

利用 FPGA 实现 UART 的设计

孟志华

(武汉科技大学信息科学与工程学院, 湖北省武汉市 430081)

摘 要: 采用 VHDL 语言作为硬件功能的描述, 硬件采用 Altera 公司的 EP1K30TC144-3 芯片, 运用模块化设计方法分别设计了 UART (通用异步收发器) 的发送器、接收器和波特率发生器。在 Max-plusII 环境下进行设计、编译和仿真, 并结合 FPGA (现场可编程门阵列) 的特点, 实现了一个可编程的 UART 模块。上位机利用 VB6.0 编程实现 PC 机与 UART 的通信。

关键词: UART; FPGA; 串行通信; MSCOMM32 OCX 控件; VHDL

中图分类号: TN919.3

0 引 言

随着计算机技术的发展和广泛应用, 尤其是在工业控制领域的应用越来越广泛, 计算机通信显得尤为重要。串行通信虽然使设备之间的连线大为减少, 但随之带来串/并转换和位计数等问题, 这使串行通信技术比并行通信技术更为复杂。串/并转换可用软件实现, 也可用硬件实现。用软件实现串行传送大多采用循环移位指令将一个字节由高位到低位 (或低位到高位) 一位一位依次传送, 这种方法虽然简单但速度慢, 而且大量占用 CPU 的时间, 影响系统的性能。更为方便的实现方法是用硬件, 目前微处理器串行接口常用的 LS 芯片是 UART (通用异步收发器)、USART (通用同步异步收发器) 和 ACIA (异步通信接口适配器) 等。不论是哪一种芯片, 它们的一种基本功能是实现串/并转换。正是这些串行接口芯片弥补了串行通信较为复杂这一缺陷。本文应用 EDA (电子设计自动化) 技术, 基于 FPGA (现场可编程门阵列)/CPLD (复杂可编程逻辑器件) 设计与实现 UART。

1 总体设计

整个设计包括两部分: 基于 FPGA 的 UART 的设计和基于 VB6.0 的上位机程序设计。UART 的设计采用模块化的设计思想, 可分为 3 个模块: FPGA 数据发送模块、FPGA 波特率发生器控制模块及数据接收模块。上位机程序采用 VB 6.0 的 MSCOMM 控件, 可分为画面设计和功能设计两部分。串口采用标准的 RS-232 协议, 主要参数选择为: 波特率 9 600 bit/s, 8 位有效位, 无奇偶校验位, 1 位停止位^[1]。

2 UART 的结构和帧格式

UART 主要包括接收器和发送器。从异步接收输入信号 SN 接收到的异步信号通过接收器完成串行/并行的转换, 形成异步数据帧; 发送器将 CPU 发出的 8 位数据进行并行/串行转换, 从 SOUT 发送出去。功能包括微处理器接口、TBR (发送缓冲器)、TSR (发送移位寄存器)、帧产生、并转串、RBR (接收缓冲器)、RSR (接收移位寄存器)、帧产生、串转并。UART 的结构如图 1 所示。

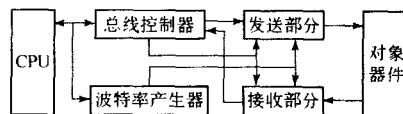


图 1 UART 的结构

UART 的帧格式包括线路空闲状态 (idle, 高电平)、起始位 (start bit, 低电平)、5 位 ~ 8 位数据位 (data bits)、校验位 (parity bit, 可选) 和停止位 (stop bit, 位数可为 1、1.5、2 位)。这种格式是由起始位和停止位来实现字符的同步。UART 内部一般有配置寄存器, 可以配置数据位数 (5 位 ~ 8 位)、是否有校验位和校验的类型、停止位的位数 (1、1.5、2 位) 等设置。

3 UART 的设计与实现

UART 是广泛使用的串行数据传输协议。UART 允许在串行链路上进行全双工的通信。串行外设使用 RS-232-C 异步串行接口, 一般采用专用集成电路即 UART 实现。如 8250、8251、NS16450 等芯片都是常见的 UART 器件, 这类芯片已经相当复杂, 有的含有许多辅助的模块 (如 FIFO), 有时不需要使用完整的 UART 的功能和这些辅助功能, 或者使用了 FPGA/CPLD, 那么就可以将所需要的 UART 功能集成到 FPGA 内部。

