

基于 FPGA 的串行接收模块的设计

The Design of Serial Receiver Model Based on FPGA

(1.中科院长春光学精密机械与物理研究所;2.中国科学院研究生院) 龚睿^{1,2} 王萍^{1,2}

GONG Rui WANG Ping

摘要: 为了使计算机能够通过串口控制 FPGA 的输出信号,笔者根据异步串行通信的原理,设计了简便易行的 FPGA 串行通信接口系统,并应用 VHDL 语言在 FPGA 内部集成了串行接收模块,具有较强的通用性和推广价值。

关键词: FPGA; RS232; VHDL; 串行接收

中图分类号: TN91; TP710 **文献标识码:** A

Abstract: In order to control the FPGA's output through serial port, According to the principles of asynchronous series communication, the author designs a simple and convenient system of FPGA's series communication interface, and also integrates a model in FPGA for receiving series data with VHDL language, which can be more useful and have more spread value.

Key words: FPGA; RS232; VHDL; series receiving

1 前言

随着 FPGA 的飞速发展与其在现代电子设计中的广泛应用,越来越多的实验和设计会运用 FPGA 与 RS232 通信。与此同时, FPGA 具有功能强大、开发过程投资小、周期短、可反复编程等特点。笔者在 FPGA 芯片上集成了串行接收功能模块,从而简化了电路、缩小了电路板的体积、提高了可靠性。本文主要介绍围绕 FPGA 所设计的符合 RS232 标准的串行接收模块。

2 异步串行通信原理

串行通信分为两种类型:同步通信方式和异步通信方式。本设计采用的是异步通信方式,其的特点是:通信的发送方和接收方各自有独立的时钟,传输的速率由双方约定。国际上规定的一个串行通信波特率标准系列是:110、300、600、1200、1800、2400、4800、9600、19200,单位是 bps。本文采用的是 19200bps。

异步传输是一个字符接一个字符传输。一个字符的信息由起始位、数据位、奇偶校验位和停止位组成。每一个字符的传送靠起始位来同步,字符的前面是一位起始位,用下降沿通知接收方传输开始,紧跟着起始位之后的是数据位,传输时低位在前、高位在后,字符本身由 5~8 位数据位组成。数据位后面是奇偶校验位,最后是停止位,停止位是高电平,标志一个字符的结束,并为下一个字符的开始传送做准备。停止位后面是不定长度的空闲位。停止位和空闲位都规定高电平,这样可以保证起始位开始处有一个下降沿,如图 1 所示。

3 硬件接口电路原理设计

在串行通信中,普遍采用的是 RS232-C 接口的标准。RS232-C 接口信号引脚的连接方式规定了 25 芯的 D 型连接器 DB-25,本设计采用的是一个 9 芯的 D 型连接器 DB-9,并且用最为简单常用的三线制接法,即地、接收数据和发送数据三脚相连。

龚睿:硕士研究生

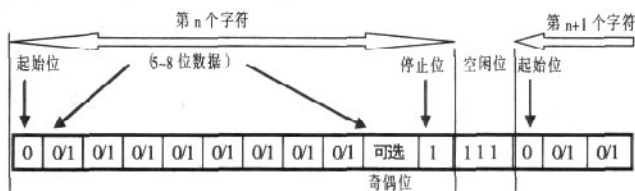


图 1 RS232 异步通信的字符格式

本设计的硬件接口电路图如图 2 所示,由三部分组成: FPGA 串口接收模块、MAX232 和 DB-9。FPGA 采用的是 Xilinx 公司的 SPARTAN 系列的 XC2S60 芯片,其封装为 TQ144。MAXIM 公司的 MAX232CPE 是为满足 EIA/TEA-232E 的标准而设计的,具有功耗低、波特率高、价格低等优点。工作电源为 +5V,外界电容仅为 1μF,为双组 RS-232 收发器。MAX232 有两个发送器,本设计只用其中一个发送器,另外一个发送器的输入端接地、输出端悬空。

异步数据接收过程可作为一个整体来实现,数据由 DB-9 的 RxD 端输入,经过 MAX232 进行电平转换由 FPGA 串口接收模块的 RxD 端进入,然后在串口接收模块内部对接收来得数据进行判断,并最终实现对 FPGA 输出信号的控制。

