

## 一种低功耗时钟树综合的寄存器聚类方法

唐俊龙<sup>1†</sup>, 卢英龙<sup>1</sup>, 戴超雄<sup>1</sup>, 邹望辉<sup>1</sup>, 李振涛<sup>2</sup>

(1. 长沙理工大学 物理与电子科学学院, 湖南 长沙 410114;

2. 湖南毅梁微电子有限公司, 湖南 长沙 410003)

**摘要:**随着集成电路制造工艺的进步与芯片集成度的提升,对于低功耗芯片的需求越来越大.时钟网络功耗占芯片总功耗的 40% 以上,优化时钟网络的功耗已成为高性能集成电路设计中最重要目标之一.本文提出了一种新的寄存器聚类方法来生成时钟树的叶级拓扑结构,通过限制群组的扇出、负载和范围,对寄存器进行合理分组,减少了缓冲器的插入数目和总布线长度,有效降低时钟网络功耗.将该方法整合到传统的时钟树综合(CTS)流程中,在 ISCAS89 基准电路上测试并分析其有效性.实验结果表明,该寄存器聚类方法在不影响时钟树最大延时的情况下,有效减少了时钟网络 20% 以上的功率耗散和 20% 以上的时钟偏移.

**关键词:**低功耗电子;时钟偏移;寄存器聚类;时钟树综合

**中图分类号:**TN402 **文献标志码:**A

## A Register Clustering Method for Low-power Clock Tree Synthesis

TANG Junlong<sup>1†</sup>, LU Yinglong<sup>1</sup>, DAI Chaoxiong<sup>1</sup>, ZOU Wanghui<sup>1</sup>, LI Zhentao<sup>2</sup>

(1. School of Physical and Electronic Sciences, Changsha University of Science and Technology, Changsha 410114, China;

2. Hunan Great-Leo Microelectronics Co. Ltd., Changsha 410003, China)

**Abstract:** With the advancement of integrated circuit manufacturing technology and the improvement of chip integration, the demand for low-power chips has been steadily increasing. The clock network is responsible for more than 40% of the total power consumption of the chip. Consequently, optimizing the power consumption of the clock network has become one of the most important goals in the design of high-performance integrated circuits. In this paper, a new register clustering method is proposed to generate the leaf level topology of the clock tree. By carefully limiting the fan-out, load, and range of the clusters to reasonably group the registers, the method effectively reduces the number of buffer insertions and the total wiring length, and the clock network power consumption is also significantly reduced. The method is integrated into the traditional clock tree synthesis (CTS) flow, and its effectiveness is tested and analyzed on the ISCAS89 benchmark circuit. Experimental results show that the register clustering method effectively reduces the power dissipation of the clock network by more than 20% and the clock offset by more than 20%, without affecting the maximum delay of the clock tree.

**Key words:** low power electronics; clock skew; register clustering; clock tree synthesis

\* 收稿日期:2022-09-30

基金项目:柔性电子材料基因工程湖南省重点实验室开放基金(202015), Open Research Fund of Hunan Provincial Key Laboratory of Flexible Electronic Materials Genome Engineering (202015)

作者简介:唐俊龙(1973—),男,湖南武冈人,长沙理工大学副教授,博士

† 通信联系人, E-mail: tangjl625@163.com

随着 VLSI 技术中集成度越来越高,时钟频率越来越快,功率消耗已经成为现代 IC 设计中的一个关键问题.动态功耗是指在对负载电容充放电过程中的功率耗散,占总功率消耗的最大部分,而在所有的电路元件中,时钟网络是动态功耗的主要贡献者,通常要消耗整个芯片功率预算的 40%<sup>[1]</sup>.因此,在数字集成电路的物理设计中,低功耗时钟树的设计具有重要意义.时钟树的负载电容由互连电容、寄存器电容和缓冲器电容组成.大部分的时钟树电容(大约 80%)都在叶级,包括所有的寄存器和连接它们的导线以及驱动缓冲器<sup>[2]</sup>.文献[3]对八个 ISPD'10 基准电路的时钟树电容进行了简单的统计,缓冲器的电容是时钟树总电容的主要贡献者,减少叶级缓冲器电容能有效减少时钟树负载电容.

