ELECTRONIC DESIGN AUTOMATION TOOLS AND DESIGN STUDY FOR HETEROGENEOUS MONOLITHIC 3D INTEGRATED CIRCUITS

A Dissertation
Presented to
The Academic Faculty

By

Sai Surya Kiran Pentapati

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology

May 2022

ELECTRONIC DESIGN AUTOMATION TOOLS AND DESIGN STUDY FOR HETEROGENEOUS MONOLITHIC 3D INTEGRATED CIRCUITS

Thesis committee:

Dr. Sung Kyu Lim
Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Callie Hao
Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Saibal Mukhopadhyay
Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Hyesoon Kim
College of Computing
Georgia Institute of Technology

Dr. Shimeng Yu
Electrical and Computer Engineering
Georgia Institute of Technology

Date approved: April 1, 2021

ACKNOWLEDGMENTS

As I approach the end of my journey as a doctoral student, I would like to take this opportunity and thank everyone who have helped me to reach this point of my journey.

First of all, I like to thank Dr. Sung Kyu Lim who has been an integral part of my Ph.D. guiding me along and helping me gain my footing in my first year. He has been a great mentor and taught me some invaluable lessons. I would also like to thank the members of the proposal and dissertation committees – Dr. Arijit Raychowdary and Dr. Shimeng Yu for their great advice during the proposal exam, and Dr. Saibal Mukhopadhyay, Dr. Hyesoon Kim, and Dr. Callie Hao for taking their time and effort in serving as part of my dissertation committee.

I'm also grateful for all the great mentors, collaborators, managers, and colleagues from the industry: Dr. Asif Khan, Nick Samra, Vassilios Gerousis, Rwik Sengupta, Harsono Simka, Dr. Gary Yeap, Nick Lafrenz, Dr. Xiaoqing Xu, Dr. Thorlindur Thorolfsson, Kurian Abraham for their technical discussions and industry insights. I also like to show my sincere appreciation to all the GTCAD members with whom I had many fruitful discussions: Dr. Sandeep Samal, Dr. Kyungwook Chang, Dr. Bonwoong Ku, Rakesh Perumal, Dr. Heechun Park, Lennart Bamberg, Jinwoo Kim, Anthony Agnesina, Da Eun Shim, Dr. Junsik Yoon, Lingjun Zhu, Yi-Chen Lu, Gauthaman Murali, Pruek Vanna-Iampikul

I am also very grateful for all my friends and family whose conversations and chats always make me more cheerful, especially, all the great times I had with my cousins, and my brother, Kaushik, who is the best friend I could ask for (although I'm scared to show this to him as he would never let it go).

Lastly, but most importantly, I want to show my heartfelt gratitude towards my parents, Sai Murali and Venkata Lakshmi, who have been very supportive of my decisions ever since I was capable of making informed decisions, and being with me every step of the way. I can never thank them enough and hope this is at least a drop in the ocean.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

The objective of this research is to explore and exploit novel design configurations possible with 3D ICs. Furthermore, several tool flows and algorithms were developed to augment and capitalize on the commercially available Electronic Design Automation (EDA) tool flows to support our exploration. While most of the technological assumptions made in our work were in early stages of research, we also develop new flows to refine the 3D IC routing with commercially available 3D IC fabrication techniques.

CHAPTER 1
INTRODUCTION

Technology scaling predicted by Moore's law [1] is gradually slowing down and new alternatives to silicon-based transistors are being explored. Some of the most promising solutions make use of materials such as carbon nanotubes [2] or ferroelectrics with negative capacitance effects [3]. While these materials bring improvements to the actual transistor structure, Three Dimensional (3D) Integrated Circuit (IC) Design is another such alternative [4] for going beyond the Moore's Law that operates orthogonally to the transistor material improvements. 3D integration improves power, performance, and area (PPA) by stacking multiple smaller 2D dies vertically instead of using a single 2D die with a larger footprint. This leads to shorter interconnects and adds an extra degree of placement freedom in the z-direction along with the traditional x,y-directions.

1.1 Fabrication and Packaging Techniques for 3D ICs

There are three main types of 3D ICs based on the fabrication or bonding techniques [5]: Monolithic 3D ICs (M3D IC), hybrid bonded 3D IC, micro-bump 3D IC.

Micro-bumping is a die-level 3D packaging technique where two known good dies are bonded together using micro-bumps usually in the order of 10 µm or larger. The large size of the bumps limits the number of bumps that can be used to connect the two dies and the high resistance and capacitance of the micro-bumps also limits the maximum connection speed achievable. While this is commercially most feasible, it can only be helpful in specific designs due to their limitations.

Hybrid bonding uses direct Cu-Cu bonds with pitch values around 1 µm at a wafer level to create the 3D IC. The smaller pitch significantly increases the allowed bandwidth between the two dies of the 3D design and is also quickly becoming a commercially available option. The fine-pitch for hybrid bonding at wafer level is still hard to accomplish with current technology processes and not many foundries offer this at present. But more research is being done into enabling sub-micron pitch for hybrid bonding [6] which can bring it to consumer electronics in the near future.

Monolithic 3D IC design is the most advanced technique for creating 3D ICs where the 3D ICs are directly fabricated unlike the packaging techniques of Micro-bumping and Hybrid-bonding. The different tiers of the 3D IC are sequentially fabricated on top of each other, removing the need for alignment of bumps/bonds. This can achieve an extremely fine via pitch of $\approx$0.1 µm. While this shows the best-case scenario for 3D ICs, the process of fabricating dies on top of each other is extremely challenging with many limitations related to the thermal budget and the materials that could be used in the fabrication. In recent years, CEA-LETI showcased a significant breakthrough in low temperature fabrication of the devices that shows potential for M3D IC manufacturing to lead the 'More than Moore' era of computing [7]. This sequential fabrication allows for a nanometer scale pitch for 3D vias that can unlock a variety of 3D IC designs.

1.2 Electronic Design Automation Flows for 3D ICs

To study the benefits and different characteristics of 3D ICs, Electronics Design Automation (EDA) tools are necessary for the placement, routing, and timing optimization of the 3D ICs. The commercially available tools such as Innovus from Cadence and IC Compiler from Synopsys do not natively support PnR for 3D ICs. As a result, two ways of tackling this issue have appeared. First is a more ground-level 3D implementation with placers such as [8, 9, 10, 11] that are mainly focused on improving the placement without regards to

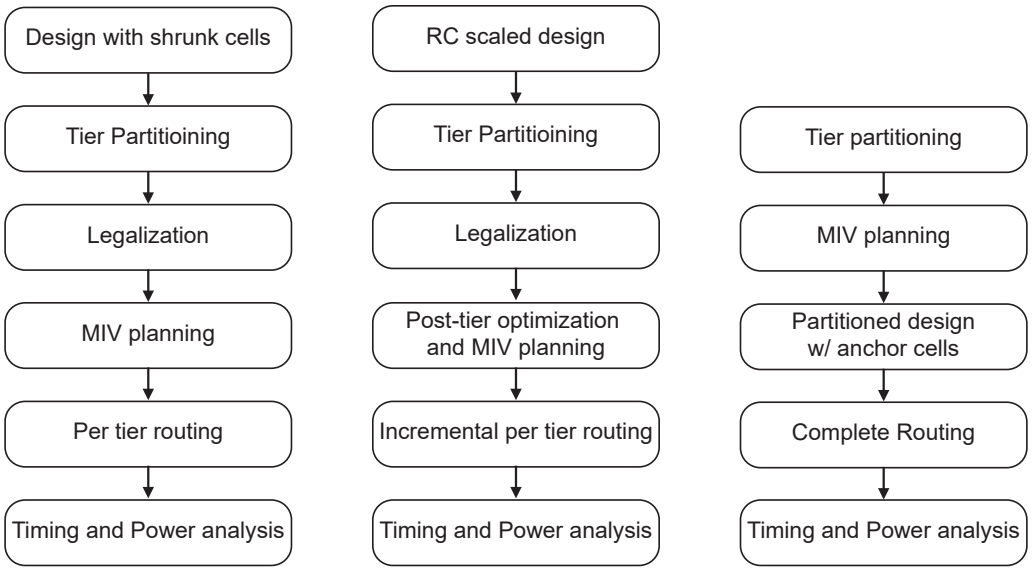
routing or the final Power, Performance, Area (PPA) results. On the other hand, tool flows such as [12, 13, 14] are used to develop a more holistic 3D IC design with Place and Route (PnR), and timing optimization using tweaks to the available commercial 2D PnR tools. Partitioning algorithms such as [15, 16] help to consider the 3D nature of the placement in such tool flows.

1.2.1 Placement in the Three Dimensional space

The academic placers take several heuristics to create the 3D IC placement. The authors in [11] perform 3D placement using a force-based algorithm that models connecting wires as a springs, and local cell density as a repelling force. The 3D placement in [9] is done based on partitioning based algorithms, that recursively partition the netlist structure and assign them to ever smaller areas, until the size of a partition makes the placement trivial. In [10], analytical solvers are employed for the 3D placement that minimize the wirelength while satisfying some density and overlap constraints. And finally, the authors in [8] perform placement by treating cells as charged particles and employing electrostatic field solvers to find the least placement of standard cells in a 3D space that has the least electric potential. While these placements might preform better at metrics like wirelength or the number of 3D-vias, they fail in terms of PPA when compared to the routing or timing driven placement of commercial tools.

1.2.2 Pseudo-3D Place and Route Flows

A tight integration between placement and routing, timing optimization is required to achieve good Power, Performance, and Area (PPA) of any IC. “Pseudo-3D flows” such as [17, 12, 18] have been proposed that utilize the commercial 2D EDA capabilities to 3D ICs. In a pseudo-3D flow, 3D designs are built using an “intermediate 2D design” and then partitioned into multiple tiers and routed to obtain the final 3D design. The placement in such flows transforms a 2D optimized placement into a placement for 3D IC designs.



(a) Shrunk2D flow as described in [17]

(b) Compact2D flow as described in [12]

(c) Cascade2D flow as described in [18]

Figure 1.1: Various Pseudo-3D Flows

Shrunk2D [17] is the first RTL-to-GDSII tool flow that creates near-optimal M3D designs using 2D EDA tools. The width and height of the standard cells are first shrunk by a factor of $1/\sqrt{2} = 0.707$ thereby halving the area of standard cells (Hence the name Shrunk2D). With the shrunk dimensions, all the cells fit in a floorplan half the area of a 2D floorplan with the same cell density of a normal 2D design. With the halved footprint, the distance between cells in an M3D design is approximately $0.707\times$ of 2D designs which gives a theoretical wirelength savings of $\sim 30\%$. The shrunk design can be treated as a proxy for M3D design because the cell distances are almost equal to that of the M3D design (ignoring the z-direction distances which are relatively short). At this stage, the design is still considered 2D and all the optimization capabilities of a commercial EDA tool such as cell sizing, buffer insertion, removal, routing, power optimization, timing closure etc., can be leveraged. This concludes the 'pseudo-3D' stage of the flow.

The intermediate shrunk2D design falls short as a proxy for 3D designs in some key aspects. In the original shrunk2D flow presented in the paper [17], the wire widths and pitches are scaled by a factor of 0.707 to allow for routing on a smaller footprint. In

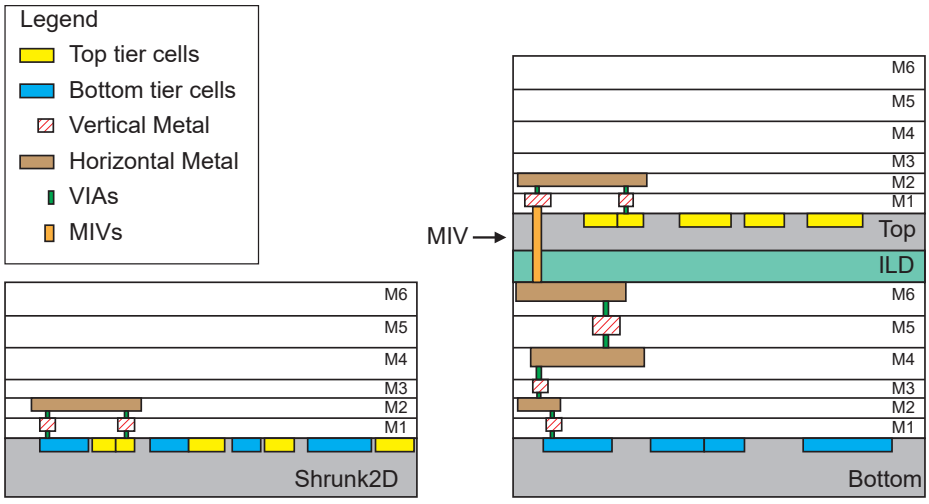


Figure 1.2: Routing overhead of a 3D net

this case, the authors did not consider the change in wire parasitics due to the geometry scaling. In [19] the authors address this issue to create better RC-lookup tables so that the parasitics in the shrunk2D stage mirror the parasitics in the M3D design. Correct parasitic estimation is necessary for EDA engines to optimize M3D designs. Another drawback is that the shrunk2D flow does not provide optimization after tier partitioning meaning that the cell movement during legalization and the routing overhead of 3D nets(Figure 1.2) are not considered.

Compact2D [12] is another RTL-to-GDSII tool flow. Compact2D simply scales the RC parasitics of the wires linearly without shrinking any geometry. This RC shrinking is a proxy for the 3D parasitics to be represented in the 2D footprint. Once the design is optimized, the cells are then mapped to a halved footprint by linearly scaling the x, y coordinates. The 3D cell placement is then achieved in the same fashion as shrunk2D using bin-based FM min-cut partitioning.

The second big improvement of compact2D is its post-partitioning optimization, as it re-optimizes the whole design after tier-partitioning to consider the cell movement during legalization and the 3D net overhead. The design is therefore optimized with 3D overhead and 3D metal layer structure in consideration. There still remain a few drawbacks

in Compact2D as the linear scaling of parasitics is inaccurate as the scaling depends on characteristics of the net. While the post-partitioning optimization helps fix the timing and power, the representation of the design at this stage leaves some design rule violations in the routing, and does not fully support a 3D clock tree optimization.

Cascade2D [18, 20] is a different type of 3D RTL-to-GDSII flow where the z-location(tier assignment) of the cells are determined before performing x-y optimization unlike the previous two flows. The 3D connections are treated as special kind of cells, and a co-iterative placement of the standard cells and the 3D cells in the two dies create the placement. Unlike Shrunk2D or Compact2D flows, the availability of z-location allows the authors to create a better pseudo-3D stage that considers the different peculiarities of 3D connections. The drawback here is that the partitioning cannot be dense as it can negatively impact the pseudo-3D representation, limiting its usage in densely connected designs. Additionally, the PPA quality is significantly affected by the partitioning which needs to be done based on the detailed micro-architecture and data-flow analysis.

1.3 Organization

The main contributions of this dissertation encompass three different themes. The first theme corresponds to *the design of better EDA design flow and heuristics* in chapter 3 and chapter 2. Here we tackle the issues of current pseudo-3D flows to create a more robust and efficient 3D IC. The second theme comprises of chapter 4 and chapter 5 where we specifically deal with the *routing in 3D IC designs*. We thoroughly analyze the different routing structures in 3D, identify potential issues with routing in advanced nodes and propose refinements to address such problems. Finally, the last theme corresponds to *the exploration of 3D IC arrangements* in chapter 6 and chapter 7. A specific partitioning type that can immensely benefit from 3D ICs is discussed in chapter 6, and a novel heterogeneous technology scheme for 3D ICs is discussed in chapter 7.

While the different chapters can be treated as a part of an encompassing theme, they are self-contained as follows:

- In chapter 2, *for the design of better EDA design flow*: we propose a Machine Learning based prediction algorithm to decrease the discrepancy between the pre and post partitioned 3D design using regression models. Our proposed model is circuit-agnostic and its performance with respect to a circuit dependent model is also studied. Furthermore, more details on the behavior and analysis of the model is considered. Overall, we achieve significant reduction in the total negative slack of the test design (3x – 16x) using the machine learning model integrated pseudo-3D flow at the expense of just −1 to 4 % increase in total power.

- In chapter 3, *for the design of better EDA design flow*: we present incremental placement, clock optimization, complete routing, and timing optimization flows for 3D ICs. Using technology file hacks, we load the complete 3D design at once including the physical and logical connectivity of the netlist and library cell timing without causing cell overlaps. This helps with better 3D optimization, as well as 3D ECO for manual changes to the design. With Pin-3D, we were able to achieve a 10× smaller total negative slack compared to the recent flows that do not support 3D timing closure. The improved placement and routing flows also produce up to a 9% further reduction in wirelength in 3D. Compared to 2D IC designs, the 3D designs with Pin-3D flow have 9-32% power reduction due to 3D wirelength savings of 24-38% including a 17-33% reduction in the leakage power. Overall compared to 2D, the reduction in Energy Delay Product of 3D ICs is between 18% to 28% depending on the design.

- In chapter 4, *for the analysis and enhancement of routing in 3D IC designs*: we analyze and quantify a specific kind of routing novel to 3D ICs. While many recent studies have shown the benefits of 3D IC design on timing and power consumption

of circuits, routing in 3D is solely done with the automatic commercial routers and has not been well studied. In this paper, we show that 3D routing is far from simple and discuss the various routing scenarios in 3D that arise from the cell partitioning and the 3D metal layer stack. Unlike 2D, the metal layer configuration in 3D depends on the orientation of the dies that are bonded together. Due to this, depending on the 3D configuration, cells in one tier tend to use routing layers from the other tier. This is referred to as Metal Layer (or) Routing Sharing which depends on the metal layer stack and the cell partitioning in 3D, as well as the via pitch used for 3D connections. By analyzing metal layer sharing in detail, we see that it can help reduce metal layer costs in 3D, as well as improve the power consumption, and in some cases, the maximum achievable performance of the circuits. Overall, the 3D BEOL cost can decrease by 9% along with an improved Power Delay Product of up to 7.5% just from the routing sharing in Monolithic 3D ICs.

- In chapter 5, *for the analysis and enhancement of routing in 3D IC designs*: we identify the problem of routing in 3D IC designs with commercially viable bonding types and/or advanced technology nodes. State-of-the-art 3D IC Place-and-Route flows fail to honor the 3D via spacing rules when realistic pitch values are used. Here, we propose an added 3D via legalization stage during routing to reduce such violations. A force-based solver, and an ML-guided bipartite-matching algorithm are presented as viable legalizers compatible with various process nodes, bonding technologies, and partitioning types. With the modified 3D routing stage, we reduce the via overlap violations by more than 10x without any performance, power, or area impact.

- In chapter 6, *for the exploration of 3D IC arrangements*: we present the benefits of M3D ICs using OpenPiton, a scalable open-source RISC-V based multi-core SoC. With a logic-on-memory 3D integration, we analyze the power and performance

benefits of two OpenPiton single-tile systems with smaller and larger memory architectures. The logic-on-memory M3D design shows 37% performance improvement compared to the corresponding tile design in 2D. And at iso-performance, M3D shows 14% total power saving.

- In chapter 7, *for the exploration of 3D IC arrangements*: we explore a novel heterogeneous design of Monolithic 3D ICs along with crucial design flow enhancements and better partitioning methods. The heterogeneous M3D ICs are designed with a combination of low-cost, low-power, and low-performance cells on one die and a higher-cost, power, and performance technology variant on the tier, for heterogeneity. These heterogeneous designs out-perform most 2D, 3D variants in Power-Delay Product and Cost metrics. Using 4 different netlists, we see up-to 23% improvement in Performance per Cost, and 16% improvement of Power Delay Product with heterogeneous M3D compared to the best 2D designs.

- Finally, in chapter 8, we summarize all of our results and benefits from each chapter.

CHAPTER 2

MACHINE LEARNING INTEGRATED PSEUDO-3D FLOW FOR MONOLITHIC 3D ICS

In this chapter, we present a machine learning integration framework for pseudo-3D flows such as [12, 17]. Better estimations of RCs early in the design stage will improve the final PPA in general. In the pseudo-3D flows for 3D ICs in particular, this RC estimation becomes more important as the timing optimization is exclusively done in the pseudo-3D stage. The 3D parasitics depend on a lot of variables as will be discussed later in subsection 2.1.2, and so a machine learning framework is well suited to learn the different interactions of the net features to estimate the 3D R and C values.

We use a total of 12 memoryless logic RTL downloaded from opencores, ISPD contests [21, 22, 23]. These are designed with a 28 nm commercial technology node. For all the 2D designs, 6 metal layers are used for signal routing. 3D designs have two tiers and a total of 12 signal routing layers (6 per each die). All the machine learning implementation are done with python3.6.

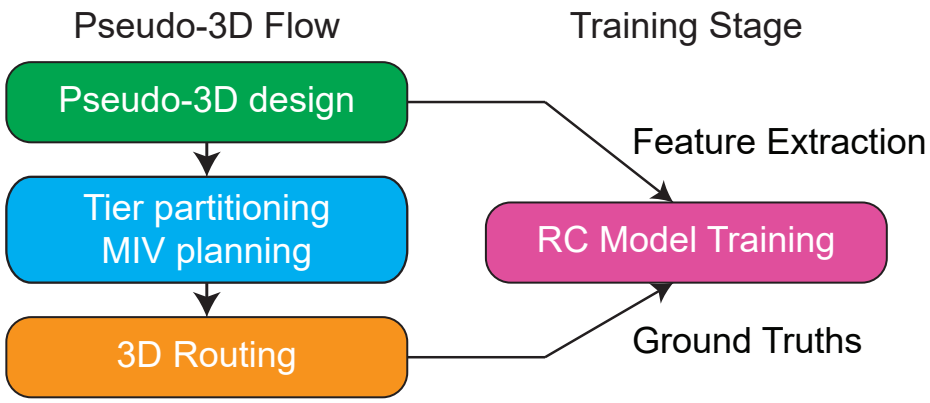


Figure 2.1: Pseudo-3D flow and training from the corresponding features

2.1 RC Analysis

2.1.1 RC breakdown of a net

In a GDS layout of an RTL, the nets are complex 3D structures whose parasitics depend on the exact shape of the net as well as the overall BEOL(Bottom-End-of-Line) dielectrics, neighboring nets. But such detailed analysis require a significant amount of time for 3D extractions and spice simulations. So the commercial EDA tools use several assumptions and simplifications to achieve a trade-off between accuracy and run-time.

Consider a rectangular wire of width W, thickness T, length L, at a distance H from the ground plane. The resistance and ground capacitance are given by $C_{wire} = \frac{\epsilon_d W L}{H}$; $R_{wire} = \frac{\rho L}{WT}$, where ϵ_d is the dielectric constant of the dielectric between the wire and the ground plane. While the resistance model is fairly simple, the total capacitance is much more complex with contributions from fringing effect of the ground capacitance, and coupling capacitance. Modern day IC designs also use multiple layers of metals and the $W, T, H, \epsilon_d, \rho$ can be different for the different metal layers. In an EDA tool, it is not feasible to calculate these capacitances from just the physical dimensions, and material properties. So, an RC look-up table is provided by the technology foundry containing pre-calculated unit length ground capacitance, coupling capacitance, and resistance values of the wires (denoted by lower-case c, r in this paper), and the vias at different scenarios (width, thickness, spacing, temperature, etc.) for each metal layer. The total resistance, and capacitance of a net with wires of length l_{M_i} on metal layer M_i, and n_{V_i} number of vias of type V_i is given by:

$$C_{net} = \sum_{M_i} c_{M_i} l_{M_i} + \sum_{V_i} c_{V_i} n_{V_i} + XCap \tag{2.1}$$

$$R_{net} = \sum_{M_i} r_{M_i} l_{M_i} + \sum_{V_i} r_{V_i} n_{V_i} \tag{2.2}$$

where $Xcap$ is the cross coupling capacitance between the net and the neighboring nets in the design. $c(r)_{M_i}$ is the capacitance (resistance) of a wire of length 1 µm, and $c(r)_{V_i}$

is the capacitance (resistance) of a via of type V_i. Note that there can be multiple types of vias from metal layer M_i to M_{i+1}. Coupling capacitance of the net is dependent on the final routing. While most of the nets have a negligible coupling capacitance, the nets that are routed in congested areas can have majority of the total capacitance as the coupling capacitance.

2.1.2 RC Evolution from Pseudo-3D to Final-3D designs

As Compact-2D's pseudo-3D stage works under the assumption that the wire RCs scale down by a factor a $\frac{1}{\sqrt{2}}$ when the design is converted from pseudo-3D to 3D stage. So, the scaling factor is applied in the pseudo-3D stage. This assumption is true in an ideal case, but the discrete row placement of cells and complex routing algorithms of commercial tools create variations in the scaling factor. Even in a global sense, the overall RC reduction is rarely as expected. Furthermore, the global scaling is applied only considering the length reduction portion in (Equation 2.1), (Equation 2.2). Number of vias on a net is much harder to predict as the routing in pseudo-3D (6 metal layers total) and final-3D (12 metal layers total) is very different. The contact resistance of vias keep increasing in smaller technology nodes, and ignoring the via resistance on the overall resistance can cause inconsistencies in the resistance of nets from the pseudo-3D to final-3D stages. In the technology node considered here, the unit values for metal layer 4 are as follows: $c_{M_4} \approx 0.20\,\mathrm{fF/\mu m}$, $c_{V_4} \approx 0.02\,\mathrm{fF/\mu m}$, $r_{M_4} \approx 10.0\,\Omega/\mu\mathrm{m}$, $r_{V_4} \approx 8.0\,\Omega/\mu\mathrm{m}$. It is clear that the contact resistance is significant even in the relatively older 28 nm node, considering the total wirelength and via count in Table 2.1

To further analyze the RC evolution in a design implementation, we design AES-128 circuit in pseudo-3D and final-3D stages. Some of the useful metrics from this implementation is shown in Table 2.1. Note that the wire length, ground capacitance, wire resistance are considerably underestimated in the pseudo-3D stage. Via Count increases by $\sim 17.5\%$, but the global scaling ($\frac{1}{\sqrt{2}}$) performed is suitable for a $\sim 30\%$ reduction in the via count.

Table 2.1: Routing statistics of the Pseudo-3D and the final 3D routed designs

	Pseudo-3D	Final-3D
Footprint (mm^2)	0.228	0.114
Metal Stack	6 Layers	12 Layers
Wire length† (µm)	1,141,179	1,234,332
Via Count	872,931	1,028,040
Ground Capacitance† (pF)	119.34	140.83
Coupling Capacitance† (pF)	35.20	31.71
Wire Resistance† (MΩ)	15.26	21.23

† Wire length, Capacitance, Resistance values of pseudo-3D are scaled by $\frac{1}{\sqrt{2}}$ to show a clear representation of the estimations

This increase in 3D is due to the halved footprint (or number of tracks per layer) and twice the number of vertical layers compared to a 2D or pseudo-3D implementation.

The scatter plot of the wire parasitics as a function of the routed wire length are in Figure 2.2 visualizes a couple of trends. One, the wire resistance can be given by a set of linear functions of the wire length, whose slope is fixed and intercept increases with the number of vias on the net. Two, the via capacitance has a negligible impact on the total wire capacitance. Three, the resistance and capacitance are linear functions of the total wirelength, i.e., they do not vary due to different distributions of the total routing on separate metal layers. This is due to the fact that the unit RC values for the 1–6 metal layers in the considered commercial technology node are very close to each other. The difference between routing in pseudo-3D and final-3D means that the via count in pseudo-3D cannot be directly used as a proxy for the final-3D via count. But it provides a new point of information for the machine learning algorithm. In later sections, we show that via count indeed has useful information regarding final 3D parasitics by verifying the null hypothesis probability.

From the stages shown in Figure 2.1, the tier partitioning and 3D routing changes layout after the pseudo-3D stage. Within tier partitioning, cell legalization is performed to get a clean placement solution in two tiers. As the location mapping from a larger 2D footprint to an halved 3D footprint in compact-2D creates cell overlaps. Additionally in this work,

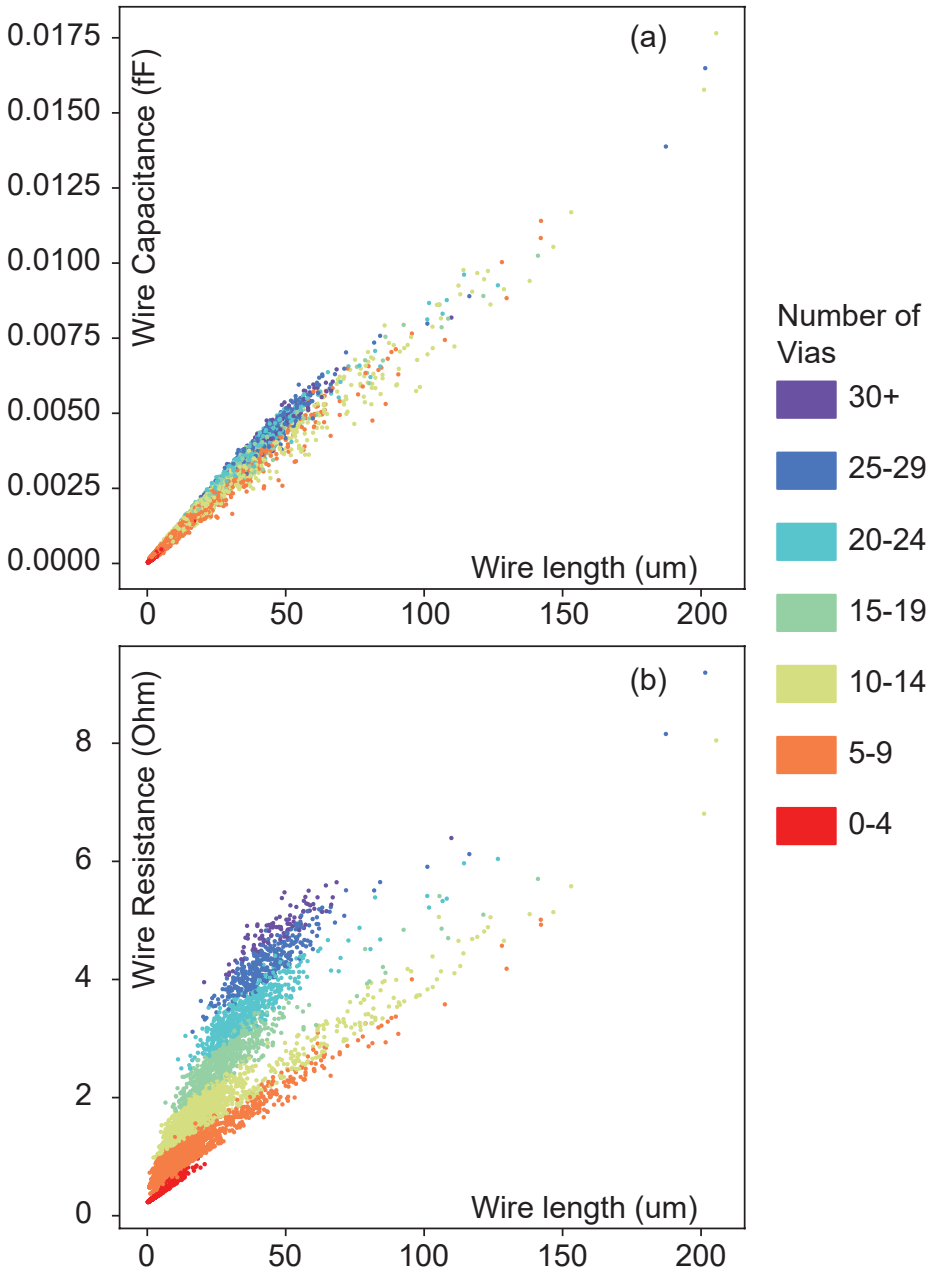


Figure 2.2: (a) Capacitance, (b) Resistance of nets in the AES-128 design w.r.t. routed wirelength of the net. The data points are color coded according to the number of vias on each net.

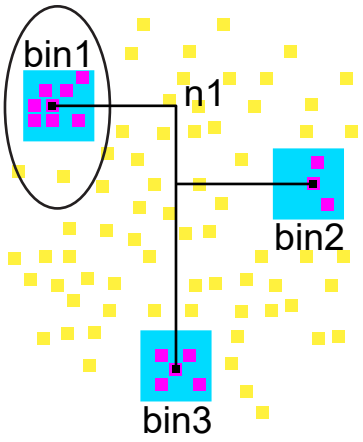


Figure 2.3: Net and its connected cells (smaller squares). The local regions at each end point of the net are shaded in blue (larger squares). bin1 has a high cell density and the contained cell will be displaced during legalization after tier partitioning

instead of legalizing the cells independently in each tier, we perform an incremental placement similar to the proposed solution in [13]. This allows for a better placement quality as die by die legalization does not consider the PPA impact. Based on cell placement and net connectivity, nets undergo different amounts of cell movement during the tier partitioning. Figure 2.3 shows an example of a net and neighboring cells. Features such as the local density in a small neighborhood to the end-points of nets can be used to learn the extent of cell movement each net undergoes.

During 3D routing, the nets are fully modified. Based on the routing, nets can be grouped into three categories as follows:

- **Single-tier nets** are the ones where the 3D routing is done entirely within top or bottom tier. These are expected to undergo the least amount of change from pseudo-3D stage as they are still routed within the 6 signal routing layers.

- **Multi-tier nets** are the nets connecting cells from different tiers after the partitioning. These have the highest difference in routing between the two design stages, as they need to be routed vertically for a proper connectivity. These would have increased wirelength and number of vias that will affect capacitance and the resistance of the nets.

- Finally, the last group are the **nets that use metal-borrowing**. Consider a net that is connecting to cells entirely within the top tier. When performing 3D routing, some of these nets can use metal layers belonging to a different tier. This is called metal layer borrowing, and such nets would have a medium variation in the parasitics in 3D.

To quantify the impact the tier partitioning and 3D routing have on the nets and to understand the extent of this impact from different net features, the following metric is useful:

$$\frac{\text{res(cap) of net(s) in pseudo-3D}}{\text{res(cap) of the net(s) in final-3D}} = \text{Scaling Error of R, C}$$

represented as $SE_{R(C)}$. This shows how well the scaling in pseudo-3D corresponds with the final 3D RCs.

Scaling error can also be defined for a group of nets by using the sum of R, C values in the numerator and denominator of the fraction. $SE_{R(C)} \approx 1$ of a group of nets implies that the scaling done in the pseudo-3D stage is close to accurate for the group.

$SE_{R(C)} \ll 1$ for a group of nets means that the estimation in pseudo-3D is much lower than the final 3D for this group. This results in worse timing after the design is 3D routed. Identifying such group of nets using combination of net is useful in properly applying scaling factor. Most of the nets in a design would fall in this group as the pseudo-3D is usually optimistic.

$SE_{R(C)} \gg 1$ occurs when the parasitic value in pseudo-3D is over-estimated. Cells on these nets would be over-sized in pseudo-3D. These cells not only consume additional but also manifests as an additional capacitance load to the connected cells.

Grouping the nets using net metrics like routed wirelength, fan-out/number of connected cells, number of MIVs, local cell density near the cells, we plot the scaling error variation of these groups in Figure 2.4.

In Figure 2.4(a), the nets are grouped based on the routed wire length. All the nets with

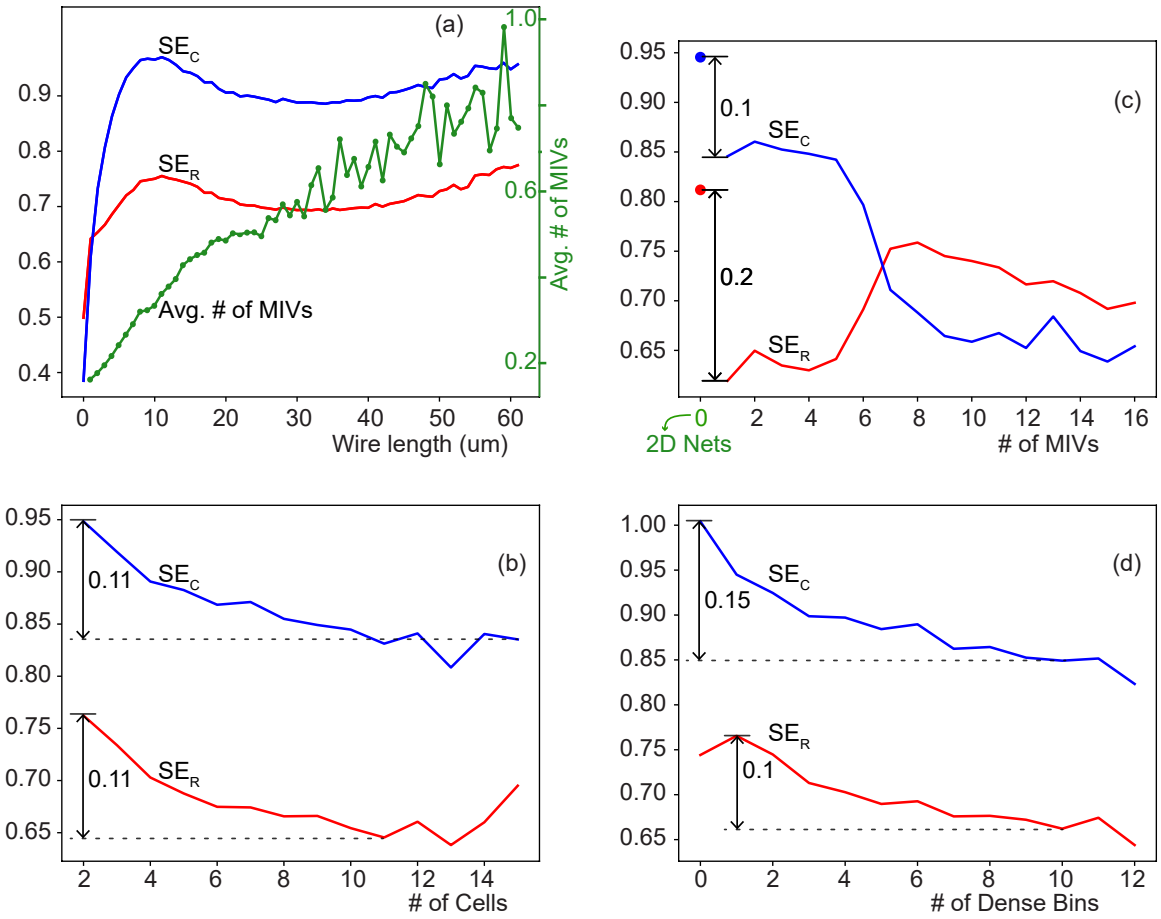


Figure 2.4: Average RC scaling errors w.r.t. various net features. (a) Wire length, (b) Number of MIVs on the net, (c) Number of cells connected to the net, (d) Number of dense bins of the net

wire length $[x, x+1)\mu\text{m}$ are put into group x. Based on this, the average scaling error of R, C, and the MIV counts are plotted. Number of MIVs is dependent on many different features such as cell count, overall connectivity graph, bin size chosen etc. In order to observe the impact of just wirelength, the other features are kept constant for this plot by only considering nets with fan-out 2 in the implementation of aes-128 with fixed bin-size. This shows how wirelength can impact the 3D routing (specifically the average number of MIVs). The average number of MIVs in each group increases as the pseudo-3D wirelength keeps getting higher. Net groups with at least a 1000 nets are considered to reduce volatility in the plot.

