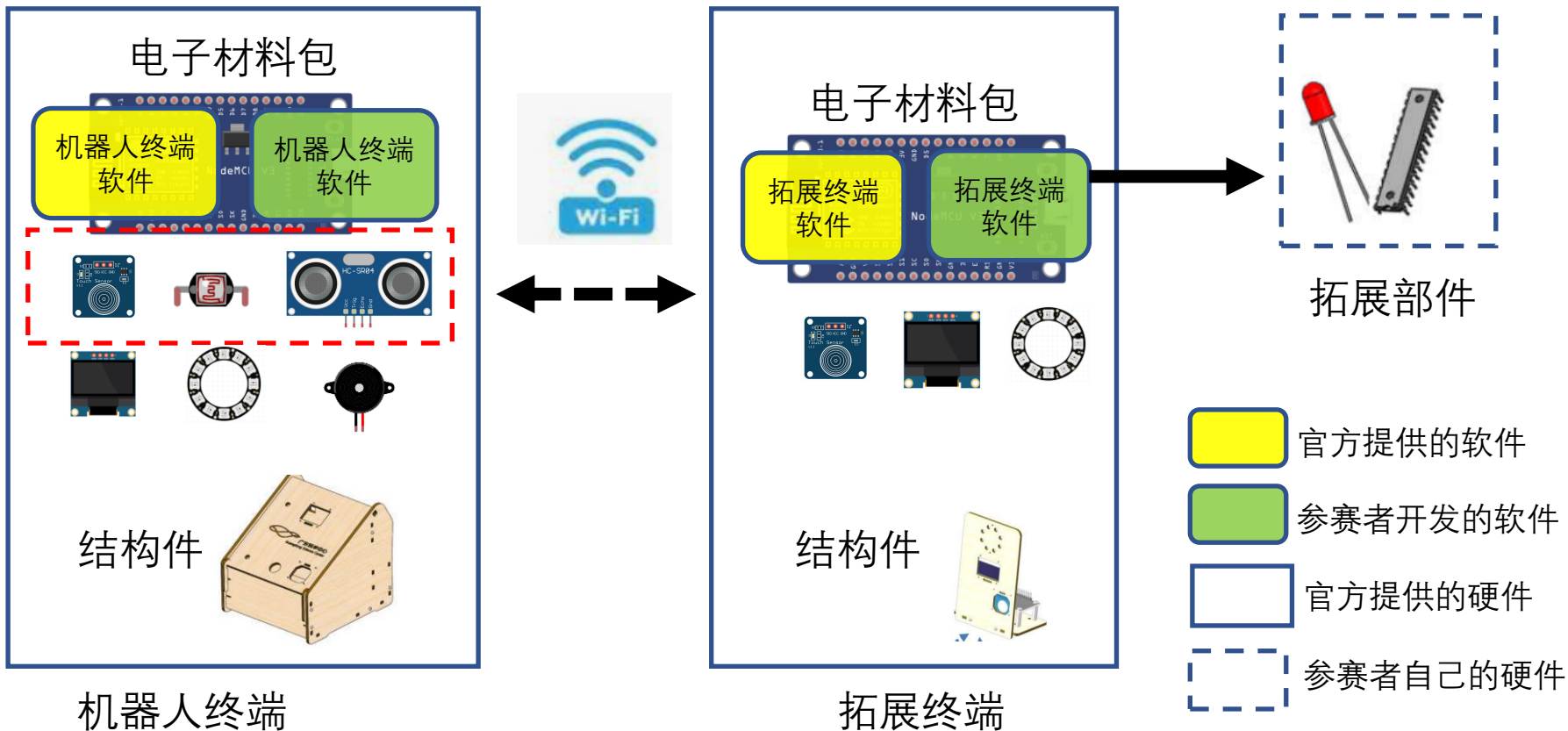
朱金辉

华南理工大学 软件学院
智能软件与机器人研究室

2021年7月



机器人的系统组成





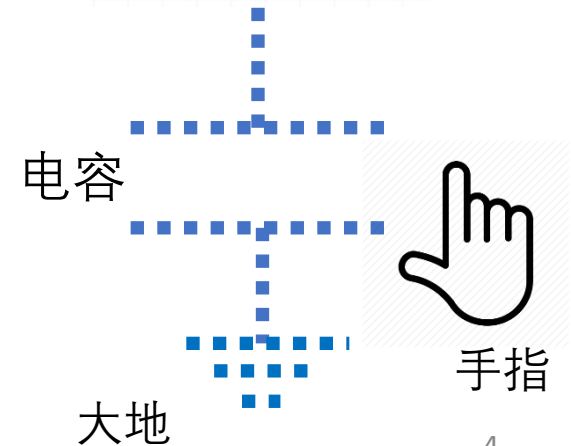
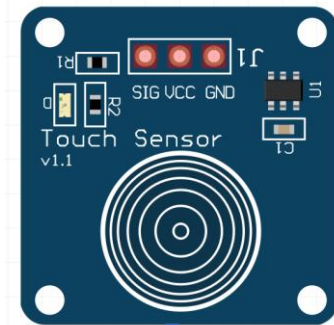
提纲

1. 触摸开关
2. 光敏电阻
3. 超声波传感器



1. 触摸开关

- 电容式触摸开关 (capacitive touch sensor)
- 任何两个导电的物体之间都存在着感应电容。一个按键即一个焊盘与大地也可构成一个感应电容，在周围环境不变的情况下，该感应电容值是固定不变的微小值。
- 当有人体手指靠近触摸按键时，人体手指与大地构成的感应电容并联焊盘与大地构成的感应电容，会使总感应电容值增加。
- 功能：机器人触觉



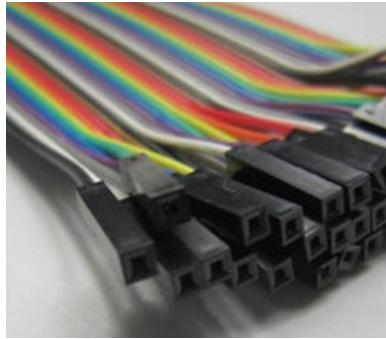


杜邦线

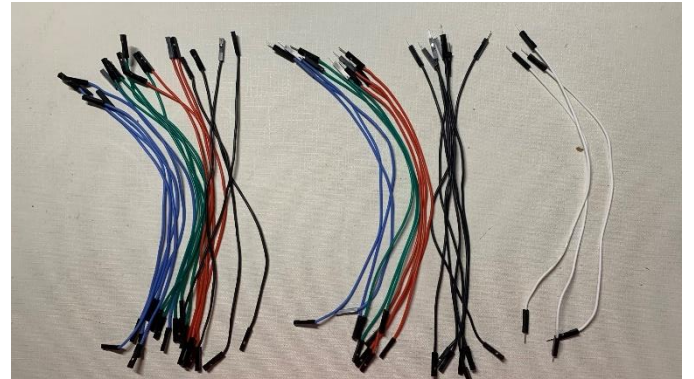
- 杜邦线 (Dupont Line) 指的是电路上常用的连接导线 (Jump wire)。
- 可用于实验板的引脚扩展, 无需焊接, 可以快速进行电路试验。



