

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,000

Open access books available

147,000

International authors and editors

185M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



# On the Efficient Design & Synthesis of Differential Clock Distribution Networks

Houman Zarrabi<sup>1</sup>, Zeljko Zilic<sup>2</sup>, Yvon Savaria<sup>3</sup> and A. J. Al-Khalili<sup>1</sup>

<sup>1</sup> *Department of Electrical and Computer Engineering, Concordia University*

<sup>2</sup> *Department of Electrical and Computer Engineering, McGill University*

<sup>3</sup> *Department of Electrical Engineering, École Polytechnique de Montréal  
Canada*

## 1. Introduction

Almost all high-performance VLSI systems in today technologies are synchronous. These systems use a *clock* signal to control the flow of data throughout the chip. This greatly facilitates the design process of systems because it provides a global framework that allows many different components to operate simultaneously while sharing data. The only price for using synchronous type of systems is the additional overhead required to generate and distribute the clock signal.

Nearly all on-chip Clock Distributions Networks (CDNs) contain a series of buffers and interconnects that repeatedly power-up the clock signal from the clock source to the clock sinks. Conventionally, CDNs consisted of only a single stage buffer driving wires to the clock loads. This is still the case for clock distribution in very small scale systems; yet contemporary complex systems use multiple buffer stages. A typical clock tree distribution network in modern complex systems is shown in Figure 1. This design is based on the reported CDNs in (O'Mahony et al, 2003; Restle et al, 1998; Vasseghi et al, 1996).

### 1.1 Hierarchy in CDNs

The clock signal is generated with a Phase Lock Loop (PLL). A PLL is a control system that generates a signal having a fixed relation to the phase of its reference signal. A PLL circuit responds to both the frequency and the phase of its input signal and automatically raises/lowers the frequency of the controlled oscillator until it matches the reference (Wikipedia, 2009). The core clock signal is then amplified through the global buffer and distributed through a hierarchical network and buffers. The system CDN is generally defined to span from the PLL to the clock *pins*. The pin is the input to a buffer that locally amplifies and distributes the clock signal to clocked storage elements within a macro, the small blocks that make up a system. There can be any number of buffer levels between the PLL and the clock pin. In modern VLSI systems, there are up to four buffer levels. The last buffer level before the clock pin is generally called a sector buffer. This stage drives the interconnect leading to the macros and the local buffers at the pins. A synchronous VLSI

system has thousands of loads to be driven by clock signal. In CDNs, the loads are grouped together creating a (sub-) block. This trend results in a hierarchy in the design of CDNs including three different levels/categories of clock distribution namely as *global*, *regional* and *local* as shown in Figure 1. At each level of hierarchy there are buffers associated with that level to regenerate and to improve the clock signal at that level.

The global clock distribution connects the global clock buffer to the inputs of the sector buffers. This level of the distribution has usually the *longest path* in CDN because it relays the clock signal from the central point on the die to the sector buffers located throughout the die. The issues in designing the global tree is *mostly related to signal integrity* which is meant to maintain a fast edge rate over long wires while not introducing a large amount of timing uncertainty. Skew and jitter accumulate as the clock signal propagates through the clock network and both tend to accumulate proportional to the latency of the path. Because most of the latency occurs in the global clock distribution, this is also a primary source of skew and jitter (Restle et al, 2001). From a design point of view, achieving low timing uncertainty is the most critical challenge at this level.

The regional clock level is defined to be the distribution of clock signals from the sector buffers to the clock pins. This level is the middle ground between global and local clock distribution; it does not span as much area as the global level and it does not drive as much load or consume nearly as much power as the local level.

The local level is the part of the CDN that delivers the clock pin to the load of the system to be synchronized. This network drives the final loads and hence consumes the most power. As a design challenge, the power at the local level is about one order of magnitude larger than the power in the global and regional levels combined (Restle et al, 2001).

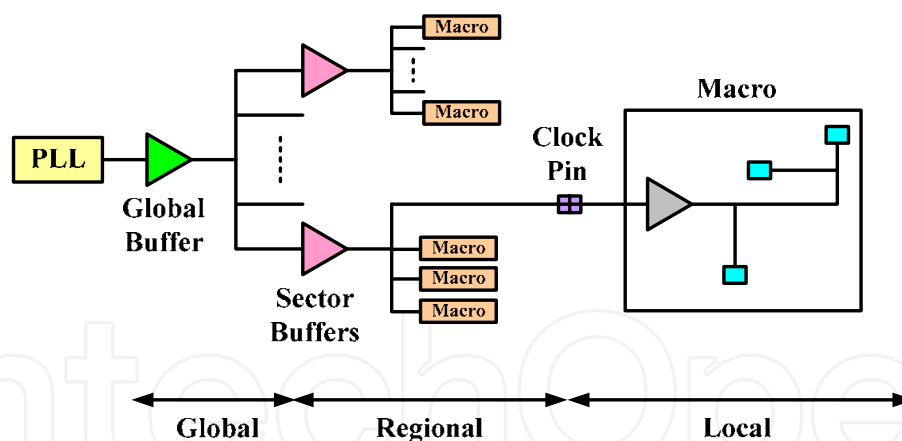


Fig. 1. A typical hierarchical CDN for a high-performance synchronous VLSI system

### 1.2 CDNs figures of merit

The main figures of merit for a CDN are the components of timing uncertainty, as well as, power consumption. All of these performance metrics have significant impacts on the design, evaluation and verification of synchronous system performance and reliability.

As mentioned previously, the advantage of a synchronous system is to regulate the flow of data throughout the system. However, this synchronizing approach depends on the ability to accurately relay a clock signal to millions of individual clocked loads. Any timing error introduced by the clock distribution has the potential of causing a functional error leading to

system malfunctioning. Therefore, the timing uncertainty of the clock signal must be estimated and taken into account in the first design stages. The two categories of timing uncertainties in a clock distribution are *skew* and *jitter*.

Clock skew refers to the absolute time difference in clock signal's arrival time between two points in a CDN. Clock skew is generally caused by mismatches in either device or interconnect within the clock distribution or by temperature or voltage variations around the chip. There are two components for clock skew: the skew caused due to the static noise (such as imbalanced routing) which is *deterministic* and the one caused by the system device and environmental variations which is *random*. An ideal clock distribution would have zero skew, which is usually unachievable.

Jitter is another source of dynamic timing uncertainties at a single clock load. The key measure of jitter for a synchronous system is the period or cycle-to-cycle jitter, which is the difference between the nominal cycle time and the actual cycle time. The first cycle, the period is the same as the clock signal period and the second cycle, the clock period becomes longer/shorter. The total clock jitter is the sum of the jitter from the clock source and from the clock distribution. Power supply noise may cause jitter in both the clock source and the distribution (Herzel et al, 1999).

Clock network also involves long interconnects which implies having lots of parasitics associated with the network contributing to the power consumption of the clock signal. Having the highest switching activity of the circuit in a chip is another fact of consuming a large amount of power of the system. This power consumption can be as high as 50% of the total power consumption of the chip according to (Zhang et al, 2000). The components of power consumption of CDN are: static, dynamic and leakage power. The power consumption due to the leakage current, in CDNs, is relatively small. In the same way, keeping the proper rise/fall times, minimizes the static power consumption. Thus the main portion of the power consumption is due to the dynamic power consumption. This is estimated as:

$$P = f C_L V_{dd} V_{swing}$$

in which  $f$ ,  $C_L$ ,  $V_{dd}$  and  $V_{swing}$  respectively represent frequency of the clock network, total load capacitances, supply-voltage and voltage-swing of clock signal. For the case of full swing (in which the clock signal swing reaches the voltage-supply level)  $V_{swing}$  is the same as  $V_{dd}$ . Accordingly, methods to reduce the power consumption are:

- a. Reduce total load capacitances ( $C_L$ )
- b. Reduce voltage-supply ( $V_{DD}$ )
- c. Reduce clock signal swing ( $V_{swing}$ )

The intrinsic load capacitance relies on the process technology and there is no handy way to improve it. Yet, from the design aspects by breaking down interconnects by repeater insertion the total interconnect load is reduced. Worth mentioning that in coupled lines, the total load is greater than that of single-node lines, thus compensating design methods should be taken into consideration for power-saving improvement. Typically, power reduction is achieved by means of supply and/or swing voltage scaling in CDNs.

