

学习知识准备与双色 LED 实验

一、Raspberry Pi 的 IO 口介绍

Raspberry Pi 的 IO ，一共有 40 个管脚，具体定义如下所示：

wiringPi Pin	BCM GPIO	Name	Header	Name	BCM GPIO	wiringPi Pin
—	—	3.3v	1 2	5v	—	—
8	R1:0/R2:2	SDA0	3 4	5v	—	—
9	R1:1/R2:3	SCL0	5 6	0V	—	—
7	4	GPIO7	7 8	TXD	14	15
—	—	0V	9 10	RXD	15	16
0	17	GPIO0	11 12	GPIO1	18	1
2	R1:21/R2:27	GPIO2	13 14	0V	—	—
3	22	GPIO3	15 16	GPIO4	23	4
—	—	3.3v	17 18	GPIO5	24	5
12	10	MOSI	19 20	0V	—	—
13	9	MISO	21 22	GPIO6	25	6
14	11	SCLK	23 24	CE0	8	10
—	—	0V	25 26	CE1	7	11
30	0	SDA. 0	27 28	SCL. 0	1	31
21	5	GPIO. 21	29 30	0V	—	—
22	6	GPIO. 22	31 32	GPIO. 26	12	26
23	13	GPIO. 23	33 34	0V	—	—
24	19	GPIO. 24	35 36	GPIO. 27	16	27
25	26	GPIO. 25	37 38	GPIO. 28	20	28
—	—	0V	39 40	GPIO. 29	21	29
wiringPi Pin	BCM GPIO	Name	Header	Name	BCM GPIO	wiringPi Pin