More importantly, we observe the scaling error of resistance and capacitance of these groups. This follows a slightly more complex trajectory. The scaling error plots are much smoother by the virtue of central limit theorem as we consider significantly more number of nets. It is interesting to note that none of the groups have an average scaling error > 1 in line with our claim that pseudo-3D under-represents the final RC values. Misrepresenting the via calculation causes the resistance to be significantly under valued. Overall, the scaling error vs. wire length plots have two main trends: a steep increase at lower values $0 \leq x < 10$ followed by a saddle-like shape for $10 < x < 60$.

At $0 \leq x < 10$, the scaling error values are the most ¡1. Since these nets are smaller, small perturbation during legalization and routing changes can cause a relatively significant increase in the final parasitics and so the $SE < 1$. As the nets become relatively large, the net are more likely to be partitioned (as evident from the avg. MIV count plot) and 3D routing is now going to have a higher impact adding more RCs in final 3D that were unaccounted for. This shows us up as the decrease in SE. And as the net length keeps on increasing, the pseudo-3D RCs increase at a rate higher than the impact of 3D routing, so the SE increases again. This interaction between the pseudo-3D RCs and the additional 3D touring manifests as the saddle shape. With a relatively low noise, this allows us to learn a scaling model as a function of routed wirelength.

Extending a similar analysis to the number of cells connected to a net vs. $SE_{R(C)}$ gives us the plot in Figure 2.4(b). These plots are close to monotonically decreasing. This shows that a highly connected net is more likely to have higher discrepancy between the pseudo-3D and final-3D stages. This is a direct result of the bin-based Fiduccia-Mattheyses partitioning done in pseudo-3D flow.

Bin-based Fiduccia-Mattheyses partitioning In this partitioning, placement layout is first divided into smaller rectangular bins and then each bin is partitioned into two tiers such that the area of cells in the two tiers is the same within a tolerance threshold. In any hypergraph partitioning, the nets with high fanout are more likely to be partitioned. For example, a net with $'c'$ cells connected has 2^c ways being split into two partitions. Apart from the two solutions where all the cells in either of the partitions, the other 2^c-2 solutions have a cut-size of 1 net. So, once such a net is forced to not be partitioned, the solution space decreases by a fraction of $\sim 2^c$. But allowing the net to be partitioned leaves the solution space almost unchanged with different configurations achieving same result. So, it is not a good move to keep a highly partitioned net constrained to a single partition as we might miss the chance of finding a better cut-size solution. So, a highly connected cell is more likely to be partitioned under hyper-graph partitioning.

With this knowledge of hyper-graph partitioning, it is easy to see the reason for the monotonically decreasing plots in Figure 2.4(b). As discussed in the wirelength analysis, a partitioned net will have increased parasitics in 3D and since the groups are not directly dependent on wirelength, the numerator (pseudo-3D parasitic value) cannot compensate for the increase in the 3D parasitics unlike in Figure 2.4(a).

The scaling error due to tier partitioning can also be seen by grouping nets based on number of MIVs. This is a final 3D metric and cannot be used in training. But this helps to understand the scaling error evolution in terms of grouping based on final 3D parameters. No MIVs on a net implies that it is fully routed in a single tier, and as discussed earlier,

such nets have the least deviation from ideal. This is seen in Figure 2.4(c) as the $SE_C \approx 0.95$, $SE_R \approx 0.80$ is highest at #MIVs=0. Both SE_R, SE_C have a significant drop when #MIVs=1. This is because nets with fewer MIVs also have a smaller avg. WL Figure 2.4). When the nets are partitioned its adds vertical routing in 3D which is particularly significant for small nets. And as resistance is especially large for vias, this causes the significantly large drop for SE_R.

Finally, number of dense bins per net is analyzed. As discussed in earlier sections, as number of dense bins per net increases, the legalization distorts the net more. When this value is 0, $SE_C = 1$ showing the group of nets that is closest to ideal in terms of pseudo-3D scaling. In this case, each bin at the endpoint is a square centered at the pin with a side of size $3 \times$ Row Height. A bin is considered dense if density is greater than a certain threshold (which is set to 75%). This again shows a monotonic drop with the range almost as large as the trend for cell count.

2.2 Design and Learning Model Implementation

2.2.1 Design Implementation

Training machine learning models requires input data to train and the output labels as the target output of the model. In our case, the input data comes from pseudo-3D design, specifically, the post-route stage of the pseudo-3D design. This is especially chosen as we can extract proper pseudo-3D parasitic values as well as routing metrics like wirelength, via count of nets. We add a few improvements to the 3D stages of the flow using improvements suggested in [13]. This allows for better legalization with full 3D connectivity and 3D congestion driven placement, and the routing is done with complete metal stack. This is useful in our model application as the target parasitic value extraction becomes more streamlined. With an independent die-by-die routing, net extraction should be performed for each die separately. By using the full 3D routing (routing both dies together), net extraction becomes more streamlined leveraging in-built query commands of the commercial

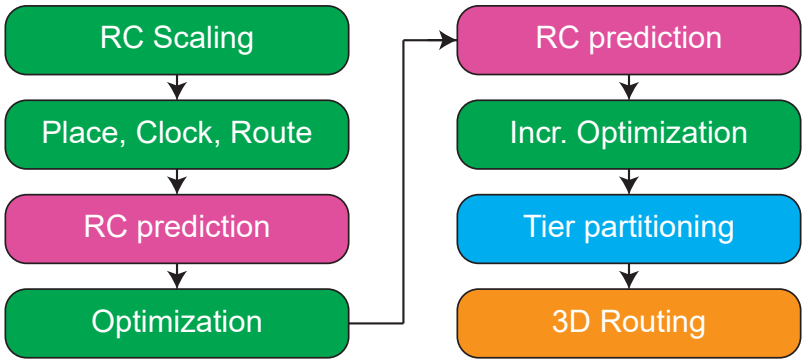


Figure 2.5: Pseudo-3D flow with integrated RC prediction results

EDA tools. So the target data are the parasitic values of the 3D routed nets done with full 3D routing.

The inputs to the model are the net features and design features from pseudo-3D stage. Net features differentiate the nets, where as the design features different between the type of designs. All the different features used are specified in Table 2.2. The evolution of a net from pseudo-3D to final-3D is design dependent and, wire dominant and cell dominant designs have very different routing and therefore very different net evolution between the two stages. Similarly global nets that span over a huge fraction of the chip width or height would have varying lengths as the design varies. Similarly, even when a circuit is fixed, different frequency implementations will add additional timing constraints to routing and additional buffers required for timing closure. And finally, the tier partitioning has an important variable that changes the partitioning type - bin width. The bin width is used to define bins within which FM min-cut is performed. So, a high bin count (small bin width) can increase number of nets partitioned. In order to generalize over all these variations, we chose 8 different training netlists, 4 target frequencies per circuit, 3 bin sizes per frequency totaling 96 implementations for training. Different circuits have different number of nets, and to make sure some large circuits do not overwhelm the training process, the number of nets per design is capped at a certain value. 4 circuits are left out at the training stage that are used during testing.

Figure 2.5 shows the overall flow after integrating the RC prediction. After the rout-

Table 2.2: Input Features used to train the XGBoost model, and their importance and/or explanation

Feature Name	Significance
	Individual Net Features
Wirelength	Long, medium, short wires have varying average $SE_{R(C)}$
HPBB	Wirelength estimation from placement; not affected by congestion
Net connectivity	Number of cells connected by the net; estimates partitioning probability
Via Count	Number of vias on the net; useful for resistance calculation
Wire Cap	capacitance in pseudo-3D design, has strong effect on final capacitance
Wire Res	resistance in pseudo-3D design, has strong effect on final resistance
Average Local density	Avg. density of the bins of a given net as shown in Figure 2.3; quantifies legalization errors
Dense Bins	Number of dense bins of each net; useful for deciding legalization errors
	Full Chip Features (Design Identifiers)
Total WL	Total routed signal wirelength, design identifier
Number of nets	Total number of nets in the design
Number of cells	Total number of cells in the design
Average Fanout	Number of cell pins/ number of nets; design connectivity information
Chip Area	Footprint of the design
Cell Area	Total standard cell area in the design
Utilization	Standard cell density
Bin Count	Number of partitioning bins to be used
Design id	Design name represented as an integer using simple hash function
	Derived Features
net WL/ total WL	Identifying global nets among various designs; detouring, cross coupling capacitance information
HPBB / $\sqrt{\text{Chip area}}$	Identifying global nets among various designs, less affected by pseudo-3D congestion
Wire Cap / Total Cap	Fractional capacitance of net w.r.t. total, importance of net within a design
Wire Res / Total Res	Fractional resistance of net w.r.t. total, importance of net within a design
net WL / Bin size	For a given wirelength, a increase in bin-size would decrease the probability of net being partitioned
HPBB / Bin size	Similar to net WL/ Bin size, but only considers placement information
HPBB + 0.1*VC	A combined HPBB, via count feature

ing stage in pseudo-3D, pre-trained models are loaded and used for RC value predictions. These values are annotated to the existing nets and the design is optimized. But post-route optimization has the ability to change the net connectivity and add/delete nets. Completely constraining the net updates at this stage will impact the optimization quality. So, during optimization, nets that do not have the RC annotations from the trained model would be introduced to the design by the EDA tool for timing closure. To remedy this, an incremental optimization is performed by re-annotating the parasitic values to all the nets in the design followed by an in-place optimization. In this stage the nets are cells are fixed in positions to minimize the changes to placement and routing structure that could invalidate the RC annotation.

2.2.2 Machine Learning Model

To find the best learning model, we first consider a couple of different options during training stage. 1. An XGBoost Regressor, 2. XGBoost Random Forest Regressor, 3. Random Forest Regressor. We chose XGBoost as it is well suited for regression type problems. RC estimation is formulated as a linear regression problem in our work with the least squared sum loss model. Random Forest Regressors are closely related to the XGBoost regressors, and having more weak tree learners in a forest could be helpful to avoid overfitting data to a single design or net type. Within each model, various hyper-parameters are varied and a 2-fold cross validation is used to choose the best combination. Specifically, they are the number of trees, number of features per tree, and max depth of each tree. In general it is better to have many weak trees to avoid overfitting. Similarly by randomly choosing a subset of all the features in each tree, the model can be generalized better and performs well with test sets. Moreover before training any model, we perform feature selection on the 24 selected features. By removing the unnecessary features, over fitting in tree learning models can be avoided. This is done using backward feature selection and null hypothesis test using a linear regression.

In backward feature selection, a linear regression model is first fitted based on an initial set of features. p-values are extracted for the features and at each stage the worst features is discarded if its p-value > 0.05. p-value shows the probability of null hypothesis i.e., the probability that a feature doesn't contribute to the target. In capacitance model training, the 'design id' was the only feature that had a p-value greater than 0.05 and is removed during model learning. In resistance model, the wire resistance was the only feature with p-value > 0.05. This is an extremely un-intuitive outcome and happens because of the influence of vias made the wire resistance less useful and redundant. Changing the design significantly changes the routing patterns (metal usage per layer, vias used etc), which has more impact on resistance model. So 'design id' has a smaller p value and is kept in resistance modelling.

2.3 Results

2.3.1 Training and Inference

With the data collected from the previous stages, a 2-fold cross-validation is used on the testing sets and the model with best cross-validation error is selected. The R2 score (explained variance) of the designs is shown in the Table 2.3. Of note are the designs with low R2 score. When R2 score equal 1, the model predicts all the variations in the target label. R2 score is zero when output is a constant equal to the target mean. The R2 score is higher in both training and testing netlists, although the increase per netlist varies. Netlists like vga, ldpc, netcard already have a relatively high R2 score in Compact-2D (C2D) stage. Interesting to note is that these three are the wire dominant designs in the netlists considered. So, the slight variances due to 3D routing is not very significant relative to the overall wirelength. MSE is also high for these circuits due to their large parasitics.

In Table 2.4, the permutation importance is given for the top 10 most important features in the model on the test set. In permutation importance, a loss metric is first calculated using a trained model, and then a feature is permuted so that values of the feature are incorrect and

Table 2.3: R2 Scores and Mean Squared Error of Compact-2D (C2D) scaling and the best model for capacitance training

	C2D - R2	ML - R2	C2D - MSE	ML - MSE
cordic	0.8284	0.9261	3.00e-7	1.30e-7
des	0.8480	0.9324	6.40e-7	2.80e-7
edit-dist	0.9247	0.9578	1.16e-6	6.50e-7
fpu	0.9289	0.9729	1.38e-6	5.30e-7
ldpc	0.9544	0.9714	3.55e-6	2.22e-6
leon3mp	0.9360	0.9697	3.41e-6	1.62e-6
matrix-mult	0.9208	0.9376	1.17e-6	0.92e-6
netcard	0.9810	0.9912	2.85e-6	1.33e-6
Test Set				
b19	0.9072	0.9704	5.40e-7	1.70e-7
ecg	0.9305	0.9719	6.90e-7	2.80e-7
tate	0.8889	0.9517	1.56e-6	0.68e-6
vga	0.9633	0.9717	3.93e-6	3.03e-6

the score is re-calculated. The difference between the scores is the permutation importance. A large value implies a feature high importance and a small value implies a low importance.

Figure 2.6 shows the RC histograms of three design stages: Model predicted RCs, Pseudo-3D RCs, True 3D RCs. We see that the pseudo-3D RCs deviate more from the ground truth in the lower RC ranges leading to more smaller RCs than final 3D. Such a design would not meet timing when RCs become worse in the actual 3D stage.

2.3.2 Full-chip PPA

Irrespective of RC prediction, PPA is the most important consideration for a full chip study, and in Table 2.5 we report the PPA of the four testing netlists used: b19, ecg, tate, vga. b19, vga are relatively small circuits and ecg, tate are significantly larger. Three different implementation PPA are reported: 3D design using circuit agnostic machine learning model, circuit specific model, and the compact-2D's global RC scaling model.

The capacitance error in each implementation is the difference between the total capacitance of its final 3D and pseudo-3D stages. $-ve$ capacitance error means that the pseudo-

Table 2.4: Permutation importance using RMSE loss of the 8 most important features per model. The RMSE loss of resistance model is $7.31\times10^{-2}\,\Omega$, and capacitance model loss is 9.22×10^{-4} fF

Resistance Model		Capacitance Model	
Feature	Importance	Feature	Importance
Via Count	3.08e-1	Wire Cap	4.48e-3
Wire Cap	1.34e-1	Wire Res	4.55e-4
Pin Count	4.16e-2	Wire Length	4.46e-4
Wire Length	3.63e-2	Via Count	6.98e-5
hpbbXwidth	4.47e-3	Pin Count	4.10e-5
hpbbXvc	4.45e-3	hpbbXwidth	9.27e-6
Dense Bins	3.94e-3	viaXhpbb	3.30e-6
Res/Total Res	3.50e-4	Chip area	2.46e-6
# nets	2.66e-4	hpbb	1.82e-6

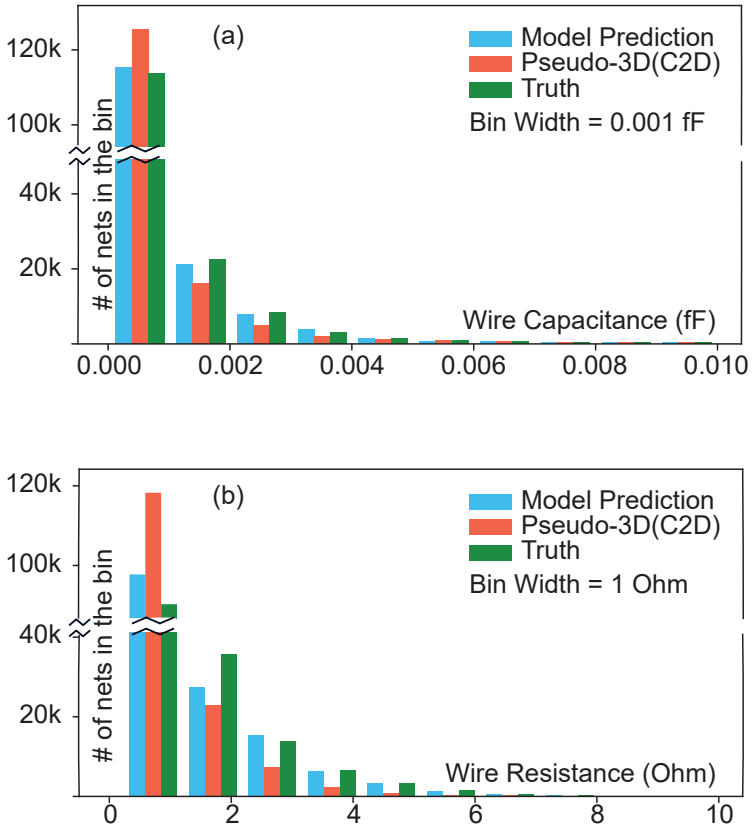


Figure 2.6: RC Histograms of ML based implementation

Table 2.5: Overall PPA of the test netlists with Circuit Agnostic ML Scaling, Compact-2D Scaling, Circuit Specific ML Scaling Models

		units	b19	ecg	tate	vga
Common Parameters	Frequency	MHz	1250	1500	1000	1250
	Chip Area	μm^2	29446.4	75130.8	116690.2	45624.6
	Cell Count	μm^2	37976	95768	155521	35613
	Cell Area	μm^2	42756.0	107115.5	233599.8	66400.9
	WL	m	0.316	0.851	1.401	1.260
Circuit Agnostic	Cap Error	%	1.22	9.77	2.78	5.36
	Total Power	mW	39.3	107.0	135.0	28.3
	WNS	ns	-0.091	-0.401	-0.441	-0.176
	TNS	ns	-13.1	-67.5	-58.5	-3.1
Compact-2D	Cap Error	%	-6.59	-3.05	-10.23	-3.84
	Total Power	mW	39.8	104.7	131.4	28.4
	WNS	ns	-0.109	-0.282	-0.764	-0.193
	TNS	ns	-80.6	-197.8	-356.1	-48.5
Circuit Specific	Cap Error	%	-0.70	5.96	1.87	2.77
	Total Power	mW	38.9	106.9	135.0	28.1
	WNS	ns	-0.068	-0.409	-1.108	-0.150
	TNS	ns	-15.6	-100.9	-84.5	-10.5

3D stage predicts a low capacitance value. Compact-2D flow always under estimates the value leading to worse slack overall. With machine learning models, the error is always positive and slightly in the positive direction. For circuits with large over or underestimates, the power consumption varies based on the method chosen (although this is a very small % of total power). For the small circuits such as b19, vga, the power consumption is actually smaller as the machine learning model is employed. This is because the accurate parasitic estimation allow for a better optimization in the pseudo-3D stage.

Total negative slack has the highest impact overall as it becomes $\approx 3 \times -16\times$ smaller with the circuit agnostic model compared to the compact-2D scaling. This is mainly because each net is assigned a better parasitic which contributes a little to the overall slack improvement. Worst negative Slack on the other hand depends on just the critical nets in 3D. These nets are not identifiable from compact-2D stages, as the 3D routing changes the criticality of the paths.The nets with negative timing slack in pseudo-3D can have a positive slack, and nets with positive slacks in pseudo-3D can become the new critical paths in final 3D. To identify the critical paths, a learning model that is tailored for timing estimation of graph networks should be used and requires a high accuracy. But the best solution would be to add a post-route optimization stage in 3D that can fix the relatively minor TNS and few WNS violations in 3D.

Finally, we also trained circuit specific models to check the benefit of having such models. While these models were able to perform better in terms of total capacitance error, they do not have significant improvement in power consumption. With regards to timing, the circuit specific models have even more volatile WNS. The TNS is better than the compact-2D scaling but is not as good as using a general model. This is slightly counter-intuitive as we would assume circuit specific model would perform better. But they suffer from over fitting and lack of different types of nets during training. Another reason is that these models are trained to learn the parasitics and cannot be directly helpful with overall timing. So the general learning model is as good as circuit specific models for parasitic

estimation and power consumption, but are significantly better in terms of timing.

2.4 Conclusion

In summary, we have presented a machine learning model that can predict final net parasitics at an early stage of the design. We have analysed several net features and how they impact the parasitic evolution in a pseudo-3D flow. We formulate new metrics and use them to achieve better circuit agnostic learning models. Using these models, we were able to achieve higher R2 score, lower MSE, better timing. We discussed the issue of critical path estimation in 3D design and showed that our general model is better than a circuit specific model. With $3\times - 16\times$ TNS reduction on test circuits, integrating these models in the pseudo-3D flows help us to minimize number of timing violations and the severity of the violations after routing.

CHAPTER 3

PIN-3D: AN EFFECTIVE MULTI-DIE CO-OPTIMIZATION METHODOLOGY FOR 3D IC DESIGN

In this chapter, we propose a well-rounded flow for creating a sign-off quality 3D IC of any partitioning or 3D pitch type using commercial PnR tools. With the help of commercial Process Design Kit (PDK), and test circuits that include two Arm Cortex processors, we show that the 3D IC designs with Pin-3D have a 9-30% power reduction along with a 24-38% wirelength reduction compared to 2D ICs. Pin-3D is the first flow to support heterogeneous Monolithic 3D IC designs (w.r.t to process nodes) for a more general partitioning type. Using Pin-3D, we successfully design a 128-bit AES encoder circuit that consists of cells from two different technology nodes: 45 nm node on bottom-die, and 15 nm node on the top. The Pin-3D methodology achieves a sign-off quality timing closure on even such heterogeneous designs. This opens up some an interesting dimension for circuit design and optimization as there can multiple process nodes on each die of 3D IC.

3.1 Background

A summary of the differences between Pin-3D and the previous Pseudo-3D flows have been provided in Table 3.1.

Shrunk-2D [17] is one of the first Pseudo-3D flows to use commercial PnR tools for a 2-tiered 3D IC design. The key idea here was to use same footprint as the 3D IC to place and route the design. With the total silicon area being fairly similar between 2D and 3D ICs, the actual footprint of a 2-tier 3D design is half the 2D IC footprint. Shrunk-2D also uses scaled Front and Back End of the Line (FEOL, BEOL) to fit all the cells and routing of the larger 2D footprint in half the area. This is the 'Pseudo-3D stage' of Shrunk-2D. Commercial tool flows can be applied at this stage without any limitations.

Table 3.1: Qualitative comparison among state-of-the-art "Pseudo-3D" physical design tools for monolithic 3D ICs and this work. "enhanced die-by-die" means the pins from both dies are visible during die-by-die optimization on a complete 3D metal stack.

	Shrunk-2D [17]	Compact-2D [24] w/o 3D Optimization	Compact-2D [24] with 3D Optimization	Pin-3D (this work)
Key idea	cell and wire shrinking	placement compaction	row halving	pin projection
3D stack	two separate dies	two separate dies	double metal stack	double metal stack
Strength	first Pseudo-3D flow	shrinking unnecessary	3D optimization	more holistic 3D flow
Weakness	shrinking causes DRC issues	under buffering/sizing	DRC due to row halving	–
Placement	2D + tier partitioning	2D + tier partitioning	2D + tier partitioning	2D + tier partitioning
Legalization	die-by-die	die-by-die	die-by-die	enhanced die-by-die
Signal Routing	die-by-die	die-by-die	true 3D (ignoring DRC)	true 3D
Clock Tree Design	2D + tier partitioning	2D + tier partitioning	2D + tier partitioning	optimized for 3D
Power/Ground Routing	manual	manual	manual	manual
Post Route Optimization	not supported	not supported	both dies at once	enhanced die-by-die
Engineering Change Order	not supported	not supported	not supported	supported
Heterogeneous 3D ICs	not supported	not supported	not supported	supported

To generate the final 3D GDS, the Pseudo-3D stage is then transformed into the two tiered 3D IC by partitioning using bin-based Fiduccia-Mattheyses min-cut algorithm [15]. The cells are then scaled up to their original sizes and are legalized to remove any overlaps during the transformation process. Monolithic Inter-Tier Vias (MIVs) that connect the nets crossing the two tiers are placed by performing routing of the 3D nets on the complete 3D stack. A die-by-die routing stage completes connectivity within the two tiers, and a final optimization is done in each die separately. The independent die optimization doesn't rectify timing from the other dies and does not create a fully optimized 3D IC.

In Compact-2D [12], a new Pseudo-3D stage was proposed in order to avoid the dimension scaling as it can lead to issues such as design rule violations, pin access issues, RC inaccuracies, software license limitations. Here, instead of scaling the footprint, it is kept the same as 2D, in turn, the unit parasitics are scaled by $1/\sqrt{2}$ as a way of replicating the smaller wirelengths and RCs of a 3D design. While this is a better Pseudo-3D stage than Shrunk-2D, it still has many of the same problems with regards to 3D timing closure.

An additional post-partitioning optimization has also been proposed for Compact-2D to close timing in 3D. To do this, the authors of this flow use a complete metal stack spanning both the 3D tiers for the complete 3D routing. The pins of the top-tier cells are maintained on their corresponding layer in the 3D metal stack. In addition, by halving the cell height and introducing two non overlapping half height row types for the cells from top and bottom dies, overlaps are removed between cells belonging to different tiers. These tweaks allow for timing optimization in 3D. While this solves most timing issues, the flow is still incomplete as there is no placement or clock tree optimization stage tailored for the 3D stack. In addition, the half height cells couldn't fully represent the design rules, and some pin access violations still remain due to the cell halving.

With Pin-3D, we do not change the physical or electrical properties of the wires or the cells at any stage of the flow allowing for a more streamlined methodology. Moreover, we also unlock incremental placement, and clock optimization in addition to the routing and

timing optimization stages for 3D ICs. Instead of halving and projecting the cells onto a single tier, we perform die-by-die type placement and buffer insertion operations, but crucially, with the entire timing and physical context of the 3D design. section 3.2 and section 3.3 provides detailed information about how this is achieved. By keeping the physical representation of the cells intact, we make Pin-3D flow generalizable to heterogeneous 3D IC designs with different process nodes on each tier. Such design configuration is not possible with any of the previous Pseudo-3D flows.

3.2 Pin-3D Flow Enablement

3.2.1 Key Idea

To enable commercial 2D PnR tools for 3D optimization, we should include the cells from both tiers in a single FEOL layer without causing unnecessary overlaps. This is because the commercial 2D PnR tools available only support a single FEOL layer for cell placement. We bypass this problem, by only keeping one die 'active' at a time, and the other die is turned into a *Transparent Die* by converting all the cells on this die to COVER cells. In current commercial tools, cover cells are a type of cells have no active area used to represent some dummy cells or feed-through cells without any logic. Using them for transparent die, would result in cells that have no placement obstructions or overlaps, letting the cells from both tiers be present at a single (x, y) location without causing placement overlaps between the two.

The entire 3D stack from both dies is also required for accurate 3D design as this would allow for accurate 3D routing in both tiers. Another crucial part of routing is the access to the standard cell pin shapes, and the representation of these shapes. The pin shapes need to be present on the correct layers in 3D so that routing can be done accurately with the double metal stack. This *Pin Projection* is enabled by hacking the BEOL and FEOL tech files for 3D. Pin Projection means that the pins of the cells on top tier are projected to the top metal layer rather than placing them right above the cells as is the standard for a normal

cell design.

Enabling commercial tools for 3D IC

As any commercial PDK only contains design files for a 2D IC design, it is necessary to make required files that can represent a 3D IC design. For the 3D BEOL, we need LEF (Layout Exchange Format) files defining all the routing rules for the metal layers, an ICT (Interconnect Technology) file with metal layer parasitic information and the complete metal stack information such as the dielectric medium present and the thickness, heights of each layer in the BEOL.

To represent the 3D FEOL, we require macro LEF files containing physical information for the standard cells and any memory macros used in the design. For the timing and power properties of the cells, a liberty format file is required with the spice characterizations of cells under various input and output scenarios. In order to represent the entire 3D design within a commercial tool, it is necessary to load all of the above technology files along with design specific files. These are the netlist for logical connectivity, constraints file for clock related information, and other optional files such as design exchange format (DEF) for physical information of the circuit. The design specific files are also similar for a 2D design with only a few modifications required.

3.2.2 Creating the 3D BEOL and FEOL files

3D BEOL Creation and Pin Projection

A 2-tier 3D IC with 6 metal layers per die, is the configuration of 3D IC considered in our work. Here, the metal layers would be M1_bottom, M2_bottom, ... M6_bottom, M1_top, M2_top, ... M6_top in order from bottom to top. In the newly created LEF file, the routing rules for each Mx_bottom and Mx_top are assumed to be same as the Mx layer from 2D LEF file provided by foundry. This makes sure the generated 3D LEF also contains all of the routing rules required by the foundry for a 2D design. A via layer is required between

every two consecutive metal layers and the via layers in the 3D LEF also follow the same methodology we used for metals, but with one notable exception. There is no parallel for the via between M6_bottom, M1_top in the 2D PDK. This is the Monolithic Inter-tier Via (MIV) that connects the two tiers together in a 3D IC design. This layer is newly created in the 3D LEF with only the base rules required for routing such as width and spacing. The ICT file with layer parasitic information and the metal stack structure is similarly extended to 3D using the corresponding 2D file.

3D FEOL Creation and Transparent Cells

To accurately represent the cells from multiple tiers together within PnR tools, it is required differentiate the cell names, and also to have pin shapes on correct layers. To do this, we first simply duplicate the 2D macro LEF files into two identical ones for top and bottom die. The bottom and top die cells are renamed by adding custom suffixes for each die. The top die cells are further modified so that the pin shape layers are shifted along the 3D stack corresponding to the top tier metal naming conventions. Since an MIV has to pass through top tier FEOL to access the M1_top, they cannot be located at the same location of cells in the top-tier. A 3D specific routing rule for each top tier cell is added by creating MIV layer obstructions the same size of the cell. This ensures every instantiation of the cell in the 3D design carries along an obstruction that restricts the tool from adding MIVs overlapping with top tier cells.

Transparent Cells A major limitation of the commercial PnR tools for 3D placement and timing optimization is that they can only have a single FEOL layer. Compact-2D uses the half-height row cells in order to avoid overlaps between cells from different tiers during 3D routing and optimization. But as discussed, this creates pin access issues and is not generalizable for placement, clock tree optimization, or heterogeneous 3D ICs. In Pin-3D, instead of cell halving we convert cells into transparent cells that do not have any active

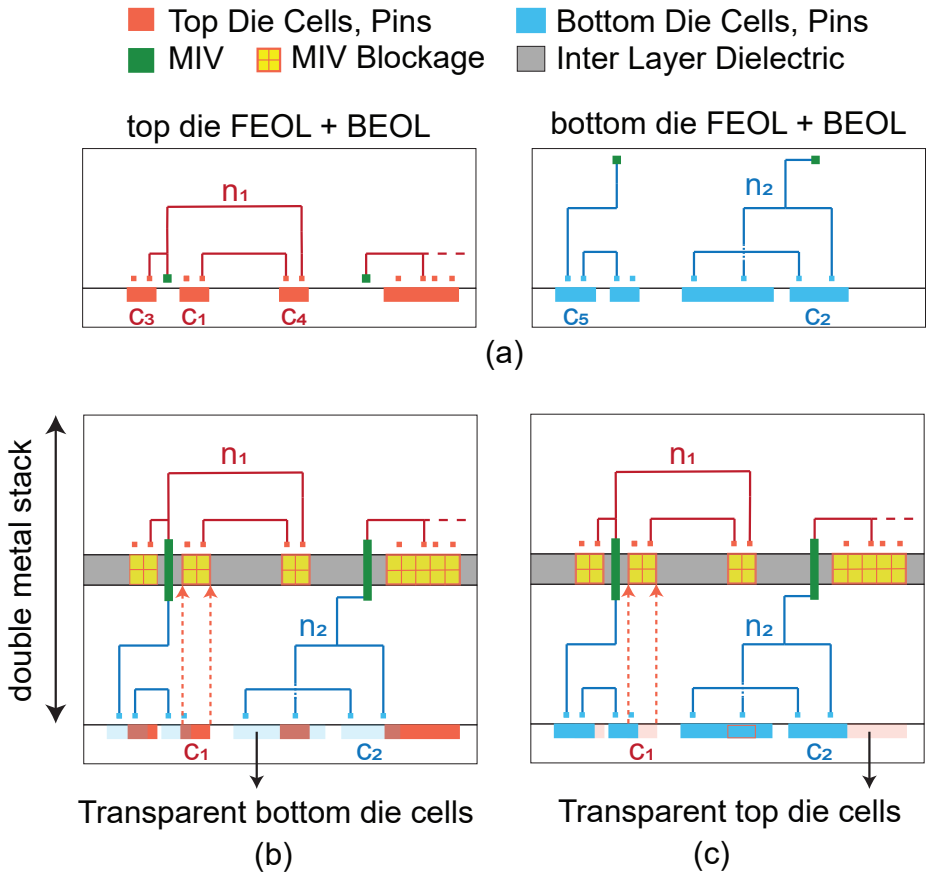


Figure 3.1: The key idea of Pin-3D: die merging and pin projection. (a) top and the bottom dies separately, (b) merged dies for the top die optimization, (c) merged dies for the bottom die optimization. Our double metal stack contains pins from both dies to provide the entire 3D context during die-by-die legalization, routing, and timing closure. Top die cells are also projected to the MIV layer to ensure no overlap between MIV and routed nets. Moreover, Pin-3D allows design with two different technology nodes as demonstrated in subsection 3.6.3

area. By selectively modelling the cells of either top or bottom die as transparent, we successfully remove the overlaps from the cells of transparent dies. The non-transparent die can undergo all of the cell sizing, insertion, deletion operations allowing for full suite of commercial capabilities. We introduce two flavors of LEF files to be used during different stages of the flow. Figure 3.1 (b) and (c) shows the two cases where the cells from different tiers are turned transparent. The bottom die cells are turned transparent during the top die optimization stages and vice-versa.

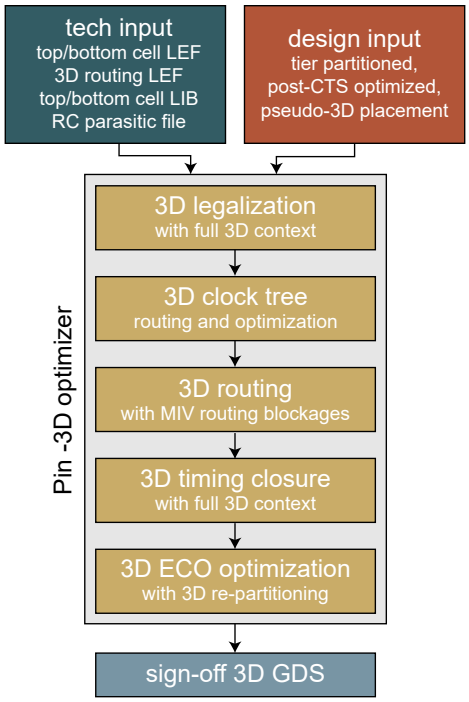


Figure 3.2: Our Pin-3D optimizer design flow.

3.3 Pin-3D Design Flow

The overall flow of the Pin-3D optimizer is shown in Figure 3.2. The Pin-3D optimizer is designed to be added on to any pseudo-3D design or any partitioning or technology formats. The technology files required to represent the 3D design are discussed above, and the first stage in the flow is cell legalization / incremental placement.

3.3.1 Incremental Placement with Global Routing

Since the pseudo-3D stage does not have any partitioning information, the placement is oblivious to 3D intricacies such as the constraint between top-die cells and MIVs, or the different layers of the cells and pins in a 3D design. During incremental placement in Pin-3D, cells are displaced to remove any overlaps as well as to optimize for the 3D connectivity, and partitioning. At the same time, the top die cells have an added soft padding of one SITE width on the left and right ends. This generates a more porous top tier cell placement that allows easier access of MIVs. The top-die standard cell placement in Fig-

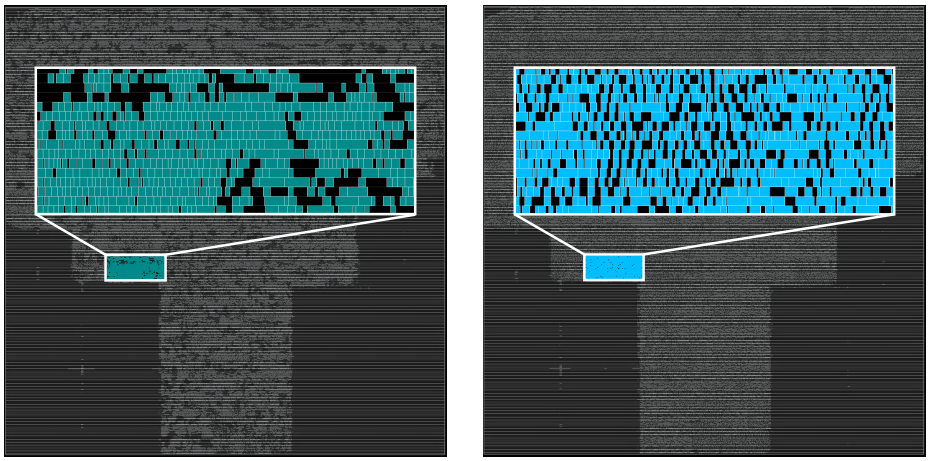


Figure 3.3: Standard cell placement of Cortex-A7 and zoom-in at a specific location using (a) Compact-2D legalization; (b) Pin-3D legalization. Dense cell clusters is bad for M3D routing. Tier-partitioning and pre-legalized cell placement is the same between the two.

ure 3.3 clearly shows the difference between legalization results of Pin-3D flow compared to C2D. Placement is still a die-by-die procedure in Pin-3D but with the full 3D context (timing, connectivity) considered at each step. This is due to the *transparent die* whose standard cells are turned into COVER cells and cannot be used for legal placement.

