文章编号:1005－0523(2005)02－0104－04

# 基于 μClinux 的 S3C4510B 的串口通讯的设计与实现

卢伟海[1],杨丰萍[1],郭家堂[2]

(华东交通大学 1.电气与电子工程学院;2.机电工程学院,江西 南昌 330013)

**摘要**:首先基于 S3C4510B 微处理器设计串口通讯电路,并介绍了 μClinux 环境下串口通讯程序的编写,最后实现基于 μClinux 的 S3C4510B 嵌入式平台与 PC 机之间的串口通讯.

**关 键 词**:μClinux;S3C4510B;串口通讯;嵌入式系统
**中图分类号**:TP311.13　　**文献标识码**:A

## 0 引 言

Linux 是一种很受欢迎的操作系统,与 unix 系统兼容,开放源代码,它可靠、安全、灵活,而且是免费的,它原本被设计为桌面系统,现在广泛的应用于服务器领域,而更大的影响在于它广泛的应用于嵌入式领域.μClinux 正是在这种氛围下产生的,是一个完全符合 GNU/GPL 公约的操作系统,完全开放源代码,现在由 Lineo 公司支持维护.μClinux 的发音"you － see － linux",它的名字来自于希腊字母"mu"和英文大写字母"C"的结合."mu"代表"微小"之意,字母"C"代表"控制器",所以从字面上就可以看出它的含义,即"微控制领域中的 Linux 系统".μClinux 的最大特点是针对无 MMU(内存管理单元)的微处理器设计的操作系统,S3C4510B 就是该类微处理器.

Samsung 公司的 S3C4510B 是基于以太网应用系统的高性价比 16/32 位 RISC 微控制器,内含一个由 ARM 公司设计的 16/32 位 ARM7TDMI RISC 处理器核,ARM7TDMI 为低功耗、高性能的 16/32 核,最适合用于对价格及功耗敏感的应用场合.

本文在硬件方面以 S3C4510B 为基础建构串口通讯电路,软件方面在 S3C4510B 微处理器上移植 μClinux 操作系统,并在 μClinux 和 S3C4510B 嵌入式平台下编写和移植串口通讯程序,然后介绍如何进行内核编译及可执行文件的装载,最后实现基于 μClinux 和 S3C4510B 嵌入式平台和 PC 机之间串口通讯.

## 1 基于 S3C4510B 串口电路的设计

串行通讯是微计算机之间一种常见的近距离通讯手段,因使用方便、编程简单而广泛使用,几乎所有的微控制器、PC 都提供串行通讯接口.通过 S3C4510B 的 UART,用 MAX232 芯片进行电平转换,即可实现与 PC 机的串口通讯.

S3C4510B 的 UART 单元提供两个独立的异步串行 I/O 口(Asynchronous Serial I/O,SIO)每个通讯口均可工作在中断模式或 DMA 模式,也即 UART 能产生内部中断请求或 DMA 请求在 CPU 和串行 I/O 口之间传送数据.

S3C4510B 的 UART 单元特性包括:

1) 波特率可编程
2) 支持红外发送与接收
3) 1～2 个停止位

**收稿日期**:2004－08－10

**作者简介**:卢伟海(1981－),男,华东交通大学硕士研究生,主要研究嵌入式系统开发及 linux 应用.

中国知网 https://www.cnki.net

4) 5、6、7 或 8 个数据位

5) 奇偶校验

RS－232C 标准是异步串行通讯中应用最广的标准总线. 该标准的全称是 EIA－RS－232C 标准(Electronic Industrial － Recommended Standard － 232C)，它是电子工业协会(EIA)推荐的，该标准对串行通讯接口的有关问题如信号线功能、电气特性等都做了明确规定. 早期它被应用于计算机和终端通过电话线和 MODEM 进行远距离的数据传输，随着微型计算机和微控制器的发展，不仅远距离，近距离也采用该通信方式. 在近距离通信系统中，不再使用电话线和 MODEM，而直接进行端到端的连接. RS－232－C 标准采用的接口是 9 芯或 25 芯的 D 型插头，以常用的 9 芯 D 型插头，各引脚定义如表 1 所示.

