

一种新的基于层次化模式实现的 SOC 时钟设计方法

王丹丹, 邹连英, 郑宽磊

(武汉工程大学 电气信息学院, 湖北 武汉 430070)

摘 要: 通过对 SOC 传统时钟设计在层次化开发模式下遇到的新问题进行分析, 提出了一种新的时钟设计方法。利用相位同步信号(Phase_sync)作为层次化模式中顶层(Top)和子设计(Sub-design)之间的桥梁, 有效解决了顶层时序收敛时对子设计内部时序路径造成的影响。同时, 规避了对时钟分频电路进行复位同步化处理, 降低了物理设计时序收敛的难度。

关键词: 平面化设计; 层次化设计; 时钟偏斜; 相位同步; 同步器

中图分类号: TN492

文献标识码: A

文章编号: 1000-7180(2011)11-0089-05

New Method of SOC Clock Design Based on Hierarchical Mode

WANG Dan-dan, ZOU Lian-ying, ZHENG Kuan-lei

(School of Electrical and Information Engineering, Wuhan Institute of Technology, Wuhan 430070, China)

Abstract: In this paper, the new problems of traditional clock design in the hierarchical mode were analyzed and a new method of clock design was proposed. A phase_sync signal was used as a bridge of the top and sub-design in this method. It effectively prevents the 'damage' to the internal timing of subchip caused by the top-level timing closure. At the same time, the application of this method avoids reset design of clock divider circuit and reduces the difficulty of back-end timing closure.

Key words: flatten design; hierarchy design; clock skew; phase_sync; synchronizer

1 引言

随着深亚微米工艺的发展, 集成电路设计的主流工艺正开始从微米级向纳米级转变, 设计规模也从百万门级发展到了千万门级。但现有的 EDA 工具, 如综合、时序分析、自动布局布线等工具的性能都不能很好的适应这种设计规模的快速增长。

为适应此变化, 很多公司开始从“平面化”(Flatten)设计模式转向采用“层次化”(Hierarchy)设计流程。图 1 所示“平面化”模式和“层次化”模式的区别。“平面化”模式是指将整个设计作为一个整体进行物理设计; “层次化”模式, 是指将较大规模的设计分离成多个小的子设计模块 SD(Sub Design), 如 SD1, SD2, ..., SDn, 再对之进行相对独立的物理设计, 然后进行顶层(TOP)的物理集成。分层设计

方法允许物理和逻辑设计协同展开, 缩短了设计周期。

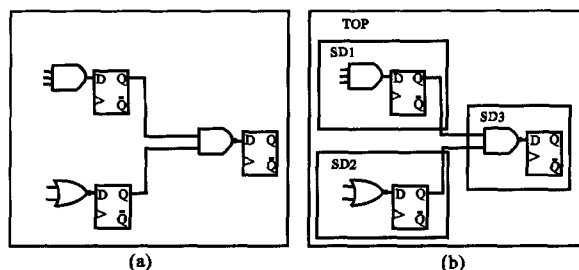


图 1 平面化模式和层次化模式区别

从物理设计的角度看, 时钟在芯片设计中占有重要的地位。如果将原来平面化模式下的时钟设计方案直接应用到层次化模式下, 物理实现时会遇到一个问题: 各个子设计(SD)完成物理实现和时序收敛后, 进行顶层集成时很可能要重新对子设计的时

钟树进行较大调整,导致子设计和顶层设计需要进行多次迭代才能完成整个 Soc 芯片的时序收敛。

如果设计流程从平面化模式转变为层次化模式,则需要考虑如何避免顶层集成对子设计时钟树造成的影响.文中提出一种新的基于层次化设计模式的时钟设计方法:通过相位同步信号即 phase_sync 信号,在 SD 内部实现时钟分频,使所有同步时钟之间有固定的相位关系.这样,在进行顶层集成时可有效避免对子设计的时钟树做调整,大幅度减少迭代次数;同时无需进行复位设计。

2 层次化设计模式下传统时钟设计方案分析

图 2 所示为层次化设计模式下使用传统时钟设计方案的例子.其中,clk1x 是 clk2x 的分频时钟; $\Delta SL1X$ 和 $\Delta SL2X$ 分别表示 SD1 的两个时钟输入 clk1x 和 clk2x 的时钟源延迟 (source latency^[1]), CTS1X 和 CTS2X 是内部时钟树,以上两个时钟进行数据交互的逻辑为 LogicA。

