

基于随机延时注入的跨时钟域信号验证方法

梁 骏¹, 唐 露², 张 明¹

(1 浙江大学 信息与通信工程研究所, 浙江 杭州 310027; 2 杭州国芯科技股份有限公司, 浙江 杭州 310027)

摘 要: 为了应对现代 SOC 复杂的时钟结构给跨时钟域信号处理带来的隐患, 分析了跨时钟域信号产生的亚稳态现象的根本原因和常用的跨时钟域信号的处理方法, 针对跨时钟域信号处理难以验证的问题, 提出了基于随机延时注入的跨时钟域仿真验证方法. 通过将亚稳态现象抽象成采样数据在时钟上的随机抖动, 使得芯片设计的 RTL 前仿真在没有时钟树物理信息的情况下能够模拟出亚稳态效应. 分析结果表明此方法能够完成 SOC 芯片的跨时钟域信号的功能验证.

关键词: 跨时钟域; 亚稳态; 随机抖动

中图分类号: TN47

文献标识码: A

文章编号: 1000-7180(2014)02-0001-04

A Random Delay Injection Based Clock Domain Crossing Verification Method

LIANG Jun¹, TANG Lu², ZHANG Ming¹

(1 Institute of Information and Communication Engineering, Zhejiang University, Hangzhou 310027, China;

2 Hangzhou Nationalchip S&T Co., Ltd, Hangzhou 310027, China)

Abstract: Complicated clock infer-structure of modern SOC brings potential problems related with CDC signal interface operations. The root cause of metastability caused by CDC and the methods dealing with it are discussed in detail. A random delay injection based CDC verification method is presented to meet the verification challenge of CDC. This method converts metastability into sample data random jitter lasting for one clock period, which makes it feasible the RTL pre-simulation when physical clock tree information is not available. The result of an example shows the method is suitable to do CDC verification for SOC designs.

Key words: CDC; Metastability; Random Jitter

1 引言

CDC(Clock Domain Crossing, 跨时钟域)信号的验证一直是复杂 SOC(System on Chip, 系统芯片)面临的问题^[1]. 比如为了满足低功耗设计的需求, 不少复杂芯片实现了 DVFS(Dynamic Voltage and Frequency Scaling, 动态电压频率调整)^[2]等先进设计方法.

这些数目众多且动态变化的时钟结构给芯片带来了复杂的时序关系. 由跨时钟域信号不正常处理导致的芯片功能错误成为导致芯片失败的一个重要

原因.

针对跨时钟域信号的验证是保证跨时钟域信号正确处理的解决方法. 常用的方法可以分成两类: 一是静态验证的方法^[3-4], 二是动态仿真验证的方法.

静态验证又可分为形式验证与文法检查两种. 形式验证方法通常局限在模块级. 文法检查验证方法是通过 RTL(Register Transfer Level, 寄存器传输级)代码的语法解析, 提取出跨时钟域信号, 并报告出没有进行跨时钟处理的信号. 它对跨时钟处理得不正确, 或对复杂跨时钟情景, 如跨时钟信号的会聚、回路等无法做出正确的判断.

收稿日期: 2013-04-06; **修回日期:** 2013-06-02

基金项目: 国家自然科学基金项目(61072081, 61271338); 国家科技重大专项(2009ZX01033-001-007)

动态验证方法是通过仿真器模拟出芯片的真实行为.但是跨时钟域信号的仿真需要在具备时钟树物理信息的版图后仿真才能进行.由于版图后仿真运行时间太长,而且 CDC 问题是以较低的概率出现,所有无法高效地对跨时钟域信号进行验证.

本文针对跨时钟域信号验证的难点,研究了跨时钟信号的特点,将在器件级的亚稳态行为进行抽象,提出跨时钟信号的行为级特征,通过 RTL 代码注入的方式使得 RTL 级的前仿真能够体现出跨时钟信号的亚稳态特征.结果表明这种方法能够在 RTL 级前仿真中高效地进行跨时钟域信号的验证.

2 跨时钟域信号

当信号的发送端的时钟与接收端的时钟不是同一个时钟时,这个信号就是跨时钟域信号.跨时钟域信号产生的问题主要是亚稳态问题及由亚稳态引发的时序问题.