公端



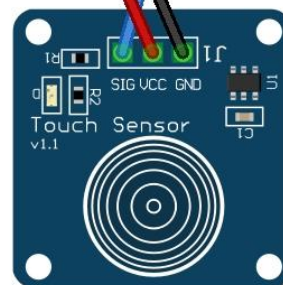
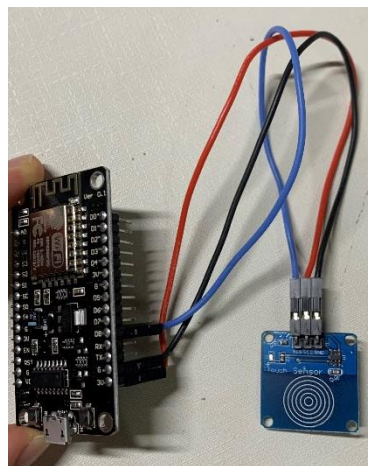
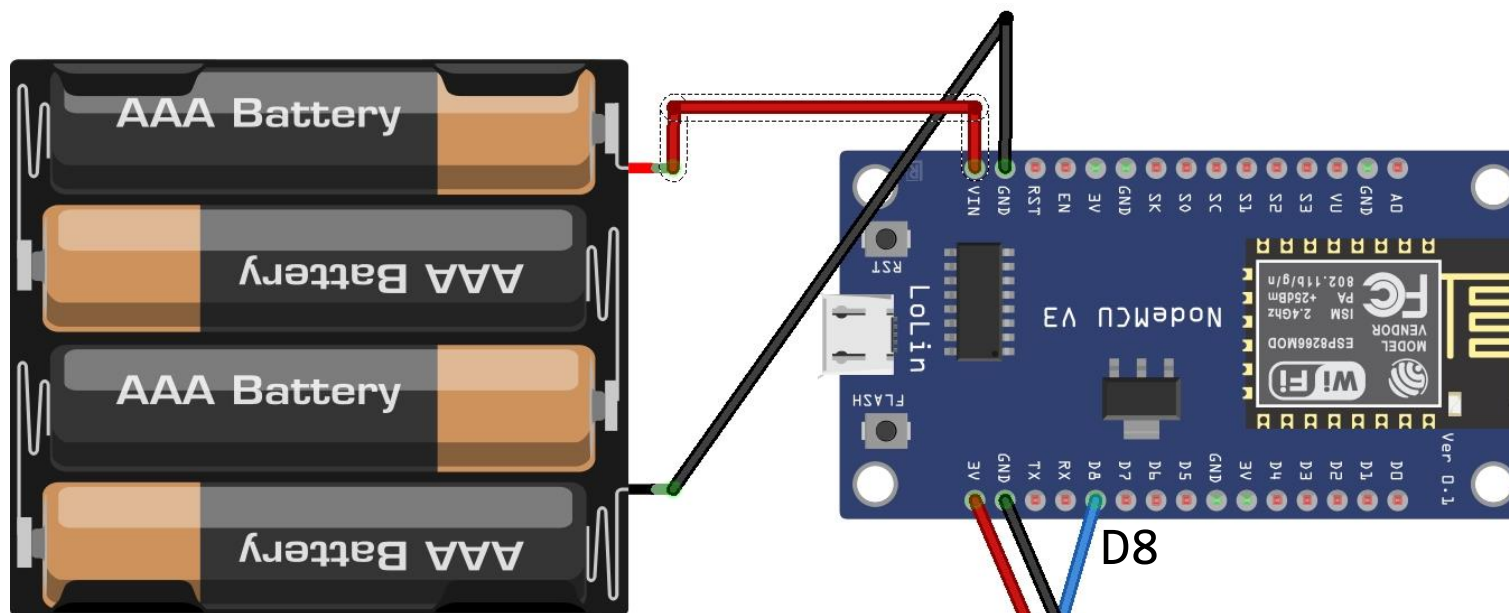
母端



充分利用杜邦线的
颜色是一个好习惯!



电路连接图





实验1. 触摸



```
1 from machine import Pin
2 import time
3
4 # touch sensor D8 GPIO15
5 button = Pin(15, Pin.IN)
6
7 print("Test button")
8 print("no press:1  press:0")
9 while True:
10     print(button.value())
11     time.sleep_ms(100)
```



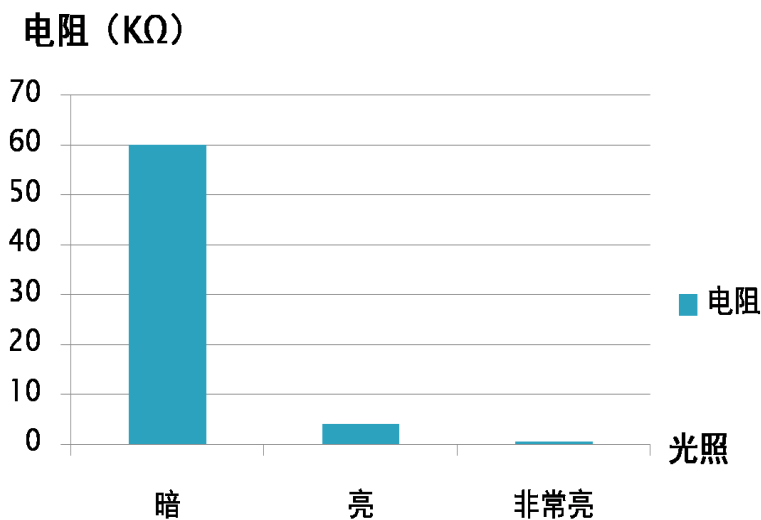
>>>

```
MPY: soft reboot
Test button
no press:1  press:0
0
0
0
0
0
0
0
0
1
1
0
0
```



