SAMSUNG

Hierarchical Power Grid Analysis for 3D-IC

Kyoungman Han, Yelim Lee, Jaeho Lee, Jiwoo Hong, YoungHoe Cheon, Daehee Lee and Jungyun Choi

Samsung Electronics, Memory Business

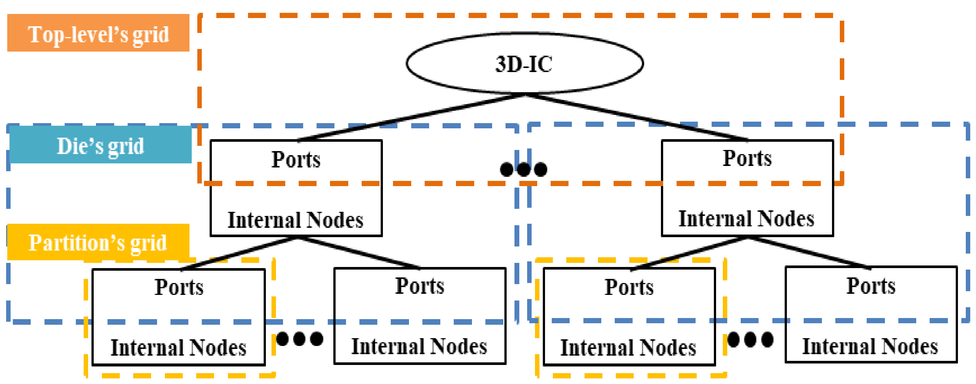
Motivation : Needs for 3D-IC IR-Drop Analysis Methodology

- **As a memory device 1) higher speed, 2) lower power, 3) more density and 4) stacked, a robust Power Distribution Network(PDN) design's importance is increased**
  - For a robust PDN design, designer run IR-Drop and Dynamic Voltage Drop(DVD) analysis and fix weak points in PDN before Mask Tape Out(MTO)
- **As a memory device size increased significantly in 3D-IC memory device, analysis run time is also increased**
  - Recently 3D-IC memory device's PDN have several billion nodes and PDN with this size cannot be analyzed in a traditional IR-Drop analysis methodology
    → For a fast IR-Drop analysis, PDN reduction is key factor and many methodologies are proposed
  - For PDN reduction, *several methodologies are proposed. These methodologies are efficient for single die but insufficient for 3D-IC with several dies * 1) Variable Reduction Method, 2) Multi Grid, 3) Model Order Reduction (MOR)
  - To overcome this limit, the hierarchical methodology is proposed
- **We propose a hierarchical methodology for 3D-IC memory device. And applied memory devices**
  - The PDN is partitioned and is analyzed each partitioned PDN
  - For an efficient hierarchical analysis, partitioning and calculation for analysis are key factors
  - Use Schur complement methodology for an efficient macro-modeling and port current/node voltage calculation
  - For efficient PDN partitioning, we use a parallel partition methodology and design information

Hierarchical IR-Drop analysis for 3D-IC memory PDN methodology

- **Hierarchical IR-Drop analysis for 3D-IC memory PDN method based on a *hierarchical methodology : has 2 steps**
  * Min Zhao, Rajendran V. Panda, Sachin S. Sapatnekar, and David Blaauw, Hierarchical Analysis of Power Distribution Networks, *IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, VOL. 21, NO. 2*, 159-168 (2002)
  - Setup step) PDN is partitioned and a macromodel of each PDN partition in top-level and die-level is made.
  - Analysis step) macromodel's port current, port voltage, and macromodel's internal node voltage are calculated
- **Two methodology are proposed for efficient analysis step**
  - Efficient macromodeling and port current/node voltage calculation methodology using Schur complement : Additional matrix operation can be reduced
  - Each macromodel analysis was implemented as a parallel process using Message Passage Interface(MPI)
- **For 3D-IC memory device, 2 partition strategies are used**
  - First, top-level partitioning is used 3D-IC memory device design information
    - 3D-IC memory device's top-level consist of dies and interconnects between die and die (Interconnections are Through Silicon Via(TSV) or wire-bonding)
  - Second, die's internal partitioning is used the partition algorithm
    - In a memory design, design isn't a hierarchical design in PDN perspective, so we use *graph partitioning algorithm

* Henning Meyerhenke, Peter Sanders, Christian Schulz, "Parallel Graph Partitioning for Complex Networks", *arXiv:1404.4797v3 [cs.DC]* 26 Jan 2015.



[Figure.] Hierarchical Methodology Structure for 3D-IC memory device

- **Traditional Flat Analysis Methodology VS. Proposal Hierarchical Analysis Methodology**

	Traditional Flat Analysis Methodology	Proposal Hierarchical Analysis Methodology
Setup step	① Make 3D-IC PDN Matrix A	① Make die's partitions's macromodel matrix Ai ② Make die's macromodel matrix Aj ③ Make top-level's matrix A_0
Analysis step	① Update Right Hand Side Vector(RHS) ② Solve 3D-IC PDN node voltage ③ Repeat 1~2	① Make die's partition's port current Si ② Make die's port current Sj ③ Update top-level's Right Hand Side Vector(RHS) ④ Solve top-level's node voltage ⑤ Solve die's node voltage ⑥ Solve die's partition's node voltage ⑦ Repeat 1~6
Matrix Sample	$\begin{bmatrix} G_{00} & G_{01} & \dots & G_{0k} \\ G_{01}^T & G_{11} & \dots & G_{1k} \\ \vdots & \vdots & \ddots & \vdots \\ G_{0k}^T & G_{1k}^T & \dots & G_{kk} \end{bmatrix} \begin{bmatrix} V_0 \\ V_1 \\ \vdots \\ V_k \end{bmatrix} = \begin{bmatrix} I_0 \\ I_1 \\ \vdots \\ I_k \end{bmatrix}$	Top' s Grid $\begin{bmatrix} G_{00} & G_{01} & \dots & G_{0k} \\ G_{01}^T & A_1 & \dots & G_{1k} \\ \vdots & \vdots & \ddots & \vdots \\ G_{0k}^T & G_{1k}^T & \dots & G_{kk} \end{bmatrix} \begin{bmatrix} V_0 \\ V_1 \\ \vdots \\ V_k \end{bmatrix} = \begin{bmatrix} I_0 \\ -S_1 \\ \vdots \\ -S_k \end{bmatrix}$ Die' s Grid $\begin{bmatrix} G_{00} & G_{01} & \dots & G_{0k} \\ G_{01}^T & A_1 & \dots & G_{1k} \\ \vdots & \vdots & \ddots & \vdots \\ G_{0k}^T & G_{1k}^T & \dots & G_{kk} \end{bmatrix} \begin{bmatrix} V_0 \\ V_1 \\ \vdots \\ V_k \end{bmatrix} = \begin{bmatrix} I_0 \\ -S_1 \\ \vdots \\ -S_k \end{bmatrix}$ … Partition' s Grid $\begin{bmatrix} G_{00} & G_{01} & \dots & G_{0k} \\ G_{01}^T & G_{11} & \dots & G_{1k} \\ \vdots & \vdots & \ddots & \vdots \\ G_{0k}^T & G_{1k}^T & \dots & G_{kk} \end{bmatrix} \begin{bmatrix} V_0 \\ V_1 \\ \vdots \\ V_k \end{bmatrix} = \begin{bmatrix} I_0 \\ I_1 \\ \vdots \\ I_k \end{bmatrix}$ …

Experimental Results

- **We implemented an analysis program in C and C++ to evaluate performance of the proposed methodology**
- **IR-Drop run time of the proposed method is -32.4% ~ -64.5% less than the previous methodology(use variable remove methodology) (run on a Linux machines with 72 cores)**

Data Case	Data Set Info		Run Time / Peak Memory Compare			
	# Stack	#Node	Analysis Methodology	# Core	Total Run Time[Hr]	Total Memory Usage[GB]
TC 1 IR-Drop	3	1.81G	Previous	72	18.27	4884.43
			Proposed	72	6.49	4324.92
	9	4.37G	Previous	72	Fail	
			Proposed	144	7.60	8347.95
TC 2 IR-Drop	8	7.47G	Previous	72	Fail	
			Proposed	256	12.05	11425.82
TC 1 DVD	2	1.38G	Previous	64	42.46	5218.75
			Proposed	64	28.72	5547.54
	9	4.37G	Previous	72	Fail	
			Proposed	288	26.46	16319.28

*** Node voltage error rate is zero**

Summary

- **Traditional PDN reduction methodology is useful for IR-Drop analysis of a single die, but inadequate for large PDN like 3D-IC with multiple dies**
- **We proposed the hierarchical methodology for 3D-IC with multi dies**
  - Experimental results show that the propose methodology is -32.4% ~ -64.5% less than traditional methodology but have error rate is zero
  - The proposed methodology's run time limit is largest PDN size of multiple die's PDNs but total PDN size of multiple dies
- **Future works**
  - How to minimize interference between each macro model when analyze the hierarchy is required
  - How to use GPU Solver to shorten analysis time
  - How to analysis a 3D-IC multiple dies with different features
  - How to simultaneously analysis Multi-Scenario using MPI or MP