

LED 闪烁实验教程

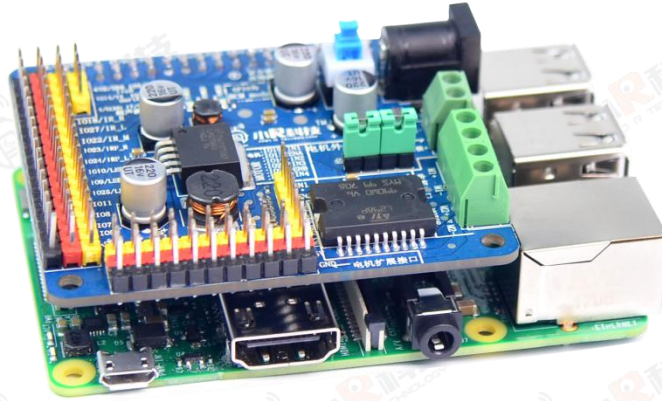
目录

一、 实验概述.....	2
二、 实验器材.....	2
三、 知识要点.....	2
四、 实验原理.....	2
五、 代码编写.....	4
六、 硬件连接及运行效果.....	6
七、 思维发散及课后作业.....	7
官 网: www.xiao-r.com	9
论 坛: www.wifi-robots.com	9
官方商城: wifi-robots.taobao.com	9
微信公众号:	9

一、实验概述

使用 PWR.A53 树莓派驱动板、树莓派主板，在树莓派上用 Python 语言编写一个小程序，让驱动板上的 LED 灯持续亮灭动作，通过这个实验，可以学习掌握最基本的 I/O 口控制、延时函数、Python 控制树莓派 IO 电平工作原理等等。

二、实验器材



1、PWR 电源板 和树莓派主板（树莓派主板在下层）

三、知识要点

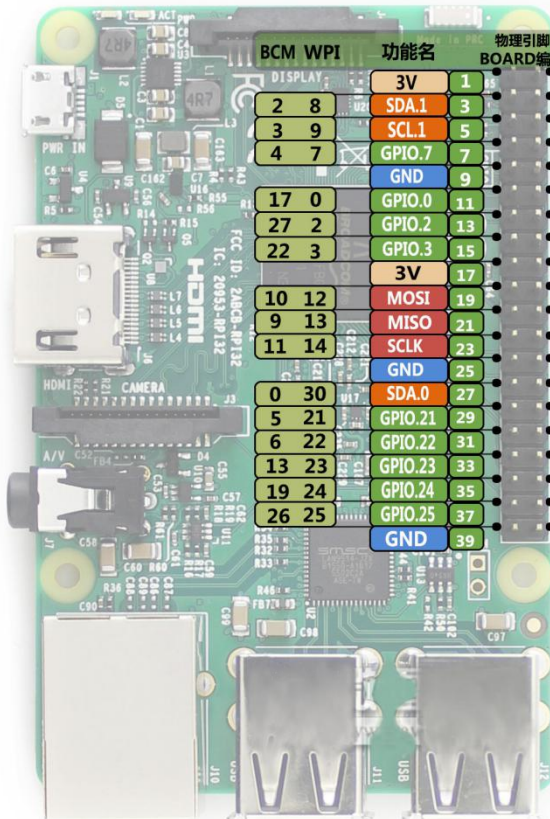
- 1、`GPIO.setmode(GPIO.BCM)`##信号引脚模式定义，使用.BCM 模式
- 2、`GPIO.setup(LED0,GPIO.OUT,initial=GPIO.HIGH)`##把指定管脚 LED0 配置为输出模式，并初始化为高电平
- 3、`GPIO.output(LED2,False)`## 让指定管脚 LED2 输出低电平

四、实验原理

LED 灯是一种基本的发光器件，给他输入 5V 电压时，LED 灯发光，输入 0V 电压时熄灭，在本节实验中，我们利用的是 PWR 电源板上的 LED0、LED1、LED2 三个灯，这三个灯在硬件电路上分别连接到了树莓派主板上的 10、9、25 三个 IO 管脚（按 BCM 模式定义），如果在代码中，我们让这三个 IO 管脚的电平输出高低变化，就可以看到 LED 灯的亮灭效果。

在树莓派中，我们可用 GPIO 库提供的接口来让指定的 IO 电平变高会变低。例如，我们想让 LED0 所对应的 10 号 IO 口电平变低，那么主要用 3 个步骤实现：

- 1、定义 LED0 变量的 IO 口，代码为：`LED0 = 10`
- 2、将当前的 GPIO 配置为输出模式，并初始化为高电平，代码为：
`GPIO.setup(LED0,GPIO.OUT,initial=GPIO.HIGH)`
- 3、使用 `GPIO.output()`函数，把 LED0 这个管脚低电平拉低，代码为：
`GPIO.output(LED2,False)`



