

# 异步 FIFO 控制器的设计

邬春学, 华 乐

(上海理工大学 光电信息与计算机工程学院, 上海 200093)

**摘 要:** 异步 FIFO 是一种先进先出电路, 可以有效解决异步时钟之间的数据传递。通过分析异步 FIFO 设计中的难点, 以降低电路中亚稳态出现的概率为主要目的, 提出了一种格雷码计数器的技术, 通过仿真验证, 有效地实现了异步 FIFO 控制器的设计。该设计将大大提高工作频率和资源利用率。

**关键词:** 异步 FIFO; 亚稳态; 格雷码计数器

中图分类号: Z429

文献标识码: A

文章编号: 1674-7720(2012)04-0023-03

## Asynchronous FIFO controller design

Wu Chunxue, Hua Le

(School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China)

**Abstract:** Asynchronous FIFO is a first-in-first out circuit, which can effectively transmit data among asynchronous clocks. This paper analyses the difficult points of asynchronous FIFO design. In order to reduce the probability of metastability in the circuit, the paper presents the technique of gray code counter. Through simulations validate, effective achieved the design of asynchronous FIFO controller. This design improves the frequency of workings and the utilization of resources greatly.

**Key words:** asynchronous FIFO; metastability; gray code counter

随着现代芯片设计规模的不断扩大, 集成电路越来越复杂, 一个系统中往往包含多个时钟, 如何设计异步时钟之间的接口电路是多时钟领域的关键问题。异步 FIFO (First In First Out) 是一种先进先出电路, 用来存储、缓冲在两个异步时钟之间的数据传输, 使用异步 FIFO 可以在两个不同的时钟系统之间快速准确地传输实时数据, 是用来解决异步时钟接口电路的一个有效方案。并且异步 FIFO 高速、可靠性好, 在网络接口、图像处理等方面都得到了广泛的应用。

### 1 异步 FIFO 控制器的设计

#### 1.1 异步 FIFO 时钟域结构设计

FIFO 主要由 FIFO 控制器和 RAM 两个部分组成。FIFO 控制器最重要的功能就是产生 RAM 的读写地址以及相应的使能信号; 产生 FIFO 的状态标志, 包括空 (Empty)、满 (Full)、溢出 (Underflow, Overflow) 以及其他根据设计需要产生的状态标志。异步 FIFO 由两个时钟域构成: push clock domain (记为 clk\_push domain) 和 pop clock domain (记为 clk\_pop domain)。

所以异步 FIFO 可以划分为下列时钟域结构, 如图 1 所示。

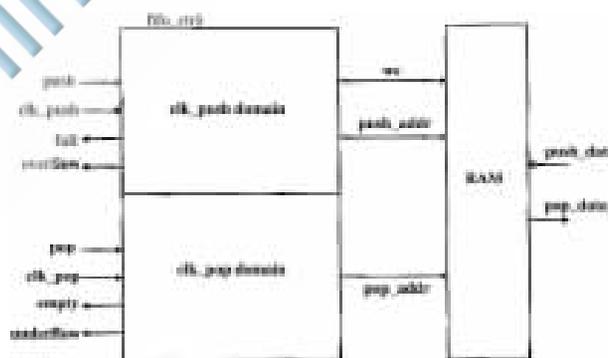


图 1 异步 FIFO 时钟域结构

#### 1.2 简单计数器实现读写地址输出结构设计

由时钟域结构可知, 作为 FIFO 控制器, 最基本的就是要根据外部的输入信号 push 和 pop, 产生对 RAM 访问的读写地址。FIFO 控制器内首先要实现对 RAM 的读写地址输出, 所以 FIFO 控制器内有一组基于 clk\_push domain 的逻辑产生写地址: push\_addr, 和一组基于 clk\_pop domain 的逻辑产生读地址: pop\_addr。然后对地址信息采用二进制编码, 每一次 push 操作, 使 push\_addr 增加 1, 即指向下一个 push 操作的 RAM 空间; 每一次

## 硬件纵横

Hardware Technique

pop 操作,使 pop\_addr 增加 1,即指向下一个 pop 操作的 RAM 空间。按照这种思路,异步 FIFO 控制器可进一步细化为如图 2 所示结构。