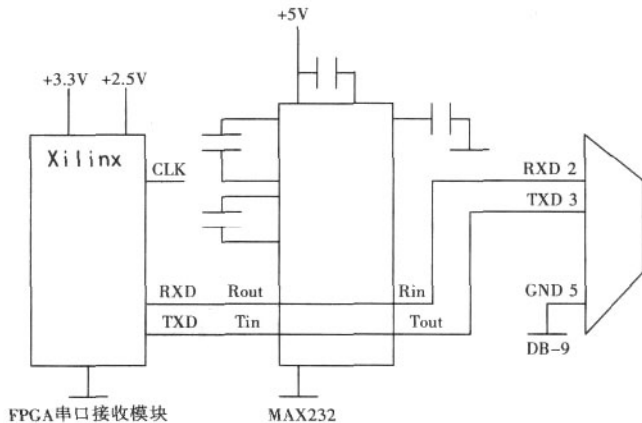


图 2 FPGA 串行通信硬件接口电路图

4 FPGA 接收串行数据的软件设计

本方案采用的串行异步通信的帧格式为: 1 位起始位+5 位数据位+1 位停止位。经检测与分析, 如果已经确定异步通信的帧格式, 那么每个字符就可以分别用固定的 7 位数据表示。比如: 字符 '0': '0000011'; 字符 '1': '0100011'。其中第 1 位数据 '0' 为起始位, 最后 1 位数据 '1' 为停止位, 中间 5 位数据为数据位。例如: 用串口调试助手软件发送字符串 '100', 则会在 FPGA 的 RxD 引脚测得如图 3 的波形。其中, st1, st2, st3 表示先后发送的 3 个字符, t1, t2, t3 则表示对每个字符进行检测时所经过的 3 种状态。

正如图 3 波形所示, 接收逻辑首先通过检测输入数据的下降沿来检查起始位。然后产生接收时钟, 利用接收时钟来采样串行输入数据。由于字符 '0' 和 '1' 在 5 位数据位中的仅第一位有区别, 因此只要准确地将第一位数据检测出来就可以得出串口调试助手所发送的字符串。再利用移位操作, 将字符串存储在缓存器(即另一组可以更新的字符串)中。至于剩下的 4 位数据位和一位停止位, 就可以不随接收时钟采样。待串口调试助手发送的字符串全部存储之后, 一起在该模块中进行判断, 并根据字符串的不同来控制 FPGA 不同的输出。接收时钟是根据数据传输的波特率产生的: 接收时钟=16×19200Hz。它始于起始位的下降沿, 终于第 5 位数据位的上升沿。下面是实现检测第 1 个字符的 VHDL 源程序。

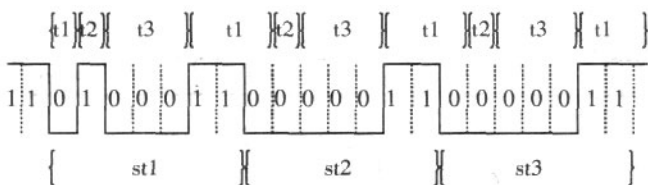


图 3 FPGA 接收的串行数据波形

```

if(clk0'event and clk0='1')then      ---- 外部时钟
case state is
when st1=>                          ---- 第 1 个字符开始
case tate is
when t1=>                            ---- 起始位开始
if(rxd='1')then                      ---- 未到下降沿不计数
cnt16 <= "0000";
cnt48 <= "000000";                  ---- 所有时钟清零
else cnt16 <=cnt16+1;              ---- 下降沿来, 起始位计数开始
end if;
if(cnt16="11111")then
tate <= t2;                          ---- 起始位完, 进入数据位
end if;
when t2=>                            ---- 进入第 1 位数据位
if(cnt16="11111")then
cnt16 <= "0000";
else cnt16 <= cnt16+1;
end if;
if(cnt16="00111")then
sdata <= rxd;                        ---- 采样第 1 位数据位
end if;
if(cnt16="01111")then
if(sdata='0')then

```

```

data0)<='0';
else data0)<='1';---- 判断采样值, 如果为 0, 则发送字符为
0, 反之亦然
end if;
end if;
if(cnt16="11111")then
tate <= t3;                          ---- 第 1 位数据位完, 进入下一状态
end if;
when t3=>                            ---- 进入第 2 位, 第 3 位和第 4 位数据状态
if(cnt16="11111")then
cnt48 <= "000000";
else cnt48 <= cnt48+1;              ---- 不对剩下的数据采样, 直接计数
end if;
if(cnt48="101111")then
sdata <='0';                        ---- 采样位清零
tate <= t1;                          ---- 进入采集下一字符的准备状态
state <= st2;                        ---- 第 1 个字符采集完, 进入下一字符
end if;
end case;
fdata0) <= data0);                  ---- 采集完的字符存入缓存区
.....

