國 立 交 通 大 學

電信工程學系

碩 士 論 文

低功率時脈網路取向多階層化電路擺置

A Multilevel Low Power Clock Network Driven Placement

研 究 生：林哲宇

指導教授：李育民　　教授

中　華　民　國 九 十 五 年 十 月

低功率時脈網路取向多階層化電路擺置

學生：林哲宇　　指導教授：李育民博士

國立交通大學電信工程學系（研究所）碩士班

摘　　要

製程進步的演化迅速，特別在進入深次微米製程後，時脈網路在許多高速的電路設計上消耗了超過40%的總功率。而現今產品廣泛地需要可攜式功能，如何將晶片的功率消耗做有效的規劃，是一個重要的課題。

我們提出了一個低功率電路時脈取向網路結構，在以各個元件的活動率為基礎下進行。我們主要採用的方法為閘控時脈網路加上使用非零時脈歪斜的時脈最佳化理論來使電路的時脈網路縮減，並且是以電路活動的機率做最佳化的考量基礎，使電路在考慮原來的時脈限制下，對功率消耗做最佳化。並且在進行電路佈局時，可將暫存器利用絕對線長法移到我們所考慮的最佳化位置，且採用的方法為多階層化電路佈局，以加快速度，在每一層都先根據該階層的狀況，進行電路佈局並且使用絕對線長設定一個暫存器的錨釘來限制其符合我們所建構最佳時脈網路，且同時考慮系統面積與系統總線長。使用一個有系統的方法減少功率的消耗。

A Multilevel Low Power Clock Network driven Placement

Student：Zhe-Yu Lin Advisor：Dr. Yu-Min Lee

Department of Communication Engineering
National Chiao Tung University

ABSTRACT

As the process technology entering the deep sub-micron design era, the power consumption of clock networks dominates over 40% of the total power in modern high performance VLSI designs. Particularly, the power design is substantial in portable electronic devices. How to optimize the power consumption of chips is an important issue.

The purpose of this thesis is to construct a low power clock network in the placement level. We use the gated clock method and non-zero clock skew optimization to reduce the total switching capacitance of clock networks. It is based on circuit block activity to minimize the clock network power under clock period constraint and it is implemented on a multilevel placer for the reason of processing complex circuits. During each level of the hierarchical placement, it uses anchors determined by the Manhatern circle to constrain the flip-flop in the desired region and it constructs an optimal clock network considering the total area and wire length. The experimental results show that our methods indeed reduce the clock network power consumption.

誌　　謝

首先要感謝指導老師李育民老師在我兩年的研究生活中，提供良好的學習環境，以及對我的指導與照顧，學生銘記在心，在此向老師致上由衷感謝。另承蒙李毅郎與陳宏明對本論文的諸多指導與建議，使其更趨完備，在此感謝每一位老師的熱心指導。

其次我要感謝至鴻和逸宏學長在研究上的傾囊相授，義琅和震軒學長在這兩年的幫助與指導，以及同窗培育、益廷、志康、國富、炳勳、獻哲、義志、政翰、琳家、家泰和俊達，在精神上及學業上的互相扶持。感謝我的女朋友芝萱一直支持我，感謝我的室友及朋友，因為有你們，由衷的感謝，還有學弟們，為這間實驗室帶來許多歡笑與活力，感謝大家。

最後，感謝我的父母與家人，感謝他們提供一個穩定且健全的環境，他們的默默支持及諄諄教誨，我常感懷於心，讓我能全心投入研究之中，無後顧之憂的完成我的學業與論文。在此致上最高的敬意，希望你們永遠健康快樂。

總之我要感謝所有關心我的、愛護我的和曾經幫助過我的人，願我在未來能有一絲的榮耀歸於最愛我的家人、老師以及朋友，謝謝你們。

林哲宇

國立交通大學

中華民國九十五年九月

國立交通大學
電信工程學系碩士班

論文口試委員會審定書

本校 電信工程學系 碩士班 林哲宇 君

所提論文(中文) 低功率時脈網路取向多階層化電路擺置

(英文) A Multilevel Low Power Clock Network driven Placement

合於碩士資格水準、業經本委員會評審認可。

口試委員：

指導教授：

系主任： 教授

中 華 民 國 95 年 10 月 2 日

國 立 交 通 大 學

博碩士論文全文電子檔著作權授權書

（提供授權人裝訂於紙本論文書名頁之次頁用）

本授權書所授權之學位論文，為本人於國立交通大學電信工程系所系統晶片組，95 學年度第一學期取得碩士學位之論文。

論文題目：低功率時脈網路取向多階層化電路擺置
指導教授：李育民

■ 同意

本人茲將本著作，以非專屬、無償授權國立交通大學與台灣聯合大學系統圖書館：基於推動讀者間「資源共享、互惠合作」之理念，與回饋社會與學術研究之目的，國立交通大學及台灣聯合大學系統圖書館得不限地域、時間與次數，以紙本、光碟或數位化等各種方法收錄、重製與利用；於著作權法合理使用範圍內，讀者得進行線上檢索、閱覽、下載或列印。

論文全文上載網路公開之範圍及時間：

本校及台灣聯合大學系統區域網路	■ 中華民國 96 年 10 月 3 日公開
校外網際網路	■ 中華民國 96 年 10 月 3 日公開

授 權 人：林哲宇

親筆簽名：林哲宇

中華民國 95 年 10 月 3 日

國 立 交 通 大 學

博碩士紙本論文著作權授權書

（提供授權人裝訂於全文電子檔授權書之次頁用）

本授權書所授權之學位論文，為本人於國立交通大學電信工程系所系統晶片組， 95 學年度第一學期取得碩士學位之論文。

論文題目：低功率時脈網路取向多階層化電路擺置
指導教授：李育民

■ 同意

本人茲將本著作，以非專屬、無償授權國立交通大學，基於推動讀者間「資源共享、互惠合作」之理念，與回饋社會與學術研究之目的，國立交通大學圖書館得以紙本收錄、重製與利用；於著作權法合理使用範圍內，讀者得進行閱覽或列印。

本論文為本人向經濟部智慧局申請專利(未申請者本條款請不予理會)的附件之一，申請文號為：_____，請將論文延至____年____月____日再公開。

授 權 人：林哲宇

親筆簽名：林哲宇

中華民國 95 年 10 月 3 日

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Temperature profiling and battery life have made power consumption be a primary optimization target for IC designs. In modern high-performance VLSI (Very Large-Scale Integration) digital systems, the clock distribution network consumes a large portion (more than 40%) of the total circuit power dissipation [21].

Since the clock signal directly influences to the synchronization of each block, the stability of clock signal always plays an important factor in the circuit design. The clock synchronization affects the whole circuit performance. The clock skew (the maximum difference in the arrival time of clock signals to the flip-flops) problem not only brings the violation of circuit functions but also results in delay of rise/ fall time of circuits. The load of clock signal wire is the most serious part in modern VLSI design and it is also sensitive to the manufacturing process.

In recent years, there has been renewal of interest in clock tree problems. Most of them are developed to solve the clock skew problems for the sake of preventing the clock race condition problem (race condition, which is the time of clock signal arriving to the flip-flops, must be constrained by zero clocking and double clocking.) In modern VLSI designs, the clock skew problem becomes extremely complex. The clock signal wires are globally distributed. Hence, if the boundary size of chip is larger, the clock skew problem becomes more serious. If we do not prevent the above effect in the early stage, more clock

wire routing resource is required in the backend stage.

Generally speaking, the researches of clock skew problem can be classified into two categories: clock skew for clock period optimization and clock tree routing (synthesis). Over the past few years, a count of literatures about clock skew problem have been proposed to optimize the clock skew problem and therefore improve circuit performance. Most of them are targeted on finding optimal clock period using clock skew scheduling which translates the clock skew problem into the linear programming problem [50, 51, 52], and the reliability of circuits can be improved by providing the timing sequential difference [51, 52]. Hu et. al. [54], proposed a novel algorithm to use the timing sequential difference to increase the reliability of circuits, and they also performed a clock skew minimization for reducing the clock wire length under a targeted clock period.

Most of clock tree routing algorithms are intended as an investigation of minimizing clock skew . They can be classified into three main groups. In the first group, it focuses on zero skew clock routing such as a H-tree in [16, 15, 18, 17, 12, 14, 13, 20]. In the second group it focuses on boundary skew clock routing like [60, 61, 62]. In the final group it targets on reducing clock wire length under clock skew optimization [54]. Inspiring the above methods, we use a linear programming method to achieve the minimization of clock network switching activity under a targeted clock period in this thesis.

Since the complexity of VLSI circuits is in order of hundreds of millions of transistors. As the purpose of reducing the complexity of design process, VLSI design process can be divided into three domains, i.e, the behavioral domain, structural domain and physical domain. It becomes extremely complex especially in physical domain. The design process in physical domain can be further divided into two phases: placement design and routing design. In this thesis, we target on placement of physical design and we translate our weighted clock tree topology to plug it into a multilevel placer “UMpack Meta-Capo9.5 ” [63] of stand cell layout.

Numerous attempts have been made by scholars to reduce the power consumption

during the placement design stage [6]–[11]. The power consumption can be divided into two types: dynamic power consumption (switching power) and static power consumption (leakage power). Recently [53] which is a low power driven placement, it can be divided into two phases. First, it analyzes the activity of all blocks and then construct a clock topology by the algorithm QTS. Hence, it computes the block activity and modify the weight of block and net of input netlist according to the switching activity. Finally, they use a commercial placement tool to achieve their goal of low power consumption. But the QTS algorithm does not focus on the low power clock tree construction, and will affect the total power consumption in more than 40%.

Clock gating is a well-known technique to reduce the dynamic power consumption in clock network of a digital circuit. It disables portions of the clock tree distribution for saving the dynamic power. The existing gated clock routing algorithms (DME like method) work in two phases. First it is bottom up to formulate a specifically objective function is switching capacitance which is based on the switching activities, capacitances of clock sinks, gated clock controlled circuits, the wire length of clock network and gated clock control network to find the set of merge points. In second step, it is top down to find the real position of each merge set and then a clock network routing is constructed.

The traditional clock network is constructed after placement procedure. As the process technology enters the deep sub-micron design era and the increasing of circuit design complexity, the clock network becomes more complex to construct and the penalty of erasing clock skew is huger. It becomes more important to emphasize a clock tree planning before the traditional clock routing stage.

According to [23], most of the publications [22, 24, 25] have suffered from the same shortcoming that there is no deeply physical level consideration while performing the gated clock routing algorithm. As the result, the clock tree constraints may not be met or the clock power dissipation of the design may actually go up after the gated clock routing. Furthermore, the trend of continuously increasing interconnect power is another

more severe problem which has been pointed out by [21].Therefore, we need to pay much more effort in reducing the interconnect power in order to save more power. These facts motivate us to develop a dedicated placement algorithm for reducing the power dissipation of clock distribution network.

In this thesis, we just consider minimizing the dynamic power. The dynamic power consumed by a module clocked at a specific clock frequency is given by $fV_{dd}^2C_L$, where V_{dd} is the supply voltage, C_L is the total load capacitance on the circuit. The formula of power consumption is given by $P = \alpha fV_{dd}^2C_L$, and α is called the switching activity. To minimize the power consumed by a CMOS synchronous system, we intend to minimize the switching activity and total load capacitance.

Because the feature size still has the trend of decreasing, the interconnection effect problems now is a highly attention. The interconnection's effect such as signal integrity, transmission line effect, and I-R drop $\cdots$etc is caused by the shrinking feature size. Among them, there is one significant problem that changes our view of power consumption, that is, the the total power consumption of interconnect now gets closer to the loading power as long as the line is sufficient long. Specifically, the clock network power consumption no longer dominate by the loading registers.

In this thesis, we first perform activity analysis and timing analysis to the circuit netlist. Secondly, we translate the information into the partition-er "hMETIS" [73] to construct a clock topology. Thirdly the clock tree information will be translated into weighted netlist to the modified multilevel placer "Umpack" [63]. Therefore, we apply our gated clock network construct algorithm to the low power driven plaer [53] to reduce total power consumption.

1.2 Our Contribution

In this thesis, our proposed algorithm, a low power clock network driven placer under a low power clock topology, utilizes three novel ideas in reducing power consumption.

- *A novel gated clock topology for reducing clock network power before placement*

  A new gated clock topology is proposed to construct a low power clock network under a delay target time constraint. The delay target time of each clock sink is estimated in a simple timing analysis. With this information of delay the clock topology is permitted to construct a more low power clock topology such as [54].

- *A low power clock skew scheduling based on linear programming method*

  The traditional clock scheduling is targeted on minimizing clock period, but our target is to minimize the clock power under a given clock period. We modify the cost function of traditional clock skew scheduling for reducing power consumption of clock network under gated clock topology. Our objective function is to minimize sum of activity clock signal wire length avoid race condition.

- *Application to a low power driven placement [53]*

  We replace the original clock tree topology construction with our low power clock topology constructor. Hence, we release the clock network clustering for reducing the general signal wire.

We propose a new flow to construct a clock tree anterior placement. First, the testing pattern activity of the circuit is analyzed for gaining the activity of each block. Secondly the simple static timing analysis is performed to obtain the delay between each register and block. We perform a clock scheduling algorithm for low power which is to perform a linear programming to minimize switching activity under a targeted clock period constraint, and we obtain the delay target time of each clock sink. Under the auspices of delay target time and activity of each block, we can construct a clock network topology.

We use a novel gated clock topology construction algorithm which considering many traditional physical constraints at the same time, such as block area, wire length and also gated clock power and clock skew problem. In order to forecast the position of clock sink before partition based placement, we choose the partition algorithm and a partition-er "hMETIS" [73] to consolidate all the constraint. To begin with partitioning, we translate the activity of each block into a weighted net. Hence, we translate delay target time of each block into a weight node. Finally, we feed the modified netlist into the "hMETIS" [73] to gain a layer of clock tree and then recursive to construct the whole topology.

The algorithm of gated clock network routing "GCR" [22, 25] is modified for our clock clock network and the placer is also modified to suit our goal. In the first place, we give the delay target time of each block to gated clock router, and modified the cost function for merging point. In the second place, we pre-place the pseudo blocks during the partition of "UMpack" [63] and applied the additional net to reduce their wire length during placement.

The experiment results are excellent. With feeding the placement of proposed placer into our modified gated clock routing algorithm. This gated clock algorithm can be saved average 20% of clock power consumption compared with its routing result using a general placement. To find a whole power consumption reduction, we imitate [53] and the result show that it reduce about 3% total interconnect power consumption after performing [53].

1.3 Organization of this Thesis

In this thesis, we focus on designing the algorithm for constructing the low power placement for gated clock circuitry. First, in Chapter 2, we will introduce the basic idea of circuit placement and clock network distribution. At the same time, we also present a survey for all the related literatures thoroughly. Specifically, the researches that focus on handling the low power design problems. In Chapter 3, we first introduce the method we

adopted for reducing clock network power which plays as an important role in our Low power clock network driven placer and second the flow chart of our algorithm. Thirdly, we will also introduce our gated clock topology constructor in detail and then we will present our algorithm and design considerations for constructing the low power driven clock network circuit placer under a gated clock topology. After the circuit placement, next step is the clock network routing.

In Chapter 4, first we will present our experimental result for the comparison between the typical placer under the universal circuit benchmarks. Secondly, we compare the total power consumption of interconnect between the placer in [53]which is implemented within our clock network construct algorithm and original one. The result is compared with traditional constraints, clock network power and total power consumption.

Finally, a brief conclusion and future work suggestion will be given in the remaining of the Chapter 5.

Chapter 2

Preliminaries and Literatures Overview

In this chapter, the basic concept of placement, clock skew optimization problem, and clock routing problem will be presented and some previous related researches will be reviewed in this chapter.