2. 光敏电阻

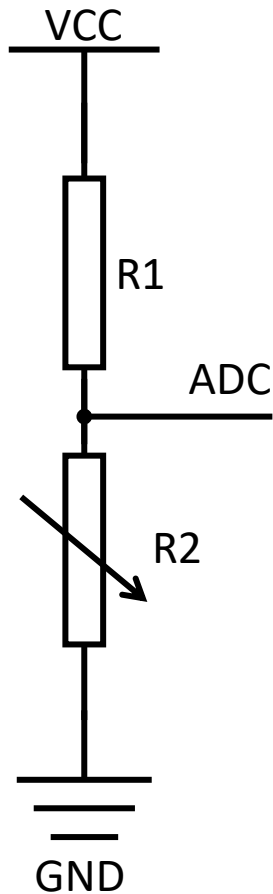
- 光敏电阻 (Photoresistor)，又称光电阻、光导体、光导管，是利用光电导效应的一种特殊的电阻。
- 它的电阻和入射光的强弱有直接关系。光强度增加，则电阻减小；光强度减小，则电阻增大。
- **功能**：检测光线强度





测光电路及分析

问题：光照变化后，AD采集的数据如何变化？



$$\begin{aligned}U_{ADC} &= I * R_2 \\ &= \frac{VCC}{R_1 + R_2} * R_2 \\ &= \frac{R_2}{R_1 + R_2} * VCC \\ &= \frac{1}{\frac{R_1}{R_2} + 1} * VCC\end{aligned}$$