聚类算法可以把寄存器聚成几个组群,并为每个组群分配一个时钟缓冲器,因此寄存器聚类方法能有效降低叶级缓冲器的电容.为了有效减少时钟网络的功耗和时钟偏移,提出了一种高效的寄存器聚类方法.在满足叶级缓冲器的最大负载电容和最大扇出约束的条件下,将距离范围内的寄存器聚集成组群,生成时钟树的叶级拓扑结构.将这种高效的寄存器聚类方法集成到传统的 CTS 流程中的实验结果表明,该方法有效减少了时钟网络的功耗和时钟偏移.

## 1 寄存器聚类方法的问题描述

在时钟树综合过程中,将寄存器聚成一个个组群的难点在于避免由寄存器分组所引起的时钟信号偏移和延迟<sup>[4]</sup>.文献[5]通过计算寄存器的时序可行域,将寄存器移动到时序可行域重叠部分,在降低功耗的前提下,减小对时序的影响.然而,芯片上不仅存在时钟网,还存在信号网,寄存器的移动往往会对信号网布线长度产生影响,造成物理设计过程中出现布线拥堵的问题.文献[6]在分析近阈值电压下时钟树设计问题的基础上,提出了时序驱动且负载均衡的寄存器分组方案,增强了时钟树时序的稳定性,但该方法没有考虑到时钟树的功率耗散.文献[7-8]提出了一种最小线长聚类算法,用于低功率时钟树合成,基于曼哈顿距离求出最小树拓扑结构,然后设置阈值,确立合理的组群数目和缓冲器插入位置,最后通过经典的 K-means 聚类算法迭代出最小线长的

组群,但没有讨论每个组群之间负载不均衡对时钟偏移的影响.

图 1 表示了简单的时钟网络,其中  $R_0$  表示时钟源驱动器的等效电阻, $R_1$  和  $R_2$  表示两个寄存器组群的叶级缓冲器的等效电阻, $R_1 = R_2$ ;  $C_i$  代表不同分支上的负载电容.

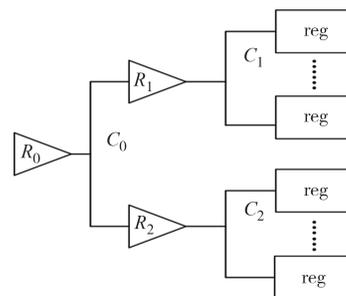


图 1 时钟网络

Fig.1 Clock network

根据 Elmore 延迟公式计算不同路径的延迟如公式(1)和公式(2)所示<sup>[9]</sup>;计算不同分支路径的时钟偏差(skew) $S$ ,如公式(3)所示.

$$t_1 = R_0(C_0 + C_1 + C_2) + R_1 C_1 \quad (1)$$

$$t_2 = R_0(C_0 + C_1 + C_2) + R_2 C_2 \quad (2)$$

$$S = t_1 - t_2 = R_1(C_1 - C_2) \quad (3)$$

由公式(3)可知组群之间负载电容的差值直接影响时钟偏移,所以在寄存器分组中,平衡不同组群之间的负载是十分必要的<sup>[10]</sup>.

对于一个给定的布局,寄存器的电容是确定的.整个时钟树的缓冲器电容由几个因素决定,包括缓冲器驱动的负载电容、时钟树综合流程优化期间的时钟偏移和最大转换时间约束等<sup>[11]</sup>.在传统的时钟树综合过程中,为了调整寄存器的延迟,减少时钟偏移,避免违反转换时间约束,通常需要插入大量的缓冲器到时钟树的叶层.高质量的寄存器聚类方法可以以最佳方式聚集寄存器,减少叶层的缓冲器插入数量,同时减少互连线电容,将局部时钟偏移限制在小范围内的同时,有效降低时钟网络的功耗<sup>[12]</sup>.寄存器聚类问题通常可以描述为:

输入:寄存器集合  $U_{\text{register}} = \{r_1, r_2, r_3, \dots, r_n\}$ , 每个寄存器  $r_i$  包括寄存器名和坐标位置  $(x_{r_i}, y_{r_i})$ .

输出:负载均衡的组群的集合  $U_{\text{cluster}} = \{c_1, c_2, c_3, \dots, c_m\}$ , 每个组群  $c_i$  包括组群内寄存器数、寄存器名和组群的中心  $ce_i$ ,  $ce_i$  的坐标  $(x_{ce_i}, y_{ce_i})$  计算公式如公式(4)、公式(5)所示.

$$x_{c_i} = (\sum_{j=1}^p x_{r_j})/p, r_j \in c_i (j = 1, \dots, p) \quad (4)$$

$$y_{c_i} = (\sum_{j=1}^p y_{r_j})/p, r_j \in c_i (j = 1, \dots, p) \quad (5)$$

且  $c_i$  和  $c_j$  满足条件式(6)~式(8):

$$c_i \neq \emptyset, (i = 1, \dots, m) \quad (6)$$

$$c_i \cap c_j = \emptyset, i, j \in (i = 1, \dots, m), i \neq j \quad (7)$$

$$U_{i=1}^m c_i = U_{\text{register}} \quad (8)$$

## 2 寄存器聚类方法的设计

文中提出一种简单、高效的寄存器聚类方法. 关键思想是在满足叶级缓冲器最大负载电容和最大扇出的同时, 通过设置最大距离, 限制组群半径, 为每个组群定义合理的范围, 使得距离组群中心较远的寄存器在聚类过程中不加入该组群内. 避免了组群内布线过长, 造成的叶级互连线电容过大的问题<sup>[13]</sup>. 寄存器分组示意图如图2所示.

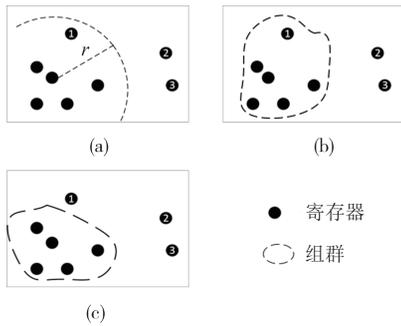


图2 寄存器分组示意图

Fig.2 Register grouping schematic

图2(a)表示以组群中心为圆心, 半径  $r$  为所设定的距离阈值, 虚线半圆为组群的聚类范围; 图2(b)中, 将满足距离约束, 同时满足组群负载和扇出约束的寄存器加入组群内, 不满足距离约束的寄存器2和3不加入该组群, 在下次遍历时加入新的组群; 图2(c)表示在满足距离约束, 而不满足组群扇出或负载约束时, 选取距离中心最远的寄存器1去除该组群, 在下次遍历时加入新的组群.

组群内的电容作为一个重要约束可以直接影响到时钟信号的转换时间. 保证时钟转换时间满足寄存器时钟端口的约束条件, 有利于时序收敛, 确保寄存器对数据通路正确采样<sup>[14]</sup>. 叶级缓冲器直接将时钟信号驱动到组群内的寄存器, 为了保证组群内寄

存器接收到满足转换时间约束的时钟信号, 叶级缓冲器的最大负载电容与组群内的电容之和应满足约束条件式(9), 单个组群内的负载电容之和  $C_{\text{cluster}}$  可以用公式(10)表示<sup>[15]</sup>.

$$C_{\text{cluster}} < C_{\text{max}} \quad (9)$$

$$C_{\text{cluster}} = \alpha \sum_{n=1}^N C_{r_n} + \beta \sqrt{N} D \quad (10)$$

式中:  $C_{r_n}$  是寄存器时钟端口的本征输入电容;  $D$  表示组群的直径, 定义为组群内最远的两个寄存器之间的曼哈顿距离;  $N$  是组群内的寄存器数量;  $\alpha, \beta$  为自定义参数, 用以调整组群内寄存器电容与互连线电容比重关系, 文中定义  $\alpha, \beta$  均等于1.  $C_{\text{max}}$  为所选时钟单元的最大负载电容, 可直接从工艺库时钟缓冲器参数说明中读取出来<sup>[16]</sup>.

本文设计的寄存器聚类方法, 另外两个重要的约束是组群的半径和组群内最大寄存器数目. 组群半径定义为寄存器与所在组群中心之间的最大距离  $d_{\text{max}}$ , 如公式(11)所示, 根据定义,  $d_{\text{max}}$  等于  $D/2$ . 结合公式(10), 组群内最大寄存器数目  $N_{\text{max}}$  如公式(12)所示.

$$d_{\text{max}} = \alpha \sqrt{\frac{WL - S}{N_{\text{all}}/N_{\text{max}}}} \quad (11)$$

$$N_{\text{max}} = \left( \frac{C_{\text{max}} - \alpha \sum_{n=1}^N C_{r_n}}{2\beta d_{\text{max}}} \right)^2 \quad (12)$$

公式(11)中,  $W$  和  $L$  分别是布局的宽度和长度,  $S$  是布局上障碍物的总面积,  $N_{\text{all}}$  是寄存器的总数. 公式(12)中,  $C_{\text{max}}$  是所选时钟单元的最大负载电容,  $C_{r_n}$  是寄存器时钟端口的本征输入电容,  $N$  是组群内的寄存器数量,  $\alpha, \beta$  为定义参数. 算法流程图如图3所示.

首先, 该算法通过添加  $r_1$  来生成第一个组群  $c_1$ , 定义此时组群的负载  $L_{c_1}$ , 组群的扇出为  $F_{c_1}$ , 计算  $c_1$  的中心, 组群半径  $d_{\text{max}}$  和最大扇出  $N_{\text{max}}$ . 然后, 对所有剩余的寄存器进行遍历. 对于每个寄存器  $r_i$ , 找到距离最近的组群  $c_{\text{near}}$ , 且  $r_i$  与  $c_{\text{near}}$  必须满足最大距离约束、缓冲器最大负载约束和最大扇出约束, 如果存在一个  $c_{\text{near}}$  满足条件, 则将  $r_i$  加入  $c_{\text{near}}$ , 更新  $c_{\text{near}}$  的中心、扇出和组群内的电容. 如果没有组群满足约束条件, 则创建一个新的组群  $c_j$ , 定义  $c_j$  的中心、扇出和负载, 然后加入组群集合  $U_{\text{cluster}}$  中, 最后更新  $U_{\text{cluster}}$ .

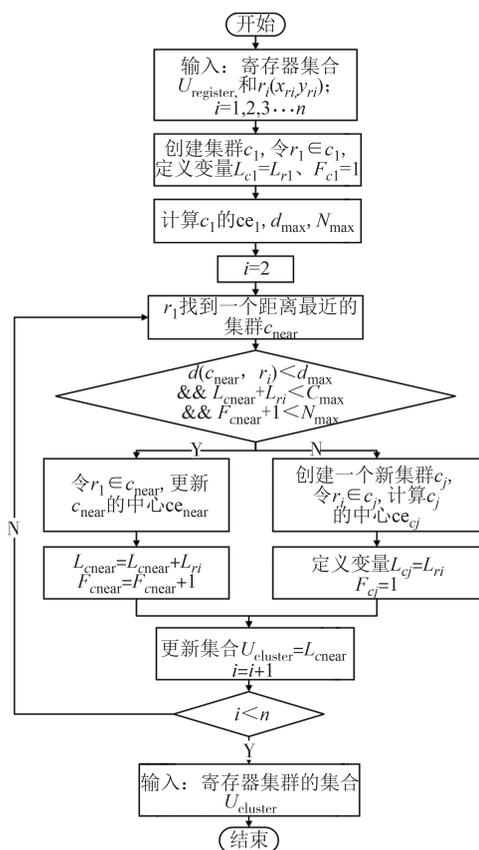


图3 寄存器聚类方法流程图

Fig.3 Flow chart of register clustering method

### 3 实验结果与分析

本文研究的低功耗时钟树设计基于寄存器分组进行时钟树拓扑结构的构建,降低时钟网络功耗的同时,尽可能地减小寄存器分组对时钟树时序性能带来的影响.提出的寄存器聚类方法实验环境在Linux操作系统下,使用C语言实现,在相同的环境下,实现文献[7]所描述的寄存器聚类方法.作为一个给寄存器分组的过程,将两种方法分别集成到传统的CTS流程中,基于110 nm物理工艺库GRACE11 LP,在被广泛应用于时钟树综合结果验证的ISCAS89基准电路上进行仿真验证<sup>[17]</sup>,对比集成两种不同寄存器聚类方法的时钟树综合结果,并对不同情况下时钟树的功率耗散和时序性能进行对比分析.

实验从ISCAS89基准电路中选择了S13207、S15850、S38417三个规模不同的单时钟同步电路结构,通过Design Compiler逻辑综合工具,将电路的硬

件描述语言形式翻译、优化并映射成对应的电路网表形式,然后通过Innovus后端布局布线工具,实现所选用电路的时钟树综合等后端物理设计,将电路网表实现成可用于掩膜制造的GDS II版图.本次实验,定义时钟信号clk\_period为5 ns, input\_delay和output\_delay为0.1 ns,选用电路综合和布局后的信息如表1所示.

在Innovus下采用集成两种不同寄存器聚类方法的时钟树设计流程,对表1中三个基准电路进行时钟树综合,然后调用StarRC和Prime Time,分别进行寄生参数提取和静态时序分析,分析不同情况下生成的时钟树的功率耗散和时序性能,如表2所示,其中skew表示全局最大时钟偏移(global skew), latency代表最长路径延时, power表示时钟网络的功耗.

表1 基准电路信息

Tab.1 Benchmark circuit information

基准电路	寄存器数目	逻辑单元个数	布局面积/(nm <sup>2</sup> )
S13207	202	432	84.32×83.54
S15850	442	1 529	145.56×144.85
S38417	1 462	5 037	265.26×264.84

表2中,对比集成了文献[7]方法的时钟树综合结果,文中提出的寄存器聚类方法在基准电路S13207、S15850和S38417上,时钟网络的功耗分别降低了20.3%、23.8%和27.8%,时钟偏移分别减少了20.8%、23.6%和21.7%.最大延迟的偏差基本维持在50 ps的范围内,除在规模相对较小的S13207基准电路上有小幅度增加以外,其余两个基准电路下latency均有所优化.实验数据表明,本文研究设计的寄存器聚类方法,通过对时钟汇点进行更为有效的分组,合理增加并均衡了组群内的寄存器数目,减少了中心缓冲器的插入数目,在不影响时钟网络最大延时的条件下,有效减少了时钟网络的功率耗散和时钟偏移.

图4和图5显示了Innovus图形界面下,文献[7]寄存器聚类方法和文中寄存器聚类方法在基准电路S38417上生成的时钟树叶级拓扑结构,图中的每个树状拓扑结构都代表一个组群.

表 2 时钟树综合结果对比

Tab.2 Comparison of comprehensive results of clock trees

基准电路	聚类方法	缓冲器插入数目	分组数	Skew/ns	Latency/ns	Power/mW	功耗优化/%	Skew 优化/%
S13207	文献[7]	26	23	0.072	0.354	1.389	20.3	20.8
	本文	19	16	0.057	0.361	1.106		
S15850	文献[7]	60	56	0.093	0.508	2.646	23.8	23.6
	本文	37	33	0.069	0.471	2.016		
S38417	文献[7]	203	189	0.101	0.578	6.134	27.8	21.7
	本文	127	118	0.079	0.558	4.468		

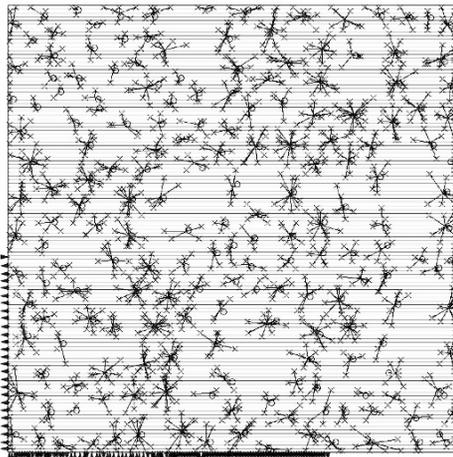


图 4 文献[7]聚类方法分组结果

Fig.4 Results of the clustering method grouping in the literature [7]

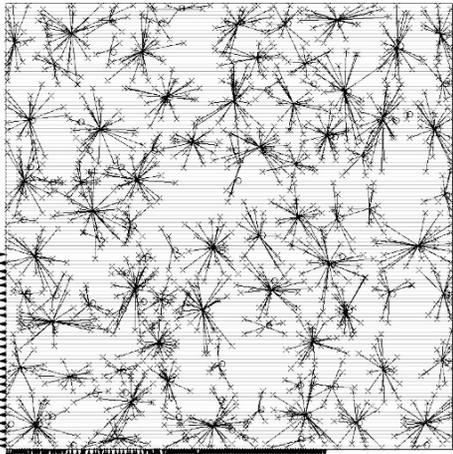


图 5 文中聚类方法分组结果

Fig.5 Results of clustering method grouping in the text

## 4 结论

高性能集成电路芯片设计中,时钟网络功耗在芯片总功耗中的占比较高,为了优化时钟网络功耗,本文提出了一种简单、高效的寄存器聚类方法.在满

足叶级缓冲器最大负载电容和最大扇出约束的条件下,通过设置距离阈值控制组群的半径,将有限范围内的寄存器聚集成组生成时钟树叶级拓扑结构.基于 110 nm 物理工艺库 GRACE11LP,将该方法集成到 ISCAS89 基准电路三个不同规模的单时钟同步电路的时钟树综合流程中进行仿真验证,并与文献[7]对比,时钟网络的功耗分别降低了 20.3%、23.8% 和 27.8%,时钟偏移分别减少了 20.8%、23.6% 和 21.7%.仿真实验结果表明:该寄存器聚类方法在不影响时钟树最大延时的情况下,有效减少了时钟网络 20% 以上的功率耗散和 20% 以上的时钟偏移.

## 参考文献

- [1] 陈宇轩,梁利平. 基于新型结构的小面积全数字 DDR 接口模块[J]. 湖南大学学报(自然科学版),2018,45(4):155-160.  
CHEN Y X, LIANG L P. A small area all digital DDR PHY based on a novel structure [J]. Journal of Hunan University (Natural Sciences), 2018, 45(4):155-160. (in Chinese)
- [2] 白创,李帆,汪东. L-DSP 片上调试电路的设计与实现[J]. 湖南大学学报(自然科学版),2020,47(8):69-73.  
BAI C, LI F, WANG D. Design and implementation of L-DSP on-chip debug circuit [J]. Journal of Hunan University (Natural Sciences), 2020, 47(8):69-73. (in Chinese)
- [3] SU H Y, CHIANG P T, SAMANTA R, et al. Clock tree aware post-global placement optimization [C]//2017 2nd IEEE International Conference on Integrated Circuits and Microsystems (ICICM). Nanjing, China: IEEE, 2018:86-90.
- [4] SEITANIDIS I, DIMITRAKOPOULOS G, MATTHEAKIS P M, et al. Timing-driven and placement-aware multibit register composition [J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2019, 38(8): 1501-1514.
- [5] SEITANIDIS I, DIMITRAKOPOULOS G, MATTHEAKIS P, et al. Timing driven incremental multi-bit register composition using a placement-aware ILP formulation [C]//2017 54th ACM/EDAC/

- IEEE Design Automation Conference (DAC). Austin, TX, USA: IEEE, 2017: 1-6.
- [6] WANG X X, DONG Y K, LIANG J W, et al. Timing-based and balanced register clustering in near-threshold voltage clock tree design [C]//2020 IEEE 15th International Conference on Solid-State & Integrated Circuit Technology (ICSICT). Kunming, China: IEEE, 2020: 1-3.
- [7] HAN K, KAHNG A B, LI J J. Optimal generalized H-tree topology and buffering for high-performance and low-power clock distribution [J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2020, 39(2): 478-491.
- [8] DENG C, CAI Y C, ZHOU Q. Register clustering methodology for low power clock tree synthesis [J]. Journal of Computer Science and Technology, 2015, 30(2): 391-403.
- [9] 董永康. 近阈值下面向抗工艺波动时钟树的寄存器分组设计 [D]. 南京: 东南大学, 2020: 26-68.
- DONG Y K. Register grouping design for anti-process variation clock tree under near-threshold [D]. Nanjing: Southeast University, 2020: 26-68. (in Chinese)
- [10] SONG X N, ZHANG S Y, ZHOU J, et al. A variation aware register clustering methodology in near-threshold region [C]//2019 IEEE 13th International Conference on ASIC (ASICON). Chongqing, China: IEEE, 2020: 1-4.
- [11] YANG F, LIN M H, SUN H M, et al. Time-efficient and TSV-aware 3D gated clock tree synthesis based on self-tuning spectral clustering [C]//2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS). Boston, MA, USA: IEEE, 2017: 1200-1203.
- [12] SONG Y T, LI P, LIU Z L, et al. Buffer reduction for congestion control during timing optimization [C]//2022 IEEE 2nd International Conference on Power, Electronics and Computer Applications (ICPECA). Shenyang, China: IEEE, 2022: 36-40.
- [13] KWON T, IMRAN M, PAN D Z, et al. Virtual-tile-based flip-flop alignment methodology for clock network power optimization [J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2020, 28(5): 1256-1268.
- [14] 胡军涛, 薛智民, 龙娟, 等. 高速 DDR3 存储控制器的时钟偏差控制和优化 [J]. 微电子学与计算机, 2018, 35(10): 103-106.
- HU J T, XUE Z M, LONG J, et al. Design and optimize clock skew for high-speed DDR3 memory controller [J]. Microelectronics & Computer, 2018, 35(10): 103-106. (in Chinese)
- [15] 朱佳琪, 陈岚, 王海永. 一种低功耗时钟树的设计和 optimization 方法 [J]. 微电子学与计算机, 2021, 38(10): 85-90.
- ZHU J Q, CHEN L, WANG H Y. A low-power clock tree design and optimization method [J]. Microelectronics & Computer, 2021, 38(10): 85-90. (in Chinese)
- [16] GE Z, FU J, WANG P D. Low power clock tree optimization by clock buffer/inverter reduction [C]//2019 IEEE International Conference on Integrated Circuits, Technologies and Applications (ICTA). Chengdu, China: IEEE, 2020: 69-70.
- [17] GABER L, HUSSEIN A I, MONESS M. Improved automatic correction for digital VLSI circuits [C]//2019 31st International Conference on Microelectronics (ICM). Cairo, Egypt: IEEE, 2020: 18-22.