



RPi Relay Board

用户手册

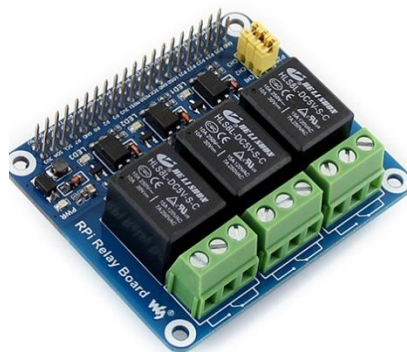
产品简介

RPi Relay Board 是适用于 Raspberry Pi A+/B+/2 代 B/3 代 B 的树莓派继电器扩展板。

主要用途:

树莓派 IO 输出的弱电流通过本产品可以控制强电流的通断,常用于智能家居等强电领域。结合树莓派的网络功能,可以实现对家电的远程控制。

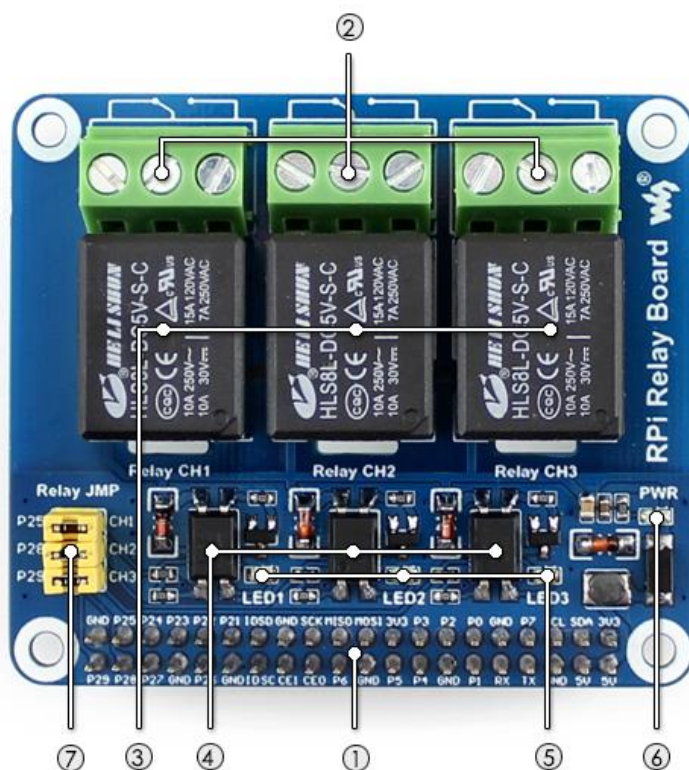
【注意】树莓派具有 40Pin 和 26Pin 两种接口,该产品只兼容 40Pin 的树莓派。



主要特性:

- 适用于树莓派 A+/B+/2B/3B
- 采用高质量的继电器
- 继电器允许接入高达 5A 250V AC 或 5A 30V DC
- 带光耦隔离,避免高电压电路干扰
- 带继电器指示灯,方便查看继电器的工作状态
- 带继电器选择跳线,方便选择树莓派其他引脚控制继电器
- 提供完善的配套资料手册(包括 wiringPi、WebioPi、shell、python 和 bcm2835 等例程)

资源简介



[接口简介]

1. **RPi 连接口**
方便接入树莓派
2. **继电器接线端子**
方便接外部被控制线

[跳线说明]

7. **继电器控制管脚选择跳线**
短接跳线：接入到示例程序指定的 I/O
断开跳线：可改为使用连接线接入自定义的 I/O

[器件简介]

3. **继电器**
4. **光耦**
PC817 光耦芯片
5. **继电器工作指示灯**
LED 亮，继电器常闭端断开，常开端闭合
LED 灭，继电器常闭端闭合，常开端断开
6. **电源指示灯**

接口说明

- **继电器通道和 Pi 引脚的对应关系：**

通道标号	RPi 引脚号	wiringPi	BCM	描述
CH1	37	P25	26	通道 1
CH2	38	P28	20	通道 2
CH3	40	P29	21	通道 3

【注意】PCB 上的丝印标示对应 wiringPi 编码。

- **Relay_JMP (6P 跳线) 用途：**

Relay_JMP 是继电器控制管脚选择跳线。当连接跳线帽时，即可使用树莓派控制继电器。

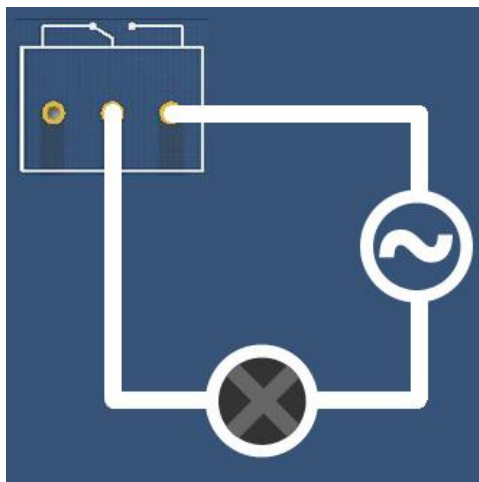
- **接线端子说明：**

本扩展板带有三个接线端子，每个接线端子上分别有三个脚，用来连接外部电路。

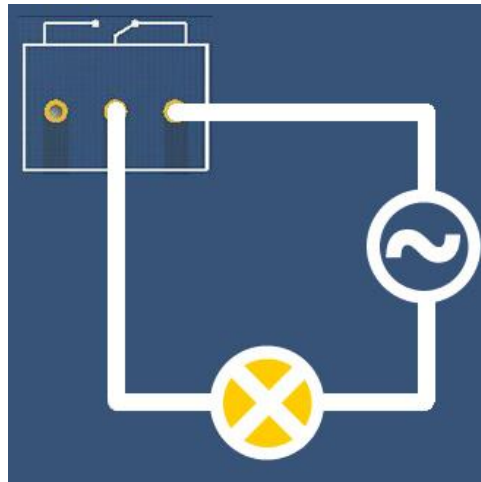
三个端子都是低电平使能。当树莓派 IO 口输出低电平时，则对应通道的 LED 灯点亮。同时继电器常开触点闭合，常闭触点断开，使得外部电路通断状态发生改变。

(PS: 不要忘记连接 Relay_JMP 跳线帽哦！)

继电器动作前：



继电器动作后：



软件例程

该模块的例程包括 Shell、bcm2835、WiringPi、python 和 Webiopi 例程。

开发环境

在使用本扩展板之前，我们需要给树莓派安装 bcm2835, wiringPi 和 python 函数库，树莓派的库函数安装和配置过程请参见[为树莓派安装必要的函数库](#)。

安装好函数库之后，还需要下载产品的示例程序。把示例程序复制并释放到系统中，例如执行 `tar -zxvf RPi_Relay_Board.tar.gz ~`，释放到/home/pi 目录下。部分示例程序拥有不同的实现方式（例如 bcm2835, wiringPi, python, Shell, Webiopi 等方式），即使实现同一个功能，也可能依赖不同的函数库。因此使用之前请务必安装好 bcm2835, wiringPi, python 和 Webiopi 函数库。

如果安装完函数库之后依然无法执行以下的示例程序，那么您可以尝试使用 `chmod +x filename` 命令给程序赋予可执行权限。

实验操作和现象

1) Shell 程序

进入 Linux 终端，在终端执行以下命令。

执行程序：

```
pi@raspberrypi ~/RPi_Relay_Board/shell $ sudo ./Relay.sh CH1 ON
```

预期结果：继电器通道 1 的 LED 被点亮，同时听到继电器接合的声音。命令中后面两个参数可改变，例如运行如下命令分别为继电器 2 接合，继电器 3 断开。

```
pi@raspberrypi ~/RPi_Relay_Board/shell $ sudo ./Relay.sh CH2 ON
```

```
pi@raspberrypi ~/RPi_Relay_Board/shell $ sudo ./Relay.sh CH3 OFF
```

2) Bcm2835 程序

编译程序：

```
pi@raspberrypi ~/RPi_Relay_Board/bcm2835 $ make
```

终端执行：

```
pi@raspberrypi ~/RPi_Relay_Board/bcm2835 $ sudo ./Relay_Module
```

预期结果: 可以看到 3 个 LED 依次点亮, 继电器依次在常开触点和常闭触点之间来回切换。同时终端会显示目前继电器在哪个触点。

按键盘 **Ctrl + C** 结束实验。

3) wiringPi 程序

编译程序:

```
pi@raspberrypi ~/RPi_Relay_Board/wiringPi $ make
```

终端执行:

```
pi@raspberrypi ~/RPi_Relay_Board/wiringPi $ sudo ./Relay_Module
```

预期结果: 可以看到 3 个 LED 依次点亮, 继电器依次在常开触点和常闭触点之间来回切换。同时终端会显示目前继电器在哪个触点。

按键盘 **Ctrl + C** 结束实验。

4) python 程序

终端执行:

```
pi@raspberrypi ~/RPi_Relay_Board/python $ sudo python Relay_Module.py
```

预期结果: 可以看到 3 个 LED 依次被点亮, 继电器依次在常开触点和常闭触点之间来回切换。同时终端会显示目前继电器在哪个触点。

按键盘 **Ctrl + C** 结束实验。

5) python-bottle 程序

本例程的网页控制是基于 python Web 框架来控制继电器的。

● 环境安装:

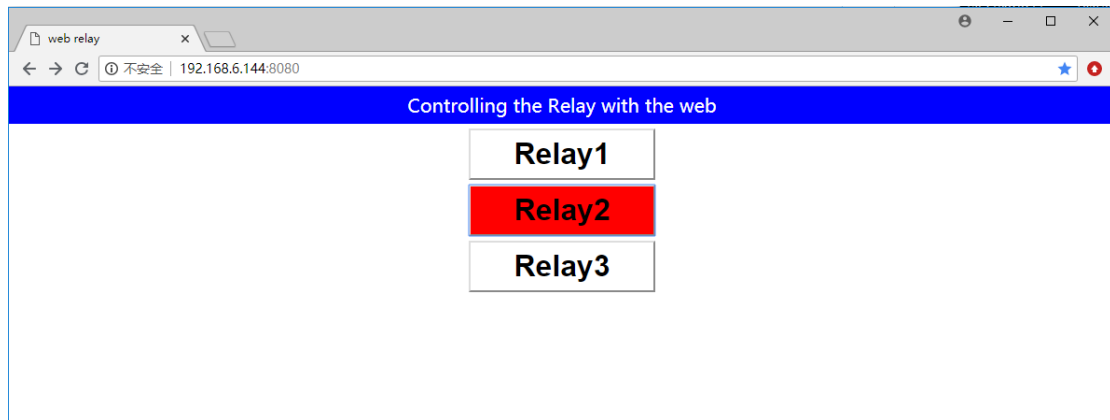
```
sudo apt-get install python-bottle
```

运行如下命令启动程序:

```
cd ~/python-bottle
```

```
sudo python main.py
```

在谷歌浏览器（其他浏览器可能不兼容）地址栏内输入树莓派 ip 地址，端口号 8080



点击网页上的不同继电器通道的控件可以控制继电器。