

HOSTED BY



ELSEVIER

Contents lists available at ScienceDirect

Engineering Science and Technology, an International Journal

journal homepage: <http://www.elsevier.com/locate/jestch>

Full length article

A buffer placement algorithm to overcome short-circuit power dissipation in mesh based clock distribution network

John Reuben^{*}, Mohammed Zackriya V., Harish M. Kittur, Mohd Shoaib

School of Electronics Engineering, VIT University, Vellore 632014, India

ARTICLE INFO

Article history:

Received 16 July 2014

Received in revised form

18 October 2014

Accepted 24 October 2014

Available online 28 November 2014

Keywords:

Mesh-based clock distribution

Mesh buffer

Short circuit power

Inter-buffer skew

Clock skew

ABSTRACT

In the recent past, Mesh-based clock distribution has received interest due to their tolerance to process variations in deep-sub micron technology. Mesh buffers are placed on the mesh to drive the large load capacitance of clock sinks and mesh wire capacitance. In this paper, we propose a buffer placement algorithm which can overcome the short circuit power dissipated in clock meshes. Our buffer placement algorithm uses clustering technique to judiciously place buffers such that short-circuit power is minimized while minimizing skew at the same time. This is verified by Monte carlo simulations incorporating process, voltage and systemic variations in NGSPICE.

Copyright © 2014, The Authors. Production and hosting by Elsevier B.V. on behalf of Karabuk University. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/3.0/>).

1. Introduction

Traditionally, tree-based clock distribution has been the preferred method in most VLSI designs. However, in Deep Sub Micron (DSM) technology, clock skew induced by process variations is posing a threat to the reliability of tree-based distribution. Mesh-based distribution, due to their redundant paths offers a variation tolerant alternative to tree-based distribution at the cost of increased power dissipation (due to increased wire capacitance) [1]. Tree driven mesh is a hybrid clock distribution scheme in which a top level clock tree feeds a leaf level mesh. Leaf level clock mesh synthesis is well studied in [1–3]. In the leaf level mesh, buffers are placed at the mesh nodes (Fig. 1) to drive the large capacitance (input capacitance of flip flops/registers called clock sinks and mesh wire capacitance).

2. Existing buffer placement algorithms

Buffer placement on the mesh is a crucial step in the design of the clock distribution network. Although buffers can be placed on

all the mesh nodes, such a liberal placement of buffers will be an inefficient way (due to power and area consumed by the buffers) to minimize the skew in a mesh. All the previous works([1–3]) place buffers at certain mesh nodes such that the clock latency to all the sinks is equalized. The buffer placed at a particular mesh node is designed to drive the capacitance load around that mesh node.

The buffers are placed using a set-cover algorithm with a discrete buffer library in Ref. [1]. In Ref. [2], the buffers are placed by Iterative Buffer Deletion algorithm (IBD). In Ref. [3], the buffers are placed at the mesh nodes according to the density of sinks in its vicinity and their proximity to the sinks. Since buffers have to be placed in hundreds of locations for designs having thousands of sinks in a clock mesh, the buffer sizing and placement is typically automated. Hence a discrete buffer library, $Library = \{b_1, b_2, \dots, b_n\}$ is used. For example, if the load capacitance at mesh nodes varies between 0 and 300 fF in a design, a buffer library of 6 buffers can be used: b_1 to drive 0–50 fF, b_2 to drive 51–100 fF, b_3 to drive 101–150 fF... b_6 to drive 251–300 fF. Each of the buffers is designed/sized to drive its load capacitance with a particular slew constraint. Since clock signals are critical timing signals, they are tightly slewed, typically at 10% of their time period. Hence an ideal buffer to feed a 1 GHz clock to a clock mesh must have a slew no greater than 100 ps when driving its corresponding load.

All the aforementioned algorithms place buffers using a discrete buffer library which may be inefficient i.e to drive a 101 fF load or a 149 fF load, the buffer placement algorithm might place the same

^{*} Corresponding author.

E-mail addresses: johnreuben@vit.ac.in (J. Reuben), mdzackriya@vit.ac.in (M. Zackriya V.), kittur@vit.ac.in (H.M. Kittur), shoaib.saleem0513@gmail.com (M. Shoaib).

Peer review under responsibility of Karabuk University.