**表 1　9 芯 D 型插头引脚信号描述**

| 引脚 | 名称 | 功能描述 |
|---|---|---|
| 1 | DCD | 数据载波检测 |
| 2 | RXD | 数据接收 |
| 3 | TXD | 数据发送 |
| 4 | DTR | 数据终端准备好 |
| 5 | GND | 地 |
| 6 | DSR | 数据设备准备好 |
| 7 | RTS | 请求发送 |
| 8 | CTS | 清除发送 |
| 9 | RI | 振铃指示 |

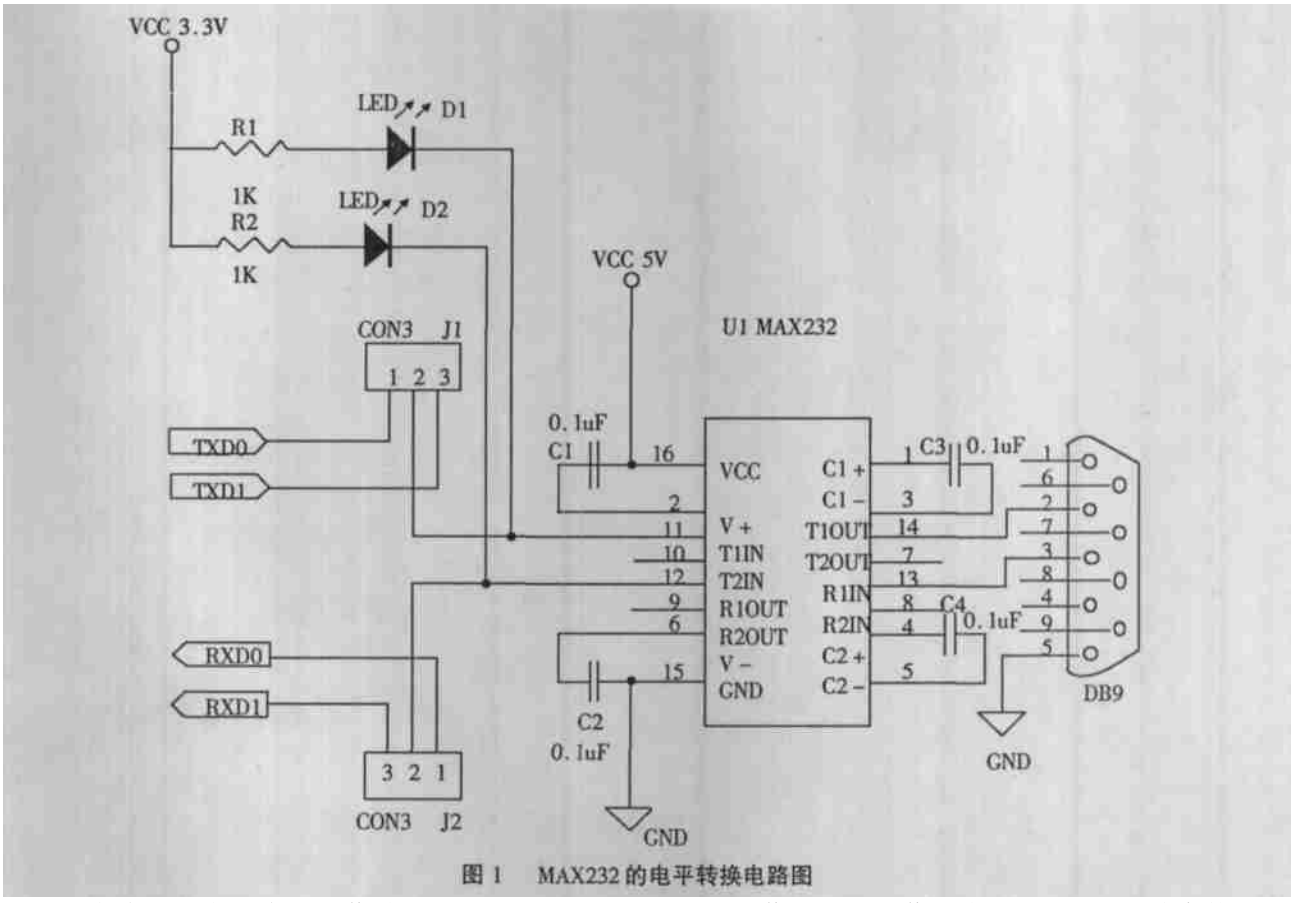


图 1　MAX232 的电平转换电路图

要完成最基本的串行通信功能，实际上只需要 RXD、TXD 和 GND 即可，但由于 RS－232－C 标准所定义的高、低电平信号与 S3C4510B 系统的 LVTTL 电路所定义的高、低电平信号完全不同，LVTTL 的标准逻辑“1”对应 2 V～3.3 V 电平，标准逻辑“0”对应 0 V～0.4 V 电平，而 RS－232－C 标准采用负逻辑方式，标准逻辑“1”对应－5 V～－15 V 电平，标准逻辑“0”对应＋5 V～＋15 V 电平. 显然，两者间要进行通信必须经过信号电平的转换，目前常使用的电平转换电路为 MAX232. MAX232 的电平转换的接线电路图如图 1 所示.

为缩小电路板面积，只设计了一个 9 芯的 D 型插头，通过两个跳线选择 S3C3410B 的 UART0 或 UART1，同时设计数据发送与接收的状态指示 LED，当有数据通过串行口传输时，LED 闪烁，便于掌握其工作状态以及进行软、硬件的调试.

中国知网　https://www.cnki.net

## 2 μClinux 环境下的串口通讯编程

Linux 操作系统从一开始就对串行口提供了很好的支持，μClinux 也很好的保持了这一特性.

### 2.1 串口操作的相关操作及函数

1) 串口操作需要的头文件

```
#include <stdio.h> /* 标准输入输出定义 */
#include <stdlib.h> /* 标准函数库定义 */
#include <unistd.h> /* Unix 标准函数定义 */
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h> /* 文件控制定义 */
#include <termios.h> /* PPSIX 终端控制定义 */
#include <errno.h> /* 错误号定义 */
```

2) 打开串口

在 μClinux 下串口文件是位于/dev 下的. 串口一为 /dev/ttyS0，串口二为/dev/ttyS1(需要注意的是本设计中串口一被 minicom 命令占用，所以串口通讯通过串口二实现). 打开串口是通过使用标准的文件打开函数操作：

```
int fd; /* 以读写方式打开串口 */
fd = open("/dev/ttyS0", O_RDWR);
if (-1 == fd){ /* 不能打开串口一 */
perror("提示错误!");}
```

3) 设置串口

最基本的设置串口包括波特率设置，效验位和停止位设置. 串口的设置主要是设置 struct termios 结构体的各成员值.

```
struct termios
{ unsigned short c_iflag; /* 输入模式标志 */
unsigned short c_oflag; /* 输出模式标志 */
unsigned short c_cflag; /* 控制模式标志 */
unsigned short c_lflag; /* local mode flags */
unsigned char c_line; /* line discipline */
unsigned char c_cc[NCC]; /* control characters */
};
```

波特率设置代码如下.

```
struct termios Opt;
tcgetattr(fd, &Opt); /* 保存串口当前设置 */
cfsetispeed(&Opt, B19200); /* 设置为 19200Bps */
cfsetospeed(&Opt, B19200);
tcsetattr(fd, TCANOW, &Opt);
```

设置效验的函数：

设置效验由函数 set_Parity(int fd, int databits, int stopbits, int parity)完成，其中 databits 设置数据位参数，取值可为 7 或 8；stopbits 停止位参数，取值可以为 1 或 2；Parity 为效验类型参数，可以为 N, E, O, S，分别代表使能校验，奇校验，偶校验，无校验.

4) 读写串口

设置好串口之后，读写串口就很容易了，把串口当作文件读写就是.

发送数据：

```
char buffer[1024];
int Length, nByte;
nByte = write(fd, buffer, Length);
```

读取串口数据：使用文件操作 read 函数读取，如果设置为原始模式(Raw Mode)传输数据，那么 read 函数返回的字符数是实际串口收到的字符数. 可以使用操作文件的函数来实现异步读取，如 fcntl，或者 select 等来操作.

```
char buff[1024];
int Len;
int readByte = read(fd, buff, Len);
```

5) 关闭串口

关闭串口就是关闭文件. 即 close(fd);

### 2.2 串口通讯流程图

在 μClinux 环境下实现 PC 机与嵌入式平台的串口通讯程序的流程图，如图 2 所示.

## 3 μClinux 环境下 S3C4510B 的串口通讯的实现

因为没有足够的资源在嵌入式平台运行开发工具和调试工具，所以通常的嵌入式系统的软件开发采用一种交叉编译调试的方式. 交叉编译调试环境建立在宿主机(即一台 PC 机)上，对应的嵌入式平台叫做目标板. 开发时使用宿主机上的交叉编译、汇编及连接工具形成可执行的二进制代码，然后把可执行文件下载到目标板上运行. 所以先将串口通讯源程序移植到宿主机的 μClinux 下，然后进行编译，最后下载到目标板进行调试，其具体步骤介绍如下：

中国知网 https://www.cnki.net

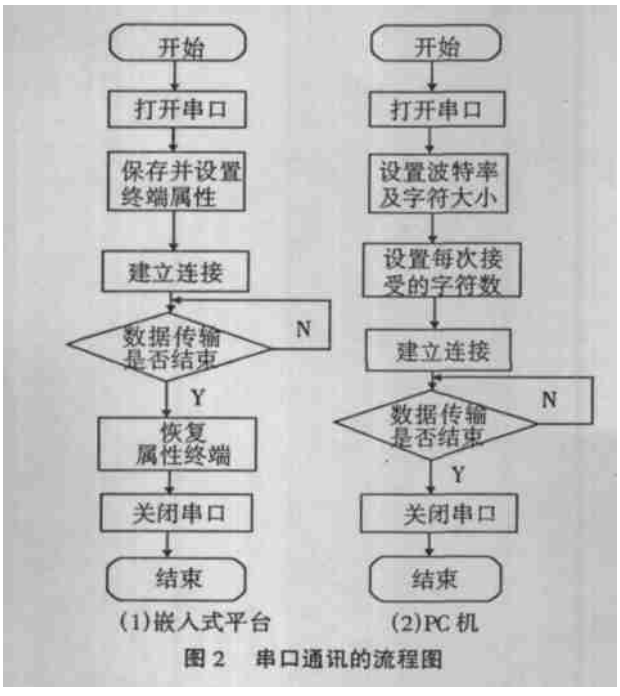

图 2　串口通讯的流程图

### 3.1　串口通讯程序的移植

1) 在 μClinux/user 目录下创建目录 serial.

2) 将应用程序目录 serial 加入编译队列，即修改 user 下的 Makefile.

3) 将应用程序 serial.c 拷贝至目录 serial 下，并将该源文件的 Makefile 也一起拷贝到目录 serial.

4) 修改 μClinux/config 目录下的 config.in 文件.

### 3.2　编译 μClinux 内核

1) 键入命令：make menuconfig，进行内核配置.

2) 键入命令：make dep，该命令用于寻找依存关系.

3) 键入命令：make clean，该命令清除以前构造内核时生成的所有目标文件，模块文件和一些临时文件.

4) 键入命令：make lib _only，该命令编译库文件.

5) 键入命令：make user _only，该命令编译用户应用程序文件.

6) 键入命令：make romfs，该命令生成 romfs 文件系统.

7) 键入命令：make image.

注意第一次做到这一步的时候会出现错误的信息提示.

8) 键入命令：make，通过各个目录的 Makefile 文件进行，会在各目录下生成一大堆目标文件.

上述步骤完成后，就完成了对 μClinux 源码的编译工作.

### 3.3　串口通讯调试

将 PC 机和嵌入式平台通过串口相连，在 PC 机上运行 minicom(它占用了串口 1)使 PC 机的显示器作为嵌入式平台的显示终端.运用 mount 或通过网络下载或烧写相应的可执行文件到嵌入式平台，最后可进行程序调试.

## 4　结束语

本文详细的讲述了在 μClinux 和 S3C4510B 嵌入式平台下串口通讯的要点，对在嵌入式设备上实现串口通讯有着重要的参考意义，本文所述的开发流程对嵌入式系统(特别是基于 μClinux 操作系统)的开发有着很好借鉴的作用.

**参考文献：**

[1] Michael R. Sweet Serial Programming Guide for POSIX Operating Systems.

[2] 天津市计算机学会单片机分会.2003 年全国单片机及嵌入式系统学术年会论文集[C]，北京航空航天大学出版社.

[3] 李驹光，聂雪媛，等. ARM 应用系统开发详解——基于 S3C4510B 的系统设计[M].北京：清华大学出版社.

[4] Loujia. μClinux 上的应用程序设计.

# Design and Implementation of Serial Communication of S3C4510B Based on μClinux

**LU Wei-hai，YANG Feng-ping，GUO Jia-tang**

(School of Electrical and Electronic Eng.，East China Jiaotong Univ.，Nanchang 330013，China)

**Abstract**：At first，the Serial Communication circuit based on S3C4510B microprocessor is designed in this paper. Then the Serial Communication program under μClinux is discussed. Finally，the Serial Communication between the S3C4510B Embedded platform based on μClinux and personal computer is realized.

**Key words**：μClinux；S3C4510B；serial communication；embedded system

中国知网　https://www.cnki.net