Timing Slack Aware Incremental Register Placement with Non-uniform Grid Generation for Clock Mesh Synthesis

Speaker: Jianchao Lu

Jianchao Lu, Xiaomi Mao, Baris Taskin

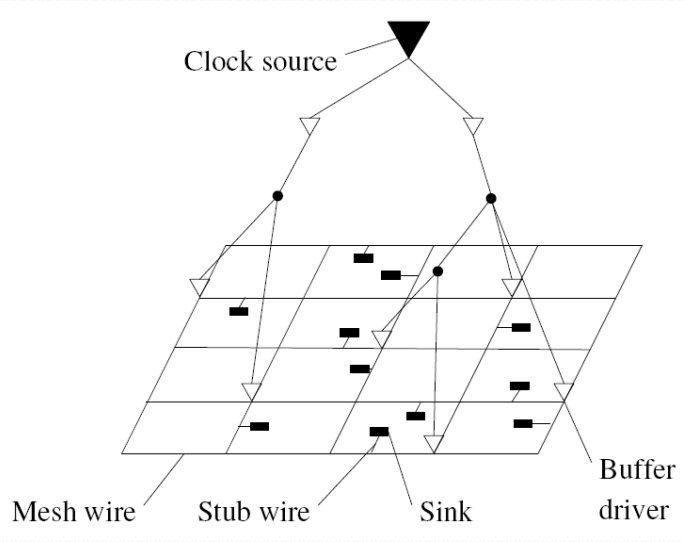
VLSI Lab

Electrical & Computer Engineering

Drexel University

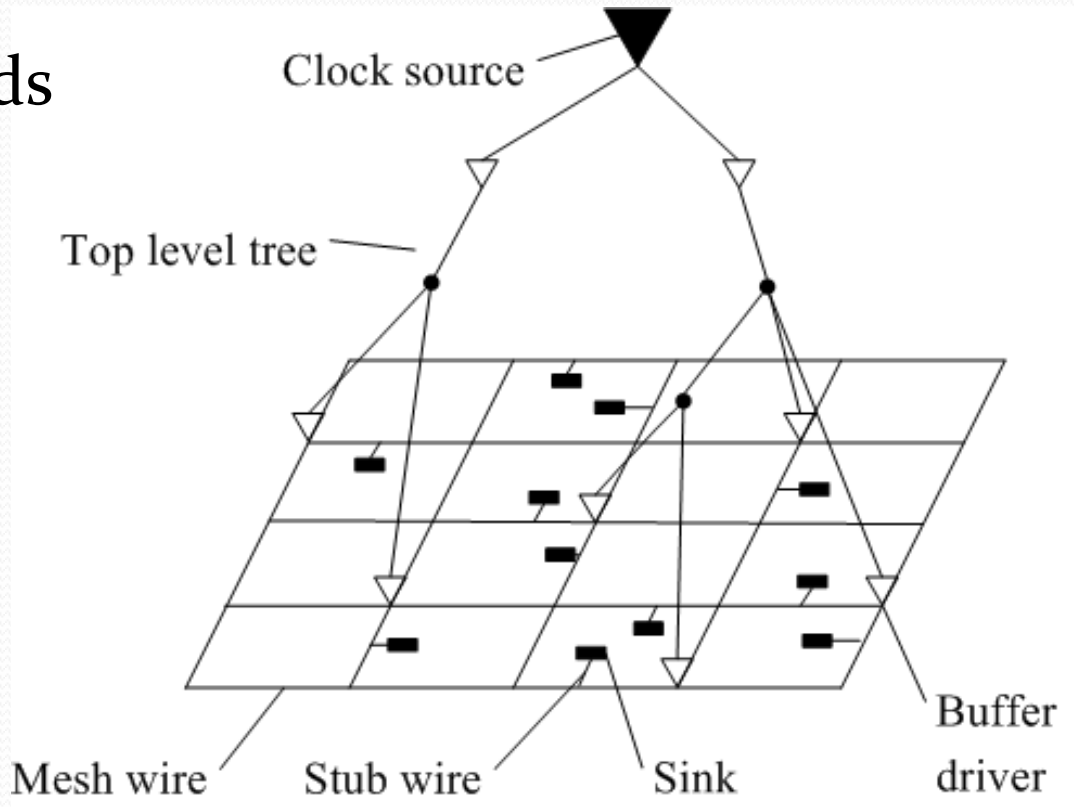
Outline

- Preliminaries
- Previous Works
- Methodology
- Experimental Results
- Conclusions



Clock Mesh Network

- Consists of top level clock tree, mesh grids and stub wires.



Power Dissipation on Clock Network

- Clock network is a global network of interconnect wires and buffers.
- Clock signal switching introduces a lot of dynamic power dissipation.
- Consumes more than 40% of the total power.

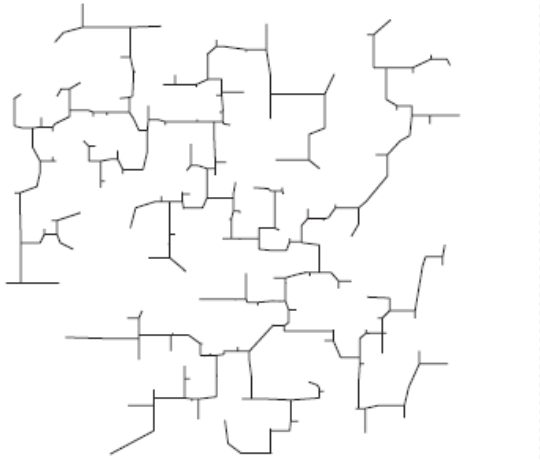
$$P = \alpha C V^2 f_{clk}$$

Switching factor

Capacitance

VDD

Frequency



Power Dissipation on Clock Network

- Clock network is a global network of interconnect wires and buffers.
- Clock signal switching introduces a lot of dynamic power dissipation.
- Consumes more than 40% of the total power.

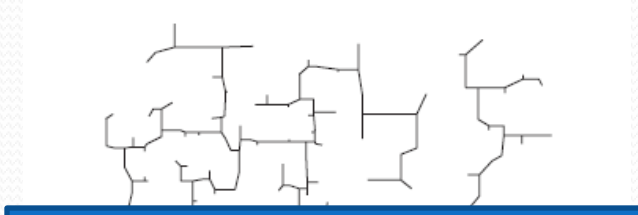
$$P = \alpha C V^2 f_{clk}$$

Switching factor

Capacitance

VDD

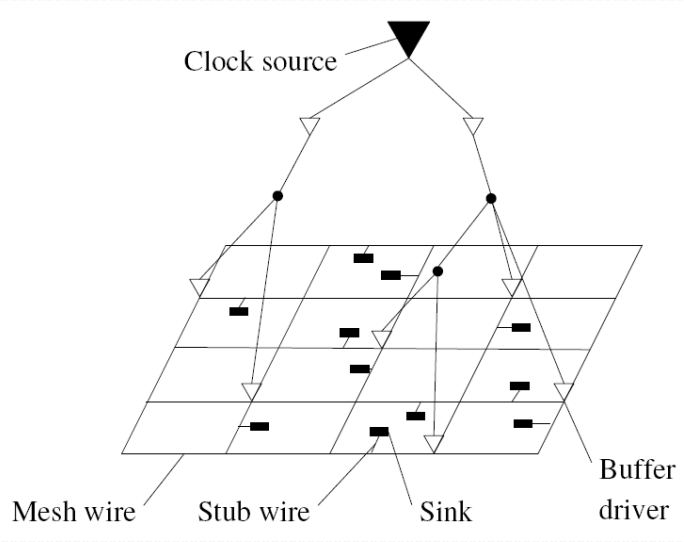
Frequen



Switching capacitance = α*C_total = (C_grid + C_stub + C_tree)

Outline

- Preliminaries
- Previous Works
- Methodology
- Experimental Results
- Conclusions

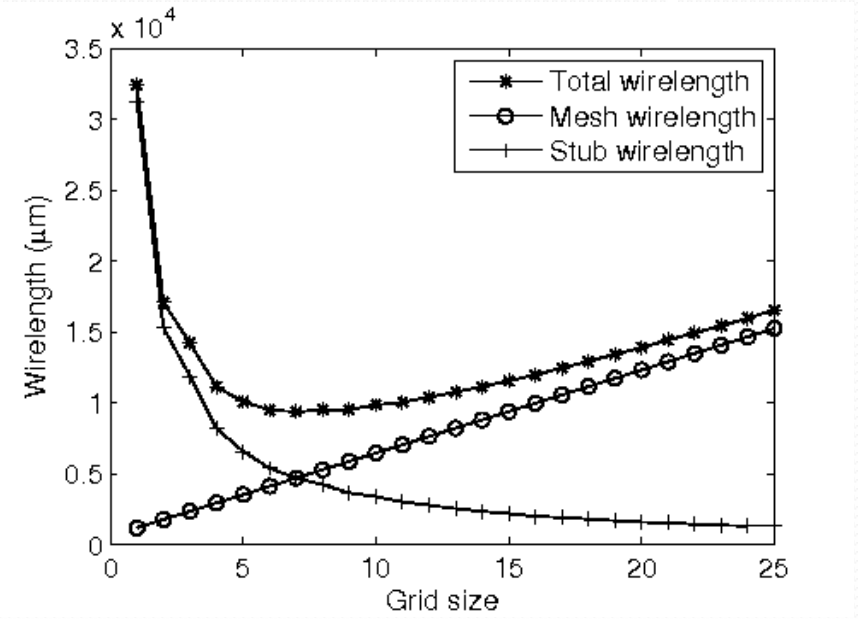


Most Relevant Previous Works

- [1] A. Rajaram and D. Pan, Meshworks: An efficient framework for planning, synthesis and optimization of clock mesh networks. In *Asia and South Pacific Design Automation Conference (ASPDAC), Jan. 2008.*
- [2] M. R. Guthaus, G. Wilke, and R. Reis, Non-uniform clock mesh optimization with linear programming buffer insertion. In *Proceedings of the ACM/IEEE Design Automation Conference (DAC), June 2010.*
- [3] Minsik Cho, David Z. Pan and Ruchir Puri, Novel Binary Linear Programming for High Performance Clock Mesh Synthesis, In *Proceedings of IEEE/ACM Int'l Conference on Computer-Aided Design (ICCAD), San Jose, CA, November 2010.*

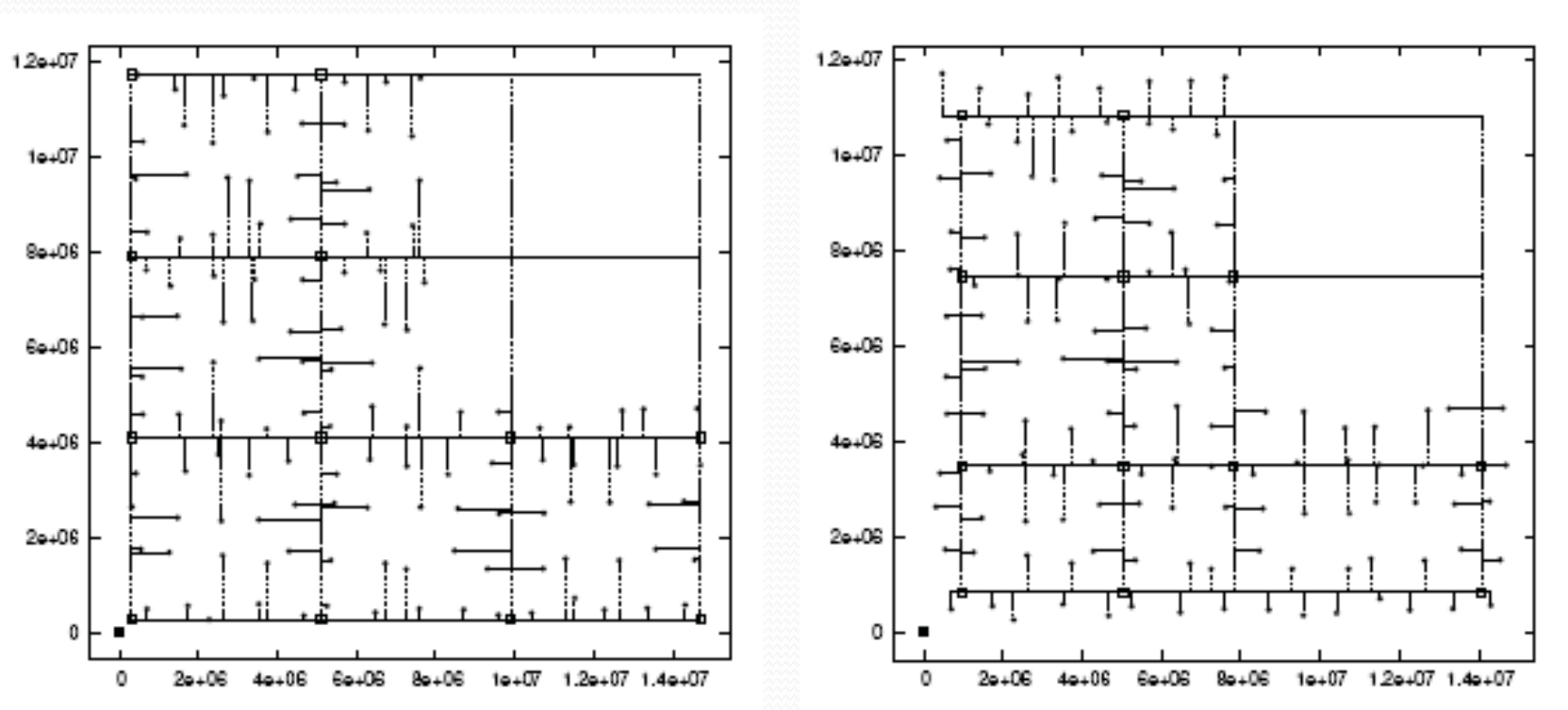
Meshworks [1]

- Identifies relationship between grid size and total mesh wire.
- Optimal grid size based on skew.
- Mesh reduction.
- Modified buffer driver insertion.



[1] A. Rajaram and D. Pan. Meshworks: An efficient framework for planning, synthesis and optimization of clock mesh networks. In *Asia and South Pacific Design Automation Conference (ASPDAC), pages 250–257, Jan. 2008.*

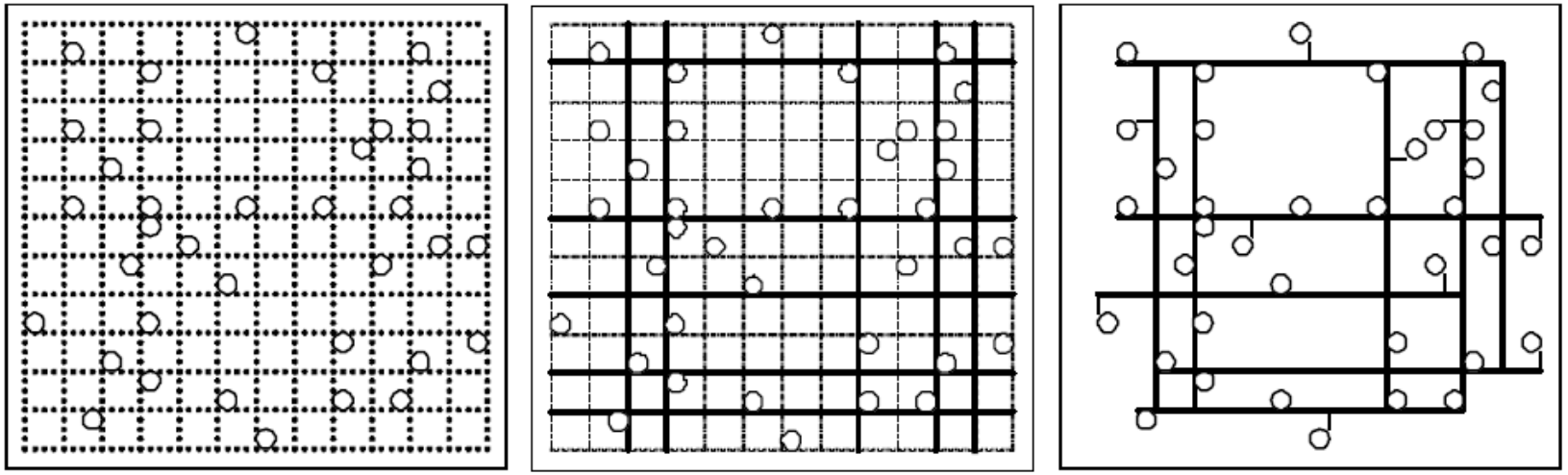
Non-uniform Mesh [2]



[2] M. R. Guthaus, G. Wilke, and R. Reis. Non-uniform clock mesh optimization with linear programming buffer insertion. In *Proceedings of the ACM/IEEE Design Automation Conference (DAC), pages 74–79, June 2010.*

ILP Based Mesh Synthesis [3]

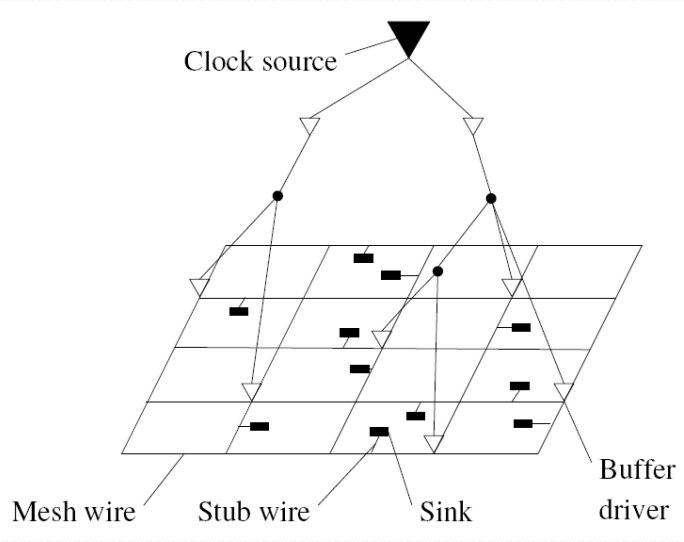
- Mesh generation and sink assignment algorithms.



[3] Minsik Cho, David Z. Pan and Ruchir Puri, Novel Binary Linear Programming for High Performance Clock Mesh Synthesis, In *Proceedings of IEEE/ACM Int'l Conference on Computer-Aided Design (ICCAD), Page 438—443, November 2010.*

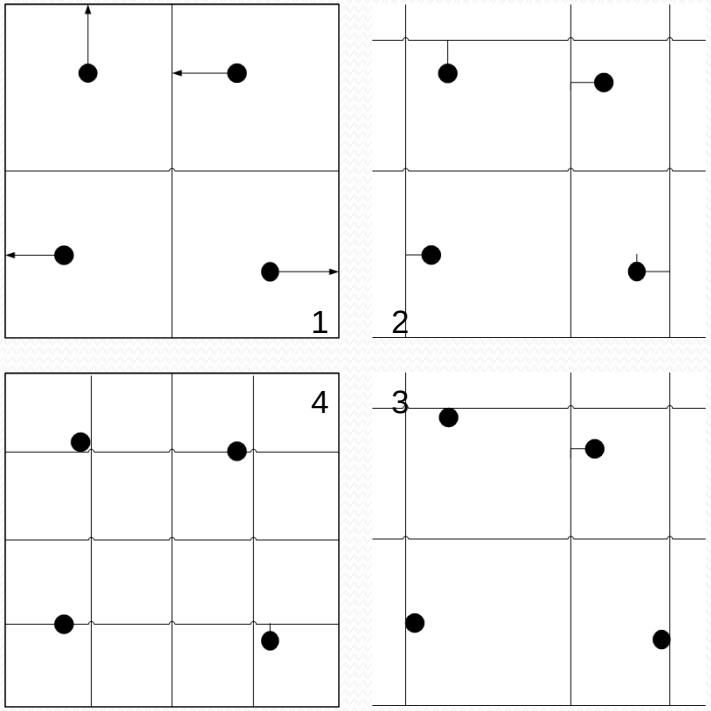
Outline

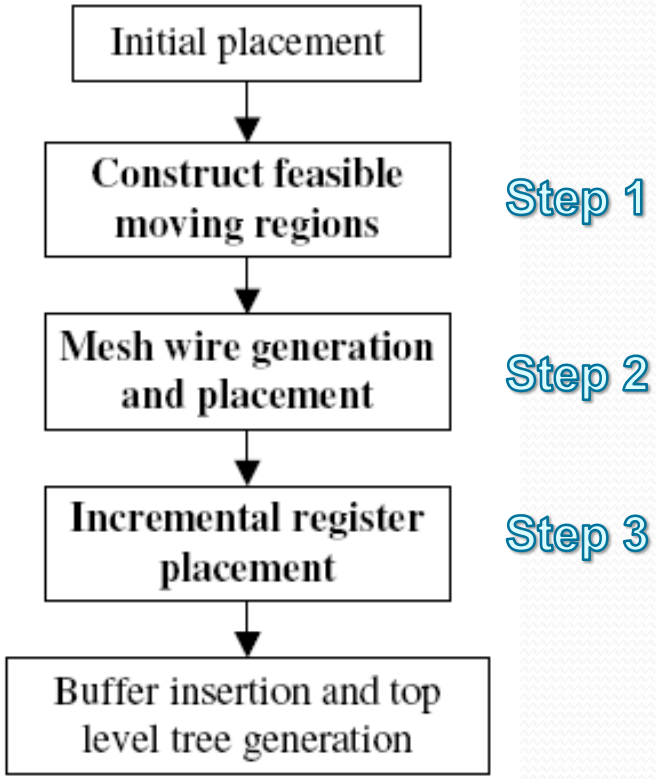
- Preliminaries
- Previous Works
- Methodology
- Experimental Results
- Conclusions



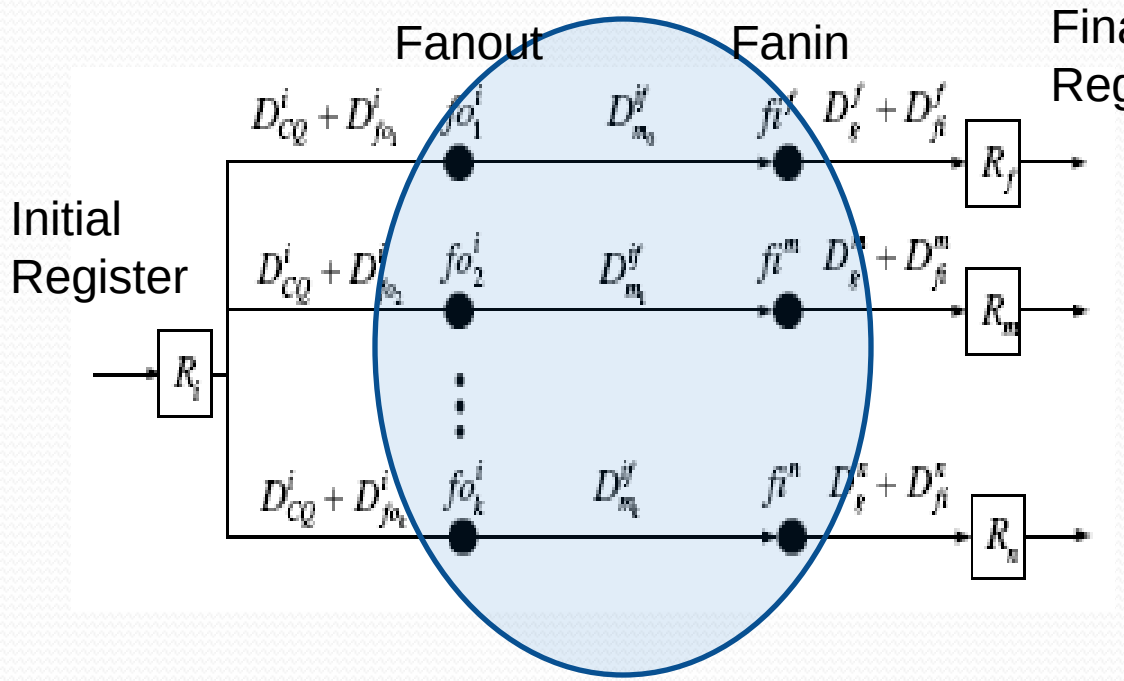
Proposed Method

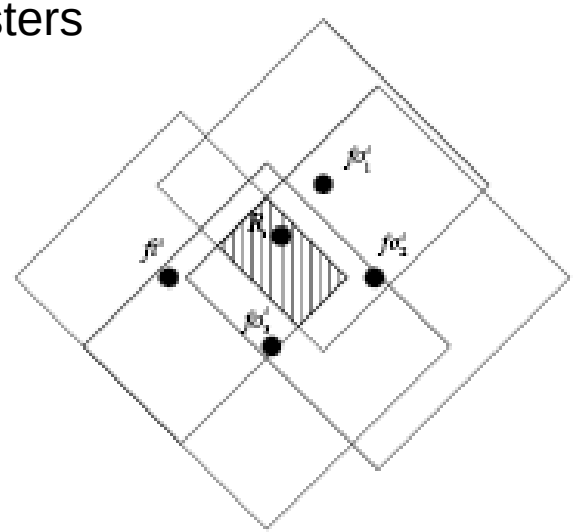
- Optimizing the placement during the clock mesh synthesis.



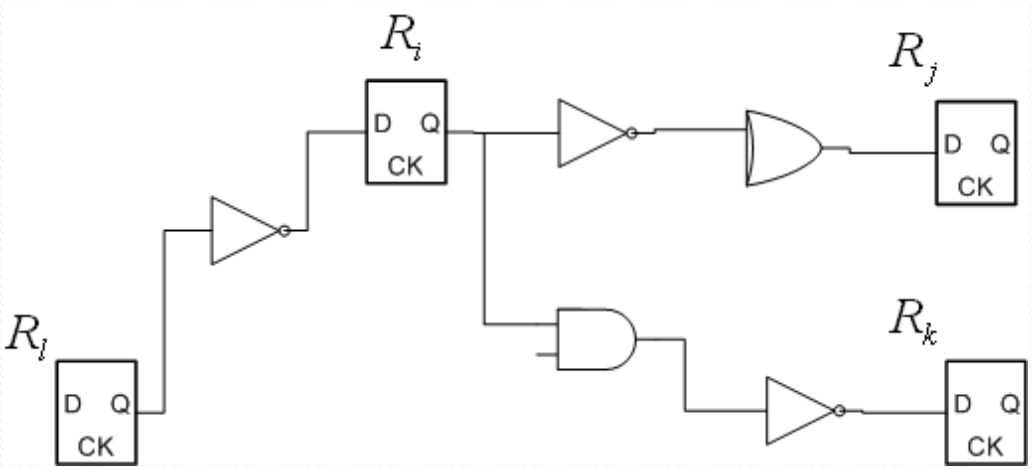


Step 1: Creating Feasible Moving Region of Each Register

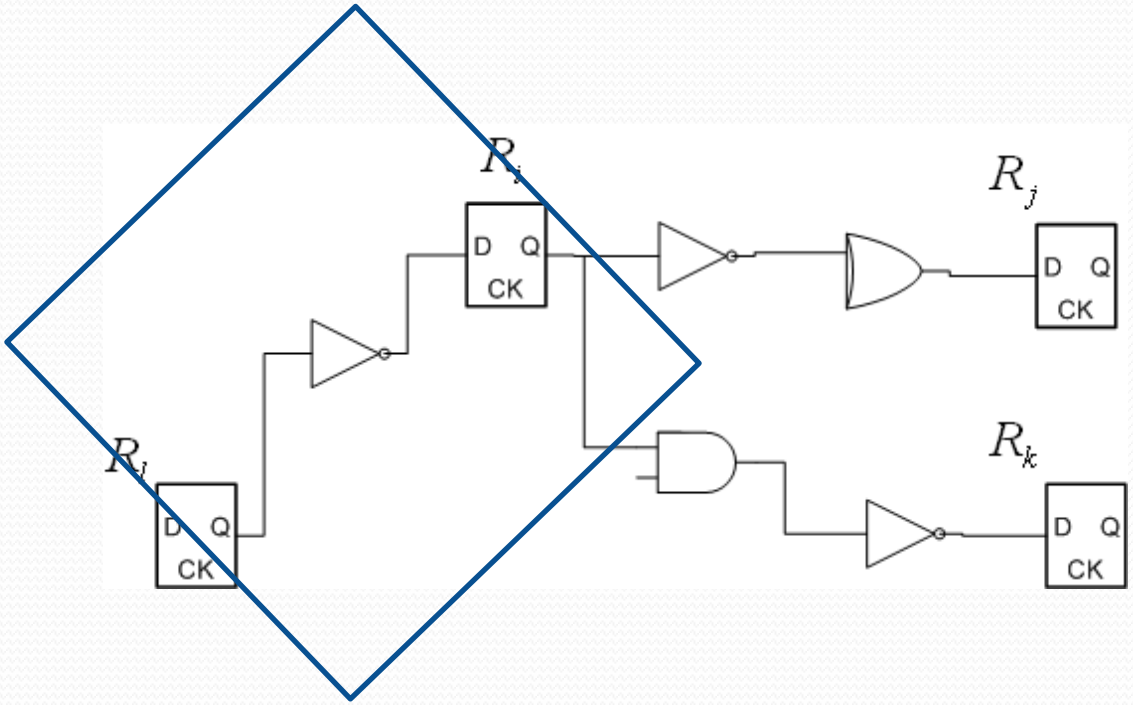




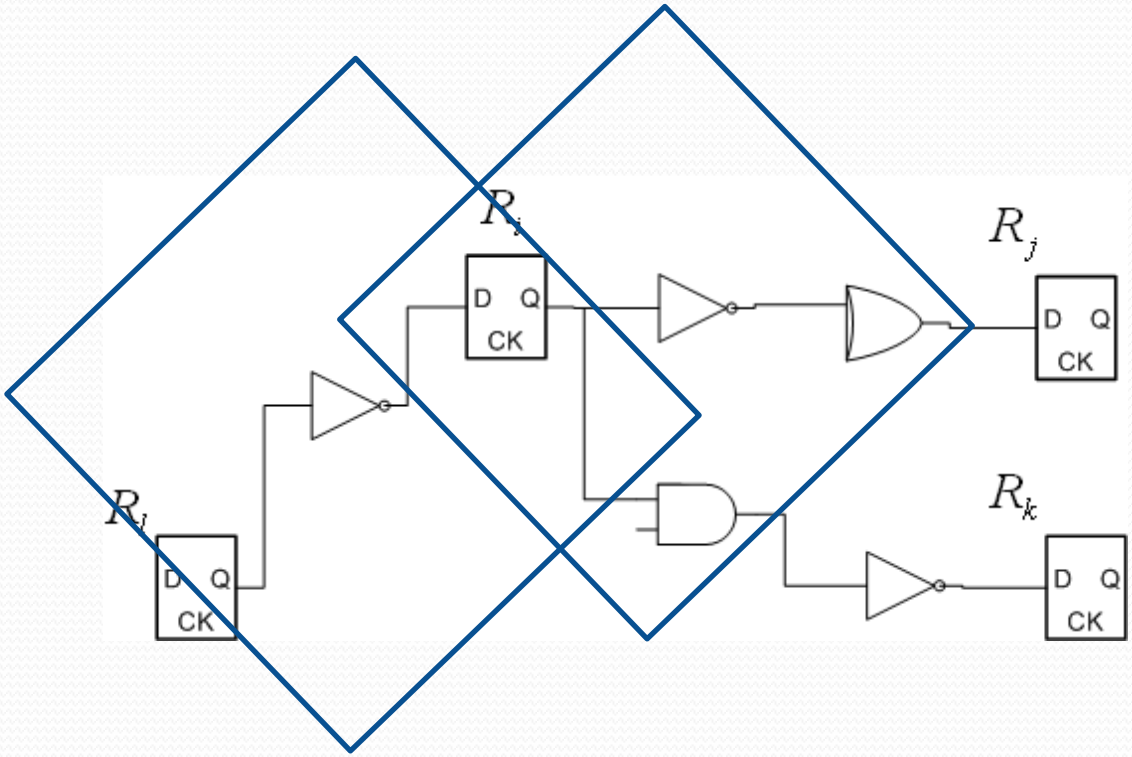
Creating Feasible Moving Regions



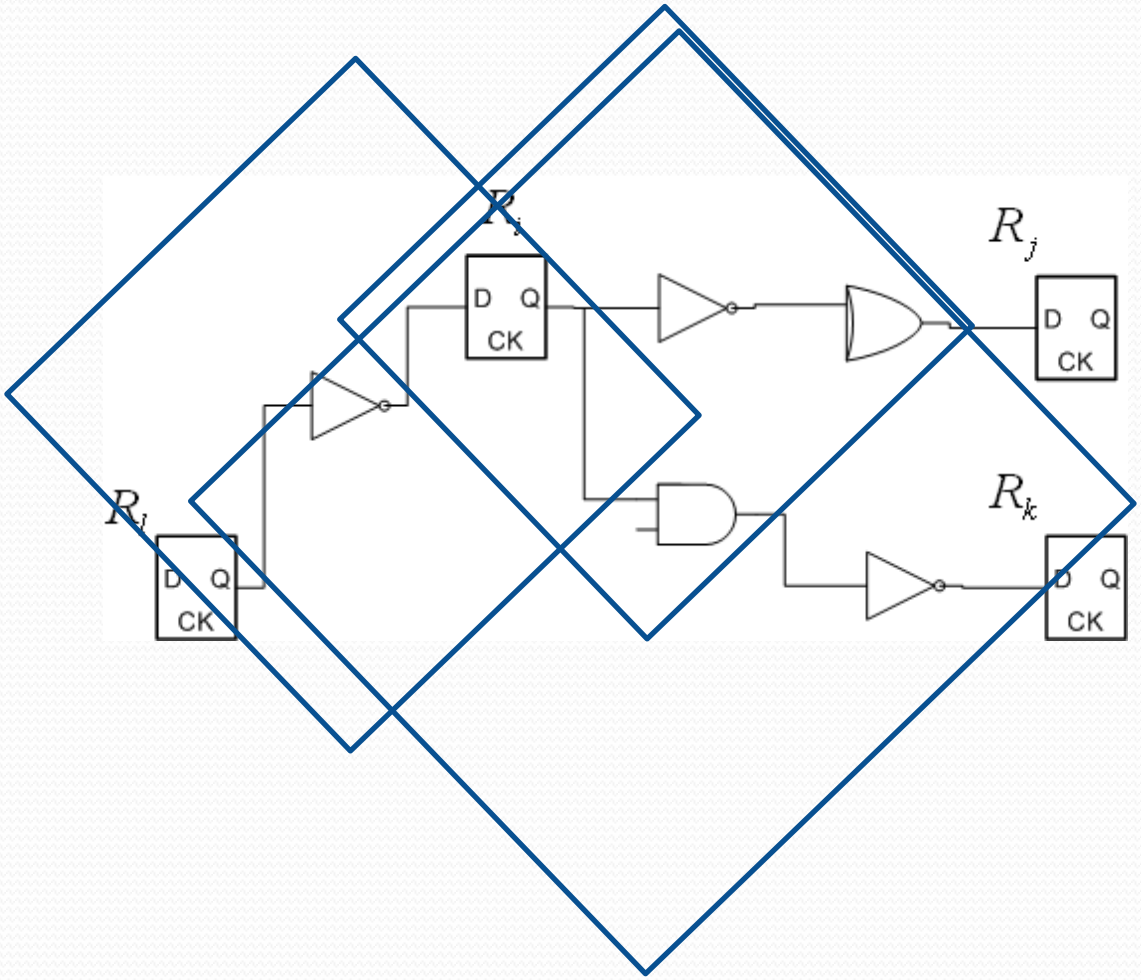
Creating Feasible Moving Regions



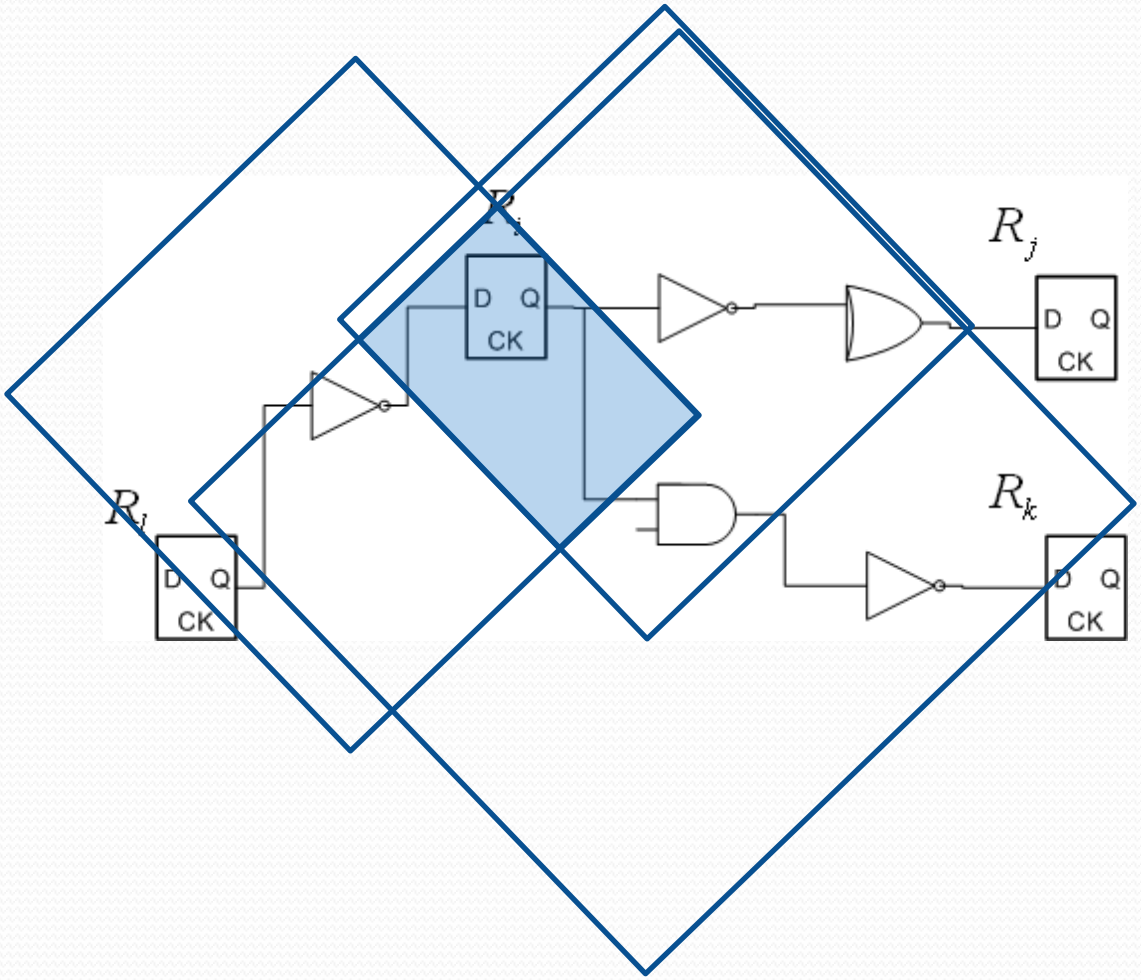
Creating Feasible Moving Regions



Creating Feasible Moving Regions

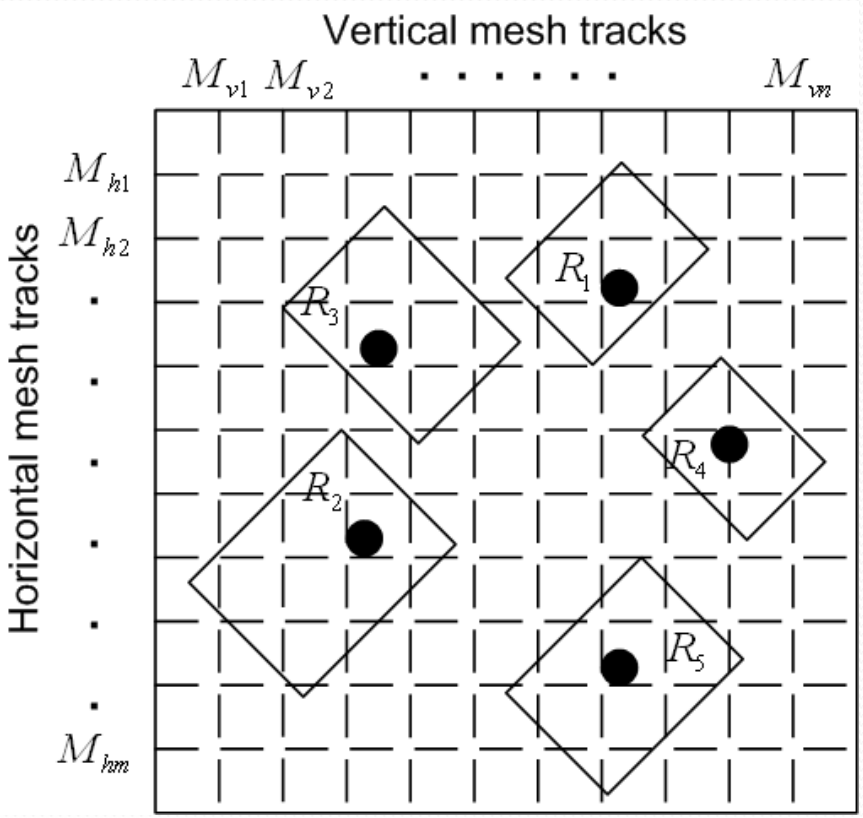


Creating Feasible Moving Regions



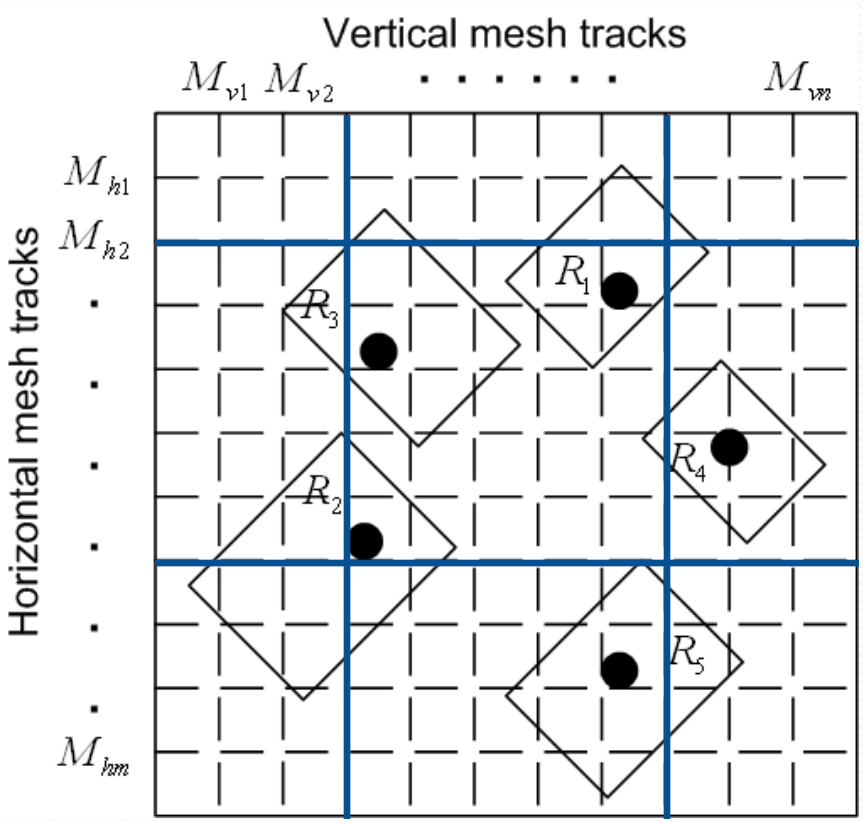
Step 2: Mesh Generations

- Registers can be moved in feasible moving regions without negative timing slack.
- Choose the minimum amount of mesh tracks that all the registers can be moved on as the mesh network.



Step 2: Mesh Generations

- Registers can be moved in feasible moving regions without negative timing slack.
- Choose the minimum amount of mesh tracks that all the registers can be moved on as the mesh network.



Mesh Generation Problem

- Problem: Assume each mesh track is a set and each register is an element. Finding the minimum amount of sets that includes all the elements is equivalent to finding the minimum amount of mesh tracks that can connect to the mesh wires.
- Greedy algorithm: Greedily add the candidate mesh track with the minimum cost.
- Cost of each grid wire = total distance of the registers from the grid/number of new elements added in the solution set.

Step 3: Incremental Register Placement

Objective: minimizing total stub wire.

Subject to:

- The timing constraints.
- The registers should be non-overlapped.

Variables:

- Registers locations.

Minimize the total stub wirelength.

$$\min \quad \sum_{\forall R_i} w^i_{stub}$$

Objective

Timing constraints

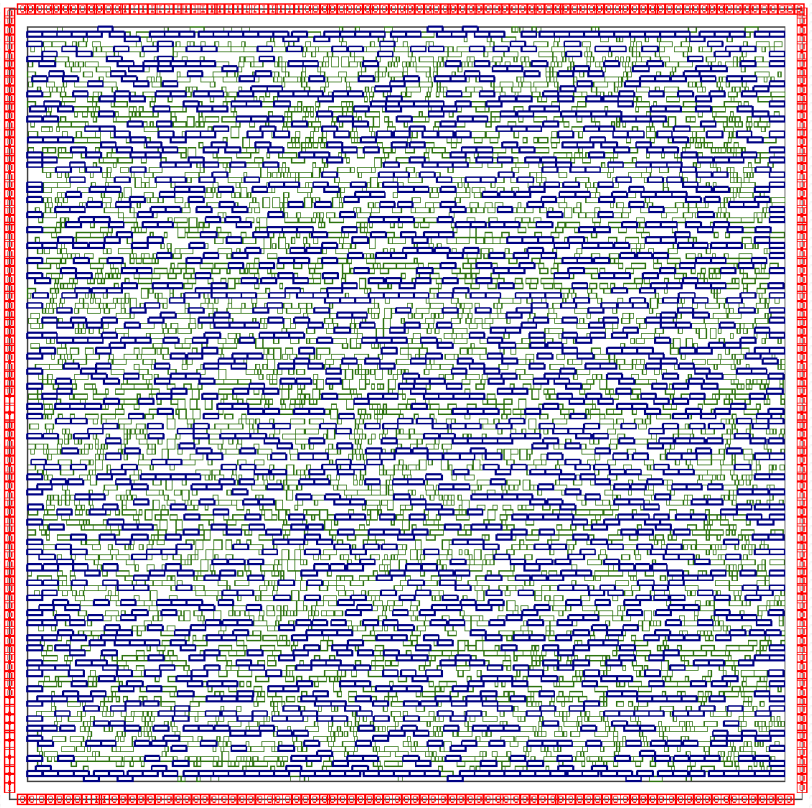
$$\text{s.t.} \quad D^i_{fo_k} + D^i_{CQ} + D^f_{fi_k} \le T - S_f - L_{if} - D^{if}_{m_k},\ \forall (R_i \rightarrow R_f)$$
$$D^i_{fo_k} = K_w C_0 w^i_{fo_k},\ \forall R_i$$
$$D^i_{CQ} = D^i_{R0} + K^i_r C_0 w^i_{fo_k},\ \forall R_i$$
$$D^f_{fi_k} = K_w C_0 w^f_{fi_k} + D^f_{G0} + K^f_G C_0 w^f_{fi_k},\ \forall R_f$$

Non-overlap constraints

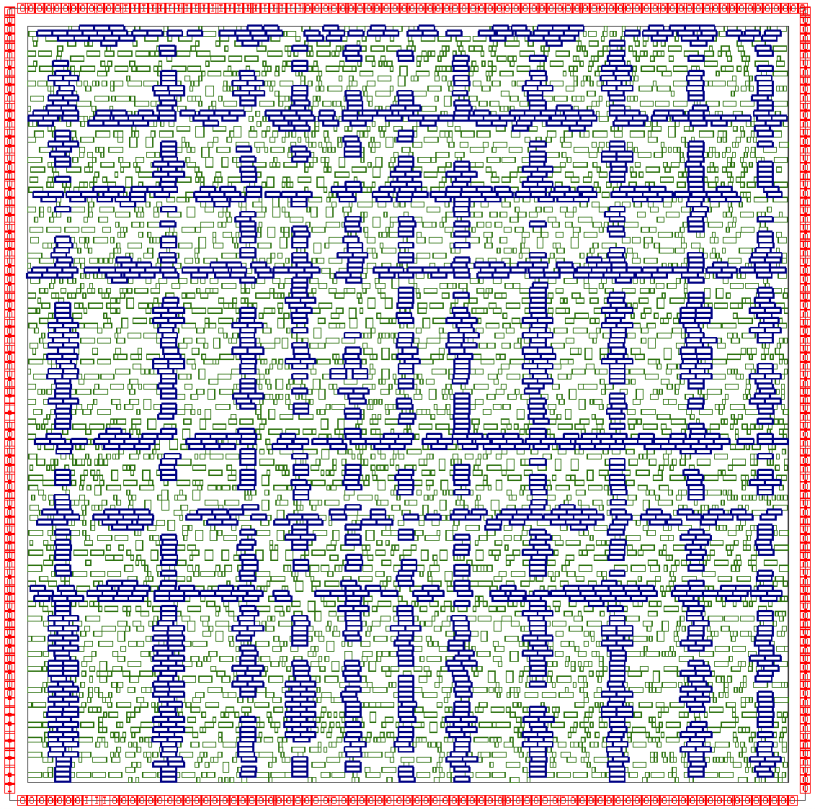
$$w^i_{stub} = xdist(R_i, M_{vj})\ (or\ ydist(R_i, M_{hj})|),\ \forall R_i$$
$$xdist(R_i, M_{vj}) \ge x_{R_i} - x_{M_{vj}},\ \forall R_i$$
$$xdist(R_i, M_{vj}) \ge x_{M_{vj}} - x_{R_i},\ \forall R_i$$
$$ydist(R_i, M_{hj}) \ge y_{R_i} - y_{M_{hj}},\ \forall R_i$$
$$ydist(R_i, M_{hj}) \ge y_{M_{hj}} - y_{R_i},\ \forall R_i$$
$$w^i_{fo_k} = xdist(R_i, fo_k) + ydist(R_i, fo_k),\ \forall R_i$$
$$xdist(R_i, fo_k) \ge x_{R_i} - x_{fo_k},\ \forall R_i$$
$$xdist(R_i, fo_k) \ge x_{fo_k} - x_{R_i},\ \forall R_i$$
$$ydist(R_i, fo_k) \ge y_{R_i} - y_{fo_k},\ \forall R_i$$
$$ydist(R_i, fo_k) \ge y_{fo_k} - y_{R_i},\ \forall R_i$$
$$w^f_{fi} = xdist(R_f, fi) + ydist(R_f, fi),\ \forall R_f$$
$$xdist(R_f, fi) \ge x_{R_f} - x_{fi},\ \forall R_f$$
$$xdist(R_f, fi) \ge x_{fi} - x_{R_f},\ \forall R_f$$
$$ydist(R_f, fi) \ge y_{R_f} - y_{fi},\ \forall R_f$$
$$ydist(R_f, fi) \ge y_{fi} - y_{R_f},\ \forall R_f$$
$$x_{R_i} - x_{R_j} \ge W_r.$$
$$or\ x_{R_j} - x_{R_i} \ge W_r.$$
$$or\ y_{R_i} - y_{R_j} \ge L_r.$$
$$or\ y_{R_j} - y_{R_i} \ge L_r.$$

The Incremental Placement Results (s35932 in ISCAS89)

Before placement

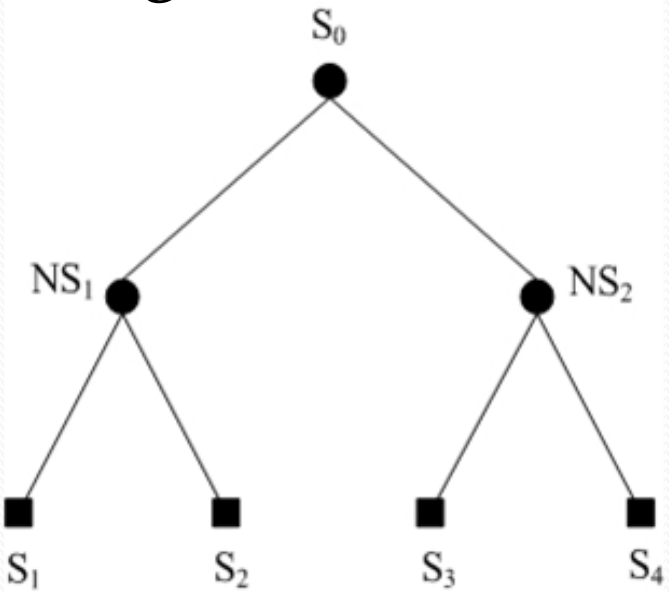


After placement



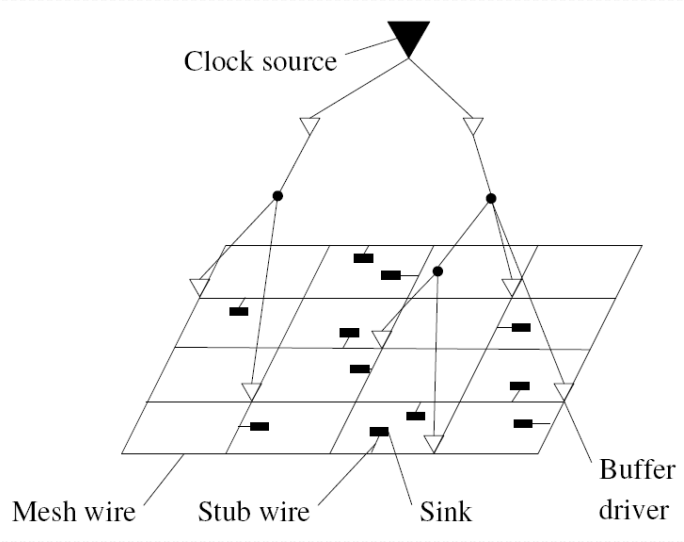
Top Level Clock Tree Generation

- Insert buffer drivers on the intersection of the mesh grid wires[1][2].
- Generate top level clock tree where the sinks are buffer drivers of the mesh grid wires. (Buffered DME)



Outline

- Preliminaries
- Previous Works
- Methodology
- Experimental Results
- Conclusions



Experimental Results

Set 1: Compare the proposed method with [2] using different grid sizes.

Circuit	Proposed	[2]
s13207	6*7	8*8
s15850	5*4	8*8
s35932	11*7	12*12
s38417*	10*9	12*12
s38584	12*7	11*11

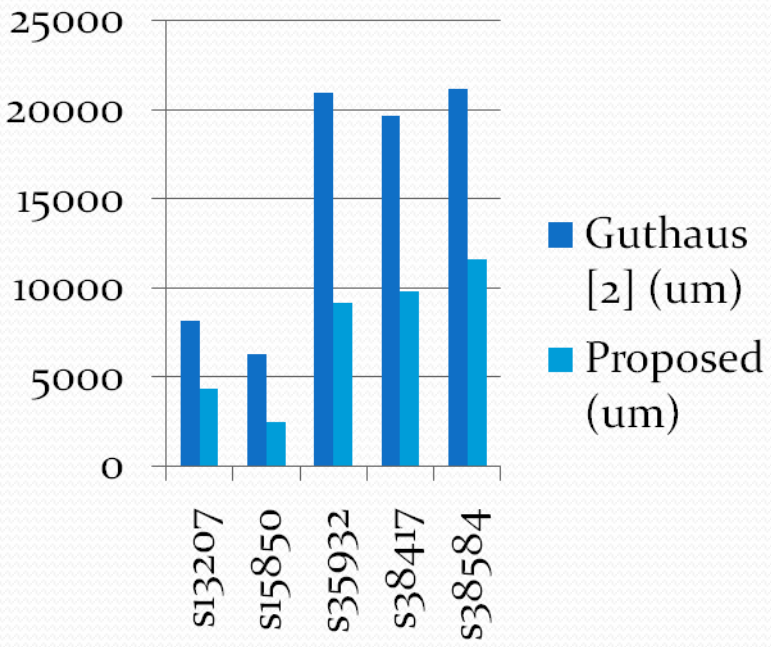
Set 2: Compare the proposed method with [2] using the same grid sizes.

Circuit	Proposed	[2]
s13207	6*7	6*7
s15850	5*4	5*4
s35932	11*7	11*7
s38417*	10*9	10*9
s38584	12*7	12*7

[2] M. R. Guthaus, G. Wilke, and R. Reis. Non-uniform clock mesh optimization with linear programming buffer insertion. In *Proceedings of the ACM/IEEE Design Automation Conference (DAC), pages 74–79, June 2010.*

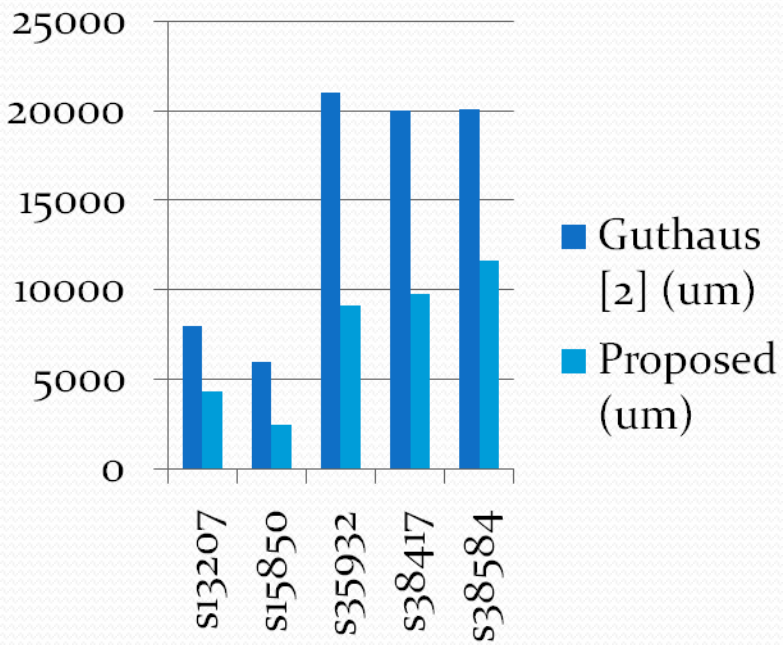
Mesh Wire Reduction

Set 1 (Different grid size)



Average improvement of 51.9%.

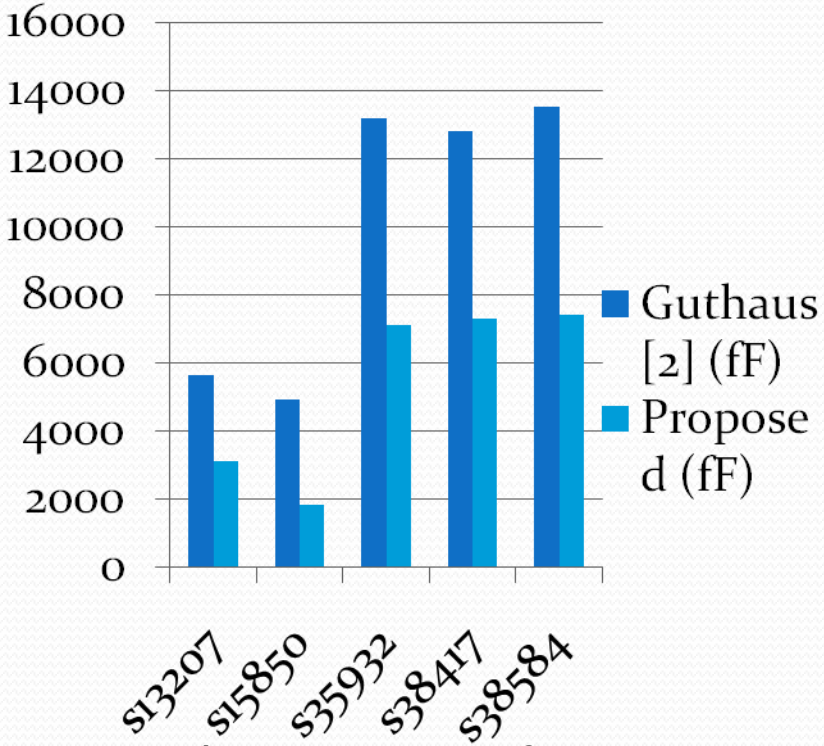
Set 2 (Same grid size)



Average improvement of 50.8%

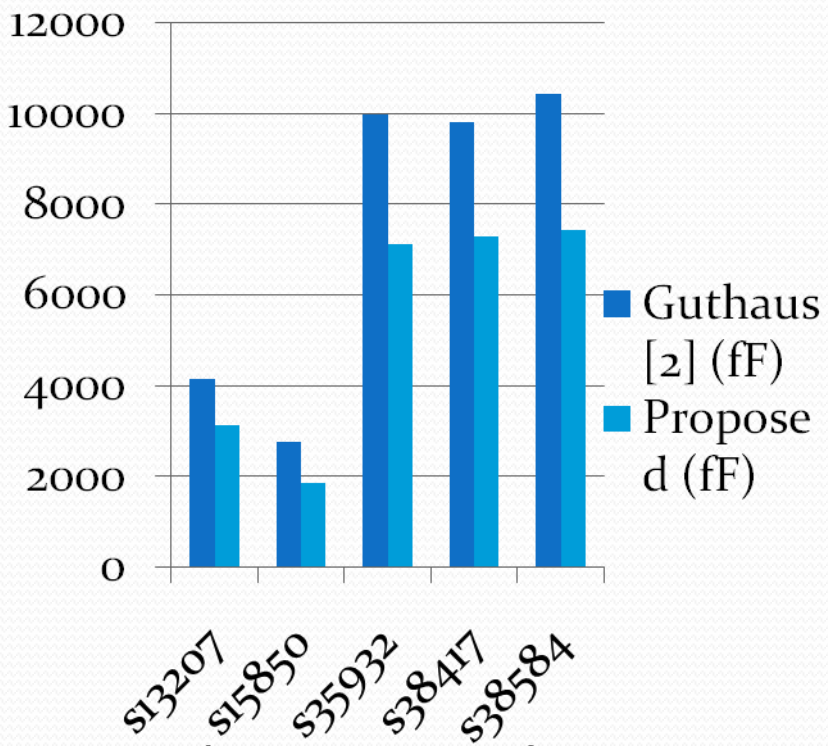
Clock Power Reduction

Set 1 (Different grid size)



Average improvement of 48.3%.

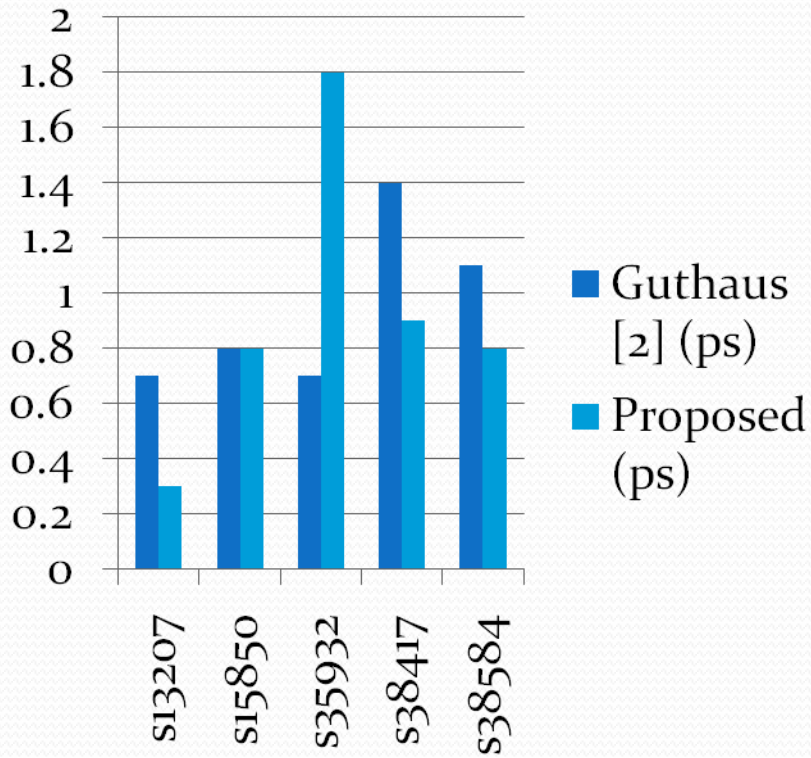
Set 2 (Same grid size)



Average improvement of 28.1%

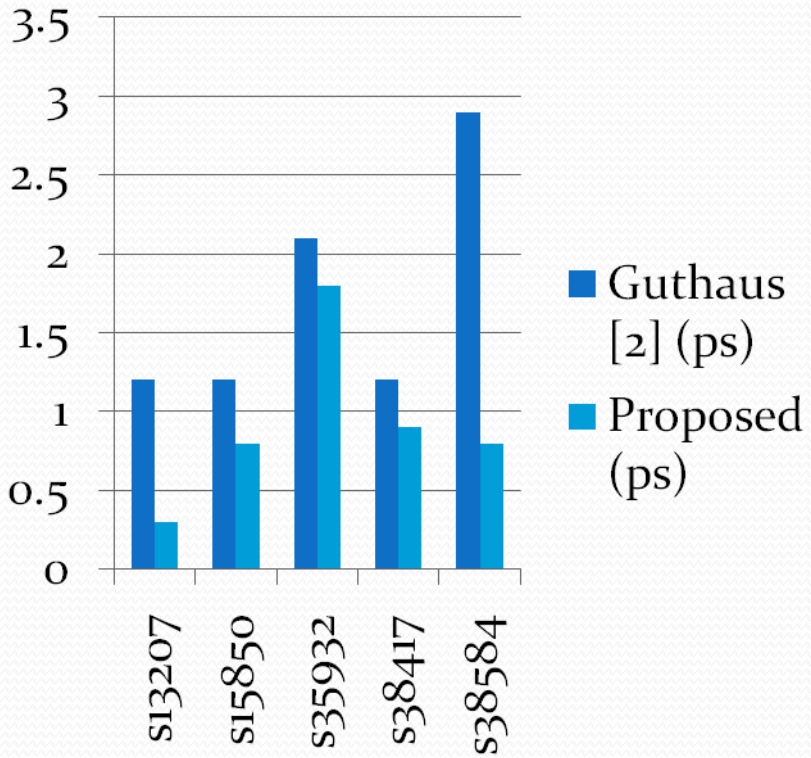
Skew Results (45nm PTM)

Set 1 (Different grid size)



Average skew is in the same range.

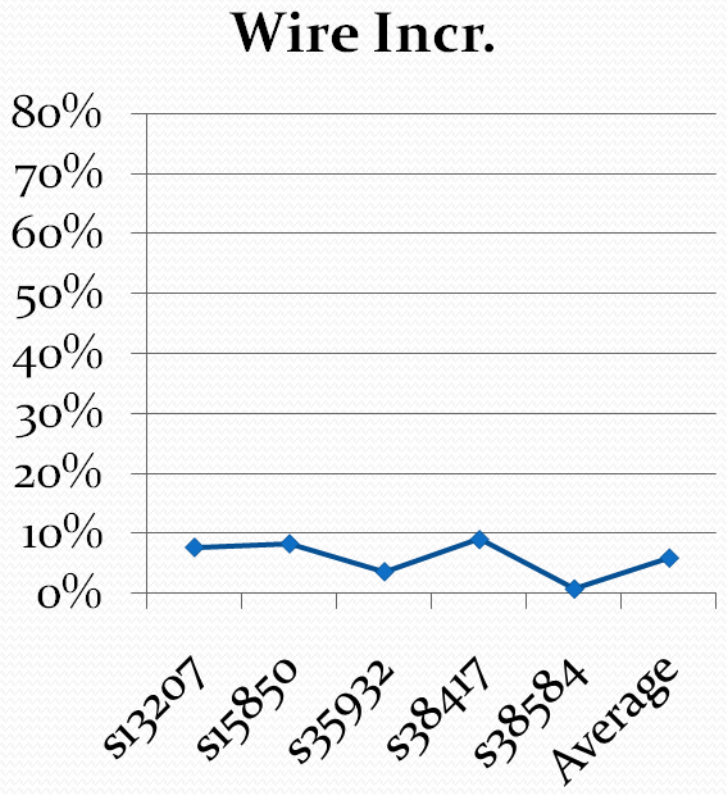
Set 2 (Same grid size)



Skew is improved by 0.8ps.

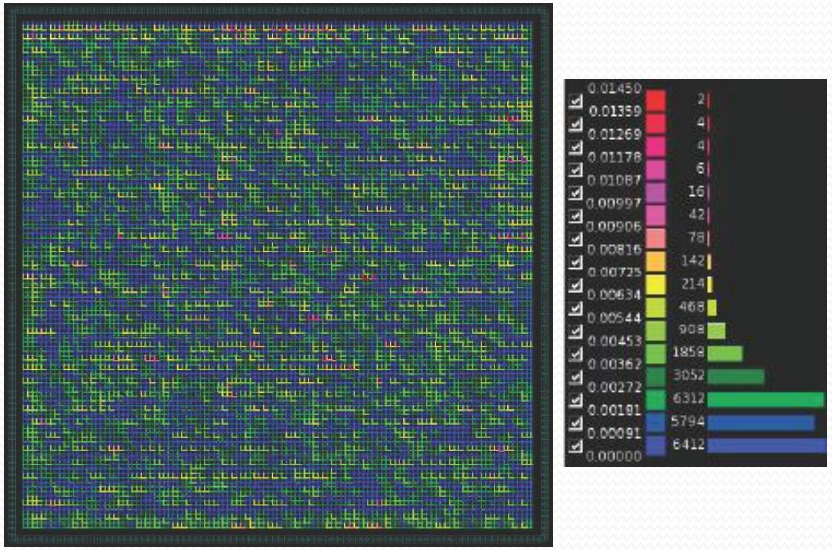
Trade-off

- The trade-off is the logic wirelength change due to the register placement.

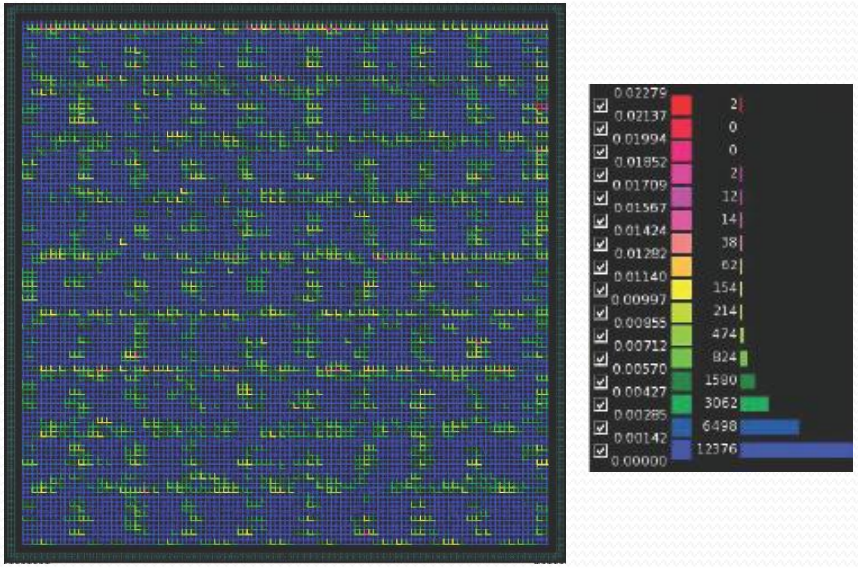


Implications of Placement Congestion

Before Register Placement

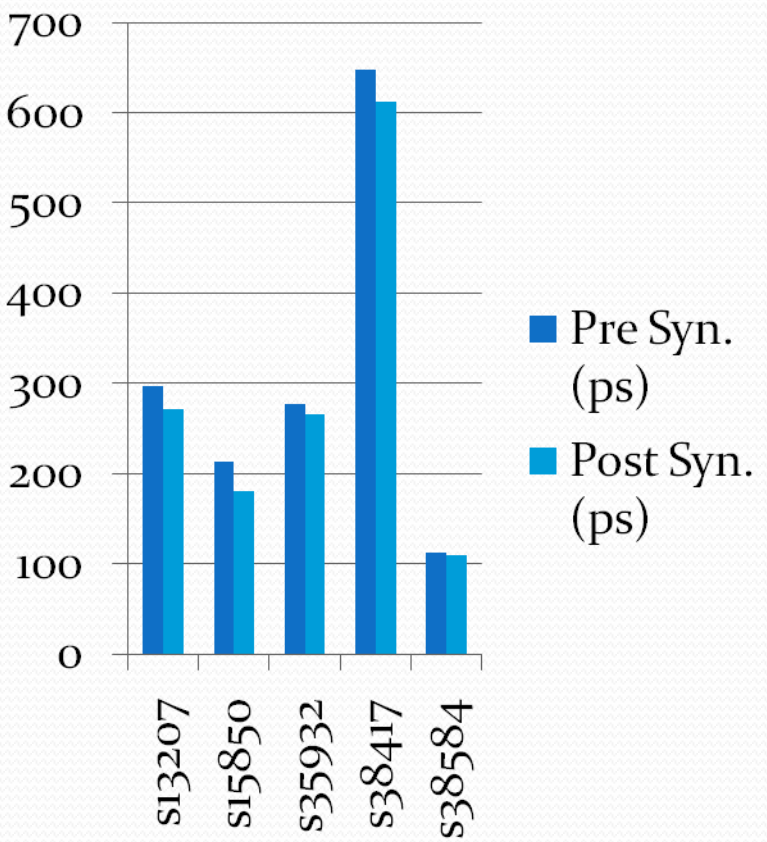


After Register Placement



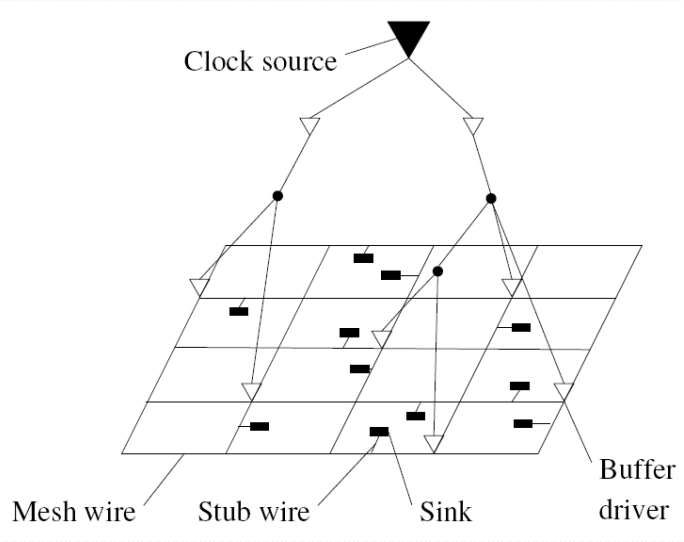
Routing Congestion

- The timing slack is decreased by an average of 22ps, which is very limited compared to the 2ns clock period.



Outline

- Preliminaries
- Previous Works
- Methodology
- Experimental Results
- Conclusions



Conclusions

- Advantages
  - Significantly reduced power dissipation.
  - Guaranteed timing slack (pre-routing).
- Disadvantages
  - Power density increase.
  - Timing slack decrease.

Thank You!