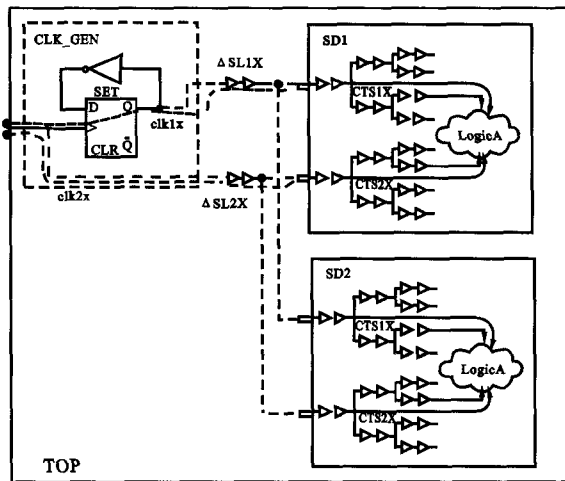


图 2 层次化设计模式下传统时钟方案

对 SD1 进行物理设计时,需提供 $\Delta SL1X$ 和 $\Delta SL2X$ 的预估值.如公式 (1) 所示,根据预估值,LogicA 的时序收敛完成后,clk1x 和 clk2x 两个时钟树之间的时钟偏斜 $SKEW_{CTS2X-CTS1X}$ ^[2] 为一个固定值。

$$\begin{aligned} SKEW_{CTS2X-CTS1X} &= |(\Delta SL2X + CTS2X) - (\Delta SL1X + CTS1X)| \\ &= |(\Delta SL2X - \Delta SL1X) + (CTS2X - CTS1X)| \quad (1) \end{aligned}$$

SD1 的物理设计完成之后进行顶层集成时,需要生成真实的顶层时钟树,但真实的时钟树必然与预估的时钟树不一致.因此,在进行顶层集成的物理设计阶段, $\Delta SL1X$ 和 $\Delta SL2X$ 这两个参数的实际值

和预估值必然存在偏差,某些情况下偏差会很大.此时时钟偏斜 $SKEW_{CTS2X-CTS1X}$ 会随之发生变化.这个结果可能导致 LogicA 的部分路径时序关系发生违反 (violation^[3]).如图 3 所示,当 $\Delta SL2X - \Delta SL1X$ 发生变化时,建立时间检查 (Setup check) 和保持时间检查 (Hold check) 将发生变化.子设计的物理工程师不得不重新进行物理设计的迭代工作。

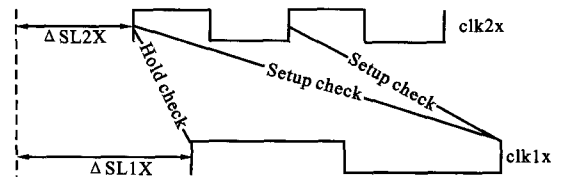


图 3 $\Delta SL1X$ 和 $\Delta SL2X$ 不确定条件下时序说明图 (LogicA 时序)

3 层次化设计模式下传统时钟方案的改进

传统时钟设计方案的问题根源在于一个子设计 SD 中有 clk1x 和 clk2x 两个同步时钟输入.现在参考下面的设计将层次化设计模式下传统时钟方案进行改进,如图 4 所示将图 2 中的分频电路移到子设计内部.这样就能保证子设计只有一个时钟输入,即 clk2x。

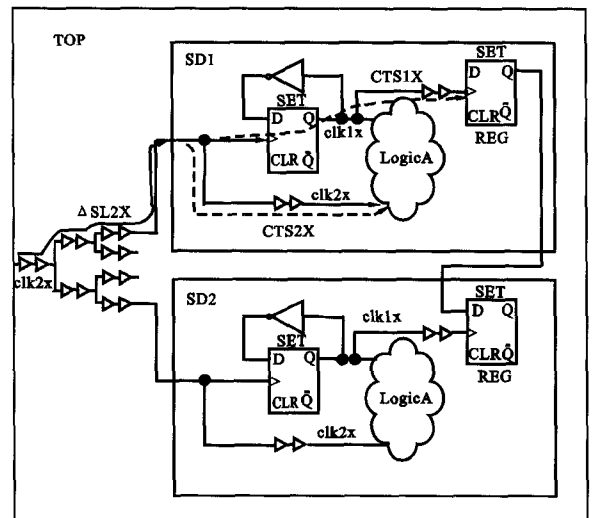


图 4 层次化设计模式下时钟方案

以图 4 中 SD1 为例,clk2x 和 clk1x 时钟源延迟均为 $\Delta SL2X$.如公式 (2) 所示 SD1 内部 clk1x 与 clk2x 两个时钟树之间的偏斜 (SKEW) 值与 $\Delta SL2X$ 的预估值无关.在进行顶层时序收敛时,可能需改变 $\Delta SL2X$ 的值以满足某些顶层逻辑的时序,但 $\Delta SL2X$ 的值的变化不会影响 SD1 内部 clk1x 与 clk2x 两个时钟树之间的偏斜 (SKEW) 值。