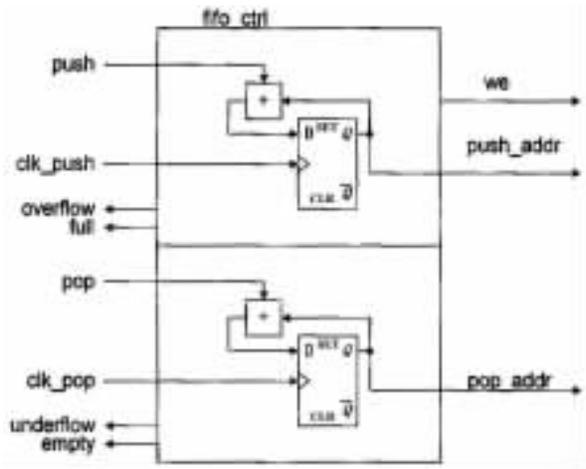


图 2 简单计数器实现读写地址输出结构

## 1.3 异步 FIFO 控制器设计中的关键问题

所谓异步是指读、写时钟是完全独立并且不一致的,或者不同频率,或者同频但不同相。读地址和空标志是由读时钟产生的,而写地址和满标志则由写时钟产生,当要产生 FIFO 的空、满标志时,必须进行读写地址的比较,地址线一般有多位,如果直接采样地址比较,就会存在问题。写地址的每一位在写时钟作用下,跳变会不一致,即产生毛刺,要过一段时间才能稳定。在未稳定期内,刚好读时钟进行采样写地址,这时就会出现误判断和逻辑错误从而导致了亚稳态的出现。一个好的 FIFO 设计的基本要求是:写满而不溢出,读空又不多读。

因而,异步 FIFO 设计主要存在两个关键问题:

(1) 如何产生空、满等相应的控制信号;

(2) 为了尽量降低电路中亚稳态出现的概率,如何同步从一个时钟域传送来的多位数据信号。

## 1.3.1 异步 FIFO 控制器空满标志产生

地址输出设计好后,接下来解决第一个关键问题,即异步 FIFO 的空满状态标志。

当 pop\_addr 追赶 push\_addr,并且赶上,即 pop\_addr = push\_addr 时,FIFO 为空,即置 empty;当 push\_addr 追赶 pop\_addr,并且赶上,即 push\_addr = pop\_addr 时,FIFO 为满,即置 full。

可以发现,不论是 empty 还是 full, pop\_addr 均与 push\_addr 相等,因而暂时无法区分到底是 empty 还是 full。所以需要增加额外的逻辑加以区分。

由于异步信号在使用前需要使用两级触发器同步才能在另一个时钟域被使用,因而在 clk\_pop domain,需要两级触发器来同步 push\_cnt;在 clk\_push domain,也需要两级触发器来同步 pop\_cnt。

在这里可以增加 almost\_full 和 almost\_empty 标志判断 empty 和 full;在计数器的复位值都必须为 0,并且为

二进制编码递增的前提下,当 FIFO 内数据少于某一预设值(低水位, low\_waterlevel)时,置位 almost\_empty;当 FIFO 内数据多于某一预设值(高水位, high\_waterlevel)时,置位 almost\_full。这时就很清楚,当 almost\_empty 有效,并且 pop\_addr = push\_addr 时,FIFO 为 empty;当 almost\_full 有效,并且 pop\_addr = push\_addr 时,FIFO 为 full。