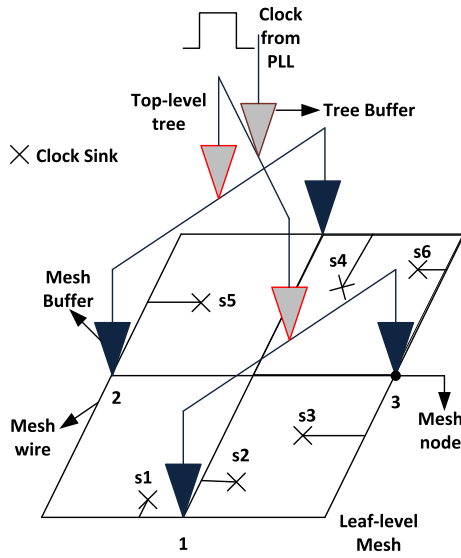


Fig. 1. Tree driven Mesh.

buffer- b_3 . Buffers contribute to area penalty which is a very costly commodity in sub micron designs. More importantly, there will be Short-Circuit (SC) power dissipation between the buffers in the mesh. As illustrated in Fig. 2, there is a short-circuit path formed between adjacent buffers when the clock reaches the mesh buffers at different times. In this paper, we refer to the power dissipated due to the formation of this short-circuit path as short-circuit power dissipation. This short-circuit power is different from the one inherent in CMOS switching which can be made negligible by careful design. The difference of clock arrival at mesh buffers is attributed to process variations in the top level tree driving the mesh ([1]). The existing buffer placement algorithms overlook this short circuit power dissipation while placing the buffers on the mesh. They only concentrate on achieving minimum skew with minimum number of buffers on the mesh. Hence we need an algorithm which places buffers such that SC power dissipation is minimized while conserving the clock skew. In this paper, we propose a clustering based buffer placement algorithm which has the following properties:

1. Our buffer placement algorithm is short-circuit power 'aware'. Since buffers are placed near the centroids of clusters, they are

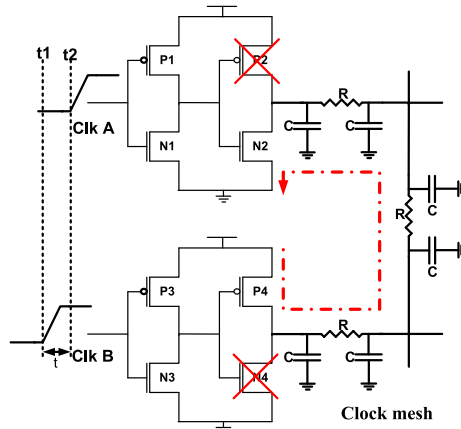


Fig. 2. Short circuit path in a clock mesh driven by buffers.

physically spread apart minimizing the possibility of forming short-circuit paths. This will reduce the SC power dissipation as will be verified by simulations in Section 5.1.

2. Computationally less intensive than the buffer placement algorithms of [1,2] (Section 5.2)
3. More robust to variations in top-level tree driving the mesh (Section 5.3).

3. Clustering based Buffer Placement Algorithm

The distribution of clock sinks is highly uneven in real designs. This is evident from the ISPD2010 clock network synthesis benchmarks ([4]) which are based on real 45 nm microprocessor designs of INTEL (Fig. 4). The proposed buffer placement is based on clustering which is a technique to find similarity in data. Clustering is often used in image processing and data mining to find patterns in images/data sets. Here we use clustering to find groups of sinks which can be assigned a single buffer. The detailed steps are:

1. Start with the sink closest to the lower left corner of the mesh. Let it be sink s_1 of the first cluster c_1
2. Find m nearest sinks to s_1 and keep adding them to cluster c_1 till the capacitance of sinks in a cluster reaches a target capacitance, say 100 fF
3. Find the capacitance centroid of the sinks in the cluster c_1 as follows

$$(x_{c1}, y_{c1}) = \left(\frac{\sum_{i=1}^m x_i \times C_{si}}{\sum_{i=1}^m C_{si}}, \frac{\sum_{i=1}^m y_i \times C_{si}}{\sum_{i=1}^m C_{si}} \right) \quad (1)$$

where $s_i, i = 1$ to m represents the m sinks belonging to cluster c_1 and C_{si} represents the capacitance of sink s_i having co-ordinates (x_i, y_i) .

4. Place a buffer which can drive the target load (100 fF) at the mesh node closest to the capacitance centroid of cluster c_1 .
5. Find the next sink closest to the last sink of cluster c_1 and repeat Steps (2)–(4) for the remaining sinks till all sinks are covered by a cluster

Fig. 3 illustrates the clustering algorithm. A direct application of the algorithm will result in clusters spanning large areas when the sinks distribution is very sparse (like cluster c_2). This will lead to increased clock latency which will affect the skew of the mesh. To avoid this, an imaginary box is used internally in Step (2) to limit the cluster area even when the target capacitance is not reached.

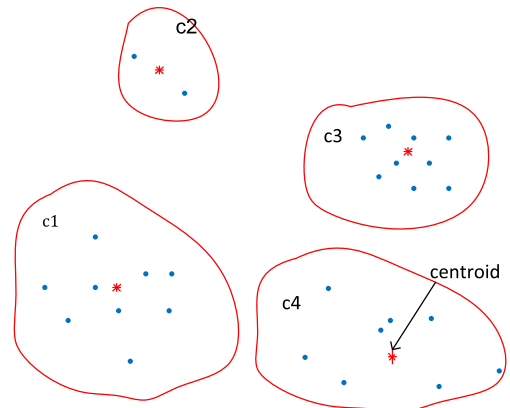


Fig. 3. Formation of clusters of clock sinks which can be assigned a single buffer.

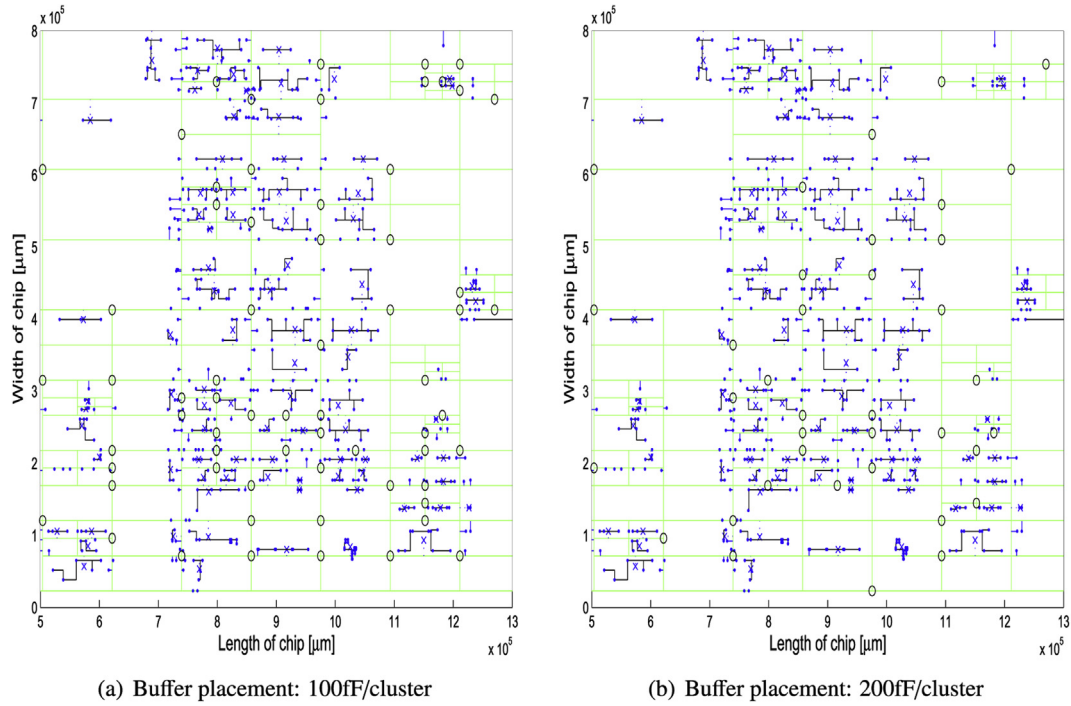


Fig. 4. Portion of ISPD2010 benchmark: 'x' denotes clock sinks, 'o' denotes buffers.

4. Mesh formation and buffer placement for benchmark circuit

In our previous work ([3]), we presented a capacitance driven mesh formation algorithm which forms a minimum wire-length, non-uniform mesh according to the density of capacitance in the chip. Clock sinks are connected to the mesh edges by a combination of steiner tree and stubs. Benchmarks 6,7 and 8 of ISPD2010 clock network synthesis contest have 981, 1915 and 1134 clock sinks respectively. The clock sinks in these benchmarks are unevenly distributed making buffer placement challenging. We formed the capacitance driven mesh of [3] for these benchmarks. We then applied our proposed buffer placement algorithm and placed the buffers at mesh nodes. Fig. 4 shows benchmark6 with capacitance driven mesh formed and buffers placed according to our proposed clustering algorithm (target capacitance of 100 fF and 200 fF). The buffers are placed only at those mesh nodes near the capacitance centroids of clusters.

5. Mesh modeling and simulation

The mesh wire has got resistance and capacitance. We use a distributed model of the wire which is an accurate model when compared to the lumped modeling. According to [5], the lumped RC model is a pessimistic model for a resistive-capacitive wire. So we used the π -model to model the mesh wires. Because of the importance of the distributed model, most circuit simulators have a built-in distributed rc models of high accuracy. NGSPICE supports the uniform-distributed rc-line model (URC) of Berkeley's SPICE3 simulator. This model approximates the rc line as a network of lumped RC segments with internally generated nodes. The mesh model is shown in Fig. 5.

Clock mesh simulation is computationally intensive due to the thousands of wire segments which have to π -modeled. Sliding window scheme ([6]) reduces the computational complexity by splitting the mesh into windows where the mesh is accurately modeled. Outside the window, the mesh is approximately modeled.

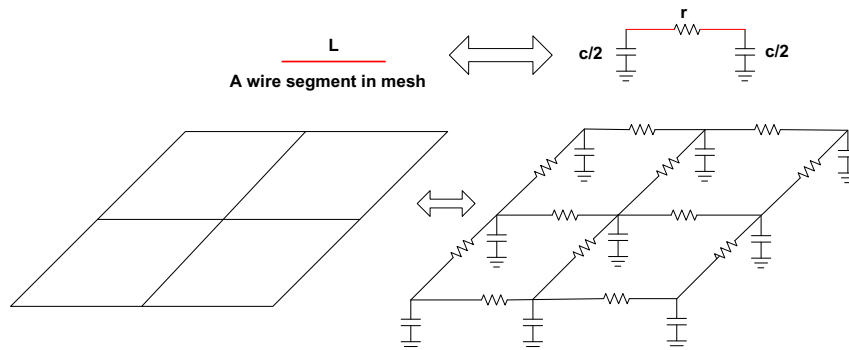


Fig. 5. Mesh model.

Table 1
Simulation conditions and variations.

Parameter	Value
V_{DD}	1 V
Clock frequency	1 GHz
Resistance of mesh wire	0.3 Ohm/ μm
Capacitance of mesh wire	0.16 fF/ μm
Parameter	Variation (Standard deviation $\sigma = 5\%$ of nominal)
Channel length	Gaussian with 3σ variation
V_{th}	Gaussian with 3σ variation
V_{DD}	Gaussian with 3σ variation
Inter-buffer skew	50 ps

i.e the resistance and capacitance of wires joining the sinks to the mesh wires (stubs) are ignored. The clock latency is evaluated for the sinks inside the window and then the window is moved to evaluate the latency at all the other sinks ([3]). We have split the entire mesh into 4 windows as illustrated in the simulation flow (Fig. 7).

5.1. Simulation conditions and results

The simulation conditions are tabulated in Table 1. 45 nm PTM files were used to simulate the mesh buffers. The clock signal feeding the mesh has a slew of 50 ps to model the degradation in the clock reaching the mesh buffers. We studied benchmark6 for 4 different cases of buffer placement – target capacitance of clusters being 100 fF, 150 fF, 200 fF and 250 fF. For each case, we placed an appropriately sized buffer which can drive the load without slew violation (Table 2).

To study the effectiveness of our buffer placement and to accurately predict the skew post-fabrication, we performed Monte

Carlo simulations with statistical variations. Threshold voltage V_{th} and channel length L_{eff} are the two key device parameters that are subject to variation in DSM technology [7]. We did not consider variations in interconnects since transistor variations are dominant compared to interconnect variations [8]. Hence we varied V_{th} and channel length L_{eff} by 5% of their nominal value according to a gaussian distribution. V_{DD} variation is also incorporated. As already stated, the clock signal will not reach all the mesh buffers at the same time due to process variations in the top level tree driving the mesh. This will lead to short circuit power dissipation between adjacent buffers in the mesh (Fig. 2). To incorporate this phenomenon in simulation, the clock arrival time at the mesh buffers is varied randomly such that the maximum difference in arrival times between any 2 buffers does not exceed 50 ps. This is called ‘Inter-buffer skew’ and illustrated in Fig. 6 (the ‘delay’ parameter of the PULSE command in NGSPICE is varied to implement different arrival times of the clock at different buffers).

Following the simulation flow in Fig. 7, we performed 200 Monte Carlo (MC) simulations on benchmark 6,7 and 8. The global skew was calculated from the clock latency at sinks. The total power was calculated by integrating the current drawn from V_{DD} . Table 3 shows the average skew and average power dissipation for the 4 different cases of buffer placement on benchmark6 after 200 MC simulations. We observe that bigger the capacitance/cluster, lesser buffers have to be placed. But the skew increases due to increased imbalance in the path between buffers and sinks. Interestingly, the total power dissipation remains the same whether 116 buffers or 60 buffers are placed on the mesh. This is because the total load (C_{load}) to be driven by all the buffers together remains the same (total load is the capacitance of sinks and the capacitance of the mesh wires) for a particular benchmark. Hence the dynamic power dissipation ($C_{load} \cdot V_{DD}^2 \cdot f$) will remain the same. As depicted

Table 2
Sizes of buffers to drive different cluster sizes.

Case	Capacitance/cluster	First inverter			Second inverter		
		W/L of P1 [nm]	W/L of N1 [nm]	W_p/W_n	W/L of P2 [nm]	W/L of N2 [nm]	W_p/W_n
1	100 fF	82/45	57/45	1.438	1656/45	1177/45	1.407
2	150 fF	114/45	82/45	1.390	2451/45	1718/45	1.426
3	200 fF	151/45	110/45	1.372	3258/45	2281/45	1.428
4	250 fF	189/45	130/45	1.454	4073/45	2852/45	1.428

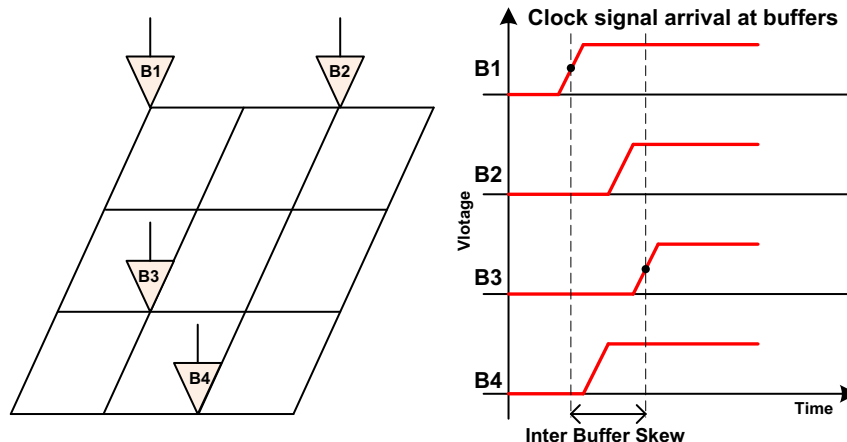


Fig. 6. Inter-buffer skew: The maximum difference in clock arrival time at buffers.

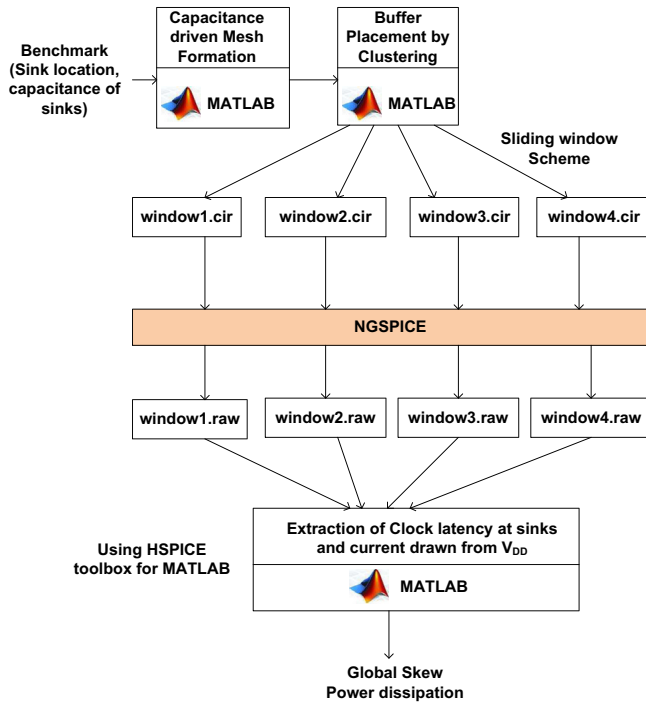


Fig. 7. Simulation flow.

in Table 3, when the capacitance/cluster is 100 fF, 116 buffers are needed while if the capacitance/cluster is 200 fF, only 60 buffers are needed because each buffer in this case is of a larger size compared to the 100 fF case (Table 2). Since the total power dissipation remains constant irrespective of the number of buffers (Table 3), it implies that short circuit power component of total power does not change with the increase in the number of buffers. This proves that the short circuit power component in Table 3 is due to the 50 ps inter-buffer skew and not due to the increased number of buffers i.e. whether 60 buffers or 116 buffers are placed, our buffer placement doesn't aggravate short-circuit power consumption.

The skew and power dissipation for benchmarks 6,7,8 for clustering based buffer placement (for capacitance/cluster of 100 fF) under the simulation/variation conditions of Table 1 is tabulated in Table 4. Among the buffer placement algorithms published in literature, the Iterative Buffer Deletion (IBD) of [2] attains the best skew and power dissipation on ISPD2010 benchmarks. As observed in Table 4, our buffer placement achieves lower power dissipation compared to IBD of [2] for benchmark6 and almost the same power dissipation for benchmarks 7 and 8, at the cost of increased skew. Although our buffer placement is inferior to [2] in terms of skew, our algorithm is computationally much less complex than [2].

5.2. Computational complexity of buffer placement algorithms: a qualitative comparison

[1] uses a set-cover algorithm to place the buffers. In this algorithm, the cost of placing a buffer at a mesh node is calculated using

a cost function and buffers are placed such that all the sinks are covered with minimum cost. The time complexity of the algorithm is $O(n \cdot \beta)$ where n is the number of candidate mesh nodes and β is the size of the buffer library ([9]). Guthaus [2] uses an iterative buffer deletion algorithm for buffer placement. In this algorithm, minimum sized buffers are placed at all the mesh nodes initially. Then, buffers are sized using sensitivity-based linear programming method. Smaller sized buffers are then removed iteratively till the skew exceeds a certain user specified bound. According to Ref. [2], the time complexity of their algorithm is 7 times more than that of the set-covering approach of [1]. In our approach, the time complexity is highly reduced because we don't do buffer sizing i.e. we use uniformly sized buffers throughout the mesh. The capacitance driven mesh forms a non-uniform mesh according to the density of clock sinks. Then, the clustering based buffer placement algorithm forms clusters of sinks according to a particular target capacitance and places uniformly sized buffers on the mesh. Hence, in general, the time complexity is $O(k)$, where k is the number of sinks. Practically, the run-time complexity of our algorithm is the time it takes to form the clusters and find the capacitance centroid of each cluster. For benchmark7 having 1915 sinks, this time was around 7.6 s when our buffer placement algorithm was run on a 3.3 GHz Intel i3 processor with 2 GB RAM. Hence our algorithm has a much lower time complexity compared to the buffer placement of [1,2]. We are not able to compare our time complexity with the buffer placement of [2] quantitatively since CPU runtime of buffer placement alone is not provided in Ref. [2].

5.3. Performance of the clock mesh with increased inter-buffer skew

To probe the tolerance of our mesh to inter-buffer skew, we performed some simulation studies with increased inter-buffer skew. According to Ref. [10], the SC power dissipation in a clock mesh is a linear function of inter-buffer skew. Wilke [11] conducted a study to find the % contribution of this SC power to the total power as a function of inter-buffer skew (The clock arrival time at the mesh buffers was varied randomly between 0 and inter-buffer skew). The study concluded that the SC power dissipation can be as high as 50% of the total power when the inter-buffer skew is 15% of the clock period (150 ps for a 1 GHz clock). To study our clock mesh design in light of [11,12]'s observations, we increased the inter-buffer skew from 50 ps to 150 ps in steps of 25 ps for benchmark6 and performed 200 MC simulations as earlier. As the inter-buffer skew at the mesh buffers increases, the skew and power in the mesh increase only marginally as shown in Fig. 8. In fact, even for 100 ps inter-buffer skew which implies the worst case top-level tree (the acceptable skew in a clock tree must be $\leq 10\%$ of clock time period) for 1 GHz clock, the total power dissipation increases by only 1.42 mW when compared to the 50 ps case. This proves the tolerance of our design to inter-buffer skew. For 150 ps inter-buffer skew, the total power dissipation increases by 3.82 mW when compared to the 50 ps case. This increase is justified because of the unreasonable input skew at mesh buffers. Such unreasonable input skew will happen in the case a very poor top-level tree design which can be avoided. Hence our buffer placement algorithm is

Table 3
Skew and Power averaged over 200 MC simulations for benchmark6.

Capacitance/cluster	No. of buffers	Clock skew	Total power dissipation
100 fF	116	26.35 ps	21.77 mW
150 fF	79	29.18 ps	21.64 mW
200 fF	60	34.9 ps	21.88 mW
250 fF	47	38.95 ps	21.41 mW

Table 4
Performance of buffer placement of [2] vs Clustering based buffer placement.

Benchmark	IBD of [2]		Ours	
	Skew (ps)	Power (mW)	Skew (ps)	Power (mW)
Benchmark6	12	26.09	26.35	21.77
Benchmark7	7	44.71	27.76	44.1
Benchmark8	7	31.96	26.58	31.9

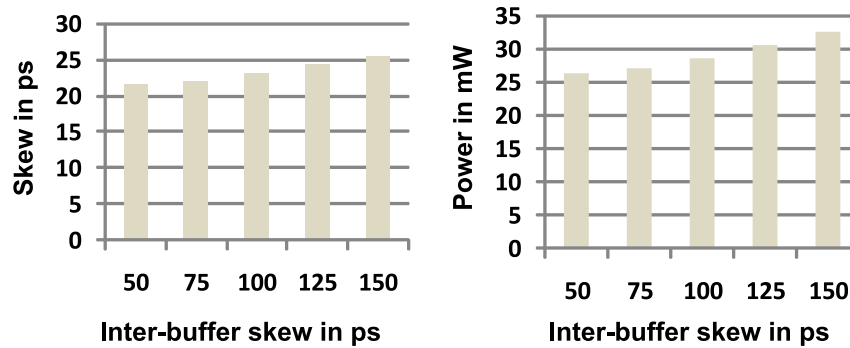


Fig. 8. Skew and power averaged over 200 MC simulations of benchmark6 with different Inter-buffer skew.

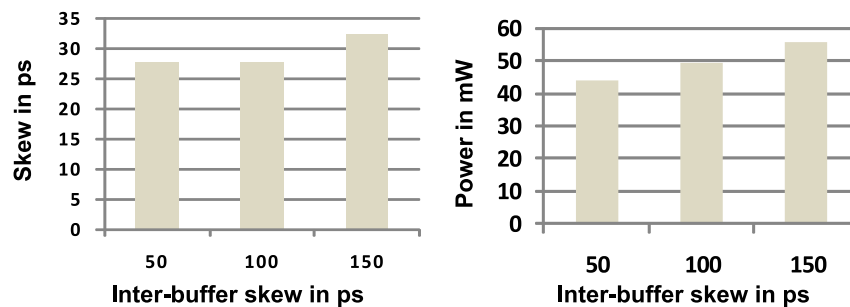


Fig. 9. Skew and power averaged over 100 MC simulations of benchmark7 with different Inter-buffer skew.

tolerant to inter-buffer skew of acceptable measure. We performed the same study on benchmark7 and plotted our observations in Fig. 9. Due to the large size of benchmark7, we performed only 100 MC simulations and varied inter-buffer skew in steps of 50 ps. Irrespective of the benchmark, our buffer placement is tolerant to inter-buffer skew and power consumption doesn't increase rapidly with increase in inter-buffer skew, as predicted by [11].

Finally, we compared the tolerance of our buffer placement with the tolerance of buffer placement of [3] by performing 200 MC simulations for an inter-buffer skew of 150 ps. While the buffer placement of [3] consumed 35.11 mW for 50 ps inter-buffer skew on benchmark6 ([3]), the total power dissipation grew rapidly to 50.96 mW for 150 ps inter-buffer skew i.e on the same mesh and under the same simulation conditions, the total power dissipation of [3] increases by 45.14% while that of the proposed work increases by 17.54% only. All the above observations establish the robustness of the proposed buffer placement algorithm.

6. Conclusion

Power dissipation in a clock mesh is increased due to short circuit power dissipation in mesh buffers. By forming clusters of sinks and placing buffers at the mesh nodes near the centroid of the clusters, we are able to minimize the short-circuit power dissipation in the mesh. When buffers are placed near centroids of clusters, they are physically spread apart, thus avoiding formation of short-circuit paths when the clock arrival time varies at the mesh buffer inputs. This comes at the cost of some increased skew but the power dissipation is reduced since short-circuit component of total power is greatly minimized. Although our buffer placement algorithm achieves the same power dissipation as the recently proposed algorithm ([2]), it is better than [2] in terms of computational complexity. We also verified that with the increase in inter-buffer skew (from 50 ps to 150 ps), the skew and power dissipation of

our mesh doesn't increase rapidly but only marginally. Hence clustering based buffer placement is also robust against variations in the top-level tree driving the mesh.

References

- [1] A. Rajaram, D. Pan, Meshworks: a comprehensive framework for optimized clock mesh network synthesis, *IEEE Trans. Comput. Aided Design Integr. Circuits Syst.* 29 (12) (2010) 1945–1958.
- [2] M.R. Guthaus, X. Hu, G. Wilke, G. Flach, R. Reis, High-performance clock mesh optimization, *ACM Trans. Des. Autom. Electron. Syst.* 17 (3) (2012) 1–33, 33.
- [3] J. Reuben, V.M. Zackriya, S. Nashit, H.M. Kittur, Capacitance driven clock mesh synthesis to minimize skew and power dissipation, *IEICE Electron. Express* 10 (24) (2013).
- [4] C.N. Sze, ISPD 2010 high performance clock network synthesis contest: benchmark suite and results, in: *Proceedings of the 19th International Symposium on Physical Design, ISPD '10*, ACM, 2010, p. 143.
- [5] Jan M. Rabaey, *Digital Integrated Circuits, a Design Perspective*, Prentice Hall, 2003.
- [6] H. Chen, C. Yeh, G. Wilke, S. Reddy, H. Nguyen, W. Walker, R. Murgai, A sliding window scheme for accurate clock mesh analysis, in: *Computer-aided Design, 2005. ICCAD-2005. IEEE/ACM International Conference on*, 2005, pp. 939–946.
- [7] S. Sarangi, B. Greskamp, R. Teodorescu, J. Nakano, A. Tiwari, J. Torrellas, Varius: a model of process variation and resulting timing errors for microarchitects, *IEEE Trans. Semicond. Manuf.* 21 (1) (2008) 3–13.
- [8] S. Abe, M. Hashimoto, T. Onoye, Clock skew evaluation considering manufacturing variability in mesh-style clock distribution, *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* E91-A (12) (2008) 3481–3487.
- [9] G. Venkataraman, Zhuo Feng, Jiang Hu, Peng Li, Combinatorial algorithms for fast clock mesh optimization, *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* 18 (1) (2010) 131–141.
- [10] I. Vaisband, E.G. Friedman, R. Ginosar, A. Kolodny, Low power clock network design, *J. Low Power Electron. Appl.* 1 (1) (2011) 219–246.
- [11] G.R. Wilke (Ph.D. thesis), *Analysis and Optimization of Mesh-based Clock Distribution Architectures*, vol. 100–101, Universidade Federal do Rio Grande do Sul, 2008.
- [12] Gustavo Wilke, Renan Fonseca, Cecilia Mezzomo, Ricardo Reis, A novel scheme to reduce short-circuit power in mesh-based clock architectures, in: *Proceedings of the 21st Annual Symposium on Integrated Circuits and System Design (SBCCI '08)*, 2008, pp. 117–122.