## 2. Differential Clock Distribution Networks (DCDNs)

In this section, based on the general overview given on CDNs, we will introduce the concepts and motivations toward the design of Differential CDNs (DCDNs). For this, we initially address the preliminaries needed for the design of DCDNs. These theories include differential signaling and differential signal integrity.

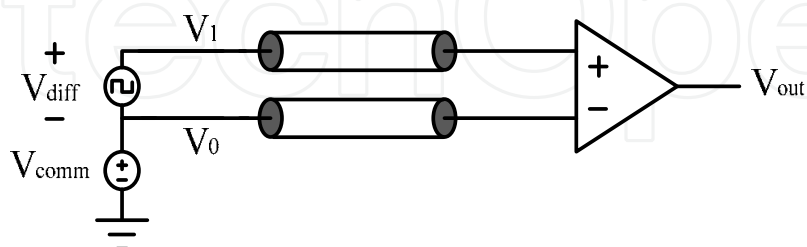


Fig. 2. Voltage-mode differential signaling

### 2.1 Preliminaries

#### 2.1.1 Differential signaling

A digital signal can be transmitted *differentially* over the medium by utilizing two conductors. One of which is used for transmitting the signal and the other is used for the complement of the signal. Figure 2 shows a differential voltage-mode signaling system. To transmit logic '1', the upper voltage source drives  $V_1$  and the lower voltage source drives  $V_0$ . For logic '0' transmission, the voltages are reversed.

As is shown in Figure 2, the following voltages are defined in a differential system:  $V_1$  is the signal on the first line with respect to common return path,  $V_0$  is the signal on the second line with respect to common return path,  $V_{diff}$  is the differential signal which is the voltage difference of the two signal pair, and,  $V_{comm}$  is the common voltage signal which is in common between both of signal pair. Differential signal  $V_{diff}$  carries the information and at the receiver the information is extracted from this voltage difference. In addition to the differential voltage there is a common-mode signal. This signal is used to give an initial biasing to the differential signal pair. In ideal conditions, the common-mode signal is constant and it does not carry any information. In this case:

$$V_{diff} = V_1 - V_0$$

Differential signaling requires more routing and wires and pins than its single-ended counterpart system. In return for this increase, differential signaling offers the following advantages over single-ended signaling:

- A differential system, serves its own reference. The receiver at the far end of the system compares the two signal pair to detect the value of the transmitted information. Transmitters are less critical in terms of noise issues, since the receiver is comparing two pair of signals together rather than comparing to a fixed reference. This results in canceling any noises in common to the signals.
- The voltage difference for the two signal pair between logic '1' and '0' is:

$$\Delta V = 2(V_1 - V_0)$$

which is twice as much as is defined for a single-ended signaling system. This shows that *the noise margin of the differential system is twice as much as the single-ended signaling system*. This doubling effect of signal swing improves the speed of the signaling system. It affects the transition times (rise/fall time) which is done in half of the transition time of single-ended signaling system.

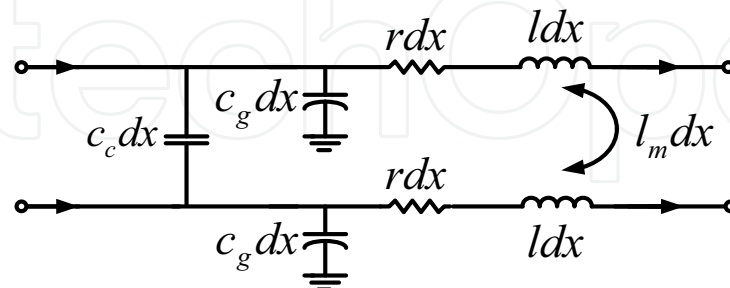


Fig. 3. A segment of a coupled interconnect

### 2.1.2 Differential signal integrity

In order to employ differential signaling, the coupled interconnects model is utilized and applied to the system. This type of interconnects not only have the intrinsic signal integrity issues, but also, they are involved with their mutual signal integrity aspects. In Figure 3, a segment of a coupled interconnect is shown.

The mutual parasitic elements are due to the adjacent line. These are mutual capacitance  $C_c$  and mutual inductance  $l_m$  in addition to the intrinsic parasitic elements  $r$ ,  $C_g$  and  $l$  which indicate intrinsic resistance, capacitance and inductance of each line. The effective capacitance  $C_{eff}$  associated with each line, depending on the direction/mode of the signaling (in-phase or out-of-phase usually called *even* and *odd* mode respectively) can be calculated from the following equations (Hall et al, 2000):

$$C_{eff}(odd) = \eta C_c + C_g$$

$$C_{eff}(even) = C_g$$

And for effective inductance we have:

$$l_{eff}(odd) = l - l_m$$

$$l_{eff}(even) = l + l_m$$

As the above equations indicate, for the case of differential signaling (or out-of-phase signaling), the effective capacitance is increased by the factor of  $\eta$  due to coupling capacitances and the effective inductance is decreased due to the effect of mutual inductance. In (Kahng et al, 2000) it was shown that  $\eta$  has the value of  $\{0, 2 \text{ and } 3\}$  depending on the mode of signaling and slew rates of the coupled signals. The typical value for  $\eta$ , for typical sharp input signals designs, is taken as 2.

2.1.3 Differential Buffers

The configuration of differential buffers is based on current steering devices, in which the output logic can be set by steering the current in the circuit. These devices are also considered as Current Mode Logic (CML) circuits. CML circuits are known to outperform the conventional CMOS circuits in Giga Hertz (GHz) operation frequency. A basic differential buffer is given in Figure 4. The current source in differential buffer is the tail current  $I_{ss}$ . When the common-mode voltage  $V_{comm}$  is applied to the differential buffer, due to the symmetry of the differential buffer, the current is split equally between the two wings ( $I_{ss}/2$ ). Increasing one of the input voltages which implies the decrease in the other one, will result in increase in current of one branch and decrease in current of the other branch. Note that the total possible current to steer is  $I_{ss}$  and when one input voltage rises, the other one decreases by the same amount. When the input differential voltage  $\Delta V=V_{in}-V'_{in}$  has passed a specific threshold, in other words when one of the transistors derives all the possible current from one branch the other transistors turn off, hence the output voltage reaches  $V_{dd}$  whereas the first branch drops to  $V_{dd}-RI_{ss}$ . Several differential loads also have been introduced in the literature (Dally et al, 1998). These loads may use resistor, current mirror and cross-coupled transistors. The differential load is characterized by its differential and common-mode impedances, known as  $r_{\Delta}$  and  $r_c$  respectively. The differential impedance determines the change in the differential current  $I_{\Delta}$  when the voltages on the two inputs of the terminal are varied in opposite directions. The common-mode impedance implies the average current changes when both input voltages are varied in the same direction. Depending on the type of application, the design may chose from these design options. Table I demonstrate the  $r_{\Delta}$  and  $r_c$  for each load.

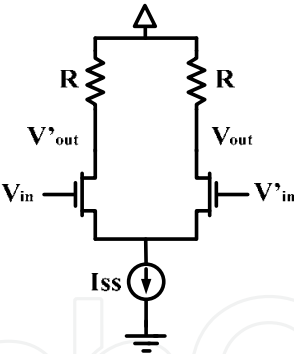


Fig. 4. A basic differential buffer

| Load           | $r_c$   | $r_{\Delta}$   |
|----------------|---------|----------------|
| Resistor       | $R$     | $R$            |
| Current-mirror | $1/g_m$ | $-1/\lambda I$ |
| Cross-coupled  | $1/g_m$ | $1/g_m$        |

Table 1. Impedance of differential loads

## 2.2 Differential Clock Distribution Networks (DCDNs)

As discussed previously, differential signaling offers higher immunity against external perturbations. Due to the complexity increase and the need for error-free operation in contemporary systems, the idea of integrating differential signaling and clock distribution is seemingly becoming a viable solution for modern and for future IC designs.