$$\text{SKEW}_{\text{CTS2X-CTS1X}}$$

$$= |(\Delta\text{SL2X} + \text{CTS2X}) - (\Delta\text{SL2X} + \text{CTS1X})|$$

$$= |\text{CTS2X} - \text{CTS1X}| \quad (2)$$

本例中的子设计 SD 中仅输入一个高频时钟,低频时钟在该 SD 内部通过高频时钟分频获取。此方法有效解决了顶层时序收敛过程对 SD 内部时序路径造成的影响。

但上述方案仍然存在缺陷。如图 4, SD1/REG 和 SD2/REG 之间有数据交互。由于 SD1 和 SD2 中各有一个随机的时钟分频电路 SD1/clk1x 和 SD2/clk1x,彼此间的相位差可能为 0° 或 180° ,因此无法对之进行时序约束。

这个问题的产生在于 SD1 和 SD2 两个子设计内部的分频时钟之间无固定的相位关系。常规解决办法是采用异步复位同步撤离的方式进行规避。如图 5 所示,RESET_GEN 模块产生 rstn 信号用于复位 SD1 和 SD2 的分频电路,通过对 rstn 信号进行同步撤离保证 SD1/clk1x 和 SD2/clk2x 的相位一致^[4]。

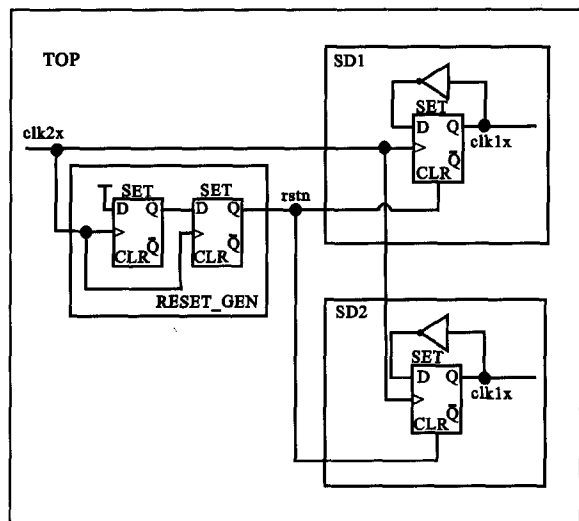


图5 复位同步化方案

这种设计方法可以解决 SD1/clk1x 和 SD2/clk1x 相位一致性问题,但需要进行复位同步化设计。如果系统时钟频率很高,进行复位同步化处理将非常困难。

4 层次化设计模式下新型时钟方案

下面提出的一种新型的设计方案,既可避免一个 SD 中同时有两个同步时钟输入,又可以避免做分频电路的复位设计。层次化设计模式下新型时钟方案如图 6 所示。

为降低高频情况下时序收敛难度,提高物理设

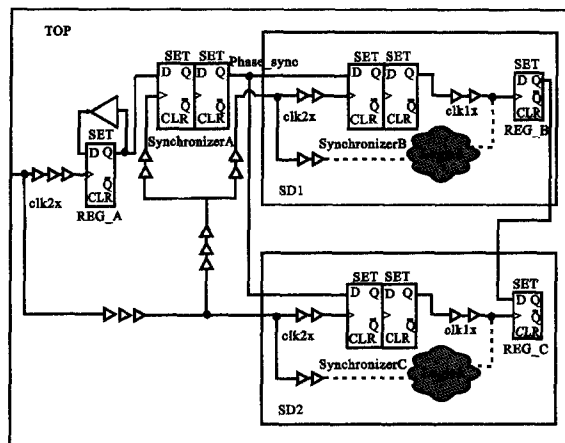


图6 改进后的时钟方案

计时序收敛的灵活性,在顶层增加一个两级同步器^[5]SynchronizerA 用来产生 phase_sync(相同步)信号。在 SD1(SD2 同)内部利用 clk2x 时钟,通过子设计内部的两级同步器 SynchronizerB 对 phase_sync 信号进行同步化处理作为 clk1x 时钟。