Pin-3D's design flow also includes global routing stage during the legalization so that the placement and routing are interdependent to create better congestion-driven and/or timing-driven placement and routing.

3.3.2 Clock Tree Optimization

Clock Tree is a crucial part of the 3D IC design, and none of the previous flows have considered optimizing the entire clock tree for 3D IC design. There have been a few studies that cluster or partition the clock buffers or flip-flops to minimize detrimental skewing between the flip-flops in 3D, but these are just heuristics and only have a limited effect. Pin-3D also enables clock tree synthesis and optimization for 3D ICs. Therefore, instead of manual tuning and controls with clustering techniques, we can let the tool optimize the setup and hold paths for the entire design.

Table 3.2: Worst and Total Negative Slack Trend in Pin-3D, and the effect of the clock optimization stage for Cortex-A7. All slacks are normalized w.r.t the clock period

	Worst Negative Slack		Total Negative Slack	
Design Stage	No ccopt	With ccopt	No ccopt	With ccopt
Place	-0.753	-0.753	-1799	-1799
Clock	–	-0.029	–	-7
Route	-0.741	-0.039	-2157	-82
Timing opt1	-0.052	-0.020	-103	-4
Timing opt2	-0.037	-0.008	-41	-1
Timing opt3	-0.032	-0.004	-24	-0

In our flow, clock tree synthesis and optimization takes place on the top-die after placement legalization. To generate a high quality clock tree in 3D, we identify and move all the clock tree buffers and inverters to the top-die before performing clock optimization. This allows the tool to fully optimize the clock tree network in a single step, rather than in a die-by-die fashion that can hurt the overall clock tree.

As discussed previously in subsection 3.3.1, legalization is a two step process and after the first step of legalization, the clock cells from bottom-tier are identified and re-partitioned to the top-tier using the ECO capabilities that are enabled with Pin-3D. ECO allows incremental global routing and placement legalization to account for the re-partitioned clock tree cells. Note that we do not change the partitioning solution of the sequential cells. In a typical clock-tree network, the number of buffers is much smaller compared to the sequential cells at the leaf nodes of clock tree. So simply changing the clock combinational cells does not significantly alter initial partitioning solution. Proper clock skew assignment with clock optimization is important and can drastically reduce the total negative slack of the entire design as the clock signal reaches every launching and capturing flip-flops.

The resulting clock tree from this methodology shows a very beneficial impact on the timing closure. This is shown in Table 3.2 where the design flow with clock optimization (ccopt) stage added to 3D design has achieved effective timing closure by the end of the flow. The place stage in Table 3.2 is combination of both bottom and top die legalization as well as the global routing, after which the timing information is extracted. We see that

the clock tree stage has effectively removed most of the negative slack after the global route stage. By accurately modelling 3D design data, the clock tree optimizer was able to generate useful skews from timing positive paths to timing negative paths. Clock Tree optimization is only done in a single die as breaking up the clock tree into different stages is not a feasible method with Pin-3D flow. Unlike placement or routing, we observed that die-by-die clock tree optimization had a detrimental effect on the previously generated trees.

3.3.3 Routing

Routing of the 3D IC is a one-step process unlike the placement, as the transparent cell method that mandates die-by-die placement does not affect the placement. Routing is simply done to lay down physical wires connecting the entire 3D design. Since the transparent COVER cells do not have any timing modifications, the router can easily consider their the delay and loading effects when routing both the standard cells and the transparent cells in a single go. We see that from Table 3.2, the worst and total negative slacks both degrade slightly after the routing (detail routing) step. This is because of the small inaccuracies in timing estimation between global and detail routing stages. This is not specific to 3D IC, and is an artefact present in any design flow with modern PnR tools due to their split global and detail routing stages.

3.3.4 Timing Closure

Timing closure of any design is crucial for achieving a sign-off quality commercial design. With Pin-3D, as the cell insertion can only occur in one die at a time, the timing closure is done in 3 distinct steps as hinted in Table 3.2.

After routing the complete 3D design in the previous stage, the design flow now continues to optimize the timing of the paths by only inserting cells in the top die. During this process, the tool can still modify the routing of the nets connected to the cells on bottom

die. As a result, a timing limited net on the bottom die can be split up by adding a buffer on the top die and resolving some of the timing issues. This will not yet be optimal as the bottom die cells are left untouched. A second round of optimization stage then concerns with the cells resizing and modifications on the bottom die. This would then solve most of the timing issues present in the design.

But a third round of optimization is run, this time on the top die, to finish the timing closure stage of Pin-3D. As the top die cells had to overcome the non-optimized nature of bottom die cells in the first iterations, there might be few aggressively scaled cells during the initial iteration of the design. Therefore, we run this third optimization stage on the top die, after optimizing the bottom die with 3D context to reclaim any leakage, area by resolving such aggressively scaled cells.

We see that both the worst and total slack improve with number of optimization stages. But the marginal improvement we get from more iterations comes at a significant run-time cost. As seen in Table 3.2, the timing is mostly resolved at the end of the 2nd timing iteration, rendering the final stage moot. This is not the case for the design without ccopt, whose worst slack has started to plateau at around -0.032 ns, but the total slack is still recovering albeit slowly.

3.3.5 ECO

Finally, ECO is an important stage of the commercial design flows, which is used to manually adjust the path timing, leakage, or other violations that might need to be addressed using custom scripts. In 3D, ECO is also necessary to change the tier allocation of cells in the post-route stage along with traditional 2D functions such as add/modify cells within a tier. Pin-3D's technology and design setup allows all of the aforementioned moves and is shown using a simple flip-flop sizing algorithm.

As the cells of the two tiers are differentiated based on names, we can move an inverter in the bottom tier to the top tier by replacing the replacing it with the top tier cell using the

Algorithm 1: ECO Technique

```
criticalRegs ⟵ launching registers of register-to-register paths with slack < 0.0;
nonCriticalRegs ⟵ launching registers of register-to-register paths with slack >
  150 ps;
foreach reg in criticalRegs do
    if reg is not maximum drive strength then
        Up-size the register;
    else
        Use the corresponding lowest V_th register;
    end
end
foreach reg in nonCriticalRegs do
    if reg is not minimum drive strength then
        Down-size the register;
    else
        Replace with corresponding low-power register;
    end
end
Perform incremental placement and routing
```

'eco' commands of PnR tools. Since the MIV blockage is defined within the cell definition, replacing the cell type from bottom to top would automatically create required blockages for routing. Similarly as the pin projection is also done within the cell definition, they also appear on the correct tier after the eco change. The incremental eco routing provided within the PnR tools can then route to the newly moved cell with accurate routing restrictions.

Here, we first note that the clock-to-output delay of the flip-flops contribute to more than 10% of the total path delays in our 3D designs. This is significant when considering that the critical paths are in some cases have a logic cell depth of ≈40. To address this, we perform ECO using algorithm 1.

By up-sizing the registers on the critical paths we can improve overall timing. Simultaneously, by down-sizing the registers on paths with large negative slacks, the overall power consumption can be kept in control. We identify these two sets of flip-flops based on the path slacks. A 150 ps threshold is used for the positive timing path groups as the path delay would degrade when replacing it with a smaller drive-strength register.

The critical registers are swapped with register of same type with higher drive-strength. In case, the register under consideration is of the highest drive-strength, we replace it with the same register, but from the lowest threshold voltage type. The same process is done with the non critical register but in the opposite direction (replacing with lower drive-strength cells, or with highest threshold voltage type).

3.4 Experimental Setup

3.4.1 Homogeneous 3D ICs

The performance and efficiency of the Pin-3D design flow is compared with the Compact-2D flow. Note that we use the version of C2D without the post-tier partitioning as we do not have access to a working version of the scripts. We also add the 2D designs for reference so that we can see the complete picture of 2D vs. 3D, as well as C2D vs. Pin3D. All the designs are implemented with a commercial 28 nm PDK, with the 3D technology files generated as specified in section 3.2.

Two application processors: Cortex-A7 and Cortex-A53, are used as test circuits that validate our flows with commercial designs. The results for these circuits are normalized w.r.t corresponding 2D designs as per our NDA with Arm. Both of the Arm processors are configured with 1 core with 32 kB L1 Instruction Cache, 32 kB L1 Data Cache, Floating Point Unit, and Arm's NEON SIMD unit. Both the netlists are dominated by logic cell area rather than memory macro area.

Two open-source pure-logic test circuits: LDPC and netcard, are also used for analyses. These help us to show the raw design metrics of different design implementations for an in-depth analysis, and to add variety to the test-bench.

3.4.2 Heterogeneous 3D ICs

As we specified, Pin-3D flow enables the design and optimization of heterogeneous 3D IC designs. A proof-of-concept of the design is shown using the open-source nangate 45 nm

Table 3.3: Pin-3D vs. Compact-2D [24] on different aspects of the 3D design. We use Cortex-A7 in 28 nm.

Cortex-A7		C2D [24]	Pin-3D
Legalization Displacements	Top-Die Avg.	1	0.42
	Bottom-Die Avg.	1	0.40
Routing Metrics	Total Wirelength	1	0.910
	MIV Count	50,474	104,219

and nangate 15 nm PDKs. While, the 3D technology file generation is same as the process described in section 3.2 for homogeneous files, there are a few other restrictions that constrain the design process. This is discussed in detail in subsection 3.6.3.

3.5 PPA benefits of the Pin-3D stages

Placement Legalization for 3D

From Table 3.3, we see that the legalization using Pin-3D manages to achieve 50-60% lower average displacements compared to the Compact-2D (C2D) flow. This is because the legalization in each die is guided by the full 3D placement. With the incremental placement of Pin-3D, the tool not only removes cell overlaps but also improves for timing, congestion, and other design metrics using global routing. With die-by-die legalization of C2D, once the dies are separated after partitioning, the displacements in one die doesn't impact timing or connectivity of the other die. The displacements resulting from the legalization stage contribute to the mismatch of placement information between the pseudo-3D and the final M3D stages. For example, a high displacement of even a single cell on the critical path will increase the wirelength and wire load connected to the cell leading to a worse delay and slack.

3D Clock Tree Optimization

The run-time and timing closure benefits of clock tree optimization were already discussed in subsection 3.3.2. Here we analyze different clock tree metrics and the impact of fixing all the clock buffers and clock logic on the top die. Figure 3.4 shows the two different scenarios during the clock tree optimization stage. As mentioned in subsection 3.3.2, during the clock optimization stage, we fix the clock combinational cells to the top-die as depicted in Figure 3.4(b). Clock optimization stage which occurs during top-die optimization during which only top-die cells are changeable by the tool, and bottom-die cells are all converted to transparent cover cell type that cannot be modified. The benefit in Figure 3.4(b) is that the tool can modify the entire buffer network. This allows for a much easier skewing of the clock paths to the sequential cells. Note that the tier assignment of sequential cells is not important for clock skewing. Since routing can be done to any top or bottom die cells without any issues, the skews can be accurately controlled.

With the clock tree as shown in Figure 3.4(a) skew between pairs of sequential cells that are driven by bottom-die buffer cannot even be adjusted directly. To do this, the tool breaks the nets and adds a lot of new clock buffers so that the skews can be controlled. Because of this we see that in Table 3.4, the clock tree obtained from without fixing the clock buffers is much larger for the same netlist. The total number of cells is more than 50% of the clock tree with buffer fixing. This in-turn creates the larger wirelength, worse latency and max skew of this clock tree. As clock is the one of most active signals in any design with highest toggle rate, minimizing the clock network area is important to reduce overall power.

3D Routing

The global routing aware placement stage in Pin-3D allows for a better routing solution. Table 3.7 show that these improvements lead to smaller routed wirelength by up to $\sim$8% with Pin-3D compared to the C2D design, and around 25% smaller wirelength than the 2D

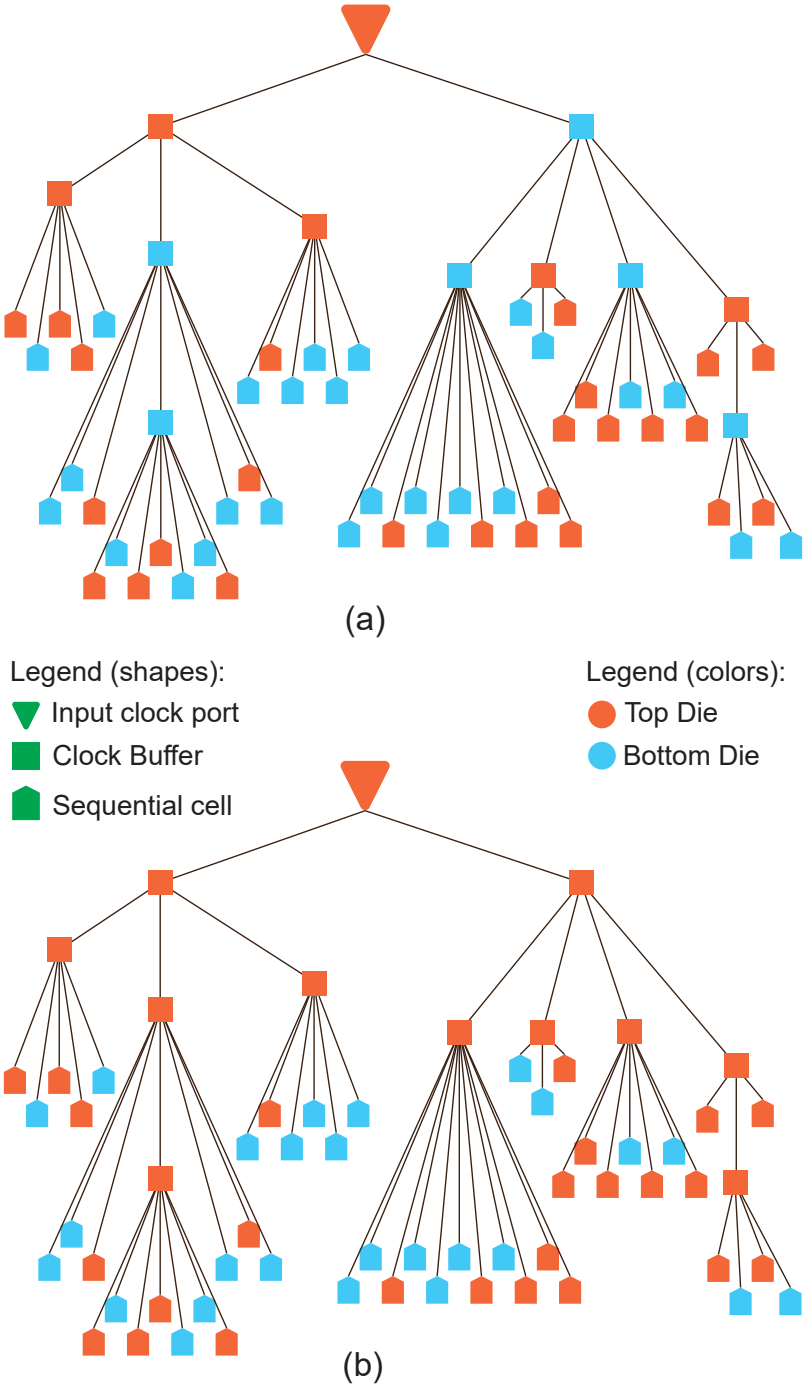


Figure 3.4: Example Clock Tree Network showing input clock, clock buffers, and sequential cells. (a) Clock Buffers allowed to be placed on both tiers. (b) Clock Buffers moved to the top-tier

Table 3.4: Clock Tree structure and other related metrics of a netlist designed with and without fixing clock combinational cells on top-die

Metric	Units	Without fixing	With fixing
Total Cells	-	855	538
Top Die Cells	-	507	506
Bottom Die Cells	-	348	32
Max depth	-	24	17
Total Area	μm^2	1217.3	871.2
Wirelength	mm	86.4	77.6
Max Latency	ns	0.818	0.685
Max Skew	ns	0.416	0.353

designs. This leads to better switching power with 3D due to the decreased wirelength and wire capacitance load. Another important difference in the routing between the C2D and Pin3D is the number of MIVs used for routing. Pin-3D designs have $\sim 2\times$ the number of MIVs of the C2D design. This is due to the full 3D routing using complete metal stack in Pin3D. From Figure 3.1, we can see that the M5_bottom, M6_bottom layers are much closer to the FEOL layer of the top tier than M5_top or M6_top. As the track usage reduces further away from FEOL, the bottom tier FEOL doesn't use much of the tracks in M5_bottom, M6_bottom leaving them open for the top tier nets. This adds additional MIVs on nets that do not require MIVs and helps to reduce congestion by distributing it across more metal layers. The routing done in the 3D metal stack is very different from the routing in the pseudo-3D stage. So cell sizing and buffering is necessary to achieve timing closure. This is also seen in Table 3.2, where the no clock opt design with just placement and legalization has large negative slacks.

Timing Closure for 3D

Figure 3.5 show the evolution of timing path delays between the pre and post partitioning stages. The final pseudo-3D stage with just the global routing parasitics is used for the pre-partitioning stage. At this stage, all the cells are placed in a single tier and the routing is done only on a 2D like BEOL. After partitioning, the cells now have a new z-location

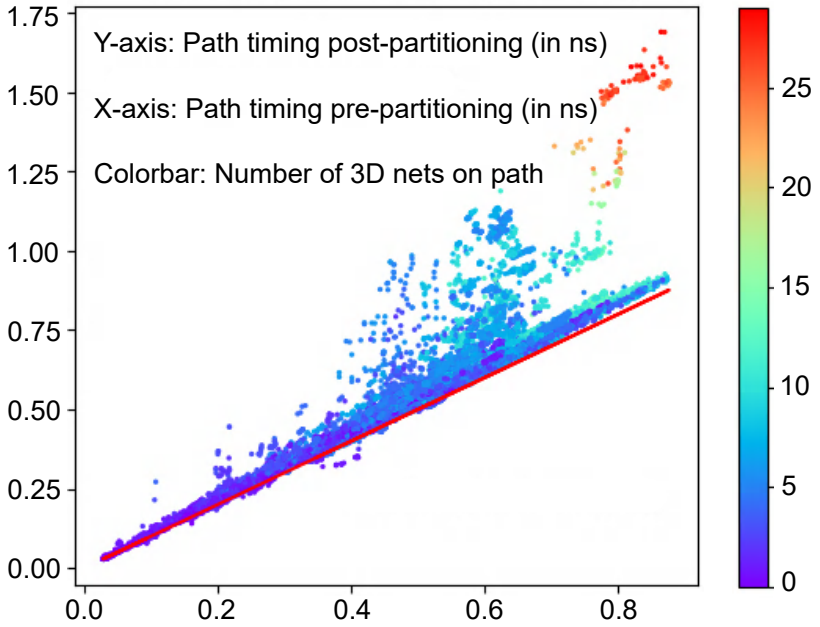


Figure 3.5: Path delays of a design before and after tier partitioning. The red line represents the line along which the delays are equal i.e., the path timing does not change after partitioning

corresponding to the tier assignment. The significant changes to routing BEOL and the placement leads to the deviations that are shown in Figure 3.5. Here, the worst critical path to each register in the design is plotted pre and post partitioning on the X and Y axes respectively. The points are colored by the number of 3D nets (nets having MIVs) on the critical path, and we see that the paths with more 3D nets deviate further away from the ideal (solid line in red) where timing of a path pre and post partitioning remain unchanged (x=y).

Using the three stage optimization methodology presented in subsection 3.3.4, we perform the timing closure to resolve the deviations caused from 3D partitioning, placement, and routing. The clock and timing optimization stages together result in a total slack reduction of up to $\sim 91\%$ compared to C2D design of the Cortex-A7 benchmark. This is from better skew assignments and cell sizing, but the increase in power consumption is $< 2\%$ showing the efficiency of Pin-3D methodology. The worst slack also improves significantly resulting in a 20% better frequency (=18% lower effective delay).

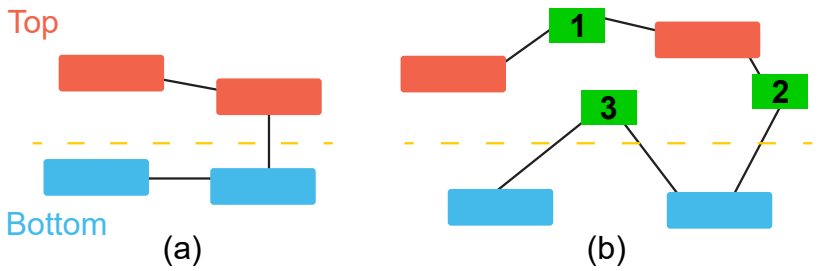


Figure 3.6: Example logical connectivity of netlist (a) Before top die optimization, (b) After top die optimization showing three different types (1, 2, 3) of buffer insertions shown in green

Efficiency of Pin-3D Optimization: A key benefit of Pin-3D is the presence of entire design information at every stage of the design as discussed before. This has two advantages in therms of design optimization. First, it allows for any change in the design to be completely aware of the entire 3D design. Second, during optimization of say bottom tier, the nets in the top tier can be modified by allowing for a buffer to be inserted on the net. Due to these net modifications of the transparent/fixed die, timing optimization is much faster than it would otherwise be. Figure 3.6 shows the three different types of buffer insertion during an optimization stage in Pin-3D.

To demonstrate, we compare two processor designs optimized two different ways with Pin-3D: including and excluding the capability of buffer insertion on fixed die nets. To do this, during top-die optimization, the nets that only connect to the cells in the bottom-die are marked as do not touch by the tool. This lets the logical structure of the net to remain unchanged while allowing any modifications to the physical layout of net during global and/or detail routing. Similarly, constraints are also applied vice versa during bottom die optimization.

Table 3.5 shows the PPA impact in Pin-3D without the additional type-3 buffer insertion. We see that it leads to a worse total slack and therefore a smaller effective frequency. The total number of buffers added during Pin-3D optimization is not very significant as it is at the post-route stage. But of the 2200 buffers inserted, 300 were of type-3 that break the net connecting cells in the transparent tier. While the PPA impact is negligible, the

Table 3.5: Efficiency of the Pin-3D optimization in timing closure. Critical parameters for Cortex-A7 are normalized w.r.t the 2D design

		OpenPiton		Cortex-A7	
	Units	Default	Exclude3	Default	Exclude3
Footprint	mm^2	0.6032	0.6032	α	1.0 α
Std. Cell Area	μm^2	0.3242	0.3237	β	1.0 β
Total Buffers	–	14051	13474	γ	0.999 γ
Worst Slack	ns	-0.483	-0.496	-0.280	-0.293
Effective Freq.	GHz	0.870	0.860	δ	0.987 δ
Number of Buffers Added During Pin-3D					
Total	–	2232	1620	616	602
Type1	–	843	783	213	278
Type2	–	1046	837	329	324
Type3	–	343	0	74	0

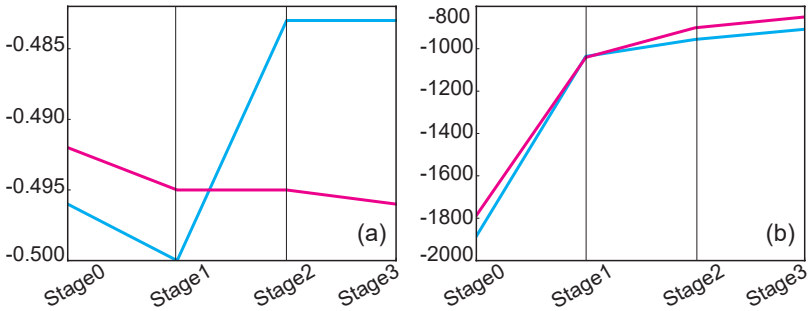


Figure 3.7: (a) Worst Negative Slack and (b) Total Negative Slack Trends during the three stages of Pin-3D optimization

main benefit of this type-3 insertion is the number of optimization stages needed for timing closure.

Figure 3.7 shows the evolution of the worst and total negative slacks in the 3D OpenPiton design. Stage0 refers to the post-route stage, where no 3D optimization is performed. Stages 1–3 are the three stages of optimization (top tier, bottom tier, top tier) of Pin-3D. First, we see that the Worst Negative Slack by excluding type-3 buffers is not able to match the default WNS value in 3D. This is where the type3 buffers show their importance. Since we only have a few buffer insertions and even fewer type-3 buffers inserted in a design, the total negative slack is not affected significantly.

ECO for 3D

The PPA results of the ECO algorithm introduced in subsection 3.3.5 are shown in Table 3.6. By identifying registers to both up-size and down-size, we see that the overall increase in the flip-flop area and therefore power is negligible. The number of violating paths (defined as unique begin-end point pairs) have reduced from 450 without ECO to 270, which in turn leads to a 33% reduction in the Total Negative Slack. It is important to note that the Worst Negative Slack doesn't improve by much with the ECO method suggested. This is because the total negative slack of a design is a compound of all the violating paths, but the worst slack is only from a single path. By adjusting the slack of each path by a small amount, we can reduce the total slack significantly. The worst path is not specifically targeted with ECO and hence it is not improved. We believe this is one of the first works showing applicability of ECO in monolithic 3D designs.

3.6 PPA Results and Analysis

3.6.1 Homogeneous 3D IC Design

Overall Design Results

The final GDSII layouts of the 2D and Pin3D implementations of the two processor designs are shown in Table 3.8, and the final PPA results of the 2D, C2D, and Pin-3D based implementations of the four test-bench circuits are given in Table 3.7.

C2D Comparison When compared to C2D, we see that Pin-3D has better results for almost all the metrics across the four benchmarks in Table 3.7. Moreover, we achieve 12% and 14% better EDP (Energy-Delay Product) than commercial 2D with our Pin-3D for commercial Cortex-A7 and Cortex-A53, respectively. In Table 3.7, PDP (Power-Delay Product) and EDP uses the effective delay ($=$ Clock Period $-$ Worst Negative Slack $=$ $^{1}/_{\text{effective frequency}}$).

Cortex-A7 As mentioned in section 3.4, the results for both Cortex-A7 and A53 are normalized w.r.t the 2D designs. Pin-3D shows better results than 2D for all the key design metrics such as cell count, wirelength, power, timing slacks. The 3D design shows $\sim 9\%$ power benefit at the same target frequency of 2D. Most of the power benefit in 3D comes from the wirelength reduction (here, 25% smaller compared to 2D) which leads to smaller wire load and therefore reduced switching power.

Switching power depends on the sum of wire cap and input gate capacitance of the cells. So the 25% wirelength reduction translated to 15% switching power reduction in the design. Internal and leakage power are a function of the cell area and cell types present in the final design. With reduced wire load, the cell strength and buffer strength can be reduced which explains the 4.4% drop in cell utilization and the smaller 1.4% drop in the cell count. This results in the 4% drop in internal power.

The reduction in leakage power drop is more drastic with 16.4% reduction compared to 2D. Leakage is significantly dependant on the threshold voltage type of the cell. The cell distribution based on V_{th} and the contribution of each cell type to leakage power is shown in Table 3.8. In 2D, we see that more than 50% of the cells are from the lowest V_{th} domain, and these contribute to around 80% of the total leakage power consumption. In Pin-3D the number of lowest V_{th} cells drop from 51% to 43% which is the main source of leakage power improvement.

Instead of looking at the total power by internal, switching, and leakage, we can also split the power consumption based on the cell type. Sequential and macro power in 3D have barely any improvements. For macros, the power is very dependent on the memory macros designs which remain the same between 2D and 3D, and so there is little reduction. Sequential power is the total internal and switching power of the sequential cells (flip-flops) which are connected to the clock network. Clock has a very high activity factor compared to many other combinational logic. Due to this, sequential power is dominated by the internal power of these cells which is not directly reduced from a 3D placement. Moreover,

as the sequential cells cannot be removed or added during the physical design stage, there is only limited optimization available during PnR.

The effective frequency ($= \frac{1}{\text{clock period} - \text{worst slack}}$) of the 3D design is 5.8% higher than 2D design due to the 3D placement and the optimization methodology used in Pin-3D. Combined with the power savings from 3D, the power delay product is 14% smaller than 2D, and the energy delay product is 19% smaller.

Cortex-A53 As both Cortex-A53 and Cortex-A7 are processor designs with L1 cache only and have a similar architecture and hierarchy, the benefits are similar between the two. The Cortex-A53 design is much larger than Cortex-A7 and we see a better reduction in overall power, cell count, cell area in 3D. With larger cell area reduction, the switching, internal, and leakage power reductions reach up to 18.6%, 7.5%, and 24.8% respectively. Overall this added up to 12.9% total static power reduction. The total negative slack also improves compared to 2D as it is 16.8% smaller. Effective frequency improvement is similar at around 5% and the overall EDP benefit compared to 2D was 21.5%. The larger size and higher combinational cell count of Cortex-A53 was useful to extract the higher power and timing savings. Wirelength reduction depends mostly from the routing complexity, wirelength distribution of the nets, and other metrics. It is not significantly dependent on the design size and so we see a reduction of 24.1% similar to Cortex-A7.

LDPC and Netcard The two open-source RTLs are used to show the raw design metrics with all the three implementations. The time related metrics are reported in ns, and power metrics are in mW for the two designs. All other units are reported in the table. LDPC and Netcard are open-source benchmarks, so the design metrics are not normalized. When compared to its baseline 2D design, LDPC Pin-3D shows a high power savings of $\approx$30% with only a relatively small frequency degradation. LDPC is a wire-dominated circuit, as can be seen from the high portion of switching power in the design. So, the wirelength reduction in M3D significantly reduces the output load of the cells, which can then be

down-sized without exceeding the delay targets. This leads to 26% reduction in the internal power, which is the most reduction across all the designs. Netcard does not have as high of a switching power proportion and still has a modest power savings of 12.3% with Pin-3D.

Memory Net Analysis

Memory macros and their placement are an important part of any design as they can create timing bottlenecks in the design. In cases with high memory workload, memory switching and the nets connected to the memories become a crucial part of power savings, and 3D is especially suited for reducing the latency and wire load of memory nets. By placing cells on top of macros, it becomes easier to connect macros and stand cells reducing the wire lengths of nets connecting to memories. We see long nets over macros in Table 3.8, with both the Cortex-A7 and Cortex-A53 2D designs. This routing over the memory macros can create huge net delays and wire cap load at the macro pins. By placing macros on top of each other, we create easier access and so there is a significant reduction in routing over the memory macros.

The impact is clearly seen in the memory net statistics portion of the processor circuits in Table 3.7. Again, due to the larger size and complexity of the design, the impact of 3D placement is more with the Cortex-A53 design. The Root Mean Square input and output net latency reduce by more than 40%. Here, Root Mean Square is used as the mean as it is more skewed towards the larger latency values which are more important in a determining bottlenecks. The net switching power of the memory macros are reduce by $\sim$25%. It is smaller with the Cortex-A7 with only 17% reduction. This is based on the 3D memory macro placement, and can be further optimized if memory nets are the primary concern.

Timing Path Analysis

The top-100 register-to-register critical paths are analyzed for a more in-depth understanding of the path-level trends in the designs. By focusing on a single path group rather than

different types of paths such as memory-to-register or register-to-output allows for better analysis and more cleaner data. Memory paths have significant portion of the path delay from the internal delay of memory macros that are not affected with 3D placement. And the paths to output ports are incomplete as they only have a portion of the logical stages in the design. The other portion would be a higher hierarchy level, and are represented by path margins that are again independent of 2D or 3D design.

Table 3.9 presents the averaged path statistics for the four circuits considered here. Since the Cortex-A benchmarks are processor designs, register-to-register paths may not be the most critical paths of the overall design. These are still useful to analyze the overall path trends. We see that in Cortex-A7 Pin-3D, the register-to-register paths have a worse average negative slack than the 2D design. This doesn't mean that the path is longer as we see that the path delay (from launch to capture registers inclusive) is smaller in Pin-3D compared to 2D. C2D has higher path delay and is representative of the path deviations shown in Figure 3.5. Clock skew on the Pin-3D paths is significantly large than the 2D designs which is the cause of the worse slack in Pin-3D. In Pin-3D, the overall critical paths of the design are connected to memories, and the clock skew is used to improve the timing slack of these paths. Depending on the path criticality, this changes, and we see that for Cortex-A53 the Pin-3D clock skews are much smaller to not cause additional timing bottlenecks for the already long paths (path delay $= 1.02 \times$ clock period). In both these designs, but particularly in Cortex-A53 we see that the wire delay contribution is much smaller in 3D compared to 2D.

For LDPC and Netcard, all the values are in ns, and the 100 paths considered are also the top-100 critical paths of the overall design. Clock tree is designed to help timing of critical paths, and we see that the average clock skews are similar within 15 ps of the 2D skew. LDPC has a very dense path connections and is difficult to obtain negative clock skew on critical paths without sacrificing timing of other paths. With more larger circuits like netcard, we do see the negative skew that helps the critical paths. Both of these circuits

also have large wirelengths and so have a higher portion of the path delay coming from wire delays.

3.6.2 Routing Analysis and Metal Layer Savings

As observed in Table 3.8, the usage of top-most metal layer in the top-die (shown in pink) is very low in M3D Cortex-A processor designs, less than 2% of the total wirelength is routed on this layer. In comparison, this number is 10% for 2D designs. This is mainly due to the difference in the routing stacks of 2D and M3D designs. Wirelength is shorter in M3D, and M3D makes better use of other metal layers as discussed in subsection 3.3.3. Thus, we achieve metal layer cost-saving by changing the number of metal layers in the 3D metal stack. Table 3.10 shows that the total power and delay in 3D Cortex-A7 are only affected by $< 1\%$ even with one less metal in 3D implementation.

3.6.3 Heterogeneous 3D IC Design

Pin-3D optimization methodology provides a versatile and robust way to incorporate cells from different technology nodes into a single circuit at a path level. As discussed in subsection 3.4.2, we design a 3D IC with process node heterogeneity (15 nm top-die, 45 nm bottom-die) for the first time at gate-level partitioning. Pin-3D flow requires an input Pseudo-3D stage to proceed with incremental 3D placement, clock tree optimization, routing, and timing optimization as shown in Figure 3.2. Since pseudo-3D is a 2D-like design, only a single technology node can be used at this stage. We start with the 15 nm process node to synthesize and obtain the input Pseudo-3D design.

During partition, the 15 nm Pseudo-3D stage is split and assigned to different technology nodes of top and bottom tiers. This creates a few constraints some of which are specific for heterogeneous 3D IC partitioning. First, for ideal silicon area usage the cell area between the two tiers should be identical. Monolithic fabrication means that the shape and size of the dies on the top and bottom tiers are the same, and a significant skew in final cell

areas on the two tiers implies that the underutilized die has unused silicon that increases the die cost unnecessarily. To do this, we use a simple global scaling factor while partitioning the 15 nm design. When a min-cut move change the cell tier from 15 nm to 45 nm, it's area is scaled by a factor α which is the average scaling factor of all cells from 15 nm $\longrightarrow$ 45 nm node. This allows us to keep track of area balance with heterogeneous cells.

Second, every cell in the 15 nm node should have a counterpart of the same type (logic type, driving strength, threshold voltage type, pin list). This allows cells partitioned to the 45 nm node have sufficient physical and timing information for PnR. Pin mapping from the 15 nm $\longrightarrow$ 45 nm nodes should also be strictly one-to-one so that no pins are not added or removed that can affect functionality of the circuit. Due to this, we only use the heterogeneous PDKs from the same source that ensures same functionality. Further the cells are limited to the basic cells such as Buffers, Inverters, NAND, NOR, XOR, Flip-Flops.

Figure 3.9 shows the placement and routing layouts of the 3D heterogeneous aes-128 design. Figure 3.9 (a) and (c) show the bottom 45 nm tier and the zoomed-in placement show the tall 1.4 µm cell rows. Similarly Figure 3.9 (b) and (d) corresponding to the bottom-tier show the much smaller and shorter 0.768 µm height cells in the 15 nm tier. The routing in this tier also shows a much thinner and closely spaced wires of the 15 nm node. As routing layers in each tier follow the design rules of their corresponding tier, there is no additional routing congestion due to the large number of cells on the 15 nm tier.

The final optimization PPA of the heterogeneous design are given in Table 3.11, and the values support the observations from Figure 3.9. Cell Area is similar between the two tiers with the help of partitioning. The 15 nm die has $\sim$70% of the total cells and 68% of the routed wirelength due to the smaller cell areas of the FEOL and thinner routing pitch of BEOL. As clock tree is important for the overall timing control of the design, it is fixed on to the 15 nm die as this is the Pseudo-3D input in which clock has been initially synthesized. The Clock Tree Statistics portion of the Table 3.11 also quantifies this fact. Sequential cells also contribute to significant cell delays and are fixed to the faster top-die as well.

The combination of large cell count, and the power-hungry clock-network and sequential cells on the top-die shows a high power consumption on this die. A significant benefit of this skewed power distribution is with thermal configuration. A heat-sink placed in contact with the top-die can absorb most of the thermal power, as the power density is very small on the bottom die that is further away from the heat sink. Thermal cooling has always been a major thorn for 3D ICs, and heterogeneous 3D ICs have an in-built power skew to tackle that problem.

As mentioned earlier, here, the heterogeneity is at a path level and this is seen in the critical path of the design. The aes-128 benchmark is a small logic block with 107 000 cells and the critical path here only has 18 stages. Of these 7 cells are on the 15 nm die and only contribute 0.051 ns to the overall delay. The 11 cells on the 45 nm die are particularly slow and contribute to almost all of the path delay. Before optimization, the paths are distributed randomly between the two tiers, and changing the cell node from 15 nm $\longrightarrow$ 45 nm for 31 000 out of 107 000 cells degrades the timing of the paths passing through the 45 nm die. This is seen in the pre optimization worst and total negative slacks of the design which are huge for the small test-bench used. Optimization with Pin-3D has helped reclaim almost all the negative timing slacks giving a heterogeneous design very close to timing closure.

