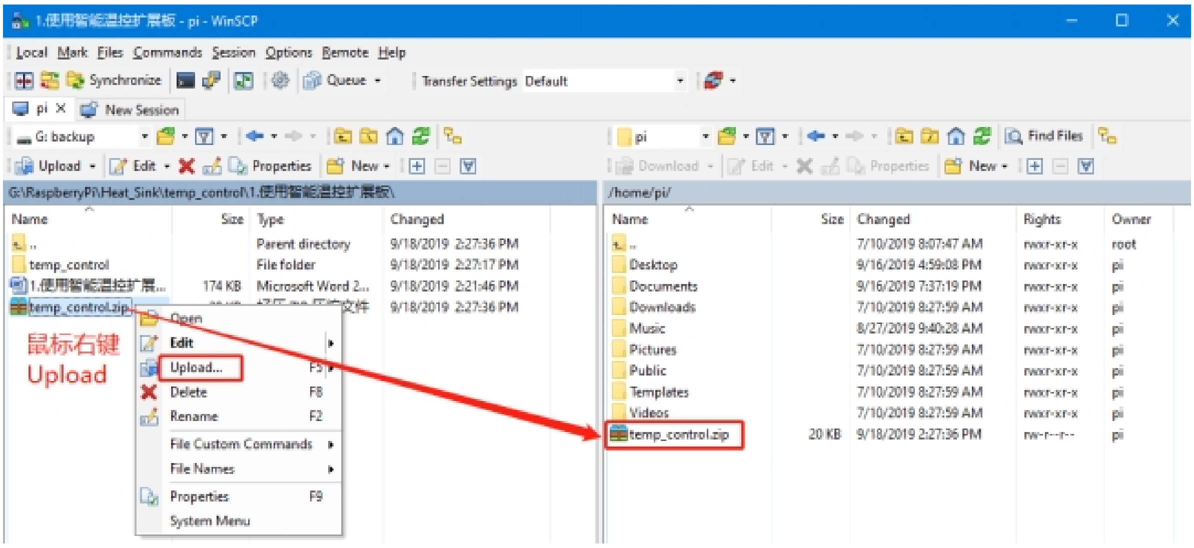
Control fan

RGB Cooling HAT needs to be correctly inserted into the GPIO port of RDK X3, and the I2C function of RDK X3 must be turned on.

The phenomenon of this experiment is that control the fan speed, start after 2 seconds of rest, increase the fan speed every second, run at the highest speed for 2 seconds and then stop.

1.File transfer

1.Install WinSCP on your computer, connect to RDK X3, and transfer the **temp_control.zip** package downloaded from the documentation to the pi directory of the RDK X3.



G:\RaspberryPi\Heat_Sink\temp_control\1.使用智能温控扩展板\

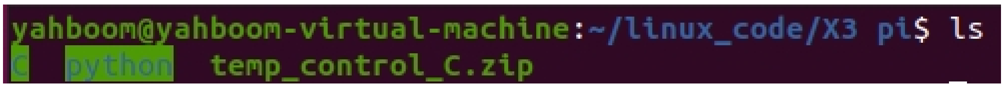
Name	Size	Type	Changed
..		Parent directory	9/18/2019 2:27:36 PM
temp_control		File folder	9/18/2019 2:27:17 PM
1.使用智能温控扩展...	174 KB	Microsoft Word 2...	9/18/2019 2:21:46 PM
temp_control.zip	[illegible]	[illegible]压缩文件	9/18/2019 2:27:36 PM

/home/pi/

Name	Size	Changed	Rights	Owner
..		7/10/2019 8:07:47 AM	rwxr-xr-x	root
Desktop		9/16/2019 4:59:08 PM	rwxr-xr-x	pi
Documents		9/16/2019 7:37:19 PM	rwxr-xr-x	pi
Downloads		7/10/2019 8:27:59 AM	rwxr-xr-x	pi
Music		8/27/2019 9:40:28 AM	rwxr-xr-x	pi
Pictures		7/10/2019 8:27:59 AM	rwxr-xr-x	pi
Public		7/10/2019 8:27:59 AM	rwxr-xr-x	pi
Templates		7/10/2019 8:27:59 AM	rwxr-xr-x	pi
Videos		7/10/2019 8:27:59 AM	rwxr-xr-x	pi
temp_control.zip	20 KB	9/18/2019 2:27:36 PM	rw-r--r--	pi