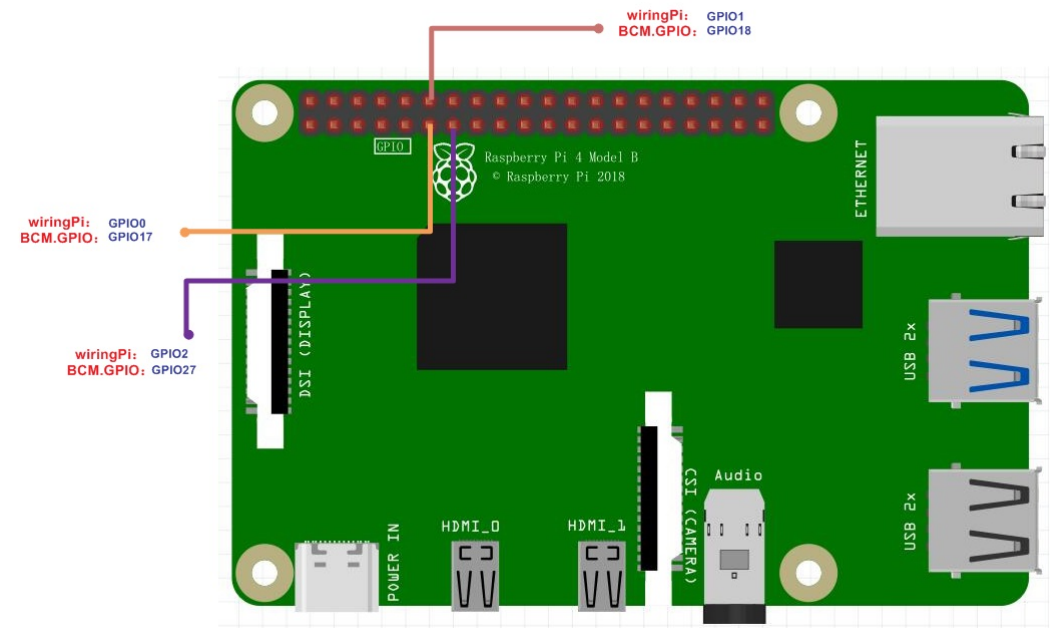
For RPi B

For RPi B+ / 2 model B

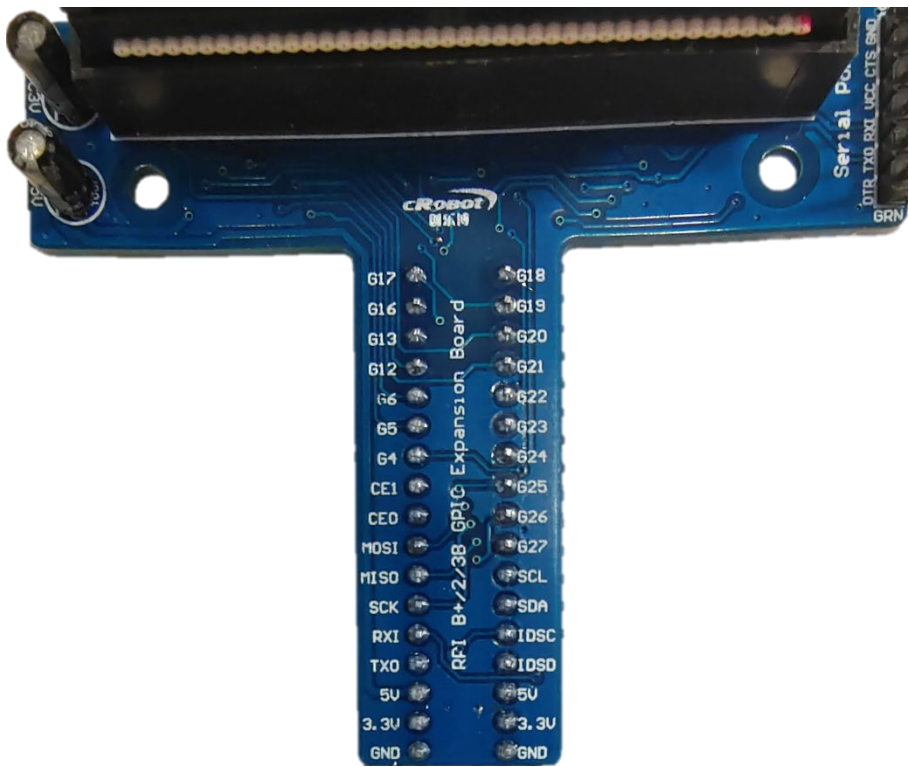
目前，Raspberry Pi 有三种引脚编号方法，分别是：根据引脚的物理位置编号(Header)、由 C 语言 GPIO 库 wiringPi 指定的编号(wiringPi Pin)、由 BCM2837 SOC 指定的编号 (BCM GPIO)。

如果想要基于 wiringPi 库以 C 语言运行 Raspberry Pi GPIO, 请选择由 wiringPi 指定的编号。从上图可以看出，wiringPi 中的 GPIO0 对应于物理位置编号的引脚 11，GPIO30 对应物理位置编号的引脚 30。