3.7 Conclusion

In this paper we proposed our Pin-3D methodology for incremental placement optimization, clock tree optimization, routing, timing optimization, and ECO for 3D ICs using the commercially available PnR tools. Compared to the current state-of-the-art 3D flows, we showed that adding our Pin-3D optimization improves every aspect of the design from placement, routing, and PPA. Especially, we see more than a $10\times$ smaller total negative slack for the Cortex-A7 design and similarly high reductions in other benchmarks as well. Compared to 2D designs, we were able to see $\sim$20% reduction in EDP for the Cortex-A series benchmarks, and $\sim$30% EDP improvement for the LDPC benchmark. 3D routing

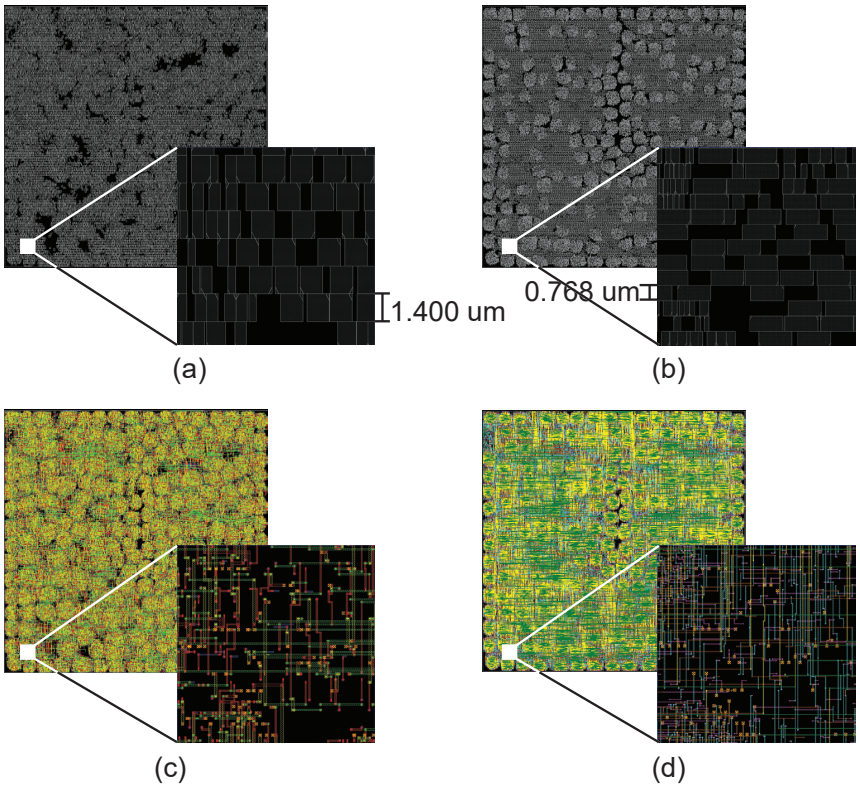


Figure 3.9: Layout of our 45 nm+15 nm heterogeneous 3D IC design of 128-bit AES benchmark using Pin-3D. (a), (b) Full placement in top and bottom dies respectively along with standard row height (c), (d) Full routing of the top and bottom dies respectively with zoom-in windows for each.

with Pin-3D also allowed for savings in the BEOL cost without any meaningful effect to the important PPA metrics. We also saw how the buffering insertion with Pin-3D methodology is superior in terms of critical path timing compared to a fairly limited die-by-die optimization. And finally, a proof-of-concept design of a heterogeneous 3D IC shows the versatility of Pin-3D as well as exploring more complex structures that are possible with 3D IC design.

Table 3.6: 3D ECO optimization result on register-to-register paths using Pin-3D ECO. We use Cortex-A7 in 28 nm.

Cortex-A7	w/o ECO	w/ ECO
Frequency	1	1.0
Sequential Cell Area	1	1.000
WNS	1	0.910
TNS	1	0.669
#Violations	449	270
Power	1	1.000

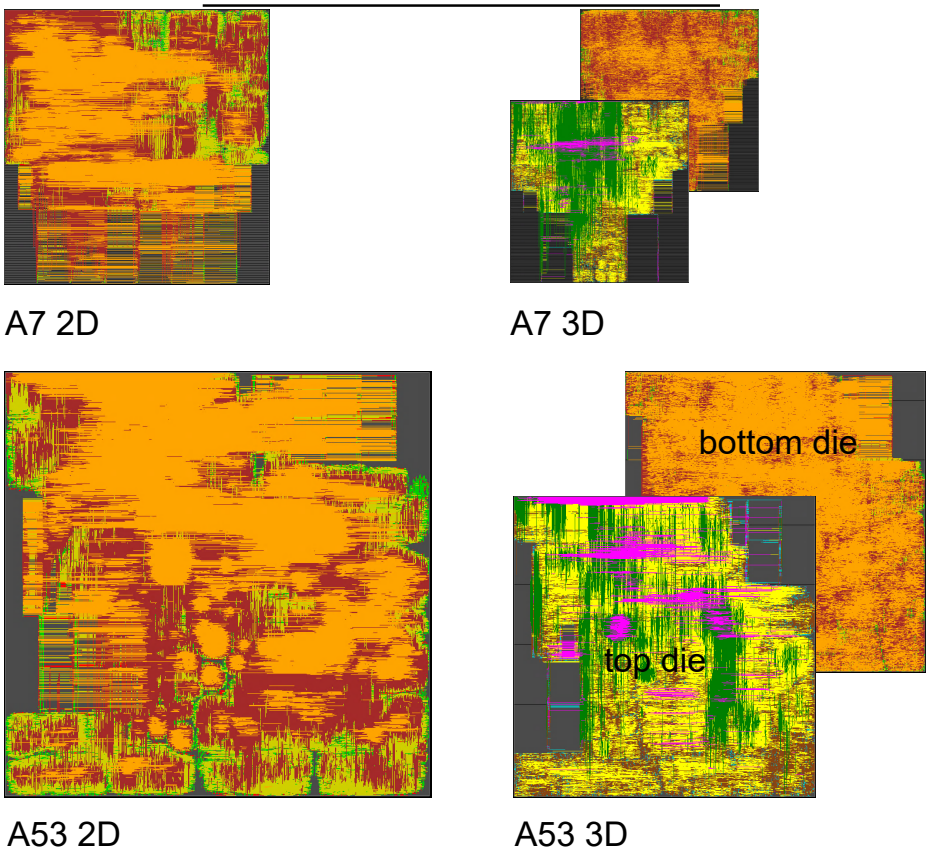


Figure 3.8: GDS layouts of our Cortex-A7 and Cortex-A53 designs. For 3D designs, we show the placement for the top die, and the routing for the bottom die. We use a TSMC 28nm technology in all designs.

Table 3.7: TSMC 28nm benchmark PPA comparisons among commercial 2D, Compact-2D [24], and Pin-3D optimized designs.

	Cortex-A7			Cortex-A53			LDPC			Netcard		
	2D	C2D	Pin-3D	2D	C2D	Pin-3D	2D	C2D	Pin-3D	2D	C2D	Pin-3D
Target Frequency (GHz)	1	1	1	1	1	1	1.500	1.500	1.500	1.250	1.250	1.250
Footprint (μm^2)	1	0.500	0.500	1	0.499	0.499	111,420	55,692	55,692	525,835	263,272	263,272
Cell utilization (%)	1	0.944	0.956	1	0.882	0.933	77.75	61.04	61.95	73.06	68.02	67.16
Gate Count	1	0.975	0.986	1	0.952	0.964	55,858	48,612	49,009	240,218	229,217	232,366
Total WL (m)	1	0.812	0.750	1	0.786	0.759	2.290	1.526	1.418	9.981	7.308	7.040
MIV Count	–	50,474	118,517	–	168,759	363,339	–	14,574	28,567	–	66,218	166,767
Internal Power	1	0.944	0.961	1	0.915	0.925	90.99	62.41	66.67	73.92	71.29	70.94
Switching Power	1	0.863	0.854	1	0.823	0.814	165.26	112.29	112.6	79.39	62.41	61.81
Leakage Power	1	0.739	0.826	1	0.743	0.752	0.19	0.12	0.133	0.298	0.207	0.199
Sequential Power	1	0.983	0.983	1	0.972	0.991	15.61	15.76	15.80	78.03	71.48	74.21
Macro Power	1	0.999	0.999	1	0.994	0.995		—			—	
Combinational Power	1	0.832	0.848	1	0.810	0.806	236.50	155.40	159.90	54.71	52.44	49.16
Clock Power	1	0.974	0.933	1	0.916	0.885	4.30	3.68	3.62	10.87	9.99	9.57
Mem Input Net Latency	1	0.680	0.724	1	0.558	0.554		—			—	
Mem Output Net Latency	1	0.640	0.578	1	0.632	0.570		—			—	
Mem Net Switching Pow	1	1.003	0.834	1	0.752	0.751		—			—	
Total Power (mW)	1	0.905	0.911	1	0.871	0.871	256.44	174.82	179.4	153.61	133.91	132.95
Total Negative Slack (ns)	1	10.786	0.999	1	7.919	0.832	-20.23	-141.21	-32.13	-2.145	-783.74	-1.221
Avg. Negative Slack (ns)	1	1.268	0.625	1	1.761	0.714	-0.011	-0.067	-0.018	-0.012	-0.032	-0.010
Total Positive Slack (ns)	1	0.587	1.512	1	0.508	1.347	662.0	562.8	672.2	5545.3	3955.3	6471.9
Effective Freq. (GHz)	1	0.843	1.058	1	0.844	1.054	1.429	1.221	1.415	1.160	0.969	1.193
Power × Delay (pJ)	1	1.074	0.861	1	1.031	0.827	179.50	143.18	126.8	132.41	138.19	111.44
Energy × Delay (pJ*ns)	1	1.275	0.814	1	1.220	0.785	125.65	117.26	89.59	114.14	142.62	93.41

Table 3.8: Cell Distribution by threshold voltage types in Cortex-A7 2D and Pin-3D designs. The threshold voltage types are labelled 1 (lowest V_{th}) — 4 (highest V_{th})

V_{th}	2D		Pin3D	
	% of Cells	% of Lkg.	% of Cells	% of Lkg.
Type1	51.4	82.7	43.4	78.2
Type2	31.7	10.6	35.0	13.5
Type3	9.0	0.7	10.1	1.0
Type4	7.9	0.2	11.5	0.3

Table 3.9: Top 100 critical path averages of register-to-register path group. The Cortex-A metrics are normalized w.r.t the clock period.

	Cortex-A7			Cortex-A53		
	2D	C2D	Pin-3D	2D	C2D	Pin-3D
Clock Period	1	1	1	1	1	1
Path Slack	-0.041	-0.285	-0.098	-0.157	-0.260	-0.102
Clock Skew	0.050	0.025	0.175	0.135	0.117	0.076
Setup Time	0.007	0.024	0.011	0.004	0.010	0.006
Path Delay	0.984	1.236	0.912	1.108	1.133	1.020
Cell Delay	0.891	1.188	0.834	0.912	1.066	0.966
Wire Delay	0.093	0.048	0.078	0.105	0.067	0.054
	LDPC			Netcard		
	2D	C2D	Pin-3D	2D	C2D	Pin-3D
Clock Period (ns)	0.667	0.667	0.667	0.800	0.800	0.800
Path Slack	-0.025	-0.123	-0.034	-0.019	-0.140	-0.012
Clock Skew	0.005	0.055	0.021	-0.026	0.017	-0.015
Setup Time	0.019	0.017	0.017	0.023	0.100	0.033
Path Delay	0.668	0.717	0.663	0.821	0.824	0.794
Cell Delay	0.578	0.663	0.611	0.597	0.688	0.627
Wire Delay	0.090	0.054	0.052	0.224	0.136	0.167

Table 3.10: Impact on PPA with one metal layer removed

Cortex-A7	Pin-3D	-1 Metal
Frequency	1	1
Total Metal Layer Count	12	11
Wire Length	1	1.002
MIV Count	104,219	104,621
Total Power	1	1.001
WNS	1	1.006
TNS	1	0.958
Eff. freq.	1	0.999

Table 3.11: PPA results of our 45 nm+15 nm heterogeneous 3D IC design of 128-bit AES benchmark using Pin-3D. We use 2GHz as the target frequency of the whole design.

Design Metric	Total	Top-Die	Bottom-Die
Technology Node	Hybrid	15 nm	45 nm
Number of Cell Rows	–	285	156
Cell Area (μm^2)	60,887	29,832	31,055
Gate Count	107,201	74,203	32,998
Buffers Added	3,160	1,091	2,069
Wirelength (mm)	832.2	572.3	259.9
MIV Count	39,237	–	–
Total Power	123.24	104.09	19.15
Critical Path Delay (ns)	0.553	0.051	0.502
Critical Path Cell Count	18	7	11
Footprint (μm^2)		48,246	
Optimization Statistics			
Pre Opt. WNS (ns)		-0.615	
Pre Opt. TNS (ns)		-278.4	
Final WNS (ns)		-0.051	
Final TNS (ns)		-1.418	
Clock Tree Statistics			
Buffer Count	463	463	0
Wirelength (mm)	19.64	19.45	0.19
Max Latency (ns)		0.116	
Max Skew (ns)		0.045	

CHAPTER 4
METAL LAYER SHARING: A ROUTING OPTIMIZATION TECHNIQUE FOR MONOLITHIC 3D ICS

The main focus of this chapter is the analysis of routing quality in various 3D bonding and orientation types. *For the first time, we analyze a type of routing that is specific to 3D ICs – metal layer sharing*. We investigate the effect of different 3D arrangements on metal layer sharing, and how it can be leveraged to efficiently use the metal layers in the 3D stack. We show that, with metal layer sharing, an entire metal layer can be dropped from the routing stack without negatively affecting the maximum performance or power efficiency of the design. We also see how this phenomenon creates better 3D designs with up to 8% higher power efficiency. Finally, we analyze the routing of the 3D ICs with respect to congestion and Design Rule Violations (DRVs) due to the metal layer sharing.

This chapter is organized as follows: section 4.1 discusses the different effects of routing in 2D and 3D ICs of different configurations. In subsection 5.5.1, we present our setup and methodology to analyze the affect of metal layer sharing. In section 4.3, we quantitatively discuss metal layer sharing in various scenarios of 3D ICs with Place and Route (PnR) simulations of a processor design. In section 4.4, we discuss several aspects of metal layer sharing in monolithic 3D IC design of three commercial processors. Finally, section 4.5, concludes the paper.

4.1 Characteristics of Routing

Here, we discuss and analyze the general routing characteristics of 2D ICs, and the various routing scenarios of 3D ICs. The routing discussion of 2D ICs helps to define the baseline routing analysis and to understand the full impact of the metal stack in 3D routing.

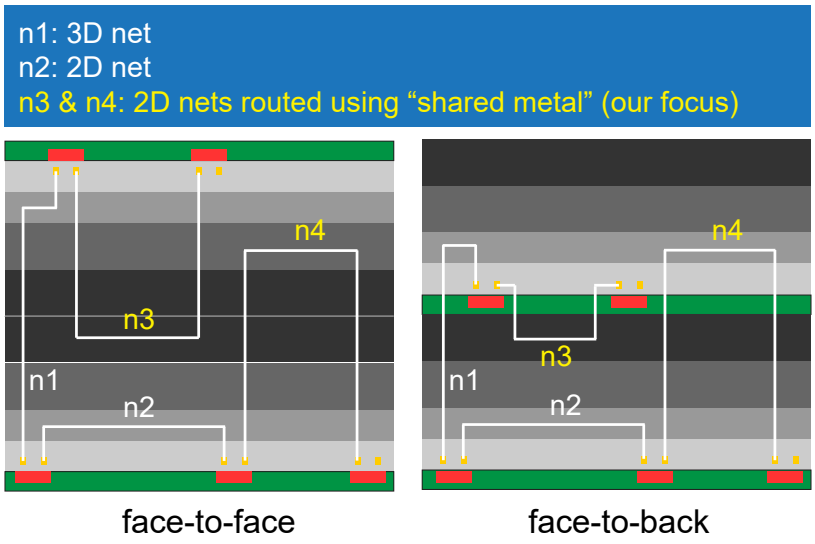


Figure 4.1: Routing layer sharing in face-to-face and face-to-back 3D ICs. Green portion represent the active FEOL layers, Gray represents the dielectric and various routing layers. The darker shade corresponding to higher thickness, pitch, and lower parasitic values of metal layers

4.1.1 2D IC Routing Characteristics

In a traditional 2D IC, there is a single layer of Front End Of the Line (FEOL) followed by multiple layers of the Back Of the Line (BEOL) for routing the cells to each other. The routing within the cells is mostly limited to the metal layer closest to the FEOL (M1). While the layer on top, M2, is only utilized in cells with more complex internal connectivity, such as flip-flops. Due to the close proximity to standard cells, M2 is used effectively to route short wires rather than longer wires. This is because long wires block a contiguous portion of the available routing tracks that can block M2 tracks over several cells. All the nets have some routing on M2 (sometimes M1 is used based on the cell placement) for pin access making this a highly utilized layer in general.

As we climb further in the metal layer stack from M3–M6 we see that the number of nets on the layer decreases while the average length of uninterrupted wire segment routed on the layer becomes longer. These two phenomenon are closely related to each other. As more nets are routed on a layer, it is effective to use it for shorter wire segments, while

Table 4.1: Inter-cell routing layer usage in OpenPiton 2D IC used as a reference. A wire segment is a single continuous piece of metal routed in a straight line.

Metal Layer	Number of nets	Avg. wire segment (μm)
M1	1400	0.77
M2	212900	0.44
M3	181100	1.27
M4	79200	1.96
M5	34800	4.87
M6	15000	10.90

letting longer portions of the nets route on higher layers. Additionally, as we go up the stack, most of the nets would have sufficient routing tracks available to complete routing without needing to route further up the stack, leaving only a few long nets as we reach M6. This wire distribution behavior of 2D ICs is shown in Table 4.1.

The metal layers are also engineered in a way to consider the wirelength trends and to reduce the wire load. One such technique is to gradually increase the metal layer width and the pitch as we move higher up the stack. In conjunction with the metal layer design, the surrounding dielectric medium and the thickness are also increased with the pitch to reduce the parasitics of the wire. The smaller width of the lower metal layers, accommodates more nets closer to the routing stack. And as the metal layer at the top do not route many nets, it can accommodate a larger pitch. This reduces the wire resistance, and the increased spacing helps to limit any coupling parasitics for the long wires to keep the wire delay in check.

4.1.2 3D IC Routing Characteristics

Figure 4.1 show example cross-sections of the two 3D IC orientations along with their BEOL. The BEOL of each tier within the 3D IC is colored to reflect the fact that the parasitics, average wire segment length decrease monotonically along the stack. By joining two different BEOL stacks in 3D IC, we see that the overall routing stack can change significantly not only from 2D, but also between the two 3D orientations. To understand different routing scenarios that are created due to this stacking, we split the nets into different cate-

gories based on connectivity and routing:

- *3D nets*: Nets connecting cells from more than one tier
- *2D nets*: Nets connecting cells located in a single tier
  - *no sharing*: 2D nets that are routed on the BEOL of its own tier. Sometimes referred to as default 2D nets
  - *with sharing*: 2D nets that borrow tracks from metal layers of other tier for their routing

Face-to-Face Bonded 3D IC

In the Face-to-Face (F2F) orientation, the two tiers are attached at the metal layer face. Assuming a single tier BEOL stack to be from metals *1* through *x* (*M1 – Mx*), the 3D BEOL stack would be as follows: *M1_bottom – Mx_bottom - Mx_top – M1_top*. In this configuration, the metal layer sharing is limited as the two FEOL layers are separated by the BEOLs of the two tiers.

Consider the 2D net $n2$ from Figure 4.1 (F2F case) that only connects the cells from a single tier (here, bottom FEOL). Specifically this is a type of net that does not use metal layer sharing as the routing is limited to its own BEOL (*M1_bottom – Mx_bottom*). As such, the routing characteristics of this net are not different than a normal net in 2D IC with only a single FEOL and a BEOL with monotonic decrease in parasitics per unit length.

From the pin connectivity, nets $n3$, $n4$ are also classified as 2D nets, as they connect to pins that are in a single tier (top for $n3$, bottom for $n4$). But these nets use routing tracks from metal layers that are not from their own BEOL. In our example Figure 4.1, this is shown by the use of *Mx_top* for the bottom-tier 2D net $n3$, and routing on metal layer *Mx_bottom* for the top-tier 2D net $n4$. As we have discussed previously, the number of nets that require higher metal layers for routing decreases gradually (from *M1* to *Mx*), and so metal layer sharing can be very limited in F2F designs.

Finally, the nets of type $n1$ are examples of the 3D nets as they connect cells from different tiers. In order to achieve full connectivity, these nets must be routed across all the layers in 3D metal stack (BEOL of bottom layer + BEOL of the top layer). This adds, what is referred to as a "3D overhead", excess routing that needs to be done for 3D nets due to taller BEOL stack and the placement of pins on either ends of the stack.

Face-to-Back Bonded 3D IC

The Face-To-Back (F2B) stacking creates a 3D metal layer stack that is the most different from any of those discussed above. By connecting the top layer of bottom tier (*Mx_bottom*) to the back-side of the top-tier FEOL, the 3D stack becomes *M1_bottom – Mx_bottom - M1_top – Mx_top*. This places the bottom tier *Mx* which usually has the least routing, easily accessible to the cell pins of the top-tier. This encourages metal layer sharing in the top-tier 2D nets such as $n3$ from Figure 4.1 (Face-to-Back case). Since the 3D vias compete with standard cells for silicon area, there will additional detour to find legal locations for the 3D vias going into bottom tier BEOL.

Moreover, sharing on the bottom-tier nets such as net $n4$ is further restricted as they are placed further away from the *Mx_top* compared to the same type of net under F2F stacking. The high track utilization of metals above the top-tier FEOL and the added dependency between 3D vias and the top-tier cell placement create additional restriction for metal layer sharing of $n4$-type nets.

3D Bonding and Via Pitch

Apart from the orientation of the tiers, the bonding type also plays an important role in determining the feasibility of metal layer sharing. While a higher pitch discourages metal layer sharing, due to fewer available connections that can be made. The parasitics of the bonding structures also play an important part. For example, with micro-bump bonding, the bumps have a high parasitic value and can significantly add delays for the paths. So,

micro-bump bonding is left out of consideration when talking about metal layer sharing. With sequential fabrication, and hybrid bonding neither the pitch size nor the parasitics become very important in the overall path delays, and can be aggressively used for metal layer sharing.

4.2 Experimental Setup

4.2.1 3D PnR and Controlling the Metal Layer Sharing

In order to perform 3D placement and routing, we use Pin-3D [13] and Macro-3D [14] tool flows in our work along with Innovus Place And Route tool version 20.15. These allow us to do a wide range of partitioning and 3D bonding types to analyze the routing across various 3D configurations. Macro-3D flow is well suited for designing memory-on-logic 3D IC designs using hybrid bonding or monolithic integration. Pin-3D flow is a more generalized flow that works well with any type of partitioning and supports both the 3D bonding types that are possible with Macro-3D.

Neither Pin-3D nor Macro-3D offer any differentiation between net types (2D or 3D) during routing stage, leaving the routing fully driven by the router. In order to analyze the metal layer sharing separately, we need to control the signal routing which is done using custom scripts in our work. In Place and Route flow, early global routing of the signals starts as early as the placement stage where the trial routing is done to improve the placement quality. Detail routing is first done at the clock tree synthesis stage, where the clock network is routed before any other nets are routed to have the best possible clock tree design. After the clock tree stage, the entire design is routed to based on the global routing.

In order to control the metal layer sharing, we identify the nets that connect to different tiers (3D nets) and the nets limited to single tier (2D nets). The 2D nets are then restricted to be routed in their respective metal layers while letting the 3D nets to be routed on the entire 3D metal stack. In addition, the clock nets are handled more strictly to be routed on only limited routing layers to limit the metal layer sharing on the clock tree. Clock tree

requires special care as the clock optimization engine does not honor the routing rules we initially set for the whole design. By controlling the metal layer sharing, we can effectively isolate and study its effects on the full chip PPA of 3D IC designs.

4.2.2 Benchmarks and Technology Setup

Benchmarks Used

In order to analyse the metal layer sharing, three different commercially available CPUs are considered. These are widely-used in consumer electronics and their names and layouts are not revealed to protect their IP as per our NDA. These CPUs are further referred to as Industry-A, Industry-B, and Industry-C. Industry-A design is a dual core processor with 512 kB of shared L2 cache and 32 kB each of L1 Instruction and Data caches. Industry-B is a larger single-core processor with 1 MB of L2 and 32 kB of L1 Instruction and Data. Industry-C is the last commercial circuit considered with 512 kB and 16 kB of L1 Instruction and Data Caches. Finally, an open-source RISC-V processor (OpenPiton) is also considered to freely discuss and show the layouts wherever applicable.

Technology Process

The 3D Process Design Kit (PDK) is heavily based on a commercial 28 nm PDK in this work. For the monolithic integration case, the 3D via (also referred to as Monolithic Inter-tier Via or MIV) is assumed to have a pitch of 0.14 µm which is close to the pitch value of an *Mx* via. In the hybrid bonded 3D IC, a larger pitch of 1.00 µm is considered to accommodate for alignment accuracy during bonding.

4.3 Metal Layer Sharing Scenarios

In this section, we first empirically analyze the difference in metal layer sharing based on, the orientation of 3D dies subsection 4.3.1), type of partitioning used for 3D design subsection 4.3.2), and the bonding type/pitch of the 3D design subsection 4.3.3). Across all these

Table 4.2: Metal layer sharing in different 3D orientations using OpenPiton RTL. #MIVs on 2D nets shows the amount of metal layer sharing.

	Units	F2B	F2F
Frequency	MHz	1400	1400
Chip Area	mm^2	0.638	0.638
# MIVs	–	120,351	3,112
# MIVs on 2D nets	–	119,317	2293
# MIVs on 3D nets	–	1034	819
Wirelength	m	6.36	5.81
Worst Neg Slack	ns	-0.384	-0.438
Effective Frequency	MHz	910.5	867.8
Total Neg Slack	ns	-864.5	-540.2
Total Power	mW	414.6	411.2

comparisons, the number of metal layers is kept constant with 6 metal layers per BEOL of each tier for efficient signal routing. The choice of the metal layer count is based on the logic-on-logic partitioned 3D subsection 4.3.2) which requires has significant routing on the metal 6 of both top and bottom tiers.

Why Analyze Metal Layer Sharing?

While 3D has multiple routing scenarios as discussed in subsection 4.1.2, the 2D nets without sharing are same as any net in a traditional 2D IC design. The 3D nets, while specific to 3D and are interesting in their own right, are unavoidable and require to be routed to achieve full connectivity of the cells. 2D net routing with sharing, on the other hand, is specific to 3D and can be controlled manually or using the commercial tools. Therefore, it is important to understand the characteristics of these nets and their usefulness in the overall physical design of 3D ICs.

4.3.1 Metal Layer Sharing with Different 3D Bonding Styles

As mentioned in subsection 4.1.2 and depicted in Figure 4.1, the F2F and F2B orientations of the 3D IC can show significantly different routing characteristics. This is analyzed and quantified in Table 4.2 using the OpenPiton RTL. The partitioning is kept constant between

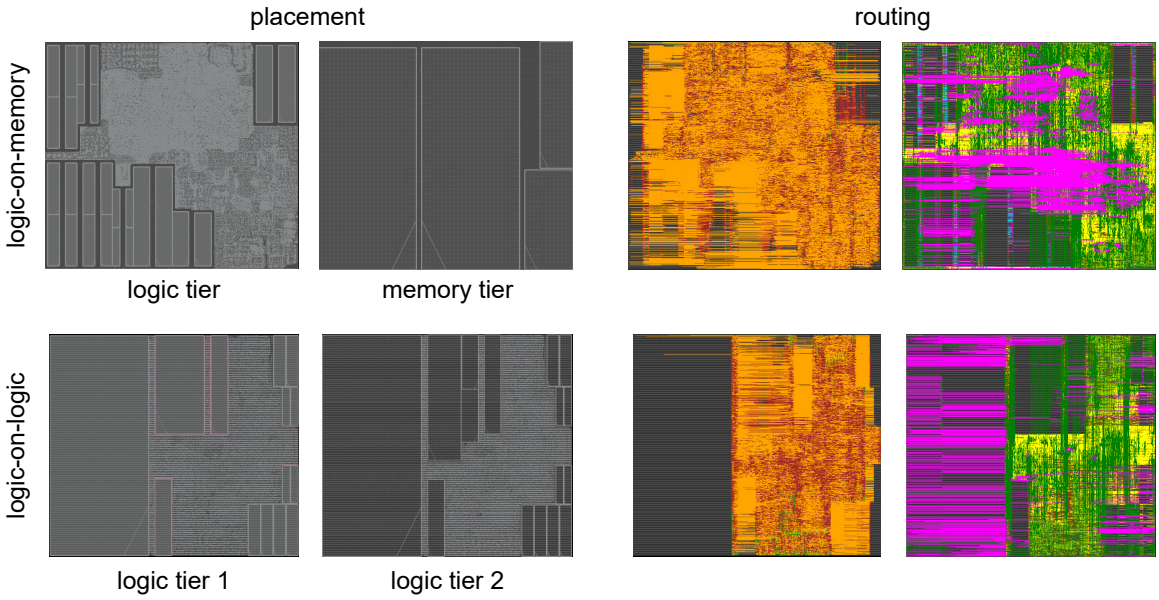


Figure 4.2: Comparing tier partitioning impact on routing in OpenPiton. The placement and routing layouts in the two tiers are provided for the two styles of partitioning. Memory tier and Logic tier 2 are the bottom FEOL in their corresponding designs.

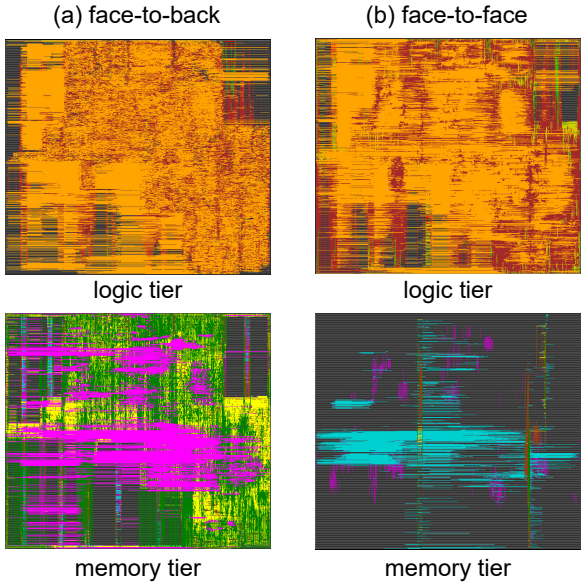


Figure 4.3: Routing comparison between two bonding styles of Logic-On-Memory 3D ICs. (a) F2B, (b) F2F. The logic tier BEOL layouts are on the top, and memory tier BEOL layouts the bottom. Each color corresponds to a routing layer.

the two with the L3 data caches and related tag blocks on the memory die, and everything else (computational core, and the L2 and L1 caches) on the logic die. Macro-3D [14] flow is used for the 3D PnR of both these styles. Additionally, in this style of partitioning and optimization with Macro-3D flow, the memory die placement is locked and no standard cells (including clock buffers) are allowed on it. The logic-on-memory placement layouts in Figure 4.2 show the partitioned layouts of the two tiers.

Comparing the design metrics from Table 4.2, we first observe a few things: the two designs are implemented at the same target frequency and with the same chip area to have a direct comparison between the two in terms of routing and metal layer sharing. The major difference between the two comes in terms of the number of MIVs in the design. The F2B design makes abundant use of 3D vias: 120 351 vias compared to just 3112 in the F2F case. A further classification of the MIVs based on the type of nets they are located on shows a clearer picture.

On the 3D nets – which are governed by the partitioning – the number of MIVs are similar between the two with F2B having a slightly larger count by 215 MIVs. The 3D net are logically the same between the two orientations, and this increased via count comes from the difference in routing the nets between the two designs. Due to the sudden change of the routing types between the layers at the interface of the 3D connection in F2B (*Mx_bottom* and *M1_top*), the routing behavior becomes more chaotic. The high routing blockage density of the *M1_top* and the high track usage of the *M2_top* compared to a lower blockage and routing densities of the *Mx_bottom* layer, the 3D nets crosses the 3D interface multiple times, thereby accruing multiple MIVs per net. This is referred to as *snaking* and is a commonly observed phenomenon for 3D ICs (especially with automatically routed F2B 3D ICs).

But most of the additional MIVs in the F2B designs are on the 2D nets. As discussed in subsubsection 4.1.2, the ease of access to the bottom tier BEOL from the top-tier nets lets creates this situation. Moreover, since a 2D net is defined as those that connect to cells

with a single tier, in order to borrow routing tracks from metal layers of the bottom BEOL, they have cross the 3D interface to access the bottom BEOL, and then cross it again to connect to their sink pins. Therefore, 2D nets undergoing metal layer sharing have at least 2 MIVs per net.

Finally, we see the macro design properties of the two designs. The wirelength of the F2B version is larger due to the added 'snaking' and the additional routing required for metal layer sharing. While this affects the total power slightly by $\approx 3\,\mathrm{mW}$, it is more beneficial for the worst timing path delay. Two factors are at play here that are dependent on the routing stack and the routing behavior. With the two FEOLs separated by both bottom and top BEOL stacks in F2F configuration Figure 4.1), the 3D nets would have longer wirelength. So paths through the 3D nets are vulnerable to delay increase in this configuration. Second, with more metal layer sharing of the F2B configuration, the metal layers are well utilized with a reduced routing usage on the top-tier BEOL stack. This decreases congestion in the design and allows for fewer detours in routing critical paths on the top-tier. The routing layouts in Figure 4.3 show a noticeable difference in the memory tier routing, which is left under-utilized by the F2B option. Consequently, we see that in the logic-tier, the F2F routing has more long wires on the top-most *Mx* layer (orange colored in Figure 4.3) compared to the F2B option.

4.3.2 Metal Layer Sharing with Different 3D Partitioning

In this section, we turn our attention to the partitioning of the RTL and its impact on routing in 3D. The two choices considered are the Logic-On-Memory (same as the partitioning discussed in subsection 4.3.1) and the Logic-On-Logic partitioning. The corresponding layouts are shown in Figure 4.2 for both placement and routing. Unlike Logic-On-Memory partitioning, where the memory tier is limited to pre-placed macro blocks, the Logic-On-Logic style has both logic and memory blocks on both tiers, and is implemented using Pin-3D flow [13]. Pin-3D flow cannot properly optimize the Logic-On-Memory design

due the largely asymmetrical partitioning, and Macro-3D cannot be applied to designs with logic block on both tiers. Because of this, the two different flows are used to handle the two partitioning types.

F2B orientation is used for both 3D partitioning options to encourage metal layer sharing. The PPA comparisons and other design metrics are shown in Table 4.3. We first see that the Logic on Memory style has a slightly larger footprint than the Logic on Logic option. This is because of the fact that the memory tier can only fit macros which do not fit well together leaving unused white space in the Logic On Memory option. This can be seen in Figure 4.2 where the memory tier has some unused white space. In terms of the MIV count, both options have a similar MIV count at around 100k and 120k. But the origin of these MIVs are very different to each other, with most of the MIVs in the Logic On Memory option coming from 2D nets, but they're are majority from 3D nets in the Logic On Logic option. We have already analyzed the MIV distribution for Logic On Memory case in subsection 4.3.1, and is not further mentioned here. In Logic-On-Logic, the coarse gate-level min-cut partitioning creates a very large cut-size that in-turn creates a high number of 3D nets and MIVs. More than 85% of the total MIVs in this case, are on the 3D net which is a stark contrast from the distribution in the Logic On memory option. Out of the 17 000 MIVs on the 2D nets, we see that virtually all of them are used for borrowing tracks from the bottom tier. This further supports our discussion of the metal stack and routing discussion in subsubsection 4.1.2. The uneven metal layer sharing across the two dies is also seen in the Logic On Memory partitioning.

As a whole, the total wirelength is significantly smaller in the logic on logic designs due to the symmetrical nature of the partitioning and the high 3D net count of this option. As the number of 3D nets increase. more nets can have shorter wirelengths due to the 3D placement, as well as the net detours. This leads to a much smaller total power consumption due to the decreased wire load. Out of the total routed wirelength, more than 25% is on borrowed metal layer tracks showing the abundance of metal layer sharing for the Logic

On Memory option. This percentage is only at 6% with the Logic On Logic option. The main reason for the decreased sharing in this case is due to the large number of 3D nets along with the symmetrical partitioning in both tiers.

With Logic On Memory partitioning, the memory die has a lot of unused routing tracks on M5 and M6 which are not used by the memory blocks for intra-cell routing. This allows for more of the logic tier (top tier) nets to flow on to the bottom tier. In contrast, the symmetrical nature of the Logic-On-Logic case (memory blocks are placed on top of each other, and the sea of logic cells of both tiers is also largely located in the same region of the tier). This makes the unused routing tracks on top of the memory macros harder to be utilized by the other nets in the design. At the same time, the routing tracks on top of the sea of logic region is heavily utilized with not enough free space for metal layer sharing. This is visualized in Figure 4.4, where the routing of layers M5 and M6 in the bottom tier is shown for both partitioning options. Note that the bottom tier M5, M6 have the highest metal layer sharing wirelength among all the layers for both options as seen in the 'Shared Wirelength' block of the Table 4.3. The wires in Figure 4.4 are colored based on their type of routing. The wires that belong to nets on the top die (routed on borrowed tracks from bottom/memory tier) are shown in red, and the other nets (default 2D and 3D net routing) is shown in yellow. The %age of wirelength in the memory tier layers (bottom tier for Logic On Logic case) used for the purpose of metal layer sharing is also calculated in Table 4.3 showing more than 95% shared wirelength on the layers M5, M6 of the memory tier. The metal layers M4–M1 in the bottom BEOL of Logic On Memory option is mostly occupied by intra-cell routing of memory macros, and for intra-die routing in Logic On Logic option.

4.3.3 Impact of Pitch on the Metal Layer Sharing

Finally, we measure the impact of pitch on the 3D routing and metal layer sharing using two the logic-on-memory partitioning in the F2B orientation. Two pitch values (0.14 µm and 1.0 µm) are used for this comparison and the results are tabulated in Table 4.4. We

Table 4.3: Metal layer sharing in 3D partitioning options: Logic+Memory, Logic+Logic. #MIVs on 2D nets shows the abundance of metal sharing in the designs.