```

对每个字符的采集过程是相同的。笔者根据实际需要只让计算机发送 3 个二进制字符, 由此能控制 FPGA 的 8 种输出状态。在整个 VHDL 源程序编写完之后, 用 Modelsim 6.0 进行仿真, 如图 4。

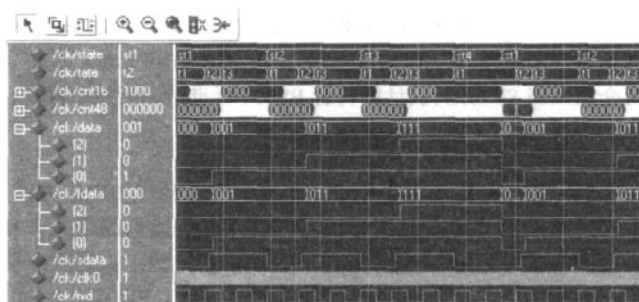


图 4 FPGA 中的接收模块 Modelsim 仿真图

鉴于发送字符与其异步传输帧格式的特殊关系, 此仿真中的 rxd 用周期为 16×clk0 的波形代替, 因此就会接收到字符串 '111'。串口调试助手需要每隔一段时间(大于 1ms)发送一组字符串, 则等待状态 st4 是必须的, 它也是接收下一字符的准备状态。缓存区为 3 位空数组 fdata, 它在接收下一字符之前必须清零。

5 结语

尽管目前串行通信速度慢的特点已经越来越明显, 但是因其传输线少且成本低, 多数电子产品开发中都会继续使用串行通信。本文所介绍的串行接收模块是笔者根据实验要求自行设计, 并在实验中成功地完成了计算机对 FPGA 的准确控制。因此, 本文也可作为一个实例, 供开发者交流。

本文作者创新点: 计算机串口按 ASCII 码发送 '0' 和 '1' 组成的字符串, 在 FPGA 内部只需判断接收到的数据流中的每个起始位之后的第一位数据位, 就可最终得出计算机发送的字符串。此种方法使 FPGA 能够简便、准确的识别串口数据, 且可以方便地修改为任何一种异步通信帧格式。(下转第 73 页)

另一个设计问题是选择什么样的仲裁算法,有多种方法可供选择,包括静态优先权、随机法、轮转法和最老优先权法。每种方法都有不同的性能特征和实现复杂度。静态优先权法实现最简单,只需要用一个简单的优先权编码器,然而在一个很大的网络里,可能会造成不确定的延迟。一般来说,给每个输入端口提供公平服务的调度算法性能会更好。轮转法需要一个额外的位,在每个时钟周期改变优先级的次序。最老优先法虽然与随机法有同样的平均延迟,但延迟变化比随机法要小一些。我们采用的一种实现最老优先法的办法是,在每个输出端口上设置一个输入端口号控制 FIFO 队列。当一个输入缓冲区请求获得一个输出端口进行传输时,就将一个请求放入输入控制 FIFO 队列中,在 FIFO 队列中最老的请求被授予输出端口。

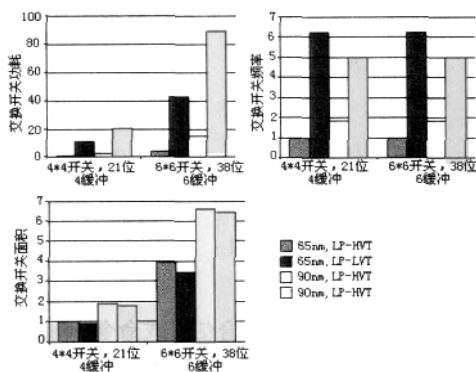


图 5 xpipes 开关在不同技术库中的分析

我们分别将 xpipes 交换开关在 4 个不同的库中进行不同的功耗/运行折衷(LP-LVT 和 LP-HVT)分析,两个用 65 纳米工艺,另两个用 90 纳米工艺。在分析中,开关被充分的布局和路由,而且增加了一个时钟树。

从结果(图 5)中可以看出,65 纳米库提供了更大的改进空间。事实上我们发现采用 65 纳米库,功耗降低了 50%,面积节省了 40%~50%。同样重要的是,在相同的技术节点处两种不同库中所得出的结果有很大的不同。在 65 纳米中,LP-HVT 库比 LP-LVT 库有更少的功耗。而且,我们的结果表明和 90 纳米库相比其操作空间增加了。例如,通过观察可行的时钟频率,LP-HVT65 纳米库可以达到的频率比 90 纳米低 50%,但是 LP-LVT65 纳米库实际上比 90 纳米库快 25%。这一趋势说明在新的技术节点中设计者可以有新的自由度。

3 结论

目前的应用和技术趋势推动了从总线结构向片上网络的转变。片上网络不仅用于解决物理设计的难题,而且提高了带宽需求。然而,片上网络的设计性能和折衷在 100 纳米以下的研究比较少。本文分析了片上网络的一个重要组成部分——交换开关。xpipes 是一个软模块库,我们将其运用于交换开关的设计。并且将其在 65 纳米和 90 纳米两种前提下进行综合分析,实验证明,65 纳米技术下的交换开关的性能更好,更适于目前低功耗的设计趋势。本文作者创新点:提出了一种新的交换开关结构,并将其在 65 纳米及 90 纳米库中进行综合分析,得出虽然 65 纳米工艺有一定的缺点但总体而言还是有很大研究价值和空间。

参考文献

[1]ITRS. International Technology Roadmap for Semiconductors

[EB/OL]. <http://public.itrs.net>. 2003.

[2]李耀荣,王兴军,梁利平. SOC 总线仲裁算法的研究. 微计算机信息, 2007, 6-2: 113-115

[3]Shashi Kumar, Axel Jantsch, et al. A Network on Chip Architecture and Design Methodology. IEEE 2002.

[4]陈国良,吴俊敏,章锋,章隆兵编著. 并行计算机体系结构. 北京:高等教育出版社, 2002

[5]J. Duato, S. Yalamanchili, and L. Ni, interconnection networks: Engineering Approach, IEEE Computer Society Press, 1997.

[6]Seitz C L, Su W K. A family of routing and communication chips based on mosaic. Proc Of Univ. Of Washington Symposium on integrated Systems. Cambridge, MA: MIT Press, 2003, 320~337.

[7]Karol M, Hluchyj M, Morgan S. Input versus output queuing on a space division packet Switch. IEEE Transactions on Communications, 2001, 35(12): 1347~1356.

作者简介: 王晓袁(1982-) 女(汉族), 辽宁省丹东人, 西安电子科技大学硕士研究生, 主要研究方向为 NOC 系统模型; 杨银堂(1962-) 男(汉族), 河北省邯郸人, 博士, 教授, 博士生导师。

Biography: WANG Xiao-yuan (1982-), female (the Han nationality), Liaoning Province, Xidian University, master, studying of the model on system-level of network-on-chips.

(710071 陕西西安 西安电子科技大学) 王晓袁

通讯地址: (710071 陕西西安 西安电子科技大学 西安 陕西) 王晓袁

(收稿日期: 2008.07.23)(修稿日期: 2008.09.05)

(上接第 138 页)

参考文献

[1]潘松, 黄继业. EDA 技术实用教程[M]. 北京: 科学技术出版社, 2002

[2]杜晓斌, 陈兴文. FPGA 和单片机串行通信接口的实现[J]. 微计算机信息, 2004, 9: 71-72

[3]聂涛, 许世宏. 基于 FPGA 的 UART 设计[J]. 现代电子技术, 2006, 2: 127-129

[4]陈炳权. 基于 FPGA 器件的 RS232-C 接口设计及其扩展[J]. 攀枝花学院学报, 2006, 10: 106-109

[5]刘国勇. MAX232/MAX232A 收发器及其应用[J]. 国外电子元器件, 1997, 1

作者简介: 龚睿(1982-) 男(汉族) 山西孟县人 中国科学院长春光机所硕士研究生 从事成像快视方面的研究。

Biography: GONG Rui (1982-), male (han), study in Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, profession is "Fast-View" imaging.

(130033 吉林 长春 中科院长春光学精密机械与物理研究所) 龚睿 王萍

(100039 北京 中国科学院研究生院) 龚睿 王萍

(Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China) GONG Rui WANG Ping

(Graduate School of the Chinese Academy of Sciences, Beijing 100039, China) GONG Rui WANG Ping

通讯地址: (130033 长春市经济技术开发区营口路 20 号研究生部 C 座 106) 龚睿

(收稿日期: 2008.07.23)(修稿日期: 2008.09.05)