

文章编码：1005-3840 (2007) 03-0003-04

基于 Vxworks 操作系统的串口通信程序设计与实现

刘亚芬

(上海师范大学, 上海 200234)

摘要：结合嵌入式实时操作系统 VxWorks 实现风洞测控系统的实际应用, 以其中的串口通信为例, 讨论了在 VxWorks 下实现串口通信的关键技术, 给出了基本原理、环境配置和程序示例。

关键词：VxWorks; 实时操作系统; 嵌入式操作系统; 串行通信

1 前言

目前市场上比较著名的实时操作系统有: VxWorks、PSOS、Nucleus、QNX、VRTX、Windows CE、Palm OS、LynxOS 等。其中, VxWorks 是一个具有微内核、可裁剪的高性能实时操作系统, 在实时操作系统市场上处于领先地位。它以其良好的可靠性和卓越的实时性被广泛地应用在通信、军事、航空、航天等高精尖技术及实时性要求极高的领域中, 如卫星通讯、弹道制导、飞机导航等。在美国的 F-16 战斗机、B-2 隐形轰炸机和爱国者导弹上, 甚至火星探测器上都使用了 VxWorks 实时操作系统。在嵌入式实时操作系统中使用串口通信, 不仅可扩展嵌入式设备通信能力, 而且可扩大其应用范围。

2 VxWorks 简介

VxWorks 是由 wRS(wind River Systems, Inc.) 公司开发的一套具有微内核、高性能、可伸缩的实时操作系统, 支持广泛的网络通信协议, 并能够根据用户的需求进行组合, 其开放式的结构和对工业标准的支持使开发者只需做最少的工作即可设计出有效的适合于不同用户要求的系统。除了性能出众的操作系统之外, wRS 公司还提供了优秀的实时操作系统开发工具 Tornado。Tornado 由三个高度集成的部分组成: Tornado 工具, 一整套强有力的交叉开发工具; VxWorks 运行系统, 是运行在目标机上的高性能、可裁剪的实时操作系统; 连接目标机和宿主机的通信选项, 加以太网、串行线路、在线仿真等。Tornado 能够支持 Windows、Unix 等流行的工作平台和 PowerPC、X86、ARM 等几乎所有的目标处理器, 所提供的工具可用于所有目标机, 并具有两种调试模式(系统模式和任务模式)。除了基本的功能和开发工具外, Tornado 还具有先进的系列网络产品, 极大地扩展了 Tornado 的网络特性, 并增强了嵌入式微处理器的网络特性^[1]。

3 串口通信基本原理

串口在嵌入式系统当中是一类重要的数据通信接口, 其本质功能是作为 CPU 和串行设备间的编码转换器。当数据从 CPU 经过串行端口发送出去时, 字节数据转换为串行的位; 在接收数据时, 串行的位被转换为字节数据。应用程序要使用串口进行通信, 必须在使用

收稿日期: 2007-05-2

之前向操作系统提出资源申请要求(打开串口),通信完成后必须释放资源(关闭串口)。串口通信的优点是开发简单,在传输数据量不大、要求速度不高而传输距离较大的通信场合得到广泛应用。

在 VxWorks 中,将 I/O 系统设计成为任何类型的设备提供一个简单、统一、独立于设备的接口,任何对于串口的操作都可以视为对一个文件的操作,而不必了解串口设备或程序驱动实现的细节。在串口通信软件的设计中,当串口初始化完成后,在使用之前利用 open()打开相应串口,然后进行配置。

VxWorks 提供终端和伪终端设备驱动。Tty 驱动针对实终端;pty 针对仿真终端的程序。Tty 设备有两种操作模式:raw 模式和 line 模式。在 raw 模式下,每个刚从设备输入的字符对读者都是有效的;在 line 模式下所有输入字符被存储,直到 NEWLINE 字符输入。设备选项字使用带 FIOSETOPTIONS 功能的 ioctl()程序来设置^[2]。

配置完成后,依据串口打开时的读写标志,调用函数 write()、read()对串口进行只读操作、只写操作或同时进行读写操作。

为提高数据接收的实时性,可采用中断方式,利用 VxWorks 提供的 select 函数的事件触发机制,将读串口的任务阻塞使其一直等待数据,当有数据来到的时候该任务会立刻自动响应,提高系统的实时性。

4 环境配置

在本实时应用系统中采用 486 机作为目标机,串口通信时目标机 VxWorks 系统启动盘的制作步骤:

(1) 修改通用配置文件\\Tornado\\target\\config\\pc486\\config.h。在 config.h 文件中加入以下宏定义:

```
#define INCLUDE_WDB
#define INCLUDE_WDB_TTY_TEST
#undef WDB_COMM_TYPE
#define WDB_COMM_TYPE WDB_COMM_SERIAL /*定义通信方式为串口联结*/
#define WDB_TTY_CHANNEL 1 /*通道号*/
#define WDB_TTY_BAUD 9600 /*串口速率,可设置至 38400*/
#define WDB_TTY_DEV_NAME "tyCo/1"
#define CONSOLE_TTY 0
#define DEFAULT_BOOT_LINE "fd=0, 0 (0, 0) hostname: /fd0/vxWorks \ h=主机
ip e=目标机 ip u=主机上的登录用户名"
```

(2) 在 Tornado 集成环境中执行菜单命令 Project > Make PC486 > Common Targets > clean 删除以前生成的文件,执行菜单命令 Project > Make PC486 > Boot Rom Targets > bootrom_uncmp 编译链接生成 bootrom_uncmp ;再选择 VxWorks Target,编译生成 vxworks ;

(3) 拷贝\\Tornado\\target\\config\\pc486\\bootrom_uncmp 至\\Tornado\\host\\bin 下

(4) 重命名文件 bootrom_uncmp 为 bootrom ;

(5) 准备一张已格式化的空盘插入软驱;

(6) 在目录\\Tornado\host\bin 下执行命令 mkboot a: bootrom ;

(7)拷贝\\Tornado\target\config\pc486\VxWorks 至软盘 ;

(8) 将系统制作盘插入目标机软驱，加电启动目标机即载入 VxWorkst 系统。[3]

5 程序示例

5.1 在程序中加入下列头文件

```
#include <vxWorks.h>
#include "strLib.h"
#include <string.h>
#include <sioLib.h>
#include <ioLib.h>
#include <stdio.h>
#include <ioctl.h>
#include <selectLib.h>
#include "types/vxTypesOld.h"
```

5.2 打开串口

```
int open_com1(void) //打开串口 1 函数
{
    int sfd; //串口设备文件描述符
    sfd=open("/tyCo /0", 0_RDWR, 0); //打开串口并返回串口设备文件描述符
    if(sfd==ERROR) //如果不能打开串口 1 则打印出错信息
        printf("You can't open port com1 !");
}
```

5.3 配置串口

```
int config_com1(void) //串口 1 配置函数
{
    ioctl(sfd, FIOSETOPTIONS, OPT_LINE); //设置串口工作模式为行模式 :
    LINE_MODE
    ioctl(sfd, FIOBAUDRATE, 9600); //设置串口波特率为 9600bps
    ioctl(sfd, FIOFLUSH, 0); //清空输入输出缓冲
    ioctl(sfd, SIO_HW_OPTS_SET, CS8|STOPB|PARENB|PARODD);
    //设置 8 位数据位，2 位停止位，带校验位，奇校验
}
```

5.4 串口接收数据

```
int accept_com1(void) //从串口 1 接收数据函数
{
    while(1)
    {
        char * accept_buf;
        FD_ZERO(&fds_data); //位码置零
        FD_SET(sfd, &fds_data); //初始化位码
        width = sfd + 1;
```

```
//任务阻塞等待读串口准备完毕；  
if(select(width, &fds_data, NULL, NULL, NULL)==ERROR)  
    return(ERROR);  
read(sfd, accept_buf, sizeof(accept_buf)); //从串口读字符  
printf("accept message is : %s \n\n", accept_buf); //输出接收到的信息  
}  
}
```

5.5 串口发送数据

```
int send_com1(void) //向串口 1 发送数据函数  
{  
    char *send_buf="Data had accept!"; //待发送数据  
    //任务阻塞等待写串口准备完毕  
if(select(width, NULL, &data_fds, NULL, NULL)==ERROR)  
    return(ERROR);  
if(FD_ISSET(sfd, &fds_data)) //检查串口准备好就向串口写数据  
write(sfd, send_buf, sizeof(send_buf));  
}  
}
```

5.6 关闭串口

```
close(sfd);
```

6 结束语

RS-232 串口通信虽速率不高，但 RS-422 标准串行口通信则采用了双线传输，大大增加了抗共模干扰的能力，最大数据传输速度可以达到 10Mb/s，这对于远程数据交换具有极大的优势。

本文描述了在实时操作系统 VxWorks 中利用串口实现数据通信的方法，并给出了示例程序，目前已经在我们的风洞测控系统中获得了应用。该方法可将其串口通信可适用于工业控制等多方面，具有广泛的适应性。在嵌入式实时操作系统中使用串口通信，不仅可扩展嵌入式设备通信能力，而且可扩大其的应用范围。

【参考文献】

- [1] 孔祥营，柏桂枝. 嵌入式实时操作系统 vxworks 及其开发环境 Tornado[M]. 北京:中国电力出版社，2002.01
- [2] 陈智育，温彦军，陈琪编著 VxWorks 程序开发实践. 北京:人民邮电出版社，2004.05
- [3] 李方敏编著 VxWorks 高级程序设计. 北京:清华大学出版社，2004.05

作者简介：刘亚芬，1980 年出生，女，上海师范大学计算机应用学院，硕士研究生