

微雪官网

产品资料

QQ登录

用户名

密码

自动登录

登

W

微雪课堂

WAVESHARE STUDY

请输入搜索内容

微雪课堂

微雪课堂

树莓派

查看内容

树莓派系列教程15：红外遥控

2015-9-2 09:00 | 发布者: MyMX1213 | 查看: 6075 | 评论: 6 | 原作者: MyMX1213

摘要: 上一章我们介绍了如果通过树莓派devicetree,将在ds18b20添加到linux系统中,并通过命令行读取温度数据,这一章我们也通过devicetree添加红外接收lirc为linux系统中红外遥控的软件,树莓派系统已经有这个模块,我们只...

上一章我们介绍了如果通过树莓派device tree,将在ds18b20添加到linux系统中,并通过命令行读取温度数据,这一章我们也通过device tree添加红外接收lirc为linux系统中红外遥控的软件,树莓派系统已经有这个模块,我们只需设置一下就已使用。

1 | sudo vi /boot/config.txt

在文件后面添加下面这一行

1 | dtoverlay=lirc-rpi,gpio_in_pin=18

红外默认输出是18管脚,如果红外接收头接到其他管脚则需修改对应管脚,(管脚为BCM编码),Pioneer 600接收头默认接到18管脚故只需要添加

1 | dtoverlay=lirc-rpi

在/boot/overlay/README文件中我们可以找到详细说明。

```
64 This causes the file /boot/overlays/lirc-rpi-overlay.dtb to be loaded. By
65 default it will use GPIOs 17 (out) and 18 (in), but this can be modified using
66 DT parameters:
67
68     dtoverlay=lirc-rpi,gpio_out_pin=17,gpio_in_pin=13
69
```

安装lirc软件

1 | sudo apt-get install lirc

运行lsmod命令查看设备是否已启动,如若没有找到可运行sudo modprobe lirc_rpi加载驱动。

```
pi@raspberrypi / $ lsmod
Module                Size  Used by
cfg80211              386508  0
rfkill                16651   1 cfg80211
i2c_dev                6027   0
snd_bcm2835           18649   3
snd_pcm               73475   1 snd_bcm2835
snd_seq               53078   0
snd_seq_device         5628   1 snd_seq
snd_timer             17784   2 snd_pcm,snd_seq
snd                   51038  11 snd_bcm2835,snd_pcm,snd_seq,snd_seq_device
wl_therm               2559   0
spi_bcm2708            5137   0
i2c_bcm2708            4990   0
wl_gpio                3465   0
lirc_rpi               6646   0
wire                  25680   2 wl_gpio,wl_therm
cn                     4636   1 wire
lirc_dev               8181   1 lirc_rpi
uio_pdrv_genirq        2958   0
rc_core                16932   1 lirc_dev
uio                    8119   1 uio_pdrv_genirq
pi@raspberrypi / $
```