下图说明了三个物理位置编号的引脚 11、12 和 13 对应的编号。例如，物理位置编号的引脚 11 对应 wiringPi 的 0 和 BCM GPIO 的 17。



我们使用的 T 型扩展板采用的是 BCM 编码。需要注意的 G27 对应的是 BCM 编码的 R1:21/R2:27 管脚。



二、wiringPi 库介绍

如果选择使用 C/C++ 编程，需要安装 wiringPi 库，安装方法如下：

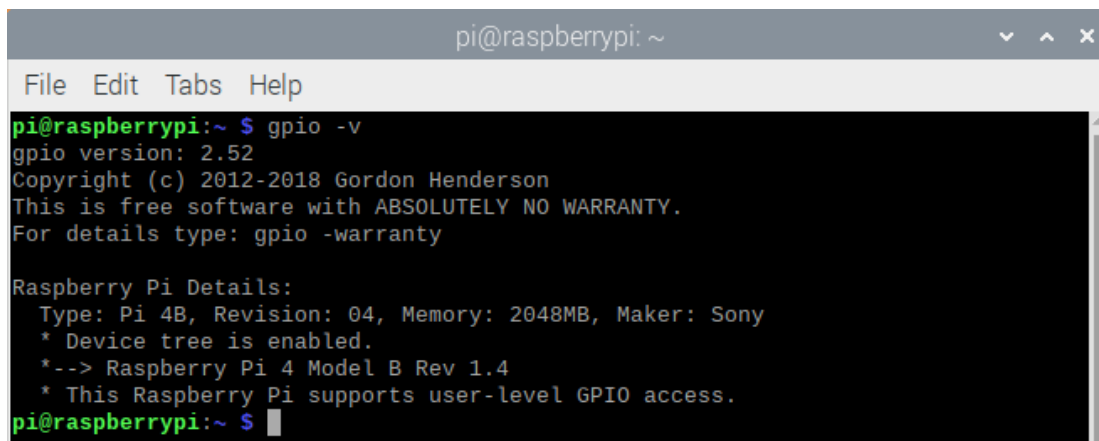
```
cd /tmp
```

```
wget https://project-downloads.drogon.net/wiringpi-latest.deb
```

```
sudo dpkg -i wiringpi-latest.deb
```