2.Extract file

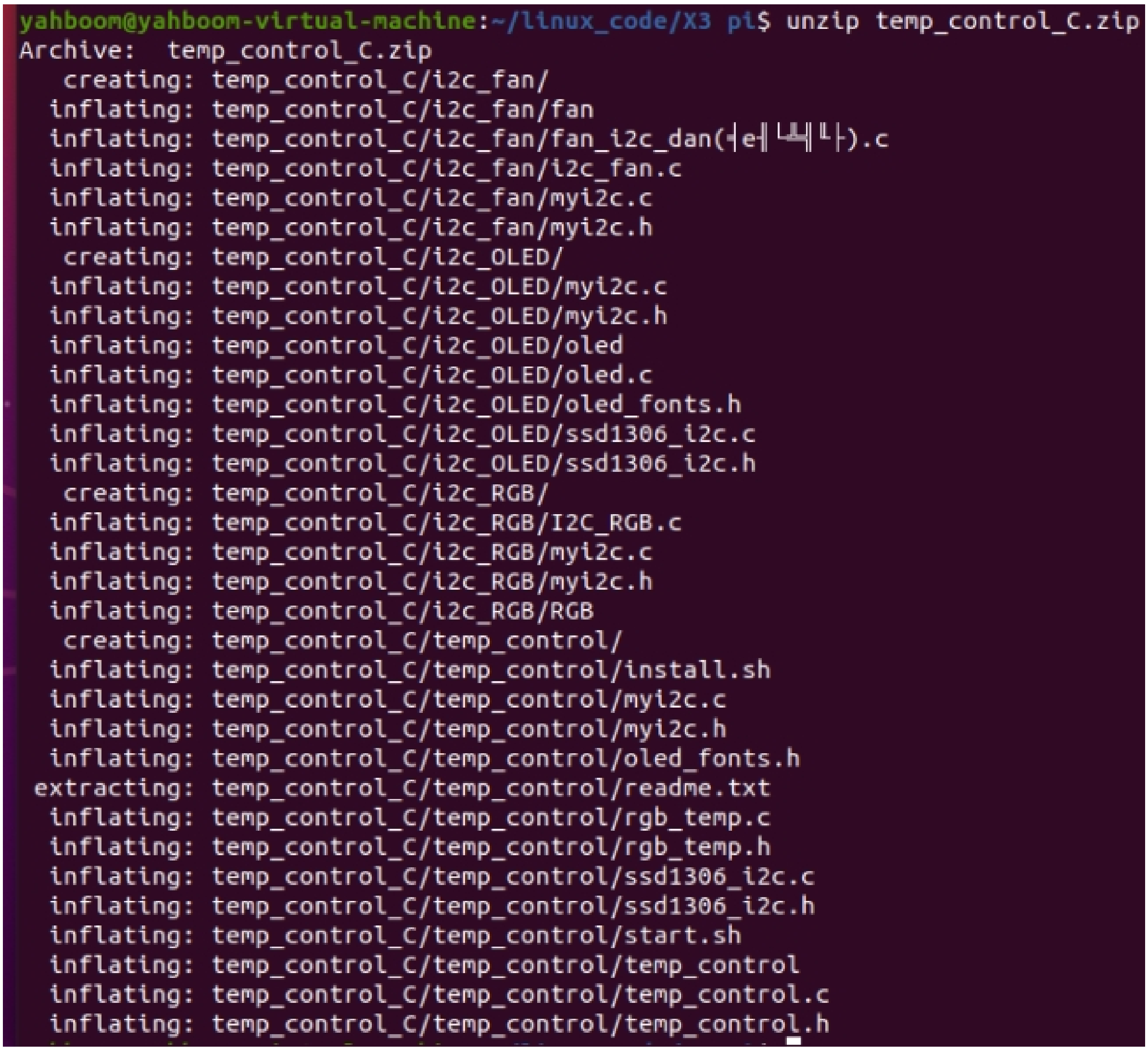
Open the terminal in RDK X3 system and find **temp_control_C.zip** file.



```
yahboom@yahboom-virtual-machine:~/linux_code/X3 pi$ ls
C  python  temp_control_C.zip
```

Enter the following command to extract the file.

```
unzip temp_control_C.zip
```

