

# Design and Analysis of Clock Gating Elements

S. Ravi\*, Subhajit Sinha, R. Adithyan and Harish M. Kittur

VLSI Division, SENSE, VIT University, Vellore - 632014, Tamil Nadu, India; msravi@vit.ac.in, subhaece1992@gmail.com; radi94@gmail.com; kittur@vit.ac.in

## Abstract

**Background/Objectives:** As the complexity of system on chip increases the timing sign-off becomes a challenging task for STA (Static Timing Analysis) engineer. **Methods/Statistical Analysis:** Due to the wide usage of IP's, change in power and skew may occur among different regions of chip. To address this issue clock gating and zero skew algorithms are mainly used. It is good design idea to turn off the clock when it is not needed. **Findings:** This paper proposes a new NOR/OR cells with different driving strengths and a new tunable delay element. These designed cells can be used in automatic clock gating which is supported by modern EDA tools. The proposed method of clock buffers with different sizes are designed and compared with ISCAS89 Benchmark circuits (s35932, s38417 and s38584). These components are designed and tested using Cadence ICFB and SOC encounter P&R tool. **Applications/Improvements:** This clock gating elements can be used in any System-on-a-Chip (SoC) application where minimum skew is required.

**Keywords :** Clock Gating, Delay Matching, Skew, Tunable Delay, Type Matching

## 1. Introduction

With the decrease of feature sizes and increase of clock frequencies in integrated digital circuits, power consumption has become a major concern for modern integrated circuit designs. Power dissipation has a dynamic component, due to the switching of active devices and a static component, due to the leakage of inactive devices. Clock gating is one of the most effective and widely used techniques for saving clock power<sup>1-15</sup>. The clock net is one of the nets with the highest switching density, resulting in high power dissipation. A promising technique is to reduce the power dissipation of the clock net is selectively stopping the clock in parts of the circuit, called "clock gating". It is very well integrated into semi-custom design flows nowadays. By gating the clock, the switching activity of the clock signal is reduced. However, clock gating circuitry itself occupies chip area and consumes additional power; therefore a judicious selection of circuit is important. So care should be taken for the size and power consumption of gated cells, clock tree consumes more than 60% of dynamic power. The components of this power are: Power consumed by combinatorial logic whose values are changing on each clock edge and power

consumed by flip-flops and the power consumed by the clock buffer tree in the design.

In this paper we proposed a new delay matching cells and a new design of tunable clock buffer which can be drop-in replacement for the existing clock buffer. The tunable clock buffer is capable of producing different delay values and equal rise and fall time.

## 2. Techniques to Implement Clock Gating

Clock gating is a technique to reduce the power consumption of network by stopping the clock input to a segment of the circuit when it is not needed to avoid unnecessary clock transition as shown in Figure 1.

Clock gating logic can be insert into a design using variety ways:

- By inserting enable into the RTL code in such a way that it can be translated into clock gating.
- Using ICG cells clock gating can be inserted into the design manually by RTL designers.
- Using clock gating tools, we can also insert clock gating logic in to the design.

\*Author for correspondence

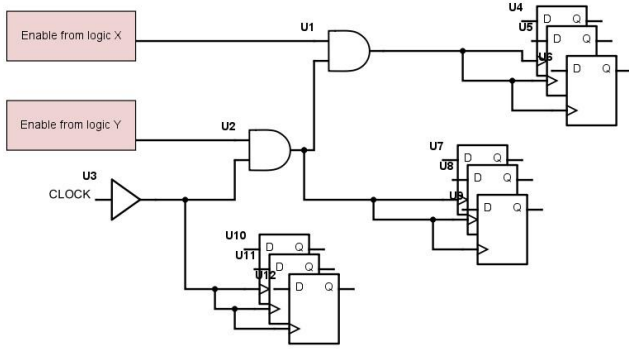


Figure 1. A basic clock gating circuit<sup>1</sup>.

There are basically two technique to implement clock gating:

- Type matching<sup>2</sup>.
- Delay matching<sup>1</sup>.

In type-matching clock gating we use same logic gates in same levels as shown in Figure 2.

We can see in level 2 we have two gates U9 and U11 both are and gates as well as in level 3 we have four gates U12, U13, U14, U15 all are same type of gate which is and gate.

In delay matching clock gating we use cells with same timing requirements as described in Figure 3.

We can see in level 2 we have gates U9 and U11 but the two gates are different. U9 is a gate and U11 is a buffer. As both the gates have same timing requirements the skew can be minimized.

The main disadvantage of delay matching clock gating is to imply delay matching clock gating we need to develop delay matching cells. The main disadvantage of type matching clock gating algorithm is described using Figure 4.

If the above tree has to be implemented in type matching tree then it has to be implemented as shown in Figure 5. As we can see that implementing the above tree in type matching concept, there is one extra level added to make as a type matching and also the complexity of the tree is increasing. It also increases chip area, latency, power consumption.

Where as in delay matching we can just implement the tree, provided all the gates in that level are having same timing properties. And in addition to this some more levels with dummy cells will be added due to load balancing algorithm<sup>11</sup>.

So that is why we choose delay matching clock gating over type matching clock gating.

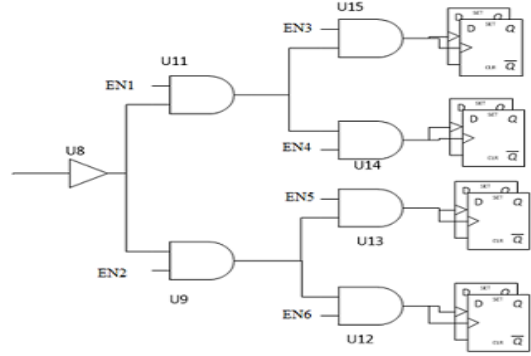


Figure 2. A typical type matching Clock tree<sup>1</sup>.

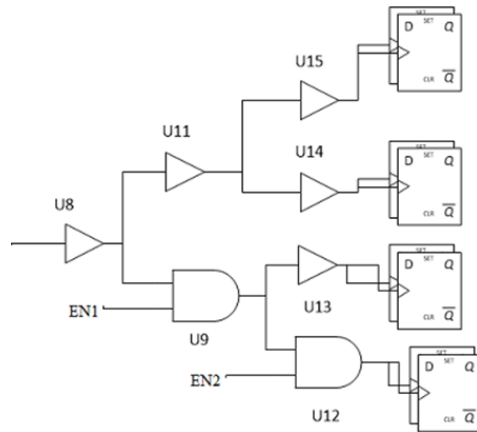


Figure 3. A typical delay matching clock tree<sup>1</sup>.

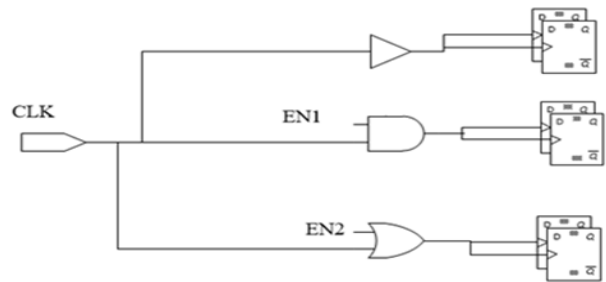


Figure 4. OR gate difficulty.

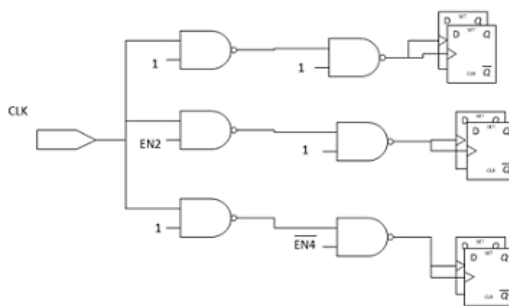


Figure 5. Implementation of Figure 4 with type matching<sup>1</sup>.

### 3. Design of Delay Matching Cells

Basically, delay matching clock gating require delay matching cells to implement. The delay matching cells must have same timing requirement. Delay matching cells must follow the below properties:

- In a standard cell library if we have clock buffer with kX driving capability, where k = 1, 2, 3, 4, 6, 8, 12, 16, 20. Then there will be logic gate cells with kX driving capability. like, if we take OR gate, the OR gate with kX driving capability called CKORkX and it's clock buffer counterpart called CKBUFkX .
- The input capacitance of CKORkX and CKBUFkX must be same.
- The rise time, fall time, rise delay, fall delay of CKORkX and CKBUFkX must be same.

To designing the delay matching cells we have followed the same procedure as in<sup>1</sup>. The input capacitance of designed gated cells are calculated and tabulated in Table 1. From the table it can be observed that the input capacitance of INVkX, NANDkX and NORkX will be approximately equal. And the input capacitances of BUFkX, ANDkX, ORkX. This is because we are forcing the timing properties of the cells to be similar. The delay-matching cells are characterized and added into the standard cell library. Such a library is called delay-matching library. The rise time and fall time tabulated in Table 1 and Table 2 are calculated after rcx extraction. It can be observed that the rise time and fall time of the CKNANDkX and CKNORkX are similar to the CKINVkX.

The cell height and Power/Ground rail widths of a delay-matching cell are made same compare with the

other cells in an industrial standard cell library GPDK 90nm. In<sup>1</sup> it should be noted that NOR/OR is implemented using all NAND gates.

### 4. Proposed Tunable Buffer

A tunable delay buffer as shown in Figure 6 designed to produce multiple delay values and it's best fit for post-silicon tuning. Basically two types of tunable buffer are used widely:

- Current starved buffer<sup>13</sup>.
- Capacitive load<sup>16</sup>.

In capacitive load<sup>16</sup> tunable buffer, we add capacitor load in between two inverter, transistor use as switch which controlled by capacitor control bank.

In current starved inverter<sup>13</sup> based tunable buffer, extra pull up and pull down transistor used to tuning delay of the buffer. In previous research they used adjustable delay buffer which can able to produce two delay values and for more delay value we need to add another ADB in parallel, which in result increase buffer area which is not acceptable. Various tunable buffer and their insertion techniques discussed in other paper<sup>17-21</sup>. In this paper we proposed a new design of adjustable delay buffer which can produce more delay values without increasing the area. Also we design upto 20X cells and values of rise time, fall time and propagation delay tabulated in Table 3.

### 5. Proposed Solution

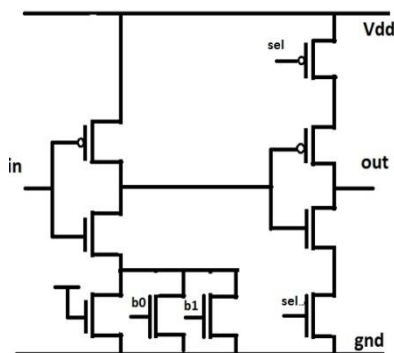
As we have seen in<sup>1</sup>, the implementation of NOR/OR using only NAND gates will result in more no. of transis-

**Table 1.** Input capacitance (pF) and Rise time ( $\mu$ s) of gated cells

	Input Capacitance (pF)						Rise Time ( $\mu$ s) of the gated cells					
	CKINV	CKNAND	CKNOR	CKBUF	CKAND	CKOR	CKINV	CKNAND	CKNOR	CKBUF	CKAND	CKOR
1X	0.0035	0.0041	0.0042	0.0045	0.0048	0.005	20.87	20.7	20.6	3.415	3.47	3.40
2X	0.007	0.00732	0.00837	0.0030	0.0043	0.0044	25.79	25.68	25.5	5.094	5.06	5.1
3X	0.0102	0.0106	0.0124	0.007	0.0075	0.008	32.48	32.4	32.7	7.4	7.8	7.94
4X	0.0135	0.014	0.0164	0.007	0.008	0.0084	37.08	37.42	37.69	10.3	10.39	10.42
6X	0.01963	0.020	0.025	0.01020	0.0107	0.01201	40.5	40.8	40.39	13.1	13.22	13.33
8X	0.02588	0.0267	0.038	0.01330	0.01387	0.1567	43.7	43.5	44.0	16.7	16.9	16.84
12X	0.03867	0.04	0.045	0.0190	0.020	0.023	46.4	46.66	46.75	20.45	20.48	20.467
16X	0.05	0.0519	0.064	0.026	0.028	0.029	48.9	48.71	48.6	25.09	25.10	25.12
20X	0.064	0.067	0.079	0.035	0.035	0.037	50.6	50.45	50.3	29.79	29.85	29.90

**Table 2.** Fall time ( $\mu\text{s}$ ) of gated cells

	CKINV	CKNAND	CKNOR	CKBUF	CKAND	CKOR
1X	20.98	20.85	20.9	3.41	3.57	3.5
2X	26.01	26.15	26.25	5.12	5.10	5.4
3X	32.69	32.71	32.84	7.6	7.75	7.79
4X	37.01	37.12	37.20	10.6	10.40	10.5
6X	40.23	40.08	40.07	13.09	13.28	13.39
8X	43.65	43.71	44.80	16.9	16.7	16.94
12X	46.57	46.89	46.71	20.5	20.64	20.48
16X	48.98	48.74	48.54	25.1	25.75	25.05
20X	50.75	50.68	50.78	29.85	29.8	29.75