安装完毕后，可以使用如下命令测试是否成功安装。

```
gpio -v
```



```
pi@raspberrypi: ~  
File Edit Tabs Help  
pi@raspberrypi:~ $ gpio -v  
gpio version: 2.52  
Copyright (c) 2012-2018 Gordon Henderson  
This is free software with ABSOLUTELY NO WARRANTY.  
For details type: gpio -warranty  
  
Raspberry Pi Details:  
Type: Pi 4B, Revision: 04, Memory: 2048MB, Maker: Sony  
* Device tree is enabled.  
* -> Raspberry Pi 4 Model B Rev 1.4  
* This Raspberry Pi supports user-level GPIO access.  
pi@raspberrypi:~ $
```

三、RPi.GPIO 介绍

如果使用 Python 编程，可以使用 RPi.GPIO 提供的 API 对 GPIO 进行编程，该软件包提供了一个类来控制 Raspberry Pi 上的 GPIO。Raspberry Pi 的 Raspbian OS 镜像中默认安装 RPi.GPIO，因此可以直接使用它。

如果需要安装 python-dev 包，则输入以下命令：

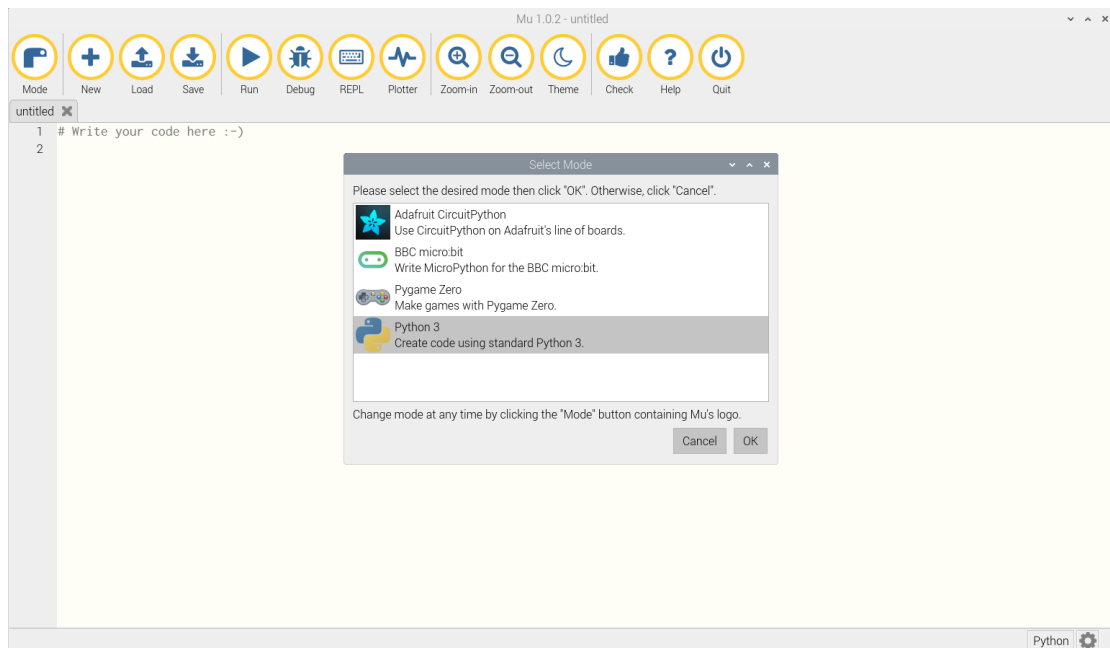
```
sudo apt-get install python-dev
```

四、使用 MU 软件进行 python 开发

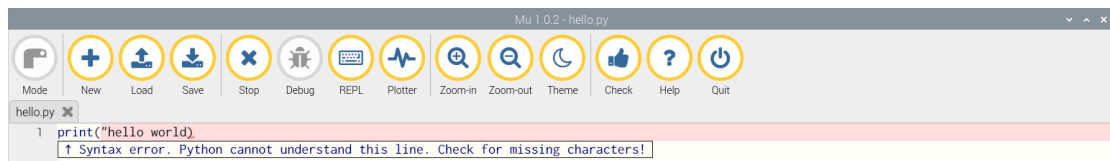
Mu 对初学者来说是一个简单易用的 Python 编辑器和 IDE(集成开发环境)。