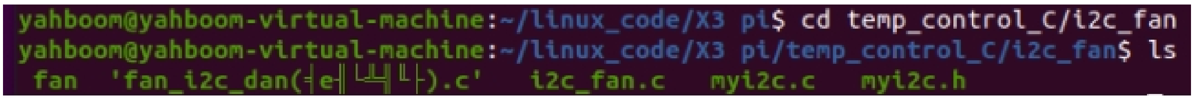


```
yahboom@yahboom-virtual-machine:~/linux_code/X3 pi$ unzip temp_control_C.zip
Archive:  temp_control_C.zip
   creating: temp_control_C/i2c_fan/
  inflating: temp_control_C/i2c_fan/fan
  inflating: temp_control_C/i2c_fan/fan_i2c_dan(┤e┤└┴┤└├).c
  inflating: temp_control_C/i2c_fan/i2c_fan.c
  inflating: temp_control_C/i2c_fan/myi2c.c
  inflating: temp_control_C/i2c_fan/myi2c.h
   creating: temp_control_C/i2c_OLED/
  inflating: temp_control_C/i2c_OLED/myi2c.c
  inflating: temp_control_C/i2c_OLED/myi2c.h
  inflating: temp_control_C/i2c_OLED/oled
  inflating: temp_control_C/i2c_OLED/oled.c
  inflating: temp_control_C/i2c_OLED/oled_fonts.h
  inflating: temp_control_C/i2c_OLED/ssd1306_i2c.c
  inflating: temp_control_C/i2c_OLED/ssd1306_i2c.h
   creating: temp_control_C/i2c_RGB/
  inflating: temp_control_C/i2c_RGB/I2C_RGB.c
  inflating: temp_control_C/i2c_RGB/myi2c.c
  inflating: temp_control_C/i2c_RGB/myi2c.h
  inflating: temp_control_C/i2c_RGB/RGB
   creating: temp_control_C/temp_control/
  inflating: temp_control_C/temp_control/install.sh
  inflating: temp_control_C/temp_control/myi2c.c
  inflating: temp_control_C/temp_control/myi2c.h
  inflating: temp_control_C/temp_control/oled_fonts.h
 extracting: temp_control_C/temp_control/readme.txt
  inflating: temp_control_C/temp_control/rgb_temp.c
  inflating: temp_control_C/temp_control/rgb_temp.h
  inflating: temp_control_C/temp_control/ssd1306_i2c.c
  inflating: temp_control_C/temp_control/ssd1306_i2c.h
  inflating: temp_control_C/temp_control/start.sh
  inflating: temp_control_C/temp_control/temp_control
  inflating: temp_control_C/temp_control/temp_control.c
  inflating: temp_control_C/temp_control/temp_control.h
```

2. Compiling and running the program

1.Enter the folder and view the files in the current folder

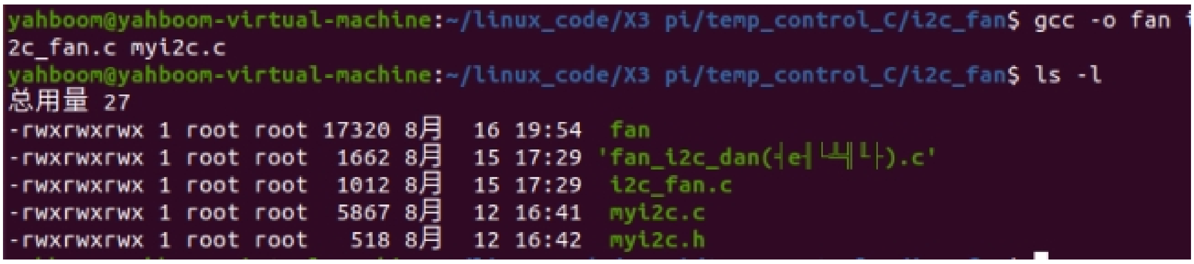
```
cd temp_control_C/i2c_fan
ls
```



```
yahboom@yahboom-virtual-machine:~/linux_code/X3 pi$ cd temp_control_C/i2c_fan
yahboom@yahboom-virtual-machine:~/linux_code/X3 pi/temp_control_C/i2c_fan$ ls
 fan   'fan_i2c_dan(┤e┤└┴┤└├).c'   i2c_fan.c   myi2c.c   myi2c.h
```

2.Compile program files

```
gcc -o fan i2c_fan.c myi2c.c
```



```
yahboom@yahboom-virtual-machine:~/linux_code/X3 pi/temp_control_C/i2c_fan$ gcc -o fan i
2c_fan.c myi2c.c
yahboom@yahboom-virtual-machine:~/linux_code/X3 pi/temp_control_C/i2c_fan$ ls -l
总用量 27
-rwxrwxrwx 1 root root 17320 8月  16 19:54  fan
-rwxrwxrwx 1 root root  1662 8月  15 17:29 'fan_i2c_dan(┤e┤└┴┤└├).c'
-rwxrwxrwx 1 root root  1012 8月  15 17:29  i2c_fan.c
-rwxrwxrwx 1 root root  5867 8月  12 16:41  myi2c.c
-rwxrwxrwx 1 root root   518 8月  12 16:42  myi2c.h
```

