

Capacitance driven clock mesh synthesis to minimize skew and power dissipation

John Reuben^{a)}, Mohammed Zackriya.V, Salma Nashit,
and Harish M Kittur

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

a) johnreuben@vit.ac.in

Abstract: Tree driven mesh is gaining popularity as a viable method to distribute clock with minimum skew in Deep Sub Micron (DSM) technology. In the design of the leaf level mesh, the density of the mesh at various parts of the chip is a crucial factor which decides the clock skew and power dissipated in the mesh. We propose a capacitance driven mesh formation methodology which forms a minimum wire length, non-uniform mesh when compared to the traditional skew-driven mesh. After connecting the sinks to the mesh by a combination of Steiner tree and stubs, appropriately sized buffers are placed at optimal locations such that skew and power dissipation are minimized. When our algorithms were tested on ISPD2010 benchmarks, the power dissipated in the mesh was found to be 25% lesser and the skew was 32% to 45% lesser than the skew driven mesh.

Keywords: clock mesh, tree driven mesh, clock skew, process variations, short-circuit power

Classification: Integrated circuits

References

- [1] C.-C. Tsai, J.-O. Wu and T.-Y. Lee: IEICE Trans. Fundamentals **E91-A** (2008) 365.
- [2] A. Rajaram and D. Z. Pan: IEEE Trans. Computer-Aided Design Integr. Circuits Syst. **29** (2010) 1945.
- [3] G. Venkataraman, Z. Feng, J. Hu and P. Li: IEEE Trans. Very Large Scale Integr. (VLSI) Syst. **18** (2010) 131.
- [4] M. R. Guthaus, X. Hu, G. Wilke, G. Flach and R. Reis: ACM Trans. Des. Autom. Electron. Syst. **17** (2012) 33.
- [5] H. Qian, P. J. Restle, J. N. Kozhaya and C. L. Gunion: IEEE Trans. Computer-Aided Design Integr. Circuits Syst. **31** (2012) 868.
- [6] A. Abdelhadi, R. Ginosar, A. Kolodny and E. G. Friedman: Integration **46** (2013) 382.
- [7] C. N. Sze: ACM International Symposium on Physical Design (2010) 149.
- [8] L. Xiao, Z. Xiao, Z. Qian, Y. Jiang, T. Huang, H. Tian and E. F. Y. Young: ICCAD (2010) 458.
- [9] H. Veendrick: IEEE J. Solid-State Circuits **19** (1984) 468.
- [10] Predictive Technology Model: (2013) <http://ptm.asu.edu/>

- [11] H. Chen, C. Yeh, G. Wilke, S. M. Reddy, H. Nguyen, W. Walker and R. Murgai: ICCAD (2005) 939.
- [12] S. M. Reddy and R. Murugai: US Patent 7802215 B2 (2010).
- [13] NGSPICE: <http://ngspice.sourceforge.net> and MATLAB toolbox to analyse raw files: http://www.cpsim.com/download_hspice_tools.html
- [14] K. Suzuki, M. Yamashina, T. Nakayama, M. Izumikawa, M. Nomura, H. Igura, H. Heiuchi, J. Goto, T. Inoue, Y. Koseki, H. Abiko, K. Okabe, A. Ono, Y. Yano and H. Yamada: IEEE J. Solid-State Circuits **29** (1994) 1464.
- [15] S. R. Sarangi, B. Greskamp, R. Teodorescu, J. Nakano, A. Tiwari and J. Torrellas: IEEE Trans. Semicond. Manuf. **21** (2008) 3.
- [16] S. Abe, M. Hashimoto and T. Onoye: IEICE Trans. Fundamentals **E91-A** (2008) 3481.
- [17] X. Hu and M. Guthaus: DAC (2011) 516.

1 Introduction

Clock distribution by constructing a tree has always been in conflict with clock distribution by mesh. The proponents of Tree Based distribution used combinatorial algorithms to construct zero skew tree (or nearly zero) with minimum wire length and power consumption [1]. But they struggled against skew induced by process variations in DSM technology. The proponents of mesh based distribution leveraged the redundant paths in the mesh to propagate mesh based distribution which are highly resistant to skew induced by process variations. This is achieved at the cost of increased wire capacitance and hence higher power consumption [2]. The mesh was not preferred since meshes were difficult to analyse and lacked automation tools. But clock mesh synthesis has received much attention since 2010 [2, 3, 4]. Tree driven mesh is the hybrid clock distribution scheme in which tree is used at the top level and mesh is used in the leaf level to distribute clock (Fig. 1). In the recent past, Tree driven mesh has received considerable attention as a viable method to distribute clock signal in large industrial designs (a few thousand clock sinks) with minimum skew [5, 6]. Since issues in clock tree design are well studied, we focus on the design of leaf level mesh in this letter. We propose

1. A “capacitance-driven” mesh formation methodology as opposed to the “skew-driven” mesh formation methodology of [2, 3, 4].
2. A new method of connecting sinks to the mesh by a combination of Steiner tree and stubs as opposed to stubs alone [2, 3, 4].
3. A buffer library design considering short circuit current and a buffer placement methodology for the mesh formed.

We test our proposed algorithms on three ISPD2010 benchmarks which are based on real 45 nm microprocessor designs of INTEL ([7]).

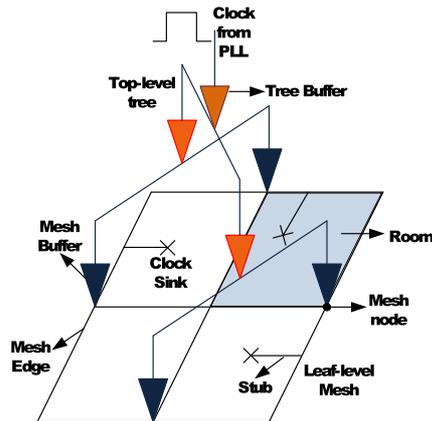


Fig. 1. Tree driven mesh: Notations used in this letter

2 Capacitance driven mesh formation

Mesh formation refers to the way in which the density of the mesh is decided—how coarse or fine the mesh should be. The general methodology which the previous works ([2, 3]) have followed is to start with a uniform mesh and increase the mesh density till the target skew is achieved. In [4], the mesh wire segments obtained by the method in [2, 3] are moved horizontally or vertically such that the stub lengths are minimized. This results in a non-uniform mesh. This methodology does not consider the density of clock sinks on the chip area while forming the initial mesh. Moreover, mesh density is increased uniformly to achieve skew requirements. In cases where the distribution of clock sinks is not uniform (ispd10cns06, 07, 08), this skew-driven mesh formation is not very efficient: more wire is used, resulting in consumption of more power. Hence, in most practical cases, the mesh obtained by [2, 3, 4] will result in an inefficient design with more power consumption.

In the proposed capacitance driven method, a non-uniform mesh is constructed based on density and capacitance of the sinks. The individual rectangles of the mesh are called rooms (Fig. 1). The mesh is repeatedly divided into four rooms till the sum of the capacitances of clock sinks in a room is less than a target capacitance (say 100 fF). Though a similar congestion driven method is used in [8], a complete framework including mesh buffer design and buffer placement has not been reported. The capacitance driven mesh is highly non uniform with some big rooms and very small rooms (Fig. 2). After the mesh is formed, we considered the possibility of connecting the sinks inside a room by a rectilinear Steiner tree since it can reduce the wire length for larger rooms. Though the sink capacitance gets accumulated in a Steiner tree, the clock will not slew since the capacitance in a room is kept below a particular threshold by the mesh formation algorithm. A trade-off was calculated between direct connections of clock sinks to mesh (stub) vs. rectilinear Steiner tree formation. As a result, all clock-sinks lying close to room edges (within 10% of length and width of each room) are connected directly using stubs. The rest of the sinks are connected by a rectilinear Steiner tree, which is then connected to the mesh.

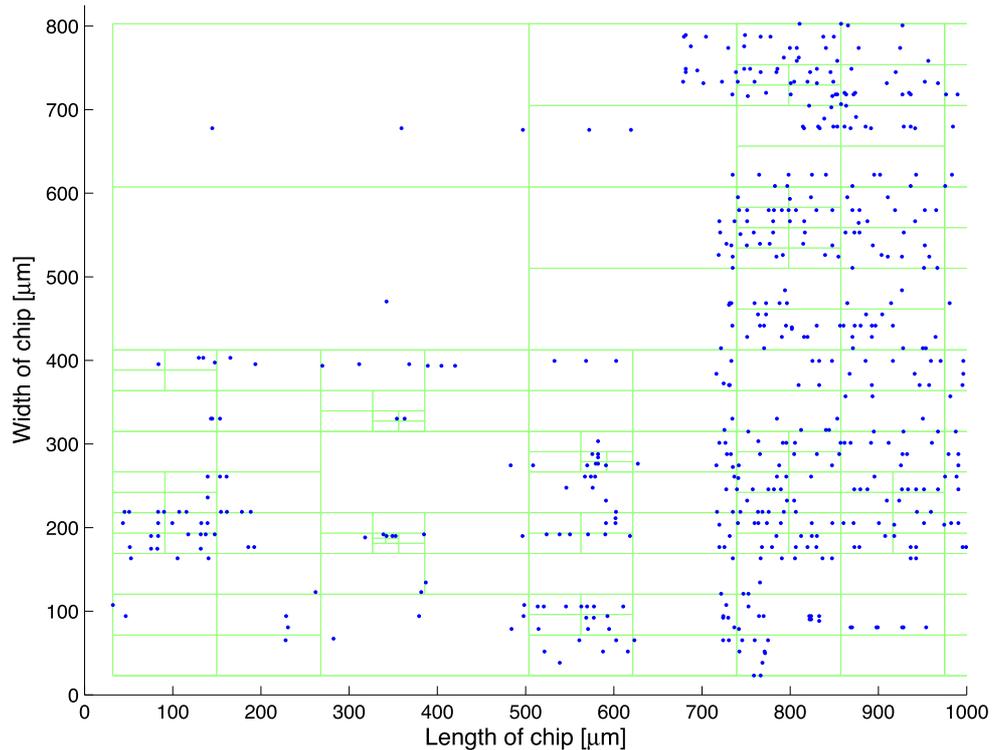


Fig. 2. Capacitance driven mesh for left half of ispd10cns06, “.” represents clock sink

It is crucial that the Steiner tree be connected to the mesh at a point such that clock signal reaches all the sinks in a room with minimum skew. The center of mass, defined in [6] is the average location of all the clock sinks, weighted by the sink capacitances. We call this the Capacitance Centroid (CC) which is a more appropriate term for the center of mass used in [6]. The CC of all vertices (v) of a Steiner tree can be expressed as:

$$(x_{cc}, y_{cc}) = \left(\frac{\sum(c(v) \times x(v))}{\sum c(v)}, \frac{\sum(c(v) \times y(v))}{\sum c(v)} \right) \quad (1)$$

where $x(v)$, $y(v)$ represent the co-ordinates of sinks represented by v with capacitance $c(v)$.

The clock is fed to the steiner tree through its CC and hence the delays to all sinks are equalized. The CC need not always lie on the Steiner tree. If it doesn't, the CC is interpolated to the nearest tree branch, which is then connected to the nearest mesh edge. The entire algorithm is presented in Fig. 3. The mesh formed for ispd10cns06 for 100 fF target capacitance is shown in Fig. 4. To compare with skew driven mesh of [2, 3], we formed an initial uniform mesh of 20×20 and divided it repeatedly to achieve a target skew of 10 ps across the mesh. (The wire length data of [2, 3] are not available for ISPD2010 benchmarks). The sinks were connected to the mesh by stubs and skew was calculated using Elmore delay model. We compared the total wire-length (mesh edges + stub + trees) of our mesh with the total wire length (mesh edges + stub) of skew driven mesh in Table I.

Key Observation: The individual sink capacitance varies from as low as

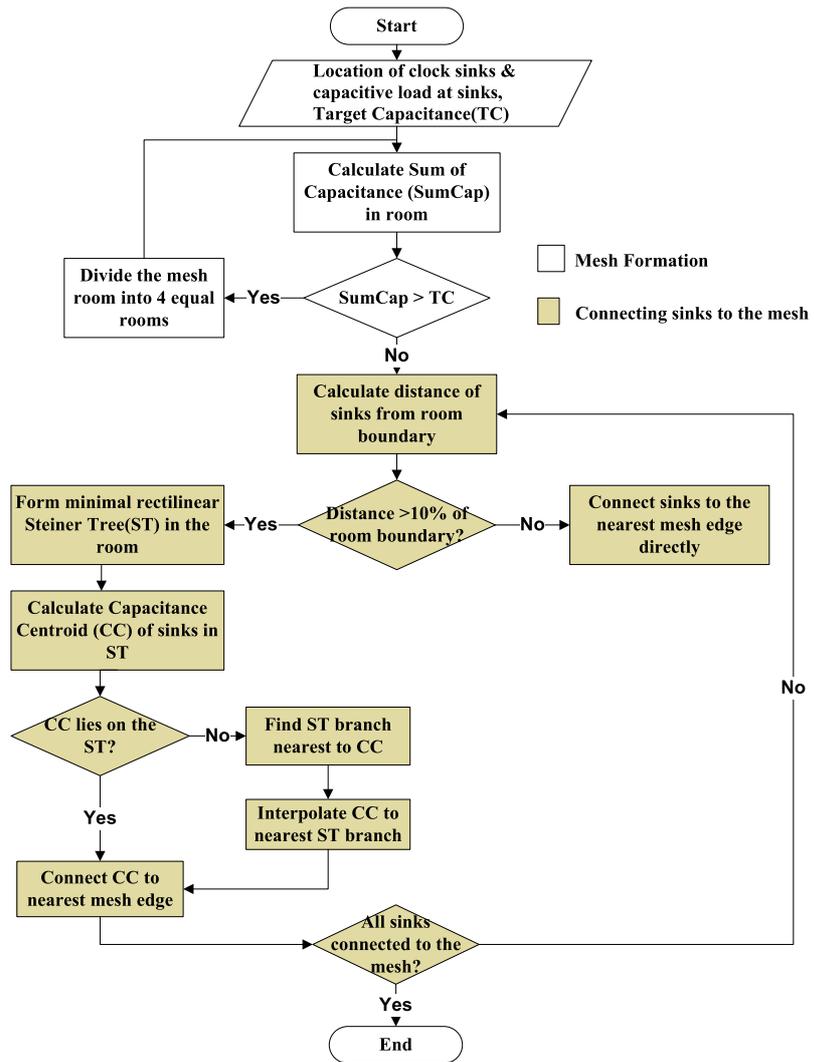


Fig. 3. Algorithm for Capacitance driven mesh synthesis

Table I. Comparison of total Wire Length (WL): Skew driven vs. Capacitance driven mesh

Target skew for Skew driven mesh = 10 ps Target capacitance for Capacitance driven mesh = 100 fF				
Benchmark circuit	Size of chip [mm ²]	Total WL of skew-driven mesh [mm]	Total WL of Cap. driven mesh [mm]	% reduction in WL
ispd10cns06	1.7	59.7	42.8	28.3%
ispd10cns07	3.7	83.7	71	15.17%
ispd10cns08	3.7	72.8	47.6	34.6%

4.25 fF to 50 fF on an average in ISPD2010 benchmarks. Hence, if there are large number of sinks of smaller capacitance in a region, the mesh will not be made denser because of the large number of sinks in that region. The density of sinks and their individual capacitance are both taken into account resulting in a mesh of minimum wire length.

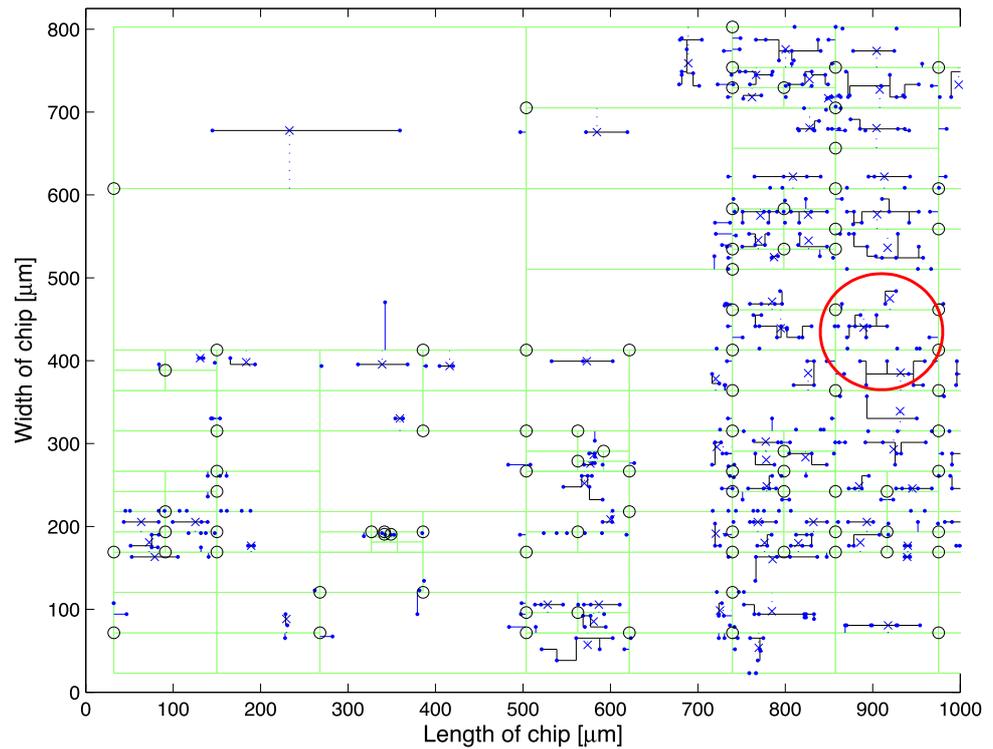


Fig. 4. Mesh of Fig. 2 with sinks connected to mesh: “X” represents CC, “o” are buffers

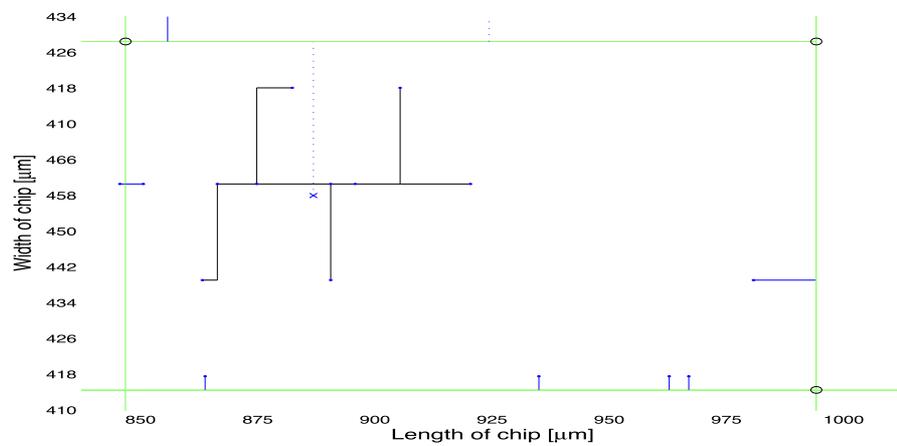


Fig. 5. The circle in Fig. 4 zoomed

3 Buffer library design considering short circuit current

The previous works ([2, 3]) considered only the capacitive load to design the mesh buffers. Power dissipation due to short circuit current in inverter must also be considered while designing the buffers in addition to being able to drive the load with a particular slew constraint [4]. To minimize short circuit current, the slope of the rise/fall of the clock signal must be same at the input and output of a circuit [9]. We define rise time as the time required by the clock to rise from 0 to 100 percent of V_{DD} . Since buffers are formed by cascading inverters, we designed the inverter stages such that slope of rise/fall of the clock signal is same at the input and output (Fig. 6). This

makes the ratio of second inverter size to first inverter size large. This design will help to minimize the short-circuit power dissipation due to inter-buffer skew (Section 5.2). The clock signal has a slew of 100 ps (worst case) for 1 GHz clock at the input to the mesh buffer. 45 nm Predictive Technology Model files ([10]) were used for the NMOS and PMOS transistors.

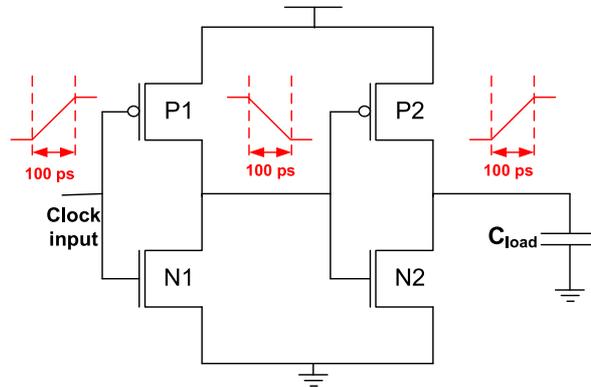


Fig. 6. Buffer library design

Even if the input slew to mesh buffer is better (less than 100 ps), this buffer design will minimize the short circuit power dissipation. Based on the capacitive load in the rooms of the mesh for ISPD2010 benchmarks, the buffer library in Table II is formed. Though the capacitance is evenly distributed as 100 fF/room, big buffers like B6-B8 are seldom needed in rare congested area where the buffer at a mesh node is required to drive the sinks in three or four rooms adjoining it.

Table II. Buffer library: Sizes of buffers to drive the corresponding load with 100 ps slew

Name	C _{load} [fF]	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
B1	0-50	44/45	32/45	1.375	840/45	575/45	1.460
B2	51-100	82/45	57/45	1.438	1656/45	1177/45	1.407
B3	101-150	114/45	82/45	1.390	2451/45	1718/45	1.426
B4	151-200	151/45	110/45	1.372	3258/45	2281/45	1.428
B5	201-250	189/45	130/45	1.454	4073/45	2852/45	1.428
B6	251-300	227/45	156/45	1.455	4887/45	3422/45	1.428
B7	301-450	340/45	233/45	1.459	7330/45	5118/45	1.432
B8	451-600	450/45	310/45	1.452	9700/45	6800/45	1.426

4 Buffer placement algorithm

The mesh buffers serve the dual purpose of reconstructing the clock signals degraded by top level tree and isolating the local mesh capacitance from upstream load capacitance seen by tree buffers. For every room which has a Steiner tree, a buffer is placed at the mesh node closest to the Capacitance

Centroid (CC) of the tree. Similarly, a buffer is placed for the sinks connected by stub. If two buffers are needed at the same location, their load capacitances are added. Finally, an appropriately sized buffer is placed at those locations according to Table II. The flowchart in Fig. 7 depicts this algorithm.

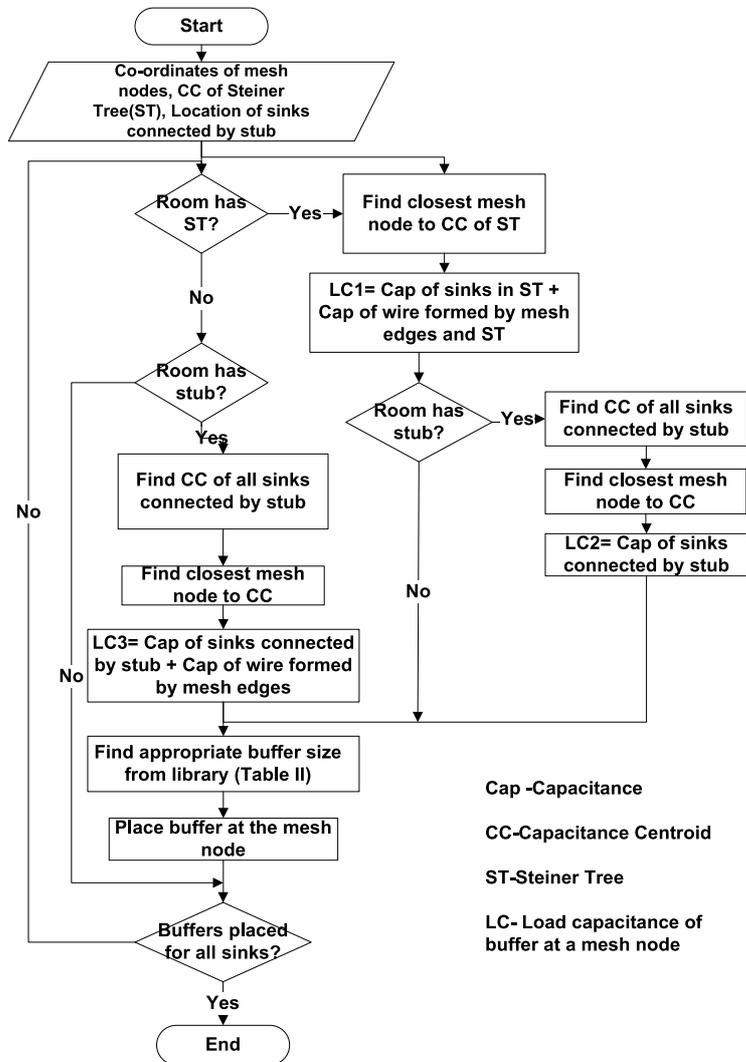


Fig. 7. Algorithm for Buffer Placement

5 Mesh simulation by Sliding Window Scheme (SWS)

The simulation of the clock mesh to analyze the skew and power dissipation is an onerous task. In fact, the difficulty in simulating the clock mesh is one of the prominent reasons for its unpopularity when compared to the tree based clock distribution. The sliding window scheme ([11, 12]) is a method of simulating a big mesh by splitting the simulation task into smaller meshes called 'windows'. The wires connecting the sinks to the mesh are modelled accurately inside the window and approximately outside the window (Fig. 8). The premise of this technique is that the attenuation of the clock signal

fed at a node in a RC mesh increases exponentially with distance from the node. Hence nodes which are far apart have less electrical impact on each other. From numerous simulations (on meshes of different sizes, with different window sizes), the authors in [11, 12] prove that the delay calculated by the SWS at the clock sinks is always within 1 percent of the clock delay calculated by a complete simulation of the whole mesh modelled accurately. Since each segment of tree and stub has to be π -modelled (Fig. 8), the SPICE file becomes so large and memory consuming that an accurate modelling of the whole mesh is not practical for large benchmarks. So we adopt the SWS to simulate our mesh.

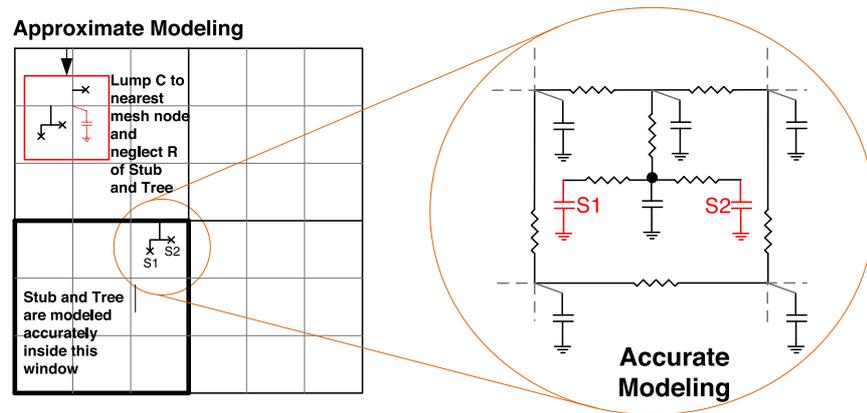


Fig. 8. SWS and mesh modelling inside and outside the window

5.1 Experimental procedure

The capacitance driven mesh formation algorithm was implemented using MATLAB and the mesh formed was simulated using NGSPICE ([13]). The capacitance driven mesh formed along with the information of mesh buffers (location and size) is used to form the SPICE (.cir) file. The mesh edges are modelled using single π model both inside and outside the window. The stub and tree are also single π -modelled (accurate modelling) inside the window (Fig. 8). Outside the window, the sink capacitances are lumped to the nearest mesh nodes and the resistance of the stub and tree are ignored (approximate modelling). We split the entire benchmark area into 4 uniform windows and a SPICE file was written for each window. These files are then executed sequentially in NGSPICE to calculate the clock latency at sinks inside the window. The raw data file thus generated is taken to MATLAB again and the clock latency at each sink location in the mesh is extracted using HSPICE toolbox for MATLAB ([13]).

To arrive at the optimum target capacitance (capacitance/room), we did a study on the ispd10cns06 by varying the target capacitance as shown in Table III. As the capacitance/room increases, the mesh becomes more and more coarse. This results in less wire being used, but the skew increases due to longer delay inside the room. We calculated total power by integrating

the current drawn from V_{DD} in one cycle which includes dynamic and short circuit power. The total power dissipation decreases due to fewer buffers. Since short circuit power due to inter-buffer skew also has to be considered, we decided to choose 100 fF because the product of skew and number of buffers is minimum for it. Beyond 100 fF, the number of buffers continues to decrease, but skew increases drastically nullifying the main motivation behind mesh based clock distribution. Comparing Table I and Table III, we note that capacitance driven mesh achieves the same skew target of 10 ps, but with 28% wire length reduction.

Table III. Arriving at optimum Target Capacitance (TC)

TC	Skew across the mesh	Number of buffers	Capacitance range of buffers	Wire length translated as capacitance	Total Power dissipation
50 fF	8.03 ps	259	14-430 fF	8.94 pF	38.25 mW
75 fF	8.12 ps	208	32-430 fF	8.26 pF	36.41 mW
100 fF	8.87 ps	156	40-590 fF	7.43 pF	34.43 mW
125 fF	18.58 ps	138	43-592 fF	7.06 pF	33.51 mW

5.2 Short Circuit (SC) power dissipation due to inter-buffer skew

Though the top level tree feeding the mesh may be designed for zero skew by efficient clock tree synthesis algorithms (like Zero skew tree, DME, LTM of [1]), there will be skew induced by process variations. Hence the clock will not reach the input of all the mesh buffers at the same time, resulting in clock skew between the mesh buffers (inter-buffer skew). Inter-buffer skew will result in short circuit path as illustrated in Fig. 9. This SC current will result in additional short circuit power dissipation in addition to the dynamic and short circuit power already dissipated in the buffers during each transition. To verify this, we constructed a 2×2 test mesh of size $120 \mu\text{m}$ by $60 \mu\text{m}$ having 2 sinks of 30 fF capacitance. This mesh was fed by two buffers appropriately sized to drive the surrounding capacitance. We observed the total current drawn from V_{DD} when this mesh was fed by a 1 GHz clock signal for the following two cases (Fig. 9):

1. The clock signal is fed to two buffers at the same time.
2. The clock signal is fed with 50 ps skew between the two buffers.

Hence the power dissipation calculated in Table III is an under-estimate of the actual power consumption in the mesh. When we do Monte Carlo simulations (Section 6) we will incorporate the inter-buffer skew which will give us an accurate estimate of the total power dissipation in the mesh. In the design of top-level tree, stage connect clock tree approach of [14] can be used to reduce the skew which will result in lesser inter-buffer skew at the mesh.

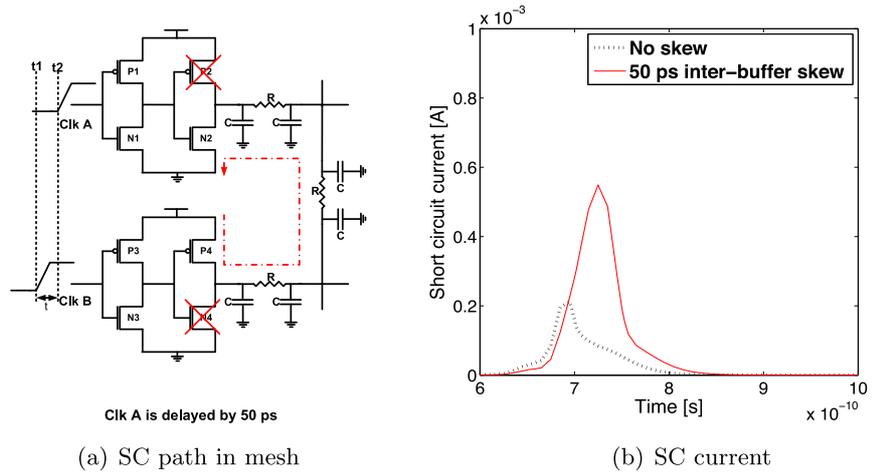


Fig. 9. Short-Circuit (SC) power dissipation due to inter-buffer skew

6 Monte Carlo simulations across variations

6.1 Experimental parameters

Our basic parameters are taken from ISPD2010 benchmarks to enable comparison with other works (Table IV). Mesh buffers were simulated using Predictive Technology Model (45 nm PTM HP from [10]).

Table IV. Simulation conditions

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

Threshold voltage V_{th} and channel length L_{eff} are the two key device parameters that are subject to variation in DSM technology ([15]). So we consider only these two variations to incorporate the process variations. Then we consider V_{DD} variation and system variation. We did not consider variations in interconnects since transistor variations are more dominant compared to interconnect variations [16]. We performed 200 Monte Carlo simulations with the variations considered in Table V. One complete simulation of ispd10cns06 (mesh generation in MATLAB + four windows simulation in ngspice sequentially) took 81 seconds on a 2.4 GHz INTEL i3 processor with 4 GB RAM. This time will reduce further if the windows are simulated simultaneously in parallel processors which is yet another significant advantage of sliding window scheme. For ispd10cns07 (largest benchmark with 1915 sinks), the windows were simulated in parallel on different computers to reduce simulation time. The skew and power dissipation of [2, 3] are not available for ISPD2010 benchmarks. So we compare our work with the results in [17] where a skew driven mesh is formed according to [2] for ISPD2010 benchmarks (Table VI).

Table V. Variations considered in MC simulations

	Parameter	Distribution	Variation
Process	Channel length	Gaussian	3σ with mean 45 nm
	V_{th}	Gaussian	3σ with mean 0.404 V for nmos and -0.384 V for pmos
Voltage	V_{DD}	Gaussian	3σ with mean 1 V
System	Inter-buffer skew	Uniform	0-50 ps

Table VI. Performance of our work considering variations

Benchmark	This work		Skew-driven [17]*		% reduction to [17]	
	Avg. skew (ps)	Avg. power (mW)	Skew (ps)	Power (mW)	Skew	Power
ispd10cns06	11.47	35.116	17	47	32.52%	25.29%
ispd10cns07	14.88	56.28	26	72.8	42.7%	22.7%
ispd10cns08	13.04	40.71	24	54.5	45.6%	25.3%

*Denoted as Meshworks grid in [17]

Though we generated the mesh for other benchmarks, we couldn't simulate them because our sliding window scheme splits the entire circuit into four equal windows. Benchmarks 1-5 of ISPD2010 have blockages and the sliding window scheme needs increased automation to handle benchmarks with blockages.

7 Conclusion

Capacitance driven mesh formation will result in lesser wire (and hence lesser power) compared to skew driven mesh. The mesh density is adjusted automatically according to the number of sinks and capacitance of the sinks in the region. Hence capacitance driven method synthesizes the most optimum mesh in cases of unevenly distributed capacitances of real industrial designs. Since capacitance is uniformly distributed in the rooms of the mesh, the capacitance each buffer has to drive will be of the same order enabling efficient buffer sizing and buffer placement. As noted in [2], the skew introduced by an inappropriately sized buffer is one of the dominant reasons for skew in a mesh. We simulated our mesh across variations and verified that apart from wire length reduction, the clock skew reduces upto 45% and the power dissipation reduces upto 25% compared to the traditional skew-driven mesh of [2].

Acknowledgments

We would like to thank Holger Vogt of NGSPICE team for helping with Monte Carlo script in NGSPICE.