2.1 Traditional VLSI Design Flow

The traditional VLSI design flow is shown in Fig 2.1. First, designers synthesize their circuit by several synthesizers, Verilog or VHDL. Hence the synthesized circuits should be partitioned into several blocks according to the circuit functions or achieve the minimum number of cut between blocks. In the floorplan and placement stage, each function block is placed on the proper location where to achieve the minimum total area, wire length, congestion, crosstalk, or power consumption, etc. After placement stage, if it is a sequential circuit, the clock tree is synthesized for avoid clock skew problem, and the routing stage is performed behind it. In general, this stage emphasizes the routability, wiring congestion, and timing improvement. When the routing is complete, the compaction, extraction, and circuit verification is performed to minimize the total area and verify the performance as well as signal integrity, respectively. Finally, taping out and finishing the design.

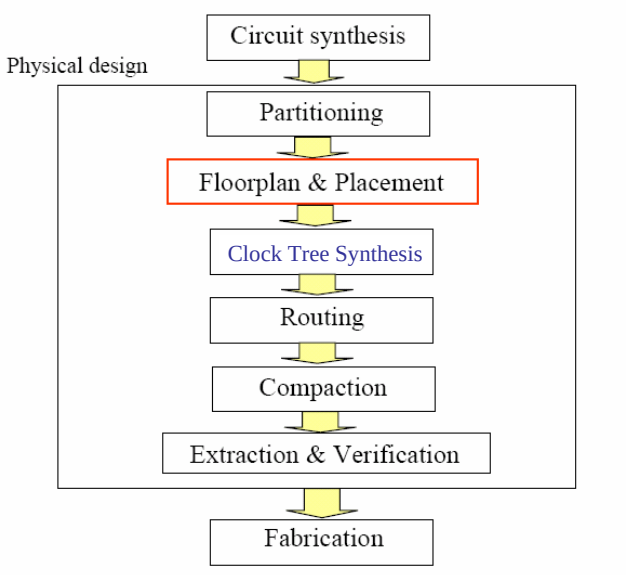


Fig. 2.1: Traditional VLSI design flow.

2.2 Partition of the Algorithms Used in hMETIS

In the rest of this section, we briefly describe the various phases of the multilevel algorithm.

2.2.1 Coarsening Phase

During the hyper-graph coarsening phase, a sequence of successively smaller hyper-graphs is constructed. The purpose of coarsening is to create a small hyper-graph, such that a good bisection of the small hyper-graph is not significantly worse than the bisection directly obtained for the original hyper-graph. In addition to that, hyper-graph coarsening also helps in successively reducing the size of the hyper-edges. That is, after several levels of coarsening, large hyper-edges are contracted to hyper-edges connecting just a few vertices. This is particularly helpful, since refinement heuristics based on the Kernighan-Lin algorithm are very effective in refining small hyper-edges but are quite ineffective in refining hyper-edges with a large number of vertices belonging to different partitions.

The group of vertices that are contracted together to form single vertices in the next level coarse hyper-graph can be selected in different ways. hMETIS implements various such grouping schemes (also called matching schemes).

2.2.2 Initial Partitioning Phase

During the initial partitioning phase, a bisection of the coarsened hyper-graph is computed. Since this hyper-graph has a very small number of vertices (usually less than 100 vertices) many different algorithms can be used without significantly affecting the overall runtime and quality of the algorithm. hMETIS uses multiple random bisections followed by the Fiduccia-Mattheyses(FM) refinement algorithm.

2.2.3 Un-coarsening and Refinement Phase

During the un-coarsening phase, the partitioning of the coarsest hyper-graph is used to obtain a partitioning for the finer hyper-graph. This is done by successively projecting the partitioning to the next level finer hyper-graph and using a partitioning refinement algorithm to reduce the cut and thus improve the quality of the partitioning. Since the next level finer hyper-graph has more degrees of freedom, such refinement algorithms tend to improve the quality. hMETIS implements a variety of algorithms that are based on the FM algorithm.

2.3 Basic Concept of General Placement

General Placement

Placement is a basic step in VLSI physical design flow. A poor placement brings about larger areas, performance degradation, higher power consumption, and even a hot spot. The placement has a deeply impact on the overall layout. The good placement gives a good initial solution of routing to solve the problem we mention above. In other words, the overall quality of the layout such as area, power, total wire-length or thermal distribution

is mainly determined as early as in the placement phase. **Multilevel placement**

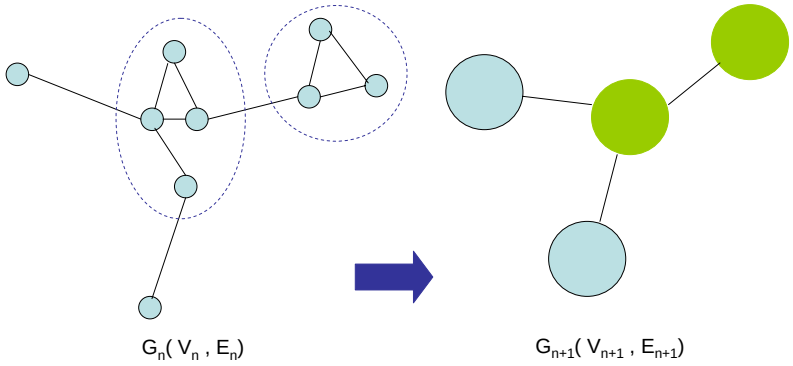


Fig. 2.2: The reduced graph of multilevel method.

In order to deal with enormous and complex circuit, most of multilevel placer use the partition based algorithm in [55, 56, 57]. This kind of algorithm is first partition the circuit into several sub-circuits in order to decrease the complexity. For the goal, it use the multilevel hierarchy architecture. Using this architecture can speed up the processing time of the placer.

We translate the circuit into graph 2.3, and the stand cell view as the vertex of graph, the net view as the edge. In the hierarchy architecture, it usually use the coarsen method the stand cell into a module, recursive until all stand cells are group into modules.

2.3.1 Problem Formulation

We now define the placement problem as follows:

Given:

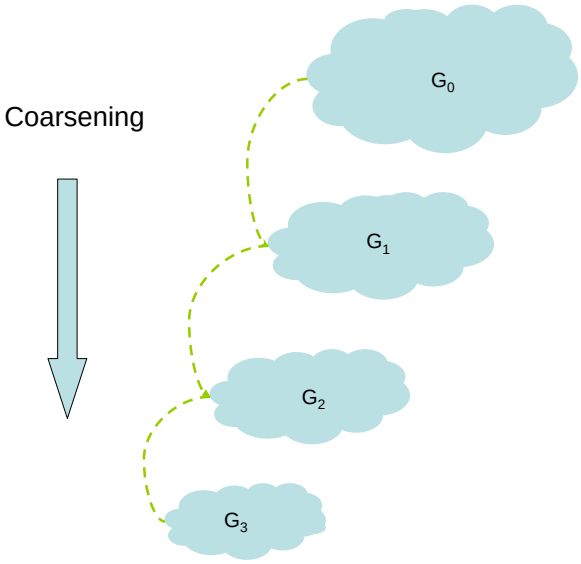


Fig. 2.3: The hierarchy reducing of multilevel method.

A set of circuit modules, a placement is to arrange those modules on the layout surface to satisfy some prescribed constraints (such as no two modules overlap or circuit boundary constraint).

Optimize certain objective functions:

Such as total area, total estimated wire-length after placement or total power consumption. A circuit is now considering as a group of rectangular modules here in placement level, and the geometry shapes of the modules are fixed. We can perform some limited operations on the modules to form the optimal placement according to the cost function such as rotation, flip, move or swap.

2.3.2 Wire-length Estimation

Although the real signal wire-length is not available at placement level, a placement algorithm needs to model the actual topology of the interconnection nets fast and accurate. An interconnection graph structure which interconnects each net is used for this purpose. The interconnection structure for two terminal trees is simply an edge between the two vertices

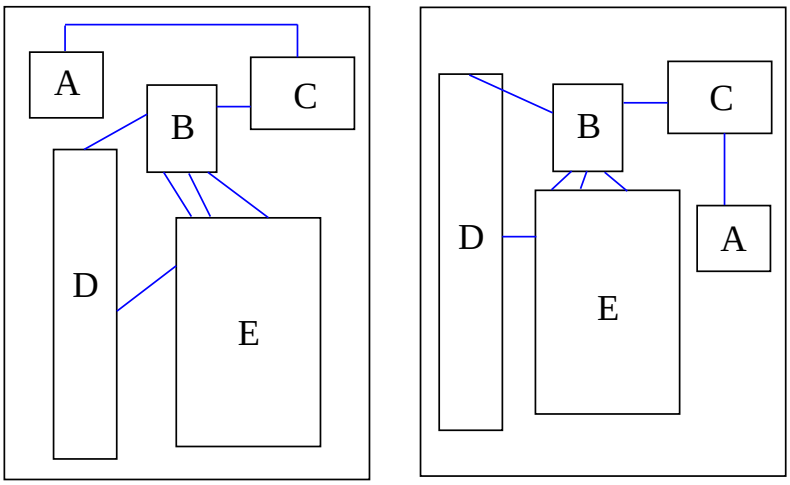


Fig. 2.4: The translation graph of general placement.

corresponding to the terminals. In order to model the net with more than two terminals, rectilinear Steiner trees are used as shown in Fig. 2.5(a) to estimate optimal wiring paths for a net. This method is usually not used by placers, because of the NP-completeness of Steiner tree problem. As a result, minimum spanning tree and semi-perimeter representations are the most commonly used structures to connect a net in the placement phase. Minimum spanning tree connections (shown in Fig 2.5(b)) allow branching only at the pin locations. Hence, the pins are connected in the form of minimum spanning tree of a graph. The other one is semi-perimeter representation (shown in Fig 2.5(c)), it could fast estimate the wire-length by bound pins with a rectangle then calculate its semi-perimeter. We admit in this paper through this estimation model. Complete graph interconnection is shown in Fig 2.5 (d)). It is easy to implement such structure. However, this method causes many redundant interconnections, and result in longer wire length.

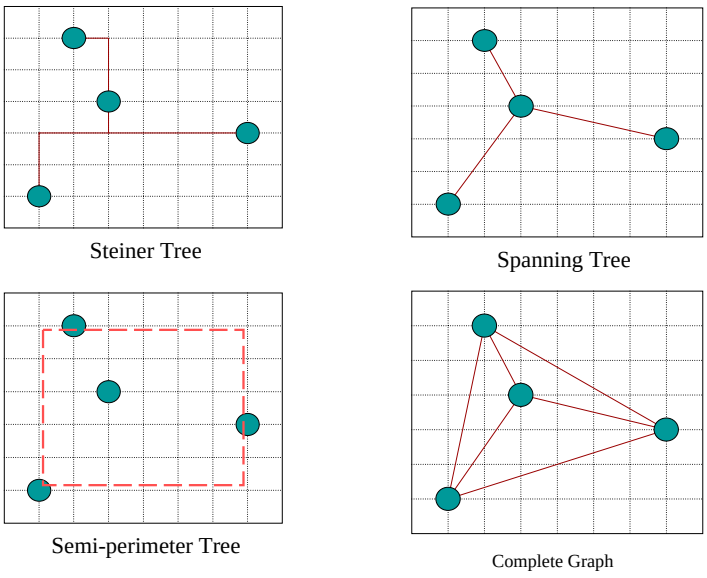


Fig. 2.5: The wire estimation model.

2.4 Previous Work Related to Low Power Placement

Many placement algorithm [6]–[11]and [53] have been developed to reduce the power consumption at this level, but none of them cooperated with the technique of clock gating. They only optimized dynamic switching power through different approaches with consideration on the combinational networks. We revisit the low power driven placement problem with a focus on gated clock and provide a novel placer to reduce the interconnect power both in gated clock network and gated clock control network. In other words, we provide a good seed for constructing an optimal low power gated clock network.

Generally, the total power consumption P_total, in a typical CMOS circuit can be formulated as follow:

$$P_{total} = \sum_{\forall block_i} P_{block_i} \quad (2.1)$$

$$P_{wire} = \alpha_{wire} C_{load} f V D D^2 \quad (2.2)$$

Where Ptotal is the total power consumption, P_blocki is the power consumption of block i, VDD is the supply voltage, α_{wire} is the activity of the net, the C_{load} is the wire capacitance.

The existing low power driven placement methods [6]–[11] optimized the following cost function through different approaches since they believed that this cost function was the only one related to the power consumption at the placement level.

In [53] which is a recently work in the low power placement algorithm, this work is a good method for low power. It performs a QTS (quick clock tree synthesis) before placement start, then performs the algorithm “RCC” which is cluster the register in order to reduce the clock power by add net weight to each cluster and promote the new boundary of cluster.

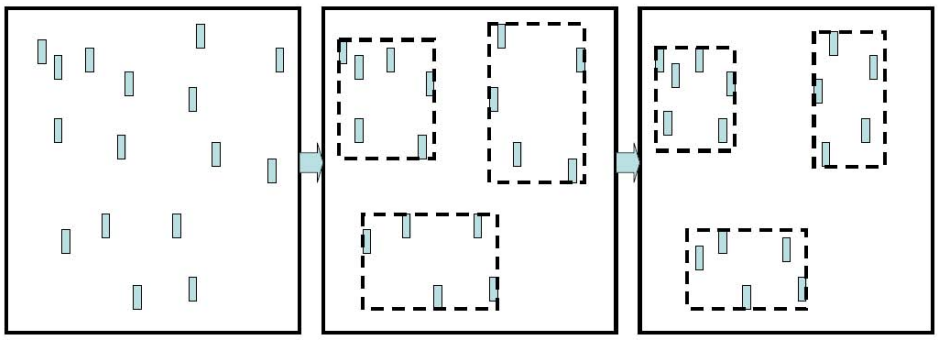


Fig. 2.6: Register clustering.

After constructing clock tree, the placer modifies the general net weight by the activity of the net. In the 2.7, the Fig 2.7 (a) is a normal place case and when the register clustering perform the block would become in Fig 2.7(b), and if we give each net activity relation in Fig 2.7(c), and the situation would become as a weight balance result Fig 2.7(d)).

Finally, the placer combine the additional power weight by the algorithm above and the net weight of timing form timing analysis. After refining the net weight and block weight, the placer call a robust placer which can deal with the weighted circuit netlist.

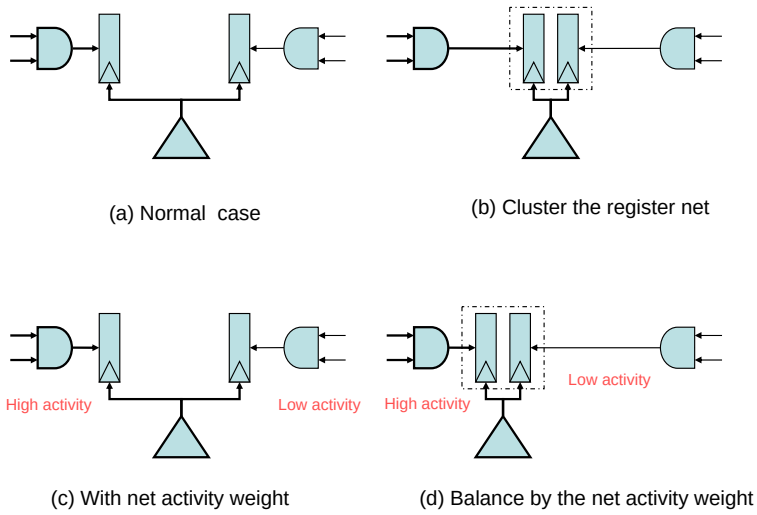


Fig. 2.7: Activity based register clustering.

2.5 Clock Skew Optimization

In a sequential VLSI circuit, due to differences in interconnect delays on the clock distribution network, this simultaneity is difficult to achieve. Improving the performance of a synchronous digital system by adjusting the path delays of the clock signal from the central clock source to individual flip-flops is investigated.

2.5.1 Problem Formulation

Given:

Each registers in the circuit, and timing information (after timing analysis) of each register.

Objective:

(1) Minimizing the clock period, while avoiding clock hazards.

(2) For a given period, maximizing the minimum safety margin against clock hazard.