运行sudo mode2 -d /dev/lirc0,按遥控上任何键,查看是否接到类似脉冲。

微雪课堂

树莓派

Arduino

C8051

PIC

STM8

FPGA

01 Alphabot树莓派教程

lede是openwrt的一个分支支持pi3

02 Alphabot树莓派教程

03 树莓派系列教程18：1

04 树莓派系列教程17：1

05 树莓派系列教程16：1

06 树莓派系列教程15：1

07 树莓派系列教程14：1

08 树莓派系列教程13：1

09 树莓派系列教程12：1

010 树莓派系列教程11

011 树莓派系列教程10

012 树莓派系列教程9：

013 树莓派系列教程8：1

014 树莓派系列教程8：

015 树莓派教程系列7：

016 树莓派教程系列6：

017 树莓派系列教程5：

018 树莓派系列教程4：

019 树莓派系列教程3：

020 树莓派系列教程3：

021 树莓派系列教程3：

022 树莓派系列教程2：

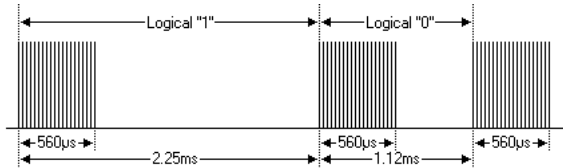
023 树莓派系列教程1：

http://www.waveshare.net/study/article-622-1.html

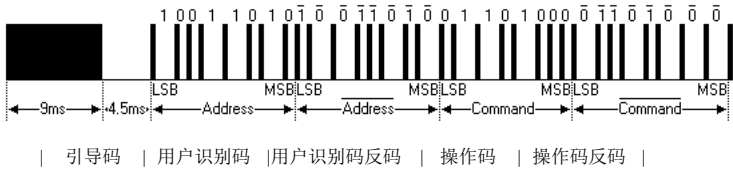
1/5



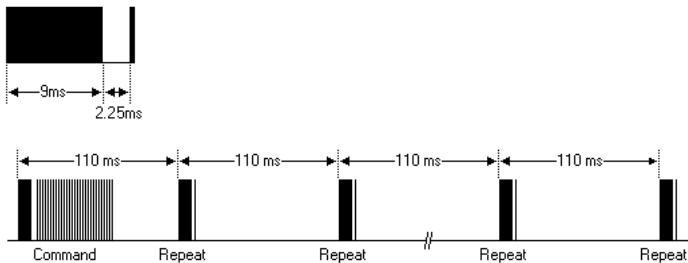
如有接到脉冲测lirc正常使用。
采用脉宽调制的串行码，以脉宽为0.565ms、间隔0.56ms、周期为1.125ms的组合表示二进制的“0”；以脉宽为0.565ms、间隔1.685ms、周期为2.25ms的组合表示二进制的“1”



协议：
上述“0”和“1”组成的32位二进制码经38kHz的载频进行二次调制以提高发射效率，达到降低电源功耗的目的。然后再通过红外发射二极管产生红外线向空间发射，如下图。



一个命令只发送一次，即使遥控器上的按键一直按着。但是会每110ms发送一次代码，直到遥控器按键释放。
重复码比较简单：一个9mS的AGC脉冲、2.25mS间隔、560uS脉冲。



bcm2835程序：
cpp代码：

```
01 #include <bcm2835.h>
02 #include <stdio.h>
03 #define PIN 18
04 #define IO bcm2835_gpio_lev(PIN)
05 unsigned char i,idx,cnt;
06 unsigned char count;
07 unsigned char data[4];
08
09 int main(int argc, char **argv)
10 {
11     if (!bcm2835_init())return 1;
12     bcm2835_gpio_fsel(PIN, BCM2835_GPIO_FSEL_INPT);
13     bcm2835_gpio_set_pud(PIN, BCM2835_GPIO_PUD_UP);
14     printf("IRM Test Program ... \n");
15
16     while (1)
17     {
18         if(IO == 0)
19         {
20             count = 0;
21             while(IO == 0 && count++ < 200) //9ms
22                 delayMicroseconds(60);
23
24             count = 0;
25             while(IO == 1 && count++ < 80) //4.5ms
26                 delayMicroseconds(60);
27
28             idx = 0;
29             cnt = 0;
30             data[0]=0;
31             data[1]=0;
32             data[2]=0;
33             data[3]=0;
34             for(i =0;i<32;i++)
35             {
```

```

36         count = 0;
37         while(IO == 0 && count++ < 15) //0.56ms
38             delayMicroseconds(60);
39
40         count = 0;
41         while(IO == 1 && count++ < 40) //0: 0.56ms; 1: 1.69ms
42             delayMicroseconds(60);
43
44         if (count > 25) data[idx] |= (1<1<cnt);
45         if(cnt== 7)
46         {
47             cnt=0;
48             idx++;
49         }
50         else cnt++;
51         if(data[0]+data[1] == 0xFF && data[2]+data[3]==0xFF) //check
52             printf("Get the key: 0x%02x\n",data[2]);
53     }
54     bcm2835_close();
55     return 0;
56 }
57
58 }

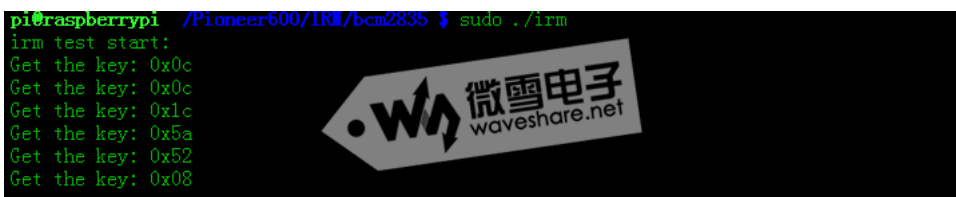
```