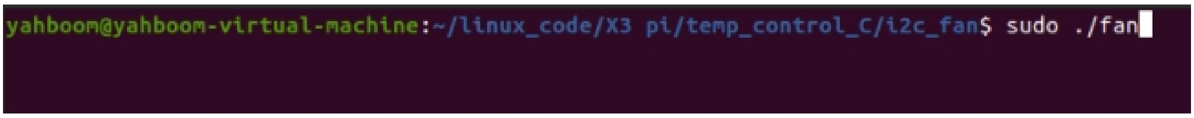
Among them, the gcc compiler is called, -o means to generate a file, followed by the generated file name, and fan.c is the source program.

3.Run program

```
sudo ./fan
```

Press **Enter** and enter the password: **sunrise**

The password is invisible when you enter it.

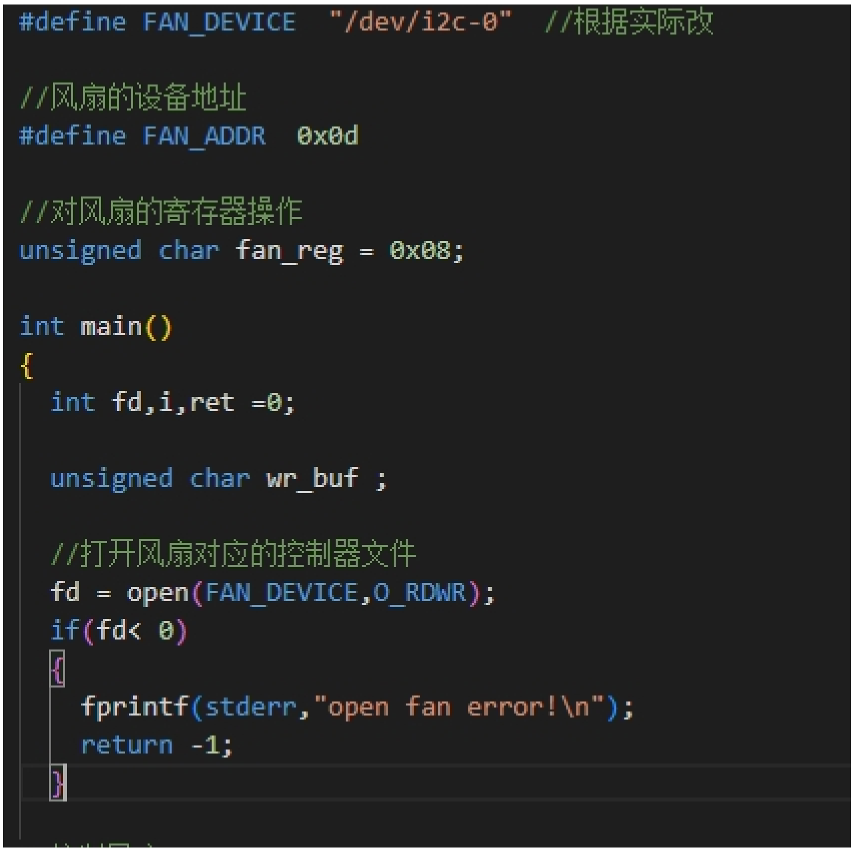


```
yahboom@yahboom-virtual-machine:~/linux_code/X3 pi/temp_control_C/i2c_fan$ sudo ./fan
```

At this time, the fan will continue to run after the user enters the speed.