	Units	Logic+Mem	Logic+Logic
Frequency	MHz	1400	1400
Chip Area	mm^2	0.638	0.603
# MIVs	–	120,351	104,606
# MIVs on 2D nets	–	119,317	17,575
# MIVs on 3D nets	–	1034	87,031
# MIVs on clk nets	–	1363	13,278
Borrow from bottom	–	119,317	17,421
Borrow from top	–	0	154
Wirelength	m	6.36	4.66
Shared Wirelength	%	25.1	6.4
Worst Negative Slack	ns	-0.384	-0.403
Effective Frequency	MHz	910.5	895.0
Total Negative Slack	nHz	-864.5	-631.6
Total Power	mW	414.6	378.3
% WL of shared nets in the memory tier			
M6	%	97.4	29.7
M5	%	95.5	18.1
M4	%	91.2	2.6
M3	%	36.0	0.1
M2	%	2.3	0.0
M1	%	0.0	0.0
Shared Wirelength			
Top Layer M1 – M6	µm	0	10,567
M6	µm	690,461	111,529
M5	µm	888,242	138,754
M4	µm	21,197	11,366
M3	µm	14,807	694
M2	µm	47	59
M1	µm	0	7

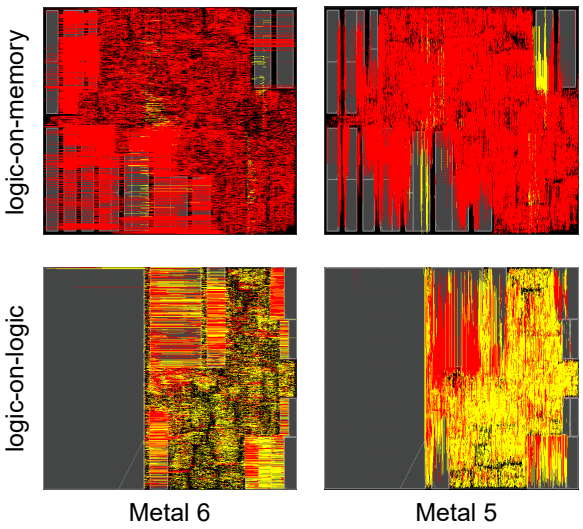


Figure 4.4: Routing in shared metal layers of 3D OpenPiton design with F2B bonding style. We show M5 and M6 of the memory tier and logic tier 2. Red are routing with metal sharing, and yellow is everything else.

quickly see a drastic reduction in the number of MIVs used. This is due to the larger pitch which increases the area occupied by each MIV by 100 $\times$. The shared wirelength also reduce by $\approx$ 45% due to this. As seen with different orientation in subsection 4.3.1, the reduced sharing decreases the wirelength but increases the worst slack.

Another key difference that undermines the usage of 1.0 µm bumps for metal layer sharing is the number of violations in the design. The larger 1.0 µm MIV pitch is significantly large ($\approx$ 10 $\times$) than the pitch of metal layers and vias around it. This creates routing violations as seen in the Table 4.4. Similarly, the height of the cells with which the MIVs compete for area in the F2B orientation is 1.2 µm making it comparable to the larger MIV pitch. This severely limits the number of MIVs that can be placed in the design. Ideal MIV occupancy roughly calculates the number of MIVs that can fit in the free area, and the 1.00 µm pitch already utilizes 30% of the available space. In comparison, the smaller MIV pitch, has $<$1% utilization even with its larger MIV count.

Due to the above-mentioned reasons, we only use an aggressive pitch of 0.1 µm. This is in-line with the 3D pitch values used in other works that consider monolithic-3D integration.

Table 4.4: Metal layer sharing with F2B oriented Logic+Memory partitioning at different pitch values

	Units	0.10 µm Pitch	1.00 µm Pitch
Frequency	MHz	1400	1400
Chip Area	mm^2	0.638	0.638
# MIVs	–	120,351	33,194
# MIVs on 2D nets	–	119,317	32,264
# MIVs on 3D nets	–	1034	930
Borrow from bottom	–	119,317	32,264
Borrow from top	–	0	0
MIV Occupied Area	mm^2	0.001	0.033
Ideal MIV Occupancy	%	<1	29.6
Wirelength	m	6.36	6.06
Cell Area	mm^2	0.512	0.514
White space	mm^2	0.136	0.134
# Routing violations	–	9353	128,162
Worst Negative Slack	ns	-0.384	-0.410
Effective Frequency	MHz	910.5	889.5
Total Negative Slack	nHz	-864.5	-934.0
Total Power	mW	414.6	408.7
Shared Wirelength			
Top Layer M1 – M6	m	0	0
Bottom Layer M1–M6	m	1.86	1.01

Takeaway Of all the configurations considered, metal layer sharing is most prevalent in the memory on logic partitioned 3D IC due to the absence of standard cells in the bottom die and the sparse track usage by the intra-cell routing of the bottom die. F2B orientation is also crucial for enabling sharing of the tracks between the tiers. And finally, the F2B orientation also necessitates a monolithic integration of the 3D IC with its fine pitch value rather than hybrid bonding. By identifying the best configuration to analyze the track sharing between BEOLs, we move on to the main portion of the analysis which investigates the different pros and cons of track sharing in the following section 4.4.

4.4 Results

Every implementation of the designs in the current section, are chosen using a maximum frequency sweep between 1000 MHz to 1500 MHz. Among these frequency values, the design with the highest effective frequency is selected as our candidate for comparison. As Power Delivery Network (PDN) is a crucial aspect of the physical design, all the designs in this section are included with a similar PDN for comparability. The three designs used across all the below results are the commercial SoCs: Industry-A, Industry-B, Industry-C.

4.4.1 Baseline Experiments

3D Metal Layer Stack

Before starting the comparisons and analysis of metal layer sharing, we first set baseline designs that would be helpful in all the later discussions. To do so we need to first find the number of metal layers required to satisfactorily route the Logic-On-Memory design with the PDN. The preliminary analysis for various designs in section 4.3 uses the metal layer structure that is limited by the logic-on-logic partitioning which requires more metal layers due to the distribution and placement of memory macro across the two tiers. This is excessive for Logic-On-Memory partitioning as evident from the sparse routing in Figure 4.3.

Memory macros with the 28 nm technology use up to 4 metal layer (*M1 – M4*) for

internal routing. So at least 4 metal layers are required in the memory BEOL. But the F2B nature of the monolithic 3D ICs create additional routing issues (especially for PDN) based on the partitioning of the macros. Figure 4.5 shows an example where this issue arises. For the macros or macro pins under the large placement blockages in the top logic tier (here, another memory macro), the signal routing needs to be routed in a small channel towards area not blocked by logic tier placement. This significantly impacts the timing closure capabilities of the 3D IC. More significantly, power delivery becomes increasingly challenging with just four metal layer in the memory BEOL. In regions where memory-tier macros are not blocked by logic-tier macros in the X-Y plane, power delivery can be done using MIVs from the top tier directly on to the power rails within the memory macros without the need for additional distribution layer in the memory BEOL. But with memory macros on logic tier, the portion of or the entire macros that are covered by them in the memory tier cannot be supplied with power and ground supply. Because of this, we use up to 5 metal layers for the memory BEOL as our baseline.

The Table 4.5 shows the percentage of available tracks in the metal layers used by the inter-cell routing in the design. The available track length per each layer is calculated by subtracting away all the routing blockages due to memory macros, logic cells and other sources of obstruction to routing that are present in the design. The % tracks removed to due blockages is also shown in this table. In the memory tier, wee see $\approx 99\%$ possible routing area blocked by the memory macros on the layers M1–M4. Out of the remaining non-blocked tracks, metals M2, M4 have 15% or more usage while the usage in M1, M3, and even M5 is limited to under $\leq 2\%$. This mainly comes from the orientation of the macros in the memory tier (and the routing channels formed by these macros), and the preferred orientation for routing in these metal layers. In a vertical channels like the one formed on the memory tier of example Figure 4.5, the metal layers with vertical preferred directions have more usage as there can be longer uninterrupted wires in such areas. In the direction orthogonal to the routing channel (vertical direction, in this case), the available

Table 4.5: Metal Layer Usage of signal and power networks in the baseline 3D metal stack. Usage is calculated as the % of available tracks used for routing. Blocked Tracks is the % tracks blocked compared to total possible tracks in the footprint. Industry-A design is used for the following calculations

Metal Layer	Signal Usage	PDN Usage	Blocked Tracks
Memory Tier			
M1	0.0	0.0	98.8
M2	15.0	0.0	98.8
M3	2.0	2.0	98.8
M4	17.9	1.0	98.8
M5	1.3	9.4	0.0
Logic Tier			
M1	0.0	0.0	25.5
M2	20.4	21.5	20.1
M3	32.2	26.6	19.5
M4	25.8	18.2	19.5
M5	24.0	17.2	0.0
M6	14.8	16.1	0.0

routing tracks are much shorter making these routing channels under-utilized.

While the results in Table 4.5 are particular to Industry-A circuit, the overall signal usage trends are similar across the other circuits in consideration too. Unlike signal routing, Power Delivery Network is fully controlled by the inputs (layers used for PDN routing, width, spacing), and are kept constant across all the designs. This generates a PDN with similar metal usage across all the circuits.

Results

With the baseline setup established, we perform the Place and Route of the three Industry circuits (referred to as Ind-A, Ind-B, and Ind-C). As mentioned at the start of this section, the designs which show best frequency is selected to represent the baselines of each circuit. The total MIVs in the design is the sum of MIVs used for Power, Ground network (# PG MIVs) and the MIVs on signal nets (# Signal MIVs). The number of Power, Ground MIVs depend on the pitch value of the nets in the top and bottom tier metal layers, as well as the overall design footprint. In these designs, the PDN is identical and the varying PG MIV

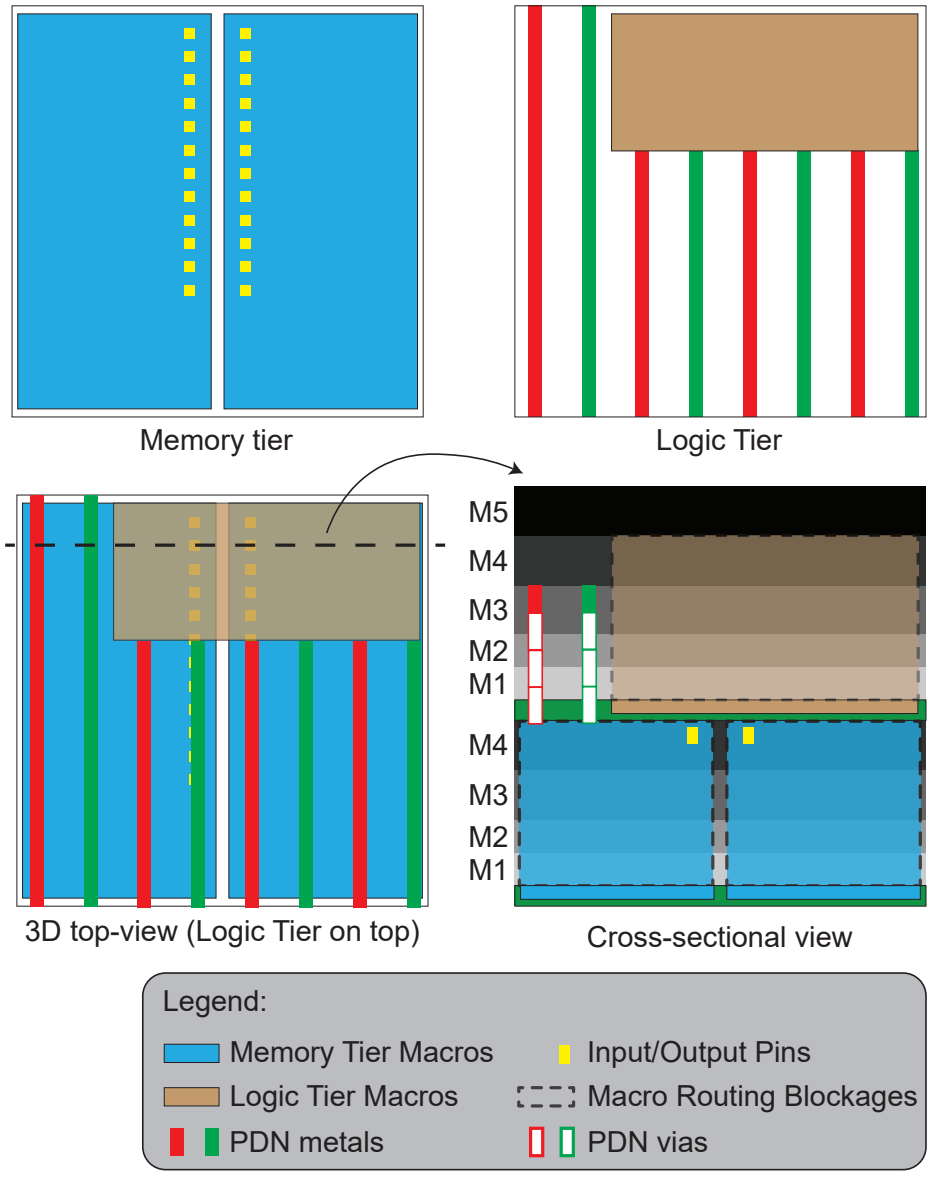


Figure 4.5: Partitioning scenario showing the obstructions caused by memory macros with just 4 layers in the bottom BEOL. The Cross-sectional view is shown at the cut-line of the 3D view

Table 4.6: Design metrics of the three RTLs considered in our work. The designs are implemented in a F2B 3D fashion. These are the baseline designs for further comparisons

	Units	Ind-A	Ind-B	Ind-C
Target Frequency	MHz	1500	1375	1375
Chip Area	mm^2	1.109	1.893	1.051
# Metal Layers	–	11	11	11
# PG MIVs	–	3620	5872	3222
# Signal MIVs	–	4157	1703	2282
# MIVs on 2D nets	–	83	94	78
# MIVs on 3D nets	–	4074	1602	2204
# MIVs on clk nets	–	91	94	96
# Routing Violations	–	1038	1201	1468
# 2D Nets	–	571650	790350	454312
# 3D Nets	–	3118	1311	1359
# Clock Nets	–	4866	6175	4280
Wirelength	m	12.214	19.438	10.541
Worst Neg Slack	ns	-0.352	-0.334	-0.286
Total Neg Slack	ns	-1648	-2287	-913
Effective Frequency	MHz	981.7	942.3	986.9
Effective Power	mW	592.3	730.5	444.9
Eff. PDP (Power×Delay)	pJ	603.3	775.2	450.8

count is strictly attributed to the difference in the footprints of the three circuits. Of the total signal MIVs, most of them are on the 3D nets which is the result of the MIV control explained in subsection 4.2.1. There exists a few MIVs on the 2D nets which is the result of the post-route optimization stage of 3D ICs. The routing constraints to control the MIVs on 2D nets are only applied to the nets present in the design during the MIV control. To solve this problem, we re-evaluate the constraints at each stage of 3D IC design. But, after the routing, constraints are only added to nets present during routing. And the nets added to the design by the automated timing optimization of the PnR tools, deletes some nets and add others. This last-stage optimization is the reason for the MIVs on the 2D nets in the design. In all the designs considered, the # MIVs on 2D nets is insignificant compared to the total number of 2D nets. As such, there shouldn't be any measurable impact on the PPA due to these nets, thereby acting as a good baseline to measure the impact of metal layer sharing.

Table 4.7: Metal Layer Usage of signal and power networks with the reduced metal layer stack with metal layer sharing. Industry-A design is used for the following calculation. All the calculations are done same as from Table 4.5

Metal Layer	Signal Usage	PDN Usage	Shared Usage
Memory Tier			
M1	0.0	0.0	0.0
M2	13.2	0.0	4.9
M3	3.6	2.4	2.2
M4	19.0	1.0	14.3
M5	13.2	9.4	12.3
Logic Tier			
M1	2.0	0.0	0.0
M2	21.9	21.6	0.0
M3	34.2	26.5	0.0
M4	29.3	18.5	0.0
M5	21.9	21.7	0.0
M6	unused		

4.4.2 Metal Layer Sharing and Cost Saving

Overall Track Usage

From Table 4.5 we have seen that although the metal M5 of the memory tier is necessary for power delivery, it is severely under-used for signal routing when metal layer sharing is not allowed. With no intra-cell routing in this metal, as evident by the lack of any routing blockages, it is helpful to use this layer for metal layer sharing. Furthermore, the top-most layer of the logic tier (M6) is also removed as the memory tier's M5 is additionally used for routing the logic tier nets. With this 5+5 metal stack, we can see a 1 Metal Layer reduction compared to the baseline. Although it is feasible to simply drop a metal layer in this case, the overall impact on the PPA should be verified.

Table 4.7 shows the track usages for all the metal layers with the reduced metal stack. The calculation of track usage is done in the same way as in Table 4.5, and since the macro placement is left untouched, the routing blockages are the same and are not mentioned. In addition, a new column for the borrowed track usage is added which signifies the metal layer sharing.

In the memory tier, we see that the track usage of the signals is fairly similar with a 2% increase for layers M2–M4. M5 of memory tier, which is the least utilized and blocked, shows the largest difference as its tracks are borrowed by the wires of the logic tier. Specifically, 12.3% of the tracks in memory tier M5 are used by logic tier. This mostly the additional wirelength rerouted from the M6 of top-tier (which has 14.8% track usage when metal sharing is not allowed).

In the logic tier, the track usage of M5 decreases as many long wires are preferentially routed on the memory M5. The track usage of the layers M1–M4 see a slight increase as they are rerouting the nets to the memory tier that would otherwise be on M6. Finally, M1 of logic tier is slightly more interesting as the usage without metal layer sharing was 0%. Because of the large prevalence of intra-cell routing in this tier, the router is discouraged from any additional routing. The only inter-cell routing is for pin-access for cells that do not have pin-shapes on higher metal layers, and the routing that is associated with pin-access. With metal layer sharing, M1 is needed for all the nets that borrow tracks from memory M5. 58 073 out of a total 561 669 nets in the design are shared 2D nets that are routed through logic M1 layer to access the memory M5. Even with additional routing, only 2% of the unobstructed tracks in M1 are utilized.

Routing Summary

The routing violations from Table 4.8 gives us a better picture of the routing quality than the track usage. Note that only 10 iterations are used for fixing the Design Rule Violations (DRVs) for all the designs considered. While using more DRV fixing iterations could have reduced the violations, we wanted to compare the type of violations that could be possible. With most of the designs having under 2000 violations, this is not a significant value compared to the entire footprint. On the memory M5, even with only a relatively lower track usage (12.3%) with metal layer sharing, we see a high DRV count. This can be mainly attributed to the routing layer setup and routing blockages in 3D, along with the

placement of MIVs.

To understand the reason behind these violations on M5, Figure 4.6 shows a zoom-in of routing in the metal M5 of the memory tier (layer with most metal layer sharing). From this, we see a few interesting aspects of the borrowed track routing using 5 metal layers in each tier. First, we see the MIVs being placed in the spaces left behind by the standard cells of the logic tier (shown in gray). When using metal layer sharing, the nets of top-tier are routed via an MIV to the bottom (memory) tier and up to the top (logic) tier via another MIV. This presents two challenges: First, since vias can only be located in certain places between the standard cells, we see more detours. Second, as the bottom tier has only 5 layers, of which 4 layers are heavily ($\approx 99\%$) blocked by intra-cell routing of macros, the routing is not always optimal. Once, the wires are routed through MIV on to the memory tier, the nets that have to be routed in both horizontal and vertical direction only have a single layer to do so.

This sub-optimality can be seen from the long jogs in the M5 routing. Table 4.8 also shows this from the % routing in the non-preferred direction values. Values for M1–M4 of memory tier can be safely ignored as they only have limited routing within the channels. M5 of memory tier has 3.5% of its wirelength routed in the non-preferred direction when metal sharing is turned on, this is relatively high compared to M3–M5 of logic tier under the same implementation. On the metal M1 and M2 of logic tier, the jog % is inherently larger due to the proximity to standard cells, and the short average wirelength. With metal layer sharing, M1 of logic tier is mostly used for routing to MIVs and has a high % of jogs. Even with the added complexity of the routing we are assured by the fact that only 112 violations remain in the logic M1 with metal sharing (lower than the baseline).

From the average wire segment lengths, we first see that the averages in the logic tier are always slightly higher in the metal sharing version. This comes from the added routing on these layers to perform metal layer sharing. On M5 metal of memory tier, the average wire segment length becomes shorter with metal layer sharing. This is mainly because of

Table 4.8: Routing Summary of the Industry-A design with different metal layer sharing options. The two columns correspond to the Industry-A columns in Table 4.9

Metal Layer	Without Metal Sharing (Baseline)	With Metal Sharing
# Design Rule Violations		
	Memory Tier	
M1	0	0
M2	0	0
M3	3	3
M4	3	3
M5	8	1097
	Logic Tier	
M1	144	112
M2	808	20
M3	71	13
M4	0	0
M5	1	0
M6	0	unused
Total Count	1038	1248
Average Wire Segment (in µm)		
	Memory Tier	
M4	6.60	2.92
M5	9.57	3.41
	Logic Tier	
M1	2.28	0.44
M2	0.54	0.56
M3	1.73	1.89
M4	3.43	4.47
M5	8.59	10.83
M6	20.00	unused
% Routing in Non-Preferred Direction (or) % Jog wirelengths		
	Memory Tier	
M4	0.2	0.6
M5	0.7	3.5
	Logic Tier	
M1	1.2	12.2
M2	6.0	5.2
M3	0.4	0.4
M4	0.3	0.2
M5	0.0	0.0
M6	0.0	unused

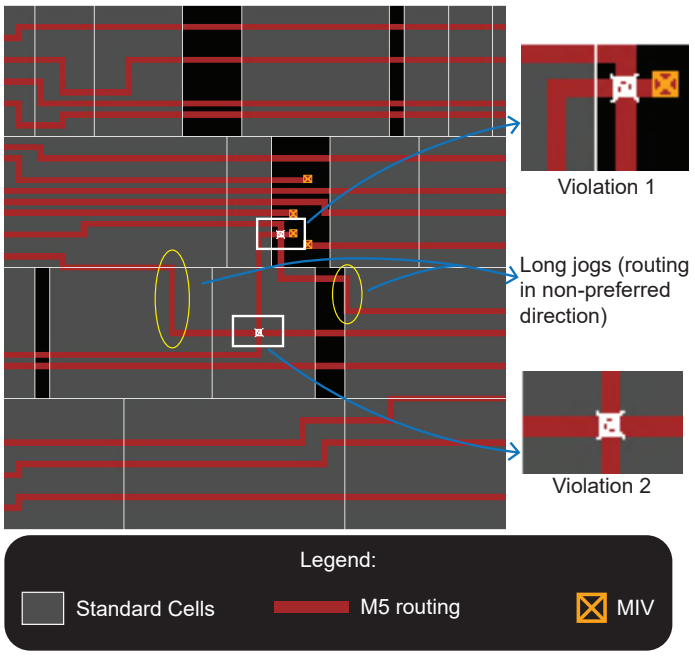


Figure 4.6: Zoom-in shot of M5 routing in the metal layer sharing design. We can see the routing jogs and shorts in this layer

the type of routing. In the baseline, only the long nets that need access to top-tier for 3D nets are routed on this layer. With metal sharing, a lot of logic tier nets access this layer as well the increased jogs add to the number of wire segments (an uninterrupted portion of wire in horizontal or vertical direction).

PPA Results

Finally, we compare the maximum performance of the three Industry circuits with the reduced metal layer stack and metal sharing. The values in Table 4.9 are a combination of $\Delta\%$ over the baseline as well as some raw numbers. The target frequencies correspond to the design which can achieve highest effective frequency similar to the baseline setup. We see a meteoric rise in the number of MIVs used for routing signal nets compared to baseline. AS the partitioning is left unchanged, the MIVs on 3D nets are very similar between the two cases. The MIVs on clock nets also see a large increase with nearly 1 MIV per clock net on average for the three circuits. We also see that, on average, the shared 2D nets as a whole use 2.66–2.98 MIVs per net depending on the design. The 3D nets on the

other hand, use much fewer MIVs per net between 1.18 and 1.82. This is in-line with our expectations, as the 2D nets that undergo metal sharing need to pass the 3D interface layer twice or more to be fully routed. Overall, there is only a negligible impact to wirelength with metal sharing compared to the overall design routing. Note that this is with one fewer layer in the logic tier.

When we compare the timing across the two, we see that the total negative slack at the target frequency is worse with sharing. This is due to the worsening of the nets that need to be shared. Although the most critical paths are not severely impacted as seen from the effective frequency, the non-critical paths are all slightly impacted leading to an overall total slack worsening. For the most critical path, except for the Industry-B design, both A and C circuits were able to reach a slightly better or similar frequency with one layer dropped. More importantly, the power benefit is significant for both Industry RTLs B and C, and the overall PDP is 5–7% better with metal layer sharing. *This shows the usefulness of metal layer sharing in efficiently utilizing the memory tier's M5, and to simultaneously drop the logic tier's M6 without negatively affecting the full-chip performance.*

4.4.3 Full-Chip Timing Improvements

Timing Improvement

To find the root of the PPA improvement, we focus our attention first to the Industry-A design. This shows a relatively large improvement in terms of maximum frequency which is solely responsible for the PDP improvement. To analyse, we look at few of the worst critical paths to analyze their behavior. These results are tabulated in Table 4.10 for the Industry-A design which shows the highest performance benefits.

The results presented here correspond to the Industry-A designs reported in Table 4.9. The values are reported at the max achievable clock, which is why the worst slack is 0 ps

Table 4.9: Max-performing design metrics of the three Industry RTLs with one fewer metal layer. For metrics reported as a $\Delta\%$, the absolute value is calculated w.r.t. the baseline designs in Table 4.6. A negative value for $\Delta\%$ implies the current design (one metal layer removed, and metal layers shared) performs worse than the baseline and vice versa.

	Units	Ind-A	Ind-B	Ind-C
Target Frequency	MHz	1375	1250	1250
Chip Area	$\Delta\%$	0.0	0.0	0.0
# Metal Layers	–	10	10	10
# Metal Layers	$\Delta\%$	9.1	9.1	9.1
# PG MIVs	–	1248		
# Signal MIVs	–	159045	312420	190302
# MIVs on 2D nets	–	154765	318074	187868
# MIVs on 3D nets	–	4280	1546	2434
# MIVs on clk nets	–	4221	6823	6720
# Routing Violations	–	1248	327	880
# 2D Nets	–	547745	745072	427412
# Shared 2D Nets	–	58073	106795	65235
# MIVs per Shared 2D Net	–	2.66	2.98	2.88
# 3D Nets	–	3162	1307	1337
# MIVs per 3D Net	–	1.35	1.18	1.82
# Clock Nets	–	4746	5761	4000
Wirelength	$\Delta\%$	-0.25	0.95	1.08
Worst Negative Slack	ns	-0.231	-0.309	-0.210
Total Negative Slack	ns	-1438.6	-1464.6	-668.3
Effective Frequency	$\Delta\%$	6.3	-4.31	0.32
Effective Power	$\Delta\%$	-0.71	11.29	7.26
Effective PDP	$\Delta\%$	5.25	7.30	7.56

Table 4.10: Timing Analysis of the Critical Paths and Clock Tree Results of Industry-A design

Metric	Units	Baseline	Metal Sharing
Effective Period	ns	1.019	0.959
Worst Slack	ns	0.0	0.0
Top-50 Register to Output Path Averages			
Path Delay	ns	0.585	0.644
Setup	ns	0.0	0.0
Skew	ns	0.417	0.271
Slack	ns	0.017	0.044
Path Length	µm	386	397
Logic Depth	–	3.8	3.4
Capture Latency	ns	0.4	0.4
Launch Latency	ns	0.817	0.671
Top-50 Register to Register Paths			
Path Delay	ns	0.801	0.885
Skew	ns	0.016	-0.040
Setup	ns	0.088	0.084
Slack	ns	0.112	0.031
Path Length	µm	458	510
Logic Depth	–	19.7	18.5
Capture Latency	ns	0.654	0.606
Launch Latency	ns	0.700	0.565
Clock Tree Results			
Maximum Latency	ns	1.060	0.952
Minimum Latency	ns	0.440	0.382
Average Latency	ns	0.680	0.746
Std. Deviation Latency	ns	0.066	0.064
Maximum Skew	ns	0.420	0.450

for both designs. Slack is derived as

$$\text{Path Slack} = \text{Clock Period} - \text{Path Delay} - \text{Setup} - \text{Skew}$$

We analyze the top-50 register-to-output and register-to-register paths in the design. By averaging over a larger number of nets than focusing on the single most critical path, we can isolate some of the eccentricities of a single path. As the two different path groups considered have very different characteristics, we separate them for analysis.

The register-to-output paths only have a portion of path in the chip as can be seen by the very small logic depth of around 3 – 4. When comparing the overall paths, we see that while the average path delay is marginally better in the baseline design, the main cause of worst period is the clock skew, and the clock tree synthesis. The capturing clock latency is the same between the two as it is an estimate based on the clock tree synthesis outside the current chip. The launch latency is significantly different between the two, as the baseline design has a worse latency on these critical paths. The results from the Clock Tree Synthesis further demonstrate this difference. The baseline design has worse overall clock latency which creates the bottlenecks on the critical paths. While the average latency is worsened with baseline, we see that the tool was able still able to do a good clock tree design as the standard deviation of the latency is similar among the two designs.

On the register-to-register paths, we see that the baseline has a better path delay as well as better slack. But, as the bottleneck paths in this design are the register-to-output paths, the slack on these paths is not helpful for the maximum frequency. In the metal sharing design, the average slack of this path group (0.031 ns) is similar to the average slack of the register-to-output path group (0.044 ns). This shows that the two path groups are much more closely distributed in terms of timing in the metal sharing design. Moreover, in the baseline, we see that the paths are shorter in length, but have more cells which is a sign of over optimization as long nets are broken down into shorter nets by adding buffers. This

Table 4.11: Energy Consumption per unit clock period at maximum frequencies for Industry-B design

Metric	Units	Baseline	Metal Sharing	Delta%
Effective Period	ns	1.061	1.108	4.43
Total Energy	pJ	775.2	718.6	-7.30
Internal Energy	pJ	391.7	360.3	-7.94
Switching Energy	pJ	382.3	357.2	-6.57
Leakage Energy	pJ	1.214	1.124	-7.41
Standard Cell Area	mm^2	1.023	0.931	-8.99
Wirelength	m	19.44	19.25	-0.98
Input Pin Cap	pF	2150	1909	-11.2
Wire Cap	pF	3084	2940	-4.67
Ground Cap	pF	2097	1988	-5.20
Coupling Cap	pF	996.8	951.6	-4.53

is required for the cells to meet timing in the bottleneck paths, which is more critical for baseline design.

Power Improvement

Two of the three designs discussed in Table 4.9 have improved PDP due to the power improvement resulting from the metal layer sharing. We analyze the two Industry-B implementations which show the highest power delta. Since power is a function of frequency, we report the power efficiency or the energy consumption per unit clock period (same as the PDP) in order to normalize the power and compare the overall trends. Splitting the total energy into internal, switching, and leakage in Table 4.11, we see that each component is reduced almost the same amount as the total energy.

Internal energy is the energy consumed within the cells, and has a high dependence on the total cell area. Note that we only used a single threshold voltage type (lowest V_{th} available) to keep the technology setup simple. The 9% reduction in cell area is the main reason for the internal energy reduction. The reduced cell area is a combination of better power and timing of the metal sharing designs. These designs reached their maximum frequency at a lower target period than the baseline in each case. Even with the lenient target, the maximum frequency reached is higher or comparable as seen from Table 4.9.

The combination of these two effects have resulted in a higher power efficiency as discussed throughout our work.

Further, the switching energy, which depends on both the routing and the cell sizing, also sees a significant reduction with metal sharing. The energy consumed due to a net switching is $\frac{1}{2}\alpha CV^2$, where α is the activity factor, C is the load capacitance (= wire cap + input pin cap), and V is the voltage of the signal on the net. With metal layer sharing, the only difference would be in the switching load (C), and some negligible differences in activity factor based on the design optimization. We see that the input pin cap reduces by 11% with metal sharing from the reduced cell area. The wire capacitance also reduces by $\approx 4.5\%$ even though the wirelength reduces by $< 1\%$. Factors such as wirelength distribution, and fanout can also affect the ground capacitance of wire. But more importantly, the unit capacitance per layer differs across the different layers of the 3D metal stack. For example, in the current 3D baseline metal stack, logic tier M5 has a capacitance of 0.155 fF/μm, logic tier M6 has 0.113 fF/μm, and the memory tier M5 has 0.112 fF/μm. So the difference in routing across the metal layers also leads to the difference in the wire ground capacitance. The cross-coupling capacitance between the nets is also reduced by 4.53% showing the improvement of routing by allowing metal layer sharing.

Leakage energy, like internal energy, is mainly dependent on the standard cells as well, and the 7.4% reduction stems from the cell are reduction. The overall power delay product improvement for the Industry-C design also follows the trend presented for Industry-B.

Takeaway Overall, we see that metal layer sharing can allow for efficient usage of metals. From using 12 metal layers for logic-on-logic 3D ICs, we were able to reduce the 3D stack to 11 metals with the help of logic on memory partitioning. Finally, the metal layer sharing was helpful to effectively use the memory layers and create further cost savings. This also came with an added PDP benefit which was presented in Table 4.9. Depending on the type of design, the improvements to clock network or the overall design in general was key to

generating the PDP benefit with metal sharing.

4.5 Conclusion

In this work, we have first analyzed the routing in various 3D IC types and found that metal layer sharing in 3D ICs is a novel phenomenon which can be controlled or enhanced in the designs based on user input. This layer sharing is only meaningful for the Monolithic 3D ICs due to the fine pitch and Face-To-Back nature of sequential fabrication. While metal sharing uses a large amount of MIVs, we see that they do not cause any routing issues in the design. Rather, they are helpful in effectively using the metal layers with a high quality routing using one fewer BEOL layer resulting in cost savings for 3D ICs. Even with the dropped metal layer, the timing and/or power consumption of the design improved with a Power Delay Product improvement of 5-7% across the three commercial processor designs.

CHAPTER 5

ON LEGALIZATION OF DIE BONDING BUMPS AND PADS FOR 3D ICS

An important shortcoming with the current state-of-the-art 3D flows is the routing stage. All the pseudo-3D flows, including the most recent Macro-3D and Pin-3D, assume a bump pitch in the order of 0.1–1 µm. Current research suggests that sub-micron pitch values for 3D bond pads are not easily realizable due to yield and manufacturability issues.

The combination of the smaller pitch values and the 28 nm process node used by authors in both [14, 13] obscures an important problem with automated 3D via placement. From Figure 5.1, we see how the 3D net routing can be impacted when a realistic Face-to-Face (F2F) bond pad pitch is used. The huge 3D via creates many routing violations with all the pseudo-3D flows used to design and optimize 3D ICs. In section 5.1, we further analyze this with the help of results from the design implementations.

In this chapter, we present and compare two different via legalization algorithms to remove the via overlaps produced by the commercial routers. In section 5.3, a force-based legalization algorithm is presented that displaces the vias to remove overlaps for hybrid bonding pads of any pitch and process node. Second, in section 5.4, we present an ML-guided bipartite-matching algorithm where the vias are optimally assigned to a grid of legal via locations with machine learning based parameter tuning. The ML-tuned bipartite matching is robust, close-to-optimal and applicable across different 3D partitioning types, technology nodes, and in both hybrid bonding and micro-bumping based 3D.

5.1 Motivation

The following terminology is frequently used throughout the paper. *3D via or cut:* F2F bond pad or micro bump. *Cut Spacing:* Edge to Edge spacing for a via. *Cut Distance:* Center to Center distance (L∞-norm) between two cuts. Given two vias centered at (x_1, y_1)

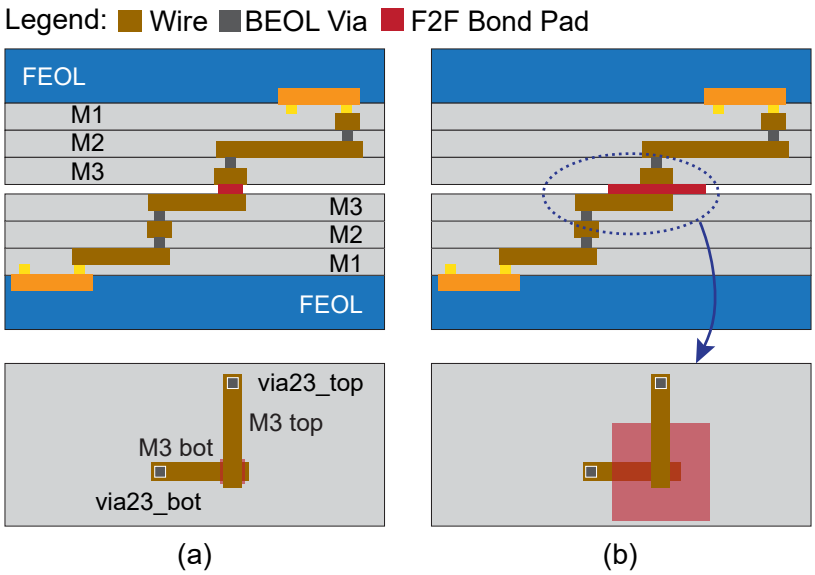


Figure 5.1: Using a commercial router to place face-to-face pads [13, 14]. (a) small F2F bond pad pitch, (b) large F2F pitch. The top-down views of the die interface are shown on the bottom.

Table 5.1: 3D Via Overlaps using the two state-of-the-art 3D flows and varying pitches.

Macro-3D [14]			Pin-3D [13]		
Pitch	#Overlaps	via_{util}	Pitch	#Overlaps	via_{util}
2 µm	2	1.1%	0.2 µm	0	1.3%
5 µm	1410	7.2%	0.5 µm	0	8.9%
10 µm	11080	28.2%	1.0 µm	5315	32.2%

and (x_2, y_2), the cut distance is $\max(|x_2 - x_1|, |y_2 - y_1|)$. *Pitch:* Minimum Cut distance defined in the technology files. *Cut Overlap:* Cut Distance ¡ Pitch, creates cut shorts and spacing violations

5.1.1 Via Overlaps with State-of-the-Art 3D flows

Impact of Pitch

With Macro-3D, a processor RTL is partitioned to have L2 and L1 Data RAMs on a memory tier, and everything else on a logic tier. The processor is implemented up to the CPU level (no L2 cache) using Pin-3D flow. This partitioning is done at logic gate level and results in a huge number of connections between the two tiers of the 3D IC.

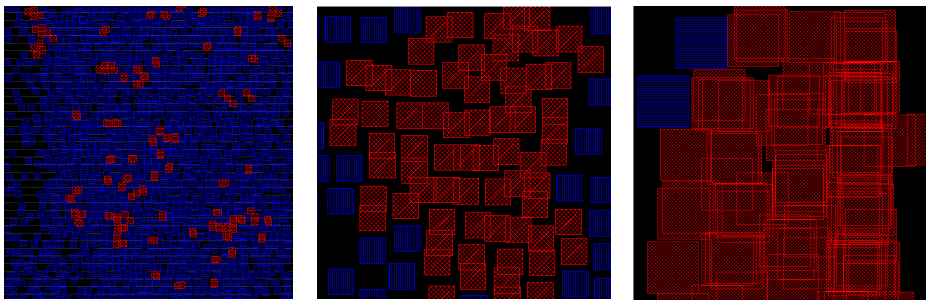


Figure 5.2: 3D Via overlaps (shown in red) with different flows and pitch values from Table 5.1. (a) Pin-3D [13] 1 µm pitch, (b) Macro-3D [14] 5 µm pitch, (c) Macro-3D [14] 10 µm pitch.

By sweeping the hybrid bond pitch used for 3D, we see the number of cut overlaps increase (given in Table 5.1). The 3D designs with Macro-3D and Pin-3D have 3000 and 50.000 3D vias, respectively, and Figure 5.2 show the various zoomed-in layouts and overlaps for the two flows. The routing stage in Pin-3D is similar to other pseudo-3D flows such as Compact-2D [12], and the automatically inserted 3D vias will have cut spacing and cut shorts in these flows as well.

Impact of Technology Node

Figure 5.3(a) show the difference in via densities of the same design implemented in different process nodes with Macro-3D flow. The number of 3D vias is $\sim$ 1200, and the 3D via pitch value is 5 µm for both cases. There can be at most 25 such vias placed legally in a 25 µm area, but we see that the 16 nm design contains bins with significantly more vias.

5.1.2 Source of Via Overlaps

In any commercial router, routing is separated into global and detail routing. During global routing, the entire footprint is separated into several global cell (gCell) grids, and nets are assigned to these grids. Detail routing then generates the exact physical routing solution of the nets, by assigning nets to the tracks within gCells.

During global routing, the router only considers the metal layer pitch for gCell assign-

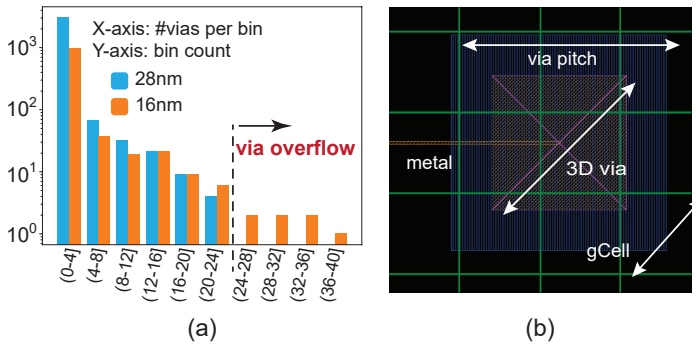


Figure 5.3: (a) Via distribution of a design in two different process nodes. Each bin is 25 µm × 25 µm, (b) Global cell grid (in green), 3D via, and metal layer in a 28 nm design.

Table 5.2: Comparing BEOL dimensions in the 28 and 16 nm nodes. The metal (Mx) layer is directly beneath the 3D via.

Metric	Unit	28 nm	16 nm
Mx pitch	µm	0.10	0.08
Via pitch below Mx	µm	0.10	0.08
3D Via pitch	µm	5.00	5.00
gCell width	µm	1.48	1.08

ment. But the 3D via pitch can be many times larger than the metal pitch and the gCell as seen in the Figure 5.3(b) with 28 nm node, and Table 5.2 for both 28 nm and 16 nm commercial process nodes. So, when various 3D nets are assigned to nearby gCells during global routing, the detail router ends up with significant routing changes and cut overlaps.

A naive way to solve this problem is to increase the gCell size. This increases the overall routing complexity exponentially and is not realistic. Even with larger gCells, the global router is not 3D via pitch, and doesn't resolve via overflow problem in gCells.

5.2 Bump/Pad Legalization Flow

In the state-of-the-art 3D flows [14, 13, 12], the implementation environment is aware of all metal layers the 3D entire stack and results in a better routing quality than a die-by-die implementation. We build the routing flow on this and modify it as shown in Figure 5.4.

The 3D nets are first routed to obtain an initial 3D via placement. The legal locations for the vias are then generated in the Via legalization step of Figure 5.4 using one of our

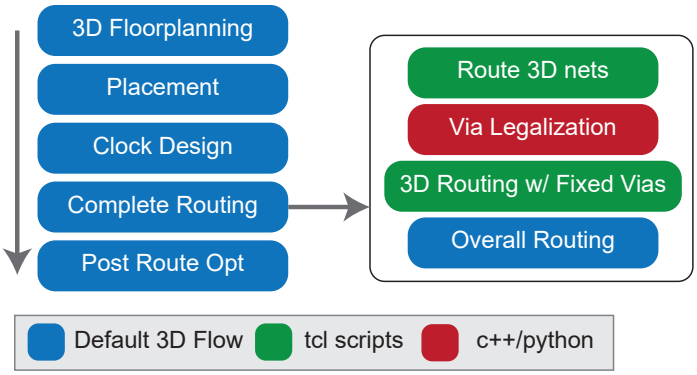


Figure 5.4: Design flow of a typical 3D IC design, and our modifications for inter-die pad/bump management.

two legalization methods. Force-solver is used for hybrid bonding pads only, and bipartite-matching can be used for both micro bumps, and bond pad assignment . The via positions are updated using the 'editMove' command capability supported by Cadence Innovus PnR tool.

A cut layer blockage on the 3D via layer is added to the discourage addition of any new 3D vias by the router, and the 3D routing is re-done to ensure full connectivity with the legalized vias. After routing the 3D nets, all the other nets are routed with the cut layer blockage intact.

5.3 Force-Based Via Legalization

Force-directed placement is a popular algorithm for cell placement in literature [25]. Inspired by this approach, we propose a force-based solution to remove the overlaps in the 3D via layer. Traditional force based global-placer calculate an equilibrium position by solving for the forces acting on each object. For simplicity and to get a detail placement, we choose a numerical approach of the force solver by incrementally moving the vias in small discrete steps.

The force-based legalizer starts with an initial solution from the commercial router which optimizes for various design metrics. At this stage, we suppress the violation fixing step of the router, and replace it with the following force solver.

5.3.1 Forces Utilized

For each overlapping via pair, we introduce two equal and opposite forces on the vias along the direction of the line joining their centers. To minimize the run-time, only the overlap neighborhood of a via is considered for force interactions. The blue rectangle in Figure 5.3(b) shows the overlap neighborhood of a via. The force $vs.$ distance relation is given as $(1+x)^{-p}$, where $x = \frac{distance}{pitch}$, along the two horizontal, vertical separately. A smooth force $F' \to 0$ as $x \to 1$ is not favorable as impact of vias close to the overlap boundaries are significantly reduced, requiring more iterations and run-time to remove overlaps. As the iteration progress, the power (p) is gradually decreased from 2 to 0.6 to increase the effect of vias further from center in the overlap neighborhood.

For each via, a small attractive force is also added that pulls a via towards initial location to reduce the maximum displacement of vias and reclaim some excess spacing between vias.

5.3.2 Overall Execution

The vias are treated as objects of mass m starting from rest, and with a resultant force F. In a time interval t, they moves a distance of $\frac{1}{2}at^2$ where $a = \frac{F}{m}$. So the displacement is $\propto F$, and the proportionality constant varies with m, t. The time t is fixed for every iteration, and the mass of vias are all assumed to be the same except for the vias marked as fixed by the router, and clock net vias. These are modelled as $10\times$ heavier than a normal via, and only move $0.1\times$ the distance under same force.

At the top level, the force solver is called multiple times until all the violations are removed. Instead of running the solver with the true pitch value, we start with a pseudo pitch value of 0.1 µm. In early stages, only the vias in a close neighborhood interact with each other, and are legalized up to the pseudo-pitch value. This value is then increased by $1.1\times$ until we reach the actual pitch value.

In each iteration of the force solver, we loop over each via as victim, and for each via,

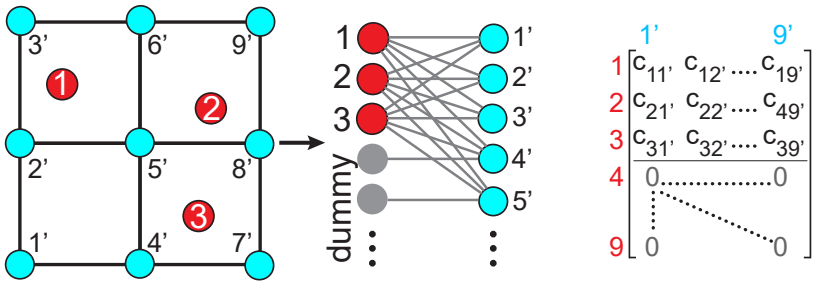


Figure 5.5: Our high-level grid assignment formulation. Vias (in red) and manufacturing grid points (in blue) are transformed into a bipartite graph, whose pairwise distances form the weight matrix, input to the LAP solver.

we visit all its neighbors. In worst case, the number of neighbors is $\propto n$, where n is the number of vias, and each iteration takes $\mathcal{O}(n^2)$. In reality, the number of neighbors is much smaller. With an average of m neighbors per via, each iteration takes $\mathcal{O}(n \cdot m)$. The number of iterations k varies based on the distribution, severity, and the number of via overlaps in the design. Overall, the run-time is $\mathcal{O}(k \cdot n \cdot m)$, or $\mathcal{O}(k \cdot n^2)$ in the worst case.

5.4 Bipartite-Matching Grid Assignment

The legalization problem can be cast into a combinatorial optimization problem of assigning vias on a grid evenly spaced with the via pitch. The grid line intersections form legal via placement points, and we find an assignment minimizing the total displacement from an initial solution by solving a minimum weighted bipartite matching problem. Unlike the force-solver, a grid-based assignment is crucial if the 3D via manufacturing grid differs from the design's.

Due to the many vias and available grid points, it is impossible to solve the problem directly in a reasonable time. Therefore, we propose a windowing technique tuned using Bayesian optimization, a machine learning method for black-box optimization, to reduce complexity and reach close-to-optimal solutions.

5.4.1 Algorithm

Legalizing vias to the grid while minimizing the total displacement (our cost metric) can be seen as an assignment problem in a bipartite graph. We want to uniquely match the set of vias $\mathcal{S}_V$ to the set of grid points $\mathcal{S}_G$, where the cost of matching a particular via v to a particular grid point g is proportional to their Manhattan distance D: $c_{v,g} \propto D(v,g) = |v_x - g_x| + |v_y - g_y|$, where (x, y) correspond to the 2D locations of the points. Typically, this is an unbalanced problem as $\mathbf{card}(\mathcal{S}_V) < \mathbf{card}(\mathcal{S}_G)$, which adds complexity, but we transform it into a balanced one by adding dummy vias with zero cost to all the grid points.

To solve the minimum cost (weight) perfect matching problem, we rewrite it as a linear assignment problem (LAP) as:

$$\begin{aligned} \min & \sum_{v,g} c_{v,g} x_{v,g}, \\ \text{s.t.} & \sum_{g} x_{v,g} = 1, \quad v \in \mathcal{S}_V, \\ & \sum_{v} x_{v,g} = 1, \quad g \in \mathcal{S}_G, \\ & x_{v,g} \geq 0, \quad v \in \mathcal{S}_V, \quad g \in \mathcal{S}_G, \end{aligned} \tag{5.1}$$

where $x_{v,g} = 1$ if $(v, g) \in M$ and 0 otherwise. We solve this problem using the shortest augmenting path algorithm [26]. Figure 5.5 depicts the transformation of the geometric problem to LAP in a matrix form, input to the shortest augmenting path algorithm.

Timing Considerations

Restricting the legalization displacement of 3D vias on the clock signal is crucial to minimize possible PPA degradation. Similarly, vias associated with unconstrained nets are not as critical as the other vias. We propose to weigh the matching cost by the timing-criticality of the connected net based on the net type and static timing analysis. We use a standard net/via weight factor used extensively in timing-driven placement [27]. Per via v, we ex-

Algorithm 2: Windowed Bipartite Matching Algorithm

Data: (x, y) locations of 3D vias from routed design;
floorplan; 3D Via pitches along X, Y; window definition;
Result: A via assignment on the manufacturing grid minimizing the total timing-driven displacement cost;
for $w \in$ *Windows* **do**
1. Query vias $\in w$ and build grid in that window;
2. Compute pairwise distances, and multiply with pre-computed timing weights to obtain the cost matrix;
3. Solve the LAP with the shortest augmenting path algorithm [26];
4. Apply the assignment solution: update locations of vias and recompute query matrix;

tract the worst timing path through v and define the weight based on the obtained slack and data arrival time as

$$w(v) = \begin{cases} 2^{\alpha} & \text{if clock net,} \\ \epsilon \ll 1 & \text{if unconstrained net,} \\ 1 & \text{if slack}(v) \geq 0, \\ \left(1 - \frac{\text{slack}(v)}{\text{arrival}(v)}\right)^{\alpha} & \text{otherwise,} \end{cases} \tag{5.2}$$

where α is the criticality exponent (=2 in our experiments). The new LAP formulation is then updated to use $c_{v,g} = w(v) \cdot D(v, g)$.

Window Sliding

The shortest augmenting path algorithm exhibits a time complexity of

$$\mathcal{O}(\max(\mathbf{card}(\mathcal{S}_V), \mathbf{card}(\mathcal{S}_G))^3)$$

The cost matrix itself has a space complexity of

$$\mathbf{card}(\mathcal{S}_V) \times \mathbf{card}(\mathcal{S}_G) \times 8$$

Table 5.3: The six windowing parameters tuned with machine learning. The 3D via pitches are noted as p_x, p_y.

Name	Unit	Type	Range	Default
Tile width	µm	float	$[5p_x, 100p_x]$	$30p_x$
Tile height	µm	float	$[5p_y, 100p_y]$	$30p_y$
Window size x	$tile_{width}$	int	[3, 10]	3
Window size y	$tile_{height}$	int	[3, 10]	3
Stride x	$tile_{width}$	int	[1,3]	2
Stride y	$tile_{height}$	int	[1,3]	2

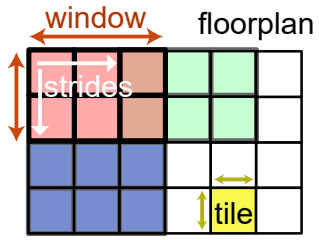


Figure 5.6: Divide-and-conquer using a sliding window. In each window, the grid assignment problem is solved optimally.

, where 8 corresponds to the number of bytes to encode *float64* weights, easily exceeding the modern RAM capacities.

Therefore, we propose a tiling/windowing method for 2D floorplan/space partitioning to reduce matrix size and solve the LAP locally in each window, following the algorithm 2. This window is slid over the 2D floorplan, similar to a 2D convolution filter. Figure 5.6 shows an example windowed floorplan. Each window contains only a few thousand vias or grid points for appropriate window sizes, and we update the via locations based on the found assignment. Moreover, to counteract the non-optimality introduced by the local solutions, we use striding to allow reassignment of previously derived locations if it reduces the total displacement.

5.4.2 Machine Learning Tuning

The quality of the assignment significantly depends on the values of the windowing parameters presented in Table 5.3. These correspond to the window configuration of Figure 5.6. For example, for a small problem size, the window can be defined to include the entire

floorplan, and an optimal solution can be found directly. However, these parameters need to be tuned for more complex problems to obtain a near-optimal solution within a reasonable runtime.

This objective is realized in the maximization of the following:

$$f(p) = w_C \tanh\left(\frac{C_0}{C(p)}\right) + w_D \tanh\left(\frac{D_{max_0}}{D_{max}(p)}\right) + \tanh\left(\frac{T_0}{T(p)}\right),$$

where p denotes the parameter settings, C denotes the total timing-driven displacement cost, D_{max} is the maximal displacement, and T is the runtime. We integrate the maximal displacement as it reflects the maximum deviation from the router's initial decision. We set $f(p) = 0$ if the LAP solver crashes due to a runtime exception from the inability to allocate enough memory for the cost matrix or shortest path algorithm. The reference values subscripted with 0 are set based on the default parameter values. The application of the $\tanh$ is to squash the differently scaled metrics into $[-1, 1]$ and make them comparable. The weights of each component can be set to realize different trade-offs of optimality vs. speed.

To maximize this objective, we use Bayesian optimization [28]. The Bayesian algorithm sequentially queries the function f and builds a surrogate function interpolating the evaluations. We use the Gaussian process as a surrogate family, with a squared exponential kernel. Moreover, we use the Upper Confidence Bound (UCB) acquisition function to pick the next candidate query point. After multiple iterations, we report and use the assignment that maximized the presented objective function. Table 5.4 shows the positive effect of the tuning on the maximal displacement.

Implementation We implement the flow in Python, based on Numpy vectorized features, and accelerate the cost matrix calculation with multithreading and SIMD through Numba just-in-time compilation. Moreover, to speed up the query of points in a given window, rather than using traditional 2D spatial query data structures, such as quadtrees or KD-trees,

Table 5.4: Displacement metrics before and after ten Bayesian optimization iterations. The design has $\sim$ 6000 vias to be legalized on a 5 µm pitch grid. The weights are $\mathbf{w_C}$=20, $\mathbf{w_D}$=10.

Metric	Unit	Before Tuning	After Tuning
Total cost		9469.1	9249.9 (–2.3 %)
Max displacement	µm	21.7	16.2 (–25.3 %)
Runtime	s	1.76	4.02

we store the indices of the list of vias in a 2D matrix Q where $Q[i][j] = \{vias \in \text{tile}(i, j)\}$. Using this matrix Q to query points is much faster than KD-trees due to the regular memory accesses. Moreover, every time via locations are changed, the matrix is quickly updated locally. Bayesian optimization is done using [29].

5.5 Results And Analysis

5.5.1 Experimental and Technology Setup

Technology Setup

To test the efficiency and the applicability of our via legalizer, we use two commercial PDKs: 28 nm node, 16 nm node. Along with the process nodes, the following bonding styles and pitch combinations are also tested: A micro-bump based 3D IC with 20 µm, 10 µm pitches, and hybrid-bond based 3D IC with 5 µm, 1 µm pitches.

PnR Flows

The micro-bump 3D ICs are designed using a die-by-die flow by pre-assigning the bump locations during 3D floorplanning stage. Without an initial routing solution to use in displacement minimization, we start by assigning the bumps to the center of macro pins connected by each 3D net. The displacement is minimized from this initial solution, and the bumps are assigned on to a bump grid with bipartite-matching. The hybrid bond flows are implemented using Macro-3D flow for memory-on-logic partitioning, and Pin-3D flow for the logic-on-logic partitioning as discussed in section 5.2.

Partitioning types and Circuits

For the memory-on-logic partitioning we implement the following circuits: 1. A dual core application processor (AP1) with 512 kB of L2 cache implemented in the 28 nm node. The memory tier contains the L2 and L1 data cache. 2. A single core processor (AP2) with 1 MB of L2 with 28 nm PDK. Only the L2 data cache in memory tier with a cut-size of $\sim$ 1500. 3. A slight variation of AP2 (referred to as AP2.1) with 512 kB is implemented in the 16 nm PDK due to the memory size restrictions of Macro-3D.

The following circuits are designed with logic-on-logic partitioning using the Pin-3D flow: Application processors AP1, AP2 without the L2 cache (referred to as AP1cpu, AP2cpu respectively) in 28 nm node. And, a Neural Processing Unit with 128 MACs (NP1) at the 16 nm node.

5.5.2 Application in different types of 3D ICs

Hybrid Bonded 3D: Memory-On-Logic

The memory-on-logic designs with hybrid bonding are given in Table 5.5. For the three designs considered, a hybrid bonding pitch of 5 µm is used, with an equal width and spacing of 2.5 µm each.

We see from Table 5.5 that both force and bipartite-matching algorithms were able to remove almost all of the spacing violations in the three cases studied. The overall wirelength is largely unaffected, as only a few 3D nets exist with memory-on-logic partitioning. The number of vias are smaller in the force, bipartite methods as the modified routing flow in Figure 5.4 suppresses the usage of 3D vias by nets other than 3D nets.

Variations in the via assignment pattern

We see from Table 5.7(b) that the via placement resulting from the force-based legalizer results in a more spread out solution than Table 5.7(c). This is also supported by com-

paring the max displacement metrics in Table 5.5. In addition to the denser packing with bipartite-matching, aligning the vias to a regular grid can improve the yield and complexity of 3D bonding. Force legalization on the other hand, only snaps vias to the smallest design manufacturing grid, and requires higher alignment accuracy. Even with the grid alignment constraint, bipartite solution was able to match or improve upon the displacements from force based solution due to the ML tuning and local optimality of our bipartite-matching solution.

Micro Bump 3D

Table 5.6 shows the results of the three designs implemented with the micro bump bonding assumption. As micro bump bonding flows generally require vias to be placed on a custom grid rather than design grid, force-based legalizer cannot be applied. So we compare the 3D bump assignment of bipartite-matching with a simple timing priority-based greedy assignment. A bump pitch of 20 µm is used for the AP2 benchmark, and a 10 µm pitch for AP1, AP2.1 due to the smaller footprints.

A greedy approach creates large displacements for bumps with lowest assignment priority, and is reflected in the max displacement values in Table 5.6. The optimal placement with the timing-weighted bipartite-matching solution provides a much better result. We see a significant improvement in the Total and Worst Negative Slacks with bipartite-matched assignment compared to a greedy solution. This shows the robustness of our proposed solution to both hybrid-bonding, and micro-bumping 3D designs.

Hybrid Bonded 3D: Logic-On-Logic

Finally, Table 5.7 shows the results for via legalization in logic-on-logic partitioning with a 1 µm 3D via pitch. With a huge 3D via count, we see that the legalization starts to degrade the design quality as the 3D interconnect complexity increases. Having such a high via utilization is also not very good in terms of manufacturability and yield due to

reduced redundancy. Therefore in such extreme cases, it is required to redesign the routing stage rather than simply adding a legalizer step. Logic-on-Logic designs are also not very feasible with advanced process nodes due to their extreme pitch requirements.

Takeaways In designs with large via counts and via utilization, our legalizers can start to slightly degrade the routing quality. Such cases might anyway require a much finer bonding pitch values that are not practical in the near future, and necessitate a fully integrated routing solution to provide a good PPA quality with clean DRCs.

With more realistic partitioning and 3D bonding, our force based legalizer can be extremely useful when only a few vias needs to be displaced without affecting the overall via placement. But more importantly, our timing-weighted bipartite-matching solution is extremely versatile and applicable for a wide range of 3D pitch values, bonding styles, and partitioning types. Compared to simpler and more traditional approaches like a force based legalization or a greedy bump assignment, the ML-guided bipartite-matching legalization has consistently outperformed in terms of maximum and average via displacements, as well as the overall PPA.

5.6 Conclusion

In summary, we have shown that when the 3D via pitch becomes comparable or larger than the global cell grid, commercial detail router fails to create a good 3D via assignment with cut short and spacing violations. Fixing these violations early with our proposed legalizer techniques can create a better routing quality with fewer DRVs, better total slack, with only a negligible run-time impact.

Table 5.5: Via Legalization results of memory-on-logic 3D ICs with hybrid bonding (pitch of 5 µm)

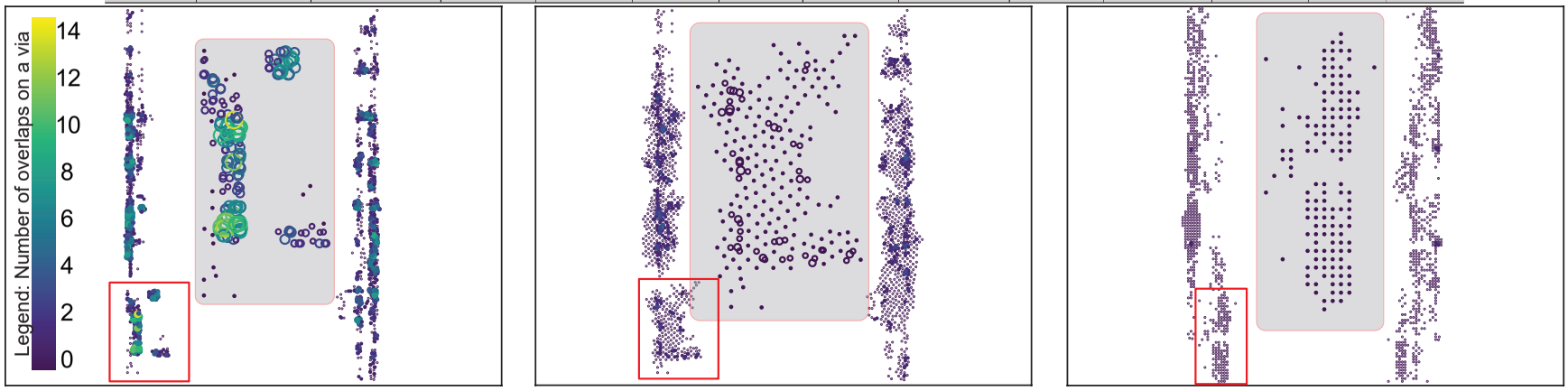
			Physical Stats				Timing and Power			Legalizer Stats			
			Freq.	Area	WL	via_{util}	WNS	TNS	Power	#Vias	#Viols	d_{max}	d_{avg}
Node	Circuit	Legalizer	(GHz)	(mm^2)	(m)	(%)	(ns)	(ns)	(mW)			(µm)	(µm)
28 nm	AP1	Default	1.25	1.11	11.62	11.3	-0.119	-202.1	677.0	5014	3538	-	-
		Force	1.25	1.11	11.76	7.4	-0.091	-113.7	678.7	3246	0	29.9	5.3
		Bipartite	1.25	1.11	11.75	7.4	-0.094	-120.4	678.5	3246	0	31.6	3.0
28 nm	AP2	Default	1.25	1.89	18.92	3.1	-0.251	-681.1	894.7	2310	893	-	-
		Force	1.25	1.89	18.89	1.8	-0.240	-605.5	895.5	1343	0	12.4	1.3
		Bipartite	1.25	1.89	18.89	1.8	-0.244	-578.9	894.7	1343	0	6.8	2.4
16 nm	AP2.1	Default	1.60	0.605	9.439	5.2	-0.011	-0.057	843.6	1272	2339	-	-
		Force	1.60	0.605	9.455	5.2	-0.013	-0.055	843.7	1243	308	44.2	10.6
		Bipartite	1.60	0.605	9.584	5.2	-0.002	-0.002	844.6	1269	10	15.0	2.8

Table 5.6: Via Legalization results of memory-on-logic 3D ICs with micro bumping (10 µm pitch for AP2, 2.1; 20 µm for AP1)

			Freq.	Area	WL	via_{util}	WNS	TNS	Power	#Vias	#Viols	d_{max}	d_{avg}
Node	Circuit	Legalizer	(GHz)	(mm^2)	(m)	(%)	(ns)	(ns)	(mW)			(µm)	(µm)
28 nm	AP1	Greedy	1.25	1.11	11.56	53	-0.118	-25.15	796.6	2957	0	601.0	127.5
		Bipartite	1.25	1.11	11.41	53	-0.014	-5.04	794.4	2957	0	385.1	68.1
28 nm	AP2	Greedy	1.25	2.18	20.06	43	-0.323	-1336.4	912.1	1187	0	435.7	160.6
		Bipartite	1.25	2.18	19.92	43	-0.295	-1237.6	907.9	1187	0	198.1	79.3
16 nm	AP2.1	Greedy	1.60	0.61	10.39	39	-0.155	-881.7	1043	1169	0	326.6	121.1
		Bipartite	1.60	0.61	9.92	39	-0.140	-835.8	1006	1169	0	102.2	44.7

Table 5.7: Via Legalization results of logic-on-logic 3D ICs with hybrid bonding (pitch of 1 µm)

Node	Circuit	Legalizer	Freq. (GHz)	Area (mm^2)	WL (m)	via_{util} (%)	WNS (ns)	TNS (ns)	Power (mW)	#Vias	#Viols	d_{max} (µm)	d_{avg} (µm)
28 nm	AP1cpu	Default	1.50	0.25	3.643	23	-0.019	-0.450	350.5	58039	5315	-	-
		Force	1.50	0.25	3.709	21	-0.014	-1.057	351.5	53416	554	5.9	0.6
		Bipartite	1.50	0.25	3.698	21	-0.023	-2.220	350.7	53347	582	4.5	0.5
28 nm	AP2cpu	Default	1.50	0.59	9.32	36	-0.032	-15.8	884.3	214123	19992	-	-
		Force	1.50	Not Feasible due to high density of vias, and large number of overlaps.									
		Bipartite	1.50	0.59	10.05	38	-0.122	-438.1	915.0	222465	1900	3.1	0.5
16 nm	NP1	Default	1.60	0.12	2.61	40	-0.466	-276.2	390.5	57936	33971	-	-
		Force	1.60	0.12	3.01	36	-0.782	-1157.4	401.5	51414	19573	31.9	6.4
		Bipartite	1.60	0.12	2.77	38	-0.853	-854.5	397.7	54619	17378	28.6	0.9



(a) No legalization (b) Force-based legalization (c) Bipartite matching legalization

Figure 5.7: Die bonding (5 µm pitch) via legalization for the AP2.1 benchmark implemented with 16 nm node. Size and color of each via represent the number of overlaps. The gray rectangle shows the zoom-in of the highlighted vias in the red box.

CHAPTER 6
A LOGIC-ON-MEMORY PROCESSOR-SYSTEM DESIGN WITH MONOLITHIC 3D TECHNOLOGY

In this work, we use two different configurations of the RISC-V based OpenPiton tile [30], as our benchmark architecture: a memory-heavy case; and a second case with smaller memory capacity. We present logic-on-memory partitioning for M3D ICs for the two different memory architectures, and show drastic PPA improvement of M3D over the respective 2D designs for a commercial 28nm PDK. Furthremore, a detailed analysis outlines the cause of performance, power and routing improvement of the logic-on-memory designs.

6.1 Monolithic 3D Integration

6.1.1 Logic-on-Memory Monolithic 3D Partitioning

The high demand for processor-to-memory bandwidth has become a challenge for modern computer systems. 3D integration of processor and memory is a promising solution to improve the memory bandwidth from the physical perspective because it reduces the wire delay between the processor and the memory by replacing long 2D interconnections with shorter 3D interconnections. Logic-on-Memory is a special structure of 3D integration as it separates the logic gates and memory blocks into different dies which allows them to be fabricated with different technologies.

There have been a lot of studies on 3D memory stacking with various 3D integration technologies [2][31][32]. Major silicon companies such as AMD and Xilinx are also using 3D integration techniques to improve the performance of memory in their next-generation products [33]. However, most of the studies use TSV-based or face-to-face bonded 3D integration technology which provide limited and predefined 3D interconnections. M3D

on the other hand, provides more flexible 3D interconnections and potential benefits on routing and clock tree optimization. In this paper, we will explore the impacts of logic-on-memory 3D integration on the performance of a RISC-V-based processor-system.

6.1.2 RTL-to-GDS Tool Flow For Monolithic 3D ICs

One of the challenges facing M3D ICs is the lack of commercial tools to perform Placement and Routing (P&R) in the three-dimensional space. Currently available commercial tools only support placement in a single 2D plane restricting their use in designing 3D ICs. A 3D placement should use the silicon area on both the top- and bottom-dies to optimize the design. A variety of flows have been developed which make use of 2D commercial tools along with various heuristics to achieve a 3D placement [34][35][36].

Shrunk-2D [34] is the first RTL-to-GDS flow developed to design commercial quality 3D ICs from RTL using the design optimization capabilities of 2D P&R tools. Compact-2D [35] flow has an added ability to perform complete routing and timing-optimization in 3D. Cascade-2D [36] performs architecture-based 3D placement. It also supports the complete routing and timing-optimization for 3D.

A poor partitioning choice undermines the benefits of 3D ICs and architecture-based or heuristic-based placement should be done carefully. In this work, we make use of an extended version of the Shrunk-2D flow tailored for P&R in logic-on-memory design. However, details on the EDA flow are outside the scope of the present publication.

6.2 Experimental Setup

6.2.1 Benchmark Architecture

We use OpenPiton[30], an open-source multi-core processor system and framework, as the benchmark architecture. It is highly configurable which makes it possible to change the core count, cache sizes, etc. The OpenPiton many-core system is shown in Figure 6.1(a). A full system consists of one or more chips and according chipsets, while chips are made

up of multiple tiles. Thus, a tile is an atomic piece out of which systems of arbitrary size are constructed. Hence, we only analyze the design of the tile while ensuring a correct functionality/timing when multiple tiles are later instantiated to create larger systems (more details on the resulting constraints in subsection 6.2.2). Thereby, we report results valid for systems with arbitrary tile-counts.

The tile structure along with the bit-widths for data-flow is illustrated in Figure 6.1(b). It consists of a 64-bit Out-of-Order (OoO) RISC-V Ariane core and three levels of cache (L1–L3). The first two levels, L1 and L2, are private to the individual cores, while the third level is shared cache-coherent among all cores of the system. Thus, the physical memory of the shared L3 cache is distributed evenly among the tiles of a system. Network-on-Chip (NoC) routers are used for the communication to provide good scale-ability up to hundreds of cores/tiles.

Two variants of the tile are analyzed, which differ in their memory capacities. Case-I is a memory-heavy case with 16 kB of L1 Instruction cache, 16 kB of L1 Data cache, 128 kB of L2 cache, and 1 MB of L3 per tile. In Case-II, smaller memories are used with 8 kB of L1 Instruction cache, 16 kB of L1 Data cache, 16 kB of L2 cache, and 256 kB of L3 cache per tile. The memory macros occupy way more than 50% of the area in both cases, showing the suitability of logic-on-memory integration not only for memory-heavy systems. The memory-macro floorplans for the 2D and M3D designs are shown in Figure 6.2(a)–(f).

6.2.2 Design Setup

In the tile design, the complete inter-tile NoC interconnection must be captured through constraints as according paths start in one tile and end in another. For example, consider a NoC path starting in one tile-instance and ending in the north adjacent tile-instance. This path is represented in the tile-design by a path starting at an NoC-output-register and ending at a North-output pin, combined with a path starting at a South-input pin ending at an NoC-input register. Thus both paths together have to finish in one clock-cycle and the north-

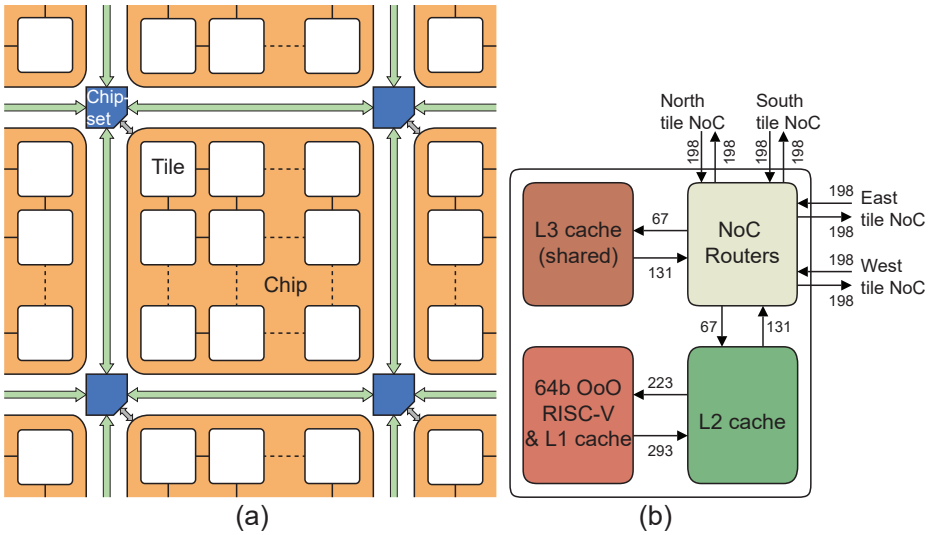


Figure 6.1: OpenPiton architecture (a) full system (adopted from [30]), (b) single tile with data-flow width.

output and the according south-input pin locations have to be aligned in a way that multiple instances of the tile can be connected without additional routing. Thus, in our tile-design input-to-NoC and NoC-to-output paths are constrained with a half-clock-cycle delay, all North-South output/input pin-pairs are horizontally aligned, all East-West output/input pin-pairs are horizontally aligned, and all pins are located in Metal 3. This ensures timing closure for full systems with arbitrary tile counts. The frequency achieved per single-tile is used as the performance metric for the OpenPiton system in this paper.

The power is calculated statistically using an input toggle rate of 0.1 and a flip-flop toggle rate of 0.2. The Power—Delay Product (PDP) is the energy consumption of the design per clock-cycle. The Energy—Delay Product (EDP) metric has a quadratic dependency on clock-period as it is the product of clock-period and the PDP.

In all the tile designs, a worst negative slack (WNS) below 3-4% of the clock-period time is considered to be within the noise limit and is used as the timing-met condition.

6.2.3 Technology Setup

Using a commercial 28nm PDK, the timing closure and analysis are done at the typical PVT corner at 0.9 V and 25 °C. Six metal-layers are used for routing in the 2D designs. For

M3D designs, we use six metal layers per die. Since each metal layer in M3D has half the XY-area of a 2D metal-layer, the overall available metal-layer area is equal for both 2D and M3D designs.

For M3D, routing-technology files are created for the M3D metal-stack used (6 metal-layers per die). The metal-layer stack of each die resembles the 2D metal-stack. A new cut-layer is added for the MIVs between the top metal of bottom-die and bottom metal of top-die. The MIV cut-layer passes through the transistor layer of top-die. This creates placement obstructions between MIVs and cells on the top die. A width and spacing of 70 nm each is used for the MIVs. For comparison, this MIV is ∼430x smaller in area than the smallest inverter in this technology. Each MIV has R=2 Ω and C=0.02 fF. In comparison, a normal routing via connecting the metal layers M1 and M2 has a footprint of 50 nmx110 nm with R=8 Ω and C=0.02 fF. Possible performance degradations due to the low-temperature M3D manufacturing are neglected as the transistors and metal-layers on both tiers are assumed to be identical.

6.3 Design and Simulation Results

6.3.1 GDS Layouts

The memory layouts presented in Figure 6.2 are chosen as they minimize the distance between closely connected blocks and allow good memory-pin access for standard cells. In logic-on-memory M3D design, standard cells are placed only on the top-die (logic-die), leaving the bottom-die (memory-die) for macros. This arrangement is critical for the M3D design as huge macro-cells on the top-die create huge obstructions for MIV insertion and restrict the MIV placement to the small channels between the memory macros. This setup would create routing congestion and increases the wirelength (WL) of the design. With the standard cells on the top-die, the MIVs can pass through the spacing between standard cells present throughout the top-die, allowing the router to place the MIVs throughout the die with only small obstructions of the standard cells.

Figure 6.3(a)–(b) show the routing results on 2D and M3D designs of Case-I OpenPiton tile. Thereby, routes of different metal-layer are represented by different colors.

6.3.2 Analysis

Max-Performance analysis

The max-performance results of all the 2D and M3D designs of the OpenPiton tile are presented in Table 6.1. In Table 1, $\Delta_{\text{M3D}} = \frac{M3D-2D}{2D} * 100\%$. Compared to the 2D design, Case-I OpenPiton tile achieves 36.8% higher performance with logic-on-memory M3D integration Table 6.1). A closer look at the delay numbers on the critical paths in 2D and M3D helps understand the difference between these max-performance designs. The total delay of the critical paths are consisted of cell delay, wire delay, pin delay (added to capture inter-tile communication), and launch latency of the clock. As mentioned before, paths that end/start in an adjacent tile are assigned a delay equal to half the clock-period. Thus, only half clock-period is available for a flop-to-pin path and the other half clock-period is left for the pin-to-flop counterpart. While the 18.3% latency decrease is on-par with the clock-period reduction of 26.3% between 2D and M3D, the main difference is the drastic reduction in wire delay portion of the critical path.

Although using NoCs reduces the interconnect bottleneck in many-core systems, the longer global wires for the NoC-based tile-communication are still found out to be the system's performance bottleneck as they heavily contribute to the critical path in 2D. M3D helps to overcome this bottleneck efficiently as long global wires are shortened in M3D. We observe that the wirelength of the critical path in M3D is 37.9% less than the critical path in 2D, giving rise to a 61.5% lower wire delay in M3D compared to 2D. In the 2D design, wire delay makes up 32.5% of the half clock-period available, whereas in M3D the wire delay is only 16.7% of the half clock-period. Further timing analysis is done in subsubsection 6.3.2. The energy of the M3D system is nearly equal to the 2D energy. As the average wirelength per net is only 9.8% smaller in the logic-on-memory M3D design,

standard cells drive similar amount of wire-load in both 2D and M3D. So a larger cell area is needed to meet timing with faster clock. This results in the power increase in M3D. Thus, the drastic performance improvement in M3D is obtained at a cost of power increase, nullifying the effect of M3D on the PDP.

In the Case-II OpenPiton design with smaller memories, we see a performance improvement of 35.0% which is on-par with the Case-I design. The footprint in Case-II 2D design is nearly $^1/_3{}^{rd}$ of the Case-I 2D design and is also smaller than M3D design in Case-I. The critical path in Case-I 2D is still between tiles, but the maximum wire-length (WL) and the wire delay portion of the total delay are smaller than 2D Case-I due to the smaller footprint of the tile. In M3D Case-II, we see that the critical path is no longer a tile-communication path, but a memory-to-memory path. This is because the Case-II M3D footprint is small enough that global wires are no longer the performance-bottleneck. In this Case-II M3D, memory latency is the performance bottleneck as it contributes to the 46.35% of the 1.1846 ns of Cell Delay on the critical path.

Another interesting aspect in the Case-II designs is that, even at a higher frequency the standard cell area in M3D design is still smaller than the cell area of the 2D design. This is an indication that M3D is able to meet timing much better than 2D. Because the average wire-length per net is 25.6% smaller in this M3D design, the standard cells meet timing easier, and removing the need to up-size the cells significantly. Therefore, the power increase is smaller than the frequency increase leading to an overall PDP benefit of 13.0% in M3D.

Iso-Performance analysis

Table 6.2 compares the Case-II 2D and M3D at iso-frequency of 500 MHz. In this comparison, we see a huge wirelength reduction of 27.8% in M3D. This is because, in Case-II M3D, the bottom-die contains both the shared L3 data-cache and the L1 cache that is part of the RISC-V core. The standard cell placement by the commercial tool in the top-die is

Table 6.1: Max-performance comparison of the 2D and M3D designs of OpenPiton.

	Case-I: Large Memory			Case-II: Small Memory		
	2D	M3D	Δ_{M3D}	2D	M3D	Δ_{M3D}
Full-Chip Stats						
Frequency (MHz)	475	650	36.8	500	675	35.0
Width (mm)	1.97	1.32	-33.0	1.00	0.82	-18.0
Height (mm)	1.97	1.46	-25.9	1.20	0.73	-39.2
Silicon Area (mm^2)	3.880	3.854	-0.7	1.200	1.201	0.04
Cell Area (mm^2)	0.481	0.502	4.2	0.311	0.310	-0.5
Memory Area (mm^2)	2.856	2.856	0.0	0.775	0.775	0.0
Total WL (m)	12.09	10.96	-9.3	7.38	5.34	-27.6
Avg. WL/net (μm)	38.30	34.56	-9.8	35.05	26.06	-25.6
MIV Count	–	252,075	n/a	–	136,295	n/a
Total Power (mW)	292.69	399.27	36.4	146.21	178.04	21.8
PDP (mW*ns)	616.19	614.26	-0.3	292.42	254.34	-13.0
EDP (mW*ns^2)	1297.24	945.02	-27.1	584.84	363.35	-37.9
Critical Path Stats						
Inter-tile path	Yes	Yes	n/a	Yes	No	n/a
Path WL (μm)	2319	1440	-37.9	1824	1587	-13.0
Longest WL (μm)	772.2	385.6	-50.1	658.2	258.9	-60.7
Clock Period (ns)	2.1053	1.5385	-26.9	2.0000	1.6667	-16.7
Launch Latency (ns)	0.4028	0.3291	-18.3	0.3987	0.1501	-62.4
Cell Delay (ns)	0.3173	0.3306	4.2	0.4019	1.1846	195
Wire Delay (ns)	0.3422	0.1318	-61.5	0.2625	0.3611	37.6
Pin Delay (ns)	1.0526	0.7692	-26.9	1.0000	–	n/a
Setup Time (ns)	–	–	n/a	–	0.0040	n/a
Capture Latency (ns)	–	–	n/a	–	0.2557	n/a
Slack (ns)	-0.010	-0.022	120	-0.0631	0.0374	-159

efficiently guided by both L1 and L3-blocks on the bottom-die. By placing logic cells on top of the L1-cache helps reducing the wire-length inside the core. This is the reason we see a much better wirelength savings in the Case-II designs using logic-on-memory M3D. The wirelength reduction leads to the 22.9% switching power savings in the M3D design. With the high switching power reduction, total power is reduced by 13.5% in iso-performance Case-II M3D design compared to its 2D counterpart.

Table 6.2: Iso-performance comparison of the Case-II (small memory architecture) 2D and M3D designs of single-tile OpenPiton.

	2D	M3D	Δ_{M3D}
Frequency (MHz)	500	500	0.0
Silicon Area (mm^2)	1.200	1.201	0.04
Cell Area (mm^2)	0.311	0.301	-3.3
Memory Area (mm^2)	0.775	0.775	0.0
Total WL (m)	7.38	5.33	-27.8
MIV Count	–	141,156	n/a
Total Power (mW)	146.21	126.53	-13.5
WNS (ns)	-0.0631	0.0626	n/a
Total Power distribution by power type			
Internal (mW)	73.12	70.02	-4.2
Switching (mW)	70.39	54.29	-22.9
Leakage (mW)	2.70	2.22	-17.8
Total Power distribution by cell type			
Sequential (mW)	38.01	35.42	-6.8
Combinational (mW)	71.61	55.24	-22.9
Macro (mW)	23.72	23.63	-0.4
Clock (mW)	12.86	12.23	-4.9
Clock Network Stats			
Clock Period (ns)	2.0000	2.0000	0.0
Clock WL (μm)	400,115	379,032	-5.3
Max Latency (ns)	0.4330	0.3059	-29.4
Max Skew (ns)	0.1464	0.1226	-16.3

MIV Count Analysis

As seen in Table 6.1 Table 6.2, the MIV count of the Case-I M3D designs is ∼250,000 and for the smaller Case-II M3D designs, it is ∼141,000. The standard cells are located on the top-die and have easier access to the top metal layers of the bottom-die. The bottom-die consists of memory macros which blocks metal layers 1–4 for internal routing and have a sparse inter-block routing. So, the routing resources on the top-metal layers are under-utilized by bottom-die macros. Commercial tools therefore access the bottom-die metals through MIVs to route the top-die interconnects. This leads to the high MIV counts observed and mitigates the routing congestion of the design.

Timing Path Analysis

To better understand the impact of the logic-on-memory floorplan on the performance benefit of OpenPiton processor-system, we analyze the Case-I 2D critical path in 2D and M3D designs in Figure 6.4(a)–(c), highlighting the path in the respective layouts. By comparing a fixed path in all the designs, we can understand the effects of footprint reduction. A detailed wire delay breakdown of the path in these designs is shown in Figure 6.4(d). Here, the total wire delay is broken into delays of individual wires. Each block in the stacked-column chart represents the delay of a wire between the output-pin of one standard cell to the input-pin of the next. Some of these wires have insignificant delays and cannot be seen in the stacked-column chart.

The 2D critical path here is part of the tile-communication and span a major portion of the width/ height of the floorplan. The tile-communication path is constrained and only half clock-period is available for the delay optimization. Out of the half clock-period the wire-delay makes up 32.5% of total delay in 2D, 18.6% in 475 MHz M3D, and 17.2% in 650 MHz M3D design. The majority of the wire delay (0.2876 ns out of 0.3422 ns) in 2D is caused due to two wires on the path as seen in Figure 6.4(d). These wires are routed over the memory modules in 2D and buffers cannot be placed to break down the long nets leading to large and wide-spread wire delays in 2D. These paths are benefited by two main characteristics of logic-on-memory M3D design: the first is due to the small footprint of M3D design reducing tile dimensions.

The second is an integral part of the logic-on-memory placement. In this placement, as the top-die is free of the huge memory blocks is is easier for placing the buffers to split long wires if necessary. This is the reason logic-on-memory based M3D design shows high performance improvement in multi-core processor systems. Thus, in the iso-performance M3D design at 475 MHz, the same timing path has a smaller wire delay of 0.1959 ns. Comparing the delay breakdown in the timing path in M3D at 475 MHz and 650 MHz, the deviation in wire-delay is also not as wide as in 2D. This is again due to the absence of the

memory-macros that obstruct placement and routing.

This discussion, also explains the presence of the long wire in Case-II 2D design in Table 6.1 even when the footprint is substantially smaller than those of the Case-I 2D and M3D designs (refer to Figure 6.2 comparing the footprints head-to-head). Because, the top and bottom portion of the 2D-die is mainly occupied by macros, the vertical tile-communication paths need to bypass over the memories with a height of over $730\,\mu\mathrm{m}$. This leads to the large maximum wirelength that is similar in Case-I and Case-II 2D designs.

6.4 Conclusion

In this chapter, we benchmarked a RISC-V single-core system with a logic-on-memory M3D-integration scheme. We demonstrated a 37% improvement in maximum performance with M3D due to the critical paths being wire delay dominated. This shows that M3D alongside a good memory macro floorplan is a very promising method for improving performance of common NoC-based processor systems whose critical paths are still dominated by global wires. Using a smaller memory size for the tiles, we observe a 13.5% power savings, demonstrating the usability of logic-on-memory designs also for low-power designs. The cost-impact of having huge MIV counts in the design is not considered here. A high MIV count can increase the cost of M3D ICs until the technology matures. Thus, an analysis of the max-performance as a function of MIV count is left for future work.

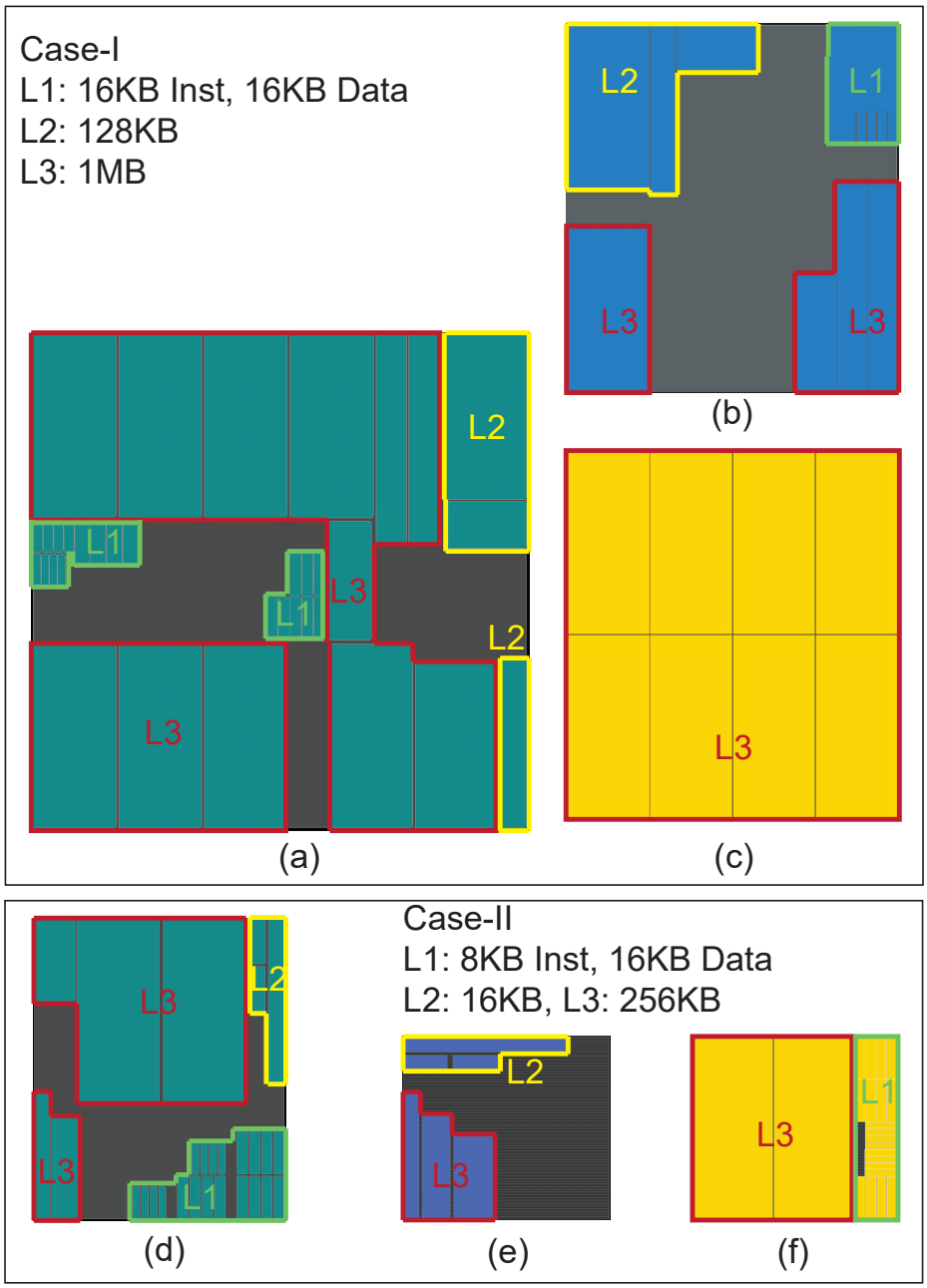


Figure 6.2: Physical layout of the memory modules. Case-I designs: (a) 2D, (b) M3D top-die, (c) M3D bottom-die; Case-II designs: (d) 2D, (e) M3D top-die, (f) M3D bottom-die

.

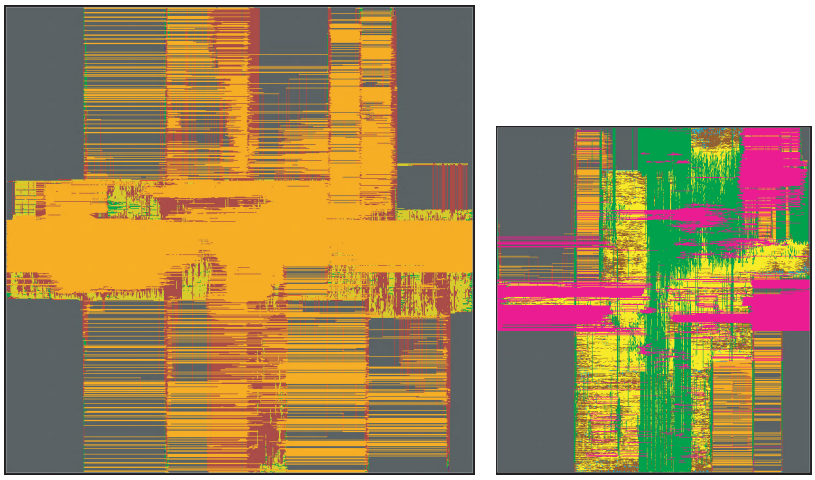


Figure 6.3: GDS layouts of single-tile OpenPiton Case-I (= large) memory architecture. (a) 2D, (b) M3D.

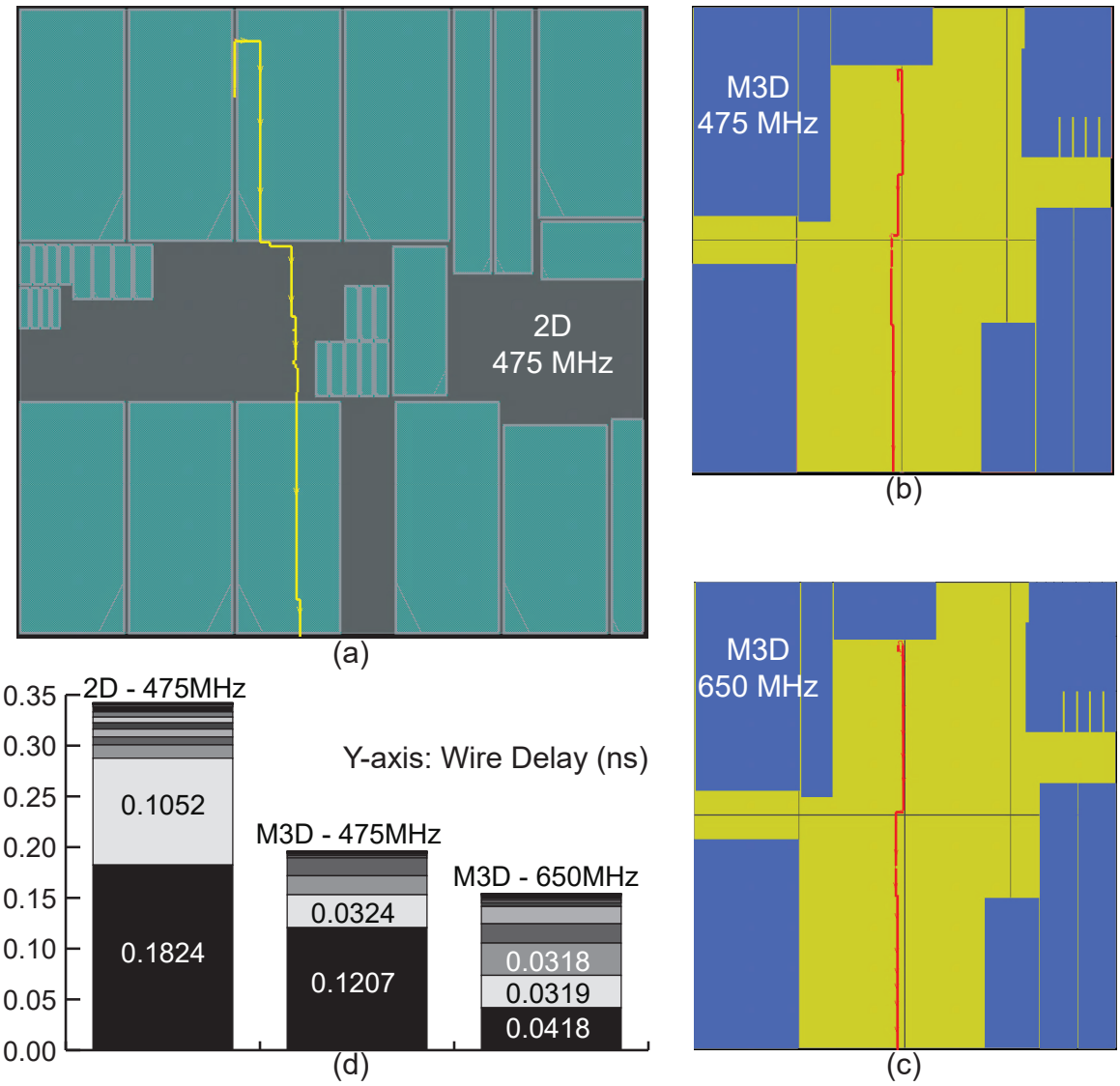


Figure 6.4: Timing critical path of Case-I 2D memory architecture design in: (a) 2D at 475 MHz, (b) M3D at 475 MHz (iso-performance), (c) M3D at 650 MHz (max-performance). (d) Detailed delay breakdown of the path in the designs.

CHAPTER 7

HETEROGENEOUS MONOLITHIC 3D ICS: EDA SOLUTIONS, AND POWER, PERFORMANCE, COST TRADEOFFS

The current state-of-the-art research in 3D IC designs improve the Power-Performance-Area (PPA) in homogeneous M3D ICs by exploiting improvements in 3D wirelength, placement, routing, or a specific 3D stacking. But, heterogeneous integration (where each tier of an M3D IC utilizes a different technology node) is a unique advantage of M3D fabrication that is not well studied.

Only the most recent pseudo-3D flow Pin-3D [13], support heterogeneous 3D IC optimization at the dense gate-level partitioning. Heterogeneity is a major focus in 2.5D ICs and is also recently fabricated with microbump bonded 3D ICs [37]. These implementations use a coarse level heterogeneity at chip-let level in 2.5D ICs, and at the block level in bonded 3D ICs. M3D ICs, with their dense pitch, can support technology heterogeneity at a finer level of partitioning. So, a single netlist can be divided into two tiers at a gate-level and manufactured with different technologies.

In this chapter, we use an enhanced Pin-3D flow to study the behavior of heterogeneous 3D, and PPAC (PPA, Cost) impact using 4 different RTLs: AES (cell-dominant design), Netcard (large design, slightly wire dominant), LDPC encoder and decoder circuit (extremely wire dominant), CPU design from a commercially available core (large, general purpose design with memory blocks). Each of these four netlists is designed in 5 technology and design configurations as shown in Figure 7.1. Overall, we see that heterogeneous design achieves better Power-Delay Product (Energy Efficiency) and PPC (Performance/(Power*Cost)).

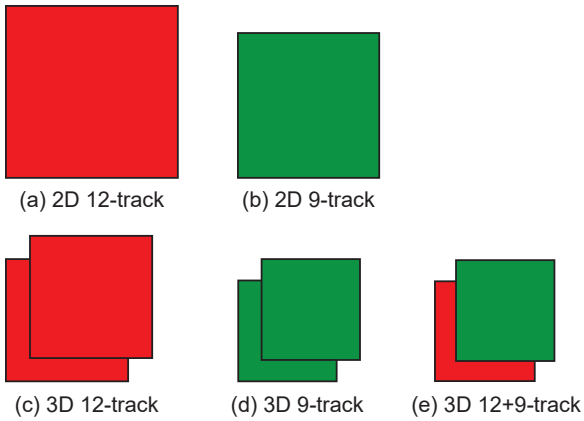


Figure 7.1: 5 different configurations (to scale, assuming equal number of cells) of 2D and 3D using 9-track and 12-track cells studied in this work. We use commercial 28nm libraries.

7.1 Technology Setup

Heterogeneous technology can be used to target different aspects of the PPAC metrics. For example, the authors in [37] use a low-power 22 nm and a high-performance 10 nm technologies to design different blocks of an SoC with dissimilar PPA requirements such as the compute core and periphery modules. In our work, we cost reduction along with power reduction at a high operational frequency subsection 7.3.1 defines our interpretation of a 'high' frequency). By studying the PPC trends of previous technology nodes, we pick a mixture of libraries from the set of foundry provided 28 nm libraries that meet our PPAC requirements.

7.1.1 Cost Trends

The equation $\textit{Cost per Element} = \textit{Wafer Cost per } \text{mm}^2 \times \textit{Area of Element}$ is a useful to understand the cost trends with scaling. In general, the increase in wafer cost at each generation is accompanied by a significant increase in transistor density, resulting in cost per element reduction for advanced technologies [38]. In such cases, it is more expensive to manufacture a chip using an older technology node. Wafer cost has been growing at an increasingly higher pace only in the recent technology nodes, and cost per chip has

Table 7.1: Cost Model Parameters and Assumptions [40]

Baseline wafer cost (FEOL+8 metals)	C'
Wafer FEOL cost	$0.3 \times C'$
Wafer BEOL cost (up-to 6 metals)	$0.66 \times C'$
3D integration cost (α)	$0.05 \times C'$
Wafer Diameter	300 mm
Defect Density (D_w)	0.2 mm^2
Wafer yield (κ)	0.95
3D Yield Degradation (β)	0.95
2D Wafer Cost (C_{2D})	$0.96 \times C'$
3D Wafer Cost (C_{3D})	$1.97 \times C'$

only increased slightly at the 5 nm node [39]. Technology scaling also improves power and performance, and advanced technology nodes always achieve better PPC. So, such technology heterogeneity is not the best candidate to achieve PPC improvement without specially designed libraries [37].

Apart from technology node scaling, using cells with a fewer tracks (shorter cell height), the cell area decreases without incurring additional wafer costs as the design rules and the mask layers complexity remain the same [41]. The smaller cells with fewer tracks have better power, but worse timing and larger cells with more tracks have worse power but better timing. So, we cannot optimize all the PPAC metrics at once using just a single cell track type. In our work, we use a 12-track (highest available) and a 9-track (smallest available) libraries from the commercial foundry 28 nm node in heterogeneous 3D ICs to extract benefits from the both track types simultaneously without incurring significant drawback. The two track variants have remarkably similar BEOL (Back-End-Of-Line), and so the routing layers in the three 3D cases are similar to each other. To analyze the die cost and PPC metrics, we use a cost model previously used for M3D ICs in [40] assuming it will hold in our 28 nm heterogeneous case. Table 7.1 shows the list of parameters used. The BEOL cost is split between the metal layers based on the routing pitch.

Table 7.2 shows the expected full-chip behavior of the technology and design configu-

Table 7.2: "Qualitative" comparisons of expected PPAC behavior of the 5 technology and design configurations at their expected maximum frequencies. 1 means the worst, and 5 the best

	9 Track (slow & small)		12 Track (fast & large)		9+12 Tracks (combined)	
	2D	3D	2D	3D	2D	3D
Frequency	1	2	4	5	-	3
Power	4	5	1	2	-	3
Power/Freq	3	4	1	2	-	5
Footprint	4	5	1	2	-	3
Si Area	5	5	1	1	-	3
Die Cost	5	4	2	1	-	3

rations. In the following subsection, we discuss the various quirks of using such heterogeneous cells in a commercial 2D tool.

7.1.2 Quirks of Heterogeneity

To achieve a heterogeneous design with significant power and area benefits without incurring a marked performance loss, we have to choose a suitable process and voltage corner for the different libraries. Here, we choose a slower process (higher threshold voltage) and a lower voltage (0.81 V) for the 9-track libraries, and a faster process (lower threshold voltage) along with higher voltage (0.90 V) for the 12-track libraries to separate the libraries further in terms of their achievable PPA. When the two libraries have similar characteristics, heterogeneous 3D cannot be much different or better than a homogeneous 3D. The voltage difference (0.09 V) is considerably smaller than the PMOS threshold voltage $V_{thp} \sim 0.3\,\mathrm{V}$ of the cells in the 0.90 V domain. This ensures that the pull-up network turns-off without the need for voltage shifters that impede dense connectivity benefits of M3D ICs.

An FO-4 inverter with the two extreme heterogeneous configurations is shown in Figure 7.2. The cells connected to other tier are referred to as 'boundary cells'. In the '**heterogeneity at driver output**' case, the heterogeneity affects the load pin capacitances and the output slew of driver. The liberty (LIB) model files capture the pin capacitance vari-

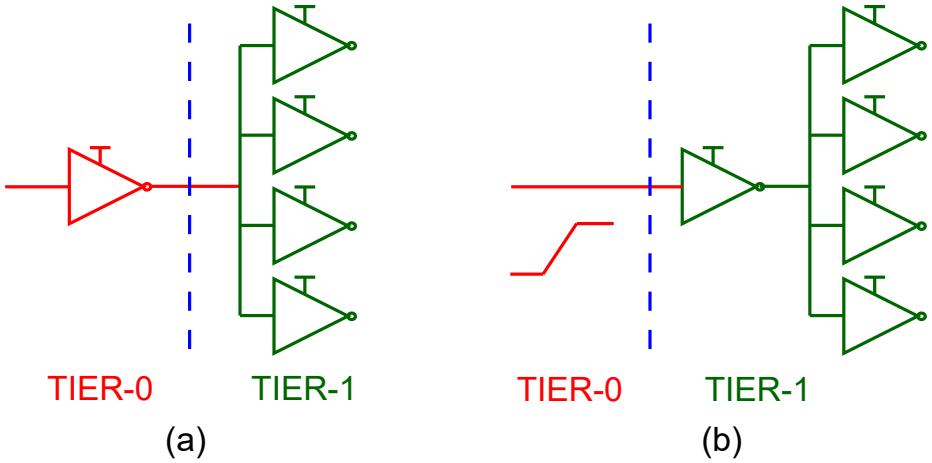


Figure 7.2: The two types of boundary conditions due to heterogeneity in a FO-4 inverter. (a) Heterogeneity at driver output, (b) Heterogeneity at driver input.

Table 7.3: Impact of heterogeneous technology when input to driver of an FO4 is from different tier (see Figure 7.2(b)). Time is in ns, Power is in µW, Voltage is in V

	Case-I	Case-II	$\Delta\%$	Case-III	Case-IV	$\Delta\%$
Tier-0	fast	slow	–	slow	fast	–
Tier-1	fast	fast	–	slow	slow	–
Driver V_G	0.9	0.81	10.0	0.81	0.9	11.1
Rise Slew	15.6	16.9	8.1	14.6	13.1	-9.9
Fall Slew	18.2	19.4	6.6	19.1	17.6	-8.1
Rise Del.	12.5	13.0	3.4	23.6	22.4	-5.3
Fall Del.	16.4	17.1	4.1	26.2	24.8	-5.1
Lkg. Pow.	0.093	0.330	250	0.003	0.002	-44.9
Total Pow.	3.86	4.21	9.2	2.00	1.99	-0.6
% Input Pins	31	15	–	33	21	–

ation due to the heterogeneous cells, and the slews are accurately calculated. Spice analysis also shows that when the driver is in a separate tier from the load, the slew changes by atmost $\pm 15\%$. So, as long as the libraries have a significant overlap in characterized slew ranges these variations lie within the characterized range that usually span 2-3 orders of magnitude.

The next type of variation at boundary cells occurs when the entire FO-4 setup is on a single tier, but the driver input comes from the other. This case is referred to as '**heterogeneity at input**' and the LIB files used here do not model differences in input logic-high level and the power-domain level. The impact of this type of boundary hetero-

Table 7.4: Improvements obtained with our heterogeneous version of Pin-3D flow [13] for the commercial CPU design

		Pin-3D [13]	Hetero-Pin-3D
Frequency	GHz	1.200	1.200
WL	m	3.22	3.07
WNS	ns	-0.489	-0.055
Total Power	mW	224.1	188.0

geneity is shown in Table 7.3. With only $\sim \pm 5\%$ difference in cell delays, the impact on total path timing is negligible. In a heterogeneous 3D design, the number of boundary input pins of the two heterogeneous cases (Cases II and IV from Table 7.3) differ at most by 1. So, even with multiple MIVs (Monolithic Inter-tier Vias) on a path that goes back and forth between the two tiers, the estimated timing using the LIB files only differ by $\sim 5\%$ of cell delay.

The main discrepancy for 'input heterogeneity' is in the leakage power, which increases $2.5\times$ when logic-high is at 0.81 V, and pull-up network connected to 0.90 V (Case-II in Table 7.3). This increase is caused by the exponential nature of I_{DS} vs. V_{GS} in the cut-off region. At full-chip level, the leakage power is just $\sim$1% of the total power estimate in our heterogeneous designs. On average only 15% of the input pins over all the four heterogeneous implementations fall in Case-II. Assuming each input pin has an equal contribution to leakage and considering a $2.5\times$ increase in individual leakage value, the leakage power increases by 37.5 and total power by just 0.375%. Even after assuming the worst-case leakage increase of 37.5%, heterogeneous designs are still better than using only fast cells in a 2D or 3D configuration as the slow cells have a $\simeq 10 - 20\times$ smaller leakage than fast cells.

7.2 Heterogeneous 3D IC Design Flow

7.2.1 Enhancing the Pin-3D Flow

To design a timing optimized heterogeneous 3D IC in [13], the authors make several assumptions regarding technology, clock tree, voltage levels, etc. Using Pin-3D in its proposed form for heterogeneous 3D, the timing and total power become significantly worse due to the unconventional clock-tree and critical path behavior of heterogeneous 3D as seen in Table 7.4. These peculiarities are revisited using full-chip analysis in subsection 7.3.3. As the post-CTS optimized pseudo-3D input to Pin-3D is done in a 2D fashion, it is based on homogeneous technology, and differs significantly from a clock tree that is optimized for a heterogeneous 3D. Moreover, the inability of the area-balanced min-cut partitioning to account for timing difference between the dies in heterogeneous circuits exacerbates the timing issues. Several enhancements are proposed to the Pin-3D flow to alleviate aforementioned issues and achieve the results in Table 7.4.

Timing-based Partitioning

When the input pseudo-3D stage for the heterogeneous 3D is designed with the faster library, a placement-driven and cut-size driven partitioning can assign cells on critical paths to the slower die to improve the cut-size. In such cases, the critical paths show an increase in worst slack, and optimization with this partitioning solution may not result in timing closure. In our experiments, we see that a path-based partitioning solution [42] cannot be applicable here. In path-based timing queries, the paths are differentiated by the begin-end pair of registers, and there can be several different paths between a same pair. The different paths between a specific register pair naturally have a large intersection in terms of the cells on the path, and the coverage increases at a very slow rate as we increase the number of paths per pair. In our experiments, even after considering up-to 1024 paths per register pair, we were only able to cover 60% of the total constrained cells in the design and eventually,

the run-time to just calculate cell coverage increased to up-to an hour with only 60-65% unique cells traversed.

Therefore, we use a cell-based partitioning where each cell is visited exactly once and the slack value of the worst timing path through the cell is assigned as 'cell-slack'. If the cell is not constrained, slack is assumed to be a large positive value. When partitioning is based only on timing, 3D placement becomes highly clustered, and the legalization significantly alters the input placement. So, only a 20-30% of the total cell area is assigned to the faster die. Bin-based FM min-cut completes the partitioning to obtain a tradeoff between the timing and placement based partitioning solutions. The area threshold depends on the design and their architecture, based on how many critical paths need to be fixed to fast die.

Supporting Heterogeneous Clock Tree

The clock tree engine within the commercial EDA tool, Innovus, doesn't treat the zero-sized overlaps as zero-area cells in Pin-3D, and causes a density overflow during the clock design. To remedy this, we use a cleaner and equally efficient approach of transparent cells using the COVER cell construct provided in LEF. This class of cells is considered by the tool to have no active area and does not break the clock engine. As the tool still uses the LIB files for accurate PPA estimations, COVER cells are a useful construct for the zero-sized cells. Based on the placement and timing optimization in Pin-3D, the clock tree optimization is done in two stages for the top and bottom tiers.

7.2.2 Re-partitioning Using ECO

Area-balancing between the tiers in a 3D design is crucial for efficient usage of silicon area in 3D. A heavily skewed utilization leads to placement and routing congestion within the high utilization die and worsens full-chip timing and power. In heterogeneous designs, the timing-based partitioning is not always sufficient to ensure area balance as not all the critical cells in 3D are identified after pseudo-3D. Additionally, a sub-par area threshold

Algorithm 3: Re-partitioning algorithm

$unbalance \leftarrow (area_{slow} - area_{fast})/area_{total}$
$d_k \leftarrow d_0; n_p \leftarrow n_0$
▷ Initialize delay threshold factor, number of critical paths considered per loop
while $unbalance > unbalance_{th}$ **do**
  $d_{th} \leftarrow d_k \times$ (avg. cell delay of n_p critical paths)
  foreach $cell \in$ *the* n_p *critical paths* **do**
    $d_{cell} \leftarrow$ delay of $cell$ on its critical path
    if $d_{cell} > d_{th}$ **then**
      $all\_crit+ = 1$
      if $cell \in$ *slow die* **then**
        $slow\_crit+ = 1$
        append $cell$ to $move\_list$
      end
    end
  end
  if $slow\_crit/all\_crit < crit_{th}$ **then**
    break; ▷ slow and fast tier cells have similar delay
  end
  Move all cells in $move\_list$ to the fast die
  update timing
  if $Delta_{WNS} < W_{th}$ || $Delta_{TNS} < T_{th}$ **then**
    Undo the cell moves done
    $d_k* = \alpha$ ▷ $\alpha < 1$, update delay threshold
  end
  update $unbalance$
end

value can end up with some critical cells on the slower die. So, we introduce incremental algorithm 3 based on 3D timing, unbalance to alleviate area unbalance after any 3D optimization stage.

7.3 Experimental Results

As discussed in subsection 7.1.1, we use a 9-track library at 0.81 V and a 12-track library at 0.90 V. In heterogeneous 3D designs, the bottom tier has the faster 12-track tech, and the top tier has the slower 9-track tech. Six metal layers are used per tier in 3D, and six metal layers are used in total in 2D to complete signal routing. As the foundry provided BEOL file is virtually identical for both the track variations, the MIV layer is the same in all the

three M3D configurations. The MIV is similar to a routing via at 140 nm tall, 70 nm wide square face, and resistance and capacitance of $\sim 2\,\Omega$ and $\sim 0.015\,\mathrm{fF}$. The BEOL parasitics, including the MIV layer parasitics, are generated using Quantus from Cadence®using the material properties and physical dimensions of the layers.

7.3.1 Methodology

The first step in RTL-to-GDS conversion is netlist synthesis. The footprint is fixed to be a square and the area is determined using a provided target utilization value and the synthesized netlist. To allow for any noise in the PNR optimization, a worse negative slack of up-to $\sim 5-7\%$ of clock period is considered as timing-met. AES can achieve a rather small clock periods, and the slack threshold of 10% the clock period is used for determining the timing-met condition. For the iso-performance comparison among various technology configurations, a 'high' frequency is used. This is defined as the value where slow 2D does not meet timing, and fast 2D shows negative timing slacks at the initial target utilization.

Homogeneous 3D ICs follow the same design flow as in [13] using the 'high' frequency from 2D implementations, and the are from its corresponding (12-track or 9-track) 2D design. Heterogeneous 3D IC design starts with pseudo-3D stage at fast-node using same footprint area as its fast 2D analogue, and the timing-based partitioning enhancement gives the partitioning solution. As the 9-track cells are 25% smaller, more than 50% ($\sim 60\%$) of the 12-track cell area should be converted to 9-track tier, to achieve area-balanced partitioning. So, the foot-print is further shrunk by $\sim 13.5\%$ to maintain chip utilization. Finally, the enhanced Pin-3D flow is used to create the heterogeneous 3D GDS.

7.3.2 Full-Chip PPAC

Heterogeneous 3D IC Results

Table 7.5 shows the raw values of the 4 RTL implementations with heterogeneous 3D. The density value reported for 3D designs is the average of the two tiers. Being heavily

Table 7.5: PPAC results of our 3D Heterogeneous Designs (raw data based on a commercial foundry 28 nm technology)

	Units	netcard	aes	ldpc	cpu
Frequency	GHz	1.750	3.000	1.125	1.200
Area	mm^2	0.384	0.126	0.216	0.390
Chip Width	µm	438	251	329	442
Density	%	82	86	64	88
WL	m	6.560	1.022	5.500	3.073
# MIVs	k	153	62	83	98
Total Power	mW	550	138	339	188
WNS	ns	-0.037	-0.028	-0.026	-0.055
TNS	ns	-0.41	-4.827	-0.902	-15.54
Effective Delay	ns	0.608	0.361	0.597	0.888
PDP	pJ	334.5	49.8	310.1	167
Die Cost	$10^{-6}C'$	6.16	1.97	3.41	6.26
PPC	$\frac{\text{GHz}}{\text{mW}\times 10^{-6}C'}$	0.517	11.06	0.946	1.02

wire dominant, LDPC cannot achieve high placement densities without causing routing congestion. 3D routing in LDPC significantly changes its critical paths, and up-to 20% area-unbalance occurs after the first optimization stage. Re-partitioning fixes this issue and brings down the final unbalance to within 5%. The other three designs are not as wire dominant, and timing-based partitioning alone limits the area unbalance to ¡5%. The Worst Negative Slack (WNS) is within $\sim 7\%$ of the clock period as a result of the heterogeneous enhancements for all the four different RTLs. Effective delay (=clock period - worst slack) value is used for PDP calculation to factor in the small WNS imperfections. Finally, PPC is calculated based on achieved frequency, power consumption, and die cost.

Heterogeneous 3D vs. 9-track 2D

In Table 7.6, the change of each metric in heterogeneous 3D (referred to as HT-3D) w.r.t. the four other design configurations is shown. In the first set of four columns, we see that when compared to 2D 9-track (9T-2D), HT-3D shows a significant total area benefit as 9T-2D designs require a strict timing target and a larger area to meet the iso-performance targets. The large CPU, netcard design still cannot meet timing targets even with higher

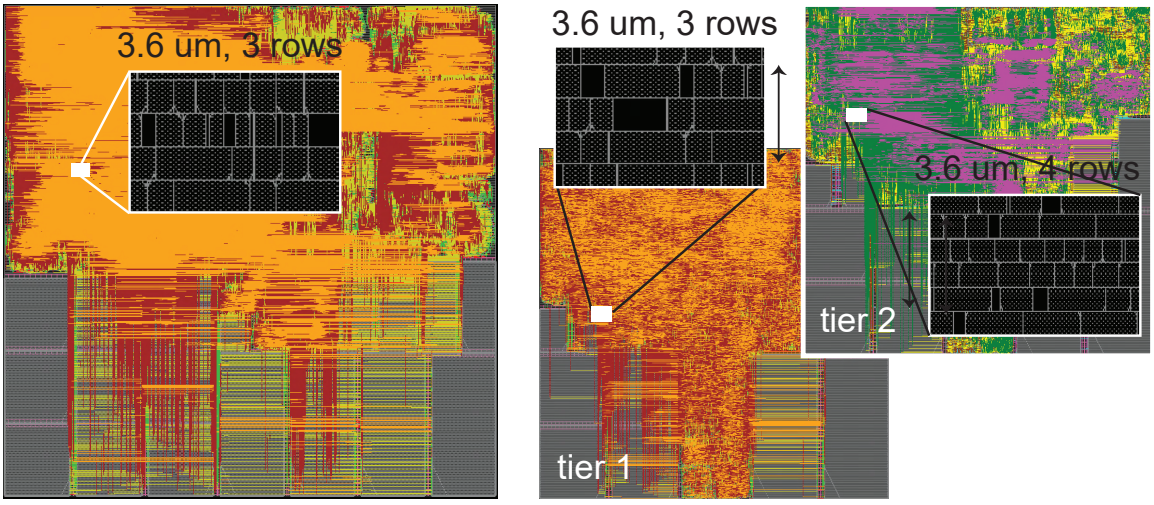


Figure 7.3: Routing and zoomed placement GDS layouts of our commercial CPU. (a) 2D 12-track, (b) 3D heterogeneous, where tier 1 is using 12-track cells and tier 2 9 track cells.

area usage. While HT-3D contains some large 12-track cells on the faster tier, it utilizes them efficiently to meet timing targets without the need for a larger area. We see that this area increase in 9T-2D is not unwarranted as the densities at this point are only 3-4% smaller than the $\sim 85\%$ density of HT-3D. Overall we see that the 'high' frequency is either unattainable or the 9T-2D fail to show any of their expected benefits from Table 7.2 at the frequency target.

Heterogeneous 3D vs. 12-track 2D

12-track technology is the faster technology used, and from Table 7.6 we see that the 12T-2D are near their max-frequency limit as the worst slack is negative, showing the limits of timing optimization. They also have a high utilization at around 75% for AES, and $>$80% for the others. The effective delay of HT-3D is very close to the 12T-2D designs showing that HT-3D can truly achieve the 'high' frequency of fast 2D. By virtue of using 9-track cells, HT-3D is consistently the best in terms of PDP and PPC, with $18.6 - 57.2\%$ improvement in the PPC. In the CPU design, the memory cell technology is left unchanged within the two dies. So, we only see a 7.8% area reduction rather than the $12 - 13\%$

reduction seen in other netlists. Even without this additional cost and power benefits due to memory macros, the CPU design still shows a 23% PPC improvement.

Heterogeneous 3D vs. 9-track 3D

Similar to 9T-2D, 9T-M3D designs have a worse area, timing, power, and cost in every case except the LDPC design. The 9T-3D of the wire-dominant LDPC has a smaller wirelength with the help of the 3D placement and routing space, and greatly benefits the power and even delay to a lesser extent. But the required area increase to meet the timing is significant drawback and PPC is improved by 10% in HT-3D. Like its 2D implementation, the CPU design still cannot meet the timing, but power and PDP in 9T-3D are slightly better than the 9T-2D case.

Heterogeneous 3D vs. 12-track 3D

12-track 3D designs are expected to perform better than their 2D equivalent in terms of power, performance, and this is what we see in most of the cases as the power, timing benefit in HT-3D is smaller w.r.t 12T-3D than 12T-2D. But the 12T-3D has a worse die cost due to 3D cost overhead, and the overall PPC benefit is still $\sim 20\%$ for HT-3D compared to 12T-3D.

7.3.3 Analysis of clock, critical path, and memory connections

Using the commercial CPU core, we analyze the clock network, critical paths, and the connections to and from memories in 12-track 2D, 12-track 3D, and heterogeneous 3D to understand the various physical design aspects of heterogeneous M3D. These results are tabulated in Table 7.7.

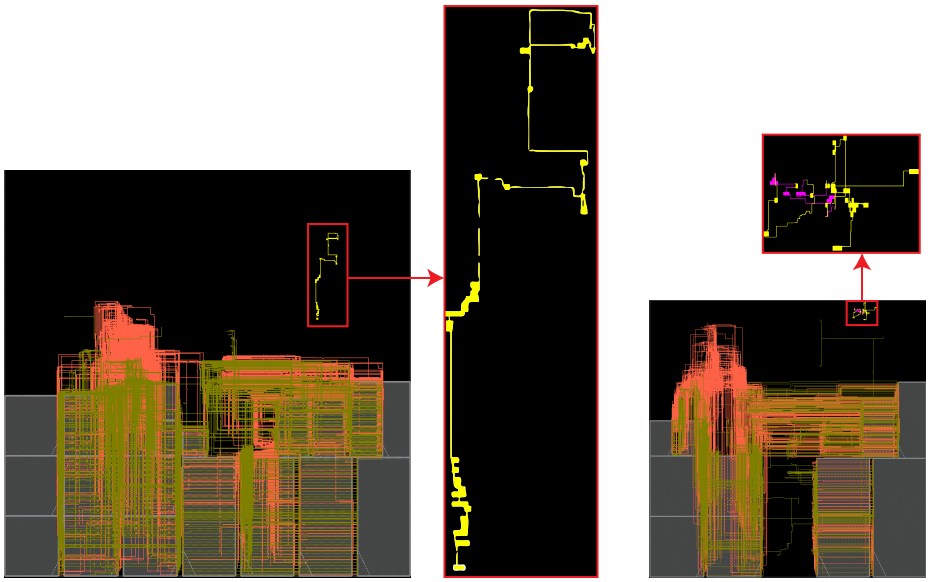


Figure 7.4: Timing critical paths and memory nets of (a) 12-track 2D, (b) Heterogeneous 3D implementations of the CPU design. Yellow: 12-Track tier wires and cells, Magenta: 9-Track tier wires and cells, Dark Red: memory output nets, and Dark Green: memory input nets.

Memory Interconnects

Root Mean Square average of the memory net latencies is used in Table 7.7 as it is more affected by the larger delay values. Memory latency is better in 3D in general due to its smaller footprint and better macro placement that achieves dense wiring as shown in Figure 7.4. In heterogeneous 3D, the smaller floorplan and pin-capacitance reduction from 9-track cells contributes to the additional latency reduction. Some of the nets connected to memory blocks in HT-3D are driven by lower V_{DD} and this contributes to additional switching power reduction.

Clock Network

The clock latency varies widely in HT-3D, resulting in large max latency as well as clock skew in heterogeneous 3D design due to the presence of clock network on both slow and fast dies as seen in Figure 7.5(b). Even though the maximum skew and latency are worse,

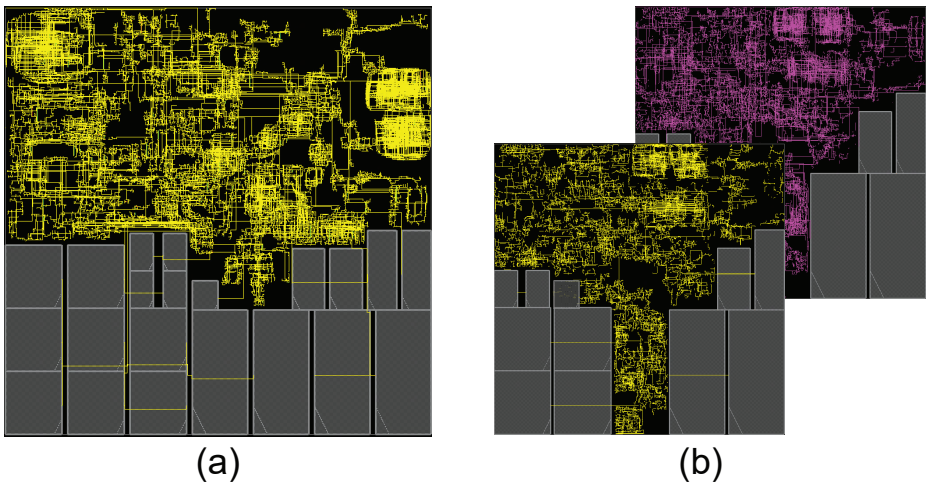


Figure 7.5: Clock tree layouts of (a) 12-track 2D, (b) Heterogeneous 3D implementations of the CPU design. Yellow: 12-Track tier clock wires, Magenta: 9-Track tier clock wires

we see the benefit of our clock tree design in the average clock skew on the first 100 critical paths. Critical path clock skew is very important as they have a significant impact on critical path slack (Hetero-3D has same critical path delay as 12T-3D but a worse slack due to the clock skew).

Critical Path

The breakdown of the critical path is the most revealing aspect of the heterogeneity. The clock period is the same ($= 0.833\,\mathrm{ns}$) for the three configurations, and path delay is mostly ($\sim 97\%$) made up of cell delay. We see that the critical path in HT-3D is shorter in length with only a few cells on the slow die, unlike the 12T-3D path whose cells are nearly evenly split between the dies. We can see from the average cell delays that without such skewed critical path partitioning, the timing would worsen in HT-3D. This skewing is achieved with timing-based partitioning. Critical path layout in Figure 7.4 also show the benefit of 3D placement as its critical path has a significantly smaller bounding box than 2D.

7.4 Conclusion

In summary, we have presented a novel arrangement for gate-level monolithic 3D ICs using heterogeneous technology integration, along with various enhancements in partitioning, clock tree, and a new re-partitioning stage to support such 3D integration. We saw that using different cell tracks on the two tiers work the best for heterogeneous 3D ICs in terms of voltage and BEOL compatibility required for densely connected M3D ICs. Overall, the 3D hetero designs achieve high frequencies very close to fast-2D, and provide a PPC improvement of $18.6 - 57.2\%$ compared to timing-met 2D designs, and $10.2 - 51.6\%$ compared to the timing-met 3D.

Table 7.6: PPAC percentage delta (= (3D hetero - config) / config × 100) of 3D heterogeneous design w.r.t. different homogeneous configurations. A -ve (+ve for PPC) value implies that heterogeneous implementation outperforms the particular configuration.

CONFIG →	2D 9-Track				2D 12-Track				M3D 9-Track				M3D 12-Track			
	netcard	aes	ldpc	cpu	netcard	aes	ldpc	cpu	netcard	aes	ldpc	cpu	netcard	aes	ldpc	cpu
Si Area	-28.6	-27.6	-13.9	-23.1	-12.3	-13.7	-13.9	-7.8	-28.6	-27.6	-13.6	-23.2	-12.3	-14.9	-13.6	-8.0
Density	3.2	2.9	-12.4	4.1	-1.8	14.5	2.2	5.4	8.7	6.9	-10.0	4.1	4.2	14.8	27.9	5.4
WL	-33.0	-30.9	-14.4	-34.8	-25.3	-17.0	-31.1	-33.5	7.8	-10.4	18.6	-8.9	-2.2	0.0	-8.7	-2.1
Total Power	-12.7	-15.0	-8.1	-21.1	-10.5	-8.0	-29.7	-14.6	-4.2	-8.6	5.4	-15.9	-6.0	-3.4	-5.6	-10.1
Eff. Delay	-14.7	-1.6	-0.8	-12.9	1.0	4.0	-2.6	-6.2	-19.3	0.0	1.2	-12.9	6.1	6.2	2.2	4.2
PDP	-25.6	-16.3	-8.8	-31.2	-9.6	-4.2	-31.5	-9.3	-22.6	-8.5	6.6	-26.7	-0.2	2.7	-3.5	-6.3
Die Cost	-27.9	-23.3	-9.5	-21.8	-9.7	-8.4	-9.5	-4.7	-29.7	-27.8	-13.9	-24.1	-12.7	-15.1	-13.9	-8.3
PPC	59.1	53.5	20.3	61.9	23.7	18.6	57.2	23.0	48.6	51.6	10.2	56.7	21.9	21.9	23.0	21.4
Width (µm)	733	417	501	712	662	382	501	650	518	295	353	504	468	272	353	460
WNS (ns)	-0.14	-0.03	-0.03	-0.19	-0.03	-0.01	-0.05	-0.01	-0.18	-0.03	-0.02	-0.19	-0.00	-0.01	-0.01	-0.02
TNS (ns)	-450	-3.61	-5.13	-357	-2.14	-0.44	-7.60	-0.03	-832	-1.95	-0.04	-316	-0.00	-0.03	-0.01	-0.59

Table 7.7: Clock Network, Critical Path, Memory Interconnect analyses of the commercial CPU design

		12T 2D	12T 3D	Hetero 3D
Memory Interconnects				
RMS Input Net Latency	ps	25.0	16.1	15.5
RMS Output Net Latency	ps	37.5	33.2	28.7
Net Switching Power	mW	5.47	4.02	3.41
Clock Network				
Buffer Count		1593	1502	1330
Buffer Area	μm^2	1277	1175	982
Wirelength	m	0.114	0.108	0.107
Max Latency	ns	0.234	0.292	0.713
Max Skew	ns	0.058	0.142	0.344
100 Path Avg. Skew	ns	-0.008	0.000	-0.011
Critical Path				
Slack	ns	-0.003	-0.012	-0.055
Clock Skew	ns	-0.014	0.013	0.045
Setup Time	ns	0.004	0.008	0.001
Path Delay	ns	0.845	0.831	0.845
# MIVs		–	9	6
Bottom Cells		45	20	25
Bottom Cell Delay	ns	0.830	0.375	0.482
Avg. Bottom Delay	ns	0.019	0.019	0.019
Top Cells		–	23	8
Top Cell Delay	ns	–	0.447	0.343
Avg. Top Delay	ns	–	0.019	0.043

CHAPTER 8

CONCLUSIONS

8.1 Machine Learning Integrated Pseudo-3D Flow for Monolithic 3D ICs

In summary, we have presented a machine learning model that can predict final net parasitics at an early stage of the design. We have analysed several net features and how they impact the parasitic evolution in a pseudo-3D flow. We formulate new metrics and use them to achieve better circuit agnostic learning models. Using these models, we were able to achieve higher R2 score, lower MSE, better timing. We discussed the issue of critical path estimation in 3D design and showed that our general model is better than a circuit specific model. With $3 \times -16\times$ TNS reduction on test circuits, integrating these models in the pseudo-3D flows help us to minimize number of timing violations and the severity of the violations after routing.

8.2 Pin-3D: An Effective Multi-Die Co-Optimization Methodology for 3D IC Design

In this chapter we proposed our Pin-3D methodology for incremental placement optimization, clock tree optimization, routing, timing optimization, and ECO for 3D ICs using the commercially available PnR tools. Compared to the current state-of-the-art 3D flows, we showed that adding our Pin-3D optimization improves every aspect of the design from placement, routing, and PPA. Especially, we see more than a $10\times$ smaller total negative slack for the Cortex-A7 design and similarly high reductions in other benchmarks as well. Compared to 2D designs, we were able to see $\sim$20% reduction in EDP for the Cortex-A series benchmarks, and $\sim$30% EDP improvement for the LDPC benchmark. 3D routing with Pin-3D also allowed for savings in the BEOL cost without any meaningful effect to the important PPA metrics. We also saw how the buffering insertion with Pin-3D method-

ology is superior in terms of critical path timing compared to a fairly limited die-by-die optimization. And finally, a proof-of-concept design of a heterogeneous 3D IC shows the versatility of Pin-3D as well as exploring more complex structures that are possible with 3D IC design.

8.3 Metal Layer Sharing: A Routing Optimization Technique for Monolithic 3D ICs

In this chapter, we have first analyzed the routing in various 3D IC types and found that metal layer sharing in 3D ICs is a novel phenomenon which can be controlled or enhanced in the designs based on user input. This layer sharing is only meaningful for the Monolithic 3D ICs due to the fine pitch and Face-To-Back nature of sequential fabrication. While metal sharing uses a large amount of MIVs, we see that they do not cause any routing issues in the design. Rather, they are helpful in effectively using the metal layers with a high quality routing using one fewer BEOL layer resulting in cost savings for 3D ICs. Even with the dropped metal layer, the timing and/or power consumption of the design improved with a Power Delay Product improvement of 5-7% across the three commercial processor designs.

8.4 On Legalization of Die Bonding Bumps and Pads for 3D ICs

In summary, we have shown that when the 3D via pitch becomes comparable or larger than the global cell grid, commercial detail router fails to create a good 3D via assignment with cut short and spacing violations. Fixing these violations early with our proposed legalizer techniques can create a better routing quality with fewer DRVs, better total slack, with only a negligible run-time impact.

8.5 A Logic-on-Memory Processor-System Design with Monolithic 3D Technology

In this chapter, we benchmarked a RISC-V single-core system with a logic-on-memory M3D-integration scheme. We demonstrated a 37% improvement in maximum perfor-

mance with M3D due to the critical paths being wire delay dominated. This shows that M3D alongside a good memory macro floorplan is a very promising method for improving performance of common NoC-based processor systems whose critical paths are still dominated by global wires. Using a smaller memory size for the tiles, we observe a 13.5% power savings, demonstrating the usability of logic-on-memory designs also for low-power designs. The cost-impact of having huge MIV counts in the design is not considered here. A high MIV count can increase the cost of M3D ICs until the technology matures. Thus, an analysis of the max-performance as a function of MIV count is left for future work.

8.6 Heterogeneous Monolithic 3D ICs: EDA Solutions, and Power, Performance, Cost Tradeoffs

In summary, we have presented a novel arrangement for gate-level monolithic 3D ICs using heterogeneous technology integration, along with various enhancements in partitioning, clock tree, and a new re-partitioning stage to support such 3D integration. We saw that using different cell tracks on the two tiers work the best for heterogeneous 3D ICs in terms of voltage and BEOL compatibility required for densely connected M3D ICs. Overall, the 3D hetero designs achieve high frequencies very close to fast-2D, and provide a PPC improvement of $18.6 - 57.2\%$ compared to timing-met 2D designs, and $10.2 - 51.6\%$ compared to the timing-met 3D.

REFERENCES

[1] G. Moore, "Cramming More Components Onto Integrated Circuits," *Proceedings of the IEEE*, 1998.

[2] M. M. Shulaker, T. F. Wu, *et al.*, "Monolithic 3D integration of logic and memory: Carbon nanotube FETs, resistive RAM, and silicon FETs," in *IEEE International Electron Devices Meeting*, 2014.

[3] Yadav *et al.*, "Spatially resolved steady-state negative capacitance," *Nature*, 2019.

[4] *International Roadmap For Devices and Systems*, 2018.

[5] E. Beyne, "The 3-D Interconnect Technology Landscape," *IEEE Design and Test*, 2016.

[6] Y. H. Chen *et al.*, "Ultra High Density SoIC with Sub-micron Bond Pitch," in *2020 IEEE 70th Electronic Components and Technology Conference (ECTC)*, 2020.

[7] L. Brunet *et al.*, "Record Performance of 500°C Low-Temperature nMOSFETs for 3D Sequential Integration using a Smart CutTM Layer Transfer Module," in *2021 Symposium on VLSI Technology*, 2021.

[8] J. Lu, H. Zhuang, *et al.*, "EPlace-3D: Electrostatics Based Placement for 3D-ICs," in *Proceedings of the International Symposium on Physical Design*, 2016.

[9] M. Hsu, V. Balabanov, and Y. Chang, "TSV-Aware Analytical Placement for 3-D IC Designs Based on a Novel Weighted-Average Wirelength Model," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2013.

[10] G. Luo, Y. Shi, and J. Cong, "An Analytical Placement Framework for 3-D ICs and Its Extension on Thermal Awareness," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2013.

[11] B. Goplen and S. Sapatnekar, "Efficient thermal placement of standard cells in 3D ICs using a force directed approach," in *International Conference on Computer Aided Design*, 2003.

[12] B. W. Ku, K. Chang, and S. K. Lim, "Compact-2D: A Physical Design Methodology to Build Two-Tier Gate-level 3D ICs," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2019.

[13] S. Pentapati *et al.*, "Pin-3D: A Physical Synthesis and Post-Layout Optimization Flow for Heterogeneous Monolithic 3D ICs," in *Proceedings of the International Conference on Computer-Aided Design*, 2020.

[14] L. Bamberg *et al.*, "Macro-3D: A Physical Design Methodology for Face-to-Face-Stacked Heterogeneous 3D ICs," in *Design, Automation Test in Europe Conference Exhibition (DATE)*, 2020.

[15] S. Panth, K. Samadi, Y. Du, and S. K. Lim, "Placement-Driven Partitioning for Congestion Mitigation in Monolithic 3D IC Designs," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2015.

[16] Y.-C. Lu *et al.*, "TP-GNN: a graph neural network framework for tier partitioning in monolithic 3D ICs," in *2020 57th ACM/IEEE Design Automation Conference (DAC)*, 2020.

[17] S. Panth, K. Samadi, Y. Du, and S. K. Lim, "Shrunk-2-D: A Physical Design Methodology to Build Commercial-Quality Monolithic 3-D ICs," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2017.

[18] K. Chang *et al.*, "Cascade2D: A design-aware partitioning approach to monolithic 3D IC with 2D commercial tools," *International Conference on Computer-Aided Design*, pp. 1–8, 2016.

[19] K. Chang, S. Pentapati, D. E. Shim, and S. K. Lim, "Road to High-Performance 3D ICs: Performance Optimization Methodologies for Monolithic 3D ICs," in *Proceedings of the International Symposium on Low Power Electronics and Design*, 2018.

[20] X. Xu, M. Bhargava, S. Moore, S. Sinha, and B. Cline, "Enhanced 3D Implementation of an Arm® Cortex®-A Microprocessor," in *2019 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, 2019.

[21] *Opencores Bemchmarks*, https://opencores.org/projects.

[22] M. M. Ozdal, C. Amin, A. Ayupov, S. Burns, G. Wilke, and C. Zhuo, "The ISPD -2012 Discrete Cell Sizing Contest and Benchmark Suite," in *Proc. ACM International Symposium on Physical Design*, 2012.

[23] ——, "An Improved Benchmark Suite for the ISPD-2013 Discrete Cell Sizing Contest," in *Proc. ACM International Symposium on Physical Design*, 2013.

[24] B. W. Ku, K. Chang, and S. K. Lim, "Compact-2D: A Physical Design Methodology to Build Commercial-Quality Face-to-Face-Bonded 3D ICs," in *Proceedings of the 2018 International Symposium on Physical Design*, 2018.

[25] H. Eisenmann, "Force-Directed Placement of VLSI Circuits," in *ISPD*, 2015.

[26] R. Jonker and A. Volgenant, "A shortest augmenting path algorithm for dense and sparse linear assignment problems," *Computing*, vol. 38, pp. 325–340, 2005.

[27] D. Z. Pan, B. Halpin, and H. Ren, "Timing-Driven Placement," in *Handbook of Algorithms for Physical Design Automation*, 2008.

[28] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian Optimization of Machine Learning Algorithms," in *NIPS*, 2012.

[29] F. Nogueira, *Bayesian Optimization: Open source constrained global optimization tool for Python*, https://github.com/fmfn/BayesianOptimization, 2014.

[30] J. Balkind, M. McKeown, *et al.*, "OpenPiton: An Open Source Manycore Research Framework," in *Proc. International Conference on Architectural Support for Programming Languages and Operating Systems*, ser. ASPLOS, 2016.

[31] D. H. Woo, N. H. Seong, D. L. Lewis, and H. S. Lee, "An optimized 3D-stacked memory architecture by exploiting excessive, high-density TSV bandwidth," in *International Symposium on High-Performance Computer Architecture*, 2010.

[32] S. K. Lim, "3D-MAPS: 3D massively parallel processor with stacked memory," in *Design for High Performance, Low Power, and Reliable 3D Integrated Circuits*, Springer, 2013.

[33] AMD, *High Bandwidth Memory*, https://www.amd.com/en/technologies/hbm.

[34] S. A. Panth, K. Samadi, Y. Du, and S. K. Lim, "Shrunk-2D: A Physical Design Methodology to Build Commercial-Quality Monolithic 3D ICs," *IEEE Trans. on Computer-Aided Design of Int. Circuits and Systems*, 2017.

[35] Ku, Bon Woong and Chang, Kyungwook and Lim, Sung Kyu, "Compact-2D: A Physical Design Methodology to Build Commercial-Quality Face-to-Face-Bonded 3D ICs," in *Proc. Int. Symp. on Physical Design*, 2018.

[36] K. Chang *et al.*, "Cascade2D: A design-aware partitioning approach to monolithic 3D IC with 2D commercial tools," in *Proc. IEEE Int. Conf. on Computer-Aided Design*, 2016.

[37] W. Gomes, S. Khushu, *et al.*, "8.1 Lakefield and Mobility Compute: A 3D Stacked 10nm and 22FFL Hybrid Processor System in 12×12mm2, 1mm Package-on-Package," in *IEEE International Solid- State Circuits Conference - (ISSCC)*, 2020.

[38] K. Flamm, "Measuring Moore's Law: Evidence from Price, Cost, and Quality Indexes," National Bureau of Economic Research, Tech. Rep., Apr. 2018.

[39] S. Khan and A. Mann, *AI Chips: What They Are and Why They Matter*, cset.georgetown.edu/research/ai-chips-what-they-are-and-why-they-matter/, Apr. 2020.

[40] Ku, Bon Woong, Debacker, Peter, *et al.*, "How much cost reduction justifies the adoption of monolithic 3d ics at 7nm node?" In *Proceedings of the International Conference on Computer-Aided Design*, 2016.

[41] P. Debacker *et al.*, "Low track height standard-cells enable high-placement density and low-BEOL cost (Conference Presentation)," in *Design-Process-Technology Co-optimization for Manufacturability XI*, 2017.

[42] S. K. Samal, D. Nayak, *et al.*, "Tier partitioning strategy to mitigate BEOL degradation and cost issues in monolithic 3D ICs," in *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2016.

PUBLICATIONS

The material in this dissertation is based on the following publications made by me with the help of my co-authors:

[1] S. **Pentapati** and S. K. Lim, "Heterogeneous Monolithic 3D ICs: EDA Solutions, and Power, Performance, Cost Tradeoffs," in *2021 58th ACM/IEEE Design Automation Conference (DAC)*, 2021, pp. 925–930.

[2] S. **Pentapati**, B. W. Ku, and S. Lim, "Machine Learning Integrated Pseudo-3D Flow for Monolithic 3D ICs," *IEEE Journal on Exploratory Solid-State Computational Devices and Circuits*, vol. 7, no. 1, pp. 35–42, 2021.

[3] S. **Pentapati**, K. Chang, V. Gerousis, R. Sengupta, and S. K. Lim, "Pin-3D: A Physical Synthesis and Post-Layout Optimization Flow for Heterogeneous Monolithic 3D ICs," in *Proceedings of the 39th International Conference on Computer-Aided Design*, ser. ICCAD '20, Association for Computing Machinery, 2020, ISBN: 9781450380263.

[4] S. **Pentapati**, B. W. Ku, and S. K. Lim, "ML-Based Wire RC Prediction in Monolithic 3D ICs with an Application to Full-Chip Optimization," in *Proceedings of the 2021 International Symposium on Physical Design*, ser. ISPD '21, Association for Computing Machinery, 2021, pp. 75–82, ISBN: 9781450383004.

[5] S. **Pentapati**, L. Zhu, L. Bamberg, D. E. Shim, A. García-Ortiz, and S. K. Lim, "A Logic-on-Memory Processor-System Design With Monolithic 3-D Technology," *IEEE Micro*, vol. 39, no. 6, pp. 38–45, 2019.

[6] S. **Pentapati**, D. E. Shim, and S. K. Lim, "Logic Monolithic 3D ICs: PPA Benefits and EDA Tools Necessary," in *Proceedings of the 2019 on Great Lakes Symposium on VLSI*, ser. GLSVLSI '19, Association for Computing Machinery, 2019, pp. 445–450.

In addition, here are some more works where I have been a contributing author and are not used for this dissertation:

[1] L. Zhu *et al.*, "High-Performance Logic-on-Memory Monolithic 3-D IC Designs for Arm Cortex-A Processors," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 29, no. 6, pp. 1152–1163, 2021.

[2] M. Hoffmann *et al.*, *Antiferroelectric negative capacitance from a structural phase transition in zirconia*, 2021. arXiv: 2104.10811 [cond-mat.mtrl-sci].

[3] Y.-C. Lu, S. **Pentapati**, L. Zhu, G. Murali, K. Samadi, and S. K. Lim, "A Machine Learning Powered Tier Partitioning Methodology for Monolithic 3D ICs," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp. 1–1, 2021.

[4] Y.-C. Lu, S. **Pentapati**, and S. K. Lim, "The Law of Attraction: Affinity-Aware Placement Optimization Using Graph Neural Networks," in *Proceedings of the 2021 International Symposium on Physical Design*, ser. ISPD '21, Association for Computing Machinery, 2021, pp. 7–14, ISBN: 9781450383004.

[5] P. Vanna-Iampikul, C. Shao, Y.-C. Lu, S. **Pentapati**, and S. K. Lim, "Snap-3D: A Constrained Placement-Driven Physical Design Methodology for Face-to-Face-Bonded 3D ICs," in *Proceedings of the 2021 International Symposium on Physical Design*, ser. ISPD '21, Association for Computing Machinery, 2021, pp. 39–46, ISBN: 9781450383004.

[6] Y.-C. Lu, S. Nath, S. **Pentapati**, and S. K. Lim, "A Fast Learning-Driven Signoff Power Optimization Framework," in *2020 IEEE/ACM International Conference On Computer Aided Design (ICCAD)*, 2020, pp. 1–9.

[7] V. Huang, D. E. Shim, J. Kim, S. **Pentapati**, S. K. Lim, and A. Naeemi, "Modeling and Benchmarking Back End Of The Line Technologies on Circuit Designs at Advanced Nodes," in *2020 IEEE International Interconnect Technology Conference (IITC)*, 2020, pp. 37–39.

[8] Y.-C. Lu, S. **Pentapati**, L. Zhu, K. Samadi, and S. K. Lim, "TP-GNN: A Graph Neural Network Framework for Tier Partitioning in Monolithic 3D ICs," in *2020 57th ACM/IEEE Design Automation Conference (DAC)*, 2020, pp. 1–6.

[9] L. Bamberg, A. García-Ortiz, L. Zhu, S. **Pentapati**, D. E. Shim, and S. Kyu Lim, "Macro-3D: A Physical Design Methodology for Face-to-Face-Stacked Heterogeneous 3D ICs," in *2020 Design, Automation Test in Europe Conference Exhibition (DATE)*, 2020, pp. 37–42.

[10] Y.-C. Lu, S. **Pentapati**, and S. K. Lim, "VLSI Placement Optimization using Graph Neural Networks," 2020.

[11] A. Agnesina, S. **Pentapati**, and S. K. Lim, "A General Framework For VLSI Tool Parameter Optimization with Deep Reinforcement Learning," 2020.

[12] S. **Pentapati**, R. Perumal, S. Khandelwal, A. I. Khan, and S. K. Lim, "Optimal Ferroelectric Parameters for Negative Capacitance Field-Effect Transistors Based on Full-Chip Implementations—Part II: Scaling of the Supply Voltage," *IEEE Transactions on Electron Devices*, vol. 67, no. 1, pp. 371–376, 2020.

[13] S. **Pentapati**, R. Perumal, S. Khandelwal, M. Hoffmann, S. K. Lim, and A. I. Khan, "Cross-Domain Optimization of Ferroelectric Parameters for Negative Capacitance Transistors—Part I: Constant Supply Voltage," *IEEE Transactions on Electron Devices*, vol. 67, no. 1, pp. 365–370, 2020.

[14] D. E. Shim, S. **Pentapati**, J. Lee, Y. S. Yu, and S. Kyu Lim, "Tier Partitioning and Flip-flop Relocation Methods for Clock Trees in Monolithic 3D ICs," in *2019 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, 2019, pp. 1–6.

[15] K. Chang, S. **Pentapati**, D. E. Shim, and S. K. Lim, "Road to High-Performance 3D ICs: Performance Optimization Methodologies for Monolithic 3D ICs," in *Proceedings of the International Symposium on Low Power Electronics and Design*, ser. ISLPED '18, Association for Computing Machinery, 2018.

VITA

Sai Surya Kiran Pentapati was born in Visakhapatnam, India in 1996. He completed his undergraduate at Indian Institute of Technology, Kharagpur with a Bachelors of Technology in Electronics and Electrical Communication Engineering in 2017. He moved to the United States for his graduate studies at Georgia Institute of Technology, where he received his Masters in Electrical and Computer Engineering in the May of 2020, and is continuing as a Ph.D. candidate.

He has been working as a Research Assistant in the Georgia Tech Computer Aided Design (GTCAD) lab since 2017 under the advisement of Dr. Sung Kyu Lim. His research interests include developing physical design solutions for 3D ICs, emerging device modelling and Design Technology Co-optimization (DTCO), and enhancing Electronic Design Automation (EDA) tools with Machine Learning and other heuristics.