Using delay model to detect clocking hazards, two linear programs are investigated: The

conventional approach to design builds the clock distributions network so as to ensure zero clock skew. An alternative approach views clock skew as a manageable resource rather than a liability, and uses them to advantage by intentionally introducing skews to improve the performance of the circuit. To illustrate how this may be done, consider the circuit shown in Fig 2.8 2.9. In Fig 2.8, this circuit cannot be properly clocked to function at a period of 2 time units, because as shown in the figure, the required arrival time of the signal at register L1 is 2 units, while the data arrives after 3 time units. It is readily verifiable that the fastest allowable clock for this circuit has a period of 3 units. But if we add a extra delay +1 in the F1 clock pin, the circuit can be work in clock period 2 in Fig 2.9.

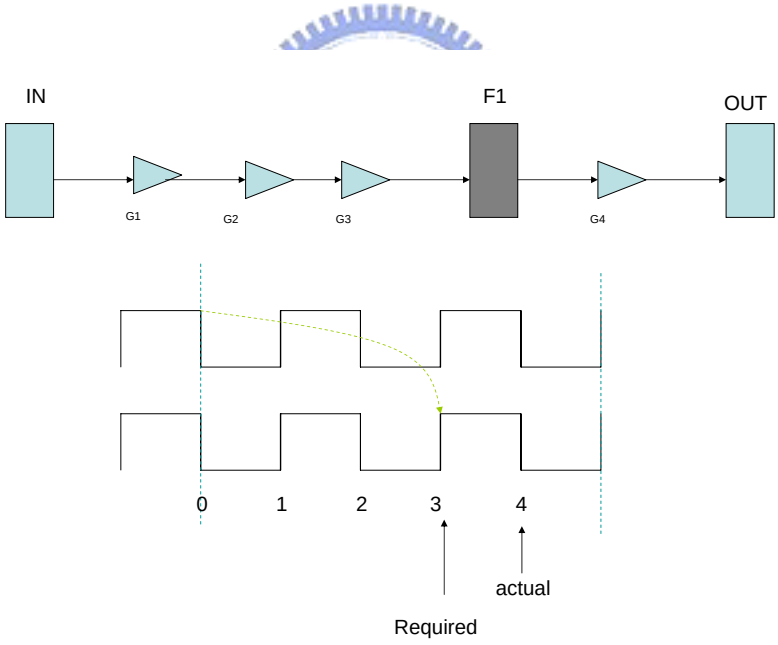


Fig. 2.8: Zero skew clock routing.

Furthermore, we can translate the clock skew problem into linear problem by avoid race condition. To meet the race condition constraint, the each clock arrives time must be satisfied those two clock constraints: zero clocking and double clocking constraint.

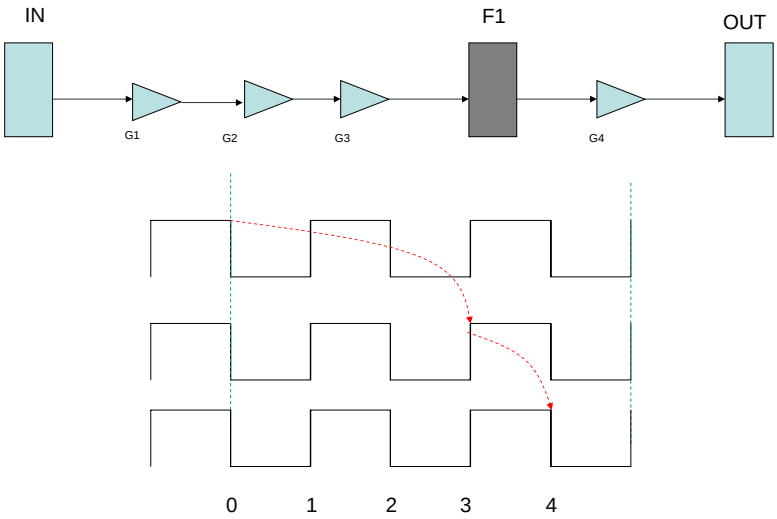


Fig. 2.9: Non-zero skew clock routing.

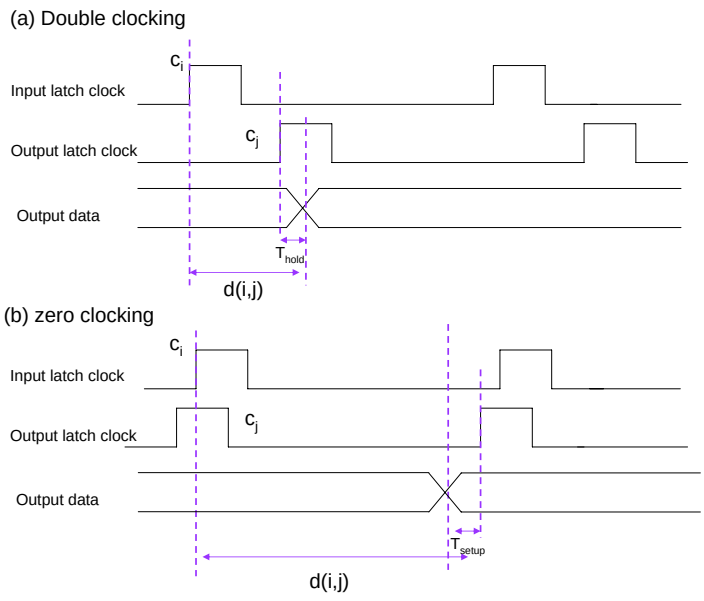


Fig. 2.10: Zero clocking and double clocking.

The constraint function can be written as follows:

$$\begin{aligned} &T_{skewi,j} \geq T_{hold} - \underline{delay}(i,j) \\ &T_{skewi,j} \leq P - T_{setup} - \overline{delay}(i,j) \\ &T_{skewi,j} = (W_i - W_j) \\ &W_i < \min(H_i, W_i); W_j < \min(H_i, W_j) \end{aligned} \tag{2.3}$$

2.6 Basic Concept of Clock Network Routing

Clock tree routing is the final stage of the clock skew solver, if the clock routing is poor and the clock it will direct influence to clock period. Clock nets should be routed with great precision, since the actual length of the path of a net from its entry point to its terminals determines the maximum clock frequency on which a chip may operate.

Clock signal is global distribution and therefore clock lines have to be very large so that they have heavy capacitances and limit the performance of the system. Therefore, a clock router needs to take several factors into account, including the resistance and capacitance of the routed wire, the noise and cross talk between wires, the type of load to be driven and the arrival time from clock source to sinks. Among the all clock routing considerations, the most important factor is "*clock skew*", the clock signal must arrive simultaneously to all flip-flop with little or no waveform distortion.

Because the clock network is such important, optimization of the clock signal can have a significant impact on the chip's cycle time, especially in high-performance designs. Non-optimal clock behavior is caused by either of two phenomena: the routing to the chip's synchronizing elements (flip-flops or registers), or in the non-symmetric behavior of the clock distribution logic. Therefore, specialized algorithms are required for clock nets due to strict specifications for routing clock network. Before the introduction of these algorithms, we discuss some fundamentals of clock network such as delay model timing and clock skew first.

2.6.1 Elmore Delay Model

In 1948, Elmore introduced a general approach for calculating the propagation delay of a linear system given its transfer function and it become widespread. The popularity of the *Elmore delay model* is mainly due to the existence of a simple tractable formula for the delay that has recursive properties making the calculation of the circuit delays highly efficient even in large circuits. [49].

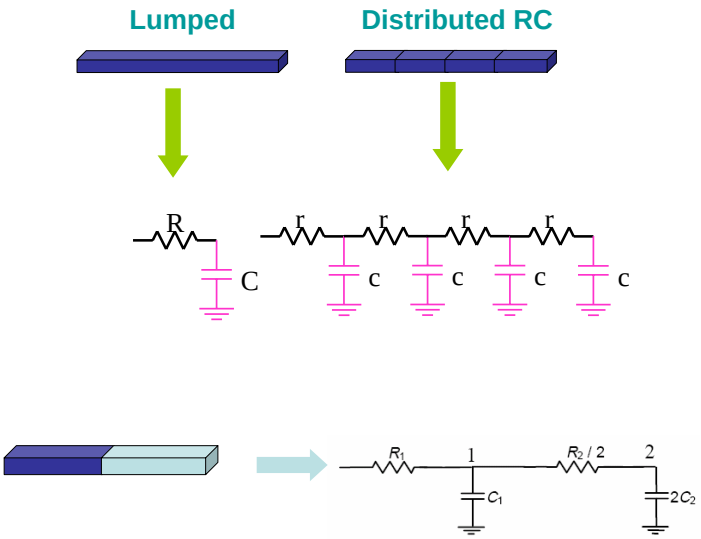


Fig. 2.11: Lump and distribution RC model.

Elmore delay Formulation :

$$R_{ik} = \sum R_j \text{s.t.}(R_j \in [path(s \rightarrow i) \cap path(s \rightarrow k)]) \tag{2.4}$$

Let's see Fig. 2.11(a), a Lump RC model is used to model a wire segment and a RC distribution. Fig. 2.11(b) is a simple RC tree which is combinate with two wires.

$$C_i = c_i + \sum_{j \in S_i} C_j \tag{2.5}$$

where S_i is the set of all the immediate successor of p_i. Let $\delta_{i,j}$ be the path between p_i

and p_j, excluding p_i and including p_j. Hence the delay between two nodes i and j is

$$t_{ij} = \sum_{j \in \delta_{i,j}} r_j C_j \tag{2.6}$$

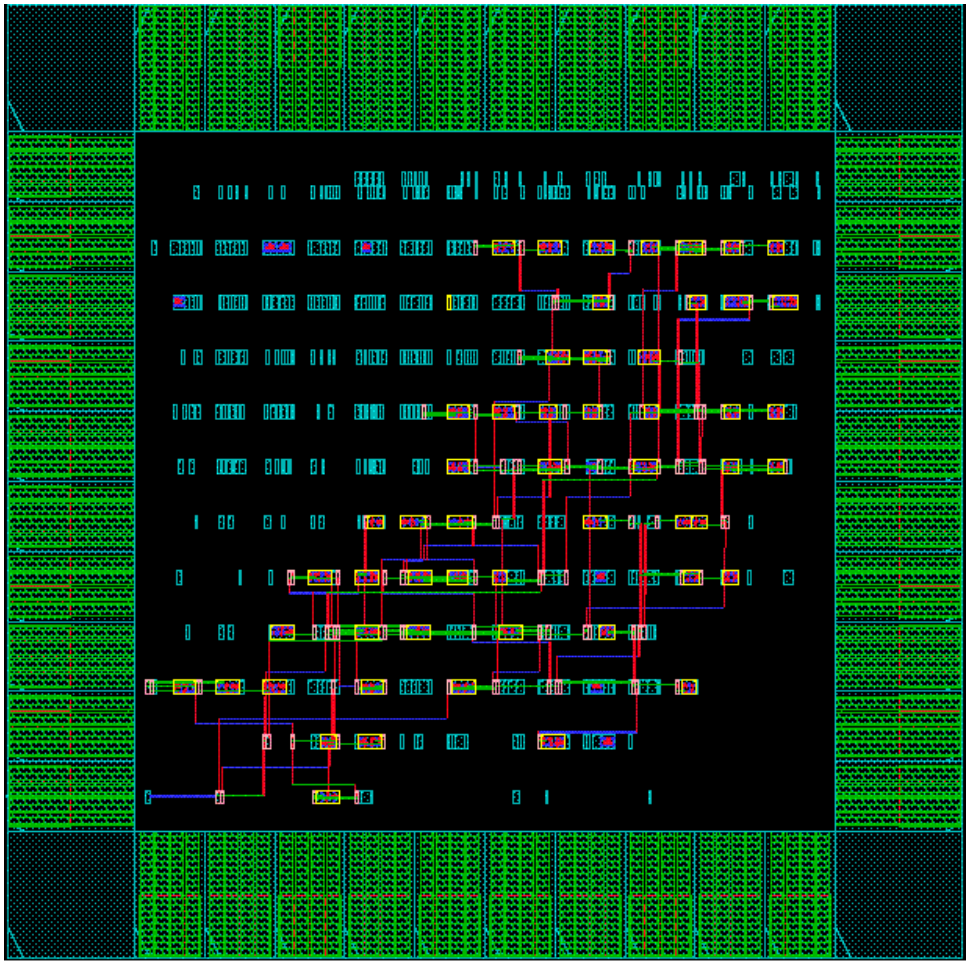


Fig. 2.12: The clock tree in IC layout.

Clock Skew

Fig 2.12 demonstrate a typical synchronous system, dotted lines represent clock path. The clock signal is often generated external to the chip and supplied to the chips through the clock source pins. All chips would have their own clock network distribution by some dedicated clock router in order to transmit clock signals from clock source pin to every functional unit which needs clock signal. Hence, the clock controls the flow of data

transmission within the synchronous system. In this figure. we omitted the internal clock networks for a brief introduction.

Every designer hope that the arrive time from clock source pin to all functional units are completely and precisely the same so that all data transformations may start at the same time without data stall and the functional unit could be designed faster to achieve maximum performance.

Actually, the clock signals do not arrive at all functional units simultaneously. We give the definition for the maximum difference from the clock source to each the clock pins*clock skew*, just as Fig.2.13 demonstrate. With the clock skew limitation, we can not design our system ideally. In other word, we should first calculated the budget or bound for the maximum clock skew could affected. As skew increases with the clock period held fixed, the efficiency of the digital system is reduced because valuable computation time is "stolen" from the total cycle time. Frequently in high-performance design environments, skew is constrained to be less than five percent of the clock period. Thus, in a 1 Ghz design, skew would be less than 5000ps. However, there is still some clock skew that we cannot solve which comes from manufacturing process.It forces us to be conservative and use little bigger clock period in order to minimize the skew effect so that we are sacrifice the maximum frequency from malfunction comes from clock skew.

We adopts the clock period which allows for the maximum completion time of all the functional units' in addition to the extra time for handling the maximum clock skew. As we know, we should minimize the clock period as we can so that we should evaluate for the second term which represent the bound of the clock skew. However, this term is hard to be evaluated since it deviated from logic style, clock routing network, register distribution and technology. More precisely, clock skew is caused by several phenomena: asymmetric routes to the clocked elements, differing interconnect line parameters, different delays through the clock distribution elements, and different device threshold voltages for the clock distribution logic.

Therefore we should keep any possibilities in our mind as to make a pessimistic design for handling the worst case. By the way, if we could minimize the skew effect in every level of the design, we will have higher probability for improving the performance.

In the following subsections, we will introduce some classic algorithm which minimize the skew effect even minimize the clock power simultaneously.

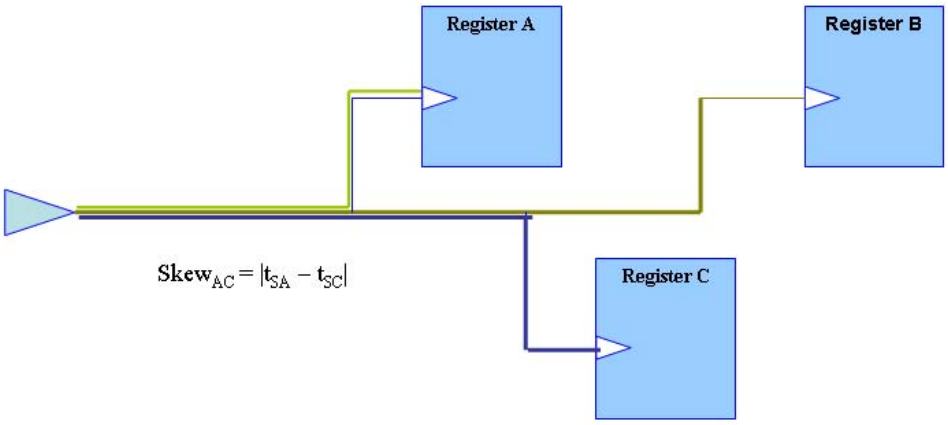


Fig. 2.13: A simple diagram which demonstrates clock skew effect.