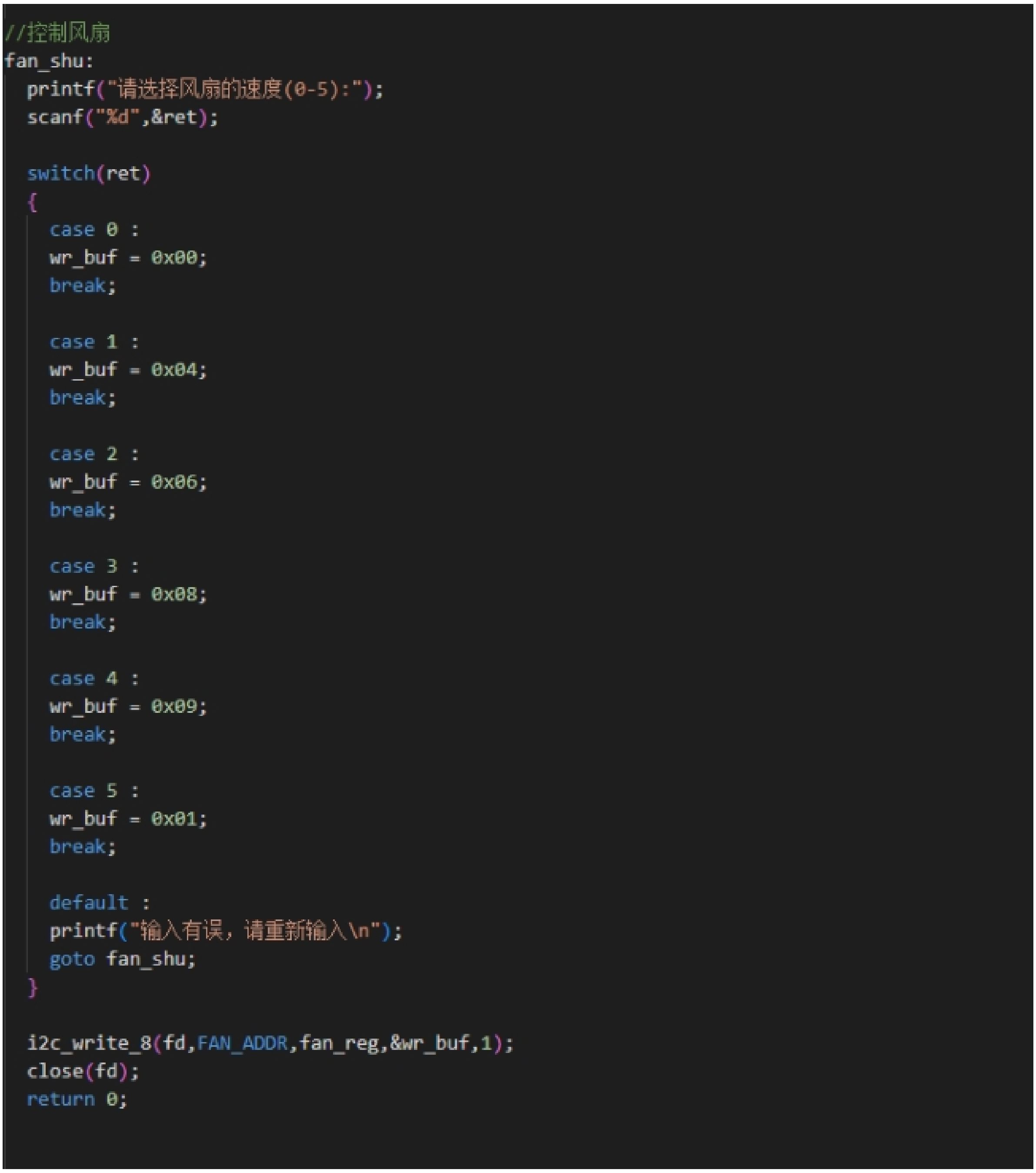
3. Code

1. Initialize RDK -X3 I2C configuration



```
#define FAN_DEVICE  "/dev/i2c-0"  //根据实际改

//风扇的设备地址
#define FAN_ADDR  0x0d

//对风扇的寄存器操作
unsigned char fan_reg = 0x08;

int main()
{
  int fd,i,ret =0;

  unsigned char wr_buf ;

  //打开风扇对应的控制器文件
  fd = open(FAN_DEVICE,O_RDWR);
  if(fd< 0)
  {
    fprintf(stderr,"open fan error!\n");
    return -1;
  }
```

2.The user controls the fan speed. According to the protocol, the fan speed level can be known, 0x00 off, 0x01 full speed, 0x02: 20% speed, 0x03: 30% speed, ..., 0x09: 90% speed.



```
//控制风扇
fan_shu:
  printf("请选择风扇的速度(0-5):");
  scanf("%d",&ret);

  switch(ret)
  {
    case 0 :
    wr_buf = 0x00;
    break;

    case 1 :
    wr_buf = 0x04;
    break;

    case 2 :
    wr_buf = 0x06;
    break;

    case 3 :
    wr_buf = 0x08;
    break;

    case 4 :
    wr_buf = 0x09;
    break;

    case 5 :
    wr_buf = 0x01;
    break;

    default :
    printf("输入有误，请重新输入\n");
    goto fan_shu;
  }

  i2c_write_8(fd,FAN_ADDR,fan_reg,&wr_buf,1);
  close(fd);
  return 0;
```

3. 0: turns off the fan, 5 is the maximum speed