

多 FPGA 设计的时钟同步

宋 威^{1,2}, 方穗明¹, 姚 丹², 张立超², 钱 程²

(1. 北京工业大学电子信息与控制工程学院, 北京 100022; 2. 北京工业大学北京市嵌入式系统重点实验室, 北京 100022)

摘 要: 在多 FPGA 设计中, 时钟信号的传输延时造成了 FPGA 间的大时钟偏差, 进而制约系统性能。为减少时钟偏差, 该文提出一种多数字延迟锁相环(DLL)电路。该电路将时钟的传输电路放入 DLL 的反馈环路。利用 DLL 的延迟锁定特性, 对 FPGA 间的时钟传输延时进行补偿, 减少 FPGA 间的时钟偏差, 解决多 FPGA 的时钟同步问题。

关键词: 现场可编程逻辑门阵列; 时钟偏差; 延迟锁相环

Clock Synchronization in Multi-FPGA Designs

SONG Wei^{1,2}, FANG Sui-ming¹, YAO Dan², ZHANG Li-chao², QIAN Cheng²

(1. College of Electronic Information & Control Engineering, Beijing University of Technology, Beijing 100022;

2. Beijing Embedded System Key Lab, Beijing University of Technology, Beijing 100022)

【Abstract】 In multi-FPGA designs, the delay of clock transfer causes a huge clock skew between FPGAs and therefore undermines the system performance. To decrease this skew, a circuit based on Delayed Locked Loop(DLL) circuits is proposed. This circuit links the clock transfer path into the feedback loop of DLLs. Thanks to the delayed locked feature, the delay of clock transfer is compensated. Therefore, this circuit reduces the skew between FPGAs and solves the synchronization in multi-FPGA designs.

【Key words】 FPGA; clock skew; Delayed Locked Loop(DLL)

时钟源到所有由该时钟驱动时钟引脚的最大延时之差称为时钟偏差。为了解决 FPGA 的时钟偏差问题, FPGA 内部同时并存几棵全局时钟树。寄存器通过可配置开关可以和任意一棵时钟树相连。并且, 为了最大程度地减少同一棵时钟树的时钟偏差, 几乎所有的 FPGA 都采用对称梳状的时钟树结构。根据文献[1]中的验证结果, 电子辅助设计(EDA)工具, 通过寄存器在时钟树上的平衡分布, 可将该时钟树内的时钟偏差减小到可容忍的范围。根据文献[2], EDA 工具缺少识别门控时钟和分频时钟的能力, 不能很好地解决同源但不同时钟树时钟之间的时钟偏差问题。然而, 通过合理利用 FPGA 内部的特殊时钟资源, 还是能将多时钟树之间的时钟偏差减小到可忽略的程度, 即达到片内时钟同步。尽管合理利用特殊时钟资源和 EDA 工具, 已经可以解决 FPGA 内部时钟同步的问题, 但是多 FPGA 的时钟同步尚待解决。

1 FPGA 的时钟结构

如前所述, FPGA 内部通过单独的一层金属布线层和相应的可配置开关, 构成一种接近中心散射的对称梳状时钟树。如图 1 所示。

所有的时钟信号都从时钟原点处的时钟缓冲器进入时钟布线层, 然后通过中央垂直方向的时钟主干, 水平方向的多条时钟枝干和局部时钟线, 最终和分布在 FPGA 内部的寄存器相连, 形成一棵时钟树。在这种结构下, 时钟原点到寄存器的时钟路径都由时钟主干、时钟枝干和局部时钟线构成。3 种连线的连接都由处于连接处的可配置开关控制。因此, 从时钟原点到每一个寄存器都经过了大致相同的连线路径和相同个数的可配置开关。再配合寄存器的平衡分布, FPGA 就能保证在同一棵时钟树上的时钟偏差较小。本文使用的 Xilinx

的 VIRTEX-4 LX100 FPGA^[3] 同时提供 16 棵时钟树。通过测试, 时钟树的最短延时为 2.5 ns, 最长延时为 3.6 ns, 最大时钟偏差为 0.9 ns。

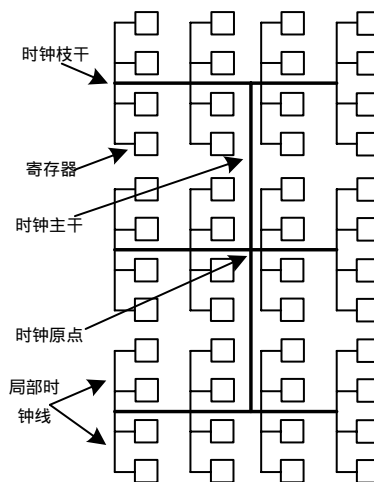


图 1 梳状时钟树结构

2 FPGA 的同步问题分析

多 FPGA 的时钟同步问题可简化为如图 2 所示的电路。

基金项目: 北京市嵌入式系统研究与开发计划基金资助项目(D03040040111)

作者简介: 宋 威(1983 -), 男, 硕士研究生, 主研方向: 大规模集成电路设计与 FPGA 的原形实现; 方穗明, 副教授; 姚 丹、张立超、钱 程, 硕士研究生

收稿日期: 2007-04-09 E-mail: benjaminweber@emails.bjut.edu.cn

当然，可以用一个时钟晶振驱动 FPGA1 和 FPGA2。但是多负载将直接导致时钟质量下降。同时，为了降低时钟偏差，必然采用板上对称布线，因而不能达到晶振芯片尽量靠近 FPGA 的要求，导致时钟信号容易被干扰。所以，在较高时钟频率的应用下，仍应采用图 2 中的时钟传递的方式。

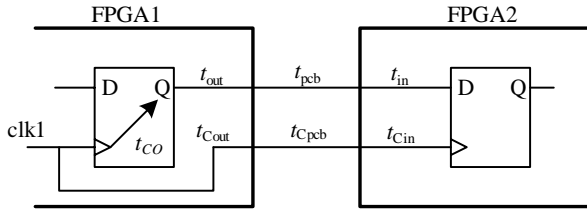


图 2 FPGA 间的数据通路简化图

定义 t_{CO} 为寄存器门延时； t_{out} 和 t_{Cout} 分别为寄存器的输出端和时钟树末端到 FPGA 输出脚的传输延时； t_{pcb} 和 t_{Cpcb} 分别为数据通路和时钟通路在电路板上的传输延时； t_{in} 和 t_{Cin} 分别为 FPGA 输入引脚到寄存器数据输入端和时钟端的传输延时。此外，再定义寄存器的建立时间为 t_{setup} ，寄存器的保持时间为 t_{hold} ，时钟周期为 T ，FPGA1 到 FPGA2 为正方向。

那么，FPGA 间正向通信的必要条件为

$$t_{Cout} + t_{Cpcb} + t_{Cin} + t_{hold} < t_{CO} + t_{out} + t_{pcb} + t_{in} < t_{Cout} + t_{Cpcb} + t_{Cin} + T - t_{setup} \quad (1)$$

该式可简单理解为数据的传输延时大于正时钟偏差同时小于时钟周期减去正时钟偏差。

根据在 Xilinx 的 VIRTEX-4 FPGA LX100 FF1512-10^[3] 上的实际试验测试报告和 VIRTEX-4 FPGA 的转换特性^[4]，式(1)中的相关变量取值范围如下：

$$\begin{aligned} t_{Cout} &= 3.5 \pm 0.5 \text{ ns}; t_{Cin} = 6.5 \pm 1.7 \text{ ns}; t_{CO} = 0.5 \pm 0.2 \text{ ns}; \\ t_{out} &= 3.5 \pm 0.5 \text{ ns}; t_{in} = 1.6 \pm 0.5 \text{ ns}; t_{hold} = 0.25 \pm 0.15 \text{ ns}; \\ t_{setup} &= 0.25 \pm 0.15 \text{ ns}; t_{Cpcb} = 0.85 \pm 0.35 \text{ ns}; \\ t_{pcb} &= 0.85 \pm 0.35 \text{ ns} \end{aligned}$$

考虑到工作温度、器件的分散参数和扇入扇出不同，取值为一个范围。具体来说， t_{in} 和 t_{out} 根据寄存器在 FPGA 内位置不同有很大变化。 t_{Cin} 的取值决定于时钟输入脚位置、时钟树的延时和时钟树自身的时钟偏差。 t_{Cpcb} 和 t_{pcb} 则决定于实际 PCB 的布线结果。在 Dini Group 的 DN8000K10PCI^[5] 原型板上，FPGA 间的连线延时 t_{Cpcb} 和 t_{pcb} 约为 $0.85 \text{ ns} \pm 0.35 \text{ ns}$ 。

综合以上的数据和式(1)，可以得到：

$$11.1 \pm 2.7 < 6.45 \pm 1.55 < T + 10.6 \pm 2.7 \quad (2)$$

明显可得，式(2)并不成立。数据延时小于正时钟偏差，违反了保持时间要求。另外，图 2 只考虑了从 FPGA1 到 FPGA2 之间的正向数据通路。如果反方向传输，式(1)应被改写为

$$\begin{aligned} -t_{Cout} - t_{Cpcb} - t_{Cin} + t_{hold} < t_{CO} + t_{out} + t_{pcb} + t_{in} < \\ -t_{Cout} - t_{Cpcb} - t_{Cin} + T - t_{setup} \end{aligned} \quad (3)$$

将各变量的取值代入式(3)，得到：

$$-10.6 \pm 2.7 < 6.45 \pm 1.55 < T - 11.1 \pm 2.7 \quad (4)$$

为了保证式(4)中的第 2 个小于号成立，即满足建立时间要求，时钟周期应当满足：

$$T > 17.55 \pm 4.25 \quad (5)$$

由式(2)和式(5)可得出结论：基于 VIRTEX-4 系列 FPGA 和图 2 的简单时钟同步方式，正向通信不满足寄存器的保持时间要求，而反向通信的最高时钟频率为 $46 \text{ MHz} \sim 75 \text{ MHz}$ 。

3 正向通信的时钟同步

分析各时间的取值范围，可发现 t_{Cin} 远大于 t_{in} 。片内对称

梳状时钟树减小了同时钟树上的时钟偏差，但对所有片内器件的时钟端增加了一个群延时。这个群延时由时钟 I/O 的门延时，I/O 到时钟原点的连线延时，位于时钟原点的时钟缓冲器的门延时和时钟树延时共同组成。显然这个群延时要远大于简单连接的数据通路的输入延时。如果能将 t_{Cin} 减少到 t_{in} 的数值，则能缓解保持时间错误的问题。

Xilinx VIRTEX-II 之后的 FPGA 中，FPGA 内部提供了内置 DLL 的电子时钟管理单元(DCM)。

通过在 FPGA2 内的时钟输入路径上使用 DCM，如图 3 所示，将时钟树末端接入反馈端(FB)，DCM 内部的 DLL 将对时钟路径进行动态延时补偿，使反馈端和输入端同相位。由于 DCM 的反馈端为时钟树的末端，与寄存器时钟端的时钟偏差小于同时钟树上的最大时钟偏差，因而寄存器的时钟端和时钟输入 I/O 之间的延时被等效为 I/O 延时和时钟偏差之和。实际测算得到在使用 DCM 之后 t_{Cin} 为 $1.9 \text{ ns} \pm 0.3 \text{ ns}$ 。这样，FPGA2 内部的时钟延时就被大大削减。

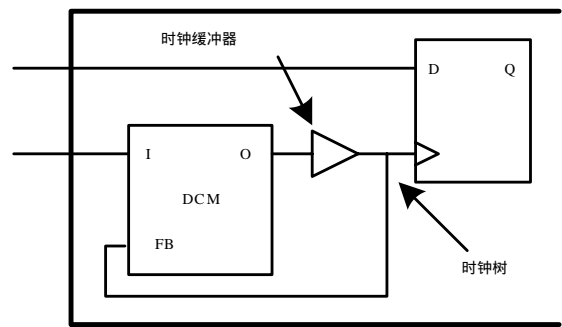


图 3 FPGA 锁相环的连接方式

然而，这并没有完全解决正向通信的保持时间问题。尽管时钟延时减少，但并不能保证数据延时大于时钟延时。Xilinx 的所有 FPGA 在 I/O 内部内置了延时模块(IOBDelay)。该延时模块可以对输入信号延时 $0 \text{ ns} \sim 5 \text{ ns}$ ，具体时间可配置。其具体设定方法请参考文献[3]。

将使用 DCM，并配置输入延时增加 5 ns 之后的 t_{Cin} 和 t_{in} 代入式(1)和式(3)可以得到式(6)和式(7)：

$$6.5 \pm 1.3 < 11.45 \pm 1.55 < T + 6 \pm 1.3 \quad (6)$$

$$-6 \pm 1.3 < 11.45 \pm 1.55 < T - 6.5 \pm 1.3 \quad (7)$$

从式(6)可得，正向数据通信的保持时间问题已解决。通过式(7)估算，反向数据通信的最高频率为 $48 \text{ MHz} \sim 66 \text{ MHz}$ ，并没有明显改善。

4 全局时钟同步

尽管在 FPGA2 内部使用 DCM 减少了时钟的输入延迟，但是并不能明显提升系统的最高时钟频率。当前 FPGA 的最高工作频率可达 400 MHz ，显然最大支持 66 MHz 并不能满足高性能系统的要求。

再考虑式(3)，其中 $-t_{Cout} - t_{Cpcb} - t_{Cin}$ 为负时钟偏差。对比 FPGA 内部，由于时钟树的作用，寄存器间的时钟偏差极小，因此最高时钟频率由最大数据路径延时决定。而在 FPGA 间通信时，时钟的传递导致了不同 FPGA 的时钟有一个大的时钟偏差。

如果将图 2 中 FPGA2 的时钟树看成 FPGA1 时钟树的子树，FPGA1 的时钟树为父树。子树原点和父树末端的延时则为 FPGA 间的通信延时，那么父树的时钟树延时和 FPGA 间通信延时之和等效为 FPGA 间的时钟偏差。如果能将这个时钟偏差消除，FPGA 间通信的最高时钟频率将由数据通路的

延时决定。

显然,最直接的办法是 FPGA1 和 FPGA2 共用时钟原点,则 FPGA 间的时钟偏差为时钟树的时钟偏差。如第 2 节所述,同时钟树上的时钟偏差最大为 0.9 ns,远远小于时钟传递所造成的时钟偏差。

但是,并不能用同一个晶振驱动 2 个 FPGA,其原因在第 2 节已经说明。所以,还是采用时钟传递,但可以让 FPGA1 的时钟主动和 FPGA2 同步。其具体的电路结构如图 4 所示。

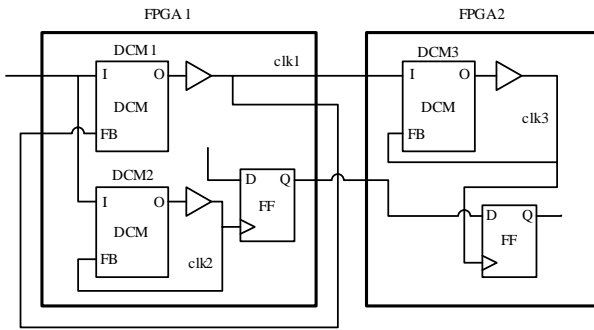


图 4 全局时钟同步的时钟树结构

晶振产生的时钟信号首先进入 FPGA1 的 DCM1,其输出时钟通过 2 个 I/O 分别传递给 FPGA2 的 DCM3 和自身的时钟反馈端。因此,DCM1 的时钟反馈是一个板级反馈。FPGA2 内部的寄存器由 DCM3 产生的 clk3 驱动。FPGA1 内部的 DCM2 使用片内 DCM1 之前的时钟,产生 clk2 作为内部寄存器真正使用的时钟。

这样,FPGA1 和 PGFA2 的寄存器分别使用 clk2 和 clk3。clk2 和 clk3 之间的时钟偏差即 FPGA 间的时钟偏差。分别计算 clk2 和 clk3 相对晶振时钟输入的延时 t_{clk2} 和 t_{clk3} 。

$$t_{clk2} = t_{Cin} + t_{DCM2} + t_{tree2} \quad (8)$$

$$t_{clk3} = t_{Cin} + t_{DCM1} + t_{tree1} + t_{Cout} + t_{Cpcb} + t_{Cin} + t_{DCM3} + t_{tree3} \quad (9)$$

根据 DCM 的延迟锁定原理可知:

$$t_{DCM1} + t_{tree1} + t_{Cout} + t_{Cpcb} + t_{Cin} = nT (n \in \mathbb{N}) \quad (10)$$

$$t_{DCM2} + t_{tree2} = nT (n \in \mathbb{N}) \quad (11)$$

$$t_{DCM3} + t_{tree3} = nT (n \in \mathbb{N}) \quad (12)$$

简单化简得到 DCM 的传输延时:

$$t_{DCM1} = nT - t_{tree1} - t_{Cout} - t_{Cpcb} - t_{Cin} (n \in \mathbb{N}) \quad (13)$$

$$t_{DCM2} = nT - t_{tree2} (n \in \mathbb{N}) \quad (14)$$

$$t_{DCM3} = nT - t_{tree3} (n \in \mathbb{N}) \quad (15)$$

代入式(8)、式(9):

$$t_{clk2} = t_{Cin} + (nT - t_{tree2}) + t_{tree2} = t_{Cin} + nT (n \in \mathbb{N}) \quad (16)$$

$$t_{clk3} = t_{Cin} + (nT - t_{tree1} - t_{Cout} - t_{Cpcb} - t_{Cin}) + t_{tree1} + t_{Cout} + t_{Cpcb} + t_{Cin} + (nT - t_{tree3}) + t_{tree3} = t_{Cin} + nT + nT = t_{Cin} + nT (n \in \mathbb{N}) \quad (17)$$

由式(16)和式(17)可以发现,FPGA1 和 FPGA2 的实际时钟 clk2 和 clk3 相对时钟输入来说延时相等。

分析电路结构,DCM2 和 DCM3 分别为 FPGA1 和 FPGA2 的局部时钟延迟补偿电路。DCM1 将 FPGA1 的时钟 clk1、时钟输出 I/O、电路板上的时钟反馈线和时钟输入 I/O 放入反馈

环路内,从而补偿了因上面 4 个因素所造成的时钟偏差,而这正好近似于时钟传递所造成的时钟偏差。因此 DCM1 将 DCM3 的时钟输入端等效为 DCM2 的时钟输入端,从而消除了 FPGA 间的时钟偏差。

当然,这个结论需要建立在 clk1、clk2 和 clk3 的时钟树延时相等,DCM1 的板级反馈线和 FPGA1 到 FPGA2 的时钟传递线延时相等的条件之上。根据在第 3 节给出的数据,这些因素造成的实际时钟偏差在 ± 4.7 ns 范围内。利用式(3)并用 ± 4.7 ns 代替 $-t_{Cout} - t_{Cpcb} - t_{Cin}$ 可得:

$$0.25 \pm 4.85 < 6.45 \pm 1.55 < T - 0.25 \pm 4.85 \quad (18)$$

经计算,该方法最大时钟频率范围为 76 MHz~3.3 GHz。由于 ± 4.7 ns 是对该方法造成的时钟偏差的最坏估计,计算得到的频率的下限 72 MHz 还是相对较低。如果合理使用该方法,一般能将时钟偏差控制在 1 ns 以内,那么时钟频率范围为 102 MHz~278 MHz。另外,如果将时钟偏差带入式(1)还可发现,随着时钟偏差的消除,正向通信时的时钟保持错误也随之解决。

该方法的电路结构可利用 Xilinx 的 Architecture Wizard 工具协助产生,具体参看 Xilinx 的 FPGA 开发平台 ISE 8.1 的相关帮助文档与参考文献[3]。此外,该方法可简单扩展到多 FPGA 的设计当中,每一个新添的 FPGA 相当于图 4 中的 FPGA2 即可。并且,此方法也可用于基于 PLL 的 Altera 的 FPGA 当中,效果类似。

5 结束语

在 DN8000K10PCI^[5] 平台实验上,该方法实现了 3 个 VIRTEX-4 FPGA(LX100 FF1515 -10, LX200 FF1513 -10 和 FX60 FF1152 -10)之间的时钟同步。实际测量得到,FPGA 间的时钟偏差小于 1 ns。同时,该方法已经使用在北京市嵌入式系统重点实验室的美国数字电视(ATSC)解调芯片的 FPGA 原形验证项目中,实现了 FPGA 间 100 MHz 的时钟同步。

针对 FPGA 间数据通信的具体问题,本文通过分析数据通路和时钟通路的延时,详细说明了该通信遇到的问题。并提出了一种多 FPGA 的全局时钟同步的方法。该方法可用于任何在 FPGA 内集成了 DLL 或者 PLL 的商用 FPGA 当中,具有较高的实际使用价值。

参考文献

- [1] Zhu Kai, Wong D F. Clock Skew Minimization During FPGA Placement[J]. IEEE Transactions on Computer-aided Design, 1997, 16(4): 376-385.
- [2] 宋威,方穗明.基于 BUFGMUX 与 DCM 的 FPGA 时钟电路设计[J].现代电子技术,2006,29(2): 141-143.
- [3] Xilinx 公司. Virtex-4 User Guide[EB/OL]. (2007-01-04). <http://www.xilinx.com>.
- [4] Xilinx 公司. Virtex-4 Data Sheet: DC and Switching Characteristics [EB/OL]. (2006-12-11). <http://www.xilinx.com>.
- [5] Dini Group. DN8000K10PCI User Manual[EB/OL]. (2006-08-08). <http://www.dinigroup.com>.