BCM	WPI	功能名	物理引脚	BOARD 编号	功能名	WPI	BCM
		3V	1		2		5V
2	8	SDA.1	3		4		5V
3	9	SCL.1	5		6		GND
4	7	GPIO.7	7		8	15	14
		GND	9		10	16	15
17	0	GPIO.0	11		12	1	18
27	2	GPIO.2	13		14		GND
22	3	GPIO.3	15		16	4	23
		3V	17		18	5	24
10	12	MOSI	19		20		GND
9	13	MISO	21		22	6	25
11	14	SCLK	23		24	10	8
		GND	25		26	11	7
0	30	SDA.0	27		28	31	1
5	21	GPIO.21	29		30		GND
6	22	GPIO.22	31		32	26	12
13	23	GPIO.23	33		34		GND
19	24	GPIO.24	35		36	27	16
26	25	GPIO.25	37		38	28	20
		GND	39		40	29	21

树莓派管脚分布图

五、代码编写

#coding:utf-8

#Python 中声明文件编码的注释，编码格式指定为 utf-8

import time #导入 time 库，可使用时间函数。

import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BCM) ##信号引脚模式定义，使用.BCM 模式

LED0 = 10 ##LED0 的 IO 口定义

LED1 = 9 ##LED1 的 IO 口定义

LED2 = 25 ##LED2 的 IO 口定义

GPIO.setwarnings(False)

GPIO.setup(LED0,GPIO.OUT,initial=GPIO.HIGH)##led0 初始化为高电平

GPIO.setup(LED1,GPIO.OUT,initial=GPIO.HIGH)##led1 初始化为高电平

GPIO.setup(LED2,GPIO.OUT,initial=GPIO.HIGH)##led2 初始化为高电平

def init_light(): ##使用 def 定义函数，可在其他地方调用此函数。未调用不执行。

 GPIO.output(LED0,False)

 GPIO.output(LED1,False)

 GPIO.output(LED2,False)###LED0,LED1,LED2 = 亮 亮 亮

 time.sleep(0.5)

 GPIO.output(LED0,True)

 GPIO.output(LED1,False)

 GPIO.output(LED2,False)###LED0,LED1,LED2 = 灭 亮 亮

 time.sleep(0.5)

 GPIO.output(LED0,False)

 GPIO.output(LED1,True)

 GPIO.output(LED2,False)###LED0,LED1,LED2 = 亮 灭 亮

 time.sleep(0.5)

 GPIO.output(LED0,False)

 GPIO.output(LED1,False)

 GPIO.output(LED2,True)###LED0,LED1,LED2 = 亮 亮 灭

 time.sleep(0.5)

 GPIO.output(LED0,False)

 GPIO.output(LED1,False)

 GPIO.output(LED2,False)###LED0,LED1,LED2 = 亮 亮 亮

 time.sleep(0.5)

 GPIO.output(LED0,True)

 GPIO.output(LED1,True)

 GPIO.output(LED2,True)###LED0,LED1,LED2 = 灭 灭 灭

 time.sleep(0.5)

for i in range(1,5): #调用 rang () 循环函数，功能类似 for (i =1;i<5;i++) 执行 4 遍

 init_light()

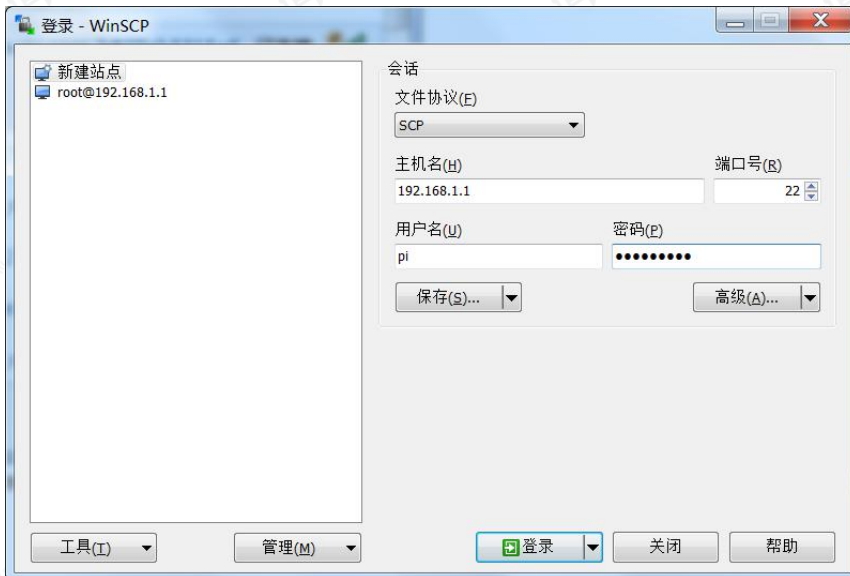
六、硬件连接及运行效果

1、硬件连接如下图所示，因为我们使用的是树莓派主板上自身的 LED 灯，所以只需要接入电池到 PWR 驱动板的电源输入接口即可。

2、打开 PWR 驱动板的开关，系统开始启动，等待 30 秒后，系统启动完毕，使用电脑的无线网卡搜索 wifi-robots.com 开头的无线信号，并连接。



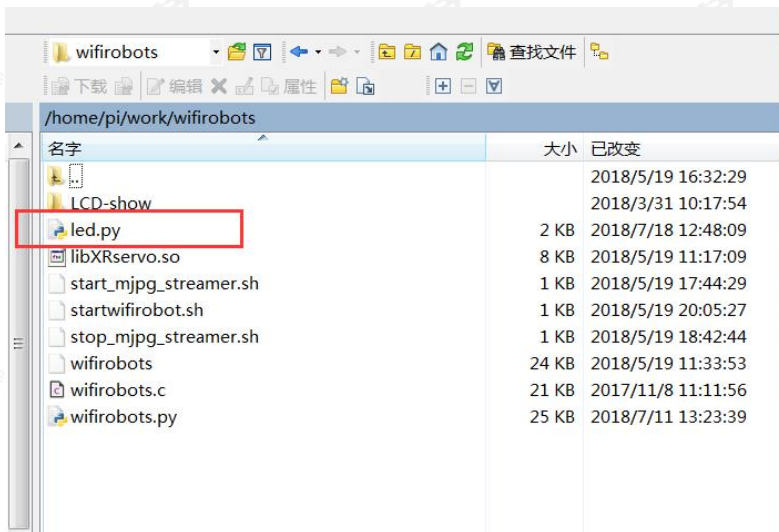
4、打开 WINSCP 软件，填写树莓派小车的 IP 等参数，其中：用户名为 pi 密码为 raspberry 点击“登录”。



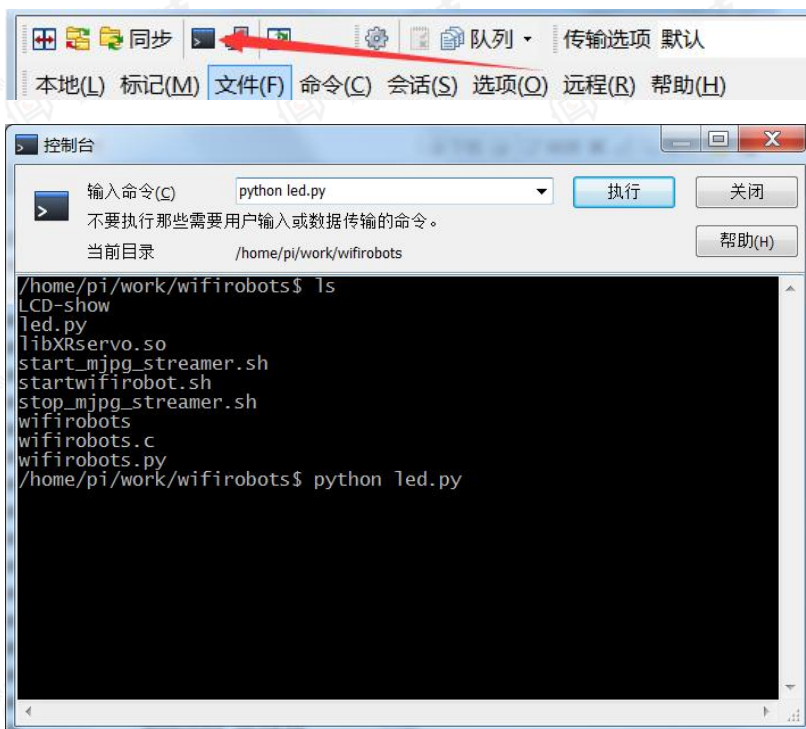
如果出现如下提示，选择更新或者添加



5、把 led.py 文件拖放到右侧的/home/pi/work/wifirobots 目录



6、点击 WINSCP 工具栏里面的“命令”按钮，在命令框中输入命令 `python led.py` 并点击“执行”



运行效果：

PWR 驱动板上的 3 个 LED 灯交替亮起、闪烁，持续 4 次，然后熄灭。

七、思维发散及课后作业

LED 灯的亮灭是最基本的 IO 操作，通过本节课程，我们掌握了如何用 python 代码来控制硬件 IO 的电平，我们也可以以此类推，去控制继电器或者其他电平控制的外部设备，同学们可以动手试一下。

官 网: www.xiao-r.com

论 坛: www.wifi-robots.com

官方商城: wifi-robots.taobao.com

微信公众号:

