Low Power Resonant Rotary Global Clock Distribution Network Design

A Thesis

Submitted to the Faculty

of

Drexel University

by

Ying Teng

in partial fulfillment of the

requirements for the degree

of

Doctor of Philosophy in Electrical and Computer Engineering

May 2014

Dedications

This thesis is dedicated to my husband Tianyi Xu and our parents, for their endless love and support.

Acknowledgments

I would like to express my sincere gratitude to all the people who helped me during my Ph.D. study at Drexel University.

First of all, I would like to thank my advisor, Dr. Baris Taskin, who led me into the world of research and also supported me to propose many ingenious results. I am always encouraged by his hard-working spirit and research enthusiasm during my whole Ph.D. study. Without his help and guidance, I could not have a productive graduate life.

I would like to thank Dr. Nagarajan Kandasamy, Dr. Prawat Nagvajara, Dr. Mark Hempstead and Dr. Antonios Kontsos, for serving on my defense committee. I have learned a lot from their valuable suggestions and encouragement during my research work.

I would also like to thank the alumni and students in our team: Dr. Vinayak Honkote, Dr. Jianchao Lu, Dr. Ankit More, Sharat C. Shekar, Julian Kemmerer, Can Sitik, Leo Filippini, Scott Lerner, Vasil Pano and Karthik Sangaiah for their help and all the fun we have had during the Ph.D. study.

Lastly and most importantly, I would like to express my greatest appreciation to my husband Tianyi Xu for his care and infinite love, and our parents for their continuous support.

Table of Contents

List of Tables

List of Figures

Abstract

Low Power Resonant Rotary Global Clock Distribution Network Design

Ying Teng

Advisor: Baris Taskin, Ph.D.

Along with the increasing complexity of the modern very large scale integrated (VLSI) circuit design, the power consumption of the clock distribution network in digital integrated circuits is continuously increasing. In terms of power and clock skew, the resonant clock distribution network has been studied as a promising alternative to the conventional clock distribution network. Resonant clock distribution network, which works based on adiabatic switching principles, provides a complete solution for on-chip clock generation and distribution for low-power and low-skew clock network designs for high-performance synchronous VLSI circuits.

This dissertation work aims to develop the global clock distribution network for one kind of resonant clocking technologies: The resonant rotary clocking technology. The following critical aspects are addressed in this work: (1) A novel rotary oscillator array (ROA) topology is proposed to solve the signal rotation direction uniformity problem, in order to support the design of resonant rotary clocking based low-skew clock distribution network; (2) A synchronization scheme is proposed to solve the large scale rotary clocking generation circuit synchronization problem; (3) A low-skew rotary clock distribution network design methodology is proposed with frequency, power and skew optimizations; (4) A resonant rotary clocking based physical design flow is proposed, which can be integrated in the current mainstream IC design flow; (5) A dynamic rotary frequency divider is proposed for dynamic frequency scaling applications. Experimental and theoretical results show: (1) The efficiency of the proposed methodology in the construction of low-skew, low-power resonant rotary

clock distribution network. (2) The effectiveness of the dynamic rotary frequency divider in extending the operating frequency range of the low-power resonant rotary based applications.

1. Introduction

Along with the increasing complexity of the modern CMOS integrated circuit (IC) designs, the power consumption of the clock distribution network in ICs has become a major contributor to the chip power consumption. In some applications, the clock distribution network can contribute to 25% [23] to 50% [53] of the total chip power consumption. Fig. 1.1 shows a power consumption breakdown of the Intel Itanium II McKinley microprocessor, in which the clock distribution network contributes to 33% of the total power consumption [49]. The power consumption of the clock distribution network consists of leakage power, short circuit power and dynamic power, in which the major part is dynamic power. The dynamic power P_{dyn} is proportional to:

$$P_{dyn} \approx C_L V_{DD}^2 f \tag{1.1}$$

where, C_L, V_{DD} and f are capacitive loading, supply voltage and clock frequency, respectively. Reducing the dynamic power consumption by lowering the clock frequency is conflicting with developing high speed VLSI systems. Thus, in conventional clock distribution network designs, low-power designs are achieved by either reducing the power supply [44] or the capacitive loading [50].

Alternatively, an efficient approach to reducing the power consumption is to use resonant clocking [21, 51, 76]. The adiabatic switching [12] property of resonant clocking provides an appealing solution to the low-power clock distribution network design by effectively recycling the dynamic power. As a result, resonant clocking technology is considered as a promising alternative to the traditional clock distribution network. Besides the advantage of low-power, resonant clocking technology is superior in generating low skew, low jitter, high frequency oscillation signals [77].

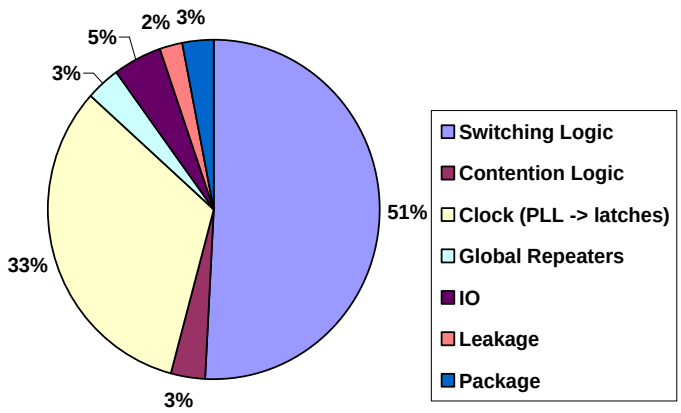


Figure 1.1: The power breakdown of the Intel microprocessor McKinley [49].

The resonant rotary oscillator, as one of the favorable resonant clocking technologies, is outstanding in providing high frequency, full amplitude, square-wave oscillation signals [76], which can be easily applied as the clock sources in the clock distribution network. Since the oscillation signals with different phases can be obtained from the resonant rotary oscillator, previous studies mainly focused on utilizing the multi-phase oscillation signals in non-zero skew circuit designs [27, 33, 80]. However, the utilization of rotary clocking in zero-skew circuit designs has been challenging. Therefore, resonant rotary clocking based low-skew clock distribution network is investigated in this dissertation. More importantly, a complete design methodology with power and clock skew optimizations is proposed.

The power consumption of the resonant oscillators is inversely proportional to the operating frequency of the resonant system, which is a common problem for all types of resonant clocking technologies. As a result, the resonant oscillators lose their advantages in low frequency ranges due to their increasing power consumption. In the study of [38], a resonant/non-resonant clock distribution network switch is proposed, with which the clock distribution network works in hybrid mode for different

frequency ranges in order to maximally reduce the power consumption. In this dissertation, a resonant rotary clocking based dynamic frequency divider is proposed, in which the frequency of the source oscillation signal is divided in order to generate a lower frequency oscillation signal. Thus, the operating frequency range of the resonant rotary clocking is greatly extended while still maintaining low power dissipation. Furthermore, the proposed dynamic frequency divider can dynamically tune its division number in order to be fitted into dynamic frequency scaling applications.

1.1 Problem Statement

This dissertation introduces the automated development of a global resonant rotary clock network, and the support for dynamic frequency scaling of the rotary clock technology. The major problems in global resonant clock network design and rotary clocking dynamic frequency scaling are described as follows.

1.1.1 Problems in Resonant Rotary Clock Distribution Network Development

There are three major problems in the automated development of global resonant rotary clock distribution network designs.

The first problem is the operating frequency inaccuracy. The operating frequency of resonant rotary clocking is determined by the total capacitance and the total inductance of the resonant system, which are contributed to by the resonant rotary oscillator and the sub-network trees. The sub-network trees are composed of the synchronous components (i.e. registers) and the interconnect wires delivering the clock signals. Thus, the construction of the resonant oscillator, which determines the oscillation frequency, greatly depends on the capacitance of the sub-network trees. A significant frequency mismatch exists if the resonant oscillator is constructed without

considering any loading capacitance. However, in a conventional top-down clock network design flow, capacitance information of the sub-network trees remains unknown until the clock network is constructed. Thus, the frequency inaccuracy of the resonant rotary clock distribution network still remains a problem.

The second problem is the resonant rotary clocking synchronization issue. Resonant rotary clocking is well known for generating multi-phase oscillation signals. Zero skew implementations are noted in literature as being accomplished through tapping off an oscillation signal with a desired phase. However, with an unknown signal rotation direction within the oscillation system, it is not guaranteed to tap off an oscillation signal with a desired phase.

The third problem is a lack of EDA design tools supporting the physical designs of resonant clocking, particularly for global-level clock distribution. Resonant rotary clocking possesses an on-chip clock generation and distribution feature, which is not supported in current mainstream CAD tools in the IC physical design flow.

1.1.2 Problems in Rotary Dynamic Frequency Scaling

An existing problem of resonant clocking technologies is that the power consumption is inversely proportional to the operating frequency of the resonant system. The power consumption problem limits the resonant rotary clocking to be applied on lower frequency circuit designs. A static frequency division is known for rotary clocking but support for Dynamic Frequency Scaling is lacking.

1.2 Contribution of This Work

The major contributions of this work are as follows: (1) The circuit topologies and novel design methodologies for the resonant rotary clock distribution network design and (2) the circuit topology for rotary dynamic frequency scaling. The pro-

posed improvements are significant on the topics of low-power, low-skew resonant rotary clock distribution network design and wide frequency range resonant dynamic frequency scaling. In order to address the issues listed in Section 1.1.1 and 1.1.2, the following studies provide solutions to the problems within the circuit topology and design methodology.

1. A novel resonant rotary circuit topology, the ROA-Brick, is proposed for zero-skew resonant rotary clock distribution network design [62]. Analysis and mathematical proofs are provided to demonstrate the uniqueness of the ROA-Brick in generating ideally zero-skew clock signals.

2. A fast synchronization scheme is proposed for large scale resonant rotary clock generation circuits [65] in order to help the circuit reach a steady state quickly after the circuit powers up.

3. A design methodology is proposed using the ROA-Brick as the basic building block in resonant rotary clock generation and distribution network for larger scale synchronous circuit designs [64].

4. A topology and placement optimization methodology for resonant rotary global network is proposed in order to make the global network better suited for non-even distribution of the synchronous components in chip designs [68].

5. A frequency-centric resonant rotary design methodology is proposed for high frequency accuracy resonant rotary clock distribution network design [63].

6. A complete physical design flow is proposed for resonant rotary clocking based circuits, which is developed with the current mainstream physical design tools, based on the conventional physical design flow.

7. A rotary dynamic frequency divider is proposed for resonant dynamic frequency scaling. The rotary dynamic frequency divider not only provides a solution for applying resonant rotary clocking at lower frequencies while still maintaining low power dissipation, but also realizes the dynamically automatic frequency adjustments [66, 67].

1.3 Organization of the Dissertation

This dissertation is organized as follows. In Chapter 2, different categories of the resonant clocking technologies and the corresponding previous works are reviewed. In Chapter 3, a novel circuit topology of the resonant rotary clocking is proposed, which is called rotary oscillator array brick (ROA-Brick). The ROA-Brick solves the critical signal rotation direction uniformity problem of the rotary clocking, so that the ROA-Brick can be served in the zero-skew clock signal generation circuits. In Chapter 4, a resonant rotary clock distribution network design methodology is proposed, in which the ROA-Brick is used as the basic component for the resonant rotary global network construction. Meanwhile, a synchronization scheme is proposed in order to help the resonant rotary clock distribution network reach frequency and signal rotation direction synchronization efficiently. In Chapter 5, a resonant rotary global distribution network topology and placement optimization methodology is proposed, in which the power and skew of the resonant rotary clock distribution network are optimized. In Chapter 6, a frequency-centric rotary clock distribution network design methodology is proposed. In the proposed methodology, the regional and local clock network capacitances are considered in the development of the resonant clocking generation circuits in order to improve the frequency accuracy. In Chapter 7, a complete physical design flow is proposed for resonant rotary clocking based circuits, which is developed with a mainstream electronic design automation (EDA) tool. In order

to utilize the resonant rotary clocking in lower frequencies, in Chapter 8, a resonant rotary clocking based dynamic frequency divider is proposed, the division number of which can be dynamically tuned for different workloads, which fits well in the modern dynamic frequency scaling designs.

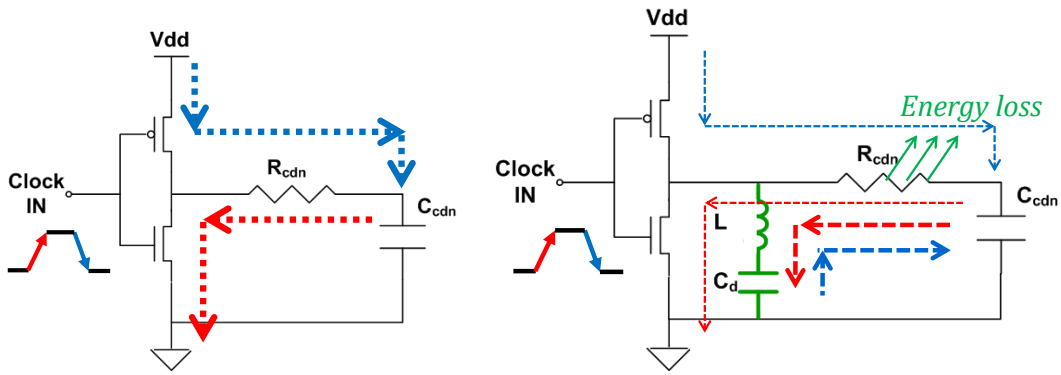
2. Overview of the Modern Resonant Technologies

Resonant clocking technologies provide a solution for low-power, high-frequency clock generation and distribution network designs [2, 12, 16, 21, 76], which is studied as a promising alternative to the conventional clock distribution network. A comparison of the energy dissipation pattern between the non-resonant and resonant clock distribution networks is shown in Fig. 2.1 [24]. In conventional clock distribution network, as shown in Fig. 2.1(a), the dynamic power eventually dissipates as heat. On the other hand, in a resonant system, the dynamic power is recycled in the resonant system.

Modern resonant clocking technologies are mainly categorized as follows:

1. Distributed LC-tank oscillator [6, 7, 8, 9, 10, 20, 21, 24, 38, 55]
2. Grid LC-tank oscillator [36, 37, 57]
3. Standing wave oscillator [2, 16, 29, 39, 43, 45, 46, 47, 51, 52]
4. Rotary traveling wave oscillator [27, 31, 32, 33, 34, 42, 59, 60, 72, 76, 77, 78, 79, 80, 81]

In order to facilitate modern VLSI designs, the resonant clock distribution networks developed based on these four kinds of resonant clocking technologies are usually constructed in three levels: The global, regional and local clock distribution networks. In the following sections, the resonant clock distribution network circuit topologies of these four kinds of resonant clocking technologies are introduced in detail.



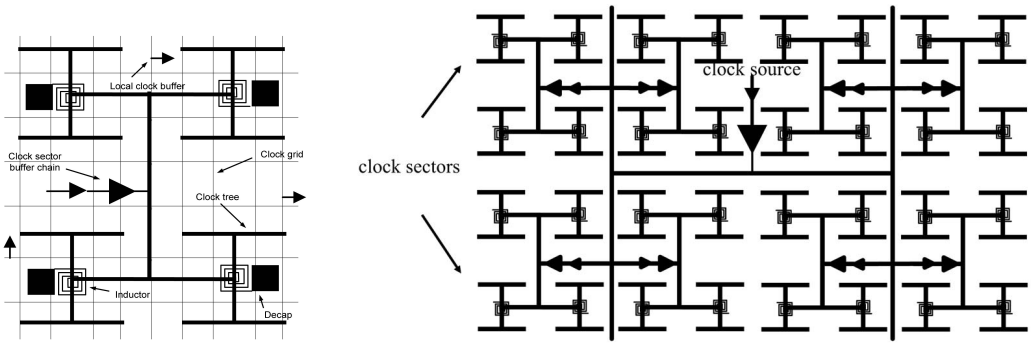
(a) Energy dissipation pattern in non-resonant clock distribution network.

(b) Energy dissipation pattern in resonant clock distribution network.

Figure 2.1: Energy dissipation pattern comparison between non-resonant and resonant clock distribution networks [24].

2.1 Distributed LC-tank Oscillator Based Clock Distribution Network

The distributed LC-tank oscillator is developed from the conventional H-tree clock distribution network for power saving purposes. Fig. 2.2(a) and Fig. 2.2(b) show the regional and global networks of the distributed LC-tank clock distribution network, respectively. The global network is symmetrical throughout the chip area. The regional network, which is named as clock sector in [10], contains the resonant components. In the design of [10], an external clock source is needed to distribute the clock signal throughout the chip. In each clock sector, each of the four spiral inductors has one end connected to the clock tree and the other end attached to a large decoupling capacitance. The decoupling capacitances are used to set a mid-rail DC voltage for the oscillation signals. The intrinsic oscillation frequency of the regional network, which is set by the parasitic capacitance of the clock tree and the the spiral inductors, are designed to resonant with the external clock signal. Therefore, a main feature of the distributed LC-tank clock distribution network is that ideally zero-skew and same amplitude clock signals can be obtained at the end of each clock sector.



(a) Regional network. (b) Global network.

Figure 2.2: Distributed LC-tank clock distribution network [10].

Due to the similarity of the topology of the distributed LC-tank clock distribution network to the traditional clock distribution network, the design automation flow of the distributed LC-tank clock distribution network can be highly integrated into the conventional physical design flow. In the studies of [38, 57], a clock distribution network is designed to operate in a hybrid mode: (1) A resonant-clock mode for energy-efficient operation in higher operating frequency range; (2) A conventional direct-drive mode to support lower frequency operations. As shown in Fig. 2.3, the resonant/conventional operating mode can be switched for different operating frequencies for power saving purposes while still maintaining high performance.

2.2 Grid LC-tank Oscillator Based Clock Distribution Network

The grid LC-tank clock distribution network is developed as a combination of the distributed LC-tank clock distribution network and the non-resonant clock mesh. In most of the distributed LC-tank clock distribution network designs, the spiral inductors are evenly distributed throughout the chip area, the total size of which is mainly determined by the operating frequency and chip area. In order to optimize the placement and the size of the spiral inductors, in the work of [37], a grid LC-tank

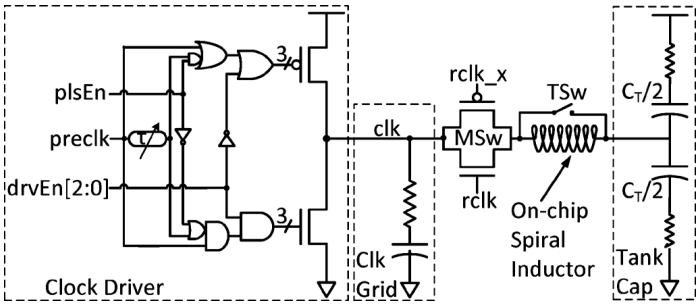
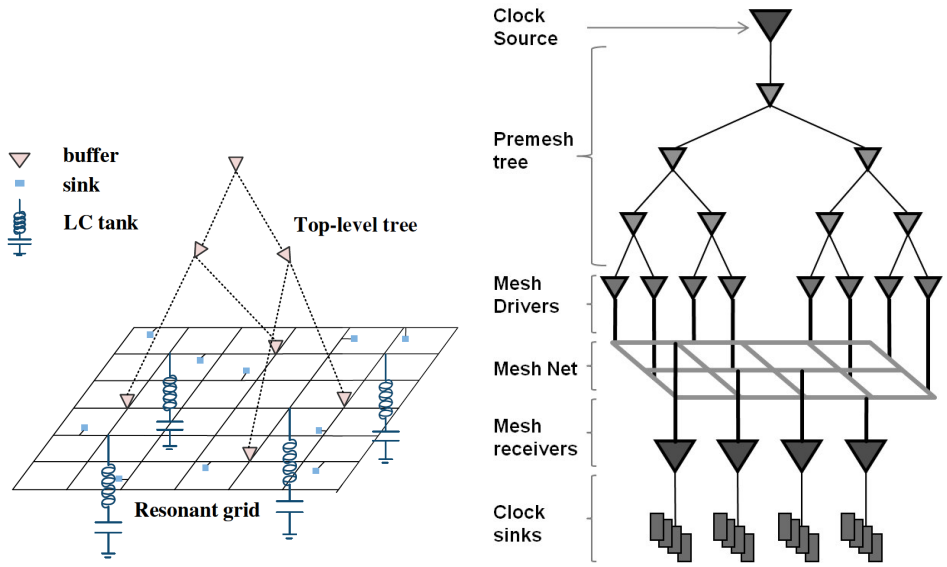


Figure 2.3: Simplified schematic representation of dual-mode clock implementation in [57].

clock distribution network is developed by applying a clock mesh into the resonant system. Similar to the conventional mesh-based clock distribution network topology, the grid LC-tank clock distribution network can be divided into global, regional and local levels. In a grid LC-tank clock distribution network, the global level clock tree is used to drive the regional level clock mesh. The local level consists of the followings: (1) Stubs directly connecting to synchronous components or unbuffered subnetwork trees and (2) distributed LC-tanks. The LC-tanks are driven by the clock mesh and are designed to resonant with an injected clock signal. Fig. 2.4(a) and Fig. 2.4(b) show the grid LC-tank clock distribution network structure [37] and the traditional non-resonant clock distribution network [5], respectively.

The traditional mesh-based clock distribution network has received significant attention in high-performance ASIC designs in the last decade [18, 25, 54, 70] due to its robustness to process and environmental variations in modern IC designs. The grid LC-tank clock distribution network inherits the robustness property of the traditional mesh-based clock distribution network by possessing a mesh grid as part of its resonant clock distribution network. Furthermore, the resonant property of the grid LC-tank clock distribution network benefits for dynamic power savings. The grid LC-tank synthesis methodology is also similar to the design methodology of the



(a) Grid LC-tank clock distribution network [37].

(b) Traditional mesh-based clock distribution network [5].

Figure 2.4: Comparison of the grids LC-tank clock distribution network and the traditional mesh-based clock distribution network.

mesh-based clock distribution network [54, 70]: (1) Create an initial clock mesh and insert buffers; (2) Place and size the LC-tanks attached to the clock mesh for a target frequency; (3) Resize the clock mesh buffers after the placement/sizing of the LC-tanks.

The most critical step of the grid LC-tank clock distribution network synthesis methodology is the placement and sizing of the LC-tanks for a target frequency. If inductors are added to every node of the grid, the grid exhibits ideal performance. However, the size of the inductors is impractically large. The inductor sizing methodology in [37] provides a better solution for saving power and area. However, the size of the optimized inductors falls into a continuous range, which makes it infeasible to implement. Based on these considerations, the work in [36] presents a library-aware resonant clock synthesis (LARCS) methodology, which optimizes the size of the inductors using a pre-made library of inductors. To this end, the grid LC-tank

clock distribution network is synthesized with a discrete inductance oscillator with a maximum total inductor area as a constraint. By applying the methodology in [36], the average area engagement of the inductors on eight ISPD10 benchmarks decreases from 69% to 66%.

2.3 Standing Wave Oscillator Based Clock Distribution Network

Along with increasing the operating frequency, the timing uncertainty introduced by the clock network, the clock skew and the clock jitter, becomes huge obstacles in modern IC designs. This is because in conventional buffered H-trees, the clock skew and the clock jitter are proportional to clock latencies, which are hard to be reduced along with the decrease of the clock cycle [55]. A resonant clock distribution network that uses standing wave oscillators has the potential to significantly reduce both clock skew and clock jitter. This is because standing waves have the unique property that phase does not depend on positions, which indicates that there is ideally no skew. The standing wave oscillator has been used on board level clock distribution network designs [12] and on chip level design in [52].

A standing wave formed when two identical waves that are propagating in opposite directions interact with each other. The property of the standing wave oscillator is expressed in Equation 2.1:

$$\begin{aligned} & A_1 cos(\omega t - \beta z) + A_2 cos(\omega t + \beta z + \phi) \\ & = \underbrace{2A_2 cos(\omega t + \frac{\phi}{2}) cos(\beta z + \frac{\phi}{2})}_{\text{Standing wave}} + \underbrace{(A_1 - A_2) cos(\omega t - \beta z)}_{\text{Traveling wave}}. \end{aligned} \tag{2.1}$$

The first part of Equation 2.1 describes the standing wave term and the second part describes the traveling wave term. The traveling wave term diminishes to zero when the amplitudes of the two waves are identical. As indicated by Equation 2.1,

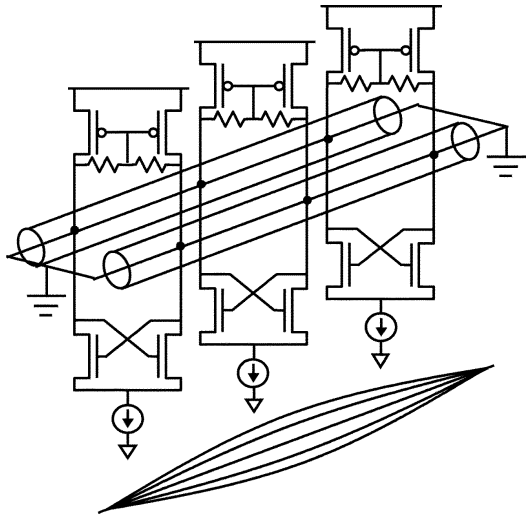
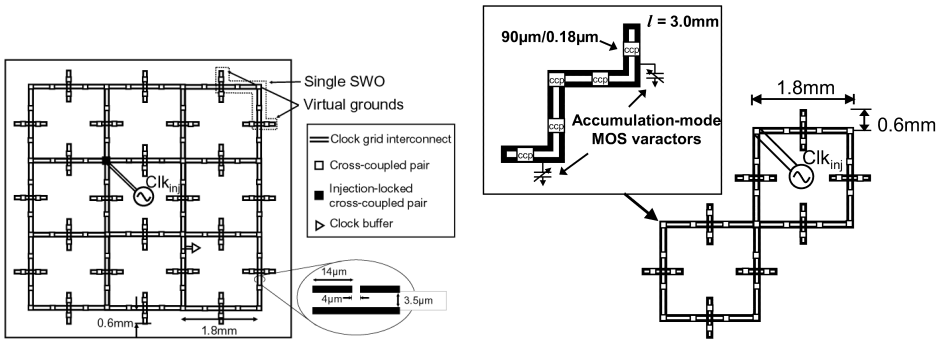


Figure 2.5: Standing wave oscillator with three cross-coupled pairs in [52].

the phase of the traveling wave signal varies linearly with positions and the phase of the standing wave signal is the same regardless of the position but the amplitude varies.

The standing wave oscillator can be generated by sending an incident wave down a transmission line and reflect it back with a short circuit. Fig. 2.5 shows the topology of the standing wave oscillators in [52], which sustains ideal standing waves on lossy transmission lines. The NMOS cross-coupled pairs provide enough gain in order to compensate the power loss on the transmission lines. The PMOS diode-connected loads are used to set a mid-rail voltage, around which the oscillation signals oscillate.

The standing wave oscillator topology is used to construct the standing wave clock distribution network. Fig. 2.6(a) and Fig. 2.6(b) show the grid structure of the standing wave clock distribution network and the topology of a single standing wave oscillator used for the grid construction, respectively. As shown in Fig. 2.6(a), single standing wave oscillator can be coupled together by simply connecting their



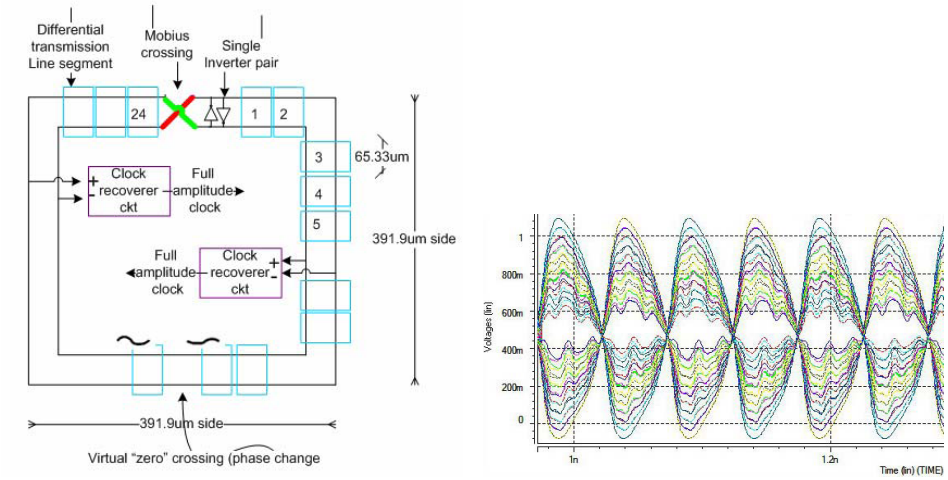
(a) Standing wave clock grid.

(b) Single standing wave oscillator for the grid construction

Figure 2.6: The topology of the standing wave clock distribution network in [52].

transmission lines to the grid structure. These standing wave oscillators can also be injection locked to a reference signal. Injection locking allows the clock frequency to be directed by an external clock signal such as a phase-locked loop (PLL) or a delay-locked loop (DLL). In the work of [52], the external reference clock is AC coupled into the gates of the PMOS/NMOS at the center of a standing wave oscillator.

Another popular standing wave oscillator topology is introduced in [16]. This standing wave oscillator topology is developed from a rotary traveling wave oscillator (RTWO) topology introduced in [76]. As shown in Fig. 2.7(a), two differential transmission lines are used to form a Möbuis connection. A single inverter pair is placed between the transmission lines. The oscillation signals on the transmission lines where the inverter pairs are placed possess the highest amplitudes and the traveling waves are propagated in opposite directions with equal power. Since there is a Möbuis connection formed by the transmission lines, the traveling waves meet with each other after traveling half of the ring perimeter with equal power and differential phases. Thus, these two traveling waves are canceled with each other as if there is a "short" on the transmission lines and the "short" point is called a virtual "zero" as shown on Fig. 2.7(a).



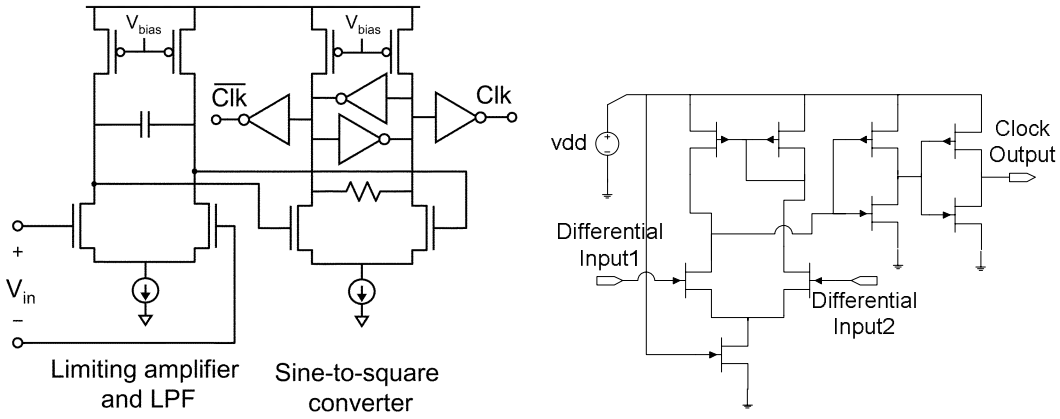
(a) Standing wave oscillator topology. (b) Simulation result of the standing wave oscillator.

Figure 2.7: The standing wave oscillator in [16].

The drawback of the standing wave oscillator is that the amplitude of the standing wave oscillation signals is position-dependent. Therefore, both of the standing wave oscillator introduced in [52] and [16] need recovery circuits to convert the sinusoid signals to digital levels in order to be used as the source of the clock distribution network. Fig. 2.8(a) shows the recovery circuit in [52], which can be divided into two stages. The first stage is composed of an amplifier and a lowpass filter which limits the amplitude-dependent skew of the output signals. The second stage is a sine wave to square wave converter. The recovery circuit in [16] is simpler and it is composed of a differential amplifier and two stages of drivers.

2.4 Rotary Traveling Wave Oscillator Based Clock Distribution Network

Rotary traveling wave oscillator (RTWO) was first introduced in [76], which provides low-skew, low-jitter, GHz range clocking with low power consumption [76] compared to its competitors introduced in Section 2.1, 2.2 and 2.3.



(a) Recover circuit in [52]. (b) Recover circuit in [16]

Figure 2.8: Recover circuits for standing wave oscillators.

2.4.1 Physical Structure of the RTWO Ring

Fig. 2.9 shows the circuit topology of an RTWO ring. The ring is composed of two differential transmission lines forming a Möbius connection. A number of anti-parallel inverter pairs are connected between the two differential transmission lines and are evenly distributed in the ring structure. The anti-parallel inverter pairs are used to overcome power losses (quasi-adiabatically) as well as providing a rotation lock. The traveling signals possess the properties as follows: (1) The oscillation signals are full-amplitude, square-wave signals since the anti-parallel inverter pairs overcome power losses during wave propagation and (2) the propagating wave inverts during consecutive rotations. Thus, an arbitrary point on the ring can provide a clock signal with a 50% duty cycle.

In order to better explain the concept of rotation lock, Fig. 2.10 shows an expanded view of an RTWO ring section, in which three anti-parallel inverter pairs are connected between the two transmission lines. When the ring starts to oscillate with a rotating wave sweeping from left to right, each anti-parallel inverter pair works independently as a single latch element. Along with the signal sweeping, the latch switches one after

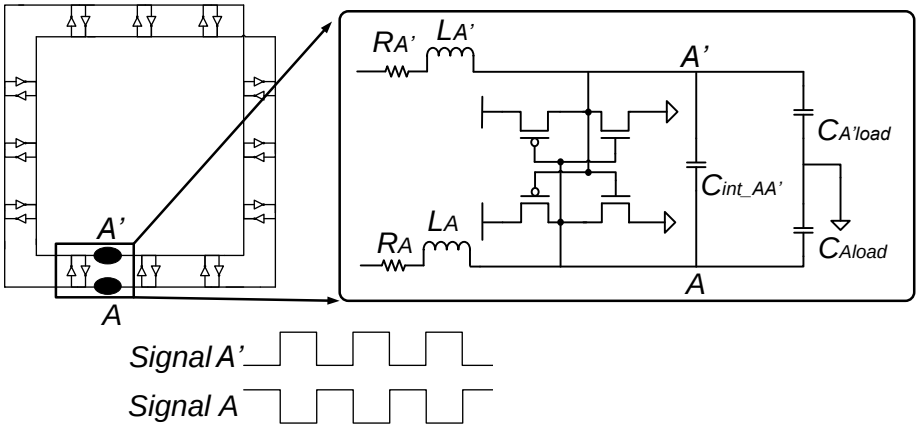


Figure 2.9: The structure of an RTWO ring [76].

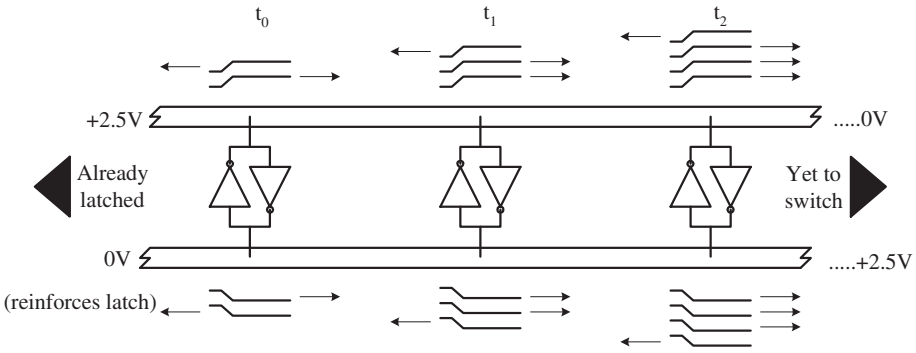


Figure 2.10: Expanded view of an RTWO ring section with transmission line sections with three cross-coupled inverter pairs [76].

another to override its own previous state. This "clash" of state changing occurs only at the rotating wavefront of the oscillation signal. Once switched, the current signal rotation direction is reinforced.