2.6.2 Problem Formulation

The clock routing problem defines as follows: Given the routing plane and a set of sinks $C = \{c_1, c_2, \cdots, c_n\}$ lying within the plane and c_s represent the clock source pin. We now wish to construct a network to connect all of the clock pins c_i with clock source c_s. At the same time, we also wish to minimize the source to sink delay, clock skew and total clock wire-length. We can formulate them as follows: $ClockRoutingCost = Minimize(\max_{\tau_{(c_s,c_j)}}, \max_{(m,n)\in C} \mid \tau_{(c_s,c_m)} - \tau_{(c_s,c_n)} \mid, wirelength_{clock})$

2.7 Previous Work Related to General and Low Power Clock Routing

Prior to delving into the clock routing algorithm, it is necessary to define its role in the context of the overall design flow. In the contemporary physical design flow, it consist three classical steps: floorplanning/placement, global routing and detailed routing.

Clock routing is a specific problem interposed between placement and global routing. Specifically, during the clock routing step, the global and detail routes of the clock net are determined and passed to the global router as blockages. The determined routes are constructed so that the clock signal behavior is optimized.

2.7.1 Zero-skew Clock Routing

The skew could be minimized by distributing the clock sinks in such a way that for each path from the clock source to sink has same delay time under some wire delay model. Measuring if the delay times are the same has many approaches so that there are many publications in this field, such as [16, 15, 18, 17, 12, 14, 13, 20]. They design their routing algorithm to solve the zero-skew clock routing problem distinctly. Although they are different, they motivate many works on low power zero-skew clock routing. Additionally, as we will introduce in the coming chapters, our LPGC placer built an "pseudo clock router" for evaluation on gated clock topology. This pseudo router were also motivated by some of the zero-skew clock router. We now introduce the classic zero-skew clock router in the coming sections.

The Method of Means and Medians

The first clock routing algorithm is the *H tree* which is a well-known algorithm. However, it could not solve the problem with unevenly distribution of clock sinks. This motivated authors in [13] presented a top-down clock routing algorithm called method of means and medians(MMM).

The MMM algorithm follows a technique very similar to the H-tree algorithm by recursively partitioning a circuit into two equal parts, and then connects the center of the mass of the whole circuit to the centers of the mass of the two sub-circuits.

The algorithm is simple and yields good results. Let L_x be the list of points P sorted according to the x-coordinate. Let P_x be the median of the L_x. Assign points in list to the left of P_x to P_L/ Assign the remaining points to P_R. Due to the geometric nature of

the problem, we may consider the partition of the point set as the partition of a region. Thus P_L and P_R partition the original region by x-median into two sub-regions with an approximately equal number of points in each sub-region. Similarly, P_B and P_T represent the division of P into two sets about the y-median.

The basic algorithm first splits P into two sets(arbitrarily in the x or y direction.). Assume that a split P into P_L and P_R is selected. Hence, the algorithm routes the center of the mass of P to each of the center of mass of P_L and P_R respectively. The regions P_L and P_R are then recursively split in the y direction (the direction opposite to the previous one). Thus splits between x and y are introduced on the set of points recursively until there is only one point in each sub-region. An example of this algorithm is shown in Fig. 2.14.

Notice that basic algorithm discussed above ignores the blockage and produces a non-rectlinear tree. It may also possible that some wires may intersect with each other. It is also possible that some wires may intersect with each other. In the second phase, each wire in the tree can be converted so that it only consists of rectlinear segments and avoids blockages and other nets.

The Geometric Matching based Algorithm

Another binary tree based routing scheme is presented by Kahng, Cong and Robins [15, 16]. In this approach, clock is achieved bu constructing binary tree using recursive *geometric matching*. We call this algorithm *Geometric Matching Algorithm(GMA)*. Unlike MMM algorithm which is a top down algorithm. GMA works bottom up. Let us start by defining the geometric matching.

Given a set P of n points, a geometric matching on P is a set of $n/2$ line segments whose endpoints are in P, with no two line segments sharing the endpoint. Each line segment in the matching define an *edge*. The cost of a geometric matching is the sum of the lengths of its edges.

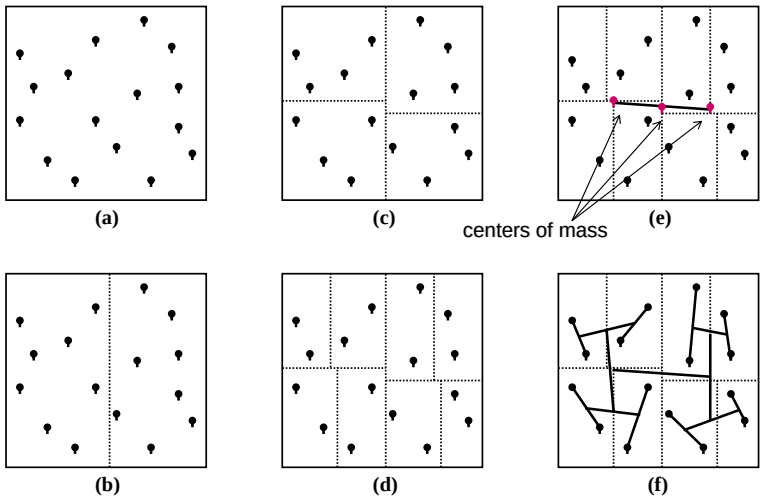


Fig. 2.14: Method of mean and median clock routing algorithm.

The Deferred Merge Embedding Algorithm

The purpose of deferred merge embedding algorithm [17, 18, 14] was to improve the two techniques which we have discussed. This DME algorithm could minimize the total clock wirelength with zero-skew under any given clock topology. In other words, this algorithm should be performed after the clock topology is determined.

The MMM and GMA are the algorithms could solve clock topology construction and clock routing at the same time while DME just solve the second one. However, this property make DME a multi-purpose clock router since different clock topology could brings different advantages and disadvantages. We will introduce the optimal clock topology which optimized the total clock power.

A generic DME algorithm is a two phase: *bottom up merge* and *top down placement*. The name placement here just referred to determine the precise locations for every steiner point (merge point) in order to eliminate the skew. First, we define *merging segment* and *Tilted Rectangle Region*(TRR) for the clarity in this section.

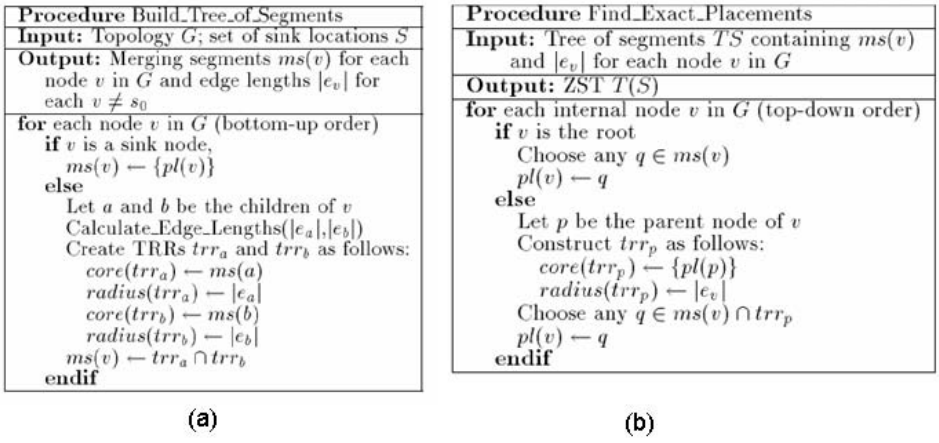


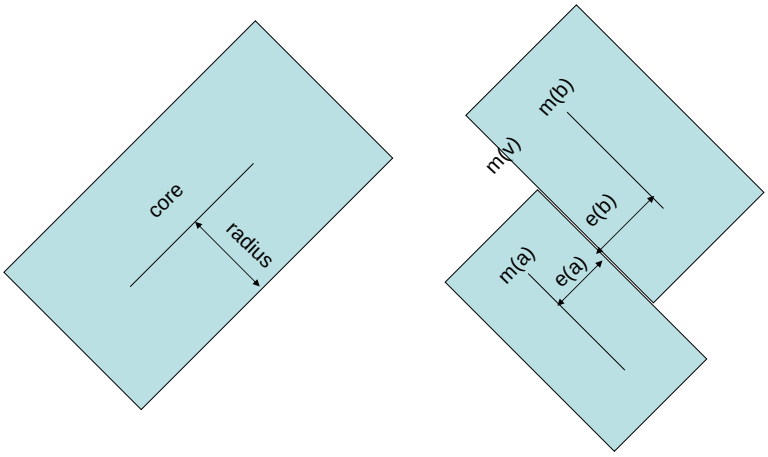
Fig. 2.15: The DME algorithm.

The bottom-up phase is recursively merging each node's TRR in order to find out all the merging segments which we call it *tree of merging segments*, Fig. 2.17 is an example. Given the tree of merging segments corresponding to a clock topology, the top-down phase resolves the exact positions of the internal nodes with minimum wire cost while preserving zero skew. DME algorithm requires the pre-determined clock topology so that this algorithm could be applied to many clock routing problem under different objective function as long as providing the related clock topology.

Solid line are merging segments and dotted lines indicated edges between merging segments. The Squares represent the clock sinks.

2.7.2 Low Power Zero-skew Clock Routing

For a long time, the general theory behind clock design was that the signal should be kept as clean as possible, and that a circuit designer should not interrupt or disable a clock signal. Traditionally, all the clock routing algorithms aim to solve the clock skew and minimum clock wirelength. However, as the technology keep improving and keep shrinking the feature size, the interconnect power now has the same magnitude as compared to gate power [3, 4, 5, 21]. Clock network is a network with long wires and highly switching activity, hence, the higher power. It has been researched that the clock power is



(a) Tiled rectangle region in DME, the radius of a TRR is the distance between its core and its boundary

(b) The merge of TRRs, the merging cost between T_{Sa} and T_{Sb} to be $|e_a| + |e_b|$

Fig. 2.16: A simple example of tiled rectangle region (TRR).

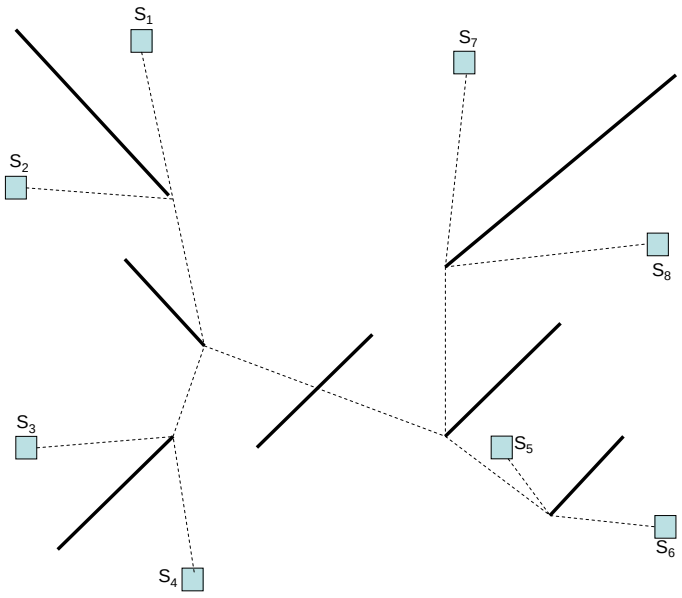


Fig. 2.17: An Example of clock tree merging segments.

at least 40% of the total power for a typical microprocessor [21].

Therefore, today's research focus on distributing a clock network with minimizing clock skew, wirelength, timing and finally clock power. It is a new field with many chances and improvements in it.

Activity Driven Clock Design

Clock power can be saved by disabling clock signals from inactive registers(flip-flops) in idle circuit parts. We can carefully insert the controlling logic gates to control whether the register that the gate controlled is active or not. by shutting down idle registers, the combinational circuitry and data transfer path would not be triggered and switched so that save a substantial amount of power. This technique has a so called name *gated clock routing*, but someone named it *clock gating*.

In this area, people tried to distribute clock networks with inserting the control gates. By performing sophisticated algorithms, one could achieved the low power clock network. Since registers that should be gated or merged together may be placed far apart, it is possible that gating control wirelength will finally be increased routing and the total clock power may finally goes up.

The authors in [24] suggested that by putting the pseudo net between commonly-gated registers together in the netlist and then the placer would tends to put them closely. However, here are the drawbacks in this idea:

1. The amount of pseudo nets would be enormously so that the weight of net which made by manually dictated would probably be failed when handling larger circuits.
2. There is no way to handle the global optimization.
3. The pseudo net would destroy the timing of critical nets.

Gated Clock Routing for Microprocessor Design

In this publication, the authors have proposed an approach for clock tree routing, that ensures zero-skew and reduces the power dissipation by minimizing the effective switched

capacitance at the internal nodes of the tree. This algorithm refined the DME algorithm which we have known as a multi-purpose clock router. The authors [25, 22] replace the traditional clock routing objective from minimizing total wirelength to minimizing total switched capacitance.

The algorithm first formulate the clock network power and clock control network power as follows: Consider a clock tree without gates, the power dissipation on a particular clock edge e_i is defined as

$$Power_{e_i} = |e_i|c_0 f V_{dd}^2 \tag{2.7}$$

where $|e_i|c_0$ represent the switched capacitance of this clock edge(c_0 is the unit wire capacitance and e_i is the wirelength of the clock edge e_i). The power dissipation for gated clock network, the Equ. refEqungate should be modified as:

$$Power_{e_i,gated} = |e_i|c_0 f V_{dd}^2 P(e_i) \tag{2.8}$$

with additional consideration on the switching activity of the clock edge e_i. Since the operation frequency and supply voltage are fixed we define *effective switched capacitance* at any edge as:

$$SC_{ei} = (|e_i|c_0 + C_i)P(e_i) \tag{2.9}$$

where the C_i represents the load capacitance of the clock edge e_i. Thus, the switched capacitance of the entire gated clock network could be define as:

$$SC_{GCN} = \sum_{\forall e_i} SC_{ei} \tag{2.10}$$

Finally, we take the gated clock control network into account. We can follow the definition as same as Equation 2.10:

$$SC_{GCCN} = \frac{1}{2} \sum_{\forall g_{e_i}} (c_0|g_{e_i}| + C_g)P_{tr}(g_{e_i}) \tag{2.11}$$

where the $|g_{e_i}|$ and $P_{tr}(g_{e_i})$represent the wirelength and transition probability of the gated clock control wire g_{e_i}. We assume that the source of the gated clock control network was

laid in the center of the chip, C_g means the control gate's capacitance. The objective of this routing algorithm is to minimize the term $SC = SC_{GCN} + SC_{GCCN}$ through DME algorithm.

First, given all the clock sinks' locations, they could built a binary tree which represent the gated clock topology which could minimize SC. Hence, they pass this topology to DME router hence force it to route this clock network with zero-skew. As we could see from the idea of this algorithm, there are many aspects which we could improve. Since they have no process for global optimization, they might achieve either a non-optimal solution or a poor solution. Our work improve these drawbacks then provide a process or a good seed for global optimization.

Chapter 3

Low Power Clock Network Placement Algorithm

The power consumption is an important problem in modern design, and the clock network power is 40 % of total power consumption. Because of power consumption, we focus on reducing clock network power. Two different methods, clock gated method and prescribed skew routing method, are used to reduce the clock power consumption.

- *Gated Clock Network*

  Clock gating is a well-known technique to reduce the dynamic power consumption in clock network of a digital circuit. It disables portions of the clock tree distribution for saving the dynamic power. The clock network power consumption can be reduced more than 30% in a general case.

- *Non-Balance Clock Tree (prescribed skew routing)*

  According to Huffman Coding algorithm, the clock network topology might be an un-balanced tree structure for saving the power consumption. However, the traditional clock tree synthesis flow tends to construct a balance tree because of the balancing of clock skew. Without the delay target information, the non-balance clock tree would cause a clock skew problem. We modify the cost of clock skew scheduling and use the linear programming method to minimize clock power consumption under the timing and skew constraints.