**Figure 6.** CSI Based adjustable delay buffer.

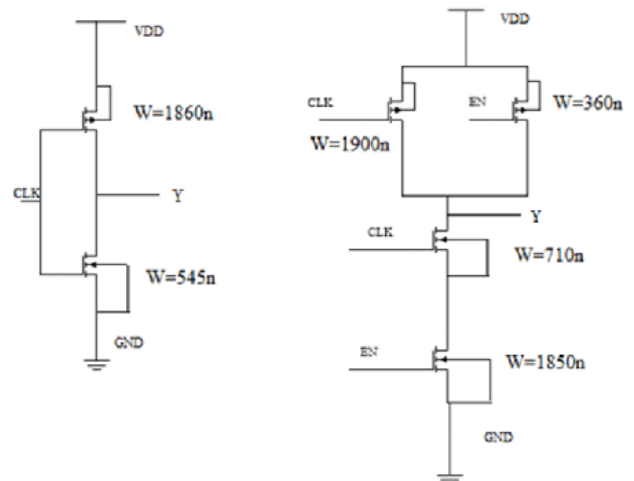
**Table 3.** Rise time, fall time and propagation delay of proposed design

	Rise Time (ps)	Fall Time (ps)	Propagation delay(ps)
1X	164.9	165.3	100.9
2X	157.2	157.6	97.29
3X	134.6	134.5	95.53
4X	167.6	167.0	94.85
6X	163.5	163.8	93.73
8X	178.0	178.0	92.75
12X	180.3	180.2	91.00
16X	163.7	163.9	86.85
20X	155.1	155.7	85.89

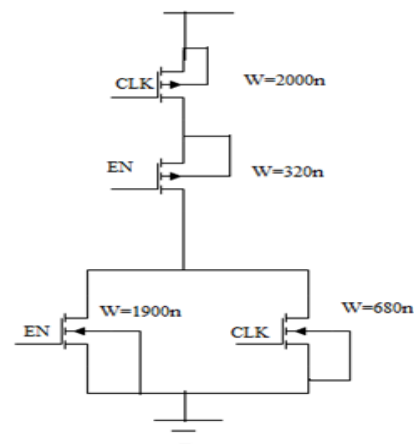
tors as shown in Figure 7 which will increase the latency, cell area and power consumed by the clock tree. Figure 8 proposes the architecture for NOR gate and OR gate respectively. Transistor count is as follows:

### 5.1 Existing Solution

No. of transistors when NOR implemented only using NAND = 16.



**Figure 7.** Implementation of NAND1X and INV1X as in<sup>1</sup>.



**Figure 8.** Implementation of NOR1X. (proposed design)

No. of transistors when OR implemented by only NAND = 12.

### 5.2 Proposed Solution

No. of transistors with NOR based implementation = 4.

No. of transistors with OR implementation = 6.

## 6. Simulation Results

For the purpose of comparisons, we implement the following clock gating methods which each of the methods are explained in detail in section 6.2 and results are tabulated in Table 4.

The following parameters are measured and simulated:

- Latency.
- Skew.

**Table 4.** The average values of the latency, skew, Clock Tree Wire length (CTW), Clock Tree Area (CTA), Power, Worst Buffer Slew, Worst sink slew of the benchmark circuits

Corners	Gating Methods	Latency (ps)	Skew (ps)	CTW (um)	CTA (um <sup>2</sup> )	Power (mW)	Worstcase Buffer Slew (ps)	Worst case Sink Slew (ps)
Best Corner	NG	112.12	16.42	12251	1264	2.87	16.33	15.83
	NML	152.95	59.76	12271	1267	2.13	24.52	25.59
	DM	116.63	21.03	12458	1673	2.39	18.85	17.75
	SmallT	202.27	13.58	13387	2558	2.58	27.90	18.44
	BigT	146.07	23.84	12517	2349	2.73	22.39	21.29
	PM	<b>114.85</b>	<b>19.20</b>	<b>12426</b>	<b>1660</b>	<b>2.25</b>	<b>17.08</b>	<b>17.68</b>
Worst Corner	NG	210.94	16.57	12251	1264	1.98	24.82	24.11
	NML	310.65	123.56	12271	1267	1.46	44.25	45.32
	DM	214.59	19.91	12458	1673	1.56	26.44	25.35
	SmallT	440.69	17.49	13387	2558	1.60	56.38	33.45
	BigT	276.08	21.73	12517	2349	1.83	35.74	33.50
	PM	<b>212.50</b>	<b>19.75</b>	<b>12426</b>	<b>1660</b>	<b>1.52</b>	<b>25.89</b>	<b>25.00</b>
Typical Corner	NG	146.32	15.97	12251	1264	2.31	19.40	18.95
	NML	206.78	80.78	12271	1267	1.70	31.63	32.46
	DM	150.10	19.84	12458	1673	1.86	21.45	20.18
	SmallT	280.47	14.45	13387	2558	1.90	38.10	23.00
	BigT	189.48	22.01	12517	2349	2.03	27.02	25.09
	PM	<b>148.08</b>	<b>19.79</b>	<b>12426</b>	<b>1660</b>	<b>1.80</b>	<b>20.0</b>	<b>19.00</b>

- Clock tree wire length.
- Clock Tree Area.
- Power.
- Worst Buffer Slew.
- Worst Sink Slew.
- NG: non-gating.
- NML: Normal clock gating which is inbuilt in encounter tool.
- DM: Delay-matching clock gating.
- SmallT: Type-matching using small-drive gated cells.
- BigT: Type-matching using also large-drive gated cells.
- PM: Proposed Methodology.
- And all of these methods are simulated in 3 library conditions.
- Typical (typical.lib).
- Worst (slow.lib).
- Best (fast.lib).

## 6.1 Bench Mark Circuits

We evaluate the above clock gating approaches using three large ISCAS89 Benchmark circuits (s35932, s38417 and s38584). Also we have simulated some smaller ISCAS89 benchmark circuits (which have less no of sequential elements). Just for the purpose of showing the formation of clock tree in a convenient and simple way.

## 6.2 Methods of Clock Gating

For the purpose of comparison we propose the following gating methods:

### 6.2.1 NG: Non Gating.

We first analyse a design without applying clock gating and perform clock tree synthesis with skew and slew constraint using cadence SOC encounter.

### 6.2.2 NML: Normal Clock Gating

In normal clock gating, we perform clock gating technique that is built in cadence SOC encounter.

### 6.2.3 DM: Delay Matching

In delay matching clock gating, we use delay matched cells were used for clock gating.

### 6.2.4 Small T: Type-matching using small-drive gated cells

Here we use cell library with less driving capability like 1X, 2X, 3X, 4X, 6X.

### 6.2.5 Big T

Here we use cell library with high driving capability like 8X, 12X, 16X, 20X.

### 6.2.6 PM: Proposed Methodology

When performing clock tree synthesis our delay matching library (which has our proposed structure of NOR/OR) has to be added instead of Standard cell library.

## 7. Transistor Count

#### Case I: s38417

No. of OR Gates in Clock tree: 25.

No. of transistors as per existing solution: 300.

No. of transistors as per proposed solution: 150.

#### Case II: s35932

No. of OR Gates in Clock tree: 20.

No. of transistors as per existing solution: 240.

No. of transistors as per proposed solution: 120.

#### Case III: s38584

No. of OR Gates in Clock tree: 28.

No. of transistors as per existing solution: 336.

No. of transistors as per proposed solution: 168.

## 8. Conclusion

As we can see from the result table we can conclude that delay matching approach which uses our OR gates is giving more feasible results than the delay matching algorithm which uses all NAND gate to construct an OR gate. We also develop a method for designing delay-matching cells. Compared with type-matching, delay-matching

attains better slew and clock latency with comparable clock skew while using much less cell area. Meanwhile delay matching ECO excels in original timing characteristics of a gated tree. There are still some problems needed to be addressed for delay-matching and type-matching approaches, the second level of the clock tree employs an inverter, a buffer and an OR gate. With type-matching, the OR gate and buffer each can be replaced by two serially connected NAND gates. However, this cannot be done for the inverter. Similarly, with delay-matching, the OR gate and buffer have similar timing characteristics, but the inverter and buffer do not have the same. So care should be taken while facing these types of cases.

## 9. References

1. Hsu SJ, Lin RB. Clock gating optimization with delay-matching. Design Automation and Test in Europe Conference and Exhibition (DATE); Grenoble. 2011 Mar 14-18. p. 1-6.
2. Chang CM, Huang SH, Ho YK, Lin JZ, Wang HP, Lu YS. Type-matching clock tree for zero skew clock gating. 45th ACM/IEEE Design Automation Conference, DAC'08; Anaheim, CA. 2008 Jun 8-13. p. 714-9.
3. Li L, Choi K, Park S, Chung MK, Novel RT. Level methodology for low power by using wasting toggle rate based clock gating. 2009 International SoC Design Conference, (ISOC); Busan. 2009 Nov 22-24. p. 484-7.
4. Arbel E, Eisner C, Rokhlenko O. Resurrecting infeasible clock-gating functions. 46th ACM/IEEE Design Automation Conference, DAC'09; San Francisco, CA. 2009 Jul 26-31. p. 160-5.
5. Wu Q, Pedram M, Wu X. Clock gating and its application to low power design of sequential circuits. IEEE Transactions on Circuits and Systems - I: Fundamental, Theory and Applications. 2000 Mar; 47(3):415-20.
6. Borkovic D, McElvain KS. Reducing clock skew in clock gating circuits. USA; Synplicity Inc: 2003.
7. Garret D, Stan M, Dean A. Challenges in clock gating for a low power ASIC methodology. International Symposium on Low Power Electronics and Design; San Diego, CA, USA. 2002 Aug 17. p. 176-81.
8. Raghavan N, Akella V, Bakshi S. Automatic insertion of gated clocks at register transfer level. 12th International Conference on VLSI Design; Goa, 1999 Jan 7-10. p. 48-54.
9. Farrahi AH, Chen C, Srivastava A, Tellez G, Sarrafzadeh M. Activity driven clock design. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 2001 Jun; 20(6):705-14.

10. Hathaway DJ. Method for making integrated circuits having gated clock trees, USA; International Business Machines Corporation: 2003.
11. Chang CM, Huang SH, Ho YK, Lin JZ, Wang HP, Lu YS. Type-matching clock tree for zero skew clock gating. 45th ACM/IEEE Design Automation Conference, DAC'08; Anaheim, CA. 2008 Jun 8-13. p. 714-9.
12. Cheung CC, Au KD. Clock gating cell for used in a cell library. USA; LSI Logic Corporation: 2003.
13. Suresh VB, Burleson WP. Variation aware design of post-silicon tunable clock buffer. 2014 IEEE Computer Society Annual Symposium on VLSI (ISVLSI); Tampa, FL. 2014 Jul 9-11. p. 1-6.
14. Lin KY, Lin HT, Ho TY. An efficient algorithm of adjustable delay buffer insertion for clock skew minimization in multiple dynamic supply voltage designs. 2011 Proceedings of the 16th Asia and South Pacific Design Automation Conference (ASP-DAC); 2011. p. 825-30.
15. Vennelakanti S, Saravanan S. Design and analysis of low power memory built in self test architecture for SoC based design. Indian Journal of Science and Technology. 2015 Jul; 8(14):1-5.
16. Kim J, Joo D, Kim T. An optimal algorithm of adjustable delay buffer insertion for solving clock skew variation problem. Proceedings of the 50th Annual Design Automation Conference (DAC); Austin, TX. 2013 May 29-Jun 7. p. 1-6.
17. Mueller J, Saleh R. A tunable clock buffer for intra-die PVT compensation in Single-Edge Clock (SEC) distribution networks. 9th International Symposium on Quality Electronic Design, ISQED'08; San Jose, CA. 2008 Mar 17-19. p. 572-7.
18. Kobenge SB, Yang H. A power efficient digitally programmable delay element for low power VLSI applications. 1st Asia Symposium on Quality Electronic Design, ASQED; Kuala Lumpur. 2009 Jul 15-16. p. 83-7.
19. Su YS, Hon WK, Yang CC, Chang SC, Chang YJ. Value assignment of adjustable delay buffers for clock skew minimization in multi-voltage mode designs. IEEE/ACM International Conference on Computer-Aided Design - Digest of Technical Papers, ICCAD 2009; San Jose, CA. 2009 Nov 2-5. p. 535-8.
20. Tsai JL, Zhang L, Chen CC. Statistical timing analysis driven post-silicon-tunable clock-tree synthesis. IEEE/ACM International Conference on Computer-Aided Design, ICCAD'05; 2005 Nov 6-10. p. 575-81.
21. Chakraborty A, Duraisami K, Sathanur A, Sithambaram P, Benini L, Macii A, Macii E, Poncino M. Dynamic thermal clock skew compensation using tunable delay buffers. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. 2008 Jun; 16(6):639-49.