

Linux编程:linux串口示例Linux认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/644/2021_2022_Linux_E7_BC_96_E7_A8_c103_644828.htm 最近看了下linux的串口，发现还是蛮容易的做一些总结和记录【这篇文章也重在备份和记录，代码都是套用别人的，所以基本只是罗列了些代码，但保证代码可用】其实串口操作也就那么几步 1.打开串口 2.设置参数 3.发送接收 4.按需关闭 而根据函式提供的形式，一般设置参数分两步进行【其实就是那么配置下，分几步都行，只是配合后面的代码了】 [1]设置波特率 [2]设置数据格式 下面还是罗列一些代码 打开串口

```
int OpenDev(char *Dev) { int fd = open( Dev, O_RDWR ). if (-1 == fd) { perror("Cant Open Serial Port"). return -1. } else return fd. } O_RDWR就是可读写的意思
```

设置波特率

```
Code int speed_arr[] = { B38400, B19200, B9600, B4800, B2400, B1200, B300, B38400, B19200, B9600, B4800, B2400, B1200, B300, }, int name_arr[] = {38400, 19200, 9600, 4800, 2400, 1200, 300, 38400, 19200, 9600, 4800, 2400, 1200, 300, }, void set_speed(int fd, int speed){ int i. int status. struct termios Opt. tcgetattr(fd, lt. sizeof(speed_arr) / sizeof(int). i ) { if (speed == name_arr[i]) { tcflush(fd, TCIOFLUSH). cfsetispeed(&Opt, speed_arr[i]). status = tcsetattr(fd, TCSANOW, &Opt). if (status != 0) { perror("tcsetattr fd1"). return. } tcflush(fd,TCIOFLUSH). } } }
```

设置参数是用到了一个专用的结构体struct termios其实也没啥说的，就是通过它去配置串口参数罢了注意tcflush,他是清空buffer用的，关于buffer这东西，里面其实挺饶，这里不多说明，只是要注意它清空的buffer并不是printf那类函式中所谓的缓冲。

另外那两个数组，其实可以弄的简单些，只不过懒得改了有点像画刷的使用，旧的一般都爱保存起来，最后还要还原。
100Test 下载频道开通，各类考试题目直接下载。详细请访问
www.100test.com