2.4.2 Signal Rotation Direction and Multi-Phase Property of the RTWO Ring

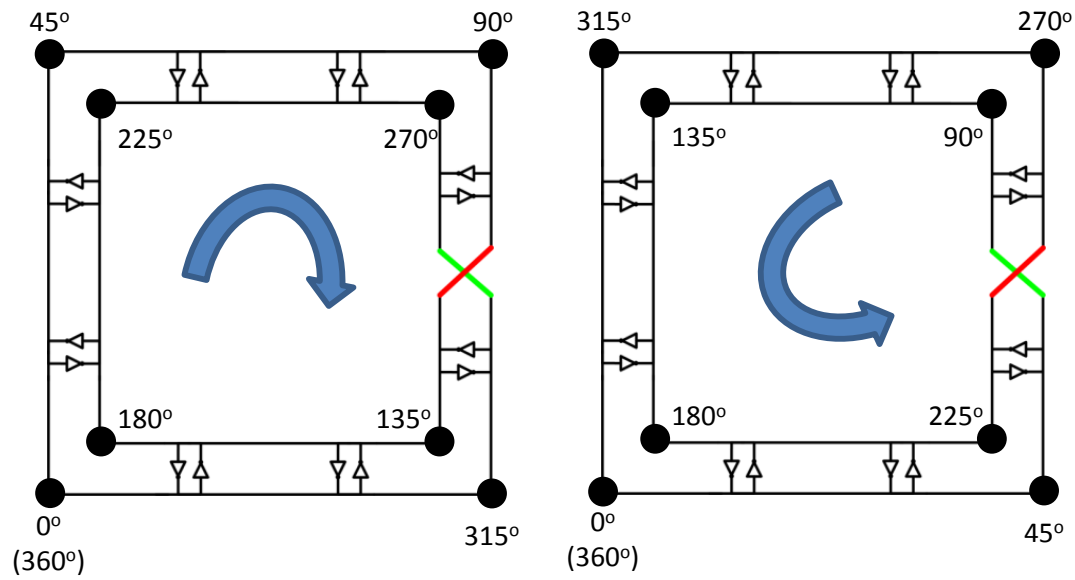
When the ring is stimulated by an external noise event or a manually performed excitement, an oscillation is formed and propagates along the transmission line. An unbiased, startup can trigger a signal propagation in either rotation direction. Early studies have detected this problem [60, 76, 79, 82, 83] and proposed that a single ring will oscillate in the direction with less capacitive loading. In order to obtain the desired signal rotation direction, varies rotation biasing mechanisms are applied, such as the directional coupler technology [76] and gate displacement [74]. Fig. 2.11 presents the different phases of the clock signals available for a ring to propagate in clockwise direction and counter clockwise direction, respectively. It is also observed that with this implementation, two corresponding points on the differential lines provide clock signals with a π phase difference.

For the signal phase information, as shown in Fig. 2.11, a reference point is selected on the ring, which is defined with clock signal delay $t = 0$ and phase $\theta = 0$. The clock signal travels along the ring and reaches back to the reference point with phase $\theta = 2\pi$. Ideally, the phase delay is evenly distributed along the ring transmission lines in the wave propagating direction. Thus, the relationship between the time delay and the phase delay can be expressed as:

$$\frac{\theta}{2\pi} = \frac{t}{T} \tag{2.2}$$

where T is the cycle time of the oscillation signal.

Since the generation of a multi-phase oscillation signal is highly practical from the ring, previous studies [27, 30, 80] have focused on utilizing the multi-phase property of the ring into non-zero skew clock distribution network design. In the study of [30],



(a) Signal phases under clock wise signal rotation direction.

(b) Signal phases under counter clock wise signal rotation direction.

Figure 2.11: Multiple phases of the oscillation signals on an RTWO ring under different signal rotation directions.

oscillation signals with optimized phases are tapped off from the transmission lines of a ring, which are used to drive the synchronous components. Consider the non-zero skew design shown in Fig. 2.12, each register R_i on the circuit has a phase requirement θ_i. For instance, consider the registers A and B, which have phase requirements of 65° and 225°, respectively. In order to satisfy the timing requirements, the tapping points marked as 45° and 135° are selected to be the tapping points for register A and register B, respectively. This is because the tapping wire (the dotted line) connecting the register to its tapping point also contributes towards the phase delay.

2.4.3 Frequency and Power Approximation of the RTWO Ring

Similar to the LC-tank oscillator, the frequency of the clock signal generated by the ring depends on the total capacitance and inductance in the system, which are

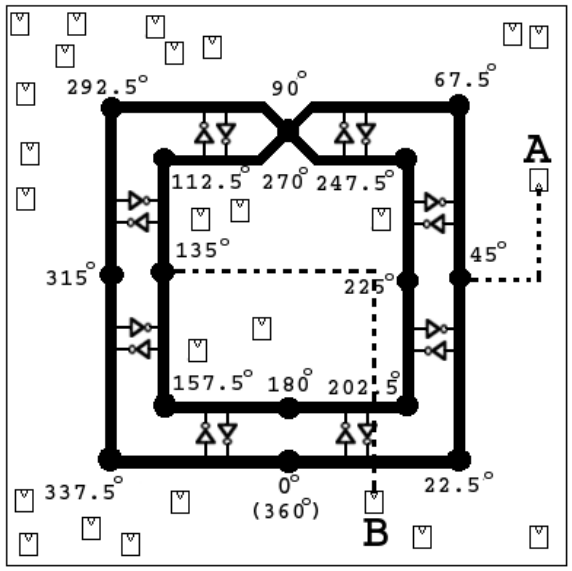


Figure 2.12: RTWO ring used for non-zero skew clock distribution network design [30].

defined by the physical implementation of rotary resonant system. The frequency of the ring is approximated as:

$$f_{osc} \approx \frac{1}{2\sqrt{C_T L_T}}. \tag{2.3}$$

The total inductance, L_T, is mainly determined by the physical parameters of the transmission lines of the ring and the total capacitance, C_T, is contributed by the following four aspects:

$$C_T = \sum C_{tline} + \sum C_{invp} + \sum C_{wire} + \sum C_{reg}, \tag{2.4}$$

As shown in Fig. 2.13, C_{tline}, C_{invp}, C_{wire}, C_{reg} are the capacitance of the transmission lines, inverter pairs, tapping wires and the register sinks, respectively. In some applications, the total capacitance C_T is supplemented by the varactor capacitance $C_{varactor}$ for post-silicon tunability, which is essential for the oscillation frequency accuracy and the directionality of the rotary oscillation.

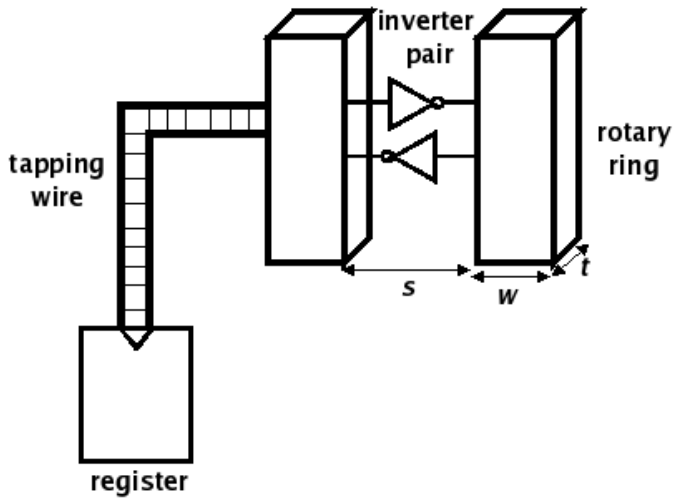


Figure 2.13: The capacitance parameters for RTWO frequency approximation [31].

The power consumption P_{osc} of the ring can be approximated by [76]:

$$\begin{aligned} P_{osc} &= P_{tline} + P_{inv} \approx \frac{V_{DD}^2}{Z_0^2} R_{tot} + P_{inv} \\ Z_0 &= \sqrt{\frac{L_T}{C_T}} \end{aligned} \tag{2.5}$$

where, the most significant power loss mechanism for the ring is the I^2R drop dissipated from the interconnects, P_{tline}. Z_0 is the differential characteristic impedance, it should be noted that the transmission line characteristics dominate over RC characteristics when [19]:

$$R_{tot} < 2Z_0 \tag{2.6}$$

Some previous works aimed at the optimization of the ring frequency and power for a target frequency. In the study of [83], a geometric programming (GP) compatible model is used to construct a single ring with geometric parameters for low power and a desired frequency operation. In the study of [79], the square-shaped ring

structures are analyzed in detail for power minimization. In the work of this dissertation, the power and frequency accuracy are optimized within the resonant rotary clock distribution network design: (1) By optimizing the topology and the placement of the resonant rotary clock global network, the power consumption is optimized; (2) The capacitive loading of the clock distribution network together with the synchronous components are considered in the frequency approximation process in order to improve the frequency accuracy.

2.4.4 Resonant Rotary Clock Distribution Network Design for Larger Scale IC Designs

Single rings can be connected in a checkerboard structure, which is called the rotary oscillator array (ROA), for the distribution of the rotary clock signals to the synchronous components in larger scale IC designs. The ROA topology is shown in Fig. 2.14, which is first introduced in [76] and adopted by most of the researchers as in [28, 30, 81]. Similar to the conventional mesh-based clock distribution network topology, the ROA-based clock distribution network can also be designed in a three level network topology.

The global network: ROA

An ROA topology, which contributes to the global network of the resonant rotary clock distribution network, is used for the generation and distribution of the rotary clock signals to the local areas on the chip. The square-shaped mesh-like global network (the ROA) is first introduced in [76] and widely used later on. The ROA consists of rings as its basic elements, in which the adjacent rings are physically touched with each other. Fig. 2.15 shows the connection between the adjacent rings in an ROA. The physical connection between adjacent rings offers the potential to

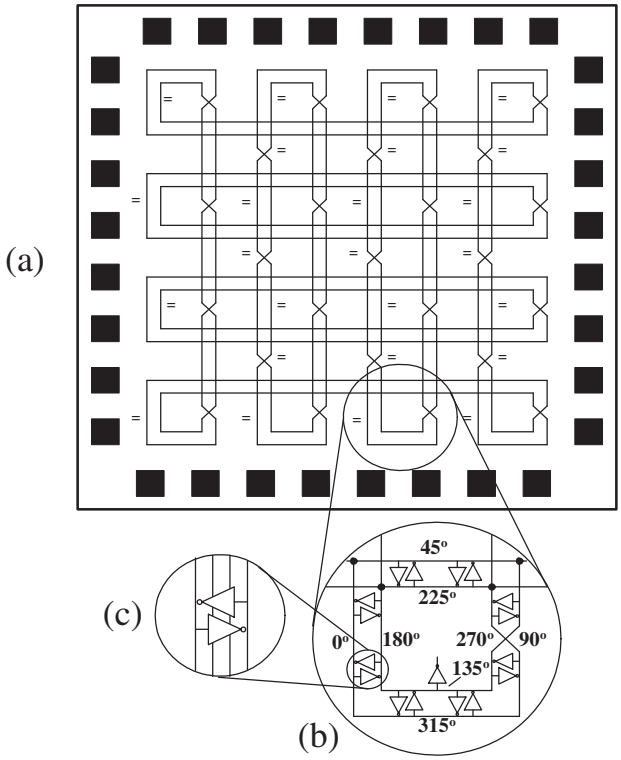


Figure 2.14: The topology of a 7X7 ROA [76].

control skew in spite of relatively open-loop time-of-flight mismatches. As shown in Fig. 2.15, adjacent two rings form a four-port junction. When all the rings in the ROA are synchronized, the oscillation signals of adjacent rings arrive at the junction simultaneously. When there exists a time-of-flight mismatch, one pulse arrives at the junction earlier than the other. The first arriving signal reflects part of its signal power, which delays the its propagation in order to "wait" for the late arrival signal from the other ring. When the second signal arrives at the junction, it then combines with the first pulse at the junction to branch into the two output ports without further reflection. This phase-locking phenomenon occurs by pulse combination at the junction of the transmission lines of different rings, which account for smaller skew among all the rings.

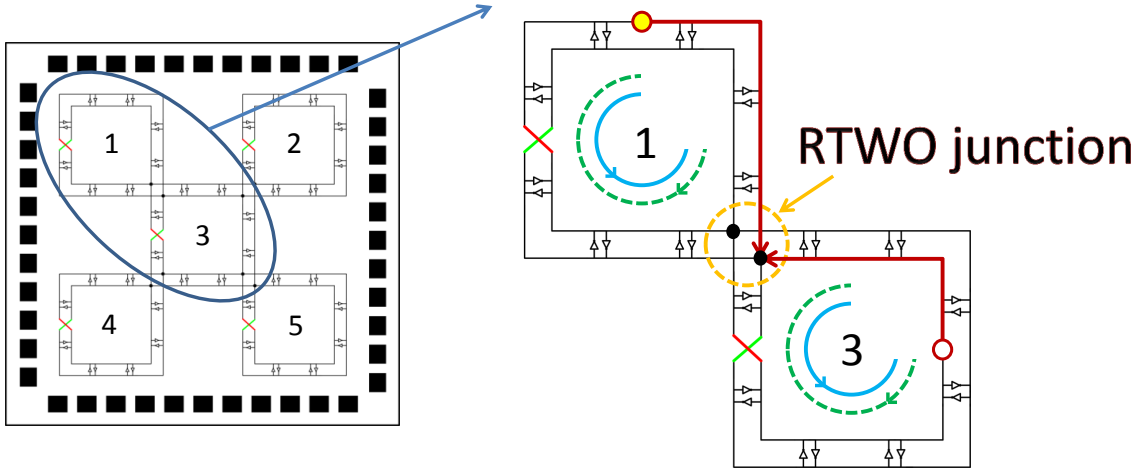


Figure 2.15: The connection between adjacent rings in an ROA.

The regional network: RTWO rings

The regional network designs can be categorized based on their usage for zero/non-zero skew network designs.

The study in [33] is focused on regional network design for non-zero skew network designs. Instead of using the traditional square-shaped ring to construct the resonant network, a custom-shaped ring topology is proposed, which is better for uneven distribution of the synchronous components within local areas. Fig. 2.16 shows the topology of the custom-ROA (CROA) which is composed of custom-shaped rings. In this design, the die area is partitioned into a number of major grids. A custom-shaped ring is placed at the center of each major grid. The topology of the ring is customized based on the synchronous components distribution in each major grid. Thus, the topology of the ring in each major grid is optimized according to the distribution of the local synchronous components independently. The total tapping wirelength in each major grid is optimized with respect to the total tapping wirelength. However, due to the custom shape of the rings, the connection between adjacent rings may be

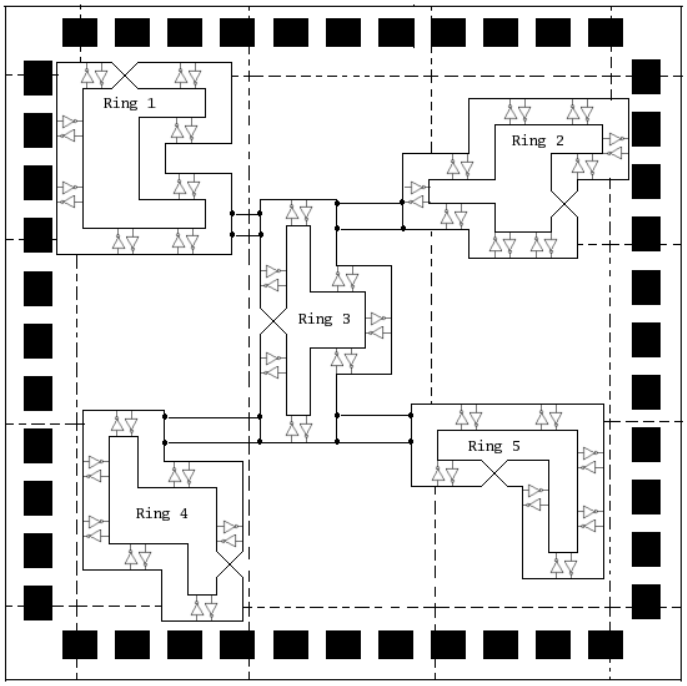


Figure 2.16: The topology of the costom-ROA [33].

built with additional transmission lines instead of the traditional physical touch. The additional transmission line connections between adjacent rings weaken the phase-lock property of the ROA, which introduces skews among the oscillation signals of different rings.

For zero skew regional network designs: Integer linear programming (ILP) is used in [72] to facilitate the design. However, the registers are still connected to the tapping points individually, so that the tapping wire is still huge. The design in [61] combines the tree network and the custom ring topology design together. Given the register locations and the required skew information, a customized topology of the rings is constructed with subnetwork trees connected to each tapping point. Through generating the custom-shaped rings and the subnetwork trees, the tapping wires are saved.

The local network: Sub-network trees

In terms of the local network design of the resonant rotary clocking, similar to the traditional mesh-based clock distribution network: (1) The synchronous components can be attached to the ring tapping points directly; (2) Local sub-network trees can be generated and attached to the ring tapping points. In the study of [72], a tapping point is optimized for each register and each register is connected to the ROA individually. On the other hand, in the study of [42], the synchronous components are clustered into a number of unbuffered zero-skew steiner trees and the number of the steiner trees is the same as the number of rings in the ROA. To this end, the root of each sub-network tree is optimized to be assigned to a tapping point on a ring in order to optimize the total tapping wirelength.

In all these designs, the mesh-like, square-shaped ROA is adopted as the global network without optimization. In this dissertation, the topology and placement are optimized for the global network, which better suits the distribution of the synchronous components. Simulation results show that the optimization of the global network of the resonant rotary clock distribution network greatly reduces power and clock skew.

3. Novel ROA Topology for Resonant Clock Distribution Network Designs

This chapter presents a topology design-based solution that addresses one of the major challenges in the design of Rotary Traveling Wave Oscillator (RTWO) based clock networks—the direction of oscillation. Rotary oscillator array brick (ROA-Brick) structure is proposed that guarantees the uniformity of the signal rotation direction of the signals on all the rings in an ROA. The ROA built from ROA-Bricks has the advantages of: (1) The same phase points on all the rings in the array can easily be tracked, (2) The same phase points set of the ROA are independent of the specific rotation direction of the signals on the ROA. HSPICE simulations demonstrate the superiority of the brick-based ROA circuit design.

This chapter is organized into the following sections. In Section 3.1, the motivation of the ROA topology analysis is proposed. In Section 3.2, the signal rotation direction analysis is performed on a popular 5-ring-ROA structure and the proposed ROA-Brick topology in order to demonstrate its advantages. In addition, a mathematical proof is provided to demonstrate the uniqueness of the ROA-Brick in maintaining the signal rotation direction uniformity. In Section 3.3, the brick-based ROA is proposed, in which the ROA-Brick is proposed to be the building block to form an ROA. In Section 3.4, experiments are presented to demonstrate the correctness of the theory. Conclusions are provided in Section 3.5.

3.1 Motivation

Rotary traveling wave oscillator [76] based resonant clocking technology is a promising approach in resonant clock distribution network designs. However, the

uncertainty of the signal rotation direction on the rings remains an issue which limits the use of an ROA for clock distribution, both for zero skew and non-zero skew clock distribution network designs. Earlier studies have detected this problem [60, 76, 79, 82, 83] and proposed that a single ring will oscillate in the direction with less capacitive loading [76]. However, the capacitive loading for all the rings in the ROA, in order to direct their rotation directions, is a challenging task given that the capacitive load distribution on and between the rings can be non-uniform. The insertion of varactors to scale the capacitive loading, proposed as early as in the pioneering paper [76], bears a sizable overhead and requires post-silicon tuning. The study in [82] has proposed another topology-design-based method for controlling the direction of single ring, however, the proposed ring physical structure involves many corner segments (significantly impacts on oscillation characteristics as shown in [33]) and is difficult for ROA construction due to its non-regularity. Thus, a more comprehensive and adaptive design method of an ROA structure that enables the control of directionality is desirable.

In this chapter, the control of the physical structure of the ring-based ROA on the wave rotation direction in each ring is analyzed critically to observe its implications on the direction of oscillation. Based on this analysis, the topology "ROA-Brick" is proposed as the building block of the ROA, in which the signal rotation direction on all the rings are uniform. The ROA-Brick is used as the basic element in forming the ROA structure, which guarantees the oscillation direction uniformity in all the rings. Mathematical proofs are provided to demonstrate the uniqueness of the ROA-Brick as the only block which maintains this property.

3.2 Proposed ROA-Brick

Despite the popularity of the 5-ring-ROA topology in resonant rotary clock distribution network design, which is shown in Fig. 3.1, this structure has a drawback that the ring elements have no guaranteed control on the rotation direction of their adjacent rings. The wave can propagate in arbitrary directions on each ring and synchronize with its neighbor rings. The inconsistency of the rotation direction of each ring makes it infeasible to tap out the same phase point from each ring element. In order to direct the rings to rotate in a desired direction, several researchers have proposed methods, including modifying the physical structure of the ring [82] and capacitive loading balancing [32]. However, these methods are aimed at controlling the rotation direction of single ring. Controlling the rotation direction of *every* ring in the ROA is infeasible (or very costly for post-silicon tunable varactors on each ring) with these methods. In other words, these methods can help influence the rotation direction of all the rings in an ROA to some level (if done aggressively), however do not guarantee the direction. Furthermore, aggressively applying the aforementioned methods (e.g. capacitive loading) to dictate the rotation direction might degrade the overall oscillation quality. In this chapter, the ROA-Bricks are shown to guarantee the consistency of the rotation direction.

3.2.1 Rotation Direction Analysis of a 5-ring-ROA

Fig. 3.1 shows the topology of the popular 5-ring-ROA structure. The rings are drawn separated from each other (they are adjacent in physical implementation) for illustration purposes.

Assume the black squares on ring `R3` are the tapping points providing the desired clock signal phase. To facilitate the analysis, symbols `1` and `2` are used to denote that the signal on the ring is propagating in either the clockwise or the counter clockwise

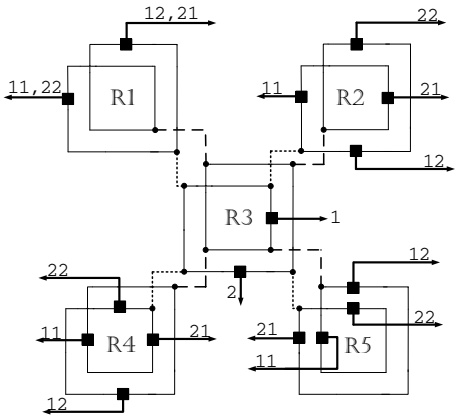
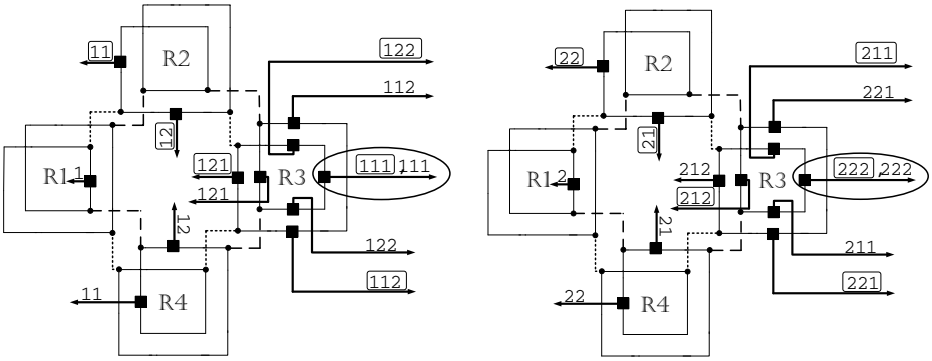


Figure 3.1: Possible same phase points of 5-ring-ROA.

direction, respectively. For instance, the symbol `1` on the tapping point on `R3` indicates that the signal on `R3` is assumed to be propagating in the clockwise direction. The same phase points (to tapping point `1` or `2` on `R3`) on the other four rings can not be fully determined since the positions are related to the rotation direction on these four rings, which are not known with certainty. For instance, consider `R2`: If the signal on `R2` is also propagating in the clockwise direction as `R3`, the same phase point is at `11` (the first symbol `1` denotes the signal rotation direction on `R3` and the second symbol `1` denotes the signal rotation direction on `R2`). Alternatively if the signal on `R2` is propagating in the counter clockwise direction, the same phase point is `12`. Thus, there are two possible same phase points on `R2`, only one of which would have the same phase as `1`. Similar analysis are performed with `R1`, `R4` and `R5` as shown in Fig. 3.1, which shows that none of the rings neighboring `R3` can provide a unique same phase point as tapping point `1`. The repeated analysis for an initial assumption of counter clockwise direction on `R3` instead (e.g. tapping point `2`) leads to similar results. Thus, the same phase points analysis on the 5-ring-ROA indicates that even if the signal propagation direction on `R3` is known with certainty (e.g. through the use



(a) clockwise direction (b) Counter clockwise direction

Figure 3.2: ROA-Brick rotation direction uniformity analysis.

of topology design, capacitive balancing or varactors), the rotation direction on the other four rings need to be tested first to determine the same phase point positions.

3.2.2 Rotation Direction Analysis of the ROA-Brick

Fig. 3.2 is the topology of the proposed ROA-Brick. For completeness, the rotation direction analysis of the ROA-Brick is performed both in clockwise and counter clockwise directions. In Fig. 3.2(a), consider that all four rings in the ROA-Brick are oscillating in a steady state and the signal on `R1` is oscillating in the clockwise direction (indicated as `1`). The sampled point `1` on `R1` has two possible same phase points each on `R2` and `R4` and these points are marked `11` and `12`. The possible same phase points on `R2` and `R4` each has four possible same phase points on `R3`. In total, there are seven possible same phase points on `R3` (the `111` point of both paths is at the same position). The possible same phase points on the propagation analysis path {`R1`, `R2`, `R3`} are circled and the same phase points on the propagation analysis path {`R1`, `R4`, `R3`} are not circled such that possible same phase points traced through different pathes are distinguishable. Fortunately, not all of these seven points are achievable in the ROA-Brick. The realizable points should satisfy two requirements:

1. When all the rings are stabilized, the same phase point on R3 should be unique, regardless of tracing through the propagation analysis path {R1, R2, R3} or through the propagation analysis path {R1, R4, R3}. Thus, the points where circled points and uncircled points coincide with each other are realizable points,

2. The last digit of the symbol on the points on R3 denotes the rotation direction of R3. Thus, this number should also be the same of both the circled and uncircled points.

Based on these requirements, only point 111 in Fig. 3.2(a) is realizable. Correspondingly, only 11 on R2 and R4 are realizable. This result indicates that R1, R2, R3 and R4 should all rotate in the clockwise direction, the assumed direction of R1. The analysis performed for the assumption of the rotation direction of R1 to be counter clockwise is illustrated in Fig. 3.2(b). Similar conclusions are drawn: In steady state, the signals on all the four rings are propagating in the counter clockwise direction.

Based on the above analysis, it is concluded with certainty that the ROA-Brick possesses the capability of forcing all the signals on the rings to propagate in a consistent direction, either clockwise or counter clockwise. The properties of the ROA-Brick are as follows:

1. The same phase point set (one point from each ring) of the ROA-Brick can be arbitrarily selected and is independent of the rotation direction of the signals on the rings.

2. The positions of the same phase points in the same row are the same, such as 1 and 111 in Fig. 3.2(a).

3. The positions of the same phase points of every other row are of π difference, such as 11 on R2 and 11 on R4. Note that the π phase difference corresponds to the closest pair of points on the inner and outer rings on the ring topology.

3.2.3 Proof of the Uniqueness of the ROA-Brick

The restricted ring rotation direction combinations of the ROA-Brick stems from its four ring structure. In the following part of this section, through definitions and theoretical analysis, a proof is provided showing that the ROA-Brick is the unique ROA (sub) structure which can maintain only two ring rotation directions.

Definitions

Definition 1. *Feasible ROA: A feasible ROA is an ROA structure, in which exists only two ring rotation direction combinations for all the rings of the topology in a steady oscillation state, P and P'. The signal rotation direction on each ring under P' is opposite to the ones under P.*

Definition 2. *Ring loop: A ring loop is an ROA structure, in which each ring is connected to only two other rings. Each point can be uniquely looped back to itself through other rings and all the other rings are traversed in the loop. For example, the topologies in Fig. 3.4 and Fig. 3.5 are ring loops. However, the topology in Fig. 3.6 is not a ring loop, as* `R5` *and* `R2` *are connected to 3 rings each.*

Analysis

In order to demonstrate the uniqueness of the ROA-Brick, three lemmas are presented as follows and a theorem is formulated based on the lemmas.

Lemma 3. *All rings in a feasible ROA should connect to at least two other rings.*

For a proof through contradiction, assume there exists a feasible ring loop, in which a ring is connected to only one of the other rings, such as `R2` shown in Fig. 3.3. When a single ring is oscillating in a steady state, considering any corresponding points A and A' on the inner and outer rings, point A' is always four edges away from

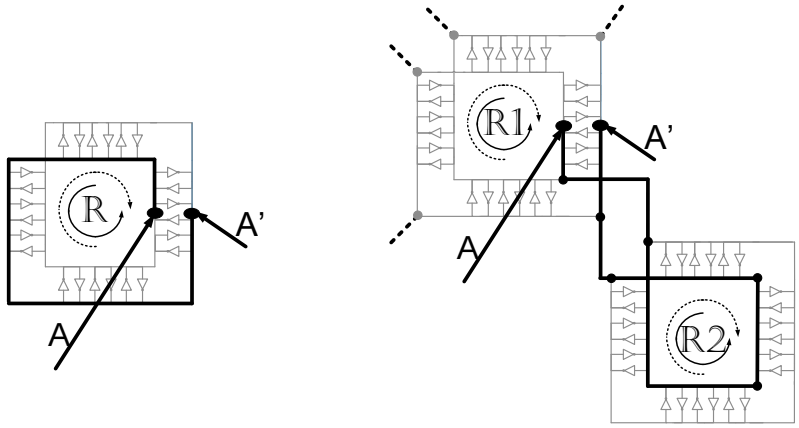


Figure 3.3: The phase difference between two rings.

A. During the signal propagation, each edge contributes $\frac{\pi}{4}$ phase difference. Thus, the oscillation signals at point A and A' has a phase difference of π regardless of the signal oscillation direction.

Assume the ring array is in a steady oscillation state under the ring rotation direction combination P. A new ring rotation direction combination P'' is defined as the signal rotation directions on all rings except `R2` are the same to the ones under P. No matter which direction the ring `R2` rotates, the phase difference between the signal that goes into and comes out of `R2` is always π. If `R2` oscillates in the opposite direction, the oscillation signals at the corresponding points on other rings, e.g. A and A', can still keep a phase difference of π. Thus, under the ring rotation direction pattern P'', the ring array is in a steady oscillation state; However, the direction pattern P'' is neither the same as P nor the opposite P', which is contradiction to that the ring array is feasible. Hence, a feasible ring loop does not have a ring connected to only one of the other rings.

Lemma 4. *ROA-Brick is the only feasible ring loop. A ring loop with more than four rings is not feasible.*

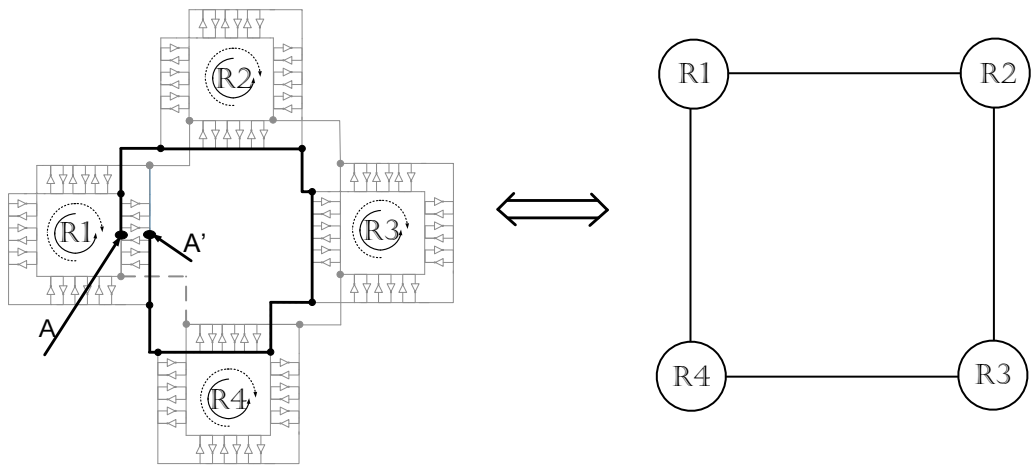


Figure 3.4: Signal propagation around an ROA-Brick.

In a ring loop, each ring is connected to its two adjacent rings. Thus, the ring topology is equivalent to a square topology. For instance, the ring loop with four rings is a 2×2 square. Based on the discussion in Section 3.2.2, the loop containing four rings is feasible. As shown in Fig. 3.4, when the signal traverses from `A` through each rings of the ROA-Brick and comes back to `A'`, the smallest loop path contains four edges in total. Under this condition, in order to maintain the π phase difference between `A` and `A'` in ring `R1`, all of the edges along the path should contribute either all $\pi/4$ or $-\pi/4$ phase difference ($| \frac{\pi}{4} \times 4 |=| \frac{-\pi}{4} \times 4 |= \pi$). The same sign of phase contributions indicates that all the rings are rotating in the same direction. Thus, the ROA-Brick is a feasible ring loop, in which only two ring rotation direction combinations can be formed: all in clockwise or all in counter clockwise.

On the other hand, consider the $M \times K$ ring loop, where $M, K \geq 3$. For example, a 3×3 ring loop (8-ring-ROA) is shown in Fig. 3.5. On each edge of the square, there is always at least one ring, which is connected to other rings at the diagonal corners, as `R2`, `R4`, `R6` and `R8` in Fig. 3.5, and define these rings as diagonal connected rings.

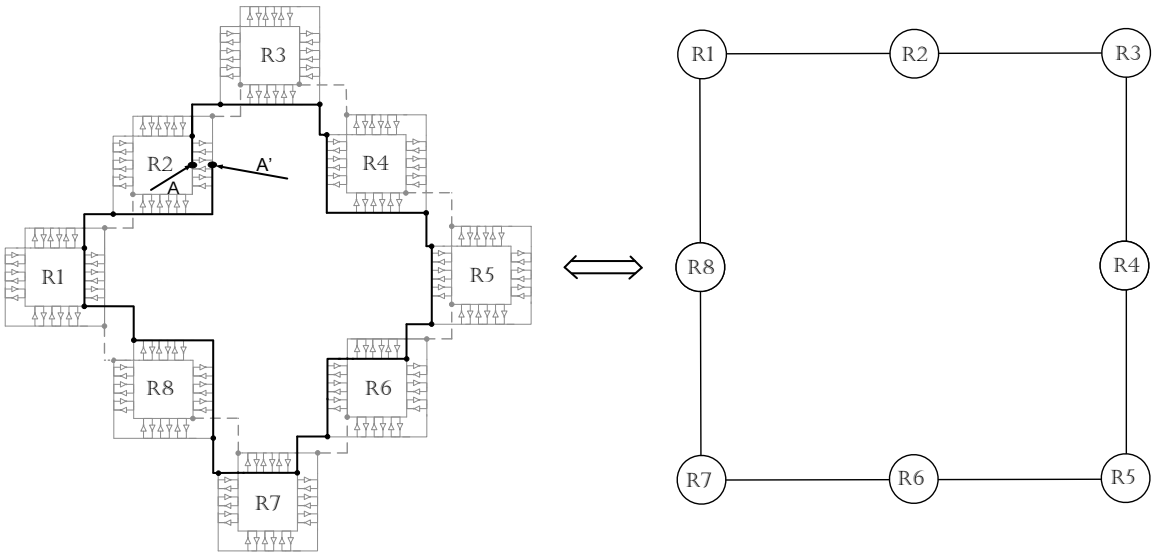


Figure 3.5: Signal propagation around an 8-ring-ROA.

Assume the $M \times K$ ring loop has already oscillated in a steady state. Consider a signal propagation loop connecting any two corresponding points on the inner and outer rings. In any signal propagation loop, the signal travels one or two edges on each ring in the loop, which corresponds to $\pm\frac{\pi}{4}$ or $\pm\frac{\pi}{2}$ phase differences. The diagonal connected ring is the one corresponds to $\pm\frac{\pi}{2}$ phase difference. The total phase difference contributed by any two diagonal connected rings could be: $-\pi$, 0 and π. By only changing the rotation directions of two diagonal connected rings, the total phase change in the signal propagation loop is either $\pm 2\pi$ or 0. For example, if the phase difference contributed by the two diagonal connected rings is π, after changing their directions, the phase difference contributed by the diagonally connected rings is $-\pi$. Thus the total phase change in the signal propagating loop is -2π, i.e., 2π. In other words, by changing the rotation direction of an even number of diagonal connected rings, the two corresponding points still have a phase difference of π. This implies that the ring loop can still reach a steady state under the new oscillation

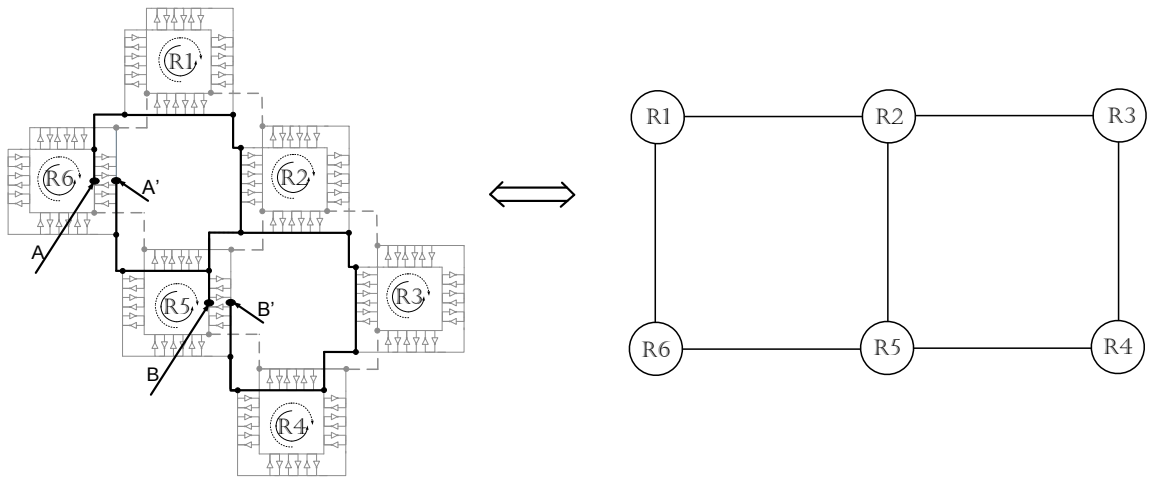


Figure 3.6: Signal propagation around a 2×3 ring topology.

rotation direction pattern. Thus, the rotation pattern is non-unique for an $M \times K$ ring loop, where $M, K \geq 3$, which makes it a non-feasible ring loop.

Lemma 5. *Any $2 \times N$ rectangular ring topology is not a ring loop, but a connection of ROA-Bricks.*

For the proof of Lemma 5, consider a $2 \times N$ rectangular ring topology. For example, a 2×3 rectangular ring topology is shown in Fig. 3.6. It is trivial to observe that the 2×3 rectangular topology is constructed by $N - 1$ ROA-Bricks. On the other side, since some of the rings in the topology has been connected to more than two other rings, the $2 \times N$ topology is not a ring loop.

Theorem 6. *ROA-Brick is the only feasible ring loop.*

Proof. The proof of Theorem 6 is based on the conclusions of the three Lemmas. First, based on Lemma 3, each ring in the feasible ring loop should connect to two other rings in order to form a ring loop. There is no ring in a ring loop which could have only one connection to other rings. Second, the conclusions of Lemma 4 and

Lemma 5 indicate that the ROA-Brick is the only feasible ring loop with the feature that only two ring rotation direction combinations can exist in the topology in a steady oscillation state. Any $M \times K$ ring loop, where $M, K \geq 3$ are not feasible. Any $2 \times N$ topology comes from a connection of ROA-Bricks and the ring loop within the topology is an ROA-Brick. As stated above, the only feasible ring loop is the proposed four ring loop, which is named as an ROA-Brick. □

3.3 Brick-based ROA

The unique property of the ROA-Brick, which guarantees the rotation direction uniformity, should be implemented in the ROA construction. In other words, instead of using single rings to form an ROA, the ROA-Brick should be used as the basic element in building ROAs. It is observed that the 3×3 ROA topology shown in Fig. 3.7 is the traditional 13-ring-ROA topology after removing the four corner rings. In other words, eliminating the four corner rings from the traditional 13-ring-ROA topology, the signal rotation direction on the remaining structure is uniform and the same phase points on each rings are known with certainty. This ROA structure, which is composed of four ROA-Bricks, is named brick-based ROA. A sample group of same phase points is shown in Fig. 3.7. These same phase points satisfy all three properties of an ROA-Brick listed in Section 3.2.2.

The brick-based ROA shown in Fig.3.7 reveals three properties:

1. Each ring belongs to at least one ROA-Brick,

2. Each ROA-Brick shares at least one of their rings with another ROA-Brick,

3. All the ROA-Bricks are linked together.

Since all the rings in a brick-based ROA are rotating in the same direction, the relative positions of the same phase points (tapping points) are fixed on a brick-based

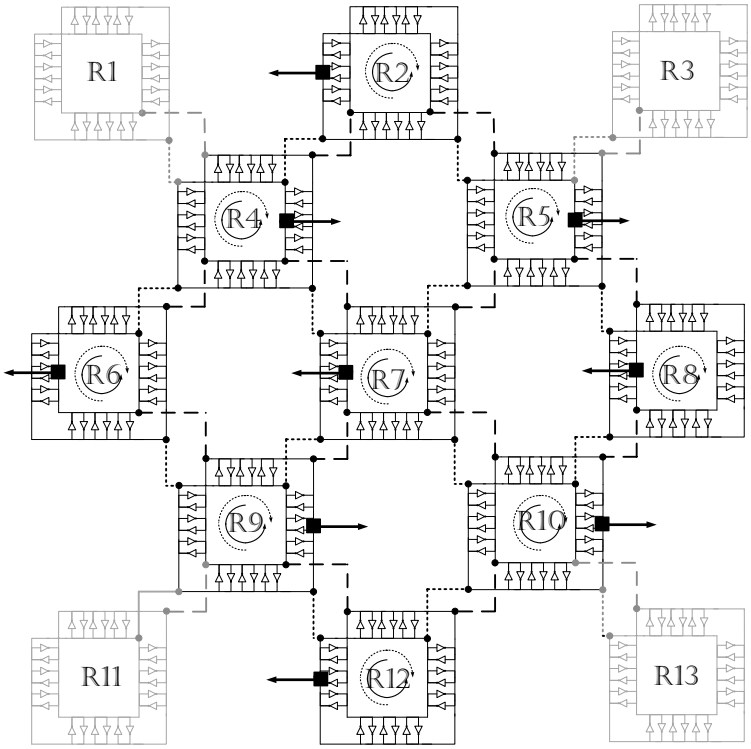


Figure 3.7: Same phase points of a 3×3 ROA.

ROA. Thus, when one point is selected to be a tapping point on a ring, the position of the tapping points on other rings are determined correspondingly.

3.4 Experiments

In this section, an ROA-Brick is built using a 90nm technology. HSPICE simulations are performed to confirm the signal rotation direction uniformity of the ROA-Brick that has been theoretically demonstrated in Section 3.2.2. The performances of the 5-ring-ROA and the 8-ring-ROA in Fig. 3.5 are also tested to empirically verify the theory. Finally, the simulation result of the brick-based ROA in Fig. 3.7 is demonstrated.

3.4.1 The Rotation Directions

HPSICE simulations are performed to test the signal rotation directions of the ROA-Brick, 5-ring-ROA and the 8-ring-ROA. The results are shown in Fig. 3.8. These simulations are performed for identical capacitive loading at frequency 6.6GHz in a 90nm technology. The ROA-Brick simulation result is shown in Fig. 3.8(a), in which the outputs `TP1`—`TP4` are the same phase points as shown in Fig. 3.2(a) from `R1`—`R4`. The simulation results in Fig. 3.8(b) show the signal rotation direction from `R1`—`R5` in the conventional 5-ring-ROA. The detected points are the consecutive points selected along the transmission line of each ring in the clockwise direction. The signal on `R1` is rotating in the clockwise direction, similar to the signals on `R3`, `R4` and `R5`. However, the signal on `R2` is rotating in the counter clockwise direction. The rotation direction inconsistency does not affect the synchronization among each ring. The simulation results in Fig. 3.8(c) show the signal rotation direction inconsistency of the rings in the 8-ring-ROA structure. The output signals are selected from the tapping points listed in Fig. 3.7 from `R2`—`R12` except `R7`. The simulation result shows that the 8-ring-ROA structure does not guarantee the signal rotation direction uniformity.

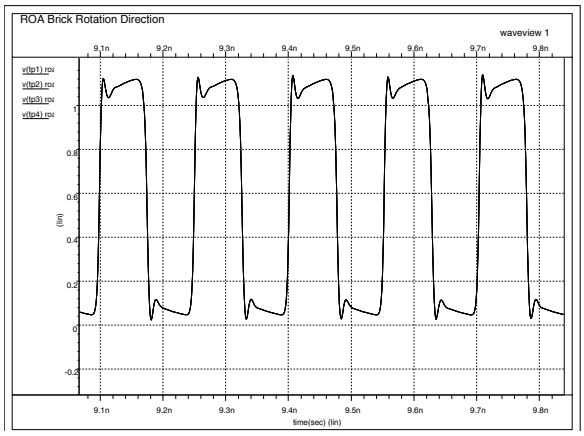
3.4.2 Brick-based ROA

The signal rotation direction simulation is performed to demonstrate the feasibility of the brick-based ROA design in Fig. 3.7, as well. The simulation result is depicted in Fig. 3.9, in which the brick-based ROA is operating at 6.6GHz. Initially, each ring starts to oscillate in its own direction, but eventually synchronizes with neighbor rings and reach rotation direction uniformity. At the beginning, if two neighboring rings are rotating in opposite directions, one of the two rings forces the other ring to oscillate in its own direction such that steady waveforms are formed in each ring. After power up, about 3ns is consumed such that all the rings in the ROA correct

their rotation direction and form a steady waveform. The start-up jitter is due to the self-oscillating and phase-locking property of rotary clocking and is observed for all methods of directionality adjustment, including varactors, capacitive loading and topology design.

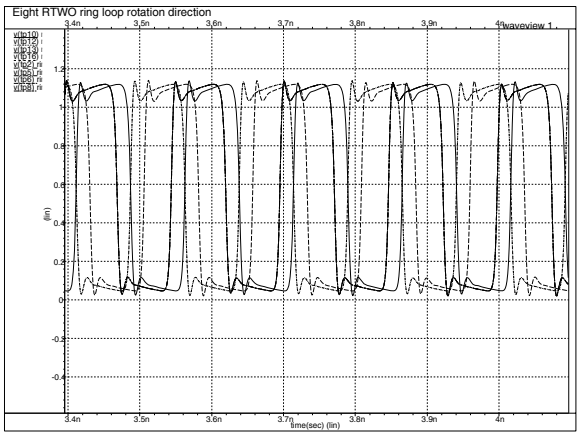
3.5 Conclusions

In this chapter, the topology of ROA-Brick is proposed which guarantees the signal rotation direction uniformity. Furthermore, This block is used in building brick-based ROA, in which the signals on all the rings are rotating in the same direction and the same phase points are easily determined. Simulation results confirm the signal rotation direction uniformity property of the ROA-Brick and the brick-based ROA.



(a) Consistent ROA-Brick rotation direction.

(b) Inconsistent 5-ring-ROA rotation direction.



(c) Inconsistent 8-ring-ROA rotation direction.

Figure 3.8: ROA-Brick, 5-ring-ROA and 8-ring-ROA rotation directions.

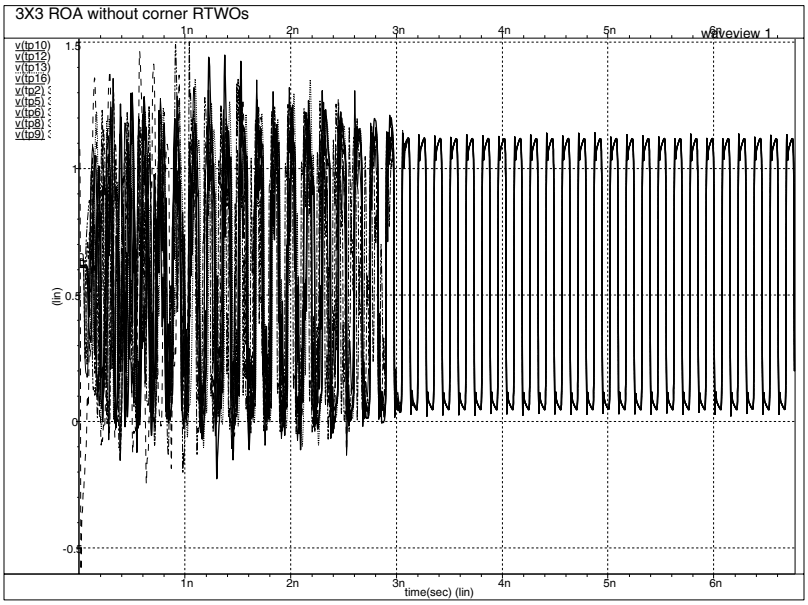


Figure 3.9: Simulation result of the rotation direction of the 3×3 ROA without corner rings.

4. Brick-based Rotary Oscillator Arrays Synchronization Scheme and Low-Skew Clock Network Design Methodology

In Chapter 3, the ROA-Brick is proposed to be used as the basic building block of the ROA instead of single rings. In this chapter, the proposed ROA-Brick circuit topology is used to develop a brick-based ROA. The brick-based ROA maintains the same properties as a single ROA-Brick:

1. Only two feasible ring rotation direction combinations in the ROA;
2. The same phase tapping points of all the rings in the ROA being identifiable;
3. The same phase tapping points of the ROA being independent from the two possible rotation directions.

4.1 Motivation

The properties of the brick-based ROA guarantee that it can be applied as the global network of resonant rotary low-skew clock distribution network. In order to solve the large scale brick-based ROA frequency synchronization and signal rotation direction uniformity problem, a brick-based ROA synchronization scheme is proposed, which can: (1) Effectively accelerate the synchronization process of the ROA network; (2) Help form a desired signal rotation direction on the brick-based ROA. Furthermore, in order to facilitate the brick-based ROA clock distribution network design, a design methodology is proposed with clock skew optimization. The proposed design methodology is tested with ISPD 10 clock benchmarks, demonstrating the GHz operation with the low-skew clock distributions through HSPICE.

This chapter is organized into the following sections. In Section 4.2, a brick-based ROA synchronization scheme is proposed, which effectively accelerates the synchro-

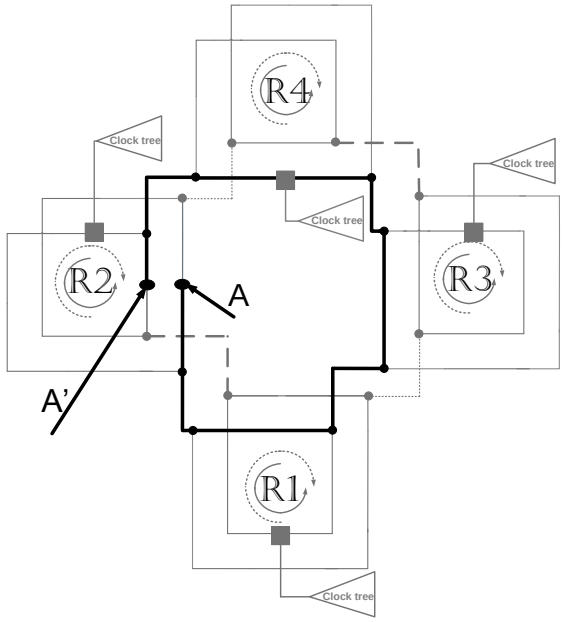


Figure 4.1: ROA-Brick driving local clock networks.

nization process of the brick-based ROA. In Section 4.3, a brick-based ROA clock distribution network design methodology is proposed. In Section 4.4, experiment results are presented to demonstrate the effectiveness of the proposed methodology in resonant rotary clock distribution network construction. Conclusions are provided in Section 4.5.

4.2 Brick-based ROA Synchronization Scheme

When the brick-based ROA starts to oscillate, all the rings of the ROA need to coordinate their individual oscillating frequencies and signal rotation directions. This process is called synchronization.

The *frequency synchronization* of an ROA is essential to provide consistent clocking between the self-oscillation frequencies of each ring connected in locked-in-phase pattern in the ROA. For instance, consider the ROA-Brick illustrated in Fig. 4.1. The

mismatch of natural oscillation frequencies of each ring is caused by the mismatch of total capacitive loading of each ring. The total capacitive loading of each ring is contributed by two parts: (1) Ring loading: The capacitive loading of the transmission lines and inverter pairs of the adjacent rings. (2) Circuit loading: The capacitance of the clock networks loaded on each ring. Due to the mismatch of the ring loading (the rings in ROA-Brick do not have ring loading mismatch) and circuit loading, each ring in the brick-based ROA needs to coordinate its own natural frequency with that of the adjacent rings and finally reach a uniform oscillation frequency for the entire ROA system.

The *signal rotation direction synchronization* is essential to provide multi-phase operation capability. The study in [76] reveals that the signal on a single ring tends to propagates in the direction of less capacitance. This dominant direction on an individual ring can be different when a ring is not connected to the ROA versus when it is connected, as necessitated in brick-based ROAs. Thus, after the brick-based ROA is started up, some rings need to change its natural signal rotation direction in order to adapt to the uniform signal rotation direction of the ROA system.

The brick-based ROA synchronization process contains both frequency and signal rotation direction synchronization. In the proposed synchronization scheme, the two are mutually coupled. This is because the uniform oscillation frequency, which is the average oscillation frequency of all the rings, can be easily obtained through phase averaging at the junctions of adjacent rings as long as all the rings are rotating in the same direction.

The proposed synchronization process divides the system startup process into two periods: Pre-synchronization period and synchronization-period. In the pre-synchronization period, only one ring is powered up and allowed to propagate its oscillation signal to the other rings in the ROA. In the synchronization-period, all the rings

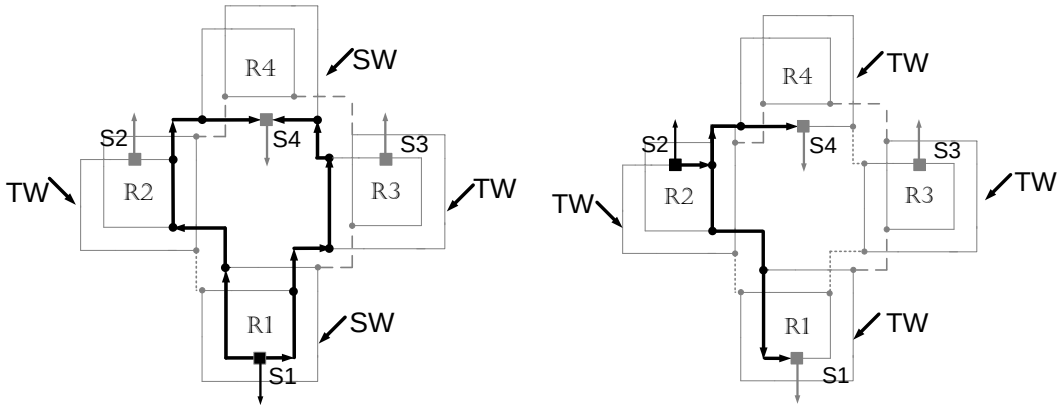
are powered up and reach synchronization with certainty. In the proposed synchronization scheme, the "master" ring to start-up initially during the pre-synchronization period is selected methodologically as it has direct implications on the signal rotation direction synchronization.

4.2.1 Master Ring Selection

The ring in an ROA which possesses the highest driving capability determines the uniform signal rotation direction and direct all the other rings to rotate in its direction [76]. In previous applications, since all the rings in the ROA are loaded with clock networks with relatively balanced capacitive loads [42], the difference of driving capability among rings are not significant. In order to start up the ROA circuit efficiently and guarantee the uniformity of the signal rotation direction of the rings in the ROA, one ring is selected as the master ring, which is assigned zero circuit loading. In other words, no registers are connected to the tapping points on the master ring, whereas the remaining rings are used to synchronize the circuit. The master ring starts to oscillate in the pre-synchronization period. The remaining rings, which are called slave rings, are delayed in the power up until the synchronization-period. Before the synchronization period, the master ring forms a steady oscillation signal, which is propagated to the other rings through the junctions between adjacent rings. The master ring is constructed based on the following rules:

1. The master ring is in only one ROA-Brick.
2. The master ring is only loaded with ring capacitance (i.e. zero circuit load capacitance).
3. The tapping point of the master ring has asymmetrical adjacent tapping points.

Rule 1 guarantees that the master ring has the smallest ring capacitive loading. The ring selected by Rule 1 only has two adjacent rings, which is the smallest ring



(a) R1 is not a master ring candidate.

(b) R2 is a master ring candidate.

Figure 4.2: Master ring selection based on traveling wave (TW) and standing wave (SW) resonance implications.

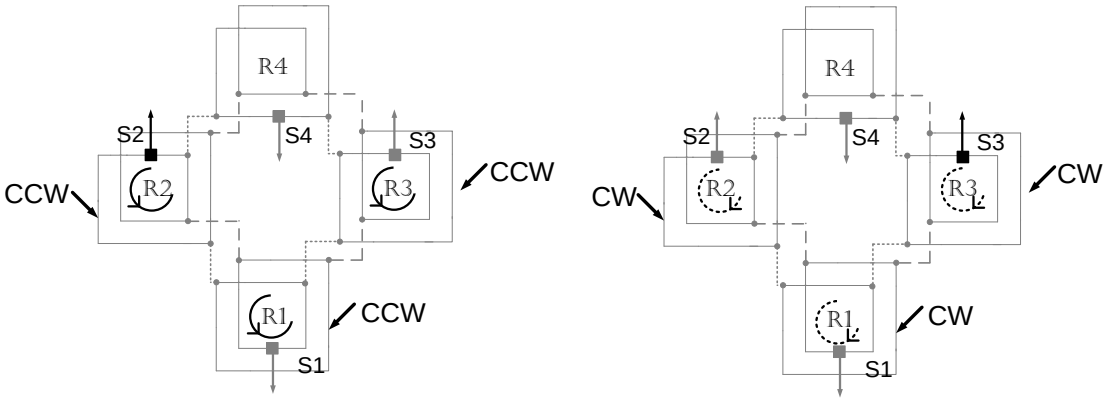
capacitive loading that could be obtained in the ROA structure. Rule 2 requires that the master ring is not assigned with circuit loading which further guarantees the master ring to maintain the highest driving capability. Rule 3 is related to the oscillation starting point on the master ring through the startup circuit. In the proposed synchronization scheme, the oscillation on the master ring starts to propagate from its tapping point position. For instance in Fig. 4.2(b), `R2` and `R3` are the candidates of the master rings. `R2` is a candidate that satisfies Rule 3 because the adjacent tapping points of `S2`—`S1` and `S4`—are asymmetrical. `S4` is 1 edge away from `S2` and `S1` is 3 edges away from `S2`. The asymmetrical tapping point location leads to asymmetrical circuit loading distribution among rings, which is leveraged to define the "known" dominant direction for oscillation during start-up.

4.2.2 Signal Rotation Direction Control

By picking `R2` or `R3` as the master ring, the uniform signal rotation direction is known through the synchronization process. As shown in Fig. 4.3(a), if `R2` is selected as the master ring, the oscillation begins to propagate from `S2` seeing imbalanced ca-

pacitive loading between the two directions. Thus, a counter clockwise traveling wave is formed on R2 during the pre-synchronization period. When the signal propagates to R1 and R4, these two rings form traveling waves due to the capacitance imbalance between the two directions seen from their junctions with R2. The traveling wave propagation directions on R1 and R4 are opposite. Usually, the circuit loading is less than the ring loading. Thus, R1 forms a counter clockwise traveling wave, the signal rotation direction of which is the same as that on R2 so as to maintain stronger oscillation signals. Referring to R4, both the circuit loading and ring loading are on the right hand side seen from the junction between R2 and R4. Thus, R4 is driven to form a clockwise traveling wave, the signal rotation direction of which is conflicting with that on R2. The signal amplitude on R4 is attenuated. When the signals of R1 and R4 converge on R3, both of these oscillation signals try to direct R3 to follow their signal rotation directions. Due to the higher power of the oscillation signal on R1 and the clock network loading position on R3, the oscillation signal on R3 will follow the signal rotation direction of R1, which is counter clockwise. Thus, during the pre-synchronization period, R2, R1 and R3 have formed a uniform signal rotation direction and the attenuated oscillation signal on R4 will be easily overwritten when all the rings are powered up.

Alternatively, as shown in Fig. 4.3(b), if R3 is selected as the master ring and the oscillation signal begins to propagate from S3, a traveling wave with clockwise signal rotation direction forms during the pre-synchronization period. Through similar analysis, it is derived that R3, R1 and R2 will have a uniform signal rotation direction and the signal rotation direction on R4 will be conflicting. The signal amplitude is also attenuated on R4 due to the conflict. Thus, when all the rings are powered up, the signal rotation direction on R4 is easy to be overwritten so as to adapt to the uniform clockwise signal rotation direction on the ROA-Brick.



(a) Counter clockwise (CCW) signal rotation direction.

(b) Clockwise (CW) signal rotation direction.

Figure 4.3: Master ring affecting rotation direction.

Consider the following analysis for the case where Rule 3 is not included in the master ring construction rules. In this undesirable case, all the rings on the periphery of the brick-based ROA can be randomly picked as the master ring. Note that, this case illustrates the current state-of-the-art, as start-up sequencing of resonant rotary clocking has not been analyzed critically. Suppose `R1` in Fig. 4.2(a) is selected as the master ring and the oscillation signals start to propagate from `S1`. The oscillation signals propagating in both directions see almost equal capacitance due to the symmetrical structure of `R2` and `R3` and the symmetrical tapping point locations of `S2` and `S3`. Thus, a standing wave (SW) is more likely to be established on `R1` rather than a traveling wave (TW) during the pre-synchronization period. Traveling waves are formed on `R2` and `R3` due to the asymmetrical capacitive loading on both sides seen from the their junctions with `R1` and the signal rotation directions are opposite. When oscillation signals on `R2` and `R3` converge on `R4`, the equal power, opposite direction traveling waves form a standing wave on `R4` similar as that on `R1` and the tapping point `S4` becomes a standing point. Thus, during the pre-synchronization period, standing waves are formed on two rings and the traveling waves on `R2` and `R3`

choose their signal rotation direction freely. When all the rings are powered up, the signals on `R1` and `R4` starts to form their traveling waves, yet their rotation directions are unknown. Thus, selecting `R1` as the master ring does not effectively assists the synchronization process.

4.3 Brick-based ROA Low-skew Clock Distribution Network Design Methodology

As shown in Chapter 3, the same phase tapping points on the brick-based ROA are always identifiable. These tapping points can be used as single-phase clock sources for clock distributions. Given a floorplan, the design process can be concluded as follows:

1. Group the synchronous components into a number of clusters according to the driving capability of the ROA (decided based on C_{reg} computation and C_{wire} estimation in Eq. 2.4) and the chip area.

2. All the clusters are built into (ideal) zero-skew sub-network trees.

3. Design an ROA and optimize the one-to-one pairs between the tapping points and the sub-network tree roots for minimizing the total stub wirelength.

4. Optimize the tapping point locations for skew reduction.

The details of these four steps of the design flow are described as follows.

4.3.1 Cluster Generation

The registers are first grouped into a number of clusters with similar capacitance. In this work, the well-known k-means algorithm [17] is used for cluster generation. Capacitance balancing is critical to form the stable oscillation signals on the rings

in the ROA [72]. However, the original k-means method does not guarantee all the clusters maintain similar total capacitance. To this end, a balanced k-means method [26] is used instead to deal with the cap-balancing problem. After the registers are grouped into a number of clusters using the k-means algorithm, some registers are transferred among adjacent clusters iteratively until the largest total capacitance difference among all the clusters falls into a tolerance region. A user-defined capacitance tolerance ratio is expressed as:

$$\tau = (C_{max} - C_{min})/C_{max} \tag{4.1}$$

Here, C_{max} and C_{min} are the maximum and minimum total capacitance among all the clusters. The value of τ should be as low as possible but can be as high as 0.3 without causing the oscillation to fail. In the experiments, τ is selected to be around 0.2 and 0.3. The capacitance tolerance ratio τ is used in Section 4.4.2 to maintain the capacitance balancing after the real sub-network trees are generated.

The detailed algorithm is shown in Algorithm 1. The inputs are the sink set R, the number of sub-network trees m, and the capacitance tolerance ratio τ. The outputs are the cap-balanced clusters $K_1, K_2, \cdots, K_m$. The initial clusters are generated using the balanced k-means algorithm. Then, the largest capacitance difference among the clusters is examined to test whether the generated clusters satisfy the capacitance balance requirements. The total capacitance of each cluster is estimated by:

$$C_i^{tot} = \sum_{j=1}^{n_i} (C_{i,j} + C_{i,j}^{wire}) \tag{4.2}$$

Here, $C_{i,j}$ is the sink capacitance of the register $r_{i,j}$ in cluster K_i, where $j = 1 \cdots n_i$ and n_i is the total number of registers in cluster K_i. The capacitance $C_{i,j}^{wire}$ indicates the Manhattan distance between register $r_{i,j}$ and the centroid of cluster K_i, cen_i.

The capacitance C_i^{tot} of each cluster i includes both the register capacitance and the wire capacitance. Thus, C_i^{tot} evaluates both the load capacitance and the density of the elements in the cluster. Here, the distance from each register to the centroid of its cluster is used to approximate the total wirelength of the cluster. After the initial clustering, the heuristic iterations are used to balance the total capacitance among all the clusters. While the largest capacitance difference among all the clusters remains above the capacitance tolerance ratio τ (Line 6), the registers are moved out of the cluster which has the largest total capacitance to its neighbor cluster in order to balance the total capacitance among all the clusters. For each register in the cluster with the largest capacitance, the distance from the register to the centroid of other clusters are evaluated. The register which maintains the smallest distance to a centroid of other clusters is selected to be moved to the target cluster (Line 7). In order to avoid the condition that a register repeatedly goes out and in to one cluster and the algorithm never converges, once a register is moved out of one cluster, it is not allowed to be moved in again. The process continues until the largest capacitance difference among all the clusters falls below the tolerance ratio.

It is a challenging problem with resonant clocks that both the capacitance from the ring itself and the capacitance contributed by the sub-network trees influence the operating frequency of the resonant system. Nonetheless, by budgeting them in design and with the use of the clustering algorithm here, an initial estimation of the final operating frequency is feasible. If a more accurate frequency target is desired, a frequency tuning process [76] should be applied instead. For a quick determination of the frequency as presented here, the estimation and clustering of capacitive loads work well.

Algorithm 1 Register cluster generation

Input: Sink set R, number of sub-network trees m and capacitance tolerance ratio τ
Output: Cap-balanced cluster set $\{K_1, \cdots, K_m\}$
1: $[\{K_1, \cdots, K_m\}, \{cen_1, \cdots, cen_m\}]$=k-means(R, m).
2: **for** i=1 to m **do**
3: $C_{max} \longleftarrow$ Largest capacitance cluster K_{i_1}.
4: $C_{min} \longleftarrow$ Smallest capacitance cluster K_{j_1};
5: **end for**
6: **while** $(C_{max} - C_{min})/C_{max} > \tau$ **do**
7: Find $\min\{d(r_{i_1,j}, cen_{i_2})\}$, where $r_i \in K_{i_1}$, $i_2 = \{1, 2, \cdots, m, i_2 \neq i_1\}$
8: $K_{i_2} = K_{i_2} \cup r_{i_1,j}$
9: Label $r_{i_1,j}$ cannot be moved back to K_{i_1}
10: Update C_{max} and C_{min}
11: **end while**

4.3.2 Zero-skew Sub-network Tree Generation

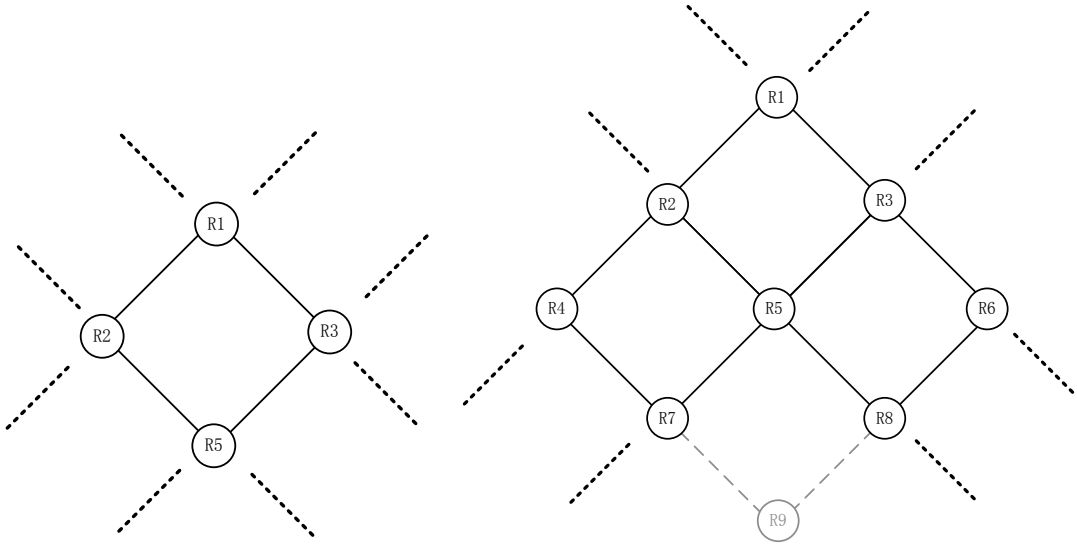
After the m clusters are generated, (ideally) zero-skew sub-network trees are grown for all the clusters. Here, the well know deferred merge embedding (DME) [3, 11, 22, 69] algorithm is used to generate a forest of m unbuffered zero-skew steiner trees. This process is similar to the study in [42] for local tree network synthesis from a regional/global clock mesh topology, where m local tree networks are generated with BST/DME [15] for a regional clock distribution of m clock mesh grids. In this work, all the sub-network trees generated are ideally zero-skew local individual trees. The clock skew among different sub-network trees is minimized together with the ROA later in the fourth step.

4.3.3 Brick-based ROA Generation

In addition to low skew clock generation, the ROA contributes to part of the clock network for globally delivering the clock signals to local sub-network trees. In this dissertation, for simplicity, the tapping points and the sub-network tree roots form

one-to-one pairs for clock delivery. This is a two stage process: (1) Brick-based ROA generation with m rings (There are m sub-network trees); (2) The optimization of the roots to tapping points pairing. In order to generate a brick-based ROA with a specified number of rings, it should be validated that a brick-based ROA can be formed with an arbitrary number of rings. As shown in Fig. 4.4(a), by adding ROA-Bricks to an existing brick-based ROA topology along the edge direction, a larger topology can be obtained and the minimum increment of the ring numbers is 2. Thus, a brick-based ROA, which contains an even number (≥ 4) of rings, is always realizable. On the other hand consider Fig. 4.4(b), where a triangle brick-based ROA structure, which is composed of 3 ROA-Bricks, exists in an existing brick-based ROA topology. By adding 1 ring in the originally unoccupied location of ring 9, the topology becomes a new brick-based ROA, which is composed of 4 ROA-Bricks. This is to say, if a triangle structure exists in a brick-based ROA containing even number of rings, by adding 1 ring, the topology can be turned into brick-based ROA with an odd number of rings (≥ 9). There is a special case that the brick-based ROA composed of 7 rings can be realized by connecting 2 ROA-Bricks by just sharing only one ring between them. However, a brick-based ROA composed of only 5 rings is not realizable. To this end, it is proved that a brick-based ROA with m rings, where $m \geq 6$, always exists. In other words, the number of rings in a brick-based ROA is as flexible as that in a ring-based ROA.

After the brick-based ROA is generated, each tapping point is used as a local clock source which provides a clock signal to a single sub-network tree. The Hungarian Algorithm [48] is used to assign each tapping point to a sub-network tree root in order to minimize the total stub wirelength.



(a) ROA composed of even number of rings

(b) ROA composed of odd number of rings

Figure 4.4: ROA composed of arbitrary integer number of rings.