亚稳态问题的起源是由于采样寄存器不满足建立(setup)与保持(hold)的时序要求.当采样寄存器的时钟来临时,如果信号也刚好处于变化的时刻,就将产生亚稳态现象.此时寄存器输出端 Q 在有效时钟沿之后比较长的一段时间内处于不确定的状态.图 1 所示为亚稳态波形的 SPICE 仿真结果.如图 1 所示,根据信号信号变化沿与时钟变化沿的时间间隔不同,最后的输出端 Q 的波形可能是一个毛刺,或表现为振荡,或为固定的某一电平,而不是等于数据输入端 D 的值.这段不确定的时间称为决断时间(resolution time).经过决断时间后 Q 端将稳定到 0 或 1 上,但是究竟是 0 还是 1,与输入信号 D 没有必然的关系^[5].

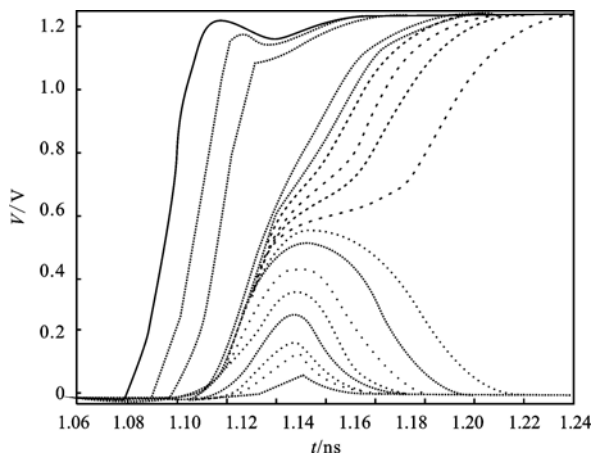


图 1 亚稳态输出波形

时序电路都是基于稳态进行分析的.亚稳态除了寄存器输出状态是 0 或 1 不定导致后续逻辑判断

出错之外,当寄存器输出介于 0~1 之间的中间电平时还会导致时序错误,可能使下一级寄存器也产生亚稳态,即亚稳态在时序电路中扩散.如不对跨时钟域信号进行正确处理,亚稳态与亚稳态的扩散将导致时序异常,芯片功能错误.

3 CDC 信号处理

跨时钟域信号处理常用 3 种处理方法:双寄存器同步、握手同步或单向握手同步、FIFO 同步^[6].

3.1 双寄存器同步

双寄存器同步是最常用的跨时钟域信号同步方法.双寄存器同步方法是指将跨时钟域信号使用接收端时钟同路 2 级后作为同步信号使用,如图 2 中的 READY 与 ACK 信号.跨时钟域信号经过两级寄存器同步后的输出基本不存在亚稳态.其原理是即使第一个寄存器的输出端以极低的概率存在亚稳态,而第二个寄存器采样到第一个寄存器的亚稳态,并导致自身出现亚稳态的概率非常小.因此第二个寄存器输出产生亚稳态的概率在工程上近似于零.

文献[7]中给出了平均两次同步失败间隔(Mean Time Between Failures, MTBF)的公式:

单同步寄存器:

$$MTBF_1(t_r) = e^{t_r/\tau} / T_o f_s f \quad (1)$$

双同步寄存器:

$$MTBF_2(t_r) = e^{t_r/\tau} / T_o f_s (MTBF_1^{-1}) \\ = MTBF_1^2 / f \quad (2)$$

式中, t_r 是信号建立时间, f_s 是采样时钟频率, f 是输入信号频率, τ 与 T_o 是与寄存器相关的时间常数.

3.2 握手同步

当有一组总线进行跨时钟处理时不能通过简单对每条信号使用双寄存器同步的方法进行处理.这时因为总线间不同信号线间的延时不同导致出现错误数据.此时可以使用握手信号进行同步.

通用握手同步如图 2 所示.发送端在准备好总线上的数据后令 READY 有效.接收端对 READY 信号进行双寄存器同步.接收端收到 READY 有效后,将采样数据总线上已经稳定的数据,并发送 ACK 应答.发送端对 ACK 进行双寄存器同步.当发送端收到 ACK 有效后,将 READY 无效,并准备发送下一个数据.