(1) 光照强，R2变小，AD变小

(2) 光照弱，R2变大，AD变大



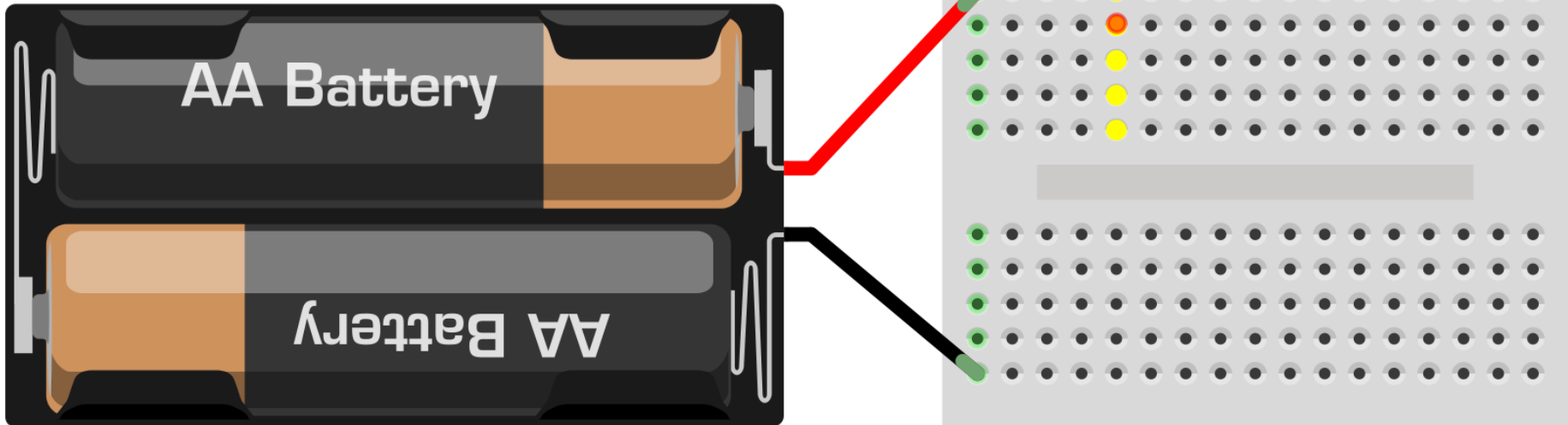
电源与面包板

电源正极

VCC

Volt Current Condenser的简写，
意思是电路的供电电压