Historically the idea of DCDN was to be utilized for off-chip clock distribution and for PCB-level synchronization. This technique was utilized to reduce and suppress the Electro-Magnetic Interference (EMI) of the neighboring circuits and systems waves. Due to the superiority of DCDN, recently there has been a couple of works on on-chip DCDN as well, such as (Sekar, 2002; Anderson et al, 2002). The idea of utilizing on-chip DCDN has not been widely used in the literature. In (Anderson et al, 2002) a DCDN is used in global level of the hierarchical CDN for Itanium Microprocessor. They reported that the use of DCDN has given the advantage of 10% less skew variation. In (Sekar, 2002) it is reported that DCDN has 25%-42% less sensitivity to power supply noises and 6% less sensitivity to manufacturing variations when they utilized H-Tree DCDN.

A general model of a DCDN is given in Figure 5. The DCDN is composed of a differential signal pair shown in two different patterns. The clock tree generally is a binary tree. The differential signal is dispersed along the clock network. Throughout the clock network at branching points the differential clock signals are regenerated by differential buffers to improve the signal integrity of the clock network. Finally at the last stage, they are all converted to single-ended signals for compatibility with the rest of the system functionality, which normally use single-ended signals. For the regenerative buffers a simple differential buffer introduced in the previous part can be utilized. The only design issue related to the buffer is the choice of differential loads. Based on the process technology, or design criteria, this item can be chosen from the design library. For final stage converters, usually the choice of current mirror load is the superior choice. As Table 1 demonstrates, current mirror loads have high differential output impedance which results in fast change in the output that is used to drive the output of the clock network.

Differential clocking eliminates the induced crosstalk due to aggression of clock signals. Clock signal is spread all over the chip area. It also has full switching activity. Also device sizes tend to shrink as technology advances. These facts show that as technology advances the clock signal aggression can be quite harmful for all system components all over the chip area. Distribution of clock with differential signals eliminates this problem to a certain extent, as both positive and negative signal values are applied and the noise would be cancelled. Furthermore, as given in (Anderson et al, 2002), DCDN offers less skew variations in the presence of external noises; it has less sensitivity in presence of supply and process variations (Sekar 2005).

The aforementioned points are of the most important criteria/solutions for reliable system design. Due to technology advances and increase in system complexity, the design with low or no parameter variation in ideal case, has become the most concerning issue. Timing error results directly in system malfunctioning. Thus designing a reliable and noise tolerant, clock distribution may help significantly for a reliable system design. As introduced in the literature, DCDN has these potentials; thus this design methodology can be a solution for future robust system design.

Plus the pros and cons of DCDN, there are some design/synthesis challenges associated with the efficient design of DCDNs. Some of most challenges may be summarized as:

- Differential signaling is involved with higher parasitic, due the existence of coupled lines. In this case the total power consumption is commonly increased.
- Coupled lines are commonly routed and synthesized using symmetrical path models (which is not the general case).
- Routing complex DCDNs may take too much computation time. Using existing routing methods is not time efficient.

Proposing solutions to address the above challenges can efficiently help the design and synthesis of DCDNs, needed for modern complex VLSI technologies. These solutions are given in the following sections.

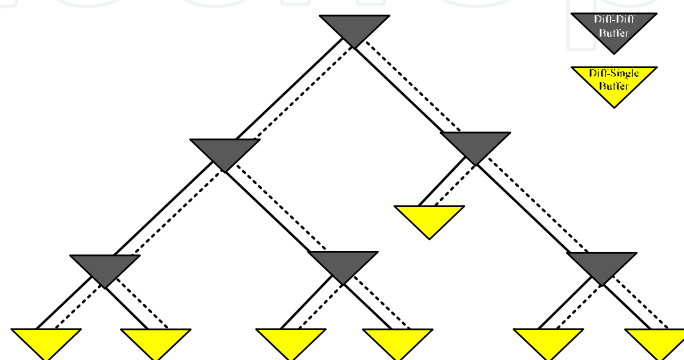


Fig. 5. A general structure for DCDN

### 3. Efficient design of DCDNs

#### 3.1 Dynamic Threshold (DT) MOS for low-voltage DCDNs

For the design and synthesis of CDNs, buffers are inserted to improve the performance of CDNs in order to reduce the overhead capacitances. In this part, Dynamic Threshold (DT) (sometimes referred to as Variable Threshold) transistors (Assaderaghi et al, 1997) are utilized in conventional differential buffer structures. These transistors outperform the conventional transistors in low voltage applications which are suitable for advanced low voltage technologies. The use of DT transistors helps improve the buffer performance. DT transistors switch faster since their threshold voltage decrease dynamically when the input is applied to their gate terminal due to body effect. Such buffers are depicted in Figure 6 (Zarrabi et al, 2006). Figure 6(a) presents a Low-Swing:Low-Swing differential buffer. DT transistors help improve the speed of these buffers when low swing inputs are applied to the buffer. The use of cross-coupled differential load with high differential impedance helps to have a fast transition of the inputs to the outputs. Figure 6(b) represents a Low-Swing: Full-Swing level converter. This buffer is used at the sinks to restore clock signals to their single-node full amplitude. The current-source pull-ups help to have asymmetrical fast transformation of differential to single-ended signals. The structure is based on Chappell amplifier which offers good common-mode noise rejection (Chappell et al, 1998).

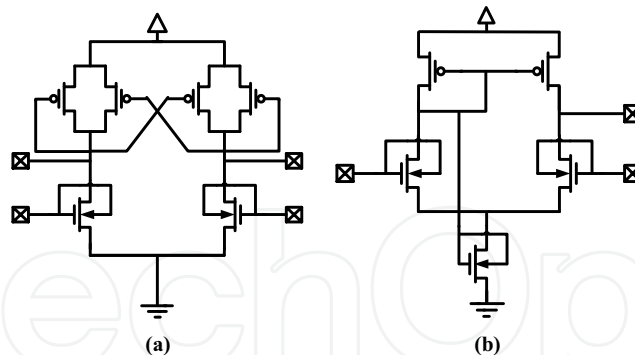


Fig. 6. (a) Low-Swing: Low-Swing (b) Low-Swing: Full-Swing DT differential buffers

### 3.2 Differential low-power buffers

Recalling from Section 2.1.1, two voltage components are associated with differential signaling: common-mode and differential. Common-mode also refers to DC voltage biasing and is the voltage used for initial biasing of the differential buffer.

In order for receivers (differential to single-ended converters) to operate efficiently and have a full/proper output swing, the common-mode voltage or DC biasing of the differential buffer should be low enough to turn the input transistors off. In the literature, the method used in order to overcome this issue is to increase the voltage swing as much as possible to be able to decrease the common-mode voltage to the sufficient supply level (usually used differential voltage of 50% of  $V_{dd}$ ) (Anderson et al, 2002; Sekar, 2005). This method results in high power consumption in DCDN. Recalling from Section 2.1:

$$V_{diff\_low} = V_{dd} - RI_{ss}$$

The above equation implies that in order to increase the differential voltage swing, the tail current need to be increased. This technique largely affects the power consumption in DCDN. Note that, it is not possible to touch the load ( $R$ ) as it directly affects the bandwidth of the clock network. Therefore, in previous works, in order to reach sufficient output swing, the differential voltage swing is increased to reduce common-mode voltage. Correspondingly, a circuit technique is proposed to address this design problem.

The proposed technique for differential receiver is given in Figure 7 (Zarrabi, 2006). The buffer configuration is based on Chappell amplifier as introduced in the previous section. Attached to the buffer are the level-shifting circuits. The buffer functionality is as follows:

The dashed parts in Figure 7 are the level shifters (also referred to as source followers) (Razavi, 2001). When the input is applied to the gate terminals of the level shifters, the outputs are dropped and follow their inputs. In other words, the voltage gain equals one (no voltage amplification), and the following relations are applicable (Broderson, 2005):

$$\begin{aligned} I &= (\beta/2)(V_{IN} - V_{OUT} - V_T)^2 \\ V_{OUT} &= IR_s \\ V_{OUT} &= (R_s\beta/2)(V_{IN} - V_{OUT} - V_T)^2 \\ V_{IN} &= V_{OUT} + V_T + [(V_{IN} - 2)/(R_s\beta)]^{0.5} \end{aligned}$$

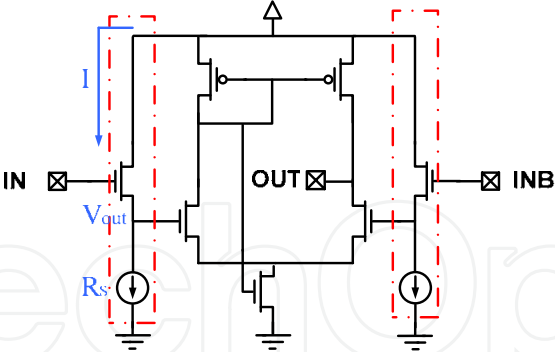


Fig. 7. Differential receiver with level shifter

The last result shows that  $V_{OUT}$  can be derived by solving the final equation iteratively. However, by making the first order approximation that  $R_S$  is large enough (especially in current sources) to make the third term equal to zero, we can conclude:

$$V_{OUT}= V_{IN}-V_T$$

This shows that the output of the source follower circuits copies the input of the gates with a shift of a transistor threshold which is a technology dependent factor. The transistor ratios for buffers sizing are the same as the ones given in (Chappell et al, 1998). However, the total size of the buffer is scaled to minimize the skew.

The above configuration for the differential receivers helps lower the common-mode (DC bias) of the internal input transistors of the receiver. Utilizing this design technique, it is possible to further reduce the differential voltage swing while maintaining a sufficient output swing at the final nodes.

In order to perform differential voltage scaling in DCDN, previously a new design for level converter was given. For the case of intermediate buffers, in order to be able to vary the differential voltage while maintaining the linearity of the buffer, the differential load should be reconfigured in a way to establish this design goal. In this part, a new configuration for differential load is proposed which enables us to have linearity in the buffer. Figure 8 shows the proposed buffer configuration.

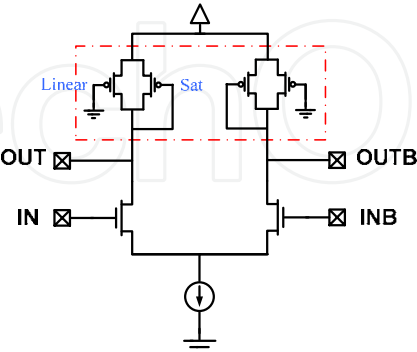


Fig. 8. Differential buffer with composite load

The dashed part demonstrates the proposed composite configuration of the differential load. Such composition enables the circuit to combine both the characteristics of the diode connected device and triode transistor together to have a linear operating load in various

voltage ranges (Dally et al, 1998). The proposed buffer based on composite differential load is a technology portable design and can be used in any available design process whereas the use of resistance is limited to current and future advanced technologies. This portable design method comes at the price of increase in area and parasitic elements. The transistor ratios (for buffers sizing) are 1 to 3 which refer to the ratio of pull up to pull down transistors ( $L=2L_{\min}$  to reduce the channel length modulation effect). The total size of the buffer is scaled to reach the objective frequency of operation.

## 4. Efficient synthesis of DCDNs

### 4.1 Zero skew DCDN routing

As seen in the overview section, DCDNs are commonly routed assuming symmetrical CDN path models. This however is not the general case. Here we will propose a method for zero skew routing of DCDNs applicable to general (asymmetric) path models. In the literature, especially in (Cong et al, 1996) a comprehensive study on efficient clock routing is studied. Tsay's method (Tsay, 1991) is one of the methods introduced for zero skew routing of clock trees. In order to route differential clock trees with zero skew characteristic, the existing methods are modified to satisfy design objectives. To achieve this aim here in this part, a line equivalent delay model is utilized. The model is applied to Tsay's method (Tsay, 1991) for zero skew routing of differential clock trees (DCDNs).

#### 4.1.1 Utilizing Tsay's method

In this method, zero skew is achieved by locating tapping points throughout the clock tree. Tapping points are the branching points at which sub-trees are chosen to maintain equal delay as shown in Figure 9.

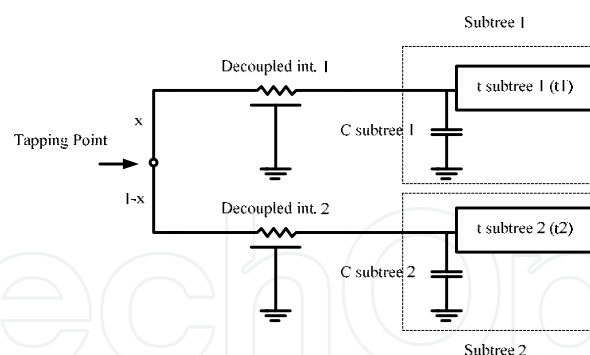


Fig. 9. Tapping point extraction through merging decoupled sub-tree(s)

As was seen in Section 2.1.2, the effective capacitance associated with each segment of a coupled line, considering both intrinsic and mutual effects is:

$$C_{eff}(odd) = \eta C_c + C_g$$

The effective capacitance is applied to both signal lines independent of each other; in this way, we call these lines as decoupled lines and we name this model as *decoupled line* model. This model is employed for the purpose of clock routing. In this way, the decoupled RC- $\Pi$  delay model is used to model interconnects. The methodology of tapping extraction is as

follows. Figure 9 shows a schematic of a decoupled clock tree branch in which each line of the branch is a decoupled distributed RC model connected to its sub-tree child, for which the distributed line propagation delay is given by  $t_{int}=0.37R_{int}C_{eff}$ . Each sub-tree is modeled by a total capacitance  $C_{subtree}$  and total propagation delay  $t_{subtree}$  as shown in Figure 9. Considering tapping location  $x$ , to satisfy the equality of the two branch delays, the following equation is realized:

$$t_{int1}+0.74R_{int1}C_{effsubtree1}+t_1=t_{int2}+0.74R_{int2}C_{effsubtree2}+t_2 *$$

In the second part of the equality, since the interconnect resistance combined with sub-tree capacitance creates a *Lumped* loop, it has the lumped propagation delay of  $0.74R_{int}C_{subtree}$ . Rewriting interconnect parasitics by per unit length parameters, we have:

$$R_{int1}=r_0xl, C_{int1}=c_0xl \\ R_{int2}=r_0(1-x)l, C_{int1}=c_0(1-x)l$$

in which  $r_0$  and  $c_0$  are the resistance and capacitance per unit length of the wire,  $l$  is total interconnection length between the two sub-trees and tapping location  $x$ . Solving Equation \* with respect to  $x$  results into:

$$x=[1.35(t_{int2}-t_{int1})+r_0l(C_{effsubtree2}+0.5c_0l)]/[r_0l(C_{effsubtree1}+c_0l+C_{effsubtree2})]$$

In case of ( $x \leq 0$  or  $x \geq 1$ ), elongation would be needed. Elongation is the process of adding extra wire length to the sub-tree which has less effective capacitance, in order to equalize the delay of both sub-trees. The length of elongation to maintain zero skew is given by:

$$L'=[-20r_0C_{effsubtree2}+2(100r_0^2C_{effsubtree2}^2+270r_0c_0(t_{int2}-t_{int1}))^{0.5}]/[20r_0c_0]$$

This methodology is applied for zero skew routing in DCDNs. The results are given in Section 5.1 will validate the efficiency of this methodology.

## 4.2 Parallel synthesis of DCDNs

CDN synthesis is one of the primary time-consuming steps, performed in the synthesis flow of VLSI systems. Especially with the growth of complex SoCs in current advanced technologies, this part has become more complicated and less computational cost-effective. Many efforts have been put into parallel computer aided design, all with the goal of reducing the computation time. In literature, methodologies have been proposed for parallel synthesis of CDN such as the ones proposed in (Banerjee et al, 1992; Banerjee, 1994). These methods however, focus mainly on the single-ended clock tree structure. In this section, the goal is to leverage distinctive features of parallel computation to reduce computational time required to synthesize DCDNs (Zarrabi et al, 2007). The methodology utilizes and extends the technique proposed in Section 4.1, to synthesize zero skew DCDNs in parallel. This is a flexible methodology, applicable to symmetric/asymmetric and hybrid (differential and/or single-ended) clock tree structures.

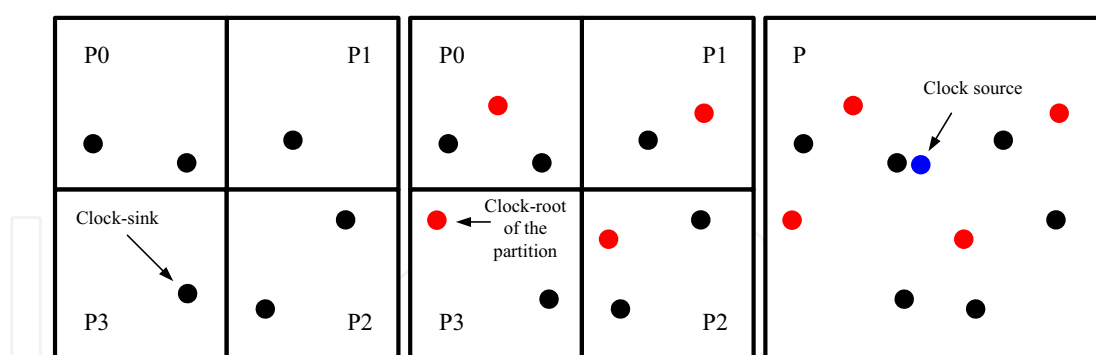


Fig. 10. Parallel DCDN distribution: a) partitioning the die area into sub-regions, b) locating the clock-root of each region, c) finding the source of the clock network

The methodology for parallel synthesis of zero skew DCDNs is as follows. Initially the total chip area is partitioned into sub-regions (partitioning phase). Later, synthesis of zero skew differential clock distribution networks is performed on each of the partitioned regions (local clock distribution phase). In the final stage, the global differential clock network is routed for each of the previously-extracted clock-roots of the sub-regions (global clock distribution phase). The obtained source of the clock network can end up anywhere in the whole chip area (Manhattan surface), regardless of the initial partitioning. The proposed scenario is illustrated in Figure 10. The proposed method may be implemented using C++ language and the Message Passing Interface (MPI) platform (MPI). A pseudo-code describing the method is given in Figure 11.

A possible negative side effect of parallel synthesis is the increase in the total wire-length in the clock network. This could be interpreted as the impact of multi-stage distribution of the clock network which results in initial local zero-skew clock networks and a final global clock network routed on top of regional clock networks. In general, this parallel processing approach results in a clock-tree different from the one routed in a single step, due to die area partitioning; thus, the characteristics of the new clock tree such as total wire-length and skew may be slightly different. This proposed methodology is flexible, as it allows having a hybrid (differential and/or single-ended) distribution of the clock network. The global CDN could be differential, while the local (lower levels) CDNs could be single-ended to alleviate routing complexity. It is possible to enhance the global/local distribution algorithm with refined interconnect models. This methodology is also applicable to all symmetric/asymmetric clock-trees.

#### Parallel Zero Skew Differential Clock Distribution

(Clock-sinks, Number of Processing-nodes)

- ```
{
    1. Partition chip area according to the number of processing nodes.
    2. Apply 'local' zero skew (differential) clock distribution to the partitioned
       areas and send the clock-tree root(s) to the root processing node.
    3. Receive the processed clock-tree root(s) from processing nodes, and, apply
       'global' zero skew (differential) clock distribution.
    4. Return the obtained final clock-tree root as the source of the (differential)
       clock distribution network.
}
```

Fig. 11. Pseudo-code for parallel synthesis of zero skew differential clock distribution

5. Results

In this section, the quantitative results related to the given design and synthesis methods for DCDNs are given.

5.1 Zero skew routing

The zero skew routing method, inspired by Tsay’s algorithm and given in Section 4.1, was applied to IBM benchmarks r1...r5 (Tsay, 1991). Modified Tsay’s method (for differential signal integrity) was implemented using C++ for clock routing, and PERL language for netlist manipulation was utilized for design and simulations. HSPICE simulation results for the proposed method are tabulated in Table 2. The delay and skew results presented throughout this work represent the average and absolute difference of clock signal phase delay at sink nodes respectively.

Two methodologies were used for routing differential lines: Single-Spaced (SS) and Double-Spaced (DS) routing. In single-spaced routing scheme, the mutual coupling effects are stronger, therefore differential characteristics of the pair is more dominant. DS offers smaller mutual coupling, consequently this reduces delay while degrading noise immunity. This is due to the fact that the stronger the coupling effect is, the stronger common-mode noise rejection becomes.

Table 2 demonstrates that, on average, clock trees generated with the proposed model show 97% skew reduction compared to those obtained using Elmore model, i.e. neglecting coupling effects. This improvement is achieved because coupling effects of differential lines are more accurately considered in the algorithm leading to tapping point selection. As technology advances the coupling effects increase and we are no longer able to neglect these effects in system modeling. In this case, neglecting the coupling effects, results in the misplacement of tapping points and reduces the effectiveness of the considered zero skew DCDNs. Simulation results also show smaller delay and skew for the DS scheme due to reduced coupling, however this design strategy as we will see degrades robustness in presence of external noise.

| Bench Mark | Single Spaced (SS) |            | Single Spaced (SS) |            | Double Spaced (DS) |            |
|------------|--------------------|------------|--------------------|------------|--------------------|------------|
|            | Elmore             | Proposed   | Proposed           | Elmore     | Proposed           | Proposed   |
|            | Skew (ps)          | Delay (ns) | Skew (ps)          | Delay (ns) | Skew (ps)          | Delay (ns) |
| r1         | 115                | 1.1        | 5.9                | 1.1        | 2.4                | 0.9        |
| r2         | 199                | 3.8        | 15                 | 3.7        | 8.8                | 3.3        |
| r3         | 341                | 5.1        | 14                 | 5.1        | 8.0                | 4.6        |
| r4         | 759                | 17.5       | 36                 | 17.1       | 20                 | 14.5       |
| r5         | 1825               | 34         | 51                 | 34         | 39                 | 28         |

Table 2. Skew and delay of DCDN for r1-r5 benchmarks in 180nm technology

5.2 Applying DT buffers

Buffers were inserted to improve the performance of benchmark clock networks. The buffer insertion procedure is as follows. Low-swing buffers were inserted at branching points and level converting buffers were inserted at sinks. Low-swing buffers are sized to reduce propagation delay throughout the clock network. To accomplish this, a base size for buffer according to its delay-power characteristic diagram was chosen to drive a unit length interconnect segment. Further, the buffers were uniformly sized according to the longest interconnect. Full-swing level converters (differential to single-ended) were composed of minimum size transistors to reduce the power consumption and skew reduction; their sizes were scaled up relative to their load capacitance.

| Bench Mark | Low-Swing Conventional |            | Low-Swing Proposed |            | Full-Swing Conventional |            |
|------------|------------------------|------------|--------------------|------------|-------------------------|------------|
|            | Skew (ps)              | Delay (ns) | Skew (ps)          | Delay (ns) | Skew (ps)               | Delay (ns) |
| r1         | 695                    | 9.6        | 545                | 7.0        | 71                      | 1.1        |
| r2         | 844                    | 10         | 679                | 7.7        | 163                     | 1.3        |
| r3         | 808                    | 10         | 667                | 7.8        | 127                     | 1.3        |
| r4         | 1388                   | 11.9       | 981                | 8.8        | 379                     | 1.6        |
| r5         | 1566                   | 13.1       | 1135               | 9.6        | 532                     | 1.9        |

Table 3. Buffered DCDNs (Full-Swing Vdd=1.8V, Low-Swing Vdd=0.5V)

Table 3 shows the skew and delay difference for similarly sized, buffered DCDN based on the conventional (Dally et al, 1998) and the proposed buffers. It shows 25% delay and skew improvement on average compared to conventional buffers in low-swing differential clocking scheme. Results show that the delays are reduced significantly while skews are degraded as compared with un-buffered DCDNs. It is believed that skews in buffered clock networks can be reduced significantly by enhancing the process of buffer insertion. For instance, differential buffers delay model should be considered when tapping points are selected in the zero skew DCDN design algorithm.

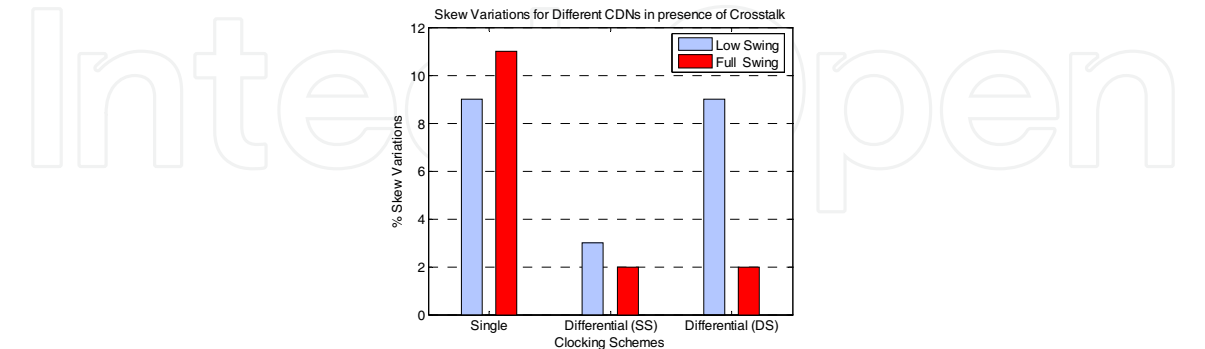


Fig. 12. Skew variations due to crosstalk

With regards to the skew sensitivity of the proposed DT DCDNs, two types of external aggressors, resulting into random skew are investigated: power-supply variations and crosstalk. Comparisons were made between similarly designed single-node CDN, single-

spaced DCDNs and double-spaced DCDNs (Figure 12). Benchmark r3 is used for simulations due to its average characteristic in terms of size and simulation time. For the low-swing scheme, the power supplies were  $V_{ddH}=1.8V$  &  $V_{ddL}=0.5V$ , whereas for full-swing scheme a single supply voltage ( $V_{ddH}=1.8V$ ) was used. In those experiments, supply-voltages were varied by  $\pm 10\%$ .

Simulation results show that for both clocking schemes, the single-spaced DCDN is the most robust design method in the presence of power-supply variations when compared to other CDNs. Skew variations increase when low-swing clocking is used. Double-spaced DCDN has less robustness to supply variations. DCDN is seen to have up to 25% less skew variations in low-swing clocking scheme and up to 9% less skew variations in full-swing clocking scheme than single-node CDN, in presence of power-supply variations.

Another source of perturbation that causes delay uncertainty in CDNs is crosstalk. For experiments, a full-swing aggressor is applied to one of the two big-child of the clock tree. The same low-swing and full-swing clocking schemes were considered. Simulation results show that the single-spaced DCDN shows 6% less skew variations when combined with low swing clocking scheme and 9% when combined with full swing clocking scheme as compared to single-node CDN subject to crosstalk.

### 5.3 Applying low-power buffers

In this part, the effect of employing low-power buffers introduced in Section 3.2 is studied. For comparison and performance evaluation of clock networks, a set of 400 MHz CDNs were designed for benchmark r3 due to its average size suitable for simulations. All reported designs were minimally sized to meet the target operating frequency (400MHz) based on 180nm technology parameters. In DCDNs, the differential signal swing was scaled by adjusting the tail current source of intermediate differential buffers. The lowest potential reached by either part of the differential signal is  $V_{dd}-RI_{ss}$  where  $V_{dd}$  is the supply voltage,  $I_{ss}$  is the tail current and  $R$  is the equivalent resistance of the transistor loads. Note that, the load resistance determines the bandwidth of the clock network; hence the only possible variable to tune is the tail current to scale the differential voltage swing. In the following, the effect of differential voltage scaling on power consumption and clock skew variations in the presence of power supply variations is explored.

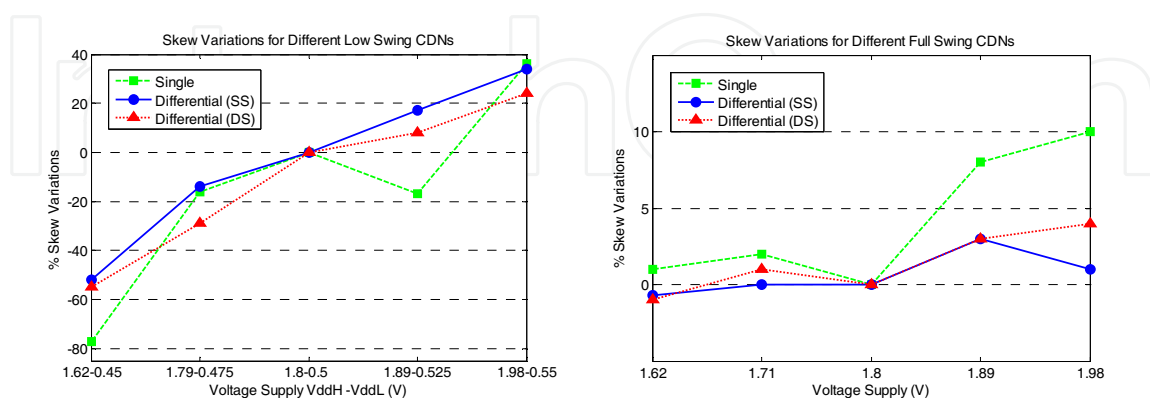


Fig. 13. Skew due to supply-voltage variations in low/full -swing schemes

### 5.3.1 Effect on power consumption

The tail current affects both the short circuit and dynamic power dissipation. Figure 14 demonstrates the effect of differential voltage scaling on the power consumption of DCDN and its single-node CDN counterpart both optimized for the same benchmark; which is obtained by scaling the tail current source of the differential buffer.

Figure 14 shows the power consumption obtained with three different clocking schemes: differential CDNs with composite-load buffers, differential CDNs with only grounded-gate load transistor buffers and the conventional single-node clocking scheme. The reason for exploring the network with only grounded-gate load transistor buffers is to demonstrate the effectiveness of composite-load buffers in terms of reduced sensitivity when large signal swings are considered. In this case, the size of the load transistors is increased (by approximately 20%) to get the same amplitude for a given tail current. The CDN used as reference is a conventional balanced single-node clocking tree, where transistors have the minimal size necessary to give full output swing while operating at the target frequency (400MHz).

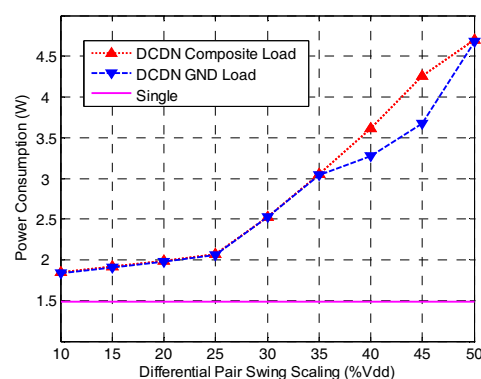


Fig. 14. Power consumption for voltage scaled DCDNs vs. a single-node CDN (r3).

Figure 14 shows that as the differential swing increases beyond 25% of supply-voltage (450mV,  $V_{dd}=1.8V$ ), the power consumption increases drastically. This emphasizes the significant impact of large differential voltage swings on the power consumption of the clock network. For differential voltage swings below 450mV, the power consumption is not reduced much if the differential swing scaling is further reduced. A lower bound of 10% of  $V_{dd}$  was imposed to the differential swing to ensure a sufficient noise margin. Another consideration is that even with a differential swing as low as 10% of  $V_{dd}$  (180mV), the power consumption of the differential clock network remains almost 30% higher than that of single-node clock distribution network. Thus, trying to match the power dissipation of a single-node network by decreasing the swing of differential networks does not appear to be a viable option. A final observation from Figure 14 is that the DCDN with grounded-gate loads (GND) consumes less power over a limited region where the differential swing is large. However, as we will see in the following, this slight reduction in dissipated power comes at a large price in clock skew variability.

### 5.3.2 Supply-voltage scaling

In the previous section, we observed that the reduction of the differential swing has a strong impact on the power consumption. For differential swings smaller than 25% of  $V_{dd}$ , the power consumption becomes less than half of that observed when the clock network operates with the voltage swing of 50% of  $V_{dd}$ . Second, as Figure 15 suggests, due to the fluctuations of common-mode voltage in the clock network, the voltage swing of 25% of  $V_{dd}$  (450mV) shows the least skew when the clock network is subject to supply variations. We also observe that DCDN with composite load is more robust. This is due to the linear characteristic of composite load buffers, as was seen in Section 3.2.

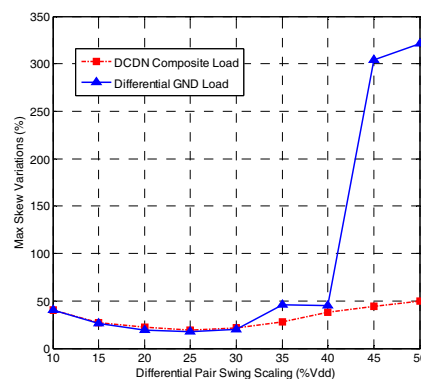


Fig. 15. Peak to peak skew variations in differential voltage scaled DCDNs.

Taking these considerations into account, we consider a design point for which the differential swing is 25% of  $V_{dd}$  (450mV) and we reduce the supply voltage to the point where we reach the same power consumption as that observed for the single-node and differential clock networks. HSPICE simulations demonstrate that for a supply voltage of 1.4V and differential swing 450mV, we obtain the same power consumption for the differential clock network. Yet, as can be observed from Figure 16, the variation of clock skew is still less than that of the comparable single-node CDN. Another interesting point that was observed during supply voltage scaling in DCDN is a *negligible signal latency difference*. This can be justified since as the tail current is lowered to achieve lower differential swings; the necessary differential voltage needed for the differential buffer to switch is also decreased. This enables the differential buffer to operate/switch faster than in the case where greater supply voltage with greater differential swing is used. Also as observed from Figure 16 and as discussed previously, the DCDN based on only grounded-gate loads is less resilient.

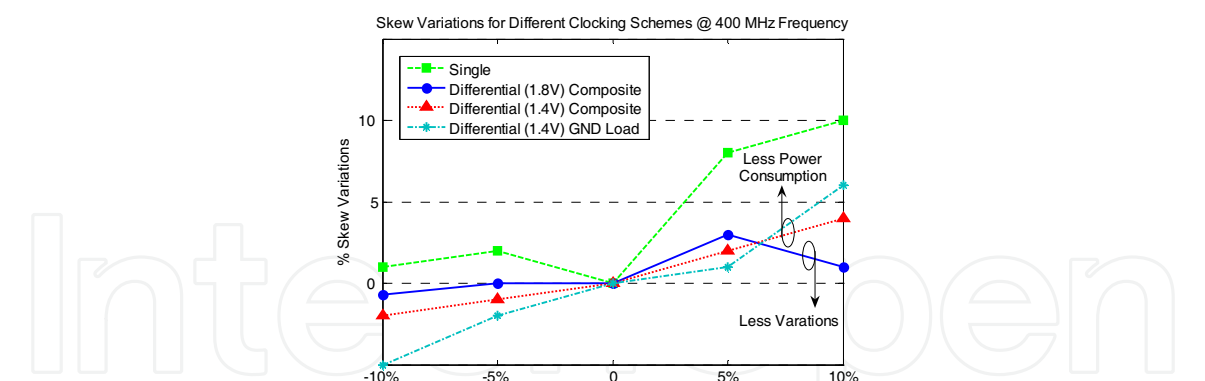


Fig. 16. HSPICE simulation show less variations in DCDN compared to single-node CDN for equal nominal power consumption.

5.4 Parallel zero skew routing

Benchmarks from (Tsay, 1991) with as many as 3101 clock sinks and total area as large as 1.4 cm \* 1.4 cm were chosen. The reported results are based on 180nm CMOS technology file, for which interconnect parameters are:  $C_g= 8e-17(F/um)$ ,  $C_c= 8e-17(F/um)$  and  $R_{int}=0.022(\Omega/um)$ . The processing time, speed up and resulting simulated skews when synthesizing zero skew DCDNs using 1 (sequential), 2 and 4 processing nodes are reported in Table 4. These results demonstrate a *nearly-linear speed-up*. It is expected that for very large benchmarks, the speed-up grows *linearly* when the number of sinks is sufficiently large compared to the number of processing nodes. Thus, the processing time overhead due to performing that synthesis in parallel was almost negligible. Figure 17 confirms these observations for 2 and 4 processing nodes. As the number of clock sink nodes increases, the speed-up (dashed lines) converges to the maximum expected speed-up (rigid lines).

| Benchmark<br>(# of sinks) | Skew (ps) |      |      | Computation Time (s) |      |      | Speed-Up |      |      |
|---------------------------|-----------|------|------|----------------------|------|------|----------|------|------|
|                           | 1 PN      | 2 PN | 4 PN | 1 PN                 | 2 PN | 4 PN | 1 PN     | 2 PN | 4 PN |
| r3<br>(862)               | 14        | 12   | 14   | 0.70                 | 0.48 | 0.21 | 1.0      | 1.45 | 3.24 |
| r4<br>(1903)              | 36        | 35   | 36   | 1.60                 | 0.90 | 0.46 | 1.0      | 1.76 | 3.46 |
| r5<br>(3101)              | 51        | 49   | 52   | 2.74                 | 1.45 | 0.73 | 1.0      | 1.89 | 3.74 |

Table 4. Run-time and speed-up results of benchmarks

In general, the parallel processing approach results in a clock-tree different from the one routed in a single step, due to die area partitioning; thus, the characteristics of the new clock tree such as total wire-length and skew may be slightly different. This proposed methodology is flexible, as it allows having a hybrid (differential and/or single-ended) distribution of the clock network. The global CDN could be differential, while the local (lower levels) CDNs could be single-ended to alleviate routing complexity. It is possible to enhance the global/local distribution algorithm with refined interconnect models. This methodology is also applicable to all symmetric/asymmetric clock-trees.

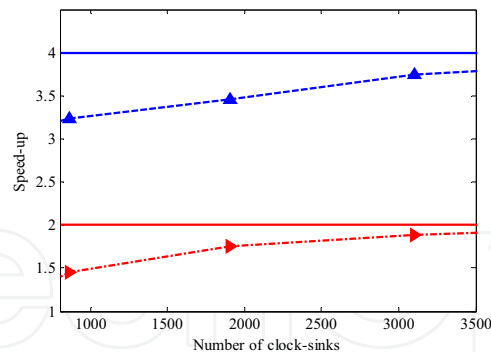


Fig. 17. Speed-up approaches its maximum, as the size of clock network increases, for 2 and 4 processing node synthesis cases.

## 6. Conclusions

In this chapter, some techniques for efficient design and synthesis of on-chip Differential Clock Distribution Networks (DCDNs) were given.

Initially design techniques were proposed that improve the performance of differential buffers which result into the performance improvement of DCDNs. This was achieved by means of introducing configurations for differential buffers based on Dynamic Threshold (DT) transistors. It was shown that for low supply-voltages, they outperform the conventional buffers with 25% delay reduction. Also, in order to overcome the high power consumption of DCDNs, a circuit configuration was proposed by which it is possible to reduce the differential voltage swings (down to 10% of  $V_{dd}$ ) which reduces the power consumption significantly (30% more than single-node CDN). Furthermore, by scaling the supply voltage of the system from 1.8V to 1.4V, we reach a design point where the DCDN consumes the same power as its single-node CDN counterpart but has less variation (in terms of skew). This however comes at the expense of delay and reduced voltage swing.

Various synthesis techniques were introduced that improve the DCDNs routing to achieve low (and possibly zero) skew. For this, a line equivalent delay model was suggested by which it is possible to route DCDNs with low (zero) skew. On average, 97% skew reduction was obtained utilizing this model compared to the classic Elmore delay model. A methodology for parallel distribution (routing) of zero skew DCDNs was also proposed. The method is applicable to all symmetric/asymmetric clock networks with ability for hybrid implementation (differential and/or single-ended). The proposed method alleviates the problem of high computational cost of such CDNs in complex VLSI systems. Utilizing this method, nearly-linear speed-up is achieved for zero skew DCDNs.

In the hierarchy of CDNs in modern high-performance complex systems, DCDNs can be effectively fit in the global level of CDNs; yet they can be used as the sole solution to the clock distribution of the system, when noise is the main design issue.

