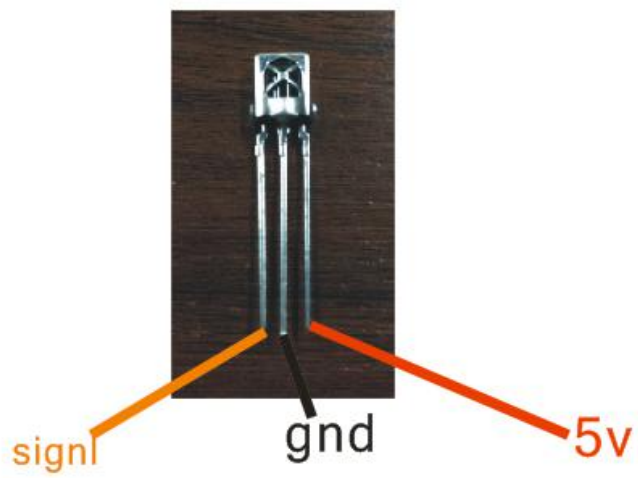
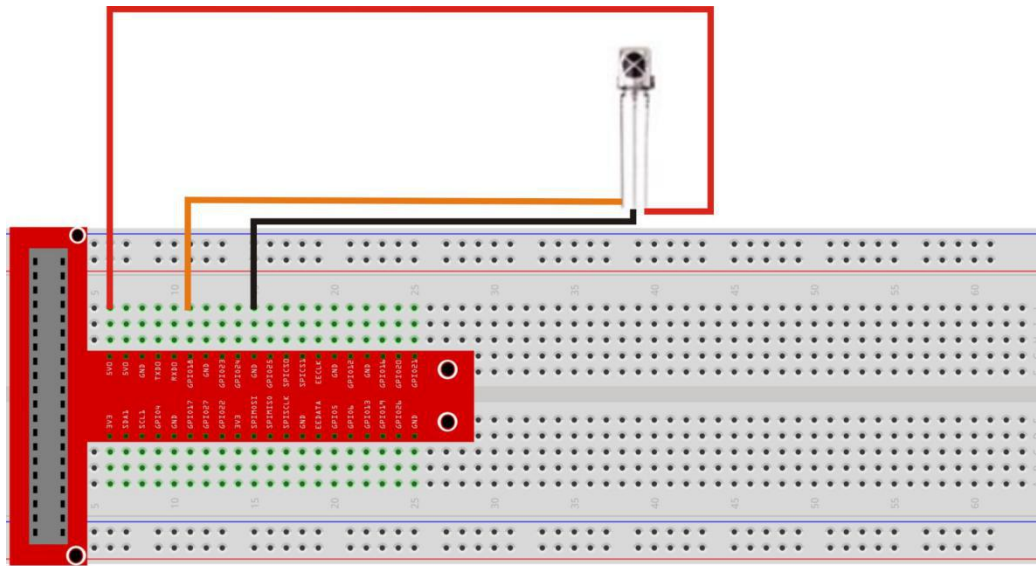


IRremote

Product pictures:



Wiring diagram:



1. Install LIRC, LIRC (Linux Infrared Remote Control) is an open source software package for Linux. Allows the Linux system to receive and send infrared signals, please be careful to connect the Raspberry Pi to the network
Use the command: `sudo apt-get install lirc`

```
pi@raspberrypi:~ $ sudo apt-get install lirc
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libftdii liblircclient0
Suggested packages:
  lirc-x setserial ir-keytable
The following NEW packages will be installed:
  libftdii liblircclient0 lirc
0 upgraded, 3 newly installed, 0 to remove and 266 not upgraded.
Need to get 389 kB of archives.
After this operation, 1,559 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

2. Use the command to enter the configuration file directory and modify it

Modify content:

LIRCD_ARGS="--uinput"

DRIVER="default"

DEVICE="/dev/lirc0"

MODULES="lirc-rpi"

The following figure is the instruction and diagram to be input. After the modification, press the key combination “ctrl+X”, then press Y, and finally press the Enter key.

```
pi@raspberrypi:~ $ cd /etc/lirc
pi@raspberrypi:/etc/lirc $ ls
hardware.conf  lircd.conf  lircmd.conf
pi@raspberrypi:/etc/lirc $ sudo nano hardware.conf
```

```
GNU nano 2.2.6 File: hardware.conf
/etc/lirc/hardware.conf
#
# Arguments which will be used when launching lircd
LIRCD_ARGS="--uinput"

#Don't start lircmd even if there seems to be a good config file
#START_LIRCMD=false

#Don't start irexec, even if a good config file seems to exist.
#START_IEXEC=false

#Try to load appropriate kernel modules
LOAD_MODULES=true

# Run "lircd --driver=help" for a list of supported drivers.
DRIVER="default"
# usually /dev/lirc0 is the correct setting for systems using udev
DEVICE="/dev/lirc0"
MODULES="lirc-rpi"

# Default configuration files for your hardware if any
LIRCD_CONF=""
LIRCMD_CONF=""
```

3. In the configuration file, add the infrared module

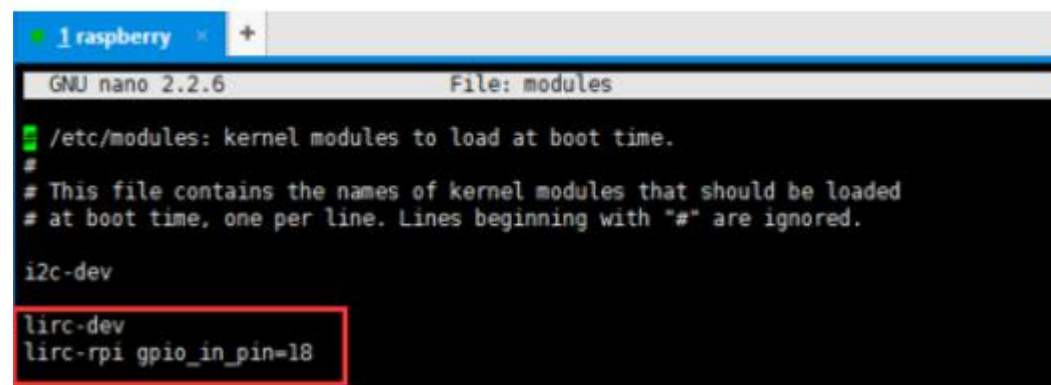
Join statement:

lirc-dev

lirc-rpi gpio_in_pin=18

Because it only uses infrared reception and does not use the infrared emission function, so only add the in_pin statement. Operate as shown below. After the modification, press the key combination “ctrl+X”, then press Y, and finally press enter Key.

```
pi@raspberrypi:/etc/lirc $ cd
pi@raspberrypi:~ $ cd /etc
pi@raspberrypi:/etc $ sudo nano modules
```



```
1 raspberry
GNU nano 2.2.6 File: modules

/etc/modules: kernel modules to load at boot time.
#
# This file contains the names of kernel modules that should be loaded
# at boot time, one per line. Lines beginning with "#" are ignored.

i2c-dev

lirc-dev
lirc-rpi gpio_in_pin=18
```

4. Modify the system config configuration file

Remove the "#" in front of the sentence in the red box, after the modification, press the key combination "ctrl + X", then press Y, and finally press the Enter key

```
pi@raspberrypi:~ $ cd /boot
pi@raspberrypi:/boot $ sudo nano config.txt
```



```
# Uncomment this to enable the lirc-rpi module
dtoverlay=lirc-rpi
```

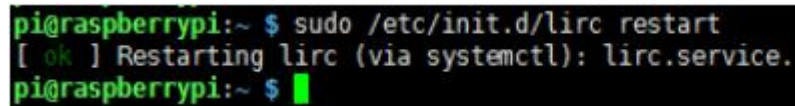
5. Restart the Raspberry Pi Controller

Use the command: sudo reboot

```
pi@raspberrypi:~ $ sudo reboot
```

6. Re-open the lirc function, so that the lirc software is configured.

`sudo /etc/init.d/lirc restart`



```
pi@raspberrypi:~ $ sudo /etc/init.d/lirc restart
[ ok ] Restarting lirc (via systemctl): lirc.service.
pi@raspberrypi:~ $
```

7. Infrared reception configuration detection

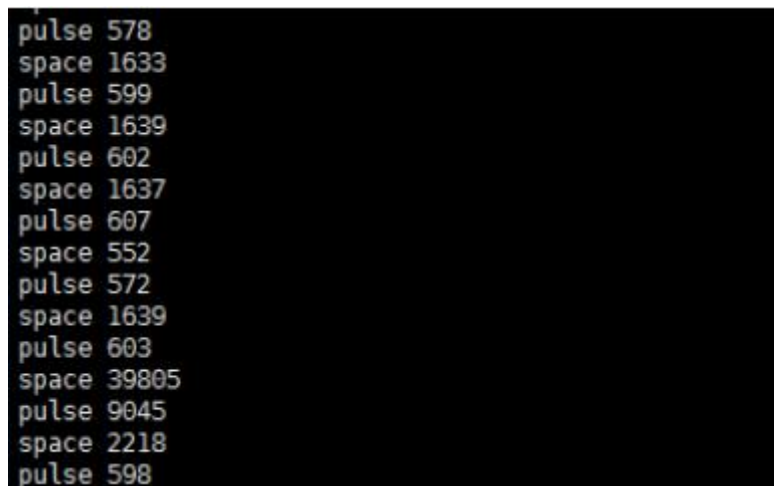
First shut down the lirc software, use the following command

`sudo /etc/init.d/lirc stop`

Then execute the following command

`mode2 -d /dev/lirc0`

Pressing any key of the remote control will return something similar to the following figure, indicating that the infrared receiving configuration is successful.



```
pulse 578
space 1633
pulse 599
space 1639
pulse 602
space 1637
pulse 607
space 552
pulse 572
space 1639
pulse 603
space 39805
pulse 9045
space 2218
pulse 598
```

8. Recording infrared code

First shut down the lirc software and look at the available key names

Use the command:

`sudo /etc/init.d/lirc stop`

`irrecord -list-namespaces`

Return to the currently available key name

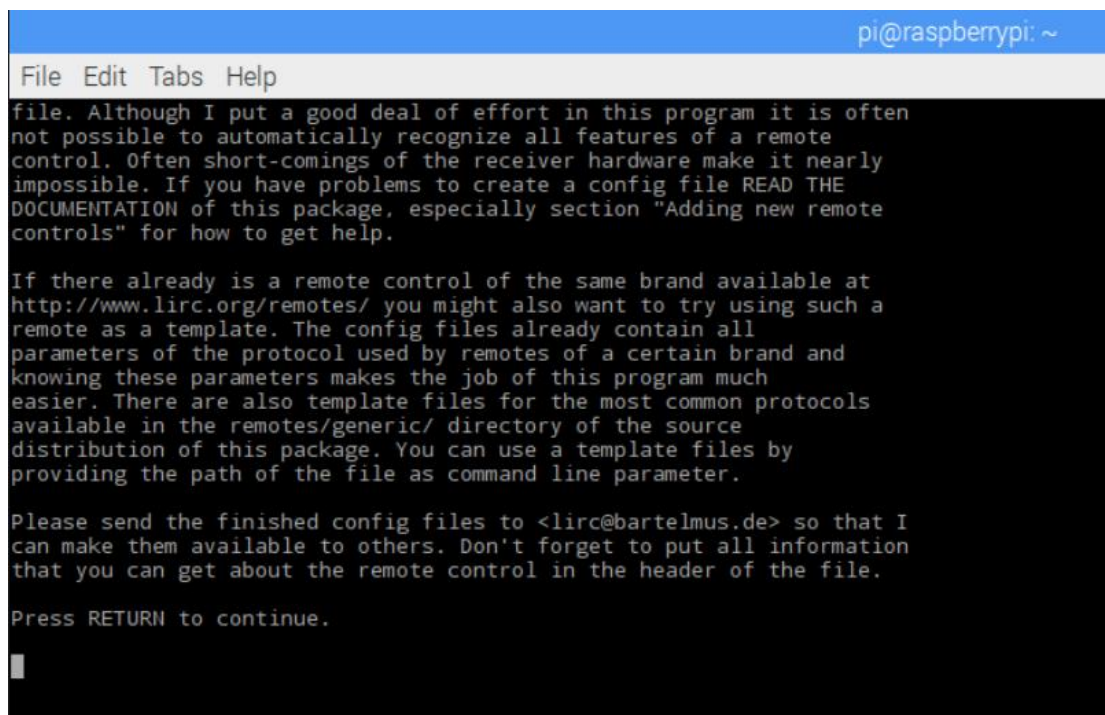
```

pi@raspberrypi:~ $ irrecord -list-namespace
KEY_0
KEY_102ND
KEY_1
KEY_2
KEY_3
KEY_4
KEY_5
KEY_6
KEY_7
KEY_8
KEY_9
KEY_A
KEY_AB

```

9. Execute the following IR coded recording command. The following picture will appear. Press the Enter button twice.

Command: `irrecord -d /dev/lirc0 ~/lircd.conf`



```

pi@raspberrypi: ~
File Edit Tabs Help
file. Although I put a good deal of effort in this program it is often
not possible to automatically recognize all features of a remote
control. Often short-comings of the receiver hardware make it nearly
impossible. If you have problems to create a config file READ THE
DOCUMENTATION of this package, especially section "Adding new remote
controls" for how to get help.

If there already is a remote control of the same brand available at
http://www.lirc.org/remotes/ you might also want to try using such a
remote as a template. The config files already contain all
parameters of the protocol used by remotes of a certain brand and
knowing these parameters makes the job of this program much
easier. There are also template files for the most common protocols
available in the remotes/generic/ directory of the source
distribution of this package. You can use a template files by
providing the path of the file as command line parameter.

Please send the finished config files to <lirc@bartelmus.de> so that I
can make them available to others. Don't forget to put all information
that you can get about the remote control in the header of the file.

Press RETURN to continue.

```

Then press any button until "... " fills up one line, then click any button, until the point is filled with the second line, and then the prompt for the name of the input button will pop up, we directly input just detected The name of the button, such as KEY_1, and then press the corresponding button on the remote control, you can complete the recording of the code. Repeat this method to record the other buttons. After the recording is completed, press the Enter button to end the recording.


```

Now start pressing buttons on your remote control.

It is very important that you press many different buttons and hold them
down for approximately one second. Each button should generate at least one
dot but in no case more than ten dots of output.
Don't stop pressing buttons until two lines of dots (2x80) have been
generated.

Press RETURN now to start recording.
.....
Found const length: 108546
Please keep on pressing buttons like described above.
.....
Space/pulse encoded remote control found.
Signal length is 67.
Found possible header: 9000 4603
Found trail pulse: 501
Found repeat code: 9001 2326
Signals are space encoded.
Signal length is 32
Now enter the names for the buttons.

Please enter the name for the next button (press <ENTER> to finish recording)
KEY_1

Now hold down button "KEY_1".

Please enter the name for the next button (press <ENTER> to finish recording)
KEY_2

Now hold down button "KEY_2".

Please enter the name for the next button (press <ENTER> to finish recording)
KEY_3

```

10. Overwrite the recorded configuration file with the corresponding file of lirc software. Here is the command:
`sudo cp ~/lircd.conf /etc/lirc/lircd.conf`

```

pi@raspberrypi:~ $ sudo cp ~/lircd.conf /etc/lirc/lircd.conf
pi@raspberrypi:~ $

```

11. Test the recording result of the infrared remote control button
input the command:
`sudo /etc/init.d/lirc start`

`irw`

Pressing the button just recorded by the infrared remote controller at this time will display the name of the button.

```

pi@raspberrypi:~ $ sudo /etc/init.d/lirc start
[ ok ] Starting lirc (via systemctl): lirc.service.
pi@raspberrypi:~ $ irw
0000000000ffa25d 00 KEY_1 /home/pi/lircd.conf
1000000000ffa25d 01 KEY_1 /home/pi/lircd.conf
0000000000ff629d 00 KEY_2 /home/pi/lircd.conf
2000000000ffe21d 00 KEY_3 /home/pi/lircd.conf
3000000000ffe21d 01 KEY_3 /home/pi/lircd.conf
0000000000ff22dd 00 KEY_4 /home/pi/lircd.conf
4000000000ff02fd 00 KEY_5 /home/pi/lircd.conf
5000000000ffc23d 00 KEY_6 /home/pi/lircd.conf
6000000000ffe01f 00 KEY_7 /home/pi/lircd.conf
7000000000ffe01f 01 KEY_7 /home/pi/lircd.conf
0000000000ffa857 00 KEY_8 /home/pi/lircd.conf
8000000000ff906f 00 KEY_9 /home/pi/lircd.conf
9000000000ff906f 01 KEY_9 /home/pi/lircd.conf
0000000000ff6897 00 KEY_A /home/pi/lircd.conf
A000000000ff9867 00 KEY_AB /home/pi/lircd.conf
0000000000ffb04f 00 KEY_F10 /home/pi/lircd.conf

```

12. The next thing is important, press the button, the Raspberry Pi to execute a command, use the key KEY_1 to reboot the system

Edit .lircrc and execute the following command

`sudo pico ~/.lircrc`

Add the following to the file

begin

 prog = irexec

 button = KEY_1

 config = sudo reboot

end

```

GNU nano 2.2.6 File: /home/pi/.l
begin
    prog = irexec
    button = KEY_1
    config = sudo reboot
end

```

"KEY_1" is the key name, "sudo reboot" is the command to restart the Raspberry Pi

The following command can be used to control the remote control, press the "KEY_1" button, it will restart the Raspberry Pi.

`irexec -d`

13. You can also add other key functions behind the file, such as "KEY_2" button to start my desktop python code, such as the following:

```
begin
    prog = irexec
    button = KEY_1
    config = sudo reboot
end
begin
    prog = irexec
    button = KEY_2
    config = sudo python /home/pi/Desktop/led.py
end
```

14. Edit rc.local to enable the raspberry pie to automatically enable infrared reception upon startup. Now execute the following command:

```
sudo nano /etc/rc.local
```

In front of exit0, add the following:

```
(sleep 3;
sudo -u pi irexec -d
)&
```

As shown below:

```
(sleep 3;
sudo -u pi irexec -d
)&
exit 0
```

After the restart, the configuration takes effect. After the Raspberry Pi is started, the remote control key functions are activated directly.