列
连
通



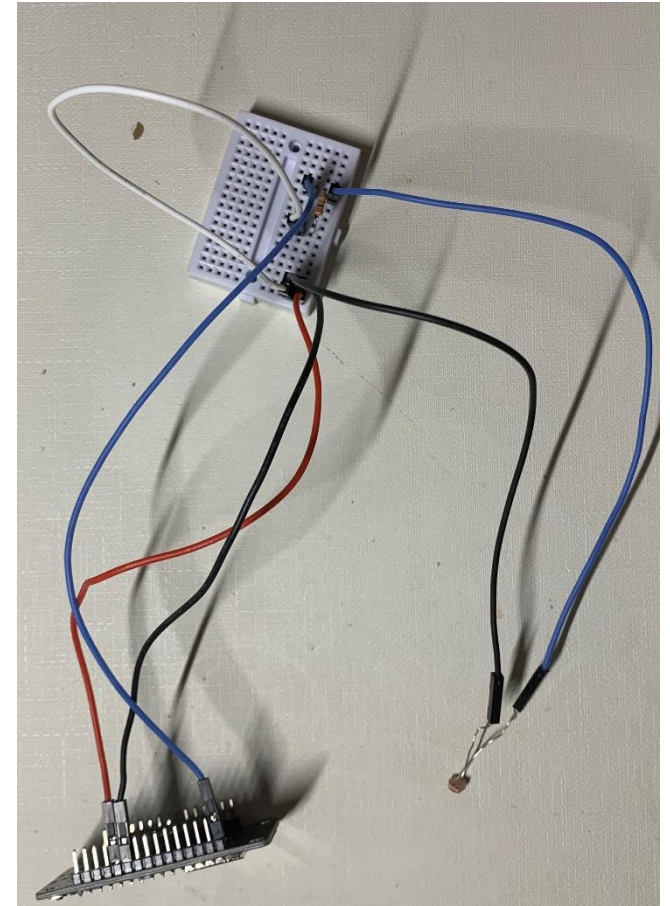
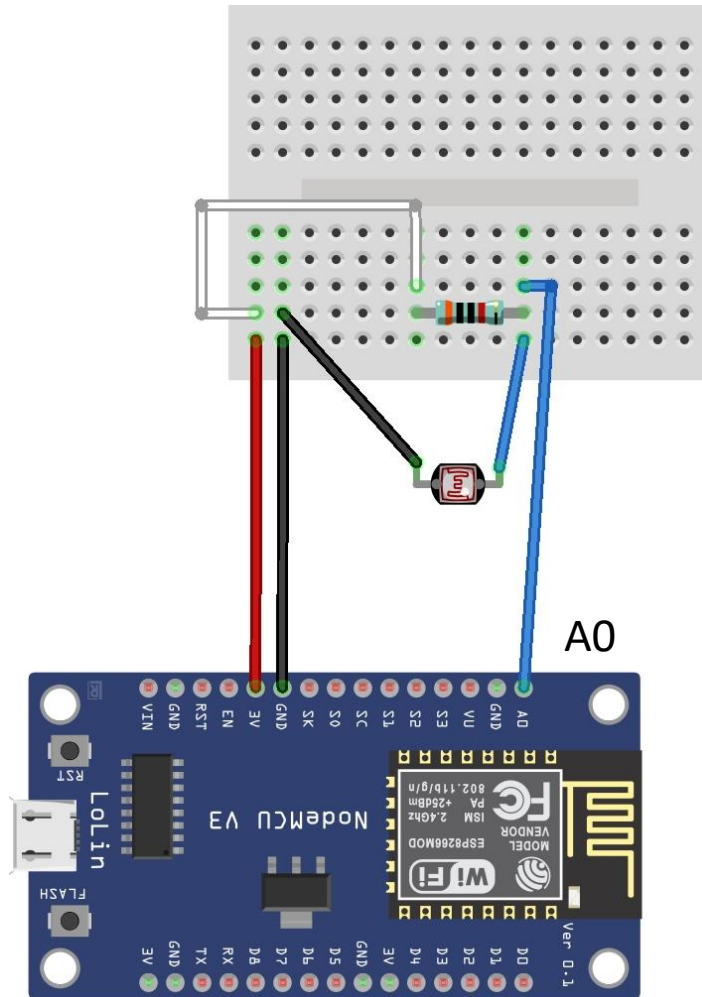
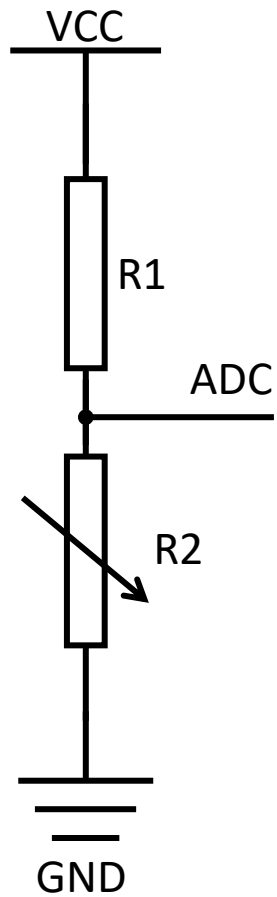
电源负极