此方案的设计思想是利用 phase_sync 信号同步 SD1 和 SD2 两个子设计的时钟相位。Phase_sync 信号分别同时进入 SD1 和 SD2 中,clk2x 驱动的同步器 SynchronizerB 和 SynchronizerC 分别对此信号进行同步处理,就会产生 SD1/clk1x 和 SD2/clk1x 两个相位一致的时钟。其时序图如图 7 所示。本例中,SD1 和 SD2 子设计中使用的同步器为两级同步器。

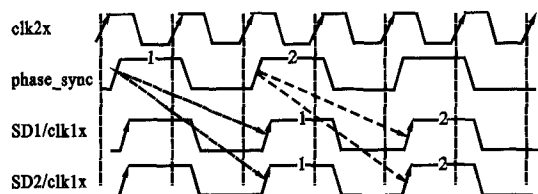


图7 两级同步器产生的 phase_sync 和 clk1x 时序图

这种设计方法的特点有两个:①每个子设计只有一个时钟输入。当进行顶层做物理设计时,对于子设计影响非常小;②各个子设计内部部分频时钟的相位一致,避免对时钟进行复位同步化处理^[6-7]。

5 实践结果分析

表 1,对子设计进行时钟树综合时,预估顶层时钟树延迟为 1500ps;实际时钟网络延迟(即图 2 中的 CTS1X 和 CTS2X)最大 skew 为 1.79679ps,总的最大 skew 为 1.79679ps。表 2,进行顶层时序收敛时,调整子设计的时钟树,得到时钟网络延迟最大 skew 为 54.41914ps,总的最大 skew 为 132.78746ps。

表 1 图 2 对应的子设计时钟树数据

单位:ps 子设计的时钟树			
寄存器 编号	时钟源延迟 (预估值)	时钟网络延迟 (实际值)	总的时钟 延迟
10	1500	256.46281	1756.46281
11	1500	255.42674	1755.42674
49	1500	255.23209	1755.23209
50	1500	256.49506	1756.49506
51	1500	255.47638	1755.47638
64	1500	254.77722	1754.77722
65	1500	254.69827	1754.69827
66	1500	254.88965	1754.88965
最大 skew	0	1.79679	1.79679

表 2 图 2 对应的顶层时钟树数据

单位:ps 顶层时钟树延迟(包含子设计延迟)			
寄存器 编号	时钟源延迟 (预估值)	时钟网络延迟 (实际值)	总的时钟 延迟
6	1408.9689	281.87915	1690.84805
7	1320.8291	282.15421	1602.98331
8	1410.9027	287.79017	1698.69287
60	1443.6368	249.09174	1692.72854
61	1489.8691	245.90167	1735.77077
72	1406.5106	238.13612	1644.64672
73	1410.9807	233.37103	1644.35173
74	1399.4618	234.90174	1634.36354
最大 skew	169.04	54.41914	132.78746

表 3,对子设计进行时钟树综合时,预估顶层时钟树延迟为 1500ps;实际时钟网络延迟最大 skew 为 1.07122ps,总的最大 skew 为 1.07122ps.表 4,进行顶层时序收敛时,调整子设计的时钟树,得到的时钟网络延迟的最大 skew 为 1.07062ps,总的最大 skew 为 52.56142ps.

表 3 图 6 对应的子设计时钟树数据

单位:ps 子设计的时钟树			
寄存器 编号	时钟源延迟 (预估值)	时钟网络延迟 (实际值)	总的时钟 延迟
27	1500	275.81348	1775.8135
28	1500	275.71428	1775.7143
29	1500	275.83006	1775.8301
30	1500	275.71179	1775.7118
31	1500	275.83849	1775.8385
32	1500	275.55076	1775.5508
77	1500	274.94695	1774.947
78	1500	274.76727	1774.7673
79	1500	274.95118	1774.9512
80	1500	274.93915	1774.9392
最大 skew	0	1.07122	1.07122

表 4 图 6 对应的顶层时钟树数据

单位:ps 顶层时钟树延迟(包含子设计延迟)			
寄存器 编号	时钟源延迟 (预估值)	时钟网络延迟 (实际值)	总的时钟 延迟
13	1381.3365	275.53638	1656.8729
14	1381.5233	275.6752	1657.1985
15	1379.2074	275.51944	1654.7269
47	1373.3721	275.64279	1649.0149
48	1373.3436	275.76889	1649.1125
49	1372.9787	275.48056	1648.4593
68	1331.1242	274.8718	1606.086
69	1330.8105	274.69827	1605.5087
70	1331.6074	274.87735	1606.4848
71	1330.9668	274.88098	1605.8478
72	1329.7687	274.86835	1604.6371
73	1330.3151	274.84167	1605.1567
最大 skew	51.7546	1.07062	52.56142

可以看出,采用新型时钟设计方案后,进行顶层集成时序收敛时,对于已经完成的子设计的影响非常小,误差为 $1.07122-1.07062=0.0006ps$;采用

传统时钟设计方案,顶层进行时序收敛后,对于子设计的影响较大,误差为 $54.41914 - 1.79679 = 52.62235\text{ps}$ 。

以上数据均在 Cadence 的 SocEncounter 平台下运行获取的。

6 结束语

这种时钟设计方法已成功应用在某大规模通信 SOC 芯片的设计中^[8-9]。实践证明,这种层次化模式下的新型时钟方案有如下优点:

(1) 分频电路通过同步器产生,无需在 SDn 内部实现二分频电路,实现简单;

(2) 进行顶层集成时,只需对单比特的 phase_sync 信号进行时序收敛,降低了时序收敛的难度;

(3) 顶层无需进行异步复位设计,降低了顶层复位设计给时序造成的难度;

(4) 每个子设计只有一个时钟输入端口,因此当 SDn 单独完成时序收敛后,顶层时序收敛对 SDn 影响很小,大大减少了迭代次数。

参考文献:

[1] Synopsys. PrimeTime fundamentals user guide[S]. Version C-2009. 6.

[2] Synopsys. Advance timing analysis user guide[S]. Version C-2009. 6.

[3] Michael D Ciletti. Advanced digital design with the verilog HDL[M]. 北京:电子工业出版社,2010.

[4] Clifford E Cummings, Don Mills. Synchronous resets? asynchronous resets? [EB/OL]. [2001-03-12]. <http://www.sunburst-design.com/papers/>.

[5] Clifford E. Cummings. synthesis and Scripting techniques for designing multi-asynchronous clock designs [EB/OL]. [2001-03-12]. <http://www.sunburst-design.com/papers/>.

[6] 千路,林平分. ASIC 后端设计中的时钟偏移以及时钟树综合[J]. 半导体技术,2008(6):33.

[7] 刘辉华,刘振,李蜀霞,等. 层次化时钟网络设计研究[J]. 微电子学与计算机,2008,25(11):52-54.

[8] 李蜀霞. 基于 ASIC 实现雷达信号处理芯片的后端设计[D]. 合肥:电子科技大学,2008.

[9] 于倩. ASIC 芯片的层次化后端实现[D]. 上海:复旦大学,2008.

作者简介:

王丹丹 女,(1977-),硕士研究生,讲师. 研究方向为检测技术与信号处理、图像处理技术

(上接第 88 页)

[3] Mei B, Vernalde S, Verkest D, et al. ADRES: an architecture with tightly coupled VLIW processor and coarse-grained reconfigurable matrix [C]// International Conference on Field-programmable Logic and Applications. Los Angeles, 2003:61-70.

[4] Panesar G, Towner D, Duller A, et al. Deterministic parallel processing[J]. International Journal of Parallel Programming, 2006, 34(4):323-341.

[5] Kissler D, Kupriyanov A, Hannig F, et al. A highly parameterizable parallel processor array architecture [C]// IEEE International Conference on Field Programmable Technology. Thailand, 2006:105-112.

[6] 李焯挺,梁利平. 一种基于 SystemC 的片上网络建模与仿真方法[J]. 微电子学与计算机,2010,27(3):78-82.

[7] 奚杰,陈杰,朱玥. 利用 SystemC 实现多核系统的快速建模[J]. 微电子学与计算机,2010,27(7):214-217.

[8] Rigo S, Araujo G, Bartholomeu M, et al. Archc: a systemC-based architecture description language [C]// 16th Symposium on Computer Architecture and High Performance Computing SBAC-PAD. Brazil, 2004:66-73.

作者简介:

潘 鹏 男,(1986-),硕士研究生. 研究方向为可重构计算、事务级建模等。

王 鹏 男,(1985-),硕士研究生. 研究方向为嵌入式系统设计、可重构计算等。

林水生 男,教授. 研究方向为嵌入式系统、通信专用集成电路与 SoC 设计。