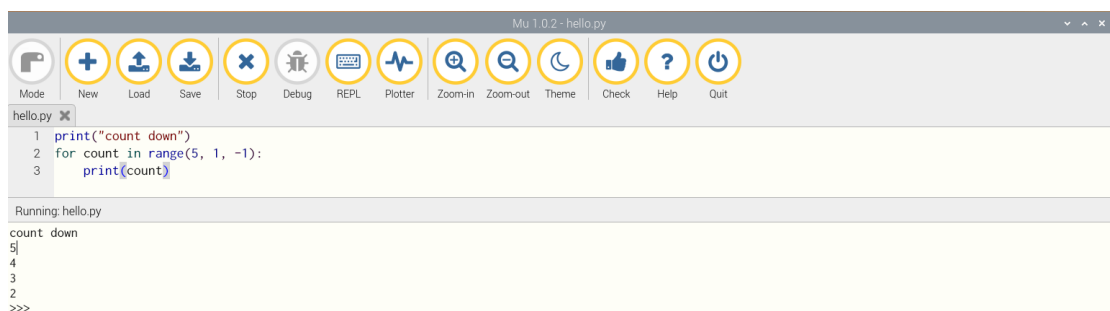
打开 Mu 软件后，选择 Python 3 模式，点击 OK。



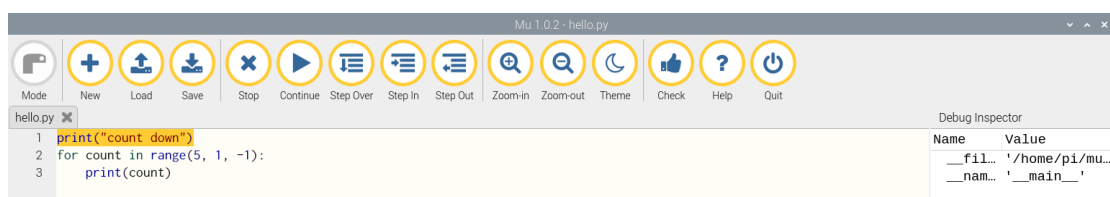
Mu 的主要区域是编写代码的地方，在 Mu 中输入以下代码，创建“Hello World”程序。点击 Save 保存程序；点击 Check 检查程序（只要在程序未运行的情况下，都可以点击 Check 检查代码是否有错误。）；点击 Run 运行程序；点击 Stop 停止运行程序。



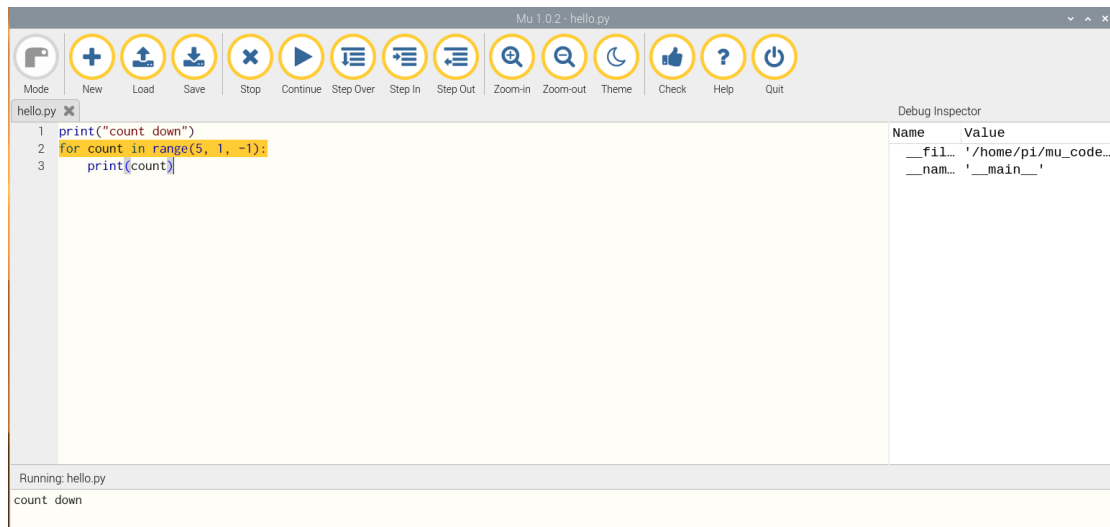
如果代码没有语法错误，但是运行结果与预期不同，则可以使用“Debug”按钮进行逐步调试，具体调试界面如下：



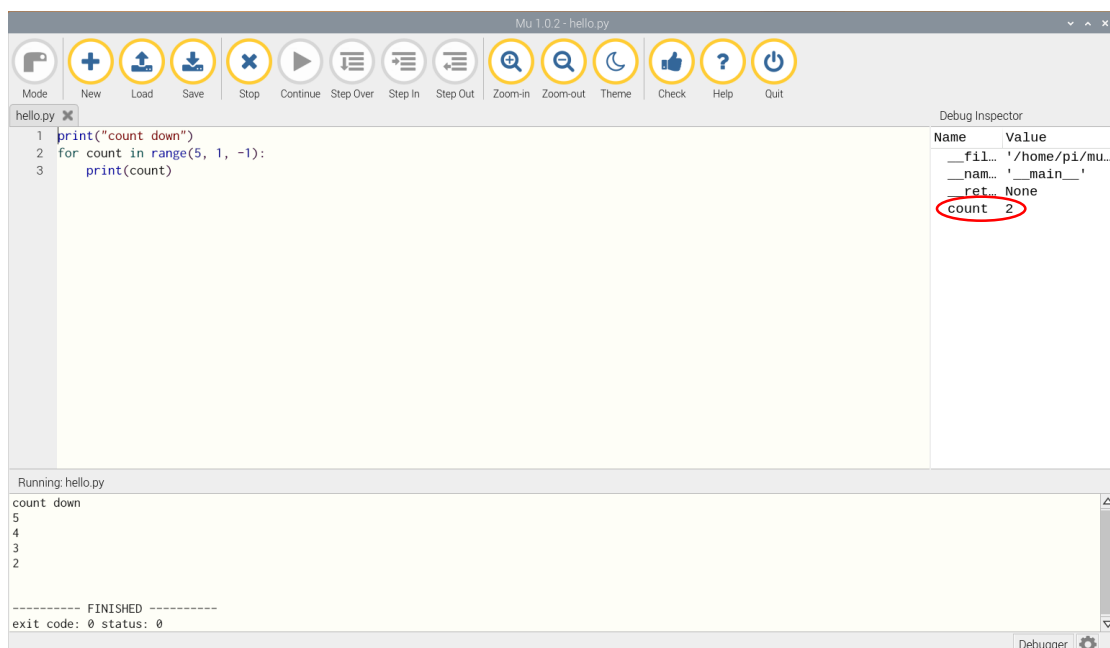
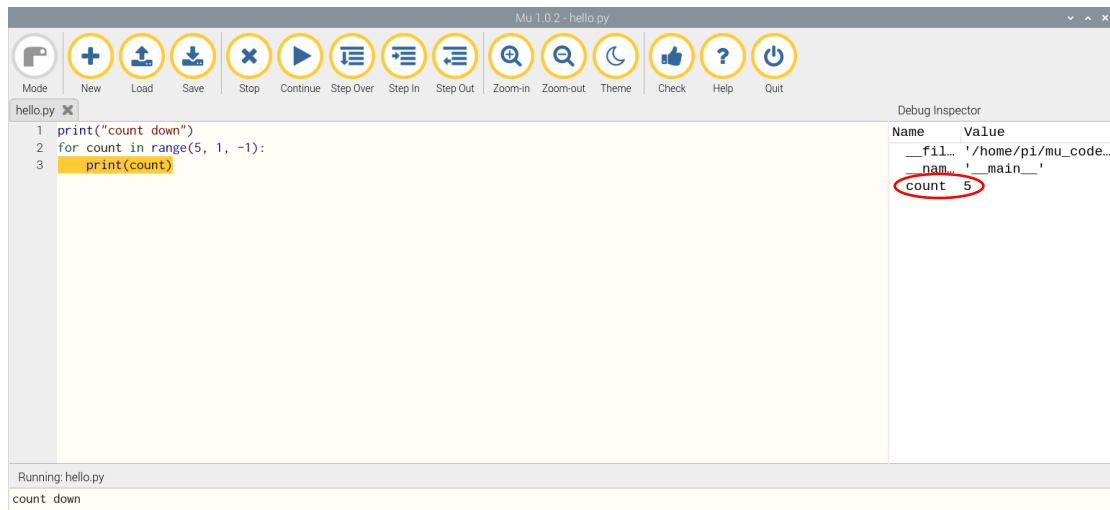
调试器有四个按钮，Continue：重新运行程序、Step Over：单步执行、Step In：如果下一行是函数，则进入该函数运行、Step Out：退出正在运行的函数。



窗口右侧 Debug Inspector，显示正在使用的任何变量的当前值。



点击 Step Over，程序逐步运行。

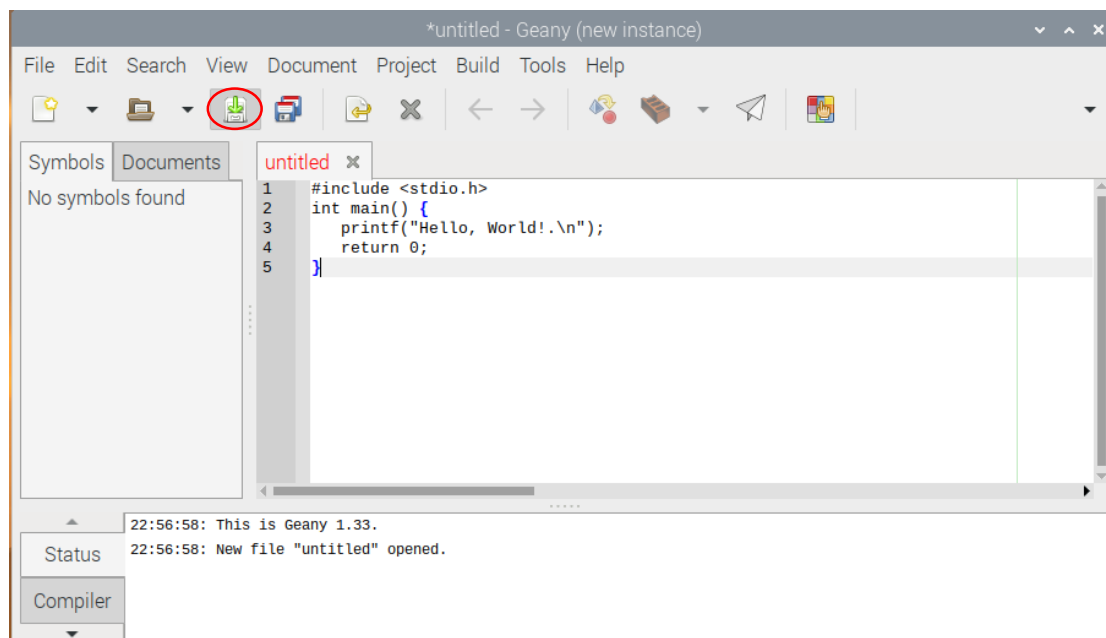


五、使用 Geany IDE 进行 C 程序开发

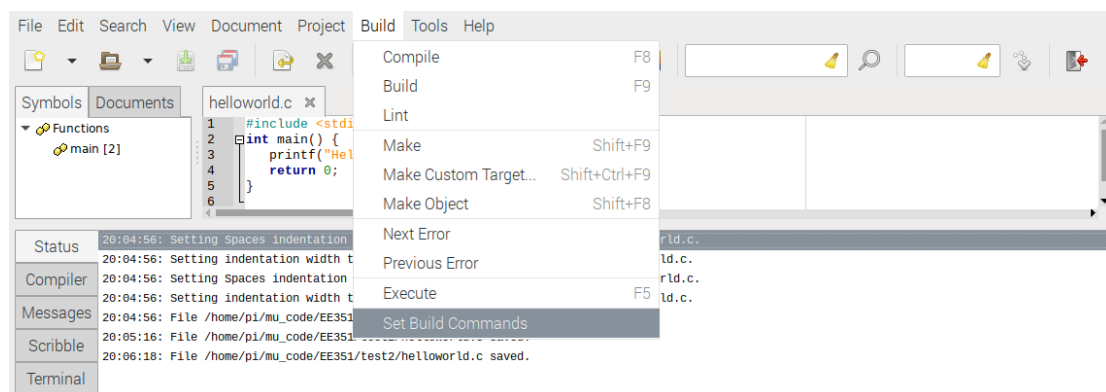
点击左上角 Programming，打开 Geany。



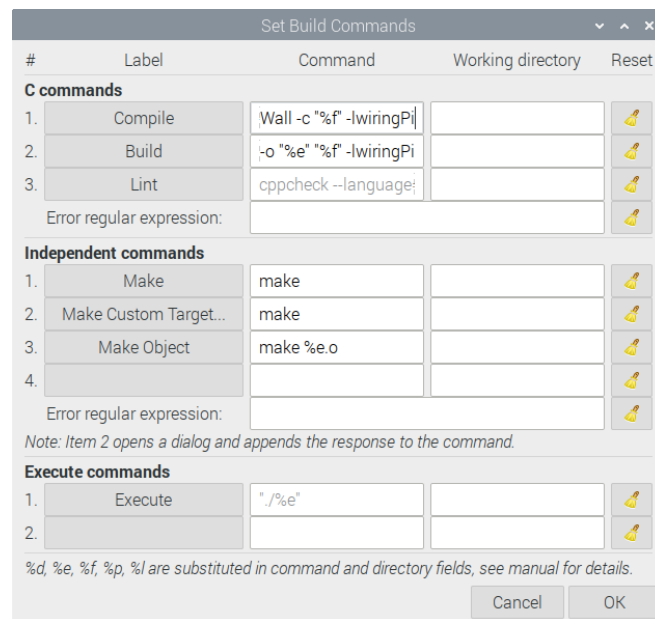
新建一个文件，点击保存，命名为 helloworld.c。（注意扩展名为.c）



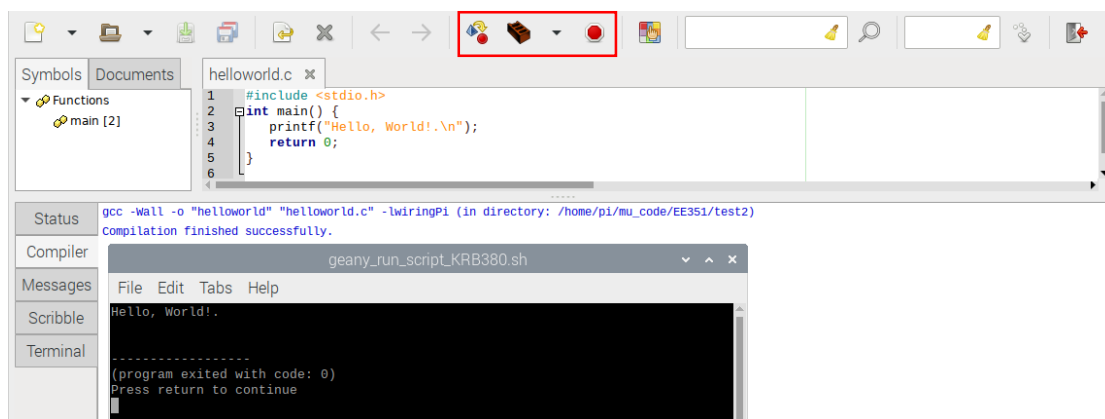
使用带有 Geany 的 wiringPi 库编译 C/C++ 程序，需要进行配置，具体方法如下。点击上侧“Build”按钮，选择 Set Build Commands。