GND

Ground的简写
意思是大地



电路连接





测光代码

```
1 from machine import ADC
2
3 # light sensor A0
4 adc = ADC(0)
5 print(adc.read())
```



```
MicroPython v1.14 on 2021-02-02; ESP module with ESP8266
Type "help()" for more information.
```

```
>>> %Run -c $EDITOR_CONTENT
```

```
137
```

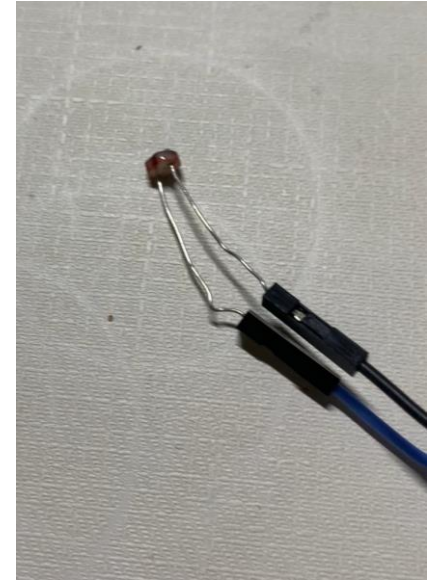
```
>>>
```



实验2：测光



```
1 from machine import ADC
2 import time
3
4 # light sensor A0
5 adc = ADC(0)
6 while True:
7     print(adc.read())
8     time.sleep_ms(100)
```



```
Shell x
0

MicroPython v1.14 on 2021-02-02; ESP module with ESP8266
Type "help()" for more information.
>>>
Backend terminated or disconnected. Use 'Stop/Restart' to restart.

MicroPython v1.14 on 2021-02-02; ESP module with ESP8266
Type "help()" for more information.
>>> %Run -c $EDITOR_CONTENT
137
>>>
```