收稿日期: 2006-10-08; 修回日期: 2007-03-20。

使用 VHDL 将 UART 的核心功能集成,从而使整个设计更加紧凑、稳定且可靠。

下面分别设计 UART 的 3 个模块(发送器、接收器和波特率产生器),并给出其仿真结果。

3.1 发送器设计

UART 串行发送器模块框图如图 2 所示。DN 为 8 位数据,其余为 1 位。



图 2 UART 串行发送器模块框图

从图 2 的框图可以看出,串行发送器中包含有一个 8 位 THR(发送保持寄存器)和 TSR(发送移位寄存器)。复位时,引脚 TRE 为高电平。当数据载入到 TSR 之后,引脚 TRE 变为低电平。发送完毕, TRE 变为高电平。当检测到输入 WRN 变为低电平时,自动使能串行数据发送过程。首先传送 1 位的起始位(逻辑电平 0),同时 THR 中的数据自动地并行载入到 TSR 中。然后,定长的数据位从 TSR 中移出,接着是校验位。最后,产生停止位(逻辑电平 1),标志着一帧的结束。串行数据帧将以内部时钟频率的 1/16 传送。如果 THR 中内容不空,当一个串行数据帧传送结束后,紧接着发送下一个数据帧。这种自动的流程使得数据帧以背靠背的方式发送,提高了数据传输的带宽。当没有数据发送时,SDO 引脚保持高电平。

发送器每隔 16 个时钟周期输出 1 位,顺序遵循 1 位起始位、8 位数据位(假定数据位为 8 位)、1 位校验位(可选)、1 位停止位。引入发送字符长度和发送次序计数器 no_bits_sent,实现的部分 VHDL 程序如下:

```

.....
elsif clk1x 'event and clk1x = '1' then
if std_logic_vector(no_bits_sent) = "0001" then
tsr <= tbr; //TBR 数据进入 TSR
tre <= '0'; //TSR 空标志置“0”
elsif std_logic_vector(no_bits_sent) = "0010" then
sdo <= '0'; //发送起始位信号“0”
elsif std_logic_vector(no_bits_sent) >= "0011" and std_logic_
vector(no_bits_sent) <= "1010" then
tsr <= tsr(6 downto 0) & '0';
sdo <= tsr(7); //从低位到高位进行移位输出至串行输出
端
elsif std_logic_vector(no_bits_sent) = "1011" then
sdo <= '1'; //停止位输出
tre <= '1'; //发送完毕标志置“1”
end if;
.....

```

发送器功能仿真结果如图 3 所示。并行输入 DN 十六进制数 56, WRN 输入由 1 变为 0, 启动发送程序, 计数器开始计数, 串行输出 SDO 为 0010101101, 发送完毕, TRE 变为高电平。起始位 0, 8 位数据位, 1 位停止位, 证明了发送模块的正确性。

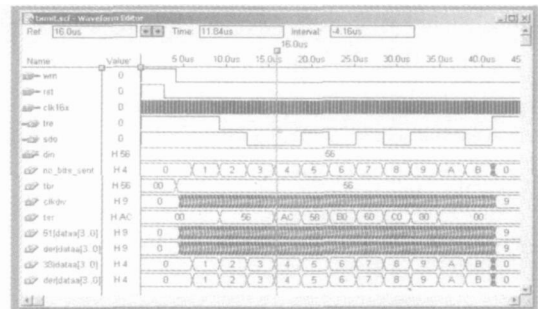


图 3 发送器功能仿真结果

3.2 接收器设计

UART 串行接收器模块框图如图 4 所示。DOUT 为 8 位数据, 其余为 1 位。接收器包含一个 8 位 RBR 和 RSR。RBR 的状态可以通过引脚 DATA_READY 来表示。当 RBR 中的数据有效时, DATA_READY 变为高电平, 向 CPU 表明可以取回数据。



图 4 UART 串行接收器模块框图

本设计只要求实现简单的收发功能, 故未设计检错程序, 程序在检测到起始位后, 计 16 个时钟周期, 便开始接收数据, 移位输入 RSR, 最后输出数据 DOUT。还要输出一个数据接收标志信号标志数据接收完。实现的部分 VHDL 程序如下:

```

.....
elsif clk1x 'event and clk1x = '1' then
if std_logic_vector(no_bits_rcvd) >= "0001" and std_logic_
vector(no_bits_rcvd) <= "1000" then
rsr(0) <= rxd; //数据帧数据由接收串行数据端移位入
RSR
rsr(7 downto 1) <= rsr(6 downto 0);
elsif std_logic_vector(no_bits_rcvd) = "1001" then
rbr <= rsr; //RSR 数据进入 RBR
.....