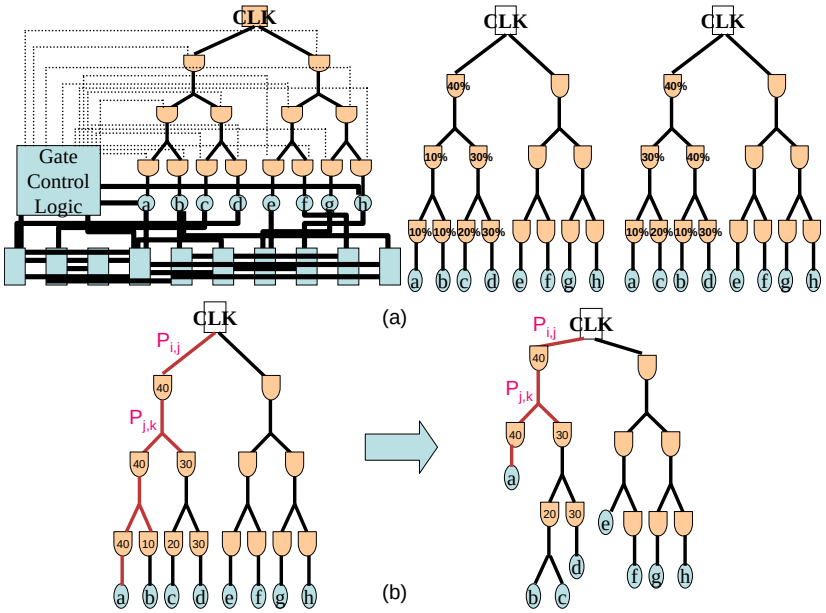


Fig. 3.1: The gated clock algorithm.

In Fig. 3.1.a, the left hand side is a clock gated system illustration, the right hand side is the optimized gated clock topology for reducing the switching activity. Each node of clock tree has corresponded activity and the clock topology is constructed by partition methods. The Fig. 3.1.b is a comparison between the balanced topology and non-balanced topology. The non-balanced tree can be constructed for minimizing clock network switching activity if the delay target time permits. In order to optimize the total power consumption, we apply block activity and delay target time to our clock topology constructor.

In the following section, we will begin with the flow chart of our algorithm and circuit analysis timing and activity. Then, a gated clock constructor and clock skew scheduling are described in detail. Finally, there are our main placement core and modified gated clock network router.

3.1 Main Flow

Our low power clock network driven placement is targeted on reducing the clock network power consumption under the traditional placement constraints. We would like to focus on the dynamic clock network power consumption which is induced by the switching capacitance. In order to reduce the clock network power consumption, first, we construct a good clock topology which is based on the gated clock algorithm and low power clock skew scheduling before placement. Then, we develop a placement algorithm to increase the realizability of this gated clock topology.

To put the matter simply, our low power clock network placement can be divided into three steps. In the first place, we perform circuit analysis. We first perform an activity analysis to obtain the activity of each block. Then, a simple timing analysis is executed to get the delay time of each block. With those activities and delay times, our clock topology can be constructed. According to Huffman Coding algorithm and the clock skew scheduling, a good un-balance gated clock topology can be established.

In the second place, we translate the above clock topology into the original circuit netlist. The weight of original netlist is modified to construct our clock network during placement, and several pseudo blocks are fixed for constraining the clock network. We utilize and modify the placer "Umpack" [63] in our algorithm to achieve our goal. In order to reduce the total power consumption, the clock topology construct algorithm and the weight of netlist are modified by the the assumption of [53] which is reviewed in chapter 2. In the third place, we perform a "GCR" [22] which is modified by receiving delay target time and merging for a non-balance low power clock gated topology.

Fig. 3.2 is the flow chart of out work which is used for the multilevel placement. At the beginning, the timing and activity of circuit netlist are analyzed. Then, a gated clock topology is generated. After that, the clock network topology is translated into a weighted netlist, and the multilevel placement is constrained by this clock topology. Finally, the modified clock router "GCR" [22] is used to verify the result of placement.

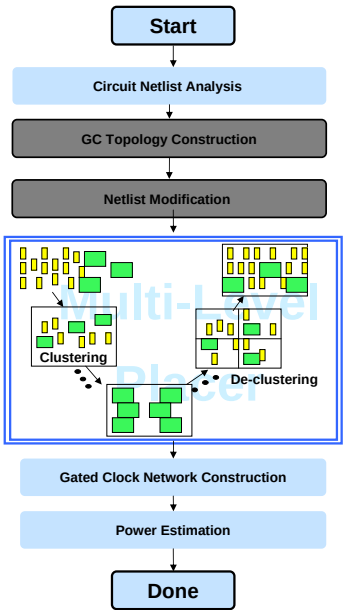


Fig. 3.2: The flow chart of low power clock network driven placement.

3.2 Circuit Netlist Analysis

The benchmarks we use are modified from ISCAS89 (ISCAS89 benchmarks [58] are translated into GSRC format by using the flow proposed in [59]). In order to find the activity relation, test patterns are generated to each block by a random number of related blocks and the total number of test patterns is determined by the number of circuit blocks. The test patterns are fed into our clock network topology constructor, and the activity relation between each block is estimated.

For the reason to find the delay time of each block especially for the flip-flop, the simple static timing analysis is performed to obtain the delay between register and block. The delay target time of each clock sink is important for creating a non-balance clock topology. The delay of each net is transfered to a net weight, and those net weights are fed into the placer for minimizing the clock period.

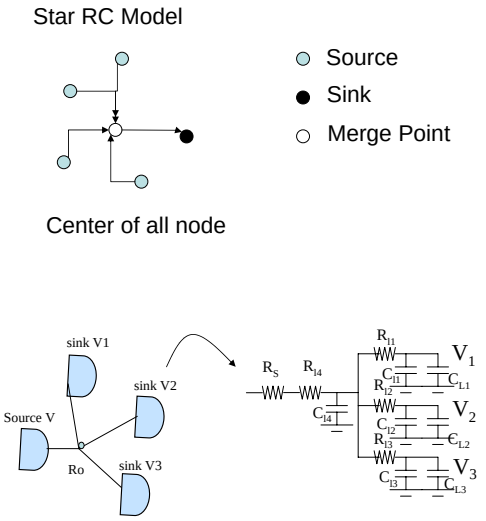


Fig. 3.3: The star RC tree model

3.3 Gated Clock Topology Construction

Many issues such as clock power, clock skew, area, wire length etc are considered for a general VLSI design. If we want construct a clock topology before placement, we need to consider those physical constraints. For this reason, we perform the algorithm of partition "hMETIS" to consider those physical constraints before performing our partition based placement and incorporate all constraints to construct a clock network topology. The test patterns are translated into weighted nets, and the delay target time of each flip-flop is translated into weighted node. The above information is fed into the partitioner "hMETIS" to recursively construct the clock topology.

The above gated clock topology construction procedure is show in Fig. 3.4 and Fig. 3.5. Fig. 3.4(a) indicates that the test patterns consist of the usage of each block and the distribution of test patterns activities. The modules M1, M2, and M5 are used by the test pattern I1, and the modules M2, M3, and M4 are used by the test pattern I2 and so on. The activities of test patterns are analyzed by a test pattern stream. After the activity an-

alyzing being done, we translate the test patterns into weighted nets as illustrated at the top of Fig. 3.4(b). We also integrate the area of each flip-flop and its skew information calculated in the next section into a weighted node for partitioning. The weight of each weight node i is equal to translate the

$$W_i = \beta \times W_i^{area} + (1-\beta) \times W_i^{timing}, \tag{3.1}$$

where W_i^{area} and W_i^{timing} are the area and timing weights of flip-flop i, respectively, and β is a constant. Finally, the topology of clock network is construct by a recursive partition shown in Fig. 3.5. The cost function and balance function are shown in Equation (3.2) and (3.3).

$$\text{cost function} = \sum_{1}^{n} \alpha_i \times N_{CUTi} \tag{3.2}$$

$$\text{balance function} = \frac{(\sum_{left} W_i - \sum_{right} W_i)}{\sum_{1}^{n} W_i} \leq k\% \tag{3.3}$$

Here, N_{CUTi} is the number of i^{th} cut. With the above cost and balance functions, the partition is performed recursively until the number of registers in each set is less than a suitable threshold.

3.4 Clock Skew Scheduling for Low Power

Generally, the clock skew scheduling is effective in reducing the clock wire length [64, 54]. In this thesis, we utilize the clock skew scheduling methodology to reduce the switching capacitances of a gated clock network, and hence reduce the power consumption. The clock network switching capacitance function can be formulated as

$$\sum_{i}^{n} \alpha_i C_0 L_i^{ck}, \tag{3.4}$$

where α_i is the activity, C_0 is the unit capacitance of clock wire, and L_i^{ck} is the length of targeted gated clock net.

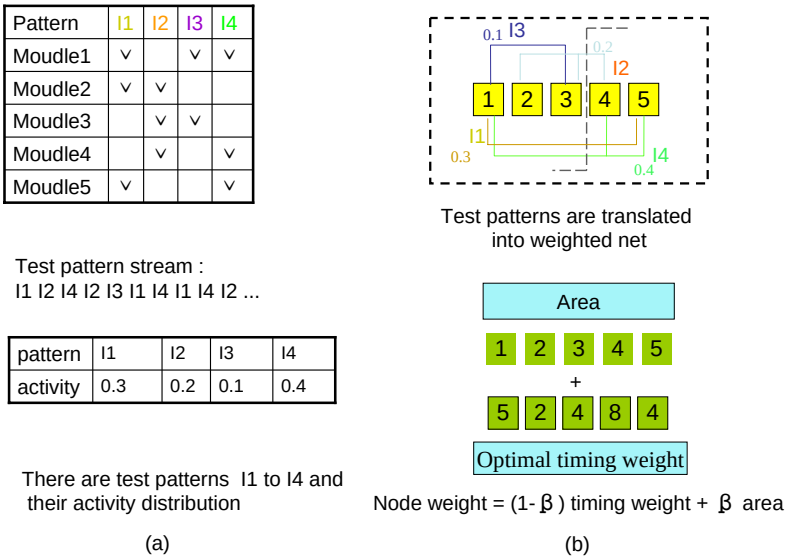


Fig. 3.4: The translation between clock topology construction and partition problem.

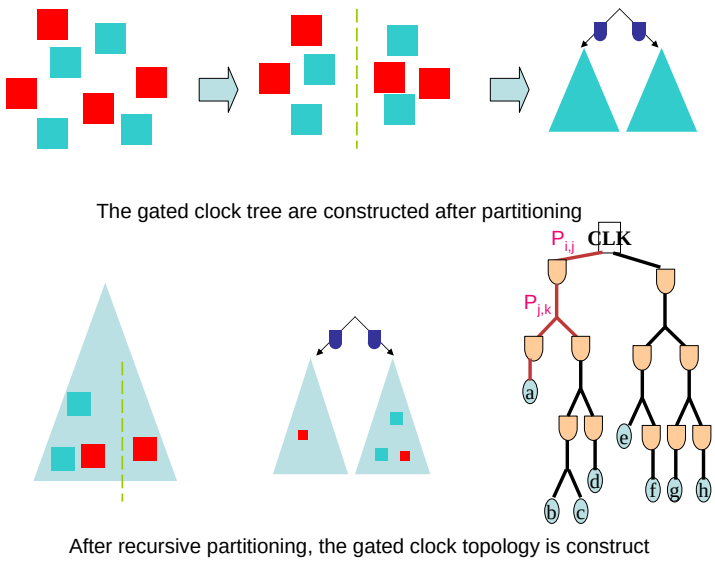


Fig. 3.5: The topology construction using recursive partition.

The traditional clock skew optimization targets on clock period minimization. However, our work is to optimize the power under the race condition with a given target clock period. To avoid the race condition, the clock skew optimization must satisfy the following two constraint equations.

$$T_{skew(i,j)} \geq T_{hold} - \underline{delay}(i,j), \tag{3.5}$$

$$T_{skew(i,j)} \leq P - T_{setup} - \overline{delay}(i,j). \tag{3.6}$$

where the $T_{skew(i,j)}$ is the difference of delay target of i_{th} and target of j_{th}, $\underline{delay}(i,j)$ is the minimum delay between i_{th} block to j_{th} block, and $\overline{delay}(i,j)$ is the maximum delay between i_{th} block to j_{th} block. P is the targeted clock period, T_{hold} is the hold time of the flip-flop and T_{setup} is the setup time of the flip-flop.

First, the delay between flip-flops translate into a constrained graph. To find the optimization of the problem we change the problem into the linear programming problem and the simplex method [70] is used to solve the problem, and the answer need a reference value to solve we assumed it = R≈ MAX(W, H) /4 in [64].

We gain the above answers for two purposes, first reason is that the answer will be used in construct clock topology and second reason is used to the gated clock routing after placement. The original answer is a delay target timing of each flip-flop and the answer translates into the term of weighted clock wire length. As we constructing the clock network topology, the clock length will be multiple a scalar xi of target as the Fig 3.6 below. The more close to the center of the circuit (we assume that we clock source is the center of circuit boundary) the X_i is more small, because it not easy to close to the center of the circuit boundary where is a small and crowded region.

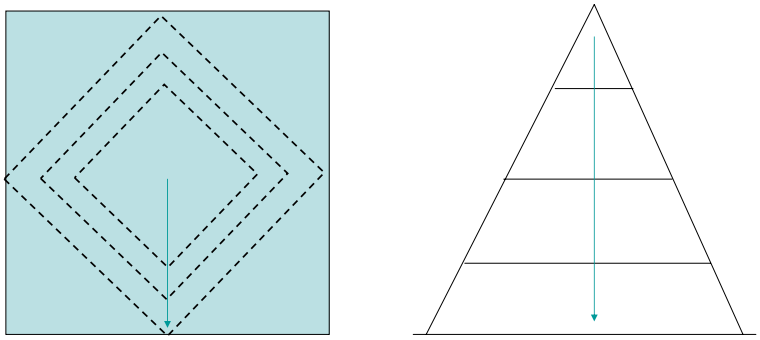


Fig. 3.6: The relation of the parameter X_i

3.5 Placement on Low Power Clock Network

Two kinds of algorithms are used here to construct this clock topology into a real clock network. In the first place, we apply “Pseudo Net and Pseudo Node” algorithm to modify original circuit netlist which is fed into the placer. The topology is translated into the pseudo weighted net and additional pseudo weighted node. The second algorithm is “Fixed Node on the Manhattan Circle” the clock gated node is fixed on Manhattan Circle which is determined by the length we estimate in the early stage. The placer “Umpack” is used in our algorithm and we modify it for second algorithm to parser our low power clock topology to the placer.

3.6 Low Power Clock Network Routing

Our clock network routing algorithm is based on the [22, 54] which is also a DME like method. The cost function in [22] is only switching capacitance. The prescribed clock skew routing is targeted on the minimization of clock network by using the total delay

time of each path is just only satisfied the delay target time constraint. We modify the cost function and provide the delay target time into the original "GCR" [22] router. The cost function of [22] is modified with not only switching activity but also maximum delay target time [54]. The delay target of each flip-flop is computed by clock skew scheduling and the constraint function is obtained by the linear programming method.

The delay constraint functions of linear programming problem are gained by those formula.

$$\begin{aligned} & MIN(Delay_{i\rightarrow X}) - MAX(MIN(Delay_{X\rightarrow j}), Clk_{target}) \\ & \quad i,j \in registers\{1,..,N\} \end{aligned} \tag{3.7}$$

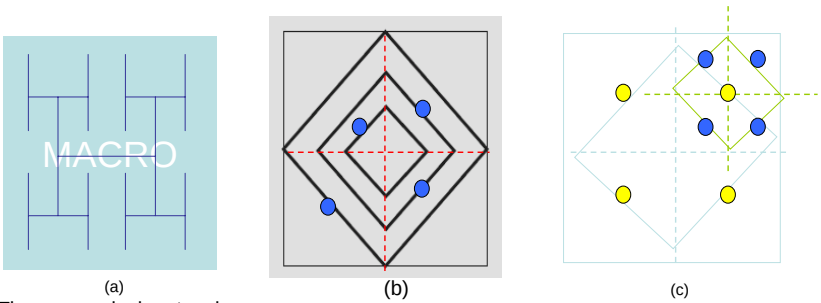
The cost function is the total sum of the delay target weight and switching capacitance.

$$\begin{aligned} & CostFunction: \\ & \quad Cost_i = (1-\alpha) \times C_{switch} + \alpha \times delaytime_{sub-treei} \end{aligned} \tag{3.8}$$

In the first place, the cost is computed for each pair of clock sinks in this clock network router and merge the minimum cost of all. When all node are merged in one topology, then the real merge point of the topology is top down estimated under the timing constraints.

3.7 Clock Tree Constraint in Placement

In order to carry out our proposed clock topology, it needs some constraints to make the flip-flopsof clock topology to place in the region we want. The pseudo net and pseudo node method we use is modified form [64, 68, 69] which is using a Manhattan Circle (every point on the Manhattan Circle to the center point is the same distance) as constraint. The Modified Manhattan Grid and Pseudo Pins Method are used here to constrain the registers in a multilevel method. If there are macro blocks in the design, we take a H-tree to estimate our constrained pseudo pins to the center of the macro blocks Fig. 3.7.(a). The pseudo nodes are fixed to constrain the flip-flop to the region that we want in Fig. 3.7(b). In Fig 3.7.(c), the method of Modified Manhattan Grid and Pseudo Pins performs each un-coarse level during the placement.



(a)
The macro clock network is regard as H tree

(b)
The pseudo pin method for prescribed skew in [10,16,17]

(c)
The Modified Manhattan Grid Method from [10]

Fig. 3.7: Algorithms of clock topology based placement

3.8 LPCN Placer Core

The placer core we use is multilevel placer "**UMpack**" from *University of Michigan EDA LAB* and modify it to suit our goal. The "**UMpack**" is a multilevel placer with good performance, robust and also a good ability in handling macro cell. The weight of the netlist (net weight and node weight) and the fixed pseudo blocks are modified by our topology constructor (partition-er) before entering the placing level. After the placement, we perform a Star Model Tree RC timing analysis in [67] which can provide a more precise estimation of timing analysis and find the new clock skew scheduling for reducing the power consumption.

3.9 Application of Our Work

Recently , [53] has good performance in reducing power consumption, but they do not focus on the low power clock network based on gated clock method. For this reason, we apply our work "low power topology based on gated clock method " to the low power driven placement [53]. Our work is to substitute our low power topology algorithm for the algorithm QTS (Quick Tree Synthesis) of [53]. To construct a low power gated clock network can release the original crowed clock network. In the Fig.(3.8. a), there

is the original register clustering method, and the flip-flop are clustered in order to reduce clock network power. In the Figure(3.8. b) there is an illustration of net activity based register clustering which is proposed in [53]. In the Fig.(3.8. c) there is application of our method “low power clock network” to low power driven placement. Because the flip-flop has high connective to the general block, and if we release the clock network constraint, more power consumption of the general signal wires can be saved.

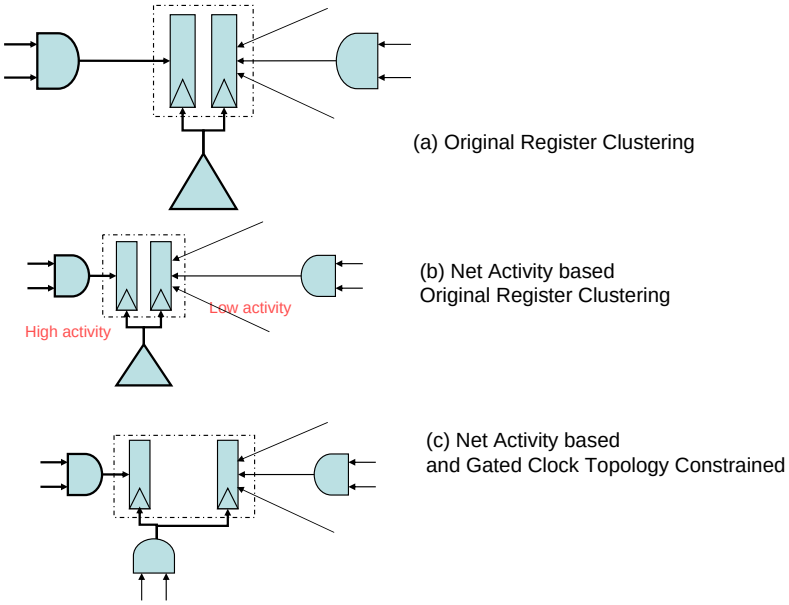


Fig. 3.8: Our application to power aware placement.

In order to integrate our work to [53], we reduce the weight of clock tree pseudo net and the block , because the power consumption of clock network has a great reduction by gated clock clock. Therefore, we do not need to construct a whole clock topology, and the execute time can be reduced. The threshold of stopping topology construction is determined by the number of flip-flops in the circuit.

Chapter 4

Experimental Results

We implement our clock network driven placement in C++ language on a dual processor 2.8GHz HP workstation with 16GB memory, and acquire the source code of Umpack [63] and a gated clock router [22] from the authors. The execute file "hMETIS" is download from the web site [73].

We use the benchmarks of ISCAS89 [58] which have the flip-flops information in our experiment. The benchmarks are translated into GSRC format, one of the most general placement format, by [59]. Those benchmarks are listed in Table 4.1. The test patterns and the activity of each benchmark are reasonably assigned. The number of test patterns is determined by the block number of each circuit, and the number of related blocks of each test pattern is randomly determined and is greater than 1. The RC process technology parameter is 0.13μm [74]. In order to demonstrate our performance, we compare two different placement flows in the same benchmark and with the same random seed. One is

Circuits	Node	Net	Register	Pin
S27	13	17	3	5
S526	214	217	21	9
S838.1	478	512	32	35
S1423	731	748	74	22
S1494	653	661	6	27
S5378	2958	2993	179	84
S9234	5825	5844	218	41
s13207	3024	2993	669	70

Table 4.1: The Benchmark Circuits.

Circuits	Area	Wirelength	Clock Period	Clock Network SC
S27	6.93E+04	1558	4.29E+05	48.8795
S526	5.84E+05	33515	2.09E+06	847.0610566
S838.1	1.02E+06	78808	1.91E+06	1897.8062
S1423	1.62E+06	105196	9.83E+06	3172.6693
S1494	1.34E+06	131132	3.68E+06	531.4452
S5378	5.24E+06	514586	4.65E+06	12736.1439
S9234	9.27E+06	800415	8.55E+06	24668.5241
S13207	1.49E+07	23465000	1.37E+07	135891.0173

Table 4.2: The Result of Original Placement.

Circuits	Area	Wirelength	Clock Period	Clock Network SC
S27	6.55E+04	1058	4.28E+05	30.22908
S526	5.84E+05	35626	1.98E+06	785.67313
S838.1	1.02E+06	56478	1.74E+06	1631.35708
S1423	1.62E+06	106949	9.80E+06	2880.15898
S1494	1.23E+06	128838	4.00E+06	167.70029
S5378	5.06E+06	558804	4.67E+06	9661.25366
S9234	8.94E+06	901713	8.90E+06	23002.12715
S13207	1.48E+07	23543100	1.13E+07	83318.33988

Table 4.3: The Result of Low Power Clock Network Driven Placement.

our proposed algorithm and the other is the traditional placement flow which considers the wire length and area. We compare the result with total circuit area, wire length (without clock wire length), clock period and clock network SC which is the switching capacitance of clock network after a gated clock routing. Table 4.2 is the original results of from the placer “Umpack”, and Table 4.3 is the placement results of our proposed algorithm.
The result comparison of Table 4.1 and Table 4.3 is shown in Fig. 4.1. This figure shows that our low power clock network driven placement method can reduce the power consumption of clock network 20 % in average without much overhead.

Because of our proposed algorithm aiming to reduce the clock network power, we apply this algorithm to a low power driven placement [53]. Table 4.4 is the result of placement which has the same assumption with [53], and Table 4.5 is the result of the placement which applies our clock network topology algorithm inside. We compare the result with the total circuit area, wire length (without clock wire length), clock period,

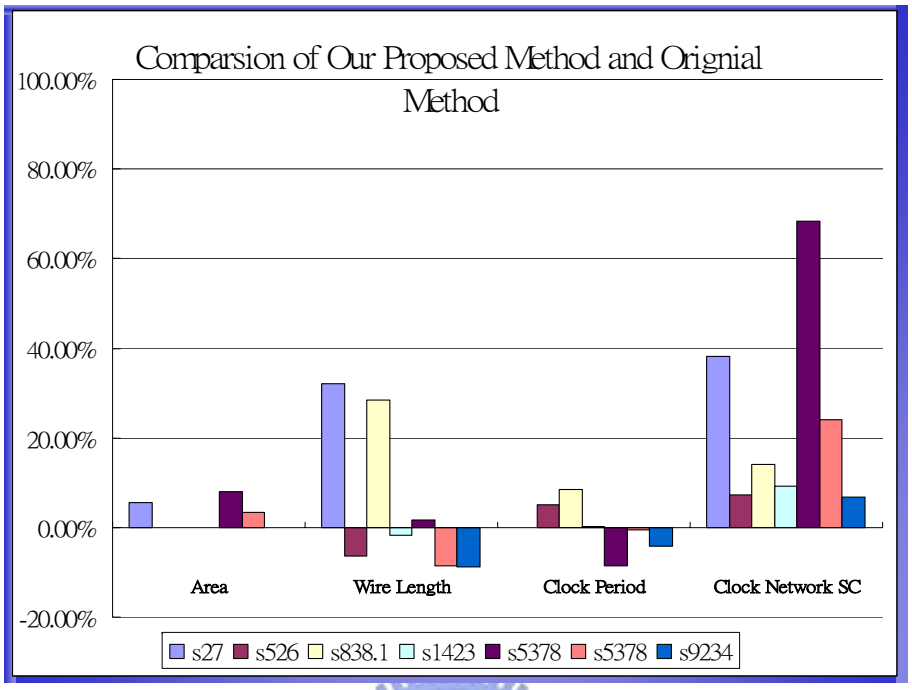


Fig. 4.1: Result of our low power clock network driven placement

clock network SC, and the signal wire SC is the total switching capacitance of signal wires.

The result comparison of Table 4.4 and Table 4.5 is shown in Fig. 4.2. It shows that our algorithm can still reduce over 3% power consumption of interconnect than the low power driven placer [53].

Fig. 4.3(a), (b), and (c) are the flip-flop distributions of the benchmark s9234 after placement with the same assumption of [53], with our algorithm, and with our algorithm

Circuits	Area	Wirelength	Clock Period	Clock Network SC	Signal Wire SC
S27	6.93E+04	1422	4.29E+05	36.59320	853.92
S526	5.84E+05	37972	1.90E+06	722.80646	19313.50
S838.1	1.02E+06	66266	1.93E+06	1685.77015	46882.10
S1423	1.62E+06	104760	9.80E+06	3064.85230	78031.40
S1494	1.25E+06	131200	4.00E+06	337.99560	36726.00
S5378	5.24E+06	610477	4.60E+06	11901.32303	748164.00
S9234	9.27E+06	927449	8.62E+07	22240.39809	3133980.00

Table 4.4: The Result of Low Power Driven Placement

Circuits	Area	Wirelength	Clock Period	Clock Network SC	Signal Wire SC
S27	6.55E+04	1153	4.28E+05	21.94929	716.64
S526	5.84E+05	34377	1.81E+06	788.00000	17540.90
S838.1	1.02E+06	58242	1.73E+06	1681.17077	46135.40
S1423	1.62E+06	103948	9.80E+06	2970.00000	76419.60
S1494	1.25E+06	126774	4.00E+06	158.09654	35467.20
S5378	55.24E+06	611002	4.62E+06	9021.12080	739592.00
S9234	9.27E+06	865521	8.90E+06	22839.54202	3048580.00

Table 4.5: The Result of Our Application to Low Power Driven Placement

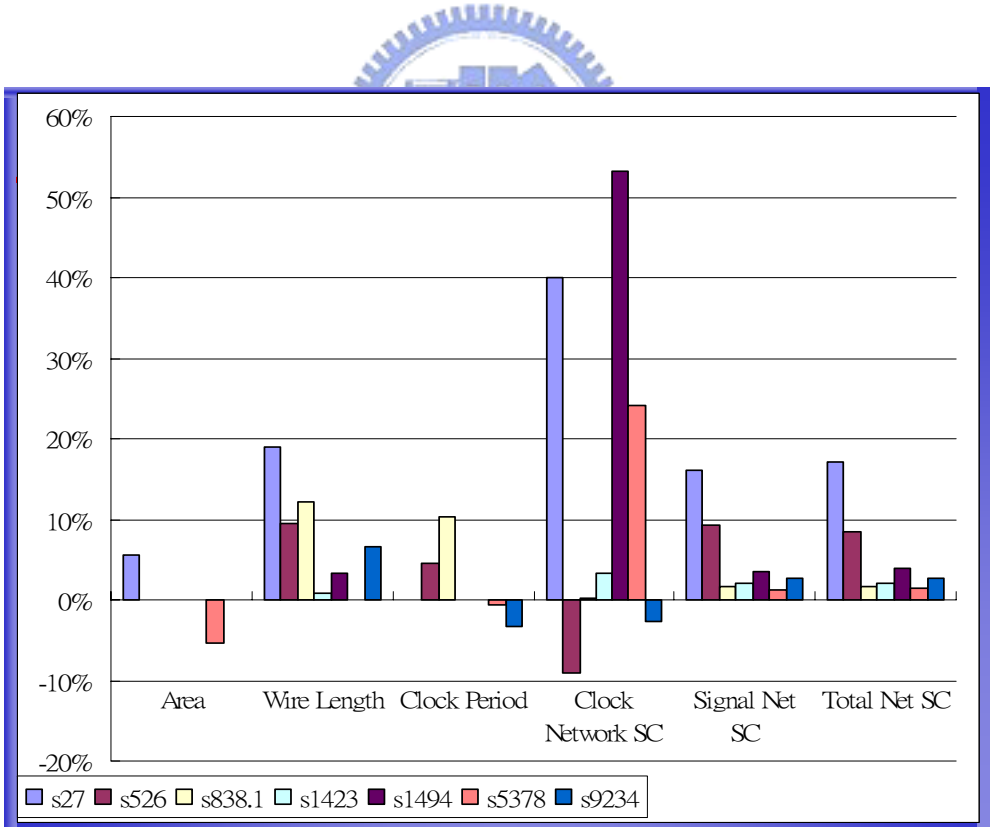


Fig. 4.2: The new application result of our work.

inside the low power driven placement [53], respectively. It shows that the clock network distributions of filp-flops cluster together tightly in (a) and (b), and that the clock network distribution of flip-flops is loose slightly in (c) to reduce the total power consumption. figure

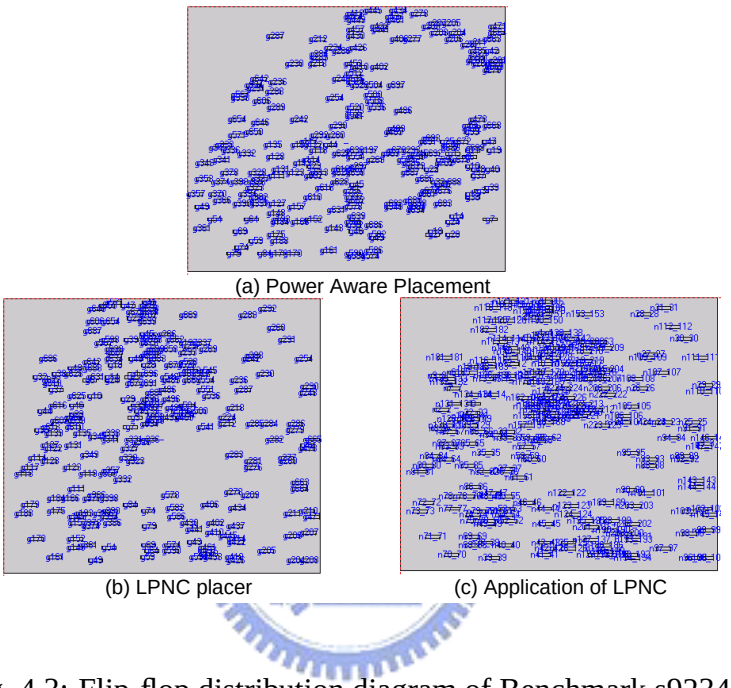


Fig. 4.3: Flip-flop distribution diagram of Benchmark s9234.

Fig. 4.4 is the a placement result of our low power clock network with the goal of minimizing the power dissipation of clock network. In order to minimize the power consumption of clock network, a few of blocks are fixed in the corner of left bottom. Fig. 4.5 is the result of our application to low power driven placement with the goal of minimizing the total power consumption of the circuit. Fig. 4.6, 4.7, 4.8, and 4.9 are the placement results of our proposed algorithm for the rest benchmarks.

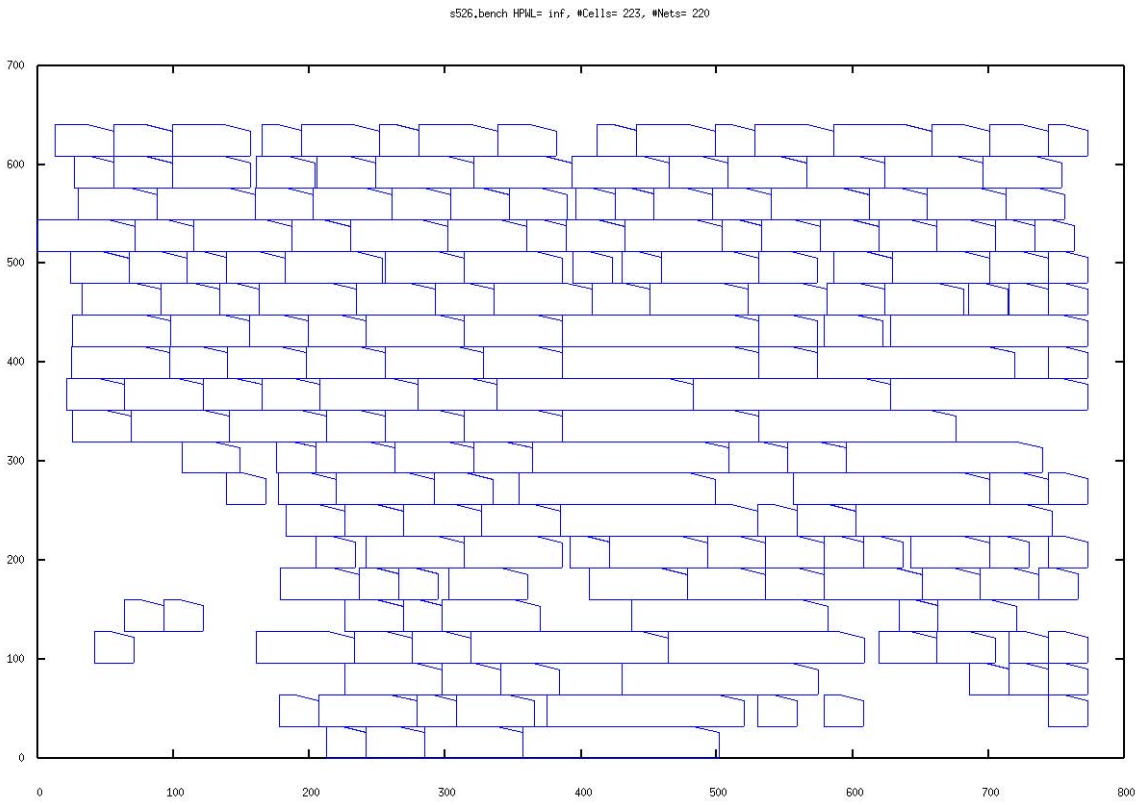


Fig. 4.4: Placement result of benchmark s526.

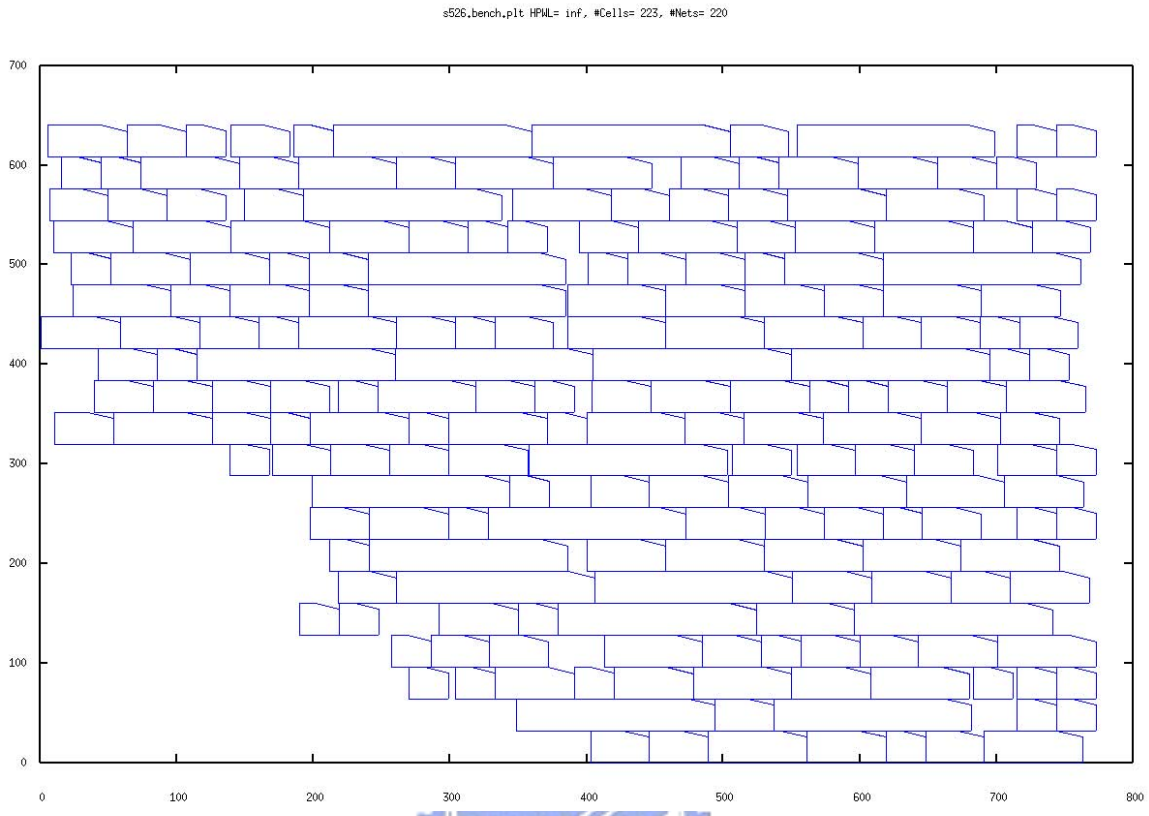


Fig. 4.5: Our application result benchmark s526.

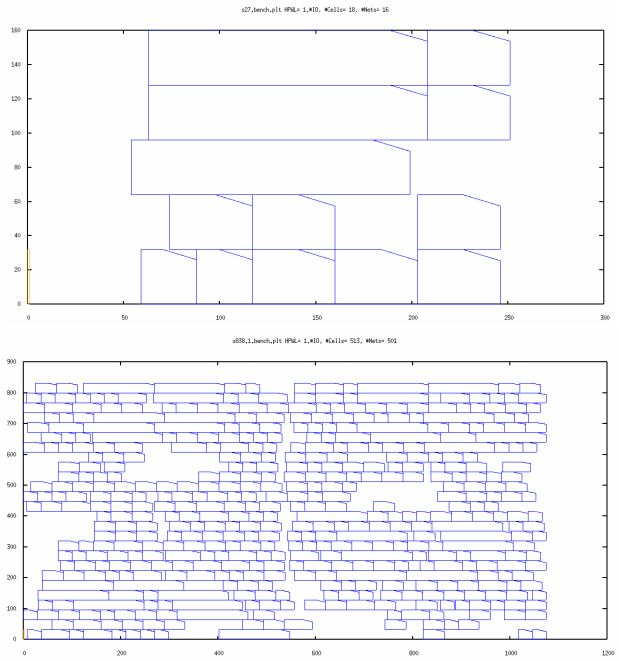


Fig. 4.6: The results of benchmark s27 and s838.1.

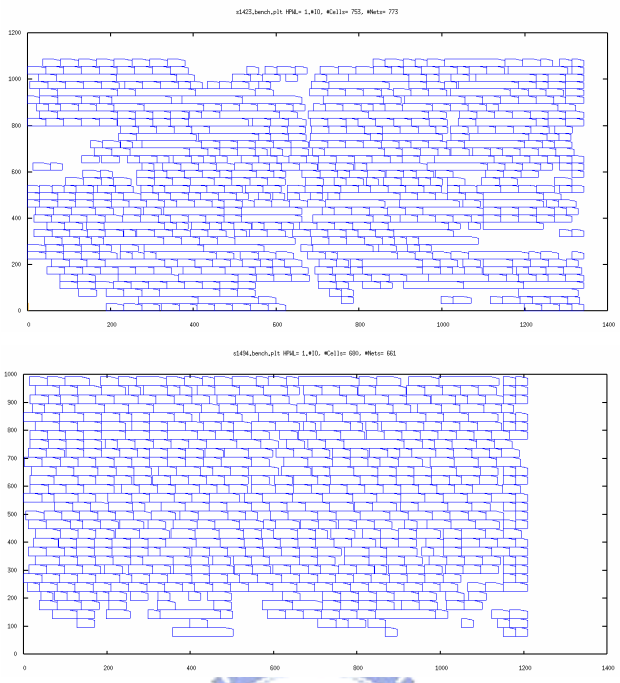


Fig. 4.7: The results of benchmark s1423 and s1494.

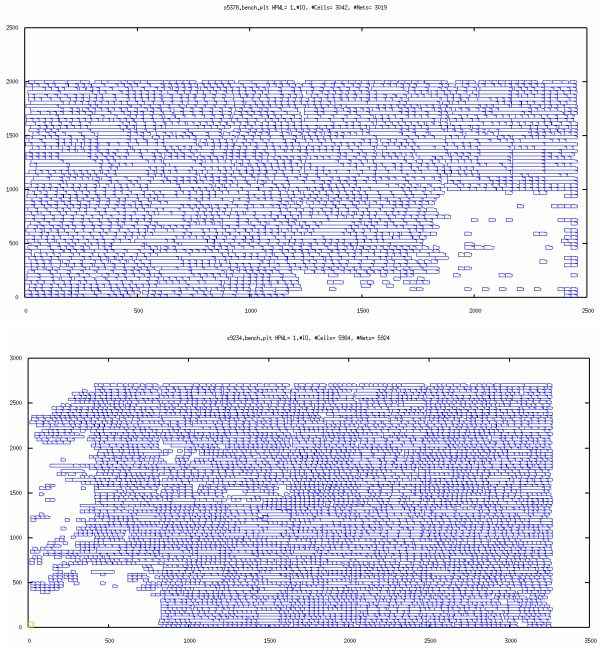


Fig. 4.8: The results of benchmark s5378 and s9234.

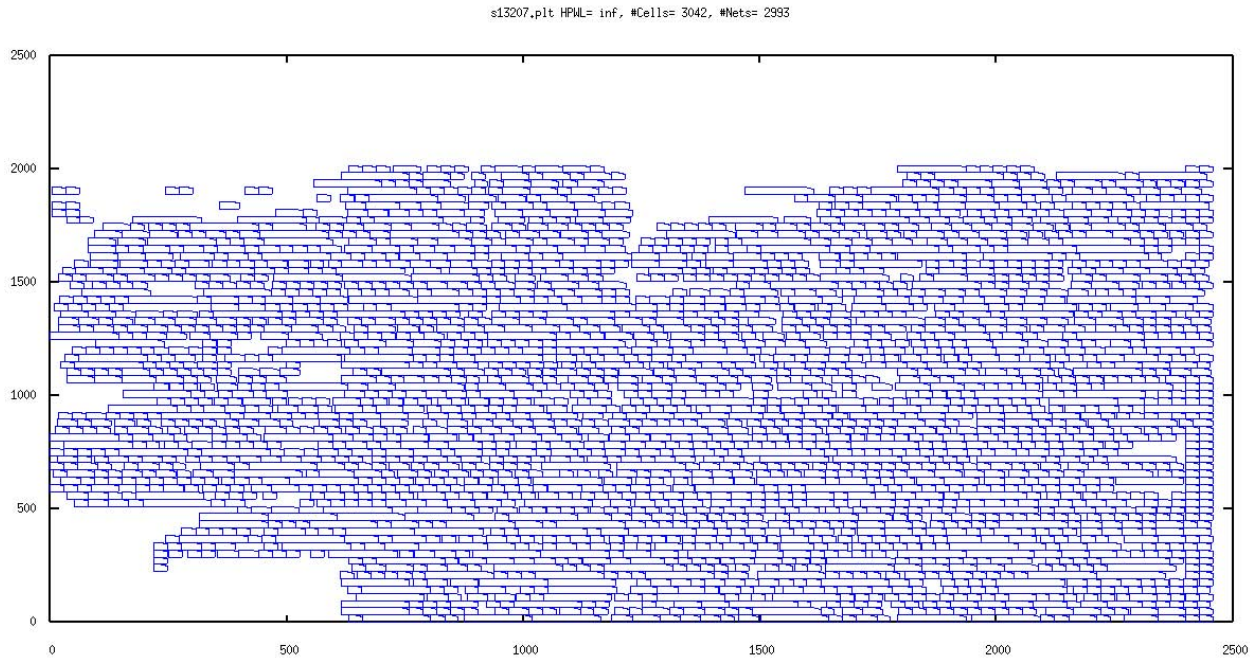


Fig. 4.9: The result of benchmark s13207.

Chapter 5

Conclusions and Future Work

5.1 Conculsions

A great deal of effort has been made on low power issues. What seems to be lacking in low power design, however, is a clock gating driven placement. Only few attempts have so far been made at register placement [64]. We propose a novel algorithm to reduce the power consumption of clock network and integrate it into placement. We propose a novel flow in placement level which minimizes the clock network power before placement. Our result can reduce the clock network switching capacitance over 20 % in average to a traditional placement result and the total power of interconnect of switching capacitance consumption over 3% than a low power driven placement [53].

5.2 Future Work

Process Variation

In modern VLSI design, the process variation becomes more and more serious. The clock network consists of interconnect wires which are sensitive to the variation of process. The researches in process variation of clock network have been developed for a long time. It is very helpful to get a more accurate estimation of clock network wire length for advanced technology if we can develop a process variation based gated clock network topology constructor.

Leakage power

It is known that the power consumption of sequential circuit due to leakage current increases rapidly in recent process. We can combine the leakage current reduction methods [74] with our clock network topology to refine our clock network topology.

Peak current

We know that positive and negative buffers can be used to reduce the peak current which is made by the switching of clock network from power grids [19]. Because of the activity property of our gated clock network, we can arrange the peak current by using positive gate or negative gate to our clock topology. Therefore we can use the gate capacitance and the switching activity of each net to improve our peak current consumption in our special clock network.

Bibliography

[1] Simon Yi-Hung Chen,“A novel Low Power driven Placement Algorithm based on Optimal Gated Clock Topology,” thesis of NCTU, 2005.

[2] Chandrakasan A. P.,Sheng S.,Brodersen R. W., “Low-Power CMOS Digital Design,” *IEEE Journal of Solid-State Circuits*, No. 4, April 1992, Page(s):473-484 vol 27.

[3] Otten, R.H.J.M.; Camposano, R.; Groeneveld, P.R, “Design Automation for Deep-submicron: present and future,” *Proceedings of the 2002 Design, and Test in Europe Conference and Exhibition*, Page(s):650 - 657.

[4] Narayanan D. Duarte, V., and Irwin M. J., “Impact of Technology Scaling in the Clock System Power,” *Proceedings of the IEEE Computer Society Annual Symposium on VLSI, ,* April 25-26, 2002, Page(s):59.

[5] Garrett David, Stan Mircea and Dean Alvar, “Challenges in clockgating for a low power ASIC methodology,” *Proceedings of the 1999 international symposium on Low power electronics and design, ,* August 16-17, 1999, San Diego, California, United States, Page(s):176-181.

[6] K.-Y. Chao, and D.F. Wong, “Floorplanning for low power designs,” *IEEE International Symposium on Circuits and Systems*, 28 April-3 May 1995, Page(s):45-48 vol.1.

[7] Vaishnav Hirendu and Pedram Massoud, “PCUBE: A Performance Driven Placement Algorithm for Low Power Designs,” *Proceedings of European Design Automation Conference,* , September 1993 Page(s):72-77.

[8] Sait Sadiq M. ,Minhas Mahmood R., and Khan Junaid A. , “Performance and low power driven VLSI standard cell placement using tabu search,” *IEEE Congress on Evolutionary Computation,* May 2002, Honolulu, Hawaii, USA, Page(s): 372-377.

[9] Sait Sadiq M. ,Youssef Habib, Khan Junaid A., and El-Maleh Aiman “Fuzzified Iterative Algorithms for Performance Driven Low Power VLSI Placement,” *19th International Conference on Computer Design (ICCD 2001),* 23-26 Sept. 2001, Page(s):484-487.

[10] Jimenez, M.A. and Shanblatt, M.; “Integrating a Low-Power Objective into the Placement of Macro Block-based Layouts,” *Proceedings of the IEEE. Midwest Symposium on Circuits and Systems,* , Ohio, Aug. 2001, Page(s):62-65.

[11] Sait Sadiq M., Youssef H., and Khan Junaid, “Fuzzy simulated evolution for power and performance optimization of VLSI placement” *International Joint INNS-IEEE Conference on Neural Networks,* , Washington DC, July 14-19, 2001, Page(s): 738-743 vol 1.

[12] Chao T.-H., Ho J.-M., Hsu Y.-C., “Zero skew clock net routing,” *Proceedings of the 29th ACM/IEEE conference on Design automation,* June 08-12, 1992, Anaheim, California, United States, Page(s):518-523.

[13] Jackson Michael A. B., Srinivasan Arvind, Kuh E. S., “Clock routing for high-performance ICs,” *Proceedings of the 27th ACM/IEEE conference on Design automation,* June 24-27, 1990, Orlando, Florida, United States, Page(s):573-579.

[14] Edahiro Masato, “A clustering-based optimization algorithm in zero-skew routings,” *Proceedings of the 30th international conference on Design automation*, June 14-18, 1993, Dallas, Texas, United States, Page(s):612-616.

[15] Cong J., A. B. Kahng, and G. Robins, “Matching-Based Methods for High-Performance Clock Routing,” *IEEE Transactions on Computer-Aided Design*, 12(8), August 1993, Page(s):1157-1169.

[16] Kahng A. B., Cong J. and Robins G., “High-Performance Clock Routing Based on Recursive Geometric Matching”, *Proc. ACM/IEEE Design Automation Conference*,” June 1991, Page(s):322-327.

[17] Boese K. D., Kahng A. B., “Zero-Skew Clock Routing Trees with Minimum Wire Length,” *IEEE International Conference on ASIC*, , Rochester, NY, September 1992, Page(s):1.1.1-1.1.5.

[18] Ting-Hai Chao, Yu-Chin Hsu, Jan-Ming Ho, Kneeth D. Boese, and Andrew B. Kahng, “Zero skew clock routing with minimum wirelength,” *IEEE Transactions on Circuits and Systems*, Volume 39, Issue 11, Nov. 1992, Page(s):799-814.

[19] Yow-Tyng Nieh, Shih-Hsu Huang, Sheng-Yu Hsu “Minimizing Peak Current via Opposite-Phase Clock Tree” *Design Automation Conference 05*,Page(s):182-185.

[20] Ran-Song Tsay, “Exact Zero Skew,” *IEEE Int. Conference on Computer-Aided Design (International Conference on Computer Aided Design)*, 1991. Nov. 1991, Page(s):336-339.

[21] M. Donno, A. Ivaldi, L. Benini, and E. Macii, “Clock-tree power optimization based on RTL clock-gating,” *ACM/IEEE 40th Design Automation Conference*, Pages 622-627, Anaheim, CA, June 2-6 2003.

[22] J. Oh, and M. Pedram, “Gated Clock Routing for Low-Power Microprocessor Design,” *IEEE Transactions on CAD/ICAS*, Vol. 20, No. 6, pp. 715-722, June 2001.

[23] Qi-Wang, and Roy.-S, “Power minimization by clock root gating,” *Proceedings of the ASP-DAC 2003* , Pages:249 - 254, 21-24,Jan.2003

[24] Amir H. Farrahi, Chunhong Chen, Ankur Srivastava, Gustavo Tellez, and Majid Sarrafzadeh, “Activity-Driven Clock Design,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems,* Vol. 20, No. 6, June 2001, pp. 705-714.

[25] J. Oh, and M. Pedram, “Gated clock routing minimizing the switched capacitance,” *Proc. of Design Automation and Test in Europe,* Feb. 1998, pp. 692-697.

[26] L. Benini, G. De Micheli, “Transformation and Synthesis of FSMs for Low-Power Gated-Clock Implementation,” *IEEE Transactions on CAD/ICAS,* Vol. 15, No. 6, pp. 630–643, June 1996

[27] Chunhong Chen, Changjun Kang and Majid Sarrafzadeh, “Activity-sensitive clock tree construction for low power,” *Proceedings of the 2002 international symposium on Low power electronics and design,* August 12-14, 2002, Monterey, California, USA

[28] Q. Wu, M. Pedram, and X. Wu. “Clock-gating and its application to low power design of sequential circuits,” *IEEE Custom Integrated Circuits Conference,* 1997, pp. 479-482

[29] Naveed Sherwani, *Algorithms for VLSI Physical Design Automation,* Kluwer Academic Publishers, 3rd edition, 1999.

[30] T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms,* 2nd Ed, MIT Press and McGraw-Hill Book Company, 2001.

[31] Sadiq M. Sait and H. Youssef, *VLSI Physical Design Automation: Theory and Practice,* World Scientific, 1999.

[32] Ralph H.J.M. Otten, ”Automatic floorplan design,” *Proceedings of the 19th conference on Design automation*, p.261-267, January 1982

[33] D. F. Wong, C. L. Liu, “A new algorithm for floorplan design,” *Proceedings of the 23rd ACM/IEEE conference on Design automation*, p.101-107, July 1986, Las Vegas, Nevada, United States

[34] Y.-C. Chang, Y.-W. Chang, G.-M. Wu, and S.-W. Wu, “B*-trees: A New Representation for Non-slicing Floorplans,” *Proc. of ACM/IEEE Design Automation Conference (DAC-2000)*, pp. 458-463, LA, CA, June 2000.

[35] Pei-Ning Guo, Chung-Kuan Cheng, Takeshi Yoshimura, “An O-tree representation of non-slicing floorplan and its applications,” *Proceedings of the 36th ACM/IEEE conference on Design automation conference*, p.268-273, June 21-25, 1999, New Orleans, Louisiana, United States

[36] Xianlong Hong, Gang Huang, Yici Cai, Jiangchun Gu, Sheqin Dong, Chung Kuan Cheng, Jun Gu, “Corner block list: an effective and efficient topological representation of non-slicing floorplan,” *Proceedings of the 2000 IEEE/ACM international conference on Computer-aided design*, November 05-09, 2000, San Jose, California

[37] Jianbang Lai, Ming-Shiun Lin, Ting-Chi Wang, Li-C. Wang, “Module placement with boundary constraints using the sequence-pair representation,” *Proceedings of the 2001 conference on Asia South Pacific design automation*, p.515-520, January 2001, Yokohama, Japan

[38] Jason Cong, Gabriele Nataneli, Michail Romesis, Joseph R. Shinnerl, “An area-optimality study of floorplanning,” *Proceedings of the 2004 international symposium on Physical design*, April 18-21, 2004, Phoenix, Arizona, USA

[39] Jai-Ming Lin, Yao-Wen Chang, “TCG: a transitive closure graph-based representation for non-slicing floorplans,” *Proceedings of the 38th conference on Design automation*, p.764-769, June 2001, Las Vegas, Nevada, United States

[40] Yuchun Ma, Sheqin Dong, Xianiong Hong, Yici Cai, Chung-Kuan Cheng, Jun Gu, “VLSI floorplanning with boundary constraints based on corner block list, *Proceedings of the 2001 conference on Asia South Pacific design automation*, p.509-514, January 2001, Yokohama, Japan

[41] Hiroshi Murata, Kunihiro Fujiyoshi, Shigetoshi Nakatake, Yoji Kajitani, “Rectangle-packing-based module placement,” *Proceedings of the 1995 IEEE/ACM international conference on Computer-aided design*, p.472-479, November 05-09, 1995, San Jose, California, United States

[42] H. Murata, K. Fujiyoshi, M. Kaneko, “VLSI/PCB placement with obstacles based on sequence-pair,” *Proceedings of the 1997 international symposium on Physical design*, April 14-16, 1997, Napa Valley, California, United States, Page(s):26-31.

[43] Shigetoshi Nakatake, Kunihiro Fujiyoshi, Hiroshi Murata, Yoji Kajitani, “Module placement on BSG-structure and IC layout applications,” *Proceedings of the 1996 IEEE/ACM international conference on Computer-aided design*, November 10-14, 1996, San Jose, California, United States, Page(s):484-491.

[44] Xiaoping Tang, D. F. Wong, “FAST-SP: a fast algorithm for block placement based on sequence pair,” *Proceedings of the 2001 conference on Asia South Pacific design automation,* , January 2001, Yokohama, Japan, Page(s):521-526.

[45] D. Duarte, V. Narayanan, M. J. Irwin, “A Clock Power Model to Evaluate Impact of Architectural and Technology Optimizations,” *IEEE Transactions on VLSI Systems*, Vol. 10, No. 6, December 2002, Page(s):844–855.

[46] Joe G. Xi, Wayne W. M. Dai, “Buffer insertion and sizing under process variations for low power clock distribution,” *Proceedings of the 32nd ACM/IEEE conference on Design automation,* , June 12-16, 1995, San Francisco, California, United States, Page(s):491-496.

[47] A. Vittal, M. Marek-Sadowska, “Low-Power Buffered Clock Tree Design,” *IEEE Transactions on CAD/ICAS,* Vol. 16, No. 9, September 1997, Page(s):965-975.

[48] Jatuchai Pangjun, Sachin S. Sapatnekar, “Clock distribution using multiple voltages,” *Proceedings of the 1999 international symposium on Low power electronics and design,* , August 16-17, 1999, San Diego, California, United States ,Page(s):145-150.

[49] W. C. Elmore, “The transient response of damped linear networks with particular regard to wide band amplifiers,” *Journal of Applied Physics,* vol. 19, January 1948, Page(s):55-63.

[50] J.P.Fishburn,“Clock Skew Optimization” *IEEE trans. on computers* Vol. 39,k NO.7 , pp.945 - 951 ,1990.

[51] R.Ader,E.G,A.Litman and I.S. kourtev,“Scale Clock Skew scheduling Technique for Improved Reliability of Digital Synchronous VLSI Circuits” *ISCAS02 circuits and System* , Vol.1,pp.I-357-360 .

[52] E.Malarasi,S.zanella,MinCao,J.Uscherson,“Impact Analysis of Process Variabity on Clock Skew” , *Quality Electronic Design,*2002 ,Page(s):129-132.

[53] Yongseok Cheon, Pei-Hsin Ho, Andrew B. Kahng, Sherief Reda, Qinke Wang “Power Aware Placement,”*Proceedings of the 42nd annual conference on Design automation 2005,*, Page(s): 176-181.

[54] Rishi Chaturvedi, Hu, J. “A simple yet effective merging scheme for prescribed-skew clock routing,”*Computer Design, 2003. Proceedings. 21st International Conference on 13-15 Oct. 2003*, Page(s):282-287.

[55] Dennis J.-H. Huang and Andrew B. Kahng, “Partitioning-Based Standard-Cell Global Placement with An exact Objective”*ISPD‘97* , 1997, Page(s):18-25.

[56] Xiaojian Yang, Maogang Wang, Kenneth Eguro, and Majid Sarrafzadeh ,“A Snap-On Placement Tool”*Proceedings of ISPD‘00*, 2000, Page(s):153-158.

[57] Maogang Wang, Xiaojian Yang, Kenneth Eguro, and Majid Sarrafzadeh ,“Dragon2000 : Fast Stand-Cell Placement for Large Circuits,”*Proceedings of International Conference on Computer Aided Design*,2000, Page(s):260-283.

[58] http://www.visc.vt.edu/ mhsiao/iscas89.html.

[59] http://www.ece.wisc.edu/ vlsi/tools/iscas-placement/index.html.

[60] J. H. Huang,A. B. Kahng and C.-W. A. Tsao, “On the Bounded-SkewRouting Tree Problem”, *Proc. ACM/IEEE Design Automation Conf., Jun. 1995,* Page(s):508-513.

[61] J. Cong and C.-K. Koh, Minimum-Cost Bounded-Skew Clock Routing, *Proc.IEEE Intl. Symp. on Circuits and Systems, Vol 1, Apr. 1995*, Page(s):215-18.

[62] Jason Cong, Andrew B. Kahng, Cheng-Kok Koh and C.-W. Albert Tsao “Bounded-Skew Clock and Steiner Routing Under Elmore Delay ” *International Conference on Computer Aided Design 5-9 Nov 1995*, Page(s):66-71.

[63] Roy,J.A., Asya,S.N.,Papa,D.A. Min-cut floorplacement“ *Computer-Aided Design of integrated Circuits and Systems,IEEE Transactions* VOL25,Issue 7,July,2006, Page(s):1313-1326.

[64] Yongqiang Lu, C. N. Sze, Xianlong Hong, Qiang Zhou, Yici Cai, Liang Huang, Jiang Hu, "Navigating registers in placement for clock network minimization ," *Proceedings of the conference on Design automation 2005,* Page(S):176-181.

[65] Chih-Hung Lee, Chin-Hung Su, Shih-Hsu Huang, Chih-Yuan Lin, Tsai-Ming Hsieh, "Floorplanning with clock tree estimation,"*ISCAS 2005* , Page(s):6244-6247 vol. 6.

[66] S.H. Huang and Y.T. Nieh, "Clock Period Minimization of Non-Zero Clock Skew Circuits", *in the Proc. of IEEE/ACM International Conference on Computer Aided Design* ,, 2003, Page(s):809-812.

[67] Guenter Stenz, Bernhard M. Riess, Bernhard Rohfleisch, Frank M. Johannes "Timing driven Placement in Interaction with Netlist Transformations"*International Symposium on Physical Design archive Proceedings of the 1997 international symposium on Physical design*, Page(s):36-41.

[68] Liang Huang, Yici Cai, Qiang Zhou, Xianlong Hong, Jiang Hu, Yongqiang Lu "Clock network minimization methodology based on incremental placement" *Design Automation Conference Asia and South Pacific archive Proceedings of the 2005 conference on Asia South Pacific design automation*, Page(s):99-102.

[69] Yongqiang Lu, C. N. Sze, Xianlong Hong, Qiang Zhou, Yici Cai, Liang Huang, Jiang Hu, "Register placement for low power clock network" *Design Automation Conference Asia and South Pacific archive Proceedings of the conference on Asia South Pacific design automation,*, Page(s):588-593.

[70] William H. Press, Sual A. Teukolsky, William T. Vetterling, Brain P. Flannery "Numerical Recipes in C++" second edition.

[71] R. B. Deokar and S. S. Sapatnekar, "A graph-theoretic approach to clock skew optimization", in *Proc. IEEE International Symposium on Circuits and Systems*, 1994, Page(s):1.407-1.410.

[72] N. Venkateswaran and D. Bhatia, “Clock-skew constrained placement for row based designs,” *in Proc. IEEE International Conference of Computer Design* , 1998, Page(s):219-220.

[73] George Karypis and Vipin Kumar “A Hypergraph Partitioning Package Version 1.5.3” 1998.

[74] Lin Yuan and Gang Qu ,“Enhanced Leakage Reduction Technique by Gate Replacement” *Design Automation Conference 2005*, Page(s):47-50.