**树莓派实验指导书**

树莓派Raspberry Pi(中文名为“树莓派”,简写为RPi，是为学习计算机编程教育而设计，只有信用卡大小的微型电脑，其系统基于Linux，我们经常利用他的强大功能实现各种想法。树莓派的官网是：<https://www.raspberrypi.org/>，在网络上搜索树莓派也能搜到大量中文、英文学习资料。不过最新版本的树莓派（即本实验所采用的树莓派）采用了一款封闭结构的芯片，芯片的文档为保密状态，这给树莓派未来的发展带来了一丝隐忧（**为什么？**）

本课程给出了一个基于树莓派的小实验，主要用于完成树莓派开发环境的搭建及熟悉树莓派的基本使用方法，为课程大实验及同学们未来的兴趣发展的打下基础。

# 一、实验简介

**实验名称**：树莓派拍照回传实验

**实验目的**：熟悉树莓派的使用、搭建简单的传输网络，自动获取远程树莓派拍摄的图片。

**实验步骤**：

1. 树莓派实验环境的搭建
2. 测试摄像头，编写开机运行脚本
3. 建立互联网路进行通信

**实验扩展**：

以下实验扩展为同学选做部分，只要选做一个实验选题即可。请不要拘泥于下面给出的选题，请自由选题。

1. PC端可发送命令，树莓派收到命令后，可发送当前时刻前n帧和后n帧的图像，n为1-3。
2. 树莓派可以连续记录视频数据，每5分钟保留一段视频文件，循环覆盖。收到PC端的命令后将当前时刻前n秒和后n秒的视频合成为一个视频文件发送给PC端。
3. 树莓派在采集视频时进行人体（或其他目标）捕捉，捕捉到人体后自动回传图片。
4. 其他新奇想法

# 二、实验环境搭建

## 1）如何面对“三无”难题

为了将树莓派顺利的用起来，网上有很多树莓派相关的教程，大家可以自行尝试。针对本实验的具体情况，同学们使用自己的笔记本电脑在非固定环境下开展实验，往往会面临“三无”状态，即无显示器、无键盘鼠标、无连接方式，因此没有办法看到树莓派的输出信息以及向树莓派输入命令，怎么办？

本实验推荐的解决方案是：

1. 利用网线借用电脑的WiFi互联网连接共享实现树莓派的连接和Internet互联；
2. 利用VncViewer工具借用笔记本电脑的桌面窗口当做树莓派的显示器；
3. 利用ssh工具和VncViewer做为树莓派的键盘鼠标。

## 2）将系统写入microSD卡中

实验环境中提供了一个microSD卡，大家可以把它理解为树莓派的硬盘，同学们需要首先在这个“硬盘”中安装上操作系统才可以把树莓派启动起来。具体方法如下：

1. **下载树莓派**

从树莓派官网（或者其他地方）下载树莓派系统的image文件（下载完成后需要解压缩），下载地址如下：

https://www.raspberrypi.org/downloads/raspbian/

1. **写入image文件**

在笔记本电脑上将上述下载好的image文件写入到microSD卡中，但是需要专门的写入工具（**为什么？**），本实验推荐使用树莓派官网推荐的工具软件Win32 Disk Imager，下载地址为：

https://sourceforge.net/projects/win32diskimager/

注意！！！，向microSD卡写入image文件之前，请一定确认写入的目标磁盘是microSD卡，以免误操作错误的把笔记本电脑中的硬盘内容全部清除。

1. **存储卡容量变小了？**

有心的同学会发现16GB的microSD卡在写入image文件后容量变为只有几十兆的大小（**为什么？**），大家不用担心，microSD卡既没有被写入工具软件损坏，也没有容量变小，可以通过一些专门的格式化工具重新格式化即可恢复原有容量大小。本实验推荐SD\_CardFormatter工具，下载地址为：

https://www.sdcard.org/downloads/formatter\_4/index.html

1. **ssh必须的一步**

完成image文件写入后，请在microSD卡中建立一个名为ssh的文件夹或者一个名为ssh的空文件，以便后续能够顺利通过网络连接树莓派。

最新版本的树莓派默认是关闭了ssh连接的权限，研发团队为什么要这么做呢？（为什么？）

## 3）从“三无”变为“三有”

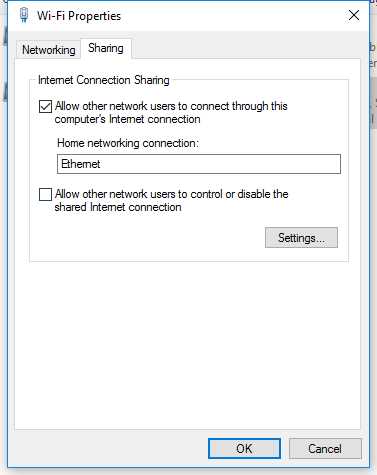
1. **树莓派加电启动**

a）将microSD卡插入树莓派中，b）通过microUSB线给树莓派供电，c）并用网线将树莓派和笔记本电脑连接起来。

以上三点缺一不可，否则下一步可能就不能成功建立网络连接。

1. **设置好笔记本电脑的网络**

本实验假定同学们笔记本电脑使用的是Windows系统，如果是Mac或Linux系统以及特殊的Windows版本，或者是硬件上有特殊的问题，请根据下文的思路自行上网寻找解决方案。

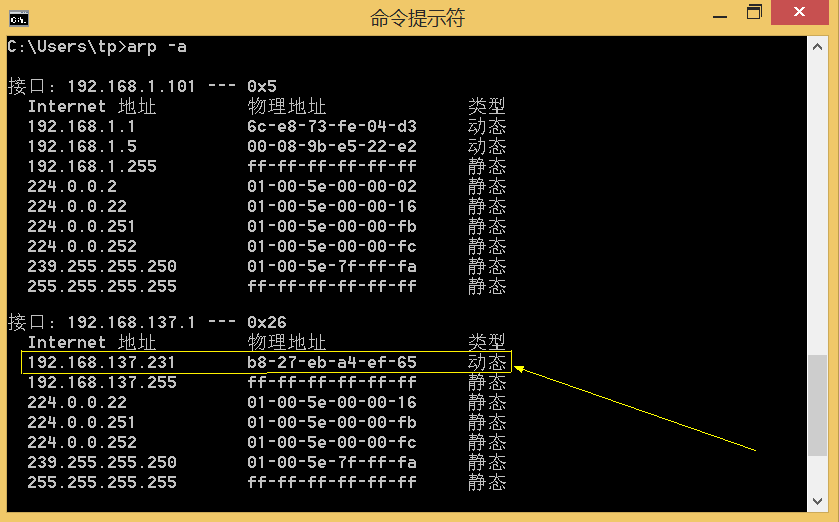


打开你的Windows的网络与共享中心-> 更改适配器选项

打开无线网卡适配器的属性->选择共享选项卡->允许其他网络用户通过这台电脑连接->下拉列表框中选择你的家庭网络连接为你的有线网卡（如上图所示），紧接着系统会提示（有可能不提示）并更改你的有线网卡的地址为 192.168.137.1。

1. **通过SSH终端连接树莓派**

网线成功连接以后，可以打开自己电脑的命令提示符，输入”arp -a”查看树莓派的IP地址，在192.168.137.1下面的就是，这个IP地址应该是[192.168.137.xxx](http://link.zhihu.com/?target=http%3A//192.168.137.xxx)，并且类型是动态的。



提示，ARP命令用于显示和修改“地址解析协议（ARP）”缓存中的项目。ARP工作时，首先请求主机发送出一个含有所希望到达的IP地址的以太网广播数据包，然后目标IP的所有者会以一个含有IP和MAC地址对的数据包应答请求主机。这样请求主机就能获得要到达的IP地址对应的MAC地址，同时请求主机会将这个地址对放入自己的ARP表缓存起来，以节约不必要的ARP通信。ARP缓存表采用了老化机制，在一段时间内如果表中的某一行没有使用（Windows系统这个时间为2分钟，而Cisco路由器的这个时间为5分钟），就会被删除。

确定了树莓派的IP地址以后，就可以通过SSH客户端软件(比如：PuTTY, pittey，SecureSSH，Xshell，CMDer等），通过ssh方式连接192.168.137.xxx登录树莓派系统，登录账号是：“pi”，默认密码是 “raspberry”。这样就可以成功的通过命令行控制和使用你的树莓派了，并且这时候树莓派是可以借用笔记本的网络连上Internet的，这相当于实现了树莓派的连接及通过命令行方式的控制与反馈。

* **扩展讨论：**

嵌入式开发是在宿主机（Host，即同学们的笔记本电脑）上对目标机（Target，即树莓派）进行开发，因此需要在二者之间建立联系。联系的方案有两种：有线、无线。

有线连接有网线连接、USB线连接、串口连接、JTAG线连接等多种方式，比较常见的是这里列出的4种。JTAG比较底层，当前嵌入式开发板基本上不再对普通开发者暴露JTAG接口；串口连接很方便，但是标准9针串口接头太大，小尺寸嵌入式开发板不愿意留出该接口，而普通插针接头不方便使用，需要使用者制作专门的接头线；USB连接需要嵌入式开发板自带USB转串口硬件功能模块及相应驱动，或者开发板直接具有内置USB驱动相关的软硬件结构；网络连接需要宿主机有网线插座，否则需要相应的转接头，但网络连接需要配置或查询IP地址信息，也不够方便易用。

无线连接方式一般是通过WiFi、蓝牙等无线方式建立宿主机与目标机之间的连接，需要解决连接确认（比如蓝牙的配对过程）、可靠的无线连接等问题，也没有在产业生态系统中取得一致的共识，形成统一、易用的解决方案。

综上所述，针对嵌入式开发板建立连接的过程，产业链还未给出公认有效的解决方案，仍然是产业和技术共同面临的一个问题。

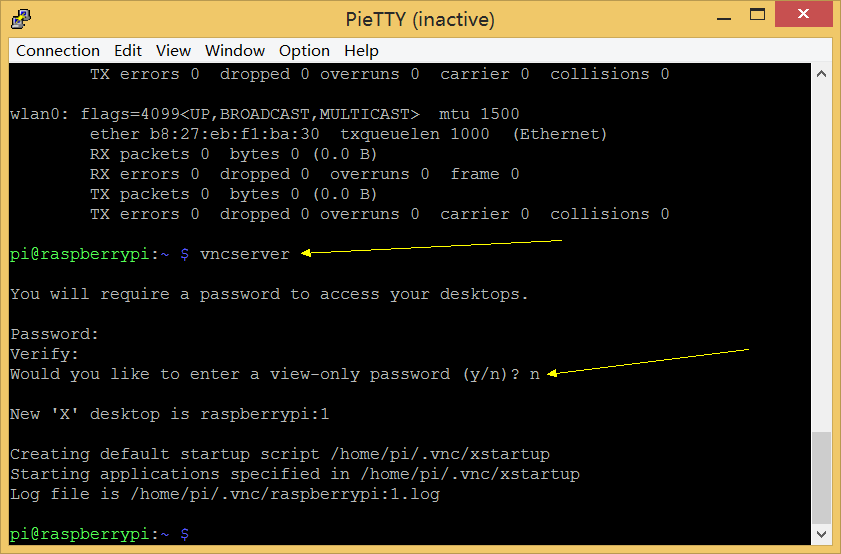
## 4）通过VNC获得图形界面显示

但是SSH连接树莓派的方式其界面还不够友好，没有图形界面，不能用鼠标点选操作。下一步可以在树莓派上面安装VNCServer，在笔记本电脑上安装VNCViewer，然后就可以用图形界面的方式和树莓派进行交互操作了。

在SSH终端中直接使用命令来安装VNCServer：

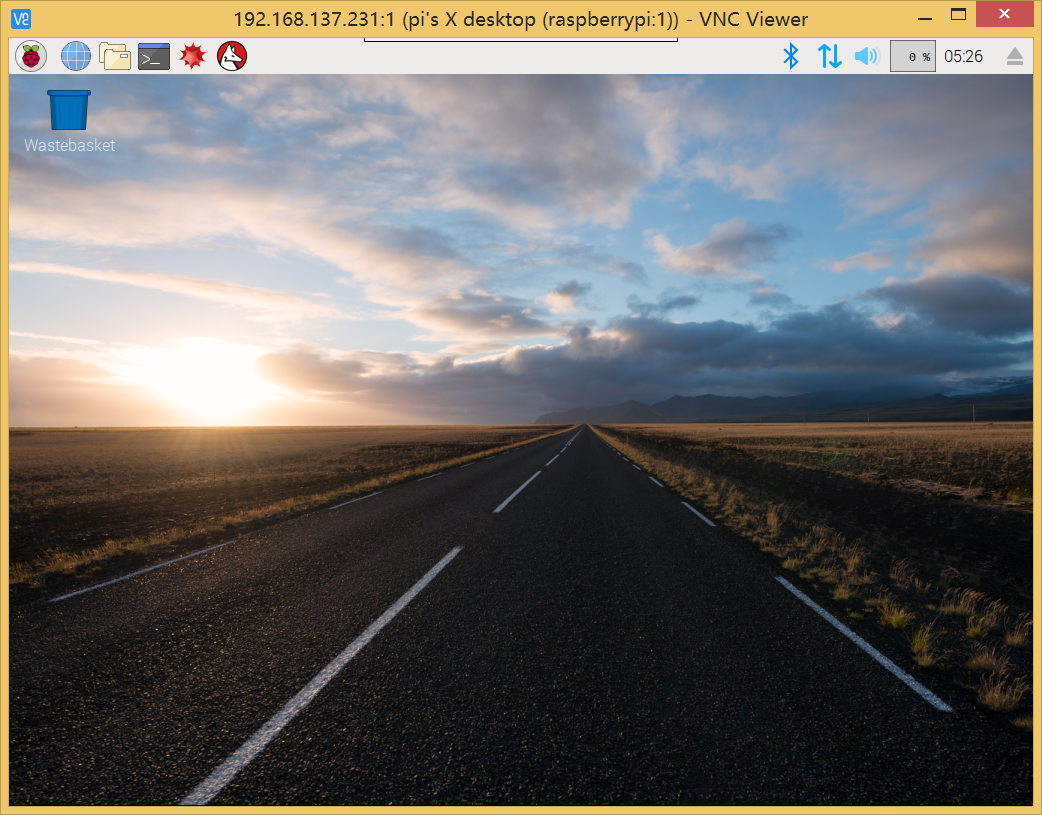
sudo apt-get install tightvncserver

成功安装以后，就可以在SSH终端中输入“vncserver”来启动远程界面服务器了，系统会提示你输入你的vncserver的登录密码，以及是否设置“只看”方式的密码，设定好以后你会看见类似下面的内容。这样你的vncserver就成功启动了。



在笔记本电脑上安装VNC Viewer软件工具，下载地址是：<https://www.realvnc.com/en/connect/download/viewer/>

最后打开你电脑上的vncviewer，在地址上输入 192.168.137.xxx:1（注意，此处IP地址后面有英文的冒号和1）,确认后输入刚刚设置的vncserver密码就可以登录到你的树莓派图形界面了。



将vncserver设置为树莓派开机自动启动，待补充。

# 三、视频处理

## 1）测试摄像头

提示：树莓派摄像头有专门拍照、记录函数，

1. **摄像头驱动安装**

首先使用 ls指令来查看是否加载到了对应的video device设备：

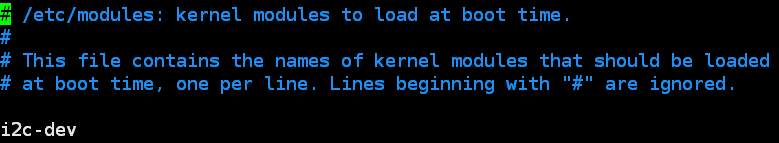
ls -al /dev/



所以没有发现我们的设备，接下来要做的是添加摄像头的驱动程序.ko文件和对应的raspiberry B3+的硬件使能问题：

1、添加驱动程序文件进来：

sudo vim /etc/modules



在最后添加如下的代码：

bcm2835-v4l2

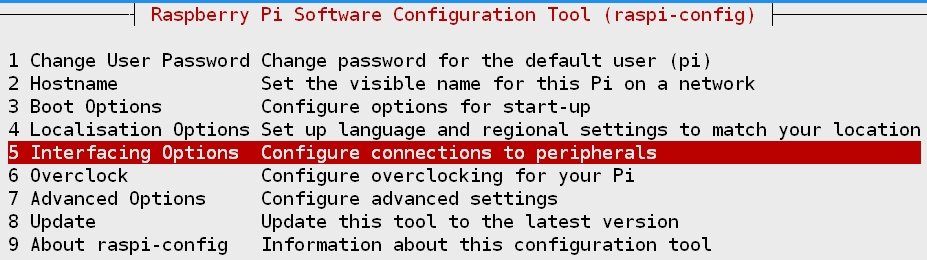
这样就完成了在启动过程中加载camera驱动的前提，注意一个问题就是/etc/modules文件的修改权限是super admin所以，记得使用sudo vim /etc/modules.

2、修改Raspberry的启动配置使能项：

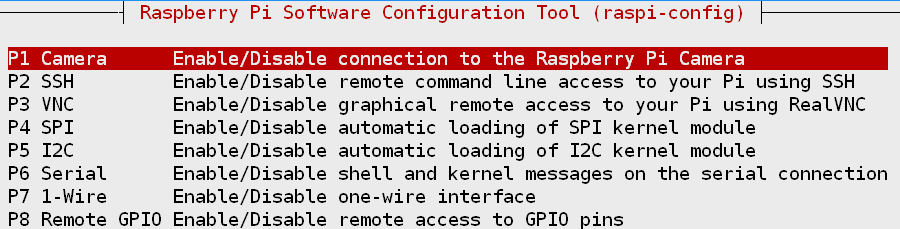
sudo raspi-config

http://images2017.cnblogs.com/blog/772331/201709/772331-20170924160535118-1858246368.png

得到如下的配置界面：



选择Interfacing Option，选中Select然后Enter进入，如下图所示：

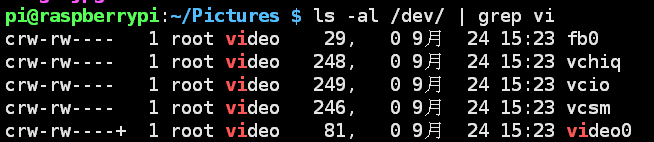


接下来机会问你是否同意使能Pi camera，选择是然后会让你重启，，重启就好了：

http://images2017.cnblogs.com/blog/772331/201709/772331-20170924160938243-536503361.png  选择 “是”

3、重启完之后，我们的基本的操作就完成了，下来来看看/dev下面是否存在摄像头设备的问题：

ls -al /dev/ | grep video



这次没问题了，找到了我们想要的看到的设备：video0设备。

**注：可能提示这样的问题（如果在以上工作都完成的情况下，摄像头还是不能正常的使用或者驱动，请先检查硬件的连接的问题，可能是排线没有很好的插稳，或者是摄像头本身的问题。）：**

mmal mmal\_vc\_component\_create failed to create component 'vc ril camera' (1:ENOMEM)

mmal mmal\_component\_create\_core could not create component 'vc ril camera' (1) mmal Failed to create camera component

mmal main Failed to create camera component

mmal Camera is not detected. Please check carefully the camera module is installed correctly

上面的问题参考这里：

<http://blog.sina.com.cn/s/blog_e4008c660101hkd9.html>

1. **摄像头功能验证**

打开终端，执行raspistill -o image.jpg 命令，验证摄像头是否完好，如果可以在VNC Viewer中看到摄像头所拍摄的图像，就表示一切正常。

raspistill命令的常用参数含义如下：

-v：调试信息查看

-w：图像宽度

-h：图像高度

-rot：图像旋转角度，只支持 0、90、180、270 度（这里说明一下，测试发现其他角度的输入都会被转换到这四个角度之上）

-o：图像输出地址，例如image.jpg，如果文件名为“-”，将输出发送至标准输出设备

-t：获取图像前等待时间，默认为5000，即5秒

-tl：多久执行一次图像抓取

执行下面的指令：

raspistill -o image%d.jpg -rot 180 -w 1024 -h 768 -t 20000 -tl 5000 –v

树莓派就会输出一系列的捕获的信息，

如图片的大小http://images2017.cnblogs.com/blog/772331/201709/772331-20170924163218540-723191324.png，

图片的总捕获时间http://images2017.cnblogs.com/blog/772331/201709/772331-20170924163248009-921464064.png，

图像是否旋转，http://images2017.cnblogs.com/blog/772331/201709/772331-20170924163533868-1060456631.png

是否水平反转垂直反转：http://images2017.cnblogs.com/blog/772331/201709/772331-20170924163600306-144602381.png

下面是最后的捕获的结果：

http://images2017.cnblogs.com/blog/772331/201709/772331-20170924163455353-92703850.png

注：这里解释一下图片的采集数量和-t参数以及-tl参数的相关关系：N\_Picture = -t(获取图像前等待时间)/-tl(多久执行一次抓取时间) + 1。所以有20000/5000+1 = 5张图片

1. **为python安装picamera模块**

执行命令：

sudo apt-get install python-pip

sudo apt-get install python-dev

sudo pip install picamera

## 2）开机运行脚本

1. **Python 代码实现连续拍照代码**

首先用命令 mkdir new 在 /home/pi/目录下建立一个名为new的目录，下面的python代码执行将每隔2秒拍摄一张照片并存放在new目录下。

|  |
| --- |
| # -\*- coding: utf-8 -\*-  import time  from picamera import PiCamera  camera = PiCamera()  camera.resolution = (1024, 768)  camera.start\_preview()  while True:  start\_time = time.time()  camera.capture('/home/pi/new/%s.jpg'%start\_time)  time.sleep(2) |

1. **Python 回传代码**

|  |
| --- |
| import os  import re  import time  path = '/home/pi/new'  l=[]  textfile = os.listdir(path)  for i in textfile:  num = i.split('.jpg')  l.append(num[0])  now = time.time()  for i in l:  name = i+".jpg"  print name  temp = 'scp ./new/'+name+' alex@192.168.1.114: '  if (now-float(i))>150:  pass  print temp  os.system(temp) |

1. **树莓派开机自动拍照**

编辑脚本，使树莓派派开机后自动拍照。运行 chmod +x /home/pi/camera\_alltime.py（提升权限）

保存脚本文件为/etc/init.d/camera\_alltime文件

|  |
| --- |
| #!/bin/sh  #/etc/init.d/ip\_acquire  ### BEGIN INIT INFO  # Provides: start\_tool  # Required-Start: $remote\_fs $syslog  # Required-Stop: $remote\_fs $syslog  # Default-Start: 2 3 4 5  # Default-Stop: 0 1 6  # Short-Description: start\_tool  # Description: This service is used to start my applaction  ### END INIT INFO  case "$1" in  start)  echo "Starting app"  python3 /home/pi/share/ip\_acquire.py  ;;  stop)  echo "Stop"  #kill $( ps aux | grep -m 1 'python3 /home/pi/share/start.py' | awk '{ print $2 }')  ;;  \*)  echo "Usage: service start\_tool start|stop"  exit 1  ;;  esac  exit 0 |

设置python脚本开机启动：

|  |
| --- |
| sudo chmod +777 /etc/init.d/camera\_alltime |

启动改脚本用service 命令就可以:

|  |
| --- |
| sudo service camera\_alltime start#开始  sudo service camera\_alltime stop#停止 |

最后设置开机启动：

|  |
| --- |
| sudo update-rc.d ip\_acquire defaults |

## 3）图片传输

为了简单直接使用了SCP进行照片传输，所以为linux机器A 安装SSH，并设置无密码登陆，用于传输图片；

1. **安装SSH**

sudo apt-get install openssh-server

尝试用ssh [root]@[IP地址]验证

1. **设置A与树莓派的无密码登陆:**

进入.ssh目录下面，机器A与树莓派上执行：

ssh-keygen -t  rsa

之后一路回车。

在机器A的目录.ssh下执行命令，cat  id-rsa.pub >> authorized\_keys；此后.ssh下面会出现authorized\_keys文件。

然后将主机A的authorized\_keys文件拷贝到树莓派的.ssh目录下，如：scp authorized\_keys xuhui@cloud002:~/.ssh/

在树梅派的.ssh目录下，会发现刚刚传输过来的文件-authorized\_keys，然后执行命令，将第二台计算机的公钥也加进来，如：cat id-rsa.pub >> authorized\_keys.

将树莓派新生成的authorized\_keys传输回机器A。覆盖掉之前的authorized\_keys。

完成后，既可以免密码ssh登陆.

## 4）自动运行

树莓派加电后一段时间后等待树莓派启动完成，机器A，SSH登陆树莓派，输入python test.py

距离此时两分钟之内的照片自动传送过来。

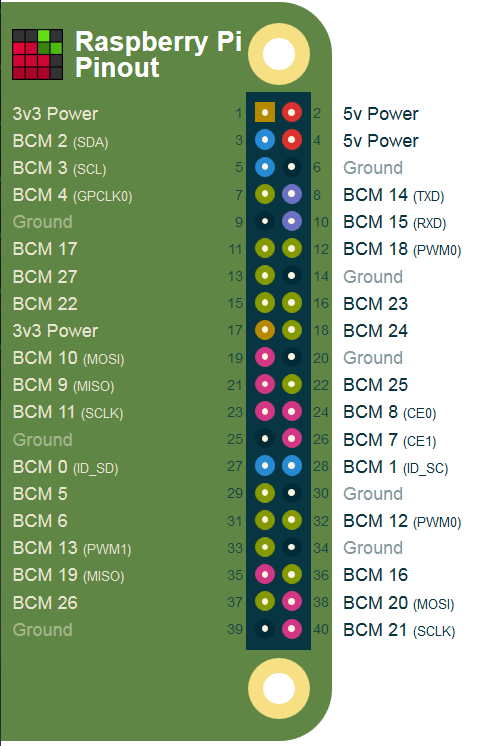
四、GPIO口的使用

**1) GPIO口简介**

General Purpose Input Output （通用输入/输出）简称为GPIO，或总线扩展器，是树莓派和外界设备进行交互的重要端口，我们在下面会介绍如何利用GPIO口及树莓派的RPi.GPIO库控制LED灯，驱动舵机，及获取光敏电阻的输入。

1. GPIO细节介绍

树莓派的GPIO口的各个引脚的功能可以查看<pinout.xyz>这个网站，网站中给出这样的示意图：



红色框框给出的是各个物理引脚的编号（BOARD模式），蓝色框框给出的是BCM的编号。在利用python进行编程的时候，需要制定引脚，要区分是使用BOARD模式，还是BCM模式。除此以外，蓝色框框中还给出了每个引脚的功能，在使用的时候需要选择合适的引脚。

1. 安装RPi.GPIO

为了在python中使用树莓派的GPIO口，需要安装RPi.GPIO库。

sudo apt-get update

sudo apt-get -y install python-rpi.gpio

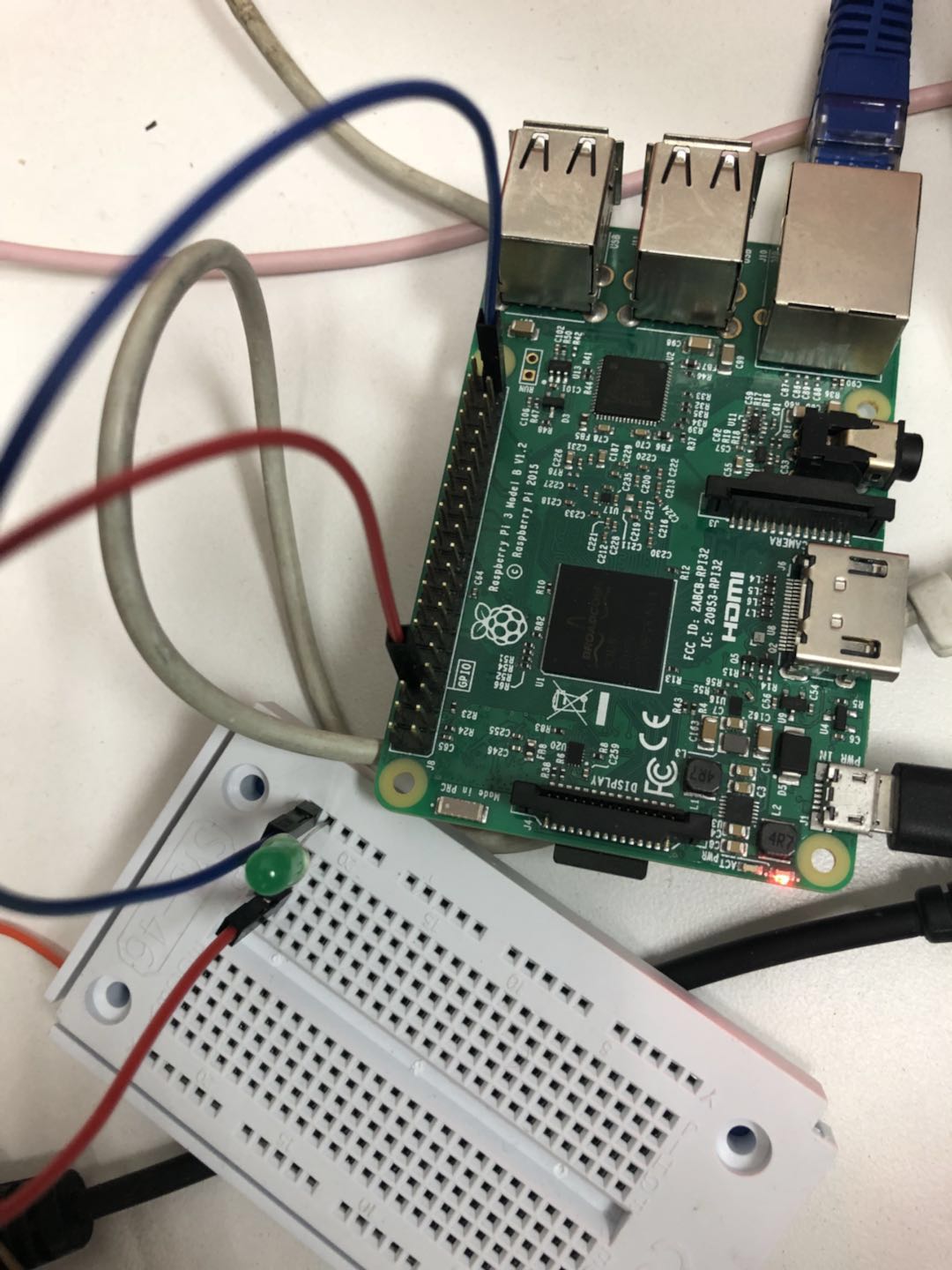
以上两行指令是为了安装这个库，下面这条指令是在python代码中引入这个库。

import RPi.GPIO as GPIO

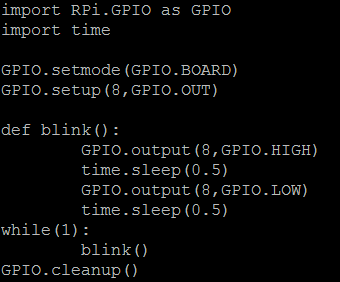
**2）LED灯的控制**

1. 接线

LED灯有两个引脚，短的引脚接地，长的引脚接信号线。在这里39号口是地线，8号口作为信号线，接线如下图：



2.代码



引入GPIO库

引入time库

设置引脚为BOARD模式

把8号引脚设为输出口

函数定义：

8号口输出高电平

延时0.5秒

8号口输出低电平

延时0.5秒

反复循环这个过程

将这个GPIO口的声明清除

以上代码给出了一个闪烁LED灯的示例。

**3）舵机驱动**

1. PWM波

PWM就是脉冲宽度调制，也就是占空比可变的脉冲波形。 舵机的控制信号是 PWM 信号，利用占空比的变化，改变舵机的位置。

舵机的控制一般需要一个20ms左右的时基脉冲，该脉冲的高电平部分一般为0.5ms~2.5ms范围内的角度控制脉冲部分。以180度角度伺服为例，那么对应的控制关系是这样的：

0.5ms--------------0度；

1.0ms------------45度；

1.5ms------------90度；

2.0ms-----------135度；

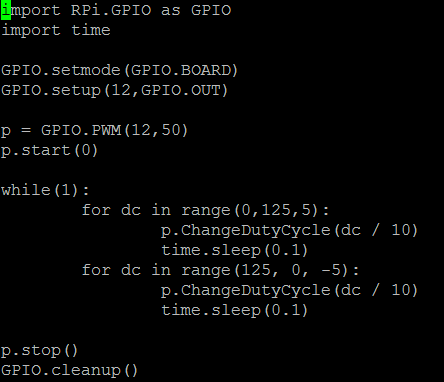
2.5ms-----------180度；

也就是大致表明在占空比为0 – 12.5内是使舵机转动0 – 180度的范围。

2.接线

舵机有三根线，棕色的是接地线，需要接树莓派的39号地线，红色的是电源线，需要接2号5V的电源线，橙色的是PWM波信号线，需要接12号PWM输出口。

3．代码



引入相应库

申明引用的12号管脚，它是输出口

定义这个口PWM波的频率为50HZ

循环过程：

占空比从0变到12.5

（舵机转动）

占空比从125变到0

（舵机逆向转动）

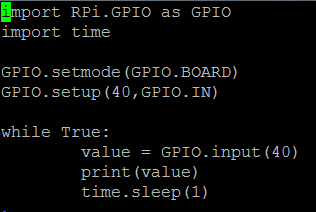
结束

**4）光敏电阻使用**

1. 接线

树莓派上三个接口，VCC接2号5V电压，GND接39号口，OUT接40号口作为树莓派的输入

2.代码



引入相应库

设置40号引脚为输入

不断获取光敏电阻数据

将获取的值输出出来

由于树莓派GPIO口缺乏AD采样功能，所以输出的值为布尔值。

五、认知服务

**1）微软认知服务简介**

**2）HTTP请求简介**

**3）人脸识别**

**4）语音识别**

**5）语音合成**