握手同步可以解决数据总线跨时钟域传输的问题.但是由于 READY 与 ACK 在两个方向都要同步,当发送与接收端的时钟频率相接近时,传输 1 个数据至少要 4 个时钟.数据总线的最高传输效率是

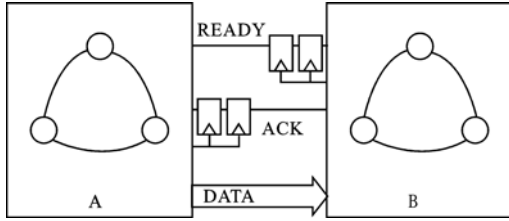


图2 握手同步

25%,传输的效率较低。

3.3 FIFO 同步

FIFO 通过写指针与读指针帮助数据写入与读出。在发送端,只要 FIFO 不满,就可以向 FIFO 写入数据。在接收端,只要 FIFO 不空,就可以从 FIFO 读出数据。跨时钟域问题被转移到 FIFO 内的空信号与满信号的产生上。通常使用格雷码对地址进行编码,以减少 FIFO 内部跨时钟域现象。

FIFO 同步是传输效率最高的一种同步方法。在当发送与接收端的时钟频率相等时,FIFO 同步方法可以 1 个时钟传输 1 个数据。传输的效率达到 100%。

4 CDC 信号行为特征与抖动注入

亚稳态现象需要确定时钟信息才能进行仿真。因为容量与运行时间的限制,SOC 芯片无法通过版图后仿真进行跨时钟域的验证。为了能够在 RTL 级前仿真上进行仿真验证亚稳态问题,必须对亚稳态的特征从器件级抽象到行为级。

考察在第 1 个采样时钟来临的时候输入信号从 0 到 1 跳变的情况。第 1 个时钟后第 1 级同步寄存器会有 3 个输出结果:输出 1;或输出 0;或输出中间电平。

当输出为 1 时,第 2 级同步寄存器在第 2 个采样时刻出现 0 到 1 的跳变。

当输出为 0 时,第 2 个采样时刻第 1 级同步寄存器采样到 1。第 2 级同步寄存器在第 3 个采样时刻出现 0 到 1 的跳变。

当输出为中间电平时,第 2 个采样时刻第 2 级同步寄存器可能为 0,也可能为 1。第 2 个采样时刻第 1 级同步寄存器采样到 1。第 2 级同步寄存器在第 3 个采样时刻输出 1。即第 2 级同步寄存器在第 2 个采样时刻或第 3 个采样时刻出现 0 到 1 的跳变。

输入信号从 1 到 0 跳变时的情况与此类似。

综上所述,跨时钟域信号的亚稳态现象在行为级的表现是跨时钟域信号出现在以采样时钟长度为单位的 1 个长度的随机延时变化。因此将跨时钟信

号随机延时 1 个时钟长度后 RTL 前仿真就可以实现对跨时钟域信号的亚稳态现象的功能验证。

以下是针对 verilog 设计代码的随机延时注入方法:

(1)使用文本静态检查的方式提取跨时钟域信号的接收端;

(2)将接收端的设计代码进行文体替换,注入随机延时注入代码;

(3)通过 RTL 前仿真完成跨时钟域信号的亚稳态验证。

以下是本方法的处理例子。

处理前的原始 verilog 代码如下:

①always @(posedge sys_clk)

②begin

③ reg1 <= value;

④end

经过代码注入后的 verilog 代码如下:

⑤always @(posedge sys_clk)

⑥begin

⑦ time pre_clk=0;

⑧ integer clk_length=0;

⑨ begin

⑩ clk_length= \$ time-pre_clk;

⑪ pre_clk= \$ time;

⑫ end

⑬ reg1 <= # (\$ random% clk_length + 0.5 * clk_length) value;

⑭ end

其中 pre_clk 存储前一个时钟的仿真时间。clk_length 是采样时钟的长度。第 7 行到第 12 行代码用于动态计算当前时钟周期长度。第 13 行代码将采样的数据随机延迟 0.5 到 1.5 个时钟长度后赋值到接收寄存器。当延迟小于 1 个周期时,相当于数据在当前时钟采样。当延迟大于 1 个周期时,相当于数据在下一时钟采样。这样就实现了跨时钟信号的延时随机注入,将跨时钟信号的行为特征体现到 RTL 代码中。修改后的 verilog 代码直接可用于 RTL 前仿真。

5 仿真结果及分析

图 3 是跨时钟域信号经过随机延时注入的 RTL 前仿真波形图。TX 是信号发送端波形。CLK 是接收端采样时钟。RX 是原始代码的接收信号。RX'是经过代码注入后的接收信号。RX'的第 1,3,

4,6 个信号跳变沿注入了 1 个时钟的延时. 第 2,5 个信号跳变沿没有注入延时. 即每个信号跳变沿都注入了随机的延时. 这也相当于将亚稳态出现的概率提高到每个跳变都出现. 相应的与亚稳态问题的验证效果也将大大提高.