在 Compile 和 Build 选框里最后面添加 “-lwiringPi”。



三个按钮分别为编译文件、生成运行文件、运行。



六、双色 LED 实验

双色 LED 灯准确来说叫双基色 LED 灯，是指模块只能显示两种颜色，一般是红色和绿色，可以有三种状态，灭、颜色 1 亮、颜色 2 亮。

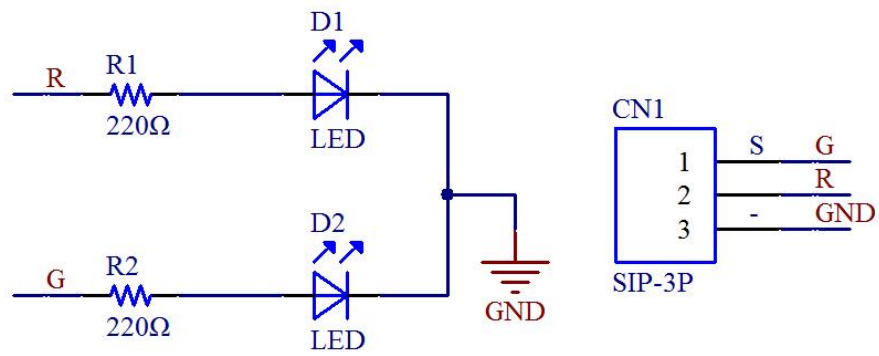


实验目标：实现 LED 灯红绿交替闪烁。

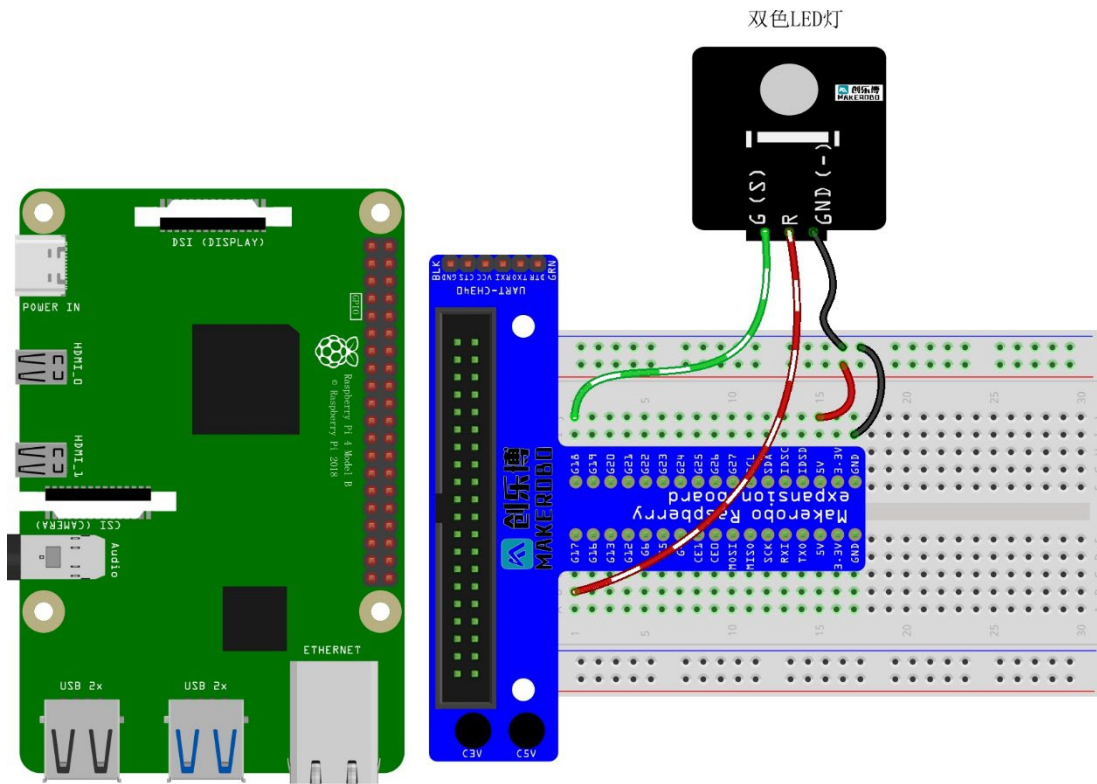
具体实现方法可以参考以下步骤：

将引脚 S（绿色）和中间管脚（红色）连接到 Raspberry Pi 的 GPIO 接口上，引脚一连接到 Raspberry Pi 的 GND 上，对 Raspberry Pi 进行编程控制，将 LED 的颜色从红色变成绿色。

该模块的原理图如下所示：



实验硬件连线图可以参考如下：



附录：使用 FileZilla 软件相互传递文件

具体配置方法如下图，可以直接使用鼠标拖拽在本机与 Raspberry Pi 之间传输文件。