4.3.4 Tapping Point Location Optimization for Skew Reduction

In the tapping point and sub-network tree roots pairing optimization process, the skew among all the sub-network trees should be considered. Clock skew may exist in the current clock distribution network: (1) There may be clock skew among the sub-network trees; (2) The root to tapping point connections may also introduce clock skew. In next step, clock skew is optimized for each sub-network tree individually. In order to minimize the clock skew among sub-network trees, many methods can be applied, such as (1) Performing wire snaking on the connections between sub-network tree root and tapping point; (2) Adding additional capacitance. Both of these methods are inefficient in introducing additional capacitance and degrading the performance of the design. On the other hand, as shown in Eq. 2.3, the phase delays are equally distributed along the transmission line in the signal propagation direction (i.e. effects of large capacitive imbalance can be considered here to recompute

phase values, if necessary). Thus, the phase delay of a point along the transmission line from the tapping point is proportional to the distance between these two points, which is very efficient for skew reduction. Thus, in this design, the location of some tapping points are optimized in order to provide extra delays (both positive and negative) for some sub-network trees for clock skew reduction. The process of clock skew reduction is shown in Algorithm 2. Since the signal rotation direction on the ROA is critical to the optimization, HPSICE simulation is performed as part of the optimization. In the initial HSPICE simulation (Line 1), the following terms are measured or detected: (1) Determine the signal rotation direction to be clockwise (CW) or counter clockwise (CCW) (Line 2); (2) The frequency of the oscillation signal, f (Line 3); (3) The initial skews among the sub-network trees $s_1, s_2, \cdots, s_m$ (Line 4). After finding the median skew s_Δ among all the skew numbers, all the skew numbers are rewritten as a relative skew to the median skew $s_{r1}, s_{r2}, \cdots, s_{rm}$ and the median skew becomes 0 (Line 7). The tapping point tuning step *step* is defined based on the largest relative skew θ among all the sub-network trees:

$$\begin{aligned} \theta &= max(|s_{r1}|, |s_{r2}|, \cdots, |s_{rm}|) \\ step &= \frac{\theta}{kT} p_{ring} \end{aligned} \tag{4.3}$$

Here, θ is defined as the largest relative skew among all the sub-network trees. A user defined parameter k, which is named as resolution controller, is used control the resolution of the skew reduction process. The tapping point tuning step *step* is determined by both θ and k. In Algorithm 2, the resolution controller is chosen as 3 (Line 8). Parameter p_{ring} is the perimeter of the ring and T is the cycle time of the oscillation signal. In this design, the total delay tuning range is $\pm\theta$. The new tapping point location information is written into an HSPICE netlist and the final simulation is performed.

Algorithm 2 Tapping point location optimization

```
Perform HSPICE simulation.
Determine rotation direction: CCW or CW
Obtain operating frequency: f
Obtain sub-network tree skews: s_1, s_2, ···, s_m
Sort the skews: s'_1, s'_2, ···, s'_m
Find the median skew: s_Δ = s_med = s'_⌊m/2⌋
Obtain the skews relative to s_med: s_r1, s_r2, ···, s_rm
step = θ/(3T)p_ring
for i=1 to n do
    if s_ri ≤ −(2/3)θ then
        tp_move_i = 2
    else if (−2/3)θ < s_ri ≤ −(1/3)θ then
        tp_move_i = 1
    else if (1/3)θ < s_ri ≤ (2/3)θ then
        tp_move_i = −1
    else if s_ri > (2/3)θ then
        tp_move_i = −2
    else
        tp_move_i = 0
    end if
end for
if CCW then
    tp_move = −tp_move
end if
Regenerate HSPICE netlist
Rerun HSPICE sim
```

4.4 Experiment

HSPICE simulations are performed using a 90nm technology with the following two primary objects:

1. The verification simulations of the effectiveness of the synchronization scheme.

2. Brick-based ROA clock generation and distribution networks are designed for ISPD 10 benchmark circuits #03 − #08 in order to show the effectiveness of the application on larger scale circuit design.

4.4.1 Signal Rotation Direction Synchronization

In order to verify the synchronization scheme presented in Section 4.2, three sets of simulations are performed by applying the synchronization scheme to an ROA-Brick. In these three simulations, `R2`, `R3` and `R1` are selected as the master ring, respectively. The simulation results shown in Fig. 4.5 show the uniform signal rotation direction when either `R2` or `R3` is selected as the master ring. The waveform on top shows that when `R3` is selected as the master ring, all the rings in the ROA-Brick are rotating in the clockwise direction. The second waveform shows that when `R2` is selected as the master ring, all the rings in the ROA-Brick are rotating in the counter clockwise direction. To verify the signal rotation direction controllability of the master ring, this experiment is performed repeatedly by continuously changing the capacitive loading on the other three rings from $10fF$—$2pF$. The waveforms shown in Fig. 4.5 is one of the simulations when the capacitive loading on the other three rings is $1pF$. All the simulation results confirm the robustness of the signal rotation direction controllability of the synchronization scheme.

When `R1` is used as the master ring for synchronization, the uniform signal rotation direction remains undetermined as postulated in the theoretical analysis based around the illustration in Fig. 4.2(a). As shown in Fig. 4.6, the ROA-Brick is restarted three times and the signal rotation directions after each time the circuit reach synchronization is inconsistent: Clockwise, clockwise and counter clockwise. If the synchronization scheme proposed in this chapter is not applied, a phase detector circuit is necessary to observe the oscillation direction and restart synchronization as many times as necessary—elongating the start-up time—or trigger circuit functionality when correct directionality is detected.

Besides the signal rotation direction controllability, the synchronization scheme present in Section 4.2 also exhibits high efficiency. The simulation results are shown

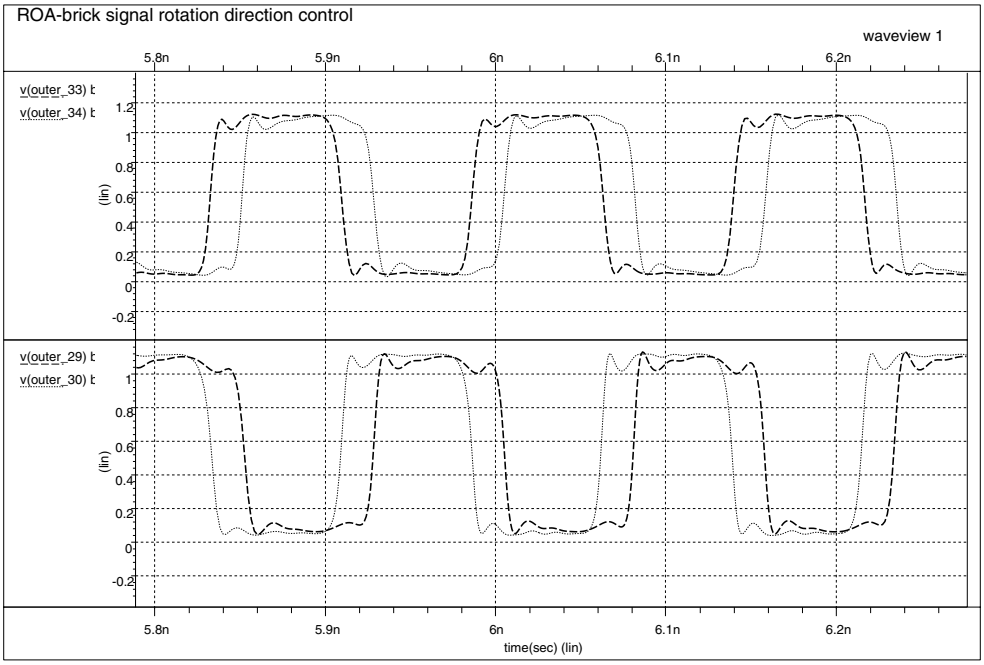


Figure 4.5: ROA-Brick signal rotation direction.

in Fig. 4.7, from top to bottom showing the synchronization results when `R1`, `R2` or `R3` are used as the master ring, respectively. As shown in the plot, the length of pre-synchronization period is $4ns$, during when only the master is powered up. When all the rings are powered up, different master rings selection costs different times to reach synchronization. When `R1` is selected as the master ring, it takes about $1.5ns$ after all the rings are powered up to reach synchronization. This is due to standing waves are formed on `R1` and `R4` during the pre-synchronization period. On the other hand, when `R2` or `R3` are used as the master ring, it takes less than $0.5ns$ to reach synchronization. This is because three out of four rings in the brick have formed a uniform signal rotation direction, which significantly accelerates the synchronization speed. More importantly, the pre-determined oscillation direction eliminates the need for repetitive start-ups and control circuitry, which substantially improves the synchronization speed (e.g. start-up time).

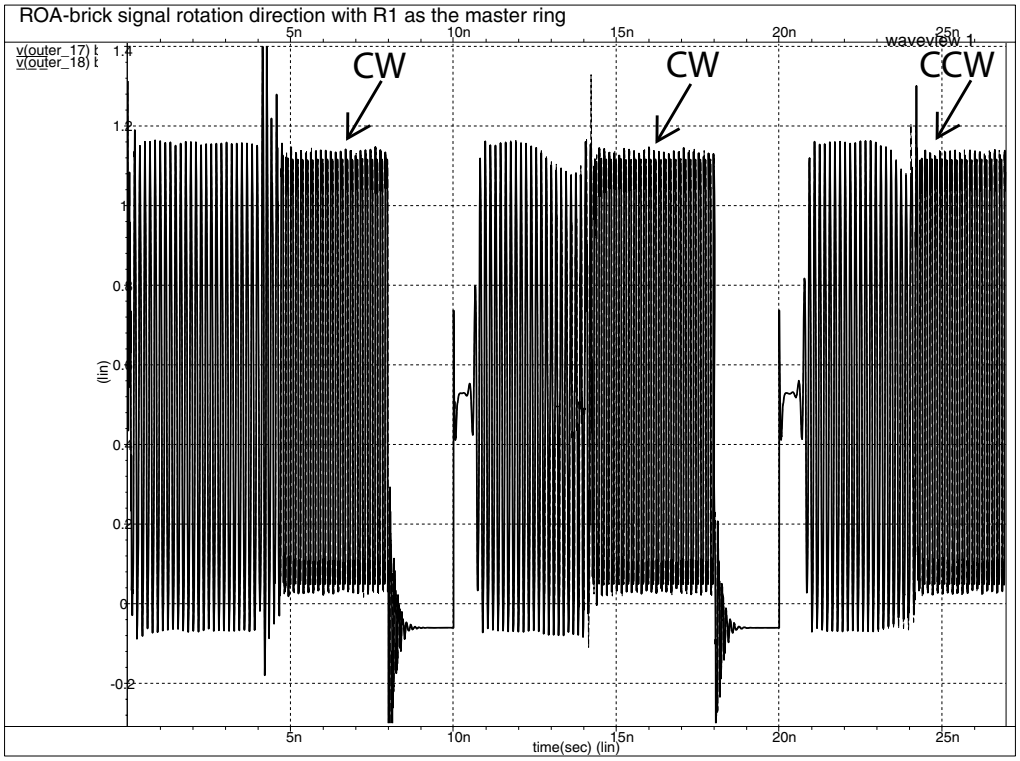


Figure 4.6: Random ROA-Brick rotation directions.

4.4.2 Brick-based ROA Clock Distribution Network Design

In this section, the application of the ROA-based clock network design on academically available clock research benchmarks is presented, which demonstrates the feasibility of using the brick-based ROA in multi-source clock distribution network design. The circuits #03 − #08 from ISPD 10 benchmarks have been utilized in the experiments.

Simulations

Fig. 4.8(a) shows the topology of the brick-based ROA designed for ISPD 10 benchmark circuit #07. In this design, 14 sub-network trees are generated with a high capacitance tolerance ratio $\tau = 0.3$. The ROA topology is composed of 14

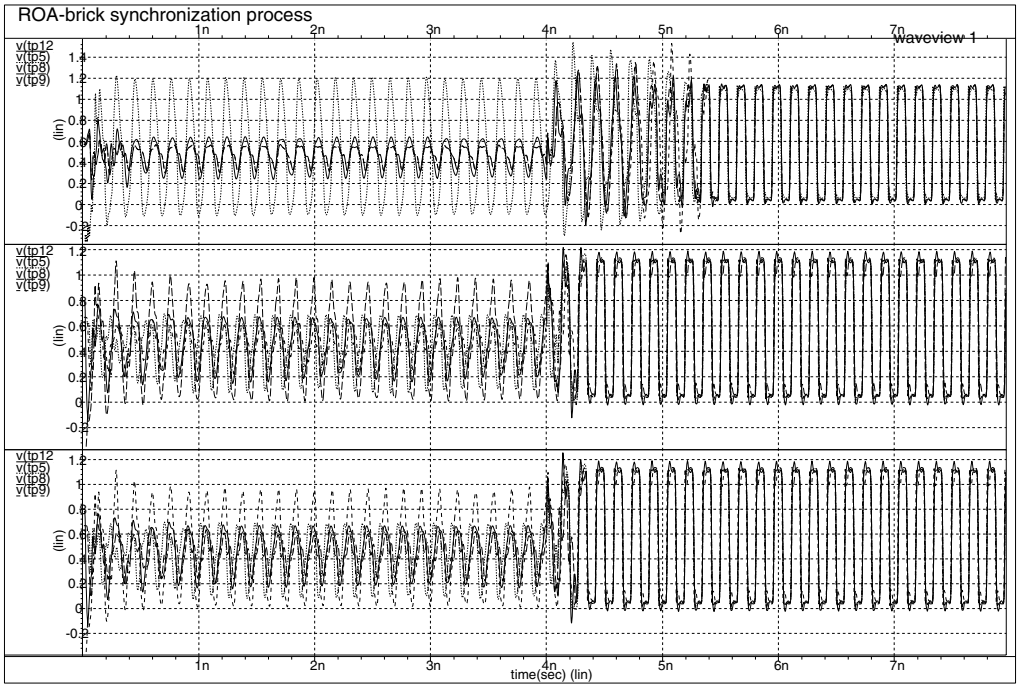
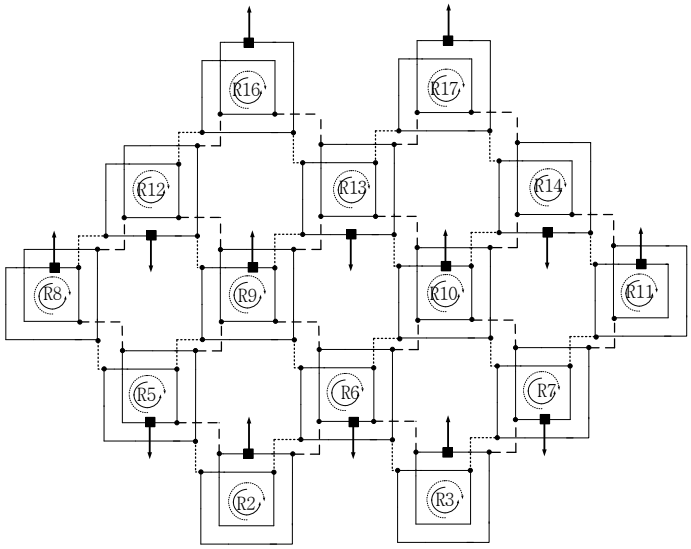


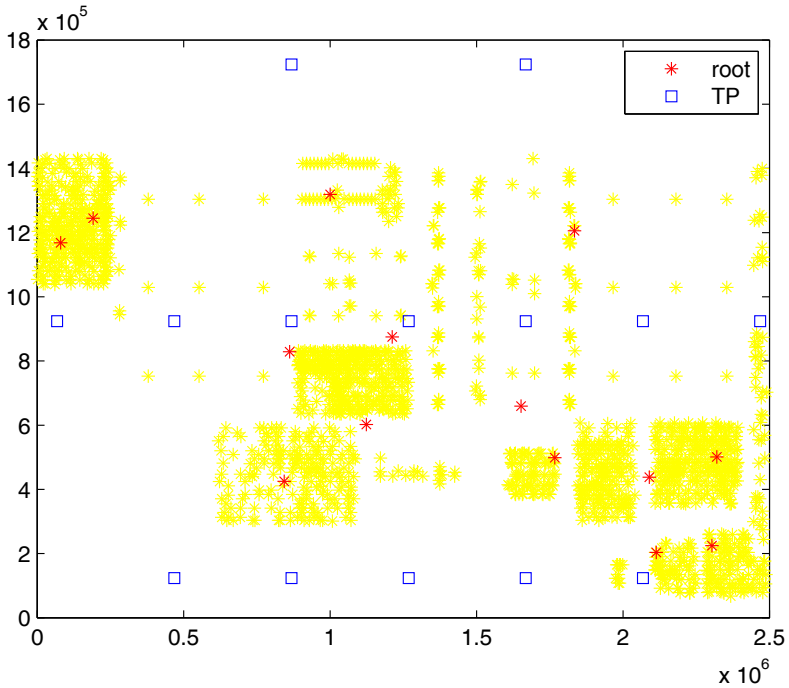
Figure 4.7: Synchronization process by different master ring selection.

rings in proposed ROA-Brick formations. Fig. 4.8(b) shows the distribution of the sub-network tree roots and the location of the tapping points after optimizations.

The HSPICE simulation result is shown in Fig. 4.9. In order to accelerate the process of synchronization among the rings, `R8` is started up $4ns$ prior to the other rings [65]. After all the rings start to oscillate, the resonant system takes around $3ns$ to reach synchronization. When the steady state is reached, all the rings in the brick-based ROA form a uniform ring rotation direction so as to have identifiable same phase point set. Fig. 4.9 shows the well aligned tapping point signals. The almost negligible phase differences among all tapping point signals (as large as $20.9ps$) are enabled by the tapping point location optimization for skew reduction. By optimizing the locations of the tapping points as explained in Section 4.3.4, there is a significant skew reduction (average 71.72%) of the clock tree network with only a small change



(a) Brick-based ROA topology designed for an ISPD 10 benchmark circuit #07.



(b) Subtree roots and tapping points locations.

Figure 4.8: Brick-based ROA for ISPD 10 benchmark circuit #07.

in the total stub wirelength (average -0.22%). This is because the phase changing of the clock signal on the transmission line of the ring is proportional to the distance from the tapping point, which is very efficient in clock skew tuning. The results of the optimization shown in Table 4.1 demonstrate this point. After the skew optimization, the average skew reduction is 71.72%, while the average total stub wirelength and power increase are -0.22% and -0.02%, respectively.

Interpretation of the Simulation Results

The simulation indicates the following aspects:

1. The tapping point location optimization in Sec. 4.3.4 guarantees the maximum skew falls between $[-\theta/k, \theta/k]$, where θ is the largest relative skew among all the sub-network trees and k is the user defined resolution controller. After the tapping point location optimization process, the final maximum skew is not closely related to the initial maximum skew of the clock network. By selecting a larger k, which corresponds to a smaller tapping point tuning step *step*, smaller skew can be obtained.

2. By optimizing the tapping point locations on the ROA to reduce the skew among clock trees, the skew reduction is significant while the impact on the clock network is negligible. This is because the ROA possesses more efficient delay tuning capability than the clock tree. In the ROA system, the delay is proportional to the clock signal traveling distance on the transmission line from the original tapping point. Thus, the stub wirelength and skew optimizations can be performed individually, which greatly simplifies the optimization process.

3. The cluster capacitance tolerance ratio τ defined in Sec. 4.3.1 plays an important role in the design. Under the control of the capacitance tolerance ratio, the

loading capacitance for all the tapping points are relatively balanced. The balanced capacitive loading helps the resonant system to reach synchronization efficiently. The capacitance tolerance ratios for the experimental benchmarks designs are chosen between $0.2 \sim 0.3$ based on experimental results. With the capacitance tolerance ratio chosen in this range, the resonant system can efficiently reach a synchronization state.

4. After the optimization, the total wirelength and power can be slightly greater or smaller than before. This is because after the optimization of the tapping point locations, the connections between the sub-network tree root and the tapping point maybe longer or shorter than before. However, as the results shown in Table 4.1 that the changes in the total wirelength and the power of the circuits are very small.

5. Another advantage of the brick-based ROA clock distribution network is that the ROA isolates the optimization of the sub-network trees. Since each sub-network tree is connected to its tapping point. The skew optimization process for a sub-network tree does not affect the performance of the other sub-network trees.

Overhead Comparisons

In order to obtain the knowledge of the signal rotation direction of the traditional ROA, a direction detector used to facilitate the design [58], which is similar to a phase detector (PD) in a delay locked loop (DLL) designed for high frequency usage [4, 35]. In order to suit for the design of the traditional ROA, the direction detector is equipped with the function of reset the synchronization process when it detects the signal direction is undesirable. To this end, the area overhead of the

Table 4.1: Results for ROA placement optimizations on ISPD 10 benchmarks

TB	Ring No.	Init. WL(μm)	Init. pwr(W)	Init. Skew(ps)	Opt. Skew(ps)	WL Incr.(%)	Pwr Incr.(%)	Skew Imp.(%)
#03	10	43078	1.66	36.8	5.72	0.46	0.10	84.46
#04	14	120437	2.37	43.1	7.45	0.06	0.00	82.71
#05	7	53069	1.09	32.6	5.86	-1.21	0.04	82.02
#06	10	54880	1.64	24.5	9.56	-0.24	-0.02	60.98
#07	14	92130	2.37	27.8	10.08	-0.51	-0.01	63.74
#08	9	53826	1.46	19.9	8.67	0.12	-0.22	56.43
Avg	–	–	–	–	–	-0.22	-0.02	71.72

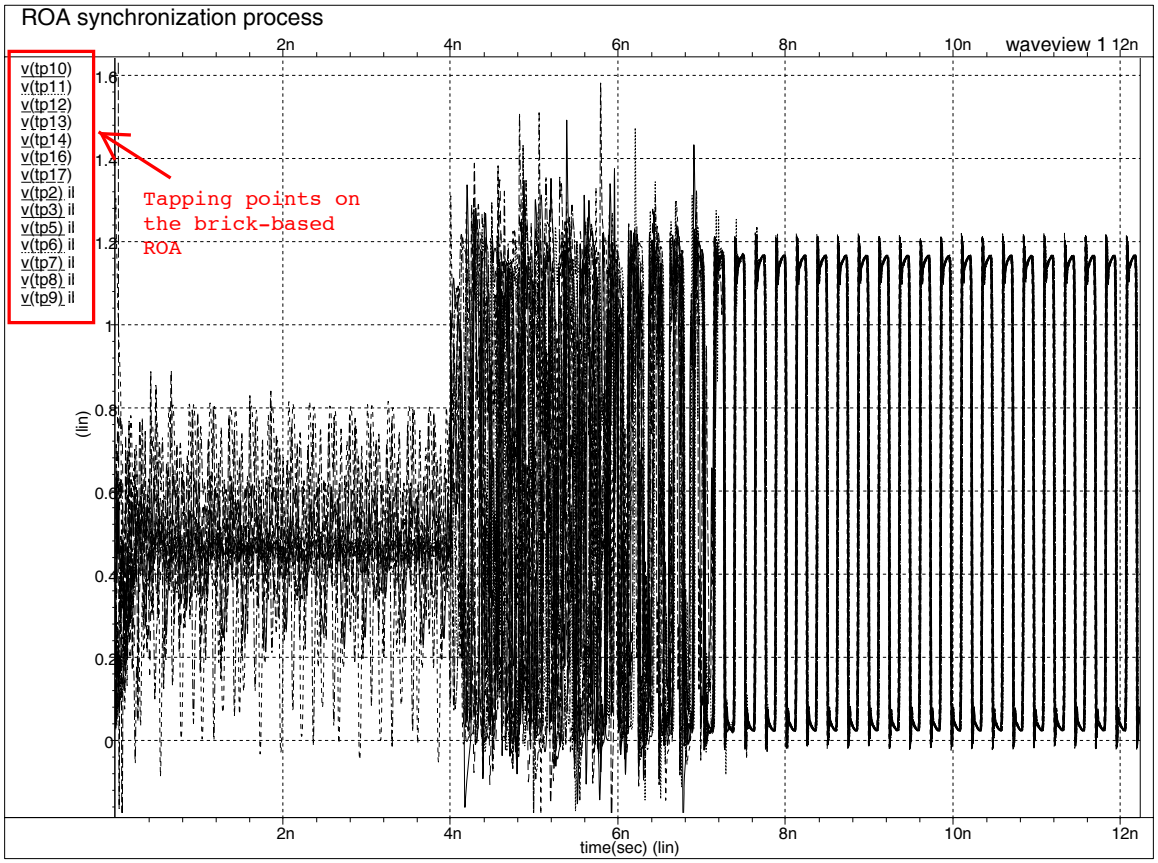


Figure 4.9: Brick-based ROA after synchronization.

direction detector tailored for a $6.6GHz$ ROA is about $2900\mu m^2$. Similarly, using varactors to control the oscillation direction is also costly. There are three typical types of varactors, namely the diode based varactor, inversion-mode MOS varactor and accumulation-mode MOS varactor. Previous research [1] shows that accumulation-mode MOS varactors outperform the other two type of varactors in term of power dissipation, phase noise and area, which has been widely used in PLL design [58, 78]. Approximating the loading capacitance for each tapping point to be approximately $1pF$, in order to direct the ring to rotate in a desired rotation direction, the maximum capacitance provided by the varactor should at least be $Cmax = 1pF$. Suppose a ring loop (non-feasible ROA) is formed by 14 rings: In order to direct all the rings

in to a desired rotation direction, the total capacitance provided by the varactor should at least be $14pF$, which corresponds to an area overhead of $1156\mu m^2$. However, by utilizing the brick-based ROA topology, both of these two previously widely used methodologies can be removed from a low-skew ROA-based clock tree distribution network design. The limited feasible rotation directions of the brick-based ROA makes it possible to remove the direction detection block circuits since the relative positions of all the zero skew points are identifiable.

4.5 Conclusions

In this chapter, a brick-based ROA synchronization scheme is proposed, which effectively accelerate the ROA synchronization process and directs all the rings in the ROA to form a uniform signal rotation direction. Therefore, the same phase points on the brick-based ROA are easily determined. Based on this feature, low-skew brick-based ROA clock distribution networks are designed for ISPD 10 benchmarks in order show the feasibility of the topology. By optimizing the tapping point locations, the skew reduction is 71.72% down to under $10ps$ without significantly changing the power consumption and total wirelength.

5. Sparse-Rotary Oscillator Array (SROA) Design for Power and Skew Reduction

This chapter presents a unique rotary oscillator array (ROA) topologythe sparse-ROA. The SROA eliminates the need for redundant rings in a typical, mesh-like rotary topology optimizing the global distribution network of the resonant clocking technology. To this end, a design methodology is proposed for SROA construction based on the distribution of the synchronous components. The methodology eliminates the redundant rings of the ROA and reduces the tapping wirelength, which leads to a power saving of 32.1%. Furthermore, a skew control function is implemented into the SROA design methodology as part of the optimization of the connections among tapping points and subtree roots, which leads to a clock skew reduction of 47.1% compared to a square-shaped ROA network design, which is verified through HSPICE.

5.1 Introduction

A rotary oscillator array (ROA) is composed of identical RTWO rings structuring the global distribution network of a mesh-like, square-shaped ROA-based resonant clocking as presented in Fig. 5.1. The ROA-based clock network design has been well studied in recent years. The studies in [42, 71, 80] addressed the optimization of the local tree by generating sub-network clock trees to the tapping points of the ROA. The study in [33] addressed the optimization of the regional clock tree in proposing custom-shaped rings. In these previous studies in literature, no attempts have been made to improve the quality of the resonant clocking through optimizing the mesh-like, square-shaped ROA, which is the global distribution network. Global network

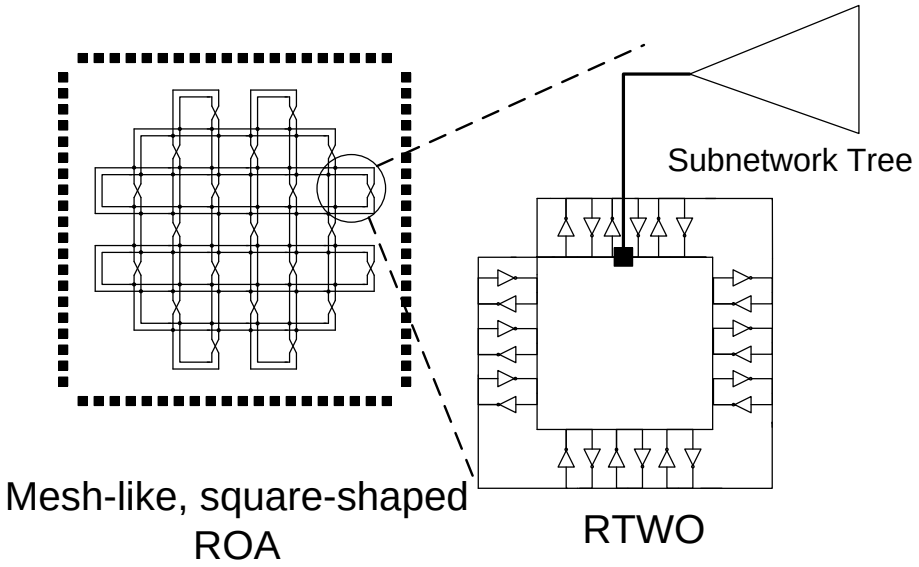


Figure 5.1: ROA-based clock network.

optimization of the ROA is pivotal as it dictates the quality of regional and local networks. Through global-network optimization of the ROA, the already low-power and low-skew characteristics of the rotary clocking technology are further improved.

In this chapter, a methodology for generating a novel ROA topology, which is aware of the distribution of the synchronous components, is proposed. The methodology incrementally places a full-mesh ROA closer to register-dense locations. Furthermore, the proposed ROA is constructed by eliminating the redundant rings from the full-mesh ROA structure according to the number and distribution of the subnetwork trees. As such, this custom-shaped ROA is called the sparse-ROA (SROA). The SROA saves the total routing wirelength both in terms of the ring transmission lines (global network) and tapping wire (local network). By eliminating the redundant rings from the square-shaped ROA, the power consumption of the resonant system is reduced significantly. Furthermore, the optimization of the connections among tapping points and subtree roots both saves the tapping wirelength and decreases

the clock skew among all the sub-network trees. The ROA startup scheme in [65] is applied in order to directs the oscillation signal on the SROA to form a uniform rotation direction quickly and effectively. In conjunction with the previous literature [33, 42, 71, 80], the SROA design methodology completes the rotary-based clock network design flow by establishing the global network design step.

This chapter is organized into the following sections. Definitions and preliminaries are provided in Section 5.2. The methodology for generating the SROA is proposed in Section 5.3. The simulations are presented in Section 5.4. Conclusions are presented in Section 5.5.

5.2 Definitions and Preliminaries

The sparse-ROA (SROA) is composed of the ROA-Bricks. Unlike a traditional ROA, the SROA does not need to maintain a square structure as long as its integrity is preserved for rotational lock between the (regional) rings of the ROA. Consequently, the SROA design methodology is synthesizing ROA-Bricks that are sparse but contiguous for integrity.

5.2.1 Definitions and Construction Rules of the SROA

A sample SROA topology composed of rings {`R1` `R2`, `R3`, `R4`, `R5`, `R6`, `R7`, `R8`, `R9`, `R10`, `R11`} is shown in Fig. 5.2. Some definitions on this SROA are given below:

1. **Brick-connection**: Two ROA-Bricks form a brick connection if they share two rings, such as ROA-Brick {`R1`, `R2`, `R5`, `R3`} and ROA-Brick {`R2`, `R4`, `R7`, `R5`}.

2. **Brick path**: A brick path is a sequence of ROA-Bricks such that every two successive ROA-Bricks form a brick-connection. For instance, Fig. 5.2 shows a brick path between ROA-Brick {`R5`, `R7`, `R10`, `R8`} and {`R6`, `R8`, `R11`, `R9`} contains two brick-connections.

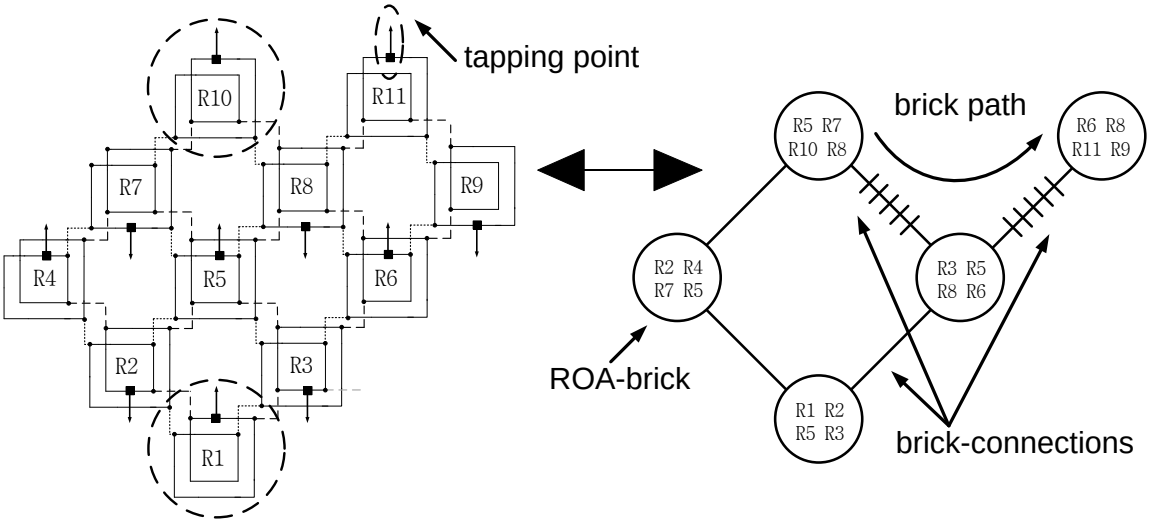


Figure 5.2: A sample topology of an SROA (brick component).

3. **Brick component**: A brick component is a group of ROA-Bricks, such that there always exists a brick path between each two ROA-Bricks. For instance, the topology in Fig. 5.2 is a brick component.

4. **Ring-connection**: Two ROA-Bricks form a ring connection if they share only one ring (and no more) between them. For instance, ROA-Brick {`R2`, `R4`, `R7`, `R5`} and ROA-Brick {`R3`, `R5`, `R8`, `R6`} form a ring-connection.

5. **Ring component**: Ring component is a group of ROA-Bricks, such that there is always a path formed by ring-connection(s) and/or brick-connection(s) between any two ROA-Bricks. As such, a brick component is always a ring component but a ring component is not always a brick component.

The brick-connection is used for SROA synchronization. If two ROA-Bricks form a brick-connection, the signal rotation direction on all the rings is synchronized. A ring connection, does not guarantee synchronization between ROA-Bricks. Based on this observation, the SROA topology is constructed with these rules:

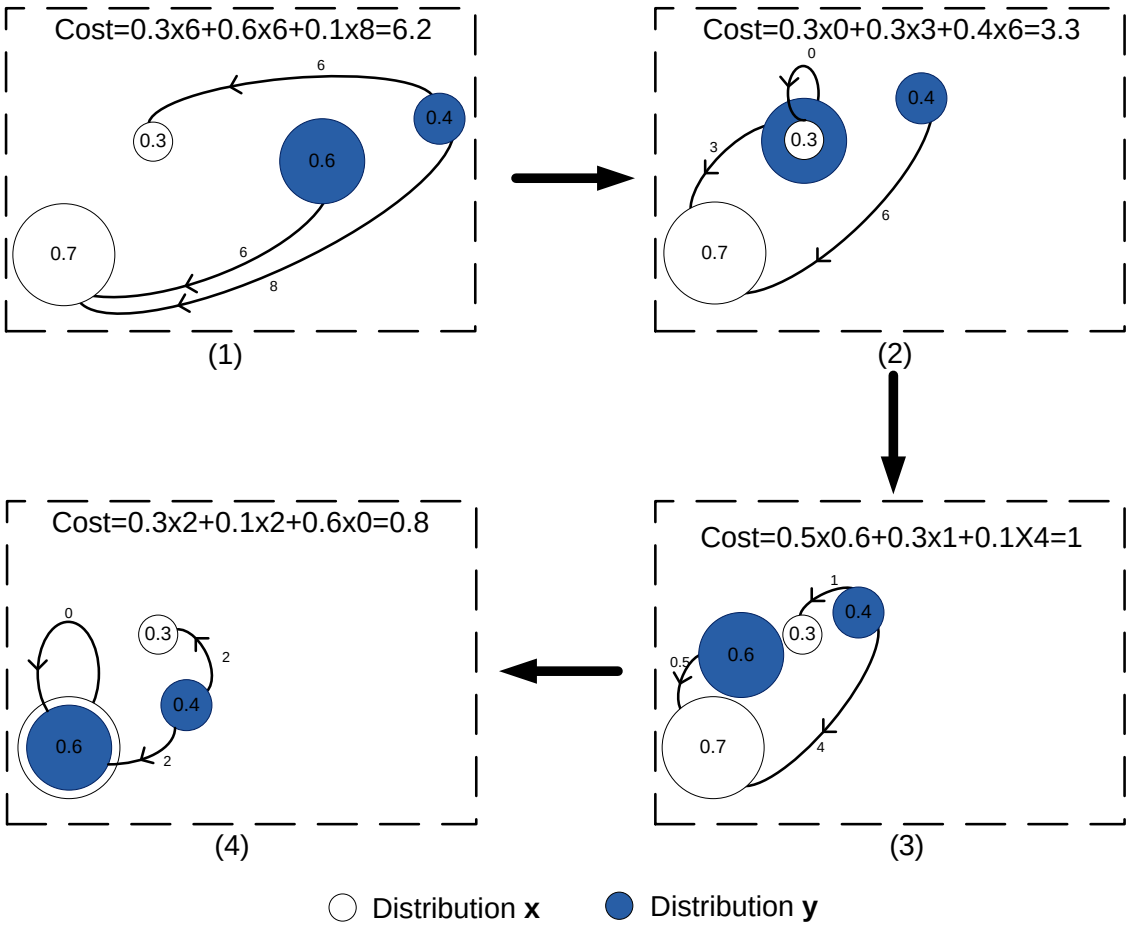


Figure 5.3: The operation process of $EMD_{\mathbf{t}}$

1. Each ring in the SROA belongs to at least one ROA-Brick.

2. The SROA should be a brick component.

Rule 1 indicates that every ring in the SROA should be contained in at least one ROA-Brick and there should be no isolated rings in the SROA. This rule is used to maintain the integrity of the SROA. Rule 2 implies: (1) There are no disjoint brick components: If there are more than one brick components, these brick components cannot be synchronized but will operate in different frequencies. (2) Only brick-connections rather than ring-connections exist in the SROA. Rule 2 is used to direct all

the rings in the SROA to form a uniform signal rotation direction. A ring-connection divides the SROA into two brick components. If these two brick components are about the same size and the signals on these components are rotating in the opposite directions, it is possible that neither brick component is able to overwritten the signal rotation direction on the other brick component through the ring-connection. Instead of reaching synchronization, the signal undesirably dies out on the ring which is connecting these two brick components.

5.2.2 Earth Mover's Distance Under Transformations

The earth mover's distance (EMD) is a distance measure between distributions, which is widely used in image retrieval and matching [56]. EMD under transformations ($EMD_{\mathbf{t}}$), which is a variant of EMD that moves one of the distributions, has the advantage of computing a transformation of one distribution in order to minimize the distance (measured by EMD) between the distributions [14].

The EMD is a distance measure between discrete, finite distributions $\mathbf{x}$ and $\mathbf{y}$:

$$\mathbf{x} = \{(x_1, w_1), (x_2, w_2), \cdots, (x_m, w_m)\} \tag{5.1}$$
$$\mathbf{y} = \{(y_1, u_1), (y_2, u_2), \cdots, (y_n, u_n)\}$$

where, the $\mathbf{x}$ distribution has a weight w_i at position $\mathbf{x_i}$ in $\mathbf{R}^2$, $i = 1, \cdots, m$ and $\mathbf{y}$ distribution has a weight u_j at position $\mathbf{y_j}$ in $\mathbf{R}^2$, $j = 1, \cdots, n$. The EMD is defined by the linear program:

$$
\begin{aligned}
\text{minimize} \quad & \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij}d_{ij}}{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij}} \\
\text{subject to} \quad & f_{ij} \geq 0 \quad i = 1 \ldots m, j = 1 \ldots n \\
& \sum_{j=1}^{n} f_{ij} \leq w_i \quad i = 1 \ldots m \\
& \sum_{i=1}^{m} f_{ij} \leq u_j \quad j = 1 \ldots n \\
& \sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij} = min(W, U)
\end{aligned}
\tag{5.2}
$$

In (5.2), d_{ij} is distance between $\mathbf{x_i}$ and $\mathbf{y_j}$. W and U are collective weights of the distributions: $W = \sum_{i=1}^{m} w_i$ and $U = \sum_{j=1}^{n} u_j$.

The earth mover's distance under transformation set $\mathbf{t}$ ($EMD_{\mathbf{t}}$) is defined as:

$$
EMD_{\mathbf{t}}(\mathbf{x}, \mathbf{y}) = min_{t \in \mathbf{t}} EMD(\mathbf{x}, t(\mathbf{y})) \tag{5.3}
$$

Equation (5.3) indicates that a transformation is performed on one distribution to move the whole distribution incrementally towards the other distribution in order to minimize the EMD. Illustration of the operation of $EMD_{\mathbf{t}}$ is shown in Fig. 5.3. The $EMD_{\mathbf{t}}$ is performed in two steps: 1) Assignment; 2) Transformation. As shown in Fig. 5.3, an assignment is performed on the two distributions $\mathbf{x}$ and $\mathbf{y}$ for the minimum total weighted distance (cost), which is 6.2. Then a transformation is performed on $\mathbf{y}$, which moves $\mathbf{y}$ towards $\mathbf{x}$ in order to reduce the cost. These two steps complete one iteration. The distributions in Fig. 5.3 require three iterations to converge to a cost of 0.8. $EMD_{\mathbf{t}}$ calculation converges very quickly. Note that, in each iteration, the assignment between two distributions may change and the transformation may also be different. The process is terminated when the assignment does not change anymore or the current cost is tolerable.

5.3 SROA Design Methodology

The design of SROA necessitates changes to local distribution networks (i.e.sub-network trees) as the sparsity of the ROA is optimized based on the register placement. To this end, a bottom-up distribution network design procedure is proposed. First, local trees are constructed through register clustering. Second, a tapping point set is generated for optimizing the total wirelength and clock skew. Third, the SROA is generated from the full-mesh ROA based on the ring selection of the tapping point set, as well as maintaining the integrity of the SROA.

5.3.1 Register Clustering

In the beginning, all the registers are clustered using a modified deferred merge embedding (DME) [3, 11, 22, 69] solution to generate a forest of unbuffered steiner tree based networks. This process is similar to the the study in [42], where local tree networks are generated with BST/DME [15]. In this work, an upper bound on the total capacitance in a register cluster is established based on the target frequency of the RTWO before generating a forest of steiner trees of balanced total capacitance. Based on this capacitive limit, the subtrees are generated.

5.3.2 Tapping Point Set Generation

The process of tapping point set generation starts with a large enough square-shaped ROA with the number of rings larger than the number of subtree roots provided by register clustering. Each ring in the ROA provides one tapping point as a clock source. Then a modified $EMD_{\mathbf{t}}$ method is used to move the square-shaped ROA around in order to optimize the one-to-one matching between the tapping points and the subtree roots for minimum total wirelength while balancing the delay among the sub-network trees. The tapping point set generation is shown in Algorithm 3

where the inputs are the set of the square-shaped ROA tapping points S and the set of the subtree roots D obtained from register clustering.

$$S = \{\mathbf{s_1}, \mathbf{s_2}, \cdots, \mathbf{s_m}\} \tag{5.4}$$
$$D = \{\mathbf{d_1}, \mathbf{d_2}, \cdots, \mathbf{d_n}\}$$

For any $\mathbf{t} \in \mathbf{R}^2$, the transformation of set S is:

$$S + \mathbf{t} = \{\mathbf{s_1} + \mathbf{t}, \mathbf{s_2} + \mathbf{t}, \cdots, \mathbf{s_m} + \mathbf{t}\} \tag{5.5}$$

The problem is described as shown in Table 5.1. $F = (f_{ij}) \in \mathbf{R^{m \times n}}$, with $f_{ij} = 1$ indicating a one-to-one matching from $\mathbf{s_i} + \mathbf{t}$ to $\mathbf{d_j}$. In order to keep the skew under control and avoiding the appearance of extremely large skews in individual sub-network trees, instead of using total wirelength, the total delay is set as the optimization object of this function. As shown in Table 5.1, the function COST is to calculate the delay from $\mathbf{s_i} + \mathbf{t}$ to a single register $\mathbf{R_{jk}}$ in $\mathbf{d_j}$. Under the Elmore model the delay between node $\mathbf{s_i} + \mathbf{t}$ and $\mathbf{R_{jk}}$ is given by:

$$\begin{aligned} & t(\mathbf{s_i} + \mathbf{t}, R_{jk}) = \\ & \mid \mathbf{s_i} + \mathbf{t} - \mathbf{d_j} \mid \cdot r(\frac{\mid \mathbf{s_i} + \mathbf{t} - \mathbf{d_j} \mid \cdot c}{2} + Cap(\mathbf{d_j})) \\ & + \sum_{\mathbf{e_w} \in Path(\mathbf{d_j}, \mathbf{R_{jk}})} \mid \mathbf{e_w} \mid \cdot r(\frac{\mid \mathbf{e_w} \mid \cdot c}{2} + Cap(\mathbf{w})). \end{aligned} \tag{5.6}$$

Here, r and c denotes the unit length wire resistance and capacitance, respectively. $Cap(\mathbf{w})$ denotes the total cap seen from point $\mathbf{w}$. Since the placement of the registers

Table 5.1: Tapping point generation problem.

Minimize the total matching cost under transformation.
min $COST(F, S, D, \mathbf{t}) =$ $\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij} \times (\mid \mathbf{s_i} + \mathbf{t} - \mathbf{d_j} \mid \cdot r \cdot Cap(\mathbf{d_j}) + const_j)$ s.t. $f_{ij} = 0 \ or \ 1, \ \ 1 \leq i \leq m, \ 1 \leq j \leq n$ $\sum_{j=1}^{n} f_{ij} \ \leq \ 1, \ \ 1 \leq i \leq m$ $\sum_{i=1}^{m} f_{ij} \ = \ 1, \ \ 1 \leq j \leq n$ $\mathbf{t} \in \mathbf{R^2}$.

in each sub-network tree are fixed before matching the tapping points to the subtree roots, the second part on right in (5.6) is a constant. Furthermore, since $\frac{|\mathbf{s_i}+\mathbf{t}-\mathbf{d_j}|\cdot c}{2} \ll Cap(\mathbf{d_j})$, (5.6) can be rewritten as:

$$t(\mathbf{s_i} + \mathbf{t}, R_{jk}) \approx \mid \mathbf{s_i} + \mathbf{t} - \mathbf{d_j} \mid \cdot r \cdot Cap(\mathbf{d_j}) + const_j. \tag{5.7}$$

Thus, the function COST, which is the sum of the total delay from set S to registers in set D under matching F and transformation $\mathbf{t}$, is as a weighted total wirelength from tapping points to subtree roots. $\mid \mathbf{s_i} + \mathbf{t} - \mathbf{d_j} \mid$ is the Manhattan distance from $\mathbf{s_i} + \mathbf{t}$ to $\mathbf{d_j}$. The total capacitance on each subtree $Cap(\mathbf{d_j})$ controls the skew among each subtrees. Experimental results show that COST provides similar total stub wirelength compared to optimize total stub wirelength, but with bounded skew.

The process is divided into two steps: (1) **best_matching**: Given the transformation $\mathbf{t}$, the problem becomes an integer programming problem, which is hard to solve. However, for this problem it can be mathematically proved that by relaxing the constraints $f_{ij} = 0 \ or \ 1$ to $0 \leq f_{ij} \leq 1$, two problems have the same solution. Thus, by giving the transformation vector $\mathbf{t}$, linear programming can be applied to find the best matching; (2) **best_move**: Keeping the matching, the best position of the square-shaped ROA is obtained by moving it around, which leads to a minimum

Algorithm 3 Tapping Point Set Generation

Input: Tapping point set S and subtree root set D

Output: Optimized tapping point set S_{opt}, optimal matching record F_{opt}, the best transformation vector $\mathbf{t_{opt}}$

1: Initialize $cost = \infty$, $cost_{new} = 0$;
2: **while** $\mid cost - cost_{new} \mid > \Delta$ **do**
3: $[F, cost_{new}] = best\_matching(S, D)$;
4: $\mathbf{t} = best\_move(S, D, F)$;
5: $cost = cost_{new}$
6: $cost_{new} = COST(F, S, D, \mathbf{t})$
7: $S = S + \mathbf{t}$;
8: **end while**
9: $S_{opt} = S$, $F_{opt} = F$, $\mathbf{t_{opt}} = \mathbf{t}$;

$COST$. In this problem, Manhattan distance is used to facilitate the calculation of the minimum total delay. The best position of the square-shaped ROA is the one with the minimum weighted total stub wirelength, the total capacitance of each subtree assists to balance the skew among all the subtrees.

The whole process is done in a number of iterations until convergence. This is because after doing the **best_move**, there may exist a better matching between $S + \mathbf{t}$ and D. The iteration ends when the improvement of $COST$ between two iterations is less than a tolerance Δ. The tapping point set contributing to the best matching is recorded as S_{opt}, the best matching result is F_{opt} and the transformation is $\mathbf{t_{opt}}$.

5.3.3 SROA Generation

The tapping point set S_{opt} generated from tapping point set generation may not be qualified to form an SROA for integrity problem. Thus, the object of SROA Generation is to exchange a portion of the rings in the tapping point set S_{opt} with the rest of the rings in S in order to satisfy the SROA integrity requirements with minimum total distance increasing. This process is illustrated in Algorithm 4. The

inputs are best matching F_{opt} and the best tapping point set S_{opt} from Algorithm 3. First, the largest ring component $RC_{largest}$ is investigated among S_{opt} located rings. All the ring connections in the $RC_{largest}$ need to be fixed in order to turn the $RC_{largest}$ into a brick component. For instance, in Fig. 5.2, ROA-Bricks {R2, R4, R7, R5} and {R3, R5, R8, R6} only have a ring-connection between them if R1 and R10 do not exist. However, by adding R1 or R10, a brick path is formed. After getting $RC_{largest}$, the next step is to find all the k possible ring-connection fixing schemes. In each iteration, the $RC_{largest}$ is fixed and turns into a brick component $BC(i)$. The isolated ring set under this fixing $R_{iso}(i)$ is generated, which is:

$$R_{iso}(i) = S_{opt} \setminus BC(i). \tag{5.8}$$

If the ring number of $BC(i)$ is smaller than that of S_{opt} (one ring corresponding to one tapping point), the isolated rings are added back to $BC(i)$ from the one with minimum cost. In order to maintain the integrity of $BC(i)$, the isolated ring is added back by a shortest brick path. The cost is expressed as:

$$cost(i) = BC_{new}(i) \backslash BC(i) \backslash R_{iso}(i) \tag{5.9}$$

where $BC_{new}(i)$ is the brick component after adding back an isolated ring. Since more than one ring may be added back to $BC(i)$ at once, the cost of adding back an isolated ring is evaluated based on both the increase of non-S_{opt} located rings and the decrease of isolated rings. After adding back one isolated ring, the $BC(i)$, $R_{iso}(i)$ are updated. The process terminates when the ring number of $BC(i)$ reaches the tapping point number of S_{opt}. Then, $BC(i)$ together with D are sent to the tapping point set generation for final optimization in case the matching and/or the location of tapping points needs to be updated. $S_{cand}(i)$ and $F_{cand}(i)$ are the best tapping point

Algorithm 4 SROA Generation

Input: The best matching F_{opt}, tapping point set S_{opt} and subtree root set D
Output: The tapping points of SROA S_{sroa}, the SROA matching F_{sroa}

```
Find the largest ring component: RC_largest.
Find all the possible k ring component fixing schemes
for each ring component fixing scheme i do
    Fix RC_largest ⇒ BC(i)
    Generate R_iso(i) based on BC(i);
    while Sizeof(BC(i)) ≤ Sizeof(S_opt) do
        Add back the minimum cost isolated tapping point;
        Update BC(i);
        Update R_iso(i);
    end while
    [S_cand(i), F_cand(i), t(i)] =
    Tapping_Point_Set_Generation(BC(i), D);
    cost_cand(i) = COST(F_cand(i), S_cand(i), D, t(i));
end for
cost_cand(j) = min(cost_cand);
S_sroa = S_cand(j), F_sroa = F_cand(j);
```

set and best matching under each ring-connection fixing scheme, respectively. The cost $cost_{cand}(i)$ under $S_{cand}(i)$ and $F_{cand}(i)$ is calculated. The index j of minimum cost among $cost_{cand}(i)$ is marked. The tapping point set cost $S_{cand}(j)$ is selected to be the tapping point set S_{sroa} of SROA. The matching $F_{cand}(j)$ of $S_{cand}(j)$ is selected to be the matching F_{sroa} of SROA.

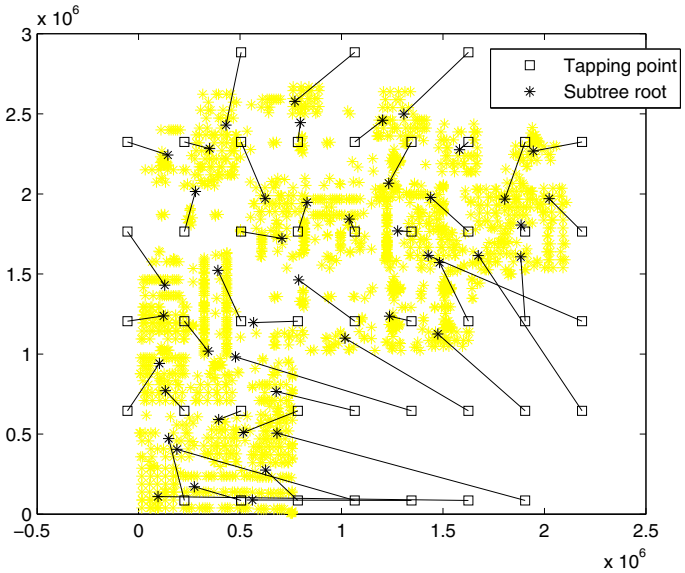
The topology S_{sroa} satisfies the SROA construction rules that are introduced in Section 5.2.1. First, every ring in the BC is contained in at least one ROA-Brick. The isolated rings are also added back to BC by brick path. Second, under each fixing scheme, the isolated rings are always added back to the same BC. Thus, this operation guarantees that the SROA is composed of only one brick component. Furthermore, the SROA generation process also guarantees that there is no ring-connection in SROA since a BC is always maintained.

5.4 Simulations

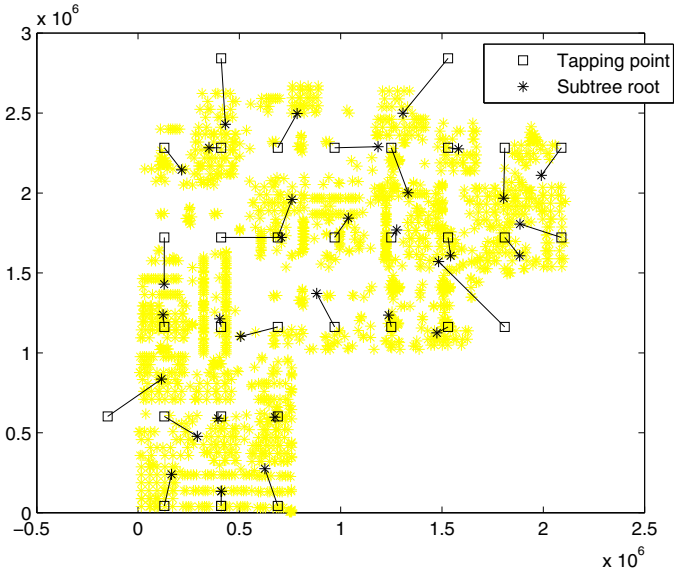
The experiments are performed on ISPD 10 benchmark circuit #03 – #08. The SROA structure generated for circuit #04 implemented in 90nm technology is shown in detail as an example. The operating frequency is arbitrarily picked between $6 - 7GHz$. Fig. 5.4 shows the matching of the tapping points on the regional rings of the global SROA network and the local subtree roots under two conditions: square-shaped ROA and SROA. It is shown that the tapping point locations of the SROA provides a better match with the sub-network tree roots in terms of total stub wirelength and clock skew.

For ISPD 10 benchmark circuit #04, 32 steiner trees are created, one of which is shown in Fig. 5.5. Using the methodology introduced in Section 5.3, not only the total tapping wire has reduced but the length of each tapping point to subtree root connection is also optimized based on the subtree total capacitance. The optimization of the tapping point to subtree root connections potentially (but not with certainty as the minimum insertion delay path can be shortened, as well the maximum insertion delay path) minimize the clock skew seen at individual sinks. After computing the structure of the clock tree network, the physical connections are transferred into a netlist and simulated using HSPICE. The power up scheme in [65] is used to facilitate the SROA synchronization process. The simulation result is shown in Fig. 5.6. The start-up jitter is due to the self-oscillating and phase-locking property of rotary clocking and is observed for all methods of directionality adjustment.

The SROA is tested on ISPD 10 benchmark, circuit #03 – #08. The complete testing results are shown in Table 5.2. The SROA leads to an average mesh wirelength saving of 35.3% and an average tapping wirelength saving of 26.4%. The maximum skew has all been reduced in the SROA topology by an average of 47.1%. The average global skew of SROA-based network is within $13.3ps$, which is very competitive



(a) Square-shaped ROA local subtree root connections to regional rings



(b) SROA local subtree root connections to regional rings

Figure 5.4: SROA-based network for ISPD 10 benchmark circuit #04.

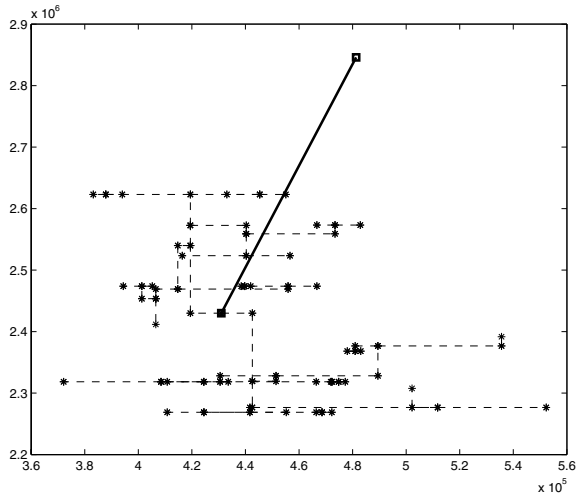


Figure 5.5: Local sub-network tree for one ring.

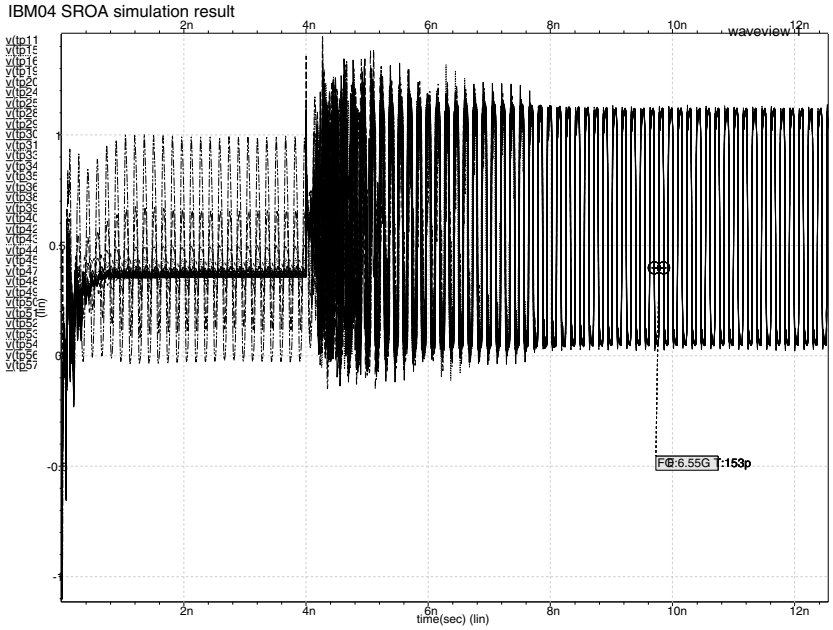


Figure 5.6: HSPICE Simulation of SROA designed for ISPD 10 benchmark #04.

with contemporary clocking techniques. The average power savings of 32.1%, on the already low-power ROA, are substantial. The combination of moving the global mesh with $EMD_{\mathbf{t}}$ *and* mesh ring reduction for SROA drives the skew reduction. The power

reduction is primarily due to the reduction in the mesh ring transmission line and tapping wirelength reduction.

5.5 Conclusions

In this chapter, the SROA topology is proposed for additional power savings, and as a by-product, generates reduced skew on the low-power rotary clocking technology. The shape of the SROA is constructed based on the distribution of the synchronous components. HSPICE simulation results show that the SROA-based clock network provides an average power savings of 32.1% and skew reduction of 47.1% compared to the square-shaped ROA.

Table 5.2: SROA testing results compared to a traditional ROA implementation that consumes ≈ 70% lower power than a clock tree [79]

TB	ROA ring No.	SROA Ring No.	Mesh Wire Red.	Tapping Wire Red.	Power Red.	ROA skew	SROA skew	Skew Red.
#03	29	16	44.8%	63.6%	40.7%	37.9ps	14.7ps	61.2%
#04	55	32	41.8%	11.9%	29.7%	25.3ps	15.4ps	39.1%
#05	55	30	45.4%	35.2%	46.0%	21.6ps	12.3ps	43.1%
#06	21	16	19.1%	16.4%	22.8%	19.2ps	11.5ps	40.1%
#07	37	27	27.0%	14.0%	25.8%	25.6ps	13.1ps	48.8%
#08	28	20	28.6%	17.4%	27.5%	22.0ps	12.5ps	43.2%
Avg	-	-	35.2%	26.4%	32.1%	-	-	47.1%

6. Frequency-Centric Resonant Rotary Clock Distribution Network Design Methodology

The geometry of a resonant rotary clocking system directly impacts its oscillation frequency. In this chapter, a frequency-centric methodology is proposed for the selection of the physical parameters of the resonant rotary clock for a target frequency. This proposed methodology is performed once for each target frequency on a semiconductor technology, in order to create a cell library of resonant rotary clock design components. A case study is performed to demonstrate the efficiency of the proposed methodology in accuracy and runtime. Simulation results show that the frequency optimization provides a good approximation for the resonant clock distribution network. At target frequencies between $4GHz$ and $6GHz$, the frequency difference is less than 0.20% and the run-time is reduced by approximately 70% compared to the traditional HSPICE simulation of an entire SROA network without the proposed simplifications for run-time improvement.

6.1 Introduction

A resonant clocking based clock distribution network has the following prime features: (1) It provides low power and low skew design solutions compared to the traditional clock network design methodologies; (2) More than one single-phase (or multi-phase) clock sources can be obtained from the resonant clock generation circuit, which suits the status quo for the distribution of clock to the synchronous components; (3) The resonant clock generation circuit is also a part of the clock distribution network, in that, it delivers the resonant clock signals to local areas for sub-network trees. Due to this interdependent nature of the clock generation circuit and the

distribution network, the frequency determination of the resonant clock distribution network is very challenging. The operating frequency of the resonant clocking based clock distribution network is hard to anticipate before the distribution network is constructed, which is counter-intuitive in the traditional physical design flow.

In previous designs, either 1) The operating frequency of the rotary oscillator array (ROA)-based clock distribution network is not considered as the main design target (the frequency of the clock distribution network is freely chosen) [32, 42, 68, 72, 80] or 2) The operating frequency of a single rotary traveling wave oscillator (RTWO) ring is investigated individually [59, 60, 83] with two shortcomings: i) ignoring the capacitance of the sub-network trees, and ii) not assessing the frequency of a global network of (RTWO) rings. For the former type of previous studies (i.e. item 1 above), such as the studies of [32, 42, 68, 72, 80], the target of the design is to optimize the structure of the clock distribution network without any consideration of the operating frequency accuracy in their clock network designs. For instance, in [68], a sparse-ROA (SROA) was constructed for the global distribution network of the rotary clock. But the frequency target was chosen without a prior information of the sub-network trees. For the latter type of previous studies (i.e. item 2 above), such as the study of [83], an analytical model is built for the computation of the frequency and power of a single ring (i.e. not a multi-ring ROA) for a target clock frequency. The analytical model is built using the physical parameters of a ring (e.g. transmission line width, length, height, separation, inverter size) but does not include the self-parasitics of sub-network trees, as the tree capacitances are unknown until much later in the design flow.

In order to optimize the operating frequency of the resonant clocking based clock distribution network, it is critical to: (1) Have a good frequency estimation of resonant clock generation circuit (i.e. a single ring, as in e.g. [83]); (2) Have a good approxima-

tion on the total capacitance over the sub-network trees; (3) Build a bottom-up design flow that starts with local sub-network trees and ends with a global resonant rotary clock distribution network of rings. To this end, it is proposed for the first time to perform a frequency optimization using "resonant circuit + sub-network trees", where the sub-network trees constitute the local level of the clock distribution network. The frequency optimization of the global resonant network including sub-network trees (as novelly proposed here) with HSPICE or analytical models is very time consuming. Instead, an efficient resonant clock distribution network frequency optimization flow is proposed. Overall, the contributions of this work are in the following aspects:

1. The frequency optimization methodology is integrated with the global resonant clock distribution network design, where the sub-network tree capacitance is considered in the frequency optimization flow. This is a first in literature in solving the frequency estimation challenge of resonant rotary clocks with high accuracy at the global (ROA or Sparse-ROA) scale.

2. The methodology embeds HSPICE simulation to characterize two limited size sub-circuit libraries: A customized inverter library and a customized transmission line library.

3. A simplification of the global ROA-based clock distribution network is applied during the ring physical structure optimization so as to speed up the optimization process.

The chapter is organized as follows: The proposed resonant clock distribution network frequency optimization methodology is presented in Section 6.2. The experiment setup and test results are presented in Section 6.3. The conclusions are presented in Section 6.4.

6.2 SROA-based Clock Distribution Network Frequency Optimization Methodology

In the proposed novel bottom-up approach, there are three steps in building the SROA-based clock distribution network for a target frequency in a specific order: (1) Subnetwork tree clustering; (2) Formation of initial SROA topology by optimizing the placement of the topology and the connections between the tapping points (of the topology) and the sub-network trees; (3) SROA generation. The design of an SROA-based clock distribution network through these three steps is a delicate process in delivering the target frequency. If the physical structure of the SROA is optimized for the target frequency without an estimation of the parasitic load of the sub-network trees, there will be large inaccuracies in the final operating frequency. On the other hand, the optimization process is very inefficient (e.g. due to speed) for an all-inclusive simulation of the SROA and the sub-network trees. The proposed methodology is a successful compromise between these two approaches. The advantages of the proposed methodology are: (1) The optimization model is simplified from the all-inclusive SROA-based clock distribution network in terms of complexity and runtime of the simulations while maintaining high frequency accuracy. (2) HSPICE simulation is embedded in the optimization flow for accuracy. The proposed methodology is a failure-aware optimization flow with self-correction in order to eliminate the influence of the invalid simulation results from some failure HSPICE runs.

6.2.1 Simplification of an SROA-based Clock Distribution Network

On the frequency optimization flow of an SROA-based clock distribution network for a given target frequency, the following simplification models are applied:

1. A representative ROA-Brick of the SROA is used as the simulation device under test (i.e. instead of a ring or the entire SROA).

2. The tapping points of the ROA-Brick are connected to singular representative capacitance elements, each with an effective average capacitance of the sub-network trees (i.e. instead of using a distributed RC network equivalent simulation model of a sub-network tree.)

In the first item, the total capacitance of each ring (of the ROA-Brick of the SROA) is contributed by its self capacitance and the sub-network tree capacitance. Thus, it is obvious that each ring in the SROA may have its own oscillation frequency if they oscillates independently. However, when rings are connected into an SROA, each ring in the topology synchronizes their own oscillation frequency with the other rings, and the SROA finally settles on a uniformed oscillation frequency. In the proposed frequency optimization process, the ROA-Brick is used, which is more accurate than using a single ring (demonstrated in experiments) and less computationally intensive than using the entire SROA. The deviations in frequency between using an ROA-Brick and an SROA are demonstrated in Section 6.3 to be negligible.

The second item is demonstrative of the capacitance balance problem [32, 72], which is a design automation challenge of rotary clocking that calls for distribution of the capacitance between the tapping points on an ROA (or SROA) to be within 20% of the max capacitive loading. In order to better synchronize all the rings in the SROA, the capacitance of the sub-network trees are maximally balanced. However, the capacitance of sub-network trees cannot be ideally equal. Since the sub-network trees are built independently using the BST/DME [22] methods, the total capacitances of all the sub-network trees are available for the frequency optimization. Thus, only one representative capacitance is used as the capacitive loading of each tapping point, set at the average effective capacitance value of the sub-networks. Experimental work in Section 6.3 demonstrates that this simplification in optimization has only a very small influence on the synchronized frequency of the SROA.

6.2.2 Reduction of the Frequency Optimization Parameters

A previous work [83] proposes a geometric programming framework for a single ring frequency optimization. In this work, a number of physical parameters are used for frequency optimization, such as the ring perimeter, the width and the separation of the transmission lines, a single inverter size and the number of inverter pairs. A finite number of simulations are performed with different combinations of the physical parameters for generating a frequency prediction analytical equation by curve fitting.

It is empirically shown in this work that using the proposed simplification model in Section 6.2.1, two physical terms are sufficient for the SROA-based clock distribution network frequency optimization:

1. The total transmission line length (ring perimeter),
2. The total PMOS/NMOS width.

The frequency optimization methodology proposed in this chapter is developed with these two design parameters. Extension to more than two design parameters, with diminishing returns on improvement, is trivial from the methodology currently hard-coded for two design parameters.

6.2.3 Failure-aware Frequency Optimization Methodology

The main flow of the optimization for a target frequency using the proposed model is shown in Fig. 6.1. The inputs are:

1. Custom-sized transmission line library (i.e. Parameter: The total transmission line length);
2. Custom-sized inverter library (i.e. Parameter: The total PMOS/NMOS width);
3. A target frequency.

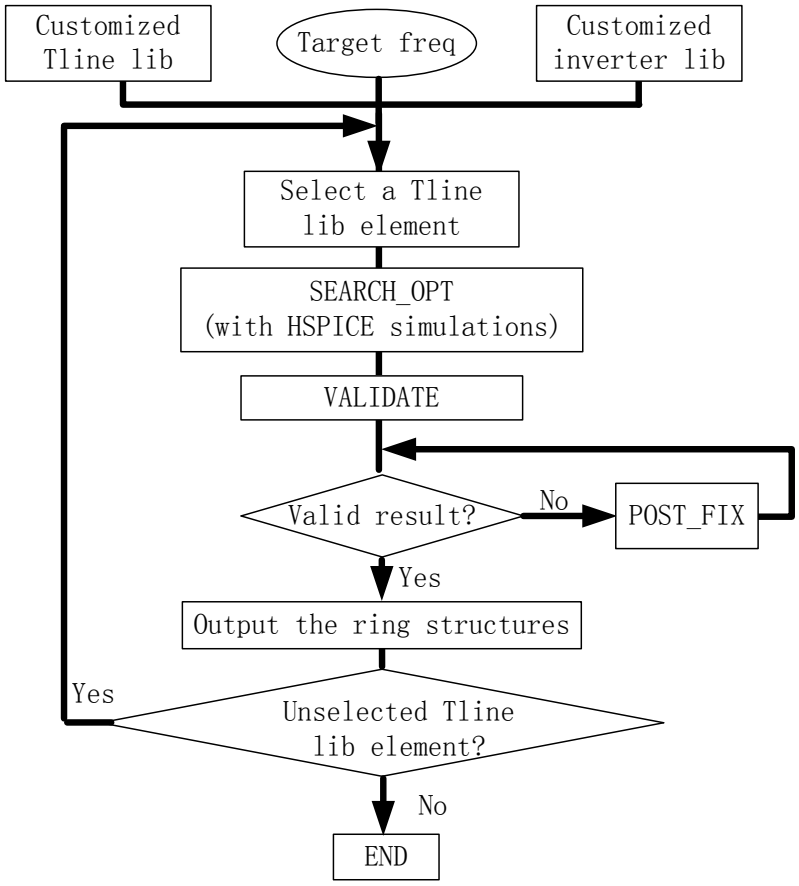


Figure 6.1: The frequency optimization flow.

Given a target frequency, different structures of the ROA-Bricks are constructed using elements from the custom, pre-characterized transmission line and the inverter libraries. The frequency optimization flow includes three main operations: (1) SEARCH_OPT; (2) VALIDATE; (3) POST_FIX. In the following paragraph, these three operations will be explained in detail.

SEARCH_OPT

The function SEARCH_OPT is used to search for an optimized inverter size for a given transmission line size in order to achieve the target frequency. A binary search

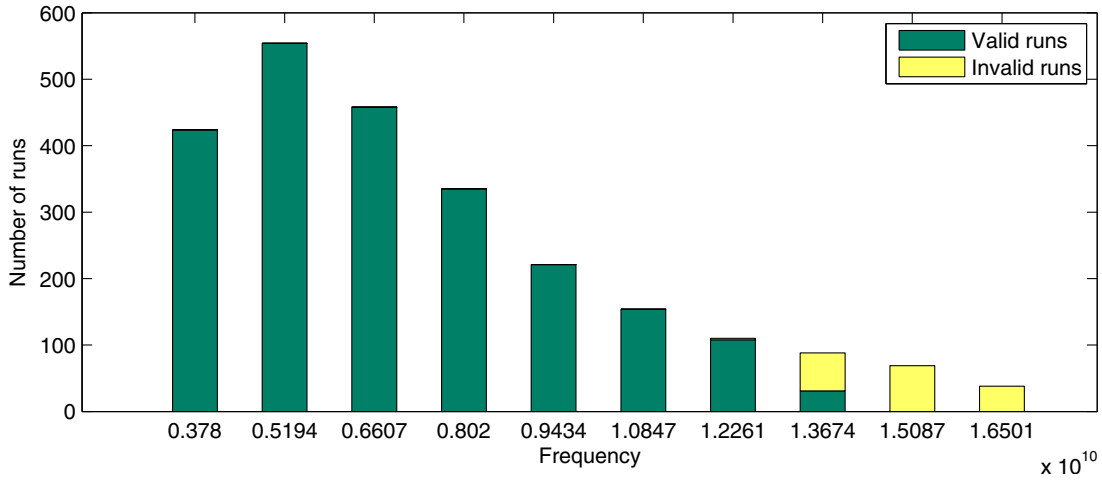


Figure 6.2: Failure count in a statistical HSPICE simulations with different ring physical structures.

is applied for pinpointing the target frequency based on the design space defined by the transmission lines and inverters provided as the inputs. HSPICE simulation is embedded inside SEARCH_OPT and the measurements are used for decision making for the binary search on frequency target.

However, the stability issue in HSPICE is a concern for resonant clock simulations at the high GHz range, due to the hardship in capturing self-oscillation in simulation. Fig. 6.2 shows the result of a sample statistical run. A customized transmission line library and a customized inverter library are used for varying ROA-Brick structures in the design space of the methodology. The sample statistical run has 66 elements in the inverter library and 38 elements in the transmission line library. The plot shows the frequency measurement results of 2508 ROA-Brick structures constructed using different transmission line and inverter size combinations. The result shows that 72 runs fail out of 2508 runs, and the fail cases mostly happen at high frequencies. These instability issues can be solved with manual intervention (e.g. infinitesimal capacitances etc.), but an automated solution is desirable for a fully-automated methodology.

Two observations can be obtained through the sample statistical run:

1. The measured frequency of a failure case is always higher than a feasibly simulated frequency.

2. Most of the failure runs can be tuned back by adding small tuning capacitance to the ring.

As shown in Fig. 6.3, there are two failure cases happens in frequency optimization process due to the convergence problem. In the case shown in Fig. 6.3(a), the frequency is measured on the huge spike of the oscillation, which is much higher than the nominal operating frequency. Failure case 1 usually happens at relative lower frequencies and can be tuned back by adding a fF scale capacitance. The second failure case as shown in Fig. 6.3(b) happens at relatively higher frequencies. The tuning process for failure case 2 is also a small, fF-scale, capacitance insertion to the netlist. However, the high frequency can sometimes require more than one iteration of capacitance insertion to reach stability.

The monitoring of the failure cases is embedded in the SEARCH_OPT function (Line 17 in Algorithm 5) using a frequency upper bound, $freq\_UB$. The value of $freq\_UB$ is continuously updated by the intermediate optimization results, which are higher than the target frequency f_{target} and lower than the current $freq\_UB$ (Line 20). On the other hand, an intermediate optimization result being higher than the current $freq\_UB$ indicates a failure (Line 17). Then the tuning process starts to work by calling HSPICE simulation with a tuning capacitance add_cap in fF scale. The tuning process stops when the measured frequency is smaller than the current $freq\_UB$ or the tuning iterations reaches its practical limit try_limit. The optimization process can be finished in two ways: First, it can be finished in the same way as the traditional binary search; second, a user defined resolution controller Δ can be used to control the resolution of the optimization results towards the target frequency.

The output of SEARCH_OPT is the optimized frequency f_{opt} for each transmission line $Tline(i)$ (ring perimeter), the corresponding power consumption p_{opt} and the selected inverter size inv_{opt}.

For most of the cases, $freq\_UB$ is effective in monitoring the emergence of a failure case within the optimization process. There is only one case for which function SEARCH_OPT could generate a wrong answer: The measured frequency of the failure case is lower than the current $freq\_UB$ and its actual frequency is lower than f_{target}. Normally, the granularity of the elements in the two sub-circuit libraries does not allow this situation to happen because an invalid frequency value is usually more than twice its actual frequency and the current $freq\_UB$ should be lower than the invalid value and detects the failure run. However, it is theoretically possible that all the intermediate runs, the measured frequencies of which are higher than f_{target}, are failure runs. In this unlikely case, the $freq\_UB$ would have been updated with invalid frequency values. This unlikely case is later differentiated by function VALIDATE and corrected by POST_FIX.

In the proposed methodology, a valid optimized frequency of SEARCH_OPT should be within the Δ region of f_{target} and lower than f_{target}, which is expressed in Equation 6.1:

$$f_{target} - \Delta \leq f_{opt} \leq f_{target} \tag{6.1}$$

VALIDATE

If the optimized frequency is valid in SEARCH_OPT, $done\_flag$ is set to value 1. There are three possibilities when $done\_flag = 0$:

1. f_{target} is not reachable for a certain transmission line size and the optimized frequency is always too high or too low towards f_{target},

Algorithm 5 SEARCH_OPT

Input: Customized transmission line library element $Tline(i)$, inverter size library Inv, and target frequency f_{target}, $freq\_UB$

Output: Optimized frequency f_{opt}, power p_{opt}, selected inverter number j and $done\_flag$

```
add_cap = 0
cnt_add_cap = 0
set try_limit
done_flag = 0
low = 1
high = cardinality of Inv
mid = ⌈(low + high)/2⌉
while low <= high do
    [f_opt, p_opt] ← HSPICE simulation (Inv(mid), add_cap)
    if | f_target − f_opt | ≤ Δ AND f_opt ≤ f_target then
        done_flag = 1
        Break
    else if cnt_add_cap > try_limit then
        add_cap = 0
        cnt_add_cap = 0
        low = mid + 1
    else if f_tmp > freq_UB then
        add_cap = add_cap + δ
        cnt_add_cap + +
    else if f_tmp > f_target then
        freq_UB = f_tmp
        add_cap = 0
        cnt_add_cap = 0
        low = mid + 1
    else
        add_cap = 0
        cnt_add_cap = 0
        high = mid - 1
    end if
    mid = ⌈(low + high)/2⌉
end while
j=mid
Return f_opt, p_opt, j, done_flag
```

2. The optimized frequency is smaller than $f_{target} + \Delta$,

3. The optimized frequency is much lower than f_{target}, which indicates it is a failure case.

The function VALIDATE differentiates between these three possibilities, leveraging the mechanics of binary search in deterministically traversing through inverter and transmission-line sizes in the custom libraries. The inputs of the function VALIDATE are: The customized transmission line library element $Tline(i)$, the target frequency f_{target}, the optimized frequency f_{opt} for $Tline(i)$, the corresponding power consumption p_{opt}, the selected inverter number j, the customized inverter library Inv and the $done\_flag$. The outputs are: The optimized frequency f'_{opt} and corresponding power consumption p'_{opt} after validation, the selected inverter number j' after validation and an error flag err_flag.

As shown on Line 7 and Line 9 of Algorithm 6, if f_{target} is unreachable, the selected inverter number should reach either end of the customized inverter library Inv (due to binary search in SEARCH_OPT). Otherwise, if it is a failure case, the wrong optimization result is below f_{target}. In addition, the wrong optimization result is next to the last wrong intermediate result, whose measured frequency is higher than f_{target} and actual frequency is lower than f_{target}. For example, assume the inverter elements in the customized inverter library are ordered in an increasing order of their sizes and the measured frequency of the ring using inverter number n_1 is higher than f_{target} and the actual frequency is not. If the wrong measurement is not detected by function SEARCH_OPT, the new searching region of the optimal inverter size will be narrowed to a region, where the inverter number are larger than n_1 (the inverter sizes are larger than that of n_1). It is easy to recognize that this is a wrong region for a search because all the inverter sizes in this region are corresponding to a lower frequency compared to that of n_1. If all the measured frequencies in this region are

Algorithm 6 VALIDATE

Input: $Tline(i)$, f_{target}, f_{opt}, p_{opt}, j, Inv, $done\_flag$
Output: f'_{opt}, p'_{opt}, j' and err_flag

```
1:  err_flag = 0
2:  f'_opt = f_opt
3:  p'_opt = p_opt
4:  j' = j
5:  if done_flag == 1 then
6:     Done
7:  else if f_opt > f_target AND j == cardinality of Inv then
8:     f_target is too low
9:  else if f_opt < f_target AND j == 1 then
10:    f_target is too high
11: else if f_opt < f_target AND f+_opt > f_target then
12:    if f+_opt < f++_opt then
13:       DONE
14:    else
15:       err_flag = 1
16:    end if
17: else if f_opt > f_target then
18:    [f'_opt, p'_opt] ← HSPICE simulation(Inv(j + 1))
19:    Inv(j') ← Inv(j + 1)
20: end if
21: Return f'_opt, p'_opt, j' and err_flag
```

all reliable, SEARCH_OPT will result in $n_1 + 1$. The frequencies corresponding to inverter number $n_1 + 1$ and n_1 are corresponding to f_{opt} and f^{+}_{opt}, respectively (Line 11). It is also easy to recognize that if inverter number n_1 corresponding to the wrong measured intermediate result, $n_1 - 1$ has never been circled into the inverter size searching region. Thus, f^{++}_{opt}, using inverter number $n_1 - 1$, is also measured. If the following expression is satisfied:

$$f_{opt} < f^{++}_{opt} < f^{+}_{opt}, \tag{6.2}$$

it indicates that f_{opt} is a wrong result. Under this situation, the error flag err_flag is labeled as 1, indicating the optimization process needs to start over. On the other side, if the following equation is satisfied:

$$f_{opt} < f_{opt}^{+} < f_{opt}^{++}, \tag{6.3}$$

it indicates that the f_{opt} is the best result can be reached, which is lower than f_{target}. The optimized result does not qualify for Equation 6.1 may because: (1) The granularity of the customized transmission line library and the customized inverter library do not provide a high enough resolution; (2) f_{target} is unreachable.

As shown on Line 17, by ruling out the other possibilities, if f_{opt} is higher than f_{target} (the inverter number j is on neither end of the inverter library in this case), it is the optimal result that the optimization process can reach. VALIDATE intentionally outputs the frequency using the inverter number $j+1$, which is the largest frequency below f_{target}.

POST_FIX

As shown in Section 6.2.3, if the optimized result is detected to be invalid, which satisfies Equation 6.2, the function POST_FIX is triggered to reapply the optimization on transmission line $Tline(i)$. The inputs of POST_FIX are the customized transmission line library element $Tline(i)$, the customized inverter library Inv, the target frequency f_{target}, the selected inverter number $j^{'}$ after validation, the error flag err_flag and a frequency upper bound $freq\_UB$. The outputs are: The optimized frequency $f_{opt}^{''}$ and corresponding power consumption $p_{opt}^{''}$ after validation, the selected inverter number $j^{''}$ after validation and the error flag err_flag.

The function POST_FIX includes the functions SEARCH_OPT_P and VALIDATE. SEARCH_OPT_P is almost the same as SEARCH_OPT and the only dif-

Algorithm 7 POST_FIX

Input: $Tline(i)$, Inv, f_{target}, j', err_flag and $freq\_UB$
Output: f''_{opt}, p''_{opt}, j'' and err_flag
1: $high = j'$
2: $[f'_{opt}, p'_{opt}, j'_p, done\_flag]$=SEARCH_OPT_P $(Tline(i), Inv, f_{target}, freq\_UB, high)$
3: $[f''_{opt}, p''_{opt}, j''$ and $err\_flag]$ = VALIDATE $(Tline(i), f_{target}, f'_{opt}, p'_{opt}, j'_p, Inv, done\_flag)$
4: Return f''_{opt}, p''_{opt}, j'' and err_flag

ference is that SEARCH_OPT_P uses the inverter number j' as its binary search upper bound. This is because: (1) VALIDATE has shown that the actual frequency of the ROA-Brick by applying inverter number j' should be lower than f_{target}. Thus, the searching region of the optimal inverter size should be narrowed to the region, in which the inverter sizes are smaller than that of inverter number j'. In this way, the searching process becomes more efficient. (2) Changing the upper bound of a binary search helps to prevent the searching process to converge into the wrong optimization result again. As shown in Fig. 6.1, the function POST_FIX keeps looping until $err\_flag = 0$, which also indicates the optimization process is finished.

The final output of the optimization flow is all the possible (non-unique) physical structures of the ROA-Bricks for the f_{target}. Users can pick from the ROA-Brick physical structure list for a proper ring structure for the SROA-based clock distribution network design based on the design-specific frequency resolution, slew and power consumption objectives.

6.3 Experiments

The performance of the proposed frequency optimization methodology for optimizing an SROA-based resonant system at $4.5GHz$ is investigated in detail. The efficacy of the proposed methodology at other frequency targets, empirically chosen

between $4.0GHz - 6.0GHz$, are also demonstrated. The results are verified through HSPICE simulations with a 90nm technology.

6.3.1 Validation of the Simplification of the SROA-based Clock Distribution Network

In Section 6.2.1, it is proposed to use: (1) An ROA-Brick instead of an SROA; (2) An equivalent average sub-network tree capacitance model for the frequency optimization. In this section, both of these two simplifications are tested with simulations for their correctness.

SROA to ROA-Brick

In this section, experiments are performed to find out the frequency differences among an SROA, an ROA-Brick and a single ring when loaded with different capacitive loading. All the rotary oscillators are designed around $4.5GHz$. Fig. 6.4 shows the frequency comparison of these three kinds of resonant clocks with the same amount of capacitive loading on each tapping point. The capacitance increases from $0 - 3pF$ for each tapping point, representative of a range from minimal loading to a maximum likely capacitance on an RTWO tapping point (derived from empirical data of capacitance per area).

Fig. 6.4 indicates that the ROA-Brick provides a better frequency approximation to the SROA than a single ring. More importantly, the frequency measurement results of the ROA-Brick to that of the SROA is within **0.12%** of the HSPICE-validated SROA frequency, and is practically error free for the average capacitance load per tapping point data point of 1.5pF/TP. It is concluded from this empirical data, with very high confidence, that an ROA-Brick can be used for the frequency optimization for the entire SROA.

The Capacitance Imbalance Tolerance of the SROA

In the SROA-based clock network design, the total capacitances of each sub-network tree are not guaranteed to be equal. Thus, it is important to analyze the fidelity of the equivalent average sub-network tree capacitance model used in the simplification model proposed in Section 6.2.1. In the following simulations, the same SROA used in Section 6.3.1, which contains 12 rings, is used for testing. The total capacitive loading of the SROA resonant system is $22.8pF$, given a $1.9pF$ capacitive loading for each tapping point. The total capacitance values are kept the same through the simulation but the maximum capacitance difference among the tapping points is swept to investigate the imbalance tolerance. Shown in Fig. 6.5 is the frequency of the SROA normalized to the original frequency of $4.5GHz$ for each capacitance load difference. The maximum capacitive loading difference is swept from 0% to 119%. The frequency changes of the SROA system are all under 0.4%. These empirical results demonstrate that the capacitance imbalance among the tapping points has a very small influence on the frequency of the whole resonant system. To this end, the equivalent average sub-network tree capacitance model can be used in the frequency optimization flow.

6.3.2 The Completeness of the Two-Parameter Optimization

In Section 6.2.2, the total transmission line length (ring perimeter) and the total PMOS/NMOS width are the two parameters used in the proposed optimization methodology. Compared to the multi-parameter optimization in [83], these two parameters can also provide multiple optimization results for power and performance trade-offs. Table 6.1 shows a group of the optimized solutions for a target frequency of $4.5GHz$. The optimization results with different ring perimeter are ranging from $5280\mu m - 7440\mu m$ and the corresponding total PMOS width for each optimization

Table 6.1: 10 non-unique feasible solutions for a frequency target of 4.5GHz with varying power and slew features

Opt. Results	Ring P.(μm)	$\sum W_p$.(μm)	f(GHz)	pwr(W)	Slew(s)
#1	5280	9240	4.47	0.32	1.50e-11
#2	5520	8760	4.49	0.31	1.53e-11
#3	5760	8160	4.45	0.29	1.57e-11
#4	6000	8040	4.48	0.29	1.61e-11
#5	6240	7680	4.48	0.28	1.65e-11
#6	6480	7200	4.48	0.26	1.70e-11
#7	6720	7080	4.49	0.26	1.73e-11
#8	6960	6840	4.48	0.25	1.77e-11
#9	7200	6600	4.48	0.24	1.80e-11
#10	7440	6360	4.48	0.24	1.84e-11

result are ranging between $9240\mu m - 6360\mu m$. The simulation results show that along with increasing the ring perimeter (and the decrease of the total PMOS width), the power of the resonant system decreases and the slew increases. Thus, the two-parameter frequency optimization methodology can be used to generate SROA structures for different design need while maintaining high frequency accuracy.

6.3.3 A Case Study

The #4 circuit of the ISPD 10 Benchmark is used for a case study. The experiments are performed from three aspects: (1) The frequency accuracy is compared between the frequency optimization model and the SROA-based clock distribution network using the optimized result. (2) The runtime is compared between the proposed optimization methodology and the SROA-based clock distribution network. (3) The accuracy of the proposed methodology is tested over a frequency range of $4.0GHz - 6.0GHz$ in order to verify its adeptness for different target frequencies.

Table 6.2: Comparison of the frequency accuracy between the optimization results and the final SROA-base clock distribution network for ISPD 10 Benchmark circuit 04, using the models in [83], [68] and the proposed model

Opt. Result	f_{model} [83]	f_{cdn1}	Diff1 (%)	f_{model} [68]	f_{cdn2}	Diff2 (%)	Prop. f_{model}	f_{cdn3}	Diff3 (%)
#1	4.49	4.39	2.26	4.49	4.42	1.63	4.47	4.47	0.09
#2	4.47	4.37	2.30	4.47	4.40	1.62	4.48	4.48	0.12
#3	4.50	4.39	2.44	4.50	4.42	1.72	4.48	4.48	0.17
#4	4.49	4.38	2.51	4.49	4.42	1.74	4.48	4.48	0.14
#5	4.47	4.39	2.53	4.47	4.39	1.78	4.49	4.49	$\lvert -0.20 \rvert$
Avg	—	—	2.41	—	—	1.70	—	—	0.15

Frequency Accuracy

For the frequency accuracy comparison, the models in [83], [68] and the proposed model are tested for a given target frequency of $4.5GHz$. Remember that i) In [83], the physical structure of a single ring is optimized for the target frequency; ii) In [68] the physical structure of an SROA is optimized for the target frequency. Both of these methodologies perform the frequency optimization without the knowledge of the capacitance of the clock sub-network trees. In order to compare the frequency accuracy of the models in [83] and [68] with the proposed model, the optimization results of [83], [68] and the proposed model from their respective targets of a ring, an SROA and an ROA-Brick with sub-network tree capacitance, are applied in SROA-based clock distribution network (CDN) construction. The measured frequency of the clock distribution networks using the optimized results of [83], [68] and the proposed model are denoted f_{CDN1}, f_{CDN2} and f_{CDN2}, respectively. Correspondingly, Diff1, Diff2 and Diff3 shows the frequency difference between each model optimization result and its corresponding clock distribution network. These results are shown in Table 6.2for five non-unique solutions. The results indicate that the frequency accuracy between the proposed model and its corresponding SROA-based clock dis-

tribution network is less than 0.20%, which is much higher than those in [83] and [68] (2.41% and 1.70%,respectively, on average).

Run-time

The run-time comparison between the proposed methodology and an SROA-based clock distribution network HSPICE simulation of the five non-unique frequency optimization results at a target frequency of $4.5GHz$ are shown in Table 6.3. The ring perimeters are varied between $5280\mu m - 6240\mu m$ and the total PMOS width of a single ring are changed between $9240\mu m - 7680\mu m$. The proposed frequency optimization methodology embeds HSPICE simulation in the optimization flow. Thus, the run-time T_{all} of the proposed methodology for a transmission line size $Tline(i)$, can be expressed as:

$$T_{all} = t_{HSPICE} * n + t_{rest}, \tag{6.4}$$

where t_{HSPICE} is a single HSPICE simulation time, n is the number of HSPICE simulations involved in the frequency optimization flow for a transmission line size $Tline(i)$. The rest of the run-time is expressed as t_{rest}. Since 1) $t_{HSPICE}*n$ dominates the run-time T_{all}, 2) the run-time of the frequency optimization flow using SROA-based clock distribution network is too time consuming to be realized, an average run-time T_{avg} is calculated from the proposed methodology to compare to a single SROA-based clock distribution network HSPICE simulation. T_{avg} is expressed as:

$$T_{avg} = T_{all}/n. \tag{6.5}$$

The results show that the proposed model saves the run-time as much as 71.20% compared to an SROA-based clock distribution network. This reduction in the run-time comes from the simplification in the proposed simulation model.

Table 6.3: Comparison of run-time between the proposed methodology and a single HSPICE run of an SROA for ISPD 10 Benchmark circuit 04

Result	Ring P.(μm)	$\sum W_p(\mu m)$	T_{avg}(s)	SROA-CDN(s)	Red.(%)
#1	5280	9240	402.21	1242.25	67.62
#2	5520	8760	387.35	1267.68	69.44
#3	5760	8160	390.25	1354.93	71.20
#4	6000	8040	398.82	1237.02	67.76
#5	6240	7680	399.11	1327.38	69.93

Table 6.4: Comparison of frequency accuracy of the proposed model and a single HSPICE run of an SROA for ISPD 10 Benchmark circuit 04 in a frequency range of $4.0 - 6.0GHz$

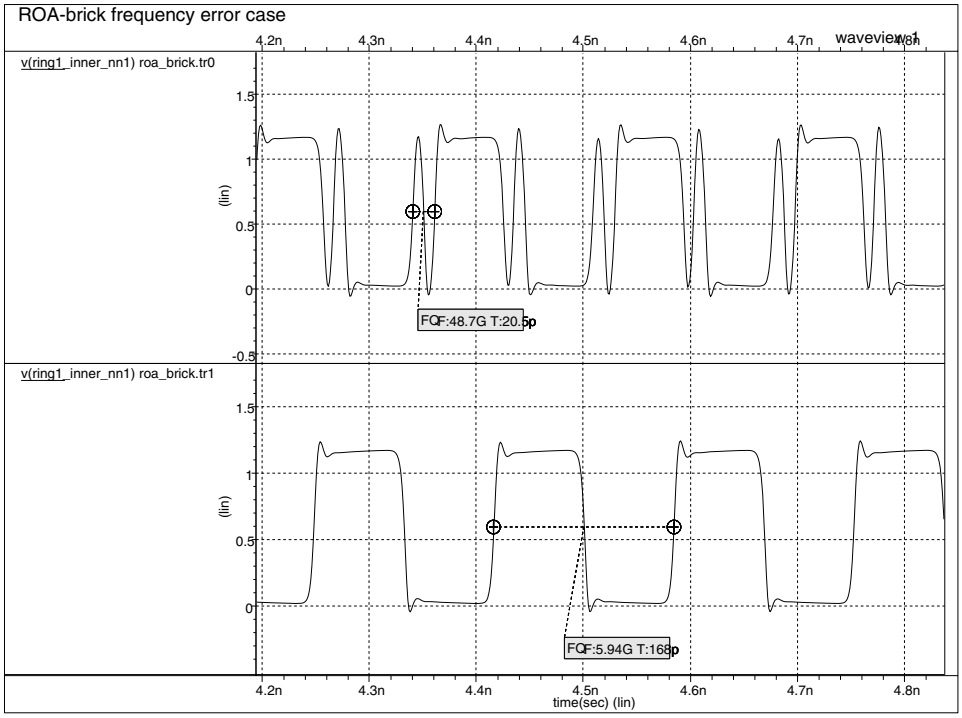
	4.0GHz	4.5GHz	5.0GHz	5.5GHz	6.0GHz
Prop.f_{model}(GHz)	3.99	4.48	4.99	5.47	5.98
Prop.f_{CDN}(GHz)	3.99	4.48	5.00	5.47	5.98
Diff(%)	0.04	0.15	0.13	0.04	$\lvert -0.10 \rvert$

Frequency Accuracy for Different Target Frequencies

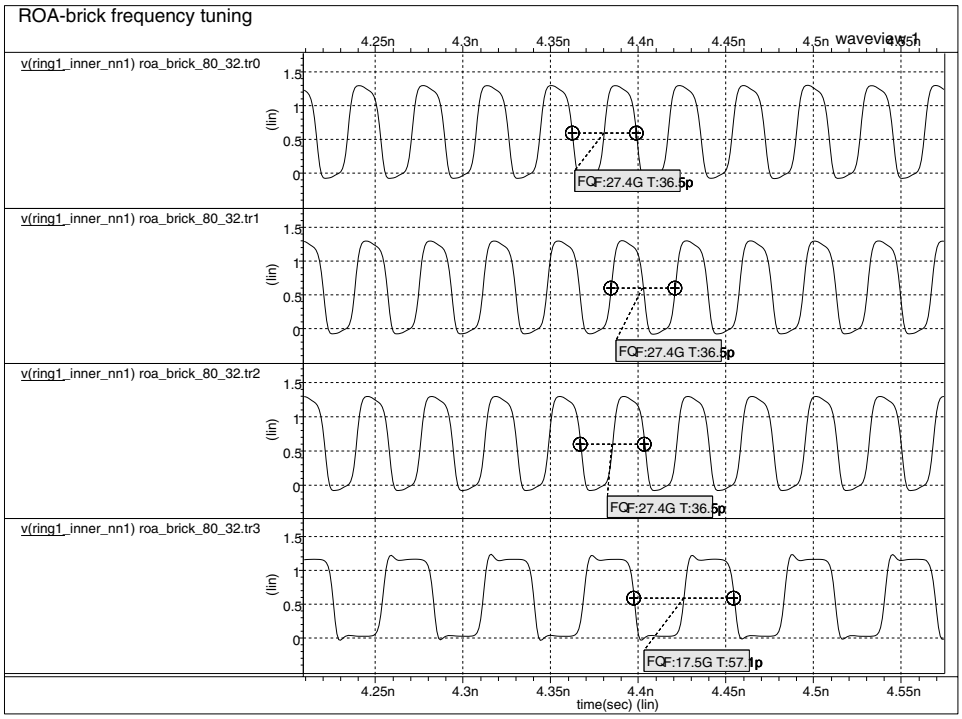
The proposed frequency optimization methodology is tested for different target frequencies, as well. To this end, the proposed methodology is used to optimize singular ROA-Brick structures in a frequency range of $4.0 - 6.0GHz$ and the interval is set as $0.5GHz$. For standardization of the comparison results, the perimeter of each ring is selected to be $6000\mu m$ and the size of the inverters are optimized for different target frequencies. The results are shown in Table 6.4. The simulation results show that the proposed frequency optimization methodology maintains a good performance over the entire spectrum of the selected $2GHz$ frequency range: The frequency difference between the proposed model and the SROA-based clock distribution network is no more than 0.1%.

6.4 Conclusions

In this chapter, a frequency optimization methodology for the SROA-based clock distribution network is proposed. In the proposed methodology, a simple model is used for frequency optimization, which is efficient while still maintaining high accuracies. The simulation results show that the frequency difference between the proposed model and the SROA-based clock distribution network is lower than 0.20% with runtime saving of 71.20%. The simulation results also show that the proposed frequency optimization methodology shows high frequency accuracy in a wide frequency range from $4.0 - 6.0GHz$.



(a) Failure case 1,



(b) Failure case 2.

Figure 6.3: Failure cases in frequency optimization for a target frequency.

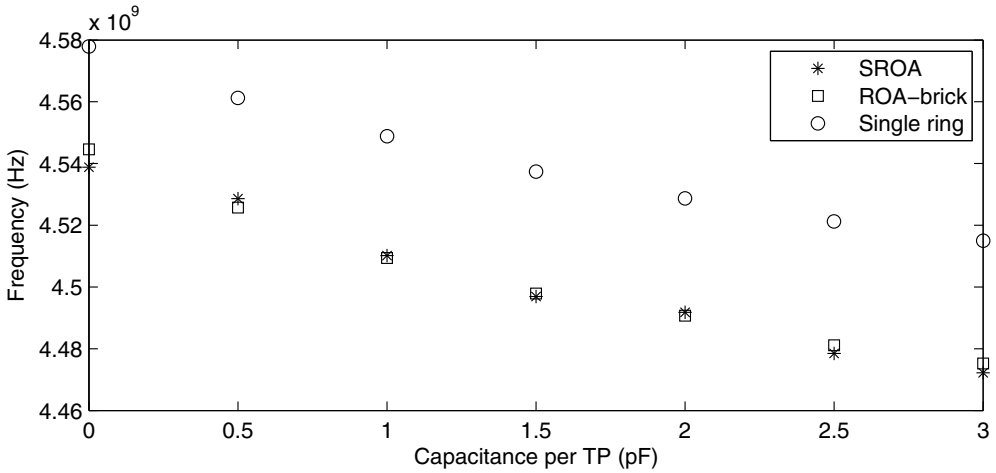


Figure 6.4: The frequency approximation among SROA, ROA-Brick and single ring, SROA is SPICE accurate but also the slowest.

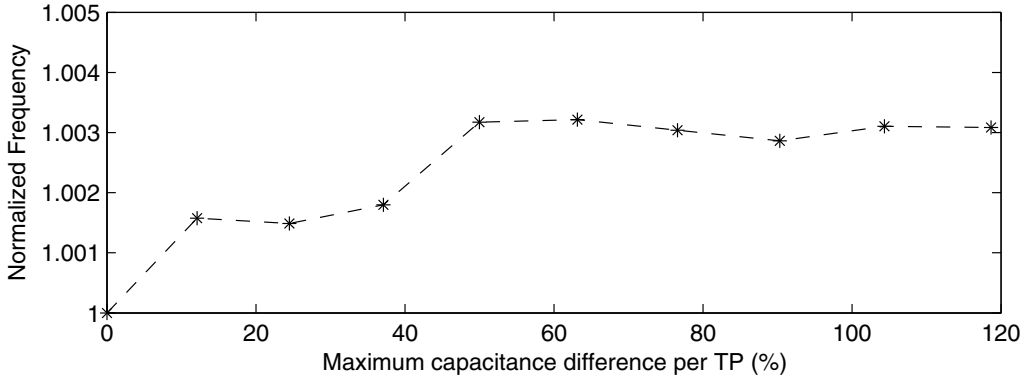


Figure 6.5: The frequency change for capacitance imbalance.

7. Physical Design Flow for Rotary Clocking Based Circuit

Resonant rotary clocking technology as a merging technology, arouses people's interests for it can generate high frequency oscillation signals with low power dissipation. However, there is currently no tool supporting the physical design of the rotary clocking based circuit. In this chapter, a novel physical design flow for rotary clocking based circuit is proposed, which is developed with mainstream design automation tools.

7.1 Introduction

The modern EDA tools provide automation facilities for the physical design of traditional non-resonant circuits, such as floorplan, placement, clock tree synthesis (CTS), routing and extraction. In order to develop the physical design flow for rotary clocking based circuit, several points need to be addressed:

1. The automation flow of the ROA layout generation for a target frequency;
2. Perform circuit synthesis with multiple clock sources;
3. The resonant clock aware floorplan, placement, CTS and routing.

This chapter is organized into the following sections. In Section 7.2, the physical design flow for rotary clocking based circuit is proposed. In Section 7.3, the conclusions are provided.

7.2 Frame of the Rotary Clocking Based Circuit Physical Design Flow

The major steps of the rotary clocking based circuit physical design flow are:

1. Brick-based ROA physical structure optimization for a target frequency;

2. Brick-based ROA layout and schematic generation;

3. Synchronous circuit synthesis with multiple clock sources;

4. The ROA aware power planning, placement, CTS and routing.

In the rest of this section, these four steps of the rotary clocking based circuit physical design flow are explained in detail with an design example. In this design example, two simplications are made:

1. An ROA-Brick is used to form the resonant global distribution network instead of a brick-based ROA.

2. The sub-network tree for each tapping point are the same.

As shown in Chapter 6, the ROA-Brick is used as an simplication model for SROA in physical structure optimization for a target frequency, which effectively accelerates the optimization process while still maintains a high accuracy in frequency approximation. In this physical design example, an ROA-Brick is used to composed the global network of the clock distribution network. In addition, in order to maximally balance the total capacitive loading for each ring tapping point [32, 72], in this design example, the same circuit is use as the sub-network tree for each tapping point.

7.2.1 ROA-Brick Physical Structure Optimization for a Target Frequency

In order to optimize the physical structure of the ROA-Brick for a target frequency, the methodology in Chapter 6 is applied. In this design, the operating frequency is set to be $4GHz$. The capacitive loading for each tapping point is approximated by the total sink (e.g. register) capacitance of each sub-network tree. The total sink capacitance may introduce some mismatch to the target frequency since the tapping wire capacitance is not considered. This frequency mismatch can be later fixed by tuning varactors of the ROA-Brick circuits [76].

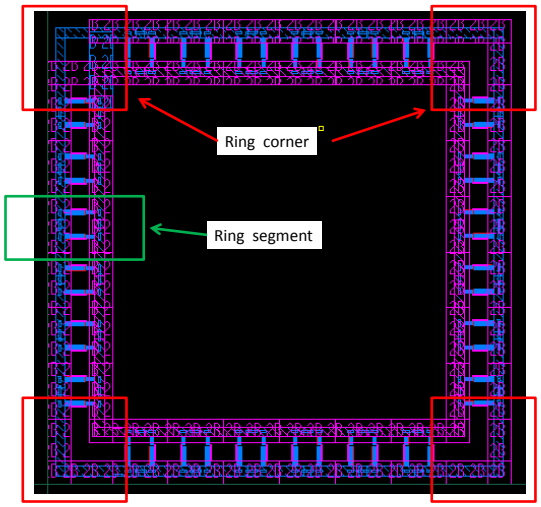
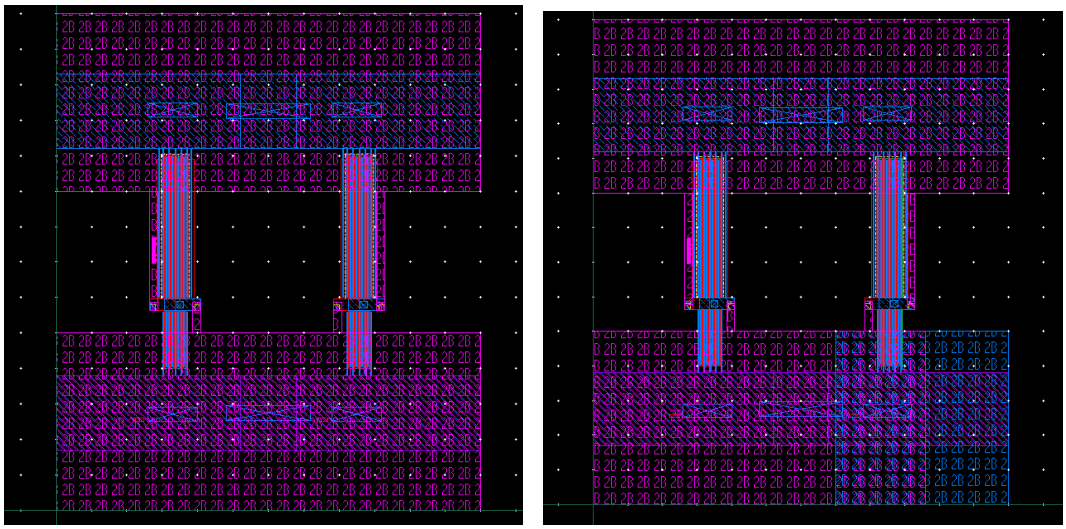


Figure 7.1: The layout of an RTWO ring and its components.

7.2.2 ROA-Brick Layout and Schematic Generation

By investigating the one cross-over ring structure, it is easy to observe that the ring layout is composed of two kinds of sub-structure layout: ring segment and ring corner. This is because the anti-parallel inverter pairs are evenly distributed between the transmission lines. As shown in Fig. 7.1, the ring segment is composed of two pieces of transmission line segments with an anti-parallel inverter pair connected between them. Specifically, two ring corner structures are needed in order to form the Möbius connection: The crossover corner and the normal corner. All the sub-structure layout can be constructed based on the following physical parameters:

1. Ring perimeter
2. Transmission line width
3. Transmission line separation
4. Inverter size (PMOS/NMOS width)



(a) Normal ring segment. (b) Ring segment for crossover connection.

Figure 7.2: Different kinds of ring segments in ring layout.

Since all these physical parameters can be obtained from the ROA-Brick physical structure optimization, the ROA-Brick layout construction is developed with Cadence SKILL scripts by applying the above physical parameters as the inputs. The layout of the ROA-Brick is generated in a hierarchical way and the schematics are generated along with all the layouts, which are used for later LVS check. Furthermore, in order to facilitate the later power planning designs in Cadence Encounter Digital Implementation System (EDI), the detailed physical parameters such as the coordinates and width of the inverter layout are obtained along with the ROA-Brick layout generation.

After the layout and the schematic of the ROA-Brick are generated, design rule check (DRC) and layout versus schematic (LVS) are performed. It is noted that, in order to form the crossover corner in the layout, the layouts of the ring segments used to connect the crossover corner is slightly different from the other ring segments. Fig. 7.2 shows two kinds of ring segments and the one shown in Fig. 7.2(b) is used

to facilitate the layer jump of the crossover corner. Though the layouts are different, these ring segments correspond to the same schematic. To this end, in order to reduce the workload of physical verifications, DRC checks are performed for every layout cell and LVS checks are performed on part of the ROA-Brick sub-structures. The physical verifications have the following characteristics:

1. The physical verifications are performed automatically with perl scripts;
2. The DRC and LVS checks are performed in a bottom up fashion. Along with a verification is finished, the verification result is verified for correctness. The verification process stops when a DRC and/or LVS error comes up.
3. The LVS checks for partial ROA-Brick sub-structures do not harm the overall checking result since the most top level design is checked.

7.2.3 Synchronous Circuit Synthesis with Multiple Clock Sources

A main feature of the rotary clocking based circuit design is the on-chip clock generation and distribution network. However, the the on-chip clock generation and distribution is not supported by the current physical design tools. In order to solve this problem, the structure of the synthesized circuits for ROA-Brick based circuit design is shown in Fig. 7.3. The clock pin CK is only used to facilitate the circuit synthesis and is removed later in clock tree synthesis step. Clock signals are later tapped off from the tapping points of the ROA-Brick during clock tree synthesis and the locations of the tapping points are labeled by the dummy cells Ck_{buff}.

In order to maximally balance the capacitive loading for each tapping point, four ISCAS Benchmarks are used as the sub-network tree of each tapping point of the ROA-Brick. Perl script is used to facilitate the circuit synthesis, the functions of which are:

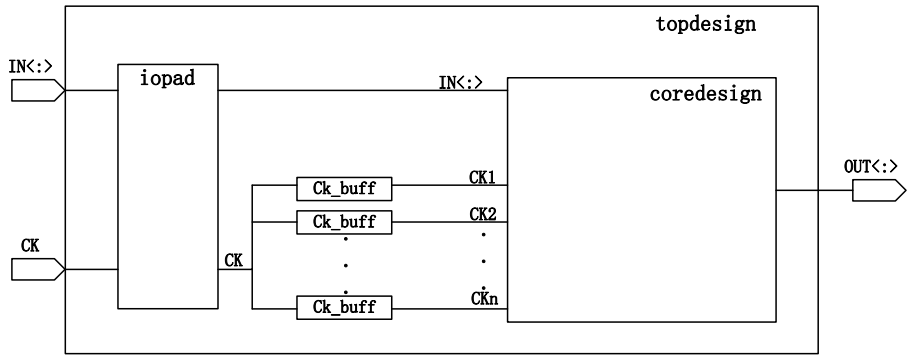


Figure 7.3: The circuit structure after synthesis.

1. Generate a top level Verilog file is used to drive these four sub-circuits.
2. Generate the corresponding design constraint file for circuit synthesis (written in Synopsys Design Compiler syntax).
3. Generate the Tcl scripts for circuit synthesis.

The outputs of circuit synthesis are:

1. A mapped Verilog file.
2. A constraint file.
3. A configuration file for Cadence EDI.

7.2.4 Power Planning, Placement, CTS and Routing with Cadence EDI

In order to perform the physical design for the ROA-Brick based circuits, the most challenging part is to avoid short circuit between the ROA-Brick and the other part of the circuits, such as power rails, routing wires and standard cells. The short circuit problem needs to be taken into account during floorplan, placement, CTS and routing. The physical design flow are performed with the following steps:

1. File preparations and floorplan;
2. Power planning 1: Power ring and power strip generations;
3. ROA-Brick placement/routing blockage generation;
4. Power planning 2: Standard cell power rail generation;
5. One-to-one pairing between the tree roots to the tapping points;
6. Placement, CTS and routing with timing optimizations;

File preparations and floorplan

In order to use the conventional physical design tool to facilitate the ROA-Brick based circuit physical design, in Section 7.2.3, the synchronous circuits are synthesized with multiple clock sources. These multiple clock sources are defined at the outputs of some dummy cells (Buffers are used here). These dummy cells are intentionally added during circuit synthesis and are placed at the tapping point locations of the ROA-Brick during placement. Perl script is used in this step to generated the IO assignment file. The design import and floorplan steps in ROA-Brick based circuit physical design are similar to the corresponding steps in conventional physical design.

Power Planning 1: Power Ring and Power Strips generation

Similar to conventional physical design, a power ring is generated around the die area, which is used to deliver the power signal from the power pad to the circuits. Then, power strips are generated in vertical direction. In order to avoid short circuit between the power strips and the ROA-Brick, the physical information of the ROA-Brick obtained from the ROA-Brick layout is used to facilitate the placement of the power strips.

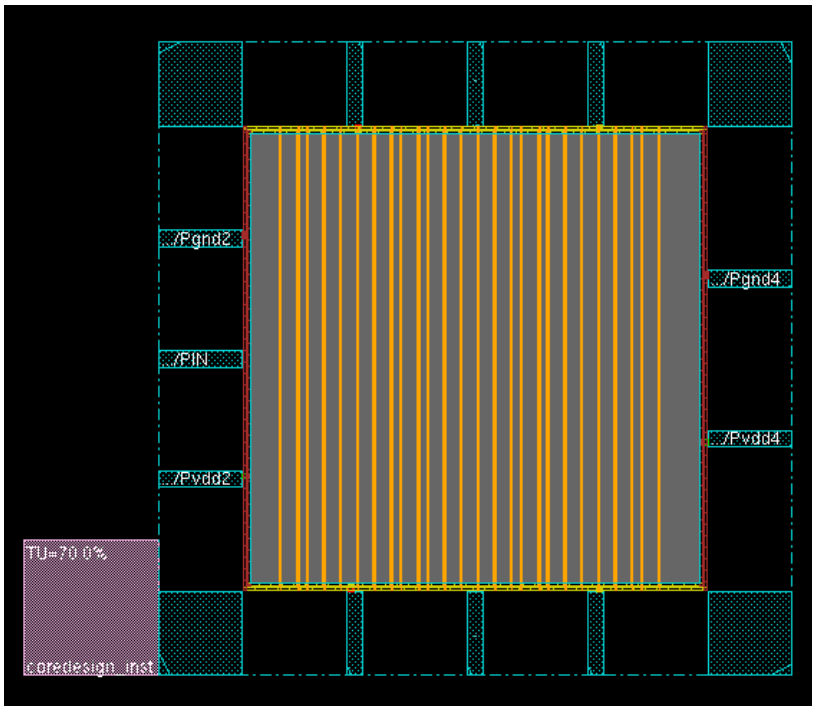
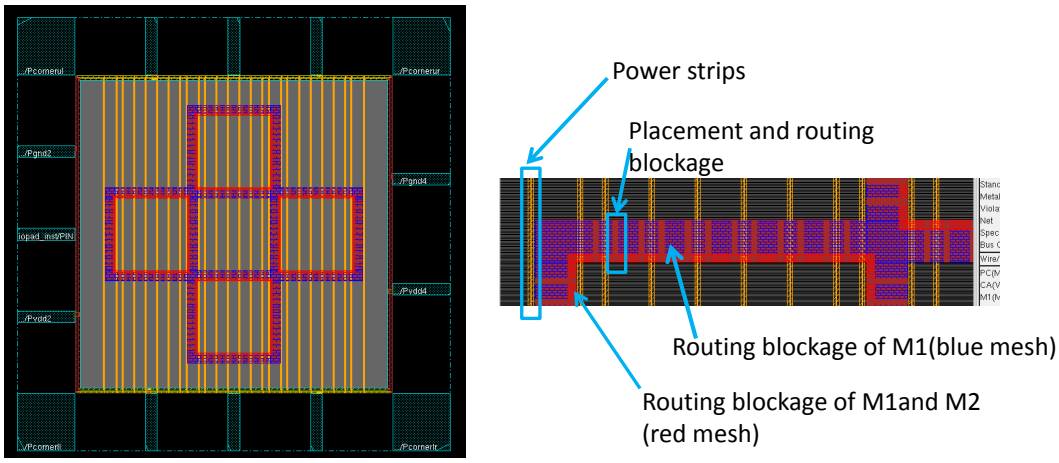


Figure 7.4: The power strips created after floorplan.

Fig. 7.4 shows the generated power strips. Similar to conventional designs, the power strip width, pitch and distance can be modified for power network optimization.

ROA-Brick placement/routing blockage generation

In order to avoid short circuit between the ROA-Brick and the synchronous circuits, the ROA-Brick is partially covered with placement/routing blockages. The placement of the placement/routing blockages is performed with Perl script. A portion of the ROA-Brick layout, such as the inverter pairs and the via stacks are covered with placement/routing blockages for specific layers in order to avoid short circuit during placement or routing.



(a) Overview of the placement/routing blockages.

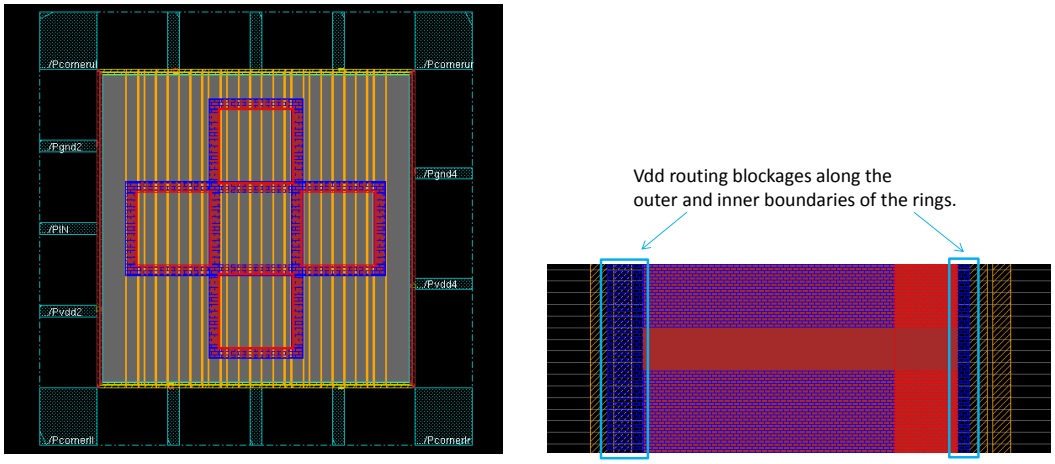
(b) Details of the placement/routing blockages.

Figure 7.5: Placement/Routing blockages for the ROA-Brick.

Power Planning 2: Standard Cell Power Rail Generation

In conventional physical designs, power rails for standard cells are a group of horizontal metal strips. Both ends of the standard cell power rails are required to be connected to power strips to avoid antenna effect. However, in the physical design of ROA-Brick based circuit, in order to avoid short circuit, some standard cell power rails are cut into isolated pieces in horizontal direction. In order to make sure that both ends of the standard cell power rails are connect to power strips, it is required that the standard cell power rails are created only outside and inside the ROA-Brick. As shown in Fig. 7.6(a), one end (or both ends) of the isolated pieces of the standard cell power rails are stopped by the power strips nearest to the ROA-Brick. Here, a few iterations are performed for standard cell power rail placements.

First, routing blockages are place for standard cell VDD rails (left vertical strip) in order to make sure one end (or both ends) of the VDD rails stop at the VDD power strip nearest to the ROA-Brick. Fig. 7.6 shows the routing blockages around the transmission lines of the ROA-Brick, which is placed before standard cell VDD



(a) Overview of the routing blockages for standard cell VDD rails.

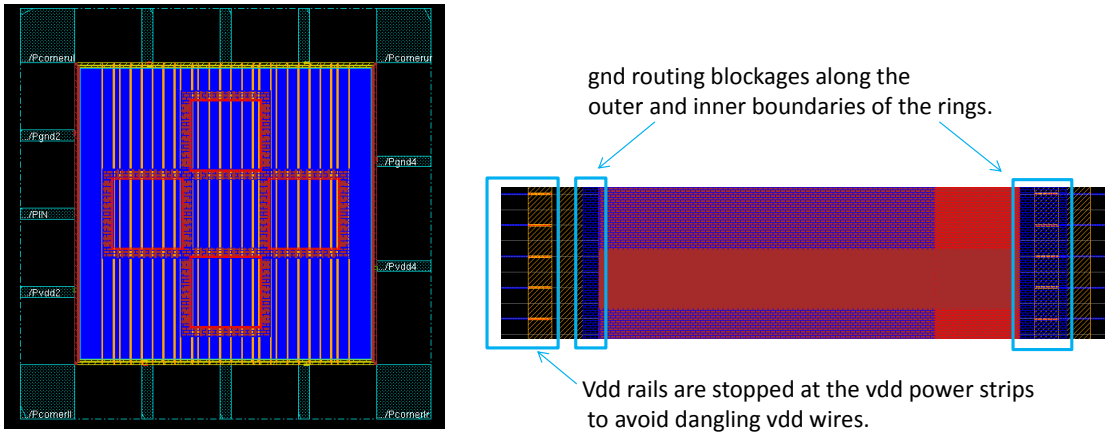
(b) Details of the routing blockages for standard cell VDD rails.

Figure 7.6: Routing blockages for standard cell VDD rails.

rail placement. Therefore, the standard cell VDD rails are stopped right on top on the power strips, which guarantees that there is no dangling power wires in the circuit.

Second, the routing blockages for standard cell VDD rails are removed and the routing blockages for standard cell GND rails are placed. As shown in Fig. 7.7, similar to the routing blockages for standard cell VDD rails, the routing blockages for standard cell VDD rails also control the start/stop location of the rails, which guarantee no dangling wires exist in the circuit.

Third, after standard cell GND rails are placed, the routing blockages are removed. Due to the length of the standard cell VDD/GND rails are not equal in horizontal direction, placement blockages are used in order to avoid the standard cells to be placed in the circled area in Fig. 7.8(b). The circled area has either no VDD or GND rail. If a standard cell is placed in this area, one of its power port would be left floating. It should be noted that the placement blockages cannot be placed before placing the standard cell power rails. Otherwise, vias are not placed for standard cell power rail the power strip connections.



(a) Overview of the routing blockages for standard cell GND rails.

(b) Details of the routing blockages for standard cell GND rails.

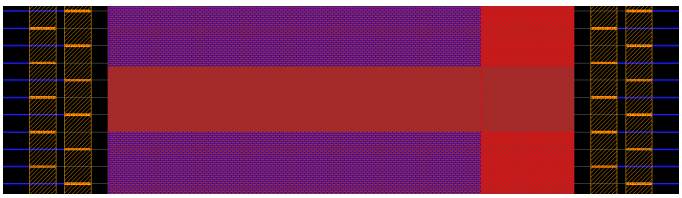
Figure 7.7: Routing blockages for standard cell GND rails.

One-to-one pairing between the tree roots to the tapping points

During the placement step, the dummy cells for multiple clock sources synthesis are manually placed at the tapping point locations of the ROA-Brick prior to the standard cell placement, which is shown in Fig. 7.9. The location calculation and the placement of the dummy cells are completed with perl script based on the physical information obtained from ROA-Brick layout.

Placement, CTS and routing with timing optimizations

The standard cell placement in ROA-Brick based circuit physical design is similar to the one in conventional physical design. Since the ROA-Brick has already been covered with placement blockages, standard cells won't be placed on top of the ROA-Brick. During the CTS step, the outputs of the dummy cells are defined as the clock sources. Since an ROA-Brick is used as the rotary clocking generation circuit, there are four tapping points in this design example. A multi-source CTS is performed with the inter-clock skew being controlled. The routing step in ROA-Brick based circuit



(a) Standard cell power rails.

(b) Placement blockages for standard placement.

Figure 7.8: Finished standard cell power rails and placement blockages for standard cell placement

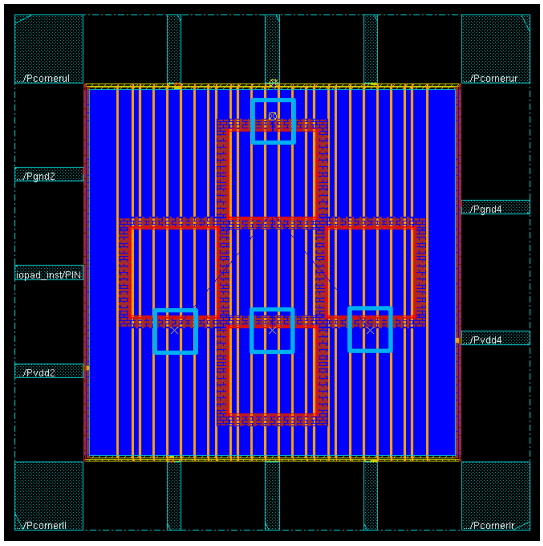


Figure 7.9: The dummy cells are placed at the locations of the tapping points.

physical design is also similar to the one in conventional physical design. Fig. 7.10 shows the circuit after routing.

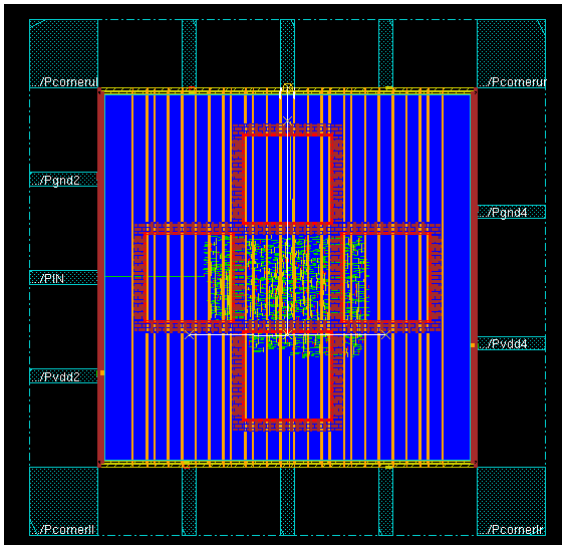


Figure 7.10: The circuit after routing.

7.3 Conclusions

In this chapter, an resonant rotary based physical design flow in proposed, which is developed with mainstream physical design tools. The proposed physical design flow realize the on-chip generation and distribution characteristic of the rotary clock distribution network.

8. Resonant Frequency Divider Design Methodology for Dynamic Frequency Scaling

In this chapter, a dynamic rotary frequency divider design methodology is proposed for dynamic frequency scaling. The proposed resonant dynamic frequency divider uses the multi-phase clock signals as its input. The proposed methodology can be used for designing a divider of any arbitrary integer divisions to be used in dynamic frequency scaling. HSPICE-based experiments are performed to test the electrical characteristics of the (RTWO) ring frequency dividers constructed from the proposed methodology. The simulation results show that the power consumption of a frequency divider is as low as around 5mW for different frequency divisions.

8.1 Introduction

High end System-on-chip (SoC) architectures consist of tens of heterogeneous processing elements and memory units. A Network-on-chip (NoC) provides the infrastructure for these processing elements and memory units to communicate. With increasing system complexity, the traffic flow from each of the processing elements can have different performance requirements [73]. It is demanding that the processing elements can be dynamically switched to work under an affordable set of operating frequencies according to their intermediate workload for power saving. Dynamic frequency scaling (DFS) is a dynamic power management scheme whereby the frequency of a microprocessor can be automatically adjusted according to the current workload of the devices [13, 40, 41]. Most of the DFS schemes are applied to processing elements, where each processing element is regarded as an individual object with its local clock domain.

Resonant clocking, including those based on distributed LC-based oscillators [10, 37], standing wave oscillator (SWO) [2] and rotary traveling wave oscillator (RTWO) [76], is an appealing approach for power reduction in clock network design [2, 10, 12, 21, 42, 76]. The advantage of the resonant clocking-based clock network design is that the resonant system can be utilized as a clock generator as well as a part of the distribution network. This is because the resonant system itself consumes part of the chip area, which can be utilized for clock delivery. In recent years, resonant clocking is also applied to NoC designs. In the study of [47], the resonant clocking is used to generate the clock signals for the NoC and the topology of the resonant system is also used to form the distribution network of the NoC. However, since the resonant-NoC is operating at a much higher speed than the processing elements, asynchronous FIFOs are used to communicate between the NoC and the processing elements. Thus, the low power resonant system only exist in the NoC but not the processing elements. Even if resonant clocking is used on processing elements, it remains unclear how popular DFS-schemes can be used on these processing elements of the SoC. The DFS of resonant clocking is a very promising approach, as it features a very low power clock network employed in the DFS. Unlike that for traditional PLL-based clocking, however, DFS for resonant clocking has previously not been explored. The two challenges of accomplishing a DFS-scheme on resonant rotary clocking, that are addressed in this work, are:

1. The power savings and the stability of the resonant rotary clocking are directly proportional with its operating frequency; thus, lower frequency implementations of rotary clocking are often inefficient. For instance, the resonant clocking implementation on AMD Piledriver chip [57] is only used at high frequencies, and the clock network is multiplexed off to be driven by a non-resonant clock source for lower frequency modes. Simply increasing the capacitance or the in-

ductance to reduce the resonant frequency of the oscillator leads an undesirable increase in power and area overhead.

2. While a frequency division circuitry exists for resonant rotary clocking (e.g. not for coupled LC oscillator-based type that is used in the AMD Piledriver in [57] but for RTWO), the rotary clock frequency divider is a static divider and cannot perform dynamic frequency scaling as targeted in this work for DFS-based architectural design support. On the other hand, traditional non-resonant frequency divider structures are not capable of maintaining the adiabatic switching property of the resonant system, either.

The second challenge listed above refers to the only prior work on resonant clock frequency division, which is a USPTO-filed patent [75]. The patent [75] describes a circuitry for frequency division designed specifically for the ring-based resonant clocking that uses the "spot-advancing" blocks (SABs) driven by multi-phases of the rotary clock. In this chapter, a ring frequency divider design methodology is proposed for *dynamic frequency scaling* based on the structure of the static divider in [75]. Using this methodology, a ring frequency divider with any arbitrary division number can be realized based on one circuit topology. The proposed methodology adds tremendous value to the integration of resonant clocking to system architecture design through enabling DFS of resonant clocking on SoCs.

The chapter is organized as follows: The structure and the operation of the SAB and the ring frequency divider in [75] are reviewed in Section 8.2. The proposed frequency divider design methodology is presented in Section 8.3. The experiment setup and test results are presented in Section 8.4. The conclusions are presented in Section 8.5.

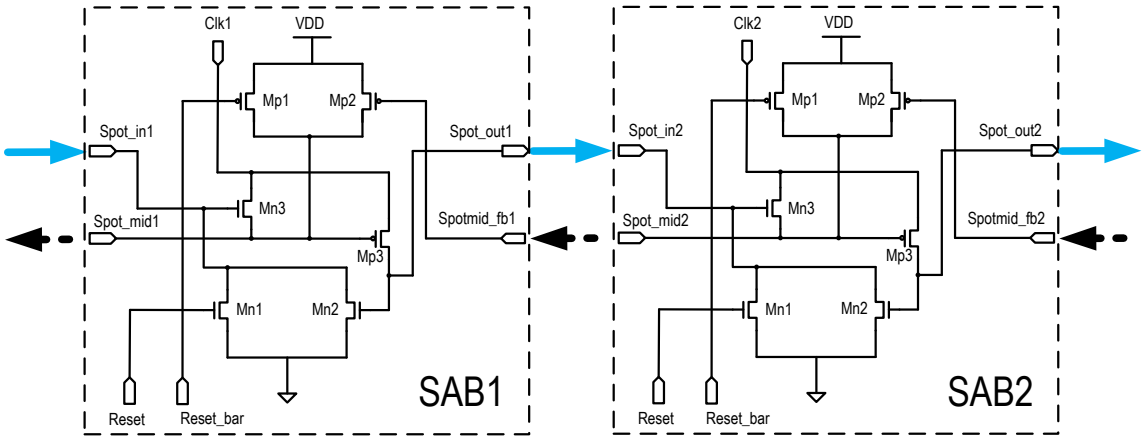


Figure 8.1: Spot-advancing block in [75].

8.2 Spot-advancing Block (SAB) Background

The circuitry of the spot-advancing block (SAB) [75] is shown in Fig. 8.1, which is the core cell in the ring frequency divider. The SAB is composed of 6 transistors. In reset mode, Reset=1/Reset_bar=0, all the Spot_out/Spot_in ports are set to 0 and all the Spotmid_fb/Spot_mid ports are set to 1, which cuts off the paths between Clk and Spot_out. When the operation starts, the SABs advance a 1 value pulse from left to right through the Spot_in port towards the Spot_out port. When Spot_out1 of SAB1 becomes 1, a connection is built between Clk2 and Spot_mid2 in SAB2. When Spot_mid2 is discharged to Clk2 and become 0, a path is formed between Clk2 and Spot_out2 in SAB2 whereas the path between Clk1 and Spot_out1 in SAB1 is closed. Thus, the Spot_out1 of SAB1 is only controlled by SAB2. When the Spot_out2 of SAB2 becomes 1, it drives the Spot_out1 of SAB1 to 0 in order to realize a 1 value pulse forwarding from SAB1 towards SAB2. The width of the 1 value pulse of Spot_out1 in SAB1 is determined by the phase delay of the input clock signals Clk1 and Clk2.

When the SABs are used to construct a frequency divider, all the SABs are connected to form a loop. The topology of a ring frequency divider with a division of 3

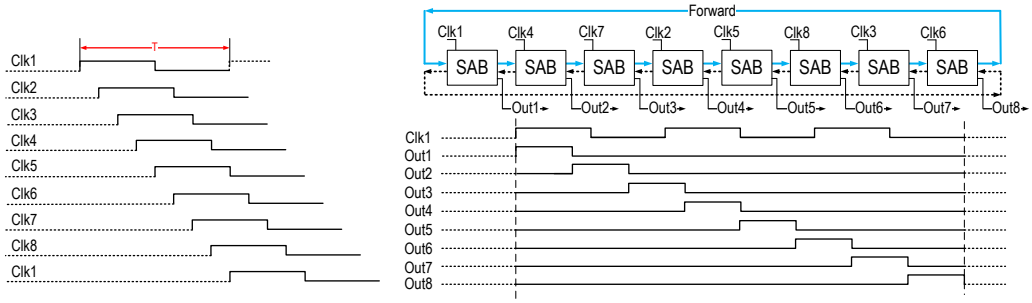


Figure 8.2: 8 SAB frequency divider in [75].

in [75] is shown in Fig. 8.2, which is composed of 8 SABs. Each SAB is assigned with an input clock signal. The input clock signals are of the same frequency and their phases are evenly distributed between 0 to 2π . Here, the number of the input clock signals is set to 8 for illustration purposes. These multi-phase input clock signals are obtained from the ring. Fig. 8.2 shows the locations on the ring off where the multi-phase input clock signals are tapped. The adjacent SABs in the divider circuit loop are connected same as in Fig. 8.1: The Spot_out port of each SAB is connected to the Spot_in port of its downstream neighbor; The Spotmid_fb port is connected to the Spot_mid port of its downstream neighbor. During normal operation, the Spot_in/Spot_out signals are propagating in downstream direction to trigger the operation of the next SAB and the Spotmid_fb/Spot_mid signals are propagating in the opposite direction to erase the operation of the previous SAB. For a frequency division of 3, the multi-phase clock signals for each SAB in the downstream direction are {Clk1, Clk4, Clk7, Clk2, Clk5, Clk8, Clk3, Clk6}. The delay between adjacent SABs is determined by the phase delay of their input clock signals. By applying this input clock sequence, the adjacent Clk signals maintain a phase delay of $\frac{3}{8} * 2\pi$, which cause the phase delay between adjacent SABs as $\frac{3}{8} * 2\pi$. When the Spot_in/Spot_out traverses the loop and comes back to the original SAB, the total phase delay is $8 * \frac{3}{8} * 2\pi = 3 * 2\pi$ so as to divide its input clock signal by 3.

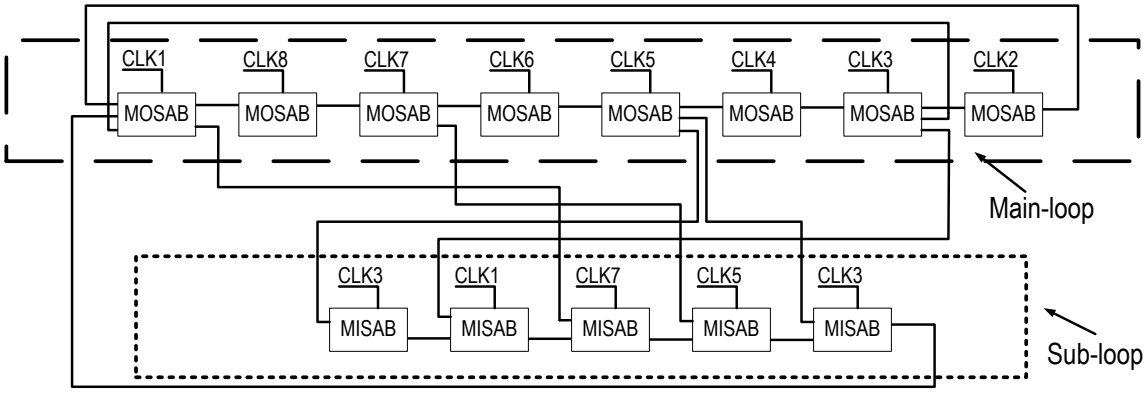


Figure 8.3: Dynamic frequency divider.

8.3 Proposed Dynamic RTWO Frequency Divider Design Methodology

The dynamic RTWO ring frequency divider design methodology proposed in this work is a generalization and extension of the design in [75]. The proposed methodology, in addition to permitting dynamic scaling enhances the practical use of the frequency divider by providing different construction methods for a given division. The proposed design methodology can realize the object division with least number of SABs and it can dynamically tune the output frequency based on SoC-driven architectural requirements of the DFS scheme.

8.3.1 Dynamic RTWO Frequency Divider Circuit Topology

The basic design principle of the dynamic RTWO frequency divider is the ability to reuse the SAB cell when forming dividers of different divisions. In order to accomplish this goal, the circuit structure of the RTWO frequency dividers for different divisions are kept as similar as possible so that the same building blocks can be used to implement different dividers to reduce the design overhead.

The proposed circuit structure of an RTWO frequency divider with division of 3 to 9 is shown in Fig. 8.3. This circuit is a 13 stage circuit topology, grouped into 8 *main*

loop stages and 5 *sub-loop stages* for the 13 stages. Not all of these stages are used for all divisions: A loop is created between the blocks on the main-loop and the blocks on the sub-loop in order to create the desired division. In other words, the division by 3 to 9 is implemented on the 13 stage circuit topology template by controlling the selection of *a circuit loop* construction between all or a subset of the 13 blocks. In the special case of a divisions by 7, all 8 blocks on the main loop are selected as *the circuit loop*, without including any of the sub-loops blocks. The theoretical analysis that drives the selection of the smallest number of connections from the template circuit topology in Fig. 8.3 is presented in Section 8.3.2. Representative designs are presented in Section 8.3.3. Discussion of the circuit topology for divisions larger than a divider of 9 is presented in Section 8.3.4.

8.3.2 Dynamic RTWO Frequency Divider Circuit Design Principle

For power saving purposes, the delays between adjacent SABs are kept as large as possible. Given m number of multi-phase input clock signals that drive the main-loop and sub-loop blocks in Fig. 8.3, the largest phase delay between the clock signals is $\frac{m-1}{m} * 2\pi$. The blocks in the main-loop are driven by clock phases that are separated by this largest delay of $\frac{m-1}{m} * 2\pi$. The phase delays of the sub-block loop is methodically selected to be the second largest phase delay for an m phase clock signal, which is $\frac{m-2}{m} * 2\pi$. The phase delay of a connection from a block in the main-loop to a block in the sub-loop (e.g. staircase connections in Fig. 8.3) is also the second largest phase delay for an m phase clock signal, which is $\frac{m-2}{m} * 2\pi$. Given that the main-loop to sub-loop connections also incur delay, and some divisions do not require main-loop to sub-loop connections (for instance division by 7), the main design parameter for a division is selected as the delay of connections between SAB blocks, rather than the number of SAB blocks in the main and sub-block loops. In other words, the smallest

implementation is defined by the smallest number of connections between the SAB blocks as opposed to the implementation with the smallest number of SAB blocks. Let n_1 be the number of SAB block to SAB block connections with the largest phase delay for an m phase clock signal, $\frac{m-1}{m} * 2\pi$, occurring exclusively in the main-loop. Similarly, let n_2 be the number of SAB block to SAB block connections with the second largest phase delay for an m phase clock signal, $\frac{m-2}{m} * 2\pi$, occurring at the sub-loop and at the main-loop to sub-loop connections. For notational convenience, let such a proposed frequency divider be defined as an (n_1, n_2)-divider. The design objective is to select the smallest ($n_1 + n_2$) number of connections on the design topology template in Fig. 8.3.

In what follows, the number of multi-phase clock input signals $m = 8$ is used for explanation and this number is also used in Section 8.4 to facilitate the simulations. When $m = 8$, the phase delays between adjacent SABs in the circuit loop are $\frac{7}{8} * 2\pi$ and $\frac{6}{8} * 2\pi$. The design principles of building a division by r is to select the smallest sum of the number of largest phase delay connections n_1 and second largest phase delay connections n_2. Mathematically, the total phase delay provided by the frequency divider can be expressed as:

$$n_1 * (\frac{7}{8}) * 2\pi + n_2 * (\frac{6}{8}) * 2\pi = r * 2\pi \tag{8.1}$$

Thus, the relationship between n_1, n_2 and r is:

$$7 * n_1 + 6 * n_2 = 8 * r \tag{8.2}$$

Eq. 8.2 indicates how n_1 and n_2 are correlated with the division value r. Note that the implementation with the smallest number of n_1 and n_2 are desirable for power saving purposes and area reduction. Lemma 7 describes the smallest solution:

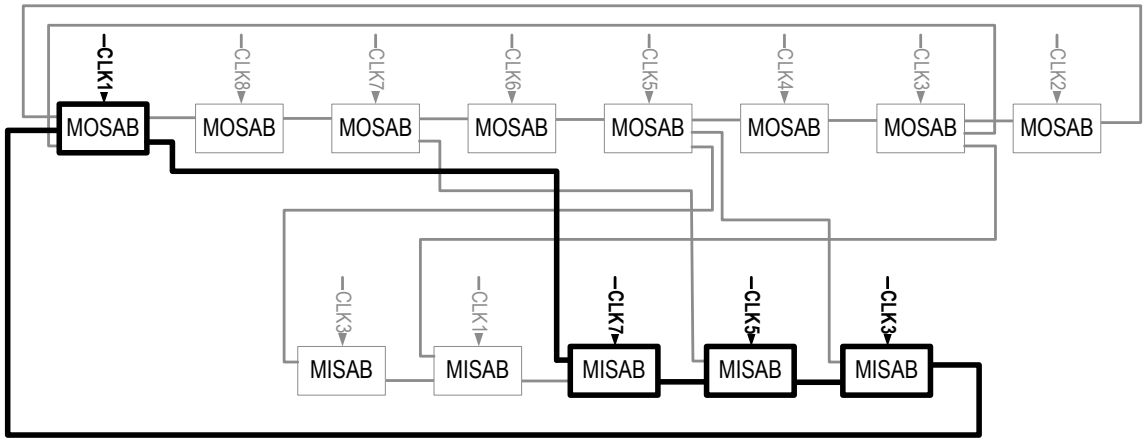
Lemma 7. *For smallest number phase delay connections* $(n_1 + n_2)$ *to be used in division by* r *on the proposed circuitry in Fig. 8.3,* n_2 *should be constrained by* $n_2 \leq 6$.

Proof. The proof is through contradiction. If $n2 \geq 7$, the frequency divider can always be turned into a $(n_1 + 6, n_2 - 7)$-divider. The $(n_1 + 6, n_2 - 7)$-divider only requires $n_1 + n_2 - 1$ blocks to construct a frequency divider, which is better for power savings than the original one. □

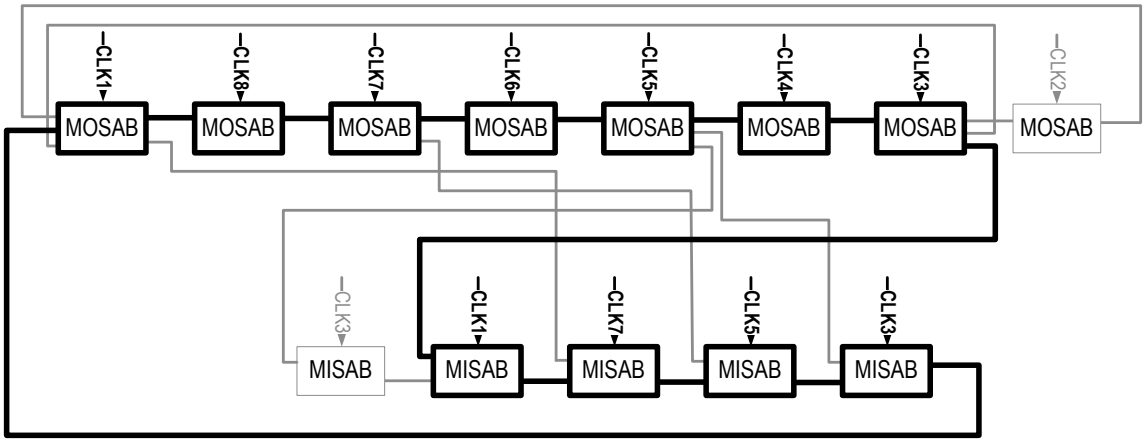
For instance, for dividers with division number $r \geq 5$, the circuit loop of dividers are composed of no less than 6 blocks, which can easily be demonstrated to follow the $n_2 \leq 6$ rule. For $r = 4$, a (2, 3)-divider is used to facilitate the design and for $r = 3$, a (0, 4)-divider is used to facilitate the design.

8.3.3 Dynamic RTWO Frequency Divider Circuit Implementation

The main-loop and sub-loop stages are created by blocks of circuitry labeled MOSAB and MISAB, respectively, in Fig. 8.3 that is proposed in this work. The original SAB is modified into the multi-output SAB (MOSAB) and the multi-input SAB (MISAB), both of which are composed of a SAB and a 4-to-1 multiplexer. The multiplexer is used to provide the dynamic frequency tuning capability to the static circuit just as described. By changing the control inputs of the multiplexers, the connections among the MOSABs in the main-loop and the MISABs in the sub-loop can be varied dynamically to facilitate the design of different division numbers. In addition, one or two of the channels of the multiplexer are designed to temporarily reset its SAB in order to save power when the SAB is not used in the divider loop. Since there is no more than three multi-input/multi-output of one stage in this topology, the 4-to-1 multiplexer can be used to realize different usage by making different circuit connections.



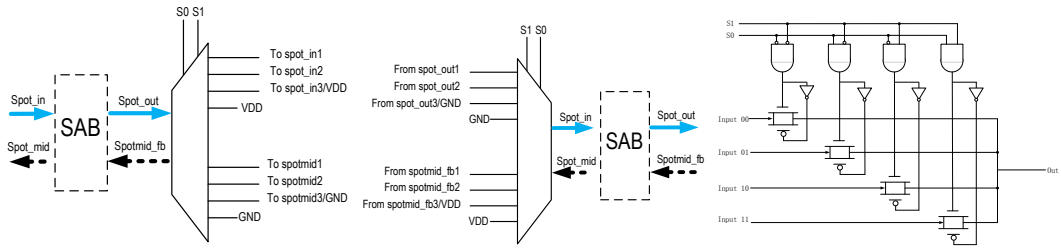
(a) Realization of division r=3 with $n_1 = 0$ and $n_2 = 4$



(b) Realization of division r=9 with $n_1 = 6$ and $n_2 = 5$

Figure 8.4: Dynamic RTWO frequency divider.

The topology shown in Fig. 8.3 is composed of two parts: (1) The main-loop is composed of 8 MOSABs with $\frac{7}{8} * 2\pi$ phase delay between adjacent stages. (2) The sub-loop is composed of 5 MISABs with $\frac{6}{8} * 2\pi$ phase delay between adjacent stages. There are 5 connections between the MOSABs in the main-loop and the MISABs in the sub-loop. The phase delay between each MOSAB and MISAB connection is also $\frac{6}{8} * 2\pi$. In Fig. 8.4, two RTWO frequency dividers are with division $r = 3$ and $r = 9$ accomplished by connecting the loop differently. For $r = 3$, as shown in Fig. 8.4(a), the phase delays between adjacent SABs are all $\frac{6}{8} * 2\pi$ as such the



(a) Multi-output SAB (MOSAB) (b) Multi-input SAB (MISAB) (c) 4-to-1 multiplexer

Figure 8.5: The basic building blocks of the RTWO frequency divider.

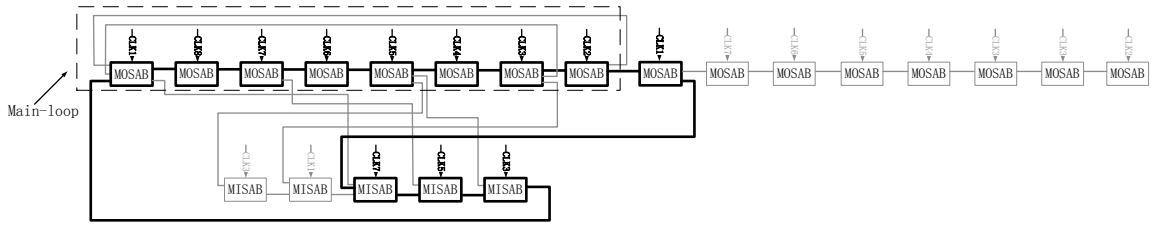


Figure 8.6: Circuit topolgy of dynamic frequency divider with r=10 with $n_1 = 8$ and $n_2 = 4$.

divider is a (0, 4) - divider. On the other hand, for $r = 9$, the divider is built as a (6, 5) - divider.

8.3.4 Dynamic RTWO Frequency Divider with Division $r > 9$

When building frequency dividers with division numbers $r > 9$, the frequency divider circuit topology in Fig. 8.3 is used as the main building block with MOSABs appended to the main-loop. Here, it is required that the first MOSAB on the left of the main loop is used in the circuit loop of every division number. Thus, the output signal of this MOSAB can be used as the output of the frequency divider of any division number. Fig. 8.6 shows the circuit topology for the RTWO frequency divider with division $r = 10$. In this circuit topology, one MOSAB is append to the main-loop so that there are 9 MOSABs and 4 MISABs in the circuit loop. It is observed that

the circuit topology is quite similar to the $r = 3$ divider circuit by adding 8 MOSABs in the main-loop. Similarly, for $r = 11$, the circuit connection is similar to $r = 4$ by adding 8 MOSABs in the main-loop. As a conclusion, frequency divider with division number $r > 9$, can always be realized from the circuit connection of the frequency divider with division $r\%7$ by appending MOSABs in the main-loop.

8.4 Experiments

In order to test the functionality of the RTWO-based frequency divider, a 90nm technology is used to facilitate the HSPICE simulations. The following experiments are performed in the following two aspects: (1) The dynamic switching between different division numbers; discussed in Section 8.4.1 (2) The power consumption of the frequency divider when building dividers with different division numbers; discussed in Section 8.4.2.

8.4.1 Dynamically Tune the Division Number

In this section, the dynamic frequency switching feature of the RTWO frequency divider is discussed and demonstrated. The frequency divider is designed to switch from division r=5 to r=9. The simulation result is shown in Fig. 8.7. As shown in the plot, the signal shown on top is the switch signal, which is triggered around 15ns. The second waveform shows the input oscillation signal generated from the ring, which is around 2.5GHz. The third waveform shows the operation of the reset signal. In order to guarantee the functionality of the circuit after switching, the reset signal is turned on when switching the circuit connection. The waveform on bottom shows the output signal of the RTWO frequency divider. It is shown that before switching, the output of frequency divider is a divided by 5 oscillation signal compared to the RTWO ring oscillation signal. On the other hand, after switching, the output of the

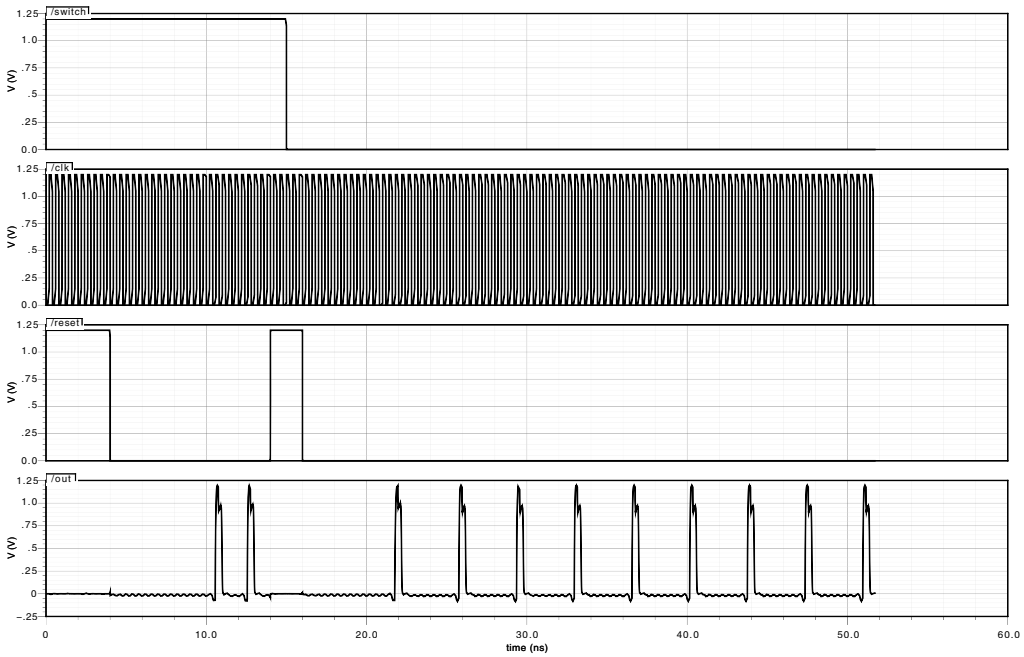


Figure 8.7: Dynamic switching the division between 5 and 9.

frequency divider changes to a divided by 9 oscillation signal compared to the RTWO ring oscillation signal. The duty cycle of the output signal is decided by the largest available phase delay ($\frac{7}{8} * 2\pi$ in this case) delay of the input signals and it is suitable for a edge triggered synchronous circuit designs.

8.4.2 Performance of RTWO Frequency Division from the Same Core Frequency

In this section, the performance of the RTWO frequency division from an arbitrarily selected core frequency of $2.5GHz$ is investigated through HSPICE simulations. The frequency dividers with division r=3 to r=9 are tested. Eight (8) multi-phase clock signals are tapped off from the RTWO to be used as the multi-phase input signals. The rising time of the input signals is 5% of the clock cycle. Fig. 8.8 shows the power consumption of 7 RTWO frequency dividers, respectively.

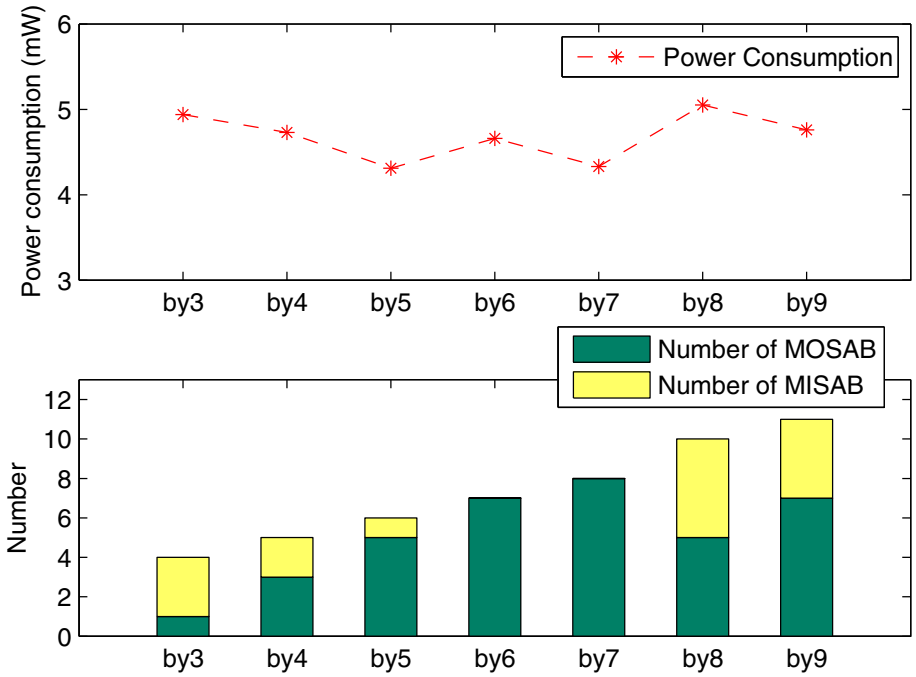


Figure 8.8: Power consumption of the RTWO frequency dividers with division 3-9.

The line plot in Fig. 8.8 shows the power consumption of the frequency dividers with divisions from 3 to 9. The frequency dividers maintain a steady power consumption between $4.31mW$ to $5.05mW$. The number of MISABs and MOSABs that are used in each frequency divider is shown in the bar chart. As shown in the figure, the sum of MISABs and MOSABs used in each frequency divider increases monotonically along with increasing of the division because the maximum phase delay is kept between adjacent MISABs/MOSABs in each frequency divider loop. Along with increasing the division number, the output frequency decreases and the number of MOSABs and MISABs increases. The changing of these two factors neutralizes their effects on the divider power consumption so as to increase the tuning range of the output frequency without increasing the power consumption.

High leakage current in deep sub-micron regimes has been a significant contributor to power dissipation. In order to investigate the leakage power dissipation of the

RTWO frequency divider, the leakage current is tested under the condition that the circuit is turned into reset mode and all the transistors in the circuit topology are shut off. Simulation results show that the leakage power consumption is 15.7μW, which contributes to a negligible 0.3% of the power consumption of the RTWO frequency divider. Thus, this topology does not exacerbate the known-to-be-favorable leakage power profile of rotary clocking at the divided frequencies and the rotary technology remains very power efficient at dynamically switched low frequencies; a feature previously considered unattainable.

8.5 Conclusions

In this chapter, a dynamic frequency divider design methodology is proposed for dynamic frequency scaling of resonant rotary clocking in modern processors. The proposed circuit topology can be used to design a dynamic RTWO frequency divider with any arbitrary division numbers. The power consumption of the proposed topology is designed to be as low as around 5.05mW stably when tuning the output frequency and the leakage power dissipation is 15.7μW when the circuit is turned into reset mode, which is very power efficient for low power digital circuit design.

9. Conclusions and Future Directions

Rotary clocking technology provides an attractive alternative to the traditional clocking schemes for the generation of high-frequency, low-skew clock signals with low power dissipation. In this dissertation, (1) circuit topologies and design methodologies are proposed to facilitate the physical design and optimization of the rotary clock distribution network, which are summarized in Section 9.1 and 9.2; (2) a physical design flow is proposed for rotary clocking based circuits, which is summarized in Section 9.3; (3) circuit topologies are proposed for rotary dynamic frequency scaling, which are summarized in Section 9.4. Future directions are discussed in Section 9.5.

9.1 Conclusions on Topology Related Work

In Chapter 3, an ROA topology—ROA-Brick—is proposed, which possesses the property that all the rings in the ROA-Brick maintain the same signal rotation direction. The signal rotation direction uniformity guarantees that the same phase points on each ring are predictable. Further, mathematical proofs are provided to demonstrate that the ROA-Brick is the unique ROA topology which possesses this feature. This is the first work that solves the ROA signal rotation direction uniformity problem, which is critical for low-skew resonant rotary clock distribution network design. This work is published in [62].

In Chapter 4, a synchronization scheme is proposed for the brick-based ROA. The synchronization scheme can help the brick-based ROA quickly reach frequency synchronization and signal rotation direction uniformity. Simulation results are encouraging in providing the feasibility of using the brick-based ROA as the global network of the rotary clock distribution network. This work is published in [65].

9.2 Conclusions on Rotary Clocking Distribution Network Design and Optimization Related Work

In Chapter 4, a brick-based ROA clock distribution network design methodology is proposed, in which the brick-based ROA is used as the global network. In addition, a tapping point location optimization methodology is proposed for clock skew reduction. This is the first work that optimizes the tapping point locations for clock network skew optimization. HSPICE simulation results show that the global skew can be reduced by 71.72%. This work is submitted to [64].

In Chapter 5, a topology and placement optimization methodology is proposed for brick-based ROA. The proposed methodology optimizes a full-meshed brick-based ROA into a sparse-ROA based on the distribution of the synchronous components. The proposed methodology leads to a power reduction of 32.1% and a skew reduction of 47.1% compared to the design results with a full-mesh ROA. This is the first work that performs the global network optimization for rotary clock distribution network. In addition, this work greatly improves the efficiency of the rotary clock distribution network design, for the global network placement and topology optimization are performed concurrently. This work is published in [68].

In Chapter 6, a frequency-centric rotary clock distribution network design methodology is proposed. Given a target frequency, the physical structure of the brick-based ROA is optimized by considering the sub-network tree capacitance. Furthermore, a simplification model is proposed to substitute the whole resonant system for brick-based ROA physical structure optimization. HSPICE simulation results show that the frequency difference between the proposed simplification model and the resonant network is less than 0.20% and the run-time is reduced by approximately 70%. This the first work that embeds the frequency accuracy methodology into the rotary clock distribution network designs. This work is accepted for publication in [63].

9.3 Conclusions on Physical Design Flow for Rotary Clocking Based Circuits

In Chapter 7, a complete physical design flow is proposed for the rotary clocking based circuit with mainstream design automation tools. The proposed physical design flow adopts the frequency-centric rotary clock distribution network design methodology proposed in Chapter 6 and realizes on-chip clock generation and distribution. This is the first design automation tool, which facilitates the physical design of the rotary clocking based circuits. The proposed physical design flow can be seamlessly integrated with the industrial physical design flow, which demonstrates the high practicality of the proposed methods.

9.4 Conclusions on Resonant Dynamic Frequency Scaling

In Chapter 8, a rotary dynamic frequency divider is proposed for resonant dynamic frequency scaling designs. The proposed rotary frequency divider circuit topology can be used to generate arbitrary division numbers greater than 2. Simulation results show that the power consumption of a resonant frequency divider is as low as around $5mW$ for different frequency divisions. This is the first work to apply rotary clocking in resonant dynamic frequency scaling, which greatly extends the operating frequency of rotary clocking for low power designs. This work is published in [66, 67].

9.5 Future Directions

As an extension to the completed work on the resonant clocking technologies, the future directions are presented in different sections as follows.

9.5.1 Fabrication of the Rotary Clocking Based Circuits

The most important work to be completed is to fabricate the rotary clocking based circuits and verify its performance. Methodologies for on-chip frequency measurements need to be investigated in order to address the following aspects in back-end measurements:

- Single ring/brick-based ROA frequency mismatch between circuit simulation and chip measurements.
- The impact of loading capacitance (from circuits and measurement tools) on the oscillation frequency deviation.
- The impact of the ring physical parameters on the oscillation signal transition time and amplitude, as well as the resonant oscillator power dissipation.
- The impact of the technology scaling on rotary ring performance.

9.5.2 On-chip Variations Analysis for Frequency-Centric Rotary Clock Distribution Network

The rotary clock distribution network is composed of global, regional and local networks. The global skews usually comes from the sub-network trees and the tapping wire difference between each tapping point and root pair. In this dissertation, the frequency-centric rotary clock distribution network design is demonstrated with high frequency accuracy. As part of the future work, a comprehensive analysis should be performed to verify the robustness of rotary clock distribution network under on-chip variations.

Bibliography

[1] P. Andreani and S. Mattisson, "On the use of mos varactors in rf vco's," *IEEE J. Solid-State Circuits*, vol. 35, no. 6, pp. 905–910, June 2000.

[2] W. Andress and D. Ham, "Standing wave oscillators utilizing wave-adaptive tapered transmission lines," *IEEE J. Solid-State Circuits*, vol. 40, no. 5, pp. 638–651, Mar. 2005.

[3] K. Boese and A. Kahng, "Zero-skew clock routing trees with minimum wirelength," in *Proc. IEEE IASICCE'92*, Sept. 1992, pp. 17–21.

[4] D. Cai, H. Fu, J. Ren, W. Li, N. Li, H. Yu, and K. Yeo, "A 2.1-ghz pll with -80dbc/-74dbc reference spur based on aperture-phase detector and phase-to-analog converter," in *Proc. IEEE A-SSCC'11*, Nov. 2011, pp. 141–144.

[5] P. Chakrabarti, V. Bhatt, D. Hill, and A. Cao, "Clock mesh framework," in *Proc. IEEE ISQED'12*, Mar. 2012, pp. 424–431.

[6] S. Chan, K. Shepard, and P. Restle, "Design of resonant global clock distributions," in *Proc. IEEE ICCD'03*, Oct. 2003, pp. 248–253.

[7] ——, "1.1 to 1.6ghz distributed differential oscillator global clock network," in *Proc. IEEE ISSCC'05*, Feb. 2005, pp. 518–519.

[8] ——, "Uniform-phase uniform-amplitude resonant-load global clock distributions," *IEEE J. Solid-State Circuits*, vol. 40, no. 1, pp. 102–109, Jan. 2005.

[9] ——, "Distributed differential oscillators for global clock networks," *IEEE J. Solid-State Circuits*, vol. 41, no. 9, pp. 2083–2094, Sept. 2006.

[10] S. C. Chan, P. J. Restle, K. L. Shepard, N. K. James, and R. L. Franch, "A 4.6ghz resonant global clock distribution network," in *Proc. IEEE ISSCC'04*, Feb. 2004, pp. 342–343.

[11] T. Chao, Y. Hsu, J. Ho, and A. Kahng, "Zero skew clock routing with minimum wirelength," *IEEE Trans. Circuits Syst. II*, vol. 39, no. 11, pp. 799–814, Nov. 1992.

[12] V. Chi, "Salphasic distribution of clock signals for synchronous systems," *IEEE Trans. Comput.*, vol. 43, no. 5, pp. 597–602, May 1994.

[13] K. Choi, R. Soma, and M. Pedram, "Off-chip latency-driven dynamic voltage and frequency scaling for an mpeg decoding," in *Proc. ACM/IEEE DAC'04*, 2004, pp. 544–549.

[14] S. Cohen and L. Guibas, "The earth mover's distance under transformation sets," in *Proc. IEEE ICCV'99*, Sept. 1999, pp. 1076–1083.

[15] J. Cong, A. Kahng, C. Koh, and C. Tsao, "Bounded-skew clock and steiner routing," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 3, no. 3, pp. 341–388, July 1998.

[16] V. H. Cordero and S. P. Khatri, "Clock distribution scheme using coplanar transmission lines," in *Proc. DATE'08*, Mar. 2008, pp. 985–990.

[17] T. Cormen, C. Leiserson, R. Rivest, and C. Stein, Eds., *Introduction to Algorithms*. New York: The MIT Press., 2009.

[18] M. Desai, R. Cvijetic, and J. Jensen, "Sizing of clock distribution networks for high performance cpu chips," in *Proc. ACM/IEEE DAC'96*, June 1996, pp. 389–394.

[19] A. Deutsch and all, "Modeling and characterization of long on-chip interconnections for high-performance microprocessors," *IBM J. Res. Develop.*, vol. 39, no. 5, pp. 547–567, Sept. 1995.

[20] L. Divina and Z. Skvor, "The distributed oscillator at 4 ghz," *IEEE Trans. Microw. Theory Tech.*, vol. 46, no. 12, pp. 2240–2243, Dec. 1998.

[21] A. Drake, K. Nowka, T. Nguyen, J. Burns, and R. Brown, "Resonant clocking using distributed parasitic capacitance," *IEEE J. Solid-State Circuits*, vol. 39, no. 9, pp. 1520–1528, Sept. 2004.

[22] M. Edahiro, "A clustering-based optimization algorithm in zero-skew routings," in *Proc. ACM/IEEE DAC'93*, June 1993, pp. 612–616.

[23] E. Friedman, "Clock distribution networks in synchronous digital integrated circuits," *Proceedings of the IEEE*, vol. 89, no. 5, pp. 665–692, May 2001.

[24] M. Guthaus, "Distributed lc resonant clock tree synthesis," in *Proc. IEEE ISCAS'11*, May 2011, pp. 1215–1218.

[25] M. Guthaus, G. Wilke, and R. Reis, "Non-uniform clock mesh optimization with linear programming buffer insertion," in *Proc. ACM/IEEE DAC'10*, June 2010, pp. 74–79.

[26] R. He, W. Xu, J. Sun, and B. Zu, "Balanced k-means algorithm for partitioning areas in large-scale vehicle routing problem," in *Proc. IEEE IITA'09*, Nov. 2009, pp. 87–90.

[27] V. Honkote, "Design automation and analysis of resonant rotary clocking technology," in *Proc. IEEE ISVLSI'10*, July 2010, pp. 471–472.

[28] V. Honkote and B. Taskin, "Custom rotary clock router," in *Proc. IEEE ICCD'08*, Oct. 2008, pp. 114–119.

[29] ——, "Capacitive load balancing for mobius implementation of standing wave oscillator," in *Proc. IEEE MWSCAS'09*, Aug. 2009, pp. 232–235.

[30] ——, "Skew analysis and design methodologies for improved performance of resonant clocking," in *Proc. IEEE ISOCC'09*, Nov. 2009, pp. 165–168.

[31] ——, "Zero clock skew synchronization with rotary clocking technology," in *Proc. IEEE ISQED'09*, Mar. 2009, pp. 588–593.

[32] ——, "Analysis, design and simulation of capacitive load balanced rotary oscillatory array," in *Proc. IEEE VLSID'10*, Jan. 2010, pp. 218–223.

[33] ——, "CROA: Design and analysis of the custom rotary oscillatory array," *IEEE Trans. VLSI Syst.*, vol. 19, no. 10, pp. 1–11, Oct. 2010.

[34] ——, "Zeroa: Zero clock skew rotary oscillatory array," *IEEE Trans. VLSI Syst.*, vol. 20, no. 8, pp. 1528–1532, Aug. 2012.

[35] C. Hsu, K. Tripurari, S. Yu, and P. Kinget, "A 2.2ghz pll using a phase-frequency detector with an auxiliary sub-sampling phase detector for in-band noise suppression," in *Proc. IEEE CICC'11*, Sept. 2011, pp. 1–4.

[36] X. Hu, W. Condley, and M. Guthaus, "Library-aware resonant clock synthesis (larcs)," in *Proc. ACM/IEEE DAC'12*, June 2012, pp. 145–150.

[37] X. Hu and M. Guthaus, "Distributed resonant clock grid synthesis (rocks)," in *Proc. ACM/IEEE DAC'11*, June 2011, pp. 516–521.

[38] A. Ishii, J. Kao, V. Sathe, and M. Papaefthymiou, "A resonant-clock 200MHz ARM926EJ-STM microcontroller," in *Proc. ESSCIRC'09*, Sept. 2009, pp. 356–359.

[39] V. Karkala, K. Bollapalli, R. Garg, and S. Khatri, "A PLL design based on a standing wave resonant oscillator," in *Proc. IEEE ICCD'09*, Oct. 2009, pp. 511–516.

[40] B. Khargharia, S. Hariri, and M. Yousif, "Autonomic power and performance management for computing systems," *Cluster Computing*, vol. 11, no. 2, pp. 167–181, 2008.

[41] C. Krishna and Y. Lee, "Voltage-clock-scaling adaptive scheduling techniques for low power in hard real-time systems," *IEEE Transactions on Computers*, vol. 52, no. 12, pp. 1586–1593, 2003.

[42] J. Lu, V. Honkote, X. Chen, and B. Taskin, "Steiner tree based rotary clock routing with bounded skew and capacitive load balancing," in *Proc. DATE'11*, Mar. 2011, pp. 455–460.

[43] T. Lu and W. Chen, "A 38/114 ghz switched-mode and synchronous lock standing wave oscillator," *IEEE Microw. Wireless Compon. Lett.*, vol. 21, no. 1, pp. 40–42, Jan. 2011.

[44] E. D. Man and M. Schobinger, "Power dissipation in the clock system of highly pipelined ulsi cmos circuits," in *Proc. IEEE IWLPD'94*, Apr. 1994, pp. 133–138.

[45] A. Mandal, K. Bollapalli, N. Jayakumar, S. Khatri, and R. Mahaptra, "A low-jitter phase-locked resonant clock generation and distribution scheme," in *Proc. IEEE ICCD'13*, Oct. 2013, pp. 487–490.

[46] A. Mandal, V. Karkala, S. Khatri, and R. Mahapatra, "Interconnected tile standing wave resonant oscillator based clock distribution circuits," in *Proc. IEEE VLSID'11*, Jan. 2011, pp. 82–87.

[47] A. Mandal, S. Khatri, and R. Mahapatra, "A fast, source-synchronous ring-based network-on-chip design," in *Proc. DATE'12*, 2012, pp. 1489–1494.

[48] J. Munkres, "Algorithms for the assignment and transportation problems," *Journal of the Society for Industrial and Applied Mathematics (J.SIAM)*, vol. 5, no. 1, pp. 32–38, Mar. 1957.

[49] S. Naffziger and G. Hammond, "The implementation of the next-generation 64b itanium microprocessor," in *Proc. IEEE ISSCC'02*, Feb. 2002, pp. 276–504.

[50] J. Neves and E. Friedman, "Minimizing power dissipation in non-zero skew-based clock distribution networks," in *Proc. IEEE ISCAS'95*, Apr. 1995, pp. 1576–1579.

[51] F. O'Mahony, C. Yue, M. Horowitz, and S. Wong, "A 10-GHz global clock distribution using coupled standing-wave oscillators," *IEEE J. Solid-State Circuits*, vol. 38, no. 11, pp. 1813–1820, Nov. 2003.

[52] ——, "Design of a 10GHz clock distribution network using coupled standing-wave oscillators," in *Proc. ACM/IEEE DAC'03*, June 2003, pp. 682–687.

[53] A. Raghunathan, N. Jha, and S. Dey, Eds., *High-Level Power Analysis and Optimization*. MA: Kluwer: Springer, 1997.

[54] A. Rajaram and D. Pan, "Meshworks: An efficient framework for planning, synthesis and optimization of clock mesh networks," in *Proc. ACM/IEEE ASP-DAC'08*, Mar. 2008, pp. 250–257.

[55] P. Restle, T. McNamara, D. Webber, P. Camporese, K. Eng, K. Jenkins, D. Allen, M. Rohn, M. Quaranta, D. Boerstler, C. Alpert, C. Carter, R. Bailey, J. Petrovick, B. Krauter, and B. McCredie, "A clock distribution network for microprocessors," *IEEE J. Solid-State Circuits*, vol. 36, no. 5, pp. 792–799, May 2001.

[56] Y. Rubner, C. Tomasi, and L. Guibas, "The earth mover's distance as a metric for image retrieval," *International Journal of Computer Vision*, vol. 40, pp. 1–20, 2000.

[57] V. Sathe, S. Arekapudi, C. Ouyang, M. Papaefthymiou, A. Ishii, and S. Naffziger, "Resonant clock design for a power-efficient high-volume x86-64 microprocessor," in *Proc. IEEE ISSCC'12*, Feb. 2012, pp. 68–70.

[58] K. Takinami, R. Strandberg, P. Liang, G. Mercey, T. Wong, and M.Hassibi, "A rotary-traveling-wave-oscillator-based all-digital pll with a 32-phase embedded phase-to-digital converter in 65nm cmos," in *Proc. IEEE ISSCC'11*, Feb. 2011, pp. 100–102.

[59] K. Takinami and R. Walsworth, "Phase error calibration technique for rotary traveling wave oscillators," *IEEE J. Solid-State Circuits*, vol. 45, no. 11, pp. 2433–2444, Nov. 2010.

[60] K. Takinami, R. Walsworth, S. Osman, and S. Beccue, "Phase-noise analysis in rotary traveling-wave oscillators using simple physical model," *IEEE Trans. Microw. Theory Tech.*, vol. 58, no. 6, pp. 1465–1474, June 2010.

[61] B. Taskin, J. Demaio, O. Farell, M. Hazeltine, and R. Ketner, "Custom topology rotary clock router with tree subnetworks," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 14, no. 3, pp. 44:1–44:14, June 2009.

[62] Y. Teng, J. Lu, and B. Taskin, "Roa-brick topology for rotary resonant clocks," in *Proc. IEEE ICCD'11*, Oct. 2011, pp. 273–278.

[63] Y. Teng and B. Taskin, "Frequency-centric resonant rotary clock distribution network design," in *Proc. ACM/IEEE ICCAD'14 (to appear)*.

[64] ——, "Roa-brick topology for low-skew rotary resonant clock network design (in review)."

[65] ——, "Synchronization scheme for brick-based rotary oscillator arrays," in *Proc. ACM GLSVLSI'12*, May 2012, pp. 117–122.

[66] ——, "Resonant frequency divider design methodology for dynamic frequency scaling," in *Proc. IEEE ICCD'13*, Oct. 2013, pp. 479–482.

[67] ——, "Rotary traveling wave oscillator frequency division at nanoscale technologies," in *Proc. ACM GLSVLSI'13*, May 2013, pp. 349–350.

[68] ——, "Sparse-rotary oscillator array (sroa) design for power and skew reduction," in *Proc. DATE'13*, Mar. 2013, pp. 1229–1234.

[69] R. Tsay, "Exact zero skew," in *Proc. ACM/IEEE ICCAD'91*, Nov. 1991, pp. 336–339.

[70] G. Venkataraman, Z. Feng, J. Hu, and P. Li, "Combinatorial algorithms for fast clock mesh optimization," *IEEE Trans. VLSI Syst.*, vol. 18, no. 1, pp. 131–141, Jan. 2010.

[71] G. Venkataraman, J. Hu, F. Liu, and C. Sze, "Integrated placement and skew optimization for rotary clocking," in *Proc. DATE'06*, Mar. 2006, pp. 1–6.

[72] V.Honkote and B.Taskin, "Skew-aware capacitive load balancing for low-power zero clock skew rotary oscillatory array," in *Proc. IEEE ICCD'99*, Oct. 2010, pp. 209–214.

[73] W. Wolf, A. Jerraya, and G. Martin, "Multiprocessor system-on-chip MPSoC technology," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 27, no. 10, pp. 1701–1713, 2008.

[74] J. Wood, "Electronic circuits," Oct. 13 2005, uS Patent App. 10/504,559. [Online]. Available: http://www.google.com/patents/US20050225365

[75] ——, "Frequency divider," Oct. 12 2010, uS Patent 7,812,648. [Online]. Available: http://www.google.com/patents/US7812648

[76] J. Wood, T. Edwards, and S. Lipa, "Rotary traveling-wave oscillator arrays: A new clock technology," *IEEE J. Solid-State Circuits*, vol. 36, no. 11, pp. 1654–1665, Nov. 2001.

[77] J. Wood, S. Lipa, P. Franzon, and M. Steer, "Multi-gigahertz low-power low-skew rotary clock scheme," in *Proc. IEEE ISSCC'01*, Feb. 2001, pp. 400–401.

[78] Z. Yu, "Rotary clock based high-frequency asic design methodology," Ph.D. dissertation, North Carolina State University, Raleigh, NC, 2007.

[79] Z. Yu and X. Liu, "Power analysis of rotary clock," in *Proc. IEEE ISVLSI'05*, May 2005, pp. 150–155.

[80] ——, "Design of rotary clock based circuits," in *Proc. ACM/IEEE DAC'07*, June 2007, pp. 43–48.

[81] ——, "Low-power rotary clock array design," *IEEE Trans. VLSI Syst.*, vol. 15, no. 1, pp. 5–12, Jan. 2007.

[82] Y. Zhang, J. F. Buckwalter, and C. Cheng, "On-chip global clock distribution using directional rotary traveling-wave oscillator," in *Proc. IEEE EPEPS'09*, Oct. 2009, pp. 251–254.

[83] C. Zhuo, H. Zhang, R. Samanta, J. Hu, and K. Chen, "Modeling, optimization and control of rotary traveling-wave oscillator," in *Proc. ACM/IEEE ICCAD'07*, Nov. 2007, pp. 476–480.

Vita

Ying Teng was born in Tianjin, China. She received the Bachelor's and Master's degrees in Electronic Information Engineering from Tianjin University, Tianjin, China in 2004 and 2007, respectively. She joined the Department of Electrical and Computer Engineering at Drexel University in September 2009, and studied with Dr. Baris Taskin. During her Ph.D. study, she focused on resonant clock distribution network design. Her research interests include VLSI physical design and optimization, clock network synthesis, timing/power analysis, mixed-signal circuit design and custom circuit design automation.

Ying Teng served as a Teaching and Research assistant at Drexel University. As a teaching assistant, she participated in teaching and developing the graduate/undergraduate level VLSI design course at Drexel University. As a research assistant, she has published several papers at prestigious journals and conferences including TVLSI, ICCAD, DATE, GLSVLSI, ICCD, ISQED and etc. She is going to join Apple at Cupertino, CA upon graduation in 2014 to continue her research in physical design field.