编译并执行，按下遥控按键，终端会显示接到到按键的键值。

```

1 gcc -Wall irm.c -o irm -lbcm2835
2 sudo ./irm

```



```

pi@raspberrypi ~/Pioneer600/IRM/bcm2835 $ sudo ./irm
irm test start:
Get the key: 0x0c
Get the key: 0x0c
Get the key: 0x1c
Get the key: 0x5a
Get the key: 0x52
Get the key: 0x08

```

python程序

```

01 #!/usr/bin/python
02 # -*- coding:utf-8 -*-
03 import RPi.GPIO as GPIO
04 import time
05
06 PIN = 18;
07
08 GPIO.setmode(GPIO.BCM)
09 GPIO.setup(PIN,GPIO.IN,GPIO.PUD_UP)
10 print('IRM Test Start ...')
11 try:
12     while True:
13         if GPIO.input(PIN) == 0:
14             count = 0
15             while GPIO.input(PIN) == 0 and count < 200: #9ms
16                 count += 1
17                 time.sleep(0.00006)
18
19             count = 0
20             while GPIO.input(PIN) == 1 and count < 80: #4.5ms
21                 count += 1
22                 time.sleep(0.00006)
23
24             idx = 0
25             cnt = 0
26             data = [0,0,0,0]
27             for i in range(0,32):
28                 count = 0
29                 while GPIO.input(PIN) == 0 and count < 15: #0.56ms
30                     count += 1
31                     time.sleep(0.00006)
32
33                 count = 0
34                 while GPIO.input(PIN) == 1 and count < 40: #0: 0.56ms; 1: 1.69ms
35                     count += 1
36                     time.sleep(0.00006)
37
38                 if count > 8:
39                     data[idx] |= 1>>cnt
40                 if cnt == 7:
41                     cnt = 0
42                     idx += 1
43                 else:
44                     cnt += 1
45             if data[0]+data[1] == 0xFF and data[2]+data[3] == 0xFF: #check
46                 print("Get the key: 0x%02x" %data[2])
47 except KeyboardInterrupt:
48     GPIO.cleanup();

```

执行，按下遥控按键，终端会显示接到到按键的键值。

```
1 sudo python irm.py
```

```
pi@raspberrypi ~/Pioneer600/IRM/python $ sudo python irm.py
IRM Test Start ...
Get the key: 0x0c
Get the key: 0x0c
Get the key: 0x18
Get the key: 0x5e
Get the key: 0x1c
Get the key: 0x42
Get the key: 0x52
|
```



48



顶一下



刚表态过的朋友 (48 人)

邀请 收藏

上一篇：树莓派系列教程14：单总线控制DS18B20
下一篇：树莓派系列教程16：RTC

🔄 相关阅读

- 树莓派系列教程2：树莓派烧写镜像
 - 树莓派系列教程3：访问树莓派之SSH篇
 - 树莓派系列教程3：访问树莓派之WiFi篇
- 树莓派教程系列6：文件共享(samba)
 - 树莓派教程系列7：wiringPi、bcm2835、python库安装

🔄 最新评论

发表评论

- 游客 2017-2-8 11:39

引用

sudo python irm.py
显示：“IRM Test Start ...”
可是遥控器按任何遥控器都没反应，是怎么回事呢？
- waveshare-admin 2016-10-26 11:45

引用

“：您好，我输入sudo mode2 -d /dev/lirc0之后，它说sudo: mode2: command not found，是因为什么原因呢？ ... ”
如果没打错是可以的，可以加下技术qq咨询下：2853908288
- 游客 2016-10-22 18:32

引用

您好，我输入sudo mode2 -d /dev/lirc0之后，它说sudo: mode2: command not found，是因为什么原因呢？ ...
- kaloha 2016-10-15 09:32

引用

“：我按这个教程试，但是什么都使用遥控器按键后，什么都显示不出来 ”
是到哪一步就不行的？有没有出错提示？
- 游客 2016-10-14 20:05

引用

我按这个教程试，但是什么都使用遥控器按键后，什么都显示不出来
- 田伟 2016-8-5 23:19

引用

doverlay=lirc-rpi,gpio_in_pin=18
dtoverlay=lirc-rpi,gpio_in_pin=18

[查看全部评论\(6\)](#)

验证码 [换一个](#)

评论