此时,异步 FIFO 控制器可进一步细化为如图 3 所示结构。

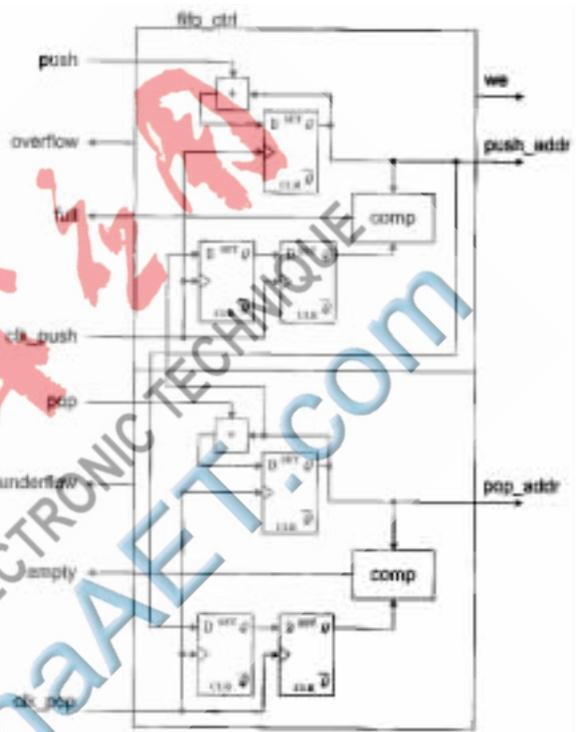


图 3 异步 FIFO 控制器的空满标志产生

## 1.3.2 亚稳态问题的存在及解决

在数字电路中,触发器需要满足 setup/hold 的时间要求。当一个信号被寄存器锁存时,如果信号与时钟之间不能满足这个要求,Q 端的值是不确定的,并且在未知的一时刻会固定到高电平或低电平,这个过程称为亚稳态。

亚稳态必然会发生在异步 FIFO 中,因为在异步 FIFO 中,电路的外部输入和内部时钟没有任何时间关系,因此存在 setup/hold 冲突是必然的,同时,在电路内部的两个没有关系的时钟域之间的信号传递也会出现 setup/hold 冲突。

虽然亚稳态是不可避免的,但是通过对写地址/读地址用格雷码可以将其降低到一个能够接受的范围之内<sup>[1]</sup>。同步多个异步输入信号出现亚稳态的概率远远大于同步一个异步信号的概率,所以对多个触发器的输出所组成的写地址/读地址需要采用格雷码。由于格雷码每次只有一个数据位变化,因而采用格雷码可以有效地减少亚稳态的产生。

按照这种思路,异步 FIFO 控制器可以设计为计数

《微型机与应用》2012 年 第 31 卷 第 4 期

## 硬件纵横

Hardware Technique

器采用 Gray Code 编码, 然后被另一个时钟域同步, 同时计数器必须按照 0、1、2、3 递增的顺序计数。于是先将 Gray Code 转换为二进制, 然后对二进制做加 1 运算, 将计算结果再转换回 Gray Code, 然后被触发器锁存。所以, 异步 FIFO 控制器的设计又可进一步细化为如图 4 所示结构。

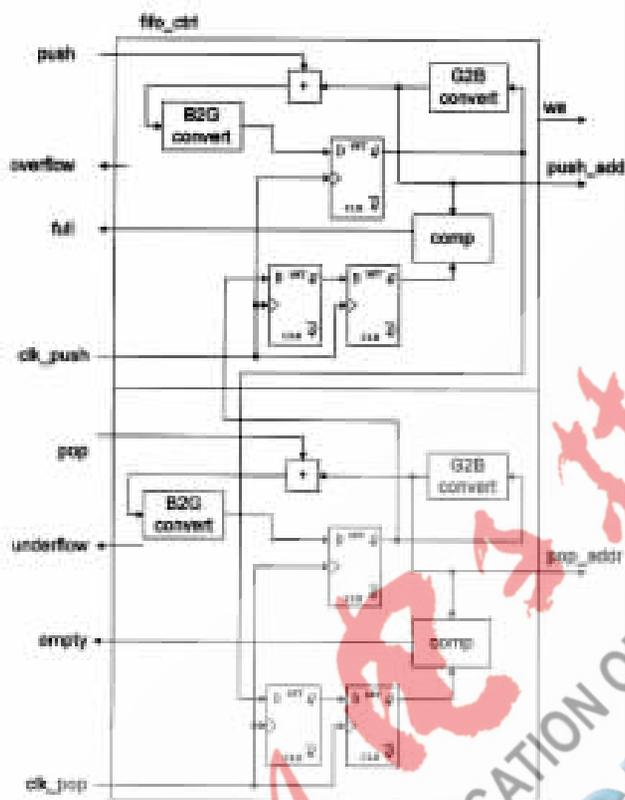


图 4 Gray Code 编码的异步 FIFO 控制器

现在一个异步 FIFO 控制器已经基本设计完成。在图上还剩 overflow、underflow、we 三个信号。在 full 时, 对 FIFO push 就会产生 overflow; 在 empty 时, 对 FIFO pop 就会产生 underflow。特别值得注意的是, 这时的 push 或 pop 都不应该使计数器继续翻转, full 后对 FIFO 的 push 操作 FIFO 控制器也不能输出有效的对 RAM 的写使能信号 we。这三个信号以及其他一些状态标志的具体实现可以根据实际应用进行设计。

## 2 用 Modelsim 仿真

在 Modelsim SE 上利用 Verilog HDL 对提出的方法进行了仿真, 仿真波形如图 5 所示。

从仿真时序图知:

(1) 复位后, 读信号和写信号均不使能(均置 1), 由于存储单元没有数据, 产生读空标志。

(2) 将写信号使能(置 0), 写入的数据与设计输入的数据一致; 将读信号使能, 读出来的数据顺序和数值与

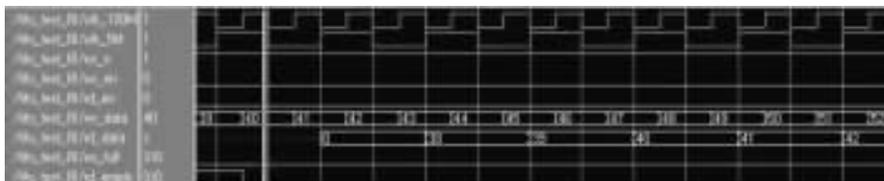


图 5 仿真波形

写入的数据一致。

(3) 将写信号置 1, 在一定的时钟下由预期设定的读地址加 1 与写地址相等时, 有读空标志产生; 接着将写信号使能, 将读信号置 1, 在一定的时钟周期下由预期设定的写地址加 1 与读地址相等时, 有写满标志产生。

(4) 将读写时钟使能, 数据的读写是正确的, 由于写时钟比读时钟快, 经过一定的时钟周期后, 有写满标志产生; 接下来, 由于写满不能再写, 故读时钟在读使能信号下读出数据时, 写满信号变为 0, 接着写满信号 1 和 0 交替出现。

(5) 系统复位后, 一个 8 位 rd\_cnt 计数器清 0, 在读使能和没有产生读空标志的条件下, 在读时钟上升沿的到来, rd\_cnt 计数器加 1, 跟随着 rd\_addr 读地址的变化, 查看得知产生的读地址变化符合预期设计; 同样, 系统复位后, 一个 8 bit wr\_cnt 计数器清 0, 在写使能和没有产生写满标志的条件下, 当写时钟上升沿的到来时, wr\_cnt 计数器加 1, 跟随着 wr\_addr 写地址的变化, 查看得知产生的写地址变化符合预期设计。

根据上述仿真波形分析, 可以看出所设计的 FIFO 控制器, 能满足需要完成的功能。

## 参考文献

- [1] KANOPOULOS, HALLENBECK J J. A First-In, First-out memory for signal processing applications[J]. IEE Transactions on circuits and system, 1986, CAS-33(05): 556-558.
- [2] 吴自信, 张嗣忠. 异步 FIFO 结构及 FPGA 设计[J]. 单片机及嵌入式系统应用, 2003(8):24-26.
- [3] CLIFFORD E. Simulation and synthesis techniques for asynchronous FIFO design [J]. SNUG San Joes, 2001(05).
- [4] 罗昊. 一种异步 FIFO 的设计方法[J]. 电子技术应用, 2004, 30(8):70-71, 74.
- [5] 魏芳, 刘志军, 马克杰. 基于 Verilog HDL 的异步 FIFO 设计与实现[J]. 电子技术应用, 2006, 32(7):97-99.
- [6] 杨青山, 蔡敏. 基于多时钟域的异步 FIFO 设计[J]. 中国集成电路, 2007(9):36-39.

(收稿日期: 2011-12-20)

## 作者简介:

华乐, 男, 1988 年生, 硕士研究生, 主要研究方向: 嵌入式计算机系统。