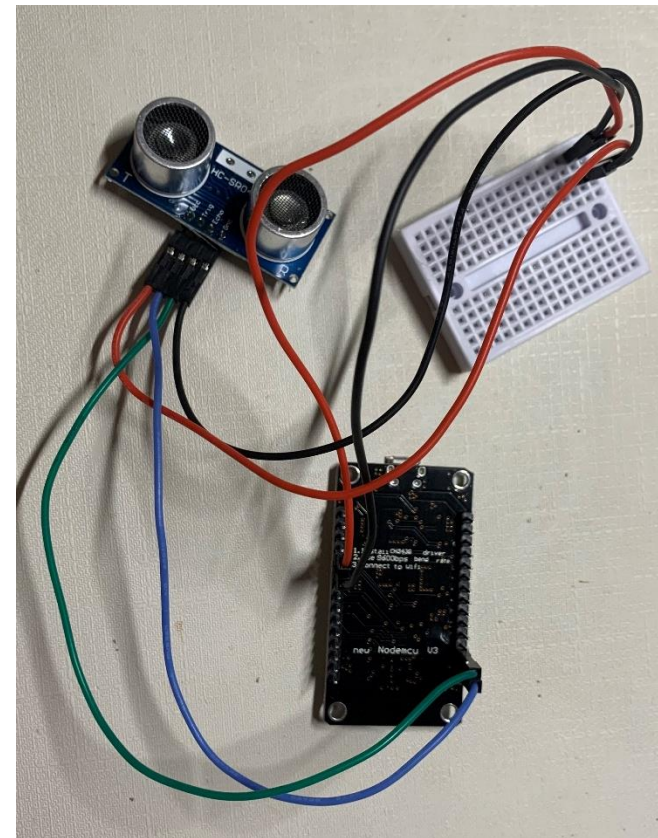
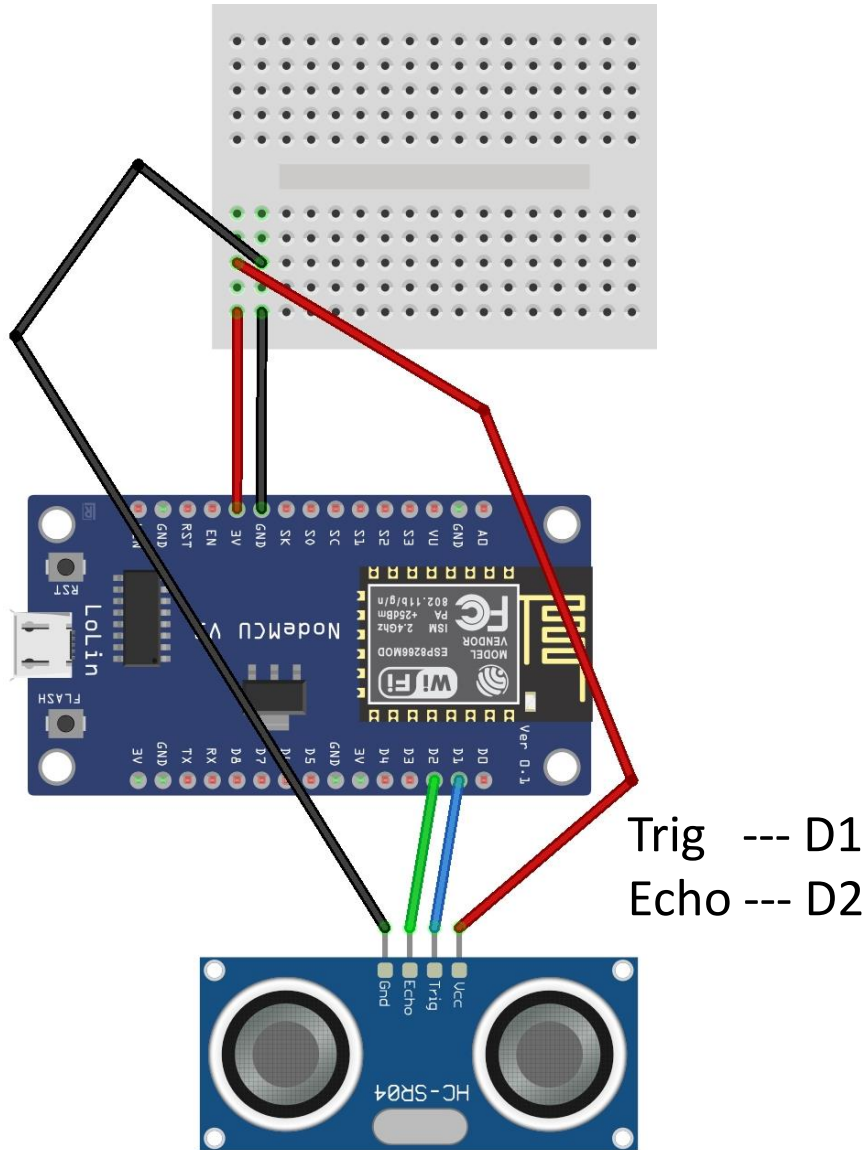
3. 超声波传感器

- 超声波传感器（Ultrasonic sensor）使用的“超声波”是一种声波。
- 20kHz以上、人耳听不到的声音称为**超声波**。
- 超声波传感器利用了压电陶瓷根据电压的方向进行伸缩并产生空气振动（超声波）。
- **功能**：测量发射波的时间，从而计算出距离





电路连接





超声波代码

```
1 import time
2 from machine import Pin
3
4 # distance sensor
5 trigPin = Pin(5, Pin.OUT) #GPIO5 D1
6 echoPin = Pin(4, Pin.IN) #GPIO4 D2
7
8 distance = measureDist()
9 print("distance: ", distance)
10
```

```
MicroPython v1.14 on 2021-02-02; ESP module with ESP8266
Type "help()" for more information.
```

```
>>> %Run -c $EDITOR_CONTENT
```

```
distance: 16
```

```
>>> %Run -c $EDITOR_CONTENT
```

```
distance: 50
```

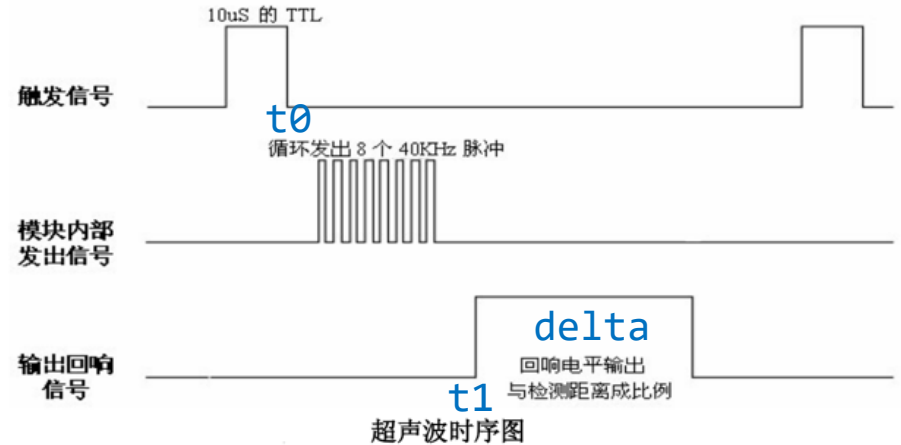
```
>>>
```



超声波代码

```
11 # Return distance (cm).
12 def measureDist():
13     trigPin.on()
14     time.sleep_us(15)
15     trigPin.off()
16
17     t0 = time.ticks_us()
18     # wait for 1
19     while not echoPin.value():
20         if time.ticks_diff(time.ticks_us(), t0)>5000: # 5s
21             return 0
22
23     t1= time.ticks_us() # time of rising edge
24     # wait for 0
25     while echoPin.value():
26         # limit measure distance 0.5m. 1/314=2.941ms
27         if time.ticks_diff(time.ticks_us(), t1)>2941:
28             # print("Out of range.")
29             return 50
30     delta = time.ticks_diff(time.ticks_us(), t1) # us
31     return int(delta*0.017) #cm
32
```

超声波时序图：





实验3：测距



```
1 import time
2 from machine import Pin
3
4 # distance sensor
5 trigPin = Pin(5, Pin.OUT) #GPIO5 D1
6 echoPin = Pin(4, Pin.IN)  #GPIO4 D2
7
8 while True:
9     distance = measureDist()
10    print("distance: ", distance)
11    time.sleep_ms(100)
```



```
Shell
MicroPython v1.14 on 2021-02-02; ESP module with ESP8266
Type "help()" for more information.
>>>
Backend terminated or disconnected. Use 'Stop/Restart' to restart.

MicroPython v1.14 on 2021-02-02; ESP module with ESP8266
Type "help()" for more information.
>>> %Run -c $EDITOR_CONTENT
distance: 16
>>> %Run -c $EDITOR_CONTENT
distance: 50
>>> |
```

谢谢！