## 7. References

- Anderson, F. E.; Wells, J. S. & Berta, E. Z. (2002). The core clock system on the next generation Itanium microprocessor, in *ISSCC Digest of Technical Papers*, pp. 146-7.
- Assaderaghi, F.; Sinitsky, D.; Parke, S.A.; Bokor, J.; Ko, P.K. & Hu, Chenming. (1997). Dynamic threshold-voltage MOSFET (DTMOS) for ultra-low voltage VLSI", *IEEE Transactions on Electron Devices*, Volume 44, Issue 3, pp.414 – 422.
- Banerjee, Prithviraj. & Xing, Zhaoyun. (1992). A parallel algorithm for zero skew clock tree routing, *International Symposium on Physical Design*, pp. 118 – 123.
- Banerjee, Prithviraj. (1994). *Parallel Algorithms for VLSI Computer-Aided Design*, PTR Prentice Hall, Englewood Cliffs, New Jersey 07632.
- Broderson, Bob. (2005). *Analog Integrated Circuits*, online material, available: <http://bwrc.eecs.berkeley.edu/People/Faculty/rb/>.
- Chappell, B.A.; Chappell, T.I.; Schuster, S.E.; Segmuller, H.M.; Allan, J.W.; Franch, R.L. & Restle, P.J. (1988). Fast CMOS ECL receivers with 100-mV worst-case sensitivity, *IEEE JSSC*, Volume 23, Issue 1, pp:59 – 67.
- Cong, J.; He, L.; Koh, C. K. & Madden, P. (1996). Performance Optimization of VLSI Interconnect Layout, *Integration, the VLSI Journal*, vol. 21, pp. 1-94.
- Dally, William J. & Poulton, John. (1998). *Digital Systems Engineering*, Cambridge University Press.
- Hall, S.H.; Hall G.W. & McCall, J.A. (2000). *High-Speed Digital system Design, A Handbook of Interconnect theory and Design Practices*. John Wiley & Sons INC.
- Herzel, F.; Razavi, B. (1999). *A study of oscillator jitter due to supply and substrate noise*, *IEEE J. Circuits and Systems*, Volume 46, pp. 56 - 62.
- Kahng, A.B.; Muddu, S.; Sarto, E. (2000). On switch factor based analysis of coupled RC interconnects, *Design Automation Conference*, pp. 79 – 84.
- MPI, Message Passing Interface, online: <http://www.mpi-forum.org/>.
- O'Mahony, Frank P. (2003). *10GHz Global Clock Distribution Using Coupled Standing-wave Oscillators*, PhD Dissertation, Stanford University.
- Razavi, Behzad. (2001). *Design of Analog CMOS Integrated Circuits*. Mc Graw Hill.
- Restle, P. J. & A. Deutsch (1998). Designing the best clock distribution network, in *Symposium VLSI Circuits Digest of Technical Papers*.
- Restle, P.J.; McNamara, T.G.; Webber, D.A.; Camporese, P.J.; Eng, K.F.; Jenkins, K.A.; Allen, D.H.; Rohn, M.J.; Quaranta, M.P.; Boerstler, D.W.; Alpert, C.J.; Carter, C.A.; Bailey, R.N.; Petrovick, J.G.; Krauter, B.L. & McCredie, B.D (2001). A clock distribution network for microprocessors, *IEEE J. Solid-State Circuits*, vol. 36, no.5, pp. 792-799.
- Sekar, D.C. (2005). Clock trees: differential or single ended? , *International Symposium on Quality of Electronic Design*, pp.548 – 553.
- Tsay, R. S. (1991). Exact zero skew, in *Proc. IEEE Int. Conf. Computer-Aided Design*, pp. 336–339, Nov.
- Vasseghi, N.; Yeager, K.; Sarto, E. & Seddighnezhad, M. (1996). 200-MHz superscalar RISC microprocessor, *IEEE J. Solid-State Circuits*, vol. 31, no. 11, pp. 1675-1685.
- Wikipedia, online: [http://en.wikipedia.org/wiki/Phase-locked\\_loop](http://en.wikipedia.org/wiki/Phase-locked_loop)
- Zarrabi, Houman. (2006). *On the design and synthesis of differential clock distribution network*, MASc Dissertation, Concordia University.

- Zarrabi, Houman; Saaied, Haydar; Al-Khalili, A. J. & Savaria, Yvon. (2006). Zero Skew Differential Clock Distribution Network, International Symposium on Circuit And Systems (ISCAS), Greece, Island of Kos.
- Zarrabi, Houman; Zilic, Zeljko; Al-Khalili, A. J. & Savaria, Yvon. (2007). A methodology for parallel synthesis of zero skew differential clock distribution networks, Joint Conference of MWSCAS/NEWCAS, Montreal, Canada.
- Zhang, H.; Varghese George & Rabaey, J. M. (2000). Low-swing on-chip signaling techniques: effectiveness and robustness, IEEE Trans. on VLSI Syst., Volume 8, Issue 3, pp. 264 – 272.



## **VLSI**

Edited by Zhongfeng Wang

ISBN 978-953-307-049-0

Hard cover, 456 pages

**Publisher** InTech

**Published online** 01, February, 2010

**Published in print edition** February, 2010

The process of Integrated Circuits (IC) started its era of VLSI (Very Large Scale Integration) in 1970's when thousands of transistors were integrated into one single chip. Nowadays we are able to integrate more than a billion transistors on a single chip. However, the term "VLSI" is still being used, though there was some effort to coin a new term ULSI (Ultra-Large Scale Integration) for fine distinctions many years ago. VLSI technology has brought tremendous benefits to our everyday life since its occurrence. VLSI circuits are used everywhere, real applications include microprocessors in a personal computer or workstation, chips in a graphic card, digital camera or camcorder, chips in a cell phone or a portable computing device, and embedded processors in an automobile, et al. VLSI covers many phases of design and fabrication of integrated circuits. For a commercial chip design, it involves system definition, VLSI architecture design and optimization, RTL (register transfer language) coding, (pre- and post-synthesis) simulation and verification, synthesis, place and route, timing analyses and timing closure, and multi-step semiconductor device fabrication including wafer processing, die preparation, IC packaging and testing, et al. As the process technology scales down, hundreds or even thousands of millions of transistors are integrated into one single chip. Hence, more and more complicated systems can be integrated into a single chip, the so-called System-on-chip (SoC), which brings to VLSI engineers ever increasingly challenges to master techniques in various phases of VLSI design. For modern SoC design, practical applications are usually speed hungry. For instance, Ethernet standard has evolved from 10Mbps to 10Gbps. Now the specification for 100Mbps Ethernet is on the way. On the other hand, with the popularity of wireless and portable computing devices, low power consumption has become extremely critical. To meet these contradicting requirements, VLSI designers have to perform optimizations at all levels of design. This book is intended to cover a wide range of VLSI design topics. The book can be roughly partitioned into four parts. Part I is mainly focused on algorithmic level and architectural level VLSI design and optimization for image and video signal processing systems. Part II addresses VLSI design optimizations for cryptography and error correction coding. Part III discusses general SoC design techniques as well as other application-specific VLSI design optimizations. The last part will cover generic nano-scale circuit-level design techniques.

### **How to reference**

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Houman Zarrabi, Zeljko Zilic, Yvon Savaria and A. J. Al-Khalili (2010). On the Efficient Design & Synthesis of Differential Clock Distribution Networks, VLSI, Zhongfeng Wang (Ed.), ISBN: 978-953-307-049-0, InTech, Available from: <http://www.intechopen.com/books/vlsi/on-the-efficient-design-amp-synthesis-of-differential-clock-distribution-networks>

# INTECH

open science | open minds

## InTech Europe

University Campus STeP Ri  
Slavka Krautzeka 83/A  
51000 Rijeka, Croatia  
Phone: +385 (51) 770 447  
Fax: +385 (51) 686 166  
[www.intechopen.com](http://www.intechopen.com)

## InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai  
No.65, Yan An Road (West), Shanghai, 200040, China  
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元  
Phone: +86-21-62489820  
Fax: +86-21-62489821

IntechOpen

IntechOpen

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen