



# Python 示例程序使用

文件状态:	当前版本:	V1.2
[ ] 正在修改	作者:	Adolph
	完成日期:	2019.4.22
[ ] 正在发布	审核:	
	完成日期:	

## 版本历史

版本号	作者	修改日期	修改说明	审核	备注
V1.0	Adolph	2019.2.13	初始版本		
V1.1	Adolph	2019.3.7	修改路径		
V1.2	Adolph	2.19.4.22	修改路径		



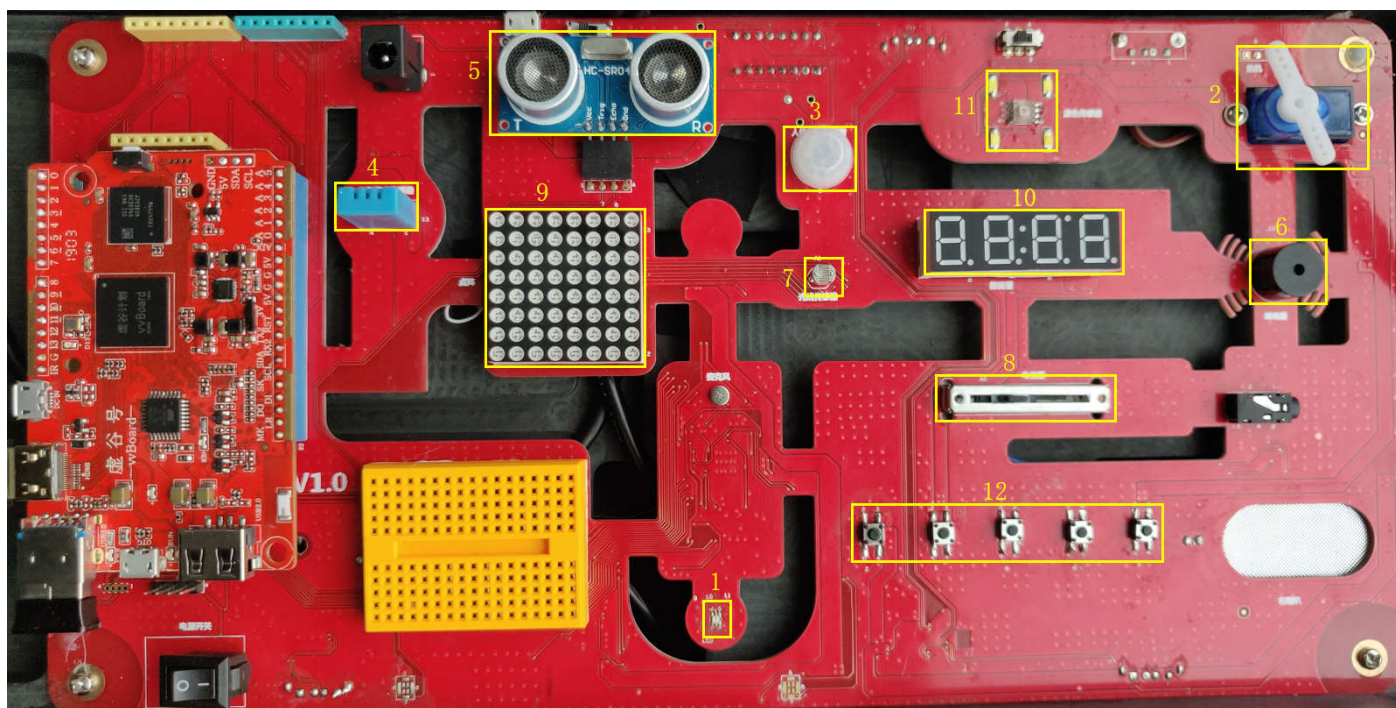
# 目录

前言: .....	3
实验准备 .....	4
实验 .....	5
一、 Led 灯实验 .....	5
二、 舵机实验.....	7
三、 人体红外感应实验.....	9
四、 温湿度测试实验.....	12
五、 超声波测距实验.....	14
六、 光敏电阻实验.....	16
七、 电位器实验.....	19
八、 蜂鸣器实验.....	21
九、 点阵实验.....	23
十、 数码管实验.....	25



## 前言：

本文档主要介绍虚谷号内置示例程序的测试方法和示例程序的说明。各 sensor 分布如下图所示：



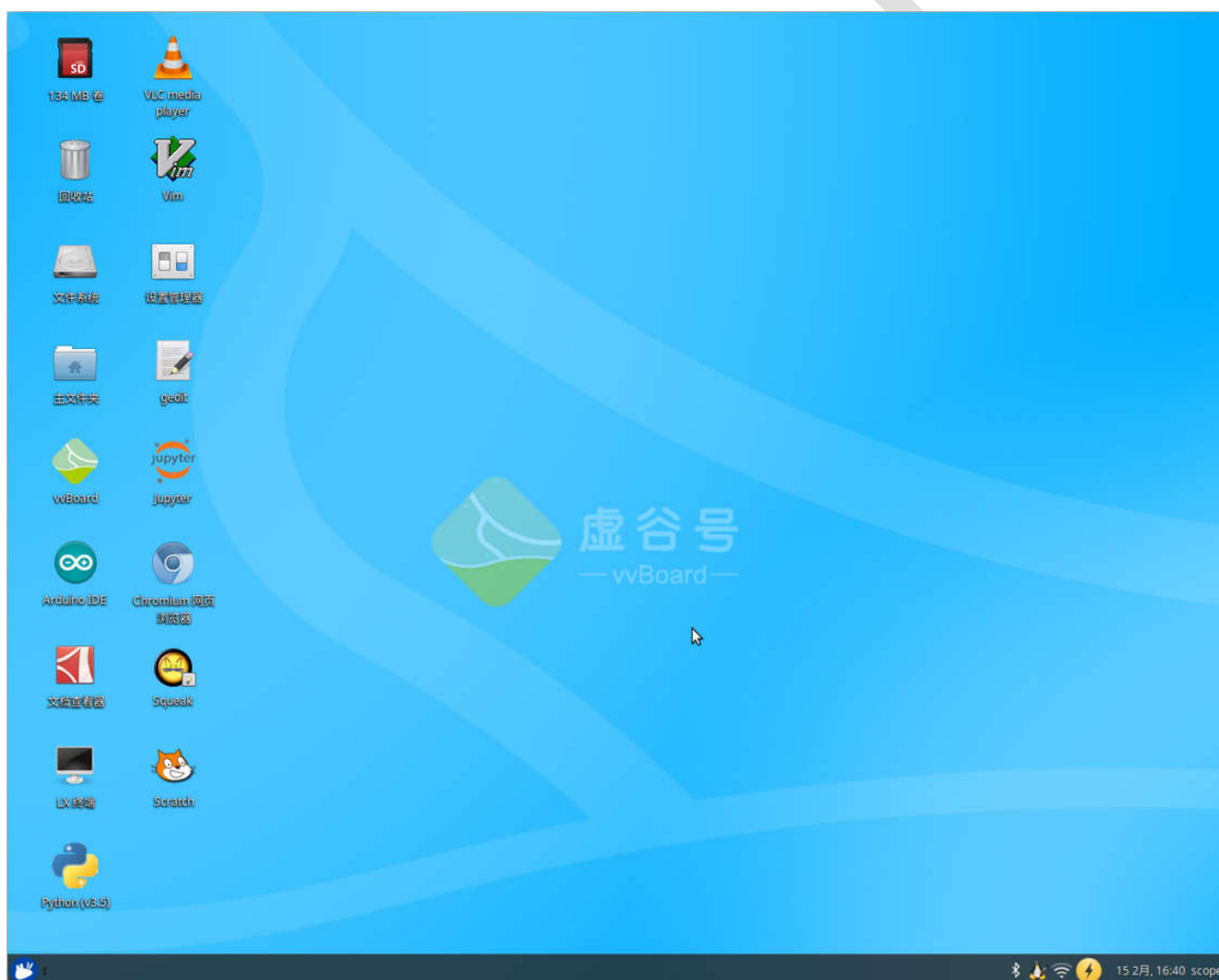
上图每个标号的 Sensor 介绍如下表：

标号	Sensor 名称
1	LED 灯
2	舵机
3	人体红外传感器
4	温湿度传感器
5	超声波传感器
6	蜂鸣器
7	光敏电阻
8	电位器
9	点阵
10	数码管
11	颜色传感器



## 实验准备

- 1、拿到实验箱并打开，先接通 HDMI 转接板的电源，后接通虚谷号板子的电源，等待进入系统桌面。开机后屏幕显示如下：



- 2、将下图所示的开关拨到指定的位置

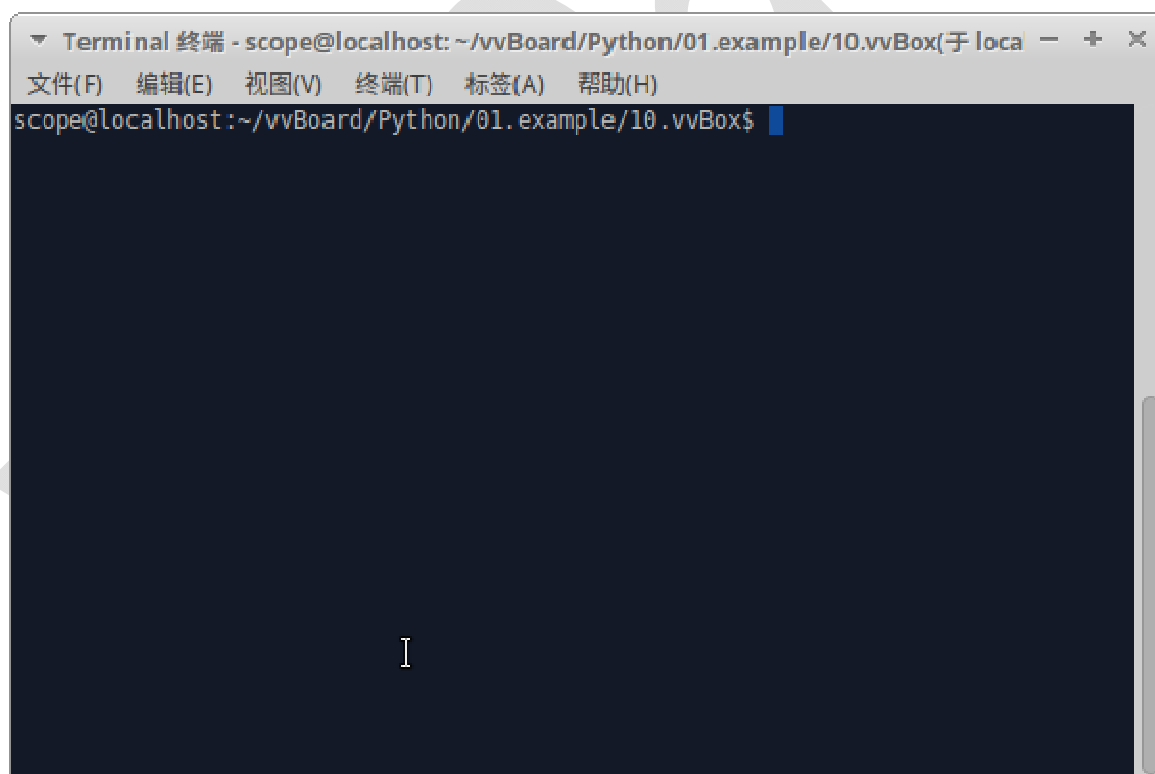


## 实验

### 一、Led 灯实验

#### 1、 实验步骤

- 1) 虚谷号进入桌面后，同时按住 Ctrl、Alt、t 按键，屏幕会出现一个命令窗口，在命令窗口中输入：`cd vvBoard/Python/01.example/10.vvBox/`回车会出现如下图所示的命令框：



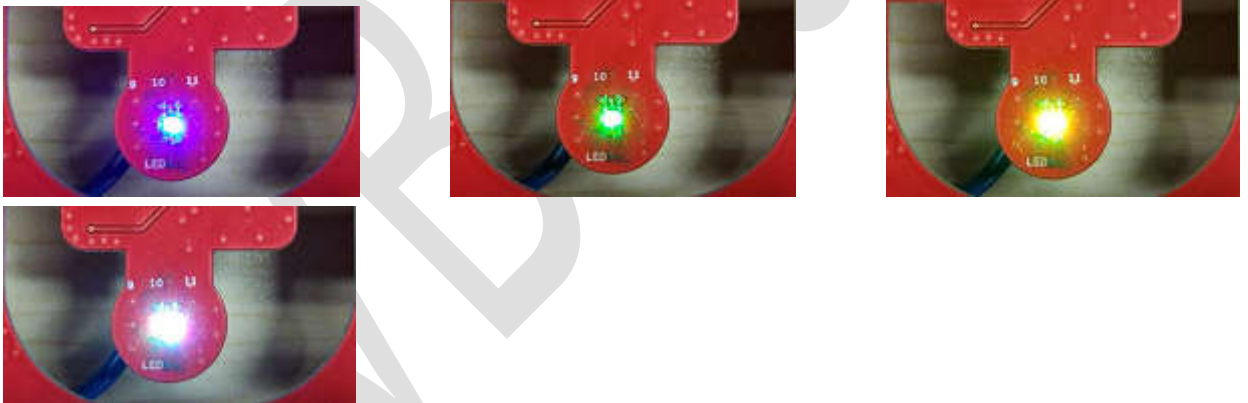
- 2) 在命令窗口中输入：`python Led_test.py` 后按下 Enter 键，LED sensor 就会依据 Led\_test.py 文件的代码执行相应的操作。运行图片如下：



```
Terminal 终端 - scope@localhost: ~/vvBoard/Python/01.example/10.vvBox(于 loca - + ×
文件(F) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)
scope@localhost:~/vvBoard/Python/01.example/10.vvBox$ python Led_test.py
pymata_aino Version 2.28 Copyright (c) 2015-2018 Alan Yorinks All rights reserved
.
Using COM Port:/dev/ttyS1
Initializing Arduino - Please wait...
Arduino Firmware ID: 2.5 StandardFirmata.ino
Auto-discovery complete. Found 20 Digital Pins and 6 Analog Pins
```

## 2、 实验现象

程序运行结束时 LED 灯会不停的闪烁不同的颜色，如下图所示：



## 3、 实验结束

运行结束后，LED sensor 闪烁也会停止。

## 4、 实验代码

Led 灯测试程序代码：

```
from xugu import * #导入 xugu 库
import time #导入时间模块
```



```
pin_r = Pin(9, Pin.OUT) #选择 soc 控制引脚
pin_g = Pin(10, Pin.OUT) #选择 soc 控制引脚
pin_b = Pin(11, Pin.OUT) #选择 soc 控制引脚
test = 20 #计数
l = [0, 1] #IO 口高低电平列表
while test > 0:
    pin_r.write_digital(l[0]) #默认写入列表的第一个参数
    time.sleep(1) #休眠一秒
    pin_g.write_digital(l[0]) #默认写入列表的第一个参数
    time.sleep(1) #休眠一秒
    pin_b.write_digital(l[0]) #默认写入列表的第一个参数
    l = l[: : -1] #列表参数交换
    time.sleep(1) #休眠一秒
    test-=1 #计数自减 1
```

## 二、舵机实验

### 1、实验步骤

- 1) 虚谷号进入桌面后，同时按住 Ctrl、Alt、t 按键，屏幕会出现一个命令窗口，在命令窗口中输入：cd vvBoard/Python/01.example/10.vvBox/回车会出现如下图所示的命令框：



```
Terminal 终端 - scope@localhost: ~/vvBoard/Python/01.example/10.vvBox(于 loca - + x
文件(F) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)
scope@localhost:~/vvBoard/Python/01.example/10.vvBox$
```

- 2) 在命令窗口中输入: `python Steering_gear_test.py` 后按下 Enter 键, 舵机 sensor 就会依据 `Steering_gear_test.py` 文件的代码执行相应的操作。运行图片如下:

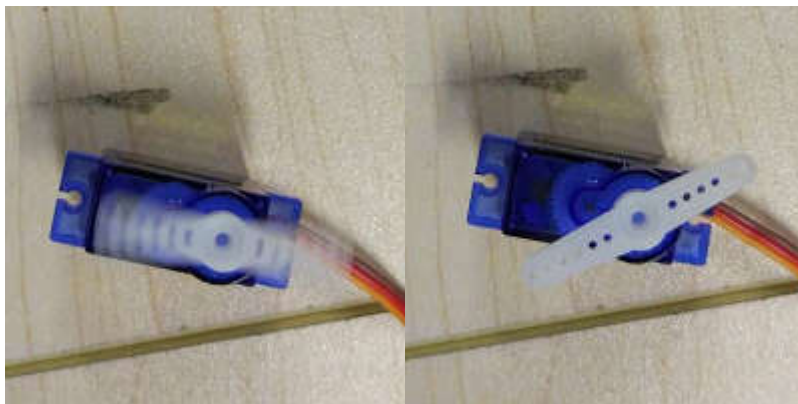
```
Terminal 终端 - scope@localhost: ~/vvBoard/Python/01.example/10.vvBox(于 loca - + x
文件(F) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)
scope@localhost:~/vvBoard/Python/01.example/10.vvBox$ python Steering_gear_test.py
pymata_aino Version 2.28 Copyright (c) 2015-2018 Alan Yorinks All rights reserved
.
Using COM Port:/dev/ttyS1
Initializing Arduino - Please wait...
Arduino Firmware ID: 2.5 StandardFirmata.ino
Auto-discovery complete. Found 20 Digital Pins and 6 Analog Pins
Task was destroyed but it is pending!
task: <Task pending coro=<PymataCore.command_dispatcher() running at /usr/local/lib/python3.5/dist-packages/pymata_aino/pymata_core.py:1400> wait_for=<Future finished result=None>>
scope@localhost:~/vvBoard/Python/01.example/10.vvBox$
```





## 2、 实验现象

程序运行时，舵机会左右转动，如下图所示：



## 3、 实验结束

程序运行结束后，舵机 sensor 会自动停止动作。

## 4、 实验代码

舵机测试程序代码：

```
from xugu import * #倒入 xugu 库
import time #倒入时间模块
servo = Servo(3) #选择 soc 控制引脚
test = 10 #计数
l = [180, 0] #角度列表
while test>0:
    servo.write_angle(l[0]) #默认写入列表的第一个参数
    l = l[: : -1] #列表参数交换
    time.sleep(1) #休眠一秒
    test-=1 #计数自减 1
```

# 三、 人体红外感应实验

## 1、 实验步骤

1) 虚谷号进入桌面后，同时按住 Ctrl、Alt、t 按键，屏幕会出现一个命令窗口，在命



令窗口中输入：`cd vvBoard/Python/01.example/10.vvBox/`回车会出现如下图所示的命令框：

```
Terminal 终端 - scope@localhost: ~/vvBoard/Python/01.example/10.vvBox(于 loca - + ×  
文件(F) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)  
scope@localhost:~/vvBoard/Python/01.example/10.vvBox$
```

- 2) 在命令窗口中输入：`python Human_body_induction_test.py` 后按下 Enter 键，人体红外 sensor 就会依据 `Human_body_induction_test.py` 文件的代码执行相应的操作。运行图片如下：



```
Terminal 终端 - scope@localhost: ~/vvBoard/Python/01.example/10.vvBox(于 loca - + ×
文件(F) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)
scope@localhost:~/vvBoard/Python/01.example/10.vvBox$ python Human_body_inductio
n_test.py

pymata_aino Version 2.28 Copyright (c) 2015-2018 Alan Yorinks All rights reserved
.

Using COM Port: /dev/ttyS1

Initializing Arduino - Please wait...
Arduino Firmware ID: 2.5 StandardFirmata.ino
Auto-discovery complete. Found 20 Digital Pins and 6 Analog Pins

no human!!!
no human!!!
no human!!!
no human!!!
no human!!!
no human!!!
no human!!!
no human!!!
no human!!!
no human!!!
no human!!!
no human!!!
no human!!!
```

## 2、 实验现象

程序运行时，当人体感应传感器检测到有人动时会显示 find human!!!，当没有检测到有人动的时候会显示 no human!!!。

## 3、 实验结束

程序运行结束后，人体红外 sensor 会停止工作。

## 4、 实验代码

人体红外测试程序代码：

```
from xugu import Pin #从 xugu 库中导入 Pin 类
import time #导入时间模块
pin = Pin(0, Pin.IN) #创建 Pin 对象
test = 30
while test>0:
    value = pin.read_digital() #读取对应 Pin 脚的电平值
    if value:
        print("find human!!!") #电平为 1 代表发现人
    else:
        print("no human!!!") #电平为 0 代表没发现人
```



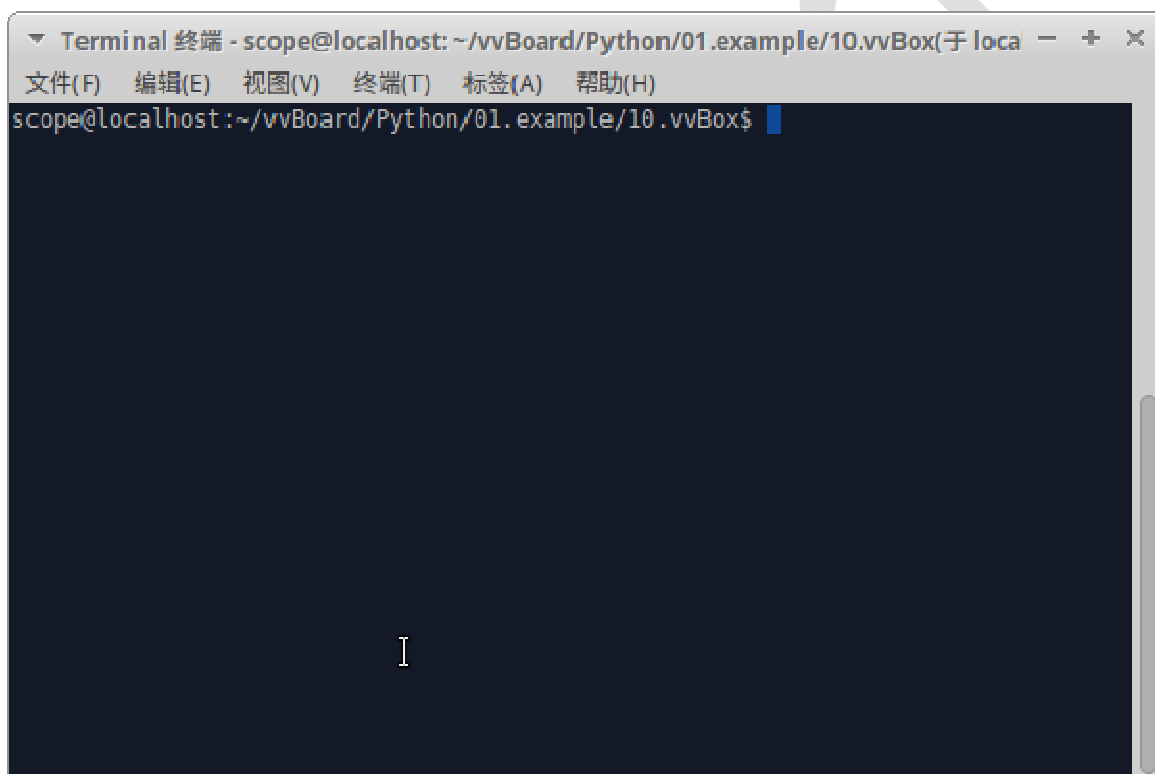
```
time.sleep(1) #睡眠 1 秒
```

```
test-=1
```

## 四、温湿度测试实验

### 1、 实验步骤

- 1) 虚谷号进入桌面后，同时按住 Ctrl、Alt、t 按键，屏幕会出现一个命令窗口，在命令窗口中输入：`cd vvBoard/Python/01.example/10.vvBox/sensor/`回车会出现如下图所示的命令框：



- 2) 在命令窗口中输入：`python Temperature_and_humidity_collection_platform.py`后按下 Enter 键，温湿度传感器就会依据 `Temperature_and_humidity_collection_platform.py` 文件的代码读取温湿度的值将显示在终端上。运行图片如下：



```
Terminal 终端 - scope@localhost: ~/vvBoard/Python/01.example/10.vvBox
文件(F) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)
scope@localhost:~/vvBoard/Python/01.example/10.vvBox$ python Temperature_and_humidity_collection_platform.py
not found firmata protocol, burn it.
Picked up JAVA_TOOL_OPTIONS:
Loading configuration...
Initializing packages...
Preparing boards...
Verifying...
Sketch uses 8292 bytes (25%) of program storage space. Maximum is 32256 bytes.
Global variables use 319 bytes (15%) of dynamic memory, leaving 1729 bytes for local variables. Maximum is 2048 bytes.
Uploading...
burn complete
温度:26.00 湿度:95.00%
温度:26.00 湿度:95.00%
温度:26.00 湿度:95.00%
温度:26.00 湿度:95.00%
温度:26.00 湿度:95.00%
温度:28.00 湿度:95.00%
温度:31.00 湿度:95.00%
温度:30.00 湿度:95.00%
温度:30.00 湿度:95.00%
温度:30.00 湿度:95.00%
```

## 2、 实验结果

程序运行时，我们对着温湿度传感器吹气，传感器会上报不同的温度和湿度上来，并显示在终端上；如下图的两个值，没吹气湿度是 95%，温度是 26 度；吹气后，湿度是 95%，温度是 28 度。

```
温度:26.00 湿度:95.00%
温度:28.00 湿度:95.00%
```

## 3、 实验结束

需要结束实验时，同时按下键盘的 Ctrl 和 C 按键结束程序

## 4、 实验代码

温湿度测试程序代码：

```
from dhtc import DHT #从 dhtc 库中导入 DHT 模块
import time #导入时间模块
dht = DHT() #创建 DHT 对象
```

```
def getTempandHum():
```

```
    """
```

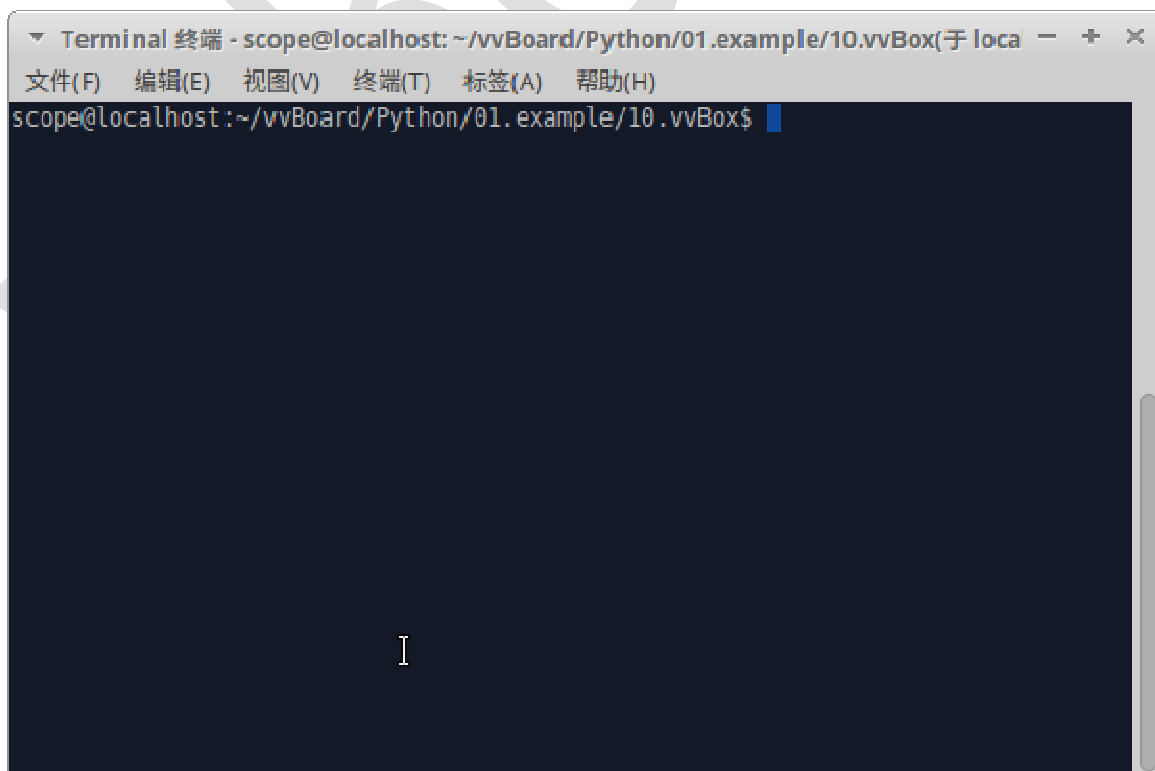


```
读取温度和湿度  
"""  
value = dht.read()  
if value != -1:  
    humi = value[:5]  
    Temp = value[-5:]  
    print("温度:%s 湿度:%s%%"%(Temp, humi))  
  
while True: #创建一个循环  
    getTempandHum()  
    time.sleep(1)
```

## 五、超声波测距实验

### 1、实验步骤

- 1) 虚谷号进入桌面后，同时按住 Ctrl、Alt、t 按键，屏幕会出现一个命令窗口，在命令窗口中输入：cd vvBoard/Python/01.example/10.vvBox/回车会出现如下图所示的命令框：





- 2) 超声波接入 sensor 板示意图如下:



- 3) 在命令窗口中输入: `python Ultrasonic_test.py` 后按下 Enter 键, 温湿度传感器就会依据 `Ultrasonic_test.py` 文件的代码读取温湿度的值将显示在终端上。运行图片如下:

```
Terminal 终端 - scope@localhost: ~/vvBoard/Python/01.example/10.vvBox
文件(F) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)
scope@localhost:~/vvBoard/Python/01.example/10.vvBox$ python Ultrasonic_test.py
not found firmata protocol, burn it.
Picked up JAVA_TOOL_OPTIONS:
Loading configuration...
Initializing packages...
Preparing boards...
Invalid library found in /home/scope/software/arduino-nightly/libraries/SR: /home/scope/software/arduino-nightly/libraries/SR
Invalid library found in /home/scope/software/arduino-nightly/libraries/SR: /home/scope/software/arduino-nightly/libraries/SR
Invalid library found in /home/scope/software/arduino-nightly/libraries/SR: /home/scope/software/arduino-nightly/libraries/SR
Invalid library found in /home/scope/software/arduino-nightly/libraries/SR: /home/scope/software/arduino-nightly/libraries/SR
Invalid library found in /home/scope/software/arduino-nightly/libraries/SR: /home/scope/software/arduino-nightly/libraries/SR
Verifying...
Sketch uses 4602 bytes (14%) of program storage space. Maximum is 32256 bytes.
Global variables use 204 bytes (9%) of dynamic memory, leaving 1844 bytes for local variables. Maximum is 2048 bytes.
Uploading...
[sudo] scope 的密码: burn complete
79
-1
75
78
```

## 2、 实验现象

程序运行时, 通过移动超声波传感器, 可以获取不同的距离, 当传感器贴近障碍物时, 距离显示为-1, 其他显示对应的距离值。

## 3、 实验结束

程序读取 30 次距离值后自动退出。



## 4、 实验代码

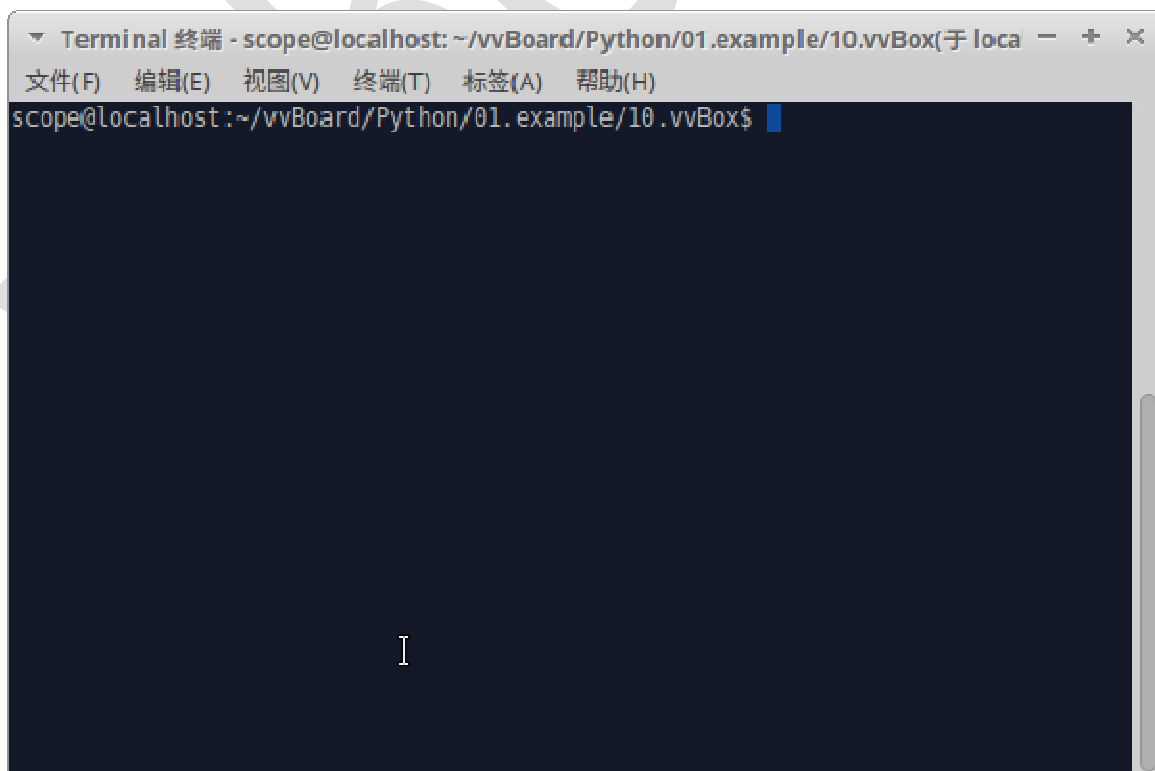
超声波测试程序代码：

```
from dhc import HC #从 dhc 模块中导入 HC 类
import time #导入时间模块
hc = HC() #创建 HC 对象
test = 30 #计数
while test>0:
    l = hc.read() #读取距离
    print(l) #在终端上输出距离
    time.sleep(1) #睡眠一秒
    test-=1 #计数自减 1
```

## 六、 光敏电阻实验

### 1、 实验步骤

- 1) 虚谷号进入桌面后，同时按住 Ctrl、Alt、t 按键，屏幕会出现一个命令窗口，在命令窗口中输入：cd vvBoard/Python/01.example/10.vvBox/回车会出现如下图所示的命令框：







- 2) 在命令窗口中输入: `python LightSensor_test.py` 后按下 Enter 键, 程序就会依据 `LightSensor_test.py` 文件的代码从 arduino 的 A1 模拟输入口读取对应的值。运行图片如下:

```
Terminal 终端 - scope@localhost: ~/vvBoard/Python/01.example/10.vvBox
文件(F) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)
scope@localhost:~/vvBoard/Python/01.example/10.vvBox$ python LightSensor_test.py

pymata_aino Version 2.28 Copyright (c) 2015-2018 Alan Yorinks All rights reserved
.

Using COM Port: /dev/ttyS1

Initializing Arduino - Please wait...
Arduino Firmware ID: 2.5 StandardFirmata.ino
Auto-discovery complete. Found 20 Digital Pins and 6 Analog Pins

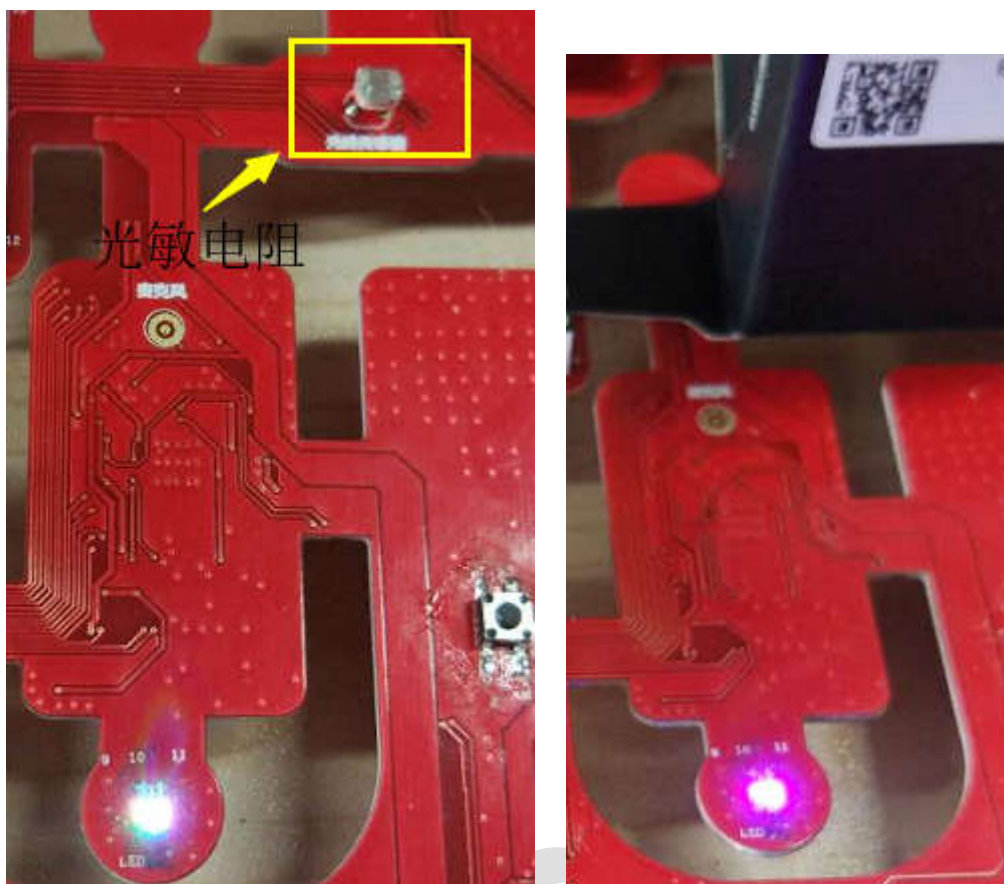
469
850
177
758
155
157
153
155
156
154
154
```

## 2、 实验现象

程序运行时, 当我们没有遮挡光敏电阻时, 读出 A1 脚在当前的亮度下的值是 96 左右; 当我们用东西遮挡住光敏电阻时, A1 脚的读数就会上升到 551 左右, 这时 LED 灯蓝灯将被点亮, 如下图所示:

光敏电阻无遮挡现象

光敏电阻有遮挡现象



### 3、 实验结束

程序在 30 次计数结束后自动退出。

### 4、 实验代码

光敏电阻测试程序代码

```
from xugu import Pin #从 xugu 库中导入 Pin 类
import time #导入时间模块
lightSensor_pin = "a1" #定义模拟输入引脚
led_pin = 11 #定义 LED 控制引脚
lightSensor = Pin(lightSensor_pin, Pin. ANALOG) #创建 Pin 对象
led = Pin(led_pin, Pin. OUT) #创建 Pin 对象
test = 30 #计数
while test > 0:
    value = lightSensor.read_analog() #读取 A1 脚的模拟输入值
```

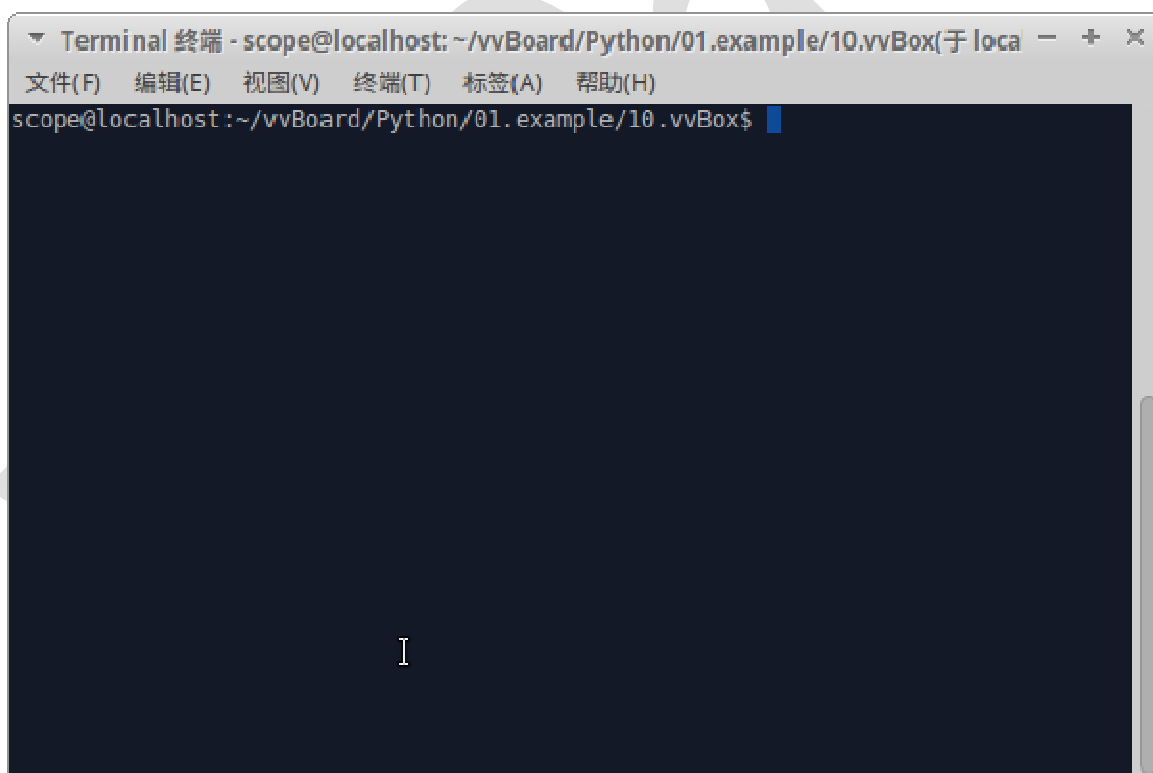


```
print(value)
if value > 200: #判断 A1 脚获取的值是否大于 200
    led.write_digital(1) #点亮蓝色的 LED 灯
else:
    led.write_digital(0) #熄灭 LED 灯
time.sleep(1) #睡眠 1 秒
test -= 1 #计数自减 1
led.write_digital(0)
```

## 七、 电位器实验

### 1、 实验步骤

- 1) 虚谷号进入桌面后，同时按住 Ctrl、Alt、t 按键，屏幕会出现一个命令窗口，在命令窗口中输入：cd vvBoard/Python/01.example/10.vvBox/回车会出现如下图所示的命令框：



- 2) 在命令窗口中输入：python Potentiometer\_test.py 按下 Enter 键后，程序就会依据 Potentiometer\_test.py 文件的代码从 arduino 的 A0 模拟输入口读取对应的值。运行图片如下：



```
Terminal 终端 - scope@localhost: ~/vvBoard/Python/01.example/10.vvBox
文件(F) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)
scope@localhost:~/vvBoard/Python/01.example/10.vvBox$ python Potentiometer_test.py
pymata_ao Version 2.28 Copyright (c) 2015-2018 Alan Yorinks All rights reserved
.
Using COM Port:/dev/ttyS1
Initializing Arduino - Please wait...
Arduino Firmware ID: 2.5 StandardFirmata.ino
Auto-discovery complete. Found 20 Digital Pins and 6 Analog Pins

Terminal 终端 - scope@localhost: ~/vvBoard/example/Python/sensor
文件(F) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)
pymata_ao Version 2.28 Copyright (c) 2015-2018 Alan Yorinks All rights reserved
.
Using COM Port:/dev/ttyS1
Initializing Arduino - Please wait...
Arduino Firmware ID: 2.5 StandardFirmata.ino
Auto-discovery complete. Found 20 Digital Pins and 6 Analog Pins

0
0
1023
1023
1023
1023
415
415
416
415
415
415
415
415
```

## 2、 实验现象

程序运行时，我们拨动电位器到如下图片的位置时



我们终端将获得对应的值，分别是 0、415 和 1023

## 3、 实验结束

程序在 30 次计数结束后自动退出。

## 4、 实验代码

电位器测试程序代码

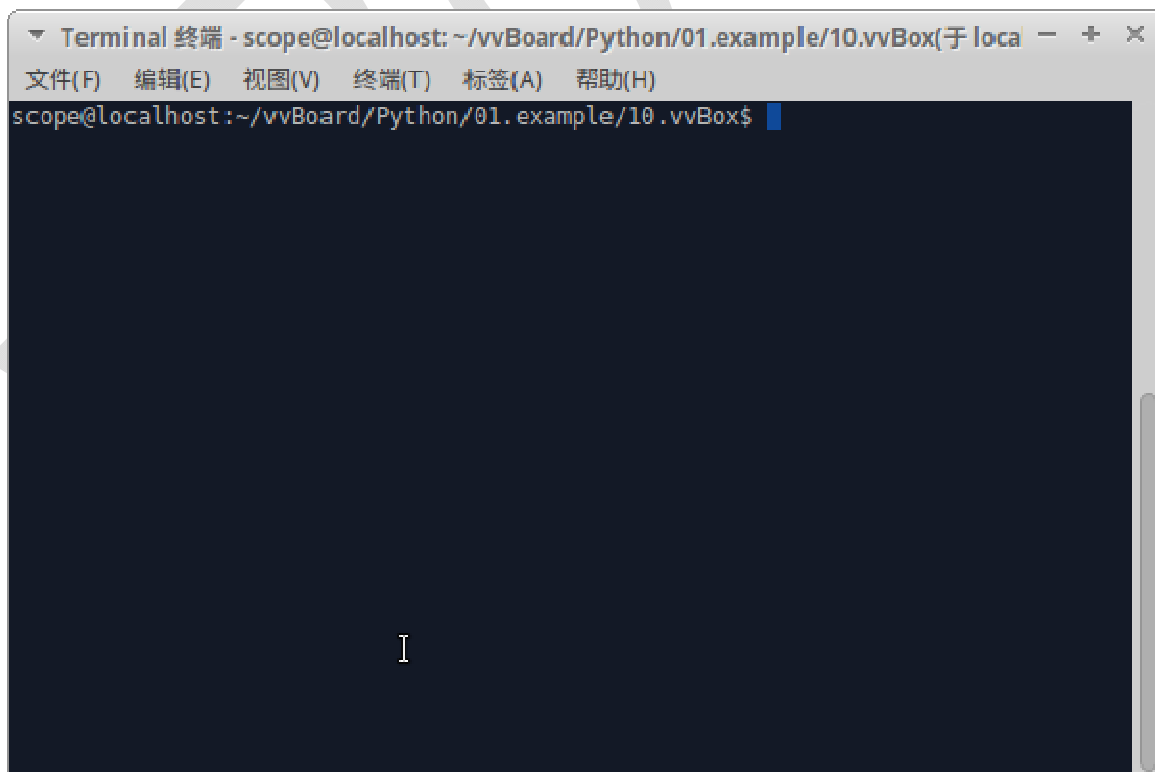


```
from xugu import Pin #从 xugu 库中导入 Pin 类
import time #导入时间模块
potentiometer_pin = "a0" #定义模拟输入引脚
potentiometer = Pin(potentiometer_pin, Pin.ANALOG) #创建 Pin 对象
test = 30 #计数
while test > 0:
    value = potentiometer.read_analog() #读取 A0 脚的模拟输入值
    print(value) #将读取的值打印到终端
    time.sleep(1) #睡眠 1 秒
    test -= 1 #计数自减 1
```

## 八、蜂鸣器实验

### 1、实验步骤

- 1) 虚谷号进入桌面后，同时按住 Ctrl、Alt、t 按键，屏幕会出现一个命令窗口，在命令窗口中输入：cd vvBoard/Python/01.example/10.vvBox/回车会出现如下图所示的命令框：



- 2) 在命令窗口中输入：python Buzzer\_test.py 后按下 Enter 键，蜂鸣器就会依据



Buzzer\_test.py 文件的代码发出声音。运行图片如下：

```
Terminal 终端 - scope@localhost: ~/vvBoard/Python/01.example/10.vvBox
文件(F) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)
scope@localhost:~/vvBoard/Python/01.example/10.vvBox$ python Buzzer_test.py
pymata_aido Version 2.20 Copyright (c) 2015-2018 Alan Yorinks All rights reserved
.
Using COM Port: /dev/ttyS1

Initializing Arduino - Please wait...
Arduino Firmware ID: 2.5 StandardFirmata.ino
Auto-discovery complete. Found 20 Digital Pins and 6 Analog Pins

Task was destroyed but it is pending!
task: <Task pending coro=<PymataCore._command_dispatcher() running at /usr/local
/lib/python3.5/dist-packages/pymata_aido/pymata_core.py:1400> wait_for=<Future fi
nished result=None>>
scope@localhost:~/vvBoard/Python/01.example/10.vvBox$
```

## 2、 实验现象

程序运行时，我们可以听到蜂鸣器发出滴滴滴的声音

## 3、 实验结束

蜂鸣器响 30 声后自动退出。

## 4、 实验代码

蜂鸣器测试程序代码

```
from xugu import * #导入 xugu 库
import time #导入时间模块
buzzer = Pin(12, Pin.OUT) #创建 Pin 对象
test = 30 #计数
while test > 0:
    buzzer.write_digital(1) #拉高 soc 的 12 脚
    time.sleep(0.1) #睡眠 100 毫秒
    buzzer.write_digital(0) #拉低 soc 的 12 脚
    time.sleep(0.1)
```



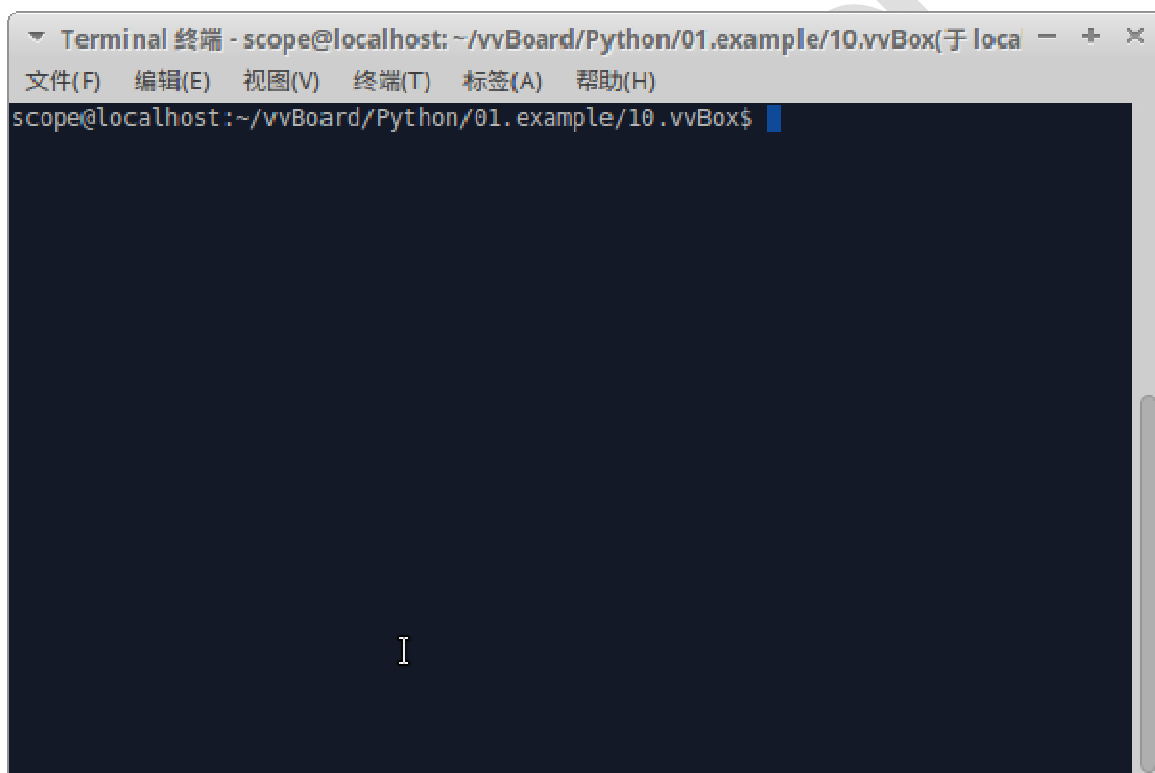
```
test -= 1 #计数自减 1
```

```
buzzer.write_digital(0) #拉低 soc 的 12 脚
```

## 九、点阵实验

### 1、实验步骤

- 1) 虚谷号进入桌面后，同时按住 Ctrl、Alt、t 按键，屏幕会出现一个命令窗口，在命令窗口中输入：`cd vvBoard/Python/01.example/10.vvBox/`回车会出现如下图所示的命令框：



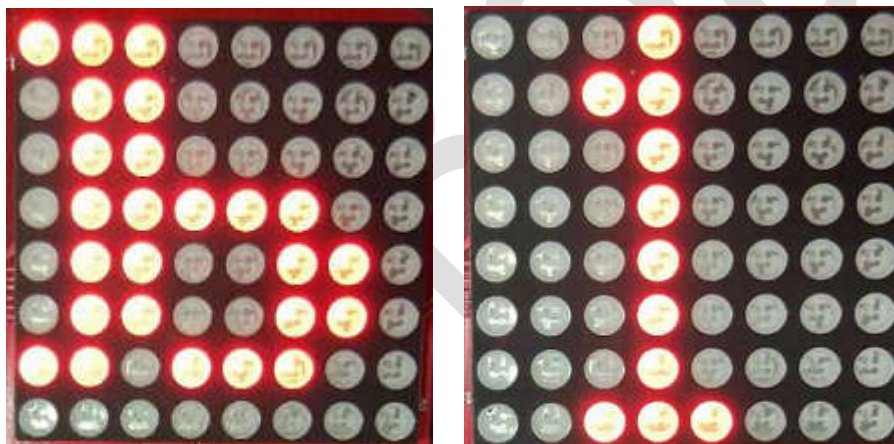
- 2) 在命令窗口中输入：`python Matrix_test.py` 按下 Enter 键后，程序就会依据 `Matrix_test.py` 文件的代码向点阵芯片写入对应的值。运行图片如下：



```
Terminal 终端 - scope@localhost: ~/vvBoard/Python/01.example/10.vvBox
文件(F) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)
scope@localhost:~/vvBoard/Python/01.example/10.vvBox$ python Matrix_test.py
not found firmata protocol, burn it.
Picked up JAVA_TOOL_OPTIONS:
Loading configuration...
Initializing packages...
Preparing boards...
Verifying...
Sketch uses 7866 bytes (24%) of program storage space. Maximum is 32256 bytes.
Global variables use 1357 bytes (66%) of dynamic memory, leaving 691 bytes for l
ocal variables. Maximum is 2048 bytes.
Uploading...
[sudo] scope 的密码: burn complete
```

## 2、 实验现象

程序运行时，点阵芯片会显示对应的字符，如下图所示：



## 3、 实验结束

需要结束实验时，同时按下键盘的 Ctrl 和 C 按键结束程序

## 4、 实验代码

```
from mtx import MTX #从 mtx 库中导入 MTX 类
import time #导入时间模块
mtx = MTX() #创建 MTX 类
val = "123abc" #将需要显示的值赋值给 val
```



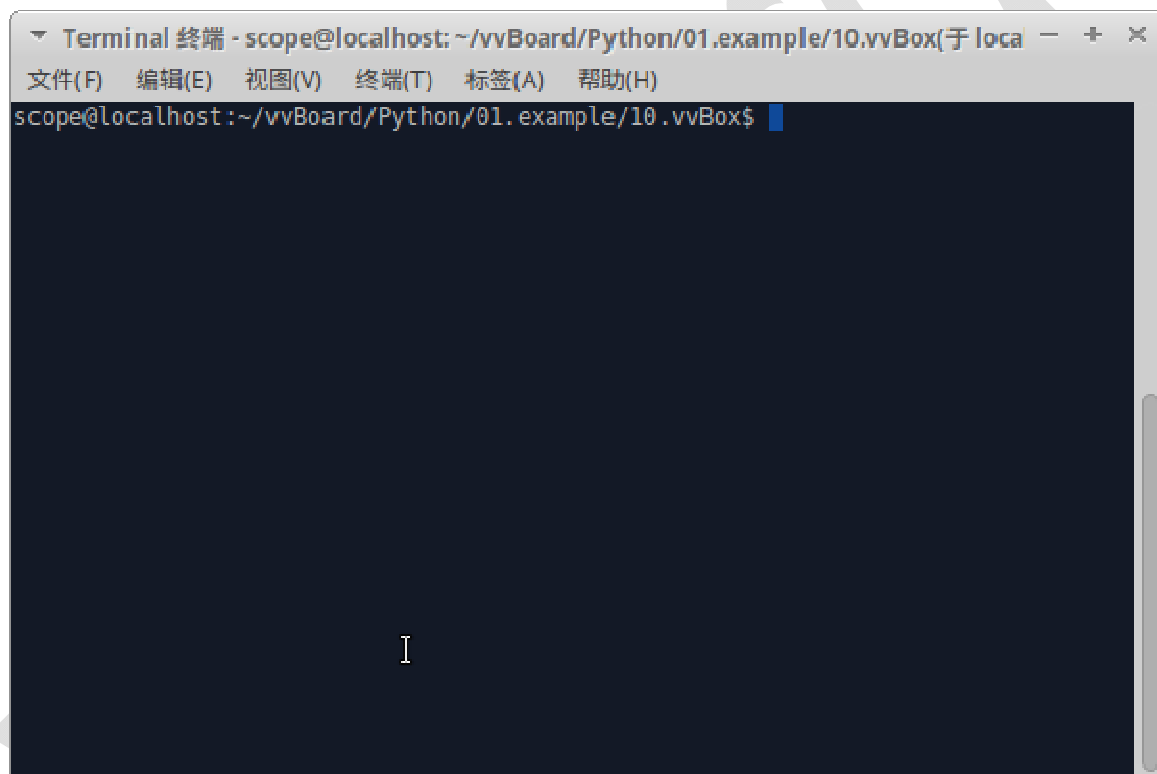


```
while True: #创建循环
    if mtx.burn_complete: #判断 arduino 驱动是否烧入完毕
        mtx.write(val) #写入 val 的值
        time.sleep(6) #休眠 6 秒
```

## 十、数码管实验

### 1、 实验步骤

- 1) 虚谷号进入桌面后，同时按住 Ctrl、Alt、t 按键，屏幕会出现一个命令窗口，在命令窗口中输入：`cd vvBoard/Python/01.example/10.vvBox/`回车会出现如下图所示的命令框：



- 2) 在命令窗口中输入：`python Nixietube_test.py` 按下 Enter 键后，程序就会依据 `Nixietube_test.py` 文件的代码向数码管芯片写入对应的值。运行图片如下：



```
Terminal 终端 - scope@localhost: ~/vvBoard/Python/01.example/10.vvBox
文件(F) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)
scope@localhost:~/vvBoard/Python/01.example/10.vvBox$ python Nixietube_test.py
not found firmata protocol, burn it.
Picked up JAVA_TOOL_OPTIONS:
Loading configuration...
Initializing packages...
Preparing boards...
Verifying...
Sketch uses 5258 bytes (16%) of program storage space. Maximum is 32256 bytes.
Global variables use 236 bytes (11%) of dynamic memory, leaving 1812 bytes for l
ocal variables. Maximum is 2048 bytes.
Uploading...
[sudo] scope 的密码: burn complete
```

## 2、实验现象

程序运行时，数码管先从左边第 0 位数码管从 0~9 点亮，之后是点亮第 1 位数码管，以此类推，现象如下图片所示：

点亮第 0 位数码管



点亮第 2 位数码管



点亮第 1 位数码管



点亮第 3 位数码管



## 3、实验结束

需要结束程序时，同时按下键盘的 Ctrl 和 C 按键结束程序

## 4、实验代码

```
from tm1637 import TM1637 #从 tm1637 库中导入 TM1637 类
import time #导入时间模块
tm1637 = TM1637() #创建 TM1637 类
val = "0123456789" #将数码管需要显示的值赋值给 val
com = "0123" #将数码管的位数赋值给 com
```



```
i = 0 #数码管位数计数
j = 0 #显示内容计数
tm1637.clearDisplay() #数码管清除显示
while True: #创建循环
    for letter in com: #遍历 com 字符串
        for letter in val: #便利 val 字符串
            tm1637.display(com[i],val[j]) # 数码管的第 i 位显示字符串的第 j 数字
            time.sleep(1) #休眠 1 秒
            j += 1 #j 计数加 1
            i += 1 #i 计数加 1
            j = 0 #将 j 计数重新赋值为 0
            i = 0 #将 i 计数重新赋值为 0
tm1637.clearDisplay() #数码管清除显示
time.sleep(1) #休眠 1 秒
```