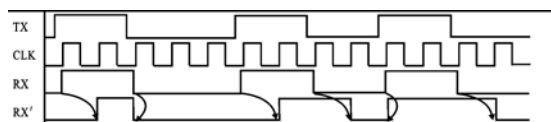


图 3 随机注入的 RTL 仿真波形

本文提出的方法在实践中对于跨时钟域汇聚和跨时钟跨时钟域环路有明显的验证效果. 如图 4 中的设计存在 CDC 会聚问题. A, B, C 和 D 分别是 4 个不同时钟域的模块. A 模块的一个信号输出同时进入 B 模块与 C 模块, B 模块的输出与 C 模块的输出在 D 模块会聚. 由于跨时钟域的亚稳态效应, B 模块接收到的信号与 C 模块接收到时的信号的有效时间可能会出现 1 个时钟的延时抖动. 然后在 B 模块发回到 D 模块与 C 模块发送到 D 模块的信号又会产生 1 个时钟的延时抖动. 最终在 D 模块产生 2 个时钟的延时抖动. 如果不对跨时钟域的亚稳态进行仿真, 是无法正确产生有 2 个时钟延时抖动的测试矢量. 而且由于在形式上每个模块都进行了跨时钟域信号的同步处理, 形式验证工具也将认为设计代码是正确的. 但是通过本文的方法可以将跨时钟域引起的亚稳态现象进行有效的建模, 通过 RTL 前仿真验证快速查找到设计缺陷.

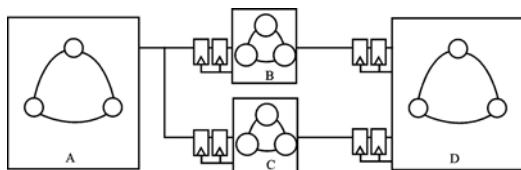


图 4 CDC 会聚

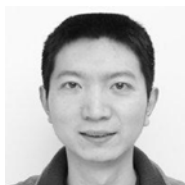
6 结束语

本文利用 CDC 的静态工具提取出跨时钟域的路径, 对 RTL 代码中的跨时钟域信号采样代码进行修改. 通过在跨时钟域路径上注入一个采样时钟长度的随机延时, 使得 RTL 的前仿真能够体现出亚稳态效应. 试验结果表明这是有效的对复杂跨时钟域信号进行验证的方法, 而且处理容量不受限制, 仿真时间不受影响. 这种方法特别适用于有复杂时钟结构的 SOC 芯片对跨时钟域信号的验证.

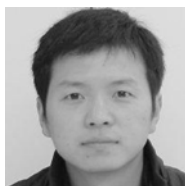
参考文献:

- [1] Li Luo, He Hongjun, Qiang Dou, et al. Design and verification of Multi-clock domain synchronizers[C] // Intelligent System Design and Engineering Application. Changsha, 2010:544-547.
- [2] Rotem E, Mendelson A, Ginosar R, et al. Multiple clock and voltage domains for chip multi processors [C] // Microarchitecture, MICRO-42, 42nd Annual IEEE. New York, NY, 2009:459-468.
- [3] Schmaltz J. A formal model of clock domain crossing and automated verification of Time-Triggered hardware[C] // Formal Methods in Computer Aided Design, 2007. FMC. Austin, TX, USA, 2007:223-230.
- [4] Bing Li, Kwok C K. Automatic formal verification of clock domain crossing signals[C] // Design Automation Conference, ASP-DAC 2009. Yokohama, 2009: 654-659.
- [5] Horstmann J U, Eichel H W, Coates R L. Metastability behavior of CMOS ASIC flip-flops in theory and test[J]. Solid-State Circuits, 1989, 24(1):146-157.
- [6] Chelcea, Tiberiu. Robust interfaces for mixed-timing systems[J]. Very Large Scale Integration (VLSI) Systems, IEEE, 2004, 12(8): 857-873.
- [7] Dike C, Burton E. Miller and noise effects in a synchronizing flip-flop[J]. Solid-State Circuits, 1999, 34(6):849-855.

作者简介:



梁 骏 男, (1978—), 博士研究生. 研究方向为 SOC 集成电路设计及高速电路设计.



唐 露 男, (1983—), 工程师. 研究方向为集成电路设计与验证.



张 明 男, (1962—), 教授, 博士生导师. 研究方向为信号与信息处理.