1. 创建出来的文件权限与mkdir一样

#include <sys/stat.h>

int main(void) {

mkdir("/home/battery/Desktop/newdir", S\_IRWXU | S\_IRWXG | S\_IROTH | S\_IXOTH );

}

windows是#include <direct.h>

2. popen的使用

#include <stdio.h>

#include <unistd.h>

#include <sys/stat.h>

const int PIPE\_BUF\_SIZE =8\*512;

int main(void) {

FILE \*rfile0 =popen("/home/battery/Desktop/echo.sh", "r");

FILE \*wfile0 =popen("cat -", "w");

if( !access("/tmp/webcamDir", F\_OK) )

mkdir("/tmp/webcamDir", S\_IRWXU | S\_IRWXG | S\_IROTH | S\_IXOTH );

FILE \*wfile1 =popen("cat - > ~/Desktop/testOutput", "w");

char buf[PIPE\_BUF\_SIZE]; int nRealSize;

while(1) {

nRealSize=fread(buf, PIPE\_BUF\_SIZE, 1, rfile0);

fwrite(buf, nRealSize, 1, wfile0);

fwrite(buf, nRealSize, 1, wfile1);

fflush(wfile0); fflush(wfile1);

}

pclose(rfile0);

pclose(wfile0);

pclose(wfile1);

return 0;

}

#echo.sh

#!/bin/bash //这不是非要在第一行

a=1

while [ true ]

do

echo -n $a

done

3. 一个理解缓冲区的程序

#include <stdio.h>

#include <unistd.h>

#include <sys/stat.h>

const int PIPE\_BUF\_SIZE =8\*512;

int main(void) {

FILE \*rfile0 =popen("~/Desktop/echo.sh", "r");

char buf[PIPE\_BUF\_SIZE]; int nRealSize;

while(1) {

nRealSize=fread(buf, PIPE\_BUF\_SIZE, 1, rfile0);

printf("%d", nRealSize);

fflush(stdout); //没有这个就是隔一段时间出来一片

}

pclose(rfile0);

return 0;

}

#echo.sh

#!/bin/bash

a=1

while [ true ]

do

echo -n $a

done

树莓派里面的gcc居然对文件名有要求：

pi@raspberrypi:~/test $ gcc back.test

back.test: file not recognized: File format not recognized

collect2: error: ld returned 1 exit status

NULL在stdio.h中定义，不过C里面的定义不是很好。

#define NULL (void \*)0

后台进程中将stdin，stdout，stderr重定向到/dev/null的问题：

有人认为对于后台守护进程做此类重定向操作浪费资源，建议直接关闭0、1、2号句

柄拉倒，这是非常不正确的。假设它们确实被关闭了，则一些普通数据文件句柄将等

于0、1、2。以2号句柄为例，某些库函数失败后会向2号句柄输出错误信息，这将破

坏原有数据。所以都是重定向到/dev/null。还有一些没看到但是最推荐的代码是：

int fd = open( "/dev/null", O\_RDWR );

/\*

\* handle failure of open() somehow

\*/

dup2( fd, 0 );

dup2( fd, 1 );

dup2( fd, 2 );

if ( fd > 2 )

{

close( fd );

｝

检测文件是否存在：使用open函数O\_EXCL | O\_CREAT 尝试打开

创建文件（包括临时）先设置UMASK 设置权限的“补码”，和chmod恰好相反。再使用tmpfile()或者mktemp()。