```

接受器功能仿真结果图略。串行输入 RXD 为 0010101101, 每一位占 16 个时钟周期, 一旦检测到输入 RXD 为 0, 计数器开始计数, 开始接收数据, 接收完毕, 标志位变为高电平。仿真结果证明了接收模块的正确性。

3.3 波特率发生器的设计

UART的接收和发送是按照相同的波特率进行收发。波特率发生器产生的时钟频率不是波特率时钟频率,而是波特率时钟频率的16倍,目的是在接收时进行精确地采样,以提出异步的串行数据^[1]。根据给定的晶振时钟和要求的波特率算出波特率分频数。实现的部分VHDL程序如下:

```
.....
elsif (clk_event and clk = '1') then
  if (clk_div = clk_div_coef) then
    clk_div <= (others => '0');
    clk16x_t <= not clk16x_t; /取反
  else clk_div <= clk_div + 1; /除数加 1
end if;
end if;
end process;
b_clk <= clk16x_t; /输出波特率
.....
```

波特率功能仿真结果图略。输入频率为20 MHz,波形周期为50 ns, $20\text{ MHz}/(9\ 600\text{ bit/s} \times 16\text{ bit}) = 130$,由仿真结果可知输出波形的半个周期为65倍的输入时钟周期,从而证明了波特率产生器模块的正确性。

4 上位机程序设计

本文使用VB 6.0进行上位机程序的设计,实现PC与FPGA的串行通信。下面是1个上位机收发测试通信程序的设计过程,通过该程序可以与FPGA进行串行通信。波特率默认值是“9600,N,8,1”,其意为所使用的通信端口是以9600 bit/s的速度传输,不作字符校验,每次的数据是8位,而停止位是1位。波特率(单位为bit/s)可为110、300、600、1200、2400、9600、14400、19200、28800。校验位为:E偶校验,N无校验,O奇校验,S空白。正确的数据位值有:4、5、6、7、8(默认值)。正确的停止位值有:1(默认值)、

1.5、2^[1]。

将UART的程序编译、仿真后,下载到FPGA的EP1K30TC144-3芯片上。引入20 MHz的晶振频率;发送使能端和复位端分别接一个开关;状态输出标志TRE和DATA-READY分别接一个二极管,指示状态;设置波特率为“9600,N,8,1”。串行数据帧的格式为:起始位0,8位数据位,无校验位,1位停止位。将UART的串行发送、接收端口分别与计算机的RS-232的串行接收、发送端口连接,以便与PC机进行串行通信;并行输入DIN接入并行输出DOUT;连好线后,执行发送测试程序。

5 结束语

在实现FPGA与PC的串行通信中,将程序下载到芯片中验证设计的正确性,目前还没有更好的工具可以在下载后实时地对FPGA的工作情况和数据进行分析。通过串行通信,可以向FPGA发控制命令让其执行相应的操作,同时把需要的数据通过串口发到PC上进行相应的数据处理和分析,以此来判断FPGA是否按设计要求工作。本文以UART为重点讨论了FPGA与上位机串行通信的实现方法。采用高级语言VB实现了上位机与FPGA的通信。

参 考 文 献

- [1] 范逸之,陈立元,等. 利用Visual Basic实现串并行通信技术[M]. 北京:清华大学出版社,2002
- [2] 褚振勇,瓮木云. FPGA的设计与应用[M]. 西安:西安电子科技大学出版社,2002
- [3] MEYER-BAESE U. 数字信号处理的FPGA实现[M]. 刘凌,胡永生,译. 北京:清华大学出版社,2003
- [4] 朱明程. FPGA原理及应用设计[M]. 北京:电子工业出版社,2001.

孟志华(1977-),男,实验员,主要研究方向为数字信号处理。

Design and Implementation of UART Based on FPGA

MENG Zhíhua

(Wuhan University of Science and Technology, Wuhan 430081, China)

Abstract: This paper adopts the VHDL language to be the hardware function description method. The hardware adopts the chips EP1K30TC144-3 of Altera company, uses the modularity method to design the transmitter module, receiver module and baud rate generator respectively. It designs, compiles and simulates the UART in MaxplusII environment. It implements a model of programmable UART by using the Field Programmable Gate Array (FPGA). The system implements the serial communication of the PC and UART by VB6.0 program.

Keywords: UART; FPGA; serial communication; MSCOMM32 OCX control; VHDL