



基于FPGA的异步FIFO硬件实现

王宏臣, 林咏海

(淮安信息职业技术学院, 江苏 淮安 223003)

摘要: 使用FPGA内部资源BlockRam实现异步FIFO, 因为未使用外挂FIFO, 使得板卡设计结构简单并减少了硬件板卡的干扰, 给硬件调试工作带来了方便, 也充分体现了FPGA的优势, 这种方法对设计异步FIFO的使用具有很好的借鉴意义。实验通过VERILOG编程实现异步FIFO, 对程序进行了功能仿真、时序仿真, 并下载到FPGA芯片中进行了硬件仿真, 实验结果达到了预期的参数要求, 完成了FIFO软硬件设计。

关键词: FPGA; 异步FIFO; VERILOG; TS流数据

中图分类号: TP302.1 **文献标识码:** A **文章编号:** 1681-1070 (2006) 12-0034-03

Asynchronous FIFO Hardware Design Based on FPGA

WANG Hong-chen, LIN Yong-hai

(Huai'an College of Information Technology, Huai'an 223003, China)

Abstract: Usage of the internal resources of FPGA realizes asynchronism FIFO, because of not using the exterior FIFO, it simplified the design construction, decrease the interference and brought the convenience to adjusting of plank card, and also well incarnate advantage of FPGA. This kind of method has great meaning to the usage of asynchronism FIFO. The experiment realizes asynchronism FIFO with VERILOG program, the result come to an anticipant request of parameter, completing the design of software and hardware in FIFO.

Key words: FPGA; asynchronism FIFO; VERILOG; TS data

1 前言

在电子设计中, 由于现场可编程门阵列(FPGA)的高逻辑密度和高可靠性以及用户可编程性, 受到了广大硬件工程师的青睐。用FPGA来实现某些专用电路, 可使整个设计更加紧凑、更小巧、灵活、稳定、可靠。本文提出了一种用Xilinx公司的FPGA芯片实现异步FIFO的设计方案, 重点强调了设计有效、可靠的握手信号EMPTY与FULL的方法, 并给出了其VERILOG语言实现的仿真图。

该异步FIFO用XILINX提供的FPGA芯片来实现, 并占用内部的BLOCKRAM资源, XC2V1000内部具有

48kbit RAM, 根据设计要求和需要设定FIFO的参数: 根据总线宽度的要求, 设定数据线32位宽度; 根据突发长度为256并实现半空控制的要求, 设定FIFO深度为512。因此只占用了13%的BLOCKRAM资源。下面论述一下FIFO模块FPGA的硬件实现。

2 异步FIFO硬件设计

在设计中, 把整个异步FIFO分成三个模块: 时钟管理单元、逻辑控制单元和数据处理单元, 如图1。

2.1 时钟管理单元

主要负责对读写时钟进行必要的频率和相位的调整, 并输出后续接口单元所需要的各种接口时钟信

号。其中 rd_clk 信号是驱动逻辑控制单元读指针 (rd_ptr) 和数据处理单元读数据 (rd_data) 的时钟信号；而 base_clk 是读数据 (rd_data) 通过 2/4 译码器的采样时钟信号，其仿真时序如图 2。

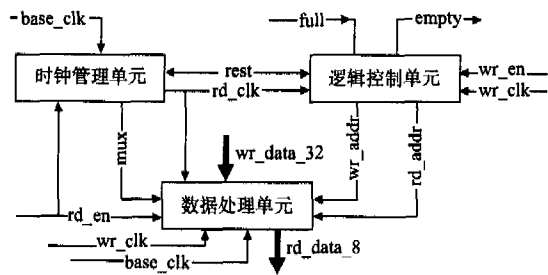


图1 异步FIFO

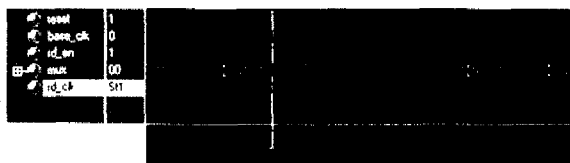


图2 时钟模块逻辑时序图

时钟的下降沿跳变，此时如若采样，则译码信号 mux 还没有度过自己的建立时间 δ_{setup} ，故会发生逻辑错误。而采用上升沿采样则不同，此时译码信号 mux 早已经建立起来(由图3可知：当此时采样时，译码信号 mux 处于最安全的时段)，故此时是安全采样，不会发生逻辑混乱。三是输出信号必须在 rd_clk 时钟的上升沿读出数据，并送入 2/4 译码器译码。

当异步清零信号 (reset) 到达下降沿时，所有的指针 (wr_ptr / rd_ptr) 都置 0，数据在写时钟 (wr_clk) 紧接着的上升沿进行采样时写入异步 FIFO 的 32-bit 缓冲内。第一次写入 0083f7e5，第二次写入 006d68c4 ……，如图 3，FIFO 数据写时序。

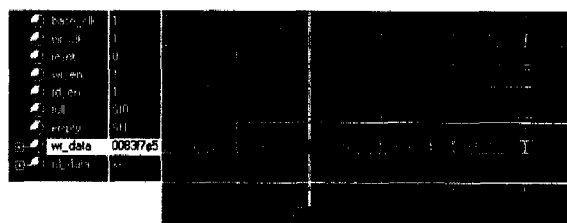


图3 FIFO数据写时序

2.2 逻辑控制单元

主要负责对输入的读写时钟信号 (rd_clk / wr_clk)、异步清零信号和读写使能信号 (rd_en / wr_en) 进行控制，并输出读写地址信号 (rd_addr / wr_addr) 和空满标志信号 (empty / full)。在由该单元传递给数据处理单元读写地址时，并非采用二进制的形式，而是采用格雷码 (Gray-Code)。二进制有其优点，就是编码简单；但是其缺点也是显而易见的：当较长的一个二进制码中出现一个逻辑电平错误时，将导致这两个二进制的距离面的不可预料性。比如 101010010B → 101110010B 时，他们的距离将会相差 32D，或者 101010010B → 101010110B 时，他们的距离只相差 4D。但是如果我们采用格雷码的话，不论是哪一位发生逻辑电平的改变，它们的距离都只有 1D，这样能降低差错的不可预料性。

2.3 数据处理单元

通过逻辑控制单元给出的读写地址 (rd_addr / wr_addr)，将数据写入 (wr_data_32) 和读出 (rd_data_8)。当读使能信号 (rd_en) 无效的时候，输出高阻态。由时钟管理单元的时序图可知：由于所有的信号都需要一个建立时间 δ_{setup} ，应该用 base_clk 时钟来当作 FIFO 缓存的读采样时钟，且必须要使用上升沿采样。如若用下降沿来采样，则采集到的状态很可能是不定状态。因为译码信号在 base_clk

数据在读时钟 (rd_clk) 上升沿进行采样，将采样得到的 32-bit 数据写入到下级的 2/4 译码器中。译码器用 mux 信号进行译码，每次译码出由异步 FIFO 读出的 32-bit 数据中的 8-bit。第一次读出 e5，第二次读出 f7 ……，可以看出这是和写入的数据对应的。此时读使能 (rd_en) 必须有效，如图 4 所示。

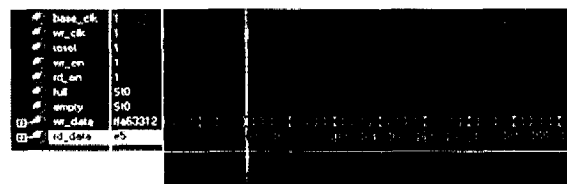


图4 FIFO数据读时序(读使能有效)

如果在读数据的时候，读使能 (rd_en) 突然变得无效，则此时 FIFO 的数据输出端口应置为高阻态 Z。并且保持读指针不变，使得当下次读使能 (rd_en) 有效的时候，能从上次的地址读出数据，避免数据的丢失，如图 5 读时序所示。

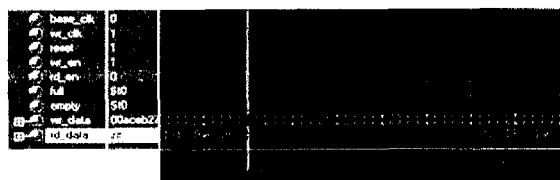


图5 FIFO数据读时序(读使能无效)

当写指针接近于读指针 ($|wr_ptr-rd_ptr|\leq 1$), 并且写指针 (wr_ptr) 超过读指针 (rd_ptr) “一圈”的时候, 满标志信号 ($full$) 有效。这时, 写指针 (wr_ptr) 不会增加; 只有等到读指针 (r_ptr) 与写指针 (wr_ptr) 的差值的绝对值大于 1 后, 这时满信号 ($full$) 变为无效, 写指针 (wr_ptr) 才会继续增加, 换句话说, FIFO 才会允许外界向其中写入数据。由仿真图 6 可以看出: 当 FIFO 装满后, 只有读指针 (rd_ptr) 工作, 而写指针 (wr_ptr) 停滞; 读指针向前“运动”某个时刻读指针 (rd_ptr) 与写指针 (wr_ptr) 差值的绝对值 $|wr_ptr-rd_ptr|>1$, 此时 $full$ 信号无效, 写时钟马上又开始工作, 短暂的无效 $full$ 信号马上又变得有效。这就是为什么 $full$ 信号会出现一个“梳状”。

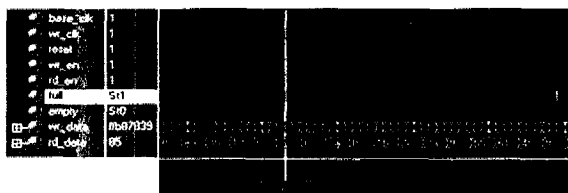


图6 FIFO 满空时序

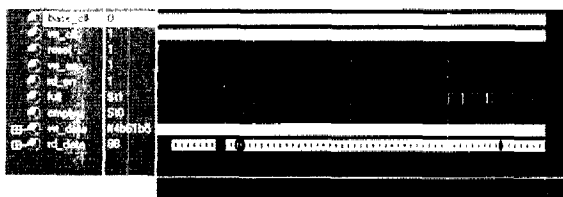


图7 FIFO 总仿真图

3 实验验证

实验仿真结果如图 2、图 3、图 4、图 5、图 6、图 7 所示。仿真结果达到了预期的参数要求, 完成了 FIFO 软硬件设计。

4 结论

文章解决了基于 FPGA 的 FIFO 握手信号 EMPTY 与 FULL 的设计。用 VERILOG 编程来实现异步 FIFO, 对程序进行了功能仿真、时序仿真, 最后下载到 FPGA 芯片中进行了硬件仿真。实验结果达到了预期的参数要求, 完成了 FIFO 软硬件设计。对其 32 位输入、8 位输出采用译码器来实现。此模块实现了缓冲输入的 TS 流数据, 从而达到同步传输数据的设计要求。

参考文献:

- [1] 王佳薇, 姚远程, 马建国. 基于 PCI 接口的 DVB 发送卡设计[J]. 电视技术, 2004, 5: 43-45.
- [2] 陈祖良. 基于 DVB-C 的 TS 流发送卡的研究[D]. 成都: 成都理工大学, 2003, 6: 23-24.
- [3] Xilinx. Spartan-II 2.5V FPGA Family: Functional Description[EB/OL]. www.direct.xilinx.com/bvdocs/publications/ds001.pdf, 2001.
- [4] 王诚. FPGA/CPLD 设计工具: Xilinx ISE 使用详解[M]. 北京: 人民邮电出版社, 2004.
- [5] J.Bhasker. Verilog HDL 硬件描述语言[M]. 北京: 机械工业出版社, 2000.
- [6] 李幼平. 共享信息的第二类网络[J]. 中国工程科学, 2002, 4(8): 56-58.
- [7] Integrated Device Technology [Z]. DT72V263 INTasheeu, 2003: 4-18.

作者简介:

王宏臣 (1976-), 男, 黑龙江北安人, 2005 年毕业于哈尔滨工业大学研究生院, 淮安信息职业技术学院机电系讲师, 主要从事单片机、PLC、FPGA 集成电路相关的机电产品的设计研发;

林咏海 (1980-), 男, 河南民权人, 2005 年毕业于河南大学研究生院, 淮安信息职业技术学院通信系讲师, 从事电子自动化研究。